Fifth International Joint Conference
on
Autonomous Agents and Multi-Agent Systems

3rd Workshop
on
Argumentation in Multi-Agent Systems

ArgMAS 2006

Future University, Hakodate, Japan

8 May 2006

Nicolas Maudet, Simon Parsons, Iyad Rahwan

Preface

Argumentation can be abstractly defined as the interaction of different arguments for and against some conclusion. Over the last few years, argumentation has been gaining increasing importance in multi-agent systems, mainly as a vehicle for facilitating "rational interaction" (i.e., interaction which involves the giving and receiving of reasons). This is because argumentation provides tools for designing, implementing and analysing sophisticated forms of interaction among rational agents. Argumentation has made solid contributions to the practice of multi-agent dialogues. Application domains include: legal disputes, business negotiation, labor disputes, team formation, scientific inquiry, deliberative democracy, ontology reconciliation, risk analysis, scheduling, and logistics. A single agent may also use argumentation techniques to perform its individual reasoning because it needs to make decisions under complex preferences policies, in a highly dynamic environment. The workshop is concerned with the use of the concepts, theories, methodologies, and computational models of argumentation in building autonomous agents and multi-agent systems.

Organizing Committee for 2006

- Nicolas Maudet
- Simon Parsons
- Iyad Rahwan

Steering Committee

- Antonis Kakas (University of Cyprus, Cyprus)
- Nicolas Maudet (University of Paris-Dauphine, France)
- Peter McBurney (University of Liverpool, UK)
- Pavlos Moraitis (University R. Descartes-Paris 5, France)
- Simon Parsons (City University of New York, USA)
- Iyad Rahwan (British University in Dubai, UAE / University of Edinburgh, UK)
- Chris Reed (University of Dundee, UK)

Program Committee

- Leila Amgoud (IRIT, Toulouse, France)
- Katie Atkinson (University of Liverpool, UK)
- Jamal Bentahar (Laval University, Canada)
- Carlos Chesnevar (Universitat de Lleida, Spain)
- Frank Dignum (Utrecht University, Netherlands)
- Rogier van Eijk (Utrecht University, Netherlands)
- Anthony Hunter (University College, London, UK)
- Antonis Kakas (University of Cyprus, Cyprus)
- Nikos Karacapilidis (University of Patras, Greece)
- Nicolas Maudet (University of Paris-Dauphine, France)
- Peter McBurney (University of Liverpool, UK)
- Jarred McGinnis (University of Edinburgh, UK)
- Pavlos Moraitis (University R. Descartes-Paris 5, France)
- Xavier Parent (King's College, UK)
- Simon Parsons (City University of New York, USA)
- Philippe Pasquier (University of Melbourne, Australia)
- Henry Prakken (Utrecht University / University of Groningen, The Netherlands)
- Iyad Rahwan (British University in Dubai, UAE / University of Edinburgh, UK)
- Chris Reed (University of Dundee, UK)
- Carles Sierra (IIIA, Spain)
- Guillermo Simari (Universidad Nacional del Sur, Argentina)
- Katia Sycara (Carnegie Mellon University, USA)
- Francesca Toni (Imperial College, London, UK)
- Paolo Torroni (Università di Bologna, Italy)
- Bart Verheij (University of Groningen, The Netherlands)
- Gerard Vreeswijk (Utrecht University, The Netherlands)
- Mike Wooldridge (University of Liverpool, UK)

External Reviewer

- P. M. Dung (Asian Institute of Technology, Bangkok, Thailand)

Schedule

09:00am–10:30am **FOUNDATIONS**

- A Generalization of Dung's Abstract Framework for Argumentation: Arguing with Sets of Attacking Arguments
  (*S. H. Nielsen and S. Parsons)*
- Towards an Argument Interchange Format for Multiagent Systems
  (*S. Willmott, G. Vreeswijk, C. Chesnevar, M. South, J. McGinnis, S. Modgil, I. Rahwan, C. Reed, and G. Simari)*
  [PANEL] 'Argument Interchange Format'

10:30am–11:00am COFFEE

11:00am–12:30pm **EXPLORATIONS**

- Managing Social Influences Through Argumentation-Based Negotiation
  (*N. C. Karunatillake, N. R. Jennings, I. Rahwan and S. D. Ramchurn*)
- Argumentation-based Learning
  (T. Fukumoto and H. Sawamura)
- Arguments and Counterexamples in Case-based Deliberation
  (*S. Ontanon and E. Plaza*)

12:30pm–02:00pm LUNCH

02:00pm–03:30pm **STRATEGIC ASPECTS**

- Argumentation and Persuasion in the Cognitive Coherence Theory
  (*P. Pasquier, I. Rahwan, F. Dignum, and L. Sonenberg*)
- An Argumentation-Based Approach for Dialogue Move Selection
  (*L. Amgoud and N. Hameurlain*)
- Lose Lips Sink Ships, a Heuristic for Argumentation
  (*N. Oren, T. J. Norman, and A. Preece*)

03:30pm–04:00pm COFFEE

04:00pm–05:30pm **STRATEGIC ASPECTS (CONT.)**

- Strategic and Tactic Reasoning for Communicating Agents
  (*J. Bentahar, M. Mbarki, and B. Moulin*)
  [**POSITION PAPER]** A Framework for Learning Argumentation Strategies
  (*C. D. Emele, F. Guerin, T. J. Norman, and P. Edwards*)
  [PANEL] 'Strategies in Argumentation and Dialogue'

Table of Contents

A generalization of Dung's Abstract Framework for Argumentation

Arguing with Sets of Attacking Arguments

Søren Holbech Nielsen[1] and Simon Parsons[2]

[1] Department of Computer Science
Aalborg University, Aalborg
Denmark
holbech@cs.aau.dk
[2] Department of Computer and Information Science
Brooklyn College, City University of New York
Brooklyn, 11210 NY, USA
parsons@sci.brooklyn.cuny.edu

Abstract. One of the most widely studied systems of argumentation is the one described by Dung in a paper from 1995. Unfortunately, this framework does not allow for joint attacks on arguments, which we argue must be required of any truly abstract argumentation framework. A few frameworks can be said to allow for such interactions among arguments, but for various reasons we believe that these are inadequate for modelling argumentation systems with joint attacks. In this paper we propose a generalization of the framework of Dung, which allows for sets of arguments to attack other arguments. We extend the semantics associated with the original framework to this generalization, and prove that all results in the paper by Dung have an equivalent in this more abstract framework.

1 Introduction

In the last fifteen years or so, there has been much interest in argumentation systems within the artificial intelligence community[3]. This interest spreads across many different sub-areas of artificial intelligence. One of these is non-monotonic reasoning [10, 19], which exploits the fact that argumentation systems can handle, and resolve, inconsistencies [12, 13] and uses it to develop general descriptions of non-monotonic reasoning [8, 18]. This line of work is summarised in [28]. Another area that makes use of argumentation is reasoning and decision making under uncertainty [5, 16, 17], which exploits the dependency structure one can infer from arguments in order to correctly combine evidence. Much of this work is covered in [9]. More recently [23, 26], the multi-agent systems community has begun to make use of argumentation, using it to develop a notion of rational interaction [4, 20].

One very influential system of argumentation was that introduced by Dung [11]. This was, for instance, the basis for the work in [8], was the system extended by Amgoud in [1, 2], and subsequently as the basis for the dialogue systems in [3, 24]. In [11],

[3] There were AI researchers who were interested in argumentation before this, for example [6, 7, 14, 21], but this interest was very localized.

Dung presents a very abstract framework for argumentation and a series of semantics for this framework. He goes on to prove a series of relationships between his framework and different varieties of formal logics, including a proof that logic programming can be seen as a special case of his framework. As a last result of the paper he provides a method for encoding systems of the argumentation framework as logic programs. 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 though Dung tried to abstract away from the underlying language and structure of arguments, he did not succeed in doing so completely. In fact if his framework is expected to be able to model all possible kinds of attack, there is an implicit assumption that the underlying language contains a logical "and" connective. This hidden assumption arises from that fact that Dung's attack relation is a simple binary relation from one argument to another, rather than a relation mapping sets of arguments to other sets of arguments.

While not explicitly analyzing the fundamental problem of Dung's framework, some previous works, most notably the efforts of Verheij, have allowed for sets of attacking arguments, although mostly as side effects. We do not find these solutions fully satisfying, and none of them can be said to be conservative generalizations of the framework of [11], that is a generalization that makes the minimum changes to the Dung framework necessary to allow it to handle sets of attacking arguments. We elaborate further on this throughout the paper.

In this paper we analyze Dung's framework, and point out the hidden assumption on the underlying language. We present a generalization of Dung's framework, keeping as close to his ideas as possible, which frees the underlying language from being closed under some logical "and" connective. We do this by allowing sets of arguments to attack single arguments, and provide new definitions and proofs mirroring Dung's results for this more general framework. We also argue why allowing sets of arguments to attack other sets of arguments does not provide further flexibility, and provide an automated encoding of systems of the new framework in Prolog, mirroring Dung's encoding of his systems as logic programs.

The paper is organized as follows: In Sect. 2 we present the essentials of Dung's framework, and then through examples illustrate how a more general attack relation is needed for a truly abstract framework. Then, in Sect. 3 we present our generalization of Dung's framework, complete with definitions, proofs, and a Prolog encoding method. Following this, in Sect. 4, we review other works on argumentation systems where sets of arguments can attack other arguments, and relate them to the approach presented in this paper. Finally, we conclude on the work presented here. Throughout the paper we use the term argumentation *system*, where [11] uses argumentation *framework*, to denote the actual mathematical structures we work with. The term *framework* we reserve

for denoting the overall approaches to describing and reasoning about the argumentation systems, such as the one represented by [11] and the ones reviewed in Sect. 4.

2 Dung's Framework

Dung [11] defines an *argumentation system* as a pair $(\boldsymbol{A}, \triangleright)$, where $\boldsymbol{A}$ is a set of *arguments*, and $\triangleright \subseteq \boldsymbol{A} \times \boldsymbol{A}$ is an *attack relation*. If for two arguments A and B we have $A \triangleright B$, then we say that A *attacks* B, and that B *is attacked by* A. As examples, we might consider the following as arguments:

E_1 "Joe does not like Jack",
E_2 "There is a nail in Jack's antique coffee table",
E_3 "Joe hammered a nail into Jack's antique coffee table",
E_4 "Joe plays golf, so Joe has full use of his arms", and
E_5 "Joe has no arms, so Joe cannot use a hammer, so Joe did not hammer a nail into Jack's antique coffee table".

As can be seen it is not required of an argument that it follows the "if X then conclude Y" pattern for reasoning, or, for that matter, that it represents sound reasoning.

As examples of attacks, we could have that $E_5 \triangleright E_3$, $E_3 \triangleright E_5$, and $E_4 \triangleright E_5$. Intuitively, and in any common-sense argumentation system, we would expect that $A \triangleright B$ if the validity of the argument A is somehow obstructing B from being valid.

Without loss of generality, we will assume that the arguments are members of some underlying language $\boldsymbol{L}$. This assumption is necessary if any kind of meaning is to be extracted from an argumentation system. For instance, in our example, $\boldsymbol{L}$ would necessarily include the strings represented by E_1 to E_5.

It seems reasonable that sometimes a number of arguments can interact and constitute a stronger attack on one or more of the other arguments. For instance, the two arguments E_1 and E_2 would jointly (but not separately) provide a case for the conclusion that Joe has struck a nail into Jack's antique coffee table, and thus provide a joint attack on argument E_5, which has the opposite conclusion. The principle of synergy among arguments is not new, and has previously been debated in connection to "accrual of arguments" (see e.g. [25, 27, 29]). The difference between that discussion and the issue addressed here is we (and Dung) do not consider arguments as having a numerical strength, and a set of defeated arguments thus cannot accrue to become undefeated, unless that set is explictly specified to defeat each argument defeating its individual members.

Going back to the example, if this synergy is to be modelled under Dung's limitations, somehow there must be a new argument:

E_6: "Joe does not like Jack *and* there is a nail in Jack's antique coffee table",

which attacks E_5. If this is taken to be a general solution, it is obviously required that the underlying language $\boldsymbol{L}$ is closed under some "and"-connective.

Furthermore, what we meant to state was that E_1 and E_2 jointly attacked E_5 and the solution does not quite suffice: It may turn out that $\triangleright$ is defined in such a manner that one (or both) of E_1 and E_2 is attacked by another valid argument, while E_6 is

not. That would mean that "Joe does not like Jack *and* there is a nail in Jack's coffee table" is a valid argument, whereas, say, "Joe does not like Jack" is not. Clearly this is nonsense, and in order to ensure that nonsense conclusions cannot arise, $\triangleright$ would have to be restricted accordingly. This muddles the clear distinction between arguments and attacks, which was the very appeal of Dung's framework.

These underlying consistency relations between arguments would seemingly be good candidates for encoding in a logical language (for example $E_1 \wedge E_2 \Rightarrow E_6$ and $E_6 \Rightarrow E_1$), and in fact an underlying logical language employing standard negation could be used to model sets of attacking arguments (i.e. $E_1 \wedge E_2 \Rightarrow \neg\texttt{conclusion}(E_5)$ with attack relations $\neg\texttt{conclusion}(A) \triangleright A$ for all arguments A), but we chose not to go this route for a number of reasons. Primarily, it adds a another level of interdependencies between arguments, which 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. Moreover, examples of joint undercutting attacks seems to be inherently argumentative in nature, and only obscurely encoded in an implicative manner. Consider for instance the following arguments:

F_1 "The Bible says that God is all good, so God is all good",
F_2 "The Bible was written by human beings", and
F_3 "Humans beings are not infallible".

F_2 and F_3 attacks the validity of F_1, but clearly it makes no sense to encode this as $F_2 \wedge F_3 \Rightarrow \neg\texttt{conclusion}(F_1)$ as the facts that human beings are not to be considered infallible and that some of them just happened to write the Bible, do not entail that God is not all good. To capture the intended meaning of the attack, one would have to add an explicit presumption, like "The Bible can be trusted on all matters" to F_1, and allow for such assumptions to be targets of attacks, which — besides requiring identification of all such implicit assumptions — can hardly be said to be as elegant as allowing attacks at the argumentative level[4]

Having argued for the necessity of allowing a set of arguments to attack another argument, we now examine settings where an entire set of arguments is attacked by either a single argument or another set of arguments. Without loss of generality, we assume that what is needed is an attack

$$\{A_1, \ldots, A_n\} \triangleright \{B_1, \ldots, B_m\} ,$$

such that the validity of all the A-arguments prevents the B-arguments from being valid. There are two distinct manners in which this can be interpreted:

1. Either the validity of the A-arguments means that each B_i cannot be valid, no matter the validity of the other B-arguments, or
2. the validity of the A-arguments mean that not all of the B-arguments can be valid at the same time.

[4] Those swayed more by practical considerations than examples should note that the original motivation for this work was to allow arguments about Bayesian networks, in which sets of attacking arguments very naturally occur.

[29] refers to these as "collective" and "indeterministic defeat", respectively — a terminology we adopt in this text.

As an example consider the following twist on the story about Jack, Joe, and the antique coffee table:

E_7 "Jack has been telling lies about Joe to Jill"
E_8 "Jack is a rabbit"
E_9 "Joe loves all animals"

If E_8 is a valid argument, then none of the arguments in the set $\{E_3, E_7\}$ can be valid: E_3 because rabbits do not own antique coffee tables, and E_7 because rabbits, being unable to speak, do not lie. This is thus an example of collective defeat. As an example of indeterministic defeat, E_9 attacks the set of arguments $\{E_1, E_8\}$ seen as a set: E_1 and E_8 cannot both be valid arguments if Joe loves all animals. However, both E_1 and E_8 can be valid seen as individual arguments, no matter how Joe feels about animals.

We claim that it is never necessary to specify a non-singleton set of arguments as attacked, as in $\{A_1, \ldots, A_n\} \triangleright \{B_1, \ldots, B_m\}$: If collective defeat is taken to heart, the attack can be reformulated as a series of attacks

$$\begin{array}{c} \{A_1, \ldots, A_n\} \triangleright B_1 \\ \vdots \\ \{A_1, \ldots, A_n\} \triangleright B_m \end{array} .$$

It is easily seen that the above attacks would imply the attack, which is intended, as the validity of the A-arguments would ensure that none of the B-arguments are valid.

If instead indeterministic defeat is required, the attack can be reformulated as

$$\{A_1, \ldots, A_n, B_2, \ldots, B_m\} \triangleright B_1 \ ,$$

which ensures that in case the A-arguments are valid, then B_1 cannot be a valid argument if the remaining B-arguments are also true, thus preventing the entire set of B-arguments from being valid at once, if the A-arguments are true. In the example above, we would state that $\{E_8, E_1\}$ attacks E_9. Notice that this "trick" is not dependent on the actual structure or language of the arguments, nor require the introduction of a new dummy argument, as was the case if only single arguments were allowed as attackers.

In conclusion, we have argued for the insufficiency of Dung's treatment, when sets of arguments are taken into account, and that an attack relation that allows for sets of arguments attacking single arguments is sufficient to capture any kind of relation between sets of arguments.

3 Argumentation with Attacking Sets of Arguments

In this section we present our generalization of the framework of [11]. In an effort to ease comparison, we have labelled definitions, lemmas, and theorems with the same numbers as their counterparts in [11], even if this means that there are holes in the

numbering (e.g. there is no Lemma 2). Furthermore, we have omitted proofs where the original proofs of [11] suffice. As a result of the tight integration with [11] most definitions and results have been worded in a nearly identical manner, even if the proofs are different and the meaning of individual words are different. Those definitions and results that differ essentially from their counterparts in [11], or which is entirely new, have been marked with an asterix (*). The rest are identical to those in [11].

Throughout the presentation, it should be clear that the framework presented here reduces to that of [11] if only singleton sets are allowed as attackers.

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

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 exists a set that attacks an argument A, then there must also exist a minimal attack on A. If for two sets of arguments $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$, there is an argument A in $\boldsymbol{S}_2$ that 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 $\boldsymbol{S}_1$.

Definition 2 (Conflict-free Sets*). *A set of arguments* $\boldsymbol{S}$*, is said to be* conflict-free *if it does not attack itself, i.e. there is no argument* $A \in \boldsymbol{S}$*, such that* $\boldsymbol{S}$ *attacks* A*.*

Let $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ be sets of arguments. If $\boldsymbol{S}_2$ attacks an 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 $\boldsymbol{S}_2$.

Definition 3 (Acceptable and Admissable Arguments*). *An argument* A *is said to be* acceptable *with respect to a set of arguments* $\boldsymbol{S}$*, if* $\boldsymbol{S}$ *defends* A *from all attacking sets of arguments in* $\boldsymbol{A}$*.*

A conflict-free set of arguments $\boldsymbol{S}$ *is said to be* admissible *if each argument in* $\boldsymbol{S}$ *is acceptable with respect to* $\boldsymbol{S}$*.*

Intuitively, an argument A is acceptable with respect to some set $\boldsymbol{S}$, if anyone believing in the validity of the arguments in $\boldsymbol{S}$ can defend A against all attacks. If a set of arguments is admissible, it means that anyone believing this set of arguments as valid is not contradicting himself and can defend his beliefs against all attacks.

Definition 4. *An admissible set* $\boldsymbol{S}$ *is called a* preferred extension *if there is no admissible set* $\boldsymbol{S}' \subseteq \boldsymbol{A}$*, such that* $\boldsymbol{S} \subsetneq \boldsymbol{S}'$*.*

Building on the intuition from before, taking on a preferred extension as your beliefs thus means that you would not be able to defend any more arguments without contradicting yourself.

Lemma 1 (Fundamental Lemma). *Let* $\boldsymbol{S}$ *be an admissible set, and* A *and* A' *be arguments that each are acceptable with respect to* $\boldsymbol{S}$*, then*

1. $\boldsymbol{S}' = \boldsymbol{S} \cup \{A\}$ *is admissible, and*
2. A' *is acceptable with respect to* $\boldsymbol{S}'$*.*

Proof. 1) As $\boldsymbol{S}$ is admissible, and A is acceptable with respect to $\boldsymbol{S}$, it is obvious that $\boldsymbol{S}$, and therefore also $\boldsymbol{S}'$, defends each argument in $\boldsymbol{S}'$. Thus we only need to prove that $\boldsymbol{S}'$ is conflict-free. Assume not. Then there is an argument $B \in \boldsymbol{S}'$ and an attack $\boldsymbol{S}'' \subseteq \boldsymbol{S}'$ on B. Since each argument in $\boldsymbol{S}'$ is defended by $\boldsymbol{S}$ it follows that $\boldsymbol{S}$ attacks $\boldsymbol{S}''$.

