Frontiers in Artificial Intelligence and Applications

COMPUTATIONAL MODELS OF ARGUMENT

Proceedings of COMMA 2006

Edited by
Paul E. Dunne
Trevor J.M. Bench-Capon

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.

Volume 144

Recently published in this series

Vol. 143. P. Ghodous et al. (Eds.), Leading the Web in Concurrent Engineering – Next Generation Concurrent Engineering
Vol. 142. L. Penserini et al. (Eds.), STAIRS 2006 – Proceedings of the Third Starting AI Researchers' Symposium
Vol. 141. G. Brewka et al. (Eds.), ECAI 2006 – 17th European Conference on Artificial Intelligence
Vol. 140. E. Tyugu and T. Yamaguchi (Eds.), Knowledge-Based Software Engineering – Proceedings of the Seventh Joint Conference on Knowledge-Based Software Engineering
Vol. 139. A. Bundy and S. Wilson (Eds.), Rob Milne: A Tribute to a Pioneering AI Scientist, Entrepreneur and Mountaineer
Vol. 138. Y. Li et al. (Eds.), Advances in Intelligent IT – Active Media Technology 2006
Vol. 137. P. Hassanaly et al. (Eds.), Cooperative Systems Design – Seamless Integration of Artifacts and Conversations – Enhanced Concepts of Infrastructure for Communication
Vol. 136. Y. Kiyoki et al. (Eds.), Information Modelling and Knowledge Bases XVII
Vol. 135. H. Czap et al. (Eds.), Self-Organization and Autonomic Informatics (I)
Vol. 134. M.-F. Moens and P. Spyns (Eds.), Legal Knowledge and Information Systems – JURIX 2005: The Eighteenth Annual Conference
Vol. 133. C.-K. Looi et al. (Eds.), Towards Sustainable and Scalable Educational Innovations Informed by the Learning Sciences – Sharing Good Practices of Research, Experimentation and Innovation

ISSN 0922-6389

Computational Models of Argument

Proceedings of COMMA 2006

Edited by

Paul E. Dunne

Department of Computer Science, The University of Liverpool, UK

and

Trevor J.M. Bench-Capon

Department of Computer Science, The University of Liverpool, UK

IOS Press

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

ISBN 1-58603-652-1
Library of Congress Control Number: 2006930652

Publisher
IOS Press
Nieuwe Hemweg 6B
1013 BG Amsterdam
Netherlands
fax: +31 20 687 0019
e-mail: 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

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

PRINTED IN THE NETHERLANDS

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Preface

The papers in this volume formed the programme of the *1st International Conference on Computational Models of Argument (COMMA)*, which was hosted by the Dept. of Computer Science of The University of Liverpool from Sept. 11th–12th, 2006. This conference originated from the ASPIC project[1] from which significant support has been received. The organisers are happy to take this opportunity to acknowledge the contribution of ASPIC towards arranging this inaugural meeting of COMMA.

The field of argumentation, once the preserve of linguistic and philosophical investigation, is now rightly seen as providing a core approach of great significance to many aspects of Artificial Intelligence. A central challenge for A.I. researchers, however, concerns how best to develop the long established body of work from more speculative disciplines, such as philosophical treatments of argument and reasoning, into effective and practical computational paradigms: one aim of COMMA, well reflected in the papers contributing to this volume, has been to engage with the issues raised by this challenge. Thus the topics addressed range from formal questions involving properties of algorithms and semantic models, through proposals for robust implementation of argumentation based systems, to reports of applications built on argumentation technology.

It is, of course, the case that the success of any conference depends not only on the quality of the research presented but also on the contributions of many other individuals. The organisers are grateful to the members of the Programme Committee and additional reviewers whose detailed reports and subsequent discussions considerably eased the diffult task of forming the final selection of papers. It is also a pleasure to thank Ken Chan, Phil Jimmieson and Dave Shield for their work in providing technical support throughout the period from the initial announcement to the conference itself, together with Thelma Williams who kept track of assorted budget and financial matters. In addition the editors appreciate the efforts of Carry Koolbergen, Maarten Fröhlich, and Paul Weij of IOS Press in promptly and efficiently handling the many questions that arose during the preparation of this volume. Finally, and by no means least, we thank Catherine Atherton who maintained the conference web pages as well as dealing with general queries.

June 2006

Paul E. Dunne
Trevor Bench-Capon
Michael Wooldridge

[1] European Commission Project, IST-FP6-002307

Organisation

Organising Committee

General Chair: Michael Wooldridge, Univ. of Liverpool
Conference Chair: Trevor J.M. Bench-Capon, Univ. of Liverpool
Programme Chair: Paul E. Dunne, Univ. of Liverpool

Programme Committee

Leila Amgoud, IRIT
Kevin Ashley, Pittsburgh
Katie Atkinson, Liverpool
Pietro Baroni, Brescia
Trevor Bench-Capon, Liverpool
Ivan Bratko, Ljubljana
Gerhard Brewka, Leipzig
Simon Buckingham Shum, Open Univ.
Martin Caminada, Utrecht
Claudette Cayrol, IRIT
Carlos Chesnevar, Lleida
Ulises Cortes, Catalonia
Frank Dignum, Utrecht
Yannis Dimopoulos, Cyprus
Sylvie Doutre, Toulouse
Phan Minh Dung,
Asian Institute of Technology
John Fox, CRUK
Massimiliano Giacomin, Brescia
Tom Gordon, Fraunhofer FOKUS
Floriana Grasso, Liverpool
David Hitchcock, McMaster
John Horty, Maryland
Anthony Hunter, UCL
Antonis Kakas, Cyprus
Jürg Kohlas, Fribourg
Paul Krause, Surrey
Ronald Loui, Washington Univ. in St. Louis
Peter McBurney, Liverpool
Nicolas Maudet, Paris-Dauphine
Sanjay Modgil, CRUK
Martin Mozina, Ljubljana
Tim Norman, Aberdeen
Simon Parsons, CUNY
Henri Prade, IRIT
Henry Prakken, Utrecht & Groningen
Iyad Rahwan, British Univ. in Dubai
Chris Reed, Dundee
Ken Satoh, NII
Carles Sierra, IIIA-CSIC
Guillermo R. Simari, Univ. Nacional del Sur
Francesca Toni, Imperial College
Paolo Torroni, Bologna
Gerard Vreeswijk, Utrecht
Doug Walton, Winnipeg

Additional Reviewers

Katsumi Nitta, Tokyo Institute of Technology
Hajime Sawamura, Niigata
Jure Zabkar, Ljubljana

Contents

This page intentionally left blank

Argument in Agent Systems

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Towards Representing and Querying Arguments on the Semantic Web

Iyad Rahwan [a,b,*] and P.V. Sakeer [a]
[a] *British University in Dubai, UAE*
[b] *(Fellow) University of Edinburgh, UK*

Abstract. This paper demonstrates the potential of the Semantic Web as a platform for representing, navigating and processing arguments on a global scale. We use the RDF Schema (RDFS) ontology language to specify the ontology of the recently proposed Argument Interchange Format (AIF) and an extension thereof to Toulmin's argument scheme. We build a prototype Web-based system for demonstrating basic querying for argument structures expressed in the Resource Description Framework (RDF). An RDF repository is created using the Sesame open source RDF server, and can be accessed via a user interface that implements various user-defined queries.

Keywords. Argumentation, Semantic Web, Agents, RDF

1. Introduction

Argumentation is a verbal and social activity of reason aimed at increasing (or decreasing) the acceptability of a controversial standpoint for the listener or reader, by putting forward a constellation of propositions intended to justify (or refute) the standpoint before a rational judge [1, page 5]. In a computational or multi-agent system, the *rational judge* could correspond to a particular choice of rules or algorithm for computing the acceptable arguments for deciding the agent that wins the argument. Moreover, the *standpoint* may not necessarily be propositional, and should be taken in the broadest sense (e.g. it may refer to a decision or a value judgement). Finally, the term *controversial* should also be taken in the broad sense to mean "subject to potential conflict."

The theory of argumentation is a rich, interdisciplinary area of research lying across philosophy, communication studies, linguistics, and psychology. Its techniques and results have found a wide range of applications in both theoretical and practical branches of artificial intelligence and computer science [2,3,4].

While argumentation mark-up languages such as those of Araucaria [5], Compendium and ASCE (see [6] for example) already exist, they are primarily a means to enable users to structure arguments through diagrammatic linkage of natural language sentences. Moreover, these mark-up languages do not have rich formal semantics, and are therefore not designed to enable sophisticated automated processing of argumenta-

*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.

tive statements. Such semantics may help improve applications of electronic deliberative democracy [7,8,9,10] by enabling citizens to annotate, query and navigate arguments and elements of arguments. Rich formal semantics may also improve capabilities for argumentation among autonomous software agents [11,12,13,14] by enabling the exchange arguments in open multi-agent systems using a standardised format.

In response to the above, an effort towards a standard Argument Interchange Format (AIF) has recently commenced [15]. The aim was to consolidate the work that has already been done in argumentation mark-up languages and multi-agent systems frameworks. It was hoped that this effort will provide a convergence point for theoretical and practical work in this area, and in particular facilitate: (i) argument interchange between agents within a particular multi-agent framework; (ii) argument interchange between agents across separate multi-agent frameworks; (iii) inspection/manipulation of agent arguments through argument visualisation tools; and (iv) interchange between argumentation visualisation tools.

This paper presents a first step towards representing arguments on the World Wide Web using open, rich, and formal semantic annotation. We present building blocks for developing Web-based systems for navigating and querying argument structures expressed in the Resource Description Framework (RDF). The RDF representation of arguments conforms to an ontology of arguments, which is based on the AIF specification and expressed in the RDF Schema language. By expressing the AIF ontology in a standard format (namely RDF), it becomes possible to use a variety of Semantic Web tools (e.g. RDF query engines) to access and process arguments. This approach opens up many possibilities for automatic argument processing on a global scale.

The rest of the paper is organised as follows. In the next Section, we summarise the current state of the Argument Interchange Format specification. In Section 3, we describe how RDF and RDF Schema can be used to specify argument structures. We discuss some related work in Section 4 and conclude the paper in Section 5.

2. The Argument Interchange Format Ontology

In this section, we provide a brief overview of the current state of the Argument Interchange Format. We will use the AIF specification as of April 2006 [15]. The AIF is a core ontology of argument-related concepts. This core ontology is specified in such a way that it can be extended to capture a variety of argumentation formalisms and schemes. To maintain generality, the AIF core ontology assumes that argument entities can be represented as nodes in a directed graph (di-graph). This di-graph is informally called an *argument network* (AN).

2.1. Nodes

There are two kinds of nodes in the AIF, namely, *information nodes* (I-nodes) and scheme application nodes or *scheme nodes* (S-nodes) for short. Roughly speaking, I-Nodes contain content that represent declarative aspects of the the domain of discourse, such as claims, data, evidence, propositions etc. On the other hand, S-nodes are applications of *schemes*. Such schemes may be considered as domain-independent patterns of reasoning, including but not limited to rules of inference in deductive logics. The present on-

	to *I-node*	to *RA-node*	to *PA-node*
from *I-node*		data/information used in applying an inference	data/information used in applying a preference
from *RA-node*	inferring a conclusion in the form of a claim	inferring a conclusion in the form of a scheme application	inferring a conclusion in the form of a preference application
from *PA-node*	applying preferences among information (goals, beliefs, ..)	applying preferences among inference applications	meta-preferences: applying preferences among preference applications

Table 1. Informal semantics of support.

tology deals with two different types of schemes, namely *inference schemes* and *attack schemes*. Potentially other scheme types could exist, such as evaluation schemes and scenario schemes, which will not be addressed here.

The ontology specifies two types of S-Nodes. If a scheme application node is an application of an inference scheme it is called a *rule of inference application node* (RA-node). If a scheme application node is an application of a preference scheme it is called a *preference application node* (PA-node). Informally, RA-nodes can be seen as applications of rules of inference while PA-nodes can be seen as applications of (possibly abstract) criteria of preference among evaluated nodes.

2.2. Node Attributes

Nodes may possess different attributes that represent things like title, text, creator, type (e.g. decision, action, goal, belief), creation date, evaluation, strength, acceptability, and polarity (e.g. with values of either "pro" or "con"). These attributes may vary and are not part of the core ontology. Attributes may be intrinsic (e.g. "evidence"), or may be derived from other attributes (e.g. "acceptability" of a claim may be based on computing the "strength" of supporting and attacking arguments).

2.3. Edges

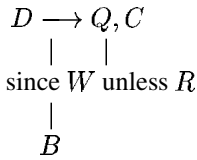
According to the AIF core ontology, edges in an argument network can represent all sorts of (directed) relationships between nodes, but do not necessarily have to be labelled with semantic pointers. A node A is said to *support* node B if and only if an edge runs from A to B.[1]

There are two types of edges, namely *scheme edges* and *data edges*. Scheme edges emanate from S-nodes and are meant to support conclusions. These conclusions may either be I-nodes or S-nodes. Data edges emanate from I-nodes, necessarily end in S-nodes, and are meant to supply data, or information, to scheme applications. In this way, one may speak of I-to-S edges (e.g. representing "information," or "data" supplied to a scheme), S-to-I edges (e.g. representing a "conclusion" supplied by a scheme) and S-to-S edges (e.g. representing one scheme's attack against another scheme).

[1]Note that this is a rather lose use of the word "support" and is different from the notion of "support between arguments" in which one argument supports the acceptability of another argument.

2.4. Extending the Ontology: Toulmin's Argument Scheme

Philosopher Stephen Toulmin presented a general argument scheme for analysing argumentation. Toulmin's scheme, which has recently become influential in the computational modelling of argumentation, consists of a number of elements which are often depicted graphically as follows:



The various elements are interpreted as follows:

Claim (C): This is the assertion that the argument backs.

Data (D): The evidence (e.g. fact, an example, statistics) that supports the claim.

Warrant (W): This is what holds the argument together, linking the evidence to the claim.

Backing (B): The backing supports the warrant; it acts as an evidence for the warrant.

Rebuttal (R): A rebuttal is an argument that might be made against the claim, and is explicitly acknowledged in the argument.

Qualifier (Q): This element qualifies the conditions under which the argument holds.

An example of an argument expressed according to Toulmin's scheme can be as follows. The war in Irat (a fictional country) is justified (C) because there are weapons of mass destruction (WMDs) in Irat (D) and all countries with weapons of mass destructions must be attacked (W). Countries with WMDs must be attacked because they pose danger to others (B). This argument for war on Irat can be rebutted if the public do not believe the CIA intelligence reports about Irat possessing WMDs (R). Finally, this argument only holds if attacking Irat is less damaging than the potential damage posed by its WMDs (Q).

Toulmin's argument scheme may be represented as an extension of the AIF core ontology. In particular, the concepts of *claim*, *data*, *backing*, *qualifier* and *rebuttal* can all be expressed as sub-classes of I-Node. The concept of *warrant*, on the other hand, is an extension of RA-Nodes. This is because the former concepts all represent passive propositional knowledge, while the warrant is what holds the scheme together. In addition, since I-Nodes cannot be linked directly to one another, we introduce two new extensions of RA-Nodes. The new *qualifier-application* nodes link qualifier nodes to claim nodes, while *rebuttal-application* nodes link rebuttal nodes to claim nodes.

3. Arguments in RDF/RDFS

In this section, we describe the specification of the AIF ontology, and its extension to Toulmin's argument scheme, in RDF Schema.

3.1. Background: XML, RDF and RDFS

The Extensible Mark-up Language (XML) is a W3C standard language for describing document structures by *tagging* parts of documents. XML documents provide means for nesting tagged *elements*, resulting in a directed tree-based structure. The XML Document Type Definition (DTD) and XML Schema languages can be used to describe different *types* of XML documents.

The Resource Description Framework (RDF)[2] is a general framework for describing Internet resources. RDF defines a resource as any object that is uniquely identifiable by an Uniform Resource Identifier (URI). Properties (or attributes) of resources are defined using an object-attribute-value triple, called a *statement*.[3] RDF statements can be represented as 3-tuples, as directed graphs, or using a standard XML-based syntax. The different notations are shown in Figure 1. Attributes are sometimes referred to as *properties* or *predicates*.

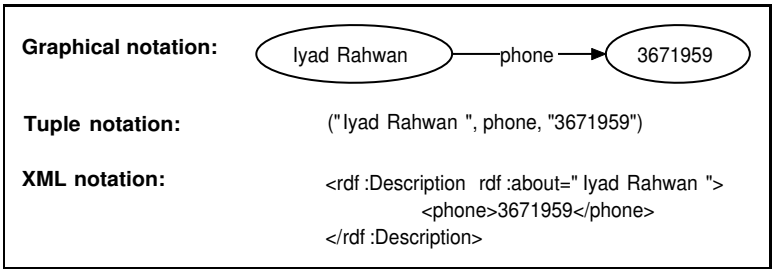


Figure 1. Different notations for RDF statements

Unlike XML, which describes document models in directed-tree-based nesting of elements, RDF's model is based on arbitrary graphs. This structure is better suited for creating conceptual domain models. RDF provides a more concise way of describing rich semantic information about resources. As a result, more efficient representation, querying and processing of domain models become possible.

RDF Schema (RDFS)[4] is an (ontology) language for describing vocabularies in RDF using terms described in the RDF Schema specification. RDFS provides mechanisms for describing characteristics of resources through, for example, domains and ranges of properties, classes of resources, or class taxonomies. RDFS (vocabulary-describing) statements are themselves described using RDF triples.

3.2. AIF and Toulmin's Scheme in RDF Schema

We have first specified the AIF core ontology in RDFS using the Protégé ontology development environment.[5] The main class `Node` was specialised to two types of nodes: `I-Node` and `S-Node`. The `S-Node` class was further specialised to two more classes: `PA-Node` and `RA-Node`. For example, the following RDFS code declares the class `PA-Node` and states that it is a sub-class of the class `S-Node`.

[2] `http://www.w3.org/RDF/`
[3] Sometimes, an *attribute* is referred to as a *property* or a *slot*.
[4] `http://www.w3.org/TR/rdf-schema/`
[5] `http://protege.stanford.edu/`

```
<rdfs:Class rdf:about="&kb;PA_Node" rdfs:label="PA_Node">
  <rdfs:subClassOf rdf:resource="&kb;S-Node"/>
</rdfs:Class>
```

Next, the following elements from Toulmin's scheme were introduced as I-Nodes: claim, data, backing, rebuttal, and qualifier. All these elements represent passive declarative knowledge. Toulmin's warrant was expressed as an RA-Node, since it holds part of the argument together, namely the data nodes and the claim. Similarly, we introduced two other types of RA-Nodes: `Rebuttal-Application` nodes are used to link rebuttal nodes to claims, while `Qualifier-Application` nodes are used to link qualifier nodes to claims. The resulting ontology is represented in Figure 2.

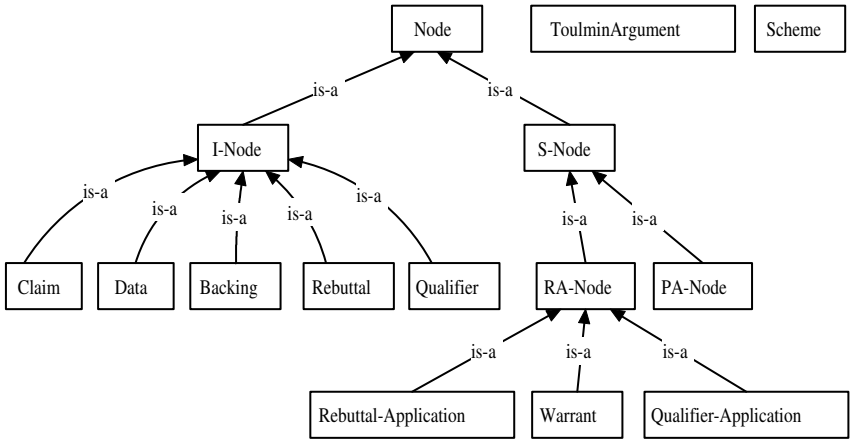


Figure 2. Toulmin argument class hierarchy as an extension of AIF ontology

Note that the concept `ToulminArgument` is a standalone concept. Instances of this concept will represent complete arguments expressed in Toulmin's scheme. Such instances must therefore refer to instances of the various elements of the scheme. The ontology imposes a number of restrictions on these elements and their interrelationships. In particular, each Toulmin argument must contain exactly one claim, exactly one warrant, exactly one qualifier, at least one backing, and at least one datum. As an example, the following RDFS code declares the property `claim` which links instances of `ToulminArgument` to instances of type `Claim`, and states that each `ToulminArgument` must be linked to exactly one `Claim`:

```
<rdf:Property rdf:about="&kb;claim"
    a:maxCardinality="1"
    a:minCardinality="1"
    rdfs:label="claim">
  <rdfs:domain rdf:resource="&kb;ToulminArgument"/>
  <rdfs:range rdf:resource="&kb;Claim"/>
</rdf:Property>
```

In our ontology, we defined various predicates to capture every type of edge, such as those that emanate from backing nodes to warrant nodes, those from warrants to claims, and so on.

Note that according to our ontology, a single claim node can belong to multiple instances of Toulmin arguments, denoting multiple reasons for believing the claim. Similarly, a single data node could contribute to multiple unrelated claims. The RDF graph model enables such flexibility.

With the ontology in place, it is now possible to create instances of the Toulmin argument scheme in RDF. Figure 3 shows the argument mentioned above for justifying the war on Irat. Each box represents an RDF resource, which is an instance of the relevant node type, while edges represent RDF predicates. In addition, all these resources are linked to an instance (named "`IratWar`") of the class `ToulminArgument`, but we omit these links for clarity purposes. In the Figure, we distinguished S-Nodes by dotted boxes although they are not treated differently from the point of view of RDF processing tools.

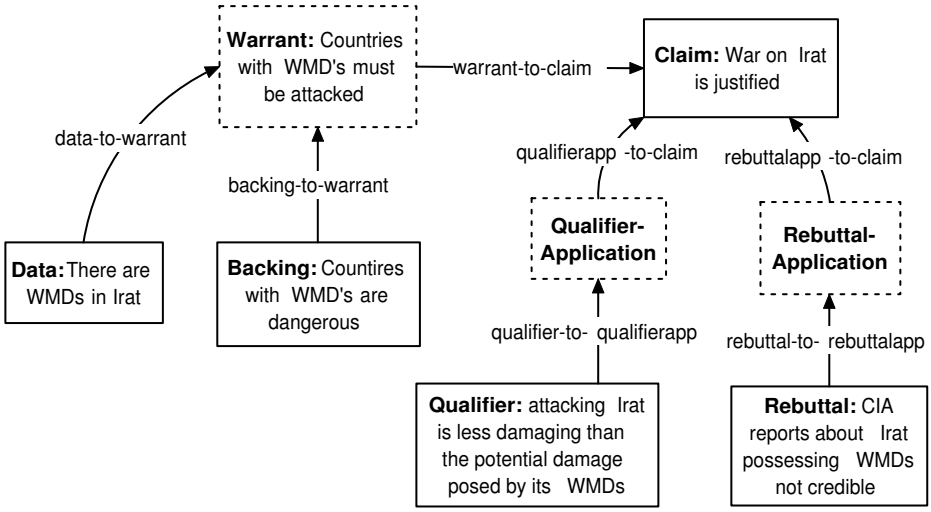


Figure 3. RDF graph for elements of Toulmin argument instance "`IratWar`"

Note that in practice, each of these elements of the argument instance may reside on a different location on the Web. For example, the backing text can be replaced by a reference to a full on-line newspaper article explaining the different dangers countries with WMDs pose. We believe that this feature of RDF could be instrumental for building a layer of argument structures on top of existing Web content.

Finally, we note that the above description is not the only way of representing the Toulmin scheme diagrammatically. Indeed, a Toulmin argument can be represented in more ways than one while, more or less, preserving its semantics. While such representations are outside the scope of this paper, we refer the interested reader to the extensive analysis by Reed and Rowe [16].

3.3. Deploying an RDF Repository of Arguments

Our ultimate aim is to provide an infrastructure for publishing semantically annotated arguments on the *Semantic Web* using a language that is semantically rich and amenable to machine processing. The choice of RDF as a representation language was motivated by its expressive power and the availability of tools for navigating and processing RDF statements.

In order to test our idea, we uploaded the argument instances on Sesame:[6] an open source RDF repository with support for RDF Schema inferencing and querying. Sesame can be deployed on top of a variety of storage systems (relational databases, in-memory, filesystems, keyword indexers, etc.), and offers a large set of tools for developers to leverage the power of RDF and RDF Schema, such as a flexible access API, which supports both local and remote access, and several query languages, such as RQL and SeRQL [17]. Sesame itself was deployed on the Apache Tomcat server, which is essentially a Java servlet container.

We have written a number of queries to demonstrate the applicability of our approach. The following query retrieves all warrants, data and backings for the different arguments in favour of the claim that "War on Irat is justified."

```
select WARRANT-TEXT, DATA-TEXT, BACKING-TEXT, CLAIM-TEXT
from {WARRANT} kb:scheme-edge-warrant-to-claim {CLAIM},
     {WARRANT} kb:text {WARRANT-TEXT},
     {DATA} kb:data-edge-data-to-warrant {WARRANT},
     {DATA} kb:text {DATA-TEXT},
     {BACKING} kb:data-edge-backing-to-warrant {WARRANT},
     {BACKING} kb:text {BACKING-TEXT},
     {CLAIM} kb:text {CLAIM-TEXT}
where
     CLAIM-TEXT like "War in Irat is justified"
using namespace    kb = http://protege.stanford.edu/kb#
```

The output of the above query returned by Sesame will be the following, showing two arguments. The first justifies war on Irat on the basis of the presence of WMDs. The second argument justifies the war on the basis of removing the country's dictator (a fictional character named "Saddad").

WARRANT-TEXT	DATA-TEXT	BACKING-TEXT	CLAIM-TEXT
Countries with WMD's must be attacked	There are WMD's in Irat	Countries with WMD's are dangerous	War on Irat is justified
Countries ruled by dictators must be attacked	Saddad is a dictator	Dictatorships pose security threats on neighbours	War on Irat is justified

Suppose that after retrieving the first argument, a user or an automated agent is interested in finding out what other claims are supported by the warrant "All Countries with WMD's must be attacked." This information can be found using the following query.

[6] http://www.openrdf.org/

```
select WARRANT-TEXT, CLAIM-TEXT
from {WARRANT} kb:scheme-edge-warrant-to-claim {CLAIM},
     {WARRANT} kb:text {WARRANT-TEXT},
     {DATA} kb:data-edge-data-to-warrant {WARRANT},
     {DATA} kb:text {DATA-TEXT},
     {BACKING} kb:data-edge-backing-to-warrant {WARRANT},
     {BACKING} kb:text {BACKING-TEXT},
     {CLAIM} kb:text {CLAIM-TEXT}
where
     WARRANT-TEXT like
                  "All Countries with WMD's must be attacked"
using namespace    kb = http://protege.stanford.edu/kb#
```

The output of this query is as follows:

WARRANT-TEXT	CLAIM-TEXT
Countries with WMD's must be attacked	War on Irat is justified
Countries with WMD's must be attacked	War on USO is justified

In this case, the same warrant used to justify the war against Irat may be used to justify war against the USO (another fictional country).

These queries demonstrate the potential of using the structure of RDF and the expressiveness of RDF query languages to navigate arguments on the Web. Query results can be retrieved via Sesame in XML for further processing. In this way, we could build a more comprehensive system for navigating argument structures through an interactive user interface that triggers such queries.

4. Related Work

A number of argument mark-up languages have been proposed. For example, the Assurance and Safety Case Environment (ASCE)[7] is a graphical and narrative authoring tool for developing and managing assurance cases, safety cases and other complex project documentation. ASCE relies on an ontology for *arguments about safety* based on *claims*, *arguments* and *evidence* [18].

Another mark-up language was developed for Compendium,[8] a semantic hypertext concept mapping tool. The Compendium argument ontology enables constructing *Issue Based Information System (IBIS)* networks, in which nodes represent *issues*, *positions* and *arguments* [19].

A third mark-up language is the argument-markup language (AML) behind the Araucaria system,[9] an XML-based language [5]. The syntax of AML is specified in a Document Type Definition (DTD) which imposes structural constraints on the form of legal AML documents. AML was primarily produced for use in the Araucaria tool. For example, the DTD could state that the definition of an argument scheme must include a name and any number of critical questions.

[7] http://www.adelard.co.uk/software/asce/

[8] http://www.compendiuminstitute.org/tools/compendium.htm

[9] http://araucaria.computing.dundee.ac.uk/

ClaiMaker and related technologies [20] provide a set of tools for individuals or distributed communities to publish and contest ideas and arguments, as is required in contested domains such as research literatures, intelligence analysis, or public debate. It provides tools for constructing argument maps, and a server on which they can then be published, navigated, filtered and visualized using the *ClaimFinder* semantic search and navigation tools [21]. This system is based on a specific ontology called the *ScholOnto* ontology [22].

The above attempts at providing argument mark-up languages share some following limitation. Each of these mark-up languages is designed for use with a specific tool, usually for the purpose of facilitating argument visualisation. They were not intended for facilitating inter-operability of arguments among a variety of tools. As a consequence, the semantics of arguments specified using these languages are tightly coupled with particular schemes to be interpreted in a specific tool and according to a specific underlying theory. For example, arguments in Compendium are interpreted in relation to a specific theory of *issue-based information systems*. In order to enable true interoperability of arguments and argument structures, we need an argument description language that can be extended in order to accommodate a variety of argumentation theories and schemes. The AIF, as captured in RDF/RDFS, has the potential to form the basis for such a language.

Another limitation of the above argument mark-up languages is that they are primarily aimed at enabling users to structure arguments through diagramatic linkage of natural language sentences [6]. Hence, these mark-up languages are not designed to process formal logical statements such as those used within multi-agent systems. For example, AML imposes structural limitations on legal arguments, but provides no semantic model. Such semantic model is needed in order to enable the automatic processing of argument structures by software agents.

5. Conclusion

In this paper, we investigated the potential of the Semantic Web as a platform for representing, navigating and processing arguments on a global scale. We used the RDF Schema (RDFS) ontology language to specify the ontology of the recently proposed Argument Interchange Format (AIF) and an extension thereof to Toulmin's argument scheme. We built a prototype Web-based system for demonstrating basic querying for argument structures expressed in the Resource Description Framework (RDF).

Our future plans include extending the AIF core ontology to other argument schemes, such as Walton's schemes for presumptive reasoning [23]. By doing so, we hope to validate the applicability of our approach and identify the limitations of RDF and RDFS for representing argument structures. A more expressive ontology language, such as OWL [24], may be needed.

Another future direction for our work is to build applications that exploit the rich semantics of arguments provided by Semantic Web ontologies. Such applications could range from sophisticated argument processing and navigation tools to support human interaction with argument content, to purely automated applications involving multiple interacting agents operating on Web-based argument structures.

Acknowledgements

The first author is grateful to *AgentLink*[10] and ASPIC [25] for financial support to attend the AgentLink Argumentation Interchange Format Technical Forum in Budapest during September 2005, and to the AIF Technical Forum Group for many stimulating discussions during and after the forum.

References

[1] F. H. van Eemeren, R. F. Grootendorst, and F. S. Henkemans. *Fundamentals of Argumentation Theory: A Handbook of Historical Backgrounds and Contemporary Applications*. Lawrence Erlbaum Associates, Hillsdale NJ, USA, 1996.

[2] H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 4, pages 219–318. Kluwer Academic Publishers, Dordrecht, Netherlands, second edition, 2002.

[3] C. I. Chesñevar, A. Maguitman, and R. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.

[4] D. Carbogim, D. Robertson, and J. Lee. Argument-based applications to knowledge engineering. *Knowledge Engineering Review*, 15(2):119–149, 2000.

[5] G. W. A. Rowe, C. Reed, and J. Katzav. Araucaria: Marking up argument. In *European Conference on Computing and Philosophy*, 2003.

[6] P. A. Kirschner, S. Buckingham Schum, and C. S. Carr, editors. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. Springer Verlag, London, 2003.

[7] K. Atkinson, T. Bench-Capon, and P. McBurney. PARMENIDES: facilitating deliberation in democracies. *Artificial Intelligence and Law – T. van Engers and A. Macintosh (editors), Special Issue on eDemocracy*, page (to appear), 2006.

[8] A. Lotov. Internet tools for supporting of lay stakeholders in the framework of the democratic paradigm of environmental decision making. *Journal of Multi-criteria decision analysis*, 12:145–162, 2003.

[9] R. Lüehrs, T. Malsch, and K. Voss. Internet, discourses and democracy. In T. Terano, T. Nishida, A. Namatame, S. Tsumoto, Y. Ohsawa, and T. Washio, editors, *New Frontiers in Artificial Intelligence*, volume 2253 of *Lecture Notes in Computer Science*, pages 67–74. Springer-Verlag, Heidelberg, Germany, 2001.

[10] C. Niculae and S. French. Bringing understanding in societal decision making: Explaining and communicating analyses. *Journal of Multi-criteria decision analysis*, 12:191–202, 2003.

[11] S. Parsons, C. Sierra, and N. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.

[12] I. Rahwan. *Interest-based Negotiation in Multi-Agent Systems*. PhD thesis, Department of Information Systems, University of Melbourne, Melbourne, Australia, 2004.

[13] I. Rahwan, S. D. Ramchurn, N. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation based negotiation. *Knowledge Engineering Review*, 18(4):343–375, 2003.

[14] F. Sadri, F. Toni, and P. Torroni. Abductive logic programming architecture for negotiating agents. In *Proceedings of the 8th European Conference on Logics in Artificial Intelligence (JELIA-2002)*, volume 2424 of *Lecture Notes in Computer Science*, pages 419–431. Springer Verlag, Berlin, Germany, 2002.

[15] S. Willmott, G. Vreeswijk, C. Chesnevar, M. South, J. McGinis, S. Modgil, I. Rahwan, C. Reed, and G. Simari. Towards an argument interchange format for multiagent systems.

[10] `http://www.agentlink.org`

In Nicolas Maudet, Simon Parsons, and Iyad Rahwan, editors, *Proceedings of the 3rd International Workshop on Argumentation in Multi-Agent Systems (ArgMAS), Hakodate, Japan*, 2006.

[16] C. Reed and G. Rowe. Translating Toulmin diagrams: Theory neutrality in argument representation. *Argumentation*, 19:267–286, 2005.

[17] P. Haase, J. Broekstra, A. Eberhart, and R. Volz. A comparison of rdf query languages. In S. A. McIlraith, D. Plexousakis, and F. van Harmelen, editors, *Proceedings of the 3rd International Semantic Web Conference, Hiroshima, Japan*, volume 3298 of *Lecture Notes in Computer Science*, pages 502–517. Springer Verlag, Heidelberg, Germany, 2004.

[18] L. Emmet and G. Cleland. Graphical notations, narratives and persuasion: a pliant systems approach to hypertext tool design. In *HYPERTEXT 2002, Proceedings of the 13th ACM Conference on Hypertext and Hypermedia, June 11-15, 2002, University of Maryland, College Park, MD, USA*, pages 55–64, New York, USA, 2002. ACM Press.

[19] J. Conklin and M. L. Begeman. gIBIS: a hypertext tool for exploratory policy discussion. *ACM transactions on office information systems*, 6(4):303–331, 1988.

[20] S. Buckingham Shum, V. Uren, G. Li, B. Sereno, and C. Mancini. Modelling naturalistic argumentation in research literatures: Representation and interaction design issues. *International Journal of Intelligent Systems, Special Issue on Computational Modelling of Naturalistic Argumentation (to appear)*, 2006.

[21] N. Benn, S. Buckingham Shum, and J. Domingue. Integrating scholarly argumentation, texts and community: Towards an ontology and services. In *Proceedings of the Fifth Workshop on Computational Models of Natural Argument (CMNA 2005)*, 2005.

[22] S. Buckingham Shum, E. Motta, and J. Domingue. ScholOnto: An ontology-based digital library server for research documents and discourse. *International Journal of Digital Libraries*, 3(3):237–248, 2000.

[23] D. N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Erlbaum, Mahwah NJ, USA, 1996.

[24] D. L. McGuinness and F. van Harmelen. Web ontology language (OWL): Overview. Technical report, W3C Working Draft, 31 March 2003.

[25] ASPIC. Argumentation service platform with integrated components, a European Commission-funded research project (no. ist-fp6-002307), 2004.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Building Agents that Plan and Argue in a Social Context

Dionysis Kalofonos [a], Nishan Karunatillake [b], Nicholas R. Jennings [b],
Timothy J. Norman [a], Chris Reed [c] and Simon Wells [c]
[a] *Department of Computing Science, University of Aberdeen*
[b] *School of Electronics and Computer Science, University of Southampton*
[c] *Division of Applied Computing, University of Dundee*

Abstract. In order for one agent to meet its goals, it will often need to influence another to act on its behalf, particularly in a society in which agents have heterogenous sets of abilities. To effect such influence, it is necessary to consider both the social context and the dialogical context in which influence is exerted, typically through utterance. Both of these facets, the social and the dialogical, are affected by, and in turn affect, the plan that the influencing agent maintains, and the plans that the influenced agents may be constructing. The i-Xchange project seeks to bring together three closely related areas of research: in distributed planning, in agent-based social reasoning, and in inter-agent argumentation, in order to solve some of the problems of exerting influence using socially-aware argument.

Keywords. Multiagent Planning, Argument Protocols, Social Reasoning, Negotiation

1. Introduction

Negotiation is a key form of interaction in multi-agent systems. It is important because conflict is endemic in such systems and because the individual agents are autonomous problem solving entities that are typically concerned with achieving their own aims and objectives. Given its importance, such negotiations come in many different shapes and forms, ranging from auctions to bilateral negotiations to argumentation. Here we focus on this latter kind of interaction because it offers perhaps the greatest degree of flexibility out of all these many different types. However, this flexibility comes at a price. Specifically, conceptualizing, designing, and building agents that are capable of argumentation-based negotiation is a major challenge. Given this fact, most work in this area is primarily directed at the theory of such agents and those implementations that do exist are somewhat primitive in nature. Moreover, much of the theoretical work in this area tends to concentrate on a specific aspect of the negotiation and fails to provide a coherent overarching framework. Against this background, we describe our work in the Information Exchange Project (*i*-Xchange) that seeks to rectify these shortcomings.

In more detail, this work seeks to integrate and pull together the following key components of an agent's activity as it relates to argumentation-based negotiation:

- The ability of an agent to devise a plan of action that takes account of the fact that the agent is situated within a multi-agent community. Thus such an agent can devise a plan that involves steps that will be performed by agents other than itself.

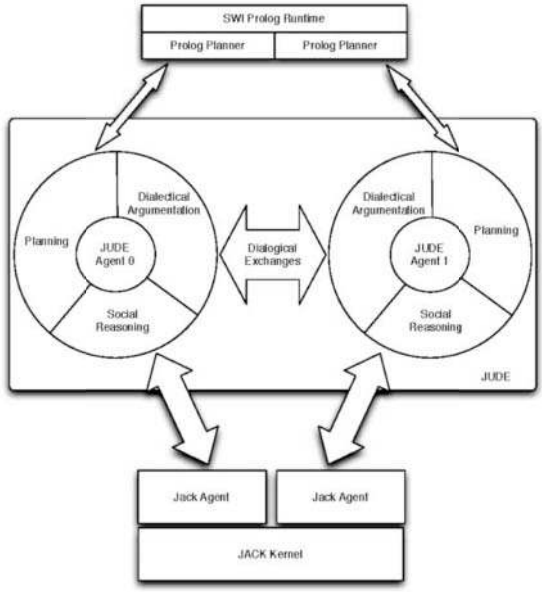


Figure 1. The structural components of the *i*-Xchange architecture

- For those actions that need to be performed by other agents, social reasoning is required to determine what agents should be chosen to perform what actions. This involves determining which agents are most suited for the task, which are likely to be available, and which are likely to be the most reliable.
- Once the appropriate agents have been identified, the agent needs to determine the most effective way of persuading these agents to perform the desired action. This dialogical goal can be achieved through a variety of means such as by offering rewards, making threats, or making use of social relationships that exist between the agents.

By bringing together these key building blocks, we are able to produce a coherent framework and software architecture for an agent that can perform a complete planning-acting cycle in which argumentation is used as the basis for all social inter-changes. As well as detailing the various components and their interfaces, we also demonstrate their operation in an e-Science scenario that has motivated much of this work.

2. The Information Exchange

The *i*-Xchange uses two multiagent system frameworks; JACK and JUDE (the Jackdaw University Development Environment. JUDE is a lightweight, flexible, industrial-strength agent platform that uses a modular approach to agent development. This enables domain specific functionality to be encapsulated into a module which can be dynamically loaded into an agent at runtime. Individual agents within the *i*-Xchange are represented by JUDE agents composed of a number of modules. A proxy module incorporating a

communications bridge allows a 1:1 relationship with JACK agents. The reason for the use and integration of multiple disparate frameworks is twofold, firstly it allows existing domain specific software to be used without reimplementation, and secondly it demonstrates that agents can be developed under different extant frameworks and integrated into a single heterogeneous MAS.

An *i*-Xchange agent is composed of three modules offering domain specific functionality. These are the planning, social-reasoning, and dialogical argumentation modules which are discussed through the remainder of this section. Figure 2 gives an overview of the components and modules that comprise the infrastructure for an *i*-Xchange MAS. Modules communicate with each other to provide aggregate behaviours. Inter-module communication is achieved by passing data objects between modules. Two such objects are the *service request* and the *proposal*. A service request is created by the Planning module to encapsulate a partial plan consisting of a set of actions and the name of an agent committed to perform the actions (initially set to $\perp$). A proposal is created by the dialogical argumentation module during a dialogue to encapsulate a service request received from another agent and any associated social issues. Figure 2 shows a complete circuit of communication for a simple enquiry dialogue between the six modules incorporated in a pair of *i*-Xchange agents.

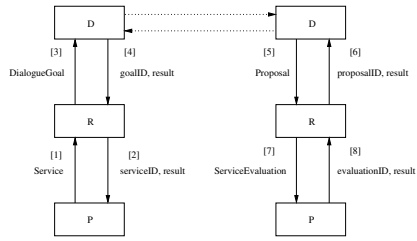


Figure 2. Basic inter-module and inter-agent communication pathways. The modules P, R, and D on the left constitute the initiating agent, iXchangeagent$_0$, and the modules P, R, and D on the right constitute the recipient agent, iXchangeagent$_1$.

2.1. Planning

The planning module makes use of the Graphplan algorithm first introduced in [1] which we have implemented in Prolog. Let us discuss the algorithm in detail and present our extensions for the extraction of services from the constructed plans and for service evaluation through the merging of services into the constructed plans.

A Planning graph is a layered graph with each layer consisting of a set of propositions and a set of actions (see figure 3). Each layer (named time-step) represents a point in time, hence a set of propositions at a time-step n represents a snapshot of the state of the world at the time-step n, while a set of actions appearing at a time-step n contains all the actions that are executable in the state of the world at time-step n.

The graph consists of two kinds of nodes namely, the proposition nodes which form the proposition sets at each time-step and the action nodes which are the instantiations

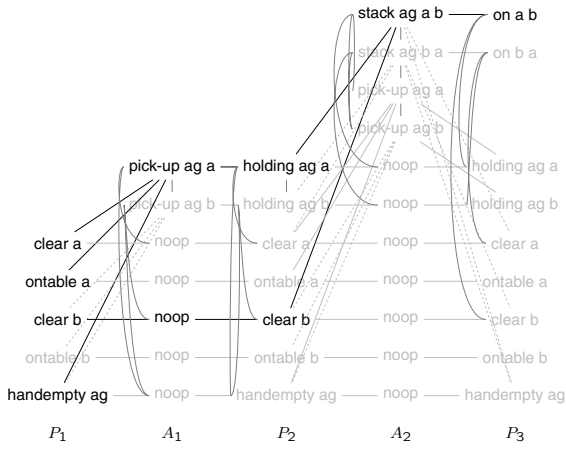


Figure 3. A planning graph generated for a very simple blocks world problem (some nodes and edges are omitted for clarity).

of the operators on the propositions of the time-step. The nodes within the graph are connected with three kinds of edges:

Precondition edges: The precondition edges connect the actions nodes of a time-step n with their preconditions appearing in time-step n.

Delete edges: The delete edges connect the action nodes of a time-step n with their negative effects appearing in time-step $n + 1$.

Add edges: The add edges connect the action nodes of a time-step n with their positive effects appearing in time-step $n + 1$.

For each action node placed into the graph a number of edges are generated that connect the action nodes with their preconditions. Those edges are the precondition edges (see figure 3:P_1 – A_1). Having placed the action nodes into the graph, Graphplan collects their effects which are placed as propositions into the subsequent time-step (see Figure 3:P_2). Subsequently, Graphplan connects the action nodes with their positive effects through a set of add edges (see Figure 3:A_1 – P_2), and with their negative effects through a set of delete edges.

Since the actions of the same time-step are applied on the same state with a finite set of resources there is a potential that the actions will interfere during the consumption of non sharable resources. In order to capture the conflicts that may arise through the parallel execution of actions, Graphplan propagates binary constraints among the actions that appear in the same time-step. The propagated constraints are named mutual exclusion relations (mutex for short) and their meaning is that only one of the two action nodes marked as mutually exclusive can be performed at that time-step. The notion of exclusivity extends to the propositions meaning that two propositions marked as mutually exclusive cannot coexist at the same time-step.

At the completion of the generation of a time-step Graphplan tries to identify the goals and ensure that the goals are mutex free at the newly generated time-step. If the

goals are present and mutex free, Graphplan performs a search within the generated graph in order to extract a plan. The search starts from the goals going backwards (i.e. towards the initial state). Moreover, the search is layered meaning that a new layer of the graph is considered only when the current goal set is fully satisfied with the actions that appear in the current layer. To make this more concrete let us have a look at our example of planning graph shown in figure 3. Graphplan starts from 3:P_3 where it tries to identify the goals. Having found the goals Graphplan tries to support all the goals with a mutex free set of actions from the previous time-step 3:P_2. If that is not possible Graphplan backtracks. If a set is found Graphplan generates a new goal set containing the preconditions of the selected actions and repeats the process with the new goal set. The search succeeds when Graphplan reaches the initial state. Upon the successful completion of the search the extracted plan is returned (the extracted plan of our example is highlighted in figure 3).

Having constructed a plan, the planning module fragments the plan into a set of services. A service consists of a unique identifier of the service, and the name of the agent that has committed to the execution of the service (initially set to $\perp$). More importantly, a service contains a set of actions which are causally related on the actor who needs to perform them. For instance the highlighted plan of figure 3 consists of a service containing the actions 'pick-up ag a' – 'stack ag a b' as the same actor who is going to pick up the block needs also to stack it. An action consists of a literal representation of its instance (in the Prolog formalism), and the point in time that it needs to be executed, hence the precedence constraints among the actions of a service are explicitly captured within the service.

Having received a service the planning module extracts the actions of the service and tries to incorporate the actions into the plan in the time-steps defined in the service. If that is achievable then the service is integrated successfully with the plan and the planning component returns a positive reply (i.e. $\top$), negative otherwise (i.e. $\perp$).

2.2. Social Reasoning

An agent planning and acting in a solitary manner has capabilities that are limited to its own. Thus, the goals it can achieve by functioning as an individual entity are rather constrained. The real potential of agents arises when these solitary entities begin to act as communities. In such a context, opportunities exist for individual agents to compensate for each other's deficiencies by acting collectively, thereby achieving higher overall performance as a system. A key mechanism for coordinating social interaction between agents is *negotiation* [2].

In abstract, negotiation is commonly viewed as a dialectic process that allows two or more parties to interact and resolve *conflicts of interest* that they have among each other with respect some issues of mutual interest [3,4]. For example, in a situation where a buyer agent attempts to purchase a car from a seller agent, there is a clear conflict of interest between the two parties with respect to the price of the car. The buyer is interested in paying the lowest price possible, whereas the seller is interested in gaining the highest price possible (thus, the conflict of interest). Negotiation provides a means for the two agents to resolve their conflict of interest by allowing them to come to a mutually acceptable agreement. Thus, it can be observed that the ultimate goal of the negotiation is to arrive at a mutual agreement and, thereby, resolve the conflict of interest present among the different parties.

Negotiation is so central and fundamental because it provides the agents with the means of influencing the behaviour of their *autonomous* counterparts. By definition, an autonomous entity cannot be forced to adopt a certain pattern of behaviour. Thus, negotiation provides agents with the means to convince their autonomous counterparts by forwarding proposals, making concession, trading options, and, by so doing, (hopefully) arriving at a mutually acceptable agreement [5]. Apart from being used as a means to achieve agreements, negotiation also underpins agents' efforts to coordinate their activities, achieve cooperation, and resolve conflicts in both cooperative [6] and self-interested [7] domains.

Increasingly it is argued that incorporating the ability to exchange arguments within such a negotiation interaction mechanism not only increases the agent's ability to reach agreements, but also enhances the mutual acceptability of the agreement [8,9,10,11]. In more detail, when agents interact within a multi-agent society, in most cases they do so with imperfect knowledge about their own capabilities, preferences, and constraints and those of their counterparts. When agents interact within such knowledge imperfections they may lead to another form of conflict between the agents, termed *conflicts of opinion*, which may hinder the the agents ability to reach agreements or lead them to suboptimal agreements with their counterparts. "Argumentation-Based Negotiation" allows the agents to exchange additional meta-information such as justifications, critics, and other forms of persuasive locutions within their interactions. These, in turn, allow agents to gain a wider understanding of each others capabilities, preferences, and constraints, thereby making it easier to resolve such conflicts that may arise due to incomplete knowledge. In the current implementation the social reasoning component considers three important decisions; namely (i) what agent to argue with, (ii) what issues to argue on, and (iii) within which ranges. The following considers these in more detail:

- **What agent** In considering which counterpart to interact with the social reasoning component considers two important aspects; first the structure of the society and secondly the experience that it has gained in its past encounters. In most instances, an agent society usually embodies a structure. Certain agents may act in certain roles within the agent society which may lead to relationships with other agents acting certain other roles. As a result of these roles and relationships agents may obtain specific obligations to others and may gain rights to influence certain others. These obligations and rights constitute social influences which can be constructively exploited in a society. In such a context, when considering the counterpart to interact with agent could constructively exploit these social influences. For instance, when negotiating for a certain service an agent may choose another which already is obliged to provide this capability through the social structure. Instead of randomly picking any agent in the society, using such a heuristic in selecting its counterpart may make the interacting more efficient. Apart from the social influences of the structure of the society, agents may also take into account the experience gained in its past encounters. In more detail, if the agent has interacted with that agent in the past to obtain a certain service, it may select the same agent when it requires the same type of service again. This may enhance the interaction being more effective since the agent already knows that it has the capability to perform the required service, which may not always be the case in selecting a random agent from the society.

- **What Issues** Once an agent has chosen its counterpart, the next main decision that it needs to make are the set of issues to negotiate with. In the negotiation literature, this set of issues is generally referred to as the negotiation object [5]. For example, when two agents are negotiating the sale of a car they will address a number of parameters such as price, warranty period, and after sale service. Each of these will be a certain negotiation issue, whereas all of these issues taken together will form the negotiation object. One of the advantages of using an argumentation-based negotiation approach is that new issues can be introduced or the existing once be retracted from the negotiation object during the argumentative encounter.
- **What ranges** Finally, the agent would need to decide the ranges (i.e., the upper and the lower limits) for each particular negotiation issue that it should adhere to during its negotiation encounter. The agent may have a certain objective for each particular issue. For instance, the buyer agent may desire to minimise the price paid while trying to maximise the quality or after-sales service parameters. The seller on the other hand may wish to maximise the price while attempting to minimise the after-sales service parameter. Thus, the upper and the lower limits as well as the direction (maximise or minimise) would depend the agents individual desire in the negotiation interaction. A rational agent wouldn't make an offer that costs the agent more than the expected benefit it aims to receive in return. Thus, the upper limit of the all the negotiation issues would have a cost less than the exacted benefit of buying that service. On the other hand, an agent would not make an offer with a negative reward since it will be irrational for another agent to accept such an offer. Between these upper and the lower bounds an agent can derive a series of offers with a combination of issue value tuples that are viable (cost is less than the benefit). This will give the set of proposals that the dialogue module can use in its encounter.

These three components (the agent, the issues, and their respective ranges) compose the dialogue object which is passed to the dialogical argumentation module. In the next section we discuss how an agent argues with the chosen counterpart to gain the service required.

2.3. Dialectical Argumentation

Agents in the *i*-Xchange system possess a number of capabilities, for example, they are able to construct plans for achieving goals (detailed in section 2.1), and they are able to reason in a social context about how to get other agents in the MAS to perform the actions required by the plan (as detailed in section 2.2). Once a plan has been constructed and a strategy for achieving the plan is devised it is necessary to interact with the other agents to engage them in performing the tasks required to satisfy the plan. In the *i*-Xchange this is achieved through the use of argumentative dialogue which is controlled by the *Dialectical Argumentation* (DA) module.

Dialogue games have been proposed as a means to model the interactions between participants during argumentative dialogues. One branch of dialogue game research is into formal dialectical games [12]. These are two-player, turn-taking games in which the moves available to the players represent the locutional acts or utterances made by the participants of a dialogue. In other words a formal dialectical system expresses a protocol

for communication between conversing agents by regulating what an agent can say and when they can say it.

Many dialectical games have been proposed based on the characterisations of a range of dialogical situations, for example, Hamblin's "simple dialectical system" [12] and Mackenzie's DC [13] are targeted towards fallacy research whilst McBurney and Parsons specify some games for use in communication between agents in MAS [14]. The formal dialectic systems used in the *i*-Xchange are represented using the unified specification format introduced in [15]. This representation is part of a unified framework for representing, rapidly implementing and deploying formal dialectic systems called the Architecture for Argumentation (A4A). To facilitate this, the framework incorporates a range of general machinery for representing dialogues and dialectical games. Each dialectical game is itself designed to model the interactions between participants in a particular dialogical situation.

An example of the most basic dialogical interaction between *i*-Xchange agents is illustrated in the sample system output in section 3 where a partial plan has been constructed by the planning module and the social-reasoning module has selected an agent to carry out the partial plan. The DA module initiates a dialogue with the nominated agent to determine whether the other agent is capable of performing the actions required of the partial plan. The dialogue uses a small range of moves to achieve this which are detailed as follows using the A4A schema:

Game

Name iXchange$_0$
Turns ⟨ Liberal, Single ⟩
Participants = {init, resp}
Stores: ⟨CStore, Init, Mixed, Set, Public⟩
⟨CStore, Resp, Mixed, Set, Public⟩

Structure

Initiation
Requirements:
$T_{current} = 0$
Effects:
$T^{init}_{next\_move}$ = ⟨Initiate, (–)⟩

Termination
Requirements:
T_{last_move} = ⟨Affirm, (–)⟩ ∨
T_{last_move} = ⟨Deny, (–)⟩
Effects:
Dialogue$_{status}$ = complete

Moves

⟨**Initiate, (S)**⟩
Requirements:
$T_{Current} = 1$
Effects:
$T^{listener}_{next\_move}$ = ⟨ Acknowledge, (–)⟩

⟨**Acknowledge, (S)**⟩
Requirements:
$T^{listener}_{last\_move}$ = ⟨ Initiate, (-)⟩
Effects:
$T^{listener}_{next\_move}$ = ⟨ Enquire, (PP)⟩

⟨**Enquire, (S)**⟩
Requirements:
$T^{listener}_{last\_move}$ = ⟨ Acknowledge, (–) ⟩
Effects:
$CStore^{speaker}_{current}$ + PP ∧ $CStore^{speaker}_{current}$ +
PP ∧ (T_{next_move} = ⟨Affirm, (PP)⟩ ∨

T_{next_move} = ⟨Deny, (PP)⟩)

⟨**Affirm, (S)**⟩
Requirements:
$T^{listener}_{last\_move}$ = ⟨ Enquire, (PP) ⟩ **Effects:**
$CStore^{speaker}_{current}$ + PP

⟨**Deny, (S)**⟩
Requirements:
$T^{listener}_{last\_move}$ = ⟨ Enquire, (PP) ⟩
Effects:
$CStore^{speaker}_{current}$ + ¬PP ∧
$CStore^{speaker}_{current}$ – PP

The iXchange$_0$ protocol is split into three parts; game, structure and moves. The game part specifies the turn structure, participants, and commitment stores. The structural part specifies the required state for legal initiation of a dialogue and the states under which the dialogue will terminate. The moves part specifies the moves which players can make during a dialogue. Moves are presented in terms of their legality requirements and resultant effects if the move is legally played. Legality requirements are formulated in terms of earlier moves during the dialogue. Effects are formulated in terms of legal responses and commitment store updates.

When all three *i*-Xchange agent modules are integrated in a single agent, the game, iXchange$_0$, is sufficient to enable an agent to engage in a simple dialogue with another and determine whether the other agent can execute a partial plan thus enabling a communication round trip between two agents composed of the basic *i*-Xchange agent modules.

3. Example

This section illustrates the use of the *i*-Xchange MAS when applied to an *e*-Science scenario. The *e*-Science domain consists of a network of host machines, a set of datasets that need to be processed, a set of data transportation mediums that can transfer the datasets between hosts and a set of data processing systems that are needed to be available in the host machine so that the dataset can be processed. The following operators are supported;

move dataset: Moves a data transportation medium loaded with a data set between hosts.

move data processing system: Moves a data transportation medium loaded with a data processing system between hosts.

move data transportation medium: Moves a data transportation between hosts.

load dataset: Loads a dataset to a data transportation medium.

load data processing system: Loads a data processing system to a data transportation medium.

unload dataset: Unloads a data set from a data transportation medium.

unload data processing system: Unloads a data processing system from a data transportation medium.

execute: Executes a data processing system at a given host.

terminate: Terminates the execution of a data processing system.

process: Processes a data set.

The *e*-Science domain has the following properties; Three classes of agents appear in the domain, namely the data processing systems, the data transportation mediums, and the host machines. The *e*-Science domain is a mixture of the transportation class of planning domains (logistics for instance) and the puzzle class of planning domains (for example blocks-world). The first property allows us to model complex societies structured as a hierarchical network of social influences. Hence the agents are provided with a rich social model to exploit the capabilities of their social reasoning component. The second property allows the agents to fully exploit their planning capabilities as the *e*-Science domain allows for the specification of very complex planning problems. The combination of the two properties provides a scenario where complex agent interactions can emerge for conflict resolution and task delegation, hence the agents can exploit their capabilities on dialectical argumentation. The following fragment illustrates system output as the planner generates a partial plan, the social reasoning module nominates an agent to execute the plan and the dialectical argumentation module engages that agent in dialogue.

```
Planner: {Agent=p1, Action=load_dps(t2, p1, m3), Time=1}
Planner: {Agent=p1, Action=move_dps(t2, p1, m3, m2), Time=2}
Planner: {Agent=p1, Action=unload_dps(t2, p1, m2), Time=3}
Planner: {Agent=p1, Action=execute(p1, m2), Time=4}
Planner: {Agent=p1, Action=process(d1, p1, m2), Time=6}
Planner: {Agent=p2, Action=execute(p2, m2), Time=1}
Planner: {Agent=p2, Action=process(d2, p2, m2), Time=2}
Planner: {Agent=p2, Action=terminate(p2, m2), Time=3}
Planner: {Agent=t1, Action=move_dtm(t1, m3, m2), Time=1}
Planner: {Agent=t1, Action=move_dtm(t1, m2, m1), Time=2}
Planner: {Agent=t1, Action=load_ds(t1, d1, m1), Time=3}
Planner: {Agent=t1, Action=move_ds(t1, d1, m1, m2), Time=4}
Planner: {Agent=t1, Action=unload_ds(t1, d1, m2), Time=5}
Planner: {Agent=t2, Action=load_dps(t2, p1, m3), Time=1}
Planner: {Agent=t2, Action=move_dps(t2, p1, m3, m2), Time=2}
Planner: {Agent=t2, Action=unload_dps(t2, p1, m2), Time=3}
ServiceImpl: ID: 77158a:10abd317fb0:-7fcf
|-> name: load_dps(t2, p1, m3) – time: 1
|-> name: move_dps(t2, p1, m3, m2) – time: 2
|-> name: unload_dps(t2, p1, m2) – time: 3
|-> name: execute(p1, m2) – time: 4
|-> name: process(d1, p1, m2) – time: 6
ServiceImpl: ID: 77158a:10abd317fb0:-7fce
|-> name: execute(p2, m2) – time: 1
|-> name: process(d2, p2, m2) – time: 2
|-> name: terminate(p2, m2) – time: 3
ServiceImpl: ID: 77158a:10abd317fb0:-7fcd
|-> name: move_dtm(t1, m3, m2) – time: 1
|-> name: move_dtm(t1, m2, m1) – time: 2
|-> name: load_ds(t1, d1, m1) – time: 3
|-> name: move_ds(t1, d1, m1, m2) – time: 4
|-> name: unload_ds(t1, d1, m2) – time: 5
ServiceImpl: ID: 77158a:10abd317fb0:-7fcc
|-> name: load_dps(t2, p1, m3) – time: 1
|-> name: move_dps(t2, p1, m3, m2) – time: 2
|-> name: unload_dps(t2, p1, m2) – time: 3
Reasoning: addServiceRequest
Reasoning: received ServiceRequest with ID:77158a:10abd317fb0:-
7fcf and actions:
|-> name: load_dps(t2, p1, m3) – time: 1
|-> name: move_dps(t2, p1, m3, m2) – time: 2
|-> name: unload_dps(t2, p1, m2) – time: 3
|-> name: execute(p1, m2) – time: 4
|-> name: process(d1, p1, m2) – time: 6
iXchangeAgent0::Reasoning: invokeAddDialogueGoal
iXchangeAgent0::DialogueModule Dialogue Goal Added
iXchangeAgent0::DialogueModule sending message to: iX-
changeAgent1 with message content:INIT DIALOGUE
iXchangeAgent1::DialogueModule handling message from: iX-
changeAgent0 with message content:INIT DIALOGUE
iXchangeAgent1::DialogueModule sending message to: iX-
changeAgent0 with message content:OK
iXchangeAgent1::DialogueModule calling invokeAddProposal in
Social Reasoning Module
iXchangeAgent1::ReasoningModule: addProposal()
iXchangeAgent1::ReasoningModule: invokeAddServiceEvaluation
Loading scheduler.pl... ok – PlanningModule: iXchangeAgent1:
Schedule for move_dtm(t1, m3, m2) at 1 – result: yes
iXchangeAgent1::PlanningModule::Schedule for move_dtm(t1,
m2, m1) at 2 – result: yes
iXchangeAgent1::PlanningModule::Schedule for load_ds(t1, d1,
m1) at 3 – result: yes
iXchangeAgent1::PlanningModule::Schedule for move_ds(t1, d1,
m1, m2) at 4 – result: yes
iXchangeAgent1::PlanningModule::Schedule for unload_ds(t1, d1,
m2) at 5 – result: yes
iXchangeAgent1::Reasoning: invokeAddProposalResponse
iXchangeAgent1::DialogueModule Response to proposal:
77158a:10abd317fb0:-7fcd iXchangeAgent1
iXchangeAgent1::DialogueModule sending message to: iX-
changeAgent0 with message content:ACCEPT: iXchangeAgent1
iXchangeAgent0::DialogueModule handling message from: iX-
changeAgent1 with message content:ACCEPT: iXchangeAgent1
iXchangeAgent0::DialogueModule calling invokeAddDialogueRe-
sult in Social Reasoning Module iXchangeAgent0::Reasoning: ad-
dDialogueResult()
iXchangeAgent0::Reasoning: invokeAddServiceResponse
iXchangeAgent0::Planning Module: Setting the agent name for
service with ID: 77158a:10abd317fb0:-7fcd to: iXchangeAgent1
ixchange.shared.ServiceImpl@97eded
```

The output demonstrates the iXchange system, starting with the planning module constructing a number of partial-plans. A service is then constructed from each partial plan requiring an action to be performed by an agent at a certain timepoint. The services are then passed to the social-reasoning module which determines which agents can execute each partial plan encapsulated in each service. The social-reasoning module then instantiates a dialogue goal which requires the dialogue module to communicate with the agent selected to perform the service and determine whether that agent will perform

the required actions. If the recipient agent can perform the action then the dialogue is successful and the planner is informed via the social-reasoning module that an agent has been found which has accepted to perform the required action at the determined time-point.

4. Conclusions

There are two facets to the results presented here. The first is that the i-Xchange is the first system to successfully integrate planning, social reasoning and argumentation. Though plans have been maintained in the context of agents communicating, this is the first time that individual agents have been equipped with modern planning techniques that are fully integrated with the communication subsystems, and where communicative “failures" (in the sense of refusals) have been taken into account by the planner on the fly. Similarly, though social structures have long formed a part of agent reasoning, and have contributed to the environment in which planners operate, this is the first time that plan refinement has explicitly involved reasoning about the social context. Finally, though argumentation has often been thought of in a social context [11], this is the first time that social reasoning has been integrated to the execution of specific argument protocols for inter-agent communication.

There is also a practical facet: the i-Xchange clearly demonstrates that the use of heterogeneous engineering techniques, different agent platforms and architectures, and a wide variety of languages and tools can provide a rich but solid foundation for a single, focused, coherent agent system. As the sophistication of individual systems continues to increase, and the scope of functionality becomes ever wider, such cross-platform heterogeneous development is going to become ever more the norm.

References

[1] A. L. Blum and M. L. Furst Fast planning through planning graph analysis. In *Artificial Intelligence*, 90: 281–300, 1997.

[2] M. N. Huhns and L. M. Stephens. Multiagent systems and societies of agents. In Gerhard Weiss, editor, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, chapter 2, pages 79–120. MIT Press, Cambridge, MA, USA, 1999.

[3] N. R. Jennings, S. Parsons, C. Sierra, and P. Faratin. Automated negotiation. In J. Bradshaw and G. Arnold, editors, *Proceedings of the 5th International Conference on the Practical Application of Intelligent Agents and Multi-Agent Systems (PAAM-2000)*, pages 23–30, Manchester, UK, 2000.

[4] A. Lomuscio, M. Wooldridge, and N. R. Jennings. A classification scheme for negotiation in electronic commerce. In F. Dignum and C. Sierra, editors, *Agent-Mediated Electronic Commerce: A European AgentLink Perspective*, volume 1991, pages 19–33. Springer Verlag, March 2001.

[5] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge. Automated negotiation: Prospects, methods and challenges. *Int. J. of Group Decision and Negotiation*, 10(2):199–215, 2001.

[6] R. Mailler, V. Lesser, and B. Horling. Cooperative negotiation for soft real-time distributed resource allocation. In *Proceedings of the Second AAMAS*, pages 576–583, 2003.

[7] J. Rosenschein and G. Zlotkin. *Rules of encounter : designing conventions for automated negotiation among computers*. MIT Press, Cambridge, MA, USA, 1994.

[8] N. R. Jennings, S. Parsons, P. Noriega, and C. Sierra. On argumentation-based negotiation. In *Proceedings of International Workshop on Multi-Agent Systems (IWMAS'98)*, Boston, USA, 1998.

[9] S. Parsons, C. Sierra, and N. R. Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, June 1998.

[10] I. Rahwan, S. D. Ramchurn, N. R. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation-based negotiation. *The Knowledge Engineering Review*, 18(4):343–375, 2003.

[11] K. Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28(3):203–242, May 1990.

[12] C. L. Hamblin. *Fallacies*. Methuen and Co. Ltd, 1970.

[13] J. D. Mackenzie. Question begging in non-cumulative systems. *Journal Of Philosophical Logic*, 8:117–133, 1979.

[14] P. McBurney and S. Parsons. Dialogue games in multi-agent systems. *Informal Logic*, 22(3):257–274, 2002.

[15] S. Wells and C. Reed. Formal dialectic specification. In I. Rahwan, P. Moraitis, and C. Reed, editors, *First International Workshop on Argumentation in Multi-Agent Systems*, 2004.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Argue tuProlog: A Lightweight Argumentation Engine for Agent Applications [1]

Daniel Bryant [a,2], Paul J. Krause [a] and Gerard A.W. Vreeswijk [b]
[a] *Department of Computing, University of Surrey, GU2 7XH, UK*
[b] *Department of Information and Computing Sciences, Universiteit Utrecht, 3508 TA Utrecht, The Netherlands*

Abstract. Argumentation is becoming increasingly important in the design and implementation of autonomous software agents. We believe that agents engaged in decision-making and reasoning should have access to a general purpose argumentation engine that can be configured to conform to one of a range of semantics. In this paper we discuss our current work on a prototype light-weight Java-based argumentation engine that can be used to implement a non-monotonic reasoning component in Internet or agent-based applications.

Keywords. Argumentation Engine, Reasoning in Agents, Argument games

1. Introduction

Agents are often cited as a key enabling technology for the next-generation of online services, such as large-scale electronic commerce [1] and Service-Oriented Computing [2]. In order to be effective agents will often need to reason about what is to be done, i.e. perform practical reasoning [3], but in these situations, consisting of large-scale open multi-agent systems, classical logic-based approaches to reasoning and decision-making are often unsuitable [4]. Accordingly, agents may benefit from the use of argumentation, a process based on the exchange and valuation of interacting arguments, to support the process of practical reasoning.

In this paper we discuss our current work on a light-weight Java-based argumentation engine that can be used to implement a non-monotonic reasoning component in Internet or agent-based applications. The core engine has been built using tuProlog [5,6], an existing open-source Prolog engine, as its foundation, which followed the same design principles that we require for our intended domain of application. Although our ultimate goal is to create a general purpose argumentation engine that can be configured to conform to one of a range of semantics, the current version of the engine implements

[1] This work was partially supported by the EU IST/STReP ASPIC project, Grant 002307, and an EPSRC PhD Studentship.
[2] Correspondence to: Daniel Bryant, Department of Computing, University of Surrey, Guildford, GU2 7XH. UK. Tel: 0044 (0) 1483 682263; E-mail: d.bryant@surrey.ac.uk

the argumentation-based framework presented in [4] (allowing our engine to generate arguments and counter arguments over an inconsistent knowledge base, determine the acceptability of arguments and construct proofs using an argument game approach to constructing proofs of acceptance [7]), and also standard PROLOG inference (allowing us to prototype a variety of metainterpretters that support other forms of argumentation.)

The motivation behind this paper is primarily to illustrate that a practical Internet-ready/agent-based implementation of argumentation is now viable. Our implementation, which we named "Argue tuProlog" (AtuP), will be made available later this year in SourceForge [8] under an Open Source licence. We have quite deliberately focused on this as an empirical application of a theoretical model of argumentation, and do not address theoretical issues directly (although we will return to some of the outstanding issues in the concluding section). This paper is structured as follows: In Section 2 we provide motivation for our work and also introduce tuProlog, the foundation of our engine. Section 3 introduces the ASPIC argumentation framework and in section 4 we discuss how we have implemented this in our engine. We conclude the paper with an overview of the planned future work. A fundamental message of this paper, and we will return to this in our final words, is that we take seriously the need for sound empirical evidence for the applicability of argumentation.

2. The Motivation for Argue tuProlog

There has been much recent work on argumentation-based engines, notably Vreeswijk's *IACAS* [9], Rock and colleagues *Deimos* [10] and García and Simari's *DeLP* [11] (and later an extension to this work, *P-DeLP*, by Chesñevar and colleagues [12]). However, none of these engines implement support for more than one form of argumentation semantics. Good practical reasoning is complex with respect to the argument schemes it can use and only in limited and well-defined domains of decision-making does it make sense to use a single scheme of practical reasoning [3]. Accordingly, it is our belief that agents engaged in reasoning should have access to a general purpose argumentation engine that can be configured to conform to one of a range of semantics.

Our prototype argumentation engine has been built using tuProlog [6] as its foundation. tuProlog is a Java-based Prolog engine which has been designed from the ground up as a thin and light-weight engine that is easily deployable, dynamically configurable and easily integrated into Internet or agent applications [5]. There are a number of advantages to using tuProlog as a foundation for our engine. Firstly, the development of tuProlog itself followed the same design principles that we require for our intended domain of application. Secondly, we are building on top of a mature code-base so that much of the functionality that is common to both argumentation and Prolog-type inference can be relied on with a high-degree of confidence. Thirdly, this ensures that in the absence of defeasible rules, our engine defaults to standard Prolog inference.

Utilising the Prolog inference provided by the tuProlog engine we can implement a series of metainterpretters for a variety of forms of argumentation. However, this way of implementing an argumentation engine has both a serious performance overhead and a less than ideal interface. In order to avoid these problems and produce an argumentation engine that fully conforms to the spirit of a light-weight Internet enabled tool, we are re-engineering tuProlog by implementing a series of core argumentation algorithms in

Java, effectively pushing the functionality of the algorithms down into the core engine. The first algorithm we have implemented in our engine is presented in [4].

3. The Acceptability of Arguments

In [4] a framework for argument games is presented that is concerned with establishing the acceptability of arguments. Argument games between two players, a proponent (PRO) and opponent (OPP), can be interpreted as constructing proofs of acceptance utilising a dialectical structure [7]. The proponent and opponent share the same (possibly inconsistent) knowledge base and the proponent starts with a a main claim to be "proved". The proponent attempts to build an admissible set to support the claim and endeavors to defend any argument against any attack coming from the opponent. The proponent wins the game (proving acceptability of the claim) if all the attacking arguments have been defeated, and the opponent wins if they can find an attacking argument that cannot be defeated. In [4] a prototype web-based implementation (coded in RUBY) of the framework algorithms, entitled "Argumentation System" (AS), is also presented.

4. The Implementation of our Engine

4.1. Overview

AtuP is currently implemented in Java and presented as a self-contained component that can be integrated into a range of applications by utilising the well defined application programming interface (API) provided. The API exposes key methods to allow an agent or Internet application developer to access and manipulate the knowledge base (including the ability to define numerical values indicating the degree of belief of each proposition), to construct rules, specify and execute queries (establishing whether a claim can be supported using the knowledge base) and analyse results (determining the support for a claim and the acceptability of arguments).

4.2. Language

As with the original AS, AtuP accepts formulae in an extended first-order language and returns answers on the basis of the semantics of credulously preferred sets (as defined in [4]). The language of AtuP is constituted of atoms, terms and rules (see Section 2.7.3 in [4] for further details) and can be considered as a conservative extension of the basic language of Prolog, enriched with numbers that quantify degree of belief. As AtuP is built on top of an existing Prolog engine, the engine naturally accepts Prolog programs.

In AtuP the numerical input values in (0, 1] represent the degree of belief (DOB), or the credibility, of a proposition [4]. As stated in [4], the DOB is currently provided to allow experimentation with different methods of argument evaluation and is not intended to express probabilities or represent values from other numerical theories to reason with uncertain or incomplete information. However, in our earlier work [13] we explored the integration of argumentation with a number of numerical calculi, such as the semi-qualitative/ordinal possibilities, and "probability of provability" for a fully numeric scale. In future work we plan to enhance arguments with possibilities as discussed in,

for example, Amgoud [14] or Chesñevar [12]. This provides us with a computationally efficient model with a well-founded semantics. We will then progress to explore the integration of a numerical calculus that has a sound probabilistic semantics.

Within AtuP there are two types of different rules, namely those with an empty antecedent (called beliefs) and those with a non-empty antecedent (called rules). Every expression of the form $t\ b.$ is a rule where t is a term and b indicates a degree of belief. Examples of beliefs include `a 0.8.` and `flies(sylvester) 0.1.` Every expression of the form t :- $t_1, ..., t_n\ b.$ is also a rule, provided t, t_i are terms, $n > 0$, and b denotes the DOB. Examples of rules include `flies(X) :- bird(X) 0.8.` and `a :- c,d 1.0.` A query is an expression of the form ?- $t_1, ...t_n.$ where $n > 0$. It is possible to include more queries in the input, but since we are usually only interested in one goal proposition, this is not typical.

4.3. Algorithms

If ?- $t.$ is a query then AtuP's main goal is to try and find an argument with conclusion t and then try to construct an admissible set (using the algorithm presented in [4]) around that argument. In AtuP every search for arguments for a particular query is encapsulated within another internal instance of an engine. Using multiple internal instances of an engine allows us to keep track of which participant (PRO or OPP) is conducting the current query and also to pause the "dialogue" at any time for further analysis. Once the first argument, say A, is found, the first engine is suspended and A is returned to AtuP. AtuP then tries to find an attacker of A. Thus for every sub-conclusion s of A, a separate engine is instantiated to search for arguments against s. If one of these remains undefeated (which is defined within [4]), then A is defeated, else A remains undefeated.

4.4. Getting Results

When AtuP has finished determining the support for a claim and the acceptability of associated arguments the engine generates a trace of the argument game dialogue (shown in the window on the right of Figure 1). In addition to providing an API to allow agent developers to utilise our engine we have also modified the existing tuProlog graphical user interface to facilitate off-line experimentation with the engine (as shown in Figure 1). We have also developed the core engine using Sun Microsystem's NetBeans integrated development environment in which we have installed the latest version of NetBeans Profiler [15], a fully functional application profiling tool. This allows us to simulate deployment of our engine within a variety of realistic scenarios, and to monitor and analyse such data as CPU usage, memory usage, program loop/branch counting, thread profiling and other basic Java Virtual Machine (JVM) behaviour. We are currently in the process of setting up several large-scale knowledge bases, and when this is complete the profiling tool will facilitate our ultimate goal of obtaining empirical evaluations of the performance of a range of argumentation models.

5. What did argumentation ever do for us?

The above question is easier to answer of the Romans than it is of argumentation. Our current work is a first step in trying to set up some real-world experiments that will help

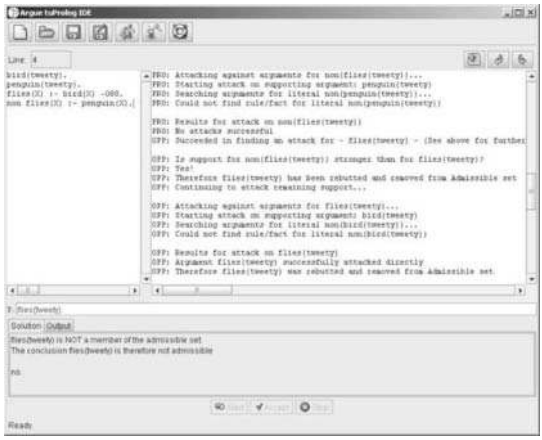


Figure 1. Screenshot of "Argue tuProlog" GUI. The left window allows manipulation of the knowledge base, the bottom window allows query entry and displays the results and the right window shows an argument game trace after a query has been executed.

us answer our question. In this paper we have presented a prototype light-weight Java-based argumentation engine which is capable of facilitating automated reasoning and decision-making, and is suitable for deployment into Internet and agent applications. We have also discussed the integration of an argumentation-based framework for determining the acceptability of arguments, as presented in [4], into our engine.

Our immediate next step is to set up some large-scale knowledge bases that will enable us to obtain empirical evaluations of the performance of the engine. This work is in hand now, and we expect to be able to report the results within the next three months. At that point we will feel confident to release the engine into the wider community, together with a clear definition of its scope and limitations. As well as gaining empirical data on the applicability and performance of a specific instance of an argumentation engine, we will also be evaluating a series of enhancements. As far as possible we are aiming towards implementing a general purpose argumentation engine that can be configured to conform to one of a range of semantics. Our basic position is that we have no prior disposition towards any one model of argumentation. Instead, our plan is to explore a range of models to provide an independent evaluation of their expressive power, performance and scalability.

Argumentation is inherently computationally challenging. As a reminder of just one point, the consistency of a set of first-order logical formulae is undecidable. Yet, most or all definitions of an argument refer to the selection of a "consistent subset" of formulae. We need to investigate theoretical approaches to easing this blocking issue for implementations (e.g. [16]), and indeed we have a parallel strand of research that is targeting this. However, we also feel that it is important to make publicly available our and other "pragmatic" implementations of argumentation, together with some large-scale benchmarking knowledge bases. This way we can also stimulate scientific evaluations of what is practically possible now, and really find out what argumentation can do for us.

Acknowledgements

We would like to gratefully thank Mariam Tariq for use of her earlier implementation work and also the tuProlog team at the University of Bologna for their enthusiastic support of our work. We also offer our thanks to the partners from the ASPIC project for useful discussions. Finally, we would like to greatly thank the anonymous reviewers for their insightful and very helpful comments.

References

[1] C. Guilfoyle, J. Jeffcoate, and H. Stark. *Agents on the Web: Catalyst for E-Commerce*. Ovun Ltd. London, 1997.

[2] M. P. Papazoglou. Service-Oriented Computing: Concepts, characteristics and directions. In *WISE '03: Proceedings of the Fourth International Conference on Web Information Systems Engineering*, page 3, Washington, DC, USA, 2003. IEEE Computer Society.

[3] R. Girle, D. Hitchcock, P. McBurney and B. Verheij. Decision support for practical reasoning. In C. C. Reed and T. J. Norman, editors, *Argumentation Machines*, pages 56–83. Kluwer Academic Publishers, The Netherlands, 2004.

[4] L. Amgoud, M. Caminada, S. Doutre, H. Prakken, and G. Vreeswijk. Draft formal semantics for ASPIC system. Technical Report ASPIC Deliverable 2.5, 2005.

[5] E. Denti, A. Omicini, and A. Ricci. Multi-paradigm java-prolog integration in tuProlog. *Sci. Comput. Program.*, 57(2):217–250, 2005.

[6] E. Denti, A. Omicini, and A. Ricci. tuProlog: A light-weight prolog for internet applications and infrastructures. In I. V. Ramakrishnan, editor, *PADL*, volume 1990 of *Lecture Notes in Computer Science*, pages 184–198. Springer, 2001.

[7] H. Jakobovits and D. Vermeir. Dialectic semantics for argumentation frameworks. In *International Conference on Artificial Intelligence and Law*, pages 53–62, 1999.

[8] SourceForge.net. *Welcome to SourceForge.net*. available from http://sourceforge.net/, 2006. Last accessed: 10 May 2006.

[9] G. A. W. Vreeswijk. IACAS: An implementation of Chisholm's principles of knowledge. In *The proceedings of the 2nd Dutch/German Workshop on Nonmonotonic Reasoning, Utrecht.*, pages 225–234, 1995.

[10] Michael J. Maher, Allan Rock, Grigoris Antoniou, David Billington, and Tristan Miller. Efficient defeasible reasoning systems. *International Journal on Artificial Intelligence Tools*, 10(4):483–501, 2001.

[11] Alejandro J. Garcia and Guillermo R. Simari. Defeasible logic programming: an argumentative approach. *Theory Pract. Log. Program.*, 4(2):95–138, 2004.

[12] Carlos I. Chesnevar, Guillermo R. Simari, Teresa Alsinet, and Lluis Godo. A logic programming framework for possibilistic argumentation with vague knowledge. In *AUAI '04: Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 76–84, Arlington, Virginia, United States, 2004. AUAI Press.

[13] P. Krause, S. Ambler, M. Elvang-Goransson, and J. Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11:113–131, 1995.

[14] L. Amgoud and H. 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.

[15] Sun Microsystems. *NetBeans Profiler*. available from http://profiler.netbeans.org/, 2006. Last accessed: 10 May 2006.

[16] A. Hunter. Approximate arguments for efficiency in logical argumentation. In *Proceedings of NMR-06*, 2006. *In press*.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

An Application of Formal Argumentation: Fusing Bayes Nets in MAS

Søren Holbech Nielsen [a], Simon Parsons [b]
[a] *Department of Computer Science*
Aalborg University, Aalborg
Denmark
[b] *Department of Computer and Information Science*
Brooklyn College, City University of New York
Brooklyn, 11210 NY, USA

Abstract. We consider agents in a multi-agent system, each equipped with a Bayesian network model (BN) of its environment. We want the agents to reach consensus on one compromise network, which may not be identical to a single one of the BNs initially held by the agents, but rather a combination of aspects from each BN. The task can be characterized as the need for agents to agree on a specific state (a BN) of a variable with an enormous state space (all possible BNs). The grandness of the task is reduced by the fact that BNs are composed of local relationships, and it should therefore be possible to reach the compromise by gradually agreeing on parts of it. In the metaphor of the variable, the agents should be able to agree on successively smaller subsets of the enormous state space. However, these same local relationship can interact, and understanding the extent to which partial agreements affect the possible final compromise is a highly complex task. In this work we suggest using formal argumentation as the reasoning mechanism for agents solving this task, and suggest an open-ended agora approach that ensures agents high quality compromises in an anytime fashion.

Keywords. Argumentation, Bayesian networks, Compromises

1. Introduction

We investigate how Bayesian networks (BNs) can be used in a multi-agent setting with the help of argumentation theory. Previously the two methodologies have mainly been studied together with a view to incorporating the efficiency and precision of BNs into argumentation theory (e.g. [1]), or as an exercise in converting models of one theory into models of the other (e.g. [2] and [3]). Here, we envision equipping each agent in a MAS with a BN, as a model of the domain it is situated in, and aim at providing a framework built on formal argumentation principles in which the agents, starting from their individual domain models, can conclude on a single network representing their joint domain knowledge. This would be useful in cases where the agents only meet occasionally and in the meantime may make small changes to their models to reflect surprising observations of their surroundings. By using the two paradigms in this manner, we hope to exploit the strengths of BNs and of argumentation: Allowing individual agents to draw inferences in

face of noisy observations using their BNs, and having agents extract a consistent "truth" from a set of conflicting ones through a distributed process built on argumentation.

The task of fusing several BNs into one compromise BN is made theoretically interesting by the fact that BNs by their graphical nature can be decomposed into several local relationships, and thus the aspect of gradually building a compromise BN bottom up is tempting. However, these very same local relationships can interact in complex ways, and the consequences of committing oneself to a partial compromise can be hard to estimate. Maybe because of this difficulty, the task has previously mainly been considered a centralized one-off operation, with little consideration given to these "cascading" effects. Furthermore, the task has been addressed with an a priori specified view to what constitutes an adequate compromise, with no apparent consensus on the goal of network fusion among authors (see [4], [5], [6], and [7]). In this paper, we do not commit ourselves to a specific compromise objective. Rather, we establish a general framework in which any kind of compromise on BNs can be reached in a gradual manner, with the exact nature of the proceedings specified by some parameter functions.

As presented the setup may be confused with a negotiation problem, where the agents would try to negotiate a compromise that is close to their individually held beliefs. However, unlike the standard negotiation setup, the parts of the problem cannot be valued in isolation, and hence, to the individual agent the value of an already agreed upon partial compromise, will depend on the compromise choices that remain to be made. For the same reason, the problem cannot be seen as a distribution of resources, as the individual agents utility of the "resources" would change according to how the remaining ones are distributed. Instead, we hope only to provide the agents with the ability to determine the extent to which they commit themselves at each step in the construction of the compromise. That is, the main focus of our work is to provide the reasoning mechanism individual agents can use for surveying the consequences of committing to partial compromises. The advantages of our approach over previous efforts include: That a general purpose argumentation engine can be implemented and reused in contexts with different definitions of compromise; that efficient distributed implementations are natural; that in cases where agents almost agree a priori, little information need to be shared among the agents; and that anytime compromises can be achieved.

2. Preliminaries And Problem Definition[1]

2.1. Bayesian Nets

A BN $\mathcal{B}$ is an acyclic directed graph (DAG) $\mathcal{G}$,[2] over a set of random variables $\boldsymbol{V}$, along with a conditional probability distribution for each variable in $\boldsymbol{V}$ given its parents in $\mathcal{G}$. The joint probability distribution P over $\boldsymbol{V}$, obtained by multiplying all these conditional probability distributions, adheres to a number of conditional (in)dependence constraints identifiable from $\mathcal{G}$ alone. Any other BN $\mathcal{B}'$ with a graph implying the exact same constraints on P is said to be *equivalent* to $\mathcal{B}$. [9] proved that the set of all BNs equivalent to some BN $\mathcal{B}$ can be uniquely characterized by a partially directed graph called the *pattern* of $\mathcal{B}$. The pattern of $\mathcal{B}$ is constructed by taking the skeleton of $\mathcal{B}$ and directing

[1] For actual examples and background on the topics, ideas, and algorithms presented here and later, see [8].
[2] We assume the reader is familiar with the basics of graph theory.

links as they appear in $\mathcal{B}$ iff they participate in a v-structure[3]. Any BN equivalent to $\mathcal{B}$ can be obtained from its pattern by exchanging links for directed arcs, while taking care that no directed cycles are introduced, and that no v-structures not already found in the pattern are introduced. Any DAG obtained from the pattern in this manner is called a *consistent extension* of the pattern. The two constraints imply that not all partially directed graphs are patterns of some BN, and furthermore that some links in a pattern are exchanged for similarly directed arcs in all consistent extensions of the pattern. Such arcs and arcs found in the pattern are called *compelled arcs*, and the partially directed graph obtained by exchanging links for compelled arcs wherever possible, is called the *completed* pattern of $\mathcal{B}$. The completed pattern of $\mathcal{B}$ is thus a unique characterization of $\mathcal{B}$'s equivalence class as well. The set of all partially directed graphs over $\boldsymbol{V}$ that are completed patterns of some BN thus constitute a complete and minimal encoding of all probabilistic dependencies for distributions expressible by BNs over $\boldsymbol{V}$. We denote this set of completed patterns $\mathfrak{C}^{\boldsymbol{V}}$ and $\mathfrak{C}$ when $\boldsymbol{V}$ is obvious from the context. [10] gave an elegant characterization of the individual elements of $\mathfrak{C}$. Next, we present how agreeing on BNs pose problems.

2.2. Compromising On Bayesian Networks

The problem we are posing arises in a MAS containing a finite number of cooperating agents. Each agent i has a BN $\mathcal{B}_i$ over a common set of domain variables $\boldsymbol{V}$, which we assume to be implicit in the remainder of the text. For ease of exposition, we furthermore assume that an arbitrary but fixed total ordering $\rightsquigarrow$ over the variables is known by all agents a priori. At some point agents 1 to k decide to pool their knowledge, as represented by $\mathcal{B}_1$ to $\mathcal{B}_k$, into a new BN $\mathcal{B}_*$. Facilitating this task is the problem addressed here. We expect $\mathcal{B}_1$ to $\mathcal{B}_k$ to be large but somewhat similar (as each describe relationships among the same variables), and therefore that having each agent communicate its entire model to each other agent is inefficient. We focus solely on the graphical structure of $\mathcal{B}_*$

As all consistent extensions of a completed pattern imply the exact same independence properties, it is reasonable to consider completed patterns as basic representations of domain knowledge, if domain knowledge is taken to be independence properties as in this text. That is, we only require the agents to agree on the completed pattern $\mathcal{G}_* \in \mathfrak{C}$ of $\mathcal{B}_*$.

To establish whether a graph is a good compromise for the agents, we need a measure for how well such graphs matches each of $\mathcal{B}_1$ to $\mathcal{B}_k$. Furthermore, as we plan to build this compromise gradually, we wish for this measure to be relative to an already agreed *partial compromise*. For example, it may be the case that an important dependency between two variables is already a consequence of a partial compromise, and further connections between the two variables may then be of little value. Contrarily, had the partial compromise not implied this dependency, connections that would ensure it are valuable. In general, we cannot assume that a partially specified graph is suitable as representation of a partial compromise, as this might include agreements on what should *not* be part of the final compromise. Therefore, we take a partial compromise $\mathcal{P} \equiv (\boldsymbol{P}_+, \boldsymbol{P}_-)$ to be two sets of sentences in some language, where $\boldsymbol{P}_+$ describe aspects that should be true of the compromise graph, and $\boldsymbol{P}_-$ describe aspects that cannot be true.

[3] A triple of variables (X, Z, Y) is a v-structure if X and Y are non-adjacent and both are parents of Z.

For any three partial compromises $\mathcal{P}$, $\mathcal{P}^a$, and $\mathcal{P}^b$, where $\boldsymbol{P}_+ \subseteq \boldsymbol{P}_+^a$, $\boldsymbol{P}_- \subseteq \boldsymbol{P}_-^a$, $\boldsymbol{P}_+ \subseteq \boldsymbol{P}_+^b$ and $\boldsymbol{P}_- \subseteq \boldsymbol{P}_-^b$, we assume that each agent i can compute its *compromise scores* $s_i(\mathcal{P}, \mathcal{P}^a)$ and $s_i(\mathcal{P}, \mathcal{P}^b)$ such that $s_i(\mathcal{P}, \mathcal{P}^a) > s_i(\mathcal{P}, \mathcal{P}^b)$ iff $\mathcal{P}^a$ describes $\mathcal{B}_i$ better than $\mathcal{P}^b$, given that $\mathcal{P}$ has already been accepted as being descriptive of $\mathcal{B}_i$. A simple example of $s_i((\boldsymbol{P}_+, \boldsymbol{P}_-), (\boldsymbol{P}_+^a, \boldsymbol{P}_-^b))$ could be the number of features described in $\boldsymbol{P}_+^a \setminus \boldsymbol{P}_+$ and $\boldsymbol{P}_-^a \setminus \boldsymbol{P}_-$, which are consistent with $\mathcal{B}_i$, minus those that are not. A more complex score could weigh each of these described features according to the empirical evidence the agent has in favor of or against them. We will assume s_i to be additive, i.e. for any three partial compromises $\mathcal{P}^0$, $\mathcal{P}^1$, and $\mathcal{P}^2$, where $\boldsymbol{P}_+^0 \subseteq \boldsymbol{P}_+^1 \subseteq \boldsymbol{P}_+^2$ and $\boldsymbol{P}_-^0 \subseteq \boldsymbol{P}_-^1 \subseteq \boldsymbol{P}_-^2$, it is the case that $s_i(\mathcal{P}^0, \mathcal{P}^2) = s_i(\mathcal{P}^0, \mathcal{P}^1) + s_i(\mathcal{P}^1, \mathcal{P}^2)$. Notice, that here we do not attempt to define what it means to be a "better description", since we believe that this issue can be dependent on the actual setting in which the framework is to be used, as stated in Section 1.

In addition to the compromise score, we also assume that the agents know the *combination function* $c : \mathbb{R}^k \to \mathbb{R}$, indicating how much trust should be put into the individual agents' models. Differences in trust can be justified by differences in experiences, sensor accuracies, etc. Formally, we define c as follows: Let $\mathcal{P}$, $\mathcal{P}^a$, and $\mathcal{P}^b$ be partial compromises. If

$$c(s_1(\mathcal{P}, \mathcal{P}^a), \ldots, s_k(\mathcal{P}, \mathcal{P}^a)) > c(s_1(\mathcal{P}, \mathcal{P}^b), \ldots, s_k(\mathcal{P}, \mathcal{P}^b)),$$

when $\mathcal{P}^a$ is a better compromise than $\mathcal{P}^b$ for the group of agents 1 to k, given that they have already agreed on $\mathcal{P}$, then c is the combination function for agents i to k. (An obvious choice for c would be a linear combination of its inputs.) We refer to $c(s_1(\mathcal{P}, \mathcal{P}^a), \ldots, s_k(\mathcal{P}, \mathcal{P}^a))$ as the *joint compromise score* of $\mathcal{P}^a$ given $\mathcal{P}$, and like s_i, we shall also assume that c is additive.

With this notation in place, we can thus restate the problem more formally as finding a partial compromise $\mathcal{P}$, which uniquely identifies some graph $\mathcal{G}^* \in \mathfrak{C}$, such that

$$c(s_1((\varnothing, \varnothing), \mathcal{P}), \ldots, s_k((\varnothing, \varnothing), \mathcal{P})) \geq c(s_1((\varnothing, \varnothing), \mathcal{P}'), \ldots, s_k((\varnothing, \varnothing), \mathcal{P}')),$$

for all other partial compromises $\mathcal{P}'$, which uniquely identifies a graph $\mathcal{G}' \in \mathfrak{C}$.

As presented here, it is clear that the problem is not of a simple binary nature, as we are not trying to establish whether some proposition is true or not, and that we are furthermore dealing with a setting in which more than two agents may interact. Consequently, we cannot utilize the vast literature on dialectic proof theories directly. Rather, the problem we are trying to solve is a distributed maximization over a super exponential hypothesis space ($\mathfrak{C}$). Furthermore, as the worth of (partial) compromises are only specified in relation to already agreed upon compromises, the problem is of a highly dynamic nature.

Our solution to the problem is divided into three parts. First, we create a finite language with which graphs and some essential properties of these can be expressed; second and most importantly, we construct an argumentation system with which the agents can reason about consequences of committing to partial compromises; and thirdly, we create an agora in which the agents can reach compromise graphs in an anytime fashion. First, however, we describe the formal argumentation framework we have selected as a reasoning mechanism.

2.3. Formal Argumentation Frameworks

Formal argumentation takes many forms, but here we see it as an approach to extracting consistent knowledge from a possibly inconsistent knowledge base. No single methodology has yet to stand out as the main approach to argumentation (see [11] for an overview of a series of approaches), so it has been necessary to pick one from a large pool of these. The framework we have picked for our purpose is the framework of [12] (which is a proper generalization of that of [13]), as this is an abstract framework, which leaves the underlying language unspecified, and thus does not force us to specify in advance the reasons to which each agent i may attribute its belief in aspects of $\mathcal{B}_i$.

An *argumentation system* is defined as a pair $\mathcal{A} \equiv (\boldsymbol{A}, \triangleright)$, where $\boldsymbol{A}$ is a set of *arguments*, and $\triangleright \subseteq (2^{\boldsymbol{A}} \setminus \{\varnothing\}) \times \boldsymbol{A}$ is an *attack relation*. The exact nature of an argument is left unspecified, but examples could be "In $\mathcal{B}$ there is an arc from X to Y and Y and Z are adjacent, so there must be an arc from Y to Z" or "Because I have observed r, I believe there is an arc from X to Y in $\mathcal{B}$". For two sets of arguments $\boldsymbol{S} \subseteq \boldsymbol{A}$ and $\boldsymbol{S}' \subseteq \boldsymbol{S}$ and an argument A, if $\boldsymbol{S}' \triangleright A$ then $\boldsymbol{S}$ is said to *attack* A. If no proper subset of $\boldsymbol{S}'$ attacks A, then $\boldsymbol{S}'$ is called a *minimal attack* on A. An example of an attack that would make sense is "There is an arc from X to Y in $\mathcal{B}$"[4]$\triangleright$"There is an arc from Y to X in $\mathcal{B}$".

A *semantics* of an argumentation framework is a definition of the arguments in the framework that should be accepted by a rational individual. [13] and [12] work with a wide range of semantics, but we only introduce those needed here: We define a set of arguments $\boldsymbol{S} \subseteq \boldsymbol{A}$ as being *conflict-free*, if there is no argument $A \in \boldsymbol{S}$ such that $\boldsymbol{S}$ attacks A. We further define a single argument A as being *acceptable with respect to a set of arguments* $\boldsymbol{S}$, if for each set of arguments $\boldsymbol{T} \subseteq \boldsymbol{A}$, where $\boldsymbol{T} \triangleright A$, there is an argument B in $\boldsymbol{T}$, such that $\boldsymbol{S}$ attacks B. A conflict-free set $\boldsymbol{S}$, where all arguments in $\boldsymbol{S}$ are acceptable with respect to $\boldsymbol{S}$, is called *admissible*.

A credulous semantics is that of a *preferred extension*, which is an admissible set that is maximal wrt. set inclusion. Finally, an admissible set $\boldsymbol{S}$ is said to be a *stable extension*, if it attacks all arguments in $\boldsymbol{A} \setminus \boldsymbol{S}$. Clearly, a stable extension is a preferred extension as well.

In general it is hard to compute a preferred extension [14], but in [15] we have adapted a technique of [16] to the problem of enumerating preferred extensions of argumentation systems of [12]: Given $\mathcal{A} \equiv (\boldsymbol{A}, \triangleright)$, we define an $\mathcal{A}$-*candidate* as a triple $(\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U} \equiv \boldsymbol{A} \setminus (\boldsymbol{I} \cup \boldsymbol{O}))$ where

- $\boldsymbol{I} \cap \boldsymbol{O} = \varnothing$,
- every argument that is attacked by $\boldsymbol{I}$ is in $\boldsymbol{O}$, and
- every argument A, for which there exists $\boldsymbol{S} \subseteq \boldsymbol{I}$ and $B \in \boldsymbol{I}$, such that $\boldsymbol{S} \cup A \triangleright B$, is in $\boldsymbol{O}$.

(Here $\boldsymbol{I}$ is supposed to capture the intuition of arguments that are *i*n the preferred extension, as opposed to *o*ut and *u*nassigned.)

Given an $\mathcal{A}$-candidate $\mathcal{C} \equiv (\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$ and an argument $A \in \boldsymbol{U}$ the triples $\mathcal{C} - A \equiv (\boldsymbol{I}_{-A}, \boldsymbol{O}_{-A}, \boldsymbol{U}_{-A} \equiv \boldsymbol{A} \setminus (\boldsymbol{I}_{-A} \cup \boldsymbol{O}_{-A}))$ and $\mathcal{C} + A \equiv (\boldsymbol{I}_{+A}, \boldsymbol{O}_{+A}, \boldsymbol{U}_{+A} \equiv \boldsymbol{A} \setminus (\boldsymbol{I}_{+A} \cup \boldsymbol{O}_{+A}))$ are given by:

$$\boldsymbol{I}_{-A} \equiv \boldsymbol{I}, \qquad \boldsymbol{O}_{-A} \equiv \boldsymbol{O} \cup A \qquad \boldsymbol{I}_{+A} \equiv \boldsymbol{I} \cup A, \text{ and} \qquad \boldsymbol{O}_{+A} \equiv \boldsymbol{O} \cup \boldsymbol{\Delta}_{\mathcal{C}+A},$$

[4] To reduce clutter, we leave out { and } for singleton sets.

where $\boldsymbol{\Delta}_{\mathcal{C}+A}$ contains all arguments in $\boldsymbol{U} \setminus A$ which need to be in $\boldsymbol{O}_{+A}$ in order for $C + A$ to be a candidate. If A does not participate in a minimal attack on itself (which is the case for all arguments of the argumentation system we construct in this paper), then both $\mathcal{C} - A$ and $\mathcal{C} + A$ are $\mathcal{A}$-candidates themselves, and we can thus construct *candidate trees*, where each node is an $\mathcal{A}$-candidate: Each $\mathcal{A}$-candidate $\mathcal{C}$ has two children $\mathcal{C} - A$ and $\mathcal{C} + A$, for some arbitrary chosen A in $\boldsymbol{U}$, except those candidates where $\boldsymbol{U} = \varnothing$, which act as leaves in the tree. A candidate tree having candidate $\mathcal{C}$ as root, is called a $\mathcal{C}$*-tree*.

It can be proven that if $\boldsymbol{I}$ is a preferred extension of $\mathcal{A}$, then there is a leaf $(\boldsymbol{I}, \boldsymbol{O}, \varnothing)$ of any $(\varnothing, \varnothing, \boldsymbol{A})$-tree. Conversely, for any leaf $(\boldsymbol{I}, \boldsymbol{O}, \varnothing)$ in a $(\varnothing, \varnothing, \boldsymbol{A})$-tree, where $\boldsymbol{I}$ defends itself, $\boldsymbol{I}$ is admissible. It follows that, by constructing an arbitrary $(\varnothing, \varnothing, \boldsymbol{A})$-tree, all preferred extensions can be enumerated.

3. Encoding Graphs

For the agents to conclude on the best compromise $\mathcal{G}_*$, a formal language $\boldsymbol{L}$ for expressing graphs and properties of graphs must be defined. For efficiency reasons we aim to make this language finite and as small as possible, while ensuring that it is still sufficiently powerful to describe any graph and its membership status in $\mathfrak{C}$.

First, we introduce a small language $\boldsymbol{L}^g$ for encoding graphs:

Definition 1 (Simple Graph Language). *The language $\boldsymbol{L}^g$ is the set containing the sentences* `Arc(X,Y)`, `Arc(Y,X)`, `Link(X,Y)`, *and* `NonAdjacent(X,Y)` *iff X and Y ($X \rightsquigarrow Y$) are distinct variables.*

A *graph knowledge base* is a set $\boldsymbol{\Sigma}^g \subseteq \boldsymbol{L}^g$. Further:

Definition 2 (Consistent Graph Knowledgebases). *Given a graph knowledge base $\boldsymbol{\Sigma}^g$, if it holds that for all pairs of variables X and Y, where $X \rightsquigarrow Y$, a maximum of one of* `Arc(X,Y)`, `Arc(Y,X)`, `Link(X,Y)`, *and* `NonAdjacent(X,Y)` *is in $\boldsymbol{\Sigma}^g$, then we call $\boldsymbol{\Sigma}^g$ a* consistent graph knowledge base (CGK).

The *graph encoded by* a CGK $\boldsymbol{\Sigma}^g$ is the graph $\mathcal{G}[\boldsymbol{\Sigma}^g]$ resulting from starting with the graph with no edges, and then for any two nodes X and Y ($X \rightsquigarrow Y$) adding an arc from X to Y if `Arc(X,Y)` is in $\boldsymbol{\Sigma}^g$, an arc from Y to X if `Arc(Y,X)` is in $\boldsymbol{\Sigma}^g$, or an undirected edge if `Link(X,Y)` is in $\boldsymbol{\Sigma}^g$. It is easy to see that graph encoded by a CGK is well-defined. Furthermore, given a graph $\mathcal{G}$ there exists at least one CGK, for which $\mathcal{G}$ is the encoded graph.

We thus have that any graph can be efficiently encoded as a CGK, and Definition 2 allows us to distinguish the graph knowledge bases, which can be interpreted as graphs, from those that cannot. Next, we extend $\boldsymbol{L}^g$ into a language powerful enough for building a reasoning engine about graphs and their membership status of $\mathfrak{C}$ on top:

Definition 3 (Graph Language). *The graph language $\boldsymbol{L}$ is the set containing all sentences in $\boldsymbol{L}^g$ and*

- `ArcNotAllowed(X,Y)`,
- `DirectedPath(X,Y)`,

- `UndirectedPath(X,Y)`,
- `UndirectedPath(X,Y)Excluding(Z,W)`,
- `¬DirectedPath(X,Y)`,
- `¬UndirectedPath(X,Y)`, *and*
- `¬UndirectedPath(X,Y)Excluding(Z,W)`,

for any choice of distinct variables[5] X, Y, Z, *and* W ($Z \rightsquigarrow W$). *Sentences of the last six kinds will be referred to as* path *sentences.*

The sentences just introduced are supposed to be used as descriptors of attributes of the graphs encoded by CGKs: `ArcNotAllowed(X,Y)` states that an arc from X to Y would not be strongly protected[6], which is required of all arcs in a completed pattern, while the remaining sentences should be self-explanatory (e.g. `¬UndirectedPath(X,Y)Excluding(Z,W)` states that there is no undirected path between X and Y, or that any such path necessarily contains either Z or W).

As $\boldsymbol{L}^g$ is a subset of $\boldsymbol{L}$, it follows that a graph knowledge base is a set of sentences in $\boldsymbol{L}$ as well, and given a set $\boldsymbol{\Sigma}$ of sentences of $\boldsymbol{L}$, we denote by $\boldsymbol{\Sigma}^g$ the set $\boldsymbol{\Sigma} \cap \boldsymbol{L}^g$. In particular Definition 2 is still applicable.

4. Graph Argumentation System

Building on the language $\boldsymbol{L}$ introduced above, we define an argumentation system for distinguishing completed patterns that could be compromises for the agents. The system that we construct enjoys the properties that a graph is a member of $\mathfrak{C}$ iff there is a preferred extension of the system which encodes this graph.

Definition 4 (Graph Argumentation System). *The* graph argumentation system $\mathcal{A}^g$ *is the tuple* $(\boldsymbol{L}, \triangleright^g \subseteq (2^{\boldsymbol{L}} \times \boldsymbol{L}))$, *where* $\triangleright^g$ *is defined as follows (*`{A,B}` *is short-hand for any one of* `(A,B)` *and* `(B,A)`*):*

1. `Arc(X,Y)` $\triangleright^g$ `Arc(Y,X)`
2. `Arc(X,Y)` $\triangleright^g$ `Link{X,Y}`
3. `Arc(X,Y)` $\triangleright^g$ `NonAdjacent{X,Y}`
4. `Link(X,Y)` $\triangleright^g$ `Arc{X,Y}`
5. `Link(X,Y)` $\triangleright^g$ `NonAdjacent{X,Y}`
6. `NonAdjacent(X,Y)` $\triangleright^g$ `Arc{X,Y}`
7. `NonAdjacent(X,Y)` $\triangleright^g$ `Link{X,Y}`
8. `¬DirectedPath(X,Y)` $\triangleright^g$ `DirectedPath(X,Y)`
9. `¬UndirectedPath(X,Y)` $\triangleright^g$ `UndirectedPath(X,Y)`
10. `¬UndirectedPath(X,Y)Excluding(Z,W)` $\triangleright^g$ `UndirectedPath(X,Y)Excluding(Z,W)`
11. `Arc(X,Y)` $\triangleright^g$ `¬DirectedPath(X,Y)`
12. `Link(X,Y)` $\triangleright^g$ `¬UndirectedPath{X,Y}`
13. `Link(X,Y)` $\triangleright^g$ `¬UndirectedPath{X,Y}Excluding(Z,W)`
14. {`DirectedPath(X,Y)`, `DirectedPath(Y,Z)`} $\triangleright^g$ `¬DirectedPath(X,Z)`
15. {`DirectedPath(X,Y)`, `UndirectedPath{Y,Z}`} $\triangleright^g$ `¬DirectedPath(X,Z)`

[5] Throughout the text we assume that the implicit set of variables $\boldsymbol{V}$ has at least five members. This assumption can easily be lifted, albeit with a more complex notation to follow.

[6] An arc is strongly protected in a graph $\mathcal{G}$ if it occurs in one of four specific sub-graphs of $\mathcal{G}$. See [10] for details.

16. $\{\texttt{UndirectedPath\{X,Y\}}, \texttt{DirectedPath(Y,Z)}\} \triangleright^g \neg\texttt{DirectedPath(X,Z)}$
17. $\{\texttt{UndirectedPath\{X,Y\}}, \texttt{UndirectedPath\{Y,Z\}}\} \triangleright^g \neg\texttt{UndirectedPath\{X,Z\}}$
18. $\{\texttt{UndirectedPath\{X,Y\}Excluding(Z,W)}, \texttt{UndirectedPath\{Y,U\}Excluding(Z,W)}\} \triangleright^g \neg\texttt{UndirectedPath\{X,U\}Excluding(Z,W)}$
19. $\texttt{DirectedPath(X,Y)} \triangleright^g \texttt{Arc(Y,X)}$
20. $\texttt{DirectedPath(X,Y)} \triangleright^g \texttt{Link\{X,Y\}}$
21. $\texttt{UndirectedPath\{X,Y\}} \triangleright^g \texttt{Arc(X,Y)}$
22. $\{\texttt{UndirectedPath\{X,Y\}Excluding(W,Z)}, \texttt{Link\{X,W\}}, \texttt{Link\{Y,Z\}}, \texttt{NonAdjacent\{X,Z\}}, \texttt{NonAdjacent\{Y,W\}}\} \triangleright^g \texttt{Link\{W,Z\}}$
23. $\{\texttt{Arc(X,Y)}, \texttt{NonAdjacent\{X,Z\}}\} \triangleright^g \texttt{Link\{Y,Z\}}$
24. $\texttt{ArcNotAllowed(X,Y)} \triangleright^g \texttt{Arc(X,Y)}$
25. $\{\texttt{Arc(Z,X)}, \texttt{NonAdjacent\{Z,Y\}}\} \triangleright^g \texttt{ArcNotAllowed(X,Y)}$
26. $\{\texttt{Arc(Z,Y)}, \texttt{NonAdjacent\{Z,X\}}\} \triangleright^g \texttt{ArcNotAllowed(X,Y)}$
27. $\{\texttt{Arc(X,Z)}, \texttt{Arc(Z,Y)}\} \triangleright^g \texttt{ArcNotAllowed(X,Y)}$
28. $\{\texttt{Link\{X,Z\}}, \texttt{Arc(Z,Y)}, \texttt{Link\{X,W\}}, \texttt{Arc(W,Y)}, \texttt{NonAdjacent\{Z,W\}}\} \triangleright^g \texttt{ArcNotAllowed(X,Y)}$

for all choices of distinct variables X, Y, Z, W, and U where the sentences obtained are in $\boldsymbol{L}$.

Loosely speaking, if $\boldsymbol{\Sigma}$ is a preferred extension of $\mathcal{A}^g$, then Bullets 1–7 ensure that $\boldsymbol{\Sigma}^g$ is a CGK; Bullets 8–18 make sure that the path sentences in $\boldsymbol{\Sigma} \setminus \boldsymbol{\Sigma}^g$ are correct wrt. $\mathcal{G}[\boldsymbol{\Sigma}^g]$; Bullets 19–28 ensure that $\mathcal{G}[\boldsymbol{\Sigma}^g]$ is a complete pattern, cf. [10]. More precisely we have:

Lemma 1. *Let $\boldsymbol{\Sigma}$ be conflict free wrt. $\mathcal{A}^g$. Then $\boldsymbol{\Sigma}^g$ is a CGK.*

Theorem 1. *Let $\boldsymbol{\Sigma}$ be a preferred extension of $\mathcal{A}^g$. Then $\mathcal{G}[\boldsymbol{\Sigma}^g]$ is in $\mathfrak{C}$.*

Theorem 2. *If $\mathcal{G}$ is in $\mathfrak{C}$, then there is a stable extension $\boldsymbol{\Sigma}$ of $\mathcal{A}^g$, such that $\mathcal{G}[\boldsymbol{\Sigma}^g] = \mathcal{G}$.*

These results are important since they guarantee that agents arguing under the restrictions specified by $\mathcal{A}^g$ can be sure that their result is a completed pattern and that they are not restricted from agreeing on any model a priori by the relations of $\mathcal{A}^g$. However, checking whether a set of arguments constitute a preferred extension is complex. It involves checks for both admissibility and maximality. We therefore state a result that yields a computationally efficient way of testing whether an admissible set of arguments of $\mathcal{A}^g$ is a preferred extension.

Theorem 3. *Let $\boldsymbol{\Sigma}$ be a preferred extension of $\mathcal{A}^g$. Then $\boldsymbol{\Sigma}$ is a stable extension.*

For proofs of all results and further elaborations, see [8].

5. Fusing Agoras

We now address the problem of having agents agree on a preferred extension of $\mathcal{A}^g$, given that each of them has its own prior beliefs, as expressed by the compromise score function s_i, and that each know the combination function c. There has not been a lot of work done in dialectics for more than two agents, where the simple proponent/opponent dualism does not suffice. The solution that we propose here is inspired by the Risk Ago-

ras of [17] and [18] and the traditional blackboard architecture of MAS of cooperating agents, without being an actual instantiation of any of them. We construct a *fusing agora*, which is a framework in which the agents can debate. The agora has the property that, if agents are allowed to run the debate to conclusion, they end up with the best possible compromise according to their joint compromise score, and that throughout the debate they maintain a compromise, which improves as the debate progresses.

In the agora we shall take a $\mathcal{A}^g$-candidate $(\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$ as a unique representatives of a partial compromise $(\boldsymbol{I}, \boldsymbol{O})$. This is possible since $\boldsymbol{I}$ and $\boldsymbol{O}$ are subsets of $\boldsymbol{L}$, and thus both contain sentences describing aspects of a graph as required, and furthermore, $\boldsymbol{U}$ is by definition determined by $\boldsymbol{I}$ and $\boldsymbol{O}$. Any leaf candidate representing a preferred extension, then uniquely identifies a completed pattern, as guaranteed by Theorem 1. Agents can explore all compromises by examining a $(\varnothing, \varnothing, \boldsymbol{L})$-tree. Continually the agents take it upon themselves to explore sub-trees of this tree, and mark other sub-trees as open for investigation by other agents. The heuristics guiding the agents choices for exploration, in addition to $s_1, \ldots, s_k$ and c, then determine the outcome.

The agora can work in a variety of ways, depending on the behavior of the individual agents (a vanilla algorithm for an individual agent is provided later in Algorithm 1), but basically builds on two elements, which we assume each agent can access in a synchronized fashion only: A *pool of candidates* $\boldsymbol{C}$ and a *current best result* $\langle \boldsymbol{I}_*, s_{\boldsymbol{I}_*} \rangle$. $\boldsymbol{C}$ consists of pairs $\langle \mathcal{C}, s \rangle$, where $\mathcal{C}$ is an $\mathcal{A}^g$-candidate and thus a sub-tree of a $(\varnothing, \varnothing, \boldsymbol{L})$-tree, and s is a real value. $\boldsymbol{I}_*$ is either the empty set or a preferred extension of $\mathcal{A}^g$, and $s_{\boldsymbol{I}_*}$ is a real value. Initially, $\boldsymbol{C}$ contains only one element $\langle (\varnothing, \varnothing, \boldsymbol{L}), 0 \rangle$, and $\langle \boldsymbol{I}_*, s_{\boldsymbol{I}_*} \rangle$ is $\langle \varnothing, -\infty \rangle$.

Each agent i can utter the following locutions:

- *ExploreFromPool*$_i(\langle \mathcal{C}, s \rangle)$ — where $\langle \mathcal{C}, s \rangle$ is a member of $\boldsymbol{C}$. The meaning of the locution is that agent i takes upon itself the responsibility to investigate the preferred extensions in a $\mathcal{C}$-tree, assuming that $\mathcal{C}$ has a joint compromise score of s.
- *PutInPool*$_i(\langle \mathcal{C}, s \rangle)$ — where $\mathcal{C}$ is an $\mathcal{A}^g$-candidate, and s is a real value. The meaning of the locution is that agent i wants someone else to investigate the preferred extensions in a $\mathcal{C}$-tree, and that $\mathcal{C}$ has a joint compromise score of s.
- *UpdateBest*$_i(\langle \boldsymbol{I}, s \rangle)$ — where $\boldsymbol{I}$ is a subset of $\boldsymbol{L}$, and s is a real value. The meaning of the locution is that agent i has identified a preferred extension $\boldsymbol{I}$ with a joint compromise score s higher than $s_{\boldsymbol{I}_*}$.
- *AskOpinion*$_i(\mathcal{C}_1, \mathcal{C}_2)$ — where $\mathcal{C}_1$ and $\mathcal{C}_2$ are $\mathcal{A}^g$-candidates. The meaning of the locution is that agent i needs to know $s_j(\mathcal{C}_1, \mathcal{C}_2)$ for all other agents j.
- *StateOpinion*$_i(\mathcal{C}_1, \mathcal{C}_2, s_\delta)$ — where $\mathcal{C}_1$ and $\mathcal{C}_2$ are $\mathcal{A}^g$-candidates, and s_δ is a real value. The meaning of the locution is that $s_i(\mathcal{C}_1, \mathcal{C}_2)$ is s_δ.

The rules governing which locutions individual agents can utter, as well as their effects, we present as a set of pre and post conditions:

- *ExploreFromPool*$_i(\langle \mathcal{C}, s \rangle)$
  - Pre: $\langle \mathcal{C}, s \rangle$ is in $\boldsymbol{C}$.
  - Post: $\langle \mathcal{C}, s \rangle$ is removed from $\boldsymbol{C}$
- *PutInPool*$_i(\langle \mathcal{C}, s \rangle)$
  - Pre: There is no $\langle \mathcal{C}', s' \rangle$ in $\boldsymbol{C}$ such that $\mathcal{C}$ is a sub-tree of some $\mathcal{C}'$-tree.
  - Post: $\langle \mathcal{C}, s \rangle$ is in $\boldsymbol{C}$.
- *UpdateBest*$_i(\langle \boldsymbol{I}, s \rangle)$

* Pre: $s > s_{\boldsymbol{I}_*}$.
* Post: $\langle \boldsymbol{I}_*, s_{\boldsymbol{I}_*} \rangle$ is set to $\langle \boldsymbol{I}, s \rangle$.

Locutions *AskOpinion*$_i$() and *StateOpinion*$_i$() have no pre or post conditions attached.

Algorithm 1 *Vanilla algorithm for agent* i

1: $\langle \mathcal{C}, s \rangle \leftarrow$SELECTCANDIDATE($\boldsymbol{C}$)
2: *ExploreFromPool*$_i(\langle \mathcal{C}, s \rangle)$
3: $\mathcal{C}' \triangleq (\boldsymbol{I}', \boldsymbol{O}', \boldsymbol{U}') \leftarrow$PRUNE($\mathcal{C}$)
4: **if** $\boldsymbol{U}' = \varnothing$ **then**
5: **if** PREFERREDEXTENSION($\boldsymbol{I}'$) **then**
6: *AskOpinion*$_i(\mathcal{C}, \mathcal{C}')$
7: $s_i \leftarrow s_i(\mathcal{C}, \mathcal{C}')$
8: **wait for** *StateOpinion*$_j(\mathcal{C}, \mathcal{C}', s_j) \forall j \neq i$
9: $s' \leftarrow c(s_1, \ldots, s_k) + s$
10: **if** $s' > s_{\boldsymbol{I}_*}$ **then**
11: *UpdateBest*$_i(\langle \mathcal{C}', s' \rangle)$
12: **go to** 1
13: **else**
14: $A \leftarrow$SELECTARGUMENT($\mathcal{C}'$)
15: *AskOpinion*$_i(\mathcal{C}, \mathcal{C}' + A)$
16: *AskOpinion*$_i(\mathcal{C}, \mathcal{C}' - A)$
17: $s_i^+ \leftarrow s_i(\mathcal{C}, \mathcal{C}' + A)$
18: $s_i^- \leftarrow s_i(\mathcal{C}, \mathcal{C}' - A)$
19: **wait for** *StateOpinion*$_j(\mathcal{C}, \mathcal{C}' + A, s_j^+)$ **and** *StateOpinion*$_j(\mathcal{C}, \mathcal{C}' - A, s_j^-) \forall j \neq i$
20: $s^+ \leftarrow c(s_1^+, \ldots, s_k^+)$
21: $s^- \leftarrow c(s_1^-, \ldots, s_k^-)$
22: **if** $s^+ > s^-$ **then**
23: *PutInPool*$_i(\langle \mathcal{C}' - A, s + s^- \rangle)$
24: $\mathcal{C} \leftarrow \mathcal{C}' + A$
25: $s \leftarrow s + s^+$
26: **else**
27: *PutInPool*$_i(\langle \mathcal{C}' + A, s + s^+ \rangle)$
28: $\mathcal{C} \leftarrow \mathcal{C}' - A$
29: $s \leftarrow s + s^-$
30: **go to** 3

The basic algorithm in Algorithm 1 corresponds to an exhaustive search, if it is followed by all agents. The search is gradual in two senses: The longer the search goes on, the average candidate in $\boldsymbol{C}$ will have more elements in its $\boldsymbol{I}$ and $\boldsymbol{O}$ sets, and thus be closer to describing a full compromise, and the current compromise held in $\boldsymbol{I}_*$ will have an increasingly higher score. Of course, in order for the search to be a success, each agent i would also need to keep an eye out for *AskOpinion*$_j(\cdot)$'s uttered by other agents, and reply to these with *StateOpinion*$_i(\cdot)$. It is relatively easy to verify that agents using Algorithm 1 are uttering locutions in accordance with the pre and post conditions of the fusing agora.

Algorithm 1 calls a number of functions, which we only describe informally: PRUNE($\mathcal{C} \equiv (\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$) uses pruning rules to investigate whether there is an argument A in $\boldsymbol{U}$ such that either $\mathcal{C} + A$ or $\mathcal{C} - A$ contains no leaves with preferred extensions. If this is the case, the method invokes itself recursively on the sub-tree that did not get

pruned away, until no further branches can be pruned. Some general pruning rules are given in [15], and more can be established for the specific case of $\mathcal{A}^g$.

SELECTCANDIDATE($\boldsymbol{C}$) picks a promising candidate from $\boldsymbol{C}$. A promising candidate could be one with a high score annotated, since these encode good partial compromises, or candidates with small $\boldsymbol{U}$ sets, as these represent partial compromises that are nearly complete. If all agents use the same criteria for picking promising candidates, this selection can be sped up by implementing the pool as a sorted list. SELECTCANDIDATE($\cdot$) is one of the areas where heuristics limiting the search space can be implemented. For instance, it makes sense to allow agents to abstain from exploring the sub-tree rooted at a candidate if it cannot contain compromises that are consistent with their own BN. This would mean that in cases where agents agree on all or most of the aspects of $\mathcal{G}_*$ only few candidates would need to be explored.

PREFERREDEXTENSION($\boldsymbol{I}$) is a Boolean valued function that returns true if the conflict-free set $\boldsymbol{I}$ is a preferred extension of $\mathcal{A}^g$. The task of answering this is simplified by Theorem 3, as it states that $\boldsymbol{I}$ is a preferred extension iff $\boldsymbol{I}$ attacks each argument in $\boldsymbol{L} \setminus \boldsymbol{I}$.

SELECTARGUMENT($\mathcal{C} \equiv (\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$) simply selects an element A of $\boldsymbol{U}$. This selection can be based on the agent's own score increase going from $\mathcal{C}$ to $\mathcal{C} + A$ or $\mathcal{C} - A$, or it might involve negotiations or argumentation with other agents.

Of course, the debate in the agora can be stopped at any time, and $\mathcal{G}[\boldsymbol{I}^g_*]$ will then be the best compromise encountered so far, as it is only ever replaced by compromises having a higher joint compromise score.

It is worth stressing that Algorithm 1 is a vanilla algorithm, and that the agora is open for more aggressive behaviour. One such behaviour could be to have agents skip the asking for opinions part in Lines 14 to 22 for most additions of arguments (and basing the decision only on the agents own beliefs), and only ask when the agent itself is indifferent. Another behaviour could be to never perform Lines 23 and 27, which would correspond to a myopic greedy construction of the compromise. Alternatively, these two lines could be carried out only when the difference between s^+ and s^- is very small. We could even have setups where the agents show different behaviours, or where individual agents change behaviour during debate depending on their available resources and utility of a good compromise. Moreover, the agora does not require that agents wait for a candidate to be in the pool, before somebody can start exploring this candidate; so even when one agent is pursuing an aggressive strategy and fails to leave candidates for others to explore, other agents can still decide to explore these. The point is, that no matter what behaviour is requested, the basics of the agora and the agents remains the same.

6. Conclusion

We have introduced a problem which we believe is a challenging one for the argumentation community, due to its mix of complexity and conditional decomposability as well as its origin in conflicting knowledge bases. Our own solution enables agents to judge the possible compromises resulting from a partial compromise, by constructing a candidate tree rooted in this partial compromise, and the agora we have proposed ensures that such exploration can take place in a distributed fashion. One problem with the vanilla algorithm we have given, is that agents exploring a branch of a candidate-tree can end up putting a lot of candidates into the pool of candidates. The space requirements for

storing the pool of candidates can be prohibitive, so it might be required that the candidates in the pool are defined in relation to each other, which imposes restrictions on which candidates an agent can choose to explore, as these are removed from the pool. Furthermore, it might be necessary to construct heuristics for thinning the pool of candidates. These issues, as well as finding good heuristics for selecting candidates to explore are challenging topics for future research.

References

[1] S. Saha and S. Sen. A Bayes net approach to argumentation based negotiation. In *ArgMAS 1*, pages 208–222. Springer, 2004.

[2] G. Vreeswijk. Bayesian inference and defeasible reasoning: suggestions for a unified framework. Working paper?, 1999.

[3] G. Vreeswijk. Argumentation in Bayesian belief networks. In *ArgMAS 1*, Lecture Notes in Artificial Intelligence, pages 111–129. Springer, 2004.

[4] I. Matzkevich and B. Abramson. The topological fusion of bayes nets. In *UAI 8*, pages 191–198. Morgan Kaufmann, 1992.

[5] J. Del Sagrado and S. Moral. Qualitative combination of bayesian networks. *International Journal of Intelligent Systems*, 18(2):237–249, 2003.

[6] M. R. and P. Domingos. Learning with knowledge from multiple experts. In *ICML 20*, pages 624–631. AAAI Press, 2003.

[7] P. M-R. II and U. Chajewska. Aggregating learned probabilistic beliefs. In *UAI 17*, pages 354–361. Morgan Kaufmann, 2001.

[8] S. H. Nielsen and S. Parsons. Fusing Bayesian networks using formal argumentation in multi-agent systems. Technical report, Aalborg University, 2006. http://www.cs.auc.dk/~holbech/-tr0106NielsenParsons06.ps.

[9] T. Verma and J. Pearl. Equivalence and synthesis of causal models. In *UAI 6*, pages 220–227. Elsevier Science Publishing Company, Inc., 1990.

[10] S. A. Andersson, D. Madigan, and M. D. Perlman. A characterization of Markov equivalence classes for acyclic digraphs. *Annals of Statistics*, 25(2):505–541, 1997.

[11] H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*. Kluwer Academic Publishers, 2000.

[12] S. H. Nielsen and S. Parsons. A generalization of Dung's abstract framework for argumentation. In *Proceedings of Third Workshop on Argumentation in Multi-agent Systems*. Springer, 2006.

[13] 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):321–358, 1995.

[14] Y. Dimopoulos, B. Nebel, and F. Toni. Preferred arguments are harder to compute than stable extension. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 36–43. Morgan Kaufmann, 1999.

[15] S. H. Nielsen and S. Parsons. Computing preferred extensions for argumentation systems with sets of attacking arguments. In *COMMA06 Proceedings*. 2006.

[16] S. Doutre and J. Mengin. Preferred extensions of argumentation frameworks: Query, answering and computation. In *Proceedings of International Joint Conference on Automated Reasoning*, volume 2083, pages 272–288. Springer, 2001.

[17] P. McBurney and S. Parsons. Risk agoras: Dialectical argumentation for scientific reasoning. In *UAI 16*. Stanford, 2000.

[18] P. McBurney and S. Parsons. Chance discovery using dialectical argumentation. In *Workshop on Chance Discovery, JSAI 15*. Matsue, 2001.

Applications

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Value-Based Argumentation for Democratic Decision Support

Katie Atkinson*
Department of Computer Science, University of Liverpool

Abstract. In this paper we discuss the integration of two systems that are based on a specific theory of argumentation: the first, an existing web-based discussion forum; the second, a method to enable autonomous software agents to perform practical reasoning based upon their subscription to social values. We show how the output from the first of these systems can be used as input to the second and how the information gathered can be reasoned about through computer support. The purpose of the approach is to demonstrate how current theories of argumentation can be used to assist with the analysis of public attitude in a particular debate, with the specific example domain used being that of eDemocracy. We also provide some discussion and comparison of these current tools with similar, earlier systems.

Keywords. eDemocracy, practical reasoning, argumentation frameworks

1. Introduction

This paper discusses the application of theories of argumentation to the domain of eDemocracy. The emergence of web technologies has led to the computerisation of numerous 'traditional' business processes in the public, as well as the private, sector. The ability of the public to interact with their rulers through online provisions has led to the emergence of a new method of governance: eDemocracy. The transformation of democracy into an electronic medium is currently making great advances, even though the field is still relatively young. Numerous countries are engaged in the trial and development of new interactive systems for eDemocracy, such as those for e-voting [1] and proposals for new systems for eGovernment are attempting to address major issues such as trust and security e.g., [2,3]. Thus, with the introduction of safe and efficient web-based services governments have the opportunity to exploit the benefits of new computer technologies to provide accessible, efficient and useful systems through which democracy can be effectively conducted. As debate and policy justification are key elements of eDemocracy, support for systems promoting such interactions can be enlisted through the implementation of theories of argumentation to underpin these systems. The work presented in this paper aims to address some of these objectives. The paper explores how a specific computer system implemented to facilitate eDemocracy can be integrated with autonomous agent systems used to reason about the justification of arguments concerning actions. In

*Correspondence to: Department of Computer Science, University of Liverpool, L69 3BX, UK. Tel.: +44 (0)151 795 4243; Fax: +44 (0)151 795 4235; E-mail: katie@csc.liv.ac.uk

section 2 we give an overview of a tool, named the PARMENIDES system, developed to foster public debate on a particular political issue. In section 3 we briefly describe an approach to argument representation for dealing with reasoning about action, which can be deployed in autonomous software systems. In section 4 we describe how a link can be established between the systems described in the previous two sections. We then illustrate this approach with a short example. In section 5 we discuss how the approach presented in this paper compares and contrasts with earlier systems of similar ambition. Finally, in section 6 we offer some concluding remarks.

2. The PARMENIDES Discussion Forum

In recent years numerous computer systems have been developed which aim to facilitate the online conveyance of democracy, e.g., Zeno [4] and DEMOS [5]. This paper focuses on one particular system – the PARMENIDES system developed by Atkinson *et al.* [6] – designed to encourage public participation and debate regarding the Government's justifications for proposed actions. The PARMENIDES (Persuasive ARguMENt In DEmocracieS) system is described in [6] and the system can be used at: `http://www.csc.liv.ac.uk/~katie/Parmenides.html`.

The idea of the system is to enable members of the public to submit their opinions about the Government's justification of a particular action. In the prototypical version the subject dealt with is the 2003 war in Iraq, with the particular question under scrutiny being, "Is invasion of Iraq justified?" (as this concerns a past action, the example debate used is for illustrative purposes only). One of the key features of PARMENIDES is the underlying model upon which it is based, as the tool is intended as an implementation to exploit a specific representation of persuasive argument. The background of this model of argument is as follows.

Atkinson *et al.* have previously described an argument scheme and critical questions that provide an account of persuasive argument in practical reasoning [7]. Their argument scheme is an extension to Walton's *sufficient condition scheme for practical reasoning* [8], and follows his method of treating such schemes as presumptive justification. His account views an argument scheme as embodying a presumption in favour of the conclusion, where presumptions are tested by posing 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. Atkinson *et al's* extended scheme, called AS1 and given below, makes Walton's notion of a goal more explicit:

AS1 In the current circumstances R,
we should perform action A,
to achieve new circumstances S,
which will realise some goal G,
which will promote some value V.

In this scheme the notion of a goal has been separated into three distinct elements: states of affairs (the effects of actions), goals (the desired features in those states of affairs) and values (the reasons why those features are desirable). Thus, values provide subjective reasons as to why states of affairs are desirable or undesirable. Additionally, values relate states of affairs, since a given state of affairs may be desirable through promoting several values, and a given value can be promoted by several states of affairs.

Instantiations of argument scheme AS1 provide *prima facie* justifications of proposals for action. Associated with this scheme are sixteen different critical questions that challenge the presumptions in instantiations of AS1. These critical questions are:

CQ1: Are the believed circumstances true?
CQ2: Assuming the circumstances, does the action have the stated consequences?
CQ3: Assuming the circumstances and that the action has the stated consequences, will the action bring about the desired goal?
CQ4: Does the goal realise the value stated?
CQ5: Are there alternative ways of realising the same consequences?
CQ6: Are there alternative ways of realising the same goal?
CQ7: Are there alternative ways of promoting the same value?
CQ8: Does doing the action have a side effect which demotes the value?
CQ9: Does doing the action have a side effect which demotes some other value?
CQ10: Does doing the action promote some other value?
CQ11: Does doing the action preclude some other action which would promote some other value?
CQ12: Are the circumstances as described possible?
CQ13: Is the action possible?
CQ14: Are the consequences as described possible?
CQ15: Can the desired goal be realised?
CQ16: Is the value indeed a legitimate value?

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. In the prototypical version a justification upholding the action of invading Iraq is presented to users of the system in the form of argument scheme AS1. Users are then led in a structured fashion through as series of web pages that pose the appropriate critical questions to determine which parts of the justification the users agree or disagree with. Once a critique has been given regarding the initial justification for action, users are then given the opportunity to state their own full justification of any action they believe should be proposed, regarding the topic in question. Users of the system are not aware (and have no need to be aware) of the underlying structure for argument representation but nevertheless, this structure is imposed on the information they submit. This enables the collection of information which has been structured in a clear and unambiguous fashion from a system which does not require users to gain specialist knowledge before being able to use the tool. All responses given by users are written to a back-end database so that information as to which points of the argument are more strongly supported than others can be gathered. The original proponent of the action, i.e., the Government, can then analyse the information gathered to review public support of its case and perhaps revise or change its justification to make the policy more amenable to public support.

This brief description of the PARMENIDES system is intended as an overview of the tool and it is described in more detail in [6]. We now briefly describe how the argument scheme and critical questions discussed in this section can be employed in an alternative application making use of autonomous software agents.

3. Reasoning About Action Using Autonomous Agents

In [9] Atkinson *et al.* describe how their argument scheme and critical questions can be transformed into a computational account for use in software systems consisting of autonomous agents based upon the popular belief-desire-intention (BDI) architecture. They provide formal definitions that specify pre-conditions for when an agent can construct a position based upon its beliefs, the actions available for performance, the agent's desires, and its values. As standard BDI architectures do not incorporate values, in [9] an account has been given that extends the architecture to include values, which provide justifications for the agent's choice of intentions, based upon its beliefs and desires. A full set of pre-conditions is specified, which when satisfied allow agents to attack a justification for action by posing any of the critical questions against the position. The output of this process is a set of presumptive arguments[1] plus attacks on them. Resolution of a chosen course of action is then done by organising the arguments and attacks into Value-Based Argumentation Frameworks (VAFs) [10], which provide an extension to Dung's Argumentation Frameworks (AFs) [11] to accommodate different audiences with different values and interests. Within a VAF, which arguments are accepted depends on the ranking that the audience (characterised by a particular preference ordering on the values) to which they are addressed gives to the purposes motivating the argument. As in Dung's AFs, the key elements in a VAF are the preferred extensions (PEs), which provide the maximal consistent set of conflict-free arguments, relative to a particular audience [10].

To demonstrate this approach Atkinson *et al.* have provided an example application in [9]. In this example they provide a reconstruction of the arguments involved in a well known legal case from property law and they show how BDI agents can reason about the justified course of action, in accordance with the above method. In the next section we show how a link can be provided between PARMENIDES and the method described above. This link is intended to show how computer support based on argumentation can be used to aid the democratic debating process, whilst accounting for differing opinions.

4. Integrating the Approaches

Given that the two systems described above are based upon the same model of argument, there is an obvious link that can be exploited between them, as we will demonstrate now.

The purpose of the PARMENIDES system is to gather public opinions regarding the justification of proposed government actions. This could potentially mean that large amounts of data are received and stored by the system and it would be useful to have a mechanism to analyse and reason about the data. The use of software agents can serve this purpose.

The database that records the information submitted through PARMENIDES stores all the critiques and counter proposals supplied by members of the public. Any such counter proposal offering a justification for action is decomposed and stored as individual entries that record each of the elements of the justification that comprise an instantiation of argument scheme AS1, i.e., the circumstances believed to be true, the action proposed given these circumstances, the consequences of performing the action that include the

[1] It is assumed in [9] that these arguments will be represented in some suitable formal logic, such as propositional logic, amenable to reasoning by a software agent.

goal of the action, and, the value promoted through achievement of this goal. Given this information, it is then possible to determine different audiences, based upon their value subscriptions, and thus ascertain the acceptability and popularity of each action suggested. An example to illustrate this is given below.

4.1. Example

On entering the PARMENIDES system, the user is presented with the Government's (hypothetical) justification for invading Iraq. This justification is as follows:

- In the current situation: Saddam has weapons of mass destruction (WMD), Saddam will not disarm voluntarily, Saddam is running an oppressive regime, Saddam is defying the UN, Saddam is a threat to his neighbours.
- The action we should take is: invade Iraq.
- Invading Iraq will: Remove the WMD, Restore democracy to Iraq, Assert the authority of the UN, Remove the threat Saddam poses to his neighbours, Cause military casualties, Cause civilian casualties.
- This will achieve: Removing WMD will promote world security, Restoring democracy will promote human rights.

As two values are involved in this justification we can split the argument into two separate justifications: one based on the presence of WMD whereby the action of invading will get rid of the WMD, promoting the value 'world security', and, one based on the existence of an oppressive regime whereby invading will dispel the regime, promoting the value 'human rights'. We shall call these two arguments Arg1 and Arg2 respectively. We are then able to take the individual elements of each justification and instantiate the beliefs, desires, goals and values of a value-enhanced BDI agent (in accordance with the formal pre-conditions described in [9]) to represent the views expressed in these justifications[2]. For convenience we will use a separate BDI agent to represent each perspective in this example. Now, returning to the PARMENIDES system, suppose that a particular user disagrees with the justification given in Arg1. Such a disagreement would be revealed in the summary of the responses of the user's critique that is displayed when he has been questioned about his views regarding each element of the justification. A textual excerpt from such a summary, showing the user's opinion of the circumstances, as displayed by PARMENIDES is given below:

- You *disagree* that Saddam has WMD,
- You *disagree* that Saddam will not disarm involuntarily,
- You *agree* that Saddam is running an oppressive regime,
- You *disagree* that Saddam is defying the UN
- You *agree* that Saddam is a threat to his neighbours.

As an example, if we examine the first item on the list we can see that this particular user believes that Saddam does not possess WMD, i.e., the user disagrees with the description of the current situation. In critiquing this element the user (without knowing or needing to know) is posing critical question CQ1 and we shall call this attack on the

[2]Due to space restrictions we assume that the pre-conditions for instantiating the agents are met and we do not provide specific details of this here. Detailed examples of how agents are instantiated can be found in [9].

justification 'Attack1'[3]. In order to construct the appropriate VAF to represent this attack, we need to identify the value endorsed by this argument. As the attack represents a disagreement as to what the facts of the situation are, we associate this argument with the value 'opinion'. Thus, the critical question posed by this attack isolates the contentious element of the justification that requires clarification, whilst recognising that this is relative to an *opinion* of a particular opponent. This value would initially be ranked as a weak argument against the justification, but it could gain strength were more users shown to have the same opinion. If a general consensus emerged that a particular element was seen as unjustifiable in the opinion of the public, then the Government may be persuaded to act upon the perceived disputable point, e.g., by clarifying the facts or altering the policy.

We can now instantiate a BDI agent that holds the belief manifest in the attack described above, i.e., that there are no WMD. If we now view the arguments considered so far as a VAF, we have the situation shown in Figure 1:

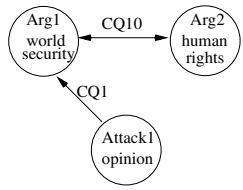


Figure 1. VAF with an attack on a justification.

The VAF in Figure 1 has nodes to represent the two arguments promoting different values and it also shows the attack on Arg1 posed by CQ1 (and note that the other critiques given in the list could also be treated in the same manner). Note also that the VAF shows an attack between Arg1 and Arg2 through the use of CQ10 to point out that although both arguments endorse the same action, they each promote different values. This distinction places importance upon the *justification* that each argument offers in support of the action. The reason these arguments are then seen to attack each other is that each sees the other's justification as being less acceptable. The importance of this point can be seen through the criticism levelled at the British Government who ostensibly invaded Iraq to remove WMD, while critics argued that their motives were actually regime change, illegal under international law. Thus, in addition to deciding which action to execute, deciding upon the most acceptable justification for an action is also an important part of practical reasoning. This point is also demonstrated in other domains, such as the legal one where the justification of actions has consequences for making future judgments.

Returning now to the example, suppose the PARMENIDES user has critiqued the original justification and has also gone on to offer an alternative action plus justification. Such an alternative, which we will call Arg3, might be recorded as follows:

- In the current situation: we believe Saddam may have WMD,
- The action we should take is: give weapons inspectors more time to investigate,
- This will: clarify whether the WMD claim is true or not,
- This will achieve: public trust in the facts.

[3]Note that in all the VAFs presented here 'Arg' is used to denote instantiations of AS1 (that may or may not arise through posing critical questions) and this is distinguished from 'Attack' which is used to denote an argument that poses a critical question but does not instantiate AS1. Thus attacks are solely negative, whereas arguments also propose an action.

The PARMENIDES database would record each of the following: the facts about the situation the user believes to be the case, the action proposed given these facts, the consequences of these facts, and the reasons (values) as to why these consequences are desirable. Such a justification is offering an alternative action, incompatible with the original action, which promotes some other value and thus it is posing critical question CQ11. We are then able to instantiate another BDI agent with the beliefs, desires and value cited in the above justification for action. The VAF showing the addition of this argument is given in Figure 2:

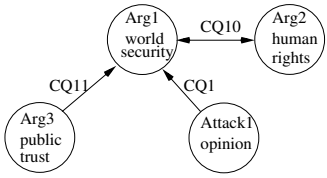


Figure 2. VAF with argument promoting a different value.

In the above VAF the attack of CQ11 on Arg1 would succeed for any audience that ranks the value 'public trust' higher than the value 'world security'. Attack1 would only succeed in defeating Arg1 if it were shown to be an opinion expressed by a sufficiently large number of users, according to a set threshold. As yet, Arg2 has no further attackers so the action of invasion, for the reasons specified in Arg2, could still be justifiable. However, submissions to PARMENIDES may reveal some critiques and counter proposals for the justification of Arg2. Again, consider a sample summary of a user's critique from the PARMENIDES database, this time concerning the consequences of the action:

You believe that invading Iraq will achieve the following:

- Remove the WMD: *Yes*,
- Restore democracy in Iraq: *No*,
- Assert the authority of the UN: *No*,
- Remove the threat that Saddam poses to his neighbours: *Yes*.
- Cause military casualties: *Yes*,
- Cause civilian casualties: *Yes*.

If we examine the second item on the list we can see that this particular user believes that invading Iraq will not restore democracy to the country, i.e., he disagrees with the consequences of the action. This critique poses critical question CQ2 and we shall call this attack on the justification 'Attack2'. As in the case of Attack1, Attack2 will also take the value 'opinion'. So, we can instantiate another BDI agent that holds this belief.

After having given his critique, a user with such views may also propose an alternative position on the matter, such as the example one given below:

- In the current situation: Saddam is running an oppressive regime, Saddam is violating human rights,
- The action we should take is: wait for a second UN resolution on the matter,
- This will mean: unjustified military intervention is not required,
- This will achieve: respect for international law.

As this justification is offering an alternative action, incompatible with the original action, which promotes some other value, it is again posing critical question CQ11. We

will call this Arg4 and we can use another BDI agent to represent this view. Both Attack2 and Arg4 can now be added to the VAF, as shown in Figure 3.

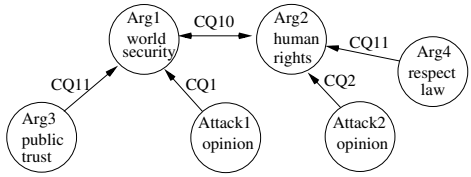


Figure 3. VAF with attacks on Arg2.

In the above scenario the attack of CQ11 on Arg2 would succeed for any audience that ranks the value 'respect for international law' higher than the value 'human rights' (as used in relation to the specific argument). Attack2 would only succeed in defeating Arg2 if it were shown to be an opinion expressed by a sufficiently large number of users.

We have shown how both original justifications can be subject to attack through users' critiques, though there are of course further attacks that could be posed against the original justifications: the attacks discussed this far are intended to show a few examples of how the position can be critiqued. In order to give some structure and analysis to the data submitted to PARMENIDES, all critiques and alternative proposals would need to be represented as VAFs. This would enable the Government to uncover any patterns in the data showing which parts of the justification are mostly frequently disagreed with, and segment the population according to their values. For example, critiques from multiple users may reveal that CQ1 is consistently being posed to disagree with the statement 'Saddam has WMD'. In such a case, the proponents of the original justification (the Government) may then try to clarify their reasons for endorsing this point, i.e., providing information on sources and their trustworthiness. Further critiques may reveal, for example, that CQ9 is consistently used to introduce arguments stating that other values have not been considered by the Government and these values are important to members of the public. In this case the Government would have to provide justification as to why the values they are endorsing are the most important ones concerned in the debate.

4.2. Reasoning About Public Opinions

The previous subsection described how the public's criticisms can be posed against the Government's position on the issue in question. However, to ensure that all opinions have been assessed in relation to each other, the reasoning process should not end here. In the same way that the original justification for invading Iraq was subject to critique, so the user-supplied arguments should also be subject to the same method of critical questioning. The PARMENIDES system currently does not provide a facility by which users can critically assess each other's views, though such an extension is desirable and would seem feasible to implement. However, it is currently possible to examine all views supplied and use the method described in the previous section to show how views between users may conflict. We now provide a short example of this.

If we examine Figure 3 we can see that it contains Arg3. This argument was constructed from an alternative position to the original justification, as supplied by a user. There are numerous ways in which this could be attacked. For example, the original proponent could counter that the goal of verifying whether Saddam has WMD could be met

through the alternative action of consulting an existing dossier profiling Iraq's WMD. This could instantiate AS1 with Arg5 as follows:

- In the current situation: we believe Saddam has WMD,
- The action we should take is: consult the previous dossiers produced by weapons inspectors on Iraq's WMD ,
- This will mean: the WMD claim is verified,
- This will: promote public trust in the facts.

This argument states that there is an alternative action that meets the same goal (verifying the WMD claim), and thus it makes use of critical question CQ6.

Looking to the attacks on Arg2 from Figure 3, we can see that Arg4 is one such attack. Again, this argument was constructed from a user-supplied alternative position and it too could be attacked in numerous ways. For example, the original proponent could counter that the alternative action proposed has side effects which actually demote the value concerned. This could instantiate AS1 with Arg6 as follows:

- In the current situation: Saddam is running an oppressive regime, Saddam is violating human rights,
- The action we should take is: wait for a second UN resolution on the matter,
- This will mean: Saddam is allowed to continue his activities,
- This will: demote respect for international law.

This argument states that the action proposed has unconsidered consequences which actually demote the value in question, ('respect for international law'), and thus it makes use of critical question CQ8. Additionally, there may be further arguments supplied by other users that also attack Arg4. For example, CQ9 could be used to state that the action has consequences which demote some other value, as in the following argument, Arg7:

- In the current situation: Saddam is running an oppressive regime, Saddam is violating human rights,
- The action we should take is: wait for a second UN resolution on the matter,
- This will mean: Saddam's enemies could be vulnerable to attack,
- This will: demote world security.

The above three arguments, Arg5, Arg6 and Arg7, can then be added to the VAF:

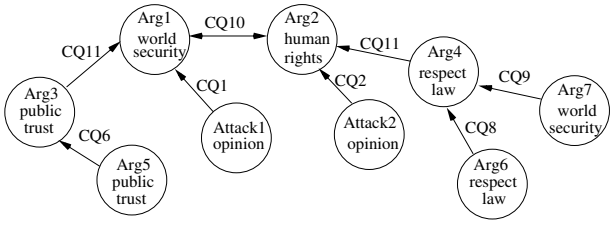


Figure 4. Final VAF.

If the reasoning were to stop here then we can see that for any audience Arg5 defeats Arg3, as the two are motivated by the same value[4]. Arg7 defeats Arg4 for any audience

[4]Following [10], where an argument attacks another argument with the same value in a VAF, the attacker always succeeds.

that ranks the value 'world security' higher than the value 'respect for international law'. If we then consider Arg6, this defeats Arg4 for any audience (again, as the two are motivated by the same value). Nonetheless, the loss of Arg4 does not mean that Arg2 will be reinstated as Arg2 could still be defeated by Attack2. Of course, the new arguments introduced to the VAF will themselves be subject to critique and were further arguments to be introduced to the debate through responses supplied to the PARMENIDES system, then the status of the VAF would need to be updated and re-evaluated accordingly.

The example arguments used here are intended to serve as an illustration of the approach, but in practice we envisage the debate encompassing a much larger range of arguments. Once a sufficiently representative number of views had been submitted to the PARMENIDES system, the Government would then be able to assess the opinions supplied and their relative importance. If the opinions revealed that particular parts of the original justification of the policy in question were viewed as being contentious, then the Government could take measures it deems appropriate to respond to public criticism. This may involve clarification of the facts, release of supporting information, or adjustment to the policy, amongst other things.

5. Related Work

Various mediation systems for deliberative debate have been proposed over the last two decades. We now briefly discuss how the work presented here relates to some similar systems. In particular, we examine the Zeno framework of Gordon and Karacapilidis [4].

Zeno, like PARMENIDES, is a "computer-based discussion forum with particular support for argumentation" [4]. The specific model of argumentation that Zeno is based upon is Rittel's Issue-Based Information System (IBIS) [12]. Zeno's main feature is a type of labelling function to represent arguments so that the relationship of positions regarding a solution to a practical issue can be assessed. From these arguments, a dialectical graph can be constructed showing the pros and cons of the choices available, in order to decide upon a solution to a practical issue. Users are able to express their preferences for particular choices and provide qualifications for these preferences. Zeno's dialectical representation graphs differ from VAFs in a number of ways. Firstly, VAFs solely encapsulate the notion of attack between arguments and as such, say nothing about the 'pros' of arguments. However, within a VAF, an attack on an attack could be construed as a type of supporting argument: if a particular argument is attacked, then a second attack made on the first may re-instate the original argument. In a Dung style AF, the notion of support is captured by considering the acceptability of an argument with reference to a *set* of arguments. An *admissible* set collectively supports all its members against all attacks from outside the set. Such defending arguments are not viewed as 'pros' within a VAF because they are only introduced into a VAF to provide rebuttals to attacks (if such defending arguments do not originally appear in the VAF). In effect, this method is prompting the audience to voice objections to the arguments presented, and any such objections will be included and evaluated as necessary, once identified. This means that only arguments relevant to the debate are included in the evaluation and arguments superfluous to it are avoided. An additional consequence of including supporting arguments in the debate is that they affect the evaluation of the acceptability of arguments. By requiring supporting arguments to be included in order to justify a position, arguments that are not attacked

cannot automatically be presumed acceptable, as they can in a VAF. Again, having this feature in a VAF means that relevant arguments are introduced to the debate only as and when necessary. However, other accounts that make use of supporting arguments have more recently been proposed, such as Amgoud *et al's* argumentation frameworks [13].

Examining now the 'value' component of VAFs, we believe that this provides extra information in the evaluation of the arguments that is not explicitly represented in Zeno. Zeno allows preferences between positions to be expressed, but these preferences are not justified in the subjective manner that is provided by the notion of *an audience* within a VAF. In Zeno's dialectical graphs, positions are regarded more like propositional statements that can be organised into a preference ordering according to the constraints defined in the debate. In VAFs however, such statements are distinguished into goal-value pairs where goal states map onto value(s) promoted by the goals. Thus, preference orderings over values are relative to particular audiences — they are not fixed constraints — and so they provide explanations as to why disagreements occur and what persuasion needs to take place in order for agreement to be reached.

In [14] the Zeno framework has also been compared against other decision support systems, such as McBurney and Parson's Risk Agora System [15]. This particular system was devised to model inquiry dialogues (and in particular, scientific inquiries), though the system is based on a different form of argumentation, namely, a dialogue game. As with most standard dialogue games, the framework specifies locutions that may be uttered by the participants (in accordance with specific pre-conditions), and it also tracks any commitments made by participants throughout the course of the dialogue. However, Risk Agora is not a fully implemented system, thus it does not provide real-time support for debates and it is intended as more of a tool to model the arguments in a debate and the relations between these arguments. Unlike PARMENIDES, it does not concern itself with justifying action through debate, as it is concerned more with inquiry dialogues.

Finally, returning to the eDemocracy domain, there are numerous approaches that have been developed in recent years that advocate the use of web-based discussion boards as a useful way of encouraging and supporting debate. Examples of such approaches can be found in [16,17]. Although such discussion boards can indeed encourage participation and debate, they generally provide no structure to the information gathered. The key advantage that the PARMENIDES system provides over such discussion boards is that it is implemented upon a firm model of argument, which is transparent to the user, but provides structure to the responses submitted. Additionally, the data submitted to PARMENIDES can be further analysed according to the techniques described in this paper. There are, of course, numerous other mediation systems that have been developed to provide support to decision making. However, consideration has been limited to the systems discussed here to illustrate the main merits of the approach presented in this paper, which combines a computational decision support system with current work on argumentation.

6. Concluding Remarks

In this paper we have shown how support can be given to systems for eDemocracy through the use of a current theory of argumentation concerning action. Our approach advocates a method integrating online public debate with current technologies based on autonomous software programs that are intended to provide computer support for rea-

soning about actions. We believe that both the systems described are of value in themselves as they are based upon a defined method of argument representation. Moreover, once integrated we believe that they have the potential to add further value to domains, such as the political one, where reasoning about and justifying actions is crucial.[5]

References

[1] E. Smith and A. Macintosh. E-Voting: powerful symbol of e-democracy. In R. Traunmüller, editor, *EGOV*, LNCS 2739, pages 240–245. Springer, 2003.

[2] J. Argyrakis, S. Gritzalis, and C. Kioulafas. Privacy enhancing technologies: A review. In R. Traunmüller, editor, *EGOV*, LNCS 2739, pages 282–287. Springer, 2003.

[3] A. Prosser, R. Kofler, and R. Krimmer. Deploying electronic democracy for public corporations. In R. Traunmüller, editor, *EGOV*, LNCS 2739, pages 234–239. Springer, 2003.

[4] T. F. Gordon and N. I. Karacapilidis. The Zeno argumentation framework. In *Proc. of the 6th International Conference on AI and Law*, pages 10–18. ACM Press, 1997.

[5] R. Lührs, S. Albrecht, M. Lübcke, and B. Hohberg. How to grow? Online consultation about growth in the city of Hamburg: methods, techniques, success factors. In R. Traunmüller, editor, *EGOV*, LNCS 2739, pages 79–84. Springer, 2003.

[6] K. Atkinson, T. Bench-Capon, and P. McBurney. PARMENIDES: Facilitating democratic debate. In R. Traunmüller, editor, *EGOV*, LNCS 3183, pages 313–316. Springer, 2004.

[7] K. Atkinson, T. Bench-Capon, and P. McBurney. Justifying practical reasoning. In F. Grasso, C. Reed, and G. Carenini, editors, *Proc. of the 4th workshop on Computational Models of Natural Argument*, pages 87–90, Valencia, Spain, 2004.

[8] D. N. Walton. *Argument Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.

[9] K. Atkinson, T. Bench-Capon, and P. McBurney. Arguing about cases as practical reasoning. In *Proc. of the 10th International Conference on AI and Law*, pages 35–44. ACM Press, 2005.

[10] T. Bench-Capon. Persuasion in practical argument using value based argumentation frameworks. *Journal of Logic and Computation*, 13 3:429–48, 2003.

[11] 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.

[12] H. W. J. Rittel and M. M. Webber. Dilemas in a general theory of planning. *Policy Sciences*, pages 155–169, 1973.

[13] L. Amgoud, C. Cayrol, and M. C. Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *Proc. of the 10th workshop on Non-Monotonic Reasoning*, pages 1–9, 2004.

[14] W. Rehg, P. McBurney, and S. Parsons. Computer decision support for public argumentation: Assessing deliberative legitimacy. *Artificial Intelligence and Society*, 19(3):203–288, 2005.

[15] P. McBurney and S. Parsons. Risk agoras: Dialectical argumentation for scientific reasoning. In *Proc. of the 16th conference on Uncertainty in AI*, pages 371–379, Stanford, CA, USA, 2000. Morgan Kaufmann.

[16] Z. Masters, A. Macintosh, and E. Smith. Young people and e-democracy: Creating a culture of participation. In R. Traunmüller, editor, *EGOV*, LNCS 3183, pages 15–22. Springer, 2004.

[17] Ø. Sæbø and H. Nilsen. The support for different democracy models by the use of a web-based discussion board. In R. Traunmüller, editor, *EGOV*, LNCS 3183, pages 23–26. Springer, 2004.

[5] A shorter version of this paper is to be presented at the 2006 EGOV conference. I am grateful to both the EGOV reviewers and the COMMA reviewers for their helpful comments and suggestions that have led to improvements in this revised and extended version. I would also like to thank Trevor Bench-Capon and Peter McBurney for their contributions to our joint work on argumentation and practical reasoning.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Argument based machine learning in a medical domain

Jure Žabkar [a,1], Martin Možina [a], Jerneja Videčnik [b] and Ivan Bratko [a]
[a] *Faculty of Computer and Information Science, University of Ljubljana, Slovenia*
[b] *Clinic for Infectious Diseases, Ljubljana, Slovenia*

Abstract. Argument Based Machine Learning (ABML) is a new approach to machine learning in which the learning examples can be accompanied by arguments. The arguments for specific examples are a special form of expert's knowledge which the expert uses to substantiate the class value for the chosen example. Možina et al. developed the ABCN2 algorithm - an extension of the well known rule learning algorithm CN2 - that can use argumented examples in the learning process. In this work we present an application of ABCN2 in the medical domain which deals with severe bacterial infections in geriatric population. The elderly population, people over 65 years of age, is rapidly growing as well as the costs of treating this population. In our study, we compare ABCN2 to CN2 and show that using arguments we improve the characteristics of the model. We also report the results that C4.5, Naïve Bayes and Logistic Regression achieve in this domain.

Keywords. Argument Based Machine Learning, Rule learning, Geriatric population, Bacterial infections

1. Introduction

The elderly population is a unique one and that is also true from the medical perspective. Compared to younger population, people over 65 years of age usually react to a disease in a different way. Many symptoms may not even be present or they are masked by others which makes it a very difficult task for a medical doctor to diagnose a condition, to decide a proper treatment or to estimate the patient's risk of death. From a wider perspective, the proportion of elderly in the population is growing rapidly and so are the costs of medical treatment, which presents an emerging economic problem.

Infections in the aging population present an increasing problem in the developed countries. Many patients that with an infection have associated chronic diseases such as diabetes, heart, kidney, lung or liver disease which makes the treatment even more complicated. The number of nursing home residents is also increasing in this population. Because of the specific living environment, these people are usually more susceptible to bacterial infections. Despite great progress in treating infectious diseases they remain one of the major causes of death in geriatric population. Some diferences in the course of illness can be observed compared to younger patients. Greater risk of severe bacterial

[1]Correspondence to: Jure Žabkar, Faculty of Computer and Information Science, University of Ljubljana, Tržaška 25, SI-1001 Ljubljana, Slovenia. Tel.: +386 1 4768 299; Fax: +386 1 4768 386; E-mail: jure.zabkar@fri.uni-lj.si.

infection is due to the patient being immunocompromised [1], immobile, nursing home resident or comorbidity. In the elderly, the infections are often present with untypical signs, such as the absence of fever [2,3,4,5], the absence of cough at pneumonia and weakness or changed mental status [6]. These usually cause a delay in making a right diagnosis. A proper and efficient antimicrobial treatment is often given too late, and the risk of a fatal outcome is increased [7,8].

In this paper we apply an approach to argument-based machine learning to this domain. The motivation for using machine learning in this study is to build a model from data which would help the physician, at the first examination of the patient, to decide the severity of the infection and consequently, whether the patient should be admitted to hospital or could be treated as an outpatient. Moreover, we would like to have an understandable model, not a black box, to see which parameters play a decisive role. Several studies are described in the literature regarding the difficulty of the course of disease [9,10,11]. Fine et al. [11] implemented a prognostic model for adult patients with documented bacterial pneumonia. An overall study, regarding the bacterial infections of different organs and taking into account so many clinical as well as laboratorial parameters has, to our knowledge, not been carried out yet.

The alternative to machine learning would be to implement with the help of domain experts an expert system and use it for diagnosing the severity of infection. The knowledge possessed by experts is usually implicit and they find it extremely difficult to elicit it in the form of a set of rules. On the other hand, it was shown that it is easier for experts to discuss certain concrete cases, instead of giving a general theory. Research from defeasible argumentation [12] proposed an alternative approach to building expert systems. Experts should first give arguments for some specific examples of possible outcomes. These arguments are then given to an argumentation engine, an expert system that can use these (possibly contradictory) arguments to make predictions for new cases. Whenever a decision from the arguments could not be inferred, the experts are again asked for additional arguments.

Our approach, Argument Based Machine Learning (ABML) [13,14], combines machine learning and argumentation. ABML is a new approach to machine learning in which the learning examples could be accompanied by arguments. The expert chooses a subset of learning examples and gives reasons, in the form of arguments, why the class value of the example is as given. We have developed an argument-based rule learning algorithm ABCN2 [15,14], an extension of the well-known CN2 algorithm, which we here apply to our medical domain hoping to improve the prediction quality of standard machine learning techniques by using arguments given by experts.

2. Argument Based Machine Learning

Argument Based Machine Learning (ABML) [14,13] is a new approach to machine learning that can learn from examples and arguments. While the standard problem of machine learning from examples is to induce a hypothesis that explains given examples, in ABML some of these examples are given arguments, and the problem of ABML is to induce a hypothesis that explains examples using these arguments. The arguments for specific examples are a special form of expert's knowledge which he/she uses to substantiate the class value for the chosen example. We believe that it is much easier for the

expert to justify the class value of the specific example than to provide some generally applicable rules. We consider this as the main advantage of the ABML approach. The other two important advantages of ABML are:

1. Arguments impose constraints over the space of possible hypotheses, thus reducing search complexity,
2. An induced hypothesis should make more sense to an expert as it has to be consistent with given arguments.

Regarding the first advantage above, it is obvious that constraining the search space should help to overcome the problem of explosive combinatorial space of possible hypotheses. Arguments do not simply reduce search complexity but they rather make it smarter, by directing the search into subspaces where better hypotheses should reside. Regarding the second advantage, we should mention that several hypotheses could explain the given examples well, but some may not be understandable to the expert. By including the arguments the induced hypotheses should make more sense to the expert.

2.1. ABCN2

Argument Based CN2 (ABCN2) is a realization of the concepts just described. It is an extension of the rule learning algorithm CN2 [16,17] in which a subset of learning examples may be given arguments. The details of the algorithm and the formalism of accepted arguments by the method are described in [15,18]. Here we shall give only a brief overview.

2.1.1. Argumented examples

A learning example E in the usual form accepted by CN2 is a pair (A, C), where A is an attribute-value vector, and C is a class value. An argumented example AE is a triple of the form:

$$AE = (A, C, Arguments)$$

As usual, A is an attribute-value vector and C is a class value. $Arguments$ is a set of arguments $Arg_1, \ldots, Arg_n$, where an argument Arg_i has one of the following forms:

$$C \text{ because } Reasons$$

or

$$C \text{ despite } Reasons$$

The former specifies a *positive* argument (speaks for the given class value), while the latter specifies a *negative* argument (speaks against the class value). $Reasons$ is a conjunction of reasons $r_1, \ldots, r_n$,

$$Reasons = r_1 \wedge r_2 \wedge \ldots \wedge r_n$$

where each of the reasons r_i is a condition on a single attribute (e.g. $X = x$, where X is the name of the attribute and x is a possible value for this attribute).

Patient	Consciousness	Trombocytes	BloodPressure	Gender	Death
Patient n.1	normal	≥ 100	≥ 100	male	no
Patient n.2	disoriented	< 100	≥ 100	male	no
Patient n.3	normal	< 100	< 100	female	yes
Patient n.4	disoriented	≥ 100	< 100	female	yes

Table 1. A simplified version of the infections database.

2.1.2. The ABCN2 Algorithm

ABCN2 is based on the version of CN2 that induces a set of unordered rules [17]. The main difference between CN2 and ABCN2 is in the definition of rule *covering*. In the standard definition (CN2), a rule covers an example if the condition part of the rule is true for this example. In argument based rule learning, this definition is modified to: A rule R *AB-covers* an argumented example E if:

1. All conditions in R are true for E (same as in CN2),
2. R is consistent with at least one positive argument of E, and
3. R is not consistent with any of negative arguments of E,

where rule R is consistent with an argument Arg if the reasons of Arg are present among conditions of R.

We will illustrate the idea of AB-covering on a simplified version of an infections database. Each example is a patient described with four attributes: Consciousness (with possible values "normal" and "disoriented"), Trombocytes (possible values "≥ 100" and "< 100"), BloodPressure ("≥ 100" and "< 100"), and Gender ("male" and "female"). The class is Death (with possible values "yes" and "no"). Let there be four learning examples as shown in Table 1.

The expert's argument for occurrence of death for Patient n.3 could be: she died because the number of trombocytes is less than hundred. Similarly, Patient n.4 died because she was disoriented. A negative argument can be: Patient n.3 died despite her consciousness being normal. Third patient would in our syntax be written as:

$$((Consciousness = normal, Trombocytes = < 100, BloodPressure = < 100, Gender = female), Death = yes, \{Death = yes \text{ because } Trombocytes = < 100, Death = yes \text{ despite } Consciousness = normal\}).$$

Arguments given for examples additionally constrain rules *covering* this example. Remember that in CN2, rules have the form:

$$\text{IF } Complex \text{ THEN } Class$$

where $Complex$ is the conjunction of simple conditions, called *selectors*. For the purpose of this paper, a selector simply specifies the value of an attribute. A rule in our domain can be:

$$\text{IF } Consciousness = normal \text{ AND } Trombocytes = < 100 \text{ THEN } Death = yes$$

The condition part of the rule is satisfied by the attribute values of third patient, so we say that this rule *covers* this example (all conditions in the rule are true for example).

As an illustration of the differences between AB-covering and the usual definition of covering, consider again our example with the argument that she died because the number of trombocytes is less than hundred and despite her consciousness being normal. Consider the following rules:

Rule 1: IF $Gender = female$ THEN $Death = yes$.
Rule 2: IF $Consciousness = normal$ AND $Trombocytes = <$ 100 THEN
$Death = yes$
Rule 3: IF $Trombocytes = <100$ AND $BloodPressure = <100$ THEN
$Death = yes$

All three rules cover the third example and have 100% accuracy on the above data set. However, Rule 1 does not AB-cover the example, because it is not consistent with the positive argument (conditions of positive argument are not satisfied). Rule 2 is consistent with positive argument, however as it is also consistent with the negative argument, it still does not AB-cover the third example. The last example AB-covers the Patient n.3 example.

We mentioned that the first requirement for ABML is that an induced hypothesis explains argumented examples using given arguments. In rule learning this means that each argumented example must be covered by at least one rule that AB-covers the example. This is achieved simply by replacing covering in original CN2 with AB-covering. As a result of this replacement, non-argumented examples may also be AB-covered. A further improvement of the original CN2 algorithm is the requirement that induced rules explain as many as possible non-argumented examples by arguments given for the argumented examples (see [15,18]). Three further important mechanisms inherited from CN2 are improved in ABCN2. These are: examples removing strategy (after a rule is learned), the evaluation function, and classification by rules. In the remainder of this section we will explain why these parts are problematic and how we improved them.

2.1.3. Removing strategy

After CN2 learns a rule, it removes examples covered by this rule and recursively continues learning on the remaining examples. This approach assumes that the algorithm induces the best possible rule for given examples - there exists no rule that would be evaluated better than this rule and cover the same examples. This assumption might be true for the original CN2, but for ABCN2, where we first learn from argumented examples (learning is constrained by arguments of argumented example), this assumption is likely to be incorrect. A rule learned from an argumented example can be seen as the best possible rule covering this example. However, this rule may not be the best rule for other examples covered by this rule. For instance, it could happen that CN2 finds a better rule for some of these examples. Therefore, removing examples after learning from argumented examples might prevent classical CN2 from learning some good rules. To avoid this drawback, in [14] we developed a probabilistic covering strategy.

2.1.4. Evaluation function

The evaluation function in rule learning algorithms is used to determine the goodness (or quality) of a rule. This measure of goodness should determine the rule's potential to predict yet unseen cases. In versions of CN2, rules are often evaluated using the Laplace formula for probability. Due to a search through a huge space of possible hypotheses, this evaluation method usually gives optimistic estimates of probability [19]. In the case of ABCN2, rules learned from argumented examples are selected from a smaller space of hypotheses than rules induced with the standard CN2 algorithm, and thus the quality of a rule learned from an argumented example is relatively underestimated when compared to a rule learned from standard CN2. We developed a novel evaluation method based on extreme value theory [19] that accounts for multiple comparisons in the search. Using this method, the evaluations of rules learned from arguments are not underestimated any more. Due to this fact, the quality of a rule becomes now a very important factor in classification.

2.1.5. Classification from rules

Most of the methods for classification by rules take into account the distribution of covered examples by these rules. However, similarly to the Laplace evaluation function, the number of positive examples in the distribution tends to be optimistic. As our evaluation function, described in the previous section, accounts for the number of candidate hypotheses, it would make sense to use the quality of a rule (instead of distribution) in classification. We developed such a method based on the Minimax theorem [20], for a detailed explanation of this classification method see [14].

3. Experiments

3.1. Data

The data for our study was gathered at the Clinic for Infectious Diseases in Ljubljana, from June 1st, 2004 to June 1st, 2005. The physicians included only patients over 65 years of age with CRP value over 60 mg/l, which indicated a bacterial etiology of the infection. The patients were observed for 30 days from the first examination or until death caused by the infection. The data includes 40 clinical and laboratorial parameters (attributes) acquired at the first examination for each of 298 patients (examples). The infections are distinguished with respect to the site where bacteria is found or on the clinical basis (respiratory, urinary tract, soft tissues, other). The continuous attributes were categorized by the physician. The distribution of the class values is the following:

- 34 examples (11.4%) for 'death = yes'
- 263 examples (88.6%) for 'death = no'

3.2. Arguments

The argumentation was done by the physician who was treating the patients and could by her expert knowledge state several positive and negative arguments for 32 examples, where all argumented examples were from class *death* = *yes*, namely she gave the reasons

Attribute	Value	
GENDER	Z	**Positive arguments**
AGE_YEARS	92	
AGE	C	DEATH=YES because RESPIRATORY_RATE_D="≥ 16"
NURSING_HOME_RESIDENT	NO	DEATH=YES because SATURATION_D="≤ 90"
COMMORBIDITY	0	DEATH=YES because BLOOD_PRESSURE_D="≤ 100"
DIABETES	NO	DEATH=YES because TEMPERATURE_D="> 37.9"
HEART	NO	DEATH=YES because LEUKOCYTES_D="≥ 12"
KIDNEY	NO	DEATH=YES because CREATININE_D="≥ 100"
LIVER	NO	DEATH=YES because BLOOD_UREA_D="≥ 13"
LUNG	NO	DEATH=YES because NA_D="> 147"
IMMUNITY	NO	DEATH=YES because AGE_YEARS is high
CENTRAL_NERVE_SYSTEM	NO	DEATH=YES because WEAKNESS=YES
MOBILITY	YES	DEATH=YES because CONSCIOUSNESS=DISSORIENTED
CONTINENCE	YES	
BEDSORE	NO	**Negative arguments**
CATHETER	NO	
IMPLANT	NO	DEATH=YES despite MOBILITY=YES
VOMITING	NO	DEATH=YES despite CONTINENCE=YES
DIABLOODPRESSUREHEA	NO	DEATH=YES despite TROMBOCYTES_D="≥ 100"
WEAKNESS	YES	DEATH=YES despite HEART_RATE_D="< 100"
CONSCIOUSNESS	DISSORIENTED	DEATH=YES despite RODS_D="< 10"
TROMBOCYTES_D	≥ 100	DEATH=YES despite CRP_D="< 150"
TEMPERATURE_D	>37.9	DEATH=YES despite COMMORBIDITY=0
RESPIRATORY_RATE_D	≥ 16	
SATURATION_D	≤ 90	
HEART_RATE_D	< 100	
BLOOD_PRESSURE_D	≤ 100	
LEUKOCYTES_D	≥ 12	
RODS_D	< 10	
CRP_D	< 150	
CREATININE_D	≥ 100	
BLOOD_UREA_D	≥ 13	
GLU_D	< 15	
NA_D	> 147	
INFECTION_TYPE	RESPIRATORY	
DEATH (class value)	YES	

Table 2. A sample argumented example from the infections database.

she believed caused death for each selected patient. A sample argumented example is shown in Table 2.

One could, at this point, ask an interesting question about these arguments: whether they would, if used as rules, describe the domain sufficiently well. We built a simple classifier from the given arguments and tested it on the same data set; for each case, we counted the number of applicable arguments for class *death = yes* and compared this number to the number of arguments for class *death = no*. The accuracy of such a classifier is only slightly above 40%, therefore there is still a large space available for machine learning to improve. Since the default accuracy in this domain is 88.6% it indicates that the knowledge which is hidden in arguments is far from perfect. However, please note that this experiment is not used to validate the expert knowledge. To do that, at least the arguments for examples from the opposite class should be given as well. Our intention is merely to show that the knowledge given by the arguments is neither perfect nor complete though it can still help to improve learning.

3.3. Results

Learning and testing was performed by 10-fold cross validation which was carried out 10 times with different random splits of examples into folds. We compared the algorithms

ABCN2 and CN2, where both methods used the improvements shown in the previous section, so that their comparison directly represents the influence of the arguments added to the learning examples. Both algorithms were then compared to Naïve Bayes (NB), decision trees (C4.5) and logistic regression (LogR). The algorithms were compared with regard to classification accuracy, area under ROC (AUC) and Brier score. All the methods and tests were implemented within Orange toolkit [21]. The results are shown in Figure 1–3.

Observing classification accuracy, that is the percentage of correct classifications, we can see that CN2, ABCN2 and C4.5 achieve similar results while NB and LogR perform significantly worse (Fig. 1). Although classification accuracy is important it should be accompanied by other estimates especially because the majority classifier itself is quite accurate in this domain due to the imbalance between the two classes. Therefore we also measure AUC and Brier score, which are applicable as all the methods also give the probability of the predicted class. AUC measures how well the method ranks examples: it is the probability that for two randomly chosen examples with different classes, the method will correctly decide classes for these examples (it is not allowed to classify both in the same class, as it knows that they are from different classes). This measure is often used to evaluate hypotheses in medical domains, where we wish to have methods that separate positive from negative examples as well as possible. Figure 2 shows that, according to AUC, ABCN2 significantly outperforms all other methods. The same effect also comes out in Brier scores (Figure 3), which measures the average quadratic error of predicted probability. It is important to note that for imbalanced domains, such as our domain, AUC and Brier score are more relevant measures of success than accuracy.

3.4. Discussion

ABCN2 achieved better results than CN2 according to all three measures by using arguments given by an expert. The question is how the induced hypotheses from both measures differ and why ABCN2 is the better method. To examine the hypotheses, we induced a set of rules from the whole data set with ABCN2 and CN2. As the arguments were given only to examples with class value *death=yes*, the induced rules for *death=no* were the same for both methods. Both methods induced 14 rules for the class *death=yes*, however there were two important differences between these two sets of rules. First, due to the restriction of hypotheses space with arguments, about half of the rules were different. While inspecting the rules that were the same in CN2's and ABCN2's set, we noticed that the quality estimates of these rules were different. For example, the rule:

IF trombocites<100 AND mobility=no THEN death=yes

was present in both rule sets. It covers 6 examples with class value *death=yes* and 1 with *death=no*, which means that the relative frequency of *death=yes* is 6/7 = 0.86. However, the evaluation function based on extreme value distributions [19] used in CN2 estimated the probability of this class (given that the conditions are true) as 0.47, which is much less than 0.86. This happens because there is a high probability that such a rule would be found by chance. On the other hand, when learning with ABCN2, the evaluation of the same rule is 0.67. In CN2, this rule was obtained by searching the whole space unguided by expert knowledge while in ABCN2 the rule was built from the argument *'death=yes BECAUSE trombocites<100'*. The search space in ABCN2 is smaller, which means that

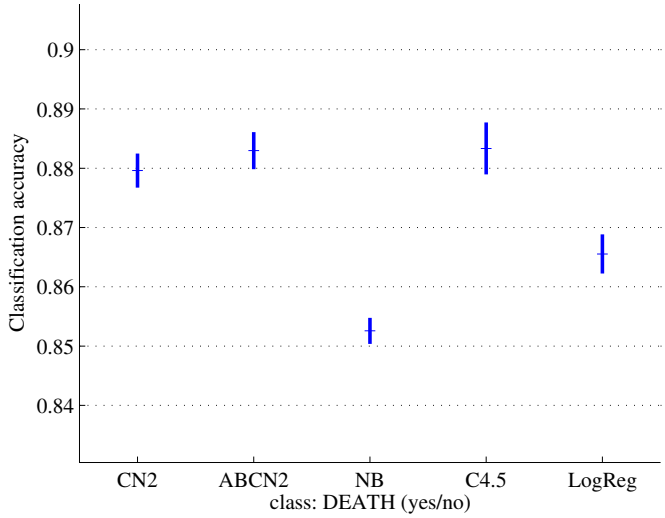


Figure 1. Mean values and standard errors of classification accuracy across tested methods.

the probability of finding such a rule by chance is lower. So, the expected quality of the rule is higher.

In the above paragraph we have shown the importance of the first expected advantage of ABML: "Arguments impose constraints over the space of possible hypotheses, thus reducing search complexity". Regarding the second advantage, that induced rules should make more sense to an expert, we asked our expert (Jerneja Videčnik) to examine the rules and compare them. Unfortunately, she could not decide which rules were more understandable to her. We believe that this occurs due to the large number of arguments with only one reason given for each example, while our restriction is that the rule must be consistent with at least one positive argument. The rule must, therefore, contain only one of the given reasons and can neglect the others.

4. Conclusion

We described the application of argument based machine learning to the medical domain dealing with severe bacterial infections in geriatric population. Our intention was to show how arguments can be used to guide a machine learning algorithm towards a better model. The use of arguments proved to be a powerful approach which offers a new insight in using expert knowledge in machine learning. This knowledge is not given as general background knowledge but is rather tied to specific examples to reason about the class value using available attributes.

We used ABCN2 which is an argument-based version of the CN2 algorithm. Our medical domain is the first real-life domain to which ABML has been applied. Several examples were given arguments by the medical doctor and used in the learning process. In our experiments we compared ABCN2 to some other popular machine learning algorithms that are not capable of using arguments, such as CN2, C4.5, Naïve Bayes and lo-

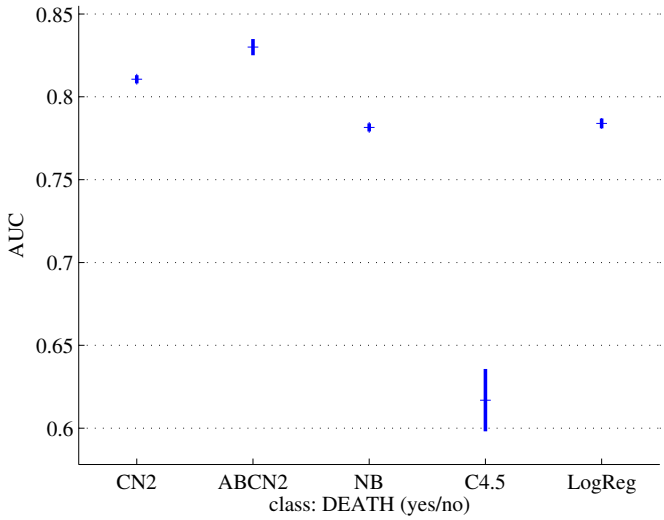


Figure 2. Mean values and standard errors of AUC across tested methods.

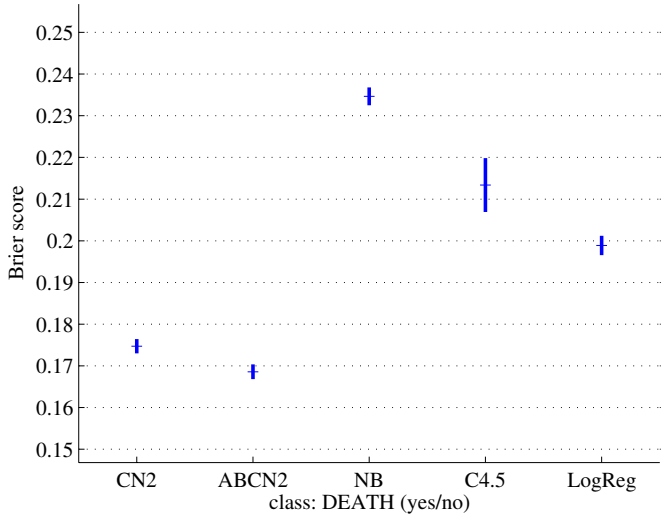


Figure 3. Mean values and standard errors of Brier score across tested methods.

gistic regression. The results show several advantages of ABCN2 over other algorithms. ABCN2 significatnly outperforms others in classification accuracy, AUC and Brier score.

For further work, it would be very interesting to see how well can an expert alone (without machine learning) would classify the examples. We would need to ask an independent expert, who had not seen these examples before, and ask her to classify them according to her knowledge. We believe that such experiment would truly show the added

value of an argument based approach. Moreover, it would also be interesting to see how the number of argumented examples influences the results and check how the results change if we select different subsets of argumented examples. In our experiment the number of arguments was quite large, which might not always happen, as argumenting examples are usually time consuming for experts. Another interesting experiment would be to have several physicians giving arguments for the examples and compare the models.

Acknowledgements

This work was partly supported by the European Commission's Information Society Technologies (IST) programme, through Project ASPIC (IST-FP6-002307), and Slovene Agency for Research and Development (ARRS).

References

[1] Ben-Yehuda A, Weksler ME. Host resistance and the immune system. Clin Geriatr Med 1992; 8: 701-11.
[2] Marco CA et al. Fever in geriatric emergency patients: clinical features associated with serious illness. Ann Emerg Med 1995; 26:18-24.
[3] Castle SC et al. Fever response in elderly nursing home residents: are the older truly colder? JAGS 1991; 39: 853-7.
[4] Gleckman R, Hibert D. Afebrile bacteriemia: a phenomena in geriatric patients. JAMA 1982; 248: 1478-81.
[5] Mellors JW et al. A simple index to identify occult bacterial infection in adults with acute unexplained fever. Arch Intern Med 1987; 147: 666-71.
[6] Rockwood K. Acute confusion in elderly medical patients. J Am Geriatr Soc 1989; 37: 150-4.
[7] Fontanarosa PB, Kaeberlein FJ, Gerson LW, Thompson RB. Difficulty in predicting bacteriemia in elderly emergency patients. Ann Emerg Med 1992; 21: 842-8.
[8] Pfitzenmeyer P, Decrey H, Auckenthaler R, Michel JP. Predicting bacteriemia in older patients. JAGS 1995; 43: 230-5.
[9] Fine MJ, Auble TE, Yeay DM, et al. A prediction rule to identify low-risk patients with community acquired pneumonia. NEJM 1997; 336: 243-50.
[10] Farr BM, Sloman AJ, Fisch MJ. Predicting death in patients hospitalized for community acquired pneumonia. Ann Int Med 1991; 115:428-36.
[11] Fine MJ, Smith MA, Carson CA, et al. Prognosis and outcomes of patients with community acquired pneumonia. JAMA 1996; 275: 134-41.
[12] Prakken H, Vreeswijk G. Handbook of Philosophical Logic, second edition, volume 4, chapter Logics for Defeasible Argumentation, pages 218-319. Kluwer Academic Publishers, Dordrecht etc, 2002.
[13] Bratko I, Možina M: Argumentation and Machine Learning. In: Deliverable 2.1 for the ASPIC project 2004.
[14] Možina M, Žabkar J, Bratko I: Implementation of and experiments with ABML and MLBA, preliminary version. ASPIC deliverable D3.4, 2006.
[15] Možina M, Žabkar J, Bratko I: Argument Based Rule Learning. Accepted for publication in Proceedings of ECAI, Riva del Garda, 2006.
[16] Clark P, Niblett T: The CN2 induction algorithm. Machine Learning Journal, 4: 261-283, 1989.

[17] Clark P, Boswell R. Rule Induction with CN2: Some Recent Improvements. In Machine Learning - Proceedings of the Fifth European Conference (ESWL-91), pages 151-163, Berlin, 1991.

[18] Možina M, Žabkar J, Bench-Capon T, Bratko I: Argument Based Machine Learning Applied to Law. Artificial Intelligence and Law, 2005; In press.

[19] Možina M, Demšar J, Žabkar J, Bratko I: Why is Rule Learning Optimistic and How To Correct It. Accepted for publication in Proceedings of ECML conference, Berlin, 2006.

[20] John von Neumann. Zur Theorie der gessellshaftsspiele. Mathematische Annalen, 100:295-320, 1928. English Translation Fin Tucker AW, Luce RD, Contributions to the Theory of Games IV, Annals of Mathematics Studies 40, 1959.

[21] Demšar J, Zupan B, Leban G (2004) Orange: From Experimental Machine Learning to Interactive Data Mining, White Paper (www.ailab.si/orange), Faculty of Computer and Information Science, University of Ljubljana.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

CBR and Argument Schemes for Collaborative Decision Making

Pancho Tolchinsky [a,*], Sanjay Modgil [b], Ulises Cortés [a] and Miquel Sànchez-Marrè [a]
[a] *Knowledge Engineering & Machine Learning Group*
Technical University of Catalonia
[b] *Advanced Computation Lab, Cancer Research UK*

Abstract. In this paper we present a novel approach for combining Case-Based Reasoning (CBR) and Argumentation. This approach involves *1)* the use of CBR for evaluating the arguments submitted by agents in collaborative decision making dialogs, and *2)* the use of Argument Schemes and Critical Questions to organize the CBR memory space. The former involves use of past cases to resolve conflicts among newly submitted arguments by assigning them a strength, and possibly submitting additional arguments deemed relevant in similar past deliberations. The latter enables use of agents' submitted arguments instantiating Argument Schemes and Critical Questions, to assess the similarity among cases. This use of CBR and argumentation is formulated with the *ProCLAIM* model, which features a Mediator Agent that directs proponent agents in their deliberation and subsequently evaluates their submitted arguments so as to conclude whether a proposed decision is valid. To motivate and substantiate the practical value of this approach, we illustrate its application in the human organ transplantation field.

Keywords. Case-Based Reasoning, Argument Schemes, Multi-Agent Systems

1. Introduction

In many domains decisions are made following established guidelines that guarantee their *correctness* and/or safety in the case of safety-critical domains. However, there are circumstances in which decisions that deviate from the guidelines are justified. In this paper we present a model – *ProCLAIM* - that provides a setting for *proponent* agents to argue over the validity of their intended decisions. The model features a Mediator Agent (MA) that directs the proponent agents in their deliberation and subsequently evaluates the submitted arguments so as to conclude whether a proposed decision is valid. The MA will generally accept as valid only those decisions that, in light of the given arguments, comply with the guidelines. However, the MA may exceptionally be persuaded to accept (resp. reject) proposed decisions whose supporting and attacking arguments indicate that, although they do not comply (resp. do comply) with the guidelines, there is evidential basis to accept (resp. reject) them.

*Correspondence to: Pancho Tolchinsky, Dept. Llenguatges i Sistemes Informàtics - Universitat Politècnica de Catalunya. c/ Jordi Girona 1-3, 08034 Barcelona, Spain. E-mail: tolchinsky@lsi.upc.edu

Hence, *ProCLAIM* defines a Case-Based Reasoning component/engine (CBRe) for evaluating, in light of the arguments used in the current and previous deliberations, whether there is sufficient evidence to accept or reject a proposed decision. To enable this functionality, the CBRe's memory space is organized using a structured set of Argument Schemes and Critical Questions [1]. This provides the CBRe with the means to compare cases on the basis of arguments used in each deliberation. Broadly speaking, two cases are similar if the submitted arguments, associated to these cases, are similar.

From an argumentation perspective, a proposed decision is itself represented by an argument that is attacked and reinstated by the submitted arguments organized into a graph of interacting arguments. Assessing the validity of a proposed decision thus amounts to determining the dialectical status of the argument representing the decision. To do so may require establishing a preference between arguments that attack each other, based on the relative strength of the mutually attacking arguments. The role of the CBRe is to use past cases in order to assign these strengths, as well as possibly submitting additional arguments deemed relevant in similar past deliberations.

To illustrate the practical value of *ProCLAIM* and in particular of the CBRe, we apply the model in a transplant scenario [2]. In the following section we describe the *ProCLAIM* model. In §3 we introduce the transplant scenario, and in §4 we show how the CBRe makes use of arguments to compare cases and how cases can be used to resolve conflicts among arguments. Finally, §5 concludes with a discussion and programme for future work.

2. The *ProCLAIM* Model

Broadly construed, the *ProCLAIM* model consist of a mediator agent, MA, directing proponent agents in an argument based collaborative decision making dialog, in which the final decision must comply with certain domain dependent guidelines. The arguments submitted by the proponent agents may also persuade the MA to accept decisions that deviate from the guidelines. For example, the MA may be able to reason that the submitted arguments supporting an alternative decision have proven to be correct in previous similar deliberations.

We believe that *ProCLAIM* is of particular value in safety-critical domains (although the scope of domain may well be wider) where the consequences ensuing from a *wrong* decision may be catastrophic. Guidelines in such sensitive environments usually exist and are created in an attempt to minimize hazardous decisions. Nonetheless, there are circumstances in which a decision is appropriate despite violating established guidelines. Moreover, in such environments, arguments supported by empirical evidence are somewhat more persuasive.

ProCLAIM defines three main tasks for the MA: *1)* Inform the proponent agents as to what are their dialectical possible moves at each stage of the deliberation; *2)* Ensure that the submitted arguments are relevant (*e.g.*, comply with the guidelines), and *3)* Evaluate the submitted arguments in order to identify the *winning* arguments and thus determine whether a proposed decision is valid. This last task may require the assignment of strengths to the given arguments and possibly submission of additional arguments. In order to undertake these tasks, MA makes use of four knowledge resources (see fig. 1):

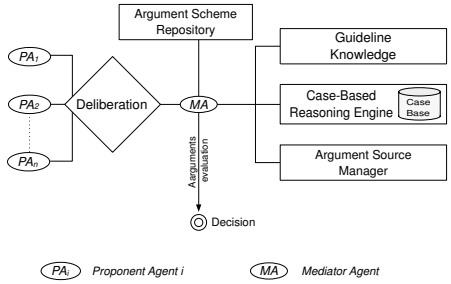


Figure 1. *ProCLAIM*'s Architecture

Argument Scheme Repository (ASR): In order to direct the proponent agents in their deliberation the MA makes use of a repository of argument schemes and their associated critical questions formalized in a way that defines a protocol based exchange of arguments (*e.g.* given a submitted argument A instantiating a scheme of ASR, the MA can reference the ASR in order to identify the schemes that, if effectively instantiated, constitute an attack on A). As we will see in §4, the ASR also structures the CBRe's memory space.

Guideline Knowledge (GK): This component enables the MA to check whether the arguments comply with the established knowledge, by checking what are the valid instantiations of the schemes in ASR (the ASR can thus be regarded as an abstraction of the GK).

Case-Based Reasoning Engine (CBRe): This component enables MA to assign strengths to the submitted arguments on the basis of their associated evidence gathered from past deliberations, as well as provide additional arguments deemed relevant in previous similar situations.

Argument Source Manager ASM: Depending on the source from whom, or where, the arguments are submitted, the strengths of these arguments may be readjusted by the MA. Thus, this component manages the knowledge related to the agents' roles and/or reputations, and/or the types of certificates, or references, that may empower agents to undertake some exceptional decision.

The agents' argument construction is based on a first order logic programming language described in [3]. This work also defines the conflict based interaction between arguments. Given the constructed arguments and their interactions we apply Dung's seminal *calculus of opposition* [4] to determine the justified or *winning* arguments. However, determining the winning arguments may require the MA to assign strengths to the submitted arguments and possibly the submission of additional arguments. This is further discussed and illustrated in sections 3 and 4. The agents' dialog and in particular, the role of the MA in directing the deliberation by referencing the ASR is defined in [5]. Agents construct and submit arguments by instantiating the schemes and critical questions in the ASR. The MA's task is then to determine which are the winning arguments in order to conclude whether the proposed decision complies with the GK. This may involve referencing the CBRe to access similar past experiences and arguments given to support an undertaken decision not compliant with the GK, but which proved to have a successful outcome. This may also involve referencing past experiences in order to resolve mutually

attacking arguments by assigning relative strengths to these arguments. These roles of the CBRe are further developed in §4. We now introduce the transplant scenario in order to illustrate the practical value of *ProCLAIM*, and in particular the value of Case-Based Reasoning in resolving conflicting arguments and the use of arguments for comparing cases.

3. The Transplant Scenario

Human organ transplantation constitutes the only effective therapy for many life-threatening diseases. However, while the increasing success of transplants has led to increase in demand, the lack of a concomitant increase in donor organ availability has led to a growing disparity between supply and demand [6]. In spite of this, an important percentage of human organs available for transplantation are discarded as being deemed non-viable for that purpose.

The human organ selection process illustrates the ubiquity of disagreement and conflict of opinion in the medical domain. What may be a sufficient reason for discarding an organ for some qualified professionals may not be for others. Hence, contradictory conclusions may be derived from the same set of facts. For example, suppose a donor with a smoking history but no history of *chronic obstructive pulmonary disease* (COPD). The medical guidelines indicate that a donor's smoking history is a sufficient reason for deeming a donor's lung as non-viable [7]. However, there are qualified physicians that reason that the donor's lung is viable given that there is no history of COPD [6]. Similarly, the guidelines suggest discarding the kidney of a donor whose cause of death was *streptococcus viridans endocarditis* (*sve*)[7]. However, some reason that by administrating *penicillin* to the recipient the kidney can safely be transplanted [8].

The transplant scenario begins when a potential donor becomes available. The donor's organs deemed non-viable by the Transplant Coordinator (which we name the Donor Agent, DA) are discarded, whereas the organs deemed viable are offered via a third-party (Transplant Organization) in a queue of Transplant Units, (which we name Recipient Agents) that may be located in different hospitals. These Recipient Agents, RA_1,...,RA_n, to which the organ may eventually be offered may accept it, in which case they may attempt to implant it to a potential recipient they are responsible for. Or, if every RA_j fails to accept the organ, it is discarded, *i.e.* not extracted from the donor.

A DA's decision to not offer an organ which he believes to be non-viable prevents other RA_j's from having the opportunity to make use of that organ. The human organ selection process is described in more detail in [2] where an alternative selection process is proposed to be managed by *CARREL*, an agent-based organization designed to improve the overall transplant process. In this alternative process a DA_i that detects a potential donor offers all the potentially transplantable organs irrespective of whether he believes the organs to be viable or non-viable. *CARREL* then distributes the offer to the appropriate RAs. Together with an organ offer, the DA_i has to provide the arguments that support his assessment over the organ's viability. In that way, a RA_j will be able to counter-argue DA_i's assessment when there is disagreement. The DA_i, in turn, will have the chance to counter-argue, and so on. Thus an argument-based dialog may take place between DA_i and RA_j. In particular, a DA_i's arguments for the non-viability of an organ may now be defeated by the RA_j's arguments for viability, and thus, RA_j may

have the opportunity to make use of that organ. In the same way, DA_i's arguments for the viability of the offered organ may be stronger than those of a RA_j for non-viability. This will result in committing RA_j to transplant the offered organ as his decision for not transplanting it would be deemed unjustified.

Therefore, the *ProCLAIM* model is instantiated in order to extend the *CARREL* System so as to support the new selection process which we believe has the potential to increase the number of organs current selection processes make available. In particular, the proponent agents are the DA_i and RA_j, the GK is instantiated by the Acceptability Criteria Knowledge Base (ACKB) that encodes the criteria the medical doctors should refer to when deciding the organs' viability. The Argument Source Manager relates to the agents' reputation. Namely, the MA may deem as stronger the arguments submitted by agents with good reputation (*e.g.* a RA_j that have in the past successfully transplanted those organs which he claimed to be viable). Finally, the CBRe allows the MA to evaluate the submitted arguments on the basis of past transplantation experiences. For example, if an agent argues that the lung of a donor with a smoking history can safely be transplanted because he did not have COPD, the MA references the CBRe in order to evaluate this argument's evidential support. Note that at the same time, the submitted arguments highlight what are the relevant factors for deciding a case. Namely, the argument graphs highlight the relevant attributes for assessing the similarity among cases.

The stage in the transplant experience in which arguments are submitted have associated different evidential weight. Arguments submitted before an organ is extracted are referred to as ***phase 1*** arguments and have associated weaker evidential wight. If an organ is deemed viable for a RA_j, the organ is extracted. At this time, new evidence may indicate that the organ is in fact non-viable, and so it is discarded. The RA_j is then obliged to provide *CARREL* with the new arguments (capturing the new evidence) as to *why* the organ is non-viable. These are referred to as ***phase 2*** (post-extraction/pre-transplantation) arguments. If complications arise after transplantation, then RA_j provides *CARREL* with arguments justifying (explaining) *how* the complications resulted in failure (eventually making the organ non-viable), or, conversely, arguments explaining *how* the complications were overcome so as to result in a successful transplant (eventually making the organ viable). These are referred to as ***phase 3*** (post-transplant) arguments and are deemed as providing stronger evidence. Hence, *phase 1* arguments are *presumptive*, submitted prior to undertaking any decision, whereas, *phase 2* and *3* arguments are submitted once the consequences of the decision is known, and so they are *conclusive* or *explanatory* arguments. We now give a shirt example of *phase 1* arguments.

Figure 2a. captures the schemes used by the agents in order to argue over the viability of an offered kidney of a donor d whose cause of death was sve. The argument graph that may result from such deliberation is illustrated in figure 2b. A deliberation must begin with the instantiation of the scheme that captures the decision under debate, the topic of the deliberation. In this case, the instantiation of the Viability Scheme ($A1$ in fig. 2b.). The later submitted arguments will attack $A1$ or reinstate it (see [5] for a more detailed description of the dialog process). Note that in fig. 2b. arguments $A5$ and $A4$ mutually attack each other. This is because the claim of $A5$ *–Recipient r will not result in having svi as a consequence of donor d having sve–* is in contradiction with the statement in $A4$ *–r may result in having svi–*. Intuitively, it remains a moot point as to whether administering *penicillin* is a sufficiently efficacious action for preventing svi in r ($A5$ wins out over $A4$) or not ($A4$ wins out over $A5$). Therefore, it cannot be concluded whether

a)

Scheme ID	Scheme Name	Scheme Description
VS	Viability Scheme	Donor *D* of organ *O* And no contraindications are known for donating organ *O* to recipient *R* Therefore, organ *O* is viable
NVS1	Non-Viability Scheme1	Donor *D* of organ *O* has contraindications *C* for donating organ *O* to recipient Therefore, organ *O* is non-viable
DDTS	Donor Disease Transfer Contraindicatiom Scheme	When transplanting organ *O* from donor *D* with condition *C1* to a recipient *R*, *R* may result in having condition *C2* and *C2* is harmful Hence, *C1* is a contraindication for donating *O*
DCCAPS	Donor Condition Course of Action Prevention Scheme	Following course of action *A* on recipient *R* prevents donor *D*'s condition *C1* resulting in condition *C2* on *R* And *A* is intended Therefore *R* will not result in having *C2* as a consequence of *D* having *C1*

b)

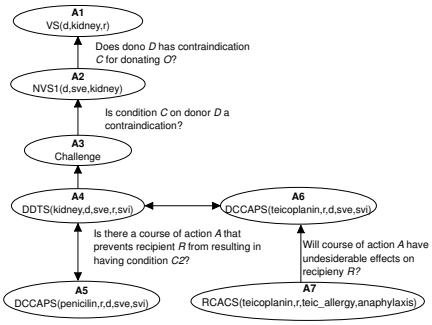


Figure 2. a) Fragment of the schemes in the ASR. b) Argument graph that results from arguing over the viability of a kidney of a donor with sve. The arrows represent the attack relation ($a \rightarrow b$, a attacks b) and the texts of the arrows are critical questions associated to the schemes. We denote $Scheme_i(x_1, .., x_n)$ as an instantiation of the scheme $Sheme_i$, with $x_1, .., x_n$ grounded and preserving the order in which the variables appear in the scheme definition. The graph also depicts the proposal for treating svi with $teicoplanin$, $A6$, but it is defeated by $A7$, the recipient r is $allergic$ to this antibiotic.

the kidney is viable or not. Applying Dung's calculus of opposition to the fig. 2b. graph only $A7$ is evaluated as winning. However, if we take $A5$ to asymmetrically defeat $A4$ (succeeds in its attack at the expense of $A4$'s attack on $A5$) then Dung's winning arguments are $A5$, $A3$ and $A1$. Thus, the organ would be deemed *viable*. But if $A4$ defeats $A5$, then $A4$ and $A2$ win and the organ would be deemed non-viable. In order to resolve this impasse in the argument evaluation, the MA makes use of the three knowledge resources: ACKB, the agents' reputation and the CBRe. Supposing *penicillin* is a novel treatment for preventing svi, the ACKB would not value argument $A5$ as reliable, and so the MA would derive that $A4$ defeats $A5$. However, supposing the agent submitting argument $A5$ has good reputation, $A5$ may be deemed stronger than $A4$, hence the MA would conclude that the kidney is viable[1]. We now describe the CBRe role in resolving conflicting arguments and the ASR structure the case base.

4. The Case-Based Reasoning Engine

Case Based Reasoning (CBR) has proven to be an appropriate reasoning and learning approach for ill-structured domains, where capturing experts' knowledge is difficult and/or the domain theory is weak or incomplete. However, CBR developers still have to face problems such as having to decide *how* to represent a case, *what* are the relevant factors for comparing them and *how* to retain new cases that encode, in a useful way, both the success and failure of the cases' proposed solutions. On the other hand, argumentation has proven to be a suitable approach for reasoning under uncertainty, with inconsistent knowledge sources, and in dialog based communication. However, one unresolved issue in argumentation is *how* to reuse the knowledge encoded in the arguments used in previous dialogs. A few approaches (see [9] and [10]) address this issue by providing support to end users for accessing or retrieving previous stored dialogs. On the other hand, [11]

[1] It is beyond the scope of this paper to discuss the conflict resolution based on the other knowledge resources.

formalizes the way in which arguments used in previous legal cases can be integrated into the current dialog, also represented as a Dung's argument graph.

In this section we propose the use of CBR together with argumentation to: *1)* make use of previous resolved deliberations for evaluating the argument graph resulting from a new deliberation. This amounts to assigning a strength to the submitted arguments and possibly submitting additional arguments deemed relevant in previous similar deliberations; and *2)* organize the case memory by making use of the structure of argument schemes and critical questions encoded in the ASR. We show that in this way, a case can simply be defined as a placeholder for the available data in an experience (*e.g.* a transplant experience) together with the agents' submitted arguments; and that it is these argument graphs associated with each case that provide the means for case comparison. Moreover, these argument graphs represent two aspects of an experience. In the first case they capture the arguments exchanged by the proponent agents in arriving at a decision; thus *presumptive* arguments (*phase 1* arguments). In the second case they capture the *downstream* outcome of actions taken as a result of the decision arrived at in the first case; thus *conclusive* or *explanatory* arguments (*phase 2* and *3* arguments). In this way, the appropriateness of the decision is fed back into the argument graph associated with the case. Hence, the success and failure of a case's proposed solution is given by the dialectical status of the argument representing the decision.

4.1. Cases and Argument-Graphs Representation

Each (transplant) experience constitutes a case. The textual (medical) information describing an experience - the ***case description*** - along with the graph of (presumptive and explanatory) arguments submitted by the agents capture the case's features. In different experiences the arguments given by the agents may be the same, *i.e.* different *cases* may share the same graph (see fig.3). Each argument graph has an associated ***evidential support*** represented by a tuple of natural numbers *(F,K)*. F indicates the degree of certainty in the decision's correctness and K is the number of *cases* that share the argument graph. Thus, graphs with bigger F and K provide stronger evidence. Note that graphs representing cases with no feedback on the decisions' correctness have a more presumptive nature (smaller F) than those whose decision is supported or attacked by factual evidence (bigger F) which are more conclusive, or explanatory in nature. In the transplant scenario this accounts for F being 1,2 or 3 according to the phase in which the transplant experience was resolved. An argument graph may be deemed as having ***sufficient evidential support***, when the evidential support is bigger than a given threshold (*e.g.* $K > 5$).

As described in §3, it may be that the argument graph G of a new case may have nodes connected by bi-directional links, *i.e.*, arguments A and A' mutually attack. One of the CBRe tasks is to decide, on the basis of argument graphs associated with past experiences, whether A defeats A' or *vice versa*, and thus help establish whether a decision should be accepted. Referring to the example in §3, this would involve determining whether the evidence represented by past cases indicates that *penicillin* is ($A5$ defeats $A4$) or is not ($A4$ defeats $A5$) effective in preventing the recipient from contracting *svi*. Another example would be the use of evidence to determine whether or not lung transplants are successful where the donor had smoking history but no COPD. In the next subsection we describe the CBRe's reasoning cycle [12]. That is, the four processes: retrieve, reuse, revise and retain, that enable CBRe to carry out its task. As we will also see, these assume an organization in the case-base memory space given by the ASR.

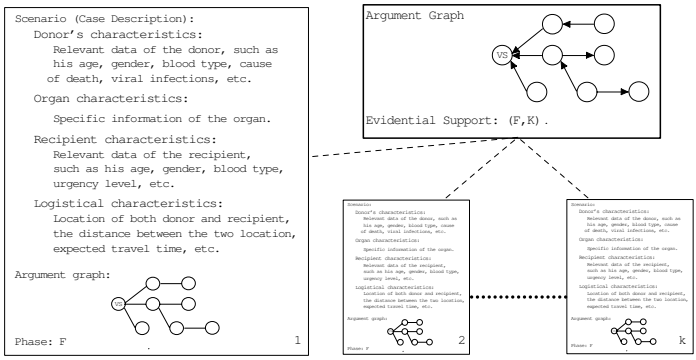


Figure 3. Case and Argument Graph Representation.

4.2. The CBRe Reasoning Cycle

Retrieval: We describe in some detail the first reasoning process in which, given a target problem, the relevant cases for solving it are retrieved from the memory. The relevant cases will be represented by the argument graphs associated to the cases. The relevant graphs to retrieve are those whose arguments apply to the new situation and such that they have sufficient evidential support. The memory from which the *relevant* argument graphs are retrieved is a set M of directed graphs whose nodes are instantiated argument schemes or critical questions of the ASR, and whose edges represent attacks or defeats between arguments allowed by the ASR. Also, every graph $G_i \in M$ contains a single node that captures the topic under debate. In the transplant scenario this account only for the Viability Scheme. In order to facilitate the retrieval process, the memory space is organized on the basis of three partial orderings:

Definition 1 *Let S be defined as the memory space M and let S' be equal to S (containing the 'same' graphs) except that in S' the edges are not directional and nodes are the identifiers of the schemes or critical questions of ASR. Thus, if for example,* VS(d,lung,r) *is a node of a graph in S its correspondent node in S' is* VS. *Let p_S be the canonical projection from S into S'. Given $G1, G2 \in S$, we say that $G2$ structurally contains $G1$, $G1 \preceq_S G2$, if and only if the graph $p_S(G1)$ is a subgraph of $p_S(G2)$.*

Given a new target problem with an associated graph G, the CBRe first identifies those graphs in its memory M that structurally contain G[2], *i.e.* the set $\{G_1, \ldots, G_n\} \subseteq M$ such that for $i = 1 \ldots n$, $G \preceq_S G_i$ (where the set S of Definition 1 is $M \cup \{G\}$). The instantiation of schemes in G_i may differ from the instantiations in G. We wish to retrieve only those G_i whose instantiations are related to that of G as determined by an ontological hierarchy of instantiating terms.

Definition 2 *Let O be the ontology whose terms instantiate the argument schemes of ASR, where O is expressed as an ordering $\prec_T$ on terms, and $\prec_T$ is interpreted as 'more specific than' (*e.g. $svi \prec_T bacterial\_infection \prec_T infection$*)*

[2]Note that these are labelled graphs, moreover they all have a single node representing the topic under debate (*e.g.* Viability Scheme), hence, the graph comparison does not result in a computational overhead.

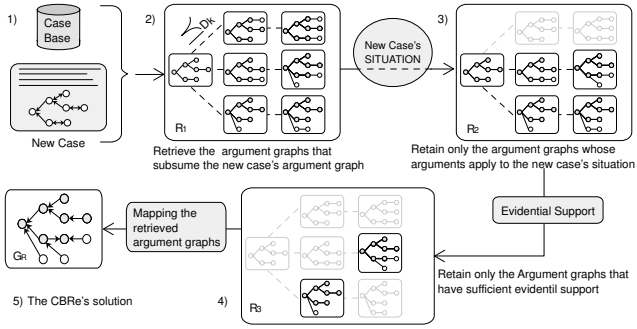


Figure 4. The Retrieval Process: steps 1 until 4. The Reuse Process, step 5

We are only interested in those G_i related to G, where the degree of similarity, or the *distance* between them, falls below a given threshold. To evaluate this, we use the distance between terms in O, denoted as δ_O (*e.g.*, $\delta_O(infection, svi) = 2$), to determine a distance δ_{sch} between scheme instantiations, and so a distance between graphs that share the same structure.

Definition 3 *Let* $G1, G2 \in M$ *such that* $G1 =_S G2$ (i.e. $p_S(G1) = p_S(G2)$) *and* $sch_1 \ldots sch_m$ *be the nodes of both* $p_S(G1)$ *and* $p_S(G2)$. *That is, for* $j = 1 \ldots m$, $sch_j(x_1, .., x_n)$ *is a node in* $G1$ *iff* $sch_j(y_1, .., y_n)$ *is a node in* $G2$. *Then the* ***distance*** *between* $G1$ *and* $G2$ *is given by:* $\delta(G1, G2) = max_{j=1}^{m}(\delta_{sch}(sch_j(x_1, .., x_n), sch_j(y_1, .., y_n)))$, *where* $\delta_{sch}(sch_j(x_1, .., x_n), sch_j(y_1, .., y_n)) = max_{i=1}^{n}(\delta_O(x_i, y_i))$.

We then state a threshold k such that the CBRe retains only those G_i such that $\delta(G, G_i) < k$. [3] To summarize, given a target graph G, CBRe retrieves the set $R_1 = \{G_1 ... G_n\}$ such that for $i = 1...n$, $G \preceq_{D_k} G_i$ (step 2 of fig. 4), where $\preceq_{D_k}$ is defined as follows:

Definition 4 *Let* $G1, G2 \in M$ *such that for some sub-graph* $G3$ *of* $G2$, $G1 =_S G3$ *(hence* $G1 \preceq_S G2$*). Then,* $G1 \preceq_{D_k} G2$ *if* $\delta(G1, G3) < k$.

From R_1, the CBRe excludes the graphs that have arguments that are not applicable in the target case, resulting in the set R_2 (step 3 of fig. 4). For example, a graph G_x in R_1 will not be retained in R_2 if G_x has an argument A_x that assumes the donor has property X which is not true in the target case. This implies searching for property X on the donor in the target case's description. Note that if this property is found in the case's description G_x will remain in R_2 and thus argument A_x. Although not belonging to the target graph it may be deemed relevant for resolving the target case. From the resulting set R_2, the CBRe selects the graphs with *sufficient evidential support* (see sub§4.1), resulting in R_3. At this stage (step 4 of fig. 4), each $G_i \in R_3$ is an argument graph that is applicable to the new case's situation, taking into account all the submitted arguments, and such that it has sufficient evidential support. Therefore, each argument in G_i is *relevant*.

Reusing: The aim of this process is to map R_3 to a solution for the target graph G. All the argument graphs in R_3 are merged into a single graph G_R such that it contains all

[3] Note that the donors and recipients are not relevant for the graph comparison, thus $\delta_O(d, d') = 0$ and $\delta_O(r, r') = 0$ for every two donors d and d' and recipients r and r'.

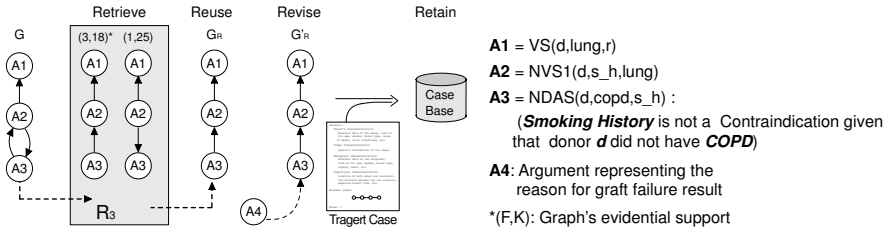


Figure 5. Smoking history example illustrating the CBRe reasoning cycle

the arguments in all graphs in R_3, and therefore in G (step 5 of fig.4), *i.e.* G_R is the minimal graph such that $G_i \preceq_{D_k} G_R$, $G_i \in R_3$. Note that in merging the graphs it may be that there are $G_1, G_2 \in R_3$ such that an argument A asymmetrically defeats A' in G_1 but A' asymmetrically defeats A in G_2. We thus must decide the direction of the defeat (the edge direction) in G_R. Recalling the mutually attacking arguments $A4$ and $A5$ in the target graph shown in fig.2b), this amounts to deciding which argument asymmetrically defeats the other given the previous graphs G_1 and G_2, (where $A4 = A$ and $A5 = A'$).

Suppose that for each edge connecting arguments A and A' of $G_i \in R_3$ such that A asymmetrically defeats A' we associate the evidential support of G_i, writing $ES(G_i, A, A') = (F, K)$. Whereas if A does not asymmetrically defeat A' then $ES(G_i, A, A') = (0, 0)$. Now, for every two connected arguments A, A' in G_R, if $max_{G_i \in R_3}(ES(G_i, A, A'))$ is **sufficiently greater**[4] than $max_{G_i \in R_3}(ES(G_i, A', A))$, then the edge in G_R will go from A to A' indicating that A defeats A'. Otherwise, A and A' will remain connected by a bi-directional edge in G_R indicating a mutual attack, which means there is no sufficient evidence to resolve the conflicting arguments.

Thus, G_R is the CBRe proposed solution, where, as described above, evidential supports are used to determine defeats and so a winning argument for viable or non-viable as described in §3. However, G_R can also determine the decision's validity given additional arguments in G_R that are not in G. G_R may identify additional arguments, not in G, that are applicable to the target case and belonging to a graph with sufficient evidential support. Thus, these additional arguments may identify new relevant factors for deciding the target problem which were not taken into account initially in G. Recall also that G will be a graph constructed from presumptive arguments (*phase 1* arguments), whereas G_R may also contain conclusive arguments (*phases 2* and *3* arguments). As described at the end of §3, *phase 2* arguments in G_R may provide conclusive evidence supporting a final decision for non-viability. *Phase 3* arguments may provide conclusive evidence supporting a final decision for non-viability or viability (*e.g.*, arguments describing post-transplant procedures that unsuccessfully, respectively successfully, dealt with post-transplant complications). To summarize, G_R provides: 1) evidential support to determine defeats amongst arguments in G and so determine the decision's validity; 2) new arguments for determining the decision's viability; 3) additional arguments that may serve, for example, as guidance to the RA_j for post-transplant management of patients.

Revising: The solution G_R must be tested in the real world, and if necessary, revised. This is achieved by requiring the agents to continue submitting arguments to G_R until

[4]The definition of 'sufficiently greater' is domain dependent. *E.g.*, $(F1, K1)$ is sufficiently greater than $(F2, K2)$ if : a)$F1 > F2$, or b) $F1 = F2$ and $F1 > 1$, $F2 > 1$, and $K1 * \alpha > K2$, with $0 < \alpha < 1$.

the (transplant) experience ends. For example, if in the smoking history example the lung is deemed viable in G_R (see fig.5) but there is a graft failure the reasons for the failure will be submitted as new argument $A4$, that will reinstate the argument for non-viability. The resulting updated argument graph G'_R will then be stored in the case base.

Retain: The aim is to store the possibly updated G'_R as a new graph in the memory. Hence, when a (transplant) experience finishes, the *case* describing this experience is retained by the CBRe. If there already exists an argument graph G_M in the memory such that $G'_R =_{D_k} G_M$ and the edges directions coincide, then the *case* is associated with G_M increasing G_M's evidential support. Otherwise, the target case is retained as associated to G'_R which is added as a new argument graph to M.

5. Conclusions and Future Work

In this paper we have proposed *1)* the use of CBR in order to evaluate the evidential support (and thus relative strength) of the agents' submitted arguments, which helps to resolve the impasse of having arguments that mutually attack each other; and *2)* the use of argument schemes and critical questions to organize the CBR's memory space, which enables comparison of cases on the basis of the submitted arguments.

This use of CBR is described as part of the *ProCLAIM* model intended for agents to argue over the validity of their intended decisions. We have shown its practical value in assessing the viability of organs for transplantation. The work described furthers our eventual objective [2], *vis-a-vis.* to increase the number of human organs current selection processes make available for transplantation.

Other works that have combined argumentation with cases can be found in the legal domain, particularly in the context of the *Common law system*, a legal system based on unwritten laws developed through judicial decisions that create binding precedent. The inherent argumentative nature of the legal domain and the particular features of the Common law system provide a scenario for developing models and systems for reasoning and arguing with precedents, *i.e.* past cases. Exponents of these works are systems such as HYPO[13] and CABARET [14] which assist users in constructing arguments from cases[5]. Intended for the same purposes is the extension to the HERMES System proposed in [10] that aims to support human agents involved in group decision making processes to retrieve, adapt and re-use past cases.

We are currently prototyping the CBRe so as to extend an existing prototype of the logical argumentation model described in [3]. This work is intended as a precursor to development of a robust large scale demonstrator with embedded argumentation components developed by the EU 6^{th} framework project ASPIC[6] (Argumentation Services Platform with Integrated Components). Future work will focus on extending the retrieval process so as to address adaptation of previous cases in order to increase the scope of the relevant cases. Another future line of work is use of the case base for searching patterns in order to propose new arguments, *i.e.* to propose new instantiations of argument schemes (*e.g.* relating a donor condition x with unsuccessful transplants: NVS1(d,x,$organ$)).

[5]It could be argued that these works are best described as systems or models for arguing with cases, rather than CBR systems in the sense of [12]. Moreover, systems such as HYPO, do not define any kind of automated procedure for retaining new cases.

[6]www.argumentation.org

The transplant scenario serves to illustrate *ProCLAIM*'s practical value. We believe *ProCLAIM*, and in particular the CBRe, may also prove to be useful in other safety-related environments. We are currently investigating the application of *ProCLAIM* as an extension to *DAI-DEPUR* [15], a decision support systems for Wastewater Treatment Plants (wwtp). In this scenario, the proponent agents would represent the wwtp operators, the Argument Source Manager would relate to the operators' hierarchy within the plant and the GK would be instantiated by the guidelines encoding compliance with the environmental legislations. The CBRe will help to establish on an evidential basis, indicating whether the operators' decisions are appropriate and thus environmentally safe, in light of their given arguments.

Acknowledgments This paper was supported in part by the Grant FP6-IST-002307 (ASPIC).

References

[1] Walton, D. N. *Argumentation Schemes for Presumptive Reasoning*, Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.
[2] P. Tolchinsky, U. Cortés, J. C. Nieves, F. Caballero and A. López-Navidad, Using Arguing Agents to increase the Human Organ Pool for Transplantation, In *3rd Workshop on Agents Applied in Health Care (IJCAI-05)*, 2005.
[3] S. Modgil, P. Tolchinsky and U. Cortés. Towards Formalising Agent Argumentation over the Viability of Human Organs for Transplantation, In *MICAI*, pages 928-938, 2005.
[4] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. In *Artificial Intelligence*, 1995.
[5] P. Tolchinsky, S. Modgil and U. Cortés. Argument Schemes and Critical Questions for Heterogeneous Agents to Argue Over the Viability of a Human Organ. In *AAAI 2006 Spring Symposium Series; Argumentation for Consumers of Healthcare*
[6] A. López-Navidad and F. Caballero. Extended criteria for organ acceptance: Strategies for achieving organ safety and for increasing organ pool. In *Clin Transplant, Blackwell Munksgaard*, 17:308-324, 2003.
[7] ONT, Organización Nacional de Transplantes. http://www.ont.es
[8] F. Caballero, A. López-Navidad, M. Perea, C. Cabrer, L. Guirado, R. Solà". Successful Liver and Kidney Transplantation From Cadaveric Donors With Left-Sided Bacterial Endocarditis. In *American Journal of Transplantation*, 5(4):781-787, 2005.
[9] S. Buckingham Shum. Design Argumentation as Design Rationale. In *The Encyclopedia of Computer Science and Technology*, 35:95-128, 1996.
[10] N. Karacapilidis, B. Trousse and D. Papadias. Using Case-Based Reasoning for Argumentation with Multiple Viewpoints. In *ICCBR 1997*.
[11] T.J.M. Bench-Capon. Representation of case law as an argumentation framework. In *Legal Knowledge and Information Systems. JURIX 2002*, pages 103–112.
[12] A. Aamodt and E. Plaza. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In *AI Commun*, 7(1):39-59, 1994.
[13] K. D. Ashley. Reasoning with Cases and Hypotheticals in HYPO. In *International Journal of Man-Machine Studies*, 34(6):753-796, 1991.
[14] D. B. Skalak and E. L. Rissland. Arguments and Cases: An Inevitable Intertwining. In *Artificial Intelligence and Law*, 1(1):3–44, 1992.
[15] U. Cortés, I. R.-Roda, M. Sànchez-Marrè, J. Comas, C. Cortes and M. Poch. DAI-DEPUR: An Environmental Decision Support System for the control and supervison of Municipal WasteWater Treatment Plants. In *ECAI*, pages 603-607, 2002.

Computational Properties of Argument

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Complexity Properties of Restricted Abstract Argument Systems

Paul E. Dunne
Department of Computer Science, The University of Liverpool, U.K.

Abstract. One difficulty that arises in abstract argument systems is that many natural questions regarding argument acceptability are, in general, computationally intractable having been classified as complete for classes such as NP, co-NP, and Π_2^p. In consequence, a number of researchers have considered methods for specialising the structure of such systems so as to identify classes for which efficient decision processes exist. In this paper the effect of a number of graph-theoretic restrictions is considered. For the class of *bipartite* graphs, it is shown that determining the acceptability status of a *specific* argument can be accomplished in polynomial time under both credulous and sceptical semantics. In contrast to these positive results, however, deciding whether an arbitrary *set* of arguments is "collectively acceptable" remains NP–complete in bipartite systems. In addition, a construction is presented by means of which questions posed of arguments in any given finite argument system may be expressed as questions within a related system in which every argument attacks and is attacked by *at most* two arguments. It follows that bounding the number of attacks on individual arguments is unlikely to produce a computationally more tractable environment.

Keywords. Computational properties of argumentation; argumentation frameworks; computational complexity

1. Introduction

Since their introduction in the seminal work of Dung [1] abstract argument systems have proven to be a valuable paradigm with which to formalise divers semantics defining argument "acceptability". In these a key component is the concept of an "*attack*" relationship wherein the incompatibility of two arguments – p and q, say – may be expressed in terms of one of these "attacking" the other: such relationships may be presented independently of any internal structure of the individual arguments concerned so that the properties of the overall argument system, e.g. which of its arguments may be defended against any attack and which are indefensible, depend solely on the attack relationship rather than properties of individual argument schemata. Among other applications, this abstract view of argumentation has been demonstrated to be a powerful and flexible approach to modelling reasoning in a variety of non-classical logics, e.g. [1,2,3].

We present the formal definitions underpinning argument systems in Section 2, including two of the widely-studied admissibility semantics – preferred and stable – introduced in [1]: at this point we simply observe that these describe differing conditions which a maximal set of mutually compatible arguments, S, must satisfy in order to be ad-

missible within some argument system comprising arguments $\mathcal{X}$ with attack relationship $\mathcal{A} \subset \mathcal{X} \times \mathcal{X}$.

Despite the descriptive power offered by abstract argument systems one significant problem is the apparent intractability of many natural questions concerning acceptability under all but the most elementary semantics: such intractability classifications encompassing NP–completeness and co-NP–completeness results of Dimopoulos and Torres [4] and the Π_2^p–completeness classifications presented in Dunne and Bench-Capon [5]. Motivated, at least to some degree, by these negative results a number of researchers have considered mechanisms by which argument systems may be specialised or enriched so that the resulting structures admit efficient decision procedures. Two main strategies are evident: the first, and the principal focus of the present paper, has been to identify purely graph-theoretic conditions leading to tractable methods for those cases within which these are satisfied; the second, which itself may be coupled with graph-theoretic restrictions, is to consider additional structural aspects in developing the basic argument and attack relationship form. Under the first category, [1] already identifies *directed acyclic graphs* (DAGs) as a suitable class, while recent work of Coste-Marquis *et al.* [6] has shown that *symmetric* argument systems – those in which p attacks q if and only if q attacks p – also form a tractable class. Graph-theoretic considerations also feature significantly in work of Baroni *et al.* [7,8].

Probably the two most important exemplars of the second approach are the *Preference–based* argumentation frameworks of Amgoud and Cayrol [9] and *Value–based* argumentation frameworks introduced by Bench-Capon [10]. While the supporting motivation for both formalisms is, perhaps, more concerned with providing interpretations and resolution of issues arising from the presence of multiple maximal admissible sets which are mutually incompatible, both approaches start with an arbitrary argument system, $\langle \mathcal{X}, \mathcal{A} \rangle$, and reduce it to an *acyclic* system, $\langle \mathcal{X}, \mathcal{B} \rangle$ in which $\mathcal{B} \subseteq \mathcal{A}$. This reduction is determined via some additional relationship $\mathcal{R}$ with the main distinction between [9] and [10] being the exact manner in which $\mathcal{R}$ is defined.

In this paper some further classes of graph-theoretic restrictions are considered: the classes of *k–partite* directed graphs and those of *bounded degree*. In the former class, for which the case $k = 2$ is of particular interest, the arguments $\mathcal{X}$ may be partitioned into k pairwise disjoint subsets – $\langle \mathcal{X}_1, \ldots, \mathcal{X}_k \rangle$ – such that every attack in $\mathcal{A}$ involves arguments belonging to *different* sets in this partition: the special case, $k = 2$, defines the class of *bipartite* directed graphs. In the bounded degree class, the number of attacks on (the argument's *in-degree*) and attacks made by (the argument's *out-degree*) any $x \in \mathcal{X}$, i.e. $|\{y \ : \ \langle y, x \rangle \in \mathcal{A}\}|$ and $|\{y \ : \ \langle x, y \rangle \in \mathcal{A}\}|$ are bounded by given values (p, q), again the special case $p = q = 2$ is of particular interest.

In the remainder of this paper formal background and definitions are given in Section 2 together with the decision questions considered. Sections 3 and 4 present the main body of results concerning, respectively, k-partite and bounded degree directed graphs. Conclusions and developments are discussed in Section 5.

2. Finite Argument Systems – Basic Definitions

The following concepts were introduced in Dung [1].

Definition 1 *An* argument system *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'. The convention of excluding "self-attacking" arguments, also observed in [6], 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 system $\mathcal{H}(\mathcal{X}, \mathcal{A})$, we say that*

a. *$s \in S$ is* attacked *by R if there is some $r \in R$ such that $\langle r, s \rangle \in \mathcal{A}$.*
b. *$x \in \mathcal{X}$ 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.*
c. *S is* conflict-free *if no argument in S is attacked by any other argument in S.*
d. *A conflict-free set S is* admissible *if every $y \in S$ is acceptable w.r.t S.*
e. *S is a* preferred extension *if it is a maximal (with respect to $\subseteq$) admissible set.*
f. *S is a* stable extension *if S is conflict free and every $y \notin S$ is attacked by S.*
g. *$\mathcal{H}$ is* coherent *if every preferred extension in $\mathcal{H}$ is also a stable extension.*

Following the terminology of [6], $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is symmetric *if for every pair of arguments x, y in $\mathcal{X}$ it holds that $\langle x, y \rangle \in \mathcal{A}$ if and only if $\langle y, x \rangle \in \mathcal{A}$.*

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.*

We make one further assumption regarding the *graph-theoretic* structure of argument systems: as an *undirected* graph, $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is *connected*. In informal terms, this states that the systems considered do *not* consist of two or more "isolated" systems.

The concepts of credulous and sceptical acceptance motivate the following decision problems that have been considered in [4,5].

Decision Problem	Instance	Question
CA	$\mathcal{H}(\mathcal{X}, \mathcal{A}), x \in \mathcal{X}$	Is x credulously accepted in $\mathcal{H}$?
SA	$\mathcal{H}(\mathcal{X}, \mathcal{A}), x \in \mathcal{X}$	Is x sceptically accepted in $\mathcal{H}$?

The questions above are formulated in terms of *single* arguments, it will be useful to consider analogous concepts with respect to *sets*. Thus $\text{CA}_{\{\}}$ denotes the decision problem whose instances are an argument system $\langle \mathcal{X}, \mathcal{A} \rangle$ together with a subset S of $\mathcal{X}$: the instance being accepted if there is a preferred extension T for which $S \subseteq T$. Similarly, $\text{SA}_{\{\}}$ accepts instances for which S is a subset of *every* preferred extension.

The results of [4] establish that CA is NP–complete, while [5] proves SA to be Π_2^p–complete.

In contrast, we have the following more positive results.

Fact 1

a. *Every argument system $\mathcal{H}$ has* at least one *preferred extension. (Dung [1])*
b. *If $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is a* DAG *then $\mathcal{H}$ has a* unique *preferred extension. This is also a* stable *extension and may be found in time linear in $|\mathcal{X}| + |\mathcal{A}|$. ((Dung [1])*
c. *If $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is* symmetric *then* CA, SA, $\text{CA}_{\{\}}$, *and* $\text{SA}_{\{\}}$ *are all polynomial time decidable. Furthermore $\mathcal{H}$ is coherent. (Coste-Marquis* et al. *[6]).*
d. *If $\mathcal{H}(\mathcal{X}, \mathcal{A})$ contains no* odd-length *simple directed cycles, then $\mathcal{H}$ is coherent. (Dunne and Bench-Capon [5])*
e. *If $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is coherent then $\text{SA}(\mathcal{H}, x)$ can be decided in co-*NP.

Fact 1 (e) is an easy consequence of the sceptical acceptance methods described in work of Vreeswijk and Prakken [11].

While Fact 1 (a) ensures the existence of a preferred extension – a property that is not guaranteed to be the case for stable extensions – it is possible that the *empty set* of arguments (which is always admissible) is the unique such extension. Whether a given argument system $\mathcal{H}(\mathcal{X}, \mathcal{A})$ has a non-empty preferred extension is unlikely to be efficiently decidable in general: [4] showing this decision problem to be NP–complete.

We also consider the effect that restricting the underlying graph structure has with respect to the value-based argument systems of [10].

Definition 2 *A* value-based argumentation framework *(*VAF*), is defined by a triple* $\langle \mathcal{H}(\mathcal{X}, \mathcal{A}), \mathcal{V}, \eta \rangle$*, where* $\mathcal{H}(\mathcal{X}, \mathcal{A})$ *is an argument system,* $\mathcal{V} = \{v_1, v_2, \ldots, v_k\}$ *a set of* k 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* α *is a* specific *audience if* α *yields a* total *ordering of* $\mathcal{V}$*.*

Using VAFs, ideas analogous to admissible argument in standard argument systems are defined in the following way. Note that all these notions are now relative to some audience.

Definition 3 *Let* $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$ *be a* VAF *and* $\mathcal{R}$ *an audience.*

a. *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* $\eta(y) \succ_{\mathcal{R}} \eta(x)$*.*
b. *An argument* x *is* acceptable to the subset S *with respect to an audience* $\mathcal{R}$ *if: for every* $y \in \mathcal{X}$ *that* successfully attacks x *with respect to* $\mathcal{R}$*, there is some* $z \in S$ *that successfully attacks* y *with respect to* $\mathcal{R}$*.*
c. *A subset* S *of* $\mathcal{X}$ *is* conflict-free with respect to the audience $\mathcal{R}$ *if: for each* $\langle x, y \rangle \in S \times S$*, either* $\langle x, y \rangle \notin \mathcal{A}$ *or* $\eta(y) \succ_{\mathcal{R}} \eta(x)$*.*
d. *A subset* S *of* $\mathcal{X}$ *is* admissible *with respect to the audience* $\mathcal{R}$ *if:* S *is conflict free with respect to* $\mathcal{R}$ *and every* $x \in S$ *is acceptable to* S *with respect to* $\mathcal{R}$*.*
e. *A subset* S *is a* preferred extension *for the audience* $\mathcal{R}$ *if it is a maximal admissible set with respect to* $\mathcal{R}$*.*
f. *A subset* S *is a* stable extension *for the audience* $\mathcal{R}$ *if* S *is admissible with respect to* $\mathcal{R}$ *and for all* $y \notin S$ *there is some* $x \in S$ *which successfully attacks* y *with respect to* $\mathcal{R}$*.*

Bench-Capon [10] proves that every specific audience, α, induces a unique preferred extension within its underlying VAF: we use $P(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle, \alpha)$ to denote this extension. Analogous to the concepts of credulous and sceptical acceptance, in VAFs the ideas of *subjective* and *objective* acceptance arise,

Subjective Acceptance (SBA)
Instance: A VAF $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$; argument $x \in \mathcal{X}$;
Question: Is there a specific audience, α for which $x \in P(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle, \alpha)$?