As $\boldsymbol{S}$ attacks $\boldsymbol{S}''$ it follows that $\boldsymbol{S}$ must attack at least one argument of $\boldsymbol{S}''$. Let C be this argument. We consider two cases: First $C \equiv A$ and second $C \not\equiv A$. If $C \equiv A$ then it follows that $\boldsymbol{S}$ attacks A. As A is acceptable with respect to $\boldsymbol{S}$, $\boldsymbol{S}$ must then necessarily attack $\boldsymbol{S}$, which contradicts the assumption that $\boldsymbol{S}$ is conflict-free. Assume then that $C \not\equiv A$. Then C must be part of $\boldsymbol{S}$, and consequently $\boldsymbol{S}$ attacks $\boldsymbol{S}$ yielding the same contradiction with the assumptions.

2) Obvious. □

Using the Fundamental Lemma the following important result, guaranteeing that an admissible set can be extended to a preferred extension, can be proven.

Theorem 1. *For any argumentation system the set of admissible sets forms a complete partial order with respect to set inclusion, and for each admissible set $\boldsymbol{S}$ there exists a preferred extension $\boldsymbol{S}'$, such that $\boldsymbol{S} \subseteq \boldsymbol{S}'$.*

As the empty set is an admissible set, we have:

Corollary 2. *Every argumentation system has at least one preferred extension.*

A more aggressive semantics is the stable semantics:

Definition 5 (Stable Semantics). *A conflict free set $\boldsymbol{S}$ is a* stable extension *if $\boldsymbol{S}$ attacks all arguments in $\boldsymbol{A} \setminus \boldsymbol{S}$.*

Lemma 3. *$\boldsymbol{S}$ is a stable extension iff $\boldsymbol{S} = \{A \mid A \text{ is not attacked by } \boldsymbol{S}\}$.*

Proof. "only if": Obvious.

"if": Assume not. Then $\boldsymbol{S}$ is either not conflict-free, or there is an argument in $\boldsymbol{A} \setminus \boldsymbol{S}$ not attacked by $\boldsymbol{S}$. The latter possibility is precluded by the definition of $\boldsymbol{S}$, so there must be a set $\boldsymbol{S}' \subseteq \boldsymbol{S}$ and an argument $A \in \boldsymbol{S}$ such that $\boldsymbol{S}'$ attacks A. But then $\boldsymbol{S}$ also attacks A, which contradicts the definition of $\boldsymbol{S}$. □

The general connection between stable and preferred semantics is given by the following result:

Lemma 4. *Every stable extension is a preferred extension, but not vice versa.*

Both preferred and stable semantics are credulous in the sense that they represent beliefs that include as much as possible. Next, we consider semantics corresponding to more skeptical points of views. For this we need the notion of a characteristic function, and some general results on this:

Definition 6 (Characteristic Function). *The* characteristic function *of an argumentation system is the function $F : \mathcal{P}(\boldsymbol{A}) \to \mathcal{P}(\boldsymbol{A})$ defined as*

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

Next, we state a couple of properties of the characteristic function F. The first result is not explicitly stated in [11], but included only as part of a proof. We make it explicit here as it is a property required of F by some proofs that have been left out.

Proposition 1 (*) *If $\boldsymbol{S}$ is a conflict-free set, then $F(\boldsymbol{S})$ is also conflict-free.*

Proof. Assume this is not the case, then there is $\boldsymbol{S'} \subseteq F(\boldsymbol{S})$ and $A \in F(\boldsymbol{S})$ such that $\boldsymbol{S'}$ attacks A. Since A is acceptable wrt. $\boldsymbol{S}$, $\boldsymbol{S}$ must attack at least one element B of $\boldsymbol{S'}$. But since B is in $F(\boldsymbol{S})$ it must be acceptable wrt. $\boldsymbol{S}$, and $\boldsymbol{S}$ must consequently attack itself. This contradicts the assumption that $\boldsymbol{S}$ is a conflict-free set. □

Lemma 5. *A conflict-free set $\boldsymbol{S}$ is admissible iff $\boldsymbol{S} \subseteq F(\boldsymbol{S})$.*

Proof. "only if": All arguments of $\boldsymbol{S}$ are acceptable wrt. $\boldsymbol{S}$, so $\boldsymbol{S} \subseteq F(\boldsymbol{S})$.
"if": As $\boldsymbol{S} \subseteq F(\boldsymbol{S})$ it follows that all arguments of $\boldsymbol{S}$ are acceptable wrt. $\boldsymbol{S}$. □

Lemma 6. *F is a monotonic function with respect to set inclusion.*

Proof. Follows since adding arguments to a set of arguments cannot cause the set to attack fewer arguments, and consequently cannot change the status of any of the arguments currently defended into being not defended. □

Now, we can introduce the most skeptical semantics possible:

Definition 7 (Grounded Extension). *The* grounded extension *of an argumentation system, is the least fix-point of the corresponding characteristic function.*

A grounded extension is thus the set of arguments that are not challenged by any other arguments, along with the arguments defended by these arguments, those defended by those, and so on. [11] does not prove that the grounded extension of an argumentation system is well-defined, but we include a proof here.

Proposition 2 (*) *If $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are both grounded extensions of an argumentation system, then $\boldsymbol{G}_1 = \boldsymbol{G}_2$.*

Proof. Assume not, and let $\boldsymbol{C} = \boldsymbol{G}_1 \cap \boldsymbol{G}_2$. As $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ are different and also minimal, it follows that none of them can be the empty set, and hence that $F(\emptyset) \neq \emptyset$. As $F(\emptyset)$ consists of the arguments that are not attacked by any arguments at all, it follows that these are acceptable wrt. any set. In particular, $F(\emptyset)$ must be a subset of both $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$, so $\boldsymbol{C}$ is non-empty. Furthermore, as Lmm. 6 assures that F is monotonic, it follows that $F(\boldsymbol{C})$ must be a subset of both $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$. But then $F(\boldsymbol{C})$ must be equal to $\boldsymbol{C}$, and is thus a fix point of F. As both $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$ were supposed to be minimal and different, this yields the desired contradiction. □

As a common class, encompassing all the semantics we have discussed so far, we introduce complete extensions:

Definition 8 (Complete Extension). *An admissible set $\boldsymbol{S}$ is called a* complete extension*, if all arguments that are acceptable with respect to $\boldsymbol{S}$ are in $\boldsymbol{S}$.*

A couple of results tie the complete extension semantics to the other semantics we have discussed:

Lemma 7. *A conflict-free set $\boldsymbol{S}$ is a complete extension iff $\boldsymbol{S} = F(\boldsymbol{S})$.*

Theorem 2. *Extensions are such that:*

1. *Each preferred extension is a complete extension, but not vice versa.*
2. *The grounded extension is the least complete extension with respect to set inclusion.*
3. *The complete extensions form a complete semi-lattice with respect to set inclusion.*

Next, we investigate classifying argumentation systems according to desirable properties of their corresponding semantics.

Definition 9 (Finitary System*). *An argumentation system is said to be* finitary *if for each argument A, there is at most a finite amount of minimal attacks on A, and each minimal attack is by a finite set of arguments.*

Lemma 8. *For any finitary system, F is ω-continuous.*

Proof. Let $\boldsymbol{S}_1 \subseteq \boldsymbol{S}_2 \subseteq \cdots$ be an increasing series of sets of arguments, and $\boldsymbol{S} = \cup_i \boldsymbol{S}_i$. We need to show that $F(\boldsymbol{S}) = \cup_i F(\boldsymbol{S}_i)$. As adding arguments to a set cannot reduce the set of arguments attacked by this set, and therefore cannot reduce the set of arguments that are acceptable with respect to it, we have that $F(\boldsymbol{S}_i) \subseteq F(\boldsymbol{S})$ for each i, and thus $F(\boldsymbol{S}) \supseteq \cup_i F(\boldsymbol{S}_i)$.

To see that $F(\boldsymbol{S}) \subseteq \cup_i F(\boldsymbol{S}_i)$, consider an argument $A \in F(\boldsymbol{S})$, and let $\boldsymbol{T}_1, \ldots, \boldsymbol{T}_n$ be the finitely many minimal attacks on A. As $\boldsymbol{S}$ attacks each attack on A, there must be an argument B_i in each $\boldsymbol{T}_i$, which is attacked by $\boldsymbol{S}$. Let $\boldsymbol{U}_i \subseteq \boldsymbol{S}$ be the minimal attack of B_i. As each minimal attack consists of a finite number of arguments, the set $\boldsymbol{U} = \boldsymbol{U}_1 \cup \cdots \cup \boldsymbol{U}_n$ is finite as well, and thus there must be a j, such that $\boldsymbol{U} \subseteq \boldsymbol{S}_j$. Consequently, A must be in $F(\boldsymbol{S}_j)$ and therefore also in $\cup_i F(\boldsymbol{S}_i)$. □

Definition 10 (Well-founded System*). *An argumentation system is* well-founded*, if there exists no infinite sequence of sets $\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots$, such that $\boldsymbol{S}_i$ is a minimal attack on an argument in $\boldsymbol{S}_{i-1}$ for all i.*

Theorem 3. *Every well-founded argumentation system has exactly one complete extension, which is grounded, preferred, and stable.*

Proof. It suffices to prove that the grounded extension $\boldsymbol{G}$ is stable. Assume this is not the case, and let $\boldsymbol{S} = \{A \mid A \notin \boldsymbol{G}$ and A is not attacked by $\boldsymbol{G}\}$, which must be nonempty if the grounded extension is not stable. We prove that each argument A in $\boldsymbol{S}$ is attacked by a minimal set $\boldsymbol{S}'$ such that $\boldsymbol{S} \cap \boldsymbol{S}' \neq \emptyset$, and therefore that the system cannot be well-founded.

Since A is not in $\boldsymbol{G}$ it is not acceptable with respect to $\boldsymbol{G}$. Therefore there must be a minimal attack $\boldsymbol{T}$ of A, not itself attacked by $\boldsymbol{G}$. Since $\boldsymbol{G}$ does not attack A, at least one element of $\boldsymbol{T}$ must be outside of $\boldsymbol{G}$. Let $\boldsymbol{T}'$ be $\boldsymbol{T} \setminus \boldsymbol{G}$, which is thus non-empty. As $\boldsymbol{G}$ does not attack $\boldsymbol{T}$, it furthermore follows that $\boldsymbol{T}'$ must be a subset of $\boldsymbol{S}$. Thus, $\boldsymbol{T}$ is the set $\boldsymbol{S}'$ we were looking for, and the proof is complete. □

Definition 11 (Coherent and Relatively Grounded System). *An argumentation system is* coherent *if all its preferred extensions are stable. A system is* relatively grounded *if its grounded extension is the intersection of all its preferred extensions.*

Let $A_1, A_2, \ldots$ be a (possible finite) sequence of arguments, where each argument A_i is part of a minimal attack on A_{i-1}. Then the arguments $\{A_{2i}\}_{i\geq 1}$ are said to *indirectly attack* A_1. The arguments $\{A_{2i-1}\}_{i\geq 1}$ are said to *indirectly defend* A_1. If an argument A is both indirectly attacking and defending an argument B, then A is said to be *controversial with respect to* B, or simply *controversial*.

Definition 12 (Uncontrovertial and Limited Controversial System). *An argumentation system is* uncontroversial *if none of its arguments are controversial. An argumentation system, for which there exists no infinite sequence of arguments $A_1, A_2, \ldots$, such that for all i, A_i is controversial with respect to A_{i-1}, is said to be* limited controversial.

Obviously, a controversial argumentation system is also limited controversial.

Lemma 9. *In every limited controversial argumentation system there exists a nonempty complete extension.*

Proof. We construct the nonempty complete extension $\boldsymbol{C}$. Since a nonempty grounded extension would suffice, we assume that it is empty. Since the system is limited controversial, every sequence of arguments, where A_i is controversial with respect to A_{i-1}, must have a last element, B. It follows that there is no argument that is controversial with respect to B. We define $\boldsymbol{E}_0$ to be $\{B\}$, and $\boldsymbol{E}_i$ to be $\boldsymbol{E}_{i-1} \cup \boldsymbol{D}_i$, where $\boldsymbol{D}_i$ is a minimal set that defends $\boldsymbol{E}_{i-1}$ from $\boldsymbol{A} \setminus \boldsymbol{E}_{i-1}$, for all $i \geq 1$. As the grounded extension is empty, each argument is attacked by some other argument, and therefore each $\boldsymbol{D}_i$ is guaranteed to exist.

We then prove by induction that, for each $i \geq 0$, $\boldsymbol{E}_i$ is conflict-free and each argument in $\boldsymbol{E}_i$ indirectly defends B.

The hypothesis trivially holds true for $i = 0$. We assume it to be true for $i - 1$ and show that it also must be true for i: From the induction hypothesis we know that $\boldsymbol{E}_{i-1}$ consists of arguments that indirectly defends B. As each argument in $\boldsymbol{D}_i$ participates in attacking an argument, which participates in an attack on an argument in $\boldsymbol{E}_{i-1}$, each of these must also indirectly defend B, and consequently this is true of all arguments in $\boldsymbol{E}_i$. Assume then that $\boldsymbol{E}_i$ is not conflict-free. Then there is a set of arguments $\boldsymbol{S} \subseteq \boldsymbol{E}_i$, that attack an argument $B \in \boldsymbol{E}_i$. But then the arguments in $\boldsymbol{S}$ are attacking an indirect defender of B, and thus are indirect attackers of B. This mean that the arguments in $\boldsymbol{S}$ are controversial with respect to B, violating the assumptions of the lemma. Thus, the induction hypothesis is proved.

Next, let $\boldsymbol{E} = \cup_i \boldsymbol{E}_i$. We prove that this set is admissible, and then let $\boldsymbol{C}$ be the least complete extension containing $\boldsymbol{E}$. We know such an extension exists as by Thm. 1 a preferred extension containing $\boldsymbol{E}$ must exists, and from Thm. 2 that extension must be a complete extension. To see that $\boldsymbol{E}$ is admissible, first let $C \in \boldsymbol{E}$ be an argument. There must be some i, such that $C \in \boldsymbol{E}_i$, and therefore a defense of C must be in $\boldsymbol{D}_i$, and consequently in $\boldsymbol{E}_{i+1}$. But then that defense is also in $\boldsymbol{E}$, and hence C must be acceptable with respect to $\boldsymbol{E}$. To see that $\boldsymbol{E}$ is conflict-free assume that it contains C and $\boldsymbol{S}$, such that $\boldsymbol{S}$ attacks C. As each argument of $\boldsymbol{S}$ must be an element of some set

$\boldsymbol{E}_i$, it follows that each of these indirectly defend B. But as C also indirectly defends B, each element of $\boldsymbol{S}$ must indirectly attack B also, and is thus controversial with respect to B. But this violates the assumption that no argument is controversial with respect to B, and there can therefore be no such $\boldsymbol{S}$ and C. □

Lemma 10. *For any uncontroversial system, with an argument A that is neither a member of the grounded extension nor attacked by it,*

1. *there exists a complete extension containing A, and*
2. *there exists a complete extension that attacks A.*

Proof. 1) Similar to the proof in [11].

2) Proof by construction. Since A is not part of the grounded extension $\boldsymbol{G}$, nor attacked by it, it is attacked by some minimal set of arguments $\boldsymbol{S}$, such that $\boldsymbol{S} \not\subseteq \boldsymbol{G}$ and $\boldsymbol{G}$ does not attack $\boldsymbol{S}$. As the system is uncontroversial, it is impossible for any member of $\boldsymbol{S}$ to participate in a minimal attack on $\boldsymbol{S}$, so the set $\boldsymbol{S}$ is conflict-free. Following a process similar to the one in the proof of Lmm. 9, substituting $\boldsymbol{S}$ for $\{B\}$, we can build a series of conflict-free sets that consists of arguments that indirectly attack A. Extending the union of these sets to a complete extension provides the sought extension. □

Theorem 4. *Every limited controversial system is coherent, and every uncontroversial system is also relatively grounded.*

Corollary 11. *Every limited controversial argumentation system possesses at least one stable extension.*

This ends our derivation of results mirroring those in [11]. [11] furthermore provides a series of results, showing how some formalisms are special cases of his framework. As Dung's framework itself is a special case of our framework, it follows that these frameworks are also special cases of our framework.

[11] ends with a procedure that turns any finitary argumentation system, as defined in [11], into a logic program, and thereby provides a tractable means for computing grounded extensions of such systems. As our framework is more general, it does not allow for Dung's procedure to be used directly. Instead we provide the following procedure for finitary systems: Given a finitary argumentation system $(\boldsymbol{A}, \triangleright)$, we define a Prolog encoding of this system as the clauses

$$\{\textbf{attacks}([\boldsymbol{S}], A) \leftarrow \ \mid \boldsymbol{S} \triangleright A\} \ ,$$

where $[\boldsymbol{S}]$ is a Prolog list declaration containing the arguments in $\boldsymbol{S}$.

Furthermore, a general interpreter for a Prolog encoding of a finitary argumentation system, is defined as:

$$\begin{aligned}\{&\textbf{acceptable}(X) \leftarrow \neg\textbf{defeated}(X);\\ &\textbf{defeated}(X) \leftarrow \textbf{attacks}(Y, X), \textbf{acc}(Y);\\ &\textbf{acc}(X|Y) \leftarrow \textbf{acceptable}(X), \textbf{acc}(Y);\\ &\textbf{acc}(X) \leftarrow \textbf{acceptable}(X);\} \ .\end{aligned}$$

4 Related Work

While not explicitly analyzing the problem of Dung's framework addressed here, nor trying to generalize it in a conservative manner, some previous works have allowed for sets of attacking arguments, although mainly as side effects. First and foremost, [29, Chapter 5] provided a framework, CumulA, with a very general attack relation, which allows sets of arguments to attack other sets. However, the framework is focused on modelling the actual dialectic process of argumentation, rather than investigating the essentials of justified and acceptable arguments, and perhaps as a consequence of this, the semantics presented by Verheij is neither as clear as Dung's nor does it allow for simple comparisons with other formalisms. Furthermore, there are some flaws in Verheij's treatment, which effectively leave CumulA with no well-founded semantics. Specifically, three requirements on allowed extensions turn out to prevent seemingly sensible systems from being analysed, and the semantics associated with an attack on sets of arguments is context dependent. For more on these problems see [22].

Later, Verheij has developed two additional frameworks that allow for sets of attacking arguments, namely Argue!, described in [30], and the formal logical framework of DefLog, described in [32] and implemented in [31]. Even though these frameworks builds on ideas from CumulA, they avoid the problems associated with that framework. However, the two frameworks have other short-comings that make us prefer a conservative generalization of Dung's framework: Argue! employs only a step-based procedural semantics, and thus lacks the analytical tools, theoretic results, and scope of [11]. DefLog, on the other hand, is well-investigated, but lacks a skeptical semantics, and allows sets of attacking arguments only as a rather contrived encoding. For instance, the attack $\{A, B\} \triangleright C$ would be encoded as

$$A \rightsquigarrow (B \rightsquigarrow \times(C)),$$

where $\rightsquigarrow$ denotes primitive implication, and $\times(\cdot)$ denotes defeat of its argument. There are two two problems with this encoding, one technical and one aesthetic. The first is that systems involving infinite sets of attacking arguments cannot be analysed. The second that the symmetry of the set of attackers is broken. Consider for instance the case where A is "X weighs less than 80 kg", B is "X is taller than 180 cms", and C is "X is obese"; here encoding the fact that A and B together defeat C as "X weighs less than 80 kg" implies that "X is taller than 180 cms, so X is not obese" seems to us to be inelegant, and the larger the set of attackers, the larger the inelegance.

The power of encoding sets of attacking arguments wielded by DefLog is due to its expressive language, which is closed under both an implicative operator and an negative operator. Some other argumentation frameworks that are based on formal languages employing similar operators also have implicit methods for encoding attacks by sets of arguments. Most notable is the framework presented in [33], which allows for any sets of sentences to attack each other by encoding rules that from each of them lead to a contradiction. Undercutting attacks are, however, not expressible without assumptions on the underlying language. [8] and [15] present frameworks based on similar ideas. However, all of these fail to abstract from the structure of arguments and as a result do not clearly distinguish between arguments and their interactions, unlike the frameworks

of [11] and this paper. Moreover, the approach of encoding attacks in a logical language restrains sets of attackers to be finite.