Objective Acceptance (OBA)
Instance: A VAF $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$; argument $x \in \mathcal{X}$;
Question: Is $x \in P(\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle, \alpha)$ for *every* specific audience α.

Regarding these questions, Dunne and Bench-Capon [12] show the former to be NP–complete and the latter co-NP–complete.

In the remainder of this paper attention will focus on the restricted forms of argument system described in the following definition.

Definition 4

a. *An argument system* $\mathcal{H}(\mathcal{X}, \mathcal{A})$ *is* k-partite *if there is a partition of* $\mathcal{X}$ *into* k *sets* $\langle \mathcal{X}_1, \ldots, \mathcal{X}_k \rangle$ *such that*

$$\forall \langle y, z \rangle \in \mathcal{A} \quad y \in \mathcal{X}_i \Rightarrow z \notin \mathcal{X}_i$$

The term bipartite *will be used for the case* $k = 2$*. It should be noted that, since there is no insistence that each of the partition members be non-empty, any* k*-partite system is, trivially, also a* $(k + t)$*-partite system for every* $t \geq 0$*. We use the notation* $\Gamma^{(k)}$ *for the set of all* k*-partite argument systems.*

b. *An argument system* $\mathcal{H}(\mathcal{X}, \mathcal{A})$ *has* (p, q)-bounded degree *if*

$$\forall\, x \in \mathcal{X} \;\; |\{\, y \in \mathcal{X} \;:\; \langle y, x \rangle \in \mathcal{A} \,\}| \leq p \text{ and}$$
$$|\{\, y \in \mathcal{X} \;:\; \langle x, y \rangle \in \mathcal{A} \,\}| \leq q$$

The notation $\Delta^{(p,q)}$ *will be used for the set of all* (p, q)*-bounded degree systems.*

The notations $\text{CA}^{(k)}$, $\text{SA}^{(k)}$, $\text{CA}^{(k)}_{\{\}}$, and $\text{SA}^{(k)}_{\{\}}$ (similarly $\text{SBA}^{(k)}$ and $\text{OBA}^{(k)}$ for the case of VAFs) will be used to distinguish the various avatars of the decision problems of interest when instances are required to be k-partite argument systems. Similarly we use $\text{COHERENT}^{(k)}$ to denote the problem of deciding whether a k-partite argument system is coherent. In instances of these problems it is assumed that $\mathcal{H}(\mathcal{X}, \mathcal{A})$ is presented using an appropriate partition of $\mathcal{X}$ into k disjoint sets $\langle \mathcal{X}_1, \ldots, \mathcal{X}_k \rangle$.[1] Table 1 below summarises the properties of k-partite systems proved in Section 3, where it is noted that *lower* bounds on the complexity of problems for k-partite systems are also lower bounds for $(k + t)$-partite systems.

3. Bipartite and k-partite Argument Systems

In this section we consider the effect on problem complexity of restricting systems to be k-partite for both standard Dung-style argument systems and the development of these described by VAFs. The results indicated in Table 1 are presented in the opening subsection and the case of VAFs, in particular the complexity of $\text{SBA}^{(k)}$ and $\text{OBA}^{(k)}$, is examined in the second part.

[1] Without this, problems arise when checking if an *arbitrary* argument system, $\mathcal{H}$, is k-partite: for $k \geq 3$ the corresponding decision question is NP–complete.

	Decision Problem	Complexity
a.	$\text{CA}^{(2)}$	Polynomial-time
b.	$\text{CA}^{(3)}$	NP–complete
c.	$\text{CA}^{(2)}_{\{\}}$	NP–complete
d.	$\text{SA}^{(2)}$	Polynomial-time
e.	$\text{SA}^{(4)}$	Π^p_2–complete
f.	$\text{SA}^{(2)}_{\{\}}$	Polynomial-time
g.	$\text{SA}^{(4)}_{\{\}}$	Π^p_2–complete
h.	$\text{COHERENT}^{(2)}$	Trivial
i.	$\text{COHERENT}^{(4)}$	Π^p_2–complete

Table 1. Complexity-theoretic Properties of k-partite Argument Systems

3.1. k-partite Dung-style Argument Systems

We first deal with the case of bipartite argument systems ($k = 2$). For other values it is noted that the classifications are largely straightforward consequences of the graph-theoretic constructions in [4,5]. Notice that it is straightforward to deal with the claim made in Table 1(h): a bipartite argument system cannot have any odd-length cycles, and thus coherence is ensured via Fact 1 (d). In contrast to *undirected* graph structures, the *absence* of odd-length directed cycles, while necessary, is not a *sufficient* condition for an argument system to be bipartite; *symmetric* systems, however, are bipartite systems if and only if the associated undirected graph contains no odd-length cycles.

The main idea underlying the algorithm of Theorem 1 is as follows: in a bipartite argument system, $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$ attackers of an argument $y \in \mathcal{Y}$ can only be arguments $z \in \mathcal{Z}$, and defences to such attacks must, themselves, also be arguments in $\mathcal{Y}$. It follows, therefore, that those arguments of $\mathcal{Y}$ that are attacked by members of $\mathcal{Z}$ upon which no counterattack exists cannot be admissible. Moreover, attacks on $\mathcal{Z}$ furnished by such arguments play no useful function (as counterattacks) and may be eliminated from $\mathcal{A}$, a process that can lead to further arguments in $\mathcal{Z}$ becoming unattacked. By iterating the process of removing indefensible arguments in $\mathcal{Y}$ and their associated attacks on $\mathcal{Z}$, this algorithm identifies an admissible subset of $\mathcal{Y}$.

Theorem 1

a. $\text{CA}^{(2)}$ *is polynomial time decidable.*
b. $\text{SA}^{(2)}$ *is polynomial time decidable.*

Proof: For (a), given a bipartite argument system, $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$ and $x \in \mathcal{Y} \cup \mathcal{Z}$, without loss of generality assume that $x \in \mathcal{Y}$. Consider the subset, S of $\mathcal{Y}$ that is formed by the following algorithm.

1. $i := 0$; $\mathcal{Y}_0 := \mathcal{Y}$; $\mathcal{A}_0 := \mathcal{A}$
2. **repeat**

   2.1 $i := i + 1$
   2.2 $\mathcal{Y}_i := \mathcal{Y}_{i-1} \setminus \{ y \in \mathcal{Y}_{i-1} : \exists z \in \mathcal{Z} : \langle z, y\rangle \in \mathcal{A}_{i-1}$ and $|\{ y \in \mathcal{Y}_{i-1} : \langle y, z\rangle \in \mathcal{A}_{i-1}\}| = 0 \}$
   2.3 $\mathcal{A}_i := \mathcal{A}_{i-1} \setminus \{ \langle y, z\rangle : y \notin \mathcal{Y}_i \setminus \mathcal{Y}_{i-1} \}$

until $\mathcal{Y}_i = \mathcal{Y}_{i-1}$

3. **return** $\mathcal{Y}_i$

If S is the subset of $\mathcal{Y}$ returned we claim that $\text{CA}^{(2)}(\mathcal{B}, x)$ holds if and only if $x \in S$.

Suppose first that $x \in S \subseteq \mathcal{Y}$. Since $\mathcal{B}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$ is a bipartite argument system it follows that S is conflict-free. Now consider any argument $z \in \mathcal{Z}$ that attacks S: it must be the case that there is some $y \in S$ that counterattacks z for otherwise at least one argument would have been removed from S at Step(2.2). In total, S is conflict-free and every argument in S is acceptable with respect to S, i.e. S is an admissible set containing x which is, hence, credulously accepted.

On the other hand, suppose that x is credulously accepted. Let S be the subset of $\mathcal{Y}$ returned and suppose for the sake of contradiction that $x \notin S$: then there must be some iteration of the algorithm during which $x \in \mathcal{Y}_{i-1}$ but $x \notin \mathcal{Y}_i$. In order for this to occur, we must have a sequence of arguments $\langle z_0, z_1, \ldots, z_i \rangle$ in $\mathcal{Z}$ with the property that $|\{ y \in \mathcal{Y}_j : \langle y, z_j \rangle \in \mathcal{A}_j \}| = 0$ with $\langle z_i, x \rangle \in \mathcal{A}_i$. Now any argument y' of $\mathcal{Y}$ attacked by z_0 cannot be credulously accepted since there is no counterattack on z_0 available. It follows that the attacks $\langle y', z \rangle$ provided by such arguments cannot play an effective role in defending another argument and thus can be removed. Continuing in this way, it follows that no argument y'' that is attacked by z_1 is credulously accepted: the only attackers of z_1 are arguments of $\mathcal{Y}$ that are attacked by z_0 and these, we have seen, are indefensible. In total, $x \notin S$ would imply that x is indefensible, a conclusion which contradicts the assumption that x was credulously accepted.

The preceding analysis establishes the algorithm's correctness. The proof of (a) is completed by noting that it runs in polynomial time: there are at most $|\mathcal{Y}|$ iterations of the main loop each taking only polynomially many (in $|\mathcal{Y} \cup \mathcal{Z}| + |\mathcal{A}|$) steps.

Part (b) follows from (a), Table 1(h) and the observation of [11] that, in coherent systems, an argument is sceptically accepted if and only if all of its attackers fail to be credulously accepted. □

Turning to the problems $\text{CA}_{\{\}}$ and $\text{SA}_{\{\}}$, [6] note that in many cases decision problems involving *sets* are "no harder" than the related questions formulated for specific arguments, e.g. for unrestricted argument systems, symmetric argument systems and DAGs, the upper bounds for $\text{CA}_{\{\}}$ and $\text{SA}_{\{\}}$ are identical to the corresponding upper bounds for CA and SA. In this light, the next result may appear somewhat surprising: although, as has just been shown, $\text{CA}^{(2)}$ is polynomial time decidable, $\text{CA}^{(2)}_{\{\}}$ is likely to be noticeably harder.

Theorem 2

a. $\text{CA}^{(2)}_{\{\}}$ *is* NP*–complete, even for sets containing exactly* two *arguments.*

b. $\text{SA}^{(2)}_{\{\}}$ *is polynomial time decidable.*

Proof: For (a), that $\text{CA}^{(2)}_{\{\}} \in \text{NP}$ is easily demonstrated via the non-deterministic algorithm that guesses a subset T, checks $S \subseteq T$ and that T is admissible.

To show that $\text{CA}^{(2)}_{\{\}}$ is NP–hard we use a reduction from the problem *Monotone 3-CNF Satisfiability* (MCS) ([13, p. 259]), instances of which comprise a 3-CNF formula over a set of propositional variables $\{x_1, \ldots, x_n\}$,

$$\Phi(x_1, x_2, \ldots, x_n) \;=\; \bigwedge_{i=1}^{m} C_i \;=\; \bigwedge_{i=1}^{m} (\, y_{i,1} \,\vee\, y_{i,2} \,\vee\, y_{i,3} \,)$$

and each clause, C_i, is defined using exactly three *positive* literals or exactly three *negated* literals, e.g. $(x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_4)$ would define a valid instance of MCS, however $(x_1 \vee \neg x_2 \vee x_3)$ would not. An instance Φ of MCS is accepted if and only if there is an instantiation, $\alpha \in \langle \top, \bot \rangle^n$ under which $\Phi(\alpha) = \top$.

Given $\Phi(x_1, \ldots, x_n)$ an instance of MCS let $\{C_1^+, \ldots, C_r^+\}$ be the subset of its clauses in which only positive literals occur and $\{D_1^\neg, \ldots, D_s^\neg\}$ those in which only negated literals are used. Consider the bipartite argument system $\mathcal{B}_{\text{MCS}}(\mathcal{Y}, \mathcal{Z}, \mathcal{A})$ in which

$$\begin{array}{l} \mathcal{Y} \;=\; \{\Phi^\neg,\; C_1^+, \ldots, C_r^+,\; \neg x_1, \ldots, \neg x_n\} \\ \mathcal{Z} \;=\; \{\Phi^+,\; D_1^\neg, \ldots, D_s^\neg,\; x_1, \ldots, x_n\} \end{array}$$

and $\mathcal{A}$ contains

$$\begin{array}{ll} \{\, \langle x_j, \neg x_j \rangle,\; \langle \neg x_j, x_j \rangle \;:\; 1 \;\leq j \leq n\} & \bigcup \\ \{\, \langle C_i^+, \Phi^+ \rangle \;:\; 1 \leq i \leq r\} \;\cup\; \{\, \langle D_i^\neg, \Phi^\neg \rangle \;:\; 1 \leq i \leq s\} & \bigcup \\ \{\, \langle \neg x_j, D_i^\neg \rangle \;:\; \neg x_j \text{ occurs in } D_i^\neg\} & \bigcup \\ \{\, \langle x_j, C_i^+ \rangle \;:\; x_j \text{ occurs in } C_i^+\} & \end{array}$$

The instance of $\text{CA}_{\{\}}^{(2)}$ is completed by setting $S = \{\Phi^+, \Phi^\neg\}$.

Suppose that there is some preferred extension, T, of $\mathcal{B}_{\text{MCS}}$ for which $\{\Phi^+, \Phi^\neg\} \subseteq T$, i.e. that $\langle \mathcal{B}_{\text{MCS}}, S \rangle$ defines a positive instance of $\text{CA}_{\{\}}^{(2)}$. Then, for each C_i^+ some argument x_j with $\langle x_j, C_i^+ \rangle \in \mathcal{A}$ must be in T (otherwise the attack $\langle C_i^+, \Phi^+ \rangle$ is undefended); similarly for each $D_i^\neg$ some argument $\neg x_k$ with $\langle \neg x_k, D_i^\neg \rangle \in \mathcal{A}$ must be in T. It cannot be the case, however, that *both* x_j and $\neg x_j$ are in T. We can, thus, construct a satisfying instantiation of Φ via $x_j := \top$ if $x_j \in T$, and $x_j := \bot$ if $\neg x_j \in T$.

On the other hand suppose the instance Φ of MCS is satisfiable, using some instantiation α. In this case the set

$$\{\Phi^+, \Phi^\neg\} \;\cup\; \{\, x_j^+ \;:\; x_j = \top \text{ under } \alpha\} \;\cup\; \{\, x_j^\neg \;:\; x_j = \bot \text{ under } \alpha\}$$

is easily seen to be admissible, so that $\langle \mathcal{B}_{\text{MCS}}, \{\Phi^+, \Phi^\neg\} \rangle$ defines a positive instance of $\text{CA}_{\{\}}^{(2)}$.

Part (b) follows easily from Theorem 1(b) since a set of arguments S is sceptically accepted if and only if each of its constituent members is sceptically accepted. □

The remaining cases in Table 1 are considered in the following Theorem.

Theorem 3

a. $\forall\, k \geq 3$, $\text{CA}^{(k)}$ *and* $\text{CA}_{\{\}}^{(k)}$ *are* NP*–complete.*

b. $\forall\, k \geq 4$, $\text{SA}^{(k)}$, $\text{SA}_{\{\}}^{(k)}$ *and* $\text{COHERENT}^{(k)}$ *are* Π_2^p*–complete.*

Proof: (*Outline*) The membership proofs are identical to those that hold for the unrestricted versions of each problem. For the cases in (a), NP–hardness follows by observing

that the argument system given in [14, Defn. 7, p. 234][2] from which NP–hardness of CA and $\text{CA}_{\{\}}$ is deduced, is 3-partite. Similarly, for (b), after applying the translation described in [5] to "suitable" quantified Boolean formulae,[3] the resulting argument system is 4-partite. □

We can, in fact, demonstrate that Thm 3(b) holds for $k \geq 3$: the proof, however, requires a non-trivial application of techniques introduced in Section 4 applied to the construction from [5] and is omitted here.

3.2. k-partite Value-Based Argumentation Frameworks

In Theorem 1 it was shown that $\text{CA}^{(2)}$ and $\text{SA}^{(2)}$ are solvable by efficient algorithmic methods, a property which continues to hold for $\text{SA}^{(2)}_{\{\}}$ even though $\text{CA}^{(2)}_{\{\}}$ becomes as hard as the unrestricted version. While there is some superficial similarity between the concepts of credulous (resp. sceptical) acceptance and subjective (resp. objective) acceptance, as is demonstrated in Bench-Capon *et al.* [15, Thm. 12], it is not possible, in general, to deduce whether an argument is subjectively accepted in a given VAF, $\langle \mathcal{X}, \mathcal{A}, \mathcal{V}, \eta \rangle$, from knowledge of whether or not it is credulously accepted in the (value-free) system $\langle \mathcal{X}, \mathcal{A} \rangle$. Nevertheless, one might hope that since bipartite systems offer a tractable class within standard argument systems, so too bipartite VAFs would admit efficient processes with which to determine subjective and and objective acceptance. In fact, as we show in Theorem 4, in complete contrast to the standard framework the problems $\text{SBA}^{(2)}$ and $\text{OBA}^{(2)}$ are as hard as their unrestricted counterparts, SBA and OBA.

Theorem 4

a. $\text{SBA}^{(2)}$ *is* NP*–complete.*
b. $\text{OBA}^{(2)}$ *is co-*NP*–complete.*

Proof: (Outline) Membership in NP (resp. co-NP) is via an identical argument to that used for the unrestricted versions as given in [12]. To show that $\text{SBA}^{(2)}$ is NP–hard we, again, use a reduction from the problem MCS introduced in the proof of Theorem 2(a). Given an instance $\Phi = \wedge_{i=1}^{m} C_i$ of MCS, consider the bipartite VAF, $\mathcal{K}_\Phi = \langle \mathcal{Y} \cup \mathcal{Z}, \mathcal{A}, \mathcal{V}, \eta \rangle$ defined from Φ in which

$$
\begin{array}{rcl}
\mathcal{Y} & = & \{\Phi,\ x_1, \ldots, x_n, \neg y_1, \ldots, \neg y_n\} \\
\mathcal{Z} & = & \{C_1, \ldots, C_m,\ \neg x_1, \ldots, \neg x_n, y_1, \ldots, y_n\} \\
\mathcal{A} & = & \{\langle x_i, \neg x_i \rangle, \langle \neg x_i, x_i \rangle, \langle y_i, \neg y_i \rangle, \langle \neg y_i, y_i \rangle \ :\ 1 \leq i \leq n\} \cup \\
 & & \{\langle C_i, \Phi \rangle \ :\ 1 \leq i \leq m\} \cup \\
 & & \{\langle \neg y_i, C_j \rangle \ :\ \neg x_i \in C_j\} \ \cup\ \{\langle x_i, C_j \rangle \ :\ x_i \in C_j\} \\
\mathcal{V} & = & \{con\} \cup \{pos_i,\ neg_i \ :\ 1 \leq i \leq n\} \\
\eta(p) & = & \begin{cases} con & \text{if } p \in \{\Phi, C_1, \ldots, C_m\} \\ pos_i & \text{if } p \in \{x_i, y_i\} \\ neg_i & \text{if } p \in \{\neg x_i, \neg y_i\} \end{cases}
\end{array}
$$

[2] This is a very minor modification of the construction in [4, Thm. 5.1, p. 227].

[3] The Π_2^p–hardness proof from [5] uses a reduction from the problem QBF_2 whose instances are *arbitrary* propositional formula $\Phi(X, Y)$ accepted if for every instantiation α of X there is some instantiation β of Y for which $\Phi(\alpha, \beta)$ holds. This problem remains Π_2^p–complete when Φ is restricted to CNF formulae, the resulting argument system for these cases in the reduction from [5] being 4-partite.

With this construction we can construct a specific audience α, such that $\Phi \in P(\mathcal{K}_\Phi, \alpha)$ if and only if Φ is a satisfiable instance of MCS. For (b), we use a similar construction, adding an argument Φ' to $\mathcal{Z}$ with $\eta(\Phi') = con$ and whose sole attacker is Φ: Φ' is objectively accepted if and only if the instance Φ of MCS is unsatisfiable. The full details are omitted for space reasons. □

4. Restricting numbers of attacks – (p, q)–bounded degree systems

In contrast to many of the results of Section 3, the restriction considered in this Section, perhaps, suprisingly does not lead to improved algorithmic methods. Our principal interest is in introducing the concept of a given class of argument systems being capable of "representing" another class. This is of interest for the following reason. Suppose that Φ and Ψ are properties of argument systems (where the formal definition of "property" will be clarified subsequently). Furthermore, suppose that any system with property Φ can be "represented" (in a sense to be made precise) by another system with property Ψ. Assuming such a representation can be constructed efficiently, we would be able to exploit algorithmic methods tailored to systems with property Ψ also to operate on systems with property Φ: given $\mathcal{H}$ (satisfying Φ), form $\mathcal{G}_\mathcal{H}$ (with property Ψ) and use an algorithm operating on this to decide the question posed of $\mathcal{H}$. In a more precise sense, we have the formalism presented below.

Definition 5 *A* property, Π *of finite argument systems is a (typically, infinite) subset of all possible finite argument systems. We say* $\mathcal{H}$ has property Π *if* $\mathcal{H}$ *is a member of* Π. *The argument system* $\mathcal{H}(\mathcal{X}, \mathcal{A})$ *is* simulated by *the argument system* $\mathcal{G}(\mathcal{X} \cup \mathcal{Y}, \mathcal{B})$ *if all of the following hold for all* $T \subseteq \mathcal{X} \cup \mathcal{Y}$ *and all* $S \subseteq \mathcal{X}$

a. $\text{CA}_{\{\}}(\mathcal{G}, T) \Rightarrow \text{CA}_{\{\}}(\mathcal{H}, T \setminus \mathcal{Y})$; $\text{CA}_{\{\}}(\mathcal{H}, S) \Rightarrow \exists\, T \subseteq \mathcal{Y}\ \text{CA}_{\{\}}(\mathcal{G}, S \cup T)$.
b. $\text{SA}_{\{\}}(\mathcal{G}, T) \Rightarrow \text{SA}_{\{\}}(\mathcal{H}, T \setminus \mathcal{Y})$; $\text{SA}_{\{\}}(\mathcal{H}, S) \Rightarrow \exists\, T \subseteq \mathcal{Y}\ \text{SA}_{\{\}}(\mathcal{G}, S \cup T)$.
c. $\text{COHERENT}(\mathcal{G}) \Leftrightarrow \text{COHERENT}(\mathcal{H})$.

A property, Δ represents *a property* Γ *if for every* $\mathcal{H}(\mathcal{X}, \mathcal{A}) \in \Gamma$ *there is some* $\mathcal{G}(\mathcal{X} \cup \mathcal{Y}, \mathcal{B}) \in \Delta$ *such that* $\mathcal{H}$ *is simulated by* $\mathcal{G}$. *We say that* Δ polynomially represents Γ *if there is some constant* k *such that, for every* $\mathcal{H}(\mathcal{X}, \mathcal{A}) \in \Gamma$ *there is some* $\mathcal{G}(\mathcal{X} \cup \mathcal{Y}, \mathcal{B}) \in \Delta$ *such that* $|\mathcal{X} \cup \mathcal{Y}| \leq |\mathcal{X}|^k$ *and* $\mathcal{H}$ *is simulated by* $\mathcal{G}$. *Finally we say that a property is (*polynomially*)* universal *if it (polynomially) represents all argument systems.*

It will be useful also to view as "polynomially universal" those properties that represent all but finitely many argument systems.[4]

The class of argument systems considered in this section are those defined by the property, $\Delta^{(p,q)}$ introduced in Defn 4(b). Our main result in this section, whose proof is given in outline only, is

Theorem 5 $\Delta^{(2,2)}$ *is polynomially universal.*

[4]The motivation is to allow a rather cleaner statement of results such as Corollary 1: the result claimed in this particular case fails to be true of exactly one graph with the property considered.

Proof: (Outline)[5] Let $\mathcal{H}(\mathcal{X}, \mathcal{A})$ be any finite argument system. Suppose $\mathcal{H} \notin \Delta^{(2,2)}$. Consider any $x \in \mathcal{X}$ for which $\{y \ : \ \langle y, x\rangle \in \mathcal{A}\} = \{y_1, y_2, \ldots, y_k\}$ and $k \geq 3$. Introduce $2k-2$ "new" arguments – $\mathcal{Z}_x^{\text{in}} = \{z_1, z_2, \ldots, z_{2k-2}\}$ – and replace the set of attacks $\{\langle y_i, x\rangle \ : \ i \geq 2\}$ by the system of attacks $\mathcal{B}_x^{\text{in}}$ defined through

$$\{\langle z_1, x\rangle,\ \langle y_{k-1}, z_{2k-2}\rangle, \langle y_k, z_{2k-2}\rangle\} \ \cup \ \bigcup_{i=1}^{k-1} \{\langle z_{2i}, z_{2i-1}\rangle\} \ \cup \ \bigcup_{i=2}^{k-2} \{\langle y_i, z_{2i-2}\rangle\}$$

Letting $\mathcal{G}_x^{\text{in}}$ be the argument system formed by $\langle \mathcal{X} \cup \mathcal{Z}_x^{\text{in}}, \mathcal{A} \setminus \{\langle y_i, x\rangle \ : \ i \geq 2\} \cup \mathcal{B}_x^{\text{in}}\rangle$. It can be shown that S is a preferred extension with $x \in S$ for $\mathcal{H}$ if and only if $S \cup \{z_{2j} \ : \ 1 \leq j \leq k-1\}$ is a preferred extension in $\mathcal{G}_x^{\text{in}}$. Noting that the construction does not change the number of attacks on arguments other than x, a similar procedure can be applied to any remaining argument attacked by at least three arguments.

A near identical construction serves when dealing with those arguments that attack more than two others. For each $x \in \mathcal{X}$ such that, $\{y \ : \ \langle x, y\rangle \in \mathcal{A}\} = \{y_1, y_2, \ldots, y_k\}$ and $k \geq 3$ we introduce $2k-2$ new arguments, $\mathcal{Z}_x^{\text{out}} = \{u_1, u_2, \ldots, u_{2k-2}\}$, replacing the set of attacks $\{\ \langle x, y_i\rangle \ : \ i \geq 2\}$ with the system of attacks $\mathcal{B}_x^{\text{out}}$ given by

$$\{\langle x, u_1\rangle,\ \langle u_{2k-2}, y_{k-1}\rangle, \langle u_{2k-2}, y_k\rangle\} \ \cup \ \bigcup_{i=1}^{k-1} \{\langle u_{2i-1}, u_{2i}\rangle\} \ \cup \ \bigcup_{i=2}^{k-2} \{\langle u_{2i-2}, y_i\rangle\}$$

Letting $\mathcal{G}_x^{\text{out}}$ be the system with arguments $\mathcal{X} \cup \mathcal{Z}_x^{\text{out}}$ and attack relation $\mathcal{A} \setminus \{\langle x, y_i\rangle \ : \ i \geq 2\} \cup \mathcal{B}_x^{\text{out}}$ it may be shown that S is a preferred extension of $\mathcal{H}$ with $x \in S$ if and only if $S \cup \{u_{2j} \ : \ 1 \leq j \leq k-1\}$ is a preferred extension of $\mathcal{G}_x^{\text{out}}$. ▫

Now, recalling that $\Gamma^{(k)}$ is the set of all k-partite argument systems we obtain

Corollary 1 *The property $\Gamma^{(4)} \cap \Delta^{(2,2)}$ is polynomially universal.*

Proof: As undirected graphs, via Brooks' Theorem ([16, Thm 6, Ch. 15, p. 337]), with one exception, every argument system in $\Delta^{(2,2)}$ is 4-colourable and, thus, 4-partite. ▫

5. Conclusions and Development

In this paper we have considered how the complexity of a number of important decision questions in both standard and value-based argument systems is affected under various graph-theoretic restrictions: the system being k-partite; each argument being attacked by and attacking some maximum number of arguments. In the first of these we obtain improved methods for both credulous and sceptical acceptance with bipartite graphs, however, no such reduction in complexity results for the related questions within VAFs. Similarly, we have outlined a construction whereby systems in which no argument attacks and is attacked by at most two arguments are sufficiently general to model the behaviour of any finite argument system, thereby suggesting that this restriction is unlikely

[5] For readers familiar with the standard translation from k-CNF to 3-CNF ($k \geq 3$) or the Chomsky Normal Form representation of context-free grammars, we note that a similar approach is used in these constructions.

to yield better algorithmic methods. The concept of "polynomial universality" introduced in Defn 5 with a concrete example of such being given in Corollary 1, may be of some interest regarding generic "normal form" representations of argument systems. One particular aspect of some interest with respect to Corollary 1 concerns multiagent implementation of argument processes: under the (reasonable) assumption that the arguments endorsed by individual agents are "internally consistent", i.e. conflict–free, one can envisage potential applications as providing a mechanism for distributing the components of a global system over (at most) four agents so that interaction regarding the status of single arguments, in the first instance, need only consider the (at most) two agents from which its attackers originate.

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*, 77:321–357, 1995.

[2] 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.

[3] Y. Dimopoulos, B. Nebel, and F. Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141:55–78, 2002.

[4] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science*, 170:209–244, 1996.

[5] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141:187–203, 2002.

[6] S. Coste-Marquis, C. Devred, and P. Marquis. Symmetric argumentation frameworks. In L. Godo, editor, *Proc. 8^{th} European Conf. on Symbolic and Quantitative Approaches to Reasoning With Uncertainty (ECSQARU)*, volume 3571 of *LNAI*, pages 317–328. Springer-Verlag, 2005.

[7] P. Baroni and M. Giacomin. Solving semantic problems with odd–length cycles in argumentation. In *Proc. 7^{th} European Conf. on Symbolic and Quantitative Approaches to Reasoning With Uncertainty (ECSQARU)*, volume 2711 of *LNAI*, pages 440–451. Springer-Verlag, 2003.

[8] P. Baroni, M. Giacomin, and G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence*, 168(1–2):162–210, 2005.

[9] L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Math. and Artificial Intelligence*, 34:197–215, 2002.

[10] T. J. M. Bench-Capon. Persuasion in Practical Argument Using Value-based Argumentation Frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[11] 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.

[12] P. E. Dunne and T. J. M. Bench-Capon. Complexity in value-based argument systems. In *Proc. 9th JELIA*, volume 3229 of *LNAI*, pages 360–371. Springer-Verlag, 2004.

[13] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of* NP*-Completeness*. W. H. Freeman: New York, 1979.

[14] P. E. Dunne and T. J. M. Bench-Capon. Two party immediate response disputes: properties and efficiency. *Artificial Intelligence*, 149:221–250, 2003.

[15] T. J. M. Bench-Capon, S. Doutre, and P. E. Dunne. Audiences in argumentation frameworks. Technical Report ULCS-05-012, Department of Computer Science, Univ. of Liverpool, December 2005.

[16] C. Berge. *Graphs and Hypergraphs*. North-Holland, Amsterdam, 1973.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Computing Preferred Extensions for Argumentation Systems with Sets of Attacking Arguments

Søren Holbech Nielsen [a], Simon Parsons [b]
[a] *Department of Computer Science*
Aalborg University, Aalborg
Denmark
[b] *Department of Computer and Information Science*
Brooklyn College, City University of New York
Brooklyn, 11210 NY, USA

Abstract. The hitherto most abstract, and hence general, argumentation system, is the one described by Dung in a paper from 1995. This framework does not allow for joint attacks on arguments, but in a recent paper we adapted it to support such attacks, and proved that this adapted framework enjoyed the same formal properties as that of Dung. One problem posed by Dung's original framework, which was neglected for some time, is how to compute preferred extensions of the argumentation systems. However, in 2001, in a paper by Doutre and Mengin, a procedure was given for enumerating preferred extensions for these systems. In this paper we propose a method for enumerating preferred extensions of the potentially more complex systems, where joint attacks are allowed. The method is inspired by the one given by Doutre and Mengin.

Keywords. Argumentation with sets, Preferred Extensions

1. Introduction

In the last fifteen years or so, there has been much interest in argumentation systems within the artificial intelligence community. This interest spreads across many different sub-areas of artificial intelligence. One of these is non-monotonic reasoning [1,2], which exploits the fact that argumentation systems can handle, and resolve, inconsistencies [3, 4] and uses it to develop general descriptions of non-monotonic reasoning [5,6]. This line of work is summarised in [7]. Another area that makes use of argumentation is reasoning and decision making under uncertainty [8,9,10], which exploits the dependency structure one can infer from arguments in order to correctly combine evidence. Much of this work is covered in [11]. More recently [12,13], the multi-agent systems community has begun to make use of argumentation, using it to develop a notion of rational interaction [14,15].

One very influential and very abstract system of argumentation was that introduced by Dung [16]. This was, for instance, the basis for the work in [5], was the system extended by Amgoud in [17,18], and subsequently as the basis for the dialogue systems

in [19,20]. The importance of Dung's results is mainly due to the fact that his framework abstracts away from details of language and argumentation rules, that the presented semantics therefore are clear and intuitive, and that relationships among arguments can be analysed in isolation from other (e.g. implicational) relationships. Furthermore, the results can easily be transferred to any other argumentation framework, by identifying that framework's equivalent of an attack. It is this generality, we believe, that has contributed to the popularity of the work, and we see it as a prime contender for becoming an established standard for further investigations into the nature of arguments and their interaction.

However, even if Dung was trying to abstract away from the underlying language and structure of arguments, his framework implicitly assumes a logical "and" connective in the underlying language, to be able to model all kinds of attacks. This hidden assumption is caused by Dung's attack relation being a simple binary relation from one argument to another, rather than a relation mapping sets of arguments to other sets of arguments. In a recent paper [21] we presented a generalisation of Dung's framework, which allows sets of arguments to attack single arguments, and thus frees the underlying language from being closed under some logical "and" connective[1]. The main motivation for that work was that sometimes it seems reasonable for a number of arguments to interact and constitute an attack on some other argument, even though the arguments of the attack does not individually attack that argument. The approach, where such joint attacks are modelled by adding to the argumentation system a new argument that represents the set of attacking arguments, and then employing traditional argumentation analysis to this system, is not satisfactory: The encoding is artificial, adding distance between the formalism and the modelled argumentation situation, and to ensure that nonsense conclusions do not arise, the relation of attack among arguments need to be restricted or an extra layer of logical relationships among arguments need to be specified. The former muddles the clear distinction between arguments and attacks, which was the very appeal of Dung's framework, and the latter makes it hard to survey the effects of one set of argument on others and calls for more specialized formalisms for analysis than Dung's. For further elaborations on this see [21].

In this paper, we build on the work in [21] and propose a method for enumerating preferred extensions of the argumentation systems defined there. In general it is hard to compute a preferred extension [23], but [24] presents a method that enumerates preferred extensions for an abstract argumentation system as presented in [16]. Moreover, [25] and [26] present methods for answering whether a specific argument is in at least one preferred extension, or if it is in all preferred extensions. Here we adapt the basic

[1] Subsequently, we have been directed to [22], which describes an argumentation framework that is a generalization of that in [16] too. The main differences between [22] and [21] are due to difference in perspectives: Bochman is motivated by the task of establishing a correspondence between disjunctive logic programming and abstract argumentation, and ends up with a framework that allows any finite set of arguments (including the empty set) to attack and be attacked by any other finite set, whereas we have tried to expand the dialogical and dialectical boundaries of abstract argumentation by allowing for arbitrary sets of attacking arguments (but the empty set), and claim that further flexibility is not needed for argumentative reasoning. (Indeed, the main example motivating attacks on entire sets of arguments turns out to be sensibly represented in our framework.) Due to his aims, Bochman construct new semantics for his framework and identifies new families of argumentation systems with nice properties (none of them coinciding with our formalism). We, on the other hand, stick as close as possible to the semantics provided by Dung, and instead show that the all of Dung's results are valid for systems with sets of attacking arguments.

technique of [24] to the more complex case of argumentation systems with joint attacks. The main problem for this adaptation, is that the argumentation systems of [16] can be viewed as directed graphs, and that this fact is exploited in the pruning rules of [24]. For the argumentation systems of [21], however, no similar graph structure exists, and new pruning rules thus have to be constructed. In particular, we lack a context independent notion of "reflective" arguments, and a context independent notion of a single argument being detrimental to a specific set of arguments.

2. Argumentation With Attacking Sets of Arguments

In this section we present our generalisation of the framework of [16], as introduced in [21].

Definition 1 (Argumentation Systems). *An* argumentation system *is a pair* $(\boldsymbol{A}, \triangleright)$*, where* $\boldsymbol{A}$ *is a set of arguments, and* $\triangleright \subseteq (\mathcal{P}(\boldsymbol{A}) \setminus \{\varnothing\}) \times \boldsymbol{A}$ *is an* attack relation.

Throughout the paper we assume an argumentation system $\mathcal{A} = (\boldsymbol{A}, \triangleright)$, and take it to be implicit.

We say that a set of arguments $\boldsymbol{S}$ *attacks* an argument A, if there is $\boldsymbol{S}' \subseteq \boldsymbol{S}$ such that $\boldsymbol{S}' \triangleright A$. In that case we also say that A *is attacked by* $\boldsymbol{S}$. If there is no set $\boldsymbol{S}'' \subsetneq \boldsymbol{S}'$ such that $\boldsymbol{S}''$ attacks A, then we say that $\boldsymbol{S}'$ is a *minimal* attack on A. Obviously, if there is a set that attacks an argument A, then there must also exist a minimal attack on A. Moreover, if $\boldsymbol{S}$ is a minimal attack on A, then it must be the case that $\boldsymbol{S} \triangleright A$. If for two sets of arguments $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$, there is an argument $A \in \boldsymbol{S}_2$, which is attacked by $\boldsymbol{S}_1$, then we say that $\boldsymbol{S}_1$ attacks $\boldsymbol{S}_2$, and that $\boldsymbol{S}_2$ is attacked by $\boldsymbol{S}_1$. If a set $\boldsymbol{S}_1$ attacks some argument in $\boldsymbol{S}_2$, and this is true of no subsets of $\boldsymbol{S}_1$, then we say that $\boldsymbol{S}_1$ is a minimal attack on $\boldsymbol{S}_2$, and relaxing notation a bit, write $\boldsymbol{S}_1 \triangleright \boldsymbol{S}_2$. If a set of arguments $\boldsymbol{S}$ does not attack itself, then we say that $\boldsymbol{S}$ is *conflict-free*.

Let $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ be sets of arguments. If $\boldsymbol{S}_2$ attacks some argument A, and $\boldsymbol{S}_1$ attacks $\boldsymbol{S}_2$, then we say that $\boldsymbol{S}_1$ is a *defense of* A *from* $\boldsymbol{S}_2$, and that $\boldsymbol{S}_1$ *defends* A *from* $\boldsymbol{S}_2$. Obviously, if $\boldsymbol{S}_3$ is a superset of $\boldsymbol{S}_1$, $\boldsymbol{S}_3$ is also a defense of A from $\boldsymbol{S}_2$. An argument A is said to be *acceptable with respect to a set of arguments* $\boldsymbol{S}$, if $\boldsymbol{S}$ defends A from all sets of attacking arguments $\boldsymbol{S}' \subseteq \boldsymbol{A}$. A conflict-free set of arguments $\boldsymbol{S}$ is said to be *admissible* if each argument in $\boldsymbol{S}$ is acceptable with respect to $\boldsymbol{S}$. This leads us to the credulous semantics we treat in this paper:

Definition 2 (Preferred Extensions). *An admissible set* $\boldsymbol{S}^*$ *is called a* preferred extension*, if there is no admissible set* $\boldsymbol{S}' \subseteq \boldsymbol{A}$*, such that* $\boldsymbol{S}^* \subsetneq \boldsymbol{S}'$.

From [16] and [21], we have that for each admissible set $\boldsymbol{S}$, there exists a preferred extension $\boldsymbol{S}^*$, such that $\boldsymbol{S} \subseteq \boldsymbol{S}^*$. Moreover, as the empty set is an admissible set, we have that every argumentation system has at least one preferred extension.

A very skeptical semantics, is the *grounded extension*, which is defined as the least fix point of the function $F : \mathcal{P}(\boldsymbol{A}) \to \mathcal{P}(\boldsymbol{A})$, defined as

$$F(\boldsymbol{S}) = \{A \ : \ A \text{ is acceptable wrt. } \boldsymbol{S}\}.$$

Example 1 (An Introductory Example): *Consider an argumentation system* $\mathcal{A}_e = (\boldsymbol{A}_e, \triangleright_e)$*, where* $\boldsymbol{A}_e = \{A, B, C, D, E, F\}$ *and* $\triangleright_e$ *is defined as:*

$$\{A, C, D\} \triangleright_e B, \quad \{A, B\} \triangleright_e C, \quad \{B\} \triangleright_e D, \quad \{C, E\} \triangleright_e D,$$
$$\{D\} \triangleright_e E, \quad \{B, F\} \triangleright_e E, \quad \{A\} \triangleright_e F, \text{ and } \{D\} \triangleright_e F.$$

It can easily be verified that the grounded extension of $\mathcal{A}_e$ *is* $\{A\}$*. The preferred extensions are* $\{A, B, E\}$ *and* $\{A, C, D\}$*, which we shall prove later in the paper.*

3. Computing Preferred Extensions

We now present a method for computing the preferred extensions for an argumentation system with sets of attacking arguments as defined in Definition 1. The method is inspired by a similar method, for computing preferred extensions for Dung's original argumentation systems, presented in [24]. The basic strategy is to enumerate all possible divisions of $\boldsymbol{A}$ into two sets, $\boldsymbol{I}$ and $\boldsymbol{O}$, where $\boldsymbol{I}$ are the arguments that are *i*n a preferred extension, and $\boldsymbol{O}$ are those that are *o*ut, and then check for each division if $\boldsymbol{I}$ is a preferred extension. Now, of course the number of divisions can be drastically reduced, by noting requirements on $\boldsymbol{I}$ imposed by Definition 2, so a full enumeration can often be avoided.

The enumeration of divisions is constructed as a tree, where each node is a partition of $\boldsymbol{A}$ into three sets $(\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$, where $\boldsymbol{U}$ is the arguments still not assigned to one of the two divisions $\boldsymbol{I}$ and $\boldsymbol{O}$. The root of the tree is a node where both $\boldsymbol{I}$ and $\boldsymbol{O}$ are empty and all arguments are assigned to the undecided partition. Each child $(\boldsymbol{I}', \boldsymbol{O}', \boldsymbol{U}')$ of a node $(\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$ is then a refinement of the division represented by the previous node, i.e. $\boldsymbol{I} \subseteq \boldsymbol{I}'$ and $\boldsymbol{O} \subseteq \boldsymbol{O}'$. The size of such a tree is exponential in the number of arguments, but fortunately we often do not have to construct the entire tree, and if only more specific queries are sought answered (such as "Is argument A included in some preferred extension?") we can sometimes get away with only inspecting parts of a few branches of the tree.

First we define the nodes we work with. These are called $\mathcal{A}$*-candidates*, or as we take $\mathcal{A}$ to be implicit, just *candidates*. For a given set $\boldsymbol{S} \subseteq \boldsymbol{A}$, define

$$\boldsymbol{S}^{\rightarrow} = \{A \in \boldsymbol{A} \,:\, \exists \boldsymbol{T} \subseteq \boldsymbol{S} \text{ s.t. } \boldsymbol{T} \triangleright A\}$$

and

$$\boldsymbol{S}^{\leftarrow} = \{A \in \boldsymbol{A} \,:\, \exists \boldsymbol{T} \subseteq \boldsymbol{S}, B \in \boldsymbol{S} \text{ s.t. } \boldsymbol{T} \cup \{A\} \triangleright B\}.$$

$\boldsymbol{S}^{\rightarrow}$ is thus the set of arguments attacked by $\boldsymbol{S}$, and $\boldsymbol{S}^{\leftarrow}$ is the set of arguments, which if added to $\boldsymbol{S}$, would make $\boldsymbol{S}$ attack itself. A candidate is then a triple $(\boldsymbol{I} \subseteq \boldsymbol{A}, \boldsymbol{O} \subseteq \boldsymbol{A}, \boldsymbol{U} = \boldsymbol{A} \setminus (\boldsymbol{I} \cup \boldsymbol{O}))$ satisfying the following properties:

$$\boldsymbol{I}^{\rightarrow} \subseteq \boldsymbol{O}, \tag{1}$$

$$\boldsymbol{I}^{\leftarrow} \subseteq \boldsymbol{O}, \text{ and} \tag{2}$$

$$\boldsymbol{I} \cap \boldsymbol{O} = \varnothing. \tag{3}$$

(If $\mathcal{C} = (\boldsymbol{I}, \boldsymbol{O}, \boldsymbol{U})$ is a triple, we will use subscripts to refer to the sets in the partition, e.g. $\boldsymbol{I}_\mathcal{C}$ denotes the set $\boldsymbol{I}$ in $\mathcal{C}$.)

Example 2 (Candidates): *We consider again the argumentation system $\mathcal{A}_e = (\boldsymbol{A}_e, \triangleright_e)$ from Example 1. A few examples of candidates are $(\{B\}, \{D\}, \{A, C, E, F\})$, $(\varnothing, \boldsymbol{A}_e, \varnothing)$, and $(\{C, D\}, \{A, B, E, F\}, \varnothing)$. Some examples of non-candidates are $(\{A\}, \{B, D\}, \{C, E, F\})$, $(\{E, F\}, \{A, D\}, \{B, C\})$, and $(\{B\}, \{B, D\}, \{A, C, E, F\})$.*

Focusing only on candidates, rather than arbitrary divisions of $\boldsymbol{A}$, is thus a restriction on the number of divisions to consider. We argue that it is sufficient below.

It follows from (1) and (3), that for any candidate $\mathcal{C}$, $\boldsymbol{I}_\mathcal{C}$ is conflict-free. For any triple $\mathcal{C}$, we denote by pref($\mathcal{C}$) the set of all preferred extensions $\boldsymbol{S}^*$, where $\boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}^* \subseteq \boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C}$. It follows, that if $\boldsymbol{U}_\mathcal{C} = \varnothing$, then pref($\mathcal{C}$) is $\{\boldsymbol{I}_\mathcal{C}\}$ if $\boldsymbol{I}_\mathcal{C}$ is a preferred extension and $\varnothing$ otherwise.

Given a triple $\mathcal{C}$ and an argument $A \in \boldsymbol{U}_\mathcal{C}$, define the triples

$$\mathcal{C} - A = (\boldsymbol{I}_\mathcal{C}, \quad \boldsymbol{O}_\mathcal{C} \cup \{A\}, \quad \boldsymbol{U}_\mathcal{C} \setminus \{A\}), \tag{4}$$

and

$$\mathcal{C} + A = (\boldsymbol{I}_\mathcal{C} \cup \{A\}, \quad \boldsymbol{O}_\mathcal{C} \cup \boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}, \quad \boldsymbol{U}_\mathcal{C} \setminus (\{A\} \cup \boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A})) \tag{5}$$

where

$$\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} = \{B \in \boldsymbol{U}_\mathcal{C} \ : \ \exists \boldsymbol{S} \subseteq \boldsymbol{I}_\mathcal{C} \text{ s.t. } \boldsymbol{S} \cup \{A\} \triangleright B\} \tag{6}$$

and

$$\begin{aligned} \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A} = \{B \in \boldsymbol{U}_\mathcal{C} \ : \ \exists \boldsymbol{S} \subseteq \boldsymbol{I}_\mathcal{C}, C \in \boldsymbol{I}_\mathcal{C} \text{ s.t. } & \boldsymbol{S} \cup \{B\} \triangleright A \\ \vee \ \boldsymbol{S} \cup \{A, B\} \triangleright C \ \vee \ & \boldsymbol{S} \cup \{A, B\} \triangleright A\}. \end{aligned} \tag{7}$$

Example 3 (Adding Arguments to Triples): *Building on Example 2, we add E to the candidate $\mathcal{C}_1 = (\{B\}, \{D\}, \{A, C, E, F\})$ and the non-candidate $\mathcal{C}_2 = (\{A\}, \{B, D\}, \{C, E, F\})$: In the first case, $\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}_1+E} = \varnothing$ and $\boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}_1+E} = \{F\}$, and in the second $\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}_2+E} = \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}_2+E} = \varnothing$. Therefore, $\mathcal{C}_1 + E = (\{B, E\}, \{D, F\}, \{A, C\})$ and $\mathcal{C}_2 + E = (\{A, E\}, \{B, D\}, \{C, F\})$.*

It is easy to verify that, given a candidate $\mathcal{C}$ and an argument $A \in \boldsymbol{U}_\mathcal{C}$, we have that

$$\boldsymbol{I}^{\rightarrow}_{\mathcal{C}+A} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}} = \boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A}, \tag{8}$$

and

$$\boldsymbol{I}^{\leftarrow}_{\mathcal{C}+A} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}} = \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}. \tag{9}$$

Given the partial division represented by a candidate, some arguments might be impossible to add to the set $\boldsymbol{I}$ without ending up with a contradiction. We therefore define the set of *reflexive arguments* with respect to a candidate $\mathcal{C}$ as follows:

$$\text{refl}(\mathcal{C}) = \{A \in \boldsymbol{U}_{\mathcal{C}} \ : \ \exists \boldsymbol{S} \subseteq \boldsymbol{I}_{\mathcal{C}}, \text{ s.t. } \boldsymbol{S} \cup \{A\} \triangleright A\}. \tag{10}$$

From the definitions, it immediately follows that if $A \in \text{refl}(\mathcal{C})$ then $\text{pref}(\mathcal{C} + A) = \varnothing$. Furthermore, we can state an important theorem, which implies that given a candidate $\mathcal{C}$, we can use the definitions of $\mathcal{C} + A$ and $\mathcal{C} - A$ to construct a tree of candidates having $\mathcal{C}$ as root:

Theorem 1. *Let $\mathcal{C}$ be a candidate, and $A \in \boldsymbol{U}_{\mathcal{C}}$. If $A \notin \text{refl}(\mathcal{C})$ then both $\mathcal{C} + A$ and $\mathcal{C} - A$ are candidates as well. Otherwise only $\mathcal{C} - A$ is a candidate.*

Proof. It is obvious that $\mathcal{C} - A$ is a candidate no matter whether A is in $\text{refl}(\mathcal{C})$ or not. We therefore only show that $\mathcal{C} + A$ is a candidate iff A is not in $\text{refl}(\mathcal{C})$.

First, assume that A is in $\text{refl}(\mathcal{C})$. This means that there is some set $\boldsymbol{S} \subseteq \boldsymbol{I}_{\mathcal{C}}$, such that $\boldsymbol{S} \cup \{A\} \triangleright A$. Consequently, $\boldsymbol{I}_{\mathcal{C}+A} = \boldsymbol{I}_{\mathcal{C}} \cup \{A\}$ contains a subset $\boldsymbol{T} = \boldsymbol{S} \cup \{A\}$, such that $\boldsymbol{T} \triangleright A$. If $\mathcal{C} + A$ was to be a candidate, (1) would therefore require that A is in $\boldsymbol{O}_{\mathcal{C}+A}$. It follows that A is in $\boldsymbol{I}_{\mathcal{C}+A} \cap \boldsymbol{O}_{\mathcal{C}+A}$, which is thus not empty. That contradicts (3), and $\mathcal{C} + A$ can thus not be a candidate.

Conversely, assume that A is not in $\text{refl}(\mathcal{C})$, and we show that $\mathcal{C} + A$ is a candidate by means of contradiction. That is, assume that $\mathcal{C} + A$ is not a candidate, which means that one of the following must be true:

(i): $\exists B \in \boldsymbol{I}^{\rightarrow}_{\mathcal{C}+A}$ s.t. $B \notin \boldsymbol{O}_{\mathcal{C}+A}$,
(ii): $\exists B \in \boldsymbol{I}^{\leftarrow}_{\mathcal{C}+A}$ s.t. $B \notin \boldsymbol{O}_{\mathcal{C}+A}$, or
(iii): $\exists B \in \boldsymbol{I}_{\mathcal{C}+A} \cap \boldsymbol{O}_{\mathcal{C}+A}$.

We show that each case is impossible. First, assume that (i) is the case. Since $\mathcal{C}$ is a candidate, we necessarily have that $\boldsymbol{I}^{\rightarrow}_{\mathcal{C}} \subseteq \boldsymbol{O}_{\mathcal{C}} \subseteq \boldsymbol{O}_{\mathcal{C}+A}$ and it must thus be the case that $B \in \boldsymbol{I}^{\rightarrow}_{\mathcal{C}+A} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}$, which according to (8) is equivalent to having $B \in \boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A}$. But according to (5), $\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A}$ is a subset of $\boldsymbol{O}_{\mathcal{C}+A}$, so $B \in \boldsymbol{O}_{\mathcal{C}+A}$ after all, which is a contradiction. Case (ii) is proved to be impossible with a similar argument.

Assume that (iii) is the case. Since $\mathcal{C}$ is a candidate, we know from (3) that $\boldsymbol{O}_{\mathcal{C}} \cap \boldsymbol{I}_{\mathcal{C}} = \varnothing$, and, since A is in $\boldsymbol{U}_{\mathcal{C}}$, which is disjoint from $\boldsymbol{O}_{\mathcal{C}}$, also that $\boldsymbol{O}_{\mathcal{C}} \cap (\boldsymbol{I}_{\mathcal{C}} \cup \{A\}) = \boldsymbol{O}_{\mathcal{C}} \cap \boldsymbol{I}_{\mathcal{C}+A} = \varnothing$. Therefore, B must be a member of $\boldsymbol{O}_{\mathcal{C}+A} \setminus \boldsymbol{O}_{\mathcal{C}} = (\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}) \subseteq \boldsymbol{U}_{\mathcal{C}}$. Furthermore, as $\boldsymbol{U}_{\mathcal{C}} \cap \boldsymbol{I}_{\mathcal{C}} = \varnothing$ it follows that B must be in $\boldsymbol{I}_{\mathcal{C}+A} \setminus \boldsymbol{I}_{\mathcal{C}} = \{A\}$. Thus, A must be in either $\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A}$ or $\boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}$. The first possibility is ruled out, since A by assumption is not a member of $\text{refl}(\mathcal{C})$. So A must be in $\boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}$.

According to the definition of $\boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}$, there must be a $C \in \boldsymbol{I}_{\mathcal{C}}$ and a set $\boldsymbol{S} \subseteq \boldsymbol{I}_{\mathcal{C}}$, so either $\boldsymbol{S} \cup \{A\} \triangleright A$ or $\boldsymbol{S} \cup \{A\} \triangleright C$. Again the first possibility is precluded by the assumption that A is not in $\text{refl}(\mathcal{C})$, so it must be the case that $\boldsymbol{S} \cup \{A\} \triangleright C$. But then A is in $\boldsymbol{I}^{\leftarrow}_{\mathcal{C}}$ and as $\mathcal{C}$ is a candidate also in $\boldsymbol{O}_{\mathcal{C}}$. That contradicts the assumption that A is in $\boldsymbol{U}_{\mathcal{C}}$, and the theorem follows. □

The theorem thus establishes that iterated use of the $\mathcal{C} + A$ and $\mathcal{C} - A$-definitions makes sense. Moreover, we have the following result on that activity:

Theorem 2. *Let $\mathcal{C}$ be a candidate and A and B be distinct arguments in $\boldsymbol{U} \setminus \text{refl}(\mathcal{C})$, such that both $(\mathcal{C} + A) + B$ and $(\mathcal{C} + B) + A$ are candidates. Then*

$$(\mathcal{C} + A) + B = (\mathcal{C} + B) + A, \tag{11}$$

$$(\mathcal{C} - A) - B = (\mathcal{C} - B) - A, \text{ and} \tag{12}$$

$$(\mathcal{C} + A) - B = (\mathcal{C} - B) + A. \tag{13}$$

Proof. We only show (11), since the others follow from similar, albeit slightly simpler arguments. It is obvious that $\boldsymbol{I}_{(\mathcal{C}+A)+B} = \boldsymbol{I}_{(\mathcal{C}+B)+A}$ and, given that $\boldsymbol{O}_{(\mathcal{C}+A)+B} = \boldsymbol{O}_{(\mathcal{C}+B)+A}$, also that $\boldsymbol{U}_{(\mathcal{C}+A)+B} = \boldsymbol{U}_{(\mathcal{C}+B)+A}$. We therefore just need to show that $\boldsymbol{O}_{(\mathcal{C}+A)+B} = \boldsymbol{O}_{(\mathcal{C}+B)+A}$:

$$\begin{aligned}
\boldsymbol{O}_{(\mathcal{C}+A)+B} =& \boldsymbol{O}_{\mathcal{C}} \cup \boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\rightarrow}_{(\mathcal{C}+A)+B} \cup \boldsymbol{\Delta}^{\leftarrow}_{(\mathcal{C}+A)+B} \\
=& \boldsymbol{O}_{\mathcal{C}} \cup (\boldsymbol{I}^{\rightarrow}_{\mathcal{C}+A} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}) \cup (\boldsymbol{I}^{\leftarrow}_{\mathcal{C}+A} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}}) \\
& \cup (\boldsymbol{I}^{\rightarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}+A}) \cup (\boldsymbol{I}^{\leftarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}+A}) \\
=& \boldsymbol{O}_{\mathcal{C}} \cup (\boldsymbol{I}^{\rightarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}) \cup (\boldsymbol{I}^{\leftarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}}),
\end{aligned}$$

where the last step is warranted by the observation that $\boldsymbol{S}^{\rightarrow} \subseteq \boldsymbol{T}^{\rightarrow}$ and $\boldsymbol{S}^{\leftarrow} \subseteq \boldsymbol{T}^{\leftarrow}$, for any two sets $\boldsymbol{S}$ and $\boldsymbol{T}$, where $\boldsymbol{S} \subseteq \boldsymbol{T}$.

Now, as $\boldsymbol{I}_{(\mathcal{C}+A)+B} = \boldsymbol{I}_{(\mathcal{C}+B)+A}$, we have:

$$\begin{aligned}
& \boldsymbol{O}_{\mathcal{C}} \cup (\boldsymbol{I}^{\rightarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}) \cup (\boldsymbol{I}^{\leftarrow}_{(\mathcal{C}+A)+B} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}}) \\
=& \boldsymbol{O}_{\mathcal{C}} \cup (\boldsymbol{I}^{\rightarrow}_{(\mathcal{C}+B)+A} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}) \cup (\boldsymbol{I}^{\leftarrow}_{(\mathcal{C}+B)+A} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}}) \\
=& \boldsymbol{O}_{\mathcal{C}} \cup (\boldsymbol{I}^{\rightarrow}_{\mathcal{C}+B} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}}) \cup (\boldsymbol{I}^{\leftarrow}_{\mathcal{C}+B} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}}) \\
& \cup (\boldsymbol{I}^{\rightarrow}_{(\mathcal{C}+B)+A} \setminus \boldsymbol{I}^{\rightarrow}_{\mathcal{C}+B}) \cup (\boldsymbol{I}^{\leftarrow}_{(\mathcal{C}+B)+A} \setminus \boldsymbol{I}^{\leftarrow}_{\mathcal{C}+B}) = \boldsymbol{O}_{(\mathcal{C}+B)+A}
\end{aligned}$$

□

Thus, no matter in what order several arguments are moved from $\boldsymbol{U}_{\mathcal{C}}$ to $\boldsymbol{I}_{\mathcal{C}}$ and $\boldsymbol{O}_{\mathcal{C}}$, the resulting candidate is the same.

Now, we wish to use a tree of candidates as enumeration of preferred extensions. Given a candidate $\mathcal{C}$, we define a $\mathcal{C}$-tree inductively as follows:

- If $\boldsymbol{U}_{\mathcal{C}} = \varnothing$ then the tree consisting of the leaf $\mathcal{C}$ is a $\mathcal{C}$-tree.
- If $A \in \boldsymbol{U}_{\mathcal{C}} \setminus \text{refl}(\mathcal{C})$ then a tree with root node $\mathcal{C}$ having the roots of a $\mathcal{C} + A$-tree and a $\mathcal{C} - A$-tree as only children is a $\mathcal{C}$-tree.
- If $A \in \boldsymbol{U}_{\mathcal{C}} \cap \text{refl}(\mathcal{C})$ then a tree with root node $\mathcal{C}$ having the root of a $\mathcal{C} - A$-tree as only child is a $\mathcal{C}$-tree.

Example 4 ($\mathcal{C}$-trees): *We continue expanding on $\mathcal{C}_1$ as in Example 3. Repeated construction of candidates gives the $\mathcal{C}_1$-tree presented in Figure 1. Notice that some branches are shorter than others. This is because some additions to $\boldsymbol{I}$ imply additions to $\boldsymbol{O}$, and hence exhaust $\boldsymbol{U}$ sooner.*

Any tree, for which there is some candidate $\mathcal{C}$ such that the tree is a $\mathcal{C}$-tree, is called a *candidate tree*. The following results guarantee that candidate trees include all divisions that encode preferred extensions.

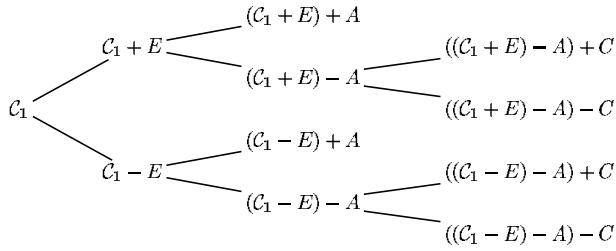


Figure 1. *A $\mathcal{C}_1$-tree*

Lemma 1. *Let $\boldsymbol{S} \subseteq \boldsymbol{A}$ be a conflict-free set, $\mathcal{C}$ a candidate, where $\boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}$ and $\boldsymbol{O}_\mathcal{C} \cap \boldsymbol{S} = \varnothing$, and A a member of $\boldsymbol{S} \setminus \boldsymbol{I}_\mathcal{C}$. Then $\mathcal{C} + A$ is a candidate, and $\boldsymbol{O}_{\mathcal{C}+A} \cap \boldsymbol{S} = \varnothing$.*

Proof. First note that A cannot be in $\text{refl}(\mathcal{C})$, as that would mean that there is a set $\boldsymbol{T} \subseteq \boldsymbol{I}_\mathcal{C}$ such that $\boldsymbol{T} \cup \{A\} \triangleright A$, which again would mean that $\boldsymbol{S}$ is not conflict-free. Therefore, Theorem 1 guarantees that $\mathcal{C} + A$ is a candidate, and we thus only need to show that $\boldsymbol{O}_{\mathcal{C}+A} \cap \boldsymbol{S} = \varnothing$.

As $\boldsymbol{O}_\mathcal{C} \cap \boldsymbol{S} = \varnothing$ it follows that $\boldsymbol{O}_{\mathcal{C}+A} \cap \boldsymbol{S} = (\boldsymbol{\Delta}^{\rightarrow}_{\mathcal{C}+A} \cup \boldsymbol{\Delta}^{\leftarrow}_{\mathcal{C}+A}) \cap \boldsymbol{S}$. If this set is non-empty, then there must be a B in $\boldsymbol{S}$, such that there is a set $\boldsymbol{T} \subseteq \boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}$ and element $C \in \boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}$, where either $\boldsymbol{T} \cup \{A\} \triangleright B$, $\boldsymbol{T} \cup \{B\} \triangleright A$, $\boldsymbol{T} \cup \{A, B\} \triangleright C$, or $\boldsymbol{T} \cup \{A, B\} \triangleright A$. But each of these imply that $\boldsymbol{S}$ is not conflict-free, and hence we conclude that $\boldsymbol{O}_{\mathcal{C}+A} \cap \boldsymbol{S} = \varnothing$. □

Theorem 3. *Let $\mathcal{C}$ be a candidate, and $A \in \boldsymbol{U}_\mathcal{C}$. Then pref$(\mathcal{C}) =$ pref$(\mathcal{C} + A) \cup$ pref$(\mathcal{C} - A)$.*

Proof. It is obvious that $\text{pref}(\mathcal{C} + A) \cup \text{pref}(\mathcal{C} - A) \subseteq \text{pref}(\mathcal{C})$, so we only show that $\text{pref}(\mathcal{C}) \subseteq \text{pref}(\mathcal{C} + A) \cup \text{pref}(\mathcal{C} - A)$.

Let $\boldsymbol{S}^* \in \text{pref}(\mathcal{C})$, i.e. $\boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}^* \subseteq \boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C}$. If A is not in $\boldsymbol{S}^*$, then it follows that $\boldsymbol{S}^* \subseteq \boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \setminus \{A\} = \boldsymbol{I}_{\mathcal{C}-A} \cup \boldsymbol{U}_{\mathcal{C}-A}$, and hence that $\boldsymbol{S}^* \in \text{pref}(\mathcal{C} - A)$. If A is in $\boldsymbol{S}^*$ we similarly get that $\boldsymbol{S}^* \supseteq \boldsymbol{I}_\mathcal{C} \cup \{A\} = \boldsymbol{I}_{\mathcal{C}+A}$ and we only need to show that $\boldsymbol{S}^* \subseteq \boldsymbol{I}_{\mathcal{C}+A} \cup \boldsymbol{U}_{\mathcal{C}+A}$, i.e. that $\boldsymbol{O}_{\mathcal{C}+A} \cap \boldsymbol{S}^* = \varnothing$. But this is guaranteed by Lemma 1, and the result follows. □

From this we immediately get:

Corollary 1. *If $\boldsymbol{S}^*$ is a preferred extension, then there is a leaf $\mathcal{C}$ of any $(\varnothing, \varnothing, \boldsymbol{A})$-tree, such that $\boldsymbol{S}^* \in$ pref$(\mathcal{C})$.*

Thus, when enumerating preferred extensions, it suffices to construct a single candidate tree, viz. a $(\varnothing, \varnothing, \boldsymbol{A})$-tree, even if candidates do not represent all possible divisions of $\boldsymbol{A}$. Furthermore, as the grounded extension of any system is a subset of any preferred extension [16,21], we have the following stronger result:

Corollary 2. *If $\boldsymbol{S}^*$ is a preferred extension, and $\boldsymbol{G}$ is the grounded extension, then there is a leaf $\mathcal{C}$ of any $(\boldsymbol{G}, \boldsymbol{G}^{\rightarrow} \cup \boldsymbol{G}^{\leftarrow}, \boldsymbol{A} \setminus (\boldsymbol{G} \cup \boldsymbol{G}^{\rightarrow} \cup \boldsymbol{G}^{\leftarrow}))$-tree, such that $\boldsymbol{S}^* \in$ pref$(\mathcal{C})$.*

4. Pruning of Candidate Trees

Depending on how a candidate tree is constructed, we might be able to prune it. In what follows we present some simple corollaries which allow for pruning of candidate trees.

Corollary 3. *Let $\mathcal{C}$ be a candidate for which pref$(\mathcal{C}) = \varnothing$. Then pref$(\mathcal{C}') = \varnothing$ for all nodes $\mathcal{C}'$ in any $\mathcal{C}$-tree.*

Thus, if during construction of a candidate tree, we create a candidate for which we know that pref$(\mathcal{C})$ is empty (e.g. by use of Theorems 6 or 7 below), then we do not have to construct the sub-tree rooted at that candidate.

Corollary 4. *Let $\mathcal{C}$ be a candidate. If $\boldsymbol{U}_\mathcal{C} = $ refl$(\mathcal{C})$, then pref$(\mathcal{C}) =$ pref$((\boldsymbol{I}_\mathcal{C}, \boldsymbol{O}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C}, \varnothing))$.*

Thus, if at some point in the construction of a candidate tree, we cannot find an argument to add to $\boldsymbol{I}_\mathcal{C}$, then we can stop exploring this branch of the tree.

Theorem 4. *Let $\mathcal{C}$ be a candidate. If $\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \subsetneq \boldsymbol{S}^*$, for some admissible set $\boldsymbol{S}^*$, then pref$(\mathcal{C}) = \varnothing$.*

Proof. Obvious from Definition 2. □

Theorem 5. *Let $\mathcal{C}$ be a candidate. If $\boldsymbol{I}_\mathcal{C}^{\leftarrow} \setminus (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C})^{\rightarrow} \neq \varnothing$ then pref$(\mathcal{C}) = \varnothing$.*

Proof. Assume otherwise, and let $\boldsymbol{S}^* \in$ pref$(\mathcal{C})$ and $A \in \boldsymbol{I}_\mathcal{C}^{\leftarrow} \setminus (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C})^{\rightarrow}$. As $A \in \boldsymbol{I}_\mathcal{C}^{\leftarrow}$ it follows that there is some argument $B \in \boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}^*$ and set $\boldsymbol{T} \subseteq \boldsymbol{I}_\mathcal{C} \subseteq \boldsymbol{S}^*$, such that $\boldsymbol{T} \cup \{A\} \triangleright B$. Furthermore, as $\boldsymbol{S}^*$ is a preferred extension, it defends itself, and thus attacks some argument in $\boldsymbol{T} \cup \{A\}$. But as $\boldsymbol{S}^*$ is conflict-free, this argument must be A, and A must thus be in $\boldsymbol{S}^{*\rightarrow} \subseteq (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C})^{\rightarrow}$, which is a contradiction. □

Theorem 6. *Let $\mathcal{C}$ be a candidate and $A \in \boldsymbol{U}_\mathcal{C}$. If*

- *for all sets $\boldsymbol{T}$, where $\boldsymbol{T} \triangleright A$, it holds that $\boldsymbol{T} \cap \boldsymbol{I}_\mathcal{C}^{\rightarrow} \neq \varnothing$, and*
- *$A \notin (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C})^{\rightarrow}$, and*
- *$A \notin (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \setminus \{A\})^{\leftarrow}$*

then pref$(\mathcal{C} - A) = \varnothing$.

Proof. Assume that there is a $\boldsymbol{S}^* \in$ pref$(\mathcal{C} - A)$, i.e. that $\boldsymbol{I}_{\mathcal{C}-A} \subseteq \boldsymbol{S}^* \subseteq \boldsymbol{U}_{\mathcal{C}-A}$, which implies that $A \notin \boldsymbol{S}^*$. Hence, either $\boldsymbol{S}^*$ does not defend A, or $\boldsymbol{S}^* \cup \{A\}$ is not conflict-free. We show that both cases are impossible.

Let $\boldsymbol{T}$ be some minimal attack on A. Since we have that $\boldsymbol{T} \cap \boldsymbol{I}_\mathcal{C}^{\rightarrow} \neq \varnothing$, $\boldsymbol{I}_\mathcal{C}$ attacks $\boldsymbol{T}$, and hence that $\boldsymbol{S}^* \supseteq \boldsymbol{I}$ defends A, ruling out the first case.

If $\boldsymbol{S}^* \cup \{A\}$ is not conflict-free, but $\boldsymbol{S}^*$ is, then there is a set $\boldsymbol{T} \subseteq \boldsymbol{S}^* \subseteq (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \setminus \{A\})$ and argument $B \in \boldsymbol{S}^* \subseteq (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \setminus \{A\})$, such that either $\boldsymbol{T} \triangleright A$, $\boldsymbol{T} \cup \{A\} \triangleright A$, or $\boldsymbol{T} \cup \{A\} \triangleright B$. But the latter of these is precluded by $A \notin (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C} \setminus \{A\})^{\leftarrow}$ and the others by $A \notin (\boldsymbol{I}_\mathcal{C} \cup \boldsymbol{U}_\mathcal{C})^{\rightarrow}$. □

Theorem 7. *Let $\mathcal{C}$ be a candidate and $A \in \boldsymbol{U}_\mathcal{C}$ an argument, which is attacked by at least one set of arguments. If, for all pairs of sets $\boldsymbol{T}$ and $\boldsymbol{R}$, where $\boldsymbol{T} \triangleright \boldsymbol{R}$ and $\boldsymbol{R} \triangleright A$, it holds that $\boldsymbol{T} \cap \boldsymbol{O}_\mathcal{C} \neq \varnothing$, then pref$(\mathcal{C} + A) = \varnothing$.*

Proof. Assume $\boldsymbol{S}^* \in \text{pref}(\mathcal{C} + A)$, implying that $\boldsymbol{I}_{\mathcal{C}+A} \subseteq \boldsymbol{S}^*$, i.e. $A \in \boldsymbol{S}^*$. As $\boldsymbol{S}^*$ is a preferred extension, it must defend A. Let $\boldsymbol{R}$ be an attack on A (whose existence is guaranteed by the assumptions of the theorem). Since $\boldsymbol{S}^*$ defends A, it follows that there is a set $\boldsymbol{T} \subseteq \boldsymbol{S}^*$ such that $\boldsymbol{T} \triangleright \boldsymbol{R}$. But then $\boldsymbol{T}$ and $\boldsymbol{R}$ fulfills the conditions in the theorem, and $\boldsymbol{T} \cap \boldsymbol{O}_{\mathcal{C}} \neq \varnothing$. It follows that $\boldsymbol{S}^* \cap \boldsymbol{O}_{\mathcal{C}} \neq \varnothing$, which implies that $(\boldsymbol{I}_{\mathcal{C}} \cup \boldsymbol{U}_{\mathcal{C}}) \cap \boldsymbol{O}_{\mathcal{C}} \neq \varnothing$, which contradicts that $\mathcal{C}$ is a candidate. □

It may be possible to establish further pruning rules, especially for families of concrete argumentation systems, where the attack relation is known to abide by some restrictions. Moreover, it might be possible to establish heuristics for checking the conditions in the above theorems, or construct data structures which allow for these to be easily tested in $\mathcal{C} + A$ and $\mathcal{C} - A$ given the answers in $\mathcal{C}$. However, this is outside the scope of this paper.

As mentioned before, the method for answering questions about preferred extensions, presented here, is based on candidate trees. The exact nature of constructing/walking the trees we leave unspecified, as it may be dependent on the question that we seek an answer to and the system at hand. In some cases it may be suitable to use a depth-first walk of a candidate tree, and in others (such as when $|\boldsymbol{A}| = \infty$) a breath-first or iterated deepening depth-first walk will be needed. However, even though we leave out an exact specification of our method, we show how to apply it to an example:

Example 5 (Full-blown Example): *We round off the example system $\mathcal{A}_e$, presented in Example 2, by identifying all preferred extensions for it. As no sets of arguments are attacking A it is clear that it belongs to the grounded extension of $\mathcal{A}$. We therefore set out with constructing a $\mathcal{C}$-tree, where $\mathcal{C}$ is a candidate having $\boldsymbol{I}_{\mathcal{C}} = \{A\}$, such as $(\{A\}, \{F\}, \{B, C, D, E\})$. We construct the tree in a depth-first manner. The final result is shown in Figure 2.*

First we construct $\mathcal{C} + B = (\{A, B\}, \{C, D, F\}, \{E\})$ and then $(\mathcal{C} + B) + E = (\{A, B, E\}, \{C, D, F\}, \varnothing)$. Here $\{A, B, E\}$ is an admissible set, and $\boldsymbol{U}_{(\mathcal{C}+B)+E}$ is empty, so the recursion stops. Next we would need to consider $(\mathcal{C}+B)-E$, but $\mathcal{C}+B$ and E satisfies the conditions in Theorem 6 so we know that the sub-tree rooted at $(\mathcal{C}+B)-E$ contains no preferred extensions, so we skip it.

Instead we back-track and construct $\mathcal{C} - B = (\{A\}, \{B, F\}, \{C, D, E\})$, $(\mathcal{C} - B) + C = (\{A, C\}, \{B, F\}, \{D, E\})$, and then $((\mathcal{C}-B)+C)+D = (\{A, C, D\}, \{B, E, F\}, \varnothing)$. This latter one contains an admissible set, viz. $\{A, C, D\}$. Next, we construct $((\mathcal{C}-B)+C)-D = (\{A, C\}, \{B, D, F\}, \{E\})$, which satisfies the conditions in Theorem 5 (the satisfying element being B). Therefore, we do not investigate that sub-tree any further. Instead we back-track and construct $(\mathcal{C} - B) - C = (\{A\}, \{B, C, F\}, \{D, E\})$ and then $((\mathcal{C} - B) - C) + D = (\{A, D\}, \{B, C, E, F\}, \varnothing)$. Here $\{A, D\}$ is not a preferred extension (it does not attack B which attacks it). Back-tracking one level, we construct $((\mathcal{C} - B) - C) - D = (\{A\}, \{B, C, D, F\}, \{E\})$. This candidate satisfies the condition in Theorem 4, as $\{A, E\}$ is a subset of $\{A, B, E\}$, which we discovered previously.

The analysis thus shows that the two admissible sets of $\mathcal{A}$ having no admissible set as supersets (i.e. the preferred extensions), are $\{A, B, E\}$ and $\{A, C, D\}$.

Due to the restriction to candidates and the pruning rules, in the example we were able to deduce the result from five total divisions (out of 64 theoretically possible di-

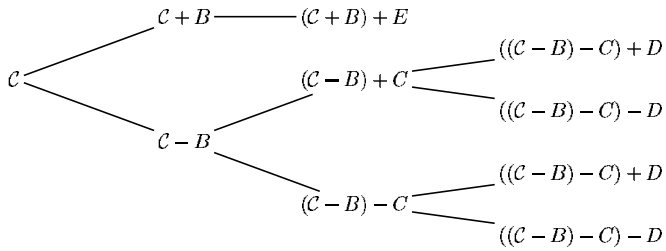


Figure 2. *Enumerating all preferred extensions of* $\mathcal{A}_e$.

visions), and with an overhead of five partial divisions. We think this is a satisfactory result, considering the highly intertwined nature of the example system. Of course, the *actual* efficiency of the method is influenced by a number of factors:

- How fast can the conditions in Theorems 4 to 7 be checked?
- In what order are candidates expanded. In the example above we went for exploring the largest sets as soon as possible, which allowed for ruling out sub-trees for smaller sets later on. Other heuristics may be better, depending on the problem being solved.

5. Conclusions

We have presented a method for enumerating the preferred extensions of argumentation system where joint attacks are allowed. We have proved that the method is complete and have presented a number of optimisation rules which should help reduce the running time of implementations. We do not claim that the set of these optimisation rules is complete, and acknowledge that details regarding implementation are still open for optimisation.

References

[1] C. Cayrol. On the relation between argumentation and non-monotonic coeherence-based entailment. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1443–1448. Morgan Kaufmann, 1995.

[2] R. Loui. Defeat among arguments: a system of defeasible inference. *Computational Intelligence*, 3:100–106, 1987.

[3] M. Elvang-Gøransson and A. Hunter. Argumentative logics: reasoning with classically inconsistent information. *Data and Knowledge Engineering*, 16:125–145, 1995.

[4] M. Elvang-Gøransson, P. Krause, and J. Fox. Dialectic reasoning with inconsistent information. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*, pages 114–121. Morgan Kaufmann, 1993.

[5] 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.

[6] F. Lin. An argument-based approach to non-monotonic reasoning. *Computational Intelligence*, 9:254–267, 1993.

[7] H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In *Handbook of Philosophical Logic*. Kluwer Academic Publishers, 2000.

[8] S. Benferhat, D. Dubois, and H. Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence*, pages 411–419. Morgan Kaufmann, 1993.

[9] J. Kohlas. Symbolic evidence, arguments, supports and valuation networks. In *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 186–198. Springer Verlag, 1993.

[10] P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11:113–131, 1995.

[11] D. V. Carbogim, D. Robertson, and J. Lee. Argument-based applications to knowledge engineering. *The Knowledge Engineering Review*, 15(2), 2000.

[12] 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.

[13] H. Prakken. Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese*, 127:187–219, 2001.

[14] L. Amgoud, N. Maudet, and S. Parsons. An argumentation-based semantics for agent communication languages. In *Proceedings of the Fifteenth European Conference on Artificial Intelligence*, 2002.

[15] P. McBurney. *Rational Interaction*. PhD thesis, Department of Computer Science, University of Liverpool, 2002.

[16] 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):321–358, 1995.

[17] L. Amgoud. *Contribution a l'integration des préferences dans le raisonnement argumentatif*. PhD thesis, Université Paul Sabatier, July 1999.

[18] L. Amgoud and C. Cayrol. On the acceptability of arguments in preference-based argumentation framework. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pages 1–7, 1998.

[19] L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proceedings of the Fourth International Conference on Multi-Agent Systems*, pages 31–38. IEEE Press, 2000.

[20] S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of formal inter-agent dialogues. *Journal of Logic and Computation*, 13(3):347–376, 2003.

[21] S. H. Nielsen and S. Parsons. A generalization of Dung's abstract framework for argumentation. In *Proceedings of Third Workshop on Argumentation in Multi-agent Systems*. Springer, 2006.

[22] Alexander Bochman. Collective argumentation and disjunctive logic programming. *Journal of Logic and Computation*, 13(3):406–428, 2003.

[23] Y. Dimopoulos, B. Nebel, and F. Toni. Preferred arguments are harder to compute than stable extension. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 36–43. Morgan Kaufmann, 1999.

[24] S. Doutre and J. Mengin. Preferred extensions of argumentation frameworks: Query, answering and computation. In *Proceedings of International Joint Conference on Automated Reasoning*, volume 2083, pages 272–288. Springer, 2001.

[25] G. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proceedings of the Seventh European Workshop on Logic in Artificial Inteligence, JELIA*, volume 1919, pages 239–253. Springer, 2000.

[26] C. Cayrol, S. Doutre, and J. Mengin. Dialectical proof theories for the credulous preferred semantics of argumentation frameworks. In *Proceedings of Sixth European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 668–679. Springer, 2001.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

An algorithm to compute minimally grounded and admissible defence sets in argument systems

Gerard A.W. Vreeswijk
CS Dept., Utrecht University

Abstract. This paper presents a query-answering algorithm to compute minimal lines of defence around an individual argument. The algorithm returns all such defence sets together with an indication whether the defence is grounded or admissible. For every argument encountered in the search process the algorithm further indicates whether that argument is IN, OUT, or UNDEC (undecided) according to the grounded semantics. The presentation of the algorithm is followed by a correctness proof and a complexity analysis of other than worst cases. The algorithm is already functional in argument analysis and visualization tools.

Keywords. Argumentation, algorithm, complexity

1. Introduction

Recently, a large number of new argument tools have been introduced. Most of these tools are meant to represent and visualize argument structures but do not show which arguments can actually be accepted [1,2].

The problem to decide which arguments may be accepted has two aspects. The first aspect, the theory, is concerned with questions such as which notions of acceptability there exist (grounded, admissible, preferred, stable, semi-stable) and how different notions of acceptability relate to each other. This part is relatively well understood [3,4,5]. The second aspect is involved with the design and analysis of algorithms that decide on acceptability. Here, the analysis falls apart in two approaches. The first approach is interested in the complexity of specific acceptability problems in worst cases. This direction is well sorted out by Dimopoulos *et al.* [6] and Dunne *et al.* [7,8]. The second approach is interested in the design of algorithms with the intention to actually use them in practice [9,10,11]. This paper follows the latter approach. More specifically, this paper proposes an algorithm that computes grounded and admissible defence sets in one pass (i.e., without walking the search tree twice) for single arguments. The presentation of the algorithm is followed by a correctness proof and a complexity analysis of other than worst cases.

Algorithms to compute grounded and/or preferred extensions have been proposed, among others, in [9,10,11,12]. However, these algorithms address one particular semantics and do not combine the search for different semantics. Indeed, surprisingly little implementations exists that actually compute credulous preferred acceptance. This paper is

connected such an implementation. Most of the existing algorithms, notably [9,12], are meant to compute full extensions, rather than to compute minimal lines of defence. Other algorithms, notably [10], are tailored to a specific argument paradigm. Finally, many proposed algorithms lack a complexity analysis. Exceptions are [6,7,8]. A problem with the latter approaches, however, is that they only address worst-case scenarios.

2. Basic concepts

The following is a rehearsal of known material, and is necessary to understand the algorithm and the motivation of the algorithm.

Definition 2.1 (Argument system) An *argument system* is a simple di-graph in which the nodes represent arguments, and the edges represent attack relations between arguments. The expression $a \leftarrow b$ is pronounced as "a is attacked by b".

An argument system may contain cycles and loops (1-cycles). Nodes adjacent to loops are called self-attacking arguments. If $a \leftarrow b$, this may be interpreted as a case in which the acceptance of argument b is a reason not to accept argument a. If X is a set of arguments such that no two arguments in X attack each other, then X is called *conflict-free*. If $a \leftarrow b$ and $b \leftarrow c$, we say that a is *defended* by c. If X is a set of arguments such that all elements are defended by (possibly other) elements in X, then X is called *self-defending*.

Caminada [13] argues convincingly that the following two axioms are fundamental.

Definition 2.2 (Reinstatement labeling) A *reinstatement labeling* is a function $L : A \rightarrow \{\text{IN}, \text{OUT}, \text{UNDEC}\}$ satisfying the following two properties:

1. An argument is IN iff all its attackers (if any) are OUT.
2. An argument is OUT iff one of its attackers is IN.

The two axioms together contain four implications so they are quite "heavy". Caminada proved that every reinstatement labeling L corresponds to a set $X = \{a \mid L(a) = \text{IN}\}$. Conversely, every set $X \subseteq A$ induces a reinstatement labeling L where all elements of X are labeled IN, and all elements that are attacked by elements of X are labeled OUT. The rest is labeled UNDEC. Different semantical notions such as grounded extensions, admissible sets, preferred extensions and stable extensions are common property and can be found in the literature or elsewhere in this volume.

The following results are from Caminada [13] and Dung [14] and are needed to motivate the semantics the algorithm operates on.

Result 2.1 (Caminada, 2006) *The following concepts are equivalent: (a) complete extensions; (b) reinstatement labelings.*

• *The following concepts are equivalent: (a) grounded extensions; (b) reinstatement labelings with minimal IN; (c) reinstatement labelings with minimal OUT; (d) reinstatement labelings with maximal UNDEC.*

• *The following concepts are equivalent: (a) preferred extensions; (b) reinstatement labelings with maximal IN; (c) reinstatement labelings with maximal OUT.*

• *The following concepts are equivalent: (a) stable extensions; (b) reinstatement labelings with empty UNDEC.*

Result 2.2 (Dung, 1995) *(1) An argument is in a complete extension iff it is in an admissible set; (2) an argument is in all complete extensions if it is in the grounded extension.*

3. Motivation

This section explains what a practical argument algorithm should compute and motivates these choices. To this end, we need to introduce the notion of a defence set. Let a be an argument. A *defence set* around a is a minimally admissible set D that contains a. A *grounded defence set* is a defence set without directed loops. Thus, if arguments a, b_1, b_2, $c_{1,1}$, $c_{1,2}$, $c_{2,1}$, are such that $a \leftarrow b_i$, $b_i \leftarrow c_{i,j}$ and $c_{1,2} \leftarrow b_1$ then $D_1 = \{a; c_{1,1}; c_{2,1}\}$ and $D_2 = \{a; c_{1,2}; c_{2,1}\}$ are defence sets of a of which only the first is grounded.

The are two types of algorithms, namely, query-based algorithms and total algorithms. Query-based algorithms compute answers for one particular argument, whether such answers are yes/no answers, defence sets or full extensions. Total algorithms compute answers for all arguments. This paper presents a query-based algorithm.

3.1. Semantics

When dealing with argument systems, questions often boil down to the following two fundamental problems:

1. Should this argument be accepted in all possible worlds? I.e., should everyone accept this argument?
2. Is there a possible world in which this argument must be accepted? I.e., can anyone defend this argument consistently?

If we assume that we are dealing with a reasonable audience (i.e., an audience that works with valid reinstatement labelings only) then the first question corresponds to the problem to determine whether a is labeled IN in all reinstatement labelings. By Result 2.1 we know that reinstatement labelings correspond to complete extensions, and by Result 2.2 we know that the intersection of all complete extensions is the grounded extension. Hence, the first question corresponds to the question whether a is contained in the grounded extension.

The second question corresponds to the question whether there exists a reinstatement labeling where a is labeled IN. By Result 2.1 we know that reinstatement labelings correspond to complete extensions, and by Result 2.2 we know that the membership question for complete extensions is equivalent to the membership question for admissible sets. Hence, the second question boils down to the question whether a is contained in an admissible set.

3.2. Multiplicity

The difference between query-answering algorithms and total algorithms has been explained at the beginning of Sec. 3. Both approaches have their pros and cons. A total algorithm would especially be relevant to GUI-based argument systems. In GUI based systems all arguments (and argument elements) that a user has constructed thus far are represented within the system and may be shown to the user. Such systems work with elements that are typically of a propositional nature and the number of elements constructed

in GUI based systems often remains within reasonable bounds. In such situations it is reasonable to expect that all argument elements have received a status-assignment and that this status-assignment is displayed in the GUI. On the other hand, there are argument systems that are based on first-order languages or equally expressive languages. These systems can only rely on query-answering algorithms. Arguments in first-order systems are constructed dynamically and therefore cannot be known in advance.

An additional advantage of a query-answering algorithm is that it can in principle take on the task of a total algorithm, simply by enumerating all arguments and querying each argument as it is enumerated. Work of Dimopoulos *et al.* [6] suggest that in terms of complexity such brute-force methods are perhaps the best one can achieve.

4. Algorithm

To explain the algorithm, we need the following concepts. Let x be an argument. A *candidate-solution* for x is a pair (L, s) where L is a conflict-free list of arguments without doublures such that $x \in L$ and $s \in \{\text{Grd}, \text{Adm}\}$. A *candidate-solution set* (CSS) for x is a (possibly empty) list of candidate solutions. A *solution* for x is a candidate solution (L, s) such that L is admissible. A *solution set* (S) is a (possibly empty) list of solutions, adjoined with an element from $\{\text{IN}, \text{OUT}, \text{UNDEC}\}$.

The input for the algorithm consists of a query r, together with a Dung-type argument system, that is, together with a simple di-graph. The algorithm returns a solution set S. Let r be the argument that is queried and let T be the tree induced by r from G. The algorithm is a depth-first search on T where restrictions on every node (called "permitted next move" in the algorithm) ensure that all branches are explored to a finite depth. These conditions are described now. Suppose either PRO or CON produces a new argument x in reply to other arguments. The following conditions are used to decide when candidate-solutions of the form $T = (L, s)$ may be dropped if x is further explored.

1. For both parties:
   (a) The argument x is attacked by one of PRO's arguments, i.e., $L \rightarrow x$.
2. For PRO:
   (a) The argument x attacks one of PRO's arguments, i.e. $L \leftarrow x$.
   (b) The argument x is attacked by a winning CON argument.

Condition (1) suffices in itself to ensure that the algorithm terminates and is correct. (The latter is proven in a moment.) The use of the Condition (2a-2b), then, is to terminate search at the earliest possible moment. The latter is important because a search for arguments in an expressive object-language is expensive.

From Condition (1) it further follows that CON may not repeat itself, and from that result, in turn, it follows that PRO does need to repeat itself throughout the entire search.

The algorithm itself consists of three parts: a pre-processing part (6-24), a loop (25-44), and a return part (45-51). The pre-processing part initializes local variables and verifies whether simple problem instances are encountered that may yield an immediate answer. If the argument under investigation has attackers and the CSS is non-empty, then attackers are explored one at a time. As long as the CSS is non-empty, results of attackers are collected inside the loop and are added to the running result or are used to expand

elements of the running result, depending on the parity of the argument. The return part returns the solution set and caches the result for possibly similar function calls in the future.

```
 1. function grd-adm(
 2.     argument,
 3.     branch,                      defaults to ∅
 4.     candidate-solutions          defaults to [[∅, Grd]]
 5. )

 6. on-pro-arg := length of branch is even ;
 7. push argument at end of branch ;
 8. status := IN ;

 #  First, verify whether argument contributes to candidate-solutions.
 #  If an argument is known to be IN or OUT or self-attacking, the candidate-
 #  solution set will in some cases be empty so that further search is unnecessary.
 9. if on-pro-arg
10.     if status-of(argument)==OUT
11.         candidate-solutions := ∅ ;
12.     else-if argument is self-attacking
13.         candidate-solutions := ∅ ;
14.         status := UNDEC if status==IN ;
15.     end-if
16. else
17.     candidate-solutions := ∅ if status-of(argument)==IN ;
18. end-if

19. if on-pro-arg
20.     add argument at the end of the first co-ordinate of each
21.     element in candidate-solutions
22. else
23.     accumulated-candidate-solutions := ∅ ;
24. end-if

25. if candidate-solutions ≠ ∅ and ATTACKERS(argument) ≠ ∅
26.     for-each attacker ∈ ATTACKERS(argument)

 #          If there is repetition in a branch, the solutions in candidate-
 #          solutions do no longer qualify as candidate-grounded solutions.
27.         if attacker ∈ branch
28.             status := UNDEC if status==IN ;
29.             change all Grd/Adm labels in candidate-solutions to Adm ;
30.         end-if

 #          Drop candidate-solutions that do longer qualify as
 #          candidate-admissible solutions.
```

```
31.          filtered := { s ∈ candidate-solutions |
32.              attacker is a permitted next move on s } ;

33.          next if filtered==∅ ; # Nothing left to explore

34.          intermediate, intermediate-status :=
35.              grd-adm(attacker, attacker, filtered) ;

 #           Status update is defined in Equation (1)
36.          status := update(status, intermediate-status) ;

37.          if on-pro-arg
38.              candidate-solutions := intermediate ;
39.              break for-each loop if candidate-solutions==∅ ;
40.          else
41.              accumulated-candidate-solutions ∪= intermediate ;
42.          end-if

43.      end-for-each
44.  end-if

45.  if on-pro-arg
46.      solution-set := candidate-solutions ;
47.  else
48.      solution-set := accumulated-candidate-solutions ;
49.  end-if

50.  pop argument at end of branch ;

51.  return [ solution-set, status ] ;
```

where status update is defined as follows:

$$\text{update}(\texttt{status}, \texttt{intermediate-status}) = \begin{cases} \text{OUT} & \text{if } \texttt{status} \neq \text{OUT} \text{ and } \texttt{intermediate-status} = \text{IN} \\ \text{UNDEC} & \text{if } \texttt{status} = \text{IN} \quad \text{and } \texttt{intermediate-status} = \text{UNDEC} \\ \texttt{status} & \text{otherwise.} \end{cases} \quad (1)$$

Many parts of the algorithm depend on the parity of the argument (i.e., whether the argument is owned by PRO or CON), so much so that it is worth considering to split the algorithm in a PRO side and a CON side. A split would eliminate if-then-else constructs that depend on argument parity. A disadvantage, however, is that common parts of the algorithm, when split, must be duplicated. To prevent such unnecessary duplications, I decided to present a monological version of the algorithm.

The notion of "permitted next move" on line 31 is defined by restrictions that are imposed on the expansion of candidate-solution sets, as described at the beginning of Sec. 4.

Examples of the algorithm can be obtained through the implementation site. (Cf. Sec. 6.)

5. Correctness

To prove correctness we need a number of concepts that demarcate parts of the search tree. Let x be an arbitrary node, and consider a specific stage in the search process at x. Let T_x^1 consist of all nodes that are visited at least once; let T_x^2 consist of all nodes that are visited twice; further let T_x^2 be transitively closed in the direction of the edges. Thus, the root is always in T_x^2. Further, let T_x^* contain x and be upwards closed. It may easily be verified that T_x^1 contains all its successors and that $T_x^2 \subseteq T_x^1$. Finally, let $T_x^{1-2} =_{Def} T_x^1 - T_x^2$. In this way, T_x^{1-2} is the arm (subset of a branch) from x to T_x^2.

Theorem 5.1 *Let a be the root argument. The solution set $\mathrm{S}(a)$ contains precisely all solutions for a.*

Proof. (Outline.) To set up an induction argument, we prove two claims.

Claim 1: upon entry, the candidate-solution set $\mathrm{CSS}(x)$ consists of all solutions for T_x^2, such that each solution is augmented with PRO-elements from T_x^{1-2}.
Claim 2: upon exit, the solution set $\mathrm{S}(x)$ consists of all solutions for $T_x^1 \cup T_x^*$.

Because a is the root node, we have $T_a^1 \subseteq T_a^*$, so that the second claim implies that the solution set $\mathrm{S}(a)$ indeed consists of all solutions for T_a^* which indeed would establish the desired result.

The two claims may be proven with induction on the number of node visits. Consider an arbitrary node x. There are three cases: x is the root, x is an internal node, or x is a leaf node. As an example let us consider the case where x is an internal node. Thus, $u \leftarrow x \leftarrow y_1, \ldots, y_n$ where "$\leftarrow$" denote attack relations and $n > 0$. With induction we may assume Claim 1 for u and Claim 2 for $y_1, \ldots, y_n$. We first prove Claim 1 for x. We may assume that $\mathrm{CSS}(u)$ contains all solutions for T_u^2, such that each solution is augmented with PRO-elements from T_u^{1-2}. Further, $T_x^2 = T_u^2$ because x and u are one the same branch. Since $T_x^1 = T_u^1 \cup \{x\}$ and $T_x^2 = T_u^2$, we are left to show that

$$\mathrm{CSS}(x) = \begin{cases} \{S \cup \{x\} \mid S \in \mathrm{CSS}(u)\} & \text{if } x \text{ is owned by PRO} \\ \mathrm{CSS}(u) & \text{otherwise.} \end{cases} \tag{2}$$

consists of all solutions for T_x^2, such that each solution is augmented with PRO-elements from T_x^{1-2}. If $x \in \mathrm{CON}$ there is nothing to prove; if $x \in \mathrm{PRO}$ then the claim follows from $T_x^{1-2} = T_u^{1-2} \cup \{x\}$ and the construction of $\mathrm{CSS}(x)$.

For internal nodes Claim 2 follows immediately from the induction hypothesis of $y_1, \ldots, y_n$ and the way how $\mathrm{S}(y_1), \ldots, \mathrm{S}(y_n)$ are combined into $\mathrm{S}(x)$. The minimality of $\mathrm{S}(x)$ follows from the way how intermediate CSSs are conditionally merged into $\mathrm{CSS}(x)$ (line 41 of the algorithm). $\mathrm{CSS}(x)$ then later becomes $\mathrm{S}(x)$.

The proofs of the two remaining cases run analogous except that Claim 2 for leaf nodes must be proven with the help of Claim 1 combined with the reason that caused the termination of the search at that branch, which can either be the absence of attackers, or the impossibility to make legal moves. □

6. Implementation

The algorithm has been implemented in the object-oriented scripting language Ruby. Its operation can be inspected through
`http://www.cs.uu.nl/~gv/code/grd_adm/`.

6.1. Testing

To test the implementation, a benchmark suite of typical argument systems (i.e., a collection of typical di-graphs) was composed. At April 2006, this collection consisted of 47 problems. Now and then the collection is extended with new problems thanks to the input of students and peers. Besides standard problems, the benchmark suite contains problems that are known to be computationally difficult or conceptually problematic [15,5]. The benchmarks can also be accessed through the above mentioned URL.

Although the collection of benchmark problems is mainly written for the purpose of testing the implementation (rather than the algorithm), experiments reveal that in many examples search is pruned either by (early) losses of PRO or else by constraints that either one of the two parties ran in to due to Conditions 1-2 (Sec. 4).

6.2. Practical use

By now the algorithm is used to compute defence sets in *Stevie*, a knowledge representation architecture for the construction of stories based on interpretation and evidence [16].

7. Complexity

In [17] it was proven that that the preferred membership problem—and hence the admissible membership problem—is NP-complete. From [17] one could conclude that the admissible membership problem has been "solved" and leave it at that. However, this is a non-productive viewpoint. Many argumentation tools are in need of an algorithm to compute grounded or admissible defences, and it may well be that in spite of the results from a worst case analysis there exist algorithms that perform acceptably in average or typical cases. This section shows that the algorithm indeed behaves exponentially in a worst case. However, it also shows that the algorithm seems to behave acceptably in other cases. This section concludes with a proposal of a definition of the average case.

7.1. Worst case

The following example shows that the algorithm may behave exponentially on the size of its input.

Example 7.1 (Outline.) Consider the family of argument systems with arguments a, b_i and $c_{i,j}$, such that $n \geq 1$, $1 \leq i \leq n$ and $j \in \{1, 2\}$, and such that $a \leftarrow b_i$ and $b_i \leftarrow c_{i,j}$ for all i, j. With a simple induction argument it can be proven that the total number of elements that needs to be verified is $\mathcal{O}(2^n)$.

7.2. Best case

Example 7.2 Let $n \geq 1$ and consider the argument system with elements a, b_1, b_2, $c_{2,1}, \ldots, c_{2,n}$, such that $a \leftarrow b_i$ and $b_2 \leftarrow c_{2,j}$.

Then a is defeated by b_1 so that further exploration of the sub-graph at b_2 is unnecessary. Hence, the input complexity goes to zero if n increases.

7.3. Other cases

This section presents examples that give insight in the decrease of complexity when one moves away from a worst case, and in the increase of complexity when one moves away from a best case.

The following example shows that the complexity drastically decreases when the worst-case example is slightly modified.

Example 7.3 Consider the collection of argument systems that arises when we take the argument system from Example 7.1 where some of the b_k's are no longer attacked at all.[1] Based on the distribution we may adopt the hypothesis that the probability that b_k is attacked equals $1/2$. If b_k is not attacked, then search immediately stops at k, and an empty set of CSSs is returned. Accordingly, the expected input complexity is

$$\frac{1}{1+3n} \sum_{k=1}^{n} \frac{1}{2^k} (3k2^{k-1} + (k+1)2^k) = \frac{5n^2 + 9n}{12n + 4}.$$

The last expression is $\mathcal{O}(n)$. □

I conclude with an example that generalizes the above approach and shows that in cases where PRO is likely to fail, the complexity of the algorithm still is acceptable.

Example 7.4 Consider the argument system that arises out of the tree T that is $2n$ levels deep and has a branching factor of B at even levels and a branching factor of 1 at odd levels. Thus, at every point, PRO's response is unique while CON may choose out of B responses.

If we denote the number of nodes in the tree owned by PRO by N, then $N = (B^{n+1} - 1)/(B - 1)$. It follows that the number of nodes in the tree owned by CON is $N - 1$ (for all PRO nodes are a unique response to a CON node, except the root node).

Since the depth of this tree is even, PRO would win every dispute. In such a case $2(1+2+3+\ldots+B^N)-1$ checks needed to be executed, which exponentially depends on N.

Now suppose that the probability that PRO is able to deliver a reply at all is, at any point, equal to p. This yields a collection of N argument systems, where the duration of the search typically depends on the place where PRO does not respond. For one particular system, he expected number of checks is two times $1 + (1 + 2p) + (1 + 2p + 3p^2) + (1 + 2p + 3p^2 + 4p^3 + \ldots + Np^{N-1})$ minus one. (I.e., one time calculated for PRO and then doubled for CON.) By applying the reduction formula for geometric series twice, this number can be reduced to an expression of $\mathcal{O}(np^n)$.

[1] Nudelman [18] calls this "Choosing a Hypothesis Space," which adequately reflects the subjectivity of the process.

Thus, PRO's success depends on N weakest links. If one such link fails we know that PRO failed entirely, so that the search stops at that point. Example 7.4 can be generalized further but this is an issue of further research.

7.4. Average case

This section presents a preliminary proposal for a definition of what constitutes to be an average case. However, it does not contain an analysis of such cases.

For the formulation of the average case I adopt the following three hypotheses:

1. Every argument has a finite number of attackers at its most.
2. The number of attackers of an arbitrary argument does not depend, on the average, on (the structure of) the argument, but on extra-logical factors such as the presence of logical material to formulate counter-arguments.
3. Two attackers of an argument are independent. Their existence does not depend on the existence of possibly other attackers for that argument.

In the theory of probability these three hypotheses are known as *ordinarity*, *stationarity*, and *lack of post-influence*, respectively, and are necessary (but not sufficient) requirements for a Poisson random distribution.

Definition 7.1 (Average case) Let N be an integer greater than zero and let $\lambda > 0$. The average input determined by (N, λ) is an argument system with arguments $A = \{a_1, \ldots, a_n\}$ with main query a_1 where the number of attackers of each argument is a Poisson random variable with mean λ where attackers are randomly selected from A.

Thus, $P(\#\text{attackers} = k) = e^{-\lambda}\lambda^k/k!$. It follows from Definition 7.1 that parts of the di-graph that are not reachable from a_1 through attackers are irrelevant for a complexity analysis. In particular, such parts need not be produced when one generates random input graphs in experiments. Instead, a straightforward way to produce random graphs would be to start at a_1 and generate attackers from there recursively.

8. Future research

The algorithm as well as the analysis of its complexity can be further improved.

8.1. Complexity of elementary checks

Currently, an elementary check amounts to verifying whether two arguments attack each other. In the first-order case, where arguments are dynamically generated, we may assume that the arguments to be checked are already generated and we may assume that their corresponding attack relation is cached.

Further, a closer look at Conditions 1-2 (Sec. 4) reveals that this particular implementation of an elementary check may be replaced by the simple verification whether an argument has been proposed by CON. We then have to take along all arguments proposed by CON but that should be no problem. If the algorithm can indeed be simplified thus, then the conceptual (not the computational) complexity of an elementary check would reduce considerably.

8.2. Empirical analysis of complexity

A possible line of research that was not explored in this article, is to empirically test the algorithm's complexity. An empirical analysis basically amounts to running the algorithm over multiple cases and measuring the amount of elementary computation steps the algorithm has executed on average. Nudelman [18] describes in detail how to such tests may be executed.

Section 6.1 (implementation and testing) and Definition 7.1 (average case) provide enough material to define experiments. Although there are several reasons in favor of running empirical complexity tests, I did not conduct such an empirical analysis. The reason not to report on empirical testing, apart from space considerations, is that I believe that the presentation of an algorithm must be accompanied by a conventional complexity analysis first, before it can be subject to practical tests.

9. Related research

Related research falls apart in two categories, viz. algorithms and complexity.

With respect to algorithms, Dung *et al.* [10] present a family of dialectic proof procedures for the admissibility semantics of assumption-based argumentation. Compared to [10], the algorithm presented in this paper returns all defence sets (instead of one) and interleaves the search for grounded and admissible defense in one pass. Further, Dung *et al.* is more focused on the construction of assumption-based arguments and on the fact that assumption-based arguments may share identical assumptions. Cayrol *et al.* [9] present decision algorithms (rather than query-answering or answer set algorithms) for credulous preferred acceptance and skeptical preferred acceptance in coherent argument systems. (An argument system is coherent iff preferred and stable extensions coincide.) Dunne *et al.* [7] showed that the problem to determine whether an argument system is coherent in the first place, is $\Pi_2^{(p)}$-complete. Every call of $\texttt{CredQA}_{\texttt{rec}}(R, d, O)$ (the function that computes credulous acceptability) contains four membership tests and two subset tests. Since such tests are known to be computationally expensive, it remains unclear how well these decision algorithms perform in practice. Verheij [12] presents argument software named ArguMed in which argument elements may be created and linked either positively (support) or negatively (attack). ArguMed contains an algorithm to compute all stable extensions (dialectical interpretations in Verheij's terminology). It does so by extending the grounded extension to (possibly different) stable extensions by means of a conventional breadth-first search procedure. It is known that plain breadth-first search procedures that suffers from multiple backtracking and trashing (resetting portions of a partial solution that had nothing to do with the reason for the failure). Unfortunately, [9,10,12] all lack a (partial) complexity analysis.

With respect to complexity, Dimopoulos *et al.* [6] and Dunne *et al.* [7,8] determine the complexity of worst-case argument decision problems, but do not venture into an analysis of average cases. Finally, [19,20] and later [18] are concerned with average-case complexity but then applied to problems outside the realm of argumentation.

Acknowledgement. Many thanks to colleagues and anonymous reviewers for sharing thoughts and counterexamples, particularly Bart Verheij, Matt South and Martin Caminada.

References

[1] P. A. Kirschner, S. J. Buckingham Shum, and C. S. Carr, editors. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. Springer-Verlag, London, 2002.

[2] C. A. Reed and G. W. A. Rowe. Araucaria: Software for argument analysis, diagramming and representation. *Int. Journal of AI Tools*, 14(3-4):961–980, 2004.

[3] C. I. Chesñevar, A. G. Maguitman, and R. P. Loui. Logical models of argument. *ACM Computing Surveys*, 32(4):337–383, 2000.

[4] J. L. Pollock. How to reason defeasibly. *Artificial Intelligence*, 57:1–42, 1992.

[5] H. Prakken and G. A. W. Vreeswijk. Logical systems for defeasible argumentation. In *Handbook of Philosophical Logic*, volume 4, pages 218–319. Kluwer, 2nd edition, 2002.

[6] Y. Dimopoulos, B. Nebel, and F. Toni. Finding admissible and preferred arguments can be very hard. In *Proc. of the 7th Int. Conf. on Principles of Knowledge Representation and Reasoning*, pages 53–61. Morgan Kaufmann, 2000.

[7] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(12):187–203, 2002.

[8] P. E. Dunne and T. J. M. Bench-Capon. Two party immediate response disputes: Properties and efficiency. *Artificial Intelligence*, 149(2):221–250, 2003.

[9] C. Cayrol, S. Doutre, and J. Mengin. On decision problems related to the preferred semantics for argumentation frameworks. *Journal of Logic and Computation*, 13(3):377–403, 2003.

[10] P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170(2):114–159, 2006.

[11] H. Jakobovits and D. Vermeir. Dialectic semantics for argumentation frameworks. In *Proc. of the 7th Int. Conf. on Artificial Intelligence and Law*, pages 53–62, New York, 1999. ACM Press.

[12] H. B. Verheij. Artificial argument assistants for defeasible argumentation. *Artificial Intelligence*, 150(1-2):291–324, 2003.

[13] M. Caminada. On the issue of reinstatement in argumentation. In *Proc. of the 10th European Conf. on Logics in AI (JELIA06)*. To appear., September 2006.

[14] 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.

[15] S. Doutre and J. Mengin. On sceptical vs. credulous acceptance for abstract argument systems. In *10th Int. Workshop on Non-Monotonic Reasoning (NMR 2004)*, pages 134–139, 2004.

[16] S. van den Braak and G. A. W. Vreeswijk. A knowledge representation architecture for the construction of stories based on interpretation and evidence. (To appear.). In *6th Workshop on Computational Models of Natural Argument (CMNA)*, 2006.

[17] Y. Dimopoulos and A. Torres. Graph theoretical structures in logic programs and default theories. *Theoretical Computer Science*, 170:209–244, 1996.

[18] E. Nudelman. *Empirical Approach to the Complexity of Hard Problems*. PhD thesis, Stanford University, Stanford, CA, October 2005.

[19] P. C. Cheeseman, B. Kanefsky, and W. M. Taylor. Where the really hard problems are. In J. Mylopoulos and R. Reiter, editors, *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI-91)*. Morgan Kaufmann, 1991.

[20] B. Selman, D. G. Mitchell, and H. J. Levesque. Generating hard satisfiability problems. *Artificial Intelligence*, 81(1-2):17–29, 2003.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Semi-Stable Semantics [1]

Martin Caminada [a]
[a] *Institute of Information and Computing Sciences, Utrecht University*

Abstract. In this paper, we examine an argument-based semantics called *semi-stable semantics*. Semi-stable semantics is quite close to traditional stable semantics in the sense that every stable extension is also a semi-stable extension. One of the advantages of semi-stable semantics is that there exists at least one semi-stable extension. Furthermore, if there also exists at least one stable extension, then the semi-stable extensions coincide with the stable extensions. This, and other properties, make semi-stable semantics an attractive alternative for the more traditional stable semantics, which until now has been widely used in fields such as logic programming and answer set programming.

Keywords. argumentation frameworks, argument based semantics, stable semantics, preferred semantics

1. Introduction

In the field of argumentation and defeasible reasoning, stable semantics is one of the oldest ways of determining which arguments or statements can be considered as justified. Well-known examples of formalisms in which stable semantics is applied are default logic [1] and stable models of logic programs [2]. Although alternative semantics have been stated over the years, like for instance grounded semantics which has its origins in Pollock's OSCAR [3] and in the well-founded semantics of logic programming [4], stable semantics has kept considerable support and is currently used even in relatively modern fields such as Answer Set Programming [5].

The popularity of stable semantics is not entirely without reason. It is a quite simple and straightforward semantics in which every argument is assigned a status of either `in` or `out` [6]. Furthermore, it is also a very credulous semantics in the sense that the intersection of the stable extensions is a superset of the intersection of the preferred extensions, which is in its turn a superset of the grounded extension. In some domains, like using argumentation for belief revision, one may prefer to use a credulous approach.

Nevertheless, stable semantics has its shortcomings, of which the potential absence of stable extensions is the most obvious one. Preferred semantics has been proposed as an alternative [7], but it has as a side effect that additional non-stable extensions can be introduced, even in situations where stable extensions already exist. An interesting question is whether one could find a semantics that is "backward compatible" to stable semantics in the sense that it is equivalent to stable semantics in situations where stable extensions exist and still yields a reasonable result (preferably quite close to stable) in

[1] This work has been sponsored by the EU ASPIC project.

situations where stable extensions do not exist. In this paper we show that a relatively simple and straightforward principle can be used to form the basis of such a semantics. We propose this semantics as a practical alternative for domains and applications where stable semantics is still being applied.

2. Basic Definitions

We first start with some basic definitions regarding abstract argumentation based on [7].

Definition 1 (argumentation framework). *An argumentation framework is a pair* (Ar, def) *where* Ar *is a finite set of arguments and* $def \subseteq Ar \times Ar$.

The shorthand notation A^+ and A^- stands for, respectively, the set of arguments defeated by A and the set of arguments that defeat A. If $\mathcal{A} \subseteq Ar$ then we write $(Ar, def)_{|\mathcal{A}}$ as a shorthand for $(\mathcal{A}, \{(A, B) \mid (A, B) \in def \text{ and } A, B \in \mathcal{A}\})$.

Definition 2 (defense / conflict-free). *Let* $A \in Ar$ *and* $Args \subseteq Ar$.
We define A^+ *as* $\{B \mid A\ def\ B\}$ *and* $Args^+$ *as* $\{B \mid A\ def\ B$ *for some* $A \in Args\}$.
We define A^- *as* $\{B \mid B\ def\ A\}$ *and* $Args^-$ *as* $\{B \mid B\ def\ A$ *for some* $A \in Args\}$.
$Args$ defends *an argument* A *iff* $A^- \subseteq Args^+$.
$Args$ *is* conflict-free *iff* $Args \cap Args^+ = \emptyset$.

In the following definition, $F(Args)$ stands for the set of arguments that are acceptable (in the sense of [7]) with respect to $Args$. Notice that the definitions of grounded, preferred and stable semantics are provided in terms of complete semantics, which has the advantage of making the proofs in the remainder of this paper more straightforward. Although these definitions are different from the ones provided by Dung [7], it is proved in the appendix that they are in fact equivalent to Dung's versions of grounded, preferred and stable semantics.

Definition 3 (acceptability semantics). *Let* $Args$ *be a conflict-free set of arguments and* $F : 2^{Args} \rightarrow 2^{Args}$ *be a function with* $F(Args) = \{A \mid A$ *is defended by* $Args\}$.

- $Args$ *is* admissible *iff* $Args \subseteq F(Args)$.
- $Args$ *is a* complete *extension iff* $Args = F(Args)$.
- $Args$ *is a* grounded *extension iff* $Args$ *is the minimal (w.r.t. set-inclusion) complete extension.*
- $Args$ *is a* preferred *extension iff* $Args$ *is a maximal (w.r.t. set-inclusion) complete extension.*
- $Args$ *is a* stable *extension iff* $Args$ *is a complete extension that defeats every argument in* $Ar \backslash Args$.

Note that there is only one grounded extension. It contains all the arguments which are not defeated, as well as those arguments which are directly or indirectly defended by non-defeated arguments.

We say that an argument is *credulously justified* under a particular semantics iff it is in at least one extension under this semantics. We say that an argument is *sceptically justified* under a particular semantics iff it is in each extension under this semantics.

3. Semi-Stable Semantics

The notion of semi-stable semantics, as put forward in the current paper, is quite similar to that of preferred semantics. The only difference is that not $\mathcal{Args}$ is maximized, but $\mathcal{Args} \cup \mathcal{Args}^+$.

Definition 4. *Let (Ar, def) be an argumentation framework and $\mathcal{Args} \subseteq Ar$. $\mathcal{Args}$ is called a* semi-stable extension *iff $\mathcal{Args}$ is a complete extension where $\mathcal{Args} \cup \mathcal{Args}^+$ is maximal.*

If $\mathcal{Args}$ is a complete extension, then $\mathcal{Args} \cup \mathcal{Args}^+$ is called its *range* — a notion first introduced by Bart Verheij [8].

The first thing to notice is that every stable extension is also a semi-stable extension.

Theorem 1. *Let $\mathcal{Args}$ be a stable extension of argumentation framework (Ar, def). $\mathcal{Args}$ is also a semi-stable extension of (Ar, def).*

Proof. Let $\mathcal{Args}$ be a stable extension of (Ar, def). Then $\mathcal{Args}$ is a complete extension that defeats every argument in $Ar \backslash \mathcal{Args}$. This means that $\mathcal{Args} \cup \mathcal{Args}^+ = Ar$. Therefore, $\mathcal{Args} \cup \mathcal{Args}^+$ is maximal (it cannot be a proper superset of Ar). Therefore, $\mathcal{Args}$ is a semi-stable extension. □

The converse of Theorem 1 does not hold. That is, it is not the case that each semi-stable extension is also a stable extension. This is illustrated by the following example.

Example 1. *Let (Ar, def) be an argumentation framework with $Ar = \{A, B, C, D\}$ and $def = \{(A, A), (A, C), (B, C), (C, D)\}$. A graphical representation is shown in figure 1. Here, $\{B, D\}$ is a semi-stable extension which is not a stable extension.*

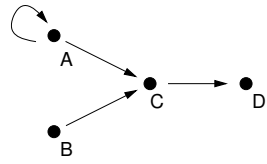


Figure 1. $\{B, D\}$ is a semi-stable but not a stable extension.

Another interesting property of semi-stable semantics is that every semi-stable extension is also a preferred extension.

Theorem 2. *Let $\mathcal{Args}$ be a semi-stable extension of argumentation framework (Ar, def). Then $\mathcal{Args}$ is also a preferred extension of (Ar, def).*

Proof. Let $\mathcal{Args}$ be a semi-stable extension of (Ar, def). Suppose $\mathcal{Args}$ is not a preferred extension of (Ar, def). Then there exists a set $\mathcal{Args}' \supsetneq \mathcal{Args}$ such that $\mathcal{Args}'$ is a complete extension. But from $\mathcal{Args}' \supsetneq \mathcal{Args}$ it follows that $\mathcal{Args}'^+ \supseteq \mathcal{Args}^+$. Therefore, $(\mathcal{Args}' \cup \mathcal{Args}'^+) \supsetneq (\mathcal{Args} \cup \mathcal{Args}^+)$. But then $\mathcal{Args}$ would not be a semi-stable extension, since $\mathcal{Args} \cup \mathcal{Args}^+$ would not be maximal. Contradiction. □

The converse of Theorem 2 does not hold. That is, it is not the case that every preferred extension is also a semi-stable extension. This is illustrated by the following example.

Example 2. *Let (Ar, def) be an argumentation framework with $Ar = \{A, B, C, D, E\}$ and $def = \{(A,B),(B,A),(B,C),(C,D),(D,E),(E,C)\}$. A graphical representation is shown in figure 2. Here, $\{A\}$ is a preferred extension which is not a semi-stable extension. The only semi-stable extension is $\{B, D\}$.*

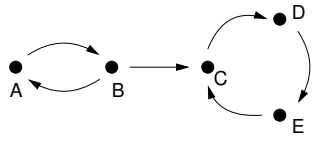


Figure 2. $\{A\}$ is a preferred but not a semi-stable extension.

The overall position of semi-stable semantics is shown in figure 3. Each stable extension is a semi-stable extension; each semi-stable extension is a preferred extension; each preferred extension is a complete extension and the grounded extension is a complete extension.

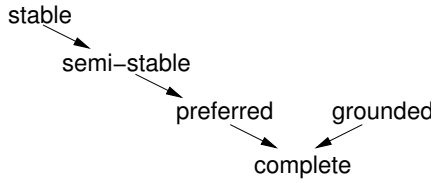


Figure 3. A brief overview of argument based semantics.

It is interesting to observe that in argumentation frameworks where there exists at least one stable extension, the semi-stable extensions coincide with the stable extensions.

Theorem 3. *Let (Ar, def) be an argumentation framework that has at least one stable extension. Let $SE = \{SE_1, \ldots, SE_n\}$ be the set of stable extensions and let $SSE = \{SSE_1, \ldots, SSE_m\}$ be the set of semi-stable extensions. It holds that $SE = SSE$.*

Proof. We need to prove that:

1. $SE \subseteq SSE$
   This follows directly from Theorem 1.
2. $SSE \subseteq SE$
   Let $SE_i \in SE$ (such an SE_i exists since it is assumed that (Ar, def) has at least one stable extension). It holds that $SE_i \cup SE_i^+ = Ar$. Therefore, every semi-stable extension SSE_i will also have to satisfy that $SSE_i \cup SSE_i^+ = Ar$ (otherwise $SSE_i \cup SSE_i^+$ would not be maximal). This means that every semi-stable extension is also a stable extension.

□

For every argumentation framework there exists at least one semi-stable extension. This is because there exists at least one complete extension, and a semi-stable extension is simply a complete extension in which some property (the union of itself and the arguments it defeats) is maximal.

Apart from the guaranteed existence of extensions, semi-stable semantics has yet another advantage to stable semantics. In determining whether an argument is sceptically or credulously justified with respect to semi-stable semantics, one only has to take into account arguments that are *relevant*.

Definition 5. *Let* (Ar, def) *be an argumentation framework. An argument* $A \in Ar$ *is* relevant *with respect to an argument* $B \in Ar$ *iff there exists an undirected path between* A *and* B.

In stable semantics, irrelevant arguments can influence whether an argument is justified or not. This is illustrated by the following example.

Example 3. *Let* (Ar, def) *be an argumentation framework with* $Ar = \{A, B, C, D\}$ *and* $def = \{(A, A)\}(B, C), (C, D)\}$. *A graphical representation is shown in figure 4. Here, arguments* B, C *and* D *are relevant with respect to each other, and argument* A *is not relevant with respect to* B, C *and* D. *Yet, argument* A *is the reason why there is no stable extension containing* B *and* D.

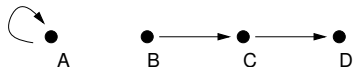


Figure 4. Stable semantics does not satisfy relevance.

Semi-stable semantics, however, does satisfy relevance. Irrelevant arguments have no influence whatsoever on the question whether an argument is justified under semi-stable semantics. To prove this, we first state two lemmas.

Lemma 1. *Let* (Ar, def) *be an argumentation framework, let* $A \in Ar$ *and* $\mathcal{A} \subseteq Ar$ *such that* $\mathcal{A}$ *is the set of arguments relevant with respect to* A. *If* $\mathcal{A}rgs$ *is a semi-stable extension of* (Ar, def) *then* $\mathcal{A}rgs \cap \mathcal{A}$ *is a semi-stable extension of* $(Ar, def)_{|\mathcal{A}}$.

Proof. Let $\mathcal{A}rgs$ be a semi-stable extension of (Ar, def). Suppose $\mathcal{A}rgs \cap \mathcal{A}$ is not a semi-stable extension of $(Ar, def)_{|\mathcal{A}}$. Then there exists a complete extension $\mathcal{A}rgs'$ of $(Ar, def)_{|\mathcal{A}}$ with $(\mathcal{A}rgs' \cap \mathcal{A}rgs'^{+}) \supsetneq (\mathcal{A}rgs \cap \mathcal{A}) \cup (\mathcal{A}rgs \cap \mathcal{A})^{+}$. As $\mathcal{A}$ is the largest (w.r.t. set inclusion) set of arguments that are relevant to each other, it holds that $(\mathcal{A}rgs \cap \mathcal{A}) \cup (\mathcal{A}rgs \cap \mathcal{A})^{+} = (\mathcal{A}rgs \cup \mathcal{A}rgs^{+}) \cap \mathcal{A}$. But then $\mathcal{A}rgs$ could not be a semi-stable extension because $\mathcal{A}rgs \cup (\mathcal{A}rgs' \backslash \mathcal{A})$ would be a complete extension with a larger range. Contradiction. □

Lemma 2. *Let* (Ar, def) *be an argumentation framework, let* $A \in Ar$ *and* $\mathcal{A} \subseteq Ar$ *such that* $\mathcal{A}$ *is the set of arguments relevant with respect to* A. *If* $\mathcal{A}rgs$ *is a semi-stable extension of* $(Ar, def)_{|\mathcal{A}}$ *then there exists a semi-stable extension* $\mathcal{A}rgs'$ *of* (Ar, def) *with* $\mathcal{A}rgs' \cap \mathcal{A} = \mathcal{A}rgs$.

Proof. Let $\mathcal{A}rgs$ be a semi-stable extension of $(Ar, def)_{|\mathcal{A}}$. Suppose there exists no semi-stable extension $\mathcal{A}rgs'$ of (Ar, def) with $\mathcal{A}rgs' \cap \mathcal{A} = \mathcal{A}rgs$. Then every complete extension $\mathcal{A}rgs'$ of (Ar, def) with $\mathcal{A}rgs' \cap \mathcal{A} = \mathcal{A}rgs$ does not have a maximal range. Let $\mathcal{A}rgs'$ be a complete extension of (Ar, def), with $\mathcal{A}rgs' \cap \mathcal{A} = \mathcal{A}rgs$, such that $\mathcal{A}rgs' \backslash \mathcal{A}$ is a semi-stable extension of $(Ar, def)_{|(Ar \backslash \mathcal{A})}$. Such an extension always exists since the arguments in $\mathcal{A}$ are not relevant with respect to the arguments in $Ar \backslash \mathcal{A}$. The fact that $\mathcal{A}rgs'$ is not a semi-stable extension of (Ar, def) means that there exists a complete extension with a bigger range. As the range of $\mathcal{A}rgs' \backslash \mathcal{A}$ is already maximal in $(Ar, def)_{|(Ar \backslash \mathcal{A})}$ this can only mean that the range of $\mathcal{A}rgs' \cap \mathcal{A}$ is not maximal in $(Ar, def)_{|\mathcal{A}}$. But as $\mathcal{A}rgs' \cap \mathcal{A} = \mathcal{A}rgs$ this means that $\mathcal{A}rgs$ would not be a semi-stable extension of $(Ar, def)_{|\mathcal{A}}$. Contradiction. □

Theorem 4. *Let (Ar, def) be an argumentation framework and let $A \in Ar$ and $\mathcal{A} \subset Ar$ such that $\mathcal{A}$ is the set of arguments that is relevant with respect to A.*

1. *There exists a semi-stable extension of (Ar, def) iff there exists a semi-stable extension of $(Ar, def)_{|\mathcal{A}}$.*
2. *A is in every semi-stable extension of (Ar, def) iff A is in every semi-stable extension of $(Ar, def)_{|\mathcal{A}}$.*

Proof. This follows directly from Lemma 1 and Lemma 2. □

As each semi-stable extension is also a preferred extension, a straightforward way of computing semi-stable semantics would be to compute all preferred extensions (using an algorithm like [9]) and then to determine which of these are also semi-stable. If one is only interested in whether an argument A is credulously or sceptically justified under semi-stable semantics, one does not have to take into account the entire argumentation framework. Instead, as stated by Theorem Theorem 4, one only has to take into account the arguments that are relevant with respect to A when calculating the preferred extensions. In many cases, however, there also exist alternative ways of determining whether an argument is credulously or sceptically justified under semi-stable semantics.

Theorem 5. *Let (Ar, def) be an argumentation framework, and let $A \in Ar$.*

1. *If A is in the grounded extension, then A is in every semi-stable extension.*
2. *If A is not part of an admissible set, then A is not in any semi-stable extension.*
3. *If A is part of an admissible set but is not defeated by any admissible set then there exists a semi-stable extension containing A.*

Proof.

1. This follows from the fact that the grounded extension is a subset of each complete extension [7], and the fact that each semi-stable extension is a complete extension.
2. This follows from the fact that each semi-stable extension is an admissible set.
3. The fact that A is not defeated by an admissible set also means that A is not defeated by a complete extension, and therefore that A is also not defeated by a semi-stable extension. That is, for any semi-stable extension $\mathcal{A}rgs$, it holds that $A \notin \mathcal{A}rgs^+$. The fact that A is part of an admissible set means that there is a preferred extension containing A. Let $\mathcal{A}rgs'$ be a preferred extension that

contains A and where (within the constraint that it contains A) $\mathcal{A}rgs' \cup \mathcal{A}rgs'^{+}$ is maximal. As for any semi-stable extension $\mathcal{A}rgs$ it holds that $A \notin \mathcal{A}rgs^{+}$, it also holds for any semi-stable extension not containing A that $A \notin \mathcal{A}rgs \cup \mathcal{A}rgs^{+}$. Thus, $\mathcal{A}rgs' \cup \mathcal{A}rgs'^{+}$ cannot be enlarged without losing A. Therefore, $\mathcal{A}rgs'$ is a semi-stable extension.

□

An example of point 3 of Theorem 5 can be found in Figure 2. Here, argument D is in an admissible set but is not defeated by an admissible set. This is because its only defeater (C) is not part of any admissible set. Hence, D is part of a semi-stable extension.

4. Discussion and Research Issues

The idea of semi-stable semantics is not entirely new. It is quite similar to Verheij's concept of an *admissible stage extension*, which fits within Verheij's general approach of using *stages* to deal with the issue of argument reinstatement [8].

Definition 6. *An* admissible stage extension *is a pair* $(\mathcal{A}rgs, \mathcal{A}rgs^{+})$ *where* $\mathcal{A}rgs$ *is an admissible set of arguments and* $\mathcal{A}rgs \cup \mathcal{A}rgs^{+}$ *is maximal.*

It can be shown that Verheij's approach of admissible stage extensions is in fact equivalent to the notion of a semi-stable semantics. This is stated and proved by Proposition 3 in the appendix.

Verheij also studied the relation between stable, semi-stable and preferred semantics, but has done so in terms of his stages approach, which received little following. This, and the fact that his work was published in a relatively small local conference has caused his work not to receive the attention that one may argue it should have received.

Semi-stable semantics can be seen as having a quite natural place within Dung's traditional semantics. One possible way of looking at the issue of argument reinstatement is to label each argument either `in`, `out` or `undec` according to the following postulate.

Postulate 1 ([6]). *An argument is labelled* `in` *iff all its defeaters are labelled* `out`*. An argument is labelled* `out` *iff it has a defeater that is labelled* `in`*.*

It can be shown that labellings satisfying this postulates coincide with complete extensions [6]. Furthermore, for labellings that satisfy Postulate 1 it holds that (1) those in which `in` is maximized coincide with preferred extensions, (2) those in which `out` is maximized coincide with preferred extensions, (3) those in which `undec` is maximized coincide with the grounded extension, (4) those in which `in` is minimized coincide with the grounded extension, and (5) those in which `out` is minimized coincide with the grounded extension. Semi-stable extensions then coincide with labellings in which `undec` is minimized (6).

One possible application of semi-stable semantics would be in the field Answer Set Programming [5]. The implementation of semi-stable semantics with respect to Answer Set Programming, however, involves more than just a change at the level of the abstract semantics. As logic programming, of which the Answer Set Programming approach can be seen as a special instance, can be regarded from the perspective of abstract argumen-

tation [7,10], the most obvious way of implementing semi-stable semantics would be at the level of the argumentation framework. Recent research, however, indicates that this may not be enough, since there is an issue regarding the potential violation of argumentation quality postulates [11,12]. For the well-founded semantics, this issue can be dealt with by stating syntactical restrictions on the content of the extended logic program in question [10]. One of our research aims is to study whether a similar approach is also possible in the context of semi-stable semantics.

References

[1] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.
[2] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proceedings of the 5th International Conference/Symposium on Logic Programming*, pages 1070–1080. MIT Press, 1988.
[3] J. L. Pollock. How to reason defeasibly. *Artificial Intelligence*, 57:1–42, 1992.
[4] A. van Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *J. ACM*, 38(3):620–650, 1991.
[5] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9(3/4):365–385, 1991.
[6] M.W.A. Caminada. On the issue of reinstatement in argumentation. Technical Report UU-CS-2006-023, Institute of Information and Computing Sciences, Utrecht University, 2006.
[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] B. Verheij. Two approaches to dialectical argumentation: admissible sets and argumentation stages. In J.-J.Ch. Meyer and L.C. van der Gaag, editors, *Proceedings of the Eighth Dutch Conference on Artificial Intelligence (NAIC'96)*, pages 357–368, Utrecht, 1996. Utrecht University.
[9] G. A. W. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proceedings of the 7th European Workshop on Logic for Artificial Intelligence (JELIA-00)*, number 1919 in Springer Lecture Notes in AI, pages 239–253, Berlin, 2000. Springer Verlag.
[10] M.W.A. Caminada. Well-founded semantics for semi-normal extended logic programs. In *Proceedings of the 11th International Workshop of Nonmonotonic Reasoning, special session on answer set programming*, 2006.
[11] M.W.A. Caminada and L. Amgoud. An axiomatic account of formal argumentation. In *Proceedings of the AAAI-2005*, pages 608–613, 2005.
[12] M.W.A. Caminada. Contamination in formal argumentation systems. In *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC)*, pages 59–65, 2005.

Appendix

Proposition 1. *Let* (Ar, def) *be an argumentation framework and let* $\mathcal{A}rgs \subseteq Ar$. *The following statements are equivalent:*

1. $\mathcal{A}rgs$ *is the grounded extension*
2. $\mathcal{A}rgs$ *is a minimal fixpoint of* F

Proof.

from 1 to 2: Let $\mathcal{A}rgs$ be the grounded extension. Suppose that $\mathcal{A}rgs$ is not a minimal fixpoint of F. Then there exists a proper subset $\mathcal{A}rgs' \subsetneq \mathcal{A}rgs$ which is a fixpoint of F. As $\mathcal{A}rgs$ is already the smallest fixpoint of F that is conflict-free, this can only mean that $\mathcal{A}rgs'$ is not conflict-free. But this is impossible as a subset of a conflict-free set is also conflict-free. Contradiction.

from 2 to 1: Let $\mathcal{A}rgs$ be a minimal fixpoint of F. As a monotonic increasing function like has a unique minimal fixpoint, the minimal fixpoint of F must be unique. From the previous point of this proof it then follows that the grounded extension is equivalent to this fixpoint.

□

Proposition 2. *Let* (Ar, def) *be an argumentation framework and let* $\mathcal{A}rgs \subseteq Ar$. *The following statements are equivalent:*

1. $\mathcal{A}rgs$ *is a preferred extension*
2. $\mathcal{A}rgs$ *is a maximal admissible set*

Proof. This follows from Theorem 25 of [7]. □

Proposition 3. *Let* (Ar, def) *be an argumentation framework and* $\mathcal{A}rgs \subseteq Ar$. *The following statements are equivalent:*

1. $\mathcal{A}rgs$ *is a semi-stable extension*
2. $\mathcal{A}rgs$ *is an admissible set of which* $(\mathcal{A}rgs, \mathcal{A}rgs^+)$ *maximal*

Proof.

from 2 to 1: A complete extension is a stronger condition than an admissible set, so we only need to prove that an admissible set $\mathcal{A}rgs$ where $\mathcal{A}rgs \cup \mathcal{A}rgs^+$ is maximal is also a complete extension. Suppose this is not the case. Then there must be an argument $B \notin \mathcal{A}rgs$ that is defended by $\mathcal{A}rgs$. This means that every argument C that defeats B is defeated by an argument in $\mathcal{A}rgs$. Therefore, $B \notin \mathcal{A}rgs^+$ (otherwise $\mathcal{A}rgs$ would not be conflict-free). This means that $\mathcal{A}rgs \cup \{B\}$ is conflict-free and self-defending, and thus an admissible set. But this would mean that $\mathcal{A}rgs$ is not an admissible set for which $\mathcal{A}rgs \cup \mathcal{A}rgs^+$ is maximal. Contradiction.

from 1 to 2: An admissible set is a weaker condition than a complete extension. We therefore only need to prove that maximality still holds under this weaker condition. Suppose that $\mathcal{A}rgs \cup \mathcal{A}rgs^+$ would not be maximal. This means there exists an admissible set $\mathcal{A}rgs'$ such that $(\mathcal{A}rgs' \cup \mathcal{A}rgs'^+) \supsetneq (\mathcal{A}rgs \cup \mathcal{A}rgs^+)$. From the previous point ("from 2 to 1") it follows that $\mathcal{A}rgs'$ would be a complete extension. But then $\mathcal{A}rgs$ would not have been a complete extension where $\mathcal{A}rgs \cup \mathcal{A}rgs^+$ is maximal. Contradiction.

□

Proposition 4. *Let* (Ar, def) *be an argumentation framework and let* $\mathcal{A}rgs \subseteq Ar$. *The following statements are equivalent:*

1. $\mathcal{A}rgs$ *is a stable extension*
2. $\mathcal{A}rgs$ *is a preferred extension that defeats every argument in* $Ar \backslash \mathcal{A}rgs$
3. $\mathcal{A}rgs$ *is an admissible set that defeats every argument in* $Ar \backslash \mathcal{A}rgs$
4. $\mathcal{A}rgs$ *is a conflict-free set that defeats every argument in* $Ar \backslash \mathcal{A}rgs$

Proof.

from 1 to 2: Let $\mathcal{A}rgs$ be a stable extension. This means that $\mathcal{A}rgs$ is a complete extension that defeats every argument in $Ar \backslash \mathcal{A}rgs$. Suppose that $\mathcal{A}rgs$ is not a preferred extension. That means that there is a complete extension $\mathcal{A}rgs' \supsetneq \mathcal{A}rgs$. But as $\mathcal{A}rgs$ defeats every argument in $Ar \backslash \mathcal{A}rgs$, this means that $\mathcal{A}rgs'$ would not be conflict-free and therefore could not be a complete extension. Contradiction.

from 2 to 1: Trivial (every preferred extension is also a complete extension).

from 2 to 3: From Theorem 2 it follows that a preferred extension is a (maximal) admissible set.

from 3 to 2: Let $\mathcal{A}rgs$ be an admissible set that defeats all arguments in $Ar \backslash \mathcal{A}rgs$. Suppose that $\mathcal{A}rgs$ is not a preferred extension. This means that there exists an admissible set $\mathcal{A}rgs' \supseteq \mathcal{A}rgs$. But as $\mathcal{A}rgs$ defeats all arguments in $Ar \backslash \mathcal{A}rgs$, this would mean that $\mathcal{A}rgs'$ is not conflict-free and therefore could not be an admissible set. Contradiction.

from 3 to 4: This follows directly from the fact that an admissible set is conflict-free.

from 4 to 3: Let $\mathcal{A}rgs$ be a conflict-free set that defeats all arguments in $Ar \backslash \mathcal{A}rgs$. Then, every argument that defeats $\mathcal{A}rgs$ is also defeated by $\mathcal{A}rgs$. This means that $\mathcal{A}rgs$ is an admissible set.

□

Argumentation Frameworks and Schemes I

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Reasoning in Argumentation Frameworks Using Quantified Boolean Formulas

Uwe Egly, Stefan Woltran
Institut für Informationssysteme 184/3
Technische Universität Wien
Favoritenstraße 9–11
A–1040 Wien, Austria
email: [uwe,stefan]@kr.tuwien.ac.at

Abstract. This paper describes a generic approach to implement propositional argumentation frameworks by means of quantified Boolean formulas (QBFs). The motivation to this work is based on the following observations: Firstly, depending on the underlying deductive system and the chosen semantics (i.e., the kind of extension under consideration), reasoning in argumentation frameworks can become computationally involving up to the fourth level of the polynomial hierarchy. This makes the language of QBFs a suitable target formalism since decision problems from the polynomial hierarchy can be efficiently represented in terms of QBFs. Secondly, several practicably efficient solvers for QBFs are currently available, and thus can be used as black-box engines in potential implementations of argumentation frameworks. Finally, the definition of suitable QBF modules provides us with a tool box in order to capture a broad range of reasoning tasks associated to formal argumentation.

1. Introduction

In daily life, we use arguments and counter-arguments in discussions in order to "convince" our opponent to our point of view. Argumentation frameworks [1] have been used to formalize the reasoning underlying argumentation. They provide what "convince" means and how arguments may be defeated by counter-arguments.

Reasoning underlying argumentation is a general principle. Many of the well-known non-monotonic reasoning formalisms [2,3] can be faithfully interpreted within argumentation frameworks [4]. Consequently, these frameworks formalize not only the mentioned reasoning underlying argumentation, but can be used to interpret, compare, and implement a wide range of different reasoning principles. Since the main difference between two distinct reasoning principles is the underlying derivability operator, the interpretation of both principles is generic except the definition of this operator. Therefore, argumentation frameworks provide not only a theoretical setting for studying different reasoning mechanisms, but also can be used as a practical underpinning for implementations. However, as shown by Dimopoulos, Nebel, and Toni [5], some combinations of derivability operators and notions of extensions make reasoning in such argumentation frameworks computationally involving as witnessed by hardness results up to the fourth level of the polynomial hierarchy.

In this paper, we propose an implementation of argumentation frameworks which is based on the satisfiability problem of quantified Boolean formulas (QBFs), an extension of classical propositional logic in which formulas may contain quantifications over propositional atoms. The motivation to consider QBFs is as follows:

First, in recent years we observed a parallel and mutually influencing development of QBF solvers on the one hand, and design of applications, on the other hand. This situation is similar to the emerge of the success of satisfiability solvers in the mid nineties, where first impressive results have been achieved by employing SAT solvers in the area of planning [6,7]. Since QBFs are a more expressive language than propositional logic, their range of application is naturally larger than that of SAT (under the reasonable assumption that reductions are computable in polynomial time). In fact, various problems from different areas have been considered as applications for QBFs, including conformant planning [8], inconsistency tolerance [9,10], nonmonotonic reasoning [11,12,13], verification [14,15], and theorem proving [16]. Moreover, there has been made a significant progress in the development of QBF solvers in the last few years [17].

Second, the different semantics captured by argumentation frameworks are all uniformly represented in our QBF setting. Our aim is, not at least, to illustrate how basic QBF modules can be used as building blocks for assembling realizations of numerous reasoning tasks in different instantiations of the framework. Notably, the different complexity behavior does not prohibit a uniform implementation method due to the power of QBFs and their solvers. In fact, our encodings provide highly complex but structured problems for benchmarking QBF solvers. Currently, such structured problems are barely going beyond the second level of the polynomial hierarchy.

The outline of the paper is as follows. After some formal preliminaries, we start with the description of abstract argumentation frameworks. The terminology is due to the fact that the underlying derivability operator remains *abstract* in the sense that only some necessary criteria have to be satisfied, but the operator is not specified in a concrete way. Then we provide corresponding abstract translation schemes by means of QBFs. Finally, we briefly describe some case studies, which instantiate the generic framework to propositional reasoning principles. In terms of the QBF framework, this is obtained by plugging in a QBF module which concretely describes the derivability operator of the respective formalism. Due to space restrictions, we shall only sketch these concrete realizations, which may serve as a basis for implementation by invoking QBF systems.

2. Formal Preliminaries

2.1. Quantified Boolean Formulas

Quantified Boolean formulas (QBFs) generalize ordinary propositional formulas by the admission of quantifications over propositional variables. In particular, the language of QBFs contains, for any atom p, unary operators of the form $\forall p$ and $\exists p$, called *universal* and *existential quantifiers*, respectively. Informally, a QBF of form $\forall p\, \exists q\, \Phi$ means that for all truth assignments of p there is a truth assignment of q such that Φ is true.

An occurrence of a propositional variable p in a QBF Φ is *free* iff it does not appear in the scope of a quantifier $\mathsf{Q}p$ ($\mathsf{Q} \in \{\forall, \exists\}$), otherwise the occurrence of p is *bound*. If Φ contains no free variable occurrences, then Φ is *closed*, otherwise Φ is *open*. Further-

more, we write $\Phi[p/\phi]$ to denote the result of uniformly substituting each free occurrence of the variable p in Φ by a formula ϕ. For a set $P = \{p_1, \ldots, p_n\}$ of propositional variables and a quantifier $\mathsf{Q} \in \{\forall, \exists\}$, we let $\mathsf{Q}P\,\Phi$ stand for the formula $\mathsf{Q}p_1\mathsf{Q}p_2 \cdots \mathsf{Q}p_n\,\Phi$.

By an *interpretation*, I, we understand a set of atoms. Informally, an atom p is true under I iff $p \in I$. In general, the truth value, $\nu_I(\Phi)$, of a QBF Φ under an interpretation I is recursively defined as follows:

1. if $\Phi = \top$, then $\nu_I(\Phi) = 1$;
2. if $\Phi = p$ is an atom, then $\nu_I(\Phi) = 1$ if $p \in I$, and $\nu_I(\Phi) = 0$ otherwise;
3. if $\Phi = \neg\Psi$, then $\nu_I(\Phi) = 1 - \nu_I(\Psi)$;
4. if $\Phi = (\Phi_1 \wedge \Phi_2)$, then $\nu_I(\Phi) = min(\{\nu_I(\Phi_1), \nu_I(\Phi_2)\})$;
5. if $\Phi = \forall p\,\Psi$, then $\nu_I(\Phi) = \nu_I(\Psi[p/\top] \wedge \Psi[p/\bot])$;
6. if $\Phi = \exists p\,\Psi$, then $\nu_I(\Phi) = \nu_I(\Psi[p/\top] \vee \Psi[p/\bot])$.

The truth conditions for $\bot$, $\vee$, $\rightarrow$, and $\leftrightarrow$ follow from the above in the usual way. We say that Φ is *true under* I iff $\nu_I(\Phi) = 1$, otherwise Φ is *false under* I. If $\nu_I(\Phi) = 1$, then I is a *model* of Φ. If Φ has some model, then Φ is said to be *satisfiable*. If Φ is true under any interpretation, then Φ is *valid*. Observe that a closed QBF is either valid or unsatisfiable, because closed QBFs are either true under each interpretation or false under each interpretation. Hence, for closed QBFs, there is no need to refer to particular interpretations. Two QBFs are *logically equivalent* iff they possess the same models.

In the same way as the satisfiability problem of classical propositional logic is the "prototypical" problem of NP, i.e., being an NP-complete problem, the satisfiability problem of QBFs in *prenex form* possessing $k-1$ quantifier alternations is the "prototypical" problem of the k-th level of the polynomial hierarchy,

Proposition 1 ([18]) *Given a propositional formula ϕ with its atoms partitioned into $i \geq 1$ sets $P_1, \ldots, P_i$, deciding whether $\exists P_1 \forall P_2 \ldots \mathsf{Q}_i P_i \phi$ is true is Σ_i^p-complete, where $\mathsf{Q}_i = \exists$ if i is odd and $\mathsf{Q}_i = \forall$ if i is even; deciding whether $\forall P_1 \exists P_2 \ldots \mathsf{Q}'_i P_i \phi$ is true is Π_i^p-complete, where $\mathsf{Q}'_i = \forall$ if i is odd and $\mathsf{Q}'_i = \exists$ if i is even.*

This complexity landscape can be extended to arbitrary closed QBFs if the maximal number of quantifier alternations along a path in the QBF's formula tree is taken into account. In turn, an arbitrary QBF can be transformed into an equivalent QBF in prenex form, although this transformation is not deterministic and crucial for the performance of QBF solvers requiring the input formula in this normal form (for details, see [19,20]).

Finally, we highlight the used reduction approach. Given a decision problem D, we aim at finding a translation scheme $\mathcal{T}_D$ into closed QBFs, such that

1. $\mathcal{T}_D(\cdot)$ is faithful, i.e., $\mathcal{T}_D(K)$ is true iff K is a yes-instance of D;
2. for each instance K, $\mathcal{T}_D(K)$ is computable in polynomial time with respect to the size of K; and
3. determining the truth of the QBFs resulting from $\mathcal{T}_D(\cdot)$ is not computationally harder (by means of Proposition 1) than the computational complexity of D.

2.2. Abstract Argumentation Frameworks

In this section, we introduce the notions around argumentation frameworks, where we basically follow the definitions in [5]. Abstract argumentation frameworks are defined on

top of a deductive system $(\mathcal{L}_\mathcal{A}, \mathcal{R})$, where $\mathcal{L}_\mathcal{A}$ is some formal language over an alphabet $\mathcal{A}$ and $\mathcal{R}$ is a set of inference rules inducing a monotonic derivability relation $\vdash$. For a theory $T \subseteq \mathcal{L}_\mathcal{A}$, we identify, as usual, its deductive closure by

$$Th(T) = \{\alpha \in \mathcal{L}_\mathcal{A} \mid T \vdash \alpha\}.$$

An *abstract (assumption-based) framework* is a triple $(T, A, \overline{(\cdot)})$, where $T, A \subseteq \mathcal{L}_\mathcal{A}$, with A being the set of *assumptions*, and $\overline{(\cdot)}$ is a mapping from A to $\mathcal{L}_\mathcal{A}$. For an $\alpha \in A$, $\overline{\alpha}$ is the *contrary* of α. An *extension* of a framework $(T, A, \overline{(\cdot)})$ is a theory $Th(T \cup S)$ with $S \subseteq A$. If no confusion can arise, an extension is often referred to as S alone. A set $S \subseteq A$ *attacks* an $\alpha \in A$ iff $T \cup S \vdash \overline{\alpha}$, and S attacks an $S' \subseteq A$ iff S attacks an $\alpha \in S'$. Consequently, S attacks *itself* iff there exists an $\alpha \in S$, such that S attacks α.

A set $S \subseteq A$ is *closed* iff $S = A \cap Th(T \cup S)$. Frameworks, in which it is guaranteed that each such S is closed, are called *flat*. Given a framework $(T, A, \overline{(\cdot)})$, a set $S \subseteq A$ is *stable* iff

1. S is closed,
2. S does not attack itself, and
3. S attacks each $\alpha \in A \setminus S$.

A set $S \subseteq A$ is *admissible* iff

1. S is closed,
2. S does not attack itself, and
3. for all closed $S' \subseteq A$, it holds that if S' attacks S, then S attacks S'.

Finally, S is *preferred* if it is admissible and maximal with respect to set inclusion. For a set $S \subseteq A$, which is stable (resp. admissible, preferred), the extension $Th(T \cup S)$ is called stable (resp. admissible, preferred).

A framework $(T, A, \overline{(\cdot)})$ is called *normal*, iff every maximal closed set not attacking itself is stable. Finally, a framework is *simple*, iff, for inconsistent T, there is no admissible extension, and otherwise there exists a least admissible extension $S = A \cap Th(T)$.

Given a framework $(T, A, \overline{(\cdot)})$, the *credulous reasoning problem* is to decide whether a given $\varphi \in \mathcal{L}_\mathcal{A}$ is contained in $Th(T \cup S)$ for *some* extension S. The *skeptical reasoning problem* is to decide whether $\varphi \in \mathcal{L}_\mathcal{A}$ is contained in $Th(T \cup S)$ for *all* extensions S.

The attentive reader might have observed that we did not define what kind of derivability operator is associated with the abstract argumentation framework. This is not an error but a feature. In the next section, we will continue with a translation of abstract argumentation frameworks to QBFs. These translations will again be independent from a concrete derivability operator which will come into the play when we instantiate the framework.

3. Abstract Translation Schemes to QBFs

In this section, we discuss the general encodings, leaving the concrete check for the derivability operator $\vdash$ unresolved. Afterwards we shall present some concrete realizations in detail.

Given the propositional language $\mathcal{L}_\mathcal{A}$ underlying an argumentation framework, we assume the language of QBFs, $\mathcal{L}_{\mathsf{QBF}}$, implicitly as defined over a sufficiently large al-

phabet, consisting of all propositional atoms $\mathcal{A}$ in $\mathcal{L}_\mathcal{A}$ plus a set of additional mutual disjoint *guessing variables* $\{g_\alpha \mid \alpha \in \mathcal{L}_\mathcal{A}\}$ which we duplicate whenever needed, i.e., $\{g'_\alpha \mid \alpha \in \mathcal{L}_\mathcal{A}\}$, $\{g''_\alpha \mid \alpha \in \mathcal{L}_\mathcal{A}\}$, etc.

We shall use these variables to guess sets of formulas: Given an interpretation I and a subset $S \subseteq \mathcal{L}_\mathcal{A}$, we say that I *characterizes* S iff it holds that $g_\alpha \in I$ iff $\alpha \in S$. More specific, given a set $T \subseteq \mathcal{L}_\mathcal{A}$ of propositional formulas and guessing variables $G = \{g_t \mid t \in T\}$, as well as an interpretation I over G, we implicitly assume that exactly those elements $t \in T$, where g_t is assigned to true in I, are contained in the currently guessed subset $S \subseteq T$. Hence, the possible interpretations over I characterize all possible subsets of T. For instance, the models of the QBF $\exists V(\bigwedge_{t\in T} g_t \rightarrow t)$, where V is the set of atoms occurring in T, characterize exactly the consistent subsets of T (see also Proposition 3 below, how the G's are concretely related with a theory T).

We proceed as follows: First, we define an abstract QBF module for encodings of $\vdash$, which is later replaced by concrete instantiations. Then we provide the general encodings for checking closure, the notion of attacking, and for characterizing stable, admissible, and preferred extensions. Afterwards, we briefly discuss simplifications for frameworks which are flat, normal, or simple.

Definition 1 *Let $(\mathcal{L}_\mathcal{A}, \mathcal{R})$ be a deductive system with an induced derivability relation $\vdash$, let $T, A \subseteq \mathcal{L}_\mathcal{A}$, and $\alpha \in \mathcal{L}_\mathcal{A}$. Moreover, let $2^{\mathcal{L}_\mathcal{A}}$ denote the power set of $\mathcal{L}_\mathcal{A}$. Then a function*

$$f^G\colon \quad 2^{\mathcal{L}_\mathcal{A}} \times 2^{\mathcal{L}_\mathcal{A}} \times \mathcal{L}_\mathcal{A} \;\mapsto\; \mathcal{L}_{\mathsf{QBF}}$$

is called an encoding *for $\vdash$, iff*

1. *$f^G(T, A, \alpha)$ has free variables $G = \{g_a \mid a \in A\}$, and,*
2. *for each interpretation I characterizing S via G, it holds that $f^G(T, A, \alpha)$ is true under I iff $T \cup S \vdash \alpha$.*

As an example, consider some theory T, a set $A = \{\beta, \gamma\}$, and an encoding f^G for $\vdash$ with free variables g_β, g_γ. Now consider, $f^G(T, A, \alpha)$ is true only under the following interpretations (over $\{g_\beta, g_\gamma\}$): $I_1 = \{g_\beta\}$, $I_2 = \{g_\gamma\}$, and $I_3 = \{g_\beta, g_\gamma\}$. Now since f^G is an encoding, we derive from these models that

(i) $T \cup \{\beta\} \vdash \alpha$,
(ii) $T \cup \{\gamma\} \vdash \alpha$, and
(iii) $T \cup \{\beta, \gamma\} \vdash \alpha$

hold, while $T \vdash \alpha$ does not hold since $f^G(T, A, \alpha)$ is not true under $I_0 = \emptyset$.

We are now well prepared to characterize all necessary ingredients for characterizing reasoning in argumentation frameworks via QBFs.

Theorem 1 *Let $F = (T, A, \overline{(\cdot)})$ be a framework over a deductive system inducing $\vdash$, f^G an encoding of $\vdash$ with free variables $G = \{g_a \mid a \in A\}$, and I an interpretation characterizing $S \subseteq A$ via G. Then the following holds.*

1. *S is closed iff I is a model of*

$$\mathit{closed}^G_F := \bigwedge_{a\in A} \big(g_a \leftrightarrow f^G(T, A, a)\big). \tag{1}$$

2. *S does not attack itself iff I is a model of*

$$noattack_F^G := \bigwedge_{a \in A} \big(g_a \rightarrow \neg f^G(T, A, \overline{a})\big). \tag{2}$$

3. *S is stable iff I is a model of*

$$stable_F^G := closed_F^G \ \wedge noattack_F^G \wedge \bigwedge_{a \in A} \big(\neg g_a \rightarrow f^G(T, A, \overline{a})\big) \tag{3}$$

$$:= closed_F^G \ \wedge \bigwedge_{a \in A} \big(g_a \leftrightarrow \neg f^G(T, A, \overline{a})\big). \tag{4}$$

4. *S is admissible iff I is a model of*

$$adm_F^G := closed_F^G \wedge noattack_F^G \wedge \forall G' \Big[closed_F^{G'} \wedge \Big(\bigvee_{a \in A} \big(g_a \wedge f^{G'}(T, A, \overline{a})\big)\Big) \rightarrow \Big(\bigvee_{a \in A} \big(g'_a \wedge f^G(T, A, \overline{a})\big)\Big)\Big]. \tag{5}$$

Observe that the third arguments in the functions f^G in $closed_F^G$ and $stable_F^G$ are different, i.e., we have a in (1) but $\overline{a}$ in (4). However, in some cases and in particular for a flat framework, the test for closure (i.e., the conjunct $closed_F^G$) can be removed from (4), resulting in

$$stable_F^G := \bigwedge_{a \in A} \big(g_a \leftrightarrow \neg f^G(T, A, \overline{a})\big). \tag{6}$$

Concerning admissible extensions, by applying Theorem 2 in [21], the encoding can now be considerably simplified for flat frameworks.

Proposition 2 *A set $S \subseteq A$ is admissible for a flat framework $(T, A, \overline{(\cdot)})$, iff S does not attack itself, and for the set $S' = \{\alpha \in A \setminus S \mid S \text{ does not attack } \alpha\} \cup S$, it holds that S' does not attack S.*

Theorem 2 *Let $F = (T, A, \overline{(\cdot)})$ be a flat framework over a deductive system inducing $\vdash$, f^G an encoding of $\vdash$ with free variables $G = \{g_a \mid a \in A\}$, and I an interpretation characterizing $S \subseteq A$ via G. Then S is admissible iff*

$$adm_F^G := noattack_F^G \wedge \exists G' \Big[\bigwedge_{a \in A} \Big(g'_a \leftrightarrow \big(g_a \vee \neg f^G(T, A, \overline{a})\big)\Big) \wedge \bigwedge_{a \in A} \big(g_a \rightarrow \neg f^{G'}(T, A, \overline{a})\big)\Big]\Big) \tag{7}$$

is true under I.

It remains to discuss the notion of preferred extensions. In order to encode the maximality test, which is employed to characterize preferred sets, we use the following concept.

Definition 2 *Let $G = \{g_a \mid a \in A\}$ and $G' = \{g'_a \mid a \in A\}$ be disjoint indexed sets of atoms with the same cardinality. Define*

$$G < G' := \bigwedge_{a \in A} \Big(g_a \rightarrow g'_a\Big) \wedge \neg \bigwedge_{a \in A} \Big(g'_a \rightarrow g_a\Big).$$

Note that an interpretation I is true under $G < G'$ iff, for each $a \in A$, it assigns g'_a to true, whenever g_a is assigned to true as well. Additionally, at least for one element a, g'_a is true under I but g_a is false under I. Hence, I characterizes two subsets of A where the set characterized by G is a proper subset of the set characterized by G'.

Theorem 3 *Let $F = (T, A, \overline{(\cdot)})$ be a framework over a deductive system inducing $\vdash$, f^G an encoding of $\vdash$ with free variables $G = \{g_a \mid a \in A\}$, and I an interpretation characterizing $S \subseteq A$ via G. Then S is preferred iff*

$$pref_F^G := adm_F^G \wedge \neg\exists G' \Big((G < G') \wedge adm_F^{G'}\Big)$$

is true under I.

Observe that the entire encoding now uses three copies of guessing variables, namely G, G', and also G'' which occurs in $adm_F^{G'}$. Also note that we can choose between two realizations of adm_F^G to be used in $pref_F^G$ depending whether F is flat (Theorem 2) or not (3. in Theorem 1). This leads to a different quantifier structure in $pref_F^G$ mirroring the different generic complexity results for the preferability semantics from [5].

To conclude this section, we turn our attention to the basic scheme to encode the reasoning problems. We denote by $stable(F)$ (resp. $adm(F)$, $pref(F)$) the set of stable (resp. admissible, preferred) extensions of a framework F.

Theorem 4 *Let $F = (T, A, \overline{(\cdot)})$ be a framework over a deductive system $(\mathcal{L}_\mathcal{A}, \mathcal{R})$, $\varphi \in \mathcal{L}_\mathcal{A}$, and $\mathcal{E} \in \{stable, adm, pref\}$. Then*

1. *φ is contained in some $E \in \mathcal{E}(F)$ iff $\exists G(\mathcal{E}_F^G \wedge f^G(T, A, \varphi))$ is true;*
2. *φ is contained in all $E \in \mathcal{E}(F)$ iff $\forall G(\mathcal{E}_F^G \rightarrow f^G(T, A, \varphi))$ is true.*

Another problem, we can solve immediately by combining the modules introduced above, is *coherence* [22], i.e., deciding whether for a given argumentation framework, each of its preferred extensions is also stable.

Theorem 5 *A framework $F = (T, A, \overline{(\cdot)})$ over a deductive system $(\mathcal{L}_\mathcal{A}, \mathcal{R})$ is coherent iff $\forall G(pref_F^G \rightarrow stable_F^G)$ is true.*

Recall that we have already discussed that, for flat frameworks, these encodings can be simplified. Moreover, there exist also shortcuts with respect to the encoded reasoning tasks. For instance, since any preferred extension is also admissible and any admissible extension is a subset of preferred extension, deciding whether φ is contained in some preferred extension is the same as deciding whether φ is contained in some admissible extension, which provides an easier encoding (i.e., an encoding with less quantifier alternations). As well, we can apply the (easier) stable encodings in order to deal with preferred extensions in the context of normal frameworks.

4. Examples for Encodings

In this section, we instantiate our abstract translation framework to concrete translations.

4.1. Basic Frameworks

First, we analyze the simple framework as treated, for instance, in [10,23]. In particular, we show that our generic approach coincides with the encodings to propositional logic given by Besnard and Doutre in [10] and thus generalizes their methodology.

We recall the definition of this basic framework.

Definition 3 *A* basic argumentation framework *is a pair* (A, R) *where* A *is a set of* arguments *and* $R \subseteq A \times A$. *If* $(a, b) \in R$ *then we say that* a attacks b. *A set* $S \subseteq A$ *attacks an argument* b *if some* $a \in S$ *attacks* b.

The attentive reader might have observed that we used the letter A for denoting the set of assumptions in the abstract framework as well as for denoting the set of arguments in basic frameworks in Definition 3. We will see that this usage is not misleading because the arguments in the basic framework play the role of assumptions in the abstract framework.

How can we represent a basic framework in our general abstract setting? We simply consider the logical system $(A, \emptyset)$, that is, the arguments are our basic vocabulary $\mathcal{L}_{\mathcal{A}}$ (i.e., a set of atomic formulas) and the set of (additional) inference rules is empty. This choice immediately implies that (i) $A \vdash a$ iff $a \in A$ and (ii) $Th(A) = A$. With a slight abuse of notation, we use $f^G(\emptyset, A, a) := g_a$ and, for any set of arguments B, $f^G(\emptyset, A, B) := \bigvee_{b \in B} g_b$ in the encodings. Then a basic argumentation framework (A, R) is simulated by the general framework $(\emptyset, A, \overline{(\cdot)})$ with $\overline{a} = \{b \mid (b, a) \in R\}$. Obviously, the framework is flat, and thus we can avoid the check for closure within our encodings.

We start with stable extensions, where our encoding (6) reduces to

$$stable_F^G := \bigwedge_{a \in A} \big(g_a \leftrightarrow \neg f^G(\emptyset, A, \overline{a})\big) := \bigwedge_{a \in A} \big(g_a \leftrightarrow (\neg \bigvee_{b \in \overline{a}} g_b)\big)$$

$$:= \bigwedge_{a \in A} \big(g_a \leftrightarrow (\bigwedge_{b:(b,a) \in R} \neg g_b)\big).$$

The latter formula coincides with the encoding from Proposition 5 in [10] by replacing the guessing atoms g_a with the corresponding atoms a, for each $a \in A$.

Admissible extensions for (A, R) are characterized using (7) from Theorem 2. By evaluating $f^G(\emptyset, A, \overline{a})$ and $f^{G'}(\emptyset, A, \overline{a})$, we get

$$adm_F^G := \bigwedge_{a \in A} \big(g_a \rightarrow (\neg \bigvee_{b:(b,a) \in R} g_b)\big) \wedge$$

$$\exists G' \Big[\bigwedge_{a \in A} \Big(g'_a \leftrightarrow \big(g_a \vee \neg \bigvee_{b:(b,a) \in R} g_b\big)\Big) \wedge \bigwedge_{a \in A} \Big(g_a \rightarrow \neg \bigvee_{b:(b,a) \in R} g'_b\Big)\Big].$$

Now we "plug in" the definition of the g'_a's from the first conjunct in the second line to the second conjunct. We then can omit these definitions and the existential quantifiers and get

$$adm_F^G := \bigwedge_{a\in A} \big(g_a \rightarrow \neg \bigvee_{b:(b,a)\in R} g_b\big) \wedge \bigwedge_{a\in A} \Big(g_a \rightarrow \neg \bigvee_{b:(b,a)\in R} \big(g_b \vee \neg \bigvee_{c:(c,b)\in R} g_c\big)\Big).$$

We rewrite the second conjunct of adm_F^G, viz.

$$\bigwedge_{a\in A} \Big(g_a \rightarrow \neg \bigvee_{b:(b,a)\in R} \big(g_b \vee \neg \bigvee_{c:(c,b)\in R} g_c\big)\Big)$$

to

$$\bigwedge_{a\in A} \Big(g_a \rightarrow \neg \bigvee_{b:(b,a)\in R} g_b\Big) \wedge \bigwedge_{a\in A} \Big(g_a \rightarrow \neg \bigvee_{b:(b,a)\in R} \bigwedge_{c:(c,b)\in R} \neg g_c\Big). \tag{8}$$

The first conjunct of this expression absorbs the first conjunct in adm_F^G, and thus adm_F^G is equivalent to (8), which itself is equivalent (module variable renaming as in the case of stable extensions above) to the encoding presented in [10].

Finally, for the encoding of the preferred extensions, we use a different concept as in [10], where the preferred extensions are characterized via *maximal* models of propositional formulas. Since we have the full power of QBFs, we can characterize these extensions via ordinary models using our encoding schema from above. Maximality is checked on the object level (i.e., within the resulting QBF). In particular, we get the following theorem.

Theorem 6 *Let (A, R) be an argumentation framework, $F = (\emptyset, A, \overline{(\cdot)})$ the corresponding abstract framework, and adm_G^F as reduced above. Moreover, let $S \subseteq A$, $I \subseteq G$, such that $a \in S$ iff $g_a \in I$ for each $a \in A$. Then S is preferred iff*

$$pref_F^G := adm_F^G \wedge \neg\exists G' \Big((G < G') \wedge adm_F^{G'}\Big)$$

is true under I.

Using our generic scheme, we additionally get immediately the encodings for the reasoning problems as discussed in Theorem 4.

4.2. Abductive Framework

We proceed with another simple framework, namely Theorist [24], which has been shown to be captured by abstract frameworks as follows. We use $(T, A, \overline{(\cdot)})$, with T and A being sets of propositional formulas. For each $a \in A$, $\bar{a}$ is just $\neg a$, and $\vdash$ is the classical derivability operator. We first have to encode $\vdash$.

Proposition 3 *For any propositional theories T and A, and for any formula φ, let V be the set of atoms occurring in T, A, or φ, and $G = \{g_a \mid a \in A\}$ be new atoms. Then,*

$$f^G(T, A, \varphi) := \forall V \Big(\big(\bigwedge_{t \in T} t \wedge \bigwedge_{a \in A} (g_a \rightarrow a) \big) \rightarrow \varphi \Big)$$

is an encoding of classical derivability in the sense of Definition 1.

With this instantiation, we can encode all reasoning tasks under consideration. In particular, we can characterize *extensions* in the sense of [24]. Such an extension is defined as $Th(T \cup S)$, where $S \subseteq A$ and S is a maximal subset of A (with respect to set inclusion), such that $T \cup S$ remains consistent. The relation to argumentation frameworks is as follows.

Proposition 4 ([4]) *Given a Theorist abductive framework (T, A), E is an extension of (T, A) iff E is a stable extension of the corresponding argumentation framework.*

Hence, using the concrete realization of the derivability operator from Proposition 3, we immediately obtain an encoding for Theorist-like extensions by plugging f^G into the abstract encodings for stable expansions given in Theorem 1.

4.3. Auto-epistemic Logic

We consider auto-epistemic logic (AEL) [2] in the context of argumentation frameworks. Then AEL has as the underlying language a modal logic with the modal operator L, but only the classical inference rules. As assumptions we have propositional atoms $L\alpha$ and $\neg L\alpha$. The contrary of $\neg L\alpha$ is α, and the contrary of $L\alpha$ is $\neg L\alpha$.

We instantiate our encodings for stable extensions following this framework. Consider $F = (T, A, \overline{(\cdot)})$, with T a modal theory, A containing literals $L\alpha$ and $\neg L\alpha$, for each subformula $L\alpha$ in T, and $\overline{(\cdot)}$ is defined as above. Since $\vdash$ is the classical inference operator, we use f_G as defined in Proposition 3. The exact relation between stable extensions of the framework and stable expansions of an auto-epistemic theory (cf. [2]) is as follows, see Theorem 3.11 in [4].

Proposition 5 *A theory E is a stable extension of the framework corresponding to a modal theory T iff E is a consistent stable expansion of T.*

Hence, our abstract encodings (together with the concrete realization for f^G as defined in Proposition 3) capture stable expansions of T. Moreover, one can show that these encodings reduce (after some simplifications) to the ones presented in [12].

However, AEL provides argumentation frameworks which are neither normal, simple, or flat. Thus, none of the previously mentioned shortcuts in the encodings can be applied and we end up, in the worst case, with QBFs possessing up to three quantifier alternations. In fact, this holds in the case of skeptical reasoning under preferred extensions, i.e., deciding whether a given formula φ is contained in all preferred extensions of a given argumentation framework. This problem was shown to be Π_4^p-complete [5] and our encodings match this intrinsic complexity. For illustration, we briefly sketch the structure of quantifier dependencies for the QBFs which encode this particular problem.

According to Theorem 4, we have formulas of the form $\forall G(pref_F^G \rightarrow f^G(T, A, \varphi))$. Observe that $pref_F^G$ has negative polarity in this formula. By inspecting the quantifiers in $pref_F^G$ (according to Theorem 3), we get additional quantifiers $\exists G' \forall G''$ the latter from the subformula $adm_F^{G'}$. The final quantifier $\exists V$ stems from the occurrences of the encodings of the classical derivability (which are present in both polarities within $adm_F^{G'}$). Hence, we end up here with quantifier dependencies $\forall G \exists G' \forall G'' \exists V$.

5. Discussion

Due to the lack of space, we just mention briefly some further potentials of our translation framework based on QBFs. First, the notions of admissible and preferred extensions provide additional semantics for non-monotonic modal logics (besides stable expansions). All of these semantics in turn can then be computed via QBFs using our generic framework. In the case of auto-epistemic logic, this is more or less straight forward since the corresponding argumentation framework relies on classical derivability, which we already encoded sufficiently in Proposition 3. With the obtained encodings, we may also provide an answer to a question raised in [5], i.e., how the preferred and admissible semantics of auto-epistemic logic relate to the semantics of parsimonious and moderately grounded expansions [25]. Since the latter have been reduced to QBFs in [12], we thus have a uniform axiomatization of all the systems in question.

A further application is to encode different tasks for default logic [3], which is also an instantiation of the abstract framework as shown in [4]. In this case, we have to express a different derivability operator, namely classical logic augmented with monotonic rules. Such an encoding has already been used in the literature, see for instance [11].

Finally, we briefly discuss an important question towards the concrete implementation of argumentation frameworks using QBF solvers. As is apparent by the presented encodings, the resulting QBFs are not in any specific normal form. However, most of the available QBF solvers require the input to be in *prenex conjunctive normal form*. Thus, a further transformation is necessary. This transformation is usually performed in two steps, namely *prenexing* and a transformation of the resulting purely propositional matrix into conjunctive normal form. The drawbacks of this transformation are an increase in both formula size and variable number, or, even worse, the formula's structure is disrupted. Moreover, prenexing cannot be carried out deterministically and the chosen normalization strategy crucially influences the runtimes (also depending on the concrete solver used), see e.g., [19]. However, there are a few solvers which are able to handle arbitrary QBFs (e.g., [20,26]) and recent results [27,20] show that non-normal form approaches are highly beneficial on certain instances. Future work thus includes a careful evaluation how QBF solvers of different types behave on encodings from our framework.

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*, 77(2):321–358, 1995.

[2] R. C. Moore. Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25:75–94, 1985.

[3] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1–2):81–132, 1980.
[4] 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.
[5] Y. Dimopoulos, B. Nebel, and F. Toni. On the computational complexity of assumption-based argumentation for default reasoning. *Artificial Intelligence*, 141(1/2):57–78, 2002.
[6] H. Kautz, D. McAllester, and B. Selman. Encoding plans in propositional logic. In *Proc. KR'96*, pages 374–384. Morgan Kaufmann Publishers, 1996.
[7] H. Kautz and B. Selman. Planning as satisfiability. In *Proc. ECAI'92*, pages 359–363. 1992.
[8] J. Rintanen. Constructing conditional pans by a theorem prover. *Journal of Artificial Intelligence Research*, 10:323–352, 1999.
[9] O. Arieli and M. Denecker. Reducing preferential paraconsistent reasoning to classical entailment. *Journal of Logic and Computation*, 13(4):557–580, 2003.
[10] P. Besnard and S. Doutre. Checking the acceptability of a set of arguments. In *Proc. NMR'04*, pages 59–64, 2004.
[11] U. Egly, T. Eiter, H. Tompits, and S. Woltran. Solving advanced reasoning tasks using quantified boolean formulas. In *Proc. AAAI'00*, pages 417–422. AAAI Press/MIT Press, 2000.
[12] T. Eiter, V. Klotz, H. Tompits, and S. Woltran. Modal nonmonotonic logics revisited: Efficient encodings for the basic reasoning tasks. In *Proc. TABLEAUX'02*, volume 2381 of *LNCS*, pages 100–114. Springer-Verlag, 2002.
[13] H. Tompits and S. Woltran. Towards implementations for advanced equivalence checking in answer-set programming. In *Proc. ICLP'05*, volume 3668 of *LNCS*, pages 189–203. Springer-Verlag, 2005.
[14] J. Katz, Z. Hanna, and N. Dershowitz. Space-efficient bounded model checking. In *Proc. DATE'05*, pages 686–687. IEEE Computer Society, 2005.
[15] A. C. Ling, D. P. Singh, and S. D. Brown. FPGA logic synthesis using quantified boolean satisfiability. In *Proc. SAT'05*, volume 3569 of *LNCS*, pages 444–450. Springer, 2005.
[16] U. Egly, R. Pichler, and S. Woltran. On deciding subsumption problems. *Annals of Mathematics and Artificial Intelligence*, 43(1–4):255–294, 2005.
[17] D. Le Berre, M. Narizzano, L. Simon, and A. Tacchella. The second QBF solvers comparative evaluation. In *Proc. SAT'04*, volume 3542 of *LNCS*, pages 376–392. Springer, 2005.
[18] C. Wrathall. Complete Sets and the Polynomial-Time Hierarchy. *Theoretical Computer Science*, 3(1):23–33, 1976.
[19] U. Egly, M. Seidl, H. Tompits, S. Woltran, and M. Zolda. Comparing different prenexing strategies for quantified boolean formulas. In *Proc. SAT'03. Selected Revised Papers*, volume 2919 of *LNCS*, pages 214–228, 2004.
[20] U. Egly, M. Seidl, and S. Woltran. A solver for QBFs in nonprenex form. Accepted for publication in *Proc. ECAI'06*, 2006.
[21] Y. Dimopoulos, B. Nebel, and F. Toni. Preferred arguments are harder to compute than stable extension. In *Proc. IJCAI'99)*, pages 36–43. Morgan Kaufmann Publishers, 1999.
[22] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(1/2):187–203, 2002.
[23] S. Doutre and J. Mengin. Preferred extensions of argumentation frameworks: Query answering and computation. In *Proc. IJCAR'01*, pages 272–288. volume 2083 of *LNAI*, pages 272–288. Springer, 2001.
[24] D. Poole. A logical framework for default reasoning. *Artificial Intelligence*, 36(1):270–48, 1988.
[25] T. Eiter and G. Gottlob. Complexity of reasoning with parsimonious and moderately grounded expansions. *Fundamenta Informaticae*, 17(1-2):31–53, 1992.
[26] M. Ghasemzadeh, V. Klotz, and C. Meinel. Embedding memoization to the semantic tree search for deciding QBFs. In *Proc. AI 2004*, volume 3339 of *LNCS*, pages 681–693, 2004.
[27] E. Giunchiglia, M. Narizzano, and A. Tacchella. Quantifier structure in search based procedures for QBFs. In *Proc. DATE'06*, 2006.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

A dialectic procedure for sceptical, assumption-based argumentation

Phan Minh Dung [a], Paolo Mancarella [b,*] and Francesca Toni [c]
[a] *AIT, Bangkok, Thailand*
[b] *Dipartimento di Informatica, Università di Pisa, Italy*
[c] *Department of Computing, Imperial College London, UK*

Abstract. We present a procedure for computing the sceptical "ideal semantics" for argumentation in assumption-based frameworks. This semantics was first proposed for logic programming in [1], extending the well-founded semantics. The proof procedure is defined by means of a form of dispute derivations, obtained by modifying the dispute derivations given in [2] for computing credulous admissible argumentation. The new dispute derivations are sound for the "ideal semantics" in all cases where the dispute derivations of [2] are complete for admissible argumentation. We prove that this is the case for the special kind of assumption-based frameworks with a finite underlying language and with the property of being "p-acyclic".

Keywords. Argumentation frameworks, Tools for argumentation

1. Introduction

We present a novel procedure for computing argumentation in the abstract, assumption-based frameworks of [3]. In these frameworks, arguments are built by means of deductions from assumptions, which are the components of the argument to be disputed by counter-arguments. These frameworks have been originally proposed for modelling default and legal reasoning [3,4], but have been equipped with powerful machinery for general-purpose argumentation in [2]. This machinery amounts to a procedure, in terms of *dispute derivations*, for computing arguments deemed acceptable according to the semantics of admissible sets of assumptions. This procedure uses *tight arguments*, which can be computed effectively by backward deductions.
The semantics of admissible sets of assumptions is *credulous*, in that it sanctions a set as acceptable if it can successfully dispute every argument against it, without disputing itself. However, there might be conflicting admissible sets. In some applications, it is more appropriate to adopt a *sceptical* semantics, whereby only beliefs sanctioned by all (maximally) admissible sets of assumptions are held. For example, in the legal domain, different members of a jury could hold different admissible sets of assumptions but a guilty verdict must be the result of sceptical reasoning. Also, in a multi-agent setting, agents may have competing plans (arguments) for achieving goals, and, when negotiating

*Correspondence to: P. Mancarella, Dipartimento di Informatica, Università di Pisa, Largo B. Pontecorvo, 3 I-56127 Pisa (Italy). Tel.: +39 050 2212710; Fax: +39 050 2212726; E-mail: paolo.mancarella@unipi.it

resources, they may decide to give away a resource only if that resource is not needed to support *any* of their plans.
Procedures for the computation of the sceptical semantics exist, e.g. the TPI procedure [12] for *coherent* argumentation frameworks [13]. However, to the best of our knowledge, no procedure exists for computing sceptical reasoning for non-coherent cases.
The procedure in this paper computes the sceptical *ideal semantics* for assumption-based frameworks. This is adapted from a corresponding semantics for logic programming, presented in [1]. The ideal semantics has the advantage of being easily computable, by a simple modification of the dispute derivations of [2], but without being overly sceptical. We prove that our procedure is sound for assumption-based frameworks with a finite underlying language and with no positive cycles (we call such frameworks p-acyclic). The proofs are omitted for lack of space, and are given in the accompanying report [5].

2. Background

In this section we briefly review the notion of assumption-based framework [3,4,6], how it applies to argumentation [2], the semantics of admissible sets of assumptions [3,9], and various possible sceptical semantics [3,9].
Any logic, viewed as a deductive system, can be extended to an assumption-based argumentation framework.

Definition 2.1 A **deductive system** is a pair $(\mathcal{L}, \mathcal{R})$ where

- $\mathcal{L}$ is a formal language consisting of countably many sentences, and
- $\mathcal{R}$ is a countable set of inference rules of the form $\alpha \leftarrow \alpha_1, \ldots, \alpha_n$ where $\alpha, \alpha_1, \ldots, \alpha_n \in \mathcal{L}$ and $n \geq 0$. [1] α is called the **conclusion** and $\alpha_1, \ldots, \alpha_n$ are called the **premises** of the inference rule.

If $n = 0$, then the inference rule represents an axiom. For notational convenience, we simply write α instead of $\alpha \leftarrow$.

Definition 2.2 A **deduction of a conclusion** α **based on a set of premises** P is a sequence $\beta_1, \ldots, \beta_m$ of sentences in $\mathcal{L}$, where $m > 0$ and $\alpha = \beta_m$, such that, for all $i = 1, \ldots, m$,

- $\beta_i \in P$, or
- there exists $\beta_i \leftarrow \alpha_1, \ldots, \alpha_n \in \mathcal{R}$ such that $\alpha_1, \ldots, \alpha_n \in \{\beta_1, \ldots, \beta_{i-1}\}$.

If there is a deduction of a conclusion α based on a set of premises P, we say that the deduction is **supported by** or **based upon** P.

Deductions are the basis for the construction of arguments, but to obtain an argument from a deduction its premises are restricted to ones that are acceptable as *assumptions*. In this paper, as in [2], we restrict ourselves to *flat* frameworks [3], whose assumptions do not occur as conclusions of inference rules. To specify when one argument attacks another, we need to determine when a sentence is the *contrary* of an assumption.

Definition 2.3 An **assumption-based framework** is a tuple $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ where

- $(\mathcal{L}, \mathcal{R})$ is a deductive system.

[1] [2] uses the equivalent notation $\frac{\alpha_1, \ldots, \alpha_n}{\alpha}$ for inference rules.

- $\mathcal{A} \subseteq \mathcal{L}$, $\mathcal{A} \neq \{\}$. $\mathcal{A}$ is the set of candidate **assumptions**.
- If $\alpha \in \mathcal{A}$, then there is no inference rule of the form $\alpha \leftarrow \alpha_1, \ldots, \alpha_n \in \mathcal{R}$.
- $\overline{\ }$ is a (total) mapping from $\mathcal{A}$ into $\mathcal{L}$. $\overline{\alpha}$ is the **contrary** of α.

Notice that, given an assumption α, $\overline{\alpha}$ may or may not be an assumption in general. Throughout the paper, following [2], we will illustrate our computational techniques by means of examples within *simplified* frameworks of the form $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$where:

- All sentences in $\mathcal{L}$ are atoms or negations of atoms (i.e. $\mathcal{L}$ is a set of **literals**).
- The contrary of any assumption p is $\neg p$; the contrary of any assumption $\neg p$ is p.

Definition 2.4 An **argument** is a deduction whose premises are all assumptions.

The only way to attack an argument is to attack one of its assumptions.

Definition 2.5

- An **argument** a **attacks an argument** b if and only if a attacks an assumption in the set of assumptions on which b is based.
- An **argument** a **attacks an assumption** α if and only if the conclusion of a is the contrary $\overline{\alpha}$ of α.
- A **set of assumptions** A **attacks a set of assumptions** B if and only if there exists an argument a based upon a set of assumptions $A' \subseteq A$ which attacks an assumption in B.

Note that the attack relationship between arguments depends solely on sets of assumptions. In some other approaches, however, such as that of Pollock [7] and Prakken and Sartor [8], an argument can attack another by contradicting its conclusion. Here, instead, such "rebuttal" attacks are reduced to "undermining" attacks, as described in [4,2].
The attack relationship is the basis of the *admissibility* semantics, first introduced in [9].

Definition 2.6

- A **set of assumptions** A is **admissible** if and only if
  A attacks every set of assumptions that attacks A, and
  A does not attack itself.
- A **belief** α is **admissible** if and only if there exists an argument for α based on a set of assumptions A_0, and A_0 is a subset of an admissible set A.

This semantics provides a non-constructive specification for which we need a practical, constructive and efficient, procedure. A major source of the non-constructivity of the specification is the monotonicity of deductive systems, implying that for every superset A' of the set of assumptions A that supports an argument a attacking another argument b, there exists an argument a' supported by A' that also attacks b. Thus, in general, there can be infinitely many arguments against another argument b. Moreover, for each such attack, there is the need to search among infinitely many candidate counter-attacks to find one that is successful. [2] proposes an alternative notion of argument, that lends itself to effective computation while maintaining correctness. This notion relies upon the use of a *selection function*, which, taken a (multi)set of sentences, returns a sentence in it.

Definition 2.7 Given a selection function:

- A **tight deduction** of a conclusion α is a (possibly infinite) sequence of multisets $S_1, \ldots, S_m, \ldots$, where $S_1 = \{\alpha\}$ and for every $1 \leq i < m$, where σ is the selected sentence occurrence in S_i:
  1. If σ is not an assumption then $S_{i+1} = S_i - \{\sigma\} \cup S$ for some inference rule of the form $\sigma \leftarrow S \in \mathcal{R}$ [2].
  2. If σ is an assumption then $S_{i+1} = S_i$.
- A **tight argument** of a conclusion α based on (or supported by) a set of assumptions A is a finite tight deduction $S_1, \ldots, S_m$ where $S_m = A$.

Tight arguments and ordinary arguments (as given in definition 2.4) are equivalent, as:
- for every tight argument of a conclusion α supported by a set of assumptions A there exists an argument of α supported by A;
- for every argument for a conclusion α supported by a set of assumptions A and for every selection function, there exists a tight argument of α supported by some subset $A' \subseteq A$.

Then, to show that a set of assumptions A is admissible, it suffices to consider only tight attacks against A and tight counter-attacks supported by assumptions in A. Indeed:

Theorem 2.1 A set of assumptions A is admissible if and only if
for every tight argument a that attacks A there exists a tight argument supported by $A' \subseteq A$ that counter-attacks a, and
no $A' \subseteq A$ supports a tight argument that attacks an assumption in A.

This theorem is the basis of the abstract procedure for argumentation via admissibility proposed in [2]. Intuitively, this is based on constructing dispute derivations between two players, the proponent $\mathcal{P}$ and the opponent $\mathcal{O}$. Here, given a sentence α to be proven to be an admissible belief, $\mathcal{P}_i$ intuitively corresponds to a multiset in a tight argument for α or counter-attacking an attack against the argument for α being constructed. Also, $\mathcal{O}_i$ corresponds to a set of multisets, each representing an argument potentially attacking the proponent's arguments. A_i is the set of all assumptions currently needed by the proponent to support its arguments. C_i is the set of all assumptions used by the opponent currently chosen to be counter-attacked by the proponent. Formally:

Definition 2.8 Given a selection function, a **dispute derivation of a defence set** A for a sentence α is a finite sequence of quadruples
$\langle \mathcal{P}_0, \mathcal{O}_0, A_0, C_0 \rangle, \ldots, \langle \mathcal{P}_i, \mathcal{O}_i, A_i, C_i \rangle, \ldots, \langle \mathcal{P}_n, \mathcal{O}_n, A_n, C_n \rangle$
where

$$\mathcal{P}_0 = \{\alpha\} \qquad A_0 = \mathcal{A} \cap \mathcal{P}_0 \qquad \mathcal{O}_0 = C_0 = \{\}$$
$$\mathcal{P}_n = \mathcal{O}_n = \{\} \qquad A = A_n$$

and for every $0 \leq i < n$, only one σ in $\mathcal{P}_i$ or one S in $\mathcal{O}_i$ is selected, and:

1. If $\sigma \in \mathcal{P}_i$ is selected then

   (i) if σ is an assumption, then

$$\mathcal{P}_{i+1} = \mathcal{P}_i - \{\sigma\} \qquad A_{i+1} = A_i \qquad C_{i+1} = C_i \qquad \mathcal{O}_{i+1} = \mathcal{O}_i \cup \{\{\overline{\sigma}\}\}$$

[2] We use the same symbols for multiset membership, union etc as for ordinary sets.

(ii) if σ is not an assumption, then there exists some inference rule $\sigma \leftarrow R \in \mathcal{R}$ such that $C_i \cap R = \{\}$ and

$\mathcal{P}_{i+1} = \mathcal{P}_i - \{\sigma\} \cup (R - A_i)$ $\quad$ $A_{i+1} = A_i \cup (\mathcal{A} \cap R)$
$C_{i+1} = C_i$ $\quad$ $\mathcal{O}_{i+1} = \mathcal{O}_i$.

2. If S is selected in $\mathcal{O}_i$ and σ is selected in S then

(i) if σ is an assumption, then

(a) either σ is ignored, i.e.

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\} \cup \{S - \{\sigma\}\}$ $\quad$ $\mathcal{P}_{i+1} = \mathcal{P}_i$
$A_{i+1} = A_i$ $\quad$ $C_{i+1} = C_i$

(b) or $\sigma \notin A_i$ and $\sigma \notin C_i$ and [3]

(b.1) if $\overline{\sigma}$ is not an assumption, then

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad$ $\mathcal{P}_{i+1} = \mathcal{P}_i \cup \{\overline{\sigma}\}$
$A_{i+1} = A_i$ $\quad$ $C_{i+1} = C_i \cup \{\sigma\}$

(b.2) if $\overline{\sigma}$ is an assumption, then

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad$ $\mathcal{P}_{i+1} = \mathcal{P}_i$
$A_{i+1} = A_i \cup \{\overline{\sigma}\}$ $\quad$ $C_{i+1} = C_i \cup \{\sigma\}$

(c) or $\sigma \notin A_i$ and $\sigma \in C_i$ [4]

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad$ $\mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad$ $A_{i+1} = A_i$ $\quad$ $C_{i+1} = C_i$

(ii) if σ is not an assumption, then

$\mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad$ $A_{i+1} = A_i$ $\quad$ $C_{i+1} = C_i$

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\} \cup \{S - \{\sigma\} \cup R \,|\, \sigma \leftarrow R \in \mathcal{R}$, and $R \cap C_i = \{\}\}$

Then, [2] proves that if there exists a dispute derivation for a sentence, then that sentence is an admissible belief (and the defence set computed by the derivation is admissible). The admissibility semantics is credulous, in that it deems a belief to be admissible whenever there exists *one* admissible set of assumptions supporting one argument for it. There are many applications where a credulous semantics is not appropriate, though. Many sceptical semantics for argumentation could be adopted, including

- the *grounded* semantics [3], defined in terms of all *complete extensions*. Complete extensions are admissible sets of assumptions A containing all assumptions α such that A counter-attacks all attacks against α;
- the *sceptical preferred* semantics [3], defined in terms of all *preferred extensions*, namely maximally admissible sets of assumptions.

[3] In [2], the condition $\sigma \notin C_i$ in case (b) and the case (b.2) were missing. Our new case here provides an additional filtering of culprits by culprits without affecting the correctness of the procedure. Moreover, case (b.2) takes into account the situation in which the contrary of the chosen culprit is an assumption in turn.

[4] In [2], this case (c) was missing. Our new case here provides an additional filtering of culprits by culprits without affecting the correctness of the procedure.

These semantics are sceptical in that they deem a belief to be held only if this belief is "agreed upon" by *all* extensions sanctioned by the semantics. In [10], we give abstract proof procedures for computing the grounded extension and the sceptical preferred semantics of a given assumption-based framework. The procedure for the sceptical preferred semantics works as follows, given a sentence α:

1. determine whether α is an admissible belief, by determining an admissible set Δ supporting an argument for α (this can be achieved by a dispute derivation);
2. let $\mathcal{D}$ be the set of all admissible sets of assumptions attacking Δ; check that, for each element E of $\mathcal{D}$, there exists an admissible set of assumptions $E' \supseteq E$ such that E' supports an argument for α;
3. if all tests at step 2. are successful, then succeed.

This abstract procedure is very expensive in practice, due to the need to compute $\mathcal{D}$ at step 2. In [10] we attempt to optimise the search for $\mathcal{D}$ by considering only tight attacks against Δ, namely by replacing step 2. above by

2′ let $\mathcal{T}$ be the set of all tight attacks against Δ; check that, for each element E of $\mathcal{T}$, there exists an admissible set of assumptions $E' \supseteq E$ such that E' supports an argument for α.

However, this optimisation is not correct in general, as shown by the following example.

Example 2.1 Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ be the assumption-based framework:

- $\mathcal{L} = \{a, b, c, f, \neg a, \neg b, \neg c, \neg f\}$
- $\mathcal{R}$ consists of
  $\neg a \leftarrow f \qquad \neg a \leftarrow b \qquad \neg b \leftarrow c \qquad \neg c \leftarrow b \qquad \neg f \leftarrow a$
- $\mathcal{A} = \{a, b, c, f\}$
- $\overline{a} = \neg a$, $\overline{b} = \neg b$, $\overline{c} = \neg c$, $\overline{f} = \neg f$.

It is easy to see that $\neg a$ does not hold in all preferred extensions of $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$, as it does not hold in the preferred extension $\{c, a\}$. If we apply the optimised algorithm above, though, this is not detected. Indeed, assume that $\Delta = \{b\}$ at step 1. (it is easy to see that this is admissible). b is attacked by the tight argument supported by the admissible $\{c\}$, thus $\mathcal{T}$ is $\{\{c\}\}$ at step 2′. Since this set can be extended to the admissible set $\{c, f\}$ in which $\neg a$ holds, the algorithm succeeds, giving an incorrect answer.

This example suggests that the sceptical preferred semantics is hard to compute in general. On the other hand, the grounded semantics is efficiently computable, but may be too sceptical, and thus not useful, in many cases, as illustrated by the following example.

Example 2.2 Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ be the assumption-based framework:

- $\mathcal{L} = \{a, b, c, \neg a, \neg b, \neg c\}$
- $\mathcal{R}$ consists of
  $a \leftarrow b \qquad a \leftarrow c \qquad \neg b \leftarrow c \qquad \neg c \leftarrow b$
- $\mathcal{A} = \{b, c\}$
- $\overline{b} = \neg b$, $\overline{c} = \neg c$.

There are two complete extensions, $\{b\}$ and $\{c\}$, both supporting a. But the grounded extension is $\{\}$ and does not support a.

In this paper, we consider an alternative sceptical semantics, defined in the next section.

3. The ideal semantics for argumentation

In [1] the ideal sceptical semantics for extended logic programs was introduced, generalising the well-founded semantics. Here, we adopt a similar approach to extend the grounded semantics for argumentation frameworks. Intuitively, the *ideal* sceptical semantics approximates better than the grounded semantics the intersection of all preferred extensions.

Definition 3.1

- An admissible set S of assumptions is **ideal** if and only if it is a subset of every preferred extension.
- A set of assumptions Δ is an **ideal extension** if and only if it is a maximal ideal set of assumptions.
- A **belief** α is **ideal** if and only if there exists an argument for α based on a set of assumptions Δ_0 and Δ_0 is a subset of an ideal extension Δ.

The ideal extension is unique and is a superset of the grounded extension. Thus, the ideal semantics is a good sceptical compromise. Moreover, as we will prove, it can be computed effectively by a simple modification of dispute derivations for admissibility. In example 2.1, $\neg a$ is not an ideal belief. Consider the following additional example.

Example 3.1 Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ be the assumption-based framework:

- $\mathcal{L} = \{a, b, c, d, \neg a, \neg b, \neg c, \neg d\}$
- $\mathcal{R}$ consists of
  $\neg a \leftarrow a \qquad \neg a \leftarrow b \qquad \neg b \leftarrow a \qquad \neg c \leftarrow d \qquad \neg d \leftarrow c$
- $\mathcal{A} = \{a, b, c, d\}$
- $\overline{a} = \neg a$, $\overline{b} = \neg b$, $\overline{c} = \neg c$, $\overline{d} = \neg d$.

There are two preferred extensions of $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$: $\{b, c\}$, $\{b, d\}$. Hence b, and thus $\neg a$, hold in the sceptical preferred semantics. b and $\neg a$ are also ideal beliefs, as $\{b\}$ is the ideal extension. Instead, the grounded extension is empty.

The following results are the basis for our proof procedure for the ideal semantics.

Theorem 3.1 α is an ideal belief if and only if the following conditions are both satisfied:

1. there is an admissible set Δ such that Δ supports an argument for α;
2. there is no admissible set of assumptions S such that S attacks Δ.

Theorem 3.2 α is an ideal belief if and only if the following conditions are both satisfied:

1. there is an admissible set Δ such that Δ supports an argument for α;
2. for each tight argument A attacking Δ there exists no admissible set S such that $S \supseteq A$.

A straightforward implementation of this result, whose correctness follows directly from theorem 3.2, is the following abstract procedure:

Algorithm 3.1 Given a sentence α:

1. Determine whether α is an admissible belief, by determining an admissible set Δ supporting an argument for α (this can be achieved by constructing a dispute derivation for α).

2. For each tight argument A attacking Δ check that there is no admissible sets S such that $S \supseteq A$.
3. If all tests at step 2. are successful, then succeed (α is an ideal belief).

We will use this procedure to provide a computational technique for sceptical argumentation with the ideal semantics, in terms of a form of dispute derivations defined next.

4. IS-Dispute Derivation

Before we introduce a dispute derivation for the ideal semantics (*IS-dispute derivation*) let us give a few new notations.
The notion of dispute derivation in definition 2.8 can be extended to a set of sentences S instead of just a single sentence α, by setting $\mathcal{P}_0$ to S. Then:

Notation 4.1 Let S be a set of sentences in $\mathcal{L}$. By *Fail(S)*, we mean that there exists no dispute derivation for S.

IS-dispute derivations are sequences of tuples of the form $\langle \mathcal{P}_i, \mathcal{O}_i, A_i, C_i, \mathcal{F}_i \rangle$, where

- the new component $\mathcal{F}_i$ holds all multisets S for which we want to prove that $Fail(S)$ (these are the potential tight attacks A in step 2. of procedure 3.1);
- $\mathcal{P}_i, \mathcal{O}_i, A_i, C_i$ are as in ordinary dispute derivations, except that sentences occurring in the multisets in $\mathcal{O}_i$ may be *marked*.

Notation 4.2 Given a set of sentences S:

- S_u is the set of *unmarked* sentences in S;
- $m(\sigma, S)$ is the set S where $\sigma \in S$ *becomes marked*;
- $u(S)$ is S where the marked sentences are *unmarked*.

Intuitively, IS-dispute derivations compute an admissible support for the given sentence α while trying to check that no admissible set attacks it. As soon as a (potential) attack is found, this is stored in the $\mathcal{F}$ component of the tuple to check that this fails to be/become admissible. Whenever a potential culprit is ignored in a potential attack, this is marked so that it will not be selected again. Selected elements in the potential attacks in the $\mathcal{O}$ component are chosen amongst the unmarked elements. Thus, we will impose that, given a multiset S in $\mathcal{O}_i$, the selection function will only select unmarked sentences in S_u.

Definition 4.1 Given a selection function, an **IS-dispute derivation of an ideal support** A for a sentence α is a finite sequence of tuples
$\langle \mathcal{P}_0, \mathcal{O}_0, A_0, C_0, \mathcal{F}_0 \rangle, \ldots, \langle \mathcal{P}_i, \mathcal{O}_i, A_i, C_i, \mathcal{F}_i \rangle, \ldots, \langle \mathcal{P}_n, \mathcal{O}_n, A_n, C_n, \mathcal{F}_n \rangle$
where

$$\mathcal{P}_0 = \{\alpha\} \qquad A_0 = \mathcal{A} \cap \mathcal{P}_0 \qquad \mathcal{O}_0 = C_0 = \mathcal{F}_0 = \{\}$$
$$\mathcal{P}_n = \mathcal{O}_n = \mathcal{F}_n = \{\} \qquad A = A_n$$

and for every $0 \leq i < n$, only one σ in $\mathcal{P}_i$ or one S in $\mathcal{O}_i$ or one S in $\mathcal{F}_i$ is selected, and:

1. If $\sigma \in \mathcal{P}_i$ is selected then

   (i) if σ is an assumption, then

$$\mathcal{P}_{i+1} = \mathcal{P}_i - \{\sigma\} \qquad A_{i+1} = A_i \qquad C_{i+1} = C_i$$
$$\mathcal{O}_{i+1} = \mathcal{O}_i \cup \{\{\overline{\sigma}\}\} \qquad \mathcal{F}_{i+1} = \mathcal{F}_i$$

(ii) if σ is not an assumption, then there exists some inference rule $\sigma \leftarrow R \in \mathcal{R}$ such that $C_i \cap R = \{\}$ and

$\mathcal{P}_{i+1} = \mathcal{P}_i - \{\sigma\} \cup (R - A_i)$ $\quad A_{i+1} = A_i \cup (\mathcal{A} \cap R)$ $\quad C_{i+1} = C_i$
$\mathcal{O}_{i+1} = \mathcal{O}_i$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i$

2. If S is selected in $\mathcal{O}_i$ and σ is selected in S_u then

(i) if σ is an assumption, then

(a) either σ is ignored, i.e.

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\} \cup \{m(\sigma, S)\}$ $\quad \mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad A_{i+1} = A_i$
$C_{i+1} = C_i$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i$

(b) or $\sigma \notin A_i$ and $\sigma \notin C_i$ and

(b.1) if $\overline{\sigma}$ is not an assumption, then

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad \mathcal{P}_{i+1} = \mathcal{P}_i \cup \{\overline{\sigma}\}$ $\quad A_{i+1} = A_i$
$C_{i+1} = C_i \cup \{\sigma\}$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i \cup \{u(S)\}$

(b.2) if $\overline{\sigma}$ is an assumption, then

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad \mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad A_{i+1} = A_i \cup \{\overline{\sigma}\}$
$C_{i+1} = C_i \cup \{\sigma\}$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i \cup \{u(S)\}$

(c) or $\sigma \notin A_i$ and $\sigma \in C_i$ and

$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\}$ $\quad \mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad A_{i+1} = A_i$
$C_{i+1} = C_i$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i \cup \{u(S)\}$

(ii) if σ is not an assumption, then

$\mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad A_{i+1} = A_i$ $\quad C_{i+1} = C_i$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i$
$\mathcal{O}_{i+1} = \mathcal{O}_i - \{S\} \cup \{S - \{\sigma\} \cup R \,|\, \sigma \leftarrow R \in \mathcal{R}\}$

3. If S is selected in $\mathcal{F}_i$ and $Fail(S)$ then

$\mathcal{O}_{i+1} = \mathcal{O}_i$ $\quad \mathcal{P}_{i+1} = \mathcal{P}_i$ $\quad A_{i+1} = A_i$
$C_{i+1} = C_i$ $\quad \mathcal{F}_{i+1} = \mathcal{F}_i - \{S\}$

Example 4.1 Consider the assumption-based framework in example 3.1. An IS-dispute derivation for $\neg a$ is $\langle \mathcal{P}_0, \mathcal{O}_0, A_0, C_0, \mathcal{F}_0 \rangle, \ldots, \langle \mathcal{P}_6, \mathcal{O}_6, A_6, C_6, \mathcal{F}_6 \rangle$ where

$\mathcal{P}_0 = \{\neg a\}$ $\quad A_0 = \{\}$ $\quad \mathcal{O}_0 = C_0 = \mathcal{F}_0 = \{\}$,

applying step (1.ii), with the second rule, we have

$\mathcal{P}_1 = \{b\}$ $\quad A_1 = \{b\}$ $\quad \mathcal{O}_1 = C_1 = \mathcal{F}_1 = \{\}$,

applying step (1.i), we have

$\mathcal{P}_2 = \{\}$ $\quad A_2 = \{b\}$ $\quad \mathcal{O}_2 = \{\{\neg b\}\}$ $\quad C_2 = \mathcal{F}_2 = \{\}$,

applying step (2.ii), we have

$\mathcal{P}_3 = \{\}$ $\quad A_3 = \{b\}$ $\quad \mathcal{O}_3 = \{\{a\}\}$ $\quad C_3 = \mathcal{F}_3 = \{\}$,

applying step (2.i.b.1), we have

$\mathcal{P}_4 = \{\neg a\}$ $A_4 = \{b\}$ $\mathcal{O}_4 = \{\}$ $C_4 = \{a\}$ $\mathcal{F}_4 = \{\{a\}\}$,

applying step (1.ii) using the second rule, we have

$\mathcal{P}_5 = \{\}$ $A_5 = \{b\}$ $\mathcal{O}_5 = \{\}$ $C_5 = \{a\}$ $\mathcal{F}_5 = \{\{a\}\}$,

applying step (3), $Fail(\{a\})$ is called (giving rise to a *Fail-dispute derivation* of $\{a\}$, given in example 6.1) and

$\mathcal{P}_6 = \{\}$ $A_6 = \{b\}$ $\mathcal{O}_6 = \{\}$ $C_6 = \{a\}$ $\mathcal{F}_6 = \{\}$,

Hence, $\{b\}$ is the computed ideal support for $\neg a$.

5. Soundness of IS-dispute derivations

IS-dispute derivations can be guaranteed to be sound for the ideal semantics if dispute derivation (for the computation of $Fail$) are complete for the admissibility semantics. As discussed in [2], dispute derivations are not complete in general. In this paper, we give a sufficient condition for their completeness, thus providing a sufficient condition for the soundness of IS-dispute derivations. For simplicity, we will restrict ourselves to the simplified frameworks used throughout the paper for the examples (see page).

Notation 5.1 Let $\mathcal{AF}$ be an assumption-based framework $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$. By $\mathcal{AF}^+$, we will denote the framework obtained by deleting all assumptions appearing in the premises of the inference rules of $\mathcal{R}$.

Below, given $\mathcal{AF}$, we use the notion of *dependency graph* of $\mathcal{AF}^+$, defined in a way similar to the *atom dependency graph* for logic programming (see, e.g. the review in [11]). The dependency graph of $\mathcal{AF}^+$ is a directed graph where:

- the nodes are the atoms occurring in $\mathcal{AF}^+$;
- a (directed) arc from a node p to a node q is in the graph if and only if there exists a rule $p \leftarrow B$ in $\mathcal{AF}^+$ such that q occurs in B.

Definition 5.1 An assumption-based framework $\mathcal{AF}$ is **positively acyclic** (or **p-acyclic** for short) if the dependency graph of $\mathcal{AF}^+$ is acyclic.

Lemma 5.1 Given a p-acyclic framework, there exists no infinite tight deduction.

In the case of p-acyclic frameworks with a finite underlying language $\mathcal{L}$ the dispute derivations of definition 2.8 are complete, in the following sense:

Theorem 5.1 Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ be an p-acyclic assumption-based framework such that $\mathcal{L}$ is finite. Then, for each literal α, if α is an admissible belief then

- there exists a dispute derivation for α;
- for each admissible set of assumptions Δ, if Δ supports an argument for α then there is a dispute derivation of defence set A for α such that $A \subseteq \Delta$ and A supports an argument for α.

We can then prove the correctness of IS-dispute derivation, for p-acyclic assumption-based frameworks with a finite underlying language.

Theorem 5.2 Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ } \rangle$ be p-acyclic with a finite $\mathcal{L}$. Suppose that there exists an IS-dispute derivation for α. Then α is an ideal belief.

6. Computing Fail(S)

$Fail(S)$ at step 3 of IS-dispute derivations can be computed by means of a new kind of dispute derivations, that we refer to as *Fail-dispute derivations*, obtained again by adapting the dispute derivations of [2].

Definition 6.1 Given a selection function, a **Fail-dispute derivation** of a multiset of sentences S is a sequence $\mathcal{D}_0, \ldots, \mathcal{D}_n$ such that each $\mathcal{D}_i$ is a set of quadruples of the form $\langle \mathcal{P}, \mathcal{O}, A, C \rangle$ where

$\mathcal{D}_0 = \{\langle S, \{\}, \mathcal{A} \cap S, \{\}\rangle\}, \qquad \mathcal{D}_n = \{\}$

and, for every $0 \leq i < n$, if a quadruple $Q = \langle \mathcal{P}, \mathcal{O}, A, C \rangle$ is selected in $\mathcal{D}_i$ then either $\mathcal{P} \neq \{\}$ or $\mathcal{O} \neq \{\}$, and

1. If an element S from $\mathcal{O}$ is selected, then
   (a) If $S = \{\}$ then $\mathcal{D}_{i+1} = \mathcal{D}_i - \{Q\}$
   (b) If $S \neq \{\}$ then let $\sigma \in S$ be the selected sentence in S:
      i. if σ is not an assumption then $\mathcal{D}_{i+1} = \mathcal{D}_i - \{Q\} \cup \{Q'\}$ where Q' is obtained from Q as in step (2.ii) of definition 2.8;
      ii. if σ is an assumption then $\mathcal{D}_{i+1} = \mathcal{D}_i - \{Q\} \cup \{Q_0, Q_1\}$ where Q_0 is obtained from Q as in step (2.i.a) and Q_1 are obtained from Q as in steps (2.i.b) or (2.i.c) (as applicable) of definition 2.8;
2. If an $\sigma \in \mathcal{P}$ is selected, then
   (a) if σ is an assumption then $\mathcal{D}_{i+1} = \mathcal{D}_i - \{Q\} \cup \{Q'\}$ where Q' is obtained from Q as in step (1.i) of definition 2.8;
   (b) if σ is not an assumption then $\mathcal{D}_{i+1} = \mathcal{D}_i - \{Q\} \cup \{Q' \mid$ there is a rule $\sigma \leftarrow R$ such that Q' is obtained from Q as in step (1.ii) of definition 2.8$\}$.

Theorem 6.1 There exists a Fail-dispute derivation for a multiset of sentences S if and only if there is no dispute derivation for S.

Example 6.1 Consider the assumption-based framework in example 3.1. We show here a Fail-dispute derivation of $\{a\}$.

$\mathcal{D}_0 = \{ \langle \{a\}, \{\}, \{a\}, \{\} \rangle \}$	applying step 2, we have:
$\mathcal{D}_1 = \{ \langle \{\}, \{\{\neg a\}\}, \{a\}, \{\} \rangle \}$	applying step (1.b), we have:
$\mathcal{D}_2 = \{ \langle \{\}, \{\{a\}, \{b\}\}, \{a\}, \{\} \rangle \}$	applying step (1.b) by selecting $S = \{a\}$ in $\{\{a\}, \{b\}\}$ we have:
$\mathcal{D}_3 = \{ \langle \{\}, \{\{\}, \{b\}\}, \{a\}, \{\} \rangle \}$	applying step (1.a) by selecting $S = \{\}$, we have:[5]
$\mathcal{D}_4 = \{\}$.	

7. Conclusions

We have proposed a new proof procedure for computing the ideal semantics for argumentation in assumption-based frameworks, adapted from [1]. We have argued that this

[5]Notice that step (2.i.b) and (2.i.c) are not applicable in this case.

is a good semantics for performing sceptical argumentation, as it is easily computed and is not overly sceptical.
The proof procedure is defined in terms of IS-dispute derivations and Fail-dispute derivations, both adapted from the dispute derivations of [2]. All these derivations extend and generalise standard SLD-based derivations in logic programming, as discussed in [2].
We have proven that it is sound for assumption-based frameworks with a finite underlying language and p-acyclic. In order to prove this soundness result, we have proven a novel completeness result, for p-acyclic frameworks, for the proof procedure proposed in [2].
There are a number of existing tools for computing sceptical argumentation, notably [12], [13] and [14]. These tools are proven to be sound and complete for *coherent* frameworks [9], i.e. frameworks for which the preferred and stable semantics coincide. Instead, our procedure is sound for non-coherent frameworks too, as soon as they are p-acyclic.

Acknowledgements
This work was partially funded by the Sixth Framework IST programme of the EC, under the 035200 ARGUGRID project. The third author has also been supported by a UK Royal Academy of Engineering/Leverhulme Trust senior fellowship.

References

[1] José Júlio Alferes, Phan Minh Dung, and Luís Moniz Pereira. Scenario semantics of extended logic programs. In A. Nerode and L. Pereira, editors, *Proc. 2nd International Workshop on Logic Programming and Non-monotonic Reasoning*, pages 334–348. MIT Press, 1993.
[2] Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170:114–159, 2006.
[3] A. Bondarenko, P.M. Dung, R.A. Kowalski, and F. Toni. An abstract, argumentation-theoretic framework for default reasoning. *Artificial Intelligence*, 93(1-2):63–101, 1997.
[4] R. A. Kowalski and F. Toni. Abstract argumentation. *Journal of Artificial Intelligence and Law, Special Issue on Logical Models of Argumentation*, 4(3-4):275–296, 1996.
[5] P.M. Dung, P. Mancarella, and F. Toni. A dialectic proof procedure for sceptical assumption-based argumentation. Technical report, AIT, 2006.
[6] Antonis C. Kakas and Francesca Toni. Computing argumentation in logic programming. *Journal of Logic and Computation*, 9(4):515–562, 1999.
[7] J. Pollock. Defeasible reasoning. *Cognitive Science*, 11(4):481–518, 1987.
[8] Henry Prakken and Giovanni Sartor. The role of logic in computational models of legal argument: A critical survey. In *Computational Logic: Logic Programming and Beyond*, volume 2408 of *Lecture Notes in Computer Science*, pages 342–381. Springer, 2002.
[9] P.M. Dung. The acceptability of arguments and its fundamental role in non-monotonic reasoning and logic programming and n-person game. *Artificial Intelligence*, 77:321–357, 1995.
[10] Phan Minh Dung, Paolo Mancarella, and Francesca Toni. Argumentation-based proof procedures for credulous and sceptical non-monotonic reasoning. In *Computational Logic: Logic Programming and Beyond*, volume 2408 of *Lecture Notes in Computer Science*, pages 289–310. Springer, 2002.
[11] Chitta Baral and Michael Gelfond. Logic programming and knowledge representation. *Journal of Logic Programming*, 19/20:73–148, 1994.
[12] G. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proc. JELIA*, pages 224–238. Springer Verlag LNCS 1919, 2000.
[13] P.E. Dunne and T.J.M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 149:221–250, 2003.
[14] C. Cayrol, S. Doutre, and J. Mengin. On decision problems related to preferred semantics a of argumentation frameworks. *Journal of Logic and Computation*, 13(3):377–403, 1003.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Evaluation and comparison criteria for extension-based argumentation semantics

Pietro BARONI [a,*] and Massimiliano GIACOMIN [a]
[a] *Dip. Elettronica per l'Automazione, Univ. of Brescia, Italy*

Abstract. In the context of Dung's theory of abstract argumentation frameworks, the comparison between different semantics is often carried out by resorting to some specific examples considered particularly meaningful. This kind of comparison needs to be complemented by more general evaluation criteria based on "example-independent" basic principles. We review several principles for argumentation semantics, identify their formal counterpart in terms of extensions, and analyze their relationships with the notion of argument justification state. Then, we evaluate and compare several semantics on the basis of the introduced principles.

Keywords. Argumentation semantics, Extensions, Argumentation frameworks

1. Introduction

According to [1], an argumentation system can be characterized along five main dimensions: the underlying logical language, the definition of what an argument is, the criteria for identifying conflict between arguments, the definition of the relevant relation of defeat between them, and, finally, the definition of the justification state of arguments. Dung's theory of abstract argumentation frameworks [2] is generic with respect to the first four dimensions and focuses on the fundamental semantics issues concerning the identification of the conflict outcome, which is represented as a set of sets of arguments, called *extensions*. An extension can be intuitively described as a set of arguments which are able to survive the conflict together. An *argumentation semantics* defines in a declarative way the properties required for a set to be an extension. In general, several sets of arguments may enjoy these properties: if a semantics prescribes the set of extensions to always be a singleton it is said to belong to the *unique-status approach*, otherwise to the *multiple-status approach*. The theory of argumentation frameworks is able to encompass a variety of semantics proposals, such as *grounded*, *stable*, and *preferred* semantics [2]. Subsequently, several alternative proposals have been formulated in the context of this theory, such as the four SCC-recursive semantics introduced in [3,4] (the most significant being $CF2$ semantics [4,5]) and the recently introduced *prudent* semantics [6,7].

The comparison of different semantics is often based on their behavior on some examples (or families of examples) of argumentation frameworks, regarded as sufficiently important and general. This is the case, for instance, of the so-called "zombie argu-

*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.

ments" [8] which reveal a fundamental difference between unique- and multiple-status approaches, or of "odd-length cycles", where stable semantics fails to prescribe extensions and whose treatment by preferred semantics has often been considered problematic [1,9,5,4]. While carefully selected examples can provide useful insights into the analysis of alternative proposals, they suffer from the inherent limitation of relying more on intuition than on formally stated principles. In fact, even in relatively simple examples there may not be a general agreement on the "desired" outcome, due to different underlying intuitions (see for instance [10]). For this reason, it has been observed that using intuition about specific examples to derive general considerations may be inappropriate and "it is better to use intuitions not as critical tests but as generators for further investigation"[11]. Disadvantages of example-based analysis are also discussed in [12,13].

In order to overcome the limits of example-driven analysis, a more general approach is needed. In this paper we provide a survey and an investigation about the definition of some general criteria for evaluating and comparing argumentation semantics and discuss their application to some literature proposals. The paper is organised as follows. After recalling the necessary background concepts in section 2, we identify two fundamental principles for argumentation semantics in section 3 and propose an articulated classification of justification states in section 4. Criteria related to the notions of *admissibility* and *reinstatement*, *directionality*, and *skepticism* are introduced and discussed in sections 5, 6, and 7, respectively. Some derived properties at the level of justification states are then examined in section 8, while section 9 compares some literature semantics in the light of the introduced criteria. Section 10 concludes the paper.

2. Background concepts and notation

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

We assume that $\mathcal{A}$ represents the set of arguments produced by a reasoner at a given instant of time, therefore in the following we will assume that $\mathcal{A}$ is finite, independently of the fact that the underlying mechanism of argument generation admits the existence of infinite set of arguments. The treatment of argumentation frameworks where $\mathcal{A}$ is infinite is left as future work. An argumentation framework $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ can be represented as a directed graph, called *defeat graph*, where nodes are the arguments and edges correspond to the elements of the attack relation. The nodes that attack a given argument α are called *defeaters* of α and form a set which is denoted as $\mathrm{par}_{\mathrm{AF}}(\alpha)$:

Definition 2 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ *and a node* $\alpha \in \mathcal{A}$*,* $\mathrm{par}_{\mathrm{AF}}(\alpha) \triangleq \{\beta \in \mathcal{A} \mid \beta \rightarrow \alpha\}$*. If* $\mathrm{par}_{\mathrm{AF}}(\alpha) = \emptyset$*, then* α *is called an* initial *node.*

Since we will frequently consider properties of sets of arguments, it is useful to extend to them the notations defined for the nodes:

Definition 3 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$*, a node* $\alpha \in \mathcal{A}$ *and two sets* $S, P \subseteq \mathcal{A}$*, we define:* $S \rightarrow \alpha \equiv \exists \beta \in S : \beta \rightarrow \alpha$; $\alpha \rightarrow S \equiv \exists \beta \in S : \alpha \rightarrow \beta$; $S \rightarrow P \equiv \exists \alpha \in S, \beta \in P : \alpha \rightarrow \beta$.

Given a generic argumentation semantics $\mathcal{S}$, the set of extensions prescribed by $\mathcal{S}$ for a given argumentation framework $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ is denoted as $\mathcal{E}_{\mathcal{S}}(\mathrm{AF})$.

We define also the *restriction* of an argumentation framework to a subset of its nodes:

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

3. Fundamental principles

As to our knowledge, all extension-based argumentation semantics proposed in the literature adhere to two fundamental principles, which in the following will be given for granted: the *conflict-free* principle, denoted as $\mathcal{CF}$, and the *I-maximality* principle, denoted as $\mathcal{IM}$. Intuitively the $\mathcal{CF}$-principle relies on the idea that, since an extension is a set of arguments which can "survive together", no conflicting arguments can be included in the same extension. The $\mathcal{CF}$-principle has a straightforward formal counterpart.

Definition 5 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$*, a set* $S \subseteq \mathcal{A}$ *is* conflict-free*, denoted as* $cf(S)$*, iff* $\nexists \alpha, \beta \in S$ *such that* $\alpha \rightarrow \beta$*. A semantics* $\mathcal{S}$ *satisfies the* $\mathcal{CF}$-principle *if and only if* $\forall \mathrm{AF}, \forall E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$ E *is conflict-free.*

The $\mathcal{IM}$-principle states that an extension can not be a proper subset of another extension and has a straightforward formal counterpart.

Definition 6 *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* $\mathcal{IM}$ principle *if and only if* $\forall \mathrm{AF}, \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$ *is I-maximal.*

Note that I-maximality is a property of the set of extensions $\mathcal{E}$ *per se*, without reference to any generic criterion. For instance grounded semantics prescribes a set consisting of a (unique) grounded extension, which therefore satisfies I-maximality according to Definition 6 though the grounded extension is not a maximal complete extension.

4. Argument justification states

In extension-based semantics, the justification state of an argument is derived from its membership to extensions. A consolidated tradition considers three justification states.

Definition 7 *[1] Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ *and a semantics* $\mathcal{S}$*, an argument* α *is:* justified *iff* $\forall E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$*,* $\alpha \in E$*;* overruled *iff* $\forall E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$*,* $\alpha \notin E$*;* defensible *otherwise i.e.* $\exists E_1, E_2 \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF}) : \alpha \in E_1, \alpha \notin E_2$.

Though largely adopted, this definition has some drawbacks. In particular it collapses to only two possible states for unique-status semantics. Since also in this case at least three justification states are needed, an alternative definition is applied [14].

Definition 8 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ *and a unique-status semantics* $\mathcal{S}$*, with* $\mathcal{E}_{\mathcal{S}}(\mathrm{AF}) = \{E\}$ *an argument* α *is:* undefeated *iff* $\alpha \in E$*;* defeated outright *iff* $E \rightarrow \alpha$*;* provisionally defeated *otherwise, i.e.* $\alpha \notin E$ *and* $E \not\rightarrow \alpha$.

The presence of two alternative definitions is quite unsatisfactory. The quest for a more appropriate definition is also witnessed by the distinction between *provisionally defeated* and *seemingly defeated* arguments mentioned in [1] and by a classification encompassing four states introduced in [15]. As shown below, a more systematic approach leads to consider seven states [16]. First, considering the relationship between an argument α and a specific extension E, three main situations[1] can be envisaged:

- α is *in* E, denoted as $in(\alpha, E)$, if $\alpha \in E$;
- α is *definitely out from* E, denoted as $do(\alpha, E)$, if $\alpha \notin E \wedge E \rightarrow \alpha$;
- α is *provisionally out from* E, denoted as $po(\alpha, E)$, if $\alpha \notin E \wedge E \not\rightarrow \alpha$.

Taking into account the existence of multiple extensions, one can consider that an argument can be in any of the above three states with respect to all, some or none of the extensions. This gives rise to 27 hypothetical combinations. It is however easy to see that some of them are impossible, for instance if an argument is in a given state with respect to all extensions this clearly excludes that it is in another state with respect to any extension. Directly applying this kind of considerations, seven possible *Justification States* emerge for an argument α with respect to a non-empty set of extensions $\mathcal{E}$:

- $\forall E \in \mathcal{E}$, $in(\alpha, E)$, denoted as JS_I;
- $\forall E \in \mathcal{E}$, $do(\alpha, E)$, denoted as JS_D;
- $\forall E \in \mathcal{E}$, $po(\alpha, E)$, denoted as JS_P;
- $\exists E \in \mathcal{E}$ such that $do(\alpha, E)$, $\exists E \in \mathcal{E}$ such that $po(\alpha, E)$, and $\nexists E \in \mathcal{E}$ such that $in(\alpha, E)$, denoted as JS_{DP};
- $\exists E \in \mathcal{E}$ such that $in(\alpha, E)$, $\exists E \in \mathcal{E}$ such that $po(\alpha, E)$, and $\nexists E \in \mathcal{E}$ such that $do(\alpha, E)$, denoted as JS_{IP};
- $\exists E \in \mathcal{E}$ such that $in(\alpha, E)$, $\exists E \in \mathcal{E}$ such that $do(\alpha, E)$, and $\nexists E \in \mathcal{E}$ such that $po(\alpha, E)$, denoted as JS_{ID};
- $\exists E \in \mathcal{E}$ such that $in(\alpha, E)$, $\exists E \in \mathcal{E}$ such that $do(\alpha, E)$, and $\exists E \in \mathcal{E}$ such that $po(\alpha, E)$, denoted as JS_{IDP}.

It is easy to see that in the case of a unique-status semantics, only JS_I, JS_D and JS_P may hold, which correspond to the state of undefeated, defeated outright and provisionally defeated, respectively. A relevant question concerns the actual existence of each of the seven states: a positive answer is given in [16], where an example for each state is provided in the context of preferred semantics. On the other hand, when $\mathcal{E}_{\mathcal{S}}(\mathrm{AF}) = \emptyset$ we assume that the justification state of the arguments of AF according to $\mathcal{S}$ is not defined.

5. Admissibility and reinstatement

The requirement of admissibility lies at the heart of all semantics discussed in [2]. A conflict-free set is admissible if it defends all of its elements, which corresponds to require the following *admissibility condition* for any extension E of AF:

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

Condition (1) includes the case where α defends itself against (some of) its defeaters. We suggest that a stronger notion of defense may also be considered where a node α

[1] The case $\alpha \in E \wedge E \rightarrow \alpha$ is prevented by the $\mathcal{CF}$-principle

cannot defend itself nor can be involved in its own defense. To formalize this requirement we introduce the notion of *strongly defended argument*.

Definition 9 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$, $\alpha \in \mathcal{A}$ *and* $S \subseteq \mathcal{A}$ *we say that* α *is strongly defended by* S *(denoted as* $sd(\alpha, S)$*) iff* $\forall \beta \in \mathrm{par}_{\mathrm{AF}}(\alpha), \exists \gamma \in S \setminus \{\alpha\}$*:* $\gamma \rightarrow \beta$ *and* $sd(\gamma, S \setminus \{\alpha\})$.

In words, α is strongly defended by S, if S includes a defeater $\gamma \neq \alpha$ for any defeater β of α. In turn, γ has to be strongly defended by $S \setminus \alpha$, namely γ needs neither α 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 $sd(\alpha, S)$ the base of this recursive definition is provided by initial nodes, which are strongly defended by any set, since they have no defeaters. The notion of strong defense is the basis of the definition of the *strong admissibility condition* given, for any extension E, by

$$\alpha \in E \Rightarrow sd(\alpha, E). \tag{2}$$

The property of *reinstatement* corresponds to the converse of the implication (1) prescribed by the admissibility condition. Intuitively, an argument α is *reinstated* if its defeaters are in turn defeated and, as a consequence, they have no effect on the justification state of α. Reinstatement can be referred to the notion of extension, by imposing that any extension E of any argumentation framework AF satisfies the following *reinstatement condition*:

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

Considering the strong notion of defense we obtain a *weak* (since it is implied by condition (3)) *reinstatement condition* expressed, for any extension E, as:

$$sd(\alpha, E) \Rightarrow \alpha \in E \tag{4}$$

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

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

The underlying notions of admissibility and reinstatement are a characterizing feature of an argumentation semantics. Actually, all semantics discussed in [2] are based on the notion of complete extension, namely a conflict-free set satisfying conditions (3) and (1). In particular, preferred extensions are complete extensions which are also maximal wrt. set inclusion. As to our knowledge, an analogous characterization of grounded semantics in terms of the adopted notions of admissibility and reinstatement has not yet been provided. We show here that it can be achieved in terms of conditions (4) and (2).

On one hand, Proposition 1 shows that, for a given semantics $\mathcal{S}$, satisfying condition (4) is equivalent to the property of *agreement* with grounded semantics, namely for any argumentation framework AF and $\forall E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$, $\mathrm{GE}(\mathrm{AF}) \subseteq E$, where $\mathrm{GE}(\mathrm{AF})$ denotes the (unique) grounded extension of AF.

Proposition 1 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ *and a set* $E \subseteq \mathcal{A}$*,* E *satisfies condition (4) if and only if* $\mathrm{GE(AF)} \subseteq E$*.*

Proof. (a) $(4) \Rightarrow \mathrm{GE(AF)} \subseteq E$.
It is known [2] that, for any finite AF, $\mathrm{GE(AF)} = \bigcup_{i \geq 1} \mathrm{F}^i_{\mathrm{AF}}(\emptyset)$, where, given a set $S \subseteq \mathcal{A}$, $\mathrm{F}_{\mathrm{AF}}(S) = \{\alpha \in \mathcal{A} : \forall \beta \in \mathrm{par}_{\mathrm{AF}}(\alpha), S \rightarrow \beta\}$, $\mathrm{F}^1_{\mathrm{AF}}(S) = \mathrm{F}_{\mathrm{AF}}(S)$, and $\mathrm{F}^i_{\mathrm{AF}}(S) = \mathrm{F}_{\mathrm{AF}}(\mathrm{F}^{i-1}_{\mathrm{AF}}(S))$. Now, any argument α belonging to $\mathrm{F}_{\mathrm{AF}}(\emptyset) = \mathrm{F}^1_{\mathrm{AF}}(\emptyset)$ (actually, the initial arguments of the defeat graph) is strongly defended by any set and necessarily belongs to E according to condition (4). Now, assume inductively that $\mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset) \subseteq E$ and $\forall \alpha \in \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset), sd(\alpha, \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset))$ (and therefore $\forall \alpha \in \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset), sd(\alpha, E)$). Then $\forall \beta \in \mathrm{F}^i_{\mathrm{AF}}(\emptyset) \setminus \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset)$, β is defended by $\mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset)$ (and therefore by E), and since $\forall \alpha \in \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset), sd(\alpha, \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset))$ then $sd(\beta, \mathrm{F}^{i-1}_{\mathrm{AF}}(\emptyset))$. Hence we have that $sd(\beta, \mathrm{F}^i_{\mathrm{AF}}(\emptyset))$ and $sd(\beta, E)$, which, by (4) implies $\beta \in E$. Then the conclusion follows.
(b) $\mathrm{GE(AF)} \subseteq E \Rightarrow (4)$.
Suppose by contradiction that $\exists \alpha \notin E : sd(\alpha, E)$. Since $\alpha \notin \mathrm{GE(AF)}$, in particular $\mathrm{par}_{\mathrm{AF}}(\alpha) \neq \emptyset$. By assumption, $\forall \beta \in \mathrm{par}_{\mathrm{AF}}(\alpha), \exists \gamma \in E \backslash \{\alpha\} : \gamma \rightarrow \beta \wedge sd(\gamma, E \backslash \{\alpha\})$. Now, if for all such γ, $\gamma \in \bigcup_{i \geq 1} \mathrm{F}^i_{\mathrm{AF}}(\emptyset)$ then also $\alpha \in \bigcup_{i \geq 1} \mathrm{F}^i_{\mathrm{AF}}(\emptyset) = \mathrm{GE(AF)}$. Therefore it must be the case that $\exists \gamma \in E \setminus \{\alpha\} : sd(\gamma, E \setminus \{\alpha\}) \wedge \gamma \notin \mathrm{GE(AF)}$. Now iterating the same kind of reasoning on γ we are led to consider an argument $\gamma' \in E \setminus \{\alpha, \gamma\} : sd(\gamma', E \setminus \{\alpha, \gamma\}) \wedge \gamma' \notin \mathrm{GE(AF)}$. This in turn leads to consider an argument $\gamma'' \in E \setminus \{\alpha, \gamma, \gamma'\} : sd(\gamma'', E \setminus \{\alpha, \gamma, \gamma'\}) \wedge \gamma'' \notin \mathrm{GE(AF)}$, and so on. In summary we are lead to consider an infinite sequence of distinct arguments within E, which is impossible due to the finiteness of $\mathcal{A}$. □

On the other hand, condition (2) implies that an extension E is included in the grounded extension.

Proposition 2 *Given an argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$ *and a set* $E \subseteq \mathcal{A}$*, if* E *satisfies condition (2), then* $E \subseteq \mathrm{GE(AF)}$*.*

Proof. We prove the statement by contradiction, assuming that $\exists \alpha \in \mathcal{A} : \alpha \in E, \alpha \notin \mathrm{GE(AF)}$. Note first that, as a consequence, $\exists \beta \in \mathcal{A} : \beta \rightarrow \alpha$, since initial nodes belong to $\mathrm{GE(AF)}$. Then, according to (2), for any such β there must be an argument $\gamma \in E \setminus \{\alpha\} : \gamma \rightarrow \beta \wedge sd(\gamma, E \setminus \{\alpha\})$. As in the proof of Proposition 1(b), it must be the case that for at least one of these γ, $\gamma \notin \mathrm{GE(AF)}$, and we are led to consider an infinite sequence of distinct arguments within E, which is impossible. □

Propositions 1 and 2 show that the conjunction of weak reinstatement and strong admissibility provides a characterization of grounded semantics.

Some paradigmatic examples often used to compare different semantics behaviors can be put in relation to reinstatement and admissibility criteria. For instance, the inability of grounded semantics to support "floating defeat" is related to its adherence to the strong admissibility condition. "Floating defeat" is typically exemplified by the argumentation framework $\mathrm{AF}_{FD} = \langle \{\alpha, \beta, \gamma, \delta\}, \{(\alpha, \beta), (\beta, \alpha), (\alpha, \gamma), (\beta, \gamma), (\gamma, \delta)\} \rangle$. Floating defeat is achieved in this case by a semantics $\mathcal{S}$ if $\mathcal{E}_{\mathrm{AF}_{FD}}(\mathcal{S}) = \{\{\alpha, \delta\}, \{\beta, \delta\}\}$ so that δ is *in* and γ *definitely out* from all extensions. However, the extensions $\{\alpha, \delta\}$ and $\{\beta, \delta\}$ do not satisfy the strong admissibility condition.

Another often debated question concerns the treatment of odd- vs. even-length attack cycles. For instance, according to preferred semantics an odd-length cycle admits only the empty set as extension while an even-length cycle admits multi-

ple nonempty extensions. Several authors [9,4] have observed that this behavior violates an intuitive *symmetry* requirement, which is however hard to formalize in general. It can be anyway observed that considering a three-length cycle $\mathrm{AF}_{3C} = \langle\{\alpha,\beta,\gamma\},\{(\alpha,\beta),(\beta,\gamma),(\gamma,\alpha)\}\rangle$, symmetry with respect to an even-length cycle can only be achieved if $\mathcal{E}_{\mathrm{AF}_{3C}}(\mathcal{S}) = \{\{\alpha\},\{\beta\},\{\gamma\}\}$, but all the relevant extensions violate the admissibility and reinstatement properties. It can be noted however that they at least satisfy the weak (4) and $\mathcal{CF}$-reinstatement (5) conditions. Therefore symmetry between odd and even-length cycles is not totally in contrast with the notion of reinstatement. In fact, $CF2$ semantics satisfies both these weaker notions of reinstatement. As to $\mathcal{CF}$-reinstatement, since any extension prescribed by $CF2$ semantics for an argumentation framework AF is a maximal conflict free set of AF (Lemma 2 of [17]), it is easy to see that $CF2$ semantics satisfies condition (5) since any extension satisfies the weaker condition $cf(E \cup \{\alpha\}) \Rightarrow \alpha \in E$. As to weak reinstatement, it is known [4] that $\forall E \in \mathcal{E}_{CF2}(\mathrm{AF})\ \mathrm{GE}(\mathrm{AF}) \subseteq E$: then (4) follows from Proposition 1(b).

6. Directionality

The directionality principle intuitively prescribes that the state of an argument is determined only by its defeaters (and then in turn by the defeaters of its defeaters, and so on) while it is not affected by the arguments it defeats. In [4] the directionality principle is introduced with reference to the notion of strongly connected components of the defeat graph. Here we provide a more general formulation based on the notions of ancestors of an argument.