5 Conclusions

In this paper we have started exploring formal abstract argumentation systems where synergy can arise between arguments. We believe that we have argued convincingly for the need for such systems, and have examined some of the semantics that can be associated with them. We have tried to do this in the most general fashion possible, by taking outset in the abstract frameworks of [11], and creating a new formalization that allows for sets of arguments to jointly attack other arguments. As we argued in Sect. 2 this degree of freedom ensures that all kinds of attacks between arguments can be modelled faithfully.

Acknowledgments: This work was partially supported by NSF IIS-0329037, and EU PF6-IST 002307 (ASPIC).

References

1. L. Amgoud. *Contribution a l'integration des préferences dans le raisonnement argumentatif*. PhD thesis, Université Paul Sabatier, Toulouse, July 1999.
2. 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.
3. L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In E. Durfee, editor, *Proceedings of the Fourth International Conference on Multi-Agent Systems*, pages 31–38, Boston, MA, USA, 2000. IEEE Press.
4. 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.
5. 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, San Mateo, CA, 1993. Morgan Kaufmann.
6. L. Birnbaum. Argument molecules: a functional representation of argument structure. In *Proceedings of the 2nd National Conference on Artificial Intelligence*, pages 63–65, Los Altos, CA, 1982. William Kaufmann.
7. L. Birnbaum, M. Flowers, and R. McGuire. Towards an AI model of argumentation. In *Proceedings of the 1st National Conference on Artificial Intelligence*, pages 313–315, Los Altos, CA, 1980. William Kaufmann.
8. 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.
9. D. V. Carbogim, D. Robertson, and J. Lee. Argument-based applications to knowledge engineering. *The Knowledge Engineering Review*, 15(2), 2000.
10. 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, San Mateo, CA, 1995. Morgan Kaufmann.

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(2):321–358, 1995.
12. M. Elvang-Gøransson and A. Hunter. Argumentative logics: reasoning with classically inconsistent information. *Data and Knowledge Engineering*, 16:125–145, 1995.
13. 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, San Mateo, CA, 1993. Morgan Kaufmann.
14. M. Flowers, R. McGuire, and L. Birnbaum. Adversary arguments and the logic of personal attacks. In W. G. Lehnert and M. H. Ringle, editors, *Strategies for natural language processing*, pages 275–294. Lawrence Erblaum Associates, Hillsdale, New Jersey, 1982.
15. A. J. Garcia and G. R. Simari. Defeasible logic programming an argumentative approach. *Theory and Practice of Logic Programming*, 4(1):95–138, 2004.
16. J. Kohlas. Symbolic evidence, arguments, supports and valuation networks. In M. Clarke, R. Kruse, and S. Moral, editors, *Symbolic and Quantitative Approaches to Reasoning and Uncertainty*, pages 186–198. Springer Verlag, Berlin, Germany, 1993.
17. P. Krause, S. Ambler, M. Elvang-Gøransson, and J. Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11:113–131, 1995.
18. F. Lin. An argument-based approach to non-monotonic reasoning. *Computational Intelligence*, 9:254–267, 1993.
19. R. Loui. Defeat among arguments: a system of defeasible inference. *Computational Intelligence*, 3:100–106, 1987.
20. P. McBurney. *Rational Interaction*. PhD thesis, Department of Computer Science, University of Liverpool, 2002.
21. R. McGuire, L. Birnbaum, and M. Flowers. Opportunistic processing in arguments. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, pages 58–60, Menlo Park, CA, 1981. American Association for Artificial Intelligence.
22. S. H. Nielsen and S. Parsons. Note on the short-comings of CumulA. `http://www.cs.aau.dk/~holbech/cumulanote.ps`.
23. 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.
24. 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.
25. John L. Pollock. *Cognitive Carpentry: A Blueprint for How to Build a Person*. MIT Press, 1995.
26. H. Prakken. Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese*, 127:187–219, 2001.
27. 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. ACM, 2005.
28. H. Prakken and G. Vreeswijk. Logics for defeasible argumentation. In D. Gabbay, editor, *Handbook of Philosophical Logic*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.
29. B. Verheij. *Rules, Reasons, Arguments. Formal studies of argumentation and defeat*. PhD thesis, Universiteit Maastricht, 1996.
30. B. Verheij. Argue! - an implemented system for computer-mediated defeasible argumentation. In H. La Poutr and H.J. van den Herik, editors, *Proceedings of the Tenth Netherlands/Belgium Conference on Artificial Intelligence*, pages 57–66. CWI, Amsterdam, 1998.
31. B. Verheij. Artificial argument assistants for defeasible argumentation. *Artificial Intelligence*, 150(1–2):291–324, 2003.

32. B. Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic and Computation*, 13(3):319–346, 2003.
33. G. A. W. Vreeswijk. Abstract argumentation systems. *Artificial Intelligence*, 90(1):225–279, 1997.

Towards an Argument Interchange Format for Multi-Agent Systems

Steven Willmott[1], Gerard Vreeswijk[2], Carlos Chesñevar[3], Matthew South[4], Jarred McGinnis[5], Sanjay Modgil[4], Iyad Rahwan[6,5], Chris Reed[7], and Guillermo Simari[8]

[1] Universitat Politècnica de Catalunya, Catalunya, Spain
[2] Universiteit Utrecht, The Netherlands
[3] Universitat de Lleida, Catalunya, Spain
[4] Cancer Research UK, UK
[5] University of Edinburgh, UK
[6] British University in Dubai, UAE
[7] University of Dundee, UK
[8] Universidad Nacional del Sur, Argentina

Joint subimission ArgMAS 2006
See (`http://x-opennet.org/aif`) for previous versions

Abstract. This document describes a strawman specification for an Argument Interchange Format (AIF) that might be used for data exchange between Argumentation tools or communication in Multi-Agent Systems (MAS). The document started life as a skeleton for contributions from participants in the Technical Forum Group meeting in Budapest in September 2005, receiving also input from third parties. The results were subsequentely improved and added to by online discussion to form a more substantial. In its current form, this document is intended to be a strawman model which serves as a point of discussion for the community rather than an attempt at a definitive, all encompassing model. The hope is that it could provide a useful input to ArgMAS discussion in paricular on the utility of common Argumentation Interchange Formats, what form they might take and a potential research / development agenda to help realise them.

1 Introduction and Background

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 [22, page 5]. 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 as outlined in various recent reviews [2, 3, 15, 18]. These applications range from specifying semantics for

logic programs [4], to natural language text generation [5], to supporting legal reasoning [1], to decision-support for multi-party human decision-making [7] and conflict resolution [20].

In recent years, argumentation theory has been gaining increasing interest in the multi-agent systems (MAS) research community [16, 17]. On one hand, argumentation-based techniques can be used to specify *autonomous agent reasoning*, such as belief revision and decision-making under uncertainty and non-standard preference policies. On the other hand, argumentation can also be used as a vehicle for facilitating *multi-agent interaction*, because argumentation naturally provides tools for designing, implementing and analysing sophisticated forms of interaction among rational agents. Argumentation has made solid contributions to the theory and practice of multi-agent dialogues.

While these efforts have made great progress there remain major barriers to the development and practical deployment of Argumentation systems. One of these barriers is the lack of a shared, agreed notation or "Interchange Format" for argumentation and arguments. The potential benefits of such a format include:

- Providing a convergence point for discussing the syntax and semantics of argumentation-related agent interaction.
- Provide a common basis for discussing and comparing Argumentation scenarios.
- Enabling the development of a variety of compatible tools/systems which share the same argumentation input/output formats.
- Facilitating the development of agents capable of interaction via argumentation using a shared formalism.

While argumentation mark-up languages such as Araucaria,[9] Compendium[10] and ASCE[11] (see [9] for example) already exist they are primarily a means to enable user to structure arguments through diagramatic linkage of natural language sentences. These mark-up languages are not designed to process formal logical statements such as those used within multi-agent systems. As a result, the aim of the Argumentation Interchange Format (AIF) workshop hosted in Budpest, Hungary in September 2005 was to sketch out a strawman document that presents an attempt to consolidate, where possible, the work that has already been done in argumentation mark-up languages and multi-agent systems frameworks. It is hoped that this effort will provide a convergence point for theoretical and practical work in this area, and in particular facilitate:

1. Argument interchange between agents within a particular multi-agent framework.
2. Argument interchange between agents across separate multi-agent frameworks.

[9] `http://araucaria.computing.dundee.ac.uk/`
[10] `http://www.compendiuminstitute.org/tools/compendium.htm`
[11] `http://www.adelard.co.uk/software/asce/`

3. Inspection/manipulation of agent arguments through argument visualization tools.
4. Interchange between argumentation visualization tools.

The remainder of this document provides a first-cut model for such a format in order that it might form a discussion point in the community.

2 Overall Approach

An Argumentation Interchange Format, like any other data representation, requires a well defined *syntax* and *semantics*. The syntax is required as a concrete representation of statements relating to arguments, and the semantics conveys the meaning of statements made using the syntax. However, beyond this basic requirement, there are a wide range of approaches which could be taken for defining both syntax and semantics. In particular, semantics may be explicit (using some previous formal notation with its own syntax and semantics) or implicit (hard coded into a piece of software which subsequently behaves in a given way for each combination of inputs), machine readable or targeted at a human audience (written notes for human consumption), formal or informal, etc. Further questions arise as to whether there should be one single AIF format defined, whether variations should be allowed for, how extensions should be dealt with, etc. Given this range of possibilities the approach taken in this document adheres to the following overall principles:

- *Machine readable syntax*: AIF representations are specifically targeted at machine read/write operations rather than human level documentation. While using formats which are human readable is desirable (for example for debugging purposes) the primary aim of the format is data interchange between software systems.
- *Explicit and (where possible) machine processable semantics*: The semantics of AIF statements are to be stated explicitly in specification documents, such that they may be implemented by multiple tool/system providers. Secondly, where possible, the nature of the semantic definition should enable the implementation of processing tools such as reasoners (for example using some existing logical framework).
- *Unified abstract model, multiple reifications*: the AIF should be defined in terms of: 1) An *abstract model* defining the concepts which could be expressed in an AIF and their relationship to one other, and 2) a set of concrete *reifications / concrete syntaxes* which instantiate these concepts in a particular syntactic formalism (such as XML, Lisp-like S-expressions, etc.). Using this even if different computational environments require different styles of Syntax, interoperability may still be facilitated by similarities at the abstract level.
- *Core concepts, multiple extensions*: recognizing that different applications may require statements about a wide array of different argumentation related concepts, the AIF will be structured as a set of core concepts (those likely to

be common to many applications) and extensions (those which are specialist to particular domains or types of applications). It is anticipated that: A) the core will evolve over time as consensus changes on what is central and applications generate experience, and that B) extensions could be generated by any user of the AIF and, if they turn out to be particularly useful, shared amongst large groups of users (potentially also being merged into the core).

3 Abstract Model / Core Ontology

The foundation for the AIF model is a set of definitions for high-level concepts related to argumentation which may need to be represented in the proposed format. These concepts are gathered into three main groups:

1. *Arguments and Argument Networks*: the core ontology for argument entities and relations between argument entities with the purpose of reification in an AIF (see Section 3.2).
2. *Communication*: the core ontology for items which relate to the interchange of arguments between two or more participants in an environment, including *locutions* and *protocols* (see Section 3.4).
3. *Context*: the core ontology for items associated with environments in which argumentation may take place. These include *participants* in argument exchanges (*agents*), *theories* contained in the environment that are used for argumentation, and other aspects which may affect the meaning of arguments/communication of arguments (see Section 3.5).

In the next subsections an overview of the above concepts is given. Definitions are drawn from existing theories when possible, but may diverge where alignment between theories is needed. Items unique to argumentation (such as the notion of an "argument" itself) are naturally treated in greater depth than items for which more general definitions are already available (such as the notion of an "agent" for example). The relationships between these groups of concepts are shown in Fig. 1.

3.1 The Notion of Argument

Before proceeding with these definitions, it is worth noting that we will not take a position on the precise definition of the notion of "argument" itself, even though later sections do provide structures for describing argument. The reason for this is that initially we found it too difficult to select a single definition acceptable to all. We contend that progress on such a definition might be better made once some consensus is reached on the necessary lower level concepts. A useful starting point for understanding philosophical notions of arguments can however be found in David Hitchcock's input to the original AIF meeting.[12]

[12] `http://www.x-opennet.org/aif/Inputs/aif2005_david_hitchcock_1.pdf`

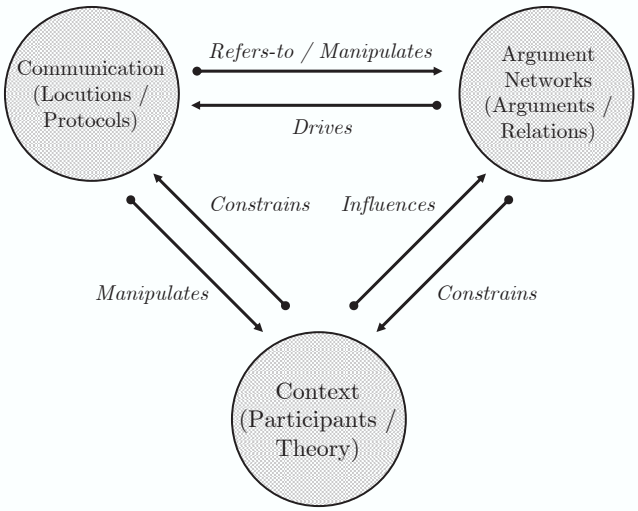


Fig. 1. Overview diagram of main groups of concepts defined by the AIF Core Ontology

3.2 Arguments / Argument Networks

The following section defines the top level concepts to be considered for an ontology of arguments and relationships between arguments.

Concepts and Relations: The starting point of this section is the assumption that argument entities can be represented as nodes in a directed graph (di-graph). This di-graph is informally called an *argument network* (AN). An example of an AN is displayed in Fig. 3. This figure will be described later, in Sec. 3.3. The rational for not to restrict ourselves to directed acyclic graphs (DAGs) or even trees is that argumentation formalisms vary to a great extent. A number of formalisms allow for cycles where others forbid them explicitly. One of our basic assumptions is that the core ontology should cater for these differences, and should be able to capture extreme cases.

Nodes: There are two kinds of nodes, namely, *information nodes* (I-nodes) and scheme application nodes or *scheme nodes* (S-nodes) for short (see Fig. 2). Note that one alternative for "scheme node" could be "application node". However, the meaning of "application" is not precise, neglecting the scheme connotation.

Whereas I-nodes relate to content and represent claims that depend on the domain of discourse, S-nodes are applications of *schemes*. Such schemes may be considered as domain-independent patterns of reasoning (that resemble rules of inference in deductive logics but broadened to non-deductive logics and not

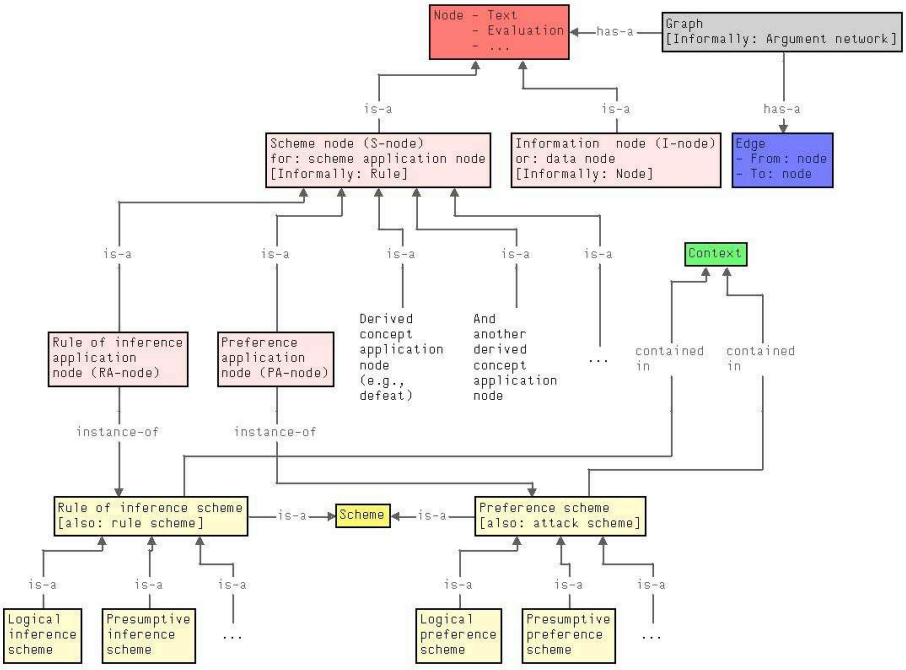


Fig. 2. Concepts and relations for an ontology of arguments

restricted to classical logical inference). The present ontology deals with two different types of schemes, namely *inference schemes* and *attack schemes*. Potentially scheme types could exist, such as evaluation schemes and scenario schemes, which will not be addressed here.

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.

Node Attributes: Nodes may possess different attributes such as "title," "text," "creator," "type" (decision, action, goal, belief), "creation date," "evaluation" (or "strength," or "conditional evaluation table"), "acceptability," and "polarity" (values either "pro" or "con"). These attributes may vary and are not part of the core ontology. The term "conditional evaluation table" is inspired by its Bayesian analogon named "conditional probability table" (CPT). Most attributes are proper, that is, essential to the node itself, while others are

derived. In this example, all attributes except "acceptability" are proper. It is imaginable that a derived attribute such as "acceptability" may be obtained from node-specific attributes through calculation. In this case, "acceptability" may be obtained from "evaluation" through mechanical inference.[13]

Edges: Let us analyze the notion of support. In the context of a graph representing argument-based concepts and relations, a node A is said to *support* node B if and only if an edge runs from A to B. This rather broad notion of support turns out to be remarkably convenient in discussions on argument ontology. Alternative terminology, more akin to graph-theory is "children of".[14]

1. Every node (i.e., every I-node and every S-node) can be supported by zero or more S-nodes.
2. Every S-node can be supported by zero or more I-nodes.

Edges do not need to be explicitly marked, labelled, or otherwise supplied with semantical pointers. A very practical example showing this would be an "edge table" representing edges between nodes. Besides an OID (object identifier) column, such an edge table does not need more than two columns: a `from_oid` field, denoting the OID of the source node, and a `to_oid` field, denoting the OID of the sink node.

If desired, edge types can be inferred from the nodes they connect. Basically 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 ("information," or "data" supplying edges), S-to-I edges ("conclusion" edges) and S-to-S edges ("warrant" edges). Table 1 summarizes the relations associated with the semantics of support. Notice that I-to-I edges are forbidden, as will be discussed further on in this section.

To distinguish scheme edges from data edges in diagrams, edges that emanate from S-nodes may be supplied with a closed arrowhead at the end, while edges that emanate from I-nodes may be supplied with an open arrowhead at the end. Edges fall into different categories, such as *support edges* (that are associated or "colored" by the scheme of the S-node they are connected to; for S-to-S edges, the nodes that they emanate from), *inference edges* (those edges that are connected to an RA-node, shown in black in Fig. 3), and *attack edges* (edges

[13] There are voices that advocate to drop derived node attributes altogether, for different algorithms may assign different statuses to arguments within one and the same argument network.

[14] Note however that the term support could be misleading when applied to preference application nodes, as preference application is intuitively associated with concepts such as negation, counterargument and preference. In such cases it may help to think of *negative support*.

	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. Semantics of support.

that are connected to an PA-node, shown in red in Fig. 3).[15] Marking edges and applying arrowheads to edges is not part of the ontology but only meant to help human beings in its interpretation.

Constructions that are not permitted: The ontology is flexible enough to allow for exceptional constructions. Still, it does not account for a number of artifacts. The following list shows a number of constructions that are not accommodated for in the present ontology:

1. I-nodes cannot be linked to other I-nodes. The reason for this restriction is that I-nodes cannot be connected without explaining why the connection is being made. There is always a reason, scheme, justification, inference, or rationale behind a relation between two or more I-nodes.
2. S-nodes may not be employed as I-nodes. Notice that it is difficult to find a compelling example that would justify the use of an S-node as an I-node. A possible example could be *"But previously you said that items that look red generally are red, so in the same way I say here that items that look like an apple generally are an apple"*. In these cases, it seems that it is not really a scheme application that is being used as an I-node-like premise, but rather something slightly different. Also, rather than using an S-node as an I-node, it seems more plausible to re-apply the scheme used for that S-node to create a new S-node.