Definition 10 *Given an argumentation framework* $\mathrm{AF} = \langle\mathcal{A},\rightarrow\rangle$ *and an argument* $\alpha \in \mathcal{A}$ *the set* $\mathrm{anc}_{\mathrm{AF}}(\alpha)$ *of ancestors of* α *is defined as follows:* $\mathrm{anc}_{\mathrm{AF}}(\alpha) = \{\alpha\} \cup \bigcup_{\beta \in \mathrm{par}_{\mathrm{AF}}(\alpha)} \mathrm{anc}_{\mathrm{AF}}(\beta)$

Definition 11 *A semantics* $\mathcal{S}$ *satisfies the directionality principle if and only if* $\forall \mathrm{AF}_1 = \langle\mathcal{A}_1,\rightarrow_1\rangle, \mathrm{AF}_2 = \langle\mathcal{A}_2,\rightarrow_2\rangle, \forall \alpha \in \mathcal{A}_1, \beta \in \mathcal{A}_2 : \mathrm{AF}_1{\downarrow}_{\mathrm{anc}_{\mathrm{AF}_1}(\alpha)} = \mathrm{AF}_2{\downarrow}_{\mathrm{anc}_{\mathrm{AF}_2}(\beta)}$, $\mathcal{AE}_{\mathcal{S}}(\mathrm{AF}_1,\alpha) = \mathcal{AE}_{\mathcal{S}}(\mathrm{AF}_2,\beta)$

where given an argumentation framework $\mathrm{AF} = \langle\mathcal{A},\rightarrow\rangle$, an argument $\alpha \in \mathcal{A}$ and a semantics $\mathcal{S}$, $\mathcal{AE}_{\mathcal{S}}(\mathrm{AF},\alpha) = \{(E \cap \mathrm{anc}_{\mathrm{AF}}(\alpha)) \mid E \in \mathcal{E}_{\mathcal{S}}(\mathrm{AF})\} \subseteq 2^{\mathcal{A}}$.

In words, given any argumentation frameworks AF_1 and AF_2 which are equal as far as the ancestors of two "corresponding arguments" α and β are concerned, the extensions prescribed by $\mathcal{S}$ for AF_1 and AF_2 have the same intersections with the ancestors of α and β respectively. This corresponds to the idea that the construction of extensions including an argument α is not affected by the nodes which strictly follow α in the defeat graph. It can be easily seen (due to space limitation we refer the reader to [4] for details) that the property of SCC-recursiveness (Definition 20 of [4]) is a special case of the directionality principle, provided that the existence of extensions for any argumentation framework is ensured. Therefore any SCC-recursive semantics satisfies it under this condition. On the other hand, stable semantics is SCC-recursive but does not satisfy the directionality principle due to non-existence of extensions in some cases.

7. Skepticism

The notion of skepticism has often been used in informal or semi-formal ways to discuss semantics behavior, for instance in the contraposition of credulous vs. skeptical attitude or when stating that a semantics is "more skeptical" than another one. In the literature, the notion of skepticism has been referred either to argument justification states or to extensions, it has however to be remarked that a systematic treatment of this notion still seems to be lacking. Let us start, at an elementary 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 arguments surviving the conflict. Since a semantics typically prescribes a set of extensions, a relation of skepticism between sets of extensions needs then to be defined. In [3] a weak and a strong skepticism relationship between sets of extensions are introduced: as discussed in [3,17] only the following *weak skepticism relation* $\preceq_W$ appears to have useful properties.

Definition 12 *Given two sets of extensions $\mathcal{E}_1$ and $\mathcal{E}_2$, $\mathcal{E}_1$ is more skeptical than $\mathcal{E}_2$, denoted as $\mathcal{E}_1 \preceq_W \mathcal{E}_2$, iff* $\forall E_2 \in \mathcal{E}_2 \; \exists E_1 \in \mathcal{E}_1 : E_1 \subseteq E_2$.

The weak skepticism relation $\preceq_W$ between sets of extensions can be applied in a twofold way. On one hand, it can be used to compare the sets of extensions prescribed by a given semantics on two different but related argumentation frameworks. This is the basis of the definition of the property of skepticism adequacy [17], which refers to a generic skepticism relation $\preceq$ between sets of extensions.

Definition 13 *Given a skepticism relation $\preceq$, a semantics $\mathcal{S}$ is $\preceq$-adequate iff for any argumentation framework* $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$*, for any* $\alpha, \beta \in \mathcal{A} : \alpha \neq \beta \wedge \alpha \rightarrow \beta$*,* $\mathcal{E}_{\mathcal{S}}(\mathrm{AF}^{\rightleftarrows(\beta,\alpha)}) \preceq \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$*, where* $\mathrm{AF}^{\rightleftarrows(\beta,\alpha)} = \langle \mathcal{A}, \rightarrow \cup \{(\beta, \alpha)\} \rangle$.

The underlying idea (see [17] for more details) is the following. Consider a generic argumentation framework AF, including a couple of nodes α and β such that $\alpha \rightarrow \beta$ while $\beta \not\rightarrow \alpha$, and an argumentation framework AF' obtained from AF by simply adding an attack relation from β to α while leaving all the rest unchanged (this is denoted as $\mathrm{AF}^{\rightleftarrows(\beta,\alpha)}$). Focusing on the skepticism relation $\preceq_W$, it is reasonable to expect that, for any semantics $\mathcal{S}$, $\mathcal{E}_{\mathcal{S}}(\mathrm{AF}') \preceq_W \mathcal{E}_{\mathcal{S}}(\mathrm{AF})$. In fact, converting a unidirectional attack into a mutual one can only make the states of the involved nodes less committed (of course they can remain the same if they are strictly determined by other arguments, independently of the attack relations between α and β). In turn, having α or β in a less committed state may only give rise to other less committed states in the nodes they attack.

On the other hand, a generic relation $\preceq$ between sets of extensions can be used to define a skepticism relation between different semantics, based on the comparison of the sets of extensions prescribed by the semantics for the same argumentation frameworks.

Definition 14 *Any skepticism relation $\preceq$ between sets of extensions induces a corresponding skepticism relation $\leq$ between semantics: $\mathcal{S}_1 \leq \mathcal{S}_2$ iff for any argumentation framework* AF*,* $\mathcal{E}_{\mathcal{S}_1}(\mathrm{AF}) \preceq \mathcal{E}_{\mathcal{S}_2}(\mathrm{AF})$.

The relation between semantics induced by $\preceq_W$ will be denoted as $\leq_W$.

8. Properties of justification states

Several properties concerning justification states can be derived from the principles introduced in sections 5, 6, and 7. First, a partial order $\trianglelefteq$ on justification states reflecting their level of commitment is induced by the skepticism relation between semantics.

Definition 15 *A justification state JS_1 is less committed than a justification state JS_2, denoted as $JS_1 \trianglelefteq JS_2$, iff there are an argumentation framework $\mathrm{AF} = \langle \mathcal{A}, \rightarrow \rangle$, an argument $\alpha \in \mathcal{A}$ and two semantics $\mathcal{S}_1, \mathcal{S}_2$ with $\mathcal{S}_1 \leq \mathcal{S}_2$, such that JS_1 and JS_2 are the justification states assigned to α by $\mathcal{S}_1$ and $\mathcal{S}_2$, respectively.*

The partial order induced by $\leq_W$ on justification states will be denoted as $\trianglelefteq_W$. It is shown in [16] that the diagram corresponding to $\trianglelefteq_W$ is the one presented in Figure 1(a). For a node α, being *in* or, similarly, *definitely out* in all extensions gives rise to the most committed states, a less committed state arises when both extensions where α is *in* and *definitely out* are present, while the least committed states are all those where there is at least an extension where α is *provisionally out*. It has to be remarked that the notion of level of commitment is clearly distinct from the one of level of credibility. In fact, both the most credible and the least credible arguments are at the highest level of commitment. A partial order of justification states according to credibility can be defined according to the existence of extensions where it is *in* (most credible), *provisionally out*, and *definitely out* (least credible), as shown in Figure 1(b).

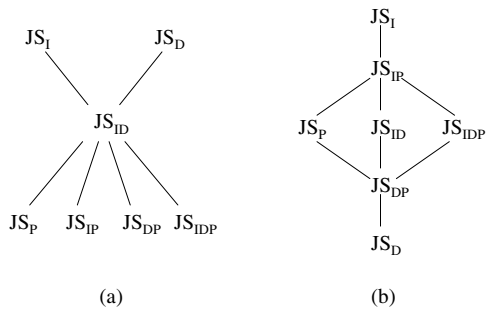


Figure 1. Diagrams of partial orders of justification states wrt. level of commitment (a) and credibility (b).

Another relevant aspect concerns the constraints on the possible justification state of an argument which are induced by the justification state of its defeaters and, possibly, by the reinstatement criterion adopted by a semantics. Each line of Table 1 shows the possible justification states of an argument α under the hypotheses that there is a defeater with a given justification state and the condition specified in the second column holds. Similarly, each line of Table 2 shows the possible justification states of an argument α under the hypotheses that all of its defeaters are assigned a given justification state and the condition specified in the second column holds. In both tables, the symbol $*$ in the second column denotes that no additional criterion is invoked, while the symbol JS_* in the third column denotes the disjunction of all seven justifications states. Due to space limitation, neither proofs (which are rather easy) nor comments to the Tables can be given in this paper.

Table 1. Constraints on justification state of α induced by the state of a defeater

State of a defeater	Additional condition	Possible states of α
JS_I	$\mathcal{CF}$	JS_D
JS_D	*	JS_*
JS_P	(1)	$JS_D \vee JS_P \vee JS_{DP}$
JS_{DP}	(1)	$JS_D \vee JS_P \vee JS_{DP} \vee JS_{IP} \vee JS_{ID} \vee JS_{IDP}$
JS_{IP}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$
JS_{ID}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$
JS_{IDP}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$

Table 2. Constraints on justification state of α induced by the state of all defeaters

State of all defeaters	Additional condition	Possible states of α
JS_I	$\mathcal{CF}$	JS_D
JS_D	*	$JS_I \vee JS_P \vee JS_{IP}$
JS_D	(3)	JS_I
JS_P	*	$JS_I \vee JS_P \vee JS_{IP}$
JS_P	(1)	JS_P
JS_{DP}	*	$JS_I \vee JS_P \vee JS_{IP}$
JS_{DP}	(1)	$JS_P \vee JS_{IP}$
JS_{IP}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$
JS_{IP}	$\mathcal{CF} \wedge$ (1)	$JS_D \vee JS_{DP}$
JS_{ID}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$
JS_{IDP}	$\mathcal{CF}$	$JS_D \vee JS_{DP} \vee JS_{ID} \vee JS_{IDP}$

9. Comparing argumentation semantics

We are now ready to apply the notions introduced in previous sections to some literature semantics, namely grounded (denoted as $\mathcal{GR}$), preferred (denoted as $\mathcal{PR}$), $CF2$ and prudent (denoted as $\mathcal{PU}$) semantics. While grounded and preferred semantics are well-known as the "best" representatives of the unique- and multiple-status approach respectively, the recently introduced $CF2$ and prudent semantics are considered here since they feature peculiar properties. $CF2$ semantics [4] is a SCC-recursive semantics which explicitly departs from the notion of admissibility in order to achieve a "symmetric" treatment of odd- and even-length cycles (further details can not be given due to space limitation, see [4]). Prudent semantics instead remains in the frame of admissibility-based semantics emphasizing the role of indirect attack: a node α indirectly attacks a node β if there is an odd-length path from α to β. While several classes of *prudent extensions* (*p-extensions*) are considered in [6,7], here we refer here to *preferred p-extensions* which are maximal admissible sets such that there are no indirect attacks between their elements.

Consider first admissibility. As already discussed in section 5, the grounded semantics is the only one to satisfy the strong admissibility condition (2) (which implies also the admissibility condition (1)). Preferred and prudent semantics satisfy condition (1) by definition, while, when dealing with odd-length cycles, $CF2$ semantics does not.

Turning to reinstatement, it is well-known that condition (3) and hence the implied conditions (4) and (5) are satisfied by grounded and preferred semantics. $CF2$ semantics does not satisfy the reinstatement condition (3), but satisfies weak and $\mathcal{CF}$-reinstatement conditions (5) and (4) as discussed at the end of section 5. As to prudent semantics, since

Table 3. Property satisfaction by argumentation semantics

	Grounded	Preferred	CF2	Prudent	??
Admissibility (1)	Yes	Yes	No	Yes	Yes
Strong admissibility (2)	Yes	No	No	No	No
Reinstatement (3)	Yes	Yes	No	No	Yes
Weak reinstatement (4)	Yes	Yes	Yes	No	Yes
$\mathcal{CF}$-reinstatement (5)	Yes	Yes	Yes	No	Yes
Directionality (Def. 11)	Yes	Yes	Yes	No	Yes
Skepticism adequacy (Def. 13)	Yes	No	Yes	No	Yes

there are preferred p-extensions where some initial node is not included (see for instance Example 2 in [7]), it is easy to see that reinstatement is not satisfied in any form.

Turning to the directionality principle, it is satisfied by grounded, preferred, and $CF2$ semantics, since, in particular, they are SCC-recursive [4]. On the other hand, Example 2 of [7] shows that there are preferred p-extensions not satisfying this principle.

As to skepticism adequacy, it is shown in [17] that grounded and $CF2$ semantics satisfy it, while preferred semantics does not. In fact, considering the argumentation framework $\mathrm{AF}_{SA} = \langle\{\alpha,\beta,\gamma,\delta\},\{(\alpha,\beta),(\beta,\gamma),(\gamma,\alpha),(\gamma,\delta)\}\rangle$, it turns out that $\mathcal{E}_{\mathcal{PR}}(\mathrm{AF}_{SA}) = \{\emptyset\}$ and $\mathcal{E}_{\mathcal{PR}}(\mathrm{AF}_{SA}{}^{\rightleftarrows(\gamma,\delta)}) = \{\{\alpha,\delta\}\}$. Using the same AF_{SA} it can be seen that prudent semantics too fails to satisfy skepticism adequacy, since $\mathcal{E}_{\mathcal{PU}}(\mathrm{AF}_{SA}) = \{\emptyset\}$ and $\mathcal{E}_{\mathcal{PU}}(\mathrm{AF}_{SA}{}^{\rightleftarrows(\gamma,\delta)}) = \{\{\delta\}\}$.

Table 3 provides a synoptic view about the satisfaction of the above mentioned principles by the considered semantics. It emerges that only grounded semantics respects all properties, but it does it in a "too strong" way, since strong admissibility may be regarded as not necessary and even undesirable. We are not aware of any semantics respecting the properties specified in the last column: such a semantics would satisfy the whole set of reasonable general properties we have identified, without imposing too strong requirements. Whether such a semantics can be found or its existence is prevented by some inherent impossibility constraint appears to be an interesting question to be investigated.

As to the skepticism relation between semantics, it has been shown in [16] that $\mathcal{GR} \leq_W \mathcal{PR}$ and $\mathcal{GR} \leq_W CF2$ (actually, this is equivalent to the property of agreement with grounded semantics, often regarded as a reasonable requirement in the literature). The examples AF_{3C} and AF_{SA} previously discussed show that $\mathcal{PR}$ and $CF2$ are not comparable as far as skepticism is concerned. As to prudent semantics, analyzing Example 2 of [7] it can be seen that it is not comparable with grounded semantics. The same examples AF_{3C} and AF_{SA} show that prudent semantics is not comparable with $CF2$ semantics. Finally, Example 4 of [7] shows that prudent semantics is also not comparable with preferred semantics (details on examples can not be given due to space limitation).

10. Conclusions

Principle-based evaluation and comparison of extension-based argumentation semantics is an important issue which has received relatively limited attention in the literature. In this paper, we have provided a systematic analysis of several general principles for argumentation semantics and discussed their relations with a novel definition of argument justification states. Due to space limitation, only a quick example of their application to

a limited number of literature proposals has been given. It emerges that, in the multiple-status approach, none of the compared semantics satisfies all the considered properties, while in the unique-status approach, grounded semantics does satisfy all of them but has also a "too strong" and possibly undesirable feature. Future work directions include a more extensive and detailed analysis of general requirements for extension-based argumentation and the study of "principle-driven" definitions of new semantics. A more detailed and comprehensive analysis covering a larger set of semantics is also envisaged.

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

References

[1] H. Prakken, G. A. W. Vreeswijk. Logics for defeasible argumentation. D. M. Gabbay and F. Guenthner, eds, *Handbook of Philosophical Logic, 2nd Ed*. Kluwer, Dordrecht, 2001.

[2] 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):321–357, 1995.

[3] P. Baroni, M. Giacomin. A recursive approach to argumentation: motivation and perspectives. *Proc. 10th Int. W. on Non-Monotonic Reasoning (NMR 2004)*, 50–58, Whistler, BC, 2004.

[4] P. Baroni, M. Giacomin, G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *Artif. Intell.*, 168(1-2):165–210, 2005.

[5] P. Baroni, M. Giacomin. Solving semantic problems with odd-length cycles in argumentation. *Proc. 7th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2003)*, 440–451, Aalborg, Denmark, 2003.

[6] S. Coste-Marquis, C. Devred, P. Marquis. Prudent semantics for argumentation frameworks. *Proc. 17th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI 2005)*, 568–572, Hong Kong, China, 2005.

[7] S. Coste-Marquis, C. Devred, P. Marquis. Sémantiques prudentes pour les systèmes d'argumentation. *Proc. 15th Congres AFRIF-AFIA Reconnaissance des Formes et Intelligence Artificielle (RFIA 2006)*, Tours, F, 2006.

[8] D. Makinson, K. Schlechta. Floating conclusions and zombie paths: Two deep difficulties in the 'directly skeptical' approach to defeasible inheritance networks. *Artif. Intell.*, 48(2):199–209, 1991.

[9] J. L. Pollock. Defeasible reasoning with variable degrees of justification. *Artif. Intell.*, 133(1–2):233–282, 2001.

[10] J. F. Horty. Skepticism and floating conclusions. *Artif. Intell.*, 135(1–2):55–72, 2002.

[11] H. Prakken. Intuitions and the modelling of defeasible reasoning: some case studies. *Proc. 9th Int. W. on Non-Monotonic Reasoning (NMR 2002)*, 91–102, Toulouse, F, 2002.

[12] G. A. W. Vreeswijk. *Studies in Defeasible Reasoning*. PhD thesis Vrije Universiteit, Amsterdam, 1993.

[13] M. W. A. Caminada. *For the Sake of the Argument; explorations into argument-based reasoning*. PhD thesis Vrije Universiteit, Amsterdam, 2004.

[14] J. L. Pollock. How to reason defeasibly. *Artif. Intell.*, 57(1):1–42, 1992.

[15] C. Cayrol, M.-C. Lagasquie-Schiex. Gradual acceptability in argumentation systems. *Proc. 3rd Int. W. on Computational Models of Natural Argument (CMNA 2003)*, 55–58, Acapulco, Mexico, 2003.

[16] P. Baroni, M. Giacomin, G. Guida. Towards a formalization of skepticism in extension-based argumentation semantics. *Proc. 4th Int. W. on Computational Models of Natural Argument (CMNA 2004)*, 47–52, Valencia, E, 2004.

[17] P. Baroni, M. Giacomin. Evaluating argumentation semantics with respect to skepticism adequacy. *Proc. 8th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, 329–340, Barcelona, E, 2005.

Argumentation Tools

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Translating Wigmore Diagrams

Glenn ROWE and Chris REED
School of Computing, University of Dundee, Dundee DD1 4HN, UK
[*growe* | *chris*]*@computing.dundee.ac.uk*

Abstract. In the early 20th century, J.H. Wigmore described a new method for analysing and laying out arguments in legal cases. His proposal was the first system of argument diagramming, and it is still in use in jurisprudence today. Wigmore diagrams offer a rich ontology of argumentation concepts which in some respects are close to ideas in other, more modern systems of argument analysis and argument diagramming – whilst in other areas, is much richer and more specific than alternatives. The features of Wigmore analyses might reasonably be expected to contribute to modern, computational approaches to argument, both in the legal domain and more broadly. This paper explores some of the key issues in representing Wigmore analyses and translating between them and other systems of analysis such as those founded upon Toulmin models and scheme-based models.

Keywords. Argument Diagramming. Argument Interchange. Argumentation Schemes. Legal Argument. Wigmore Diagrams

Introduction

Analysing argumentation with diagrams has long been a technique developed for pedagogy in argument and critical thinking [1], and has also played an important role in developing both logical and argumentation theoretic accounts of the structure of monological and dialogical argument [2]. More recently, it has become the subject of widespread investigation in the AI and computer science community [3], [4], [5].

Recently [6] analyzed the issues involved in translating between Toulmin diagrams and 'standard' argument diagrams (the latter so called because they represent the most common approach in argumentation theory and critical thinking), with a view to arriving at a general language for representation of argument. Analysis of Toulmin and standard argument diagrams is facilitated by the use of *Araucaria* [4], a software package which (in its version 3.0 form) allows text to be marked up and diagrammed using either method.

Araucaria has recently been extended to version 3.1, which now allows Wigmore diagrams to be created from marked-up text. Araucaria allows translation between Wigmore diagrams and both standard and Toulmin diagrams. Implementing this translation has given rise to a number of interesting issues which this paper explores.

1. Wigmore diagrams

Wigmore [7] introduced a method of portraying legal arguments in diagrammatic form. Wigmore diagrams are superficially similar to standard diagrams, in that they use a type of 'box and arrow' structure: a given statement is supported by one or more other statements, and in turn can form part of the support of another statement above it. There are no divergent arguments (i.e. arguments where one statement can be used to support two or more other statements). Though more recent authors (most prominently, Schum [8]) have developed and refined Wigmore's charting mechanism, the original remains not only a landmark in evidential charting, but also a tool in practical use by legal professionals around the world.

Wigmore has, however, constrained the types of support allowed by classifying the various types of statement and inference according to their roles in a court case, and the resulting diagrams are intricate and complex, made up from over 30 separate diagrammatic components and conventions. Each statement in a Wigmore diagram is referred to as *evidence* of one type or another. Some evidence nodes can be *affirmatory* (i.e supporting another statement) or *negatory* (i.e. arguing against another statement). Support arrows in the diagram are referred to as *forces*, and can have varying degrees, ranging from 'no effect' to very strong positive or negative force upon the statement to which it leads.

We will not give a complete catalogue of the various types of evidence and forces that Wigmore uses in his diagrams, since we are concerned more with the general categories of evidence and force and how these translate into other diagram types. Therefore, we begin with a summary of these categories.

1.1.Types of evidence

The various evidence nodes in a Wigmore diagram fall into several broad categories. First, evidence can be classified by the party that offers it. In a typical court case, there are two sides to the argument: the prosecution and the defense.

From another viewpoint, evidence can be *testimonial*, *circumstantial*, *explanatory* or *corroborative*. Testimonial evidence is evidence that is stated by the witness as a fact, usually because they observed or knew something directly. Thus a witness M may testify that the defendant had the knife because M saw the defendant holding it.

Circumstantial evidence is evidence that requires some inference. For example, the police may testify that they found the knife at the scene of the crime where the defendant was known to have been at the time, therefore the defendant must have used the knife to commit the crime.

Explanatory evidence is evidence that is put forward to counter or lessen the impact of testimonial or circumstantial evidence. For example, for the testimonial evidence given above, the witness may have been too excited to see who was holding the knife, or in the circumstantial case, there could have been a third party who dropped the knife at the scene of the crime.

Corroborative evidence is evidence that supports testimonial or circumstantial evidence. For example, in the testimonial case, the witness may state that they were not excited at the time and clearly saw the defendant holding the knife. In the

circumstantial case, the police may testify that no footprints other than those of the plaintiff and defendant were seen so no third party could have dropped the knife.

All four of these types of evidence can be offered by either side in the case. In addition, testimonial and circumstantial evidence can be either affirmatory or negatory (explanatory and corroborative evidence only exist in one form). Adding up all the possibilities gives a total of 12 different types of evidence (e.g. negatory testimonial evidence offered by the defendant is one particular type of evidence, while affirmatory testimonial evidence offered by the defendant is another, and so on).

1.2.Types of force

The support arrows linking one bit of evidence to another can have a number of symbolic modifiers attached to them. The main categories in which we are interested here are *affirmatory* and *negatory* force (i.e. the inference either supports or detracts from the conclusion). These force categories are applied only between a statement and its supporting testimonial and circumstantial evidence. The force provided by explanatory and corroborative evidence can vary only by degree.

Each evidence node in the diagram can have up to three groups of other evidence nodes influencing it. The testimonial and circumstantial nodes are grouped below the supported node; the explanatory nodes are grouped to the left, and the corroborative nodes are grouped to the right. Each *group* of nodes may be given a *net probative value* (in Wigmore's words); that is, the net effect of all the evidence in the group can be considered and assigned a single symbol on the support arrow indicating its net effect on the conclusion. The assignment of grades and degrees of force on the support arrows is largely subjective and must be decided by the person constructing the chart.

1.3.Example

An example adapted from [7] is shown in Figure 1.

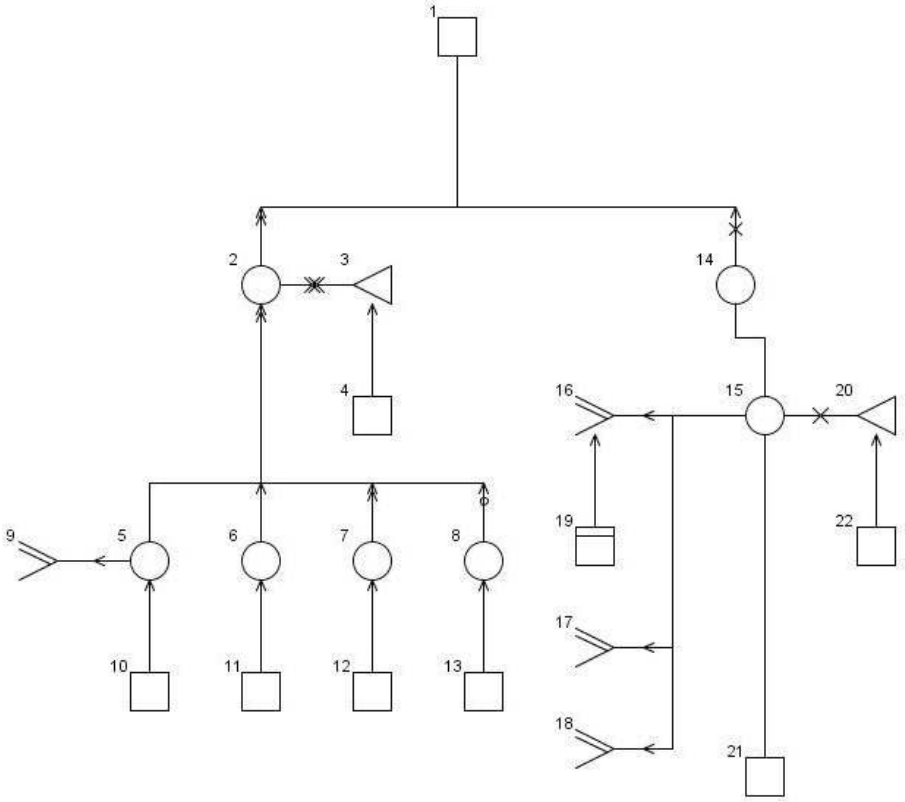


Figure 1. A sample Wigmore diagram, generated by Araucaria

In Figure 1, node 1 is the conclusion which the prosecution is attempting to prove. In the diagram, square nodes are testimonial evidence, circular nodes are circumstantial evidence, nodes with > symbols (such as nodes 9, 16, etc) are explanatory and closed triangular nodes (such as nodes 3 and 20) are corroborative evidence.

Nodes with a double line near the top (9, 16, 17, 18 and 19 in Figure 1) are defendant's evidence; all other nodes are prosecution's evidence. In this diagram, therefore, the prosecution is putting forward most of the evidence and the defense is providing explanatory evidence to counter the prosecution's argument at nodes 5 and 15.

The various symbols on the support arrows indicate the degrees of support. A single arrow indicates the direction of support, so that node 19 supports node 16, for example. A double arrow, such as from nodes 2 and 7, indicates strong support. The arrow on the edge between nodes 16 and 15 indicates that node 16 detracts from the support to node 15, which is to be expected since node 16 is explanatory and attempts to lessen the effect of node 15. The X on the edge between nodes 15 and 20 indicates that the corroborative node 20 supports node 15.

The small circle on the edge leading out of node 8 indicates a negatory force, so that node 8 detracts from the support to node 2. The double arrow just below node 2 indicates the net probative value of nodes 5, 6, 7 and 8.

2. Translating Wigmore diagrams

2.1.Desiderata

Our experience with translation between Toulmin diagrams and their "standard treatment" counterparts [6] has yielded desiderata for the process:

1. Translation should be deterministic, always providing the same output for any given input
2. Translation should be "symmetrical", i.e. translation from A to B should be 1:1 and onto, as should backtranslation from B to A, so that backtranslation from translation is always equivalent to identity
3. Translation should make maximal use of a common interlingua where possible
4. Where (3) cannot be met, theory specific analysands should be included by extending the interlingua

In the context of Araucaria [4], the interlingua is the Argument Markup Language, AML, a standard XML-based language which may be used to represent arguments. Here we explore the translation of Wigmore diagram types into standard notation, and from standard it is further possible to derive a Toulmin diagram interpretation. There are two main considerations in translating Wigmore diagrams: evidence nodes and support forces (the arrows between nodes). We take them in turn.

2.2.Translating Wigmore evidence nodes

2.2.1.Type of Evidence

A testimonial or circumstantial evidence node may have up to three supporting groups of nodes: other testimonial or circumstantial evidence, explanatory evidence and corroborative evidence. Each of these three groups of nodes are represented in the diagram by a set of nodes that have support edges converging on a single edge which then supports the parent node.

There is a superficial diagrammatic resemblance between the Wigmore notation for a group of supporting nodes and the linked argument structure in the standard diagram. It is tempting, therefore, to infer an equivalence between these two structures. However, we believe this correspondence is illusory. The linked argument in a standard diagram implies that all the premises making up the linked group of nodes are required for the connection between these nodes and the node they support. Common examples of linked arguments are found in argumentation schemes: the argument from expert opinion, for example, requires both that the expert have appropriate domain knowledge, and that the proposition they are advocating lies within that domain. In a

Wigmore diagram, however, *all* nodes of a given type that support another node are grouped together, regardless of whether some of these nodes form linked arguments and others stand alone as support for the parent node.

A Wigmore diagram also strongly reinforces pictographically the tripartite grouping of all evidence. One possible way of representing a Wigmore analysis is therefore to introduce virtual "aggregation" nodes in the argument that aggregate all the corroborative evidence supporting a node, all the explanatory evidence supporting a node, and all the other (i.e. testimonial or circumstantial) evidence supporting a node. These intermediate nodes might then be further supported in their turn by convergent arguments from the various premises. An analysis such as Figure 2a, for example, might be rendered at a deep level by the representation in Figure 2b, with C1, Ev1 and Ex1 aggregating the corroborative, testimonial and explanatory evidence for claim 1, respectively.

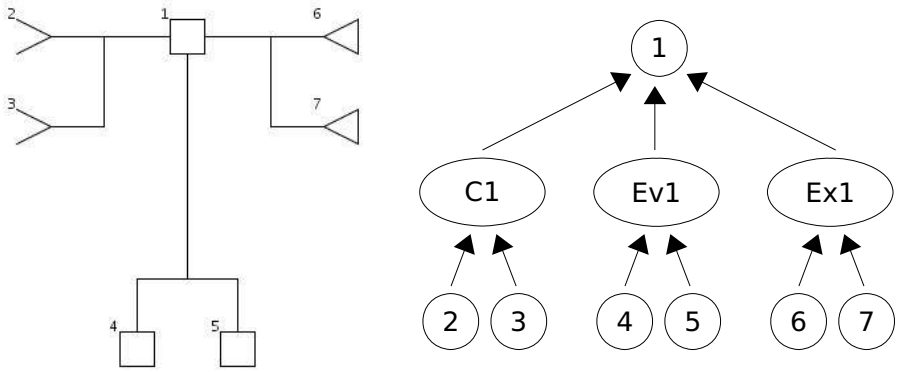


Figure 2. A sample Wigmore diagram (a) and possible deep structure representation (b)

In this way, the ontological status of nodes in the Wigmore analysis (i.e. whether they are corroborative, explanatory or testimonial/circumstantial) is captured by structural features in the AML deep representation. Unfortunately, this misrepresents the arguments in an important way. The role of "corroborating" evidence is, as the terminology suggests, one of working with elements of testimonial and circumstantial evidence to support a claim. In this respect, it is most similar to traditional linked argumentation – but the linkage crosses the groupings in Figure 2b – so, for example, it might be that 2 and 4 form a linked argument, and 3 and 5 form a linked argument. The analysis in Figure 2b not only makes such relationships opaque, it absolutely proscribes the representation of such relationships.

The problem is compounded in that an analysis performed in the Wigmore style provides no mechanism for determining which premises of a claim are linked and which are not. Thus we have no choice but to represent all the nodes supporting another node in a Wigmore diagram as single, unlinked nodes in a standard diagram. Similarly, there is no distinction in a standard diagram between the concepts of explanatory, corroborative, testimonial or circumstantial evidence, so all nodes from all these groups must be treated equally when drawn in a standard diagram.

We can use similar considerations to translate in the reverse direction: from standard to Wigmore. A standard diagram does not contain any information on the type of evidence represented by a node, so we really have no choice but to represent all standard nodes, linked or convergent, as one node type in Wigmore. For convenience, Araucaria interprets all standard nodes as testimonial affirmatory nodes (represented by a plain square) in Wigmore.

The reader may be wondering how these rules conform to our desire to use the AML structure to represent all arguments as standard and then translate to other diagram types. If Wigmore diagrams contain properties not representable in standard, how do we store these properties in AML, thereby ensuring that our second desideratum is met? The answer is that no interchange format will be able, a priori, to cater for all possible representational and operational schemes that involve argument [5]. Instead, AML is designed to support extensibility through a simple "role" mechanism that allows new ontological categories to be catered for in the representation, without the representation having to revise existing analyses. Specifically, individual propositions within an analysis can be marked as taking on a particular role in a particular class. So, for example, in the Toulmin class, a proposition might be marked as a "warrant" - a concept that only makes sense in the context of Toulmin analyses. Of course, if these extensions are not only numerous but also individually significant, then the benefits of an interchange language such as AML are eroded. The exponentially expensive problem of translation between the different classes returns. AML takes a pragmatic solution, providing as much generic capability as possible, and supporting extensions that are intended to be small scale. If particular software systems aim to make use of these extensions in translation then they are not prohibited from doing so.

In the Wigmore case, the four basic types each represent different roles: corroborative, explanatory, testimonial and circumstantial.

2.2.2.Ownership of Evidence

A further complication arises in that Wigmore diagrams distinguish explicitly between evidence offered by prosecution and that offered by defence (the extra top-most bar indicates diagrammatically the latter). Though neither Araucaria nor AML pretend to be able to handle either dialogue or a record of dialogue, they nevertheless both support identification of "owners" in standard analyses – i.e. the identity of the individual, group of viewpoint of which a given proposition is claimed. This is useful for analysing arguments in which, for example, a counter-argument to the author's position is presented and countered. The same machinery can be put to use for distinguishing between prosecution and defence arguments, inasmuch as Wigmore analyses allow the specification of just exactly those two owners and no others. This is an example of desideratum (3) driving representational re-use.

2.2.3.Evidence sense

Wigmore explicitly distinguishes between evidence that is *affirmatory* and evidence that is *negatory*. Unfortunately, Wigmore's presentation leaves it unclear as to exactly what is meant by negatory evidence (and there are few examples of it in his writings). There are three possible interpretations:

1. Evidence can only be defined as negatory with respect to other evidence (implicit or explicit) that is affirmatory. So for example, the claim that "the murderer was in the garden" might be classified as negatory with respect to another claim that "the murderer was in the house".
2. There is something intrinsic to negatory evidence which means that a human can inspect a claim and determine whether or not it is negatory. Such a determination could conceivably be related to burden of proof (so, e.g., a claim such as "there is no evidence that the murderer was in the house" as affirmatory).
3. Negatory means virtually nothing at all, making only a rhetorical distinction rather than a truth functional one (so that, e.g., "the murderer was not in the house" is negatory whilst "the murderer was in the garden" is not).

Option (1) is at the heart of most concepts of negation and contrariness: in propositional accounts, ~p derives its interpretation from the meaning of p; in Araucaria-style analyses, a refutation links a claim and counterclaim; in the Toulmin diagram [9], a rebuttal works to cancel the data-claim connection. Yet there is no indication that this was what Wigmore intended, and the few examples suggest that evidence can be negatory quite independently of other claims that are available. Option (2) would require highly contentious linguistic and philosophical assumptions, but in any case, is computationally intractable and therefore of limited interest here. Option (3) though perhaps one of the most disappointing from a formal point of view seems to resonate most closely with Wigmore's account. There is social psychological evidence that positively presented evidence may be looked upon more favourably than negatively presented evidence [10]. Perhaps therefore, it is this linguistic or rhetorical effect that Wigmore is tackling with his "negatory" class (given that juratorial presentation is a constant motivation for Wigmore). For a representation scheme, this requires nothing more than a single additional role tag for the evidence "sense" indicating whether a piece of evidence is affirmatory or negatory. We return to the problem of "negatoriness" in the context of the relations between propositions, below.

2.3.Translating Wigmore support forces

2.3.1.Premise support

The categories of support forces in a Wigmore diagram offer interesting scope for finding corresponding structures in a standard diagram. Looking back at Figure 1, we see that there are various symbols such as arrowheads, double arrowheads, Xs, double Xs, little circles and so on that are drawn on the support edges. These symbols all indicate either the degree or force with which that edge implies support for the node to

which it leads, or whether the force is affirmatory (supports the conclusion) or negatory (detracts from the conclusion).

The degree of support has a natural correspondence in the 'evaluation' feature of a standard diagram (which has been equated with the qualifier in a Toulmin diagram [6]). We can therefore use the Wigmore description of the force as an evaluation label in a standard diagram. For example, the single arrowhead on the support edge from node 4 to node 3 in Figure 1 indicates 'provisional' support, while the double arrowhead on the edge leading out of node 7 indicates 'strong' support. Other symbols have similar meanings: a complete list can be found in [7]. One oddity is the "detracts" force, which could be equated with negatory support. Wigmore, however, does not do so, and therefore neither does Araucaria's interpretation of Wigmore analysis – even though that leaves diagrams in which "support" arrows are, somewhat counterintuitively, labelled with "detracts".

2.3.2.Evidence Set Support

An important complication is that Wigmore analyses permit a very slightly finer-grained analysis of these evaluative components. For each premise, an evaluation is possible – in Figure 1, for example, premises 5, 6, 7 and 8 can each have independent evaluations. In addition, however, the set of testimonial evidence (composed of premises 5, 6, 7 and 8) can also itself have an evaluation that is separate again. Recall from the previous section that the ontological categories into which evidence is divided are simply being marked as "role" tags on the evidence nodes themselves in AML, with the result that there are no nodes in the deep representation corresponding to the set of testimonial evidence. There is, therefore, no edge in that deep representation to which an evaluation can be attached. Where then does such evaluation belong? The solution is to recognise that these evaluations are intimately tied to the claim to which they lead – i.e. the evaluation on a set of testimonial evidence is not attached to any particular member of the set, but rather to the claim that the set putatively supports. For each of the three sets that a given claim can have (corroborative, explanatory, testimonial/circumstantial), a new role tag is provided that takes the evaluative force marked for that edge. This role tag is attached to the claim.

2.3.3.Negatory Support

The presence of a small circle on an edge in a Wigmore diagram (such as that on the edge leading from node 8) indicates *negatory* force, which means that the node argues *against* its parent. This clearly suggests some relation to the *refutation* in the standard model (or the *rebuttal* in Toulmin). When translating the Toulmin rebuttal into a standard analysis, the closest match is to introduce an "added negation", so that in essence a rebuttal is the contrary of an implicit warrant [6]. In the Wigmore case, it may seem that we have a more straightforward situation, since Wigmore does not consider the subtle nuances of the Toulmin datum-warrant-rebuttal model. If a node supports another node with negatory force, then in Wigmore, the implication is that the first node counters or refutes the statement being made in the second node. Thus it may seem that we could simply map any node with negatory force on another into a refutation in the standard model, as suggested in Figure 3.

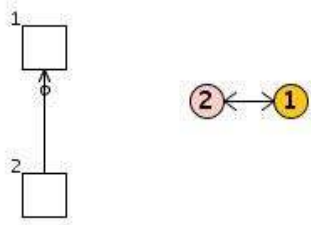


Figure 3. Simplistic negatory/refutational translation from Wigmore (a) to Standard (b)

The problem here is that the standard model (with its heritage in a propositional account) only allows a maximum of one refutation for any given node (i.e. refutation is a relationship between a proposition and its contrary, between *p* and not-*p*). In Wigmore, however, any number of nodes may support another node with negatory force. More importantly, Wigmore's use of negatory force seems to be functioning in a different way, typically functioning not as straightforward refutation, but rather much more like the rebuttal in a Toulmin diagram. The challenge can be addressed by exploiting this similarity with the Toulmin case: by introducing an 'added negation' which is refuted directly by the node with negatory force. This added negation node in turn supports (positively) the node supported directly in the Wigmore diagram. In fact, the simplest way of understanding the translation is not by comparing it with the standard treatment at all, but rather, by considering its translation to a Toulmin diagram (which then, of course, yields a standard analysis by existing translation mechanisms). Figure 4 demonstrates the idea using an example from [11]:

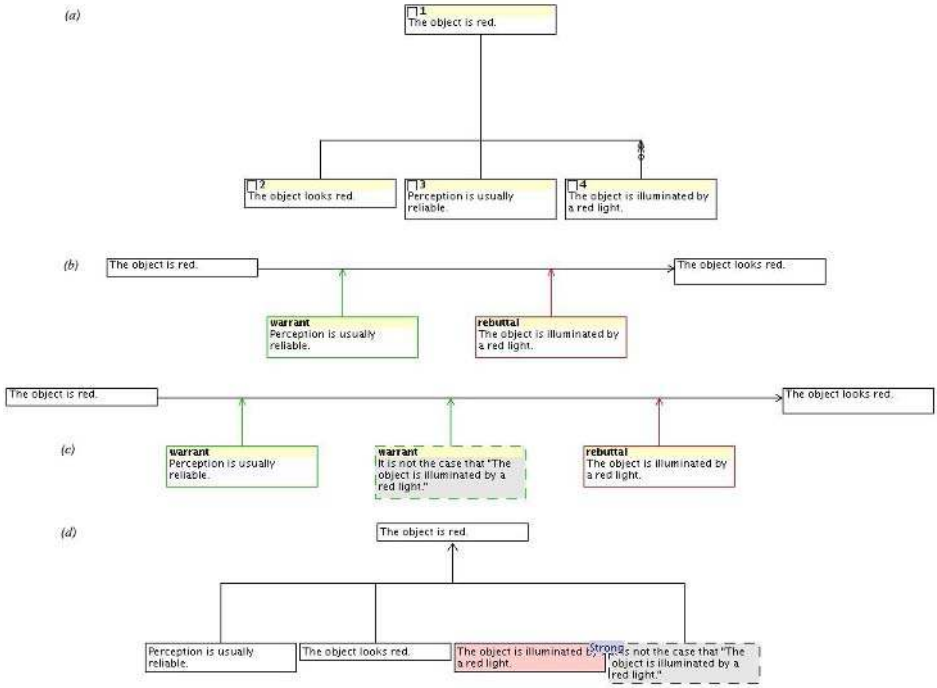


Figure 4. Pollock's example (a) analysed as a Wigmore diagram; (b) its translation into a Toulmin diagram; (c) the Toulmin diagram showing its implicit "added negation"; and (d) the translation to a standard analysis (NB. Qualifiers have been omitted to improve clarity)

2.4. Example

Drawing together all of these aspects of the translation, Araucaria implements a scheme by which the Wigmore diagram shown in Figure 1 yields the following in a standard treatment analysis:

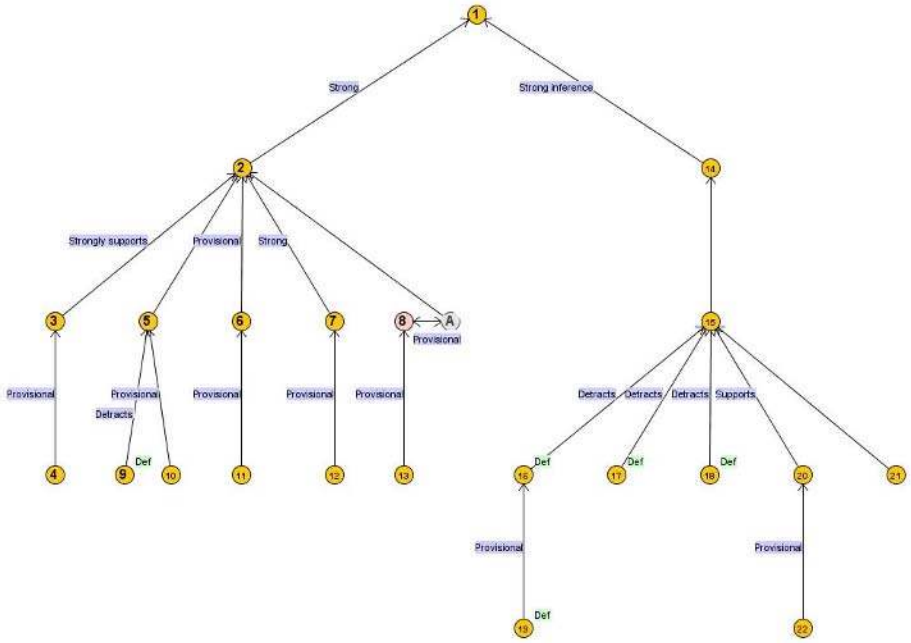


Figure 5. The standard diagram translation produced by Araucaria from the Wigmore diagram in Figure 1

It can be seen that all numbered nodes are translated directly as convergent arguments in standard. Where the Wigmore analysis has a force indicated, the translation here shows that force textually. Where the Wigmore analysis explicitly marks a node as being part of the defence's argument, it is marked “Def” in Figure 5 (anything not marked Def is assumed to be a part of Prosecution's argument). Node 8, which supported node 2 with negatory force, has an added negation node A inserted. Node 8 becomes the refutation of A, and A in turn supports node 2. Finally, any information not displayed (such as the arrangement of claims into corroborative, testimonial and explanatory groups) is latent in the deep AML representation. In this way, both the Wigmore analysis of Figure 1, and this standard version in Figure 5 can be recovered from the AML representation.

3. Conclusions

As with any inter-translation between theories with different backgrounds, goals, and working methods, there is a limit to what can be achieved in automation. The Araucaria research programme has taken a pragmatic approach, building on points of theoretical correspondence and aiming to develop a system that is at the same time usable for adherents of a particular theory of analysis, and also intuitive in its conversion of materials from one theory to another. In adding Wigmore analysis to the set of techniques supported by Araucaria, the project has encountered interesting challenges in both code and theory, but the result is, at the very least, a tool that supports analysis using the Wigmore style, which is in itself unique. This fact means that professional users of such analytical techniques (primarily in the judiciary) have the option of using a software tool to speed their analyses, and, as a further benefit, academic study of argumentation can in some situations have access to those analyses.

Because of the support for translation, the work also means that the large online database of analysed argumentation can be presented in Wigmore style, and can be further extended by analyses natively produced using Wigmore concepts. The translation mechanism meets the full list of desiderata laid out in section 2.1. Finally, from an academic point of view, the exercise has demonstrated the extensibility of AML, which augurs well for future developments in supporting argument interchange between diverse user groups, different disciplines, and various implemented systems.

References

[1] Walton, D. (2006) *Fundamentals of Critical Argumentation*, Cambridge University Press.
[2] Walton, D. and Reed, C. (2003) "Diagramming, Argumentation Schemes and Critical Questions" in F.H. van Eemeren, J.A. Blair, C.A. Willard & A.F. Snoek Henkemans (eds)*Anyone Who Has a View: Theoretical Contributions to the Study of Argumentation*, Kluwer, Dordrecht, pp195-211.
[3] Kirschner, P.A., Buckingham Shum, S.J. and Carr, C.S. (2003)*Visualizing Argumentation*, Springer.
[4] Reed, C. and Rowe, G.W.A. (2004) "Araucaria: Software for Argument Analysis, Diagramming and Representation", *International Journal of AI Tools*, **14 (3-4)**, pp961-980.
[5] Willmott, S., Vreeswijk, G., South, M., Chesnevar, C., McGinnis, J., Rahwan, I., Reed, C. and Simari, G. (2006, to appear) "Towards an Argument Interchange Format for Multiagent Systems" in Maudet, N., Parsons, S. & Rahwan, I. (eds) *Proceedings of the 3rd International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2006)*, Springer.
[6] Reed, C. and Rowe, G.W.A. (2005) "Translating Toulmin Diagrams: Theory Neutrality in Argument Representation", *Argumentation* **19 (3)**, pp267-286.
[7] Wigmore, J.H. (1931) *The Principles of Judicial Proof* (2nd Edition) Little, Brown & Co.
[8] Schum, D., *Evidential Foundations of Probabilistic Reasoning*, John Wiley & Sons, Toronto, 1994.
[9] Toulmin, S.E. (1958) *The Uses of Argument*, Cambridge University Press.
[10] McGuire, W.J. (1969) *The nature of attitudes and attitude change*, in: G. Lindzey, E. Aronson (Eds.), The Handbook of Social Psychology, vol. 3, 2nd ed., Addison-Wesley, Reading, MA, pp. 136-314.
[11] Pollock, J.L. (1995) *Cognitive Carpentry: How to Build a Person*, MIT Press.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

First Steps Towards the Automatic Construction of Argument-Diagrams from Real Discussions

Daan Verbree [a], Rutger Rienks [a,1], Dirk Heylen [a]
[a] *Human Media Interaction (HMI)*
University of Twente, Enschede, The Netherlands
{Verbree, Rienks, Heylen}@ewi.utwente.nl
Home page: http://hmi.ewi.utwente.nl

Abstract. This paper presents our efforts to create argument structures from meeting transcripts automatically. We show that unit labels of argument diagrams can be learnt and predicted by a computer with an accuracy of 78,52% and 51,43% on an unbalanced and balanced set respectively. We used a corpus of over 250 argument diagrams that was manually created by applying the Twente Argument Schema. In this paper we also elaborate on this schema and we discuss applications and the role we foresee the diagrams to play.

1. INTRODUCTION

Argumentation has been proposed as constituting human kind's primary means of making progress [1]. It is pervasive in everyday life and plays an important role in human communication. Argumentation is situated in discussions, conversations and meetings, the arenas where one argues with another and one or more sides are attempting to win the approval of the opponent or of a designated audience.

Within organizations the outcomes of conversations or meetings are normally not much more than what participants are able to recall. When lucky some notes were taken, or more formal meeting minutes were made with a list of action items. Generally, a lot of energy and information that has been put into the actual outcome is never seen again.

We have tried to find an approach that is able to capture the lines of the deliberated arguments in meeting discussions. This approach, the TAS-schema, was introduced in [2] and promises to be a valuable technique for capturing organizational memory. The structure that the argument trees encapsulate reveals information about the trail or path that has been taken in a meeting. It shows the line of reasoning at specific moments in time. The method can aid querying and summarization systems and is being used in meeting browsers (See fig 1). The possibility of preserving the arguments and their coherence relations for future explorations make them potentially valuable documents that contain a tacit representation of otherwise volatile knowledge [3,4].

[1]Correspondence to: Rutger Rienks, University of Twente, Human Media Interaction, PO BOX 217, 7500 AE Enschede. Tel.: +31-53-4893740.

For end users of the representations, argument diagrams constitute a representation of the content of a conversation that leads to quicker comprehension, deeper understanding. They enhance the ability to detect weaknesses or flaws in the argumentation [5,6]. Furthermore is has been claimed that they aid the decision making process and that they can be used as an interface for communication to maintain focus, prevent redundant information and to save time [7,8].

In this paper we present our initial research efforts in this area. Before we elaborate in more detail on how we created a corpus of annotations in Section 3, Section 2 provides an introduction of the TAS-schema. Section 5 is devoted to the learnability of (a subset of) the schema and investigates if an automatic tagger can one day produce the actual schemes autonomously.

2. The Twente Argument Schema

The Twente Argument Schema (TAS) is a schema designed to define argument diagrams for meeting discussion transcripts. Following most of the existing diagramming techniques, application of the method results in a tree structure with labelled nodes and edges. The nodes of the tree contain complete speaker turns or parts of speaker turns whereas the edges represent the type of relation between the nodes. The complete label set is shown in Table 1.

Node labels	Relation labels
Statement	Positive
Weak statement	Negative
Open issue	Uncertain
A/B issue	Request
Yes/No issue	Specialization
	Elaboration
	Option
	Option exclusion
	Subject-to

Table 1. The labels of the Twente Argument Schema

The TAS trees are away to capture the most important conversational moves in dialogues in which participants discuss the pros and cons of certain solutions to a problem, marking the arguments in favor of or against the various solutions. TAS distinguishes acts in which issues are raised (questions put forward) from statements in favor of a particular position. The schema allows one to distinguish strong from weak statements. Three types of issues can be marked: open issues, issues for which a choice of solutions is presented, and yes/no issues. There are various kinds of relations that are marked. In many cases statements are not simply supporting or undercutting other statements but rather are (near) synonymous. So, besides a marking for positive/negative, also relations such as restatements, specializations or generalizations have been introduced. More details on the nodes and relation labels are provided below.

TAS was constructed in a way that it preserves the conversational flow. By applying a left-to-right, depth-first walk through the resulting trees, the reader is able to read the nodes as they unfolded in time. This is realized by assuring that in principle every next contribution of a participant becomes a child of the previous contribution, unless the current contribution relates more to an ancestor. The resulting diagrams thus provide a comprehensive overview of the discussion relating the contributions from the individual participants. For a video about the TAS-schema and its applications see: *http://hmi.ewi.utwente.nl/video4ami/UT_argumentation.wmv*. An example of a TAS argument diagram, embedded in a meeting browser application, is shown in Figure 1.

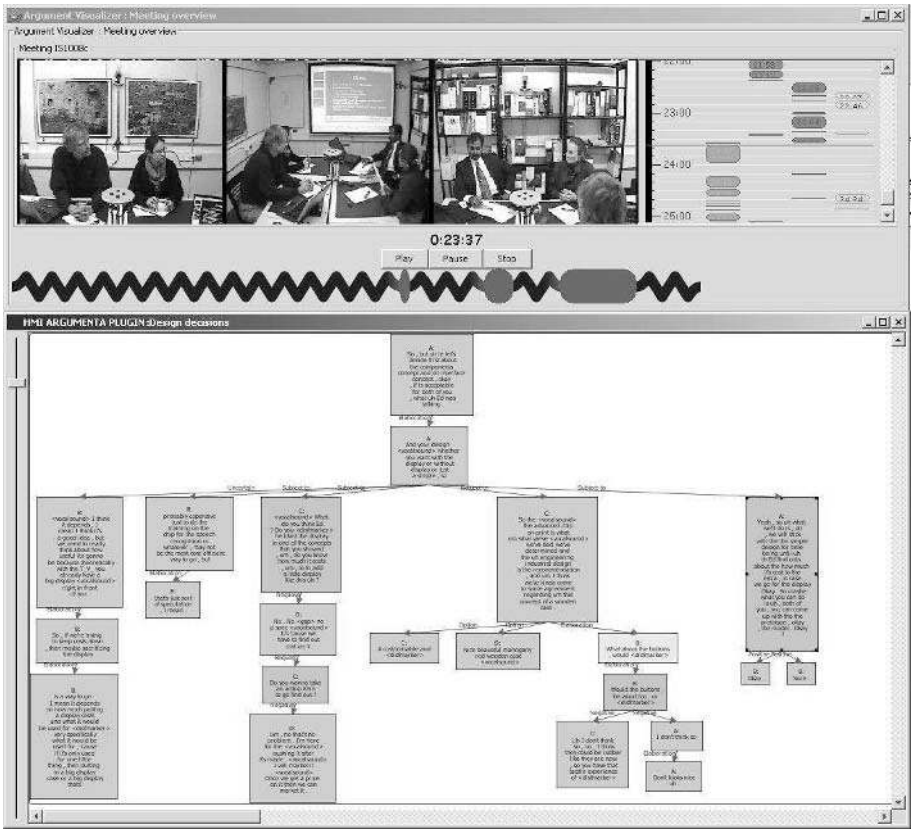


Figure 1. TAS Argument Diagrams in use as part of a meeting browser.

2.1. Related theories

There exist a number of different theories for labelling the contents of conversations. The TAS schema combines insights from various approaches among which are schemes for semantic and pragmatic relations between utterances such as Rhetorical Structure Theory (RST) [9], Dialog Act (DAs) annotation schemes [10], and theories or schemes that focus on the argumentative relations such as Toulmin's model [11], and the IBIS scheme [12]. For a more elaborate background about the motivations and ancestory of the schema the

reader is referred to [2] and [13]. In the following paragraphs we introduce TAS in more detail.

2.2. The Unit Labels

The content of the nodes are labeled with speech acts. If an utterance contains more than one act, they are split up into more than one node. In line with [14] backchannel utterances such as 'uhhuh' and 'okay' are filtered out and to be neglected, since they are generally used by listeners to indicate they are following along, and not necessarily indicating (dis)agreement. The nodes in our model are labeled either as issues or as statements.

Issues can also be found in the IBIS model. There, they are represented as questions [12] as they can be seen as utterances with a direct request for a response. Kestler distinguishes two fundamental types of question with respect to conversational moves. These are *yes-no questions* and *why questions* [15]. A yes-no question admits only two kinds of answer, be it either supportive, or negative but rules out the uncertainty *option* 'I don't know'. The *why questions* are a subclass of a more general type of *open question*. The number of positions participants can take on such an issue depends on the set of possible options enabled by the type of question or issue.

In our Schema we have defined three different labels for our nodes to represent the issues: The **'Open issue'**, the **'A/B issue'** and the **'Yes-No issue'**. The open issue allows any number of possible replies possibly revealing positions or options that were not considered beforehand. This in contrast with the A/B issue, that allows participants to take a position for a number of positions which should be known from the context (c.f. 'Would you say ants, cats or cows?'). The yes-no issue, in line with the yes-no question directly requests whether the participants positions agree or disagree with the issue. A *why question* in TAS is modelled as an open question with a clarification relation (see below).

The positions that participants take are generally conveyed through the assertion of a **statement**. The content of a statement always contains a proposition which can be a description of facts or events, a prediction, a judgement, or an advice ([16]). Statements can vary in their degree of force and scope. Meeting participants may indicate that they are not sure if what they say is actually true. In [11] *qualifiers* provide an indication of the force of *claims*. As [17] points out, the force of an argument can also be derived from lexical cues such as the words 'likely' and 'probably'. Such statements, in which the speaker does not commit himself fully to the opinion are labeled as **'weak statements'** in TAS.

2.3. The Relation Labels

In normal texts and conversations, the statements forms a coherent whole, partly, because they are connected through semantic and pragmatic relations which in Rhetorical Structural Theory are called *rhetorical relations* [9]. The TAS schema concentrates on typical argumentative relations in conversations.

When engaged in a discussion or debate, the elimination of misunderstandings is one of the prerequisites to understand each other and hence to proceed [18]. According to Neass, participants in a discussion eliminate misunderstandings by clarifying, or spec-

ifying their statements. These moves can e.g. be observed in the criteria definition phase, of the decision making process.

The **'Generalization/Specialization'** label can be applied when a particular issue generalizes or specializes another issue. The contribution 'Which animal is the most intelligent?' can be specialized with the following proceeding contribution 'Is an ant or a cow the most intelligent animal?' which again can be specialized if one for instance asks 'Are ants the most intelligent animal?'. It is also possible that a person is not satisfied with the information or the argument explained. He can then explicitly invite the previous speaker to elaborate on his earlier statements. For these situations we define the relations **'Request'**. The **'Elaboration'** label is used if a person continues his previous line of thought and adds more information to it.

Whenever an issue is raised, an exchange of ideas about the possible solutions occurs in the decision making process. As questions call for answers, issues call for opinions expressed through statements. Whenever a statement is made as a response to an open-issue or an A/B-issue it might reveal something about the opinion of the participant on the solution space. In general a participant provides an **'Option'** to settle the issue at hand. For example when a speaker asks 'Which animal is the most intelligent?' and the response from someone else is 'I think it's an ant' the option relation is to be applied. The opposite of the option relation is the **'Option-exclusion'** relation, and it is to be used whenever a contribution excludes a single option from the solution space.

With respect to a yes/no-issue the contributions that can be made are not intended to enlarge or to reduce the solution space, but to reveal one's opinion to the particular solution or option at hand. Contributions from participants are either supporting, objecting to the issue, or express uncertainty. For this purpose the labels **'Positive'**, **'Negative'** and **'Uncertain'** are introduced. The positive relation can exist for example between a yes/no-issue and a statement that is a positive response to the issue or between two statements agreeing with each other. When one speaker states that cows can be eliminated as being the most intelligent animals and the response from another participant is that cows don't look very intelligent, then the relation between these statements is positive. The negative relation is to be applied in situations where speakers disagree with each other or when they provide a conflicting statement as a response to a previous statement or a negative response to a Yes/No-issue. In a case where it is not clear whether a contribution is positive or negative, but that there exists some doubt on the truth value of what the first speaker said, the uncertain relation is used.

The final relation of our set is applied when the content of a particular contribution is required in order to figure out whether another contribution can be true or not. We termed this the **Subject to** relation. It is related to the concession relation in Toulmin's model. It is applied for example in the situation where someone states 'If you leave something in the kitchen, you're less likely to find a cow' and the response is 'That depends if the cow is very hungry'.

3. Creating a corpus of Meeting Discussions

TAS was initially devised to create argumentation diagrams for the meetings recorded in the Augmented Multiparty Interaction (AMI) Project. The AMI project is focused on the use of advanced signal processing, machine learning models and social interaction

dynamics to improve human-to-human communications. In particular the development of tools and models that provide insight into the decision making process are of primary concern. Over one hundred hours of meetings was captured for the AMI meeting corpus. All the meetings followed a script that described the global theme and the global structure of the meeting. There were no constraints on the way participants gave content to their contribution.

The recordings consist of four people meetings constituting a design team from a small company, RealReactions. In these meetings, the participants, take four different roles: a project manager (PM), user interface specialist (UI), marketing expert (ME), and industrial designer (ID). The teams design a new kind of remote control from start to finish over a series of four meetings. Transcriptions were created for all the meetings in the AMI corpus, following strict annotation guidelines [19]. For more information about the AMI corpus, see [20].

To perform the manual TAS annotations, the annotation tool *ArgumentA* was designed by using a number of components described in [21]. ArgumentA allows annotators to select text on a transcription-view pane and label them. The label is assigned by selecting the unit text with the mouse from the transcription pane and then pressing a button that makes a label selection window pop-up from which the unit label can be picked. The labelled units appear on a canvas where they can be attached to the graph via an intuitive drag and drop interface. Once attached, a popup window appears from which the relation-label can be chosen. The resulting trees can be saved in different XML formats.

Three annotators were trained in several iterations. Apart from collectively developing the schema, elaborate discussions were held after a number of training sessions about when and why to pick a particular label in that particular case. The corpus, as it stands, comprises a total of 256 annotated discussions (diagrams) including over 5000 unit labels and 5000 relation labels.

4. Reliability of the TAS Schema

With respect to the issue of reliability one should first note that it is very well possible to end up with several diagrams from one discussion as there are likely to be more than one possible interpretation. [22] for instance showed that various different argument diagrams can be instantiated by one single text. Moreover, in Rhetorical Structure Theory (RST) [9], which addresses similar issues as the TAS scheme, the suggestion is made that the analyst should make *plausibility judgements* rather than absolute analytical decisions, implying that more than one reasonable analysis may exist.

To measure the reliability of the scheme we therefore compared the unit labels on pre-segmented discussions for four meetings (12 discussions) between two annotators. The reliability issue for the relation part of the scheme is still under investigation. It turned out that, especially in first trials the value of Cohen's kappa (κ) [23] were rather low (0.50) as a lot of confusion existed amongst the labels 'other' and 'statement'. This was resolved by a consensus definition, after which κ rose to a more acceptable value (0.87).

We also experimented with other ways to obtain reliability score based on more data. We applied techniques comparable to those introduced in [24], by setting out the results

of a classifier trained on (unit label) annotations of one annotator against the values provided by another annotator. (See Section 5).

5. Tagging the TAS-unit labels

Eventually we aim to build a system that can automatically detect discussion segments, tag individual contributions with TAS-unit-labels, depict and label the relations between the units using the TAS-relation-labels and generate a visualization of the argument diagram. In this section we report on our first experiments related to the automatic classification of the TAS unit labels.

5.1. Features

Except for the *lastlabel* feature, we only used lexical features.

? and OR A good indicator for an issue is a question mark. The *?-feature* gives a binary value whether a question mark is present or not. If a question mark is available, the number of times the word *or* appears is counted and used as a feature. (If the classification is based on transcripts derived from automatic ASR, a substitute for the question mark feature is needed.)

Length The length (number of words) of each segment is a feature. This feature helps to make a distinction between the *statement* and *other* labels.

Last Label Since discussions have the property of having some coherence we might expect that given the label of a segment the conditional chance of the label of the next segment might differ from the unconditional chance. Therefore the *lastlabel* feature, which is a bigram of the previous two labels, is used.

N-gram points The n-gram-point feature is used to reduce the number of features. At first, all bi-, tri- and quadri-grams are computed for all segments. Then, for each label a predictivity score is computed and the X most predictive n-grams are selected. The predictivity score is equal to the product of the times the ngram occurs in nodes labeled X *and* the part of this 'ngram-space' occupied by nodes of type X. For example, the score for the ngram 'what do you' (see table 2) for type *statement* is $\frac{3}{3+0+100+97+2+0} \times 3 = 0.045$.

Using the ngrams selected points, an utterance is assigned ngram points by computing all ngrams in an utterance and enumerating all the occurrences of all ngrams per order and label. If for example the trigrams listed in Table 2 are found in an utterance and the occurrences of the ngrams in the training set are as shown in the table, than this utterance will get 69 points for the *statement - trigram* feature, 31 for the *weak statement - trigram* feature and so on.

POS-ngram points The POS n-gram-point features are quite similar to the n-gram point features. But instead of attributing points to words, points are attributed to n-grams of Part-of-Speech tags.

trigram	statement	weak statement	open issue	a/b issue	y/n issue	unknown
what do you	3	0	100	97	2	0
do you think	3	1	97	92	100	0
we have to	63	30	50	1	93	4

Table 2. Examples of a trigrams found in an utterance and available in the training set

Perl scripts were used to extract the features *? and OR*, *Length*, and *Last Label* from our XML-format. The construction of n-grams was done using the N-gram Statistic Package (NSP) [25]. Using the Stanford Part-of-Speech tagger all segments were tagged to make POS-n-gramming possible [26].

5.2. Baseline

The corpus as it stands is unbalanced, consisting of 4245 *statements*, 199 *weak statements*, 244 *open issues*, 72 *a/b issues*, 460 *yes/no issues* and 3061 *others*. As a baseline we have used the implementation of a one-rule classifier resulting in a correct score of 69.1%. To see how our features would perform on a balanced corpus we also constructed a balanced corpus, having an equal number of nodes for each unit type. The baseline was again computed using a one-rule classifier, which resulted in an accuracy of 28.33%.

5.3. Results

We tried out different Machine learning techniques to produce our results, but looked into most detail at Weka's **J48** implementation of the C4.5 decision tree algorithm [27], since this classifier gave the best results as a baseline classifier compared to seven other classifiers available in Weka. Furthermore Weka's **DecisionTable** and **MultilayerPerceptron** were used on our most promising results. All our results were obtained after a 10 fold cross-validation. Here we only present our best results, a more extensive presentation of experiments and results can be found in [28].

Our best result on the unbalanced corpus is 78.52% which shows an improvement of 9.4% on the best baseline. The combined confusion matrix produced by the J48, (Table 3) shows that improvement could be obtained by features that distinguish between utterances with the label *statement* or *unknown*. The table also shows that a label such as *ab_issue* is often incorrectly classified as it has only few occurrences.

a	b	c	d	e	f	$<--$ classified as
19	15	22	1	0	15	a = ab_issue
7	116	47	9	0	65	b = open_issue
8	31	3722	388	36	60	c = statement
1	9	668	2365	2	16	d = unknown
0	2	162	21	11	3	e = weak_statement
15	45	121	9	1	269	f = yn_issue

header

Table 3. Confusion matrix of unbalanced J48-classifier on our best result

On the balanced corpora our best result was 51.43% which shows an improvement of 23.1% on the best baseline.

5.4. Elaborating on The Reliability Issue

In section 4 κ-measures were computed for the TAS annotation of the HUB corpus. Two problems met there were the small amount of discussions that could be compared and the absence of utterances of type *A/B issue* in each annotation. To get more insight in the reliability of our corpus we performed experiments where the J48 classifier was trained using parts of the corpus annotated by one annotator (row) and was tested on a part of the corpus annotated by another annotator (column). This resulted in the performances shown in table 4. When both training and test sets were picked from the same annotator, we used 10-fold cross-validation.

Trained / Tested on	Annotator 1	Annotator 2	Annotator 3
Annotator 1	84.4%	75.7%	70.3%
Annotator 2	75.6%	79.5%	66.2%
Annotator 3	67.0%	66.2%	82.2%

Table 4. Performance amongst annotators

Such a table presents an alternative view on the reliability scores.

6. Discussion and Future work

6.1. Relation with DA-Tagging

The classification task described in this paper is very similar to dialog-act tagging. Research in this field mostly concentrates on cues that are either manually [29] or automatically [30] selected. The biggest difference for our approach in comparison to earlier dialogue act classifying approaches is the use of an ngram selection method. This method selects the most predictive ngrams from the total set of ngrams acquired. We have also experimented with *compressed* feature sets. The compression decreases the size of our feature vector and therefore also decreases our computing time. This of course, by itself not an advantage, unless we maintain accuracy. In addition to the compression, we also made use of n-grams of POS-tags as has previously been done in research on the generation of backchannels in a spoken dialogue system [31]. Using the same ngrams an accuracy of 78.52% was obtained without making use of compression and a result 77.20% when using compression. These results are based on the use of the J48 classifier.

6.2. Research on other ngram-selecting methods

Our work has mostly concentrated on ngrams of words an POS-tags. Results of the experiments show that for each classifier the ngram-selecting method strongly influences the performance. More research on scoring algorithms might results in better ngram selection methods and therefore a better performance on the classification task. It is not just

the selection of the right ngrams that influences the performance of our classifiers based on ngrams, however. Also the points attributed to a feature when a ngram is present are important. In our study we have used the number of occurrences of an ngram as a feature value. It might be worth the effort to research other possible values one could assign to an ngram.

6.3. Researching the punctuation features

The use of the presence or absence of a question mark as a feature could be regarded as a form of 'cheating', since in automatic speech recognition it is very hard to recognize wether an utterance is a question or not and thus deciding on placing a question mark in the output or not (See e.g. [32]). Since we like to have or classification of a discussion using TAS to be applicable to discussion transcribed using automatic speech recognition we are considering the omission of this particular feature. Ongoing work investigates the influence of the *? and or* feature on the performance.

6.4. Applications

A plug-in has been developed for the JFerret meeting browser [33]. Users are able to access the discussions depicted on a meeting time line. For each discussion the resulting argument diagram appears allows a quick grasp of the content of the on-going discussion. Clicking on the nodes in the diagram shifts the browser directly towards the corresponding moment in the meeting.

Eventually the possible applications for meetings annotated with the TAS schema are endless. They can be used for automatic summarization purposes, or aid processes aiming to find out who adhered to a specific opinion at any given moment. They can be used to see who proposed the accepted solution, or who objected to most of the discussed points. Managers can use the diagrams to investigate what went well or wrong in the discussion and which arguments were made in favor or against a specific proposal. For more information about the sorts of applications we foresee to emerge refer to [34].

6.5. Future Work

There are currently three lines of research that we are engaged in with respect to the Argumentation Schema.

Up till now we have focused on node classification only. We are currently working on relation classification as well. Our first approach to the classification of relations are discussed in [28].

In the end, the system we would like to have the system work in real time. We are therefore considering to run tests directly on the ASR output.

Finally, investigations have started to measure the actual benefit of the use of argument diagrams in a meeting browser. Does presenting a Argument Diagram really improve the system? (i.e. are user queries answered quicker with a higher satisfaction rate?) This is certainly an important topic [35].

7. Conclusions

This paper showed some of the first steps we have taken to derive at the automatic generation of argument diagrams. A corpus containing over 250 argument diagrams deriving from real-meeting discussions has been created. Machine learning experiments on automatic tagging the unit-labels resulted in a performance of 78.52% on our unbalanced and an average of 51.43% on our balanced test set using a J48 classifier.

8. ACKNOWLEDGEMENTS

This work was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811, publication AMI-188).

References

[1] T.J. van Gelder. Argument mapping with reason!able. The American Philosophical Association Newsletter on Philosophy and Computers, 2002.

[2] R.J. Rienks and D. Heylen. Argument diagramming of meeting conversations. In A. Vinciarelli and J-M. Odobez, editors, *Multimodal Multiparty Meeting Processing, Workshop at the 7th International Conference on Multimodal Interfaces (ICMI)*, pages 85–92, Trento, Italy, October 2005.

[3] S. Buckingham Shum. Negotiating the construction and reconstruction of organisational memories. *Journal of Universal Computer Science*, 3(8):899–928, 1997.

[4] V. Pallotta, J. Niekrasz, and M. Purver. Collaborative and argumentative models of meeting discussions. In *Proceeding of CMNA-05 international workshop on Computational Models of Natural Arguments (part of IJCAI 2005)*, July 2005.

[5] D. Schum and A. Martin. Formal and empirical research on cascaded inference in jurisprudence. *Law and Society Review*, 17(1):105–152, 1982.

[6] G. Kanselaar, G. Erkens, J. Andriessen, M. Prangsma, A. Veerman, and J. Jaspers. Desiging argumentation tools for collaborative learning. In *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*, pages 51–73. Springer Verlag, London, UK., 2003.

[7] J. Yoshimi. Mapping the structure of debate. *Informal Logic*, 24(1), 2004.

[8] A Veerman. *Computer-supported collaborative learning through argumentation*. PhD thesis, University of Utrecht, 2000.

[9] W.C. Mann and S.A. Thompson. Rhetorical structure theory: toward a functional theory of text organization. *Text*, 8:243–281, 1988.

[10] H.C. Bunt. Conversational principles in question-answer dialogues. *The theory of questions*, 1979.

[11] S. Toulmin. *The uses of argument*. Cambridge University Press, 1958.

[12] W. Kunz and H.W.J. Rittel. Issues as elements of information systems. Working Paper WP-131, Univ. Stuttgart, Inst. Fuer Grundlagen der Planung, 1970.

[13] E. Van der Weijden. Structuring argumentation in meetings : Visualizing the argument structure. Master's thesis, University of Twente, November 2005.

[14] M. Galley, K. McKeown, J. Hirschberg, and E. Shriberg. Identifying agreement and disagreement in conversational speech: Use of bayesian networks to model pragmatic dependencies. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 669–676, 2004.

[15] J.L. Kestler. *Questioning Techniques and Tactics*. McGraw-Hill, 1982.
[16] F. Van Eemeren, R. Grootendorst, and F. Snoeck Henkemans. *Argumentation*. Lawrence Erlbaum Associates, 2002.
[17] F.H. Van Eemeren. A glance behind the scenes: The state of the art in the study of argumentation. *Studies in Communication Sciences*, 3(1):1–23, 2003.
[18] A. Neass. *Communication and argument. Elements of applied semantics.* George Allen & Uwin Press, 1966.
[19] J. Moore, M. Kronenthal, and S. Ashby. Guidelines for AMI speech transcriptions. Technical report, IDIAP, Univ. of Edinburgh, February 2005.
[20] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, G. Lathoud, M. Lincoln, A. Lisowska, I. McCowan, W. Post, D. Reidsma, and P. Wellner. The ami meetings corpus. In *Proceedings of the Measuring Behavior 2005 symposium on "Annotating and measuring Meeting Behavior"*, 2005. AMI-108.
[21] D. Reidsma, D.H.W. Hofs, and N. Jovanovic. A presentation of a set of new annotation tools based on the nxt api. Poster at Measuring Behaviour 2005, 2005. AMI-105.
[22] D.N. Walton. *Argument Structure, A pragmatic Theory*. University of Toronto Press, 1996.
[23] J. Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, (20):37–46, 1960.
[24] S. Steidl, M. Levit, A. Batliner, E. Nöth, and H. Niemann. "of all things the measure is man" automatic classification of emotion and intra labeler consistency. In *ICASSP 2005, International Conference on Acoustics, Speech, and Signal Processing*, 2005.
[25] S. Banerjee and T. Pedersen. The design, implementation, and use of the Ngram Statistic Package. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, February 2003.
[26] K. Toutanova, D. Klein, and C. Manning. Feature-rich part-of-speech tagging with a cyclic dependency network. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 173–180, Morristown, NJ, USA, 2003. Association for Computational Linguistics.
[27] J. Quinlan. *C4.5 : programs for machine learning*. Morgan Kaufmann, San Mateo, CA, USA, 1993.
[28] A. T. Verbree. On the structuring of discussion transcripts based on utterances automatically classified. Master's thesis, University of Twente, June 2006.
[29] J. Hirschberg and D. Litman. Empirical studies on the disambiguation of cue phrases. *Comput. Linguist.*, 19(3):501–530, 1993.
[30] N. Reithinger and M. Klesen. Dialogue act classification using language models. In *Proceedings of EuroSpeech-97*, pages 2235–2238, 1997.
[31] N. Cathcart, J. Carletta, and E. Klein. A shallow model of backchannel continuers in spoken dialogue. In *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 51–58, Morristown, NJ, USA, 2003. Association for Computational Linguistics.
[32] J. Huang and G. Zweig. Maximum entropy model for punctuation annotation from speech. *Proc. Eurospeech*, 2002.
[33] P. Wellner, M. Flynn, and M. Guillemot. Browsing recorded meetings with ferret. In *In Proceedings of MLMI'04*. Springer-Verlag, 2004.
[34] R.J. Rienks, A. Nijholt, and P. Barthelmess. Pro-active meeting assistants : Attention please! In *Social Intelligence Design*, Osaka, Japan, March 2006.
[35] T.J. Van Gelder. How to improve critical thinking using educational technology. In *Proceedings of the 18th annual conference of the Australasian Society for Computers in Learning in Tertiary education*, pages 539–548, 2001.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

The Carneades Argumentation Framework

Using Presumptions and Exceptions to Model Critical Questions

Thomas F. Gordon [a,*] and Douglas Walton [b]
[a] *Fraunhofer FOKUS, Berlin*
[b] *Dept. of Philosophy, University of Winnipeg, Manitoba, Canada*

Abstract. We present a formal, mathematical model of argument structure and evaluation, called the Carneades Argumentation Framework, which applies proof standards [1] to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis. Carneades uses three kinds of premises (ordinary premises, presumptions and exceptions) and information about the dialectical status of statements (undisputed, at issue, accepted or rejected) to model critical questions in such a way as to allow the burden of proof to be allocated to the proponent or the respondent, as appropriate.

Keywords. Argumentation Framework, Argument Evaluation, Argumentation Schemes, Critical Questions

1. Introduction

The work in this paper flows from previous attempts to solve a key problem common to AI and argumentation theory concerning the using of the device of critical questions to evaluate an argument. Critical questions were first introduced by Arthur Hastings [2] as part of his analysis of presumptive argumentation schemes. The critical questions attached to an argumentation scheme enumerate ways of challenging arguments created using the scheme. The current method of evaluating an argument that fits a scheme like that for argument from expert opinion is by a shifting of the burden of proof from one side to the other in a dialog [3]. When the respondent asks one of the critical questions matching the scheme, the burden of proof shifts back to the proponent's side, defeating or undercutting the argument until the critical question has been answered successfully. At least this has been the general approach of argumentation theory. Recently, however, it was observed [4] that critical questions differ with respect to their impact on the burden of proof. These observations led to two theories about the shifting of the burden of proof when critical questions are asked. According to one theory, when any critical question is asked, the burden shifts to the proponent's side to answer the question and, if no answer is given, the argument fails. According to the other theory, merely asking a critical question

*Correspondence to: Dr Thomas F. Gordon, Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, Berlin, Germany. E-mail: thomas.gordon@fokus.fraunhofer.de

is not enough to shift the burden of proof back to the proponent. On this theory, to make the argument fail, the question needs to be supported by further argument. Some critical questions fit one theory better, while others fit the other theory better. This duality has posed a recurring problem for the project of formalizing schemes.

In this paper, we put forward a new model for evaluating defeasible arguments that solves this problem, continuing work we began in 2005 [5,6]. The current paper presents a formal, mathematical model of argument evaluation which applies proof standards [1] to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis. The formal model is called the Carneades Argumentation Framework, in honor of the Greek skeptic philosopher who emphasized the importance of plausible reasoning [7, vol. 1, p. 33–34].

The formal model has been implemented, using a functional programming language. The implementation, also called Carneades, is being developed to support a range of argumentation tasks, including argument construction ("invention"), evaluation and visualization, primarily for use in the legal domain.

Arguments in Carneades are identified, analyzed and evaluated not only by fitting premise-conclusion structures that can be identified using argumentation schemes. Arguments also have a dialectical aspect, in that they can be seen as having been put forward on one side or the other of an issue during a dialog. The evaluation of arguments in Carneades depends on the stage of the dialog. Whether or not a premise of an argument holds depends on whether it is undisputed, at issue, or decided. One way to raise an issue is to ask a critical question. Also, the proof standard applicable for some issue may depend on the stage of the dialog. In a deliberation dialog, for example, a weak burden of proof would seem appropriate during brainstorming, in an early phase of the dialog. The Carneades Argumentation Framework is designed to be used in a layered model of dialectical argument [8] for various kinds of dialogs, where higher layers are responsible for modeling such things as speech acts, argumentation protocols and argument strategies.

The rest of the paper is structured as follows. Section 2 defines the structure of arguments and illustrates this structure with examples from related work by Toulmin, Pollock and others. Section 3 formally defines how arguments are evaluated in terms of the acceptability of statements, the defensibility of arguments, and the satisfiability of proof standards. The paper closes in Section 4 with a presentation of conclusions and ideas for future work.

2. Argument Structure

We begin by defining the structure of arguments. Unlike Dung's model [9], in which the internal structure of arguments is irrelevant for the purpose of determining their defensibility, our model makes use of and depends on the more conventional conception of argument in the argumentation theory literature, in which arguments are a kind of conditional linking a set of premises to a conclusion. Intuitively, the premises and the conclusion of arguments are statements about the world, which may be accepted as being true or false. In [5] the internal structure of statements was defined in such a way as to enable the domain of discourse to be modeled in a way compatible with emerging standards of the Semantic Web [10]. These details, however, need not concern us here. For the purpose

of evaluating arguments, the internal structure of statements is not important. We only require the ability to compare two statements to determine whether or not they are equal.

Definition 1 (Statements) *Let* $\langle \text{statement}, = \rangle$ *be a structure, where* statement *denotes the set of declarative sentences in some language and* $=$ *is an equality relation, modeled as a function of type* $\text{statement} \times \text{statement} \rightarrow \text{boolean}$.

Next, to support defeasible argumentation and allow the burden of proof to be distributed, we distinguish three kinds of premises.

Definition 2 (Premises) *Let* premise *denote the set of premises. There are three kinds of premises:*

1. *If* s *is a* statement*, then* $\text{premise}(s)$ *is a premise. These are called* ordinary premises*. As a notational convenience, we will use a statement* s *alone to denote* $\text{premise}(s)$*, when the context makes it clear that the statement is being used as a premise.*
2. *If* s *is a* statement*, then* $\bullet s$*, called a* presumption*, is a premise.*
3. *If* s *is a* statement*, then* $\circ s$*, called an* exception*, is a premise.*
4. *Nothing else is a premise.*

Now we are ready to define the structure of arguments.

Definition 3 (Arguments) *An* argument *is a tuple* $\langle c, d, p \rangle$*, where* c *is a* statement*,* $d \in \{\text{pro}, \text{con}\}$ *and* $p \in \mathcal{P}(\text{premise})$*. If* a *is an argument* $\langle c, d, p \rangle$*, then* $\text{conclusion}(a) = c$*,* $\text{direction}(a) = d$ *and* $\text{premises}(a) = p$*. Where convenient, pro arguments will be notated as* $p_1, \ldots, p_n \rightarrow c$ *and con arguments as* $p_1, \ldots, p_n \multimap c$*.*

This approach, with two kinds of arguments, pro and con, is somewhat different than the argument diagramming model developed by Walton in [11] and implemented in Araucaria. There counterarguments are modelled as arguments pro some statement which has been asserted to be in conflict with the conclusion of the other argument, called a *refutation*. Our approach, with its two kinds of arguments, is not uncommon in the literature on defeasible argument [12,13,14,15].

We assume arguments are asserted by the participants of a dialog. We have specified and implemented a simple communication language and argumentation protocol to test Carneades, but that is a subject for another paper. For our purposes here, it is sufficient to note that argument moves, i.e. speech acts, are modelled as functions which map a state of the dialog to another state. (Again, this is a purely functional model, so states are not modified.) A dialog state is a tuple $\langle t, h, G \rangle$, where t is a statement, the *thesis* of the dialog, h is a sequence of moves, representing the history of the dialog, and G is an *argument graph*.[1]

It is these argument graphs which concern us here. An argument graph plays a role comparable to a set of formulas in logic. Whereas in logic the truth of a formula is defined in terms of a (consequence) relation between sets of formulas, here we will define the *acceptability* of statements in argument graphs. An argument graph is not merely a set of arguments. Rather, as its name suggests, it is a finite graph. There are two kinds of

[1] In prior work [16,15], Gordon has referred to argument graphs as *dialectical graphs*.

nodes, statement nodes and argument nodes. The edges of the graph link up the premises and conclusions of the arguments. Each statement is represented by at most one node in the graph.

To illustrate argument graphs, suppose we have the following (construed) arguments from the domain of contract law:

a1. $\text{agreement}, \circ\, \text{minor} \rightarrow \text{contract}$
a2. $\text{oral}, \bullet\, \text{estate} \multimap \text{contract}$
a3. $\text{email} \rightarrow \text{oral}$
a4. $\text{deed} \multimap \text{agreement}$
a5. $\bullet\, \text{deed} \rightarrow \text{estate}$

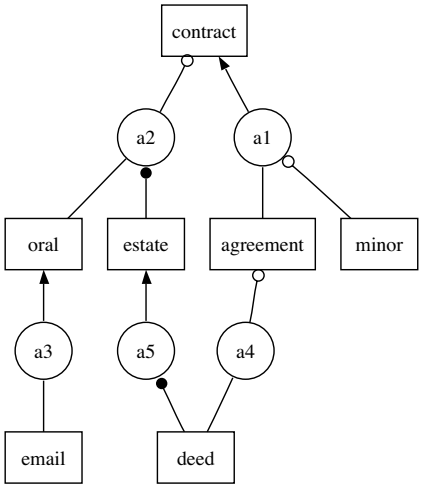


Figure 1. Argument Graph

The argument graph induced by these arguments is shown in Figure 1. In this figure, statements are displayed as boxes and arguments as circles. Different arrowhead shapes are used to distinguish pro and con arguments as well as the three kinds of premises. Pro arguments are indicated using ordinary arrowheads; con arguments with open-dot arrowheads. Ordinary premises are represented as edges with no arrowheads, presumptions with closed-dot arrowheads and exceptions with open-dot arrowheads. (The direction of the edge is implicit in the case of ordinary premises; the direction is always from the premise to the argument.) Notice that the premise type cannot be adequately represented using statement labels, since argument graphs are not restricted to trees. A statement may be used in multiple arguments and as a different type of premise in each argument. The above example illustrates this point. The fourth and the fifth arguments each use the statement 'deed' in a premise. In the fourth argument it is used in an ordinary premise but in the fifth it is used in a presumption. Walton has called this use of shared premises a *divergent argument structure* [11, p. 91].

Although argument graphs are not restricted to trees, they are not completely general; we do not allow cycles. This restriction assures the decidability of the defensibility and acceptabilty properties of arguments and statements, respectively.

Definition 4 (Argument Graphs) *An* argument-graph *is a labeled, finite, directed, acyclic, bipartite graph, consisting of* argument *nodes and* statement *nodes. The edges link the argument nodes to the statements in the premises and conclusion of each argument.*

This completes the formal definition of the structure of arguments and argument graphs. Let us now discuss briefly the expressiveness of this model, beginning by comparing our approach with Toulmin's model [17]. Recall that arguments in Toulmin's model consist of a single premise, called the *datum*; a conclusion, called the *claim*; a kind of rule, called the *warrant*, which supports the inference from the premise to the conclusion of the argument; an additional piece of data, called *backing*, which provides support for the warrant; an exception, called a *rebuttal*; and, finally, a *qualifier* stating the probative value of the inference (e.g. presumably, or necessarily). Of these, the datum and conclusion are handled in a straightforward way in our model. The set of premises of an argument generalizes the single datum in Toulmin's system. Claims are modeled comparably, as conclusions. Rebuttals are modeled with con arguments. The probative weight of an argument is handled as part of our model of proof standards, as will be explained shortly.

This leaves our interpretation of warrants and backing to be explained. Our model does not directly allow arguments about other arguments. (The conclusion of an argument must be a statement.) Rather, the approach we prefer is to add a presumption for the warrant to the premises of an argument. If an argument does not have such a presumption, the argument graph can first be extended to add one. We leave it up to the argumentation protocol of the procedural model to regulate under what conditions such *hidden premises* may be *revealed*. In effect, the datum and warrant are modelled as minor and major premises, much as in the classical theory of syllogism. Backing, in turn, can be modelled as a premise of an argument supporting the warrant.

For example, here is a version of Toulmin's standard example about British citizenship.

Datum. Harry was born in Bermuda.
Claim. Harry is a British subject.
Warrant. A man born in Bermuda will generally be a British subject.
Backing. Civil Code §123 provides that persons born in Bermuda are generally British subjects.
Exception. Harry has become an American citizen.

The argument can be reconstructed in our framework as illustrated in Figure 2.

This approach generalizes Toulmin's model, by supporting arguments pro and contra both warrants and backing, using the same argumentation framework as for arguments about any other kind of claim. Indeed, Toulmin appears to have overlooked the possibility of arguing against warrants or making an issue out of backing claims.

Our model of argument is rich enough to handle Pollock's concepts of rebuttal, premise defeat and undercutting defeaters [12]. Rebuttals can be modeled as arguments in the opposite direction for the same conclusion. (If an argument a_1 is pro some statement s, then some argument a_2 con s is a rebuttal of a_1, and vice versa.) Premise defeat can be modeled with arguments con an ordinary premise or presumption, or pro an exception.

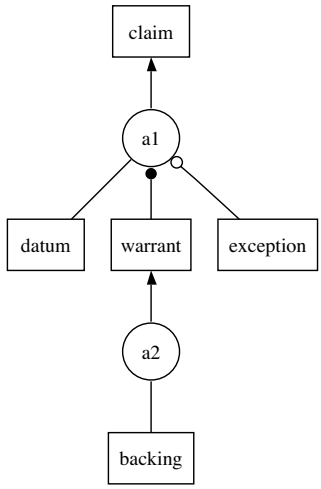


Figure 2. Reconstruction of Toulmin Diagrams

Undercutting defeaters are a bit trickier. The idea of an undercutting defeater is to argue against the argument itself, or the rule or warrant which was applied to create the argument. We model undercutting defeaters by revealing and then attacking premises, similar to the way we handled warrants in the reconstruction of Toulmin's system. Consider Pollock's example of things which look red but turn out to be illuminated by a red light:

Red. The object is red.
Looks Red. The object looks red.
Applicable. The general rule "Things which look red are red." applies to this object.
Illuminated. The object is illuminated by a red light.

An argument graph for this example is shown in Figure 3. Rather than undercutting argument a_1 (the object is red because it looks red) directly, with an argument contra a_1, we undercut the argument by first revealing a presumption (about the general rule being applicable in this case) and then assert an argument contra this presumption. Notice by the way that another presumption is still implicit in this example, namely a presumption for the "warrant" about things which look red being red.

Walton [11] distinguishes two kinds of arguments, called *convergent* and *linked* arguments. Convergent arguments provide multiple reasons for a conclusion, each of which alone can be sufficient to accept the conclusion. Convergent arguments are handled in our approach by multiple arguments for the same conclusion. Linked arguments, on the other hand, consist of two or more premises which all must hold for the argument to provide significant support for its conclusion. Linked arguments are handled in our approach by defining arguments to consist of a set of premises, rather than a single premise, and defining arguments to be defensible only if all of their premises hold. (The concept of argument defensibilty is formally defined below.)

Presumptions and exceptions are a refinement of Walton's concept of *critical questions* [18]. Critical questions enumerate specific ways to defeat arguments matching some

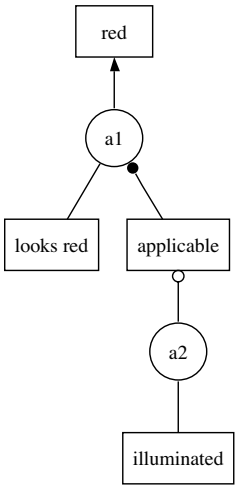


Figure 3. Undercutting Defeater Example

argumentation scheme. But so long as an issue has not been raised by actually asking some critical question, we would like to be able to express which answer to presume. The distinction between presumptions and exceptions here provides this ability.

Consider the scheme for arguments from expert opinion [19]:

Major Premise. Source E is an expert in the subject domain S containing proposition A.
Minor Premise. E asserts that proposition A in domain S is true.
Conclusion. A may plausibly be taken as true.

The scheme includes six critical questions:

CQ1. How credible is E as an expert source?
CQ2. Is E an expert in the field that A is in?
CQ3. Does E's testimony imply A?
CQ4. Is E reliable?
CQ5. Is A consistent with the testimony of other experts?
CQ6. Is A supported by evidence?

When the scheme for arguments from expert opinion is instantiated to create a specific argument, the critical questions can be represented, in our model, as presumptions and exceptions. Whether a presumption or exception is appropriate depends on the burden of proof. If the respondent, the person who poses the critical question, should have the burden of proof, then the critical question should be modeled as an exception. If, on the other hand, the proponent, the party who used the scheme to construct the argument, should have the burden of proof, then the critical question should be modeled as a presumption.[2]

[2]We agree with Verheij [20] that critical questions which are entailed by the premises of the argument scheme are redundant and may be omitted. This is arguably the case in the example for the first three critical questions.

Our model does not require that premises for critical questions be made explicit at the time the argument is first made. Rather, they can be *revealed* incrementally during the course of the dialog. The conditions under which a premise may be left implicit or revealed raise procedural issues which need to be addressed in the protocol for the type of dialog. Our contribution here is to provide an argumentation framework which can be used for modeling such protocols.

3. Argument Evaluation

By argument evaluation we mean determining whether a statement is *acceptable* in an argument graph. As we will see soon, this in turn will depend on the *defensibility* of arguments in the graph. Notice that our terminology is somewhat different than Dung's [9], who speaks of the acceptability of arguments, rather than their defensibility. Also, for those readers familiar with our preliminary work on this subject in [5], please notice that the terminology and other details of the current model are different, even though the basic ideas and general approach are quite similar.

The definition of the acceptability of statements is recursive. The acceptability of a statement depends on its *proof standard*. Whether or not a statement's proof standard is *satisfied* depends on the defensibility of the arguments pro and con this statement. The defensibility of an argument depends on whether or not its premises *hold*. Finally, we end up where we began: whether or not a premise holds can depend on whether or not the premise's statement is acceptable. Since the definitions are recursive, we cannot avoid making forward references to functions which will be defined later.

To evaluate a set of arguments in an argument graph, we require some additional information. Firstly, we need to know the current *status* of each statement in the dialog, i.e. whether it is accepted, rejected, at issue or undisputed. This status information is pragmatic; the status of statements is set by speech acts in the dialog, such as asking a question, asserting an argument or making a decision. Secondly, we assume that a proof standard has been assigned to each statement. We do not address the question of how this is done. Presumably this will depend on domain knowledge and the type of dialog. In the following, let $\{\mathrm{SE}, \mathrm{PE}, \mathrm{DV}, \mathrm{BRD}\}$ be an enumeration of some proof standards. Their meaning will be defined shortly, below. Finally, one of these proof standards, PE, makes use of numerical weights, comparable to conditional probabilities. To use this proof standard, we require a weighing function.

Let us formalize these requirements by postulating an *argument context* as follows.

Definition 5 (Argument Context) *Let $\mathcal{C}$, the argument context, be a tuple $\langle G, \mathrm{status}, \text{proof-standard}, \mathrm{weight}\rangle$, where G is an* argument-graph*,* status *is a function of type* $\mathrm{statement} \rightarrow \{\mathrm{accepted}, \mathrm{rejected}, \mathrm{undisputed}, \mathrm{issue}\}$*,* proof-standard *is a function of type* $\mathrm{statement} \rightarrow \{\mathrm{SE}, \mathrm{PE}, \mathrm{DV}, \mathrm{BRD}\}$ *and* weight *is a function of type* $\mathrm{statement} \times \mathrm{statement} \rightarrow \{0, \ldots, 10\}$.

Intuitively, a statement which has been used in a dialog is initially $\mathrm{undisputed}$. Later in the dialog, an issue can be made out of this statement. Presumably after arguments pro and con have been collected for some period of time, a decision will be taken and the statement will be either $\mathrm{accepted}$ or $\mathrm{rejected}$. The details of how this is done need

not concern us further here. These are matters which need to be addressed fully when modeling protocols for dialogs.

Definition 6 (Acceptability of Statements) *Let* acceptable *be a function of type* $\text{statement} \times \text{argument-graph} \rightarrow \text{boolean}$. *A statement is acceptable in an argument graph if and only if it satisfies its proof standard in the argument graph:* $\text{acceptable}(s, ag) = \text{satisfies}(s, \text{proof-standard}(s), ag)$.

Definition 7 (Satisfaction of Proof Standards) *A proof standard is a function of type* $\text{statement} \times \text{argument-graph} \rightarrow \text{boolean}$. *Let* f *be a proof standard in* $\text{satisfies}(s, f, G) = f(s, G)$.

Four proof standards are defined in this paper. We do not claim these particular proof standards are exhaustive. Others can be defined similarly.

SE. A statement meets this standard iff it is supported by at least one defensible pro argument.

PE. A statement meets this standard iff it is supported by at least one defensible pro argument and its strongest defensible pro argument outweighs its strongest defensible con argument, if any. This standard balances arguments using probative weights.

DV. A statement meets this standard iff it is supported by at least one defensible pro argument and none of its con arguments are defensible.

BRD. A statement meeets this standard iff it is supported by at least one defensible pro argument, all of its pro arguments are defensible and none of its con arguments are defensible.

The names of three of these standards are meant to suggest three legal proof standards: scintilla of evidence, preponderance of the evidence and beyond a reasonable doubt. However, we do not claim that the definitions of these standards, above, fully capture their legal meanings. What these standards have in common with their legal counterparts is their relative strength: $\text{BRD} > \text{DV} > \text{PE} > \text{SE}$. If a statement satisfies a proof standard, it will also satisfy all weaker proof standards.

The name of the DV proof standard is an acronym for *dialectical validity*, a term used by Freeman and Farley [1]. They defined five proof standards. In addition to the four we have defined here, they included a fifth, called *beyond a doubt*, which was defined to be an even stronger standard than *beyond a reasonable doubt*.

The preponderance of evidence (PE) standard compares the weight of arguments. The weight of an argument is defined to be the same as the weight of its *weakest premise*, i.e., to be precise, the same as the weight of the premise with the lowest weight. Recall we assume a weighing function, weight, as part of the context to provide this information. The weight of a premise p for a conclusion c is $\text{weight}(p, c)$. Other proof standards which aggregate and compare weights are conceivable. For example, one could sum the weights of the arguments pro and con and compare these sums.

We have defined weights to be natural numbers in the range of 0 to 10. We originally considered using real numbers in the range of 0.0 to 1.0, as in probability theory. However, on the assumption that the weights will be estimated by human users, we prefer to use a simpler ordinal scale, since we are skeptical that users can estimate such weights with a greater degree of accuracy.

All of the proof standards defined above depend on a determination of the *defensibility* of arguments. Defensibility is defined next.

Definition 8 (Defensibility of Arguments) *Let* $\mathrm{defensible}$ *be a function of type* $\mathrm{argument} \times \mathrm{argument\text{-}graph} \rightarrow \mathrm{boolean}$. *An argument* α *is defensible in an argument graph* G *if and only if all of its premises* hold *in the argument graph:* $\mathrm{defensible}(\alpha, G) = \mathrm{all}(\lambda p.\, \mathrm{holds}(p, G))(\mathrm{premises}\, \alpha)$.[3]

Finally, we come to the last definition required for evaluating arguments, for the holds predicate. This is where the status of a statement in the argument context and the distinction between ordinary premises, presumptions and exceptions come into play. Accepted presumptions and ordinary premises hold. Rejected presumptions and ordinary premises do not hold. Undisputed presumptions hold. Undisputed ordinary premises do not hold. An exception, $\circ s$, holds only if $\mathrm{premise}(s)$ does not hold.

Definition 9 (Holding of Premises) *Let* holds *be a function of type* $\mathrm{premise} \times \mathrm{argument\text{-}graph} \rightarrow \mathrm{boolean}$. *Let* $\sigma = \mathrm{status}(s)$. *Whether or not a premise holds depends on its type (ordinary, presumption, or exception). Thus, there are the following three cases:*

If p *is an ordinary premise,* $\mathrm{premise}(s)$*, then*

$$\mathrm{holds}(p, G) = \begin{cases} true & \text{if } \sigma = \mathrm{accepted} \\ false & \text{if } \sigma = \mathrm{rejected} \\ acceptable(s, G) & \text{if } \sigma = \mathrm{issue} \\ false & \text{if } \sigma = \mathrm{undisputed} \end{cases}$$

If p *is a presumption,* $\bullet s$*, then*

$$\mathrm{holds}(p, G) = \begin{cases} true & \text{if } \sigma = \mathrm{accepted} \\ false & \text{if } \sigma = \mathrm{rejected} \\ acceptable(s, G) & \text{if } \sigma = \mathrm{issue} \\ true & \text{if } \sigma = \mathrm{undisputed} \end{cases}$$

Finally, if p *is an exception,* $\circ s$*, then*

$$\mathrm{holds}(p, G) = \neg\, \mathrm{holds}(\mathrm{premise}(s), G)$$

The important thing to notice is that whether or not a premise holds depends in this model not only on the arguments which have been asserted, but also on the kind of premise (ordinary, presumption, or exception) and the status of the premise's statement in the argument graph (undisputed, at issue, accepted, or rejected). We assume that the status of a statement progresses in the course of the dialog:

[3]Here 'all' is a higher-order function, not a quantifier, applied to an anonymous function, represented with λ, as in lambda calculus.

1. Initially, statements used in arguments are undisputed. Whether or not a premise which uses this statement holds at this stage of the dialog depends on the kind of premise. Ordinary premises do not hold; presumptions do hold. *This is the only semantic difference between ordinary premises and presumptions in our model.* An exception holds at this stage only if it would not hold if it were an *ordinary premise*. Notice that exceptions are not the dual of presumptions. As undisputed presumptions hold, an undisputed exception would not hold if we had defined exceptions to hold only if they would not hold if they were presumptions. But this is not the semantics we want. Rather, both undisputed exceptions and undisputed presumptions hold.
2. At some point a participant may make an issue out of a statement. Now ordinary premises and presumptions which use this statement hold only if they are acceptable, i.e. only if the statement meets its proof standard, given the arguments which have been asserted. Exceptions at issue hold only if the statement is not acceptable. We presume that arguments will be exchanged in a dialog for some period of time, and that during this phase the acceptability of statements at issue will be in flux.
3. Finally, at some point a decision will be made to either accept or reject some statement at issue. The model does not constrain the discretion of users to decide as they please. Unacceptable statements may be accepted and acceptable statements may be rejected. This remains transparent however. Any interested person can check whether the decisions are justified given the arguments made and the applicable proof standards. Anway, after a decision has been made, it is respected by the model: Accepted statements hold and rejected statements do not hold, no matter what arguments have been made or what proof standards apply.

4. Conclusion

The Carneades Argumentation Framework is a formal, mathematical model of argument evaluation which applies proof standards to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis. The main original contribution of Carneades is its use of three kinds of premises (ordinary premises, presumptions and exceptions) and information about the dialectical status of statements (undisputed, at issue, accepted or rejected) to model critical questions in such a way as to allow the burden of proof to be allocated to the proponent or the respondent, as appropriate. Both of these elements are required for this purpose: presumptions hold without supporting arguments only so long as they have not been put at issue by actually asking the critical question.

The Carneades Argumentation Framework is a semantic model of argumentation, not a calculus. However, since this semantic model is formulated in terms of computable functions, the lambda calculus may be used as a formal system for deriving inferences and functional programming languages can be used to implement the model in software. Indeed, the formal model has been fully implemented in this way, using the Scheme programming language [21].[4] As a semantic model, the question of the soundness or completeness of Carneades does not arise. Rather, the relevant question concerns the validity

[4] An earlier version of Carneades was implemented in Standard ML [22].

of the semantic model. Are these semantics sufficient for providing the kind of argumentation support required by our intended application scenarios? This question cannot be answered by purely formal means, but rather requires experiments with realistic test cases. We have already tested Carneades on a number of examples from the Artificial Intelligence and Law literature, thus far yielding intuitively acceptable results. In a companion paper, also in this volume [23], Carneades has been used to model and evaluate the arguments in the majority and dissenting opinions of Pierson vs. Post, an important benchmark in the Artificial Intelligence and Law field. More work is required to validate the models of the various proof standards, in particular the model of prepondernance of the evidence, which uses weights. For this purpose, we plan to reconstruct examples of reasoning with evidence. Although our test cases thus far have all been legal, Carneades is intended to be a general model of argumentation, not restricted to some application domain. Outside the legal context, we plan to evaluate the suitability of Carneades for practical reasoning in deliberation dialogs.

When completed, Carneades will support a range of argumentation use cases, including argument construction, evaluation and visualization. Although the focus of this paper is argument evaluation, it contains some initial ideas on argument visualization. One of our next tasks will be to refine the diagramming method used here to illustrate the argumentation framework.

References

[1] K. Freeman and A. M. Farley. A model of argumentation and its application to legal reasoning. *Artificial Intelligence and Law*, 4(3-4):163–197, 1996.

[2] A. C. Hastings. *A Reformulation of the Modes of Reasoning in Argumentation*. PhD thesis, Northwestern University, Evanston, Illinois, 1963.

[3] D. Walton. *Appeal to Expert Opinion*. Penn State Press, University Park, 1997.

[4] F. Bex, H. Prakken, C. Reed, and D. Walton. Towards a formal account of reasoning with evidence: Argumentation schemes and generalizations. *Artificial Intelligence and Law*, 11(2-3), 2003.

[5] T. F. Gordon. A computational model of argument for legal reasoning support systems. In P. E. Dunne and T. Bench-Capon, editors, *Argumentation in Artificial Intelligence and Law*, IAAIL Workshop Series, pages 53–64. Wolf Legal Publishers, 2005.

[6] D. Walton and T. F. Gordon. Critical questions in computational models of legal argument. In P. E. Dunne and T. Bench-Capon, editors, *Argumentation in Artificial Intelligence and Law*, IAAIL Workshop Series, pages 103–111, Nijmegen, The Netherlands, 2005. Wolf Legal Publishers.

[7] P. Edwards. *The Encyclopedia of Philosophy*, volume 1. Macmillian and Free Press, 1972.

[8] H. Prakken. From logic to dialectic in legal argument. In *Proceedings of the Fifth International Conference on Artificial Intelligence and Law*, pages 165–174. Maryland, 1995.

[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(2):321–357, 1995.

[10] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):34–43, May 2001.

[11] D. Walton. *Argument Structure : a Pragmatic Theory*. Toronto studies in philosophy. University of Toronto Press, Toronto ; Buffalo, 1996. Douglas Walton. ill. ; 24 cm.

[12] J. L. Pollock. How to reason defeasibly. *Artificial Intelligence*, 57:1–42, 1992.

[13] B. Verheij. *Rules, Reasons, Arguments. Formal Studies of Argumentation and Defeat*. Ph.d., Universiteit Maastricht, 1996.
[14] J. Hage. A theory of legal reasoning and a logic to match. *Artificial Intelligence and Law*, 4(3-4):199–273, 1996.
[15] T. F. Gordon and N. Karacapilidis. The Zeno argumentation framework. In *Proceedings of the Sixth International Conference on Artificial Intelligence and Law*, pages 10–18. Melbourne, Australia, 1997.
[16] T. F. Gordon. The Pleadings Game — an exercise in computational dialectics. *Artificial Intelligence and Law*, 2(4):239–292, 1994.
[17] S. E. Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.
[18] D. Walton. *Argumentation Schemes for Presumptive Reasoning*. Erlbaum, 1996.
[19] D. Walton. *Argumentation Methods for Artificial Intelligence in Law*. Springer, 2005.
[20] B. Verheij. *Virtual Arguments*. TMC Asser Press, The Hague, 2005.
[21] R. Kelsey, W. Clinger, and J. Rees. Revised (5) report on the algorithmic language Scheme. *Higher-Order and Symbolic Computation*, 11(1):7–105, August 1998.
[22] R. Milner, M. Tofte, R. Harper, and D. MacQueen. *The Definition of Standard ML, (Revised)*. MIT Press, 1997.
[23] T. F. Gordon and D. Walton. Pierson vs. Post revisted — a reconstruction using the Carneades Argumentation Framework. In P. E. Dunne and T. Bench-Capon, editors, *Proceedings of the First International Conference on Computational Models of Argument (COMMA 06)*, Liverpool, 2006. IOS Press.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Pierson vs. Post Revisited

A Reconstruction using the Carneades Argumentation Framework

Thomas F. Gordon [a,*] and Douglas Walton [b]
[a] *Fraunhofer FOKUS, Berlin*
[b] *Dept. of Philosophy, University of Winnipeg, Manitoba, Canada*

Abstract. The Pierson vs. Post case [1] has become an important benchmark in the field of AI and Law for computational models of argumentation. In [2], Bench-Capon used Pierson vs. Post to motivate the use of values and value preferences in his theory-construction account of legal argument. And in a more a recent paper by Atkinson, Bench-Capon and McBurney [3], it was used to illustrate a formalization of an argumentation scheme for practical reasoning. Here we offer yet another reconstruction of Pierson vs. Post, using our Carneades Argumentation Framework, a formal mathematical model of argument structure and evaluation based on Walton's theory of argumentation [4], and compare it to this prior work. Carneades, named in honor of the Greek skeptic philosopher who emphasized the importance of plausible reasoning, applies proof standards [5] to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis.

Keywords. Legal Argument, Carneades Argumentation Framework, Pierson vs. Post

1. Introduction

The Pierson vs. Post case [1] has become an important benchmark in the field of AI and Law for computational models of argumentation. Pierson vs. Post is a classic property law case, widely used in legal education. Don Berman and Carole Hafner were the first in the Artificial Intelligence and Law community, to our knowledge, to use Pierson vs. Post, and related well-known property cases, as part of their research on the role of teleological reasoning in the law [6,7]. In a special issue of the Artificial Intelligence and Law Journal, in memory of Donald Berman, several articles presented models of teleological reasoning using the Pierson vs. Post case as a benchmark, including a paper by Bench-Capon [2]. Since then, Bench-Capon and his colleagues have continually made use of Pierson vs. Post as a testbed for their research on legal argumentation, including the 2005 ICAIL paper with Atkinson and McBurney [3].

In this paper, we offer yet another reconstruction of Pierson vs. Post, as a further step towards validating our Carneades Argumentation Framework. (Carneades is presented in a companion paper in this volume, [8]. We highly recommend reading this other paper first.)

*Correspondence to: Dr Thomas F. Gordon, Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, Berlin, Germany. E-mail: thomas.gordon@fokus.fraunhofer.de

The rest of the paper is structured as follows. Section 2 presents our reconstruction of Pierson vs. Post, using Carneades. Section 3 compares this reconstruction with the one of Atkinson, Bench-Capon and McBurney [3]. The paper closes in Section 4 with a recapitulation of its main points and some problems for future research.

2. Reconstructing Pierson vs Post

The full text of the opinion of the Supreme Court of New York in Pierson vs. Post is available on the web.[1] To help evaluate the naturalness of the Carneades model, the reconstruction here will attempt to model the arguments in the opinion in the order they were presented by the court. Of course, as in all models, many details will be abstracted away. In the interest of saving space, we will not quote the entire text of the opinion here, but instead refer the reader to the online version mentioned above.

To help make this presentation more comprehensible and self-contained, let us first summarize the basic facts of the case and present the main issue before the court. Pierson and Post were both fox hunting "upon a certain wild and uninhibited, unpossessed and waste land". Post was persuing the fox, with his dogs, when Pierson intervened, shot and killed the fox and carried it off. Post sued and won. Pierson appealed. More precisely, Pierson petitioned the Supreme Court of New York for a so-called "writ of certiorari". In this appelate proceeding, somewhat confusingly, Pierson is named as the plaintiff. As we will see, the main issue before the court is whether Post's pursuit of the fox with his dogs should be deemed sufficient, as a matter of law, to acquire possession of the fox, and thus obtain property rights to the fox. Justice Tompkins, writing for the majority, argues that such a pursuit is not sufficient, by interpreting treatises and precedents narrowly and making several policy ("teleological") arguments: legal certainty, avoiding a "fertile source of quarrels and litigation" and preserving "peace and order in society". Justice Livingston, in his dissent, tries unsuccessfully to broaden existing doctrine. He proposes a rule which would deem a wild, "noxious" animal to be "mortally wounded", and thus according to existing doctrine in the possession of the hunter, if the animal is being chased, as in this case, by large hunting dogs. The argument asserted by Justice Livingston to support this proposed rule is also teleolgical: to protect farmers by encouraging and rewarding the hunting of noxious wild animals.

Now, let us begin to model Justice Tompkins' arguments.

He starts off by stating the main issue to be decided:

> The question ... is, whether ... Post, by the pursuit with his hounds ... acquired ... property in, the fox ...

The main issue is quickly reduced to the question of possession ("occupancy"), with the following argument:

> It is admitted that a fox is an animal ferae naturae, and that property in such animals is acquired by occupancy only. These admissions narrow the discussion to the simple question of what acts as occupancy ...?

This argument can be reconstructed as shown in Figure 1. The statements which have been accepted are shown with a gray background. Statements which are acceptable

[1] http://www.saucyintruder.org/pages/pierson.html

given the arguments, as evaluated by Carneades, will also be visualized with a gray background, unless they have been rejected.[2]

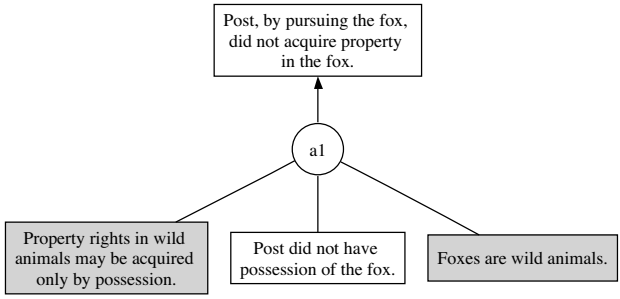


Figure 1. Argument 1

Next, Justice Tompkins argues the pursuit alone is not sufficient to constitute possession, by reference to several treatises, i.e. jurisprudential works by academic lawyers:

> Justinian's Institutes, lib. 2, tit. 1, s.13, and Fleta, lib. 3, c.2, p. 175, adopt the principle, that pursuit alone vests no property or right in the huntsman; and that even pursuit, accompanied with wounding, is equally ineffectual for that purpose, unless the animal be actually taken. The same principle is recognized by Bracton, lib. 2, c.1, p. 8.

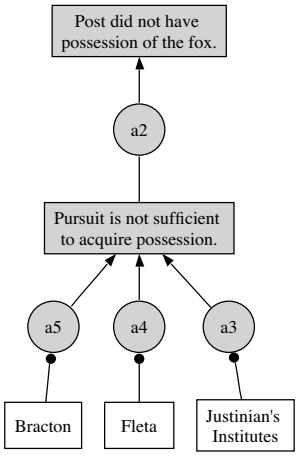


Figure 2. Argument 2

The three treatises have been modelled as presumptions, as shown in Figure 2, both because they were accepted implicitly without argument and to illustrate this feature of Carneades. Alternatively they could have been modelled as ordinary premises, but then

[2]Thus this simple visualization method does not distinguish statements which have been accepted but are not acceptable or statements which are acceptable but have been rejected, even though the underlying Carneades formal model does support these distinctions.

it would have been necessary to accept them, for the arguments to "go through", even though they were not explicitly accepted in the reported opinion.

The DV proof standard ("dialectical validity") will be used throughout this reconstruction. Other proof standards will be considered later, in the discussion section.

Notice that this argument is already sufficient to support the statement that Post did not have possession of the fox, which is now acceptable and hence displayed with a gray background. The gray backgrounds of four arguments shown in Figure 2 indicate they are currently defensible.

Justice Tompkins is not content to leave it at that. Even though he now has a defensible argument for the decision of the court, that Post did not, by pursuing the fox, acquire property in the fox, he provides a further argument, by asserting that actual, physical ("corporal") possession of the fox is required:

> Puffendorf, lib. 4, c.6, s.2, and 10, defines occupancy of beasts ferae naturae, to be the actual corporal possession of them, and Bynkershoek is cited as coinciding in this definition.

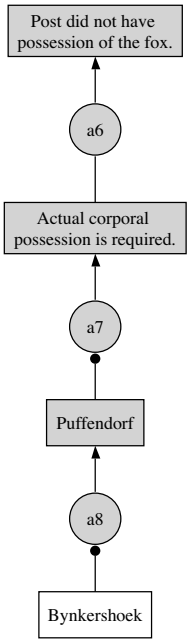


Figure 3. Argument 6

The reconstruction of this additional argument is shown in Figure 3. Notice again that references to treatises have been modelled as presumptions. Thus, Justice Tompkins' reference to Bynkershoek's treatise was not strictly necessary; citing Puffendof alone would have sufficed. Either he was closing off a potential avenue of attack or just wanted to drive this point home with additional support.

Next, Justice Tompkins discusses another treatise, by Barbeyrac, which takes the position that possession may be deemed in certain circumstances, without requiring actual physical possession, in particular when the animal has been "mortally wounded":

> Barbeyrac, in his notes on Puffendorf ... affirms, that actual bodily seizure is not, in all cases, necessary to constitute possession of wild animals. ... the mortal wounding of such beasts, ... may ... be deemed possession ... Barbeyrac seems to have adopted .... the more accurate opinion of Grotius ...

Justice Tompkins dismisses this argument by simply suggesting that none of these counterarguments apply in this case.

> The case now under consideration is one of mere pursuit, and presents no circumstances or acts which can bright it within the definition of occupancy by Puffendorf, or Grotius, or the ideas of Barbeyrac upon that subject.

The recontruction of these arguments is shown in Figure 4. We've restricted our attention to the exception for mortally wounded animals, since, as we will see, this is the exception that Justice Livingston uses in his dissent. Notice that Barbeyrac's position has been modelled as a rebuttal, i.e. con argument. Justice Tompkins' argument against the conclusion that the fox was mortally wounded could have been modelled explicitly in Carneades as a con argument. We have not done so, however, since the premise of this argument would simply have been the denial of the conclusion. Rather, we interpret Justice Tompkins instead challenging Justice Livingston to provide arguments supporting this conclusion. Indeed, as we will see soon, Justice Livingston accepted this invitation.

Notice that the conclusion of the argument shown in Figure 4, that "actual corporal possession is required", is not acceptable in the diagram, even though Justice Tompkins has argued that such possession is required. This is only because this part of Justice Tompkins' argument is modeled here in isolation, rather than integrated with his prior argument, as shown previously in Figure 3.

We are nearing the conclusion of Justice Tompkins' opinion for the majority. He next distinguishes a precedent case, Keeble vs. Hickergill (referred to as 11 Mod. 74-130 in the quotation below), which deems the owner of property to be in possession of wild animals on his property, at least if the owner hunts these animals for a living. Interestingly, Keeble is the only precedent cited in the entire opinion, by either Justice Tompkins or Justice Livingston. Pierson vs. Post may be an atypical U.S. appellate court opinion.[3]

> The case cited from 11 Mod. 74-130, I think clearly distinguishable from the present; inasmuch as there the action was for maliciously hindering and disturbing the plaintiff in the exercise and enjoyment of a private franchise; and ... the ducks were in the plaintiff's decoy pond, and so in his possession ...

Figure 5 shows our reconstruction of this argument. It is much like the prior argument about pursuit being sufficient if the animal had been mortally wounded. Justice Tompkins accepts the major premise or, to use Toulmin's term, the warrant of the counterargument, but simply denies that the minor premise is satisfied by the facts of this case. No arguments have been asserted supporting the premise that Post was pursuing his livelihood on his own land.

[3]Pierson vs. Post appears as one of a series of cases in the property law case books used in legal education. Prior work on case-based reasoning in AI and Law has used the whole series as a test bed. The goal typically has not been so much to model the reasoning of the court in Pierson vs. Post, as to understood how precedent cases may be used to generate arguments to help resolve issues in further cases.

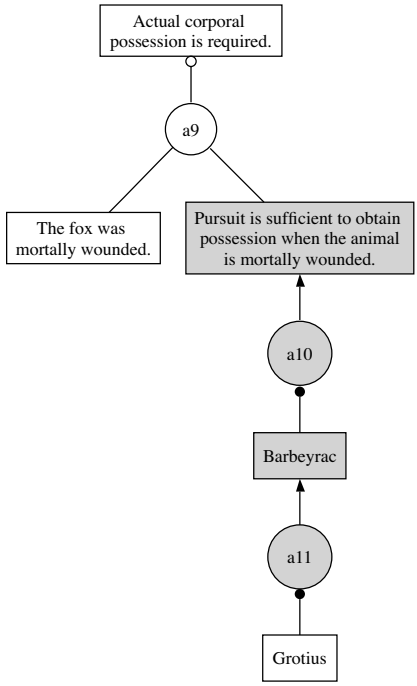


Figure 4. Argument 9

Justice Tompkins' final argument is "teleological", i.e. a policy argument about social values:

> We are the more readily inclined to confine possession or occupancy of beasts ferae naturae, within the limits prescribed by the learned authors above cited, for the sake of certainty, and preserving peace and order in society. If the first seeing, starting, or pursuing such animals, without having so wounded, circumvented or ensnared them, so as to deprive them of their natural liberty, and subject them to the control of their pursuer, should afford the basis of actions against others for intercepting and killing them, it would prove a fertile source of quarrels and litigation.

Our reconstrution of this teleological argument is displayed in Figure 6. Both of the premises of this argument have been accepted in Justice Tompkins's majority opinion; Hence the background of these premises is gray in the figure.

As Atkinson, Bench-Capon and McBurney point out [3], such teleological arguments can be viewed as instances of an argument scheme for practical reasoning. Here one premise states some action or policy would have some effect ("A bright-line rule creates legal certainty, preserving peace and order.") and the other premise states this effect is desirable, as it satisfies some goal or promotes some value ("Peace and order is an important social value.").

Depending on the formulation of the argumentation scheme for practical reasoning used, some of the premises of the argumentation scheme might not be instantiated in this particular instance of the scheme. But recall that not all premises of a scheme need to be made explicit in an argument. Arguments with implicit premises are called "en-

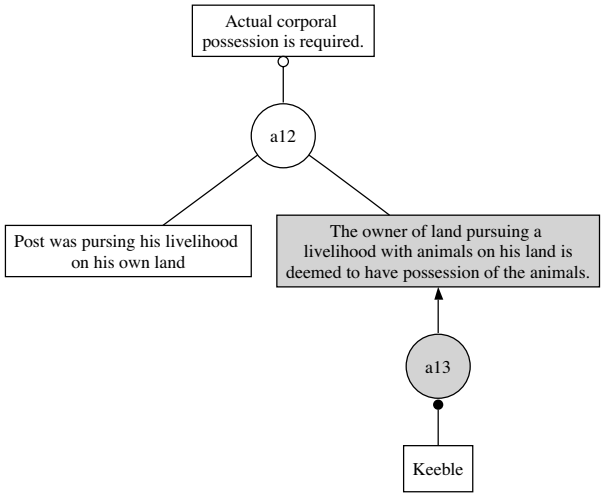


Figure 5. Argument 12

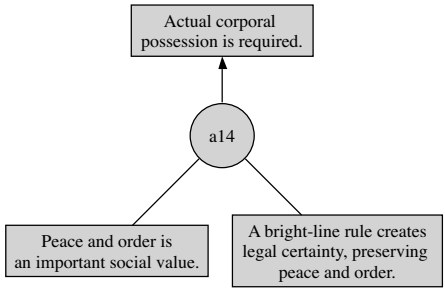


Figure 6. Argument 14

thymemes". These implicit premises can be revealed during the course of the dialog, for example by asking critical questions. In the example, one premise left implicit states that the conclusion of the argument ("Actual corporal possession is required") is indeed a bright-line rule.

Justice Tompkins concludes his majority opinion by stating the judgment of the court:

> However uncourteous or unkind the conduct of Pierson towards Post, in this instance, may have been, yet his act was productive of no injury or damage from which a legal remedy can be applied. We are of opinion the judgment below was erroneous, and ought to be reversed.

Notice that this judgment does not explictly state that Post did not acquire a property right in the fox. But surely this is what it means, since the opinion of the lower court is reversed. This text can also be understood as making an additional argument, rejecting the idea that Post's unsportmanlike conduct provides grounds for a legal remedy, but we have not modeled this argument in our reconstruction.

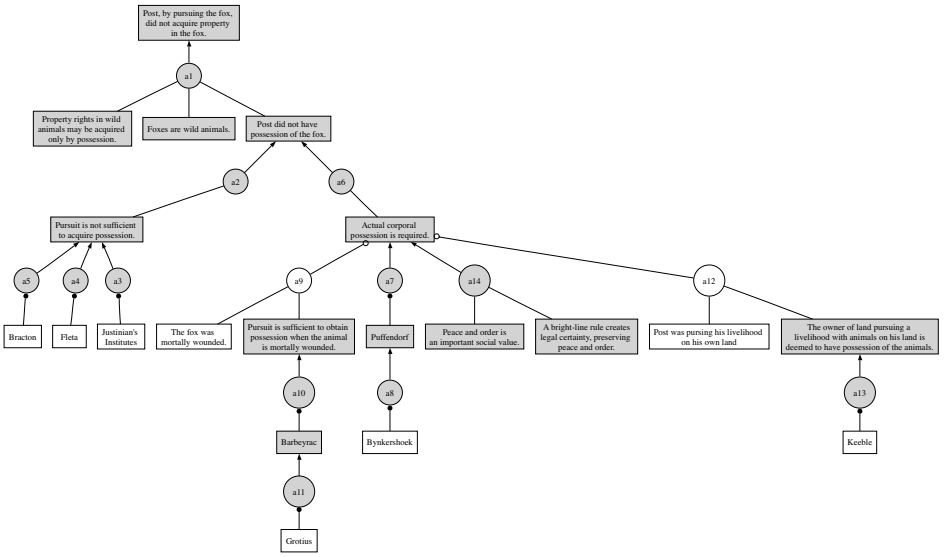


Figure 7. Tompkins' Opinion, for the Majority

Figure 7 integrates all of the arguments in our reconstruction of Justice Tompkins' opinion into a single argument graph. Here we can easily see that the judgment of the court is acceptable given the arguments in the opinion, at least in our formal reconstruction using the Carneades Argumentation Framework. Of course, this does not mean that the decision is necessarily correct or beyond criticism. It only means the court's reasoning can be understood using Carneades.

Let us now turn our attention to Justice Livingston's dissenting opinion. He focuses on a single issue, whether or not a fox should be considered mortally wounded if it is being pursued by a number of large dogs, and thus, following the position of Barbeyrac, accepted by the majority, be considered to be in possession of the hunter whose dogs are doing the pursuing. Justice Livingston asserts three arguments; the first argument refers to the pleadings to support the proposition that foxes are noxious beasts. The second argument proposes a rule, deeming noxious animals persued by large hounds to be mortally wounded, and supports this rule with an appeal to policy, by arguing that such a rule would encourage and reward hunting, thus protecting farmers. That is, Justice Livingston too applies an argumentation scheme for practical reasoning to make a telelogical argument. The third argument, less explicit in the text of the opinion, applies this proposed rule to the accepted facts of the case to reach the conclusion that the fox was mortally wounded, resolving the issue.

> ... By the pleadings it is admitted that a fox is a "wild and noxious beast" His depredations on farmers and on barn yards have not been forgotten; and to put him to death wherever found, is allowed to be meritorious, and of public benefit. Hence it follows, that our decision should have in view the greatest possible encouragement to the destruction of an animal ... But who would keep a pack of hounds; or what gentlemen, at the sound of the horn, and at peep of day, would mount his steed, and for hours together, "sub jove frigido" or a vertical sun, pursue the windings of this wily quadruped, if, just as night came on, and his stratagems and strength were nearly

exhausted, a saucy intruder, who had not shared in the honours or labours of the chase, were permitted to come in at the death, and bear away in triumph the object of pursuit?
... After mature deliberation, I embrace that of Barbeyrac ... If at liberty, we might imitate the courtesy of a certain emperor, who ... ordained, that if a beast be followed with large dogs and hounds, he shall belong to the hunter, not to the chance occupant; and in like manner, if he be killed or wounded with a lance or sword; but if chased with beagles only, then he passed to the captor, not to the first pursuer. ...
...a pursuit like the present ... must inevitably ... terminate in corporal possession ...

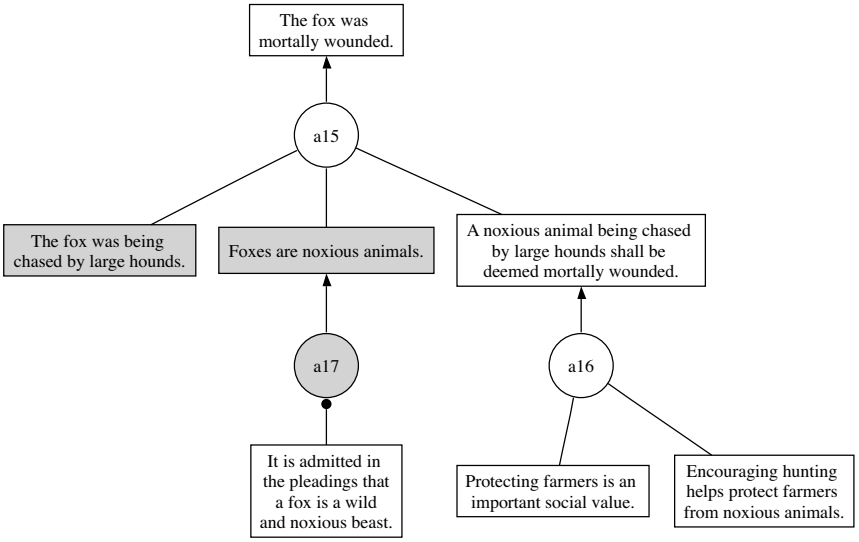


Figure 8. Livingston's Dissenting Opinion

Figure 8 visualizes our reconstruction in Carneades of the core arguments of Justice Livingston's dissent. As expected, since this is the dissenting opinion, his argument fails. The reason is simple: Justice Livingston's proposed rule was not accepted by the majority. Indeed, there is no indication in the published opinion that the majority accepts the premises of Justice Livingston's policy argument, about the importance of encouraging hunting so as to protect farmers. Justice Tompkins does not even mention this argument, let alone respond to it.

3. Discussion

Let us now consider whether our reconstruction of Pierson vs. Post in Carneades can shed any light on some prior models of legal argumentation, which also made use of this case, in particular work by Bench-Capon in [2] as well the work by Atkinson, Bench-Capon and McBurney in [3].

In [2], Bench-Capon's primary concern is to analyse the role of telelogical reasoning in legal argument, motivated by the seminal paper by Berman and Hafner [6], which

identified limitations of the HYPO approach to case-based reasoning in the law [9]. Bench-Capon's central idea in [2] is that the rules and rule preferences cannot be derived solely from factors in precedent cases, but must also be informed by the purposes of the rules, i.e. by the values promoted by the rules. Shortly thereafter, Bench-Capon, in collaboration with Sartor, developed this basic idea into a theory-construction model of legal argument [10]. In this model, legal theories are constructed from precedent cases in a process which takes values and value preferences into consideration to derive and order rules, which may then be applied to the facts of cases to reach decisions.

The paper by Atkinson, Bench-Capon and McBurney [3] views legal reasoning as a kind of practial reasoning, following [11], and illustrates this view using Pierson vs. Post. Towards this end, an argumentation scheme for practical reasoning is developed and applied to model a simulated dialog among four agents, based on the facts and arguments in Pierson vs. Post.

In our view, each of these papers used Pierson vs. Post to illustrate computational models of particular argument schemes, rather than attempting to provide a general framework which can accomodate all the argumentation schemes actually used in the case.

In [12], many examples are presented illustrating the rich variety of argumentation schemes used in legal argumentation. Although the Carneades Argumentation Framework does not yet include a formal model of argumentation schemes, we can nonetheless attempt to manually identify some of the argumentation schemes applied in Pierson vs. Post. Seven of the arguments in Pierson are arguments from authority or perhaps expert opinion, i.e. from legal treatises written by jurisprundential scholars (Braction, Fleta, Justinian's Insitutes, Barbeyrac, Grotius, Puffendorf and Bynkershoek). Interestingly, there is but a single argument from legal precedent (Keeble) and only only two arguments, in our opinion, may be understood as instances of an argumentation scheme for practical reasoning.

Let us now discuss the schemes used in some of the arguments. In the Pierson vs. Post case it says "if we have recourse to the ancient writers on general principles of law", and then it talks about sources like Justinian's Institutes as having adopted the principle that pursuit alone vests no property or right in the huntsman. The problem is to judge what type of argumentation scheme this represents. It looks like it could be argument from precedent, assuming these judgments represent previous rulings of courts. On the other hand, it uses the expression "ancient writers", suggesting an appeal to authority. This may suggest that the argumentation scheme is that for the argument from expert opinion. However, it has been recognized in the argumentation literature that there are different types of appeal to authority. One is appeal to expert opinion, but it has also been recognized that there is a species of appeal to institutional or judicial authority of a kind that is different from appeal to expert opinion, although related to it (argumentum ad judicium). In [13, p. 76] it is shown how a distinction can be drawn between two meanings of the expression 'appeal to authority'. One meaning refers to an authority who has expertise in a domain of knowledge or skill. This type is sometimes called cognitive authority. Another important meaning of 'authority' refers to what is often called administrative authority: "the right to exercise command over others, or to make rulings binding on others through an invested office or recognized position of power". Both kinds of authority are clearly very important in law, where they may even be combined.

The term "principle" is used in the court opinion to describe the statement that pursuit alone vests no property or right in the huntsman. This seems to suggest that the source cited has adopted what amounts to a general rule. But it is hard to decide whether this "principle" is being cited as a rule of law that was previously accepted, or as a generalization stated or implied in the writings of a legal authority.

One reference is to Puffendorf's cited definition, defining occupancy of wild beasts to be the actual corporal possession of them. Another source, Bynkershoek, is cited as agreeing with this definition. This move is interesting because it cites the argument from authority in an unusual way as supporting a definition. Thus the move combines argument from authority with argument from a definition, both well-known argumentation schemes.

Justice Tompkins writes that Barbeyac, in his notes on Puffendorf, does not agree with the latter's definition of occupancy, and advocates a different one. Justice Tompkins appears to be examining the two texts side by side, identifying the differences between them. He even examines the objections of the one to the definitions and principles of the other. This is a typical case of argumentation where a third party is examining the writings of two previous parties, identifying their views, checking the points at which they appear to be inconsistent, and even pitting the arguments of one against those of the objections, replies and counter-arguments of the other. This paragraph is fascinating, because it illustrates an application scenario for computational models of argument. One can imagine a future judge using such models to reconstruct the pro and con arguments of jurisprudential authorities. Moreover, it would appear that what is going on here is not just a simple case of appeal to authority. Rather, Justice Tompkins is critically analyzing the authorities, trying to reveal weaknesses in their arguments.

Over and above all these factors, Pierson vs. Post is a brilliant illustration of how argumentation in a legal case can turn on trying to find or apply exceptions to defeasible rules, as Justice Livingston does here when he tries to apply the exception identified by Barbeyrac to the requirement of physical possession, for animals which have been mortally wounded.

4. Conclusions

The Carneades Argumentation Framework is a formal, mathematical model of argument evaluation which applies proof standards to determine the defensibility of arguments and the acceptability of statements on an issue-by-issue basis. The formal model has been fully implemented, in the Carneades system, and tested on a number of examples from the Artificial Intelligence and Law literature, thus far yielding, we claim, intuitively acceptable results. This validation work is continuing.

The focus of this paper has been our attempt to reconstruct the actual arguments in the majority and dissenting opinions of the Pierson vs. Post case, which has become something of a benchmark in the AI and Law field. We feel this attempt has been successful. Using Carneades, we have been able in our model both to capture the structure of the arguments, at a high level of abstraction, and to evaluate these arguments automatically. The result of this evaluation is compatible with the decision of the court; the judgment of the court appears acceptable given the arguments in the opinion. This does not mean that the decision is necessarily correct or beyond criticism. On the contrary,

the model, and also its visualization, helps us to understand the arguments in the opinion and to reveal their weaknesses.

Pierson vs. Post has been used in the AI and Law field to illustrate computational models of particular argument schemes. But legal argumentation has in common with argumentation in general the application of a large variety of argumentation schemes. Our reconstruction of Pierson vs. Post in Carneades illustrates how a variety of argumentation schemes can be used together in a single case.

Not all features of Carneades could be evaluated by reconstructing the arguments in Pierson vs. Post. For example, more work is required to validate the models of the various proof standards, in particular the model of prepondernance of the evidence, which uses weights. For this purpose, we plan to reconstruct examples of legal reasoning with evidence.

References

[1] Pierson vs. Post. 3 Cai R 175 2 Am Dec 264 (Supreme Court of New York), 1805.

[2] T. Bench-Capon. The missing link revisted: The role of teleology in representing legal argument. *Artificial Intelligence and Law*, 10(1-3):79–94, September 2002.

[3] K. Atkinson, T. Bench-Capon, and P. McBurney. Arguing about cases as practical reasoning. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, pages 35–44, Bologna, Italy, 2005.

[4] D. Walton. *Argument Structure : a Pragmatic Theory*. Toronto studies in philosophy. University of Toronto Press, Toronto ; Buffalo, 1996. Douglas Walton. ill. ; 24 cm.

[5] K. Freeman and A. M. Farley. A model of argumentation and its application to legal reasoning. *Artificial Intelligence and Law*, 4(3-4):163–197, 1996.

[6] D. H. Berman and C. D. Hafner. Representing teleological structure in case-based legal reasoning: The missing link. In *Proceedings of the 4th International Conference on Artificial Intelligence and Law*, pages 50–60, New York, 1993. ACM Press.

[7] C. Hafner and D. H. Berman. The role of context in case-based legal reasoning: Teleological, temporal, and procedural. *Artificial Intelligence and Law*, 10(1-3):19–64, 2002.

[8] T. F. Gordon and D. Walton. The Carneades argumentation framework — using presumptions and exceptions to model critical questions. In P. E. Dunne and T. Bench-Capon, editors, *Proceedings of the First International Conference on Computational Models of Argument (COMMA 06)*, Liverpool, September 2006.

[9] K. D. Ashley. *Modeling Legal Argument: Reasoning with Cases and Hypotheticals*. Artificial Intelligence and Legal Reasoning Series. MIT Press, Bradford Books, 1990.

[10] T. Bench-Capon and G. Sartor. A model of legal reasoning with cases incorporating theories and values. *Artificial Intelligence*, 150(1–2):97–143, November 2003.

[11] K. Greenwood, T. Bench-Capon, and P. McBurney. Towards a computational account of persuasion in law. In G. Sartor, editor, *International Conference on Artificial Intelligence and Law*, pages 22–31, Edinburgh, 2003. ACM.

[12] D. Walton. *Legal argumentation and evidence*. Pennsylvania State University Press, University Park, PA, 2002. Douglas N. Walton.

[13] D. Walton. *Appeal to Expert Opinion*. Penn State Press, University Park, 1997.

This page intentionally left blank

Reasoning About Action

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Argumentation and Persuasion in the Cognitive Coherence Theory

Philippe Pasquier [a,*], Iyad Rahwan [b,d], Frank Dignum [c] and Liz Sonenberg [a]
[a] *University of Melbourne, Australia*
[b] *British University of Dubai, UAE*
[c] *Utrecht University, The Netherlands*
[d] *(Fellow) University of Edinburgh, UK*

Abstract.
This paper presents a coherentist approach to argumentation that extends previous proposals on cognitive coherence based agent communication pragmatics (inspired from social psychology) and propose (1) an alternative view on argumentation that is (2) part of a more general model of communication. In this approach, the cognitive aspects associated to both the production, the evaluation and the integration of arguments are driven by calculus on a formal characterization of cognitive coherence.

1. Introduction

"Argumentation is a verbal, social and rational activity aimed at convincing [...] of the acceptability of a standpoint by putting forward a constellation of propositions justifying or refuting the proposition expressed in the standpoint." [1, page 1].

In AI and MAS, argumentation frameworks have been put forward for modelling inference, non-monotonic reasoning, decision making and argumentation-based communication has been introduced as a way to refine multiagent communication [2,3,4,5]. The syntax and semantics of argumentation have been extensively studied, but the pragmatics of argumentation (theory of its use in context) has not been inquired. While the conventional aspects of pragmatics have been taken into account in the formalisms proposed for argumentation dialogues, the cognitive aspects of argumentation have been less studied: when does an agent argue, with whom, on what topic? What are the cognitive effects of arguments (in terms of persuasion and integration)? What is the utility of the argumentation? Are the agents satisfied with their dialogue?

Cognitive coherence theory [6,7,8] has been put forward as a way to model the cognitive aspects of agent communication pragmatics (Section 2). Inspired by social psychology theories, cognitive coherence provides a native yet realistic modelling of the cognitive aspects of communication through the concept of *attitude change* which captures the persuasive aspect inherent to all communications (Section 3). In this paper, we extend the cognitive coherence approach to argumentation and show how this extension allows to model the generative aspect of argumentation communication as well as the

*Correspondence to: Philippe Pasquier. Tel.: +61 4 3341 2533; Fax: +61 3 9349 4596; E-mail: pasquier@unimelb.edu.au

cognitive response to persuasive arguments using a single set of principles (Section 4). Finally, links with previous approaches are discussed (Section 6).

This paper thus extends the state of the art by (1) proposing an alternative (coherentist) view on argumentation that is (2) part of a more general model of communication (including the cognitive aspect of pragmatics) and (3) giving a fully computational characterization of this new model.

2. The cognitive coherence framework

In cognitive sciences, cognitions gather together all cognitive elements: perceptions, propositional attitudes such as beliefs, desires and intentions, feelings and emotional constituents as well as social commitments.

In cognitive or social psychology, most cognitive theories appeal to the concept of homeostasis, i.e. the human faculty to maintain or restore some physiological or psychological constants despite the outside environment variations. All these theories share as a premise the *coherence principle* which puts coherence as the main organizing mechanism: *the individual is more satisfied with coherence than with incoherence*. The individual forms an opened system whose purpose is to maintain coherence as much as possible.

The core of our theoretical model is the unification of the dissonance theory from Festinger [9] and the coherence theory from Thagard [10]. In that context, our main and original theoretical contribution has been to extend that model to communication (which has not been treated by those two theorists) and to develop a formalism suited to MAS.

2.1. Formal characterization of cognitive coherence

While several formal characterizations of cognitive coherence have been made (logic-based [11], neural network or activation network based [12], probabilistic network [13], decision-theoretic, ...), we present one that is constraint satisfaction based resulting in a simple symbolic-connexionist hybrid formalism (we refer the reader to [14] for an introduction to this family of formalisms).

In this approach, cognitions are represented through the notion of elements. We denote $\mathbb{E}$ the set of all elements. *Elements* (i.e. cognitions) are divided in two sets: the set $\mathcal{A}$ of *accepted elements* and the set $\mathcal{R}$ of *rejected elements*. A closed world assumption which states that *every non-explicitly accepted element is rejected* holds. Since all the cognitions are not equally modifiable, a *resistance to change* is associated to each element of cognition. In line with Festinger [9], a cognition's resistance to change depends on its type, age, as well as the way in which it was acquired: perception, reasoning or communication. Resistances to change allow to differentiate between beliefs that came from perception, beliefs that came from reasoning and beliefs that came from communication as well as to represent the individual commitment strategies associated with individual intention. Resistance to change can be accessed through the function $Res : \mathbb{E} \longrightarrow \mathbb{R}$

Those elements can be cognitively related or unrelated. For elements that are directly related, two types of non-ordered binary constraints represent the relations that hold between them in the agent's cognitive model:

- *Positive constraints*: positive constraints represent positive relations like facilitation, entailment or explanatory relations.

- *Negative constraints*: negative constraints stand for negative relations like mutual exclusion and incompatibility relations.

We note $\mathcal{C}^+$ (resp. $\mathcal{C}^-$) the set of positive (resp. negative) constraints and $\mathbb{C} = \mathcal{C}^+ \cup \mathcal{C}^-$ the set of all constraints. For each of these constraints, a weight reflecting the importance degree for the underlying relation can be attributed[1]. Those weights can be accessed through the function $Weight : \mathbb{C} \longrightarrow \mathbb{R}$. Constraints can be satisfied or not.

Definition 1 (Cognitive Constraint Satisfaction) *A positive constraint is satisfied if and only if the two elements that it binds are both accepted or both rejected, noted* $Sat^+(x,y) \equiv (x,y) \in \mathcal{C}^+ \wedge [(x \in \mathcal{A} \wedge y \in \mathcal{A}) \vee (x \in \mathcal{R} \wedge y \in \mathcal{R})]$. *On the contrary, a negative constraint is satisfied if and only if one of the two elements that it binds is accepted and the other one rejected, noted* $Sat^-(x,y) \equiv (x,y) \in \mathcal{C}^- \wedge [(x \in \mathcal{A} \wedge y \in \mathcal{R}) \vee (x \in \mathcal{R} \wedge y \in \mathcal{A})]$. *Satisfied constraints within a set of elements* $\mathcal{E}$ *are accessed through the function* $Sat : \mathcal{E} \subseteq \mathbb{E} \longrightarrow \{(x,y) | x,y \in \mathcal{E} \wedge (Sat^+(x,y) \vee Sat^-(x,y))\}$

In that context, two elements are said to be *coherent* if they are connected by a relation to which a satisfied constraint corresponds. And conversely, two elements are said to be *incoherent* if and only if they are connected by a non-satisfied constraint. These relations map exactly those of dissonance and consonance in Festinger's psychological theory. The main interest of this type of modelling is to allow defining a metric of cognitive coherence that permits the reification of the coherence principle in a computational calculus.

Given a partition of elements among $\mathcal{A}$ and $\mathcal{R}$, one can measure the *coherence degree* of a non-empty set of elements $\mathcal{E}$. We note $Con()$ the function that gives the constraints associated with a set of elements $\mathcal{E}$. $Con : \mathcal{E} \subseteq \mathbb{E} \longrightarrow \{(x,y) \mid x,y \in \mathcal{E}, (x,y) \in \mathbb{C}\}$.

Definition 2 (Cognitive Coherence Degree) *The* ***coherence degree*** $C(\mathcal{E})$*, of a non-empty set of elements,* $\mathcal{E}$ *is obtained by adding the weights of constraints linking elements of* $\mathcal{E}$ *which are satisfied divided by the total weight of concerned constraints. Formally:*

$$C(\mathcal{E}) = \frac{\sum_{(x,y) \in Sat(\mathcal{E})} Weight(x,y)}{\sum_{(x,y) \in Con(\mathcal{E})} Weight(x,y)} \tag{1}$$

The general coherence problem is then:

Definition 3 (Cognitive Coherence Problem) *The general* ***coherence problem*** *is to find a partition of the set of elements into the set of accepted elements* $\mathcal{A}$ *and the set of rejected elements* $\mathcal{R}$ *that maximizes the cognitive coherence degree of the considered set of elements.*

It is a constraint optimization problem shown to be NP-complete in [16]. An agent can be partially defined as follows:

Definition 4 (Agent's State) *An* ***agent's state*** *is characterized by a tuple* $W = \{\mathcal{P}, \mathcal{B}, \mathcal{I}, SC, \mathcal{C}^+, \mathcal{C}^-, \mathcal{A}, \mathcal{R}\}$*, where:*

- $\mathcal{P}$,$\mathcal{B}$,$\mathcal{I}$ *are sets of elements that stand for perceptions, beliefs and individual intentions respectively,* SC *is a set of elements that stand for the agent's agenda,*

[1]This is a way of prioritizing some cognitive constraints as it is done in the BOID architecture [15].

that stores all the social commitments from which the agent is either the debtor or the creditor;

- $\mathcal{C}^+$ *(resp.* $\mathcal{C}^-$*) is a set of non-ordered positive (resp. negative) binary constraints over* $\mathcal{P} \cup \mathcal{B} \cup \mathcal{I} \cup \mathcal{SC}$ *such that* $\forall (x, y) \in \mathcal{C}^+ \cup \mathcal{C}^-$, $x \neq y$;
- $\mathcal{A}$ *is the set of accepted elements and* $\mathcal{R}$ *the set of rejected elements and* $\mathcal{A} \cap \mathcal{R} = \varnothing$ *and* $\mathcal{A} \cup \mathcal{R} = \mathcal{P} \cup \mathcal{B} \cup \mathcal{I} \cup \mathcal{SC}$.

Beliefs coming from perception ($\mathcal{P}$) or from reasoning ($\mathcal{B}$) as well as intentions ($\mathcal{I}$) constitute the *private cognitions* of the agent, while public or social cognitive elements are captured through the notion of social commitments (as defined in [17]). Social commitment has proven to be a powerful concept to capture the interdependencies between agents [18]. In particular, it allows to represent the semantics of agents' communications while respecting the principle of the asymmetry of information that indicates that in the general case what an agent say does not tell anything about what he thinks (but still socially commits him).

This agent model differs from classical agent modelling in that motivational attributes are not statically defined but will emerge from the cognitive coherence calculus. Concretely, this means that we don't have to specify the agent's desires (the coherence principle allows to compute them) but only potential intentions or goals. Examples to be given in this paper will highlight the *motivational drive* associated with cognitive coherence.

Incoherence being conceptually close to the notion of conflict, we use a typology borrowed from works on conflicts [19].

Definition 5 (Internal vs. External Incoherences) *An incoherence is said to be* ***internal*** *iff all the elements involved belong to the private cognitions of the agent, else it is said to be* ***external****.*

2.2. Local search algorithm

Decision theories as well as micro-economical theories define utility as a property of some valuation functions. A function is a *utility function* if and only if it reflects the agent's preferences. In the cognitive coherence theory, according to the afore-mentioned coherence principle, coherence is preferred to incoherence which allows to define the following expected utility function[2].

Definition 6 (Expected Utility Function) *The expected utility for an agent to attempt to reach the state* W' *from the state* W *(which only differ by the acceptance state of a subset* E *of the agent's elements) is expressed as the difference between the incoherence before and after this change minus the cost of the change (expressed in term of the resistance to change of the modified elements):* $G(W') = C(W') - C(W) - \sum_{X \in E} Res(X)$.

At each step of his reasoning, an agent will search for a cognition acceptance state change which maximizes this expected utility. If this cognition is a commitment, the agent will attempt to change it through dialogue and if it is a private cognition (perceptions, beliefs or intentions), it will be changed through attitude change.

A recursive version of the local search algorithm the agents use to maximize their cognitive coherence is presented in [8] and consists of four phases:

[2]Note that our expected utility function does not include any probabilities. This reflects the case of equiprobability in which the agent has no information about other's behaviour. Notice that integrating algorithms to progressively learn such probabilities is an obvious perspective of the presented model.

1. For each element e in the agent state, calculate the expected utility and the gain (or loss) in coherence that would result from flipping e, i.e. moving it from $\mathcal{A}$ to $\mathcal{R}$ if it is in $\mathcal{A}$, or moving it from $\mathcal{R}$ to $\mathcal{A}$ otherwise.
2. Produce a new solution by flipping the element that most increases coherence, or with the biggest positive expected utility if coherence cannot be improved. Update the resistance to change of the modified element to avoid looping.
3. Repeat 1 and 2 until either a social commitment is encountered (a dialogue is needed as an attempt to flip it) or until there is no flip that increases coherence and no flip with positive expected utility.
4. Return result. The solution will be applied if and only if the cumulated expected utility is positive.

Since it does not make any backtracking, the complexity of this algorithm is polynomial: $\mathcal{O}(mn^2)$, where n is the number of elements considered and m the number of constraints that bind them[3]. We don't have a proof of correctness of this greedy algorithm in regards to the general coherence problem but, it behaved optimally on tested examples. We refer the interested reader to [8] for full justification and discussion of this algorithm. Traces of execution will be provided along with the examples in this paper.

2.3. Cognitive coherence applied to agent communication

Applied to agent communication, the cognitive coherence theory supplies theoretical and practical elements for automating agent communication. This framework has been implemented and exemplified as presented and discussed in [20] and [7]. The presented practical framework relies on our dialogue games based agent communication language (DIAGAL) and our dialogue game simulator toolbox (DGS)[21].

3. Attitude change and persuasion

From the set of all private cognitions result *attitudes* which are positive or negative psychological dispositions towards a concrete or abstract object or behaviour.

For contemporary psychologists, attitudes are the main components of cognition. These are the subjective basis to rational action [22]. Theoretically, an agent's behaviour is determined by his attitudes. The basic scheme highlighted by those researches is that beliefs (cognition) and desires (affect) lead to intentions which could lead to actual behaviours or dialogical attempts to get the corresponding social commitments depending on their nature.

From another point of view, it could happen (due to hierarchies and roles, power relations, persuasive argumentation, material constraints, ...) that an agent comes to accept a counter-attitudinal course of action or proposition. In that case, *attitude change* might occur. Since cognitive coherence theory is built over five decades of research on attitude change in social psychology, it provides a native yet realistic modelling of the cognitive aspects of persuasion through this concept of attitude change. Within our characterization of cognitive coherence, attitude change refers to the change of acceptance states of some private element of cognition in order to restore coherence with external interdependencies, i.e. social commitments.

[3] n coherence calculus (sum over m constraints) for each level and a maximum of n levels to be searched.

4. Argumentation in the cognitive coherence theory

Argumentation has not been introduced in the cognitive coherence approach yet. However, this extension follows naturally from previous work by saying that argumentation, explanation and justification are the processes by which an agent shows to the other agents why his (or a given) position is coherent. In that context, we do not distinguish between argumentation, explanation and justification which all aim to convince in some way. More specifically, the idea behind argumentation is that agents can construct, exchange and weigh up arguments relevant to conflicting issues, in the context of an explicit external incoherence.

The argumentation process can be modelled using three steps: (1) argument generation, (2) argument evaluation and (3) argument integration. The next Sections present and exemplify how cognitive processes associated with those steps are computed in the cognitive coherence framework.

4.1. Argument generation

Argumentation is a means to an end, the end being persuasion, that is attitude change. But at the same time, argumentation is a type of information disclosure and competitive (or malicious) agents can use this information to endorse non-cooperative behaviour. In this paper, we won't address strategic issues related to argumentation.

In the cognitive coherence framework, argumentation will be used in a systematic way when an explicit external incoherence is not solved otherwise (for example by referring to an authority relation or a social norm). When this precondition will be met, the agents will disclose the private part of the connected component related to the discussed issue, i.e. element. Previous work has been made about argumentation as constraint propagation in the field of distributed constraint satisfaction [23].

Definition 7 ***(Argument)*** *An* argument *for an element acceptance or rejection is a set of elements (along with their acceptance states and resistances to change) and constraints (along with their weights) that form a connected component in the network of cognitions of the agent. More formally, an argument w is a pair $w = \langle H, h \rangle$ such that:*

1. $H \subseteq \mathbb{E}, h \in \mathbb{E}; H \cap \{h\} = \varnothing$;
2. $\forall x, y \in H \cup \{h\}, \exists z_1, ..., z_n \in H \cup \{h\}, (x, z_1), ..., (z_n, y) \subseteq \mathbb{C}$ *(connexity condition);*

H is called the support of the argument while h is the conclusion of the argument.

Definition 8 ***(Argument types)***

Arg_X stands for the set of all possible arguments that can be generated from the agent's bases included in X. It is useful to differentiate between:

- *belief arguments: $\langle H, h \rangle$ is a belief argument iff $(H \cup \{h\}) \subset Arg_{\mathcal{P} \cup \mathcal{B}}$;*
- *practical arguments: $\langle H, h \rangle$ is a practical argument iff $(H \cup \{h\}) \subset Arg_{\mathcal{P} \cup \mathcal{B}} \wedge h \in \mathcal{I}$;*
- *social arguments: $\langle H, h \rangle$ is a social argument iff $(H \cup \{h\}) \subset Arg_{\mathcal{I} \cup SC} \wedge (H \cup \{h\}) \cap SC \neq \varnothing$;*

4.2. Issues in argument evaluation and integration

Argument evaluation and integration are complex issues, and social psychology (which has studied that problem on experimental basis for half a century now) indicates that

there is a large number of aspects to be considered [22]. Here is a simplified listing of those:

- *evaluation of the source*: authority, trust, credibility, attractiveness;
- *evaluation of the message*: comprehension and quality of argument, number and order of arguments, one- and two-sided messages, confidence, fear;
- *characteristics of the audience*: intelligence and self-esteem, psychological reactance, initial attitudes, heterogeneity, sex differences;
- *characteristics of the medium*: media and channel of communication, media functions, temporality of the communication.

Furthermore, many studies indicate that the regularities in that area are difficult to find and that argumentation evaluation and integration are also linked to cognitive learning and thus depend on the dynamics of the learner [24]. However, a characterization of rational agent argumentation may not take all of these into consideration. We thus restrict the discussion to the salient elements that are already considered in cognitive agent modelling and MAS:

- *trust and credibility*: the levels of trust and credibility associated with the protagonist influence the argument evaluation and integration process. The model presented in [11] (inspired by cognitive coherence approach) has inquired this link further. For the sake of simplicity, in this paper, we will consider that the levels of trust and credibility are the highest possible;
- *initial attitude toward the standpoint defended by the argument*: it is clear that the initial attitude of the antagonist agent will intervene in argument evaluation and integration especially in conjunction with trust and credibility. Social psychology, in particular the theory of social judgment [25], showed that each agent maintains some acceptability intervals in which arguments may be taken into account while arguments falling out of those intervals will be considered too extreme and won't be taken into account. However, because we model rational agents that usually operate in quite precise and well known domains, we will make the assumption that all arguments will be considered;
- *initial attitude toward the protagonist of the argument*: this issue is related to the level of trust and cooperativeness that the antagonist shows toward the protagonist. Will the agent integrate the other's point of view in his own cognitive model and act accordingly (which would be very cooperative) or will he compare his point of view with the other's and then substitute those two if his own is weaker and reject the other's one if it is (subjectively) evaluated as weaker? In this paper, we make the assumption that the agents will fully integrate the other argument in their mental states;
- *Heterogeneity of the participants*: we call *objective evaluation* the case where all the participants share the same evaluation function and we name *subjective evaluation* the case in which they all have their own. This aspect depends on the type of system addressed. While objective evaluation might be possible in cooperative systems, open system where agents may be heterogeneous will most probably rest on subjective evaluation. In this paper, we will make the assumption that the agents share the same evaluation function to be described.
- *number and quality of arguments*: in this paper, we will focus on cognitive factors which will tend to reduce argument evaluation to this last category. We will also make the assumption that'all arguments are valid and meaningful.

There are two ways of dealing with evaluation and integration: (1) do a selective evaluation and then integration of arguments after some adjustment of strength due to the evaluation, (2) doing integration and seing evaluation as the side-effect consequence of the memorization of arguments by the agents. While in previous work [26] we have explored the first path, this paper explores the second one.

4.3. Argument integration

Here, we make the hypothesis that each agent fully integrates the other's point of view in his own cognitive coherence calculus. This means that the perceptions and beliefs as well as goals and social commitments supporting the other's point of view are integrated in the cognitive model of the agent regardless of their strength. This corresponds to a fully cooperative and trustful cognitive behaviour. Many other integration strategies are possible and will be discussed and compared as part of our future work.

Cooperation in cognitive coherence theory results from the fact that once an agent is aware (even partially) about the other's cognitive constraints, he will be able to take them into account in his own coherence seeking. This argument integration procedure is fully cooperative since the others' arguments will be fully taken into account in future reasoning.

4.4. Argument evaluation

The main consequence of this integration procedure is that we don't need argument evaluation. Argument evaluation and eventual persuasion (attitude change) will be done by the cognitive coherence calculus as a result of the argument integration. According to our hypothesis, the behaviour, decision or beliefs of the agent may be changed (or not) depending on the effect of the integration (i.e. memorization) of the argument encountered. This is quite intuitive according to our hypothesis of cooperation, sincerity and validity of the arguments.

5. Example

As an example, we consider the two agents W and J that are driving a car (a shared resource) and have to decide which way to go next. Their initial states are represented by Figure 1, part (a). Using the approach presented in this paper (on top of the one presented in previous work), they will generate the following dialogue (using DIAGAL dialogue games instead of natural language):

$J.1$*: I would turn right.*
$W.1$*: I would rather turn left. there is a rugby match and there gona be lots of traffic on the right road.*
$J.2$*: But, there 's a lot of lights on the left road, that will slow us down and the right road is shorter. Can't we turn right instead?*
$W.2$*: Ok, fine, lets turn right then.*

Part (d) of Figure 1 indicates the coherence calculus that leads J to initiate a dialogue and utter an offer ($J.1$) as an attempt to get the social commitment to turn right accepted. This explicits an external incoherence for W whose cognitive coherence calculus leads to a different view, i.e. turning left (Part (c) of Figure 1). W then refuses J's offer, counter offers and discloses his arguments ($W.1$). That also makes explicit the external

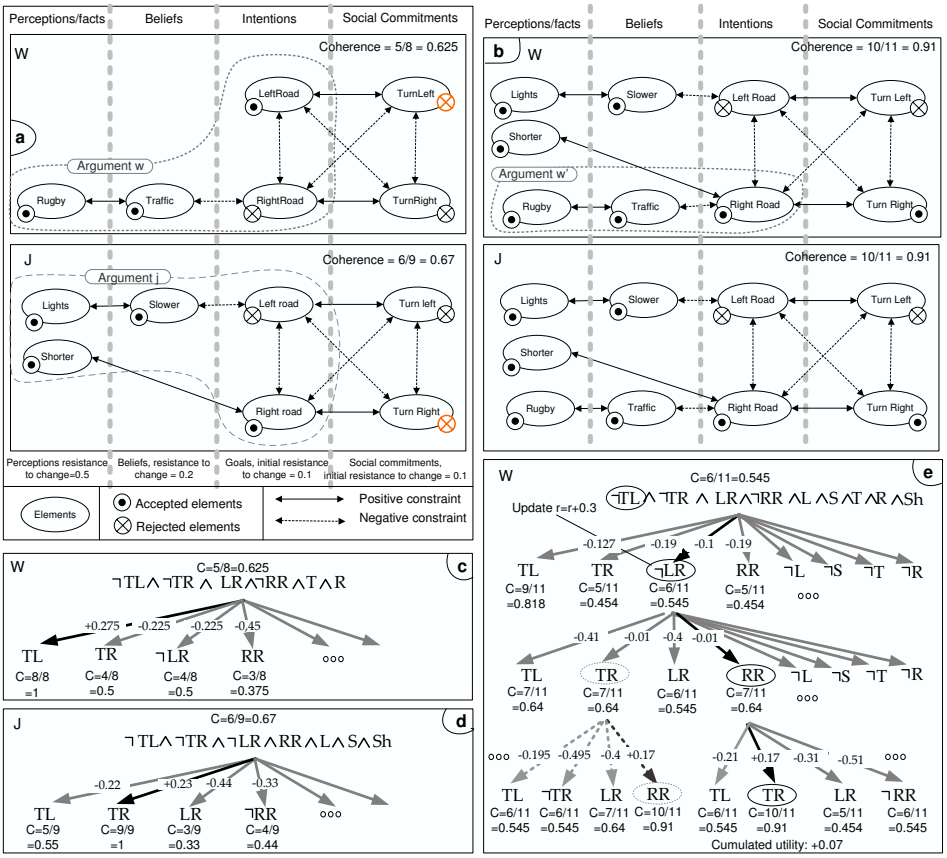


Figure 1. Parts (a) and (b) represents initial and final states for W and J, parts (c),(d) and (e) are traces of the local search algorithms, where arcs are labeled with the expected utility values. All the weights of the constraint are equal to 1. Resistances to change are initialized so that perceptions are more resistant to change than beliefs that are a bit more resistant to change than intentions and social commitments. Updates of the resistance to change when a change occurs is of $+0.3$. Other choices would avoid fanatism (i.e. looping) as well.

incoherence for J who counter argues ($J.2$). Part (e) of Figure 1, shows how W's cognitive coherence calculus leads him to an attitude change and an acceptance of J's point of view after memorizing its argument (without forgetting his own). Also notice that in part (e) a non deterministic choice is done by the local search algorithm, the part in dashed grey indicates what would have happened if the other path would have been explored: the same result would have occurred except that W's acceptance of J's request ($W.2$) would have occurred after rather than before the adoption of the corresponding intention. Finally, Part (b) of Figure 1 indicates the agents' states after that dialogue as well as their now shared and increased cognitive coherences.

Notice how, the local search algorithm (i.e. the agent cognition process) was used to drive both the structuration of the dialogue and the cognitive response to argumentation (in our case, W being convinced by J's argumentation).

6. Discussion

6.1. Comparison with Dung's approach to argumentation

If we represent our example of Figure 1 within the classical argumentation approach defined in [27], in which we call J's argument j and W's one w, we obtain the following argumentation framework: $< \{w, j\}, \{(w, j), (j, w)\} >$, composed of the two arguments and their attack relation. This particular argumentation framework has two *acceptable stable preferred extensions* (namely $\{w\}$ and $\{j\}$), which doesn't say much about persuasion. According to the semantics of acceptability in Dung's and subsequent approaches, a credulous agent accepts all acceptable extensions while a sceptical one only accepts the intersection of all acceptable extensions (which is void here). In other words, as noted in [28], Dung's approach to argumentation does not allow to fully treat persuasion.

In a multi-agent setting, preferences are needed in order to conclude (as shown by Amgoud and al. [29]). In our approach, preferences are implicit and follow from the coherence principle that coherence is preferred to incoherence. Since this is true both at the qualitative and quantitative levels, we don't need any extra treatment for taking preferences into account.

6.2. On bipolarity in the cognitive coherence approach

While Dung's framework only considers one type of interaction between arguments (i.e. attacks), it has been extended to take into account bipolarity, that is the fact that supportive and negative arguments may be differentiated, which has been shown to be useful in a number of applications [30].

In our framework, the notion of argument can be refined to consider supportive argument as well as negative argument. Here, we provide the following definitions:

Definition 9 (Supportive Argument) *A **supportive argument** for an element acceptance (resp. rejection) is (1) an argument in the sense of definition 7 that is (2) optimally coherent with the acceptance (resp. rejection) of the conclusion.*

Definition 10 (Negative Argument) *A **negative argument** for an element acceptance (resp. rejection) is (1) an argument in the sense of definition 7 for which (2) there exist an assignation that would be more coherent than the current one in which the conclusion is rejected (resp. accepted).*

For example, in Figure 1, $argumentw$ is a supportive practical argument for the acceptance of the intention to go by the left road (noted LR), while w' is a negative practical argument for the acceptance of RR.

Because of the use of social commitments and the lack of links between social commitment and private cognition the integration part is usually not modeled in AI approaches to argumentation issued from formal dialectics. Agents just do evaluation (of acceptable arguments), i.e. manage the public commitment store. We provide a more realistic model that includes links between social and private cognitions [31] and where the integration (i.e. memorization) of the others arguments is fully accounted. In that context evaluation, and more generally persuasion (i.e. eventual attitude change) is the result of the cognition process on the updated set of cognitive elements.

Further relation(s) with previous work and other approaches to argumentation are left as future work.

7. Conclusion

In this paper, we have highlighted the persuasive aspects inherent to every communication (thus including argumentation) by providing a model in which the cognitive response to persuasive message was modelled (by reifying the concept of attitude change when necessary). The strength of the proposed approach resides in the facts that: (1) all the steps of argumentation are computed using a single set of measures, i.e. the cognitive coherence metrics, (2) the approach is grounded in behavioural cognitive sciences rather than in dialectics and is part of a more general theory of mind, which covers many dimensions of the cognitive aspects of pragmatics and (3) our characterization is computational.

The presented framework has been developed in order to fill the need (that is not covered by previous approaches) of implementable argumentation based frameworks that are integrated to a more general agent architecture and communication framework.

References

[1] F. H. van Eemeren and R. Grootendorst. *A Systematic Theory of Argumentation: the Pragma-Dialectical Approach*. Cambridge University Press, 2004.
[2] I. Rahwan, S. Ramchurn, N. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation based negotiation. *Knowledge Engineering Review*, 18(4):343–375, 2003.
[3] B. Moulin, H. Irandoust, M. Bélanger, and G. Desbordes. Explanation and argumentation capabilities: Towards the creation of more persuasive agents. *Artificial Intelligence Review*, 17(3):169–222, 2002.
[4] ASPIC Consortium. Theoretical framework for argumentation. Prepared for the european commission, ASPIC(Argumentation Service Platform with Integrated Components), http://www.argumentation.org/, 2004.
[5] ASPIC Consortium. Review on argumentation technology: State of the art, technical and user requirements. Prepared for the european commission, ASPIC(Argumentation Service Platform with Integrated Components), http://www.argumentation.org/, 2004.
[6] P. Pasquier and B. Chaib-draa. The cognitive coherence approach for agent communication pragmatics. In *Proceedings of The Second International Joint Conference on Autonomous Agent and Multi-Agents Sytems (AAMAS'03)*, pages 544–552. ACM Press, 2003.
[7] P. Pasquier and B. Chaib-draa. Agent communication pragmatics: The cognitive coherence approach. *Cognitive Systems*, 6(4):364–395, December 2005.
[8] P. Pasquier. *Aspects cognitifs des dialogues entre agents artificiels : l'approche par la cohérence cognitive*. PhD thesis, Laval University, Quebec, Canada, August 2005.
[9] L. Festinger. *A Theory of Cognitive Dissonance*. Stanford University Press, 1957.
[10] P. Thagard. *Coherence in Thought and Action*. The MIT Press: Cambridge, MA, USA, 2000.
[11] J-P. Sansonnet and E. Valencia. Dialogue between non-task oriented agents. In *Proceedings of the 4th Workshop on Agent Based Simulation (ABS'04)*, Montpelier, France, april 2003. http://www.limsi.fr/Individu/jps/research/buzz/buzz.htm.
[12] R. Shultz and R. Lepper. *Cognitive Dissonance : progress in a pivotal theory in social psychology*, chapter Computer simulation of the cognitive dissonance reduction, pages 235–265. American Psychological Association, 1999.
[13] P. Thagard. Probabilistic network and explanatory coherence. *Cognitive science Quaterly*, (1):91–114, 2000.
[14] R. Sun. *Connectionist-Symbolic Integration*, chapter An introduction to hybrid connectionist-symbolic models. Lawrence Erlbaum Associates., 1997.
[15] J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. Van der Torre. The BOID architecture: Conflicts between beliefs, obligations, intention and desires. In *Proceedings of the Fifth International Conference on Autonomous Agent*, pages 9–16. ACM Press, 2001.

[16] P. Thagard and K. Verbeurgt. Coherence as constraint satisfaction. *Cognitive Science*, 22:1–24, 1998.
[17] P. Pasquier, R. A. Flores, and B Chaib-draa. Modelling flexible social commitments and their enforcement. In *Proceedings of the Fifth International Workshop Engineering Societies in the Agents World (ESAW'04)*, volume 3451 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 153–165. Springer-Verlag, 2004.
[18] M. P. Singh. An ontology for commitments in multiagent systems: Toward a unification of normative concepts. *Artificial Intelligence and Law*, 7:97–113, 1999.
[19] F. Dehais and P. Pasquier. Approche Générique du Conflit. In D.L. Scapin and E. Vergisson, editors, *Ergonomie et Interaction Homme-Machine (ErgoIHM 2000)*, pages 56–63, France, 2000. ESTIA (École Supérieure des Technologies Industrielles Avancées).
[20] P. Pasquier, N. Andrillon, and B. Chaib-draa. An exploration in using cognitive coherence theory to automate BDI agents' communicational behavior. In F. Dignum, editor, *Advances in Agent Communication - International Workshop on Agent Communication Languages (ACL'03)*, volume 2922 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 37–58. Springer-Verlag, 2003.
[21] B. Chaib-draa, M. Bergeron, M.-A. Labrie, and P. Pasquier. Diagal: An agent communication language based on dialogue games and sustained by social commitments. *Journal of Autonomous agents and Multi-agents Systems (to appear)*, 2005.
[22] P. Erwin. *Attitudes and Persuasion*. Psychology Press, 2001.
[23] H. Jung, M. Tambe, and S. Kulkarni. Argumentation as distributed constraint satisfaction: Applications and results. In *Proceedings of the International Conference on Autonomous Agents (Agents'01)*, pages 324–331, Montreal, Canada, 2001. ACM Press.
[24] A. G. Greenwald. *Psychological Foundations of Attitude Change*, chapter Cognitive Learning, Cognitive Response to Persuasion and Attitude Change, pages 147–170. Academic Press, New York, 1968.
[25] M. Sherif and C. I. Hovland. *Social Judgement*. Yale University Press, New Haven, USA, 1961.
[26] P. Pasquier, I. Rahwan, F. Dignum, and L. Sonenberg. Argumentation and persuasion in the cognitive coherence approach: preliminary repport. In *Proceedings of the third workshop on Argumentation in Multiagent Systems (ArgMas)*, Hakodate, Japan, 2006.
[27] 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.
[28] T. J. M. Bench-Capon. Persuasion in practical argument using value based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
[29] L. Amgoud and C. Cayrol. On the acceptability of arguments in preference-based argumentation. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI 1998)*, pages 1–7, San Francisco CA, USA, 1998. Morgan Kaufmann Publishers.
[30] L. Amgoud, C. Cayrol, and M.-C. Lagasquie-Schiex. On the bipolarity in argumentation frameworks. In *10th International Workshop on Non-Monotonic Reasoning (NMR-2004)*, pages 1–9, 2004.
[31] P. Pasquier and B. Chaib-draa. Linking individual intention and social commitments. In *Proceedings of the fifth internationnal conference on Autonomous Agents and Multiagent Systems (AAMAS'06)*, Hakodate, Japan, 2006. ACM Press.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Knowing When To Bargain

The roles of negotiation and persuasion in dialogue

Simon Wells and Chris Reed
School of Applied Computing, University of Dundee

Abstract. In this paper two formal dialectic systems are described, a persuasion protocol (PP_0) and a negotiation protocol (NP_0), together with a method for shifting from an instance of a persuasion dialogue to an instance of a negotiation dialogue. The rationale for this kind of shift is explored in the context of the fallacy of bargaining. Such a dialectical shift is proposed as a valuable way to enable the participants in an argumentative dialogue to proceed towards a practical settlement when they are otherwise unable to persuade each other.

Keywords. Argumentation, Communication Protocol, Multiagent System, Formal Dialectical System

1. Introduction

A typical situation in argumentative dialogue occurs when one party attempts to persuade another party to accept some standpoint. This involves notions of attack and defence as the parties attempt to justify their own position whilst refuting that of their opponent. However, because the participants are autonomous entities they will each evaluate the proffered arguments on their own terms. An argument that party A believes is sufficient to persuade party B isn't necessarily the same argument that B would accept and thus be persuaded. What should occur when A cannot persuade B? If getting B to accept the standpoint is important to A, then A should have available an alternative tactic for reaching agreement in those situations where a sufficiently persuasive argument cannot be brought to bear.

In real-world argument many people resort to bargaining when they are unable to persuade their opponent. For example, Harry and Sally are arguing about who should do the washing up. Both have stated that they will not do the washing up and that the other should do it. Sally tries to persuade Harry to do the washing up and defends her position, when it is inevitably attacked, by stating that she always does the washing up and asks why Harry can't do it for a change. Harry justifies his refusal to do the washing up with the defense that he has just hoovered the living room and so he shouldn't have to do both jobs. Domestic conflicts such as this are a common occurrence that are often resolved when an offer is made, for example, Harry concedes he will do the washing up if Sally will take the rubbish out. This is not a concession based upon Sally's superior persuasive argument but based upon a wider view of the situation and the need to reach a practical settlement. The fact that the rubbish needed to be taken out was not an issue that was

raised in the preceding persuasion dialogue but was an issue that could be raised during a negotiation dialogue.

When a party cannot get their standpoint accepted through justification of that standpoint an alternative strategy is to enter into some sort of negotiation over the issue to determine; what it would take to get the standpoint accepted by the other party, and, failing that, to determine what alternative (possibly reduced) standpoint B might accept if it turns out that the original standpoint is unlikely ever to be acceptable.

This kind of situation can be characterised as the movement within a dialogue from a persuasion-type sub-dialogue to a negotiation-type sub-dialogue. This paper introduces two formal dialectic systems named Persuasion Protocol $_0$ (PP_0) and Negotiation Protocol $_0$ (NP_0), together with a method for moving from a persuasion sub-dialogue carried out in accordance with PP_0 to a negotiation sub-dialogue carried out in accordance with NP_0. The aim is to demonstrate that this particular shift, from persuasion to negotiation, can be a useful way to procede when a persuasion dialogue is unlikely to reach a stable agreement. These results can then be applied to computational models of argument such as those for use in multiagent systems. Agents may have many more capabilities than those that are relevant to the current persuasion dialogue. If $agent_1$ cannot persuade $agent_2$ then $agent_1$ may use the opportunity to shift to a negotiation dialogue in which a concession might be won.

2. Background

This paper deals with a number of topics in argumentation including the use of formal dialectic systems to model the interactions between participants in an argumentative dialogue, the recognition that dialogues conform to a number of distinct types, and that given a formal dialectic system which models the interactions in a particular type of dialogue, there will arise the need to shift from a dialogue of one type to a dialogue of another type, and hence transition from one dialectic system to another.

Dialogue games have been proposed as a means to model the interactions between participants during argumentative dialogues. One branch of dialogue game research is into formal dialectic systems [1]. These are two-player, turn-taking games in which the moves available to the players represent the locutional acts or utterances made by the participants of a dialogue. Many dialectic systems have been proposed based on the characterisations of a range of dialogical situations, for example, Hamblin's system [1] and Mackenzie's DC [2] are targeted towards fallacy research whilst Walton and Krabbe's system PPD_0 [3] models the interactions between parties in a permissive persuasion dialogue. Girle introduces a number of systems which are aimed at modelling belief revision in A.I. systems [4,5,6]. McBurney and Parsons specify some games for use in communication between agents in multiagent systems [7]. Bench-Capon *et al.* introduce a system for modelling dialectical argument called the Toulmin Dialogue Game [8] that is based upon the argument schema of Toulmin [9].

Dialogues can be categorised into types distinguished by a range of characteristics such as initial situation, overall goal and the participant's individual aims. An influential but partial typology of such dialogue types which includes information-seeking, persuasion, negotiation, deliberation, and inquiry can be found in [3]. This paper is concerned with the negotiation and persuasion types of dialogue but can be extended to incorporate

other dialogue types. In multiagent systems research, negotiation is often characterised as a means to distribute limited resources between competing agents. Negotiation dialogues can be used to determine the distribution of those resources between the conflicting parties. In the Walton and Krabbe typology negotiation dialogues are characterised by a conflict of interests and a need for cooperation leading to a practical settlement. Persuasion dialogues occur when there is a conflict and the participants attempt to reach a stable agreement or resolution of the issue that gave rise to the conflict. Walton and Krabbe specify a formal dialectic system to model the interactions during persuasion dialogues name PPD_0.

The notion of embedding an instance of one type of dialogue within an instance of another type of dialogue was proposed in [3] which introduced the system PPD_1 which governs the embedding of rigourous persuasion dialogues within permissive persuasion dialogues. Various other approaches have been proposed including Reed's Dialogue Frames [10], and the layer model of McBurney and Parsons [11]. The core idea is to enable the participants in a dialogue to move from a sub-dialogue of one type to a sub-dialogue of another type where each sub-dialogue has its own specification of rules governing how a dialogue of that type should progress. The notion of embedding persuasion sub-dialogues within an ongoing negotiation dialogue has been explored quite extensively by Sycara in relation to the PERSUADER system [12], and by Rahwan [13] in relation to argument-based negotiation in multiagent systems. However the converse situation of embedding negotiation sub-dialogues within a persuasion dialogue has not been explored specifically except as a by-product of enabling embeddings and shifts in general.

3. The fallacy of Bargaining

Walton and Krabbe identify in [3] that shifts from one type of dialogue to another may be either licit or illicit. A licit shift occurs when the shift is constructive and agreed to by all parties. When a shift is concealed or otherwise inappropriate then it is illicit. Walton argues that a characteristic of many fallacies is that they occur where shifts in the dialogue are illicit [14]. In [3] the fallacy of bargaining is identified as occuring when participants are engaged in a dialogue which starts out as a persuasion but that at some point during the course of the dialogue an illicit shift occurs from persuasion to negotiation.

The example of the fallacy of bargaining used by Walton and Krabbe involves a government minister of finance who has been caught profiting from certain tax exemptions. The minister argues that those tax exemptions should be allowed temporarily and not be penalized. The minister then goes on to propose to his critics that if they abstain from moving for penalties for the exemptions, then he will not oppose a bill that the critics will benefit from. In this case, instead of satisfying his burden of proof with respect to his position on the tax exemptions, the minister substitutes an offer for an argument, a move which is not permissible in persuasion dialogues. By making an offer during the persuasion dialogue the minister has reneged on his commitment to defend his position, *vis a vis* the tax exemptions, and caused an illicit shift to a negotiation dialogue.

However, the shift from persuasion to negotiation need not always be an instance of the fallacy of bargaining. As Walton and Krabbe recognise, illicit shifts occur when the

shift is concealed or inappropriate and a fallacy can occur as a result, If the shift occurs in an open way, and is demonstrated to be appropriate then there is no need to characterise it as fallacious. Where conflicting participants in a dialogue have exhausted their persuasive arguments and are in a position that is unlikely to be resolved through continuation of the persuasion dialogue then it is acceptable for the participants to try some other way to break the deadlock. In an agent situation the failure to reach agreement can be undesireable, requiring that the agents replan which is computationally expensive. Given that both participants actually wish to resolve the conflict, which is the reason why they are still engaged in the dialogue at this point, a shift to another type of dialogue enables the participants to continue. If the shift is from a persuasion dialogue to a negotiation dialogue then the participants may be able to reach a practical settlement and so be able to move forward.

The dialogue protocols presented in this paper together with the associated machinery to effect dialogue shifts are aimed at demonstrating two points. Firstly that not all shifts from persuasion to negotiation dialogues need be instances of the fallacy of bargaining, and secondly that these kinds of shifts can be utilised to enable participants who would otherwise have reached an impasse to continue.

4. The systems: PP_0 and NP_0

The two formal dialectic systems, PP_0 and NP_0 are represented using the unified specification format introduced in [15]. This representation is part of a unified framework for representing, rapidly implementing and deploying formal dialectic systems called the Architecture for Argumentation (A4A). To facilitate this, the framework incorporates a range of general machinery for representing dialogues and dialectic systems. This machinery is then tailored to the needs of a specific dialectic system. The dialectic system itself is designed to model the interactions between participants during a particular dialogical situation. In this case PP_0 is formulated to model persuasion dialogues and NP_0 is formulated to model negotiation dialogues. PP_0 is a protocol tailored towards persuasion-type dialogues.

System Name PP_0
Turn Structure = ⟨Determinative, Single-Move⟩
Participants = {init, resp}
Artifact Stores :
⟨CStore, init, Mixed, Set, Light, Global⟩
⟨CStore, resp, Mixed, Set, Light, Global⟩
Global Rules :

Initiation
Requirements:
$T_{current} = 0$
Effects:
$T^{init}_{next\_move}$ = ⟨Request, (goal)⟩
Progression
Requirements:
$S \in CStore^{init}_1 \wedge S \in CStore^{init}_{current} \wedge (S' \rightarrow S) \in CStore^{init}_{current}$
$\wedge\ T^{resp}_{last}$ = ⟨ Reject, (S) ⟩

Effects:
(System=NP_0) ∨ (System=PP_0)

Termination

Requirements:
$S \in CStore_1^{init} \wedge (S \notin CStore_{current}^{init} \vee S \in CStore_{current}^{resp}) \vee$
$T_{last\_move} = \langle Withdraw(-) \rangle$
Effects:
$Dialogue_{status}$ = complete

Moves :

⟨**Request, (S)**⟩

Requirements:
Ø
Effects:
$T_{next\_move}^{listener} = \langle$ Accept, (S) $\rangle \vee \langle$ Reject, (S) $\rangle \vee \langle$ Challenge, (S) $\rangle \wedge$
$CStore_{current}^{speaker}$ + S

⟨**Accept, (S)**⟩

Requirements:
$T_{last\_move}^{listener} = \langle$ Request, (S) $\rangle$
Effects:
$CStore_{current}^{speaker}$ + S ∧ $CStore_{current}^{speaker}$ – ¬S

⟨**Reject, (S)**⟩

Requirements:
$T_{last\_move}^{listener} = \langle$ Request, (S) $\rangle$
Effects:
$T_{next\_move}^{listener} = \langle$ Challenge, (S) $\rangle \vee \langle$ Withdraw, (–) $\rangle \wedge$
$CStore_{current}^{speaker}$ + ¬S ∧ $CStore_{current}^{speaker}$ – S

⟨**Challenge, (S)**⟩

Requirements:
$T_{last\_move}^{listener} = \langle$ Request, (S) $\rangle \vee \langle$ Reject, (S) $\rangle \vee \langle$ Defense, (S′→S) $\rangle$
Effects:
$T_{next\_move}^{listener} = \langle$ Defense, (S′→S) $\rangle \vee \langle$ Reject, (S) $\rangle \vee \langle$ Withdraw, (–) $\rangle$

⟨**Defense, (S′→S)**⟩

Requirements:
Ø
Effects:
$T_{next_move}^{listener} = \langle$Challenge, (S)$\rangle \vee \langle$Challenge, (S′)$\rangle \vee \langle$Challenge, (S′→S)$\rangle \vee$
$\langle$reject, (S′→S)$\rangle \vee \langle$reject, (S)$\rangle \vee \langle$reject, (S′)$\rangle \vee$
$\langle$accept, (S′→S)$\rangle \vee \langle$accept, (S)$\rangle \vee \langle$accept, (S′)$\rangle$
$CStore_{current}^{speaker}$ + S ∧ $CStore_{current}^{speaker}$ + S′ ∧ $CStore_{current}^{speaker}$ + S′→S

⟨**Withdraw, (–)**⟩

Requirements:
$T_{last\_move} = \langle Challenge(S) \rangle \vee \langle Reject(S) \rangle$
Effects:
Ø

PP_0 enables two players named *init* and *resp* to engage in a persuasion dialogue. Players can make one move per turn, starting with init. The turn structure means that turns proceed automatically, after one player makes their move, the next player has their

turn and so on, such that it can be seen from examination of the current turn index which players move it is. The actual moves that are played cannot influence which player is assigned the speaker role in the next turn and thus cannot influence whose turn it is. Each player is assigned an artifact store named CStore. The remaining parameters specify that the store can contain a mixture of commitment types, for example a player can incur commitment to just the content of a move or to the entire move, that the store is a light side store [16] which stores a set of commitments and that the stores are to be shared between sub-dialogues of differing types. PP_0 incorporates three types of global rule. These rules specify the requirements for starting a new instance of a PP_0 sub-dialogue, the requirements for initiating a progression from an instance of a PP_0 sub-dialogue to a new instance of another sub-dialogue type, and the conditions for terminating a PP_0 dialogue.

When a new sub-dialogue of type PP_0 is begun, the initiation rules require only that the very next move, in this case the first move of the new sub-dialogue, must be a request. For a progression to be legal it is required that the player who played the first move of the PP_0 instance still be committed to their initial thesis, that *init* has expressed at least one argument in support of their initial thesis, and that the last move played in the immediate previous turn was a rejection of that initial thesis by the respondent. These conditions establish that a progression is legal at this point in the dialogue, and that the next move may be from the set of moves allocated to the NP_0 system. The current player may elect to continue in the current dialogue without progressing to another dialectic system. For example, the progression rules of PP_0 only establish that a transition is legal, not that it must occur. To actually initiate a progression at this point requires the player to make a legal move from the NP_0 move set according to the initiation rules for NP_0.

PP_0 allows six distinct moves. Each move specification incorporates a formulation of requirements for when the move is legal, and a formulation of effects that must be applied when the move is played. The request move is an utterance of the form “S?", and has no requirements. The effects of playing the request move are that the content of the move is added to the speaker’s commitment store and that the legal responses are the accept, reject and challenge moves. The accept move enables a player to agree to a request and is of the form “OK S". Conversely the reject move enables a player to disagree with a request and is of the form “Not S". The challenge move is formulated to enable a player to get justification for a previous request, reject or defense move and is of the form “why S?". The defense move enables a player to defend their challenged position by providing a supporting statement of grounds and by stating an inferential link between the challenged position and the justifying statement. The withdraw move is essentially an utterance of the form “I withdraw from this dialogue", and the rationale is to allow either player the opportunity to withdraw from the dialogue. If either player determines that the dialogue is unlikely to end successfully then it is more computationally efficient to leave the dialogue cleanly at the first subsequent opportunity rather than continue.

PP_0 only allows a player to manipulate the contents of their own commitment store and does not allow a player to incur commitments in their opponents commitment store. This is achieved through the formulation of effects for each move which only update the commitment store of the speaker. The only moves which incorporate a commitment effect are the request, accept, reject and defense moves. The challenge move does not incorporate a commitment effect, like the commitment to challenges of DC [2], but rather allows the receiver of the challenge to immediately withdraw from the dialogue without

penalty. This enables the participants to produce a number of different justifications in response to a challenge by engaging in several iterations of the challenge-defense sequence. This enables some tactical play to emerge in PP_0 persuasion dialogue whereby a player can repeatedly challenge a statement to uncover the underlying justifications for that statement, but if the player is too persistent then their opponent may choose to withdraw from the dialogue entirely. To avoid withdrawal, it is incumbent upon the challenging player to determine when they are unlikely to be able to persuade their opponent and may have more success engaging in a negotiation dialogue instead. As established earlier, the progression rules set out only when it is legal to transition to a new sub-dialogue, not that that transition must occur.

A progression is only legal, at the very earliest, after a request has been made, an argument in support of the request has been made, and the request has still been rejected by the respondent. It is only in the event that the initiator has no argument to justify their position and must make an offer in lieu of a defence or withdraw from the dialogue, that it is in the initiators interests to move straight to a negotiation dialogue. The progression rules enable the initiator to avoid the kind of fallacy of bargaining attributed to the minister of finance in the Walton and Krabbe example because the initiator has provided a defense of their initial thesis thereby discharging the burden of proof required to satisfy the persuasion dialogue and thereby avoid an illicit shift to a negotiation dialogue.

NP_0 is a protocol tailored towards negotiation-type dialogues. PP_0 is aimed at persuading a player to accept a request through successive rounds of challenge and justification. This type of dialogue requires that arguments be brought to bear which hold direct relations to the issue in question. For example, it is assumed that the defense of a challenged request lends at least some support to the request which was challenged in the first place. Likewise, an argument that is extended in defense of a request should provide relevant support for why that request should be accepted. In a negotiation the players may make offers formulated to win acceptance of their goal from their opponent. The offers however need not pertain directly to the goal. Walton and Krabbe recognise in [3] that the swapping of one concession for another is a characteristic of negotiation. In the context of a multiagent system implementation, the agents may have many different capabilities, many of which are not pertinent to the issue at hand but which may be offered as part of a deal in order to get the goal accepted. This kind of dialogue is characterised by offer-counter offer sequences. The rules of NP_0 are as follows;

System Name NP_0
Turn Structure = ⟨Determinative, Single-Move⟩
Participants = {init, resp}
Artifact Stores :
⟨CStore, init, Mixed, Set, Light, Global⟩
⟨CStore, resp, Mixed, Set, Light, Global⟩

Global Rules

Initiation
Requirements:
$S \in CStore_1^{init} \wedge S \in CStore_{current}^{init} \wedge S \notin CStore_{current}^{resp}$
Effects:
$T_{next\_move}^{speaker} = \langle Offer, (S, proposal)\rangle$
Termination
Requirements:

$S \in CStore_1^{init} \wedge (S \notin CStore_{current}^{init} \vee S \in CStore_{current}^{resp}) \vee$
$T_{last\_move} = \langle Withdraw(-) \rangle$
Effects:
$Dialogue_{status}$ = complete

Moves

$\langle$**Offer, (goal, proposal)**$\rangle$
Requirements:
$\langle Offer, (goal, proposal) \rangle \notin CStore_{current}^{speaker}$
Effects:
$(T_{next\_move}^{listener} = \langle Accept, (proposal) \rangle \vee \langle Reject, (proposal) \rangle \vee \langle Offer, (goal, proposal') \rangle$
$\vee \langle Offer, (goal', proposal) \rangle \vee \langle Offer, (goal', proposal') \rangle \vee \langle Withdraw, (-) \rangle) \wedge$
$CStore^{speaker}$ + goal $\wedge$
$CStore^{speaker}$ + proposal $\wedge$
$CStore^{speaker}$ + offer(goal, proposal)

$\langle$**Accept, (goal, proposal)**$\rangle$
Requirements:
$T_{last\_move}^{listener} = \langle Offer, (goal, proposal) \rangle$
Effects:
$CStore^{speaker}$ + goal $\wedge$
$CStore^{speaker}$ + proposal $\wedge$
$CStore^{speaker}$ + offer(goal, proposal)

$\langle$**Reject, (goal, proposal)**$\rangle$
Requirements:
$T_{last\_move}^{Hearer} = \langle Offer, (goal, proposal) \rangle$
Effects:
$(T_{next\_move}^{listener} = \langle Offer, (goal, proposal') \rangle \vee \langle Offer, (goal', proposal) \rangle \vee \langle Offer, (goal', proposal') \rangle$

$\langle$**Withdraw, (–)**$\rangle$
Requirements:
$T_{last\_move} = \langle Offer(goal, proposal) \rangle \vee \langle Reject(goal, proposal) \rangle$
Effects:
Ø

The initial setup for an NP_0 dialogue is similar to that for a PP_0 dialogue. Both systems utilise the same number and types of commitment store, the contents of which are preserved between progressions from one sub-dialogue to another. Both players retain their participant identifiers in an NP_0 sub-dialogue, that were established in the preceding PP_0 sub-dialogue, due to the formulation of progression rules. Only the initiator of the PP_0 dialogue is able to initiate a new NP_0 dialogue and hence retains their *init* identifier throughout both sub-dialogues. The similar setups are necessary to enable a clean progression from one sub-dialogue to the next, and a possible subsequent return to the original dialogue type. This approach also enables a consistent representation of supporting machinery between the two systems as required by the A4A.

The global rules for NP_0 specify initiation and termination rules. The initiation rules establish that the initiator has some initial thesis in their commitment store and that that same initial thesis is not present in the respondent's commitment store. The initiation rules also establish that an NP_0 dialogue must begin with an offer move in which the initiator states the goal that they are trying to achieve, in this case the goal is actually the

initial thesis which was established at the very beginning of the encompassing persuasion dialogue, along with a proposal that they are willing to concede to get the goal accepted. An NP_0 dialogue can terminate when either the initiator has withdrawn their initial thesis, or the respondent has accepted the initial thesis, or the withdraw move is uttered.

NP_0 incorporates four moves which enable basic bargaining behaviour. The offer move, in the context of a negotiation over action, can be assumed to have the following form, "If you accept X, I will concede Y", where X is some goal that the offerer wants the offeree to achieve and Y is the concession that the offerer is willing to make to achieve X. The offer move requires that the speaker has not previously made the same bid. In the case above, all of X, Y, and the utterance *offer(X, Y)* will be added to the speakers commitment store, so NP_0 allows commitment to offers as well as commitment with respect to the individual statements that comprise the offers. The requirements for this move stop the speaker from repeating a bid that they have already offered.

The offer move can be followed in a subsequent turn by a counter offer. NP_0 recognises four varieties of offer. The first is the initial offer in a negotiation. The remainder are various types of counteroffer in which either, the goal remains the same and the proposal is altered, the goal is altered and the proposal remains the same, or the goal and the proposal are both altered. In the two instances of counteroffers where the goal is altered, it is assumed that the goal is a reduced or related version of the initial goal but the rules do not enforce this. Given the initial offer, "If you accept X, I will concede Y", it should be noted that in the counter-offers the participants are inverted so that the offer should be read as the inversion of the previous offer; for example the first variety of counteroffer is of the form, "I will accept X, If you concede Y′", the second variety is of the form, "I will accept X′, if you concede Y", and lastly the final type of counteroffer is of the form, "I will accept X′, If you concede Y′". Notice that because NP_0 dialogues are not entirely symmetrical it is always the case that the goal refers to something that the initiator wants the respondent to accept and that the proposal refers to something that the initiator is conceding. After an initial offer is made the next move can be either outright acceptance or rejection of the offer, or one of the varieties of counteroffer. The accept move enables a player to agree to a given offer and adds the components of the offer and the offer itself to the speakers commitment store so that a player actively commits themself to accept an offer. The reject move enables a player to not accept a proposed offer. Finally the withdraw move is similar to that for withdraw in PP_0.

It should be noted that NP_0 includes no progression rules to govern either return to the parent persuasion dialogue or to enter a new instance of persuasion or negotiation dialogue as a child of the current NP_0 dialogue. This was a purposeful omission partly to aid clarity and partly because although a nice capability it is not required to demonstrate either the use or the utility of the progression from persuasion to negotiation during a dialogue. The machinery of the A4A architecture is sufficiently flexible to enables such transitions to be specified as required either in a manner similar to that used for PP_0 or by specification of a particular move which leads to a progression as part of the effects of playing that move.

5. Example Dialogue

The following dialogue fragment illustrates the canonical embedding of an NP_0 sub-dialogue within a PP_0 dialogue. The dialogue is situated within a multiagent distributed

computation scenario. Each agent has various capabilities, tasks that it can perform. A key aspect is that no single agent knows all other agents within the system or has complete knowledge of the system. The dialogue fragment is as follows:

Turn	Player	Move	CStore^{init}	CStore^{resp}
1	init	Request(S_1)	S_1	–
2	resp	Challenge(S_1)	–	–
3	init	Defence($S_2 \rightarrow S_1$)	S_2, $S_2 \rightarrow S_1$	–
4	resp	Challenge(S_1)	–	–
5	init	Defense($S_3 \rightarrow S_1$)	S_3, $S_3 \rightarrow S_1$	–
6	resp	Reject(S_1)	–	–
7	init	Offer(S_1, S_4)	S_4, Offer(S_1, S_4)	–
8	resp	Offer(S_5, S_6)	–	S_5, S_6, Offer(S_5, S_6)
9	init	Offer(S_1, S_7)	S_7, Offer(S_1, S_7)	–
10	resp	Accept(S_1, S_7)	–	S_1, S_7, offer(S_1, S_7)

The fragment involves two agents, agent$_1$ and agent$_2$. The dialogue is initiated by agent$_1$ who becomes the initiator and requests of agent$_2$ who becomes the respondent to perform task S_1. S_1 is added to the initiator's commitment store. In turn 2 the respondent challenges the request which means that the initiator must defend the standpoint established in turn T_1. At T_3 the initiator defends their standpoint and the defence is added to the initiator's commitment store. At T_4 the respondent is not persuaded by the initiator's defence and again challenges S_1. The initiator responds at T_5 with another defence of S_1 and the initiator's commitment store is again updated. In T_6 the respondent rejects the initiator's standpoint S_1. At this point the requirements of the progression rules of PP_0 are met and a shift can legally occur from the PP_0 dialogue to an NP_0 dialogue. The initiator need not utilise this progression however. If the initiator, for some reason, still has an argument that it can use to support S_1 then the PP_0 dialogue can continue. In this case though the initiator does not have a further argument to support S_1 so takes the opportunity to shift to an NP_0 dialogue. The initiator achieves this by playing the offer move at T_7 in accordance with the initiation rules of NP_0. Following the shift to the NP_0 dialogue and the initiatory offer move, the respondent responds in T_8 with a counter-offer which includes both a different goal and a different proposal to that offered in T_7. At T_9 the initiator makes another counter-offer again involving the initiators original standpoint, but this time including a new concession S_7. the concessions extended in the offer moves may, in the context of the multiagent system scenario, correspond to particular capabilities of the participating agents who offer to perform certain actions in exchange for acceptance of the initial standpoint. At T_{10} the respondent accepts the offer extended in T_9 which incorporates the standpoint originally established in T_1. At this point the termination rules of NP_0 are met and the status of the dialogue is complete.

This fragment illustrates the use of PP_0 to engage in a persuasion dialogue followed by a shift to a negotiation dialogue when the arguments of the initiating player are rejected. This is a very useful capability because it means that once the participant's persuasive arguments are exhausted they still have techniques which can allow them to reach an agreement. Without the negotiation protocol and the mechanism for shifting from a persuasion dialogue to a negotiation dialogue the dialogue would have ended much sooner without an acceptable outcome.

6. Conclusions

In this paper a situation was characterised in which the participants in an argumentative dialogue are unable to resolve their conflict through persuasive arguments. The notion of the fallacy of bargaining was introduced as a real-world tactic that is used to get agreement whereby instead of defending their standpoint from attack, the defendent makes an offer to their challenger which involves some unrelated concession. Such a fallacy involves an illicit shift from a persuasion dialogue to a negotiation dialogue. The following proposal was made; so long as a shift is licit, i.e. that the shift is clearly and transparently made, and that the shift is not made in order to escape the burden of proof of defending a standpoint, then such a shift does not lead necessarily to a fallacy of bargaining ocurring.

Given this, then in the failed persuasion scenario the participants could shift from a persuasion dialogue to a negotiation dialogue once they ran out of arguments, either to persuade their opponent or to justify their own position. Once in the negotiation dialogue the participants could make offers to each other in relation to the original issue. Such offers, instead of involving persuasive justifications of their standpoints, involve proposing concessions that could be made which aren't necessarily related to the issue at hand. To illustrate the situation, a pair of formal dialectic systems named PP_0 and NP_0 were introduced along with a mechanism for facillitating the required dialogue shift.

The next step is to refine the formulations of PP_0 and NP_0 into PP_1 and NP_1 to enable bi-directional shifts between PP and NP dialogues as well as shifts to sub-dialogues of other types.

References

[1] C. L. Hamblin. *Fallacies*. Methuen and Co. Ltd, 1970.

[2] J. D. Mackenzie. Question begging in non-cumulative systems. *Journal Of Philosophical Logic*, 8:117–133, 1979.

[3] D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue*. SUNY series in Logic and Language. State University of New York Press, 1995.

[4] R. A. Girle. Dialogue and entrenchment. In *Proceedings Of The 6th Florida Artificial Intelligence Research Symposium*, pages 185–189, 1993.

[5] R. A. Girle. Knowledge organized and disorganized. *Proceedings of the 7th Florida Artificial Intelligence Research Symposium*, pages 198–203, 1994.

[6] R. A. Girle. Commands in dialogue logic. *Practical Reasoning: International Conference on Formal and Applied Practical Reasoning, Springer Lecture Notes in AI*, 1085:246–260, 1996.

[7] P. McBurney and S. Parsons. Dialogue games in multi-agent systems. *Informal Logic*, 22(3):257–274, 2002.

[8] T. J. M. Bench-Capon, T. Geldard, and P. H Leng. A method for the computational modelling of dialectical argument with dialogue games. *Artificial Intelligence and Law*, 8:223–254, 2000.

[9] S. Toulmin. *The Uses Of Argument*. Cambridge University Press, 1958.

[10] C. Reed. Dialogue frames in agent communication. In *Proceedings of the 3rd International Conference on Multi Agent Systems*, pages 246–253. IEEE Press, 1998.

[11] P. McBurney and S. Parsons. Agent ludens: Games for agent dialogues. In *Game-Theoretic and Decision-Theoretic Agents (GTDT 2001): Proceedings of the 2001 AAAI Spring Symposium*, 2001.

[12] K. Sycara. Persuasive argumentation in negotiation. *Theory And Decision*, 28:203–242, 1990.
[13] I. Rahwan. *Interest-based Negotiation in Multi-Agent Systems*. PhD thesis, University of Melbourne, 2004.
[14] D. N. Walton. Types of dialogue, dialectical shifts and fallacies. In *Argumentation Illuminated*, pages 133–147, 1992.
[15] S. Wells and C. Reed. Formal dialectic specification. In I. Rahwan, P. Moraitis, and C. Reed, editors, *First International Workshop on Argumentation in Multi-Agent Systems*, 2004.
[16] D. N. Walton. *Logical Dialogue-Games And Fallacies*. University Press Of America, 1984.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Justifying Actions by Accruing Arguments

Trevor J.M. Bench-Capon [a], Henry Prakken [b]
[a] *Department of Computer Science, University of Liverpool, Liverpool, UK*
[b] *Department of Information and Computing Sciences, Universiteit Utrecht & Faculty of Law, University of Groningen, The Netherlands*

Abstract. This paper offers a logical formalisation of an argument-based account of reasoning about action, taking seriously the abductive nature of this form of reasoning. The particular question addressed is *what is the best way to achieve a specified goal?* Given a set of final goals and a set of rules on the effects of actions, the formation of subgoals for a goal is formalised as the application of an inference rule corresponding to the practical syllogism well-known from practical philosophy. Positive and negative applications of the practical syllogism are then accrued as a way to capture the positive and negative side effects of an action. Positive accruals can be attacked by negative accruals and by arguments for alternative ways to achieve the same goal. Defeat relations between accrued action arguments are determined in terms of the values promoted and demoted by the actions considered in the arguments. Applying preferred semantics to the result then yields the admissible ways to achieve the desired goal.

Keywords. Practical reasoning, argumentation, choice, goals, values

1. Introduction

In this paper we will address the problem of practical reasoning, which embraces questions such as: what is the best way to achieve a given purpose? how can an action be justified? and what should be done in a given situation? Here we will focus the first two of the questions, and discuss why this approach does not answer the third.

In philosophy the centre of discussion has been the practical syllogism, originally proposed by Aristotle [1]. Modern formulations take a form such as:

PS1: Agent P wishes to realise goal G
If P performs action A, G will be realised
Therefore, P should perform A

Problems with the practical syllogism as noted by, e.g. Kenny [2] include its abductive nature, and the need to consider alternatives and negative side effects before applying it. Walton [3] treats the practical syllogism as an argument scheme: instantiating the scheme supplies a presumptive reason for A, but this instantiation is then subject to a characteristic set of critical questions, which must be answered satisfactorily if the argument is to stand and the presumption upheld. These critical questions relate to the difficulties noted by Kenny. Atkinson [4] elaborated Walton's argument scheme by distinguishing the goal

into three elements: the state of affairs brought about by the action; the features of that state of affairs which are desired; and the social end or value which make those features desirable for the agent. These distinctions extended the critical questions from Walton's four to sixteen.

In this paper we aim to develop a logical formalisation of Atkinson's account within a logic for defeasible argumentation. We aim in particular to take the abductive nature of the practical syllogism seriously; its defeasible nature will be captured by stating negative answers to critical questions as counterarguments. A key ingredient in our formalisation is the use of [5]'s accrual mechanism for arguments to deal with side effects of an action. More precisely, given a set of final goals and a set of rules on the effects of actions, the formation of subgoals is formalised as the application of an inference rule expressing a positive or negative version of the scheme PS1. Both the positive and the negative applications are then accrued to capture the positive and negative side effects of an action. Positive accruals can be attacked by negative accruals and by arguments for alternative ways to achieve the same goal. Defeat relations between accrued arguments for actions are determined in terms of the values promoted and demoted by the actions advocated by the arguments. The admissible arguments are then computed within the logic using preferred semantics: if alternative ways to achieve the same goal are admissible, an ultimate choice has to be made outside the logic.

The remainder of the paper will proceed as follows. In Section 2 we recall Atkinson's account of PS1 and identify the aspects we will formalise. In Section 3 we will give some logical preliminaries, after which we present our main contribution in Section 4. Section 5 illustrates our approach with an example of a judge who must choose an appropriate sentence in order to punish a guilty person and we end in Sections 6 and 7 with a discussion of related research and some concluding remarks.

2. Atkinson's analysis of the practical syllogism

In this section we recall Atkinson's systemization of the practical syllogism and its sixteen critical questions, and we indicate which of these critical questions will be formalised in this paper. Atkinson's version of the practical syllogism is: *in the current circumstances, action A should be performed to bring about circumstances in which goal G is achieved, as this promotes value V*. The sixteen critical questions which can be posed against this argument scheme are:

CQ1: Are the believed circumstances true?
CQ2: Assuming the circumstances, does the action have the stated consequences?
CQ3: Assuming the circumstances and that the action has the stated consequences, will the action bring about the desired goal?
CQ4: Does the goal realise the value stated?
CQ5: Are there alternative ways of realising the same consequences?
CQ6: Are there alternative ways of realising the same goal?
CQ7: Are there alternative ways of promoting the same value?
CQ8: Does doing the action have a side effect which demotes the value?

CQ9: Does doing the action have a side effect which demotes some other value?
CQ10: Does doing the action promote some other value?
CQ11: Does doing the action preclude some other action which would promote some other value?
CQ12: Are the circumstances as described possible?
CQ13: Is the action possible?
CQ14: Are the consequences as described possible?
CQ15: Can the desired goal be realised?
CQ16: Is the value indeed a legitimate value?

Addressing all these questions is beyond the scope of this paper. Five of the questions cater for differences between agents: in language (CQ12, CQ14 and CQ15); in the evaluation of states of affairs (CQ4); and in what counts as a value (CQ16). We will consider only a single agent, and so these questions do not arise.

CQ1 and CQ13 relate to the state of affairs in which the agent finds itself: CQ13 representing preconditions of the action and CQ1 preconditions for the action to have the desired effect. CQ2 on the other hand represents an undercutter of the defeasible rule that the action will achieve the goal if these preconditions are satisfied. These questions are internal to the argument deriving from the practical syllogism and can be considered answered if there is a (defeasible) proof. By embedding the practical syllogism in a general formalism for defeasible argumentation, we address these questions. In contrast, CQs5-11 all involve a separate argument, which attacks or reinforces the original argument, and so require a means of comparing arguments.

CQs5-7 concern alternatives to the proposed action. We will not consider further the distinction between state and goal: this is important only if a distinction between observable and inferred states is important. Although we will distinguish between goal and value, in the limiting case where there is a one-to-one correspondence between goals and values CQ6 and CQ7 collapse. On these assumptions, only CQ6 need be considered.

CQs8-10 all concern side effects. CQ8 and CQ9 refer to adverse side effects: for this we will require a negative form of the practical syllogism, so that we can conclude that we should refrain from an action. CQ10 refers to positive side effects and the existence of an argument here will encourage the performance of the action. CQ11 is different again in that it arises when the performance of an action achieves a goal which is incompatible with the goal which motivates some other action, thus preventing the simultaneous performance of both actions.

Questions relating to side effects (CQ8-10), positive and negative, all provide extra reasons for and against performing the action. To determine the net effect of these arguments we need to accrue them, and the use of a mechanism to allow this is a main idea of this paper. Before considering alternatives we need first to establish that the action provides a net benefit, which will determine the strength of the case for performing the action. Once the beneficial actions have been identified, the best should be chosen, and now alternatives must be considered, both alternative ways of achieving a goal (CQ6) and alternative goals (CQ11). Values are used in both comparisons. We will now present our formalisation of the argument scheme and the selected critical questions.

3. Logical preliminaries

The formalism used in this paper is based on Dung's [6] abstract approach to defeasible argumentation instantiated with a familiar tree-style approach to the structure of arguments [7, 8] and incorporating an accrual mechanism of arguments [5]. Here only the main definitions of these formalisms will be given; for the full technical details the reader is referred to the original sources.

The abstract framework of [6] assumes as input a set of unstructured arguments ordered with a binary defeat relation and defines various semantics for argument-based inference, all designating one or more conflict-free sets of arguments as so-called argument extensions. Two often-used semantics are grounded semantics, which always produces a unique extension, and preferred semantics, which produces more than one extension when a conflict between arguments cannot be resolved. In this paper we will adopt preferred semantics, since reasoning about action often involves an ultimate choice between various admissible courses of action. The basic notions of [6] that we need are defined as follows.

Definition 3.1 An *argument system* is a pair $\mathcal{H} = (\mathcal{A}, \mathcal{D})$, in which $\mathcal{A}$ is a set of *arguments* and $\mathcal{D} \subseteq \mathcal{A} \times \mathcal{A}$ is the *defeat* relationship for $\mathcal{H}$. When $(a, b) \in \mathcal{D}$ we say that a *defeats* b. For $S \subseteq \mathcal{A}$ we say that

1. $a \in \mathcal{A}$ is *acceptable w.r.t* S if for every $b \in \mathcal{A}$ that defeats a there is some $c \in S$ that defeats b.
2. S is *conflict-free* if no argument in S is defeated by an argument in S.
3. S is *admissible* if iS is conflict-free and every argument in S is acceptable w.r.t S.
4. S is a *preferred extension* of $\mathcal{H}$ if it is a $\subseteq$-maximal admissible subset of $\mathcal{A}$.
5. An argument is *justified w.r.t* $\mathcal{H}$ if it is in every preferred extension of $\mathcal{H}$.
6. An argument is *defensible w.r.t.* $\mathcal{H}$ if it is in some but not all preferred extensions of $\mathcal{H}$.

As for the *structure of arguments*, we assume they have a tree-structure where applications of *strict* and *defeasible* inference rules are chained into trees. Support relations between arguments are thus captured in the internal structure of arguments. Strict inference rules will be those of a monotonic propositional modal logic (see Section 4 below), while defeasible inference rules will be a modus ponens rule for defeasible conditionals, a rule for accrual of arguments and positive and negative versions of the practical syllogism. As for notation, all knowledge is expressed in a *logical language* $\mathcal{L}$. Strict inference rules are written as $\varphi_1, \ldots, \varphi_n \rightarrow \varphi$ and defeasible rules as $\varphi_1, \ldots, \varphi_n \rightsquigarrow \varphi$ (where each φ and φ_i is a formula of $\mathcal{L}$). For any rule r its premises and conclusion are denoted, respectively, by $prem(r)$ and $conc(r)$. Each defeasible rule $\varphi_1, \ldots, \varphi_n \rightsquigarrow \varphi$ has a possibly empty set of *undercutters*, which are inference rules with conclusion $\neg\lceil\varphi_1, \ldots, \varphi_n \rightsquigarrow \varphi\rceil$. (For any term φ from the informal metalanguage of $\mathcal{L}$ the expression $\lceil\varphi\rceil$ denotes the object-level translation of φ in $\mathcal{L}$; cf. [7].)

The logical language $\mathcal{L}$ is divided into two sublanguages $\mathcal{L}_0$ and $\mathcal{L}_1$, where $\mathcal{L}_0$ is the language of a propositional modal logic to be defined in more detail in Section 4 and $\mathcal{L}_1$ is a rule language defined on top of $\mathcal{L}_0$. More specifically, $\mathcal{L}_1$ consists of so-called *defeasible conditionals*, or *defaults* for short, of the form $\varphi \Rightarrow \psi$, where ψ is a propo-

sitional literal and φ a conjunction of propositional literals of $\mathcal{L}_0$. (Note that defeasible conditionals, which express domain-specific knowledge in the object language, are not the same as defeasible inference rules, which express domain-independent inference principles in the metalanguage.) Defaults are assumed to satisfy the following inference rule of *defeasible modus ponens*:

DMP: $\varphi, \varphi \Rightarrow \psi \rightsquigarrow \psi$

Reasoning operates on a *theory* $T = (F, D)$ where F, a consistent set of $\mathcal{L}_0$ formulas, is a set of *facts* and D is a set of defaults. *Arguments* chain inference rules into AND trees, starting with a theory (F, D). For any argument A, its *formulas*, $form(A)$, are all the nodes in A, its *premises*, $prem(A)$, are all the leaf nodes of A, its *conclusion*, $conc(A)$, is its root and its *top rule*, $top(A)$, is the rule connecting the root of A with its children. An argument A is a *subargument* of an argument B if both have the same premises and $conc(A)$ is a node in B. An argument is *strict* if all its rules are strict, otherwise it is *defeasible*. A partial preorder $\leq_{\mathcal{A}}$ on the set $\mathcal{A}$ of arguments is assumed, where $A \leq_{\mathcal{A}} B$ means that B is at least as preferred as A. Related symbols are defined and subscripts omitted as usual. The preorder is assumed to satisfy the basic requirement that whenever A is strict and B defeasible then $A > B$.

As for *conflicts between arguments*, we include Pollock's [7] two ways of defeating defeasible arguments: they can be *rebutted* with an argument for the opposite conclusion and they can be *undercut* with an argument whose conclusion is that the defeasible reason applied in the attacked argument does not apply in the given circumstances. In Section 4 we will define a third form of attack for practical arguments, to deal with alternative ways to achieve the same goal. Non-undercutting conflicts between arguments will be adjudicated in terms of preference relation on arguments that takes into account the goals and values promoted and frustrated by an action.

Our formal definition of *defeat* follows common definitions in the literature.

Definition 3.2 (Defeat) Let A be an argument and B a defeasible argument.

- A *rebuts* B if $conc(A) = \neg conc(B)$ or vice versa, and $A \not< B$
- A *undercuts* B if $conc(A) = \neg\lceil top(B)\rceil$
- A *defeats* B if A rebuts or undercuts a subargument of B.

The following useful observation holds:

Observation 3.3 For all arguments A and B and preferred extensions E:

1. if A defeats a subargument of B then A defeats B;
2. if A is a subargument of B and $A \notin E$ then $B \notin E$.

Finally, as for *accrual of arguments*, [5] explains why it is worthwhile formalising this as an inference principle. Here we just recall its formalisation. The idea is that conclusions of defeasible arguments are labelled with their premises and that various defeasible arguments for the same conclusion are accrued by a defeasible inference rule that 'delabels' their conclusions. So, for instance, defeasible modus ponens now has the following form:

DMP: $\varphi, \varphi \Rightarrow \psi \rightsquigarrow \psi^{\{\varphi, \varphi \Rightarrow \psi\}}$

In the examples below the labels will for readability often be abbreviated with, possibly indexed, letters.

Next the definitions of conflicts between arguments are adjusted such that for rebutting the opposite conclusions must either be both unlabelled or have the same labels and that undercutting attack requires that the attacking arguments have unlabelled conclusions. Then a new accrual inference rule is added to the system, of the following form (in fact, the rule is a scheme for any natural number i such that $1 \leq i \leq n$):

$\varphi^{l_1}, \ldots, \varphi^{l_n} \rightsquigarrow \varphi$ (Accrual)

This inference rule and its undercutter below are the only ones that apply to labelled formulas; all other inference rules only apply to unlabelled formulas. Also, arguments are now required to have subset-minimal sets of premises to infer their conclusion, otherwise many irrelevant arguments would enter an accrual. Finally, to ensure that all relevant reasons for a conclusion are always accrued, the following undercutter scheme is formulated for any i such that $1 \leq i \leq n$.

$\varphi^{l_1}, \ldots, \varphi^{l_n} \rightsquigarrow \neg\lceil\varphi^{l_1}, \ldots, \varphi^{l_{n-i}} \rightsquigarrow \varphi\rceil$ (Accrual-undercutter)

The latter says that when a set of reasons accrues, no proper subset accrues. This undercutter is not needed if accruing arguments cannot weaken the case for the conclusion but this does not hold for all domains. For counterexamples see [5].

4. Arguments, conflict and defeat in practical reasoning

In this section we present our main contribution, a formalisation of reasoning with the practical syllogism. First we complete the definition of the logical language and its logic. The language $\mathcal{L}_0$ is a propositional modal logic with a single modality D standing for *desire*. Occurrences of D cannot be nested. To keep things simple, we abstract from the distinctions actions vs. states, procedural vs. declarative goals and achievement vs. maintenance goals: we only assume that the propositional part of $\mathcal{L}_0$ can be divided into *controllable* and *uncontrollable* formulas. Intuitively, the truth of controllable formulas is within an agent's control, but that of uncontrollable formulas (e.g. that it is raining) is not, so that only controllable formulas can be the subject of desires. The logic of D is assumed to be of type KD. Most importantly, this means that it validates $\neg(\mathrm{D}\varphi \wedge \mathrm{D}\neg\varphi)$, so that an argument for $\mathrm{D}\varphi$ can be extended by strict reasoning into an argument for $\neg\mathrm{D}\neg\varphi$.

Again for simplicity, we impose some further syntactic restrictions. Firstly, defaults cannot contain the modality D, and the only formulas in F that may contain D are of the form $\mathrm{D}\varphi$ where φ is a propositional literal from $\mathcal{L}_0$. We call the set of all such formulas in F the *goal base* G. Note that since it is a subset of F, it is assumed consistent. At first sight this would seem to prevent conflicting desires but as we will see below, we will allow for desires that turn out to be conflicting given the course that the world has taken; such 'contingent' conflicts between desires will then be subjected to our defeasible-reasoning mechanism. Contingent desire conflicts are inevitable and so our model must account for them, but it seems irrational to have desires that conflict no matter what happens.

Secondly, defaults now take one of the following forms, where all of a, r, r' and p are propositional literals and a is a controllable formula, r and r' are uncontrollable formulas and p is any propositional literal:

(i) $a \wedge r \Rightarrow p$
(ii) $a \Rightarrow p$
(iii) $r \Rightarrow r'$

Formulas of type (i) express that realising a in circumstance r achieves p, formulas of type (ii) say the same without referring to a circumstance, and formulas of type (iii) express that one circumstance typically implies another circumstance. In (i) and (ii), if p represents a state then the conditional is a causal rule, while if p represents an action the conditional is an action abstraction rule or 'counts as rule' [9], as in 'raising one's arm at an auction counts as making a bid'. Finally, formulas of type (iii) express defeasible kowledge about the world.

Next we formulate two defeasible inference rules for practical reasoning, viz. a positive and negative instance of the practical syllogism. Informally, if an agent who desires p and believes r also believes that realising a in circumstance r realises p, then this is a reason for desiring a, while if the agent believes that realising a in circumstance r instead realises $\neg p$, then this is a reason not to desire a. Note that thus practical and epistemic reasoning are interleaved, since r must be derived by epistemic reasoning. The new inference rules have the following form:

PPS: $a \wedge r \Rightarrow p, \mathrm{D}p, r \rightsquigarrow \mathrm{D}a$
NPS: $a \wedge r \Rightarrow \neg p, \mathrm{D}p, r \rightsquigarrow \neg \mathrm{D}a$

Applications of PPS can be rebutted as usual, for instance, by applications of NPS, but they can also be attacked by alternative applications of PPS to the same goal. In fact, the definition of alternatives attack is more complex than this, to deal with accrual of PPS applications to different goals.

Definition 4.1 Let A and B be two arguments.

1. A is an *alternative to* B if
   (a) $conc(A) = \mathrm{D}\varphi$ and $conc(B) = \mathrm{D}\psi$ $(\varphi \neq \psi)$; and
   (b) the last inferences in A, respectively, B apply the accrual inference rule to formulas $\mathrm{D}\varphi^{l_1}, \ldots, \mathrm{D}\varphi^{l_j}$, respectively, $\mathrm{D}\psi^{l_k}, \ldots, \mathrm{D}\psi^{l_n}$, such that:
      i. each such formula is the conclusion of a PPS application; and
      ii. at least one such PPS application in A shares a premise $\mathrm{D}\chi$ with at least one such a PPS application in B.
2. Argument A is a *sufficient alternative* to argument B if A is an alternative to B and $A \not< B$.
3. A *defeats* B if A rebuts, undercuts or is a sufficient alternative to a subargument of B.

In this paper we assume for simplicity that goals are neither already achieved nor already prevented. This assumption could be relaxed by providing undercutters of PPS and NPS in terms of what can be concluded about whether p and a hold.

The next thing to address is the preference ordering on arguments. Following [10] we first formally define the notion of a value promoted by a goal.

Definition 4.2 Let V be a set of *values* ordered by a partial preorder $\leq_V$. The function v assigns to each formula $\mathrm{D}\varphi$ a, possibly empty, subset of V of values *promoted* by $\mathrm{D}\varphi$.

We allow that $\mathrm{D}\varphi \in V$ so that as a limiting case each goal just promotes itself. Note that this ranking of values may not only differ from agent to agent, but will also be dependent on the context in which the agent is reasoning. That this is how it should be is clear from [9], where it is persuasively argued that orderings of values often emerges from the reasoning process rather than being an input to it. In particular, when considering the question of the best way to achieve a particular goal, the value promoted by that goal must be given overriding importance, since the context presupposes that the decision to achieve that goal has already been taken, and that goal must be achieved if the question is to be answered. In other contexts, when considering how best to promote other goals, the values promoted by those other goals will take on greater importance. Now the idea is that the preference relation between conflicting practical arguments is determined by the sets of values promoted and demoted by the actions considered in the arguments, where an action demotes a value if it prevents the achievement of a goal promoting it. Alternative arguments will be compared by comparing pairs of sets: for each argument the pair contains the sets of values promoted, respectively demoted, by the argument.

As for notation, for any argument A and $\mathcal{L}_0$ formula φ, the *epistemic closure* $e(A, \varphi)$ of A under φ is the set of all propositional formulas that can be derived from $\mathit{prem}(A) \cup \{\varphi\}$ with epistemic reasoning, i.e., by using only strict inference rules and Defeasible Modus Ponens.

Definition 4.3 For any argument A with conclusion $\mathrm{D}\varphi$ or $\neg\mathrm{D}\varphi$ the pair $v(A) = (p_A, d_A)$ of *values promoted and demoted by* A is defined as follows.

1. If *conc*(A) = $\mathrm{D}\varphi$ then
   (a) $p_A = \{v \in V \mid v \in v(\mathrm{D}\psi)$ for some $\mathrm{D}\psi \in \mathit{form}(A)$ such that $\psi \in e(A, \varphi)\}$
   (b) $d_A = \{v \in V \mid v \in v(\mathrm{D}\psi)$ for some $\mathrm{D}\psi$ such that there exists an argument B with conclusion $\neg\mathrm{D}\varphi$ and $\mathrm{D}\psi \in \mathit{form}(B)$ and $\neg\psi \in e(B, \varphi)\}$
2. If *conc*(A) = $\neg\mathrm{D}\varphi$ then if $A_1, \ldots, A_n$ are all maximal proper subarguments of A for which $v(A_i)$ is defined $(1 \leq i \leq n)$ then
   (a) $p_A = p_{A_1} \cup \ldots p_{A_n}$
   (b) $d_A = d_{A_1} \cup \ldots d_{A_n}$

Let E be the set of all pairs (p_A, d_A) thus defined. Then $\leq_E$ is a partial preorder on E.

In clause (1), the function p_A simply collects A's initial goal and the goals derived from it using PPS, while d_A collects all initial and derivable goals that are prevented if A's final desire is carried out. To find these prevented goals, d_A looks at all rebuttals of A and computes their epistemic closures under A's final desire. The rationale of clause (2) is that in our setting the only ways to derive a conclusion of the form $\neg\mathrm{D}\varphi$ are to derive it from a positive desire by either NPS or $\mathrm{D}\neg\varphi \rightarrow \neg\mathrm{D}\varphi$. In other words, a negative desire always 'protects' a positive desire so that it seems reasonable that they have the same sets of promoted and demoted values.

We now impose the following constraint on the argument ordering $\leq_{\mathcal{A}}$. Let A be a defeasible argument with conclusion $\mathrm{D}\varphi$ and B a defeasible argument with conclusion $\mathrm{D}\psi$ or $\neg\mathrm{D}\varphi$. Then:

- $A \leq_{\mathcal{A}} B$ iff $v(A) \leq_E v(B)$

The idea now is that $\leq_E$ is defined in terms of $\leq_V$. Clearly, many reasonable definitions are possible and a discussion of them is beyond the scope of this paper; see [11] for some related definitions.

5. An example

In this section we illustrate our formalism with an example of a judge who must determine the best way to punish (pu) a criminal found guilty. He has three options: imprisonment (pr), a fine (fi) and community service (cs). Besides punishment there are three more goals at stake, deterring the general public (de), rehabilitating the offender (re) and protecting society from crime (pt). The judge must ensure that the offender is punished, and so pu will be the most important goal, but the method of punishment chosen will depend on the other goals that can be achieved by the various methods of punishing the offender. The judge believes that imprisonment promotes both deterrence and protection of society, while it demotes rehabilitation of the offender. He believes that a fine promotes deterrence but has no effect on rehabilitation or the protection of society since the offender would remain free, and he believes that community service has a positive effect on rehabilitation of the offender but a negative effect on deterrence since this punishment is not feared. This gives (with all $\mathcal{L}_0$ formulas controllable):

$pr \Rightarrow pu$	$pr \Rightarrow de$	$fi \Rightarrow de$	$cs \Rightarrow \neg\, de$
$fi \Rightarrow pu$	$pr \Rightarrow pt$		
$cs \Rightarrow pu$	$pr \Rightarrow \neg\, re$		$cs \Rightarrow re$

Finally, the judge's goal base $G = \{\mathrm{D}pu, \mathrm{D}pt, \mathrm{D}de, \mathrm{D}re\}$. These goals just promote themselves while no other goal promotes anything: in other words, the three possible sentences are purely instrumental in achieving goals in G.

The relevant arguments are depicted in Figures 1, 2 and 3. Assuming an equality

$$\dfrac{\dfrac{\dfrac{pr \Rightarrow pu \quad \mathrm{D}pu}{\mathrm{D}pr^{l_1}} \quad \dfrac{pr \Rightarrow de \quad \mathrm{D}de}{\mathrm{D}pr^{l_2}} \quad \dfrac{pr \Rightarrow pt \quad \mathrm{D}pt}{\mathrm{D}pr^{l_3}}}{\mathrm{D}pr}}{Pr^{+}:} \qquad \dfrac{\dfrac{pr \Rightarrow \neg re \quad \mathrm{D}re}{\neg \mathrm{D}pr^{l_4}}}{\neg \mathrm{D}pr} \; Pr^{-}:$$

Figure 1. Accruals concerning imprisonment

$$\dfrac{\dfrac{fi \Rightarrow pu \quad \mathrm{D}pu}{\mathrm{D}fi^{l_5}} \quad \dfrac{fi \Rightarrow de \quad \mathrm{D}de}{\mathrm{D}fi^{l_6}}}{\mathrm{D}fi} \; Fi^{+}:$$

Figure 2. Accrual concerning fining

$$\dfrac{\dfrac{cs \Rightarrow pu \quad \mathrm{D}pu}{\mathrm{D}cs^{l_7}} \qquad \dfrac{cs \Rightarrow re \quad \mathrm{D}re}{\mathrm{D}cs^{l_8}}}{\mathrm{D}cs} \qquad \dfrac{cs \Rightarrow \neg de \quad \mathrm{D}de}{\dfrac{\neg \mathrm{D}cs^{l_9}}{\neg \mathrm{D}cs}}$$

Figure 3. Accruals concerning community service

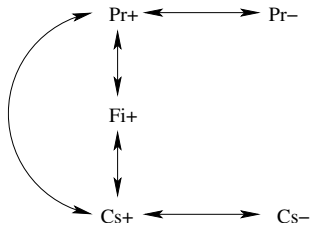


Figure 4. Partial defeat graph

argument ordering for the moment and ignoring subarguments this induces the defeat graph of Figure 4:

To adjudicate these conflicts, we must consider the values promoted and demoted by these arguments. We have that

$$\begin{array}{llll} v(Pr^+) = & (\{pu, de, pt\}, \{re\}) & v(Pr^-) = & (\{re\}, \emptyset) \\ v(Fi^+) = & (\{pu, de\}, \emptyset) & & \\ v(Cs^+) = & (\{pu, re\}, \{de\}) & v(Cs^-) = & (\{de\}, \emptyset) \end{array}$$

Recall that our question is *what is the best way to punish the offender?* We make pu an essential value, able to defeat any combination of other values, since no action that does not promote it can be an answer. This is enough to ensure that Pr^+ defeats Pr^- and Cs^+ defeats Cs^-. This leaves us with three ways to achieve our goal. Suppose that next to punishment we desire rehabilitation, and that promoting this is considered to be more important than deterrence and protection put together. Now Cs^+ will defeat Pr^+. Next we must consider whether promoting rehabilitation while demoting deterrence is preferable to promoting deterrence. If we think it is, we will accept Cs^+: if not we will accept Fi^+; and if we have no preference we will have two preferred extensions, and the choice of action must be made outside of this reasoning system. Suppose we in fact choose promoting rehabilitation while demoting deterrence over promoting deterrence: that will mean that community service is our best way to achieve punishment. The justification for our choice will then be that given that we must punish the offender, we choose to do so in a way which will aid his rehabilitation.

We cannot now, however, go on to pose the question of what is our best *set* of actions in the situation. The problem is that both the actions of sending to prison and levying a fine have had the argument for them defeated because they are (given our preference for rehabilitation) inferior alternatives to community service with respect to punishment. But if these actions were compatible with community service we might wish to perform them for the sake of their other effects. We do not, however, have any undefeated arguments to justify this. We could, of course, develop a fresh set of arguments relating to the situation where community service is performed and its goals achieved, and use this

new framework to find the best way to achieve some other goal. Such a process would, however, be dependent on the order in which goals were focussed on, and so would not provide a good answer to this question. This identifies a limitation in our approach, which we will need to address in future work.

Finally, we briefly illustrate the interleaving in our approach of practical and epistemic reasoning. Consider a refinement of the rule that community service achieves rehabilitation with a noncontrollable condition that the offender is motivated:

$$cs \wedge mo \Rightarrow re$$

The condition *mo* must now hold to make PPS applicable; this gives rise to epistemic defeasible reasoning, where the new argument for $\mathrm{D}cs$ may be defeated because the subargument for *mo* is rebutted or undercut.

6. Related work

Because of space constraints we can only briefly discuss related work.

Thomason [12], Broersen et al. [13] and van Riemsdijk et al. [14] formalise defeasible reasoning about action using default logic as a way to deal with conflicting desires. They do not formalise abductive goal generation.

Pollock [15] argues that epistemic reasoning and planning should be interleaved and models this in his OSCAR system, adopting an abductive notion of goal regression. While we especially focus on choosing an action to achieve a particular goal, Pollock's focus is more on reasoning about plans for carrying out certain actions.

Most closely related to our work is Amgoud [11], who presents a unified model of argument-based inference and decision making within the same general framework adopted by us. Her counterpart to our positive and negative form of subgoal generation is a division of the goal base into goals to achieve and goals to avoid. Abductive goal generation is allowed but cannot be chained. Also, conflicts between alternatives do not arise in the logic but are subject to a separate decision-making process in which the logically justified action arguments are further compared. Amgoud's approach also allows for 'modus ponens' generation of subgoals applied to conditional desires. Since we allow for arbitrary chains of abductive goal generation, introducing conditional desires is not trivial in our case, for which reason we leave this for future study.