Derived concepts: Concepts from an extension ontology, in particular concepts such as *rebut*, *undercut*, *defend*, and *defeat* can in principle be derived from the concepts in the diagram that is displayed in Fig. 2. Thus, an argument qualified with derived concepts can in principle be described in terms of basic concepts in a mechanical manner. Nevertheless, such derived concepts may still

[15] Note that in the color printed version of the document different colors are visible for edges for clarity – however, they are not essential to intepretation.

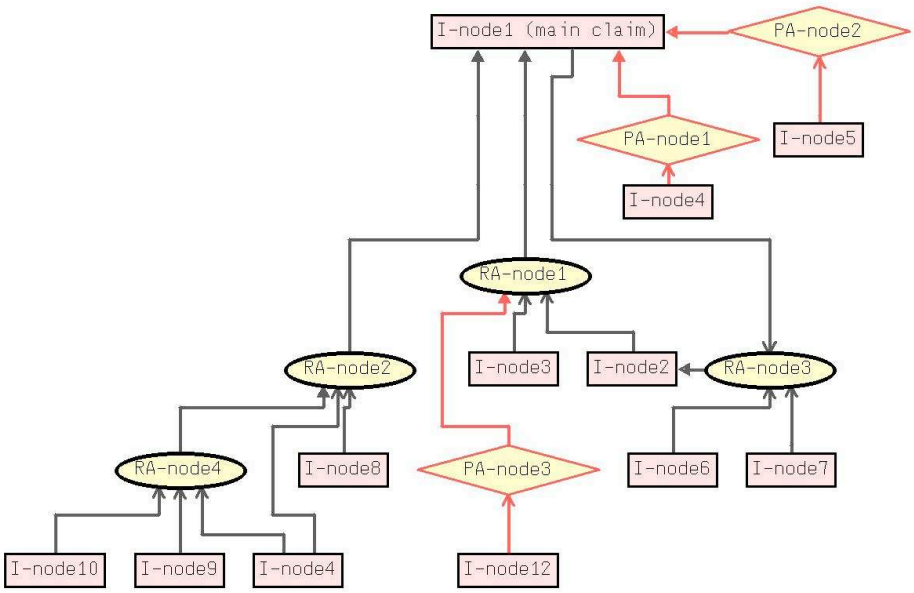


Fig. 3. Sample argument network.

have an important place that should be respected by their inclusion in an extension ontology that we might call "derived concepts" (see further discussions in Section 5).

3.3 Examples

This section presents three examples: an abstract example that shows most of the features of the ontology, a translation of Toulmin's scheme, and a simple concrete example.

Abstract example of an argument network: An abstract example of an argument network is displayed in Fig. 3. This network contains eleven I-nodes, namely I-node1 , . . . , I-node11 and six rule application nodes, namely RA-node1, RA-node2, RA-node3, RA-node4, PA-node1, and PA-node2. This abstract example is meant to demonstrate the flexibility of the core ontology, stretching the limits of the model. Obviously, most existing argument formalisms would not support the constructions shown in this example. Some observations that can be drawn from the diagram:

1. The main claim is supported by two inference applications and two attack applications.

2. Scheme-to-conclusion (SC) edges are drawn with an arrowhead, while premise-to-scheme (PS) edges are drawn without arrowheads. This is for two reasons. The first reason is to distinguish PS from SC edges. The second reason is to disambiguate direction when two S-nodes are connected by an SC edge (notice the SC-edge from RA-node4 to RA-node2). The arrowhead distinction is for diagrammatic purposes only, and it has no added value for representation and interchange formats.
3. I-node4 shows that our model allows for multiple node references. Thus, nodes may be referred to more than once. In particular, an argument network (AN) need not be a tree.
4. The two inference applications

$$\begin{aligned} \text{I-node2, I-node3} &-(\text{RA-node1})\rightarrow \text{I-node1 (main claim)} \\ \text{I-node1, I-node6, I-node7} &-(\text{RA-node3})\rightarrow \text{I-node2} \end{aligned}$$

   show that cycles in theory may occur.
5. If I-nodes are attacked, then the premises connected to the intermediate PA-node are called *rebutters*. For example, I-node1 is rebutted by I-node4 and I-node-5 through PA-node-1 and PA-node-2, respectively. These are two independent rebutters.
6. If RA-nodes are attacked, then the premises connected to the intermediate PA-node are called *undercutters*. For example, RA-node1 is undercut by I-node12 through PA-node3. In general, every type of node may be attacked, including attack nodes themselves. The diagram does not contain an instance of the latter.

Argumentation *à la* Toulmin. Example: Toulmin's scheme as depicted in (Eq. 1) is constituted of six essential elements, namely data (D), warrant (W), backing (B), qualifier (Q), rebuttal (R) and claim (C). A (somewhat liberal) translation is displayed in Fig. 4. The shadow-encircled nodes together relate to the original backing B.

$$\begin{array}{c} D \longrightarrow Q, C \\ \quad | \quad\quad | \\ \text{since } W \text{ unless } R \\ \quad | \\ \quad B \end{array} \tag{1}$$

Notice that in Fig. 4, R (rebuttal) attacks C (claim) rather than W (warrant). It is not clear from *"The Uses of Argument"* [21] whether R should attack C or W. Since an attack on C is called a rebuttal, and since an attack on W is called an undercutter in our terminology, we have chosen the one which is consistent with it. Nevertheless, R can reasonably be taken to attack C, to support not-C, to attack W, or to attack an implicit warrant (the dots). This document does not advocate a mechanism for translation but merely that any of those translations should be representable in the present ontology.

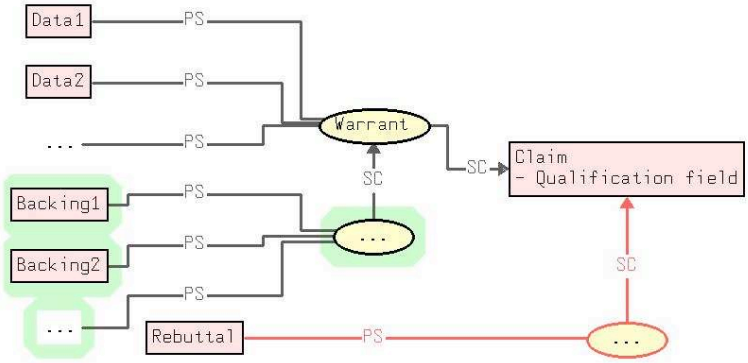


Fig. 4. Toulmin scheme.

A concrete and simple example: In Fig. 5 we show a concrete and simple example of an argument network for handling the well-known AI example of modelling the flying abilities of birds and penguins, and reasoning about whether a particular penguin `opus` can fly. In this case there are two arguments, one for `fly(opus)` and one for `~fly(opus)`.

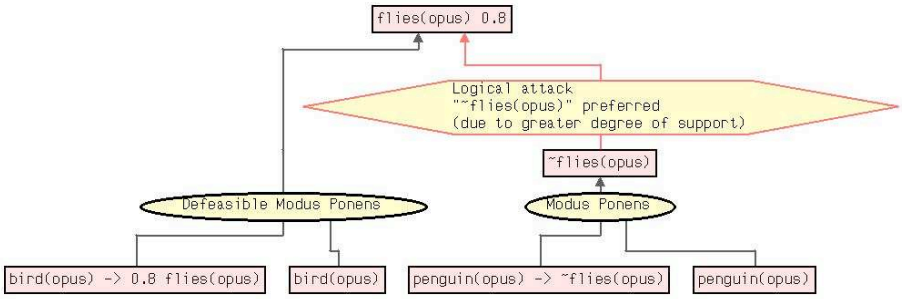


Fig. 5. Concrete example of an argument network.

The argument for `~fly(opus)` is composed of one scheme-application, namely *Modus Ponens* (MP). A simplistic version of MP reads as follows: *if there are two information nodes* `A(x)` *and* `A(x)->B(x)` *then conclusively infer* `B(x)`. The argument for `fly(opus)` is composed of one scheme-application, namely *defeasible Modus Ponens* (dMP). A simplistic version of dMP reads as follows: *If there are two information nodes* `A(x)` *and* `A(x)-(qualifier)->B(x)` *then defeasibly*

infer `B(x)`. The conflicting nodes `fly(opus)` and `~fly(opus)` are related by a PA-node that says that the argument against `fly(opus)` is conclusive and therefore preferred over the argument for `fly(opus)`. This PA-node is an instance of a more general scheme saying that deductive arguments always win out over non-deductive arguments.

3.4 Communication: Locution / Protocols

The second group of concepts identified in discussions are those which concern communication in the context of argumentation, for example, concepts which capture:

- The utterance of a statement containing an argument or argument network by an agent.
- A sequence of legal statements making reference to arguments/argument networks which could be made by a set of agents in order to make a decision or reach some other goal.

In turn, as with arguments / argument networks, communication also takes place in a context –elements of which may affect the interpretation of statements (such as references to the participants in a dialogue, the ontologies applying, the semantic models adopted etc.). Presentation here is initially split into two parts:

- *Locutions*: individual words, phrases or expressions uttered by an agent.
- *Interaction Protocols*: sequences of locutions involving one or more (usually at least two) agents and usually designed to achieve a specific goal (such as reaching an agreement or giving information).

Hence locutions form the basic building blocks of protocols. It is important to note however that there are different "schools of thought" on how the semantics for locutions and protocols should be defined in terms of one another. One approach, such as FIPA ACL [6], holds that semantics are attributed to individual locutions –and the semantics of a protocol are a compound of the semantics of individual locutions. Another approach holds that the semantics of locutions vary depending on their context (e.g. the commitments made thus far) and hence their place in a particular protocol [11].

Locutions: A rich literature exists on locutions of various types and their semantics. In terms of general agent communication, languages such as FIPA-ACL and KQML define sets of general locutions such as *inform*, *request*, *query*, *tell* and so on –each with an associated formal logical semantics. However, while these languages may provide useful resources, it is also clear from more specific argumentation literature that the types of locutions which occur frequently are more specific / different to those found in FIPA-ACL / KQML. Examples include: *assert*, *accept*, *challenge*, *question*, *concede*, and *prefer*.

While different authors use different labels for different locutions, there seem to often be similarities in semantics. Work such as that by McBurney, Parsons and Wooldridge [14], McBurney and Parsons [13], Maudet and Chaib-draa [10] and Mcburney, Hitchcock, and Parsons [12] provides a starting point for potentially determining a limited number of locutions which could form the core of an AIF, others potentially being added as extensions. In this setting, at a more general level however an AIF core ontology should usefully define the notions of:

- *Locution*: the notion of a locution, and its associated properties, which might include (taken from the FIPA-ACL message structure specification [6]):[16]
  - *Sender*: the agent uttering a locution (note that a distinction could also be made between the sender who makes and utterance and the originator(s) – an agent or group of agents responsible for generating the utterance.)
  - *Receiver* or *Receivers*: Agents "hearing" an utterance (distinctions could be made between intended recipients, those intentionally made aware of the message but not the intended recipients and those who unintentionally become aware of an utterance).
  - *Ontologies*: the ontologies which hold and define elements of the content.
  - *Language*: the content language used in the content part of the message (which should itself have a formal semantics).
  - *Protocol*: the protocol a locution is part of.
  - *Content*: the object of the locution.
  - Message management elements: items such as a *message-identifier*, a *conversation-identifier*, *in-reply-to field* etc.
- *Individual Locutions*: potentially a set of subclasses of the class of locutions which capture individual locutions such as those listed at the beginning of this section.

Interaction Protocols: It is possible to construct comprehensive standards of language usage for computational systems that are widely used and relatively precise. This is the case for programming language standards (such as PROLOG, dialects of ADA, *etc*). By contrast, in areas where standardization of more abstract concepts is required, consensus appears to be much harder to achieve, because abstract concepts are difficult to pin down uniquely in a simple way. In this circumstance it is often expedient to define precisely a core standard, containing only those elements essential to getting the job done, and then allow extensions to this core in a controlled (but perhaps less precise) way. An example of this form of standardization is the Process Interchange Format (PIF) which is a standard for describing processes. The PIF core contains a small number of very generic concepts at the heart of that standard and then allows those with specific process description needs to meet their own requirements by building on that core.

[16] Note that additionally one could add a slot for *semantics* which points to the defined formal semantics for the locution.

$$
\begin{aligned}
Model &:= \{Clause, \ldots\} \\
Clause &:= Role :: Def \\
Role &:= a(Type, Id) \\
Def &:= Role \mid Message \mid Def\ then\ Def \mid Def\ or\ Def \\
Message &:= M \Rightarrow Role \mid M \Rightarrow Role \leftarrow C \mid M \Leftarrow Role \mid C \leftarrow M \Leftarrow Role \\
C &:= Constant \mid P(Term, \ldots) \mid \neg C \mid C \wedge C \mid C \vee C \\
Type &:= Term \\
Id &:= Constant \mid Variable \\
M &:= Term \\
Term &:= Constant \mid Variable \mid P(Term, \ldots) \\
Constant &:= \text{lower case character sequence or number} \\
Variable &:= \text{upper case character sequence or number}
\end{aligned}
$$

Fig. 6. LCC syntax

The definition of a interaction protocol language as part of an argument interchange format provides a number of advantages. If the language can be used for computation then the standard is, effectively, a programming standard and history suggests that such standards tend to be durable because they connect to practice (or fail to connect and then die cleanly). If it is also declarative – and hence independent of current fashion in low level implementation languages or basic communications protocols– then it can support formal analysis and verification more readily. In addition, the use of a high level language arguably facilitates human readability. For software engineers there is a natural notion of pattern in the design of protocols and this is one approach to extension from a core protocol syntax to a (more interesting) set of extensions via patterns.

Protocols are an area where traditional computer science helps supply standards. For example, Figure 6 defines the syntax of the Lightweight Coordination Calculus (LCC) that uses a combination of traditional specification drawn from CCS and logic programming (for details on LCC see [19]). An interaction model in LCC is a set of clauses, each of which defines how a role in the interaction must be performed. Roles are described by the type of role and an identifier for the individual agent undertaking that role. The definition of performance of a role is constructed using combinations of the sequence operator ('*then*') or choice operator ('*or*') to connect messages and changes of role. Messages are either outgoing to another agent in a given role ('⇒') or incoming from another agent in a given role ('⇐'). Message input/output or change of role can be governed by a constraint defined using the normal logical operators for conjunction, disjunction and negation. Notice that there is no commitment to the system of logic through which constraints are solved –on the contrary, we would expect different agents to operate different constraint solvers. Hence the standardization in LCC is on the generic language for describing interaction (only) and in this sense it is "core". It also has the added benefit of having a style of description that is

close to computation –in this case quite close to logic programming (despite the process operators) where we already have a successful ISO standard.

3.5 Context: General Context / Participants / Theory

The third group of concepts in the ontology is that of elements which form the context in which argumentation takes place. In keeping with the distinction already made between concepts for communication and those for arguments / argument networks, concepts related to context may also be usefully grouped into these two areas.

Communication Context: Here, context captures information relevant to argument-based dialogues. These include:

- *Participants:* We may require references to agents taking place in the dialogue, possibly including:
  1. Participant ID: an identifier for a participant.
  2. Participant role: the role of the participant in relation to the dialogue (e.g. pro, con, persuader, buyer, seller, etc.). This may influence the way dialogue proceeds.
- *Dialogue topic:* This refers to the main issue under discussion (e.g. the question under enquiry, or resource under negotiation).
- *Dialogue type:* a reference to the type of the dialogue (e.g. persuasion, negotiation [23]). This can be simply a name, or it can be a pointer to more elaborate dialogue typology.
- *Background theory:* This includes statements that participants agree upon (e.g. legal rules), and which may be used to construct arguments within the dialogue.
- *Commitment stores:* This is a data structure that allows agents to add and remove commitments during their dialogues [8].
- *Commitment rules:* These are rules that specify how dialogue participants may modify the content of commitment stores.

Argument Network Context: Here, context captures information relevant to the interpretation and processing of the argument network.

- *Argumentation theory rules:* These are the rules that specify the way arguments are constructed and interpreted. In a way, they represent the underlying formal argumentation theory. These include:
  1. Inference rules: These can be thought of as the specifications of the types of inference application nodes that can be used in the argument network.
  2. Preference rules: Similarly, these can be thought of as the specifications of the types of preference application nodes that can be used in the argument network.

- *Background theory:* This includes statements taken for granted (e.g. legal rules), and which may be used to interpret or process arguments.
- *Domain ontologies:* One could add references to ontologies that may be used to interpret argument networks. For example, suppose an argument network represents claims and justifications of the medical properties of a particular drug. In order to process these arguments automatically, we may benefit from a specialized medical drug ontology while interpreting these arguments.

4 Reifications

Reifications of the concepts defined in the AIF are *concretizations from abstract to more concrete definitions*. In particular the primary use of reifications in AIF is to define concrete syntaxes which can be unambiguously serialized and de-serialized for transmission between two communicating participants exchanging arguments or between two software tools using the AIF:

- More than one reification may exist.
- Two different reifications may not be interoperable. That is, serializers for one reification may produce output which is not readable by parsers for another.
- While individual reifications will each aim to capture the semantics of the concepts defined in the AIF ontologies, they may also be influenced by the semantics of the encoding language used. Hence minor semantic differences as well as syntactic differences may arise.

A simple example of what is meant by a reification can be seen in the AIF input document by Willmott, Fox and Reed to the original AIF event.[17]

5 Conclusions and Open Issues

As described in the introduction, the development of an AIF is a highly challenging endeavor and this document is intended as a discussion starter and not a fully fledged proposal. Further, as noted in Section 3.2, the current model may well not capture all types of argumentation that are of interest. Specific significant open issues which arose during discussion included:

1. Currently no distinction is being made for AIF formalisms which might be used in GUI/Tool import-export type application and those which might be used in agent-to-agent communication. While the core concepts may be the same it remains an open issue as to whether one format can really adequately cover both cases.

[17] `http://x-opennet.org/aif/Inputs/aif2005_steven_willmott_2.pdf`

2. Given the potential richness of the communication concepts ontology it remains an open issue as to how close to generic Agent Communication Languages (ACLs – such as FIPA-ACL, KQML etc.) AIF definitions may get. This affects possible re-use of ACL concepts and/or overlap with them and/or worries about tractability issues which affected ACL semantics also affecting the semantics of concepts defined here.
3. How should the community of users around the AIF organize themselves to agree on core concepts and extensions?
4. How should reifications be generated in detail from high level concepts (e.g. development of specific RDF / XML schemas or other syntax forms?

A longer version of this document, initial inputs, previous versions and a discussion forum for feedback can be found on the AIF website at `http://x-opennet.org/aif/`.

5.1 Acknowledgments

Support is also gratefully acknowledged from Agentlink III [18] European Commission and the ASPIC (FP6-IST-002307)[19] research funded projects. Additional inputs and contributions are also gratefully acknowledged from all of the following: Leila Amgoud, Trevor Bench-Capon, Jamal Bentahar, Ivan Bratko, Martin Caminada, Sylvie Doutre, John Fox, Dan Grecu, David Hitchcock, Tsakou Ioanna, Paul Krause, Nicolas Maudet, Peter McBurney, Maxime Morge, Martin Mozina, Simon Parsons, Henri Prade, Henry Prakken, Chris Reed, Dave Robertson, Michael Rovatsos, Carles Sierra, and Michael Wooldridge.

While efforts have been made to reach a consensus on the content of this document, it is important to note that it remains the integration of a wide range of inputs, hence the final result *may not necessarily reflect the opinion of everybody who contributed* – authorship or being listed as contributor does not necessarily imply complete agreement with the text.

References

1. T. J. M. Bench-Capon. Argument in artificial intelligence and law. *Artificial Intelligence and Law*, 5(4):249–261, 1997.
2. D. Carbogim, D. Robertson, and J. Lee. Argument-based applications to knowledge engineering. *Knowledge Engineering Review*, 15(2):119–149, 2000.
3. C. I. Chesñevar, A. Maguitman, and R. P. Loui. Logical models of arguments. *ACM Computing Surveys*, 32(4):337–383, 2000.
4. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning and logic programming and n-person games. *Artifical Intelligence*, 77:321–357, 1995.
5. M. Elhadad. Using argumentation in text generation. *Journal of Pragmatics*, 24:189–220, 1995.

[18] `http://www.agentlink.org`
[19] `http://www.argumentation.org`

6. FIPA. Communicative Act Library Specification. Technical Report XC00037H, Foundation for Intelligent Physical Agents, 10 August 2001.
7. T. F. Gordon and N. Karacapilidis. The Zeno argumentation framework. In *Proceedings of the Sixth International Conference on AI and Law*, pages 10–18, New York, NY, USA, 1997. ACM Press.
8. C. L. Hamblin. *Fallacies*. Methuen, London, UK, 1970.
9. P. A. Kirschner, S. J. B. Schum, and C. S. Carr, editors. *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. Springer Verlag, London, 2003.
10. N. Maudet and B. Chaib-draa. Commitment-based and dialogue-game based protocols: new trends in agent communication languages. *The Knowledge Engineering Review*, 17(2):157–179, June 2002.
11. N. Maudet and B. Chaib-draa. Commitment-based and dialogue-game based protocols – new trends in agent communication language. *Knowledge Engineering Review*, 17(2):157–179, 2003.
12. P. McBurney, D. Hitchcock, and S. Parsons. The eight-fold way of deliberation dialogue. *Intelligent Systems (In press)*, 2005.
13. P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *Journal of Logic, Language and Information, Special Issue on Logic and Games*, 11(3):315–334, 2002.
14. P. McBurney, S. Parsons, and M. Wooldridge. Desiderata for agent argumentation protocols. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, pages 402–409. ACM Press, July 2002.
15. H. Prakken and G. A. W. Vreeswijk. Logics for defeasible argumentation. In D. Gabbay and F. Günthner, editors, *Handbook of Philosophical Logic*, volume 4, pages 219–318. Kluwer Academic Publishers, 2002.
16. I. Rahwan. (Editor) Special Issue on Argumentation in Multi-Agent Systems. *Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)*, 11(2):115–206, September 2005.
17. I. Rahwan, P. Moraitis, and C. Reed, editors. *Argumentation in Multi-Agent Systems: First International Workshop, ArgMAS 2004, New York, NY, USA, July 19, 2004, Revised Selected and Invited Papers*, volume 3366 of *Lecture Notes in Artificial Intelligence*. Springer Verlag, Berlin, Germany, 2005.
18. C. Reed and T. J. Norman, editors. *Argumentation Machines: New Frontiers in Argument and Computation*, volume 9 of *Argumentation Library*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.
19. D. Robertson. Multi-agent coordination as distributed logic programming. In *International Conference on Logic Programming*, pages 416–430, Sant-Malo, France, 2004.
20. K. Sycara. The PERSUADER. In D. Shapiro, editor, *The Encyclopedia of Artificial Intelligence*. John Wiley & Sons, January 1992.
21. S. Toulmin. *The Uses of Arguments*. Cambridge University Press, 1958.
22. 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.
23. D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. SUNY Press, Albany NY, USA, 1995.

Managing Social Influences through Argumentation-Based Negotiation

Nishan C. Karunatillake[1], Nicholas R. Jennings[1],
Iyad Rahwan[2], Sarvapali D. Ramchurn[1]

[1] School of Electronics and Computer Science, University of Southampton, Southampton, UK.
{nnc02r,nrj,sdr}@ecs.soton.ac.uk
[2] Institute of Informatics, The British University in Dubai, P.O.Box 502216 Dubai, UAE.
(Fellow) School of Informatics, University of Edinburgh, Edinburgh, UK.
irahwan@acm.org

Abstract. Social influences play an important part in the actions that an individual agent may perform within a multi-agent society. However, the incomplete knowledge and the diverse and conflicting influences present within such societies, may stop an agent from abiding by all its social influences. This may, in turn, lead to conflicts that the agents need to identify, manage, and resolve in order for the society to behave in a coherent manner. To this end, we present an empirical study of an argumentation-based negotiation (ABN) approach that allows the agents to detect such conflicts, and then manage and resolve them through the use of argumentative dialogues. To test our theory, we map our ABN model to a multi-agent task allocation scenario. Our results show that using an argumentation approach allows agents to both efficiently and effectively manage their social influences even under high degrees of incompleteness. Finally, we show that allowing agents to argue and resolve such conflicts early in the negotiation encounter increases their efficiency in managing social influences.

1 Introduction

Autonomous agents usually operate as a multi-agent community performing actions within a shared social context to achieve their individual and collective objectives. In such situations, the actions of these individual agents are influenced via two broad forms of motivations. First, the *internal influences* reflect the intrinsic motivations that drive the individual agent to achieve its own internal objectives. Second, as agents reside and operate within a social community, the social context itself influences their actions. For instance, within a structured society an agent may assume certain specific roles or be part of certain relationships. These, in turn, may influence the actions that an agent may perform. Here, we categorise such external forms of motivations as *social influences*.

Now, in many cases, both these forms of influence are present and they may give conflicting motivations to the individual agent. For instance, an agent may be internally motivated to perform a specific action. However, at the same time, it may also be subject to an external social influence (via the role it is enacting or the relationship that it is part of) not to perform it. Also an agent may face situations where different social influences motivate it in a contradictory fashion (one to perform a specific action and the other not

to). Furthermore, in many cases, agents have to carry out their actions in environments in which they are not completely aware of all the roles, relationships, or the ensuing commitments that they and their counterparts enact. Thus, in such instances, an agent may not be aware of the existence of all the social influences that could or indeed should affect its actions and it may also lack the knowledge of certain specific social influences that motivate other agents' actions. Therefore, when agents operate in a society with incomplete information and with such diverse and conflicting influences, they may, in certain instances, lack the knowledge, the motivation and/or the capacity to abide by all their social influences.

However, to function as a coherent society it is important for these agents to have a means to resolve such conflicts, manage their internal and social influences, and to come to a mutual understanding about their actions. To this end, *Argumentation-Based Negotiation* (ABN) has been advocated as a promising means of resolving conflicts within such agent societies [7, 12]. In more detail, ABN allows 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 the internal and social influences affecting their counterparts, thereby making it easier to resolve certain conflicts that arise due to incomplete knowledge. Furthermore, the negotiation element within ABN also provides a means for the agents to achieve mutually acceptable agreements to the conflicts of interests that they may have in relation to their different influences.

Against this background, this work advances the state of the art in the following ways. First, our main contribution is to propose a novel ABN approach that allows agents to detect, manage, and resolve conflicts related to their social influences in a distributed manner within a structured agent society. In order to demonstrate the performance benefits of our method, we use our proposed ABN framework to design a number of ABN strategies to manage such conflicts and then use an empirical evaluation to assess their impact. Specifically, we show that allowing agents to argue about their social influences provides them with the capability to not only manage their social influence more effectively, but to do so more efficiently as a society. Furthermore, we show that giving these agents the capability to challenge their counterparts and obtain their reasons for violating social commitments (instead of simply attempting to claim the penalty charges to which they are entitled) allows the agents to manage their social influences even more efficiently. Our second main contribution is to the ABN community. Here, we present a complete ABN framework which allows agents to argue and negotiate and resolve conflicts in the presence of social influences. Furthermore, we demonstrate the versatility of that framework; first, by mapping it to a specific computational problem of a multi-agent task allocation scenario and second, by using it to design a number of ABN strategies to resolve conflicts within a multi-agent society.

To this end, the remainder of the paper is structured as follows. First, Section 2 highlights the theoretical model of our ABN framework. Section 3 then maps this model to a computational context detailing the different representations and algorithms used. Subsequently, Section 4 details the experimental setting, presents our results and an analysis of the key observations. Next, Section 5 discusses the related work and Section 6 concludes.

2 Social Argumentation Model

In this section, we give a brief overview of our formal and computational framework for arguing and negotiating in the presence of social influences. In abstract, our framework consists of four main elements: (i) a *schema* for reasoning about social influence, (ii) a set of *social arguments* that make use of this schema, (iii) a *language and protocol* for facilitating dialogue about social influence, and (iv) a set of *decision functions* that agents may use to generate dialogues within the protocol. In the following sub-sections, we discuss each of these elements in more detail.[1]

2.1 Social Influence Schema

The notion of *social commitment* acts as our basic building block for capturing social influence. First introduced by Castelfranchi [3], it remains simple, yet expressive, and is arguably one of the fundamental approaches for modelling social behaviour among agents in multi-agent systems. In essence, a social commitment ($\mathrm{SC}_\theta^{x \rightarrow y}$) is a commitment by one agent x (termed the *debtor*) to another y (termed the *creditor*) to perform a stipulated action θ. As a result of such a social commitment, the debtor is said to attain an *obligation* toward the creditor, to perform the stipulated action. The creditor, in turn, attains certain rights. These include the right to demand or require the performance of the action, the right to question the non-performance of the action, and, in certain instances, the right to demand compensation to make good any losses suffered due to its non-performance. We refer to these as *rights to exert influence*. This notion of social commitment, resulting in an obligation and rights to exert influence, allows us a means to capture social influences between two agents. In particular, obligations reflect the social influences an agent is subjected to, while rights reflect the social influences the agent is capable of exerting on others.

Given this basic building block for modelling social influence between specific pairs of agents, we now proceed to explain how this notion is extended to capture social influences resulting due to factors such as roles and relationships within a wider multi-agent society (i.e., those that rely on the structure of the society, rather than the specific individuals who happen to be committed to one another). Specifically, since most relationships involve the related parties carrying out certain actions for each other, we can view a relationship as an encapsulation of social commitments between the associated roles. For instance, a relationship between the roles *supervisor* and *student* may be associated with a social commitment "to hand over the thesis in a timely manner." This social commitment, in turn, gives the student an obligation toward the supervisor to hand in the thesis, and gives the supervisor the right to exert influence on the student by either demanding that he does so or through questioning his/her non-performance. In a similar manner, the supervisor may be influenced to review and comment on the thesis. This again is another social commitment associated with the relationship. In this instance, it subjects the supervisor to an obligation to review the thesis while the student gains the right to demand its performance. In this manner, social commitment again provides

[1] It is important to note that here we only give a basic recap of our model to enable the reader to gain an overall understanding. A comprehensive formal representation of the framework can be found in [8, 9].

an effective means to capture the social influences emanating through roles and relationships of the society (independently of the specific agents who take on the roles). Given this descriptive definition of our model, we now formulate these notions to capture the social influences within multi-agent systems as a schema (refer to Figure 1 and formulae (1) through (6)):

Definition 1: For $n_A, n_R, n_P, n_\Theta \in \mathbb{N}^+$, let:

- $A = \{a_1, \ldots, a_{n_A}\}$ denote a finite set of agents,
- $R = \{r_1, \ldots, r_{n_R}\}$ denote a finite set of roles,
- $P = \{p_1, \ldots, p_{n_P}\}$ denote a finite set of relationships,
- $\Theta = \{\theta_1, \ldots, \theta_{n_\Theta}\}$ denote a finite set of actions,
- Act : $A \times R$ denote the fact that an agent is acting a role,
- RoleOf : $R \times P$ denote the fact that a role is related to a relationship, and
- In : $A \times R \times P$ denote the fact that an agent acting a role is part of a relationship.

If an agent acts a certain role and that role is related to a specific relationship, then that agent acting that role is said to be part of that relationship (as per Cavedon and Sonenberg [4]):

$$\text{Act}(a, r) \wedge \text{RoleOf}(r, p) \rightarrow \text{In}(a, r, p) \qquad \text{(Rel. Rule)}$$

Definition 2: Let SC denote a finite set of social commitments and $\text{SC}_\theta^{x \rightarrow y} \in SC$. Thus, as per [3], $\text{SC}_\theta^{x \rightarrow y}$ *will result in the debtor attaining an obligation toward the creditor to perform a stipulated action and the creditor, in turn, attaining the right to influence the performance of that action:*

$$\text{SC}_\theta^{x \rightarrow y} \rightarrow [\text{O}_\theta^{x \rightarrow y}]_x^f \wedge [\text{R}_\theta^{y \rightarrow x}]_y \,, \qquad \text{(S-Com Rule)}$$

where:

- $[\text{O}_\theta^{x \rightarrow y}]_x^f$ represents the obligation that x attains that subjects it to an influence of a degree f (refer to [9] for more details) toward y to perform θ and
- $[\text{R}_\theta^{y \rightarrow x}]_y$ represents the right that y attains which gives it the ability to demand, question, and require x regarding the performance of θ.

Definition 3: Let:

- DebtorOf : $(R \cup A) \times SC$ denote that a role (or an agent) is the debtor in a social commitment,
- CreditorOf : $(R \cup A) \times SC$ denote that a role (or an agent) is the creditor in a social commitment,
- ActionOf : $\Theta \times SC$ denote that an act is associated with a social commitment, and
- AssocWith : $SC \times P$ denote that a social commitment is associated with a relationship.

If the roles associated with the relationship are both the creditor and the debtor of a particular social commitment, then we declare that social commitment is associated with the relationship (as per Section 2.1).

Applying the Rel. Rule to a society where: $a_i, a_j \in A \wedge r_i, r_j \in R \wedge p \in P$ s.t. $\text{Act}(a_i, r_i)$, $\text{Act}(a_j, r_j)$, $\text{RoleOf}(r_i, p)$, $\text{RoleOf}(r_j, p)$ hold true, we obtain:

$$\text{Act}(a_i, r_i) \wedge \text{RoleOf}(r_i, p) \rightarrow \text{In}(a_i, r_i, p) \qquad (1)$$

$$\text{Act}(a_j, r_j) \wedge \text{RoleOf}(r_j, p) \rightarrow \text{In}(a_j, r_j, p). \qquad (2)$$

Now, consider a social commitment $\text{SC}_\theta^{r_i \rightarrow r_j}$ associated with the relationship p in this society. Applying this to Definition 3 we obtain:

$$(\text{DebtorOf}(r_i, \text{SC}) \wedge \text{RoleOf}(r_i, p)) \wedge (\text{CreditorOf}(r_j, \text{SC}) \wedge \text{RoleOf}(r_j, p)) \\ \wedge \text{ActionOf}(\theta, \text{SC}) \rightarrow \text{AssocWith}(\text{SC}_\theta^{r_i \rightarrow r_j}, p). \qquad (3)$$

Applying the S-Comm rule to $\text{SC}_\theta^{r_i \rightarrow r_j}$ we obtain:

$$\text{SC}_\theta^{r_i \rightarrow r_j} \rightarrow \left[\text{O}_\theta^{r_i \rightarrow r_j}\right]_{r_i}^f \wedge \left[\text{R}_\theta^{r_j \rightarrow r_i}\right]_{r_j}. \qquad (4)$$

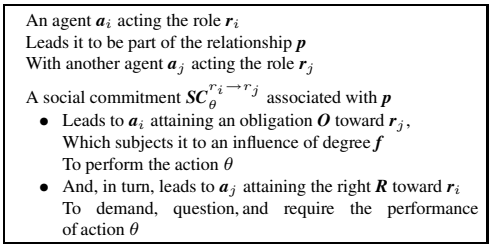


Fig. 1. Schema of Social Influence.

Combining (1), (3) and (4) we obtain:

$$\text{In}(a_i, r_i, p) \wedge \text{AssocWith}(\text{SC}_\theta^{r_i \rightarrow r_j}, p) \rightarrow \left[\text{O}_\theta^{a_i \rightarrow r_j}\right]_{a_i}^{f}. \tag{5}$$

Combining (2), (3) and (4) we obtain:

$$\text{In}(a_j, r_j, p) \wedge \text{AssocWith}(\text{SC}_\theta^{r_i \rightarrow r_j}, p) \rightarrow \left[\text{R}_\theta^{a_j \rightarrow r_i}\right]_{a_j}. \tag{6}$$

2.2 Social Arguments

Having captured the notion of social influence into a schema, we now show how agents can use this schema to systematically identify social arguments to negotiate in the presence of social influences. Specifically, we identify two major ways in which social influence can be used to change decisions and, thereby, resolve conflicts between agents.

Socially Influencing Decisions. One way to affect an agent's decisions is by arguing about the validity of that agent's practical reasoning [2]. Similarly, in a social context, an agent can affect another agent's decisions by arguing about the validity of the other's social reasoning. In more detail, agents' decisions to perform (or not) actions are based on their internal and/or social influences. Thus, these influences formulate the *justification* (or the reason) behind their decisions. Therefore, agents can affect each other's decisions indirectly by affecting the social influences that determine their decisions. Specifically, in the case of actions motivated via social influences through the roles and relationships of a structured society, this justification to act (or not) flows from the social influence schema (see Section 2.1). Given this, we can further classify the ways that agents can socially influence each other's decisions into two broad categories:

1. Undercut the opponent's existing justification to perform (or not) an action by disputing certain premises within the schema that motivates its opposing decision (i.e., dispute a_i is acting role r_i, dispute SC is a social commitment associated with the relationship p, dispute θ is the action associated with the obligation O, etc.).
2. Rebut the opposing decision to act (or not) by,
   i. Pointing out information about an alternative schema that justifies the decision not to act (or act as the case may be) (i.e., point out that a_i is also acting in role r_i, that SC is also a social commitment associated with the relationship p, that θ is the action associated with the obligation O, etc.).
   ii. Pointing out information about conflicts that could or should prevent the opponent from executing its opposing decision (i.e., point out conflicts between two existing *obligations*, *rights*, and *actions*).

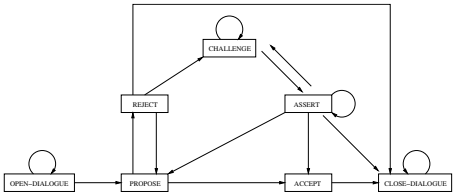


Fig. 2. Dialogue Interaction Diagram.

Negotiating Social Influence. Agents can also use social influences within their negotiations. More specifically, as well as using social argumentation as a tool to affect decisions (as above), agents can also use negotiation as a tool for "trading social influences". In other words, the social influences are incorporated as additional parameters of the negotiation object itself. For instance, an agent can promise to (or threaten not to) undertake one or many future obligations if the other performs (or does not perform) a certain action. It can also promise not to (or threaten to) exercise certain rights to influence one or many existing obligations if the other performs (or does not perform) a certain action. In this manner, the agents can use their obligations, rights, and even the relationship itself as parameters in their negotiations.

2.3 Language and Protocol

To enable agents to express their arguments, we define two complimentary languages: the *domain language* and the *communication language* (see [8] for a complete formal specification). The former allows the agents to express premises about their social context and also the conflicts that they may face while executing actions within such a context. The communication language, on the other hand, enables agents to express premises about the social context in the form of arguments and, thereby, engage in their discourse to resolve conflicts. This consists of seven elocutionary particles (i.e., *OPEN-DIALOGUE*, *PROPOSE*, *ACCEPT*, *REJECT*, *CHALLENGE*, *ASSERT*, and *CLOSE-DIALOGUE*). These locutions can be used together with content expressed in the domain language in order to allow agents to make *utterances* (e.g., assert a particular social premise, challenge a premise, make a specific proposal, and so on).

The protocol, which indicates the legal ordering of communication utterances, has six main stages: (i) *opening*, (ii) *conflict recognition*, (iii) *conflict diagnosis*, (iv) *conflict management*, (v) *agreement*, and (vi) *closing*. The opening and closing stages provide the important synchronisation points for the agents involved in the dialogue, the former indicating its commencement and the latter its termination [11]. The conflict recognition stage, the initial interaction between the agents, brings the conflict to the surface. Subsequently, the diagnosis stage allows the agents to establish the root cause of the conflict and also to decide on how to address it (i.e., whether to avoid the conflict or attempt to manage and resolve it through argumentation and negotiation [7]). Next, the conflict management stage allows the agents to argue and negotiate, thus, addressing the cause of this conflict. Finally, the agreement stage brings the argument to an end, either with the participants agreeing on a mutually acceptable solution or agreeing to disagree due to the lack of such a solution. In operation, it is defined as a dialogue game protocol which gives locutions rules (indicating the moves that are permitted), com-

Algorithm 1 Decision making algorithm for *PROPOSE*.

1: **if** $(\textit{Capable}(do(a_i, \theta_i)) \wedge B^{a_i}_{do(a_j,\theta_j)} > C^{a_i}_{do(a_i,\theta_i)})$ **then**
2: *PROPOSE*$(do(a_j, \theta_j), do(a_i, \theta_i))$
3: **end if**

Algorithm 2 Decision making algorithm for *ACCEPT* or *REJECT*.

1: **if** $(\textit{Capable}(do(a_j, \theta_j)) \wedge B^{a_j}_{do(a_i,\theta_i)} > C^{a_j}_{do(a_j,\theta_j)})$ **then**
2: *ACCEPT*$(do(a_j, \theta_j), do(a_i, \theta_i))$
3: **else**
4: *REJECT*$(do(a_j, \theta_j), do(a_i, \theta_i))$
5: **end if**

mitment rules (defining the commitments each participant incurs with each move), and structural rules (specifying the types of moves available following the previous move). Figure 2 presents these locutions and structural rules in abstract.