7. Conclusion

In this paper we have formalised a philosophically plausible approach to practical reasoning as defeasible argumentation, to address the question of what is the best way to achieve some particular goal. We have especially focussed on the abductive nature of reasoning about desires on the basis of beliefs and goals and on the accrual of positive and negative side effects of actions. Having said this, much future work remains. The restriction to contexts in which a goal to achieve has already been selected needs to relaxed. We also need to study extension to conjunctive desires, as well as refinements of the logical language to distinguish between actions and states, declarative and procedural goals, and achievement and maintenance goals. Finally, we should explore the various ways in which value orderings influence the comparison of arguments.

Acknowledgements

This work was partially supported by the EU under IST-FP6-002307 (ASPIC). Discussions within the ASPIC project have been very helpful, especially with Leila Amgoud, Peter McBurney and Sanjay Modgil.

References

[1] Aristotle. *Topics*. Clarendon Press, Oxford, 1997. Translated by R. Smith.

[2] A.J.P. Kenny. Practical reasoning and rational appetite. In J. Raz, editor, *Practical Reasoning*, pages 63–80. Oxford University Press, Oxford, 1978.

[3] D.N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, 1996.

[4] K. Atkinson. *What Should We Do?: Computational Representation of Persuasive Argument in Practical Reasoning*. PhD Thesis, Department of Computer Science, University of Liverpool, Liverpool, UK, 2005.

[5] H. Prakken. A study of accrual of arguments, with applications to evidential reasoning. In *Proceedings of the Tenth International Conference on Artificial Intelligence and Law*, pages 85–94, New York, 2005. ACM Press.

[6] 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.

[7] J.L. Pollock. *Cognitive Carpentry. A Blueprint for How to Build a Person*. MIT Press, Cambridge, MA, 1995.

[8] G.A.W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90:225–279, 1997.

[9] J.R. Searle. *Rationality in Action*. MIT Press, Cambridge, MA, 2001.

[10] T.J.M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13:429–448, 2003.

[11] L. Amgoud. A unified setting for inference and decision: an argumentation-based approach. In *Proceedings of the IJCAI-2005 Workshop on Computational Models of Natural Argument*, pages 40–43, 2005.

[12] R.H. Thomason. Desires and defaults: a framework for planning with inferred goals. In *Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning*, pages 702–713, San Fransisco, CA, 2000. Morgan Kaufmann Publishers.

[13] J. Broersen, M. Dastani, J. Hulstijn, and L. van der Torre. Goal generation in the BOID architecture. *Cognitive Science Quarterly Journal*, 2:428–447, 2002.

[14] M.B. van Riemsdijk, M. Dastani, and J.-J. Ch. Meyer. Semantics of declarative goals in agent programming. In *Proceedings of the Fourth International Conference on Autonomous Agents and Multiagent Systems (AAMAS-05)*, pages 133–140, 2005.

[15] J.L. Pollock. The logical foundations of goal-regression planning in autonomous agents. *Artificial Intelligence*, 106:267–335, 1998.

Argumentation Frameworks and Schemes II

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Handling controversial arguments in bipolar argumentation systems

Claudette Cayrol [a], Caroline Devred [b] and Marie-Christine Lagasquie-Schiex [a]
[a] *IRIT, UPS, 31062 Toulouse, Cédex 9, {ccayrol, lagasq}@irit.fr*
[b] *CRIL, Université d'Artois, 62307 Lens, Cédex, devred@cril.univ-artois.fr*

Abstract. We consider bipolar argumentation frameworks, which extend Dung's argumentation frameworks by handling two independent kinds of interaction between arguments, attack and support. In this bipolar context, we propose new semantics for coping with the problem of controversial arguments (arguments which indirectly attack and indirectly defend a same argument).

Keywords. Argumentation Frameworks

1. Introduction

Argumentation has become an influential approach to model defeasible reasoning and dialogues between agents, based on the exchange of interacting arguments (see e.g. [1,2, 3,4,5]). The following illustrative example presents the arguments exchanged during the meeting of the editorial board of a newspaper.

Example 1

Arg. a**:** If we have the agreement and without censoring, the important information I on the person X must be published.
Arg. b_1**:** I concerns the prime minister X who may use the right of censoring.
Arg. c_1**:** The prime minister has resigned. So, I no longer concerns a prime minister.
Arg. d**:** The resignation will be announced officially this evening on TV Channel 1.
Arg. b_2**:** I is a private information and X does not agree for publication.
Arg. c_2**:** Any information concerning the prime minister is a public information and not a private information.
repetition of Arg. c_1 **and** d**:** ...
Arg. c_3**:** But I is of national interest, so I cannot be considered as a private information.

In most existing systems, the interaction takes the form of a conflict, usually called attack. For example, an argument can be a pair (set of assumptions, conclusion), where the set of assumptions entails the conclusion according to some logical inference schema. Then, a conflict occurs between two arguments for instance if the conclusion of one of them contradicts an assumption of the other one. In Example 1, b_1 (resp. b_2) is in conflict with a. The main issue of any argumentation system is the selection of acceptable sets of arguments, based on the way arguments interact. Intuitively, an acceptable set of arguments must be in some sense coherent and strong enough (e.g. able to defend

itself against all attacking arguments). The concept of acceptability has been explored through the use of argumentation frameworks, such as the fruitful Dung's argumentation framework [6], abstracting from the nature of the arguments. In such an abstract framework, from a set of arguments and a binary "attacks" relation, different semantics for acceptability are proposed, each one being characterized by several requirements that a *set of arguments* must satisfy in order to be selected. These selected sets of arguments are called extensions.

However, Dung's semantics do not always lead to expected conclusions, faced with the so-called controversial arguments. Roughly speaking, an argument c is *controversial w.r.t. an argument* a iff c indirectly defends a (e.g. c attacks an attacker of a) and also indirectly defends an attacker of a. Intuitively, even there is no direct conflict between c and a, it seems uncautious to accept together both arguments. In Example 1, c_1 defends a (against b_1) and also defends b_2 which is an attacker of a. So, c_1 is controversial w.r.t. a. In some sense, c_1 reinstates an attacker of a. That's why we find uncautious to accept both c_1 and a in the *same extension* because we are interested in the definition of "coherent" *sets of arguments*[1] (the simplest notion of "coherent" set proposed by Dung is the notion of conflict-free set). Moreover, since c_1 is the unique defender of a against b_1, it seems also uncautious to derive a from the discussion. However, whatever Dung's semantics, the unique extension contains a, c_1 and c_3. This problem has motivated the definition of new prudent semantics by [7,8][2], in which the notion of coherence is enforced: pairs of arguments which conflict indirectly cannot belong to a same extension.

Moreover, recent work on argumentation [9,4,5] has advocated for the representation of another kind of basic interaction between arguments. Indeed, it can be the case in a dialog that an agent brings to light some new piece of information and so advances an argument which justifies an assumption used by an argument provided by another agent (agents are assumed independent).This kind of interaction between arguments is not captured by the notion of defence. It is rather a kind of *support*. In Example 1, we consider that the argument d given by an agent Ag_1 supports the argument c_1 given by another agent Ag_2. It is not only a "dialog-like speech act": a new piece of information is really given. In [9], Dung's framework has been extended to cope with both kinds of interaction, into a so-called bipolar abstract argumentation framework. *Bipolarity* refers to the existence of two independent kinds of information which represent repellent forces. New semantics for acceptability have been defined, based on a more complex notion of attack, called the supported attack. In Example 1, the fact that d supports an attacker of b_1 may be considered as a supported attack on b_1 by d. These new semantics ensure that no supported attack can occur within an extension. However, a new kind of controversial arguments appears in a bipolar argumentation framework. On Example 1, d supports an indirect attacker of a and also supports a defender of a. Even if d is not directly controversial w.r.t. a, it seems uncautious to accept d and a in the same extension. So the purpose of this paper is to propose new semantics which handle this new kind of controversial arguments in a bipolar setting.

We first present the background concerning Dung's framework, its extension for handling controversial arguments and the bipolar extension (Section 2). Then, we present our contribution (Section 3). The key issue is to define a new kind of controversial arguments, the b-controversial arguments, and a notion of conflict which encompasses both

[1] which is different from the definition of the derivability of an argument.

[2] There exist some other propositions for solving this problem, see Sect. 5.

indirect attacks and supported attacks. Then, we propose an analog of the prudent semantics for a bipolar framework. Despite the apparent complexity of this new framework, most of the properties satisfied by the prudent semantics are preserved in the bipolar setting (Section 4).

2. Background

2.1. Dung's framework

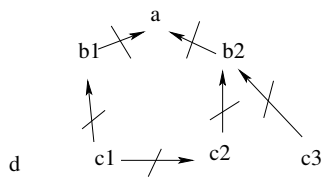
Let us present some basic definitions at work in Dung's theory of argumentation [6].

Definition 1 *A* finite argumentation framework *is a pair* $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ *where* $\mathcal{A}$ *is a finite set of so-called arguments and* $\mathcal{R}$ *is a binary relation over* $\mathcal{A}$ *(a subset of* $\mathcal{A} \times \mathcal{A}$*), the attacks relation.*

An argumentation framework can be represented by a directed graph in which each argument is a vertex and the edges are defined by the attacks relation: $\forall a, b \in \mathcal{A}$, $a\mathcal{R}b$ is represented by $a \nrightarrow b$.

Example 1 (cont)

This example can be formalized by the framework AF_1 represented by the following interaction graph (note that the argument d remains isolated because it cannot be linked to the other arguments using the attacks relation):



In the following, we consider a finite argumentation framework $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ and its associated interaction graph $\mathcal{G}$. The first important notions are the notion of acceptability and the notion of conflict which are used for defining the extensions:

Definition 2 *Let* $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ *be a finite argumentation framework.*

- *Let* $a \in \mathcal{A}$ *and* $S \subseteq \mathcal{A}$. a *is* acceptable *w.r.t.* S *iff* $\forall b \in \mathcal{A}$ *s.t.* $b\mathcal{R}a$, $\exists c \in S$ *s.t.* $c\mathcal{R}b$. *A set of arguments is acceptable w.r.t.* S *when each of its elements is acceptable w.r.t.* S.
- S *is* conflict-free *iff* $\nexists a, b \in S$, *s.t.* $a\mathcal{R}b$.
- *A subset* S *of* $\mathcal{A}$ *is* admissible *for* AF *iff* S *is conflict-free and acceptable w.r.t.* S.
- S *is* a preferred extension *of* AF *iff it is maximal w.r.t.* $\subseteq$ *among the admissible sets for* AF. S *is* a stable extension *of* AF *iff it is conflict-free for* AF *and* $\forall a \in \mathcal{A} \setminus S$, $\exists b \in S$ *s.t.* $b\mathcal{R}a$. S *is* the grounded extension *of* AF *iff it is the least element w.r.t.* $\subseteq$ *among the admissible sets s.t. each argument acceptable w.r.t.* S *belongs to* S.

In Example 1, $E_1 = \{c_1, c_3\}$ is an admissible set for AF_1 and $E_2 = \{a, c_1, c_3, d\}$ is the preferred extension, the stable extension and the grounded extension of AF_1.

Formally, the grounded extension of AF can be characterized as the least fixed point w.r.t. $\subseteq$ of the characteristic function $\mathcal{F}_{\mathsf{AF}}$.

Definition 3 *The* characteristic function*, denoted $\mathcal{F}_{\mathsf{AF}}$, of* AF *is defined by:* $\mathcal{F}_{\mathsf{AF}} : 2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$ *and* $\mathcal{F}_{\mathsf{AF}}(S) = \{a \mid a \text{ is acceptable w.r.t. } S\}$.

Dung has also identified problematical arguments, the controversial arguments.

Definition 4 *Let* $a, b \in \mathcal{A}$. a indirectly attacks b *iff* $\exists$ *an odd-length path from* a *to* b *in* $\mathcal{G}$. a indirectly defends b *iff* $\exists$ *an even-length path from* a *to* b *in* $\mathcal{G}$ *(length* ≥ 2*).*

a *is* controversial *w.r.t.* b *iff* a *indirectly attacks* b *and* a *indirectly defends* b.

In Example 1, c_1 indirectly attacks a, c_1 defends a, c_1 is controversial w.r.t. a.

2.2. Handling indirect conflict

Let $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ be a finite argumentation framework. In order to handle controversial arguments [7,8] define new semantics, the *p(rudent)* semantics. [7,8] refine Dung's notions of conflict-free and admissibility by exploiting the notion of indirect attack proposed by Dung.

Definition 5 *Let* $\mathsf{AF} = \langle \mathcal{A}, \mathcal{R} \rangle$ *be a finite argumentation framework. Let* $S \subseteq \mathcal{A}$.

- S *is* p(rudent)-admissible *for* AF *iff* S *is acceptable w.r.t.* S *and* $\forall a, b \in S$, a *doesn't indirectly attack* b.
- S *is* a preferred p-extension *of* AF *iff it is maximal w.r.t.* $\subseteq$ *among the p-admissible sets for* AF. S *is* a stable p-extension *of* AF *iff* $\forall a, b \in S$, a *doesn't indirectly attack* b *and* $\forall a \in \mathcal{A} \setminus S$, $\exists b \in S$ *s.t.* $b \mathcal{R} a$.

In Example 1, $E_1 = \{c_1, c_3\}$ is a p-admissible set for AF_1. $E_2 = \{c_1, c_3, d\}$ is the preferred p-extension of AF_1. AF_1 has no stable p-extension.

A grounded p-extension is defined using the *p-characteristic function*.

Definition 6

- *The* p-characteristic function *of* AF *is defined as follows:* $\mathcal{F}^p_{\mathsf{AF}} : 2^{\mathcal{A}} \longrightarrow 2^{\mathcal{A}}$ *and* $\mathcal{F}^p_{\mathsf{AF}}(S) = \{a \mid a \text{ is acceptable w.r.t. } S \text{ and } \forall c, b \in S \cup \{a\}, c \text{ doesn't indirectly attack } b.\}$.
- *Let* j *be the lowest integer s.t. the sequence* $(\mathcal{F}^{p,i}_{\mathsf{AF}}(\varnothing))_{i \in \mathbb{N}}$ *is stationary from rank* j. $\mathcal{F}^{p,j}_{\mathsf{AF}}(\varnothing)$ *is the* grounded p-extension *of* AF.

Contrariwise to $\mathcal{F}_{\mathsf{AF}}$, $\mathcal{F}^p_{\mathsf{AF}}$ is in general nonmonotonic w.r.t. $\subseteq$. This prevents from defining a notion of grounded p-extension as the least fixed point of $\mathcal{F}^p_{\mathsf{AF}}$. However, the sequence $(\mathcal{F}^{p,i}_{\mathsf{AF}}(\varnothing))_{i \in \mathbb{N}}$ is monotonic w.r.t. $\subseteq$. In Example 1, E_2 is the grounded p-extension of AF_1.

2.3. Bipolar argumentation frameworks

As already said, arguments may be conflicting. These conflicts are captured by the attacks relation in an argumentation framework, and may be considered as negative interactions. Then, the concept of defence has been introduced in order to reinstate some of the attacked arguments, namely those whose attackers are in turn attacked. So, most

logical theories of argumentation assume that if an argument a_3 defends an argument a_1 against an argument a_2, then a_3 is a kind of support for a_1, so a positive interaction. It holds in the basic Dung's framework: only negative interaction is explicitly represented by the *attacks* relation, and positive interaction is implicitly represented through the notion of defence. In this case, support and attack are *dependent* notions. It is a parsimonious strategy, but it is not a correct description of the process of argumentation in realistic examples: in Example 1, the link between the argument d and the other arguments cannot be expressed with the attacks relation. So, we need a more complex argumentation framework, in order to formalize situations where two *independent* kinds of interaction are available: a positive and a negative one. Following [4,5], [9,10] propose a bipolar argumentation framework. This new framework[3] is an extension of the basic Dung's framework in which a new kind of interaction between arguments is represented by the *supports* relation[4]. This framework presents the following features:

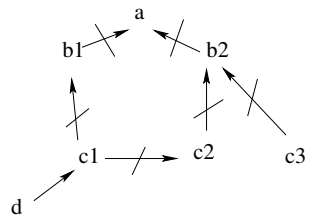
- an abstract point of view: arguments and interactions are considered as initial data;
- the existence of the support interaction is justified by the independence of the sources in a multi-agent system: different agents propose and exchange different arguments related to their own knowledge;
- the inference mechanism is based on the selection of acceptable sets of arguments and does not use a dialectical proof mechanism[5]: this selection is performed after the agents have exchanged their arguments.
- The supports relation is assumed to be totally independent of the attacks relation.

Definition 7 *A finite* bipolar argumentation framework $\langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ *consists of a finite set* $\mathcal{A}$ *of arguments, a binary relation* $\mathcal{R}_{att}$ *on* $\mathcal{A}$ *called the* attacks relation *and another binary relation* $\mathcal{R}_{sup}$ *on* $\mathcal{A}$ *called the* supports relation.

In the following, we consider a finite bipolar argumentation framework $\mathsf{BAF} = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$. Note that BAF can still be represented by a directed graph $\mathcal{G}_b$ called the *bipolar interaction graph* with two kinds of edges, one for the attacks relation and another one for the supports relation. Consider $a, b \in \mathcal{A}$, $a\mathcal{R}_{att}b$ is represented by $a \not\rightarrow b$ and $a\mathcal{R}_{sup}b$ is represented by $a \rightarrow b$.

Example 1 (cont)

The whole discussion during the editorial board meeting can now be formalized by the bipolar framework BAF_1 represented by:



The fact that d supports an attacker of b_1 may be considered as a kind of negative interaction between d and b_1, which is however weaker than a direct attack. From a

[3]The bipolar argumentation framework presented here is a simplified version of what has been discussed in [9,10,11].

[4]If the support relation is removed, we retrieve Dung's framework.

[5]Even if there exist links between the selected acceptable sets and some particular dialectical proofs – see [12,13].

cautious point of view, such arguments cannot appear together in a same extension. In order to address this problem, [9,10] introduce a new kind of attack which combines a sequence of supports with a direct attack.

Definition 8 *A* supported attack *for an argument b by an argument a is a sequence of supports followed by one attack: $a_1\mathcal{R}_1 \ldots \mathcal{R}_{n-1}a_n$, $n \geq 3$, with $a_1 = a$, $a_n = b$, s.t. $\forall i = 1 \ldots n-2$, $\mathcal{R}_i = \mathcal{R}_{sup}$ and $\mathcal{R}_{n-1} = \mathcal{R}_{att}$.*

In Example 1, there is a supported attack for b_1 by d.

3. Controversial arguments in a bipolar framework

In this paper, we are interested in handling the controversial arguments in a bipolar framework. Because we want to take into account the supports, some particular configurations may appear: in Example 1, d supports c_1 which is controversial w.r.t. a and it seems uncautious to accept d and a in the same extension, even if d is not directly controversial w.r.t. a. So, d and a illustrate a new kind of controversial arguments: the *b(ipolar)-controversial* arguments.

Definition 9 *Let $a, b \in \mathcal{A}$. a is* b-controversial *w.r.t. b iff a supports (by a sequence of supports) an argument x which indirectly attacks b and a supports (by a sequence of supports) an argument y which indirectly defends b.*

In this paper, we propose an analog of the p-semantics for a finite bipolar argumentation framework $\mathsf{BAF} = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ in order to manage the b-controversial arguments: if a is b-controversial w.r.t. b, a and b cannot belong to the same extension.

So, the first step consists in enforcing the notion of coherence.

Definition 10 *Let $S \subseteq \mathcal{A}$. S is* bp-conflict-free[6] *iff $\nexists a, b \in S$ s.t. there exists a sequence $a_1\mathcal{R}_{sup} \ldots \mathcal{R}_{sup}a_n\mathcal{R}_{att} \ldots \mathcal{R}_{att}a_{n+m}$, $n \geq 1$, with $a_1 = a$, $a_{n+m} = b$, and m is an odd number.*

In Example 1, $\{a, d\}$ is not bp-conflict-free (via c_1, c_2, b_2).

When the sequence of supports is empty, this notion is exactly the notion defined by [7,8] applied to the partial framework $\langle \mathcal{A}, \mathcal{R}_{att} \rangle$. A bp-conflict-free set does not contain indirect attacks.

From this notion of bp-conflict-free, and keeping the notion of acceptability (Def. 2), different semantics for the acceptability can be proposed: the *b(ipolar)p(rudent)-semantics*.

Definition 11 *Let $S \subseteq \mathcal{A}$. S is* bp-admissible *for* BAF *iff S is bp-conflict-free and acceptable w.r.t. S. S is a* preferred bp-extension *of* BAF *iff S is maximal for $\subseteq$ among the bp-admissible sets for* BAF. *S is a* stable bp-extension *of* BAF *iff S is bp-conflict-free and $\forall a \notin S$, $\exists b \in S$ s.t. $b\mathcal{R}_{att}a$.*

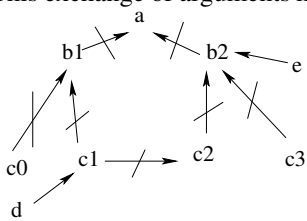
[6] bp means b(ipolar)p(rudent).

These new semantics are illustrated with the following example which is a complexification of Example 1.

Example 2 Some arguments are added to the dialog of Example 1 (arguments c_0 and e which are emphasized).

Arg. a**:** If we have the agreement and without censoring, the important information I on the person X must be published.
Arg. b_1**:** I concerns the prime minister X who may use the right of censoring.
Arg. c_0**:** *We are in democracy and even the prime minister cannot use the right of censoring.*
Arg. c_1**:** The prime minister has resigned. So, I no longer concerns a prime minister.
Arg. d**:** The resignation will be announced officially this evening on TV Channel 1.
Arg. b_2**:** I is a private information and X does not agree for publication.
Arg. e**:** *I concerns X's son which is suspected to be a spy of a foreign country.*
Arg. c_2**:** Any information concerning the prime minister is a public information and not a private information.
repetition of Arg. c_1 **and** d**:** ...
Arg. c_3**:** But I is of national interest, so I cannot be considered as a private information.

This exchange of arguments may be formalized by BAF_2 which is represented by:



$\{d, c_0, c_3, a\}$ does not contain indirect attacks but is not bp-conflict-free.
In this case, $\{a, c_0, c_3\}$ and $\{c_0, c_1, c_3, d, e\}$ are the two preferred bp-extensions of BAF_2 and BAF_2 has no stable bp-extension.

We define the bp-characteristic function as in Def. 6.

Definition 12 *The* bp-characteristic function *of* BAF *is defined as follows:* $\mathcal{F}^{bp}_{\mathsf{BAF}} : 2^{\mathcal{A}} \longrightarrow 2^{\mathcal{A}}$ *and* $\mathcal{F}^{bp}_{\mathsf{BAF}}(S) = \{a \mid a \text{ is acceptable w.r.t. } S \text{ and } S \cup \{a\} \text{ is bp-conflict-free}\}$.

The bp-characteristic function is in general nonmonotonic. An important property of the p-characteristic function fails for the bp-characteristic function: the sequence $(\mathcal{F}^{bp,i}_{\mathsf{BAF}}(\varnothing))_{i \in \mathbb{N}}$ is nonmonotonic w.r.t. $\subseteq$. So, it is not possible to define a "grounded bp-extension"[7] as in Def. 6.

4. Some properties

Let $\mathsf{BAF} = \langle \mathcal{A}, \mathcal{R}_{att}, \mathcal{R}_{sup} \rangle$ be a finite bipolar argumentation framework. AF denotes the partial argumentation framework $\langle \mathcal{A}, \mathcal{R}_{att} \rangle$. First of all, note that, when $\mathcal{R}_{sup} = \varnothing$, the bp-semantics correspond exactly to p-semantics.

The following results establish links between bp-semantics and Dung's semantics or p-semantics.

Proposition 1

[7]However, structural restrictions of BAF will allow for such a definition (it could be the subject of a future work). It is also possible to define a weaker notion of grounded extension inspired from the work of [14].

(i) *Every bp-admissible set for* BAF *is also admissible and p-admissible for* AF. *The converse does not hold.*

(ii) *Every stable bp-extension of* BAF *is also a stable extension and a stable p-extension of* AF. *The converse does not hold.*

Proof:

(i) Obvious from the definition of bp-admissible. The negative result for the converse is given by Example 2: $\{a, c_0, c_3, d\}$ is an admissible set and a p-admissible set for AF_2 , but it is not a bp-admissible set for BAF_2.

(ii) Let S be a stable bp-extension of BAF. S contains no bp-conflict and attacks each argument outside of S. So S contains no indirect attack (and S is conflict-free) in AF and attacks each argument outside of S. Accordingly, S is a stable extension of AF and a stable p-extension of AF. The negative result for the converse is illustrated by $\mathsf{BAF} = \langle \mathcal{A} = \{a, b, c, d\}, \mathcal{R}_{att} = \{(d, b), (b, a)\}$ and $\mathcal{R}_{sup} = \{(c, b)\}\rangle$: $\{c, d, a\}$ is a stable extension and a stable p-extension of AF , but it is not a stable bp-extension of BAF.

□

Basic properties of Dung's framework are preserved. Since $\varnothing$ is a bp-admissible set, a bipolar argumentation framework has always at least one preferred bp-extension. Moreover, we have :

Proposition 2 *The set of all the bp-admissible sets for* BAF *forms a complete partial order w.r.t.* $\subseteq$. *And, for each bp-admissible set* S *for* BAF, *there exists a preferred bp-extension* E *of* BAF *s.t.* $S \subseteq E$.

Proof: The set of all the bp-admissible sets for BAF has a least element w.r.t. $\subseteq$ since $\varnothing$ is bp-admissible. Since $\mathcal{A}$ is a finite set, every chain of bp-admissible sets for BAF has a least upper bound w.r.t $\subseteq$ (namely the union of these sets). So, the set of all the bp-admissible sets for BAF is a complete partial order w.r.t. $\subseteq$. The second point follows immediately from the fact that $\mathcal{A}$ is finite. □

Proposition 3 *Each stable bp-extension is a preferred bp-extension. The converse is false.*

Proof: Let S be a stable bp-extension of BAF. From Prop. 1, S is a stable extension of AF. So, S is acceptable w.r.t. S (see [6]). Moreover, S contains no bp-conflict. Accordingly, S is bp-admissible for BAF. Then, for each argument a not belonging to S, there exists a conflict in $S \cup \{a\}$, which is also a bp-conflict. So, S is maximal for $\subseteq$ among the bp-admissible sets for BAF. That is S is a preferred bp-extension of BAF.

The negative result for the converse is given by Example 2: $\{c_0, c_1, c_3, d, e\}$ is a preferred bp-extension of BAF_2, but it is not a stable bp-extension.

□

The bp-characteristic function characterizes the bp-admissible sets and the preferred bp-extensions.

Proposition 4 *Let $S \subseteq \mathcal{A}$.*

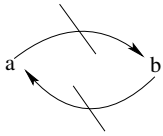
(i) *S is a bp-admissible set iff $S \subseteq \mathcal{F}^{bp}_{\mathsf{BAF}}(S)$.*

(ii) *If S is a preferred bp-extension, we have $S = \mathcal{F}^{bp}_{\mathsf{BAF}}(S)$. The converse does not hold.*

Proof: $S \subseteq \mathcal{F}^{bp}_{\mathsf{BAF}}(S)$ means that for each $a \in S$, a is acceptable w.r.t. S and $S \cup \{a\} = S$ contains no bp-conflict. It means exactly that S is bp-admissible for BAF.

Assume that S is a preferred bp-extension. It remains to prove that $\mathcal{F}^{bp}_{\mathsf{BAF}}(S) \subseteq S$. Let a be an element of $\mathcal{F}^{bp}_{\mathsf{BAF}}(S)$. a is acceptable w.r.t. S and $S \cup \{a\}$ contains no bp-conflict. So, $S \cup \{a\}$ is bp-admissible. If a does not belong to S, there is a contradiction with the maximality of S.

The negative result for the converse is given by the following example:



In this case, $\mathcal{R}_{sup} = \varnothing$. Assume that $S = \varnothing$, $\mathcal{F}^{bp}_{\mathsf{BAF}}(S) = \varnothing$. However, the two preferred bp-extensions are $\{a\}$ and $\{b\}$. □

To sum up, every finite bipolar argumentation framework has at least one preferred bp-extension, and zero, one or many stable bp-extensions. A stable bp-extension is never empty (when $\mathcal{A} \neq \varnothing$).

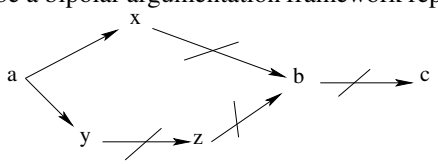
Our main purpose was to provide highly prudent semantics for which extensions cannot contain b-controversial arguments. This requirement is fulfilled, as shown by :

Proposition 5 *Let $a, b \in \mathcal{A}$. If a is b-controversial w.r.t. b then $\{a, b\}$ cannot be included in any bp-admissible set.*

Proof: If a is b-controversial w.r.t. b, there is a bp-conflict in $\{a, b\}$. □

The above result does not prevent a or b from belonging to a bp-admissible set, but not to the same one.

Example 3 Let BAF_3 be a bipolar argumentation framework represented by:



The argument a is b-controversial w.r.t. c. The bp-extensions are $\{a, x, y\}$ and $\{x, c\}$. So, no bp-admissible set contains both a and c.

As a consequence of Prop. 5, we obtain that no argument belonging to an odd-length cycle in AF can belong to a bp-admissible set. It departs from [15] who handles in the same way odd-length and even-length cycles in an argumentation framework $\langle \mathcal{A}, \mathcal{R}_{att} \rangle$.

However, some other Dung's properties are not preserved.

The set of arguments which are not attacked is included in a preferred bp-extension, but not in all of them. In Example 3, $\{a, x, y\}$ is the set of arguments which are not attacked. It is a preferred bp-extension, but there is another one $\{x, c\}$.

Unlike preferred extensions, a bipolar argumentation framework can have more than one preferred bp-extension even if it is well-founded[8] (see BAF_2). Indeed, we have:

Proposition 6 *For every preferred bp-extension, E_{bp} of* BAF*, there exists at least one preferred extension E of* AF *such that $E_{bp} \subseteq E$.*

For every preferred bp-extension, E_{bp} of BAF*, there exists at least one preferred p-extension E_p of* AF *such that $E_{bp} \subseteq E_p$.*

Proof: A preferred bp-extension is bp-admissible, and from Prop. 1, it is also admissible and p-admissible. Dung has proved that every admissible set is included in a preferred extension. And [7] has proved that every p-admissible set is included in a preferred p-extension. □

As a consequence of Prop. 6, when AF has a unique preferred extension E (for instance, when AF is well-founded, or without any even-length cycle), E includes every preferred bp-extension of BAF.

5. Related works

The work reported in this paper extends work by [7,8] on prudent semantics.

Some other works in the literature have extended Dung's semantics to cover the problem of controversial arguments.

In [16,17], the work is carried out in the specific context of argumentation in logic programming. The knowledge base is a logic program P. An argument A for a goal G is a set of negative hypotheses of the form (`not` p) such that G can be derived from P extended with A. A set of hypotheses A conflicts with another set A' when A is an argument for p with (`not` p) belonging to A'. The approach focusses on sets of hypotheses, not on sets of arguments. All the definitions for the different semantics are given for sets of hypotheses. So coherence is not defined for a set of arguments, as in our approach, but only for a set of hypotheses.

In [15], the issue is to handle in the same way odd-length and even-length cycles in an argumentation framework. It departs from our approach, since every argument belonging to an odd-length cycle is bp-controversial w.r.t. any argument of the cycle [18], no such argument can belong to a bp-extension.

[19] presents another semantics which handle the controversial arguments: the new careful semantics according to which two arguments cannot belong to the same extension whenever one indirectly attacks a third argument while the other one indirectly defends the third one (so, the controversial arguments cannot belong to the careful extensions).

As for handling support interaction, few works have been published [9,4,5]. Note that a notion of support has also appeared recently in work by [20]. This work is carried out in a specific logical framework of argumentation. The arguments are built using a knowledge base (containing a non defeasible part, called the background, and a defeasible part), classical inference and priorities between rules. Interaction between arguments is modelled by an attack relation which combines classical refutation and priorities. Then, an abductive component is added as follows. Given an available set of as-

[8] *i.e.* there does not exist an infinite sequence of attacks.

sumptions, the idea is to complete the background by some assumptions in order to be able to build new arguments. Namely, S is a supporting information for a goal G if a good argument for G can be provided from the defeasible knowledge, taking into account the background augmented with S. This kind of support is very different from the notion presented in this paper. Supporting information aims at providing new arguments, for instance for attacking old arguments. In our proposal, support occurs between arguments as a new kind of positive interaction which is essential in a multi-agent context.

6. Conclusion

Prudent semantics have been proposed in Dung's framework to cope with the problem of controversial arguments. The idea is to exclude indirect attacks in an extension. In a bipolar argumentation framework, where the presence of two kinds of interaction between arguments, conflict and support, enables to define more complex attacks, the supported attacks, the problem of controversial arguments becomes more complex because a new kind of controversial arguments appears, the b-controversial arguments. So, we have revisited the prudent semantics in a bipolar setting. We have shown that most of the properties satisfied by the preferred prudent semantics and the stable prudent semantics were preserved in the bipolar setting. However, it is no longer possible to define an equivalent for the grounded prudent semantics.

A direction for future work concerns computational issues. Algorithms described in [11] can serve as a basis for developing algorithms for computing prudent extensions in a bipolar framework. Moreover, it will be interesting to study the credulous decision problem associated with these prudent semantics: "given an argument (advanced for instance in a dialogue), is this argument in at least one prudent extension ?". For that purpose, an idea is to include heuristics into the dialectical proof theories described in [18] for the credulous decision problem.

A particular notion of attack can represent the notion of exception in nonmonotonic inheritance reasoning but it is not the only meaning[9]. Nevertheless, another direction for a future work could be the study of these particular cases.

Acknowledgements

We would like to thank the referees for their comments which helped us to improve this paper. The authors have been partly supported by the Région Nord/Pas-de-Calais, the IRCICA Consortium and the European Community FEDER Program.

References

[1] P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox. A logic of argumentation for reasoning under uncertainty, *Computational Intelligence*, 11, 113–131, 1995.

[9] In our example, the agents propose arguments using their beliefs. These beliefs either are confirmed (supported), or invalidated (attacked) by other agents' arguments.

[2] H. Prakken and G. Vreeswijk. Logics for defeaseable argumentation. In D.M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic, 2nd edition*, volume 4, pages 219–318. Kluwer Academic Publishers, Dordrecht, 2002.

[3] L. Amgoud, N. Maudet, and S. Parsons. Arguments, dialogue and negotiation. In *16th European Conf. on Artificial Intelligence (ECAI 2004)*, pages 338–342, 2004.

[4] N. Karacapilidis and D. Papadias. Computer supported argumentation and collaborative decision making: the HERMES system. *Information systems*, 26(4):259–277, 2001.

[5] B. Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation*, 13:319–346, 2003.

[6] 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):321–358, 1995.

[7] S. Coste-Marquis, C. Devred, and P. Marquis. Prudent semantics for argumentation frameworks. In *The 17th IEEE International Conf. on Tools with Artificial Intelligence (ICTAI 2005)*, pages 568–572, 2005.

[8] S. Coste-Marquis, C. Devred, and P. Marquis. Sémantiques prudentes pour les systèmes d'argumentation. In *15e congrès francophone AFRIF-AFIA Reconnaissance des Formes et Intelligence Artificielle (RFIA 2006)*, 2006.

[9] C. Cayrol and M-C. Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *8th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, volume LNAI 3571, pages 378–389, 2005.

[10] C. Cayrol and M-C. Lagasquie-Schiex. Gradual valuation for bipolar argumentation frameworks. In *8th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2005)*, volume LNAI 3571, pages 366–377, 2005.

[11] M. Mardi, C. Cayrol, and M-C. Lagasquie-Schiex. Bipolarité en argumentation : acceptabilité et algorithmes. Rapport interne IRIT 2005-20-R, IRIT-UPS, Octobre 2005.

[12] C. Cayrol, S. Doutre, and J. Mengin. Dialectical proof theories for the credulous preferred semantics of argumentation framework. In *6th European Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-2001)*, volume LNAI 2143, pages 668–679, 2001.

[13] P.E. Dunne and T.J.M. Bench-Capon. Two party immediate response disputes: Properties and efficiency. *Artificial Intelligence*, 149(2):221–250, 2003.

[14] S. Coste-Marquis, C. Devred, and P. Marquis. Contrained argumentation frameworks. In *Proc. of the 10th International Conf. on Principles of Knowledge Representation and Reasoning (KR' 06)*, pages 112–122, 2006.

[15] P. Baroni and M. Giacomin. Solving semantic problems with odd-length cycles in argumentation. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty,7th European Conf., ECSQARU 2003*, volume LNAI 2711, pages 440–451, 2003.

[16] A.C. Kakas, P. Mancarella, and P.M. Dung. The acceptability semantics for logic programs. In *Proc. of 11th International Conf. on Logic Programming (ICLP)*, pages 504–519, MIT Press, 1994.

[17] A.C. Kakas and F. Toni. Computing Argumentation in Logic Programming. *Journal of Logic and Computation*, 9(4):515–562, 1999.

[18] S. Doutre. *Autour de la sémantique préférée des systèmes d'argumentation*. PhD thesis, Université Paul Sabatier de Toulouse, 2002.

[19] S. Coste-Marquis, C. Devred, and P. Marquis. Inference from controversial arguments. In *The 12th International Conf. on Logic for Programming Artificial Intelligence and Reasoning (LPAR 2005)*, volume LNAI 3835, pages 606–620, 2005.

[20] A.C. Kakas and P. Moraïtis. Argumentation based decision making for autonomous agents. In *Proc. of International Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 883–890. ACM Press, 2003.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

On Acceptability in Abstract Argumentation Frameworks with an Extended Defeat Relation

Diego C. Martínez, Alejandro J. García and Guillermo R. Simari
Artificial Intelligence Research and Development Laboratory,
Department of Computer Science and Engineering, Universidad Nacional del Sur,
Email: {dcm,ajg,grs}@cs.uns.edu.ar

Abstract. Defeat between arguments is established by a combination of two basic elements: a conflict or defeat relation, and a preference relation on the arguments involved in this conflict. We present a new abstract framework for argumentation where two kinds of defeat are present, depending on the outcome of the preference relation: an argument may be a *proper defeater* or a *blocking defeater* of another argument. An operator is used to characterize the set of accepted arguments. This operator also provides a method for identifying controversial situations.

Keywords. Abstract argumentation, argumentation semantic, preference relation

1. Introduction

The area of Knowledge Representation and Reasoning has been enriched during the past two decades with the addition of Argument-Based Reasoning Systems [1,2,3] to mention a few. Two interesting surveys on argumentation are [4,5] and the reader is referred to them for details on the different proposals.

The study of the acceptability of arguments is one of the main concerns in Argumentation Theory. In formal systems of defeasible argumentation, arguments for and against a proposition are produced and evaluated to test the acceptability of that proposition following a dialectical process [6]. The main idea in these systems is that a proposition will be accepted as true if there exists an argument that supports it, and this argument is acceptable according to an analysis between it and its counterarguments. This analysis requires a process of comparison of conflicting arguments in order to decide which one is preferable [1,7,8,9,10]. After this dialectical analysis is performed over the set of arguments in the system, some of them will be *acceptable* arguments, while others will be not. Argumentation is used as a form of non-monotonic or defeasible reasoning [11] and it is suitable for modeling dialogues between intelligent agents [12].

Abstract argumentation systems [13,3,9] are formalisms for argumentation where some components remain unspecified. Usually, the actual structure of an argument is abstracted away. In this kind of system, the emphasis is put on the semantic notion of finding the set of accepted arguments. Most of them are based on the single abstract concept of the *attack* represented as a binary relation, and extensions are defined as sets

of possibly accepted arguments. This primitive notion of defeat between arguments is the basis of the study of argumentation semantic, but a more detailed model will be useful to capture specific behaviour of concrete systems.

We define a framework where the defeat relation between arguments is decomposed into two basic elements: symmetric conflicts and a preference criterion. Finding a preferred argument is essential to determine a defeat relation [1,8,9,10]. However, the task of comparing arguments to establish a preference is not always successful. In this case, the classic abstract attack relation is no longer useful as a modelling tool. In the next section, we present an abstract framework for argumentation where conflicts and preference between arguments are considered, and the associated semantic operator is defined.

2. Argumentation Framework

Our argumentation framework is formed by four elements: a set of arguments, and three basic relations between arguments.

Definition 1 *An abstract argumentation framework (AF) is a quartet* $\langle Args, \sqsubseteq, \mathbf{C}, \mathbf{R}\rangle$*, where* $Args$ *is a finite set of arguments,* $\sqsubseteq$ *is the subargument relation,* $\mathbf{C}$ *is a symmetric and anti-reflexive binary conflict relation between arguments,* $\mathbf{C} \subseteq Args \times Args$*, and* $\mathbf{R}$ *is a preference relation among arguments.*

Here, arguments are abstract entities [13] that will be denoted using calligraphic uppercase letters. No reference to the underlying logic is needed since we are abstracting the structure of the arguments (see [1,8,11] for concrete systems). The symbol $\sqsubseteq$ denotes subargument relation: $\mathcal{A} \sqsubseteq \mathcal{B}$ means "$\mathcal{A}$ is a subargument of $\mathcal{B}$".

The conflict relation between two arguments $\mathcal{A}$ and $\mathcal{B}$ denotes the fact that these arguments cannot be accepted simultaneously since they contradict each other. For example, two arguments $\mathcal{A}$ and $\mathcal{B}$ that support complementary conclusions l and $\neg l$ cannot be accepted together. Also an argument with hypotesis h cannot be accepted together with an argument for $\neg h$. The set of all pairs of arguments in conflict on Φ is denoted by $\mathbf{C}$. 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 set *Conf*($\mathcal{A}$) is the set of all arguments $\mathcal{X} \in Args$ such that $(\mathcal{A}, \mathcal{X}) \in \mathbf{C}$.

The constraints imposed by the conflict relation lead to several sets of possible accepted arguments. Therefore, some way of deciding among all the possible outcomes must be devised. In order to accomplish this task, the relation $\mathbf{R}$ is introduced in the framework and it is used to evaluate arguments, modelling a preference criterion based on a measure of strength. If $\mathcal{A}\mathbf{R}\mathcal{B}$ but not $\mathcal{B}\mathbf{R}\mathcal{A}$ then $\mathcal{A}$ is preferred to $\mathcal{B}$, denoted $\mathcal{A} \succ \mathcal{B}$. If $\mathcal{A}\mathbf{R}\mathcal{B}$ and $\mathcal{B}\mathbf{R}\mathcal{A}$ then $\mathcal{A}$ and $\mathcal{B}$ are 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, denoted $\mathcal{A} \bowtie \mathcal{B}$.

Preference is usually based on structural properties of arguments, as the number of logical rules used to derive the conclusion or the number of propositions involved in that process. Other non-trivial preferences may be captured by $\mathbf{R}$, for example, the fact that an argument with conclusion $\neg h$ is preferred to an argument whith hypotesis h. As the comparison criterion is treated abstractly, we do not assume any property of $\mathbf{R}$ but, as stated in [3], several conditions must be satisfied, for example, that an argument is always preferred (or equivalent in conclusive force) to any superargument. Therefore, if $\mathcal{A} \succ \mathcal{B}$ then $\mathcal{A} \succ \mathcal{C}$ for any superargument $\mathcal{C}$ of $\mathcal{B}$. Any concrete framework may

establish additional requirements for decision making. The conflict relation should also exhibit a rational behaviour regarding subarguments. If $(\mathcal{A}, \mathcal{B}) \in \mathbf{C}$, then $(\mathcal{A}, \mathcal{B}_1) \in \mathbf{C}$, $(\mathcal{A}_1, \mathcal{B}) \in \mathbf{C}$, and $(\mathcal{A}_1, \mathcal{B}_1) \in \mathbf{C}$, for any arguments $\mathcal{A}_1, \mathcal{B}_1$, $\mathcal{A} \sqsubseteq \mathcal{A}_1$ and $\mathcal{B} \sqsubseteq \mathcal{B}_1$. We call this property *conflict inheritance*: if an argument $\mathcal{A}$ is in conflict with an argument $\mathcal{B}$ then that conflict is still present when considering superarguments of $\mathcal{A}$ or $\mathcal{B}$.

Example 1 $\Phi = \langle Args, \sqsubseteq, \mathbf{C}, \mathbf{R} \rangle$ *is an AF where* $Args = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, \mathcal{E}\}$, $\mathbf{C} = \{\{\mathcal{A}, \mathcal{B}\}, \{\mathcal{B}, \mathcal{C}\}, \{\mathcal{C}, \mathcal{D}\}\}, \{\mathcal{C}, \mathcal{E}\}\}$[1] *and* $\mathcal{A} \succ \mathcal{B}, \mathcal{B} \succ \mathcal{C}$, $\mathcal{E} \bowtie \mathcal{C}$ *and* $\mathcal{C} \equiv \mathcal{D}$.

For two arguments $\mathcal{A}$ and $\mathcal{B}$ in $Args$, such that the pair $(\mathcal{A}, \mathcal{B})$ belongs to $\mathbf{C}$ the relation $\mathbf{R}$ is considered. If a concrete preference is made ($\mathcal{A} \succ \mathcal{B}$ or $\mathcal{B} \succ \mathcal{A}$), then a defeat relation is established. It is said that the preferred argument is a *proper defeater* of the non-preferred argument. If the arguments are *indifferent* according to $\mathbf{R}$, then they have the *same* relative conclusive force. For example, if the preference criterion establishes that smaller arguments are preferred, then two arguments of the same size are indifferent. On the other hand, arguments may be *incomparable*. For example, if the preference criterion states that argument $\mathcal{A}$ is preferred to $\mathcal{B}$ whenever the premises of $\mathcal{A}$ are included in the premises of $\mathcal{B}$, then arguments with disjoint sets of premises are incomparable. This situation must be understood as a natural behaviour. When two conflictive arguments are indifferent or incomparable according to $\mathbf{R}$, the conflict between these two arguments remains unresolved. Due to this situation and to the fact that the conflict relation is a symmetric relation, each of the arguments is *blocking* the other one and it is said that both of them are *blocking defeaters* [1]. 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 $\mathcal{A}$. In example 1, argument $\mathcal{A}$ is a proper defeater of argument $\mathcal{B}$, while $\mathcal{C}$ is a blocking defeater of $\mathcal{D}$ and vice versa.

Well known semantics for abstract argumentation frameworks are based on defeat relations, usually called *attack* relations [13,3,14]. These formalisms assume the existence of a binary relation of attack (not necessarily symmetric) defined over the set of all possible arguments, such that if $(\mathcal{A}, \mathcal{B})$ are in the attack relation then in order to accept $\mathcal{B}$ it is necessary to find out if $\mathcal{A}$ is accepted or not, but not the other way around. The acceptance relation should be derived from a conflict relation between arguments and a suitable comparison criterion, and that criterion usually remains unspecified in the abstract system. This remark on the attack relation is seldom made. It is our contention that an extended semantics for argumentation will be useful. This semantics will be based on the two defining characteristics of an argumentation system: the conflict relation between arguments and the comparison criterion used to evaluate such arguments.

Arguments can be classified as *accepted* arguments or *non-accepted* or *rejected* arguments according to their context in the framework. Any set of accepted arguments should not contain arguments in conflict. A set of arguments S is said to be *conflict free* if for all $\mathcal{A}, \mathcal{B} \in S$ then $(\mathcal{A}, \mathcal{B}) \notin \mathbf{C}$. In example 1 the set $\{\mathcal{A}, \mathcal{C}\}$ is a conflict free set.

Given a set of arguments S, two kinds of arguments are easily identified as accepted arguments: first, those arguments not involved in any conflict in S; second, those arguments actually involved in a conflict, but preferred to the arguments that are in conflict with them, according to relation $\mathbf{R}$. Both kinds of special arguments are called *defeater free* arguments. An argument $\mathcal{A}$ is defeater-free in a set S if no argument in S is a de-

[1] When describing elements of $\mathbf{C}$, we write $\{\mathcal{A}, \mathcal{B}\}$ as an abbreviation for $\{(\mathcal{A}, \mathcal{B}), (\mathcal{B}, \mathcal{A})\}$, for any arguments $\mathcal{A}$ and $\mathcal{B}$ in $Args$.

feater of $\mathcal{A}$. Defeater-free arguments must be accepted, since no (preferred) contradictory information is provided in the framework. Note that this classification is relative to the set in which the argument is included. The semantic of $\mathbf{C}$ states that when an argument $\mathcal{A}$ is accepted, any argument in $Conf(\mathcal{A})$ should be rejected. The following definition captures a subset of arguments that should be rejected in the framework.

Definition 2 *Let S be a set of arguments in $\langle Args, \sqsubseteq, \mathbf{C}, \mathbf{R}\rangle$. An argument $\mathcal{A} \in S$ is said to be suppressed in S if one of the following cases hold: (a) there is a defeater-free argument $\mathcal{B}$ in S such that $\mathcal{B}$ is a proper defeater of $\mathcal{A}$, or (b) there is a blocking defeater $\mathcal{B}$ of $\mathcal{A}$in S, and there is no other argument $\mathcal{C}$ ($\mathcal{C} \neq \mathcal{A}$) in S such that $\mathcal{C}$ is a defeater of $\mathcal{B}$.*

The first case is clear since any argument involved in a conflict must be suppressed when its counterpart in this conflict is accepted (has no defeater). The second case reflects the situation in which two arguments are taking part of an unsolved conflict and from the point of view of one of them ($\mathcal{A}$) its opponent is not attacked by a third argument. The argument $\mathcal{A}$ should be suppressed since the threat of $\mathcal{B}$ cannot be avoided, despite other attacks on $\mathcal{A}$. Note that if $\mathcal{A}$ is only defeated by $\mathcal{B}$ then both arguments should be suppressed because the blocking condition is symmetrical.

Given a set S of arguments it is as easy to identify obviously suppressed arguments as it is to identify inevitably accepted ones. The following function $\Upsilon : 2^{Args} \longrightarrow 2^{Args}$ characterizes the set of arguments not directly suppressed in a given set S.

$$\Upsilon(S) = \{\mathcal{A} : \mathcal{A} \in S \text{ and } \mathcal{A} \text{ is not suppressed in } S\}$$

It is easy to see that if S is a conflict-free set of arguments, then $S = \Upsilon(S)$. However, the converse is not true, as shown in the next example:

Example 2 *Let $\langle Args, \sqsubseteq, \mathbf{C}, \mathbf{R}\rangle$ be an AF, where $Args = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}\}$ and $\mathbf{C} = \{\{\mathcal{A}, \mathcal{B}\}, \{\mathcal{B}, \mathcal{C}\}, \{\mathcal{C}, \mathcal{D}\}, \{\mathcal{D}, \mathcal{A}\}\}$ and for all arguments $\mathcal{X}$ and $\mathcal{Y}$, $\mathcal{X} \bowtie \mathcal{Y}$. No argument in $Args$ is a defeater-free argument, therefore $\Upsilon(Args) = Args$.*

By definition, $\Upsilon(S)$ includes some (or all) of the arguments in S. In the set $\Upsilon(S)$ some arguments may now be classified as *defeater-free* arguments, since its defeaters are suppressed arguments in S. It is then possible to repeatedly apply function Υ to the set of arguments in the framework. This process may continue until a fixpoint is reached.

Definition 3 *Υ^n is defined as: Υ^0 is $Args$, and $\Upsilon^{(n+1)} = \Upsilon \circ \Upsilon^n$. The set of arguments $\Upsilon^k, k \geq 0$ such that $\Upsilon^k = \Upsilon^{k+1}$ is denoted Υ^ω.*

Example 3 *Let $\Phi_2 = \langle Args, \sqsubseteq, \mathbf{C}, \mathbf{R}\rangle$ be an AF where $Args = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}\}$, $\mathbf{C} = \{\{\mathcal{A}, \mathcal{B}\}, \{\mathcal{B}, \mathcal{C}\}, \{\mathcal{C}, \mathcal{D}\}\}$ and $\mathcal{A} \equiv \mathcal{B}$, $\mathcal{B} \bowtie \mathcal{C}$ and $\mathcal{C} \succ \mathcal{D}$. In this framework, $\Upsilon^1 = \{\mathcal{A}, \mathcal{D}, \mathcal{C}\}$, because $\mathcal{B}$ is a suppressed argument, as $\mathcal{A}$ is a blocking defeater not defeated by a third argument. $\Upsilon^2 = \{\mathcal{A}, \mathcal{C}\}$ because $\mathcal{D}$ is defeated by $\mathcal{C}$ which is now defeater-free in Υ^1. Because $\Upsilon^2 = \Upsilon^3$ then $\Upsilon^\omega = \{\mathcal{A}, \mathcal{C}\}$.*

Trivially, no argument is suppressed in Υ^ω. An argument in Υ^ω which is not in conflict with any other argument in the same set is an accepted argument. The set of accepted arguments in Υ^ω is denoted $\Upsilon^{\omega+}$. Therefore, if Υ^ω is a conflict-free set (as in example 3, but not in example 2), then any argument in Υ^ω is an *accepted* argument.

The previously defined conflict inheritance leads to a common sense property of argumentation frameworks. For any argument $\mathcal{A}$, if $\mathcal{A} \in \Upsilon^{\omega+}$ then $\mathcal{B} \in \Upsilon^{\omega+}$ for all $\mathcal{B} \sqsubseteq \mathcal{A}$. Suppose $\mathcal{A}_1 \sqsubseteq \mathcal{A}$ is not in Υ^{ω}. Then $\mathcal{A}_1$ is a suppressed argument, because one of the conditions of definition 2 holds in some $\Upsilon^i, i > 0$. But if $\mathcal{A}_1$ is suppressed in Υ^i then also $\mathcal{A}$ is suppressed in Υ^i because they share defeaters (because of conflict inheritance) and therefore is also suppressed. The reader is referred to [15] for the role of subarguments in well structured argumentation, using the framework of definition 1.

In the framework of example 2, no arguments should be accepted as it is not possible to establish a concrete preference. Here, Υ^{ω} is not a conflict-free set. This is related to the presence of some special arguments involved in a cicle of defeaters, a common situation called a *fallacy*. Any argument involved in a fallacy is usually called *fallacious*. The most important premise in defeasible argumentation is that an argument must be *accepted* only when none of its defeaters are. However, no fallacious argument can exhibit this property, because at least one of its defeaters is also a fallacious argument [2]. Therefore, any argument of this kind should not be accepted. An AF is said to contain a fallacy if Υ^{ω} is not a conflict-free set of arguments.

3. Related Work and Conclusions[3]

We introduced a new abstract framework for argumentation where two kinds of defeat are present, depending on the outcome of the preference relation. A fix-point operator is used to characterize the set of accepted arguments. This operator also provides a method for identifying controversial situations.

Since the introduction of Dung's seminal work [13] on the semantics of argumentation this area has been extremely active. This approach begins by defining an abstract framework in order to characterize the set of accepted arguments independently of the underlying logic. We followed this line in this work. In Dung's presentation no explicit preference relation is included, and the basic interaction between arguments is the binary, non-symmetric, *attack* relation. This style of argument attack is used in a number of different abstract frameworks, but none of them separates the notion of preference criteria from the conflict relation, as it is usually done in concrete systems. The classic attack relation allows the definition of mutual defeaters: two arguments attacking each other. This is not very realistic, as there is not an attack situation (in the sense of being conflictive and preferred to the opponent) but a controversial situation due to the lack of decision in the system. In our framework, this leads to blocking defeaters. The fixpoint semantic defined here results more credulous than the classic grounded extension [13], as it can be noted in example 3, where according to Dung the grounded extension is the empty set.

Several frameworks do include a preference relation. Vreeswijk, in [3], defines a popular abstract framework, making important considerations on comparison criterions. Interesting frameworks that consider the issue of preference relations are introduced in [9], [16] and in [17]. In these frameworks the basic interaction between agents is the classic *attack* relation, and the preference order is used as a defense against conflictive arguments. The defeat relation arises when the preferences agree with the attack.

[2] Because any non-fallacious defeater has been previously suppressed.

[3] Space limitations prevent us of a more complete review of related work

Bench-Capon, in [18], also defines an argumentation framework that includes a way to compare arguments. A set of values related to arguments is defined in the framework. Since a preference relation is defined on the values promoted by arguments, those arguments can be weighted in order to resolve attacks. However, only a single notion of defeat is derived. This defeat occurs when the value promoted by the attacked argument is not preferred to the value promoted by the attacker. Again, the preference order is used to check if the attacker argument is preferred, not to elucidate symmetric conflicts as it is used in our framework.

References

[1] G. R. Simari and R. P. Loui. A Mathematical Treatment of Defeasible Reasoning and its Implementation. *Artificial Intelligence*, 53:125–157, 1992.

[2] R. A. Kowalski and F. Toni. Abstract argumentation. *Artificial Intelligence and Law*, 4(3-4):275–296, 1996.

[3] G. A. W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90(1–2):225–279, 1997.

[4] C. I. Chesñevar, A. G. Maguitman, and R. P. Loui. Logical Models of Argument. *ACM Computing Surveys*, 32(4):337–383, December 2000.

[5] H. Prakken and G. A. W. Vreeswijk. Logical systems for defeasible argumentation. In D.Gabbay, editor, *Handbook of Philosophical Logic, 2nd ed.* Kluwer Academic Pub., 2000.

[6] G.R. Simari, C.I. Chesñevar, and A.J. García. The role of dialectics in defeasible argumentation. In *XIV Intl. Conf. of the Chilenean Computer Science Society*, pages 111–121, 1994.

[7] D. L. Poole. On the Comparison of Theories: Preferring the Most Specific Explanation. In *Proc. 9th IJCAI*, pages 144–147. IJCAI, 1985.

[8] H. Prakken and G. Sartor. A system for defeasible argumentation, with defeasible priorities. In *Proc. of the Intl. Conf. on Formal and Applied Practical Reasoning (FAPR-96)*, volume 1085 of *LNAI*, pages 510–524, June 3–7 1996.

[9] L. Amgoud. Using preferences to select acceptable arguments. In *Proc. of European Conf. in Artificial Intelligence (ECAI'98) , Brighton*, pages 43–44, August 1998.

[10] F. Stolzenburg, A.J. García, C.I. Chesñevar, and G.R. Simari. Computing generalized specificity. *Journal of Aplied Non-Classical Logics*, 13(1):87–113, January 2003.

[11] A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.

[12] L. Amgoud, S. Parsons, and N. Maudet. Arguments, dialogue and negotiation. In *Proc. of the 14th European Conf. on Artificial Intelligence*, pages 338–342. ECAI, 2001.

[13] P. M. Dung. On the Acceptability of Arguments and its Fundamental Role in Nomonotonic Reasoning and Logic Programming. In *Proc. of the 13th IJCAI*, pages 852–857, 1993.

[14] C. Cayrol, S. Doutre, M. C. Lagasquie-Schiex, and J. Mengin. "Minimal Defence": a refinement of the preferred semantics for argumentation frameworks. In *Proc. of the 9th Intl. Workshop on Non-Monotonic Reasoning*, pages 408–415, July 2002.

[15] D.C. Martínez, A.J. García, and G.R. Simari. Progressive defeat paths in abstract argumentation frameworks. In *Proc. of the 19th Conf. of the Canadian Society for Computational Studies of Intelligence*, pages 242–253, 2006.

[16] L. Amgoud and C. Cayrol. On the acceptability of arguments in preference-based argumentation. In *14th Conf. on Uncertainty in Artificial Intelligence*, pages 1–7, 1998.

[17] L. Amgoud and L. Perrussel. Arguments and Contextual Preferences. In *Computational Dialectics-Ecai workshop (CD2000) , Berlin*, August 2000.

[18] T.J.M. Bench-Capon. Value-based argumentation frameworks. In *Proc. of Nonmonotonic Reasoning*, pages 444–453, 2002.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Automating Argumentation for Deliberation in Cases of Conflict of Interest

Alison Chorley, Trevor Bench-Capon, Peter McBurney
Department of Computer Science, The University of Liverpool, Liverpool, UK

Abstract. One approach to deliberation about the choice of what to do in a particular situation is to use an argument scheme based on the practical syllogism to supply a presumptive reason for the action, which is then subjected to a process of critical questioning to see if the presumption can be maintained. In this paper we describe an implemented realisation of this approach. We describe a representation which permits the instantiation of the argument scheme and the generation and consideration of critical questions to it. In order to automate the process we supply operational interpretations of the argument scheme, the critical questions and rebuttals to those questions. Our realisation is illustrated with a detailed example of a particular case.

Keywords. Reasoning about action with argument, decision making based on argumentation, argument schemes

1. Introduction

One important use of argument is in the context of rational choice of actions, what is termed "practical reasoning" in philosophy. The correctness of a choice of action typically cannot be demonstrated conclusively, since it often must be made in the face of uncertainty and incomplete information as to the current situation; the effect of the action is often unpredictable and dependent on the choices of other agents or the environment; and there is usually an element of subjectivity in that the best choice will vary from agent to agent dependent on their interests, aspirations and values. Moreover, even if the choice is a good one, it may involve rejecting better choices, or curtailing future options. Justifying an action, either prospectively in deliberation, or retrospectively when challenged to explain oneself, involves putting forward reasons for the choice, that is an argument for why it is the right choice for the person concerned in the particular situation in which it was made.

Traditionally, justifications of such arguments have taken the form of the practical syllogism (e.g. [1]), which states that the action concerned is a sufficient means to realise a goal desired by the agent concerned. This was adapted into an argument scheme by Walton in [2], the sufficient condition scheme. The essence of Walton's approach is that being able to instantiate this scheme provides a presumptive justification for an action, which is then subject to a set of critical questions characteristic of the particular scheme, and any such questions that are posed must be answered satisfactorily if the presumption

is to be maintained. Walton gives four critical questions: Will alternative actions achieve the goal?; Is the action possible?; Are there other goals to consider?; and, Will the action have undesirable side effects? Walton's idea was elaborated by Atkinson and her colleagues ([3] gives the fullest description) to distinguish the goal into three elements: the state of affairs resulting from the action; the specific features of that state which are desired by the agent; and the social value or interest of the agent which makes those features desirable. This elaboration correspondingly extended the set of critical questions to sixteen.

In this paper we will provide a description of a realisation of deliberation about a choice of action based on this approach, in a situation where multiple agents have conflicting values. We will use the argument scheme proposed in [3] to generate presumptive arguments for and against actions, and then subject these arguments to critical questioning. Our critical questions will differ from those described in [3], in that some found there are inapplicable in our particular situation. We will rephrase the applicable questions in terms of our representation and supply characteristic rebuttals and counter-rebuttals of the various critical questions we use.

Throughout this paper we will illustrate our approach with a particular example, based on an ethical dilemma, also used in [4]. In section 2 we will describe the problem and our representation of this problem, together with the underlying logical formalism used, which is based on an extension of Alternating-time Temporal Logic (ATL) [5]. Section 3 will describe the argument scheme, the associated critical questions, and the ways of replying to those questions. Section 4 will describe the program which realises the approach. In section 5 we will apply the program to give an example of how agents will reason in a particular situation. Finally section 6 will offer some concluding remarks.

2. Representation of the Insulin Problem

We base our considerations on the representation and discussion of a specific example, a well-known problem intended to explore a particular ethical dilemma discussed by Coleman [6] and Christie [7], amongst others and also extensively discussed in [4]. The situation involves two agents, Hal and Carla, both of whom are diabetic. Hal, *through no fault of his own*, has lost his supply of insulin and urgently needs to take some to stay alive. Hal is aware that Carla has some insulin kept in her house, but does not have permission to enter Carla's house. The question is whether Hal is justified in breaking into Carla's house and taking her insulin in order to save his life. By taking Carla's insulin, Hal may be putting her life in jeopardy. One possible response is that if Hal has money, he can compensate Carla so that she can replace her insulin. Alternatively if Carla has money, she can replenish her insulin herself. There is, however, a serious problem if neither have money, since in that case Carla's life is really under threat. Coleman argued that Hal may take the insulin to save his life, but should compensate Carla. Christie's argument against this was that even if Hal had no money and was unable to compensate Carla he would still be justified in taking the insulin by his immediate necessity, since no one should die because of poverty. Christie then argues he cannot be *obliged* to compensate Carla even when he is able to.

We need to extend the representation given in [4] in order to make explicit the implicit information used there to block fruitless arguments. For the purposes of our repre-

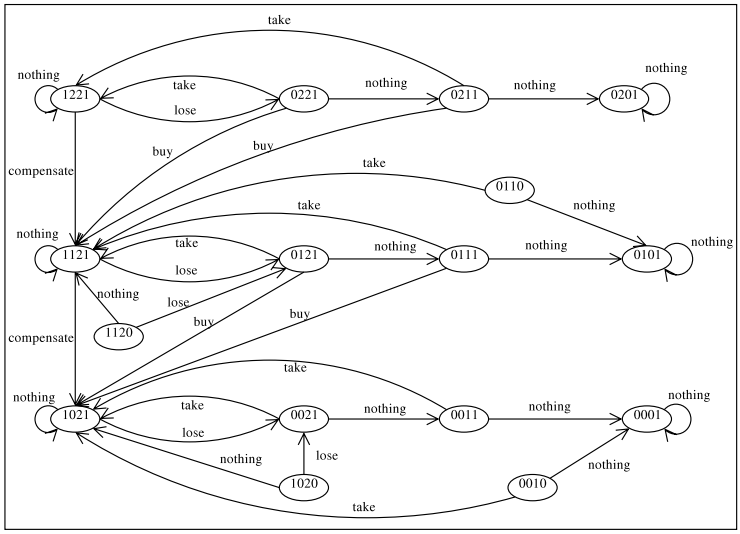


Figure 1. State Transition diagram for each agent.

sentation four attributes of agents are important: whether they have insulin (I), whether they have money (M), whether they are alive (A) and the time in the world (W). The world attribute represents the fact that the shops are shut when Hal loses his insulin and so he cannot buy insulin and has to take Carla's, whereas later the shops are open so Carla is able to buy insulin if she can afford it. The state of an agent may thus be represented as a vector of four digits, *IMAW*, with I and W equal to 1 if the agent has insulin and the shops are open and 0 if these things are false. M and A can have three values each with M equal to 0 if the agent has no money, 1 for enough money to buy insulin and 2 if they have more than enough money. A is equal to 0 if the agent is dead, 1 if the agent is in a critical state (which Hal is because he urgently needs insulin) and 2 if the agent is healthy (which Carla is initially). Since I cannot be true and A zero or one (the agent will be in good health if he or she has insulin), an agent may be in any one of sixteen possible states shown in Figure 1. Because there are two agents, the system has 16x16=256 possible states. We may now represent the actions available to the agents by depicting them as automata, as shown in Figure 1. An agent with insulin may lose its insulin; an agent with money and insulin may compensate another agent; an agent with no insulin may take another's insulin, or, with money, buy insulin provided the shops are open. In any situation when it is alive, an agent may choose to do nothing; if dead it can only do nothing. The nodes representing distinct world-states are labelled with the values of the vector $IMAW$. Arcs are labelled with actions.

To represent the interaction between the agents we draw upon the approach of Wooldridge and van der Hoek [5] which formally describes a normative system in terms of constraints on actions that may be performed by agents in any given state. The semantic structures used in [5] are known as Action-based Alternating Transition Systems (AATSs) and were developed for modelling game-like, dynamic, multi-agent systems.

An *Action-based Alternating Transition System* (AATS) is an $(n + 7)$-tuple $S = \langle Q, q_0, Ag, Ac_1, ..., Ac_n, \rho, \tau, \phi, \pi \rangle$, where:

- Q is a finite, non-empty set of *states*;
- $q_0 \in Q$ is the *initial state*;
- $Ag = \{1, ..., n\}$ 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 $i \neq j \in Ag$;
- $\rho : Ac_{Ag} \rightarrow 2^Q$ is an *action precondition function*, which for each action $\alpha \in Ac_{Ag}$ defines the set of states $\rho(\alpha)$ from which α may be executed;
- J_{Ag} is the set of *Joint Actions* such that every $j \in J_{Ag}$ is a tuple $\langle \alpha_1, \alpha_2, ..., \alpha_k \rangle$ where for each α_i ($i \leq k$) there is some $i \in Ag$ such that $\alpha_i \in Ac_i$.
- $\tau : Q$ x $J_{Ag} \rightarrow Q$ 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);
- ϕ is a finite, non-empty set of *atomic propositions*; and
- $\pi : Q \rightarrow 2^\phi$ 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.

We have extended the AATS to include a set of values (V_n) and a set of functions which determine whether these values are promoted or demoted by a transition between states.

We now turn to representing the Hal and Carla scenario as an AATS. As noted above, we have 256 possible states for the two agents, $q_1..q_{256}$. Normally both agents will have insulin, but we are specifically interested in the situations that arise when one of them (Hal) loses his insulin and is in critical health. For reasons of space we will consider only the initial state of the scenario where both agents have money, $q_0 = q_{253}$. The initial state is thus the one in which I_H = 0, M_H = 1, $A_H = 1$ and W_H = 0 and I_C = 1, M_C = 1, $A_C = 2$ and W_C = 0. As the shops are closed in the initial state, represented by W_H = 0, Hal's only option is to take Carla's insulin or do nothing. If Hal does nothing, neither agent can act further because Hal dies. If Hal takes Carla's insulin then Hal can then compensate Carla or do nothing. Similarly, after Hal takes the insulin, Carla can buy insulin or do nothing. The possible developments from the initial state are shown in Figure 2. States are labelled with the two vectors $I_H M_H A_H W_H$(on the top row) representing Hal's state of the world and $I_C M_C A_C W_C$ (on the bottom row) representing Carla's state of the world, and the arcs are labelled with the joint actions and with the value promoted or demoted by the joint action. The instantiation of the problem as an AATS is summarised in Table 1.

3. Constructing Arguments and Attacks

The argument scheme proposed in [3] is:

> In the current situation R action A should be performed to bring about a new situation S which realises Goal G which promotes value V.

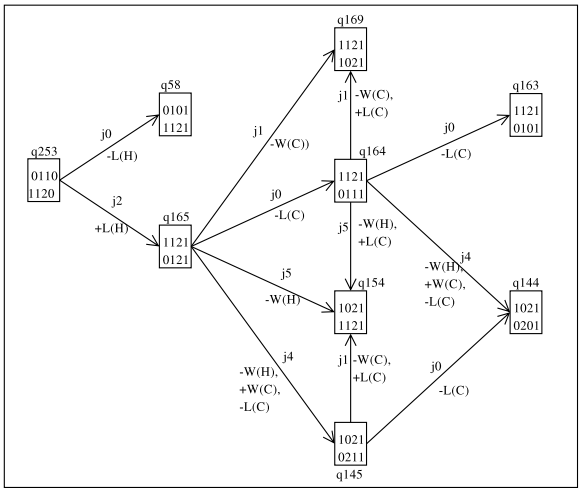


Figure 2. Part of the State Space when Both Agents have money at the initial state.

3.1. Refining the Argument Scheme

In our formalism the agent is in some particular state, Q_n. From that state there are a number of transitions available to reach new states. Each transition is achieved through some joint action J_n, which requires the agent to play its part by performing some action Ac_n. Each resulting state can be compared with Q_n to see how it changes the situation with respect to the values of the agent. We can then label the transitions with V_n+ or V_n- as appropriate to indicate that the value V_n is promoted or demoted, respectively.

Now for each transition with a positive label V_n+ we can produce the argument scheme:

PA1: In Q_n action Ac_n should be performed by Ag_i to reach Q_m which realises ϕ which promotes V_n.

Where the label is negative the argument scheme is slightly different:

PA2: In Q_n action Ac_n should not be performed by Ag_i to avoid Q_m which realises ϕ which demotes V_n.

This bipolarity of arguments, depending on whether they provide a reason to act or refrain has been noted in [8], although the treatment there differs from that described below, as we use such arguments as critical questions to arguments of the form PA1. Finally, it may be that nothing good can be done, in which case we may wish to perform an action which will avoid harm. Thus, if the transition neither promotes nor demotes any value, the argument scheme is:

PA3: In Q_n action Ac_n should be performed by Ag_i to reach Q_m which is neutral with respect to all values.

Table 1. Instantiation of the Insulin Problem

States and Initial States

$Q = \{q_0, ..., q_{256}\}$

Initial State q_{253}

Agents, Actions and Joint Actions

Ag = {H, C} $Ac_H = \{take_H, compensate_H, doNothing_H\}$

$Ac_C = \{buy_C, doNothing_C\}$

$J_{Ag} = \{j_0, j_1, j_2, j_3, j_4, j_5\}$, where $j_0 = \langle doNothing_H, doNothing_C\rangle$,

$j_1 = \langle doNothing_H, buy_C\rangle$, $j_2 = \langle take_H, doNothing_C\rangle$, $j_3 = \langle take_H, buy_C\rangle$,

$j_4 = \langle compensate_H, doNothing_C\rangle$, $j_5 = \langle compensate_H, buy_C\rangle$

Propositional Variables

$\phi = \{hasInsulin_H, hasMoney_H, isAlive_H, world_H,$

$hasInsulin_C, hasMoney_C, isAlive_C, world_C\}$

Values

$V_n = \{L_H, L_C, W_H, W_C\}$

Transitions/Pre-Conditions/Interpretation

$q \backslash j$	j_0	j_1	j_2	j_3	j_4	j_5	$\pi(q)$
q_{253}	q_{58}	-	q_{165}	-	-	-	$\{0_H, 1_H, 1_H, 0_H, 1_C, 1_C, 2_C, 0_C\}$
q_{58}	-	-	-	-	-	-	$\{0_H, 1_H, 0_H, 1_H, 1_C, 1_C, 2_C, 1_C\}$
q_{144}	-	-	-	-	-	-	$\{1_H, 0_H, 2_H, 1_H, 0_C, 2_C, 0_C, 1_C\}$
q_{145}	q_{144}	q_{154}	-	-	-	-	$\{1_H, 0_H, 2_H, 1_H, 0_C, 2_C, 1_C, 1_C\}$
q_{154}	-	-	-	-	-	-	$\{1_H, 0_H, 2_H, 1_H, 1_C, 1_C, 2_C, 1_C\}$
q_{163}	-	-	-	-	-	-	$\{1_H, 1_H, 2_H, 1_H, 0_C, 1_C, 0_C, 1_C\}$
q_{164}	q_{163}	q_{169}	-	-	q_{144}	q_{154}	$\{1_H, 1_H, 2_H, 1_H, 0_C, 1_C, 1_C, 1_C\}$
q_{165}	q_{164}	q_{169}	-	-	q_{145}	q_{154}	$\{1_H, 1_H, 2_H, 1_H, 0_C, 1_C, 2_C, 1_C\}$
q_{169}	-	-	-	-	-	-	$\{1_H, 1_H, 2_H, 1_H, 1_C, 0_C, 2_C, 1_C\}$

From the schemes PA1, PA2 and PA3 the agent can instantiate a set of arguments, one or more per transition, depending on how many values are affected by a transition. The agent can now order these arguments, beginning with the argument promoting its most favoured value down to that promoting its least favoured value, through any neutral arguments to the argument demoting its least favoured value and finally to the argument demoting its most favoured value. Each argument will be considered in turn, as providing the currently best presumptive justification.

3.2. Critical Questions

Consideration of the arguments involves posing critical questions. Our approach will not use all the critical questions of [3] because the aim there was to generate arguments that could arise amongst a group of agents with different views as to what is the case and/or different values and interests, whereas here we have a single agent, with a particular view of the situation, posing critical questions to itself, although in a context where other agents may influence the outcome of its actions. This is of course applicable to a range of problem scenarios.

For the purposes of this paper we will assume that the agent knows which state it is in. As [4] shows, uncertainty as to the current situation has an important role to play in

determining how acceptable a given justification is, but for purposes of deliberation the agent must act as if a state held, and must choose one state consistent with its beliefs on which to base its reasoning. We also consider that the agent is aware of its possible actions, and which state they will bring about for a given action of the other agent. Critical questions pertaining to verifying that the agent is aware of the consequences of its choice are therefore obviated. Since we are concerned with the deliberations of a single agent, questions as to the validity of its values cannot arise. Finally the ordering of the arguments discussed in Section 3.1 ensures that there is no better alternative to the argument under current consideration.

This leaves us with three critical questions: Whether the action will demote a more important value; Whether performing the action will realise the desired goal; and whether the action will preclude some more desirable future actions. In terms of our representation, we may state these questions as:

PCQ1 Might the action lead to states that the agent will wish to avoid?

PCQ2 Might the other agent fail to act so as to perform the desired joint action?

PCQ3 Is the desired state in fact a local optimum, so that all subsequent states will result in a state worse than the current one?

PCQ1 relates to whether we have a stronger argument against performing the action. This argument may be from an unfortunate side effect of the target state itself, in that it demotes a value we prefer to the one it promotes. Remember, however, that the state we reach from performing an action may not be the one we wish to reach, since the chosen action only determines a set of joint actions. Thus the choice of the other agent may mean that performing this action will take us to an unfavourable state: this risk can only be avoided by refraining from the action. In either case there will be present in the set of arguments an argument or arguments of the form PA2, which may point wither to demote values in the target state, or the risk of demoted values if other agents behave unexpectedly. Each of these poses a critical question of form PCQ1.

The rebuttal to PCQ1 involves considering the arguments available to the other agent. On the assumption that the other agent is rational, it will be reasoning in a similar fashion. And if the other agent also has a reason to avoid the undesired state, we can discount the risk. Thus if the other agent has available an argument of the form PA2 instructing it to avoid the undesired state, we may consider rejecting PCQ1. PCQ1, however, may be re-instated if the other agent has a counter-rebuttal: that is if the other agent has a better reason (in terms of its own value ordering) to reach the undesired state. That is, an argument of the form PA1 in favour of reaching the state to promote a value that agent prefers. In this case we must consider PCQ1 unanswered and reject the argument it attacks.

PCQ2 also involves the other agent. In this case the other agent may have a reason for avoiding the state we wish to reach, that is, have an argument of form PA2 recommending that it avoids the state we wish to reach. In this case, there is no point in acting to reach the state since we will expect the other agent to frustrate our attempt. The rebuttal to PCQ2 is that the other agent has a stronger reason to reach the state we desire, namely an argument of the form of PA1 relating to a value preferred (on its own value ordering). Given such an argument we may expect it to cooperate and participate in the joint action which will reach this state.

PCQ3 arises from the possibility that the state we are trying to reach may be initially promising, but ultimately lead to unfortunate consequences. Thus we have a reason to avoid a state, even if it promotes a value, if all subsequent choices that can be made in that state will result in us being worse off than we were in the initial state. This involves looking ahead to some final state. In the case where paths do not terminate, some cut-off to keep within resource bounds must be applied. Again the rebuttal of this question involves the other agent having a compelling argument to avoid the state with this property, and no stronger argument to reach it.

3.3. Damage Limitation

Note that the arguments so far considered all supply a reason to act, so as to reach a particular state which promotes a value. Arguments to refrain from an action to avoid a state demoting a value have appeared only in the critical questions. This is as it should be, since in our formalism we can only refrain from an action by choosing to perform a different one, and we need to justify this choice against alternative ways of refraining. If we have no positive arguments able to withstand the critical questions, we need choose the least damaging action. We pick the argument which demotes the least favoured value and perform the action despite these bad consequences. Even though the argument is demoting a value, that value is only the one that is the least preferred value for the agent. We can therefore rewrite the arguments of the form PA2 as an argument of the form PA4.

PA4: In Q_n action Ac_n should be performed by Ag_i to reach Q_m even though it realises ϕ which only demotes V_n.

PA4 can now be subjected to critical questions of the form PCQ1 if Ac_n reaches a state demoting a value preferred to V_n. PCQ2 does not arise in the situation when the target state is already not desirable. PCQ3 can be used because the state may still be only locally the least bad; it may be that subsequent moves will eventually result in a much worse state, which could have been avoided by choosing an initially less attractive option.

In the next section we will briefly describe a program which instantiates these argument schemes and critical questions, so as to deliberate on the best choice of action for an agent with given value preferences.

4. Program

We have implemented the above in Java to produce a program which takes the description of the problem given in Table 1 and generates the transition matrix, also given in Table 1, for the initial state of interest, represented by Figure 2. The program then calculates the values that are promoted or demoted by each action when the agents move from state to state shown as the labels on the arcs of these figures. It then generates the arguments for making or not making a particular action from each state depending on the values promoted or demoted by instantiating the argument schemes PA1-4. Using the arguments, the program attacks the arguments using the three critical questions so as to select the justified action. This operation will be illustrated by the example in the next section.

5. Example

What an agent should do will vary according to the ordering the agent places on values. Agents can make different choices depending on whether they are selfish or act in a selfless manner. In [4], the authors described five different value orderings: Morally Correct, Selfish, Self-Interested, Noble and Sacrificial. For all agents life is preferred to wealth. For a Selfish agent, each agent prefers its own interests to any interests of the other agent. For Hal, $L_H > W_H > L_C > W_C$ and for Carla, $L_C > W_C > L_H > W_H$. We will consider the case where both agents are selfish.

From the initial state, q_{253}, four arguments are produced, two for Hal and two for Carla. The best argument for Hal is A3 and the best argument for Carla is A4:

A3: In q_{253} $take_H$ should be performed by Hal, to reach q_{165} which realises $isAlive_H = 2$ which promotes L_H

A4: In q_{253} $doNothing_C$ should be performed by Carla, to reach q_{165} which realises $isAlive_H = 2$ which promotes L_H

There are no critical questions posed for A3 so Hal performs the $take_H$ action, and there are no critical questions posed for A4 so Carla performs the $doNothing_C$ action. This produces the joint action $\langle take_H, doNothing_C \rangle$ and the agents reach q_{165}.

For q_{165} there are twelve arguments produced, six for Hal and six for Carla. Each agent has one PA1 argument and five PA2 arguments. The arguments are listed below.

A1: In q_{165} $doNothing_H$ should not be performed by Hal, to avoid q_{164} which realises $isAlive_C = 1$ which demotes L_C

A2: In q_{165} $doNothing_C$ should not be performed by Carla, to avoid q_{164} which realises $isAlive_C = 1$ which demotes L_C

A3: In q_{165} $doNothing_H$ should not be performed by Hal, to avoid q_{169} which realises $hasMoney_C = 0$ which demotes W_C

A4: In q_{165} buy_C should not be performed by Carla, to avoid q_{169} which realises $hasMoney_C = 0$ which demotes W_C

A5: In q_{165} $compensate_H$ should not be performed by Hal, to avoid q_{145} which realises $hasMoney_H = 0$ which demotes W_H

A6: In q_{165} $doNothing_C$ should not be performed by Carla, to avoid q_{145} which realises $hasMoney_H = 0$ which demotes W_H

A7: In q_{165} $compensate_H$ should be performed by Hal, to reach q_{145} which realises $hasMoney_C = 2$ which promotes W_C

A8: In q_{165} $doNothing_C$ should be performed by Carla, to reach q_{145} which realises $hasMoney_C = 2$ which promotes W_C

A9: In q_{165} $compensate_H$ should not be performed by Hal, to avoid q_{145} which realises $isAlive_C = 1$ which demotes L_C

A10: In q_{165} $doNothing_C$ should not be performed by Carla, to avoid q_{145} which realises $isAlive_C = 1$ which demotes L_C

A11: In q_{165} $compensate_H$ should not be performed by Hal, to avoid q_{154} which realises $hasMoney_H = 0$ which demotes W_H

A12: In q_{165} buy_C should not be performed by Carla, to avoid q_{154} which realises $hasMoney_H = 0$ which demotes W_H

Hal starts with his PA1 argument, A7. A7 is attacked by PCQ1 using the PA2 argument A5 because the action $compensate_H$ could reach the state q_{145} where the value W_H is demoted which Hal prefers to the value W_C being promoted by A7. A5 is in turn attacked by the PCQ1 rebuttal using A6 because Carla also does not want to reach q_{145} because of the demoted value W_H. A6 is then attacked and defeated by the PCQ1 counter-rebuttal using A8 because Carla has a better reason to reach q_{145} than to avoid it. This is because q_{145} promotes W_C which Carla prefers over W_H. A5 is then attacked and defeated by the PCQ1 rebuttal using A10 because Carla has a reason to avoid q_{145} and does not have a better reason to reach it. This is because q_{145} demotes L_C which is the most important value for Carla. A7 thus survives this line of questioning.

A7 is then attacked by PCQ1 using A9 because Hal does not want to use the action $compensate_H$ because he may reach q_{145} which demotes the value L_C which Hal prefers to the value W_C promoted by A7. A9 is attacked by the PCQ1 rebuttal using A6 which is then attacked and defeated by the PCQ1 counter-rebuttal using A8. A9 is finally attacked and defeated by the PCQ1 rebuttal using A10.

A7 is then attacked by PCQ1 using A11 because Hal does not want to perform the action $compensate_H$ and reach the state q_{154} because the value W_H would be demoted. A11 is attacked and defeated by the PCQ1 rebuttal using A12 because Carla has a reason to avoid q_{154} and does not have a better reason to reach it. A7 is finally attacked and defeated by PCQ2 using A10 because Carla has a reason not to want to reach q_{145} and no rebuttal argument to want to reach it.

Now Hal has exhausted his supply of PA1 arguments so he must now use his PA2 arguments. These are ordered in reverse order of his value preferences so that the order Hal uses is (W_C = A3) > (L_C = (A1, A9)) > (W_H = (A5, A11)). Hal takes the PA2 argument with the least favoured value, A3, and rewrites it in the form of PA4.

A3' In q_{165} $doNothing_H$ should be performed by Hal, to reach q_{169} even though it realises $hasMoney_C = 0$ which only demotes W_C

A3' is attacked by PCQ1 using A1 because the action $doNothing_H$ can reach the state q_{164} where the value L_C is demoted which Hal prefers to the value W_C demoted in A3'. A1 is attacked and defeated by the PCQ1 rebuttal using A2 because Carla has a reason to avoid q_{164} and no better reason to reach it.

A3' is not attacked by PCQ3 because not all of the states reachable from q_{169} are bad states. This means that A3' is not defeated so Hal will perform the $doNothing_H$ action even though it demotes the value W_C.

Carla first starts with her PA1 argument, A8. A8 is attacked by PCQ1 using A2 because Carla does not want to do the action $doNothing_C$ to avoid q_{164} where the more preferred value of L_C is demoted. A2 is then attacked and defeated by the PCQ1 rebuttal using A1 because Hal also does not want to reach q_{164} and does not have a better reason to reach it. A8 is then attacked by PCQ1 using A10 because Carla does not want to do the action $doNothing_C$ to avoid q_{145} where the more preferred value of L_C is demoted. A10 is then attacked and defeated by the PCQ1 rebuttal using A5 because Hal also does not want to reach q_{145}. Finally A8 is attacked and defeated by PCQ2 using A5 because Hal does not want to reach q_{145} because the value of W_H is demoted.

Carla has now used all of her PA1 arguments and now must use the set of PA2 arguments. These are ordered in reverse to her value preferences and the order Carla uses

is (W_H = (A6, A12)) > (W_C = A4) > (L_C = (A2, A10)). Carla takes the first of these arguments, A6, and rewrites it in the form of PA4.

A6' In q_{165} $doNothing_C$ should be performed by Carla, to reach q_{145} even though it realises $hasMoney_H = 0$ which only demotes W_H

A6' is attacked by PCQ1 using A2 because Carla does not want to do the action $doNothing_C$ to avoid q_{164} where the more preferred value, L_C, is demoted. A2 is then attacked and defeated by the PCQ1 rebuttal using A1 because Hal also does not want to reach q_{164}, since he prefers L_C to W_C. A6' is then attacked by PCQ1 using A10 because Carla does not want to do the action $doNothing_C$ and risk q_{145} where the more preferred value L_C is again demoted. A10 is then attacked and defeated by the PCQ1 rebuttal using A5 because Hal also does not want to reach q_{145}. Finally A6' is attacked by PCQ3 because all of the end states reachable from q_{145} are worse than q_{145}. However this argument is attacked and defeated by the PCQ3 rebuttal using A5 because Hal does not want to reach q_{145} either. This means that A6' is not defeated and so Carla will perform the action $doNothing_C$. Based of this joint action the agents reach q_{164}. Note that the effect is for both agents to wait since neither wishes to spend their money if they can avoid it.

From the state, q_{164}, sixteen arguments are produced, eight for Hal and eight for Carla. The best argument for Hal is A5 and the best argument for Carla is A6:

A5: In q_{164} $doNothing_H$ should be performed by Hal, to reach q_{169} which realises $isAlive_C = 2$ which promotes L_C

A6: In q_{164} buy_C should be performed by Carla, to reach q_{169} which realises $isAlive_C = 2$ which promotes L_C

There are no critical questions posed for either argument so the joint action $\langle doNothing_H, buy_C \rangle$ is performed and the agents reach q_{169}. This means the agents both have insulin. Note that here, because Hal is selfish, Carla is forced by the threat to her life which is now immediate to buy her own insulin. Different value preferences for the different agents produce different outcomes, as described in [4].

6. Concluding Remarks

In this paper we have described an implemented realisation of the approach to deliberation using presumptive argumentation and associated critical questions. The program is built on a representation of the problem domain as an Alternating Time Transition System with agents represented as automata. The program instantiates the arguments scheme and then subjects it to critical questioning. We have explored automation of argumentation for practical reasoning by a single agent in a multi-agent context, where agents may have conflicting values. Traditionally, reasoning about actions between potentially self-interested agents in a multi-agent context has used Game Theory. Game Theory, however, has not been concerned with rational justification or explanation of action, which is a key focus of work in argumentation.[1]

[1]The authors are grateful to EC project ASPIC (IST-FP6-002307) for financial support.

References

[1] A. J. P. Kenny. Practical Reasoning and Rational Appetite, 1975. Reprinted in [9].
[2] D. N. Walton. Argument Schemes for Presumptive Reasoning. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.
[3] K. Atkinson. What Should We Do?: Computational Representation of Persuasive Argument in Practical Reasoning. PhD Thesis, Department of Computer Science, University of Liverpool, Liverpool, UK, 2005.
[4] K. Atkinson and T. Bench-Capon. Addressing Moral Problems Through Practical Reasoning. In: Proceedings of the Eighth International Workshop on Deontic Logic in Computer Science (DEON 2006). To appear.
[5] M. Wooldridge and W. van der Hoek. On Obligations and Normative Ability: Towards a Logical Analysis of the Social Contract. In Journal of Applied Logic, 3:396-420, 2005.
[6] J. Coleman. Risks and Wrongs. Cambridge University Press, 1992.
[7] C. G. Christie. The Notion of an Ideal Audience in Legal Argument. Kluwer Academic, 2000.
[8] L. Amgoud, C. Cayrol, M. Lagasquie-Schiex. On the Bipolarity in Argumentation Frameworks. NMR 2004: 1-9.
[9] J. Raz, editor. Practical Reasoning. Oxford University Press, Oxford, UK, 1978.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

The Logic of Multiple-Valued Argumentation and its Applications to Web Technology

Extended Abstract

Hajime Sawamura [a,1], Toshiko Wakaki [b] and Katsumi Nitta [c]
[a] *Niigata University, Niigata, Japan*
[b] *Shibaura Institute of Technology, Saitama, Japan*
[c] *Tokyo Institute of Technology, Tokyo, Japan*

Abstract. Extended Annotated Logic Programming (EALP) for argumentation knowledge and Logic of Multiple-valued Argumentation (LMA) built on top of EALP are outlined together with its implementation. We then explore a new application area of argumentation in the present rapidly growing web technology, and demonstrate the basic features and advantages of Automated Argument System based on Logic of Multiple-Valued Argumentation by specializing it to two argument systems derived from LMA: (i) an argument-based recommender system, and (ii) an integrated system of semantic web reasoning and argument-based reasoning.

Keywords. argumentation, multiple-valuedness, semantic web reasoning, uncertainty, description logic

1. Introduction

Arguing is a powerful tool we use individually and socially in daily life. This is also true in the agents' world. Computational argumentation has been accepted as a social computing mechanism in the multi-agent systems research community.

Argumentation is by nature a logic of reasoning under an uncertain or changing information environment. The Logic of Multiple-valued Argumentation (LMA) was built on top of Extended Annotated Logic Programming (EALP) [1]. LMA takes into account two kinds of intrinsic uncertainty: (1) agents can not have a perfect knowledge/belief base for the universe of discourse in resolving goals and making decisions, and (2) propositions in a knowledge/belief base can hold only partial or vague information. EALP and LMA have been devised to satisfy these two aspects of uncertainty recognition simultaneously, and differently from other approaches to argumentation theory and practice [2] [3]. It is also a very general logic of multiple-valued argumentation in the sense that it allows us to specify various types of truth values depending on application domains and

[1]Correspondence to: Hajime Sawamura, 8050, 2-cho, Ikarashi, Niigata, Japan. Tel.: +81 25 262 6753; Fax: +81 25 262 6753; E-mail: sawamura@ie.niigata-u.ac.jp

to deal with uncertain arguments under the specified truth values. Furthermore, LMA is notable for allowing us to deal with culturally unique arguments, that is, not only logos-oriented arguments in the West, but also tetralemma-oriented ones in the East [4].

In this paper, after briefly introducing EALP and LMA by describing some of their features and an automated argument system based on them in the next section, we explore a new application area in web technology that seems to be presently a very hot area in information technology. Section 3 describes an argument-based recommender system. We bring LMA to web search engines in order to raise the quality of the web search results so that they are more suited to users' preferences. Section 4 describes an integrated system of semantic web reasoning and argument-based reasoning. We develop a novel attempt showing how argumentation can contribute to the Semantic Web whose vision and aim are to enrich the existing Web with meta-data and processing methods so as to provide web-based systems with advanced capabilities such as argumentative reasoning.

2. Outline of EALP and LMA

EALP (Extended Annotated Logic Programming) is an expressive logic programming language we extended for argumentation by incorporating default negation into the Generalized Annotated Logic Programming of Kifer and Subrahmanian [5]. EALP has two kinds of explicit negation: Epistemic Explicit Negation '$\neg$' and Ontological Explicit Negation '$\sim$', and the default negation '**not**'. They are intended to yield a driving force for argumentation and dialogue in LMA below. The basic language constituents are literals with truth-values or epistemic states of agents explicitly annotated. The structure of truth-values is required to be a complete lattice so that the paraconsistency of an agent's knowledge base is guaranteed under the ideals-based semantics [1]. LMA is a logic of multiple-valued argumentation constructed on top of EALP. LMA allows agents to construct arguments under uncertain knowledge in EALP and to argue with other agents on uncertain issues in an open networked heterogeneous environment. As we can specify truth values for every application domain that has its own proper uncertainty in EALP, such diversity of truth values brings us an extensive applicability of LMA. In what follows, we will illustrate how uncertain arguments proceed in LMA using a simple argument with a somewhat deviant use of truth values, without presenting lengthy definitions.

Example 1 (Job schedule management) *Let us consider an argument about monthly job schedule management. Here we use an unconventional complete lattice of truth values which is the power set* $\mathcal{P}(\{1,\ldots,31\})$ *of the set of the monthly dates, ordered by set inclusion. An annotated atom* $work(a) : \{5,6\}$*, for example, reads "Agent* a *works on the 5th and the 6th". It asserts that the proposition* $work(a)$ *is true only in a certain time interval.* $\sim work(a) : \{5,6\}$ *reads "Agent* a *does not work on the 5th and the 6th". We define the epistemic explicit negation to be* $\neg A : \mu = A : \neg\mu$ *and* $\neg\mu = \{1,\ldots,31\} - \mu$ *as in GAP [5], and thus* $\neg work(a) : \{5,6\}$ *reads "Agent* a *works on every day except the 5th and the 6th". The difference between the ontological and epistemic explicit negations is obvious. Under this complete lattice of truth values, we consider* $MAS = \{KB_m, KB_a, KB_b, KB_o\}$*, where the knowledge base* KB *of each agent is, in EALP,*

$KB_m = \{\ finish(project):\{6\} \leftarrow work(a):\{3,4,5\}\ \&\ arrive(component):\{5\},$
$work(a):\{3,4,5\} \leftarrow,\ arrive(component):\{5\} \leftarrow,\ pay(upcharge):\{8\} \leftarrow\ \},$
$KB_a = \{\ \sim work(a):\{5\} \leftarrow \mathbf{not}\ work(b):\{5\}\ \&\ holiday:\{5\},$
$\sim work(a):\{12\} \leftarrow \mathbf{not}\ work(b):\{12\}\ \&\ holiday:\{12\},\ holiday:\{5,6,12,13\} \leftarrow\ \},$
$KB_b = \{\neg work(b):\{12,19,26\} \leftarrow,\ holiday:\{5,6,12,13\} \leftarrow\ \},$
$KB_o = \{\sim arrive(component):\{5\} \leftarrow \mathbf{not}\ pay(upcharge):\phi\ \}.$
KB_m, KB_a, KB_b and KB_o denote the knowledge bases of a manager agent m, employee agents a, b and a subcontractor agent o respectively. Agent m's argument which has the conclusion $finish(project):\{6\}$ (the project should finish on the 6th) is justified by the dialectical proof theory as shown in Figure 1. In the winning dialogue tree,

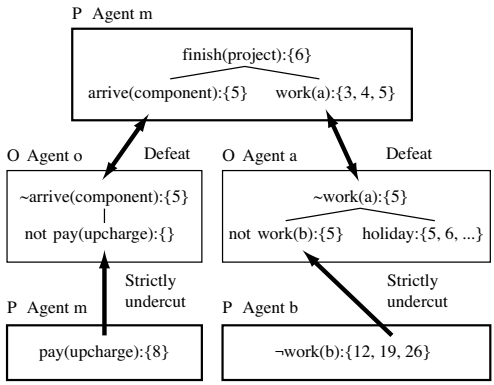


Figure 1. The winning dialogue tree in Example 1

initially Agent m (P: Proponent) says "if a component will arrive on the 5th, and Agent a works on the 3th, 4th and 5th, then the project will finish on the 6th": the formal argument is constructed from its knowledge base KB_m as
[$finish(project):\{6\} \leftarrow work(a):\{3,4,5\}\ \&\ arrive(component):\{5\},\ work(a):\{3,4,5\} \leftarrow,$
$arrive(component):\{5\} \leftarrow$].
There are two places that can be attacked by the other party (O: opponent). In the left branch of the dialogue tree, Agent o defeats it as follows "I will be unable to bring a component on the 5th if the additional charge is not paid". But Agent m strictly undercuts o's argument by saying "I will pay it to you on the 8th". In the right branch of the dialogue tree, the first argument of Agent m, is defeated by Agent a (O: Opponent) saying "the 5th is a holiday, and if the coworker b does not work, I do not want to work on the 5th": the formal argument is constructed from its knowledge base KB_a as
[$\sim work(a):\{5\} \leftarrow \mathbf{not}\ work(b):\{5\}\ \&\ holiday:\{5\},\ holiday:\{5,6,12,13\} \leftarrow$].
This is a semantically correct argument since $holiday:\{5\}$ can be resolved upon $holiday:\{5,6,12,13\}$ with the condition $\{5\} \leq \{5,6,12,13\}$ in GAP and EALP. Agent a can put forward such a counter-argument since the conclusion of Agent a's argument $\sim work(a):\{5\}$ conflicts with the second rule of Agent m, $work(a):\{3,4,5\} \leftarrow$. This is due to the defeat (rebut) relation that $A:\mu$ conflicts with $A:\rho$ each other provided that $\mu \geq \rho$ or $\rho \geq \mu$ in LMA. In fact, Agent a claims that I do not want to work on the 5th, but Agent m asserts that it works on {3, 4, 5} which is a superset of {5}.

However Agent b (P: Proponent) strictly undercuts this by saying "I will work on days except the 12th, 19th and 26th": with formal argument constructed from its knowl-

edge base KB_b as: [$\neg work(b):\{12,19,26\} \leftarrow$].
This is equivalent to $\neg work(b):D \leftarrow$, where $D = \{1,...,31\} - \{12,19,26\}$ and hence can undercut the first rule of Agent a's counter-argument above. This is due to the strict undercut relation that $A:\mu$ can attack **not** *$A:\rho$ in one way provided that $\mu \geq \rho$ in LMA. In fact, Agent b claims that it works on dates D including {5}, but Agent a asserts that it does not work on the 5th.*

There are no further arguments at this stage of the argumentative dialogue, and it finishes at the proponent's move. As a result the first argument of Agent m is justified.

3. Two Contributions to Web Technology through Argumentation

3.1. Argument-based recommender system

Chesñevar and Maguitman proposed ArgueNet, a recommender system that integrates a traditional web search engine with a defeasible argumentation framework [6]. ArgueNet presents web search results classified according to preference criteria declaratively and qualitatively specified by the user in Defeasible Logic Programaming (DeLP). It is remarkable not only as a prominent exploration of the application area of argumentation, but also as a promising approach to further augmenting existing web search engines.

We developed a refined version of ArgueNet by using our LMA in place of their two-valued argumentation framework of DeLP, so that LMA with multiple values as its truth-values specified by users allows for uncertain argumentation, resulting in more refined classified web search results. In our system called LMA-WebArgNet, user preferences for prioritizing and classifying search results are represented in EALP. Then we employed two types of truth-values: the complete lattice $\mathcal{FOUR} = (\{\bot, \mathbf{t}, \mathbf{f}, \top\}, \leq)$, where $\forall x, y \in \{\bot, \mathbf{t}, \mathbf{f}, \top\}\ x \leq y \Leftrightarrow x = y \vee x = \bot \vee y = \top$ and a complete lattice of the unit interval of real numbers $\Re[0,1]$.

In Arguenet, the search results, $[s_1, s_2, ..., s_n]$, are classified into three sets as the result of arguing issues of the form $rel(s_i)$ to represent its relevance to users' preferences: warranted search results S^w, undecided search results S^u, and defeated search results S^d, and the recommended priority order is $S^w > S^u > S^d$. In LMA-WebArgNet, it is further refined as follows in the case of $\mathcal{FOUR}$, for example:

- $S^r(Relevant\ search\ results)$: those search results s_i having a justified argument supporting $rel(s_i):\mathbf{t}$ and no arguments supporting $rel(s_i):\mathbf{f}$.
- $S^{mr}(Possibly\ relevant\ search\ results)$: those search results s_i having a justified argument supporting $\sim rel(s_i):\mathbf{f}$ and no arguments supporting $rel(s_i):\mathbf{t}$.
- $S^{ri}(Both\ relevant\ and\ irrelevant\ search\ results)$: those search results s_i having a justified argument supporting $rel(s_i):\top$.
- $S^{mi}(Possibly\ irrelevant\ search\ results)$: those search results s_i having a justified argument supporting $\sim rel(s_i):\mathbf{t}$ and no arguments supporting $rel(s_i):\mathbf{f}$.
- $S^r(Irrelevant\ search\ results)$: those search results s_i having a justified argument supporting $rel(s_i):\mathbf{f}$ and no arguments supporting $rel(s_i):\mathbf{t}$.
- $S^u(Undecided\ search\ results)$: otherwise.

In ArgueNet, $S^r(Relevant\ search\ results)$, $S^r(Irrelevant\ search\ results)$ and $S^u(Undecided\ search\ results)$ have been dealt with as S^w, S^d and S^u respectively.

The other cases in LMA-WebArgNet yield more detailed uncertainty in judging web search results and hence a much finer classification of web search results. Then, LMA-WebArgNet recommends these in the order $S^r > S^{mr} > S^{ri} > S^u > S^{mi} > S^i$.

3.2. An integrated system of semantic web reasoning and argument-based reasoning

In argument-based multi-agent systems developed so far, each agent interacts with other agents to negotiate or achieve a mutually acceptable agreement using only rule-based knowledge where ontological knowledge has not been taken into account. Aiming at a promising reasoning facility for the Semantic Web, we have attempted to build an integrated system in which an LMA-based argument system can inquire of the description logic reasoning system about ontological knowledge that has not been sufficiently supplied to agents or is lacking in agents' knowledge bases. In doing so, we exploit the following scenario and technical results for realizing computationally feasible ontology reasoning and augmenting argumentative reasoning associated with it (see Figure 2 for the relation of each component in the system architecture of the integrated system):

1. Ontology translation from OWL DL to DL $\mathcal{SHOIN}$(D) (and vice versa)
2. DL reasoning system (i.e. the theorem prover for description logics)
3. LMA augmented with a capability of inquiring about ontology.

For the first component, it is shown that ontology entailment in OWL Lite and OWL DL reduces to knowledge base (un) satisfiability in description logics such as the DL $\mathcal{SHIF}$(D) and the DL $\mathcal{SHOIN}$(D) respectively [7]. Following this result, an ontology expressed in OWL DL on the Semantic Web is translated into semantically equivalent ones in DL $\mathcal{SHOIN}$(D) expression by means of XSLT.

The second component, DL reasoning system, decides whether a query α given by an agent is a *logical consequence* of the description logic knowledge base $\mathcal{K}$, denoted $\mathcal{K} \models \alpha$. For reasoning with the knowledge base $\mathcal{K}$ expressed by DL $\mathcal{SHOIN}$(D), we adopted Heymans et al's method [8], which enables a simulation of description logics with DL-safe rules by using the normal finite answer set programming (ASP). Our DL reasoning system has been implemented using the ASP solver DLV [9].

The LMA-based argument system has been slightly augmented with a reasoning interface between the DL reasoning system and LMA, through which the LMA-based argument system is allowed to consult the DL reasoning system for unknown or unresolved literals in putative arguments in the process of argumentation. Consulting in this manner amounts to the following theoretical setting: each agent A_i virtually possesses the following knowledge base P_i in EALP on the complete lattice $\mathcal{T} = (\{\mathbf{t}, \mathbf{f}\}, \leq)$ such that $\mathbf{f} \leq \mathbf{t}$ and attends the argumentation: $P_i = KB_i \cup \{l : \mathbf{t} \leftarrow \mid \mathcal{K} \models l \; for \; l \in Lit_{\mathcal{K}}\}$, where KB_i is a set of EALP rules and $Lit_{\mathcal{K}} \stackrel{def}{=} \{a | a \in H_{\mathcal{K}}\} \cup \{\neg a | a \in H_{\mathcal{K}}\}$ with an ontological knowledge base, $\mathcal{K}$ and the Herbrand base of $\mathcal{K}$, $H_{\mathcal{K}}$. Here, it is assumed that each agent A_i $(1 \leq i \leq n)$ has its own knowledge base KB_i and the ontological knowledge base, $\mathcal{K}$, is shared among agents concerned.

4. Concluding Remarks

We have described the elements of our logic of multiple-valued argumentation by using a simple argument example, and then pursued its applicability to web technology. The

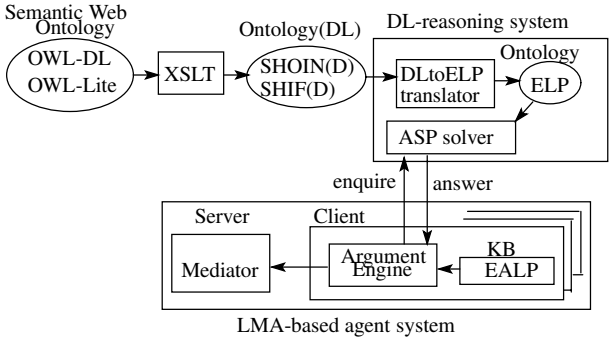


Figure 2. Architecture of the integrated system of ontology reasoning and argumentative reasoning

first application might be very time-consuming for big search data since the finer the classification is, the longer the processing time is. Speeding up argumentation in one way or another would become critical. The second application is obviously a novel attempt worthy of special mention since it concerns argumentative reasoning associated with ontology reasoning. We believe the integration of these two reasoning techniques opens up a new vista of the future of the Semantic Web and semantic web reasoning.

Acknowledgments The systems were implemented by our students: T. Fukumoto, K. Matsunaga, N. Desaki, T. Mukai and H. Miyashita. We are also grateful to Prof. Carlos Chesñevar for informing us of his stimulating paper on ArgueNet.

References

[1] T. Takahashi and H. Sawamura. A logic of multiple-valued argumentation. In *Proc. AAMAS 2004*, pages 800–807. ACM, 2004.

[2] C. I. Chesnevar, G. Maguitman, and R. P. Loui. Logical models of argument. *ACM Computing Surveys*, 32:337–383, 2000.

[3] H. Prakken and G. Vreeswijk. Logical systems for defeasible argumentation. In *D. Gabbay and F. Guenther, editors, Handbook of Philosophical Logic*, pages 219–318. Kluwer, 2002.

[4] H. Sawamura, T. Takahashi, and K. Matsunaga. An eastern specialization of logic of multiple-valued argumentation to tetralemma originated in india. In *Proc. of 2nd Indian International Conf. on Artificial Intelligence (IICAI-05)*, pages 1274–1291, 2005.

[5] M. Kifer and V. S. Subrahmanian. Theory of generalized annotated logic programming and its applications. *J. of Logic Programming*, 12:335–397, 1992.

[6] C. I. Chesñevar and Ana G. Maguitman. Arguenet: An argument-based recommender system for solving web search queries. In *Proc. of the 2nd. International IEEE Conf. on Intelligent Systems (IS 2004)*. IEEE Press, 2004.

[7] I. Horrocs and P. F. Patel-Schneider. Reducing owl entailment to description logic satisfiability. In *Proc. ISWC'2003*, volume LNCS 2870, pages 17–29. Springer, 2003.

[8] S. Heymans, D. V. Nieuwenborgh, and D. Vermeir. Nonmonotonic ontological and rule-based reasoning with extended conceptual logic programs. In *Proc. of ESWC'05*, volume 3532, pages 392–407. Springer, 2005.

[9] T. Eiter, N. Leone, C. Mateis, G. Pfeifer, and F. Scarcello. A deductive system for nonmonotonic reasoning. In *Proc. of LPNMR 1997*, volume 1265, pages 364–375. Springer, 1997.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Value Based Argumentation in Hierarchical Argumentation Frameworks

S. Modgil
Advanced Computation Lab, Cancer Research UK, London WC2A 3PX

Abstract. Hierarchical argumentation frameworks organise Dung argumentation frameworks into a hierarchy so that argumentation over preference information in a level n Dung framework is used to resolve conflicts between arguments in a level n-1 framework. In this paper we formalise and extend value based argumentation [1] in a hierarchical argumentation framework and illustrate application of the resulting framework to argumentation over action.

1. Introduction

Dung's influential theory of argumentation [2] evaluates the status of arguments by applying a 'calculus of opposition' to a framework $(Args, \mathcal{R})$. The structure of arguments $Args$ and definition of the conflict based binary relation $\mathcal{R}$ on $Args$ is left unspecified. This enables different argumentation systems with their own defined language, construction of arguments, definition of conflict and relation $\mathcal{R}$, to instantiate a Dung framework in order to evaluate the status of the system's constructed arguments. Evaluation usually requires some notion of preference to resolve conflicts between arguments. In argumentation terms this means that the defined $\mathcal{R}$ accounts for a preference ordering on arguments based on their relative strength. However, information relevant to establishing a preference ordering ('preference information') may itself be incomplete, uncertain or conflicting. Hence, in [3] we present what we believe to be the first framework for reasoning about - indeed **arguing** about - preference information.

Starting with a Dung framework containing arguments $A1$ and $A2$ that conflict with each other, one could in some meta-logic reason that: 1) $A1$ is preferred to $A2$ because of c (= $B1$), **and** 2) $A2$ is preferred to $A1$ because of c' (= $B2$). Hence, to resolve the conflict between $A1$ and $A2$ requires 'meta-argumentation' to determine which of the conflicting *arguments* $B1$ or $B2$ is preferred. Of course, one may need to ascend to another level of argumentation if there are conflicting arguments $C1$ and $C2$ respectively justifying a preference for $B1$ over $B2$ and $B2$ over $B1$. Requirements for this type of 'hierarchical argumentation' arise from the fact that different principles and criteria [4] may be used to valuate the strength of arguments. For example, $A1$ may be preferred to $A2$ by the 'weakest link' principle [4] whereas $A2$ may be preferred to $A1$ based on the 'last link' principle [5]. One may then need to ascend to another level of argumentation to construct contextual arguments justifying use of one principle in preference to the other. Also, for any given principle, the valuations of arguments may vary according to perspective. One perspective or source of information for valuating argument strength may indicate

that $A1$ is preferred to $A2$, whereas from another perspective $A2$ is preferred to $A1$. To resolve the conflict requires arguing for a preference between perspectives.

We have therefore formalised a hierarchy of Dung frameworks in which level n arguments refer to level $n-1$ arguments and conflict based relations and preferences between level $n-1$ arguments. The generality of our approach resides in the fact that the level 1 framework makes no commitment to the system instantiating it, a minimal set of commitments are made to first order logic based argumentation systems instantiating frameworks at level $n > 1$, and any one of a number of principles can be used for valuating argument strength. In this paper we substantiate the generality of our approach by formalising and extending value based argumentation [1] in a hierarchical argumentation framework (HAF).

In value based argumentation frameworks (VAF), an argument $A1$ successfully attacks (defeats) $A2$ only if the value promoted by $A2$ is not ranked higher than the value promoted by $A1$ according to some strict partial ordering on values. By formalising value based argumentation in a HAF, we extend the former in a number of ways:

- Arguments can promote (or demote) values to a given degree, so that if $A1$ and $A2$ promote the same value V, we can have that $A1$ successfully attacks $A2$ if it promotes V to a greater degree than $A2$. Requirements for extending VAF in this way are highlighted in [6] and in particular arise in value based argumentation over action [7].
- VAF acknowledges that different orderings on values may apply [1]. Hence, viewing orderings as analogous to the 'principles' referred to above, two conflicting arguments may defeat each other according to different orderings, requiring that one construct some context based justification for preferring one ordering over another in order to resolve the conflict.
- Information sources may ascribe different values to arguments (the motivating values for an action in the case of argumentation over action), or, for any given value sources may disagree on the degree to which an argument promotes that value. This may also lead to contradictory preference information and result in conflicting arguments defeating each other. Hence, resolution of the conflict requires argumentation over which source is preferred.

The remainder of this paper is structured as follows. Section 2 reviews Dung's theory and our formalisation of hierarchical argumentation frameworks. This work is also reported on in [3]. In section 3 we formalise value based argumentation in HAF and show how the extensions described above are required in the context of argumentation over action. In particular, we illustrate with an example taken from [7] in which value based argumentation is applied to arguments for action instantiating a presumptive schema, and attacking arguments instantiating critical questions associated with the schema. Section 4 concludes with a discussion of future and related work.

2. Hierarchical Argumentation Frameworks

Argumentation systems are built around a logical language and associated notion of logical consequence $\Gamma \vdash \alpha$. If $\Delta \subseteq \Gamma$ is the set of premises from which α is inferred, then an argument A claiming α can be represented by the pair (Δ, α). We say that:

- *support*(A) = Δ and *claim*(A) = α.
- A is *consistent* if *support*(A) is consistent (*support*(A) $\nvdash \bot$)
- A' is a *strict sub-argument* of A if *support*(A') $\subset$ *support*(A).

The conflict based *attack* relation is then defined amongst the constructed arguments, whereupon the *defeat* relation is defined by additionally accounting for the relative strength of (preferences between) the attacking arguments. A Dung framework [2] can then be instantiated by the system's constructed arguments and their relations. Here, we define two notions of a Dung framework:

Definition 1 Let Args be a finite set of arguments. An attack argumentation framework AF_{at} is a pair (Args, $\mathcal{R}_{at}$) where $\mathcal{R}_{at} \subseteq$ (Args $\times$ Args). A defeat argumentation framework AF_{df} is a pair (Args,$\mathcal{R}_{df}$) where $\mathcal{R}_{df} \subseteq$ (Args $\times$ Args)

If (A, A'),$(A', A) \in \mathcal{R}_{at}$ then A and A' are said to symmetrically attack or *rebut* each other, denoted by $A \rightleftharpoons A'$. If only $(A, A') \in \mathcal{R}_{at}$, then A asymmetrically attacks, or *undercuts* A', denoted by $A \rightharpoonup A'$. Where there is no possibility of ambiguity we also use $\rightleftharpoons$ and $\rightharpoonup$ to denote symmetric and asymmetric defeats. We also use this notation to denote frameworks, e.g., ($A \rightleftharpoons A'$, A'') denotes ($\{A, A', A''\}, \{(A, A'), (A', A)\}$).

An argument is justified if it belongs to all *acceptable* extensions of a framework, where the notion of acceptability is defined for different semantics [2]. Here, we focus on the preferred semantics.

Definition 2 Let E be a subset of $Args$ in AF = AF_{at} or AF_{df}, and let $\mathcal{R}$ denote either $\mathcal{R}_{at}$ or $\mathcal{R}_{df}$. Then:

- E is *conflict-free* iff $\nexists$ A, $A' \in$ E such that (A,A') $\in \mathcal{R}$
- An argument A is *collectively defended* by E iff $\forall A'$ such that $(A', A) \in \mathcal{R}$, $\exists$ $A'' \in$ E such that $(A'', A') \in \mathcal{R}$.

Let E be a *conflict-free* subset of $Args$, and let F: $2^{Args} \rightarrow 2^{Args}$ such that $F(E)$= $\{$A $\in Args$ | A is collectively defended by $E\}$.

- E is an an admissible extension of AF iff $E \subseteq F(E)$
- E is a preferred extension of AF iff E is a maximal (w.r.t set inclusion) admissible extension

Let $\{E_1, \ldots, E_n\}$ be the set of all preferred extensions of AF. Let A $\in Args$. Then A $\in$ *justified*(AF) iff A $\in \bigcap_{i=1}^{n} E_i$

Hierarchical argumentation aims at argumentation over preference information so as to define the *defeat* relation on the basis of the *attack* relation and thus enable resolution of conflicts between attacking arguments. In general, A defeats A' if A attacks A', and A' does not 'individually defend' itself against A's attack, ie.:

$$\mathcal{R}_{df} = \mathcal{R}_{at} - \{(A, A') \mid defend(A', A)\}$$

where A' individually defends itself against A if A' is preferred to (and in some cases may be required to attack) A. Hence, given AF_{at_1} = (Args_1, $\mathcal{R}_{at_1}$) instantiated by some argumentation system, then to obtain AF_{df_1} = (Args_1, $\mathcal{R}_{df_1}$) we can reason in some first order logic about the strengths and relative preferences of arguments in Args_1, to infer wff of the form $defend(\mathcal{A}', \mathcal{A})$ (where $\mathcal{A}'$ and $\mathcal{A}$ name arguments $A', A \in \text{Args}_1$).