2.4 Decision Making Functionality

The protocol described above gives agents a number of different options, at various stages, as to what utterances to make. For instance, after a proposal the receiving agent could either accept or reject it. After a rejection, the agent may choose to challenge this rejection, end the dialogue, or forward an alternative proposal. An agent, therefore, still requires a mechanism for selecting a particular utterance among the available legal options. To this end, for each of the possible dialogue moves, we specify general decision making algorithms to give the agents that capability. Specifically, Algorithms 1 and 2 show two such examples, the former for generating a proposal and the latter for evaluating such a proposal. In abstract, a proposal in our formulation has two aspects; the request and the reward. Thus, when generating a proposal the agent would assess two aspects (i) if it is capable of performing the reward and (ii) the benefit it gains from the request ($B^{a_i}_{do(a_j,\theta_j)}$) is greater than the cost of reward ($C^{a_i}_{do(a_i,\theta_i)}$) (Algorithm 1). On the other hand, when evaluating a proposal, the agent will consider (i) if it is capable of performing the request and (ii) that the benefit of the reward ($B^{a_j}_{do(a_i,\theta_i)}$) is greater than the cost incurred in performing the request ($C^{a_j}_{do(a_j,\theta_j)}$) (Algorithm 2).

3 Argumentation Context

To evaluate how our argumentation model can be used as a means of managing social influences, we require a computational context in which a number of agents interact in the presence of social influences and conflicts arise as a natural consequence of these interactions. To this end, we now proceed to detail how we map our general framework into a specific multi-agent task allocation scenario. We first provide an overview description of the scenario and then proceed to explain how we map the notion of social influence within it. Finally, we detail how the agents can use our ABN model to interact within this social context and manage conflicts related to their social influences.

3.1 The Scenario

The argumentation context is based on a simple multi-agent task allocation scenario (similar to that presented in [7]) where a collection of self-interested agents interact

Table 1. A Sample Scenario

Time	a_0 $c_{(0,0.9)}, c_{(1,0.1)}$	a_1 $c_{(0,0.1)}, c_{(1,0.9)}$	a_2 $c_{(0,0.4)}, c_{(1,0.5)}$
t_0	$\theta_0 : [c_{(0,0.5)}, 200]$	$\theta_0 : [c_{(1,0.2)}, 500]$	$\theta_0 : [c_{(1,0.5)}, 700]$
t_1	$\theta_1 : [c_{(1,0.3)}, 900]$	$\theta_1 : [c_{(0,0.4)}, 300]$	$\theta_1 : [c_{(1,0.7)}, 100]$
t_2	$\theta_2 : [c_{(1,0.1)}, 400]$	$\theta_2 : [c_{(0,0.8)}, 900]$	
t_3	$\theta_3 : [c_{(0,0.9)}, 600]$		

to obtain services to achieve a given set of actions. In abstract, the context consists of two main elements. On one hand, each agent in the system has a list of *actions* that it is required to achieve. On the other hand, all agents in the system have different *capabilities* to perform these actions. In this context, agents are allowed to interact and negotiate between one another to find capable counterparts that are willing to sell their services to perform their actions. The following introduce these main elements in more detail:

Capability: All agents within the domain have an array of capabilities. Each such capability has two parameters: (i) a type value (x) defining the type of that capability and (ii) a capability level ($d \in [0, 1]$) defining the agent's competence level in that capability (1 indicates total competence, 0 no competence). Given this, we denote a capability as $c_{(x,d)} : [x, d]$.

Action: Each action has four main parameters: (i) the specified time (t_i) the action needs to be performed, (ii) the capability type (x) required to perform it, (iii) the minimum capability level (d_m) required, and (iv) the reward (r_i; distributed normally with a mean μ and a standard deviation σ) the agent would gain if the action is completed. Given this, we denote an action as $\theta_i : [t_i, c_{(x,d_m)}, r_i]$.

Each agent within the context is seeded with a specified number of such actions. This number varies randomly between agents within a pre-specified range. Table 1 depicts one such sample scenario for a three agent context (a_0, a_1, and a_2) with their respective capabilities and actions.

3.2 Modelling Social Influences

Given our argumentation context, we now describe how social influences are mapped into it. In order to provide the agents with different social influences, we embody a role-relationship structure into the multi-agent society. To do so, first, we define a specific number of roles and randomly link them to create a web of relationships. This defines the role-relationship structure. Figure 3(a) shows an example of such a representation between 3 roles: r_1, r_2, and r_3, where 1 indicates that a relationship exists between the two related roles, and 0 indicates no relationship.

Given this role-relationship structure, we now randomly specify social commitments for each of the active relationship edges (those that are defined as 1 in the mapping). A social commitment in this context is a commitment by one role, to another, to provide a certain type of capability when requested. As per Section 2.1, an important component of our notion of social commitment is its associated degree of influence. Thus, not all social commitments influence the agents in a similar manner (for more

	r_0	r_1	r_2
r_0	0	1	0
r_1	1	0	1
r_2	0	1	0

(a) Rol-Rel mapping.

	r_0	r_1	r_2
r_0	[0:0]	[200:0]	[0:0]
r_1	[400:100]	[0:0]	[200:600]
r_2	[0:0]	[700:200]	[0:0]

(b) Social commitment mapping.

	r_0	r_1	r_2
a_0	1	0	0
a_1	0	1	1
a_2	0	1	0

(c) Ag-Rol mapping.

Fig. 3. Social Influence Model.

details refer to [9]). Here, we map these different degrees of influence by associating each social commitment with a decommitment penalty. Thus, any agent may violate a certain social commitment at any given time. However, it will be liable to pay the specified decommitment value for this violation (this is similar to the notion of levelled commitments introduced in [14]). Since all our agents are self-interested, they prefer not to lose rewards in the form of penalties, so a higher decommitment penalty yields a stronger social commitment (thereby, reflecting a higher social influence). The following represents such a mapping. For instance, in Figure 3(b) the entry [400:100] in row 1, column 2 indicates that the role r_0 is committed to provide capabilities c_0 and c_1 to a holder of the role r_1. If the agent holding the role r_0 chooses not to honour this commitment it will have to pay 400 and 100 (respectively for c_0 and c_1) if asked. Having designed this social structure and the associated social commitments, finally we assign these roles to the actual agents operating within our system as shown in Figure 3(c).

From this representation, we can easily extract the rights and the obligations of each agent within our system. For instance, the agent-role mapping shows the fact that agent a_0 acts the role r_0. Given this, its obligations and rights can be extracted as follows:

- *Obligation to provide*:
  - c_0 to an agent acting r_1; obliged to pay 400 if decommitted.
  - c_1 to an agent acting r_1; obliged to pay 100 if decommitted.
- *Rights to demand*:
  - c_0 from an agent acting r_1; right to demand 200 if decommitted.

Given this global representation of social influence, we will now detail how we seed these agents with this information. Since one of the aims in our experiments is to test how agents use argumentation to manage and resolve conflicts created due to incomplete knowledge about their social influences, we generate a number of settings by varying the level of knowledge seeded to the agents. More specifically, we give only a subset of the agent-role mapping.[2] We achieve this by randomly replacing certain 1s with 0s and give this partial knowledge to the agents during initialisation. Thus, a certain agent may not know all the roles that it or another agent may act. This may, in turn, lead to conflicts within the society, since certain agents may know certain facts about the society that others are unaware of. By controlling this level of change, we

[2] Theoretically it is possible to introduce imperfections to all the premises within the schema (i.e., Act(a_i, r_i), RoleOf(r_i, p), AssocWith$(SC^{r_i \leftarrow r_j}, p)$, InfluenceOf$(O, f)$ etc.; see Section 2.1). However, since the objective of our experiments is to prove the concept of how arguments can resolve conflicts, instead of designing an exhaustive implementation with all possible imperfections and arguments, we chose to concentrate on the first two premises. Increasing the imperfections would merely increase the reasons why a conflict may occur, thus, bringing more arguments into play. However, this would have little bearing on the general pattern of the results.

Algorithm 3 The ***negotiate***() method.

```
1: [p_0, p_1, ..., p_max] ← generateProposals()
2: p ← p_0
3: isAccepted ← false
4:
5: {Loop till either the agent agrees or the last proposal fails.}
6: while (isAccepted ≠ true || p ≤ p_max) do
7:    response ← PROPOSE(p)
8:    if (response = "accept") then
9:       isAccepted ← true
10:   else
11:      if (p ≠ p_max) then
12:         p ← getNextViableProposal()
13:      end if
14:   end if
15: end while
16: return isAccepted
```

Algorithm 4 The ***argue***() method.

```
1: {Challenge for the opponent's justification}
2: H_o ← challenegeJustification()
3: {Generate personal justification}
4: H_p ← generateJustification()
5:
6: if (isValid(H_o) = false) then
7:    {Assert invalid premises of H_o}
8: else
9:    {Adopt premises of H_o into personal knowledge}
10: end if
11: if (isValid(H_p) = false) then
12:    {Correct invalid premises of H_p within personal knowledge}
13: else
14:    {Assert H_p}
15: end if
```

generate an array of settings ranging from perfect knowledge (0% missing knowledge) in the society, to the case where agents are completly unaware of their social influences (100% missing knowledge).

To explain this further, consider for instance that when initialising a_0 we seeded it with an incomplete agent-role map by replacing the 1 in column 1, row 1 with a 0. Thus, a_0 is unaware that it is acting the role r_0. As a result, it is not aware of its ensuing obligations and rights highlighted above. Now, when agents interact within the society this may lead to conflicts between them. For example, if a_0 refused to provide c_0 to a_1, it may request that the violation penalty of 400 be paid. However, since a_0 is unaware of its obligation it will not pay the amount. On the other hand, when initialising a_0 if we replace the 1 in column 2, row 3 with a 0, a_0 would now be unaware of its obligations towards agent a_2 since its lacks the information that its counterpart a_2 acts the role r_1. This, in turn, would also lead to conflicts with the society. In these situations, agents can use the argumentation process explained in Section 3.3 to argue and resolve such conflicts.

3.3 Agent Interaction

Having detailed the multi-agent context, we now proceed to discuss how the agents can use our ABN model to interact within this social setting. As mentioned in Section 3.1, agents within the system argue and negotiate with each other to find willing and capable partners to accomplish their actions. In essence, an agent that requires a certain capability will generate and forward *proposals* to another selected agent within the community requesting it to sell its services in exchange for a certain reward (Algorithm 1). If the receiving agent perceives this proposal to be viable and believes it is capable of performing it, then will *accept* it. Otherwise it will *reject* the proposal (Algorithm 2). In case of a reject, the original proposing agent will attempt to forward a modified proposal. The interaction will end either when one of the proposals is accepted or when all valid proposals that the proposing agent can forward are rejected (Algorithm 3). In this context, the two main elements of the negotiation interaction are:

Proposal Generation: When generating a proposal, an agent needs to consider two aspects (Algorithm 1): (i) whether it is capable of carrying out the *reward* and (ii) whether

the *benefit it gains from the request* is greater than the *cost incurred while performing the reward*. To simplify the implementation, we constrain our system to produce proposals with only monetary rewards. Thus, the generic proposal from an agent a_i to an agent a_j takes the form *PROPOSE*$(do(a_j, \theta_j), do(a_i, m))$ where θ_j is the requested action and m the monetary reward. In this context, calculating the benefit and the cost becomes straight forward. The benefit is the request r_j associated with the action θ_j and the cost of reward is m the monetary reward. Given this, the agent would generate an array of proposals with increasing amounts of monetary rewards, the lowest being 1 and the highest being $(r_j - 1)$.

Proposal Evaluation: When the receiving agent evaluates a proposal it also considers two analogous factors: (i) whether it is capable of performing the *request* and (ii) if the *benefit it gains from the reward* is greater than the *cost of carrying out the request* (Algorithm 2). To evaluate capability, the agent compares its own level with the minimum required to perform the action. In this case, the cost is the current opportunity cost. Here, all agents have a minimum asking price (set to μ the mean reward value, see Section 3.1) if they are not occupied, or, if they are, the cost is the reward plus the decommitment cost of the previously agreed action. The benefit, in the simplest case, is the monetary value of the reward m. However, if the agent has a social commitment to provide that capability type to the requesting agent, then the benefit is the monetary reward plus the decommitment penalty of this social commitment.

Given the negotiation interaction, we will now detail how agents argue (Algorithm 4) to resolve conflicts within the multi-agent society (such as the one highlighted in Section 3.2). Agents first detect conflicts by analysing the decommitment penalties paid by their counterparts for violating their social commitments. In more detail, when an agent with the right to demand a certain capability claims the penalty form another for violating its obligation and the amount paid in response is different from the amount it expects to receive, the agents would detect the existence of a conflict. Once such a conflict is detected agents attempt to resolve it by exchanging their respective justifications. These justifications would take the form of the social influence schema (see Equations 5 and 6 in Section 2.1) and are then analysed to diagnose the cause of the conflict. If there are inconsistencies between them, social arguments (Section 2.2; Type-1) are used to highlight these. If they are both valid, then each agent would point-out alternative justifications via asserting missing knowledge (Section 2.2; Type-2). The defeat-status is computed via a validation heuristic, which simulates a defeasible model such as [1].

4 Managing Social Influences

As mentioned in Section 1, when agents operate within a society with incomplete knowledge and with diverse and conflicting influences, they may, in certain instances, lack the knowledge, the motivation and/or the capacity to enact all their social commitments. In some cases, therefore, an agent may violate specific social commitments in favour of abiding by a more influential internal or external motivation. In other cases it may inadvertently violate such commitments simply due to the lack of knowledge of their existence. However, to function as a coherent society it is important for these agents to have a means to resolve such conflicts and manage their social influences in

Algorithm 5 Claim-Penalty-Non-Argue strategy.

```
1: isAccepted ← negotiate()
2: if (isAccepted = false) then
3:    compensation ← demandCompensation()
4: end if
```

Algorithm 6 Claim-Penalty-Argue strategy.

```
1: isAccepted ← negotiate()
2: if (isAccepted = false) then
3:    compensation ← demandCompensation()
4:    if (compensation < rightToPenalty) then
5:       argue()
6:    end if
7: end if
```

a systematic manner. Against this background, we will now investigate a number of different interaction strategies that allow the agents to manage their social influences within a multi-agent context. The underlying motivation for these strategies is our social influence schema (see Section 2.1), which gives the agents different rights; namely the right to demand compensation and the right to challenge non-performance of social commitments. Specifically, in the following we use our ABN model to design both arguing and non-arguing strategies to implement these forms of interactions and assess their relative performance benefits.

The experiments are set within the context described in Section 3 with 20 agents, each having 3 capabilities with different levels of competence (varied randomly). The number of actions each agent has vary between 20 and 30, while their respective rewards are set according to a normal distribution with a mean 1,000 and a standard deviation 500. We use two metrics to evaluate the overall performance of the different strategies (similar to [7, 13]): (i) the *total earnings* of the population as a measure of effectiveness (the higher the value, the more effective the strategy) and (ii) the *total number of messages* used by the population as a measure of efficiency (the lower the value, the more efficient the strategy). Here all reported results are averaged over 40 simulation runs to diminish the impact of random noise, and all observations emphasised are statistically significant at the 95% confidence level.

4.1 Demanding Compensation

If an agent violates a social commitment, one of the ways its counterpart can react is by exercising its right to demand compensation. This formulates our baseline strategy which extends our negotiation algorithm by allowing the agents to demand compensation in cases where negotiation fails (Algorithm 5). Once requested, the agent that violated its social commitment will pay the related penalty.[3] However, in imperfect information settings, a particular agent may violate a social commitment simply because it was not aware of it (i.e., due to the lack of knowledge of its roles or those of its counterparts). In such situations, an agent may pay a decommitment penalty different to what the other believes it should get, which may, in turn, lead to conflicts. In such situations, our second strategy allows agents to use social arguments to argue about their social influences (as per Section 2.2) and, thereby, manage their conflicts (Algorithm 6). Our hypothesis here is that by allowing agents to argue about their social influences we are providing them with a coherent mechanism to manage and resolve their conflicts and,

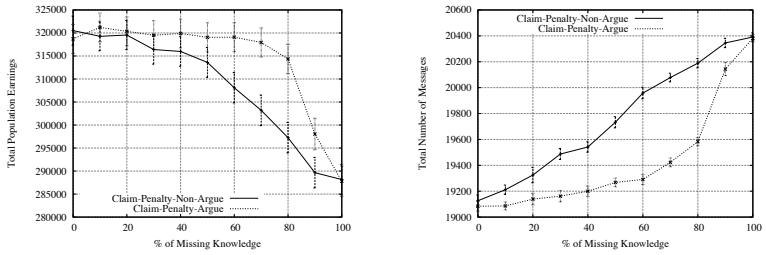
[3] To reduce the complexity, here, we assume that our agents do not attempt to deceive one another. Thus an agent will either honour its obligation or pay the penalty. We could drop this assumption and make it more realistic by incorporating trust and reputation mechanism into the system. However, this is beyond the scope of this paper.

thereby, allowing them to gain a better outcome as a society. To this end, the former strategy acts as our control experiment and the latter as the test experiment. Figures 4 and 5 show our results from which we make the following observations:

Observation 1: *The argumentation strategy allows agents to manage their social influences even at high uncertainty levels.*
If agents are aware of their social influences, they may use them as parameters within their negotiation interactions. Thereby, agents can endorse certain actions which may otherwise get rejected (see Section 2.2). This would, in turn, increase the population earnings as more actions are accomplished. However, if the agents are not aware of their social influences they may not be able to use these influences to endorse such actions. Therefore, we can observe a downward trend in the population earnings for both strategies as the agent's knowledge level about their social influences decrease (0 on the X-axis indicates perfect information, whereas, 100 represents a complete lack of knowledge about the social structure). However, we can observe that the non-argue strategy falls more rapidly than the argue one. This is because the argue method allows agents to manage and resolve conflicts of opinion that they may have about their social influences. For instance, if a certain agent is unaware of a role that another acts, it may correct this through arguing with that agent. Thus, arguing allows agents to correct such gaps in their knowledge and, thereby, resolve any conflicts that may arise as a result. In this manner, ABN allows the agents to manage their social influences even at high uncertainty levels. Thereby, as a society, the agents can accomplish more of their actions and gain a higher total earnings value. The non-arguing approach, which does not allow them to argue about their social influences and manage such conflicts, reduces the population earnings as knowledge imperfections increase within the social system.

Observation 2: *In cases of perfect information and complete uncertainty, both strategies perform equally.*
The reason for both strategies performing equally when there is perfect information (0 level) is because there are no knowledge imperfections. In other words, agents do not need to engage in argumentation to correct conflicts of opinions simply because such conflicts do not exist. On the other hand, the reason for both strategies performing equally when there is a complete lack of knowledge is more interesting. Since, none of the agents within the society are aware of any social influences (even though they exist) they are not able to detect any conflicts or violations. Consequently, agents do not resort to arguing to manage such conflicts (see *conflict recognition* stage in Section 2.3). Thus, when there is a complete lack of knowledge, the strategy that uses the argue strategy performs the same as the non-argue one.

Observation 3: *At all knowledge levels, the argumentation strategy exchanges fewer messages than the non-arguing one.*
Figure 4(b) shows the number of messages used by both strategies under all knowledge levels. Apart from the two end points, where argumentation does not occur (see Observation 2), we can clearly see the non-arguing strategy exchanging more messages (is less efficient) than the argue one. The reason for this is that even though agents use some number of messages to argue and correct their incomplete knowledge, thereafter the agents use their corrected knowledge in subsequent interactions. However, if the agents do not argue to correct their knowledge imperfections, they negotiate more fre-



(a) Total Population Earnings (b) Total Number of Messages

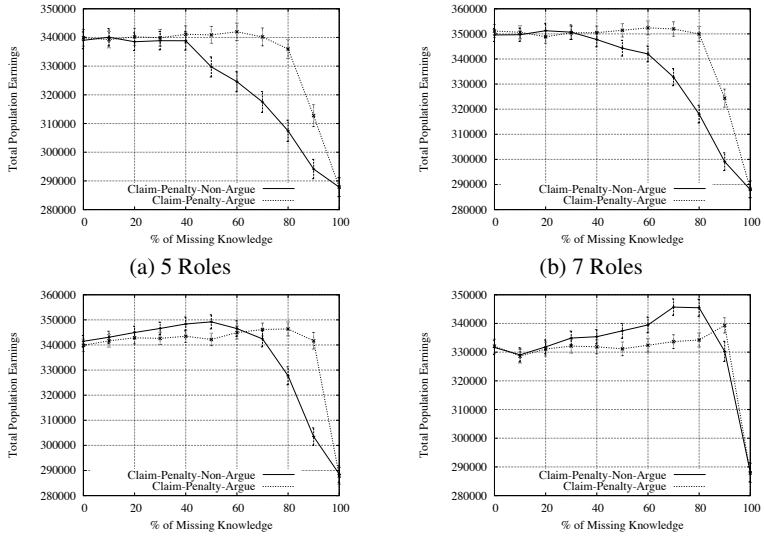
Fig. 4. Efficiency and Effectiveness of the Argue and Non-Argue strategies with 20 Agents and 3 Roles.

quently since they cannot use their social influence. Thus, this one-off increase of argue messages becomes insignificant when compared to the increase in the propose, accept, and reject messages due to the increased number of negotiations.

Observation 4: *When there are more social influences within the system, the performance benefit of arguing is only significant at high levels of knowledge incompleteness*. Figure 4(a) and Figures 5(a) through 5(d) show the effectiveness of both the strategies as the number of roles increases within the society. One of the key observations here is the decline rate of the non-argue strategy. We can see that as the number of roles increase, the rate of decline of the non-argue method becomes less pronounced. Furthermore, the crossover point where the non-argue method starts to be less effective than the argue strategy also shifts increasingly to the right (higher knowledge imperfections). In Figures 5(a) though 5(d) this level is roughly 50%, 70%, 80%, 90%. This again is a very interesting observation. As agents gain a higher number of roles, they aquire an increasing number of social influences. Now, as explained in Observation 1, the agents use these social influences as a resource to endorse their actions. Thus, when an agent has a higher number of social influences, its lack of knowledge about a certain particular influence makes little difference. The agent can easily replace it with another influence (which it is aware of) to convince its counterpart. Therefore, under such conditions, agents arguing about their social influences to correct their lack of knowledge would have little reward since the non-argue method can more simply replace it with another known influence and still achieve the same end. Only when an agent has a near complete lack of knowledge (i.e., 80%, 90%) does the argue strategy yeild significant performance gains. This observation complements our previous emperical study on the worth of argumentation at varying resource levels [7]. There we show that the benefit of arguing is more pronounced at low resource settings and under higher resource conditions the benefit is less.

4.2 Questioning Non-Performance

In the event that a particular social commitment is violated, apart from the right to demand compensation, our social influence schema also gives the agents the right to challenge and demand a justification for this non-performance (see Section 2.1). It is generally argued in ABN theory that allowing agents to exchange such meta-information in the form of justifications gives them the capability to understand each others' rea-



(a) 5 Roles (b) 7 Roles

(c) 10 Roles (d) 15 Roles

Fig. 5. Total population earnings with 20 agents and a varying number of roles.

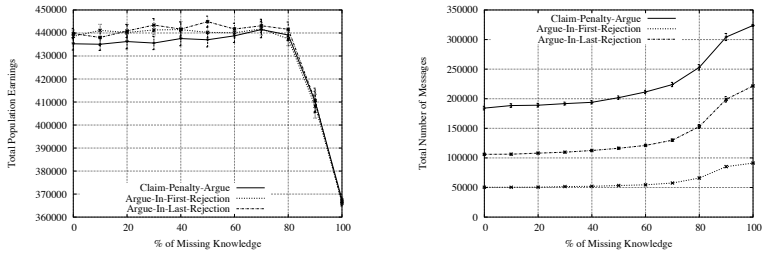
sons and, thereby, provides a more efficient method of resolving conflicts under uncertainty [12]. In a similar manner, we believe that providing the agents with the capability to challenge and demand justifications for violating social commitments also allows the agents to gain a wider understanding of the internal and social influences affecting their counterparts, thereby, providing a more efficient method for managing social influences in the presence of incomplete knowledge.

This intuition forms the underlying hypothesis for our next set of experiments. More specifically, we use our previous best strategy *Claim-Penalty-Argue* as the control experiment and design two other strategies (*Argue-In-First-Rejection* and *Argue-In-Last-Rejection*) to experiment with the effect of allowing the agents to challenge non-performance at different stages within the negotiation encounter. The former allows the agent to challenge after the receipt of the first rejection and the latter after the last rejection. Thus, the two differ on when the agent attempts to find the reason (in the first possible instance or after all proposals have been forwarded and rejected).[4] Figures 6(a) and 6(b) show our results and the following highlight our key observations:

Observation 5: *The effectiveness of the various argumentation strategies are broadly similar.*

Figure 6(a) shows no significant difference in the effectiveness of the three ABN strategies. This is due to the fact that all three strategies argue and resolve the conflicts even though they decide to argue at different points within the encounter. Therefore, we do not expect to have any significant differences in number of conflicts resolved. Thus, the effectiveness stays the same.

[4] Due to space restrictions we avoid specifying the algorithms for these two strategies here.



(a) Effectiveness (b) Efficiency

Fig. 6. Efficiency and Effectiveness of the various argumentation strategies.

Observation 6: *Allowing the agents to challenge earlier in the dialogue, significantly increases the efficiency of managing social influences.*

Figure 6(b) shows a significant difference in the number of messages used by the three strategies at all levels of knowledge. In more detail, the number of messages used by the *Argue-In-Last-Rejection* strategy is significantly lower than our original *Claim-Penalty-Argue* one. Moreover, the *Argue-In-First-Rejection* strategy has the lowest number of messages exchanged. The reason for this behaviour is based on how the agents use these reasons exchanged during the argue phase. In the *Claim-Penalty-Argue* strategy the main objective of arguing is to resolve the conflict regarding the penalty value that should be paid. However, it does not attempt to find out the reason for why its counterpart rejected its proposal. For instance, one reason could be the lack of capability. Another could be the reward of the proposal is not high enough to cover the cost. By challenging the reason for the rejection, the latter two strategies gain this meta-information which the agents constructively use in their subsequent interactions. For instance, if the counterpart rejected the proposal due to lack of capability, it can be excluded in future if the agent requires a capability which is equal or greater. In this way such reasons give useful meta-information to the agents for their future negotiations. So these strategies allow the agents to exploit such information and interact more efficiently as a society. Arguing in the first rejection provides this information earlier in the negotiation, which, in turn, gives the agent more capacity to exploit such information (even in the present negotiation) than getting it in the last encounter. Given this, we can conclude that in our context allowing the agents to challenge non-performance earlier in the negotiation allows them to manage their social influences more efficiently as whole.

5 Related Work

As highlighted in Section 1, to function as a coherent society, agents operating within a multi-agent society need the ability to detect, manage, and resolve conflicts in a systematic manner. Here, we will compare our ABN approach with two others suggested in the multi-agent literature. First, we note the work of [5] on electronic institutions where commitments of agents resulting due to social influences are managed through a performative structure. In more detail, they use a central authority to ensure that such commitments are upheld by controlling the type of locutions agents can issue in certain contexts based on the state of their commitments. In a similar vein, [6] provides a mechanism to control, verify, and manipulate commitments through the use of a state

machine. Now, one of the key distinctions of our approach from these is the absence of a central authority. Ours is a decentralised model where agents detect, manage and resolve conflicts about their social influences by arguing between each other. Another key feature in our method is its ability to function under incomplete knowledge. On the other hand, both the above approaches assume complete information within the central entity.

Our ABN framework also extends current ABN research by allowing the agents to argue, negotiate and manage conflicts in a multi-agent society. When compared against the model of Kraus *et al* [10] our framework has two distinct advantages. First, ours expressly takes into account the impact of society by way of social commitments, whereas their main focus is in formulating interactions between two agents. Second, they do not take into account the impact of incomplete information. In contrast, our social arguments captured in Section 2.2 allow agents to argue about their social influences and overcome such conflicts within a society. The work of Sierra *et al*. [15] is an important initial attempt to extend the work of [10] to a social context. Similar to our approach (and unlike [10]) they allow agents to argue in social contexts with imperfect information. However, they only consider authority based relationships, which we believe only capture a specialised form of social context (i.e., institutions or formal organisations). Our work, on the other hand, presents a more generic way of capturing social influences of roles and relationships (i.e., using social commitment with different degrees of influence), thus allowing agents' the ability to argue, negotiate and resolve conflicts under disparate social influences.

6 Conclusions and Future Work

The incomplete knowledge and the diverse conflicting influences present within a multi-agent society may prevent agents from abiding by all their social influences. In such situations, in order to function as a coherent society, agents require a mechanism to manage their social influences in a systematic manner. To this end, this paper develops a novel ABN approach that allows agents to argue, negotiate and, thereby, achieve a consensus, about their social influences. Furthermore, in order to assess the performance benefits of our proposed method, we carry out an empirical analysis by implementing such an ABN approach in a multi-agent task allocation context. Our results can be summarised as three main points. *First*, our method is shown to be both a more efficient and a more effective strategy in managing social influence even at high uncertainty levels when compared to a non-arguing approach. *Second*, we show that our approach can be further enhanced in terms of efficiency by allowing agents to challenge one another earlier in the negotiation encounter and using the meta-information that is gained to guide future negotiation encounters. *Third*, we show that both under complete uncertainty and when there are abundant social influences available in the society, the effectiveness of our approach is not significantly different from a non-arguing one.

In the future, we aim to expand our approach by allowing the agents to explicitly trade social influences in the form of threats and promises (as per Section 2.2) and examine the effect of so doing. At the moment agents only implicitly use these social influences to endorse their proposals. In such a system, we also plan to experiment with the effect of using different argument selection strategies in order to identify if certain strategies allow the agents to argue more efficiently or effectively than others.

7 Acknowledgements

This research is funded by EPSRC under the Information Exchange project (GR/S0370-6/01). We thank Xudong Luo, Peter McBurney, Timothy J. Norman, Pietro Panzarasa, and Chris Reed for their thoughts, contributions and discussions. We also extend our gratitude to the three anonymous reviewers for their valuable comments and suggestions, and also to AOS Ltd. for their JACK agent framework and support.

References

1. L. Amgoud and H. Prade. Reaching agreement through argumentation: A possibilistic approach. In D. Dubois, C. A. Welty, and M.-A. Williams, editors, *Proc. of the Ninth International Conference (KR2004)*, pages 175–182, Canada, 2004.
2. K. Atkinson, T. Bench-Capon, and P. McBurney. A dialogue game protocol for multi-agent argument over proposals for action. In *Argumentation in Multi-Agent Systems*, LNAI 3366, pages 149–161, NY, USA, 2004.
3. C. Castelfranchi. Commitments: From individual intentions to groups and organizations. In *Proc. of the first Int. Conf. on Multi-agent Systems (ICMAS'95)*, pages 41–48, San Francisco, CA, 1995.
4. L. Cavedon and L. Sonenberg. On social commitment, roles and preferred goals. In *Proc. of the third Int. Conf. on Multi-Agent Systems (ICMAS'98)*, pages 80–86, 1998.
5. M. Esteva, J. A. Rodríguez, C. Sierra, P. Garcia, and J. L. Arcos. On the formal specifications of electronic institutions. *LNAI*, 1991:126–147, 2001.
6. N. Fornara. *Interaction and Communication among Autonomous Agents in Multiagent Systems*. Phd thesis, Universit della Svizzera italiana, Facolt di Scienze della Comunicazione, 2003.
7. N. C. Karunatillake and N. R. Jennings. Is it worth arguing? In *Argumentation in Multi-Agent Systems*, LNAI 3366, pages 234–250, NY, USA, 2004.
8. N. C. Karunatillake, N. R. Jennings, I. Rahwan, and T. J. Norman. Arguing and negotiating in the presence of social influences. In *Proc. of the fourth Int. Central and Eastern European Conference on Multi-Agent Systems (CEEMAS'05)*, LNAI 3690, pages 223–235, Budapest, Hungary, 2005.
9. N. C. Karunatillake, N. R. Jennings, I. Rahwan, and T. J. Norman. Argument-based negotiation in a social context. In *Proc. of the second Int. Workshop on Argumentation in Multi-Agent Systems (ArgMAS'05)*, pages 74–88, Utrecht, The Netherlands, 2005.
10. S. Kraus, K. Sycara, and A. Evenchik. Reaching agreements through argumentation. *Artificial Intelligence*, 104(1-2):1–69, 1998.
11. P. McBurney, R. van Eijk, S. Parsons, and L. Amgoud. A dialogue-game protocol for agent purchase negotiations. *Autonomous Agents and Multi-Agent Systems*, 7(3):235–273, 2003.
12. 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.
13. S. D. Ramchurn, N. R. Jennings, and C. Sierra. Persuasive negotiation for autonomous agents: A rhetorical approach. In *Computational Models of Natural Argument, IJCAI*, pages 9–18, 2003.
14. T. W. Sandholm and V. R. Lesser. Advantages of a leveled commitment contracting protocol. In *Proc. of the 13th Conference on Artificial Intelligence (AAAI'96)*, pages 126–133, OR, USA, 1996.
15. C. Sierra, N. R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In *Proc. of fourth Int. Workshop on Agent Theories Architectures and Languages (ATAL'97)*, pages 167–182, 1998.

Argumentation-Based Learning

Taro Fukumoto[1] and Hajime Sawamura[2]

[1] Graduate School of Science and Technology, Niigata University
8050, 2-cho, Ikarashi, Niigata, 950-2181 JAPAN
fukumoto@cs.ie.niigata-u.ac.jp
[2] Institute of Natural Science and Technology Academic Assembly, Niigata University 8050, 2-cho, Ikarashi, Niigata, 950-2181 JAPAN
sawamura@ie.niigata-u.ac.jp

Abstract. Computational argumentation has been accepted as a social computing mechanism or paradigm in the multi-agent systems community. In this paper, we are further concerned with what agents believe after argumentation, such as how agents should accommodate justified arguments into their knowledge bases after argumentation, what and how agents acquire or learn, based on the results of argumentation. This is an attempt to create a new learning method induced by argumentation that we call Argument-Based Learning (ABL). To this end, we use our logic of multiple-valued argumentation LMA built on top of Extended Annotated Logic Programming EALP, and propose three basic definitions to capture agents' beliefs that should be rationally acquired after argumentation: knowledge acquisition induced by the undercut of assumptions, knowledge acquisition induced by difference of recognition, and knowledge acquisition induced by rebut. They are derived from two distinctive and advantageous apparatuses of our approach to multi-valued argumantation under : Paraconsistency and multiple-valuedness that EALP and LMA feature. We describe two overall argument examples to show the effectiveness and usefulness of the agent learning methods based on argumentation.

1 Introduction

In the last years, argumentation has been accepted as a promising social computing mechanism or paradigm in the multi-agent systems community. It has proven to be particularly suitable for dealing with reasoning under incomplete or contradictory information in a dynamically changing and networked distributed environment. The main concern, however, has lain in characterizing a set of acceptable (justified) arguments just as ordinary logics are concerned with characterizing validity and provability [3] [12]. In our view, there is one important missing angle in the past works on argumentation, which we should promote one more step further. It is such a view point that our objectives of making arguments or dialogue are not only for reaching to agreements, understanding with our social partners, and making decisions, but also for learning or acquiring information unknown or valuable to us. In this paper, we are concerned with

how agents should accommodate those justified arguments into their knowledge bases after argumentation, or what and how agents acquire or learn, based on the results of argumentation, just as we know each other, learn a lot and grow, through argumentation or dialogue in the daily, business or academic life. This paper describes a first step towards learning and growing or evolving agents through argumentation. To this end, we take a logic programming approach to argumentation since it can provide agents with both knowledge representation language and reasoning procedure in a integrated framework as well as in a computationally feasible way. We address ourselves to our purpose stated above in our Extended Annotated Logic Programming EALP and Logic of Multiple-Valued Argumentation LMA. EALP is an underlying knowledge representation language to which we extended GAP [7] for argumentation under uncertainty. It is very general and expressive as well as computationally feasible, allowing to deal with diverse types of truth values for various kinds of uncertain information. LMA is an argumentation framework on top of EALP, enabling agents to argue under their own knowledge bases with uncertainty [14]. We here emphasize that the most distinctive and advantageous point of our approach to argument-based learning (ABL) is to employ EALP and LMA with paraconsistency. As the result, we can be completely emancipated from the fear of inconsistency of knowledge bases and can concentrate on learning or knowledge acquisition itself in a manner more fused with argumentation, differently from the other approaches [1] [2] [6]. Furthermore, the multiple-valuedness that EALP and LMA feature brings us more refined knowledge acquisition methods than those of two-valued cases[1] [2] [6]. The paper is organized as follows. In Section 2 an 3, we outline Extended Annotated Logic Programming EALP and Logic of Multiple-valued Argumentation LMA respectively, to make the paper self-contained. In Section 4, we propose three definitions for learning or knowledge acquisition that is to be accomplished after argumentation. In Section 5, we illustrate two overall learning scenarios based on both argumentation and knowledge acquisition. In particular, we discuss a dynamically changing argument in which agents are involved in not only a single argument at a time but a process of consecutive arguments over time, and agents gradually become wise through repeated argumentation. In Section 6, we briefly describe some related works although there is nothing for us to be able to directly compare with ours. The final section summarizes contributions of the paper, and future work.

2 Overview of EALP

EALP is an underlying knowledge representation language that we formalized for our logic of multiple-valued argumentation LMA. EALP has two kinds of explicit negation: Epistemic Explicit Negation '$\neg$' and Ontological Explicit Negation '$\sim$', and the default negation '**not**'. They are supposed to yield a momentum or driving force for argumentation or dialogue in LMA. We here outline EALP.

2.1 Language

Definition 1. (**Annotation and annotated atoms**[7]). *We assume a complete lattice $(\mathcal{T}, \leq)$ of truth values, and denote its least and greatest element by $\bot$ and $\top$ respectively. The least upper bound operator is denoted by $\sqcup$. An annotation is either an element of $\mathcal{T}$* (constant annotation)*, an annotation variable on $\mathcal{T}$, or an annotation term. Annotation term is defined recursively as follows: an element of $\mathcal{T}$ and annotation variable are annotation terms. In addition, if $t_1, \ldots, t_n$ are annotation terms, then $f(t_1, \ldots, t_n)$ is an annotation term. Here, f is a total continuous function of type $\mathcal{T}^n \to \mathcal{T}$. If A is an atomic formula and μ is an annotation, then $A\!:\!\mu$ is an annotated atom. We assume an annotation function $\neg : \mathcal{T} \to \mathcal{T}$, and define that $\neg(A : \mu) = A : (\neg\mu)$. $\neg A : \mu$ is called the epistemic explicit negation*

Definition 2. (**Annotated literals**). *Let $A : \mu$ be an annotated atom. Then $\sim (A : \mu)$ is the ontological explicit negation (o-explicit negation) of $A : \mu$. An annotated objective literal is either $\sim A\!:\!\mu$ or $A\!:\!\mu$. The symbol $\sim$ is also used to denote complementary annotated objective literals. Thus $\sim\sim A\!:\!\mu = A\!:\!\mu$. If L is an annotated objective literal, then* **not** *L is a default negation of L, and called an annotated default literal. An annotated literal is either of the form* **not** *L or L.*

Definition 3. (**Extended Annotated Logic Programs (EALP)**). *An extended annotated logic program (EALP) is a set of annotated rules of the form: $H \leftarrow L_1 \,\&\, \ldots \,\&\, L_n$, where H is an annotated objective literal, and L_i $(1 \leq i \leq n)$ are annotated literals in which the annotation is either a constant annotation or an annotation variable.*

For simplicity, we assume that a rule with annotation variables or objective variables represents every ground instance of it. In this assumption, we restrict ourselves to constant annotations in this paper since every annotation term in the rules can evaluate to the elements of $\mathcal{T}$. We identify a distributed EALP with an *agent*, and treat a set of EALPs as a *multi-agent system*.