For example, suppose $\text{AF}_{at_1} = (A1 \rightleftharpoons A2)$. Neither $A1$ or $A2$ are justified. Inferring $defend(\mathcal{A}1, \mathcal{A}2)$ we obtain $\text{AF}_{df_1} = (A1 \rightharpoonup A2)$. $A1$ is now justified.

However, one might be able to infer that $A1$ is preferred to and so defends $A2$'s attack, **and** that $A2$ is preferred to and so defends $A1$'s attack. Hence the requirement that the first order logic itself be the basis for an argumentation system instantiating $\text{AF}_{at_2} = (\text{Args}_2, \mathcal{R}_{at_2})$ (practical systems for first order argumentation are described in [8]). Arguments B and B' in Args_2, with respective claims $defend(\mathcal{A}2, \mathcal{A}1)$ and $defend(\mathcal{A}1, \mathcal{A}2)$, attack each other. If B is justified then $A2$ asymmetrically defeats $A1$, else if B' is justified then $A1$ asymmetrically defeats $A2$ in AF_{df_1}. Of course, to determine which of B and B' are justified requires determining which asymmetrically defeats the other in AF_{df_2}, and so 'ascending' to a framework AF_{at_3}. If we can exclusively construct an AF_{at_3} argument C for $defend(\mathcal{B}, \mathcal{B}')$ (or $defend(\mathcal{B}', \mathcal{B})$) then we are done. Otherwise we may need to ascend to AF_{at_4}, and so on.

Hence, a hierarchical argumentation framework (HAF) is of the form $(AF_{at_1}, \ldots, AF_{at_n})$, from which we obtain the defeat frameworks $(AF_{df_1}, \ldots, AF_{df_n})$. For $i > 1$, $AF_{at_i} = (Args_i, \mathcal{R}_{at_i})$ is instantiated by a first order logic based argumentation system where $Args_i$ are constructed from a theory Γ_i of wff in a first order language $\mathcal{L}_i$ (note that from hereon we assume the usual axiomatisation of real numbers in any first order theory). Each Γ_i contains a mapping $\mathcal{M}_{i-1} : (Args_{i-1}) \rightarrowtail \wp(\mathcal{L}_i)$. These wff can be used in the inference of valuations of the strength of arguments in $Args_{i-1} = \{A, A', \ldots\}$. These valuations can in turn be used to construct arguments in $Args_i$ with claims of the form $preferred(\mathcal{A}', \mathcal{A})$ and $defend(\mathcal{A}', \mathcal{A})$. The latter requires that each Γ_i ($i > 1$) also axiomatise the notion of individual defense. There exist two such notions in the argumentation literature:

$$preferred(\mathcal{A}', \mathcal{A}) \wedge attack(\mathcal{A}', \mathcal{A}) \rightarrow defend(\mathcal{A}', \mathcal{A}) \qquad \textbf{(N1)}$$

or, A' is simply preferred to A:

$$preferred(\mathcal{A}', \mathcal{A}) \rightarrow defend(\mathcal{A}', \mathcal{A}) \qquad \textbf{(N2)}$$

The choice of axiomatisation only makes a difference in the case of undercut attacks. If $A \rightharpoonup A'$, then assuming **N1**, A asymmetrically defeats A' irrespective of their relative strength (preference), since the latter does not attack the former and so one cannot infer $defend(\mathcal{A}', \mathcal{A})$. In this case we call $A \rightharpoonup A'$ a *preference independent undercut*. An example, is where A' makes a non-provability assumption and A proves (claims) what was assumed unprovable by A', e.g. [5]. Assuming **N2**, A asymmetrically defeats A' only if it is not the case that A' is preferred to A. In this case we call $A \rightharpoonup A'$ a *preference dependent undercut*. Undercuts of this type will be illustrated and discussed in section 3.

Definition 3 Let $AF = (Args, R_{at})$ and let Γ, Γ' be first order theories.

- Let $\Gamma' = \{\textbf{N1}\} \cup \{attack(\mathcal{A}, \mathcal{A}') | (A, A') \in R_{at}\}$. Then Γ axiomatises preference independent undercuts in AF if $\Gamma' \subseteq \Gamma$ and neither predicate attack/2 or defend/2 appear in $\Gamma - \Gamma'$
- Γ axiomatises preference dependent undercuts in AF if $\textbf{N2} \in \Gamma$

We now formally define hierarchical argumentation frameworks and the defeat frameworks obtained from the attack frameworks in a HAF:

Definition 4 A hierarchical argumentation framework is an ordered finite set of argumentation frameworks Δ = ((Args$_1$, $\mathcal{R}_{at_1}$),. . .,(Args$_n$, $\mathcal{R}_{at_n}$)) such that for i > 1 :

• $\mathcal{L}_i$ is a first order language whose signature contains the binary predicate symbols '*preferred*', '*attack*' and '*defend*' and a set of constants $\{\mathcal{A}_1,\ldots,\mathcal{A}_n\}$ naming arguments $Args_{i-1} = \{A_1, \ldots, A_n\}$

• Args$_i$ is the set of consistent arguments constructed from a first order theory Γ_i in the language $\mathcal{L}_i$, where Γ_i axiomatises preference dependent or independent undercuts in $AF_{at_{i-1}}$ and Γ_i contains some set $\mathcal{M}_{i-1}(Args_{i-1})$ s.t. $\mathcal{M}_{i-1} : Args_{i-1} \rightarrowtail \wp(\mathcal{L}_i)$

• $\{(A, A')|A, A' \in Args_i,$ claim(A) = *defend*$(\mathcal{X}, \mathcal{Y})$, claim($A'$) = *defend*$(\mathcal{Y}, \mathcal{X})\}$ $\subseteq \mathcal{R}_{at_i}$.

Definition 5 (AF_{df_1},. . .,AF_{df_n}) is obtained from Δ = (AF_{at_1},. . .,AF_{at_n}) as follows:

1) For i = 1. . .n, Args$_i$ in AF$_{df_i}$ = Args$_i$ in AF$_{at_i}$

2) $\mathcal{R}_{df_n} = \mathcal{R}_{at_n}$

3) For i = 1. . .n-1, $\mathcal{R}_{df_i} = \mathcal{R}_{at_i} - \{$(A,A') | defend($\mathcal{A}', \mathcal{A}$) is the claim of a justified argument of AF$_{df_{i+1}}\}$

We say that $A \in$ justified(Δ) iff $A \in$ justified(AF_{df_1})

3. Formalising and Extending Value Based Argumentation as a Hierarchical Argumentation Framework

In this section we demonstrate the applicability of hierarchical argumentation by formalising and extending value based argumentation as a HAF. In what follows we will make use of the following definitions of first order argument construction [8], and definition of an attack relation given a pre-existing relation of conflict:

Definition 6 An argument A constructed from a first order theory Γ is a pair (Δ, α) such that: i) $\Delta \subseteq \Gamma$; ii) $\Delta \vdash_{FOL} \alpha$; iii) Δ is consistent and set inclusion minimal. We say that Δ is the support and α the claim of A

Let A be an argument with claim α, A' an argument with claim β. Then:

- A rebut attacks A' iff conflict(α, β)
- A undercut attacks A' iff there exists a strict sub-argument A'' of A', such that claim(A'') = γ and conflict(α, γ)

In value based argumentation frameworks (VAF) [1] the success of one argument's attack on another depends on the comparative strength of the *values* advanced by the arguments. To model this, Dung frameworks are extended to define VAFs of the form $\langle Args, \mathcal{R}_{at}, Values, val, P\rangle$ where val is a function from $Args$ to a non-empty set of $Values$, and P is a set $\{a_1, \ldots, a_n\}$, where each a_i names a strict partial ordering (audience) on $Values \times Values$. An audience specific VAF - an AVAF - is a 5-tuple:

$$\vartheta = \langle Args, \mathcal{R}_{at}, Values, val, a\rangle.$$

The justified arguments of an AVAF ϑ are the justified arguments of the framework $(Args, \mathcal{R}^a_{df})$ as defined in definition 2, where $\forall A, A' \in Args$:

$(A,A') \in \mathcal{R}^a_{df}$ iff $(A, A') \in \mathcal{R}_{at}$ and it is not the case that $val(A') > val(A)$ according to a **(V1)**

We now formalise and extend value based argumentation in a HAF. To help motivate our formalisation we refer to application of value based argumentation over proposed actions [7]. This work builds on the account of Walton [9] by proposing a presumptive scheme **AS1** justifying/motivating a course of action:

In the current circumstances R
we should perform action A
to achieve new circumstances S
which will realise some goal G
which will promote some value V

The authors then describe an extensive set of critical questions associated with **AS1**. If $A1$ is an argument instantiating **AS1**, then the critical questions serve to identify arguments that attack $A1$. For example, an argument $A2$ stating that the action in $A1$ has an unsafe side-effect undercut attacks $A1$. $A2$ responds to the critical question - *does the action have a side effect which demotes some value?*. Every argument advances (promotes or demotes) a value. Given an ordering on these values, the arguments can be organised into an AVAF in order to determine the justified arguments. Note that two or more arguments instantiating **AS1** may represent alternative actions for realising the same goal, and hence rebut (symmetrically attack).

In formalising value based argumentation in a HAF, we start with a framework $AF_{at_1} = (Args_1, \mathcal{R}_{at_1})$ where $Args_1$ and $\mathcal{R}_{at_1}$ correspond to the arguments and attack relation in a VAF $\langle Args, \mathcal{R}_{at}, Values, val, P\rangle$. We then define a first order argumentation system instantiating $(Args_2, \mathcal{R}_{at_2})$, where $Args_2$ are constructed as in definition 6 from a first order theory Γ_2. In defining Γ_2 we will make use of the following sets of wff:

Definition 7 Given a set of arguments $Args$ and a strict partial ordering named a on a set of values:

• $Args_{val}$ denotes a set of first order wff used in inferring valuations $val(S, \mathcal{A}, V, X, P)$ of arguments $A \in Args$, where $P = +$ or $-$, V is the value promoted (if $P = +$) or demoted (if $P = -$) by A to degree X (denoting a real number) according to source S.

• $>_a$ denotes the usual first order axiomatisation of a strict partial ordering on values such that $>_a \vdash_{FOL} >(a, V, V')$ iff $V > V'$ according to a

Γ_2 will contain:

1. A set $Args_{1_{val}}$. The need to allow values to be advanced to a given degree is highlighted in [6]. We additionally note that it enables resolution of cycles in the same value. Suppose two mutually attacking arguments for action $A1$ and $A2$ instantiating **AS1** and motivated by the same value V. Then by **V1**, $A1$ and $A2$ defeat each other. However, it may be that $A1$ promotes V to a greater degree than $A2$, and so should defeat $A2$.

2. a **set** $\{>_{a_1}, \ldots, >_{a_n}\}$ of partial orderings on values.

3. $val(S1, \mathcal{A}1, V1, X1, P1) \wedge val(S2, \mathcal{A}2, V2, X2, P2) \wedge >(a_i, V1, V2) \rightarrow preferred(\mathcal{A}1, \mathcal{A}2)$ **(P1)**

4. $val(S1, \mathcal{A}1, V1, X1, P1) \wedge val(S2, \mathcal{A}2, V2, X2, P2) \wedge (\text{V1} = \text{V2}) \wedge (X1 > X2) \rightarrow preferred(\mathcal{A}1, \mathcal{A}2)$ **(P2)**

5. As indicated by **V1** we require that Γ_2 axiomatise *preference dependent* undercuts in AF_{at_1}, i.e., **N2** $\in \Gamma_2$. Note that this means that if $A2$ instantiating a critical question undercuts $A1$ instantiating **AS1** as described above, and $A1$ is preferred to $A2$, then neither defeat each other and both appear in a conflict free subset of arguments in the defeat framework AF_{df_1}. This is acceptable since the arguments do not logically contradict; the action is justified while acknowledging that it has an unsafe side-effect.[1]

Now note that each argument's valuation is parameterised by the source of the valuation. This allows for sources (agents) to ascribe different degrees of promotion/demotion of a value to an argument. Furthermore, we allow for representation of more than one ordering on values. The example concluding this section will demonstrate how these features may result in arguments $B1$ and $B2$ in $Args_2$ with claims of the form defend($\mathcal{A}1, \mathcal{A}2$) and defend($\mathcal{A}2, \mathcal{A}1$). Hence, one may need to argue in a framework AF_{at_3} providing some contextual justification for preferring one ordering to another, or preferring one source to another (in principle one may in turn need to ascend to AF_{at_i}, $i > 3$). Note also that it may be that different motivating values are ascribed to the same action. For example, consider two agents engaged in a deliberative dialogue [10] over a joint donation of a sum of money to a charity. One may consider *altruism* as the motivating value, the other *self interes*t ("it makes me feel good about myself!"). Now if an argument attacking the action promotes the value of *pragmatism* (the joint donation will imply complicated changes to the accounting system), and we have the value ordering *self interest* > *pragmatism* > *altruism*, then evaluating the success of the attack depends on first coming to an agreement as to the motivating value for the action.

Given the preceding discussion and description of Γ_2, we can now define the notion of a value based HAF:

Definition 8 A value based HAF is of the form (($Args_1$, $\mathcal{R}_{at_1}$), ($Args_2$, $\mathcal{R}_{at_2}$), . . ., ($Args_n$, $\mathcal{R}_{at_n}$)), where:

- $Args_2$ are constructed as defined in def.6 from a first order theory $\Gamma_2 \supseteq$ {**N2, P1, P2**} $\cup\ Args1_{val} \cup >_{a_1} \cup \ldots \cup >_{a_n}$
- $\mathcal{R}_{at_2}$ is defined as in def.6, where conflict(α, β) if:
  - $\alpha \equiv \neg\beta$
  - α = *defend*($\mathcal{A}1, \mathcal{A}2$), β = *defend*($\mathcal{A}2, \mathcal{A}1$)
  - $\alpha = val(S, \mathcal{A}, V, X, P)$, $\beta = val(S', \mathcal{A}, V, Y, P)$ and $X \neq Y$
  - $\alpha = val(S, \mathcal{A}, V, X, P)$, $\beta = val(S', \mathcal{A}, V', Y, P')$ and $V \neq V'$

[1] In [3] we argue that if an argument $A1$ undercuts $A2$ where the conflict is based on logical contradiction, then the undercut should be either formalised as preference independent, or reformulated as a rebut, otherwise it may be that logically contradictory arguments coexist in a conflict free subset of a defeat framework.

It is straightforward to show the following (from hereon an underscore '_' denotes some arbitrary variable):

Proposition 1 Let ϑ be the AVAF $\langle Args_1, \mathcal{R}_{at_1}, Values, val, a\rangle$, and Δ = $((Args_1, \mathcal{R}_{at_1}), (Args_2, \mathcal{R}_{at_2}))$ a value based HAF s.t. $Args_2$ are constructed from Γ_2 = {**N2, P1, P2**} $\cup >_a \cup Args_{1_{val}}$, where $Args_{1_{val}}$ = $\{val(\_, \mathcal{A}, V, \_, \_) \mid A \in Args_1, V = val(A)\}$.

Then, $A \in$ justified(Δ) iff $A \in$ justified(ϑ)

We conclude now with a medical treatment example from [11], formalised as an AVAF in [7]. The latter work models decision making over action in a framework of agents based on the Belief-Desire-Intention model. The action scheme **AS1** and an extensive list of associated critical questions are made more computationally precise through representation in terms of propositions, *States* $(R, S, \ldots)$, *Actions* $(A, A' \ldots)$, *Goals*, ternary relations of the form $apply(A, R, S)$, a function mapping goals to Value-Sign pairs, etc. For example, an argument $A1$ for action A instantiating **AS1** requires that the truth value assignment to the propositions in state R holds in the current situation, $(A, R, S) \in apply$, $S \models G$, and $value(G) = \langle V, + \rangle$. An argument $A2$ undercutting $A1$, responding to the critical question *does the action have a side effect which demotes the value it promotes?*, can be made if:

Attack 8 : There is a goal $H \in Goals$, $H \neq G$ s.t. $(A, R, S) \in apply$ with $S \models H$, and $value(H) = \langle V, - \rangle$

In the following example we show a subset of the arguments and their attacks described in [7].

Example 2 The action to be chosen concerns the appropriate treatment for a patient threatened by blood clotting. We show the framework AF_{at_1} below, and descriptions of each argument conforming to schemes and critical questions in the table below:

$$AF_{at_1} = A3 \rightharpoonup A2 \rightharpoonup A1 \rightleftharpoons A4 \leftharpoondown A5$$

$A1$: As platelet adhesion is high, we should administer aspirin, since this results in low platelet adhesion, so that blood clotting is reduced, which will promote the value of safety	$A4$: As platelet adhesion is high, we should administer chlopidogrel, since this results in low platelet adhesion, so that blood clotting is reduced, which will promote the value of safety
$A2$: Since there is a history of gastritis and assuming no proton pump inhibitor is available, we should not administer aspirin, as this would result in dangerous acidity levels, which would risk gastric perforation, which will demote the value of safety	$A5$ As the cost of chlopidogrel is high, we should not administer chlopidogrel, as this will result in large expense, which will exceed the allocated budget per patient, which will demote the value of cost
$A3$: Your assumption that there is no proton pump inhibitor available is false. A proton pump inhibitor is available	

Note that $A2 \rightharpoonup A1$ is an instance of **attack 8** above. $A4 \leftharpoondown A5$ represents a similar attack, but differs in that the value demoted is not the same as the value promoted by the action in $A4$. Finally, $A3 \rightharpoonup A2$ since $A3$ denies that state R is true in the given circumstances (this argument will therefore be regarded as promoting the value of truth)[2].

Now, let $\Delta = (AF_{at_1}, AF_{at_2}, AF_{at_3})$ be a value based HAF defined as in def.8, where AF_{at_1} is the above framework. We describe the argumentation systems instantiating AF_{at_2} and AF_{at_3} (see def.8 for definition of construction of $Args_2$ and R_{at_2}).

AF_{at_2} = $(Args_2, R_{at_2})$:

In what follows, let $ct1$, $ct2$ and *bnf* respectively denote clinical trials 1, 2 and 'british national formulary' (http://www.bnf.org/bnf/). The trials report on the relative efficacy of aspirin and chlopidogrel actions w.r.t reducing blood clotting (and hence these actions promote safety). The formulary reports on hazards (and their levels of seriousness) resulting from administration of treatments when contraindicated. If an argument $A1$ for action promotes safety to degree X (based on a clinical trial report), and an attacking argument $A2$ states that the action has a hazardous side-effect that is an absolute contraindication, then the latter demotes safety to some degree $Y > X$ and thus should defeat $A1$.

Let $Args_2$ be constructed from:

$$\begin{aligned}\Gamma_2 = \{&\textbf{N2}, \textbf{P1}, \textbf{P2}\} \cup \{{>}(a1, truth, safety), {>}(a1, safety, cost)\} \cup \\ &\{val(ct1, \mathcal{A}1, saf, 5, +), val(ct1, \mathcal{A}4, saf, 3, +), val(ct2, \mathcal{A}1, saf, 3, +), \\ &val(ct2, \mathcal{A}4, saf, 5, +), val(bnf, \mathcal{A}2, saf, 7, -), val(\_, \mathcal{A}3, truth, \_, \_), \\ &val(\_, \mathcal{A}5, cost, \_, \_)\}\end{aligned}$$

We obtain the following arguments and attacks:

$claim(B1) = val(ct1, \mathcal{A}1, saf, 5, +)$
$claim(B2) = val(ct2, \mathcal{A}1, saf, 3, +)$
$claim(B3) = val(ct1, \mathcal{A}4, saf, 3, +)$
$claim(B4) = val(ct2, \mathcal{A}4, saf, 5, +)$

$$\begin{array}{c} B1 \rightleftharpoons B2 \\ \downarrow \quad \downarrow \\ B6 \rightleftharpoons B5 \\ \uparrow \quad \uparrow \\ B7 \leftharpoondown B3 \rightleftharpoons B4 \end{array}$$

$claim(B5)$ = $defend(\mathcal{A}1, \mathcal{A}4)$, $support(B5)$ includes $claim(B1)$, $claim(B3)$, **P2**, **N2**

$claim(B6)$ = $defend(\mathcal{A}4, \mathcal{A}1)$, $support(B6)$ includes $claim(B2)$, $claim(B4)$, **P2**, **N2**

$claim(B7)$ = $defend(\mathcal{A}4, \mathcal{A}5)$, $support(B7)$ includes ${>}(a1, safety, cost)$, $claim(B4)$, $val(\_, \mathcal{A}5, cost, \_, \_)$, **P1**, **N2** (notice that we could have also included $B8$ claiming $defend(\mathcal{A}4, \mathcal{A}5)$ based on $claim(B3)$ rather than $claim(B4)$, where $B8$ would be undercut by $B4$)

[2]Notice that a more elegant formulation would not require the assumption in $A2$. Rather, $A3$ would be an argument for the action of giving a proton pump inhibitor, the goal of which would be to deny the relation between action 'aspirin' and effect 'increased acidity'. However, critical questions licensing attacks of this type are not formalised in [7]

Note that given the seriousness of the hazard represented in $A2$, and that safety is not ordered above truth by $a1$, we cannot infer $defend(\mathcal{A}1, \mathcal{A}2)$ and $defend(\mathcal{A}2, \mathcal{A}3)$ respectively.

AF_{at_3} = ($Args_3, R_{at_3}$):

$Args_3$ are constructed (as defined in def.6) from Γ_3 axiomatising ***preference independent undercuts*** in AF_{at_2}, where in addition to **N1** and $\{attack(\mathcal{B}, \mathcal{B}')|(B, B') \in \mathcal{R}_{at_2}\}$, Γ_3 also contains:

- $\mathcal{M}_2(Args_2)$ = $\{source(\mathcal{B}, S, \mathcal{A}, V, X) \mid B \in Args_2, claim(B) = val(S, \mathcal{A}, V, X, P)\} \cup$ $\{ordering(\mathcal{B}, \mathcal{A}, \mathcal{A}', \mathsf{U})|B \in Args_2, claim(B) = defend(\mathcal{A}, \mathcal{A}'),$ **P1** $\in support(B), {>}(\mathsf{U}, V1, V2)$ is a conjunct in the antecedent of **P1**$\}$

- $source(\mathcal{B}, S, \mathcal{A}, V, X) \wedge source(\mathcal{B}', S', \mathcal{A}, V, Y) \wedge (X \neq Y) \wedge pref\_source(S, S')$ $\rightarrow preferred(\mathcal{B}, \mathcal{B}')$

- $source(\mathcal{B}, S, \mathcal{A}, V, X) \wedge source(\mathcal{B}', S', \mathcal{A}, V', Y) \wedge (V \neq V') \wedge pref\_source(S, S')$ $\rightarrow preferred(\mathcal{B}, \mathcal{B}')$

- $ordering(\mathcal{B}, \mathcal{A}, \mathcal{A}', \mathsf{U}) \wedge ordering(\mathcal{B}', \mathcal{A}', \mathcal{A}, \mathsf{U}') \wedge pref\_ordering(\mathsf{U}, \mathsf{U}') \rightarrow$ $preferred(\mathcal{B}, \mathcal{B}')$

- $trial\_design(T, crossover) \wedge trial\_design(T', parallel) \rightarrow pref\_source(T, T')$[3]

- $trial\_design(ct2, crossover), trial\_design(ct1, parallel).$

If α, β are $\mathcal{L}_3$ wff, then conflict(α, β) iff $\alpha \equiv \neg\beta$ or α = $defend(\mathcal{B}, \mathcal{B}')$, β = $defend(\mathcal{B}', \mathcal{B})$ and R_{at_3} is defined as in def.6. From Γ_3 we obtain arguments $C1$ with claim $defend(\mathcal{B}2, \mathcal{B}1)$ and $C2$ with claim $defend(\mathcal{B}4, \mathcal{B}3)$, each of which are based on a source preference for trial $ct2$ over $ct1$.

Applying definition 5 to Δ = ($AF_{at_1}, AF_{at_2}, AF_{at_3}$) obtains the following defeat frameworks with justified arguments shown in bold:

AF_{df_3} $\quad$ AF_{df_2} $\quad$ AF_{df_1}

C1 $\quad$ **C2** $\quad$ $B1 \leftharpoonup$ **B2** $\quad$ **A3** $\rightharpoonup A2 \rightharpoonup A1 \leftharpoonup$ **A4** $\quad$ **A5**

$\downarrow \quad \downarrow$

B6 $\rightleftharpoons B5$

$\uparrow \quad \uparrow$

B7 $\leftharpoonup B3 \leftharpoonup$ **B4**

[3]Crossover trials are usually preferred to parallel designs since only the former expose trial subjects to both drugs being assessed

Administering chlopidogrel is the preferred course of action, since trial 2 is preferred to trial 1; hence the argument for chlopidogrel defeats the argument for aspirin since it promotes safety to a greater degree than aspirin. Since safety is ordered higher than cost, then the preference dependent undercut from $A5$ to $A4$ is removed in the obtained defeat framework AF_{df_1}. Both $A5$ and $A4$ are justified. Notice that if in addition to $a1$, Γ_2 contained another value ordering $a2$ that ordered cost above safety, then one would be able to construct an additional AF_{at_2} argument $B8$ with claim $defend(\mathcal{A}5, \mathcal{A}4)$ that rebuts $B7$. Hence, one then needs to resolve in AF_{at_3}, possibly constructing a contextual argument $C3$ with claim $pref\_ordering(a2, a1)$ based on the fact that resources are low (the harsh reality is that such a trade of between safety and cost is made in medical contexts when financial resources are low) and so $C4$ with claim $defend(\mathcal{B}8, \mathcal{B}7)$. This in turn would result in the following **A3** $\rightharpoonup A2 \rightharpoonup$ **A1** $\leftharpoondown A4 \leftharpoondown$ **A5**, i.e., administering aspirin is now the preferred course of action.

4. Conclusions

In this paper we have formalised value based argumentation in a hierarchical argumentation framework. We have extended the notion of an argument promoting/demoting a value to allow for the degree of promotion/demotion. In this way, conflicts between mutually attacking arguments promoting the same value, but to differing degrees, can be resolved. We have also motivated and allowed for representation of more than one ordering of values, and parameterised the valuations of arguments by the information sources for these valuations. This may result in conflicting preferences between arguments that are resolvable through hierarchical argumentation over preference information. We illustrated our approach with an example from [7] in which agents deliberate over an appropriate course of action. This substantiates our claim that hierarchical argumentation can address challenges raised by applications of argumentation theory in agent and multi-agent contexts [12,13,10] in which interacting arguments over different epistemological categories will require different notions of conflict and conflict based interaction, and different principles by which the relative strengths of arguments are evaluated, all within a single system. For example, argumentation-based dialogues require that agents justify their preference for one argument over another, and have this justification itself challenged (e.g., [10]).

Reasoning about preferences is also explored in [14,15,5], in which the object level language for argument construction is extended with rules that allow context dependent inference of possibly conflicting relative prioritisations of *rules*. However, these works exclusively base argument strength on the priorities of their constituent sentences (rules). Furthermore, a clean formal separation of meta and object level reasoning is necessary if one is to reason about strengths of arguments as opposed to their constituent sentences (e.g., consider argument strength based on the depth/length of the proof that constitutes the argument, or the value promoted by the argument). Finally, one of our basic aims has been to put the general idea of meta-argumentation on the map. We share this aim with [16] in which the focus is on reasoning about the construction of arguments rather than preference information.

References

[1] T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[2] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reason, logic programming, and N-person games. *Artificial Intelligence*, 77:321–357, 1995.

[3] S. Modgil. Hierarchical argumentation. In *10th European Conf. on Logics in Artificial Intelligence (JELIA) (to appear)*, 2006. available at http://www.acl.icnet.uk/lab/PUBLICATIONS/ms422.pdf.

[4] L. Amgoud and C. Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *International Journal of Automated Reasoning*, Volume 29 (2):125–169, 2002.

[5] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-Classical Logics*, 7:25–75, 1997.

[6] K. Atkinson, T. J. M. Bench-Capon, and P. McBurney. Persuasive political argument. In C. Reed F. Grasso and R. Kibble, editors, *Proc. First International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2004)*, pages 44–51, Edinburgh, Scotland, 2004.

[7] K. Atkinson. *What Should We Do?:Computational Representation of Persuasive Argument in Practical Reasoning. PhD Thesis*. PhD thesis, Department of Computer Science, University of Liverpool, Liverpool, UK, 2005.

[8] P. Besnard and A. Hunter. Practical first-order argumentation. In *AAAI*, pages 590–595, 2005.

[9] D. N. Walton. *Argument Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, USA, 1996.

[10] D. Hitchcock, P. McBurney, and S. Parsons. A framework for deliberation dialogues. In H. V. Hansen et.al, editor, *Proc. Fourth Biennial Conference of the Ontario Society for the Study of Argumentation (OSSA 2001)*, Canada, 2001.

[11] S. Modgil and J. Fox. A guardian agent approach to safety in medical multi-agent systems. In *Proc. of the First International Workshop on Safety and Security in Multi-Agent Systems (SAEMAS'04 at AAMAS 2004)*, New York, 2004.

[12] L. Amgoud and S. Kaci. On generation of bipolar goals in argumentation-based negotiation. In I. Rahwan, P. Moraitis, and C. Reed, editors, *Proc. 1st Int. Workshop on Argumentation in Multi-Agent Systems*, New York, New York, 2004. Springer.

[13] S. Modgil. Nested argumentation and its application to decision making over actions. In *Proc. Second Int. Workshop on Argumentation in Multi-Agent Systems*, Netherlands, 2005.

[14] G. Brewka. Well-founded semantics for extended logic programs with dynamic preferences. *Jnl. of Artificial Intelligence Res.*, 4, 19–36, 1996

[15] A. Kakas and P. Moraitis. Argumentation based decision making for autonomous agents. In *Proc. Second international joint conference on Autonomous agents and multiagent systems*, pages 883–890. ACM Press, 2003.

[16] M. Wooldridge, P. McBurney, and S. Parsons. On the meta-logic of arguments. In *AAMAS '05: Proc. Fourth international joint conference on Autonomous agents and multiagent systems*, pages 560–567, NY, USA, 2005. ACM Press.

Dialogue and Decision

This page intentionally left blank

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Combining sceptical epistemic reasoning with credulous practical reasoning

Henry Prakken
Department of Information and Computing Sciences, Faculty of Science, Universiteit Utrecht & Faculty of Law, University of Groningen, The Netherlands

Abstract. This paper proposes an argument-based semantics for combined epistemic and practical reasoning, taking seriously the idea that in certain contexts epistemic reasoning is sceptical while practical reasoning is credulous. The new semantics combines grounded and preferred semantics. A dialectical proof theory is defined which is sound and complete with respect to this semantics and which combines existing argument games for sceptical reasoning in grounded semantics and credulous reasoning with preferred semantics.

Keywords. Practical vs. epistemic reasoning, argumentation, credulous and sceptical reasoning

1. Introduction

This paper is about the relation between epistemic and practical reasoning, or the relation between reasoning about beliefs and reasoning about action, where the later is understood as reasoning with motivational attitudes, such as having goals, desires or intentions. Since goals and desires often conflict or can be fulfilled in alternative ways, several researchers have proposed to formalise practical reasoning within a nonmonotonic logic. Some have used default logic [1,2] while others have proposed argument-based accounts. Fox & Parsons [3] study the combination of medical diagnostic and treatment arguments in the argumentation logic of [4]. Pollock [5] combines epistemic and practical reasoning in his OSCAR system. Atkinson and her colleagues [6,7] give an account based on argument schemes [8]. Finally, Amgoud [9] proposes a combined model of inference and decision making in a logic with tree-style defeasible arguments.

As is well-known, in nonmonotonic logics two different kinds of inference relations can be defined, viz. for *credulous* and for *sceptical* reasoning. They differ only when a conflict between defaults or arguments cannot be resolved. In such a case, credulous consequence notions branch into alternative sets of defeasible conclusions while sceptical consequence notions stick to one such set and leave both conclusions involved in an unresolvable conflict out of this set.

The current work on combining defeasible epistemic and practical reasoning essentially applies the same defeasible inference relation to both kinds of reasoning: either all reasoning is credulous, as in e.g. [1,2], or all reasoning is sceptical, as in e.g. [9]. (Amgoud allows for a credulous choice between alternative ways to achieve a goal, but this

choice is formalised as a separate decision-making phase after all sceptically acceptable options have been computed).

However, in this paper I want to make a case for the claim that in certain contexts reasoning about beliefs should be sceptical while reasoning about action should be credulous. Consider a university lecturer John who wants to finish a paper on Friday but who has also promised to give a talk in a remote small town Faraway on the same day. John sees only two ways to travel to Faraway: by car and by bus; in both cases he will not be able to work while travelling (in the bus he always gets sick when reading or writing). So he sees no way to fulfil one desire without giving up the other and he sees no intrinsic reason to prefer one desire over the other. Then it seems rational for John to make a choice which desire he wants to fulfil. If this choice is formalised as reasoning, it must be formalised as credulous reasoning. However, let us now suppose that John's friend Bob tells him that there is a railway connection to Faraway, so that he could work while travelling and also finish his paper. Then John's other friend Mary warns him that there will be a railway strike on Friday, so that there will be no trains after all. Bob, however, says he does not believe there will be such a strike. So to form the goal of taking the train, John must first find out whether it will run on Friday. If he has to find this out on the basis of his current beliefs, his task is one of epistemic reasoning. Now suppose that John has no reason to trust one of his friends more than the other. Then it seems rational for him not to act on the credulous belief that there will be a train to Faraway on Friday.

The kind of rationality that is assumed here is that a rational agent should map out all credulously acceptable action alternatives that have sceptically acceptable epistemic support and then make a choice between them. Objections might be raised against this view of rationality; some of them will be discussed in Section 6, after the present view has been formalised.

The technical contribution of this paper is a combined formalisation for sceptical epistemic reasoning interleaved with credulous practical reasoning. More precisely, a unified formal framework will be defined for sceptical epistemic reasoning according to grounded semantics and credulous practical reasoning according to preferred semantics. The choice for sceptical grounded and credulous preferred semantics [10] has a pragmatic and a philosophical reason. The pragmatic reasoning is that sceptical grounded and credulous preferred semantics are presently the only two argument-based semantics with elegant proof-procedures in argument-game form. See e.g. [11,12] for sceptical grounded and [13,14,15,16] for credulous preferred semantics. A philosophical reason is given by Caminada [17] in his defence why sceptical reasoning should be modelled with grounded semantics and credulous reasoning with preferred semantics. Since his argument is too detailed to repeat it here, the interested reader is referred to his paper.

The rest of this paper is organised as follows. In Section 2 the framework for argument-based reasoning assumed in this paper will be introduced. Among other things, this section introduces the notions of a logical language, argument construction and defeat between arguments. These three notions will in Section 3 be refined by dividing the language into an epistemic and a practical sublanguage and by distinguishing epistemic vs. practical arguments and epistemic vs. practical ways to defeat an argument. Section 2 has also summarised grounded and preferred semantics and two argument games for these semantics. The semantics and argument games will in Section 4 be merged into a unified semantics and proof theory for combined epistemic and practical reasoning. The

new formalism will be illustrated with some examples in Section 5 and some objections against its underlying account of rationality will be discussed in Section 6.

2. Logical preliminaries

The analysis of this paper is within Dung's [10] abstract approach to defeasible argumentation. First the basic notions of [10] will be summarised as far as needed here, adapting some notation of [14].

Definition 2.1 An *argument system* is a pair $\mathcal{H} = (\mathcal{A}, \mathcal{D})$, in which $\mathcal{A}$ is a set of *arguments* and $\mathcal{D} \subseteq \mathcal{A} \times \mathcal{A}$ is the *defeat* relationship for $\mathcal{H}$. When $(a, b) \in \mathcal{D}$ we say that a *defeats* b; when moreover $a \in S \subseteq \mathcal{A}$ we also say that S defeats b. For $S \subseteq \mathcal{A}$ we say that

1. $a \in \mathcal{A}$ is *acceptable with respect to* S if for every $b \in \mathcal{A}$ that defeats a there is some $c \in S$ that defeats b.
2. S is *conflict-free* if no argument in S is defeated by S.
3. S is *admissible* if S is conflict-free and every argument in S is acceptable with respect to S.
4. S is a *preferred extension* of $\mathcal{H}$ if it is a maximal (with respect to set inclusion) admissible subset of $\mathcal{A}$.
5. Let $F : \mathcal{A}^2 \longrightarrow \mathcal{A}^2$ be a function that for each subset S of $\mathcal{A}$ returns the set of all arguments that are acceptable with respect to S. Then S is the *grounded extension* of $\mathcal{H}$ if S is the least fixpoint of F.

As shown by [10], grounded semantics always produces a unique extension while preferred semantics may produce multiple extensions when a conflict between arguments cannot be resolved. This motivates the following well-known definitions. An argument is *justified with respect to* $\mathcal{H}$ in a semantics if it is in every extension of $\mathcal{H}$ according to that semantics. An argument is *defensible with respect to* $\mathcal{H}$ in preferred semantics if it is in some but not all preferred extensions of $\mathcal{H}$. Finally, an argument is *defensible with respect to* $\mathcal{H}$ in grounded semantics if it is not in the grounded extension of $\mathcal{H}$ but not defeated by it. Now *credulous* argument-based reasoning is interested whether an argument is defensible while *sceptical* argument-based reasoning checks whether an argument is justified.

This abstract approach will be instantiated with a familiar tree-style structure of arguments [18,19,20], where *strict* and *defeasible* inferences are chained into trees. The inference rules apply to a logical language $\mathcal{L}$ closed under negation. Strict inference rules, which are usually taken to be those of standard propositional or first-order logic, are written as $\varphi_1, \ldots, \varphi_n \rightarrow \varphi$ and defeasible rules as $\varphi_1, \ldots, \varphi_n \rightsquigarrow \varphi$ (where each φ and φ_i is a well-formed formula of $\mathcal{L}$).

Arguments chain inference rules into AND trees, starting with a subset of $\mathcal{L}$. For any argument A, its *premises*, written as $prem(A)$, are all leaf nodes of A, and its *conclusion*, $conc(A)$, is its root. An argument A is a *subargument* of an argument B if both have the same premises and $conc(A)$ is a node in B. An argument is *strict* if all its rules are strict, otherwise it is *defeasible*. It is defeasible inferences that make an argument subject to defeat.

The defeat relation will not be fully formalised in this paper; any full definition is assumed to satisfy the following conditions (which are satisfied by most argumentation systems in the literature).

Assumption 2.2 For all arguments A, B and C: if A defeats B and B is a subargument of C then A defeats C.

Observation 2.3 If an argumentation system satisfies Assumption 2.2 then for all arguments A and B and grounded or preferred extensions E:

- if A is a subargument of B and $A \notin E$ then $B \notin E$.

Finally, we define the notion of an argument game and provide two instantiations for grounded and preferred semantics. The following definitions use some notation of [21] and are relative to an unspecified argumentation system.

Argument games are between a *proponent* P and an *opponent* O of an argument from $\mathcal{A}$. The set M of *moves* of a game is defined as $\mathbb{N} \times \{P, O\} \times \mathcal{A} \times \mathbb{N}$, where the four elements of a move m_i are denoted by, respectively: $id(m)$, the *identifier* of the move, $pl(m)$, the *player* of the move, $a(m)$, the *argument* put forward in the move, $t(m)$, the *target* of the move.

The set of *argument games* (or games for short), denoted by M, is the set of all sequences $d = m_1, \ldots, m_i, \ldots$ from M such that

- each i^{th} element in the sequence has identifier i,
- $t(m_1) = 0$,
- $pl(m_i) = P$ if i is odd and $pl(m_i) = O$ if i is even;
- for all $i > 1$ it holds that $t(m_i) = j$ for some m_j preceding m_i in the sequence,
- if $d \neq \emptyset$ then $a(m_i)$ defeats $a(t(m_i))$.

The set of *finite games*, denoted by M^*, is the set of all finite sequences that satisfy these conditions. For any game $d = m_1, \ldots, m_n, \ldots$, the sequence $m_1, \ldots, m_i$ is denoted by d_i, where d_0 denotes the empty game. When $t(m) = id(m')$ we say that m *replies to* m' in d and also that m' is the *target of* m in d. Slightly abusing notation, $t(m)$ sometimes denotes a move instead of just its identifier.

A *protocol* on M is a set $R \subseteq M^*$ satisfying the condition that whenever d is in R, so are all initial sequences that d starts with. A partial function $Pr : M^* \longrightarrow \mathcal{P}(M)$ is derived from R as follows:

- $Pr(d) =$ undefined whenever $d \notin R$;
- $Pr(d) = \{m \mid d, m \in R\}$ otherwise.

The elements of $dom(Pr)$ (the domain of Pr) are called the *legal finite games*. The elements of $Pr(d)$ are the moves allowed after d. If d is a legal game and $Pr(d) = \emptyset$, then d is said to be a *terminated* game.

Within this framework an argument game for grounded semantics proposed by [12] can be stated as follows:

Definition 2.4 [G-games] An argument game is a *G-game* if for all moves m_i and finite legal games d it holds that $m_i \in Pr(d)$ iff:

1. If $i \neq 0$ then $t(m_i) = m_{i-1}$;

2. if $pl(m_i) = P$ then $a(m_i)$ was not moved by P in d.

Player pl *wins* a G-game d iff d is terminated and pl made the last move in d.

This game was by [12] proven to be sound and complete with respect to grounded semantics in the sense that proponent has a winning strategy for argument A if and only if A is a member of the grounded extension.[1]

In [13] the following game for preferred semantics was defined. In this definition, *a game line* is a sequence of the game where each non-initial move responds to the preceding move.

Definition 2.5 [P-games] An argument game is a *P-game* if for all moves m_i and finite legal games d it holds that $m_i \in Pr(d)$ iff:

1. if $pl(m_i) = P$ then:
   (a) If $i \neq 0$ then $t(m_i) = m_{i-1}$;
   (b) $a(m_i)$ was not moved by O in d.
2. if $pl(m_i) = O$ then $a(m_i)$ was not moved by O in d in the same game line.

Player pl *wins* a P-game d if d is terminated and pl made the last move in d. Furthermore, P wins d if d is infinite and O wins d if O repeats any argument earlier moved by P.

This game was by [13] proven to be sound and complete with respect to credulous preferred semantics in the sense that proponent has a winning strategy for argument A if and only if A is a member of some preferred extension.

3. Knowledge, arguments and defeat in epistemic and practical reasoning

In this section we adapt the notions of a logical language, argument construction and defeat between arguments to the distinction between epistemic and practical reasoning. Generally, two styles of reasoning about action can be recognised, which could be called *quasi-deductive* and *abductive*. The quasi-deductive approach, taken by e.g. [1,2], formalises decision rules as "If I want to satisfy my appetite I must prepare a meal" and applies such rules to beliefs or desires in a forward, modus-ponens style of reasoning. Conflicting desires, as with "if someone opens the door for me I should thank him" vs. "I should not speak while eating", are then managed by a priority mechanism defined on the rules. The abductive approach, taken by e.g. [6], essentially applies the well-known practical syllogism from philosophy: if I believe that "preparing a meal can make me satisfy my appetite" and I want to satisfy my appetite, I can form the desire to prepare a meal with a 'backward', abductive reasoning step. Alternative ways to fulfil my desires, such as given by the knowledge that "going to a restaurant can satisfy my appetite", must then be regarded as conflicting arguments.

The formalism proposed in this paper is intended to apply to both forms of reasoning and therefore it must abstract from their particularities. To this end, this paper will confine

[1] In fact [12] required that the target of a P move does not defeat it but this condition is redundant for this result.

itself to some partial assumptions on the language, the rules for argument construction and the nature of the defeat relation.

First it is assumed that the logical language is divided into two disjoint sublanguages $\mathcal{L}_e$ of *epistemic formulas* and $\mathcal{L}_p$ of *practical formulas*. A full definition of these two sublanguages is far from trivial but depends on the chosen particular argumentation system and cannot therefore be addressed in this paper.

Next a further condition is imposed on arguments. An instantiated inference rule r is called *epistemic* if $prem(r) \cup conc(r) \subseteq \mathcal{L}_e$ and it is called *practical* if $conc(r) \in \mathcal{L}_p$. Then all inferences in an argument must be either epistemic or practical. An argument is called an *e-argument* if all its inferences are epistemic, otherwise it is called a *p-argument*. For any argumentation system $\mathcal{H} = (\mathcal{A}, \mathcal{D})$, the set $\mathcal{A}$ is accordingly divided into two subsets $\mathcal{A}_e$ and $\mathcal{A}_p$; note that these sets are disjoint. Note also that these constraints rule out that inferences with an epistemic premise have a practical formula as a premise. This cannot be expressed as a condition on inference rules if the strict rules contain, for instance, full propositional or first-order logic, since if an epistemic formula φ is universally valid then any practical formula strictly implies it. The constraints in effect formalise the principle that no *Is* can be derived from a practical *Ought*. The reverse principle could also be formalised but for present purposes this will not be needed.

At first sight, the following example (due to Trevor Bench-Capon, personal communication) would seem to cast doubt on the constraint that no *Is* can be inferred from an *Ought*. Consider the default rule "John tends to achieve his goals" and assume that John has selected his goal to finish his paper to be carried out. Then it seems that these two premises, of which one is practical, give rise to an argument for the conclusion that John will in fact finish his paper. However, it should be noted that it is not John's goal that is a premise of this argument but the observer's belief that John has adopted it, and the latter is an epistemic formula. This is reminiscent of the distinction in deontic logic between logics of norms (expressing obligations and permissions) and logics of normative propositions (describing the content of normative system from an external perspective); see [22] for a clear account of this distinction.

An argumentation system is now called an *e-p-argumentation system* if its set of arguments consists of disjoint sets of e- and p-arguments. Moreover, for any two arguments A and B such that A defeats B we say that A *e-defeats* B if A is an epistemic argument and that A *p-defeats* B otherwise. We assume that if A defeats B such that A does not defeat a proper subargument of B, then B is an e-argument if A is an e-argument and B is a p-argument otherwise. Note, by the way, that if A e-defeats B then B may very well be a practical argument; in that case A defeats an epistemic subargument of B.

Observation 3.1 If an argumentation system satisfies Assumption 2.2 then for all arguments A and B and grounded or preferred extensions E:

1. if A e-defeats a subargument of B, then A e-defeats B;
2. if A e-defeats B, then there exists an epistemic subargument B' of B such that A e-defeats B';
3. no p-argument defeats an e-argument.

4. Combining epistemic and practical inference

In this section the new inference notion for combined epistemic and practical reasoning will be defined. At first sight, it would seem to suffice to first determine the grounded extension of all belief arguments, then add all justified beliefs as facts to the knowledge base and then construct the preferred extensions of the new theory. However, this does not work, since it does not respect that reasoning about beliefs and actions is interleaved (cf. also [5]). Often it is practical reasoning that determines which beliefs are relevant. For instance, in the example of the introduction John's beliefs about whether the train will run to Faraway on Friday is relevant only if he considers his goal of giving the talk. It does not make much sense for John to reason about this irrespective of his goals and desires. What is needed therefore is a single proof procedure for both kinds of reasoning. And this proof procedure in turn needs a semantics, which has the following form.

Definition 4.1 Let $\mathcal{H} = (\mathcal{A}, \mathcal{D})$ be an e-p-argumentation system with grounded extension G. Let $\mathcal{H}_g = (\mathcal{A}_g, \mathcal{D}_g)$ be obtained from $\mathcal{H}$ by:

- removing from $\mathcal{A}$ all e-arguments that are not in G plus all arguments of which they are a subargument,
- and restricting $\mathcal{D}$ to $\mathcal{A}_g$.

Then S is an *e-p-extension* of $\mathcal{H}$ iff S is a preferred extension of $\mathcal{H}_g$.

Observation 4.2 For any e-p-argumentation system $\mathcal{H} = (\mathcal{A}, \mathcal{D})$ with grounded extension G, no argument in $\mathcal{A}_g$ is defeated by G.

The corresponding argument game is now defined by combining the G-game with the P-game as follows.

Definition 4.3 [GP-games] An argument game is a *GP-game* if for all moves m_i and finite legal games d with last move m_l it holds that $m_i \in Pr(d)$ iff:

1. if $pl(m_i) = P$ or $a_l \in \mathcal{A}_e$ then $t(m_i) = m_l$;
2. if $pl(m_l) = O$ and $a(m_l)$ repeats an argument earlier moved by P then $a(m_i) \in \mathcal{A}_e$;
3. if $pl(m_i) = P$ then:
   (a) If $a(m_i) \in \mathcal{A}_e$ then $a(m_i)$ was not moved by P in d;
   (b) If $a(m_i) \in \mathcal{A}_p$ then $a(m_i)$ was not moved by O in d;
4. if $pl(m_i) = O$ then $a(m_i)$ was not moved by O in d in the same game line.

Player pl *wins* a GP-game d if d is terminated and pl made the last move in d. Furthermore, if d is infinite then P wins d if d contains no e-arguments while O wins d otherwise.

In this definition, clause (1) captures that P may not backtrack in the P game and neither player may backtrack in the G-game. Clause (2) captures that when O repeats a p-argument of P, then O in fact wins the P-part of the GP-game so the game continues as a G-game. Clause (3a) captures that P may not repeat his own e-arguments in the G-part of the game while clause (3b) expresses that P may not repeat O's p-arguments in the

P-part of the game. Finally, clause (4) repeats the rule of the P-game that O may only repeat her own moves in different lines of the dialogue.

The GP-game will now be proven sound an complete with respect to the semantics of Definition 4.1 in the sense that proponent has a winning strategy for argument A if and only if A is a member of some e-p-extension.

Lemma 4.4 let d be a GP-game.

1. d consists of a possibly empty sequence of p-arguments (denoted by d_p) followed by a possibly empty sequence of e-arguments (denoted by d_e);
2. d_p is a P-game;
3. the maximal subsequence in d_e starting with a move by P is a G-game.

Theorem 4.5 For any e-p-argumentation $\mathcal{H} = (\mathcal{A}, \mathcal{D})$ that satisfies Assumption 2.2 and any argument $A \in \mathcal{A}$, proponent has a winning strategy for A in a GP-game if and only if there exists an e-p-extension of $\mathcal{H}$ that includes A.

Proof: ($\Rightarrow$) For soundness, suppose P has a winning strategy S for A. Then P can win any game for A. Consider any such game d in which O has played optimally.

Consider first any e-argument A moved by P in d. Note that A is in d_e. Since O has played optimally and P has won d, it holds that P also wins d_e according to the G-game. Then by soundness of the G-game all of P's arguments in d_e are in the grounded extension G. Then, since all of O's e-arguments are in d_e, none of them are in G.

Consider next any p-argument A moved by P in d. As just proven, no e-defeater of A moved by O is in G. Note next that any e-defeater of A moved by P is in d_e and so can be used by O to defeat A and win d. Since O has played optimally, she has moved some such argument if it exists, but this contradicts the assumption that P has won d. Hence $A \in G$ so all p-arguments A moved by P in d are in G.

Now suppose that the set S of all P arguments in d is not admissible on the basis of $\mathcal{H}_g$ and suppose first that S is not conflict-free. Then there exists A and B in S such that A defeats B, so O can at some point in d repeat A after P has moved A. Then since P has to reply to A with an e-argument but all his e-arguments are in G, by Observation 4.2 O can win d by moving A. Since O has played optimally, O has moved A in d. But this contradicts the assumption that P has won d. Suppose next that there exists a non-defeated $A \in \mathcal{A}_g$ defeating some $B \in S$. Then O can win d by moving A which again contradicts the assumption that P has won d. So S is admissible with respect to $\mathcal{H}_g$. Then by a result of [10] S is included in a preferred extension of $\mathcal{H}_g$.

($\Leftarrow$) For completeness, suppose A is in some e-p-extension of $\mathcal{H}_g$. By completeness of the P-game P has a winning strategy S for A in the P-game. Then as long as O moves p-arguments, P can reply by picking p-arguments from S. Consider next any e-argument $B \in \mathcal{A}$ defeating A. By Observation 4.2 then $B \notin \mathcal{A}_g$. If there does not exist a $C \in G$ that e-defeats B, then some epistemic subargument of A defeated by B (which exists by clause (2) of Observation 3.1) is not in G; but then $A \notin \mathcal{A}_g$ by construction of $\mathcal{A}_g$, which contradicts the assumption that $A \in \mathcal{A}_g$. So there exists a $C \in G$ that e-defeats B. Then by completeness of the G-game and clause (1) of Definition 4.3, it holds that P can win by following a winning strategy for C in the G-game. □

5. Examples

In this section the new formalism will be illustrated with some symbolic examples. Since the formalism abstracts from the full details of particular logics, the examples will have to be semiformal. Consider first quasi-deductive, forward goal generation. Assume the following rules (where D stands for 'desire'):

$$\{\ p{\Rightarrow}\mathrm{D}q,\ r{\Rightarrow}\neg\mathrm{D}q,\ s{\Rightarrow}p,\ t{\Rightarrow}\neg p\ \}$$

And let s and r be given as facts. Now let all formulas $(\neg)\mathrm{D}L$ where L is a propositional literal and all rules with occurrences of D be practical formulas and let the rest be epistemic formulas. Suppose an argument A defeats an argument B if A's conclusion contradicts a conclusion of a subargument of B. The following GP-games can be played:

P_1: s, $s{\Rightarrow}p$, $p{\Rightarrow}\mathrm{D}q$, so $\mathrm{D}q$ $\quad$ O_1: r, $r{\Rightarrow}\neg\mathrm{D}q$, so $\neg\mathrm{D}q$
P_2: repeats P_1, now in reply to O_1.

Opponent lost this game with only p-arguments but could have won with a g-argument:

P_1: s, $s{\Rightarrow}p$, $p{\Rightarrow}\mathrm{D}q$, so $\mathrm{D}q$ $\quad$ O'_1: t, $t{\Rightarrow}\neg p$, so $\neg p$
P'_2: s, $s{\Rightarrow}p$, so p $\quad$ O'_2: repeats O'_1

So P has no winning strategy for $\mathrm{D}q$. However, he has one for $\neg\mathrm{D}q$:

P'_1: r, $r{\Rightarrow}\neg\mathrm{D}q$, so $\neg\mathrm{D}q$ $\quad$ O''_1: s, $s{\Rightarrow}p$, $p{\Rightarrow}\mathrm{D}q$, so $\mathrm{D}q$
P''_2: repeats P_1

So the only action alternative with justified support is for $\neg\mathrm{D}q$. This agrees with the semantics: the e-arguments P'_2 and O'_1 defeat each other so they are not in the grounded extension of $\mathcal{H}$. Then they are not in $\mathcal{A}_g$ so the p-argument P_1, which has P'_2 as a subargument, is also not in $\mathcal{A}_g$. So $\mathcal{H}_g$ has a unique e-p-extension, containing $O_1 = P'_1$.

Suppose now that p becomes known as a matter of fact. Then P also has a winning strategy for $\mathrm{D}q$ since O cannot now win as in the second game above:

P_1: s, $s{\Rightarrow}p$, $p{\Rightarrow}\mathrm{D}q$, so $\mathrm{D}q$ $\quad$ O'''_1: t, $t{\Rightarrow}\neg p$, so $\neg p$
P'''_2: p

Assuming that purely factual arguments cannot be defeated, P wins this game. Nevertheless, still a choice must be made what to do since the trivial winning strategy for $\neg\mathrm{D}q$ still stands. Again this agrees with the semantics: P_1's subargument P'_2 is now in the grounded extension of $\mathcal{H}$ so P_1 is in $\mathcal{A}_g$. Since P_1 defeats its only defeater in $\mathcal{A}_g$, which is O_1, there are now two e-p-extensions, one containing P_1 and the other containing O_1.

Consider next a symbolic example with abductive goal generation, in which from "doing a in circumstance s achieves g" and $\mathrm{D}g$ the desire $\mathrm{D}a$ can be inferred, and from the same rule and $\mathrm{D}g$ instead $\neg\mathrm{D}a$ can be inferred. Consider

$$\{\ a_1 \wedge s \Rightarrow p,\ a_2{\Rightarrow}q,\ r{\Rightarrow}s,\ a_1{\Rightarrow}\neg q,\ a_2{\Rightarrow}\neg p\ \}$$

and suppose we have the desires $\mathrm{D}p$ and $\mathrm{D}q$. Let a_i be action descriptions while s is a circumstance beyond the agent's control. As before, formulas with D are in $\mathcal{L}_p$ while the rest is in $\mathcal{L}_e$. Defeat can now also happen by providing an alternative way to fulfil a desire. Now P has a winning strategy for an argument for $\mathrm{D}p$:

P_1: r, $r \Rightarrow s$, so s; also $a_1 \wedge s \Rightarrow p$ and $\mathrm{D}p$, so $\mathrm{D}a_1$
O_1: $a_2 \Rightarrow p$ and $\mathrm{D}p$, so $\mathrm{D}a_2$
P_2: $a_2 \Rightarrow \neg q$ and $\mathrm{D}q$, so $\neg \mathrm{D}a_2$ ("a_2 prevents another desire, so don't do it")
O_2: $a_1 \Rightarrow \neg p$ and $\mathrm{D}p$, so $\neg \mathrm{D}a_1$ (alternative reply to P_1)
P_3: repeats P_1 in reply to O_2.

In a similar way there is a winning strategy for $\mathrm{D}a_2$ so there are two e-p-extensions, one with arguments for $\mathrm{D}a_1$ and $\neg \mathrm{D}a_2$ and one with arguments for $\mathrm{D}a_2$ and $\neg \mathrm{D}a_1$. Note also that $\mathcal{H}_g = \mathcal{H}$ so these are also preferred extensions of $\mathcal{H}$.

Assume now that we also know that $t \Rightarrow \neg s$ while t holds as a matter of fact. Then in the last argument game O can win by attacking P_1 with an e-argument for $\neg s$, defeating its subargument for s (but also defeated by it). So the arguments for s and $\neg s$ are both not in $\mathcal{A}_g$, so P_1 is also not in $\mathcal{A}_g$ so there is a unique e-p-extension, containing arguments for $\mathrm{D}a_2$ and $\neg \mathrm{D}a_1$.

6. Discussion

This paper has proposed an argument-based semantics and proof theory for combined epistemic and practical reasoning, taking seriously the idea that in certain contexts epistemic reasoning is sceptical while practical reasoning is credulous. As mentioned in the introduction, the kind of rationality assumed by the new formalism is that a rational agent should map out all defensible action alternatives that have justified epistemic support and then make a choice between them. Several objections might be raised against this view of rationality. For instance, it might be argued that in cases where an agent has only defensible support for his action alternatives, the present approach implies that he cannot rationally do anything. (This objection is due to Trevor Bench-Capon, personal communication.) Two alternative solutions to this problem could be proposed.

For the first, consider again the example of the introduction. It could be argued that what John should do is to compute the expected utility of his actions, incorporating the uncertainty about whether there will be a train to Faraway on Friday into a probability distribution over the possible outcomes of taking the train. This may be a sound approach if all uncertainty and utility can be quantified but it hides the fact that John will often have to *reason* about whether there will be a train (see also [23]). Moreover, often only partial and qualitative information about probability and preference is available and nonmonotonic logics are primarily meant for use in such cases.

A second option is to say that a logic for practical reasoning should simply present all credulously acceptable arguments to a user of the system and let the user decide upon which argument to act. This may be the best approach in some contexts but it also seems to blur a fundamental difference between epistemic and practical conflicts. In the first case, truth is at stake, so it is rational to do further investigations to resolve the conflict (John could phone the railway company). However, with conflicts on action there is no counterpart of truth and an arbitrary choice between credulously acceptable alternatives is perfectly rational.

Having said this, the choice between the best way to combine epistemic and practical reasoning may depend on the context of application. If precise knowledge about probabilities and utilities is available, decision-theoretic approaches may be the most suitable. Otherwise, a qualitative approach seems preferred, where in some contexts all reasoning

can best be credulous while in other contexts epistemic reasoning can better be sceptical (it makes less sense to make all reasoning sceptical, since practical reasoning inevitably involves choice and a logic for such reasoning should reveal the alternatives). A merit of this paper is that it has provided a formalism for contexts in which the latter approach seems best. One such context is legal reasoning, where the facts of a case have to be proven but may lead to alternative normative consequences.

Acknowledgements

This work was partially supported by by the EU under IST-FP6-002307 (ASPIC). I thank Trevor Bench-Capon for useful discussions on the topic of this paper.

References

[1] R.H. Thomason. Desires and defaults: a framework for planning with inferred goals. In *Proceedings of the 7th International Conference on Principles of Knowledge Representation and Reasoning*, pages 702–713, San Fransisco, CA, 2000. Morgan Kaufmann Publishers.

[2] J. Broersen, M. Dastani, J. Hulstijn, and L. van der Torre. Goal generation in the BOID architecture. *Cognitive Science Quarterly Journal*, 2:428–447, 2002.

[3] J. Fox and S. Parsons. On using arguments for reasoning about actions and values. In *Proceedings of the AAAI Spring Symposium on Qualitative Preferences in Deliberation and Practical Reasoning*, Stanford, CA, 1997.

[4] P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11(1):113–131, 1995.

[5] J.L. Pollock. Planning agents. In M. Wooldridge and A. Rao, editors, *Foundations of Rational Agency*. Kluwer Academic Publishers, Dordrecht, 1999.

[6] K. Atkinson. *What Should We Do?: Computational Representation of Persuasive Argument in Practical Reasoning*. PhD Thesis, Department of Computer Science, University of Liverpool, Liverpool, UK, 2005.

[7] K. Atkinson, T.J.M. Bench-Capon, and P. McBurney. Computational representation of persuasive argument. *Synthese, special section Knowledge, Rationality and Action*. To appear.

[8] D.N. Walton. *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, NJ, 1996.

[9] L. Amgoud. A unified setting for inference and decision: an argumentation-based approach. In *Proceedings of the IJCAI-2005 Workshop on Computational Models of Natural Argument*, pages 40–43, 2005.

[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. Logic programming as dialog game. Unpublished paper, Division of Computer Science, Asian Institute of Technology, Bangkok, 1994.

[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] G.A.W. Vreeswijk and H. Prakken. Credulous and sceptical argument games for preferred semantics. In *Proceedings of the 7th European Workshop on Logic for Artificial Intelligence (JELIA'2000)*, number 1919 in Springer Lecture Notes in AI, pages 239–253, Berlin, 2000. Springer Verlag.

[14] P.E. Dunne and T.J. Bench-Capon. Two party immediate response disputes: Properties and efficiency. *Artificial Intelligence*, 149:221–250, 2003.

[15] C. Cayrol, S. Doutre, and J. Mengin. On decision problems related to the preferred semantics for argumentation frameworks. *Journal of Logic and Computation*, 13, 2003. 377-403.
[16] P.M. Dung, R.A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170:114–159, 2006.
[17] M. Caminada. On the issue of reinstatement in argumentation. In *Proceedings of the 10th European Conference on Logic for Artificial Intelligence*, 2006. To appear.
[18] J.L. Pollock. *Cognitive Carpentry. A Blueprint for How to Build a Person*. MIT Press, Cambridge, MA, 1995.
[19] G.A.W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90:225–279, 1997.
[20] L. Amgoud, M. Caminada, C. Cayrol, S. Doutre, M.-C. Lagasquie, H. Prakken, and G.A.W. Vreeswijk. Towards a consensus formal model: inference part. Deliverable D2.2: Draft formal semantics for inference and decision-making, ASPIC Project, 2004.
[21] H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15:1009–1040, 2005.
[22] C.E. Alchourrón. Logic of norms and logic of normative propositions. *Logique ét Analyse*, 12:242–268, 1969.
[23] J.L. Pollock. Rational choice and action omnipotence. *Philosophical Review*, 111:1–23, 2003.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Multi-agent agreements about actions through argumentation [1]

Paolo Torroni [a,2]
[a] *DEIS, Università di Bologna, Italy*

Abstract. In this work, we propose a declarative multi-agent argumentation framework for reasoning and argument about actions, equipped with a sound operational model. The foundations of this framework rely on previous results from ALP and from Dung's studies on argumentation. Our approach features declarative knowledge representation and logic based reasoning, agent interaction by argumentation dialogues, and a notion of agreement about actions.

Keywords. argumentation framework, computational logic-based multi-agent systems, practical reasoning in MAS, abductive logic programming, $\mathcal{S}$CIFF, IFF

Introduction

A conspicuous body of literature in multi-agent research addresses the following problem: how to improve an agent's ability to reason effectively about what is the most appropriate course of action to take in a given situation. We believe that the theories and logics of argumentation are a very promising approach to this problem, and we think that the notion of agreement should play a centra role in this picture. Drawing inspiration from previous work on arguments and action persuasion protocols, we propose an Abductive Logic Programming (ALP) framework that supports multi-agent agreements about actions. Such a framework relies on results from ALP and from Dung's studies on argumentation, and it consists of several components:

- a declarative reasoning framework based on the $\mathcal{S}$CIFF operational model;
- a notion of *attack* and an admissible sets semantics;
- a notion of (argumentation) dialogue;
- a notion of agreement about actions.

Agents will be able to reason about their beliefs, goals, and possible actions, and confront themselves with other fellow agents through a collaborative and argumentative process, based on dialogue. The purposes of this framework are manyfold:

- to accommodate a declarative representation of goal, beliefs, actions and constraints in agent;

[1]The author wishes to thank Federico Chesani for his very helpful comments on this article, and the anonymous referees for their generous feedback. This research has been partially supported by the MIUR PRIN 2005 project No 2005-011293, *Specification and verification of agent interaction protocols*.
[2]Correspondence to: Paolo Torroni, DEIS, Università di Bologna, V.le Risorgimento 2, 40136 Bologna, Italy. Tel.: +39 051 209 3767; Fax: +39 051 209 3073; E-mail: paolo.torroni@unibo.it.

- to make agents capable of reasoning upon such knowledge, in order to decide a possible course of actions;
- to let agents interact with each other (by argumentation dialogues) and eventually reach agreements, whenever possible;
- to be able to use agreements made by agents, as possible future internal goals: such agreements will express what are possible future evolutions of the world, in terms of actions and constraints that describe future circumstances achieving goals;
- to achieve a number of formal properties, that insure a consistent and meaningful system evolution.

We will next provide some background on argument and action persuasion protocols and on the logical foundations of our framework, then we will outline the main components of the framework.

Arguing and reaching agreements about actions

The PARMA action persuasion protocol [1] considers a general argument schema for a rational position proposing an action, and handles possible attacks on one or more elements of a general argument schema. Attacks arise from disagreements originating from different sources. PARMA uses four categories of *locutions*, for dialogue control (*C*), action proposal (*P*), inquiry (*A*), and denial (*D*) of existence/validity of elements of a position. Such elements could be goals, circumstances, and actions (not) to be taken. While Atkinson and colleagues focus on addressing divergences on all elements of a position, we focus instead on a more restricted number of issues, and adopt only a small set of locutions. In particular, we only consider some control locutions (*C*) and some proposal/denial locutions about circumstances and actions (*P*/*D*). We do not yet accommodate locutions about logical consequences, purposes, goals or values, which are indeed an important feature of PARMA, and which we plan to address in future developments.

Interestingly, Atkinson and colleagues believe that an Argumentation Framework (AF), such as that developed by Dung, can be used to resolve factual disagreements [1, pag. 159]. We do agree with it. Moreover, the intrinsic complexity of the domain encompassed by PARMA does raise some implementation issues [1, pag. 166], but at the same time it makes PARMA an interesting abstract reference model. We aim to complement Atkinson and colleagues' approach, by giving the foundation of a possible operational multi-agent argumentation framework based on Abductive Logic Programming.

ALP is a computational paradigm aimed to introduce hypothetical reasoning in the context of Logic Programming (see [2] for an introduction to LP and [3] for a survey on ALP). A logic program $\mathcal{P}$ is a collection of clauses, with an associated notion of entailment, usually indicated by $\models$. In ALP, some predicates ("*abducibles*"), belonging to a special set $\mathcal{A}$, can be assumed to be true, if need be. In order to prevent unconstrained hypothesis-making, $\mathcal{P}$ is typically augmented with expressions which *must be true at all times*, called *integrity constraints* ($\mathcal{IC}_S$). An *abductive logic program* is the triplet $\langle \mathcal{P}, \mathcal{A}, \mathcal{IC} \rangle$, with an associated notion of abductive entailment.

$\mathcal{S}$CIFF is an ALP proof-procedure defined by Alberti *et al.* [4,5] as an extension of Fung and Kowalski's IFF [6], and it is the reference ALP framework for this work. One distinguishing feature of $\mathcal{S}$CIFF is its notion of *expectations* about events. Expectations

are abducibles denoted as $\mathbf{E}(X)$ (*positive* expectations) and $\mathbf{EN}(X)$ (*negative* expectations), where $\mathbf{E}(X)$/$\mathbf{EN}(X)$ stand for "X is expected/expected not to happen". Variables in events, expectations and in other atoms can be subject to CLP constraints and quantifier restrictions.

For the purposes of our work, two fundamental concepts are those of *consistency* and *entailment*. We report their definition below.

Definition 1 (Consistent sets of hypotheses) *A set of hypotheses* Δ *is* consistent *if and only if* $\forall$ (ground) $p, \{p, not\, p\} \not\subseteq \Delta$ *and* $\{\mathbf{E}(p), \mathbf{EN}(p)\} \not\subseteq \Delta$

Definition 2 (Entailment) *A (SCIFF) ALP* $\mathcal{S} = \langle \mathcal{P}, \mathcal{A}, \mathcal{IC} \rangle$ entails *a goal* G *(written* $\mathcal{S} \models_\Delta G$*), if and only if:*

$$\begin{cases} \mathit{Comp}(\mathcal{P} \cup \Delta) \cup \mathit{CET} \cup T_\chi \models G\sigma \\ \mathit{Comp}(\mathcal{P} \cup \Delta) \cup \mathit{CET} \cup T_\chi \models \mathcal{IC} \end{cases}$$

where Comp is the symbol of completion, CET is Clark's equality theory, $\models$ *is Kunen's logical consequence relation for three-valued logic,* σ *is a substitution of ground terms for the variables in* G*,* T_χ *the theory of constraints, and* Δ *a* consistent *subset of* $\mathcal{A}$*.*

$\mathcal{S}$CIFF operates by considering G together with $\mathcal{IC}$ as the initial goal, and by calculating a *frontier* as a disjunction of conjunctions of formulae, using at each step one among the inference rules defined in [4]. Given the frontier, at any step a selection function can be used to pick one among all the equally true disjuncts in the frontier. When no more inference rule applies (*quiescence*), if there exists at least one disjunct which is not false, then $\mathcal{S}$CIFF has succeeded, and Δ contains an answer to G. The $\mathcal{S}$CIFF proof-procedure is sound, and under reasonable restrictions it is also complete [4]. $\mathcal{S}$CIFF has been implemented and instantiated into a huge number of scenarios involving agent communication, and it can be downloaded from its web site.[3]

Following Kakas and Toni [8], we map arguments onto abducibles. For example, an assumption $\mathbf{E}(p)$, "p is expected", could be considered as a argument which possibly supports some goal g. Arguments can be circumstances (in the sense of [1]), actions, and related constraints. Thus an agent may justify a goal g by saying, e.g., "in order to achieve a goal g, under the circumstances c and the constraints x, actions a_1 and a_2 should be carried out." In order to take this kind of position, an agent will utter the various elements of it (the circumstances, the goal, the actions, the constraints) via a suitable argumentation language and using the appropriate locutions. Argumentation dialogues will provide implicit links among such uttered elements.

Our notions of argumentation framework, *attacks* relation, and preferred semantics are an (original) instantiation of Dung's work [7] and of the abstract computational framework developed by Kakas and Toni [8].

Definition 3 *A set of arguments* A *attacks another set* Δ *if and only if at least one of the following expressions is true:*

(1) $\mathcal{S} \models_A not\, p$*, for some* $p \in \Delta$*;*
(2) $\mathcal{S} \models_A \mathbf{E}(p)$*, for some* $\mathbf{EN}(p) \in \Delta$*;*

[3] http://lia.deis.unibo.it/research/sciff/.

(3) $\mathcal{S} \models_A \mathbf{EN}(p)$*, for some* $\mathbf{E}(p) \in \Delta$*;*

Definition 4 *An Argumentation Framework (AF) is the pair* $\langle \mathcal{S}, attacks \rangle$.

Definition 5 *A set* Δ *of arguments is said to be* conflict-free *if there are no arguments* A *and* B *in* Δ *such that* A *attacks* B.

It turns out by the definitions above that all arguments A such that $\mathcal{S} \models_A p$ are *admissible sets* of arguments for $\mathcal{S}$, i.e., for all sets of arguments A, if A attacks Δ, then Δ attacks $A \setminus \Delta$.

In a multi-agent context, agents can locally reason about circumstances, constraints, and actions (not) to be taken, based on the $\mathcal{S}$CIFF AF, and produce – at the social level – dialogues in the style of Atkinson and colleagues'.

Definition 6 (Agent system) *An* agent system *is a finite set* A*, where each* $x \in A$ *is a ground term, representing the name of an agent, equipped with a* $\mathcal{S}CIFF$ *program* $\mathcal{S} = \{\mathcal{P}, \mathcal{A}, \mathcal{IC}\}$.

Definition 7 (Performative or dialogue move) *A* performative *or* dialogue move p *is an instance of a schema tell*$(a, b, L[, Arg])$*, where* a *is the* utterer*,* b *is the* receiver*,* L *is the* locution *and (optionally)* Arg *is the* argument *of the performative. For a given* p*, utterer*$(p) = a$*, receiver*$(p) = b$*, locution*$(p) = L$ *and argument*$(p) = Arg$ *(if present). The set of all possible performatives is called* argumentation language*.*

Definition 8 (The argumentation language $\mathcal{L}_{arg}$**)** *The* argumentation language $\mathcal{L}_{arg}$ *is the set of all performatives* p*, such that:*

- *locution*$(p) \in \{$*'enter dialogue'*, *'leave dialogue'*, *'term finished'*, *'accept denial'*, *'state circumstances'*, *'deny circumstances'*, *'state actions'*, *'deny actions'*, $\}$*, and*
- *argument*(p) *is a conjunction of abducible atoms (possibly including* $\mathbf{E}$*/*$\mathbf{EN}$ *expectations) and CLP constraints.*

We have thus defined a concrete language for argumentation, $\mathcal{L}_{arg}$, which includes four dialogue control locutions (type *C*), two proposal locutions (*P*) and two denial locutions (*D*). Agents conversing in $\mathcal{L}_{arg}$ will not exchange formulae stating e.g. consequences of actions, such as implications, but only conjunctions of atoms.

Definition 9 (Multi-agent argumentation framework) *A* multi-agent argumentation framework $\mathcal{M}$ *is a pair* $\langle A, Actions \rangle$ *where* A *is a multi-agent system of agents with the same* $\mathcal{A}$ *which communicate using* $\mathcal{L}_{arg}$*, and* $Actions$ *is a finite set, where each element is a ground term, representing the name of an action.*

Beside assuming a common language, we also assume a common ontology (thus in Definition 9 $\mathcal{A}$ is the same for all agents in A). Otherwise some ontological middleware will have to be used so that, for example, in a position involving a sales, "buy" and "purchase" converge down to the same meaning. This is most necessary in open systems, to prevent misunderstandings arising from the use of terminology.

Definition 10 (Dialogue) *Given an agent system* A*, a* dialogue $\mathcal{D}$ *in a language* $\mathcal{L}$*, between two agents* $x, y \in A$*, is an ordered set of performatives* $\{p_0, p_1, \ldots\} \subseteq \mathcal{L}$*, such that* $\forall p_j = tell(a_j, b_j, L_j, A_j) \in \mathcal{D}, (a_j, b_j) \in \{(x, y), (y, x)\}$

This is a general definition, and it can be instantiated by choosing a concrete language, e.g. $\mathcal{L} = \mathcal{L}_{arg}$. As the dialogue proceeds, the state of affairs involving the two participants also changes, by effect of it.

We record relevant elements of argument in what we call the "state" of a dialogue.

Definition 11 (State of a dialogue) *Given a dialogue $\mathcal{D}$, for each j, $1 < j < |\mathcal{D}|$ the* state of the dialogue*, state($\mathcal{D}, j$) is a tuple $\langle \Psi_j^{sc}, \Psi_j^{dc}, \Psi_j^{sa}, \Psi_j^{da}, \Psi_j^{aa} \rangle$, defined based on the* dialogue history $\mathcal{D}_j = \{p_0, p_1, \ldots, p_{j-1}\}$*, where Ψ_j^{sc} is the* set of stated circumstances*, Ψ_j^{dc} is the* set of denied circumstances*, Ψ_j^{sa} is the* set of stated actions*, Ψ_j^{da} is the* set of denied actions*, and Ψ_j^{aa} is the* set of agreed actions*.*

Intuitively, the state of the dialogue at a step j with respect to circumstances/actions is determined by the last relevant move made. $state(D, j)$ should be defined independently of control locutions, and so that locutions *'state circumstances'* and *'state actions'* reset somehow the current state. If an agent utters, e.g., *'state circumstances'* at step j, the set of stated circumstances will only contain the new circumstances Ψ_j^{sc}, until some agent again states *'state circumstances'*. Based on this notion of state, we now define what we mean by argumentation dialogue, and then by agreement.

Definition 12 (Argumentation dialogue) *Given a multi-agent AF $\mathcal{M} = \langle A, Actions \rangle$, an* argumentation dialogue $\mathcal{D}$ *between $x, y \in A$* about a goal G_x *is a dialogue such that:*

1. $p_0 = tell(x, y,$ *'enter dialogue'*$, G_x)$*;*
2. $\forall p_j = tell(a_j, b_j, L_j, A_j) \in \mathcal{D}$*, the arguments of a_j and b_j follow from their respective knowledge and are "consistent" with some previous dialogue state;*
3. $\nexists p_j, p_k \in \mathcal{D}$ *such that* $p_j = p_k \ \wedge \ j \neq k$*,*

where for a given set Δ, $actions(\Delta) = \{\mathbf{E}(a) \in \Delta$ such that $a \in Actions\}$.

Thus, in an argumentation dialogue, the agents focus on a specific goal (1). They do not exchange purely "dialogical" arguments, but genuine products of their own reasoning based on the knowledge available to them (2). For example, we may require that circumstances/actions stated are supported by the uttering agent, and for those denied the utterer of the denial is able to produce an attacking argument based on the goal subject of the dialogue. We require that an agent does not utter the same performative twice (3).

At each step, the dialogue develops by agent reasoning on previous steps' states, to propose new arguments to the receiver. Dialogue moves need not directly address the previous move, but should be left free to refer to moves uttered in the past, in the course of the same dialogue. This leaves either agents the option to try alternative arguments, so that the dialogue can proceed even if an agent does not have an answer to the last move.

We do not define when a dialogue is terminated, but instead we define what it means that agents reach an agreement:

Definition 13 (Agreement) *Given a multi-agent argumentation framework $\mathcal{M}$, an* agreement *between two agents $x, y \in \mathcal{M}$ about a goal G_x is a set $\mathcal{C}$ such that there exists an* argumentation dialogue $\mathcal{D} = \{p_0, p_1, \ldots\}$ *between x and y about G_x, whose $state(D, j)$ is such that $\Psi_j^{aa} = \mathcal{C}$ for some j.*

In other words, we say that two agents reach an agreement when they come up in the course of the same dialogue with a set $\mathcal{C}$ which contains the same actions. In a

concrete definition of argumentation dialogue, agreements should be supported by the same arguments (circumstances) from both sides.

Indeed, when combining this notion of argumentation dialogue with the agent reasoning based on the $\mathcal{S}$CIFF AF, we must ensure that arguments produced inside dialogues are conflict-free and admissible for at least one of the two agents involved.

Final remark

We have proposed a declarative framework for multi-agent reasoning and argumentation about actions, a sound operational model, an admissible sets semantics, a notion of (argumentation) dialogue and a notion of agreement about actions. Our approach accommodates a declarative representation of the agent knowledge, upon which agents can reason, and interact by argumentation dialogues. Argumentation dialogues are useful because through them agents may eventually reach mutual agreements, which they can directly use, for example by adopting them as possible future internal goals.

We believe that $\mathcal{S}$CIFF could be an interesting framework to approach practical reasoning in agent systems. Relevant work of literature offers considerable insights into practical reasoning, but stands in need of concrete realisation. Moreover, since the practical syllogism underlying a general argument schema has an essentially abductive flavour, ALP seems a promising approach. In future work, we intend to give a precise semantic characterization of the general framework presented, in which argumentation dialogues are one of the main tools for ensuring a consistent and meaningful system evolution, and we intend to start a thorough empirical analysis of what can be done in practice with it. Indeed, this initial work leaves out many distinctive problems of practical reasoning, such as side effects, alternatives and potential differences in two arguing agents' interests and values, which we also intend to address in future developments.

References

[1] K. Atkinson, T. Bench-Capon, and P. McBurney. A dialogue game protocol for multi-agent argument over proposals for action. *Journal of Autonomous Agents and Multi-Agent Systems*, 11:153–171, 2005.

[2] J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag, 2nd extended edition, 1987.

[3] A.C. Kakas, R.A. Kowalski, and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6):719–770, 1993.

[4] M. Alberti, F. Chesani, M. Gavanelli, E. Lamma, P. Mello, and P. Torroni. Verifiable agent interaction in abductive logic programming: the $\mathcal{S}$CIFF proof-procedure. Tech. Rep. DEIS-LIA-06-001, U. of Bologna (Italy), 2006.

[5] M. Alberti, M. Gavanelli, E. Lamma, P. Mello, and P. Torroni. The $\mathcal{S}$CIFF abductive proof-procedure. In *Advances in Artificial Intelligence, LNAI* **3673**:135–147. Springer-Verlag, 2005.

[6] T. H. Fung and R. A. Kowalski. The IFF proof procedure for abductive logic programming. *Journal of Logic Programming*, 33(2):151–165, November 1997.

[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] A.C. Kakas and F. Toni. Computing argumentation in logic programming. *Journal of Logic and Computation*, 9:515–562, 1999.

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Suspicion of Hidden Agenda in Persuasive Argument

Paul E. Dunne
Department of Computer Science, The University of Liverpool, U.K.

Abstract. The process of argument may be aimed at persuading an agent to perform a particular act which its proposer views as advantageous to their own interests. The proponent's primary motivation for desiring a given action to be executed may, however, differ radically from those advertised reasons given as justification: in particular, an agent may wish to hide the true purpose of proposing an action, since knowledge of such motives would be very likely to result in it not being performed. This scenario has been described as the proponent agent having a "hidden agenda", and is a widely-recognised phenomenon in persuasive argument settings such as electioneering. The purpose of this paper is to consider various issues arising when an agent seeks to assess a case for action presented to it as motivated by a potentially damaging hidden agenda.

Keywords. dialogue based on argument, decision making based on argumentation

1. Introduction

One may pursue an argument by making a persuasive case; the object of persuasion being to move another to an action whose completion is viewed as advantageous. The nature of such advantage, that is to say the motivation underlying the call to action, may involve many divers factors; and while proponents may be prepared to divulge certain of these in furtherance of their case, equally so there may be other consequences which there is a reluctance to expose in the course of debate since knowledge of such would be likely to result in the sought after action not being executed. The term "*hidden agenda*" has been coined to describe those scenarios in which the predominant motivating reasons are precisely those a proponent seeks to avoid revealing in advance of any commitment to action being obtained.

The deployment of hidden agenda is, of course, a widely-recognised aspect of electoral campaigns, from the most parochial special interest organisations through to supranational councils. In such arenas a single party solicits support from a constituency of electors – i.e. the action desired is that of a vote being cast for the party – adducing as reasons a programme whose implementation will be in the voters' interests. The presumption underlying such scenarios is that the chosen candidate will enjoy a position of influence (usually considered to be advantageous) and, thereby, able to bring about the promised programme. In such contexts, however, the primary aim of attaining a position of influence may be to realise policies that, were the electorate fully aware of such, would leave the majority disinclined to provide support. For example, a party desirous

of significantly boosting military spending – an investment concerning which no rational electorate is likely to feel sympathy in the absence of any immediate threat – might promote its intention of imposing punitive tax increases (when addressing less affluent social groups) whilst pledging to slash welfare expenditure (when canvassing among more economically advantaged groups). If its campaign is successful the party intends to honour both commitments but to deploy the consequential additional revenue neither to reduce taxes (as highly renumerated supporters might have hoped from savings in welfare spending) nor to improve welfare provision (which could be subsidised through increased taxes): the revenue gained will, instead, be used to realise the (hidden) aim of bolstering defence expenditure.

The position whereby a notionally "damaging" but primary motivation is kept secret may arise at the level of individual, i.e. one-to-one, interaction in addition to one-to-many contexts such as electioneering. As, an example,[1] a resident of an apartment in the Soviet Union was approached by the father of her neighbour, who was terminally ill. The father said that his daughter wanted to divorce her husband, and asked for information about the husband to facilitate this. This information being provided was then used to get the divorce. Following the neighbour's death, however, her former husband did not get possession of the (state-owned) apartment, as he would have done without the divorce: it went instead to the neighbour's father whose reason for seeking the information and arranging the divorce was to get the apartment for himself.

While there has been a significant body of work touching on issues relevant to the main theme of this paper, detailed treatments of hidden agenda in dialogue and argument from a computational perspective appear to have been neglected. So, for example, the issue of strategies for "hiding" private data in multiagent systems contexts has recently been the subject of work by van Otterloo [1] and Paruchiri *et al.* [2]: both concern strategies that avoid revealing "private information" and, to some extent, the scenarios which are the principal focus can be seen as originating from the observations of Spence [3] who considered how the choice of actions adopted by an agent leads to the undesired side-effect of allowing private information about the agent to be deduced as a consequence. In these papers the aim of preserving privacy is treated as a desirable end in itself, i.e. an agent is not, necessarily, trying to disguise some ulterior purpose.

Rather closer to the concerns of the current paper, however, is the recent article by Silverman [4] regarding the notion of hidden agendas in patient-doctor consultations. Silverman attributes the phrase "hidden agenda" to Barsky's 1981 article [5], in which it describes the phenomenon of patients withholding details of their exact concerns until an extremely late stage of the consultation. Recognising that this has adverse effects on diagnosis and treatment, and that the reasons for such reticence are not necessarily linked to the nature of the complaint, [4] proposes a schedule of questions and procedures designed to encourage "early disclosure" of such hidden agendas. While the consultation dialogue underlying this scenario is of interest from the view of argumentation settings, there are some differences between the notion of "hidden agenda" as studied by [4] and the current article's perspective of interest. Notably, in the medical setting there is a presumption that a patient is not *deliberately* hiding what concerns them – at most there may be some reluctance openly to discuss these issues. In scenarios such as the electioneering setting above, however, the principal focus of a protoganist is in *conciously avoiding* re-

[1]The author is grateful to David Hitchcock for providing this example.

vealing the true motivation. The purpose being not simply that of protecting privacy but to expedite a specific action which would be unlikely to happen were its executors fully appraised of the proposer's reasons.

Thus, in the context of interagent dialogues, there are issues such as:

a. Factors in the *environment* within which a case is presented that an agent might consider render it vulnerable, i.e. engendering caution that seemingly harmless actions will have damaging (and therefore hidden) consequences, e.g. the previous history of interaction between the agents.
b. Factors in the *presentation* of an argument that could awaken suspicion that its proponent has a hidden agenda.

In addition, although we do not pursue these topics here, there are questions regarding the form of challenges that an agent might pose which, if not treated in a "satisfactory" manner would confirm suspicion that such an agenda is present, and of appropriate strategies that an agent might bring to bear in order to discover the the exact nature of a proponent's reasons for desiring some action.

Our intention in the current article is to propose a number of criteria that can be translated into computational terms in tackling those issues raised in (a) and (b) above. These elements are discussed in the next section and illustrated with respect to a detailed example study in Section 3. Conclusions and directions for further research are offered in Section 4.

2. Factors occasioning suspicion of a hidden agenda

The focus of subsequent discussion considers the following scenario: there are two agents – A and C. The latter wishes A to perform some action, α but, were he do so, the result although furthering C's interests, would be detrimental to A. While C is aware of this (and, indeed, such may be her primary motivation for desiring α), A is not and, in presenting her case, she must avoid revealing her principal reasons and the consequences of A performing α. It is assumed that A neither naively accepts every proposal other than those which are self-evidently damaging, nor is ultra-sceptical to the extent that every action, other than those clearly to his advantage, is refused. In sum, A is cautiously prepared to entertain C's wishes but may require a convincing case to be made before reaching his final decision.

2.1. Environment

Prior to any formal argument being presented there may be a number of factors that an agent might identify as leaving it vulnerable. The significance of such factors is that – if recognised – these may suffice to change a moderately cautious attitude into one that is extremely sceptical and, thereby, much less inclined to act upon the requests of another. The term *environmental factors* is introduced to describe these: such arise from the *context* in which a case will be made, i.e. indepedent of its presentational *style*.

Thus, having had an action, α, proposed by C, one has questions such as:

E1. Is there anything in my past interaction with C that I should consider?
E2. Are there unusual features of the *present* context within which α is requested?

E3. Are there aspects of my *advertised* attitude to C that she might regard as detrimental to her interests?

For example, suppose α is a request by C to borrow $£100$. If, in some earlier transaction, A had borrowed $£50$ from C but never reimbursed her, then A in recalling this outstanding debt via E1 (or, indeed, E3) may suspect that C is not only trying to recover what she is owed but also wants to leave A in this same state of "losing" $£50$. Again, under E2, even if A does not have (or cannot recall) a history of defaulting, he may become suspicious that C will not repay him if, for example, she accompanies her request by offering to pay an exorbitant interest rate. Finally, an example of E3 would be that A has expressed his opinion (perhaps, humourously) to others that C has a poor history of repaying small loans, even though he does not, necessarily, believe this himself.

It can be seen that in order to give (E1)–(E3) appropriate consideration, A must be capable of some non-trivial degree of objective introspection. This, potentially, poses difficulties in applying these criteria in a computational setting: typically it will not be possible to retain information respecting all details of all interactions with all agents in a given system, thus the extent to which (E1) can be judged may be limited only to comparatively recent history. Similarly, while for certain classes of request – e.g. financial transactions – there may be specific indicators that an agent may highlight as "unusual features", it will not be possible to account for *every* such case.

Thus, while such environmental factors may prove valuable in recognising vulnerability to hidden agenda actions, such may be problematic to identify.

2.2. Presentation

Suppose that A is not disposed to act as C requests indicating his indisposition with a refusal that may leave open the possibility of further debate. Notice that A may choose to act thus, even though he has failed to identify environmental factors giving cause (or has deemed such to be of insufficient weight). Faced with an initial refusal, C, must present an argument whose force will suffice to change A's stance and it is in the nature of such persuasive cases that A has some further possibility of suspecting α has hidden advantages for C (therefore, remaining hidden, α may be presumed detrimental to A).

While it is not claimed that the following, some of which are developed in more detail, represent the totality of such *presentational* indicators, these certainly describe an important subset, their overall effect being to convey a sense that the proposer lacks confidence in their ability to promote a sound case.

P1. **Logical Fallacy**: e.g. *non sequitur*, *petitio principi* in arguing the case.
P2. **Inconsistency**: can occur as a particular instance of (P1), but another example is the use of incompatible arguments dependent on audience, e.g. the electioneering example given above.[2]
P3. **Prevarication**: periphrasis – e.g. "at this precise moment in time" rather than "now"; lengthy digressions largely irrelevant to questions raised; repetition, etc.
P4. **Obfuscation**: responding to challenges by answering a question other than that asked; introduction of material not germane to the point at issue, etc.

[2]Inconsistency may, also, appear within a *single* statement unrelated to previous arguments used, e.g. Antony's provocation of the Roman mob "It will inflame you, it will make you mad / 'Tis good you know not that you are his heirs. " (*Julius Caesar*, Act III, ii, ll. 124–5).

P5. **Ambiguity**: expression of ideas in such a way as to admit multiple interpretations, e.g. substituting "give" for "lend" in contexts where the latter is being debated; statements carrying subtexts, e.g. "You were the first person I thought of".
P6. **Mendacity**: from, apparently, supportive but unverifiable assertions through to deliberate falsehood, e.g. "it's not for me it's for someone else".
P7. **Appeals** *ad hominen*: ranging from sycophancy attempting to exploit an audience's personal vanity through suggestions of loss of status (e.g. "what would people think if they knew you'd refused?") to direct denigration (e.g. "only a complete coward could act like this").
P8. **Excess**: use of extreme emotive language, highly-coloured imagery, etc. For example, "It will be an absolute disaster if you refuse; I've no idea who I'll be able to turn to for support if that happens."
P9. **Obscurity**: e.g. phrasing of statements so that their intention is unclear.

Recalling that these are not promoted as an exhaustive set of criteria, the question that now arises is to what extent can these be interpreted as computationally recognisable features of a persuasive case.

In order to discuss this issue let us return to the example in which α is the act of advancing $\pounds 100$. Consider this dialogue between the two protagonists A and C:

C. Will you lend me $\pounds 100$, please? I'll be able to pay you back next week.
A. No, I don't think so. What do you need the money for anyway?
C. I don't want the money for myself, it's just that I owe B $\pounds 50$ and I know he needs to pay you $\pounds 25$, so if you think it about it, you would only be giving me $\pounds 75$ (not $\pounds 100$). I know, you're thinking, "what about the other $\pounds 50$?": I shouldn't really say what that's for because it's meant to be a surprise and it will be spoiled if I say anything. As I said, the money isn't for my own use, it's just that I saw something at this sale, where was it now? You know the place, they have one every few months and there's always some really good offers available; never mind I'll remember the name in a little while, I'm sure. Of course, you were the very first person I thought of to ask: everyone says you're so generous and relaxed when it comes to money and it's such a trivial amount that I'm sure you wouldn't be so miserly as to refuse. I really would be so enormously grateful if you were able to do this one tiny favour for me. I can promise you that it will be absolutely the very last time I ask you for anything and you can be sure no one will ever know about it. If you're uncomfortable with the idea then I do understand: we'll just say you owe me a favour.

Examining C's response to A's initial rejection of her request identifies its basic form as containing (in order of their occurence) the elements listed below:

a. The money isn't for my own use.
b. The amount *given* is *only* $\pounds 75$ not $\pounds 100$ (someone I owe would repay you).
c. I'm not able to say what will happen with the remainder.
d. The money isn't for my own use. (again)
e. I've seen something [I'd like to buy (implied)] in a sale.
f. You were the first person I thought of to approach.
g. You are generous and it's not that much.
h. I really need the money.

i. I won't bother you again and no one will know about it.
j. It's all right if you refuse but then you'll be under an obligation to me.

Within this breakdown one finds all of the factors discussed earlier. Both (b) and (j) involve *non sequitur* fallacies (P1): the miscalculation in (b) is self-evident, while, for (j), it is claimed that failing to carry out the desired action implies that A will be in debt to C: yet why this should be so remains unexplained. Although C pre-empts the charge of inconsistency, prior to (c), (P2) is still apparent in reconciling (a) and (d) with (e) and (h). The lengthy digression surrounding (e) provides a clear sign of (P3); the refusal to give a direct answer in (c) being an indicator of (P4). There are a number of aspects suggesting the factor (P5): the use of "only" and "given" in (b), the latter suggesting that the money will *not* be treated as a loan; the wording of (f) raises the subtext issue of *why* exactly A is so favoured; similar considerations arising in (i). While it is not immediately evident that (a), (d), (f) and (i) are deliberate falsehoods, each of these is, at best, unverifiable, signalling the presence of (P6) in C's argument. The manner in which (g) is expressed presents evidence of (P7) (in both positive and negative senses), whereas the style of (h) is phrased so as to signal (P8). Finally, it is unclear what rôle the statements (f), and (i) are meant to fulfil in the argument: a sign of (P9).

Notice that, unlike the computational overhead that may arise in recognising environmental factors, a number of the presentational aspects can, albeit rather naively, be detected and challenged. So, potential logical deductive fallacies ("if X then Y") can be investigated, e.g. in response to (j), the obvious challenge is "Why **refusal⇒obligation**?". Again, statements that seem irrelevant to the debate, e.g. (i), may be noted, and the proposer subsequently challenged to clarify their relevance. More subjective factors – (P7) and (P8) for instance, may be rather more problematic, however, one could envisage use of mechanisms such as thesauri to flag the more blatant uses of these.

3. Detailed Example Study: persuasive argument in Aeschylus' *Agamemnon*

Tragedy, the banalities of t-shirt slogans notwithstanding, does not simply "happen". So, the pivotal event in Aeschylus' *Agamemnon*, the murder of the title character by his wife Clytemnestra upon his return after the ten year siege of Troy, is calculated and premeditated; its motivation rooted in a complex family history and the circumstances surrounding the death of their eldest daughter (Iphigenia).

The dialogue between Agamemnon and Clytemnestra in Aeschylus' treatment affords a particularly fine illustration of persuasive argument in pursuit of an extreme hidden agenda. The discussion presented in this section considers the dialogue between the the two central characters analysing its pursuit of a hidden agenda with respect to the overview given earlier: thus, considering the reasons as to why Agamemnon ought to suspect malicious intentions; the action that Clytemnestra tries to persuade him to carry out; and the critical points within the ensuing dialogue[3] where Agamemnon shifts from an initial outright refusal to his final compliance with Clytemnestra's expressed wish. In

[3]In discussing this dialogue, I use Campbell's 1936 edition of the text [6], primarily because of one radical difference from Smyth's better known 1926 version, [7]. The section of interest is between ll. 855–974 (in both editions). Campbell, however, transposes a key speech of Clytemnestra's to ll. 930–944 from its traditional placing at ll. 958–972.

order to appreciate the relevant context, a brief review of the dramatic conventions within which the play was first presented is given.

3.1. Conventions and Context

In keeping with their origin from religious festivals, an underyling theme of many classical tragedies concerns the fate visited upon those who fail to accord due reverence and respect for the gods, this failure often dramatically epitomised by acts of *hubris* [4] committed by the central character. The nature of such acts may range from open defiance of or contempt for particular deities, e.g. Pentheus' attempts to suppress the cult of Dionysus in Euripides' *The Bacchae*, but these may also be rather more subtle errors of judgement unwittingly committed, e.g. Oedipus' presumption that he himself cannot be the direct cause of the misfortunes afflicting Thebes, in Sophocles' *Oedipus Tyrannos*.

In observing this convention that the fate of even the most powerful figures can be determined by a single ill-judged act, Aeschylus was faced with resolving a problem relating to the internal and external dynamics of the plot development, i.e. between the attitudes and viewpoints held by the principal characters *in persona* and the perceptions and knowledge of the contemporary audience. Thus, Clytemnestra seeks to argue Agamemnon into carrying out an act that will cause him to lose favour and thereby provide a propitious context for her plan to kill him: her case must convince Agamemnon, firstly that the action she proposes is "harmless", and, secondly, that she has no hidden agenda. In this context, Agamemnon must be presumed not to be aware of what Clytemnestra's intentions are. For the external audience in order to maintain a sense of dramatic credibility, her arguments must also sound plausibly convincing. This audience, however, already knows that she *will* be successful[5] thus the dramatic focus is not on the issue of whether her plan will succeed but rather on the manner in which it is effected and the consequent weaknesses of Agamemnon's character that it exploits, e.g. a television audience watching an episode of *Columbo* knows from the outset who has committed the crime being investigated; interest is maintained by seeing how the eponymous lieutenant – with similar insight but without concrete evidence – entraps the guilty party.

The dialogue between the two central characters is now examined in detail in terms of Clytemnestra's motives (and the extent to which Agamemnon shounld be cautious).

3.2. Grounds for Suspicion of Hidden Agenda

All of the environmental factors discussed above can be identified. The spectator has already been alerted to one of Clytemnestra's motives (and, arguably, the principal cause of her subsequent action) in the long choral passage preceding Agamemnon's entrance: in order to appease the goddess Artemis and so alleviate the adverse weather conditions

[4] The Greek word – $\upsilon\beta\rho\iota\varsigma$ – has no exact English equivalent: its senses, paraphrasing [8], include "wanton violence arising from the pride of strength, passion, etc.; riotouness, insolence, lewdness, licentiousness; outrage, spiteful treatment, gross insult to others; acts occasioning harm, detriment, loss, etc." In total, indicative of an attitude of over-weening pride, arrogance and thoughtlessness.

[5] In much the same way as the plots of Shakespeare's dramas are familiar to us and the events described in the cycles of mediaeval "Mystery plays" equally so to their audiences, the 5th Century B.C. Athenian spectator would be fully aware that Clytemnestra does kill Agamemnon, that her son, Orestes, will "avenge" him by an act of matricide (in *The Choephori*), and that Orestes will, ultimately, be acquitted of this crime (in the final part of the trilogy, *The Eumenides*).

preventing the army under his command sailing from Aulis to Troy, Agamemnon had ritually sacrificed their daughter, Iphigenia, (ll. 184–247).[6] A second is the absence of his son: still a child when Agamemnon left for Troy, Orestes would now be recognisable as a potential successor, but he is not among the company welcoming him back. Finally, there is the presence in his entourage of the Trojan king Priam's daughter, Cassandra, whom Agamemnon has taken as concubine.

Of these factors, some discussion of the first is given in the following section, while as regards Cassandra, although neither protoganist refer to her directly, it is symptomatic of Agamemnon's essential weakness of character that he fails to realise (or, possibly, is unconcerned) that Clytemnestra might, justifiably, interpret the prominent status given to Cassandra as a personal slight.

It is, however, the second that is of particular interest. Agamemnon makes no mention of Orestes' absence in his opening speech (ll. 810–854), and it is Clytemnestra herself who draws attention to it in her welcome (ll. 855–913) citing concerns about the numerous rumours of Agamemnon's death leading to popular revolt as the reason (ll. 879–886). That Clytemnestra attempts to justify a situation which Agamemnon has left unmentioned, ought in itself to engender some unease as to her intentions. It is, however, the manner in which she raises this issue which indicates what her plans are. The actual text presents a remarkable richness of meaning, so that Orestes' absence is introduced in such a style as to hint at Iphigenia's.

The effective sense of ll. 877–879[7] is rendered as "*It is for such a cause our child stands not beside me, our pledge of love, Orestes.*". The significance and combination of meanings in these lines should not be underestimated: Clytemnestra delays identifying "our child" until the last word of her sentence, naming Orestes but intending Iphigenia.[8] The phrase "our pledge of love" combines two layers of meaning: on the one hand a somewhat sentimental reference to Orestes ("our" being Agamemnon and Clytemnestra); on the other hand, with "our" the royal use of first person plural for first person singular it refers to Iphigenia and Clytemnestra's commitment (i.e. pledge) to avenge her. Noting this also lends a double sense to "for such a cause": as a forward reference to ll. 879–886 presenting the reasons underlying Orestes' absence ; but also, with Iphigenia in mind, "such a cause" is the Trojan war and the extreme steps taken to promote its successful conclusion. Overall, within these three lines, Clytemnestra describes her real concerns whilst cloaking these under cover of an apparently innocuous explanation: a powerful indicator of the presentational factor P5.

In sum, the three environmental factors identified in Section 2 are present: an event in the past relationship of the protagonists (Iphigenia's death); an unusual feature of the

[6]That this is a significant factor, note, e.g. ll. 1412–1425 wherein Clytemnestra castigates the city elders for their hypocrisy in condemning her having cravenly accepted this act of her late husband ten years earlier. The great modern reworking of this cycle – O'Neill's *Mourning Becomes Electra* with its transposition to the post-American Civil war period – focuses on the liaison between Clytemnestra (Christine Mannon) and Aegistheus (Adam Brant) – a cousin of Agamemnon's (Ezra Mannon) – as the primary motivating factor, making no attempt to parallel Iphigenia.

[7]i.e., $\epsilon\kappa\ \tau\omega\nu\delta\epsilon\ \tau o\iota\ \pi\alpha\iota\varsigma\ \epsilon\nu\theta\alpha\delta'\ o\upsilon\ \pi\alpha\rho\alpha\sigma\tau\alpha\tau\epsilon\iota\ /\ \epsilon\mu\omega\nu\ \tau\epsilon\ \kappa\alpha\iota\ \sigma\omega\nu\ \kappa\upsilon\rho\iota o\varsigma\ \pi\iota\sigma\tau\omega\mu\alpha\tau\omega\nu,\ /\ \omega\varsigma\ \chi\rho\eta\nu,\ O\rho\epsilon\sigma\tau\epsilon\varsigma$.

[8]Oddly, for what is a near literal translation, [7] misses these subtleties: Smyth distorts the word ordering from the Greek and, further, renders "$\pi\alpha\iota\varsigma$" as the gender-specific "boy" rather than the equally valid gender-neutral "child", $\pi\alpha\iota\varsigma$ being the source of English words such as "paediatric", "paedagogue", etc.

current environment (Orestes' absence); and the suggestion that one agent is treated in a detrimental fashion (Cassandra's presence).

3.3. Action and Persuasion

The circumstances of her daughter's death and the consequent motivation these provide have been alluded to above: their nature marking, undeniably, an act of hubris on Agamemnon's part.[9] As such (and in affronting the deities honouring the status of motherhood) there might seem to be ample propitious signs for Clytemnestra's plans to succeed. Yet she endeavours to persuade Agamemnon into one further ill-judged action: summoning attendants she orders these to lay out a path of expensive silks dyed a deep purple shade, and invites Agamemnon to walk, unshod, from his chariot to palace upon these.[10] Both the material and its colour were indicative of great wealth: purple, particularly in the rich dark shade denoting royalty, was a notoriously expensive and difficult dye to produce: Clytemnestra wishes the conspicuous squandering of a rare commodity to be seen as a simple act of homage to Agamemnon honouring his victory over Troy, but hopes that by complying with her request, he will be judged as arrogant, triumphalist and profligate, so losing favour and ensuring her plan's success. It is observed that, although Agamemnon's past actions are such that it should be unnecessary to persuade him into further rash acts, from Clytemnestra's view there is a difficulty: the prosecution of the Trojan war was successful and Agamemnon has returned safely, in spite of his flagship encountering a violent storm (ll. 650–660). Thus, two of the "obvious" means of (divine) retribution – death during the siege itself or prior to reaching his homeland – have not resulted: Agamemnon appears still (divinely) favoured and, in order to end this status, she must arrange for him to perform one final ill-considered act.

Agamemnon enters and in his opening speech (ll. 810–854) formally thanks the gods for his victory and safe return to Argos, briefly recounts the the occasion of Troy's downfall, and promises to put the affairs of state duly in order.[11] In total, the address is formal and reserved – no member of his family is mentioned either by name or implication – and it is apparent that Agamemnon has does not suspect any conspiracy against himself.

The reaction to the long oration by Clytemnestra (ll. 855–913), however, seems to change this. In contrast to that to which it replies, this speech is gushing, sycophantic and makes use of excessive (near hysterical) imagery: she speaks of her love for Agamemnon (ll. 855-858); the anxiety she suffered waiting alone for his return and her distress at the many reports of his death (ll. 859–872) and how on many occasions she had been close to suicide as a result (ll. 873–876). Clytemnestra continues by talking of Orestes' absence (ll. 877–886), expresses her relief at Agamemnon's safe return (ll. 887–905); and, in conclusion, bids her attendants to lay out a path of purple silk, calling on Agamemnon to walk on this to his palace (ll. 906–913).

Agamemnon's response ([6, ll. 914–929]), at first suggests that the extreme style in which Clytemnestra's welcome is couched has alerted him to the fact that she has some,

[9] That this is, indeed, the case has been noted at a number of earlier points in the the play, e.g. the suggestion in l. 188 that Agamemnon had acted too hastily.

[10] cf. the traditional "red carpet" ceremonial of the present-day.

[11] At this juncture, (l. 854), [7] interpolates the stage direction "He descends from his chariot". This, however, makes little sense given Clytemnestra's (later) command, as a prelude to the act into which she wishes to lure him, "εκβαιν' απηνης τησδε, μη χαμαι τιθεις / τον σον ποδ', ωναξ, Ιλιου πορθητορα." (ll. 907–908): "Your chariot dismount, but setting not upon the common ground, the foot that trampled Troy."

as yet unknown, ulterior purpose. He reacts with disdain to the address that has just been delivered, mocking its length (ll. 914–916)[12] and dismissing her flattery as inappropriate. Finally, (ll. 920–929), he indicates exactly why he will not enter his home in the style she requests, recognising such an act as one that would invite envy and likely to incur the enmity of the gods: in sum, because to do so would have exactly the outcome that Clytemnestra intends.

At this point it seems that Clytemnestra's plan has failed and she will need to rely on "chance" (independent of any divine opprobium toward Agamemnon) in expediting his murder: Agamemnon has indicated his awareness of the risk and the forthright dismissive tone of his response suggests not only that he considers his decision to be final, but furthermore indicates a refusal even to countenance any further debate.[13]

In her first address, Clytemnestra had hoped that exploiting the hints of triumphalism in Agamemnon's opening speech together with his inherent vanity would suffice to make a persuasive case. His curt refusal indicates that, if she is to succeed, a rather more subtle argument is called for: not only must this be comparatively brief (cf. ll. 915–16: potentially Agamemnon's suspicions have been aroused by one of the presentational factors considered above: P3), but its tone must also be calmer and more reasoned (cf. ll. 917–919: her earlier argument is promoted too forcefully, another indicator: P8). Taking both of these factors into account, her response ([6, ll. 930–944]) forms one of the most famous surviving speeches of ancient drama,[14] its purpose not so much to bring about execution of the act itself but rather to create a state of indecision conducive to further argument. Three counterarguments are put against Agamemnon's refusal: that the sources of the materials involved are effectively unlimited (silk and the natural dye used, ll. 930–932) that, as a proportion of their actual wealth, the quantity squandered is trivial (ll. 933–934); and, finally, that had she been advised to do so she would have readily dedicated significantly larger quantities (ll. 935–937). The concluding section (ll. 938–944) uses complex figurative language that while ostensibly welcoming the restoration of good fortune to their home actually hints at the nature of her true intentions.

Now this speech is significantly shorter than its predecessor (15 as opposed to 59 lines) and its tone is, despite the richness of language and metaphor, rather more measured and calm: it contains, however, two elements which signal a hidden agenda with respect to the criteria discussed in Section 2. Firstly, the supporting arguments are fundamentally fallacious, i.e. P1. Informally, the justifications for conspicuous waste are, in turn: "there's plenty more where that came from" (rebutted by observing that, e.g. although a commodity in its natural state may be plentiful, considerable labour might be needed to transform such into a desirable artefact, e.g. handwoven cloth, diamond, oil, etc.); "the amount involved is as 'nothing' to us" (akin to arguing that, to external perceptions, "$100,000 is as 'nothing' to Bill Gates", except that to Bill Gates, himself, it is still $100,000); and, finally, "if I'd been asked to I would have done even more than this" (the classical fallacy whereby justification of an action is provided via an unsubstantiated premise). The second element in the argument's presentation, principally evident in its concluding lines, is its ambiguity and obscurity, i.e. the factors P5 and P9: as in ll. 877–

[12] ll. 915–916, provide one of the play's rare humourous instants: "well matched was your speech to its theme of my absence: for, indeed, both went on too long."

[13] It is from this point that [6] and [7] diverge.

[14] The text, my (rather liberal) translation and a more detailed commentary in terms of (P1)–(P9) are given in the Appendix.

879 discussed earlier, Clytemnestra reveals her true agenda under cover of apparently innocuous sentiments. Even within the "persuasive" section of her speech, she makes veiled references to Iphigenia's fate, e.g. ll. 935–937 hint that Agamemnon accepts *some* oracles despite the consequences for his daughter yet he baulks at the thought of ruining some tapestries in the face of others.

It is clear from his response, (l. 945), that these indications have passed unnoticed. Now hesitant and uncertain, he provides an invitation to further debate – "With untroubled mind, I would act in all things". This, together with the thirteen lines that follow (ll. 945–957), now form an interchange of seven arguments: on each occasion Clytemnestra responds to a point raised by Agamemnon or introduces new issues. So (l. 946) she resorts to a direct challenge ("Speak not to go against my wishes"), receiving only the blustering response that she can be certain he will not change his mind. She then attempts to provoke him by taunting that his reluctance indicates weakness (l. 948, "Only from fear could you have sworn to act thus"), to which Agamemnon merely reiterates his stance. It is only in the final two exchanges (ll. 954–957) that Agamemnon appears to raise direct challenging questions: pointing out the unusual vigour with which her argument is being pursued (but not asking *why* this is so); and, finally, suggesting that his compliance can be of no obvious benefit to her (but, again, not pursuing this issue). Having failed to detect (or even to probe for) her true motivation, Agamemnon concedes despite his concern over the consequences (ll. 958–971).

4. Conclusions and Development

The main aim of this article has been to highlight elements in an argument that could be interpreted as evidence of a "hidden agenda" underlying it. While such indications, if recognised, may suffice to create suspicion that such an agenda is present, there are two further developments which are of interest. Having become cautious as to its proposer's true motivations, one would expect a rational agent to attempt to confirm whether such suspicion is justified: a natural way to proceed in such circumstances would be to identify specific challenges to be put to the proposer, the answers to which would allow an agent to decide if the action put to it is best ignored. Identifying the structure and form of such challenges is the subject of continuing work. Finally, and potentially rather more problematic, given that an agent has concluded (on the basis of the argued case and subsequent challenges) that a hidden agenda *is* indeed present, an issue of some interest concerns the strategies an agent can adopt in order to discover its precise form.

References

[1] S. van Otterloo. The value of privacy. In *Proc. AAMAS'05*, pages 1015–1022, 2005.
[2] P. Paruchiri, M. Tambe, F. Ordóñez, and S. Kraus. Security in multiagent systems by policy randomization. In *Proc. AAMAS'06*, 2006. (to appear).
[3] A. M. Spence. *Market Signaling*. Harvard Univ. Press, 1974.
[4] J. Silverman. Hidden agendas and how to uncover them. *Medicine*, 33(2):27–29, 2005.
[5] A. J. Barsky IIIrd. Hidden reasons some patients visit doctors. *Ann. Intern Med.*, 94:492–498, 1981.
[6] Aeschylus. Agamemnon. In A. Y. Campbell, editor, *The Agamemnon of Aeschylus a revised text with brief critical notes*, Liverpool, 1936. University Press of Liverpool.

[7] Aeschylus. Agamemnon. In H. W. Smyth and Lloyd-Jones H, editors, *Aeschylus II*, volume 146 of *Loeb Classical Library*, Cambridge, 1926. Harvard University Press.
[8] H. G. Liddell and R. Scott. *A Greek-English lexicon*. Clarendon Press, Oxford, 1890.

Appendix – Clytemnestra's Speech ll. 930–944

930	εστιν θαλασσα, τις δε νιν κατασβεσει;	There is the sea – and who could drain its depths? –
931	τρεφουσα πολλης πορφυρας ισαργυρον	profuse with purple, costly as silver,
932	κηκιδα παγκαινιστον, ειματων βαφας.	the ever replenished dye for fine fabrics.
933	οικος δ' υπαρχει τωνδε συν θεοις αναξ	Of these our Heaven-favoured house with ample store
934	εχειν· πενεσθαι δ' ουκ επισταται δομος.	is filled enow: poverty reigns not over this realm.
935	πολλων πατησμον δ' ειματων αν ευξαμην,	Why many more would I have strewn to tread upon
936	δομοισι προυνεχθεντος εν χρηστηριοις,	had augury ordained so much be offered
937	ψυχης κομιστρα τησδε μηχανωμενης.	as due ransom for sure passage of one life.
938	ριζης γαρ ουσης φυλλας ικετ' ες δομους,	For, the root still living, leaf spreads once more
939	σκιαν υπερτεινασα σειριου κυνος.	and shade shall cool the dog star's searing heat.
940	και σου μολοντος δωματιτιν εστιαν,	Now returnéd to your hearth and home,
941	θαλπος μεν εν χειμωνι σημαινεις μολον·	as Spring may temper Winter's chill,
942	οταν δε τευχηι Ζευς απ' ομφακος πικρας	or Zeus transform the bitter grape
943	οινον, τοτ' ηδη ψυχος εν δομοις πελει,	to wine, so too fresh spirit fills these halls
944	ανδρος τελειου δωμ' επιστρωφωμενου.	as their rightful lord is rightly received.

Commentary – relationship of the text to P1–P9

930–2 The *non sequitur* "unlimited supply justifies waste" (P1), but also subtle use of ambiguity (P5). Hence, "the sea" is a standard metaphor for humanity in general and, in this case, the turbulent history of Agamemnon's family in particular; "purple" is the colour symbolising blood in classical literature. The subtext of these lines recalls this violent history and its continuation ("ever replenished"): in this setting the phrase "dye for fine fabrics" presents a vivid image that leaves no doubt as to Clytemnestra's intentions.

933–4 (P1) again: "wealth justifies waste".

935–7 (P1), "if (X then Y) therefore Z", but X, i.e. augury and its outcome, is unsubstantiated. Also potentially (P6) and (P8) ("many more"), however, the important sign is (P5) firstly via the subtext of Agamemnon's attitude to oracles but also in the two readings of "sure passage of one life": a safe return from Troy and the "passage" from life to the underworld.

938–9 Elements of (P3), (P4), a suggestion of (P7), primarily (P9) (and, again, (P5): "root" and "leaf" being metaphors for the past history and the cycle of revenge it leads to; similarly with the possible readings of "shade shall cool").

940–4 (P9) in the use of two obscure similes; (P7) with "rightful lord" and "fresh spirit". Again, (P5) with the subtext of vengeance indicated in "rightly received".

Computational Models of Argument
P.E. Dunne and T.J.M. Bench-Capon (Eds.)
IOS Press, 2006

Author Index

This page intentionally left blank