2.2 Interpretation

Definition 4. (**Extended annotated Herbrand base**). *The set of all annotated literals constructed from an EALP P on a complete lattice $\mathcal{T}$ of truth values is called the extended annotated Herbrand base $H_P^{\mathcal{T}}$.*

Definition 5. (**Interpretation**). *Let $\mathcal{T}$ be a complete lattice of truth values, and P be an EALP. Then, the interpretation on P is the subset $I \subseteq H_P^{\mathcal{T}}$ of the extended annotated Herbrand base $H_P^{\mathcal{T}}$ of P such that for any annotated atom A,*

1. *If $A\!:\!\mu \in I$ and $\rho \leq \mu$, then $A\!:\!\rho \in I$ (downward heredity);*
2. *If $A\!:\!\mu \in I$ and $A\!:\!\rho \in I$, then $A\!:\!(\mu \sqcup \rho) \in I$ (tolerance of difference);*
3. *If $\sim A\!:\!\mu \in I$ and $\rho \geq \mu$, then $\sim A\!:\!\rho \in I$ (upward heredity).*

Definition 6. (**Inconsistency**). *Let I be an interpretation. Then,*
1. *$A{:}\mu \in I$ and $\neg A{:}\mu \in I \Leftrightarrow I$ is epistemologically inconsistent (e-inconsistent).*
2. *$A{:}\mu \in I$ and $\sim A{:}\mu \in I \Leftrightarrow I$ is ontologically inconsistent (o-inconsistent).*
3. *$A:\mu \in I$ and **not** $A:\mu \in I$, or $\sim A:\mu \in I$ and **not** $\sim A:\mu \in I \Leftrightarrow I$ is inconsistent in default (d-inconsistent).*

When an interpretation I is o-inconsistent or d-inconsistent, we simply say I is *inconsistent*. We do not see the e-inconsistency as a problematic inconsistency since by the condition 2 of Definition 5, $A{:}\mu \in I$ and $\neg A{:}\mu = A{:}\neg\mu \in I$ imply $A{:}(\mu \sqcup \neg\mu) \in I$ and we think $A{:}\mu$ and $\neg A{:}\mu$ are an acceptable differentia. Let I be an interpretation such that $\sim A{:}\mu \in I$. By the condition 1 of Definition 5, for any ρ such that $\rho \geq \mu$, if $A{:}\rho \in I$ then I is o-inconsistent. In other words, $\sim A{:}\mu$ rejects all recognitions ρ such that $\rho \geq \mu$ about A.

Definition 7. (**Satisfaction**). *Let I be an interpretation. For any annotated objective literal H and annotated literal L and L_i, we define the satisfaction relation denoted by '$\models$' as follows.*
- $I \models L \;\Leftrightarrow\; L \in I$
- $I \models L_1 \,\&\, \cdots \,\&\, L_n \;\Leftrightarrow\; I \models L_1,\; \ldots,\; I \models L_n$
- $I \models H \leftarrow L_1 \,\&\, \cdots \,\&\, L_n \;\Leftrightarrow\; I \models H$ *or* $I \not\models L_1 \,\&\, \cdots \,\&\, L_n$

3 Overview of LMA

In formalizing logic of argumentation, the most primary concern is the rebuttal relation among arguments since it yields a cause or a momentum of argumentation or dialogue. The rebuttal relation for two-valued argument models is most simple, so that it naturally appears between the contradictory propositions of the form A and $\neg A$. In case of multiple-valued argumentation based on EALP, much complication is to be involved into the rebuttal relation under the different concepts of negation. One of the questions arising from multiple-valuedness is, for example, how a literal with truth-value ρ confronts with a literal with truth-value μ in the involvement with negation. In the next subsection, we outline important notions proper to a logic of multiple-valued argumentation LMA in which the above question is reasonably solved.

3.1 Annotated arguments

Definition 8. (**Reductant and Minimal reductant**).
Suppose P is an EALP, and C_i $(1 \leq i \leq k)$ are annotated rules in P of the form: $A:\rho_i \leftarrow L_1^i \,\&\, \ldots \,\&\, L_{n_i}^i$, in which A is an atom. Let $\rho = \sqcup\{\rho_1, \ldots, \rho_k\}$. Then the following annotated rule is a reductant of P.
$A{:}\rho \leftarrow L_1^1 \,\&\, \ldots \,\&\, L_{n_1}^1 \,\&\, \ldots \,\&\, L_1^k \,\&\, \ldots \,\&\, L_{n_k}^k$.
A reductant is called a minimal reductant when there does not exist non-empty proper subset $S \subset \{\rho_1, \ldots, \rho_k\}$ such that $\rho = \sqcup S$

Definition 9. (**Truth width [7]**). *A lattice $\mathcal{T}$ is n-wide if every finite set $E \subseteq \mathcal{T}$, there is a finite subset $E_0 \subseteq E$ of at most n elements such that $\sqcup E_0 = \sqcup E$.*

The notion of truth width is for limiting the number of reductants to be considered in argument construction. For example, the complete lattice $\mathcal{FOUR} = (\{\bot, \mathbf{t}, \mathbf{f}, \top\}, \leq)$, where $\forall x, y \in \{\bot, \mathbf{t}, \mathbf{f}, \top\}\ x \leq y \Leftrightarrow x = y \vee x = \bot \vee y = \top$, is 2-wide, and the complete lattice $(\Re[0,1], \leq)$ of the unit interval of real numbers is 1-wide.

Definition 10. (**Annotated arguments**). *Let P be an EALP. An annotated argument in P is a finite sequence $Arg = [r_1, \ldots, r_n]$ of rules in P such that for every i $(1 \leq i \leq n)$,*

1. *r_i is either a rule in P or a minimal reductant in P.*
2. *For every annotated atom $A{:}\mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $A{:}\rho$ $(\rho \geq \mu)$ is head of r_k.*
3. *For every o-explicit negation $\sim A{:}\mu$ in the body of r_i, there exists a r_k $(n \geq k > i)$ such that $\sim A{:}\rho$ $(\rho \leq \mu)$ is head of r_k.*
4. *There exists no proper subsequence of $[r_1, \ldots, r_n]$ which meets from the first to the third conditions, and includes r_1.*

We denote the set of all arguments in P by $Args_P$, and define the set of all arguments in a set of EALPs $MAS = \{KB_1, \ldots, KB_n\}$ by $Args_{MAS} = Args_{KB_1} \cup \cdots \cup Args_{KB_n}$ $(\subseteq Args_{KB_1 \cup \cdots \cup KB_n})$.

3.2 Attack relation

Definition 11. (**Rebut**). *Arg_1 rebuts $Arg_2 \Leftrightarrow$ there exists $A{:}\mu_1 \in concl(Arg_1)$ and $\sim A{:}\mu_2 \in concl(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A{:}\mu_1 \in concl(Arg_1)$ and $A{:}\mu_2 \in concl(Arg_2)$ such that $\mu_1 \leq \mu_2$.*

Definition 12. (**Undercut**). *Arg_1 undercuts $Arg_2 \Leftrightarrow$ there exists $A : \mu_1 \in concl(Arg_1)$ and $\mathbf{not}\, A : \mu_2 \in assm(Arg_2)$ such that $\mu_1 \geq \mu_2$, or exists $\sim A : \mu_1 \in concl(Arg_1)$ and $\mathbf{not}\ \sim A{:}\mu_2 \in assm(Arg_2)$ such that $\mu_1 \leq \mu_2$.*

Definition 13. (**Strictly undercut**). *Arg_1 strictly undercuts $Arg_2 \Leftrightarrow Arg_1$ undercuts Arg_2 and Arg_2 does not undercut Arg_1.*

Definition 14. (**Defeat**). *Arg_1 defeats $Arg_2 \Leftrightarrow Arg_1$ undercuts Arg_2, or Arg_1 rebuts Arg_2 and Arg_2 does not undercut Arg_1.*

When an argument defeats itself, such an argument is called a *self-defeating argument*. For example, $[p : \mathbf{t} \leftarrow \mathbf{not}\, p : \mathbf{t}]$ and $[q : \mathbf{f} \leftarrow\!\sim q : \mathbf{f},\ \ \sim q : \mathbf{f}]$ are all self-defeating. In this paper, however, we rule out self-defeating arguments from argument sets since they are in a sense abnormal, and not entitled to participate in argumentation or dialogue.

Definition 15. (x/y-**acceptable and justified argument** [4]). *Let x and y be attack relations on $Args$. Suppose $Arg_1 \in Args$ and $S \subseteq Args$. Then Arg_1 is x/y-acceptable wrt. S if for every $Arg_2 \in Args$ such that $(Arg_2, Arg_1) \in x$ there exists $Arg_3 \in S$ such that $(Arg_3, Arg_2) \in y$. The function $F_{Args,x/y}$ mapping from $\mathcal{P}(Args)$ to $\mathcal{P}(Args)$ is defined by $F_{Args,x/y}(S) = \{Arg \in Args \mid Arg$ is x/y-acceptable wrt. $S\}$. We denote a least fixpoint of $F_{Args,x/y}$ by $J_{Args,x/y}$. An*

argument Arg is x/y-justified if $Arg \in J_{x/y}$; an argument is x/y-overruled if it is attacked by a x/y-justified argument; and an argument is x/y-defensible if it is neither x/y-justified nor x/y-overruled.

In this paper, we employ $J_{Args,d/su}$ to specify the set of justified arguments where stands d for defeat and su for strictly undercut. Justified arguments can be dialectically determined from a set of arguments by the dialectical proof theory.

Definition 16. (d/su**-dialogue** [14]). *An d/su-dialogue is a finite nonempty sequence of moves $move_i = (Player_i, Arg_i), (i \geq 1)$ such that*

1. *$Player_i = P$ (Proponent) iff i is odd; and $Player_i = O$ (Opponent) $\Leftrightarrow$ i is even.*
2. *If $Player_i = Player_j = P$ $(i \neq j)$ then $Arg_i \neq Arg_j$.*
3. *If $Player_i = P$ $(i \geq 3)$ then $(Arg_i, Arg_{i-1}) \in su$; and if $Player_i = O$ $(i \geq 2)$ then $(Arg_i, Arg_{i-1}) \in d$.*

In this definition, it is permitted that $P = O$, that is a dialogue is done by only one agent. Then, we say such an argument is a self-argument.

Definition 17. (d/su**-dialogue tree** [14]). *An d/su-dialogue tree is a tree of moves such that every branch is an d/su-dialogue, and for all moves $move_i = (P, Arg_i)$, the children of $move_i$ are all those moves $(O, Arg_{i+1,j})$ $(j \geq 1)$ such that $(Arg_{i+1,j}, Arg_i) \in d$.*

We have the sound and complete dialectical proof theory for the argumentation semantics $J_{Args,d/su}$ [14]. In the learning process described in the next section, we will often take into account deliberate or thoughtful agents who put forward deliberate arguments in the dialogue.

4 Learning by Argumentation

We have outlined notions and definitions provided in EALP and LMA that are to be underlain in considering learning by argumentation. The most common form of machine learning is learning from examples, data and cases such as in inductive learning [13]. There are some argumentation-related learning methods [5][8] . They, however, are concerned with introducing traditional learning methods from examples. From this section, we will address ourselves to a new approach to machine learning that draws on some notions and techniques of EALP and LMA. Although there are so many aspects, methods and techniques already known on learning [13], our motivation for machine learning comes from argumentation since we learn and grow through argumentation or dialogue with our partners, friends, colleagues or even enemies in the daily life and scientific communities, as well as through self-deliberation that can be though of as self-argumentation. Actually, we benefit a lot from argumentation, and we believe argumentation is a desideratum to learning.

In this paper, we propose three basic approaches to learning by argumentation, which naturally reflect our intuitions and experiences that we have had in

the daily life so far. They are conceptually methods: (i) to correct wrong knowledge, (ii) to reconsider, (iii) to have a second thought, through argumentation. These are considered exhaustive in its types of argument-based learning on the basis that the learning process is presumably triggered by the attack relation such as the rebut and undercut of LMA. Below, let's take up simple but natural arguments to see shortly what they are like.

(i) *Correct wrong knowledge:* Here is an argument on a soap to slim between Mr. A and Mr. B. They argue about whether the soap to slim works or not.

Mr. A: I do not have experienced its effect, but I think that it is effective because TV commercial says so.

Mr. B: I have not become thin.

Mr. A: Now that you haven't, I may not become thin either.

After such an argumentation, we as well as Mr. A would usually correct or change our previous belief that the soap is effective, into its contrary. Like this, we may correct wrong knowledge and learn counter-arguments. Technically, the first assertion in Mr. A's locution is considered as having an assumption "the soap is effective to slim". And Mr. B argues against Mr. A. It amounts to *undercut* in terms of LMA. In the next subsection 4.1, we formally capture this type of learning by argumentation, calling it *knowledge acquisition induced by the undercut of assumptions*.

(ii) *Reconsider:* Let's consider the evaluation of a movie.

Mr. C: The story of the movie is so fantastic! I recommend it.

Mr. D: The performance of the actors in the movie is unskilled, so I do not recommend it.

Agent D states an opinion contrary to Agent C, but does not intend to refuse and take back Agent A's opinion. In the dialogue, they simply state their own opinion on the evaluation of the movie. They are not necessarily in a conflict with each other, and simply made it sure that they had a contrary opinion on the matter. Through the dialogue, they will know or learn that there are facets or aspects on the movie that can be evaluated and can not. In the subsection 4.2, we formally capture this type of learning by argumentation, calling it *knowledge acquisition induced by difference of recognition*.

(iii) *Have a second thought:* Let's see the third type of learning by argumentation with an argument on which is correct, the Copernican theory or Ptolemaic theory.

Mr. E: I agree with the Ptolemaic theory because we see the Sun go around us, and the Bible also tells us so.

Mr. F: I agree with the Copernican theory because the Earth moves according to our observation.

People who have believed the Ptolemaic theory may have a second thought if the Copernican theory is *justified* by a (scientific) argumentation. Or, they may reach such an eclectic conclusion that both the Ptolemaic theory and the Copernican theory are partial knowledge. In the subsection 4.3, we formally capture this type of learning by argumentation, calling it *knowledge acquisition induced by rebut*.

4.1 Knowledge acquisition induced by the undercut of assumptions

In this paper, we think that the momentum of knowledge acquisition or learning comes when agents recognize right and wrong of arguments. And we identify it with the notion of *justification* for arguments in Definition 15. The first learning definition based on it is the following.

Definition 18. (**Knowledge acquisition induced by the undercut of assumptions**). *Suppose* $KBs = \{KB_1, \ldots, KB_n\}$ *is a set of EALPs. We denote the set of all arguments in* KB_i *by* $Args_{KB_i}$ $(1 \leq i \leq n)$*, and* Arg *is an argument in* $Args_{KB_i}$*. Let* JA *be the set of justified arguments. If there exists an argument* $Arg' \in JA$ *such that it undercuts* Arg*, we say Agent* i *acquires* Arg'*, letting* $KB_i' = KB_i \cup \{Arg'\}$*.*

After argumentation, Agent i acquires all the rules included in Arg' with this definition.

Corollary 1. *The new knowledge base of* KB_i' *after knowledge acquisition induced by the undercut of assumptions is inconsistent in default (d-inconsistent), that is, the interpretation* I *such that* $\forall B \in KB_i' \models B$ *is d-inconsistent.*

The proof is straightforward, and importantly we do not need to have a fear of such an inconsistency since our EALP is advantageously paraconsistent [14]. If the underlying complete lattice of truth values is 1-wide, then we have

Corollary 2. *JA is preserved by knowledge acquisition induced by the undercut of assumptions.*

The proofs are straightforward. Taking up the previous argument example, we illustrate how the definition operates.

Example 1. Let $\mathcal{T} = \langle \mathcal{R}[0\ 1], \leq \rangle$ be a complete lattice on the unit interval of real numbers. Suppose Agent A and B have the following knowledge bases KB_A and KB_B on a soap to slim respectively.
KB_A = { $become\_slim : 0.8 \leftarrow medical\_rationale : 0.7 \,\&\, information\_from\_TV : 0.8 \,\&\, \mathbf{not}\, experience\_of\_effect : 0.0, medical\_rationale : 0.8 \leftarrow, information\_from\_TV : 0.9 \leftarrow$},
KB_B = { $become\_slim : 0.0 \leftarrow experience\_of\_effect : 0.0,\ experience\_of\_effect : 0.0 \leftarrow$}.
Then, the sets of arguments $Args_{KB_A}$ and $Args_{KB_B}$ are;
$Args_{KB_A}$ = {[$become\_slim : 0.8 \leftarrow medical\_rationale : 0.7 \,\&\, Information\_from\_TV : 0.8 \,\&\, \mathbf{not}\, experience\_of\_effect : 0.0, medical\_rationale : 0.8 \leftarrow, information\_from\_TV : 0.9 \leftarrow$], [$medical\_rationale : 0.8 \leftarrow$], [$information\_from\_TV : 0.9 \leftarrow$]},
$Args_{KB_B}$ = {[$become\_slim : 0.0 \leftarrow experience\_of\_effect : 0.0$, $experience\_of\_effect : 0.0 \leftarrow$], [$experience\_of\_effect : 0.0 \leftarrow$]}.
These are representations of verbal and natural arguments described in (i) *Correct wrong knowledge* above. The set of justified arguments JA is as follows.
JA = { [$become\_slim : 0.0 \leftarrow experience\_of\_effect : 0.0,\ experience\_of\_effect : 0.0 \leftarrow$], [$medical\_rationale : 0.8 \leftarrow$], [$information\_from\_TV : 0.9 \leftarrow$],

[$experience\_of\_effect : 0.0 \leftarrow$] }.

JA can be seen as a set of agreements on various issues among agents concerned. Agents then get down to acquiring knowledge with JA based on Definition 18. Suppose Agent A put forward the following argument Arg_1.
$Arg_1 =$ [$become\_slim : 0.8 \leftarrow medical\_rationale : 0.7 \,\&\, information\_from\_TV : 0.9$ $\&$ **not** $experience\_of\_effect : 0.0$, $medical\_rationale : 0.8 \leftarrow$, $information\_from\_TV : 0.9 \leftarrow$].
However, it can be seen that it is undercut by justified arguments $Arg_2 =$ [$experience\_of\_effect : 0.0 \leftarrow$]. Therefore, agent A acquires Arg_2, and builds a new knowledge base KB'_A as follows.
$KB'_A = \{$ $become\_slim : 0.8 \leftarrow medical\_rationale : 0.7 \,\&\, information\_from\_TV : 0.9 \,\&$ **not** $experience\_of\_effect : 0.0, medical\_rationale : 0.8 \leftarrow$, $experience\_of\_effect : 0.0 \leftarrow$, $information\_from\_TV : 0.9 \leftarrow\}$.

It is noted that agent A is no more entitled to put forward the previous argument Arg_1 with KB'_A since the newly added rule '$experience\_of\_effect : 0.0 \leftarrow$' immediately blocks it by undercut under deliberate argumentetion in Definition 17. Note also that the new knowledge base of an agent after argumentation does not coincide with the set of justified arguments JA that has been obtained before the learning process . For example, $KB'_A \neq JA$ in general. This means that the learning is a genuine process to raise a agent' mind under selective attention. This property also applies to the succeeding two knowledge acquisition approaches below.

4.2 Knowledge acquisition induced by difference of recognition

In this section, we describe the second learning method inspired by the notion of difference of recognition.

Definition 19. (**Difference of recognition**). *Let $KBs = \{KB_1, ..., KB_n\}$ be a set of EALPs, $Args_{KB_i}$ and $Args_{KB_k} (1 \leq i, k \leq n)$ be the sets of all arguments in KB_i and KB_k respectively, and Arg be an argument in $Args_{KB_i}$. If there exist $A : \mu_1 \in concl(Arg_i)$ and $A : \mu_2 \in concl(Arg_k)$ such that $\mu_1 \neq \mu_2$, agent i and agent k have different recognition about the proposition A.*

Example 2. Let a lattice $\mathcal{T} = \mathcal{R}[0, 1]$ and a multi-agents system $KB_1 = \{$ $p : 0.8 \leftarrow q : 0.4, q : 0.5 \leftarrow$ $\}$. $KB_2 = \{$ $p : 0.5 \leftarrow q : 0.2, q : 0.5 \leftarrow$ $\}$. $Args_{KB_1}$ and $Args_{KB_2}$ are basically the same as above as follows. $Args_{KB_1} = \{$ [$p : 0.8 \leftarrow q : 0.4, q : 0.5 \leftarrow$], [$q : 0.5 \leftarrow$] $\}$ $Args_{KB_2} = \{$ [$p : 0.5 \leftarrow q : 0.2, q : 0.5 \leftarrow$], [$q : 0.5 \leftarrow$] $\}$
Then, agent 1 and agent 2 have different recognition about p.

In argumentation, we did not pay attention to difference of recognition agents hold, which does not produce any conflict between them, but simply represents their own views mutually. From learning point of view, however, we suppose agents wish to know and learn the other party's opinion or view. Based on the two notions of difference of recognition and justified arguments, we capture this

by classifying it into three cases: (1) both Arg_1 and Arg_2 are justified, (2) either Arg_1 or Arg_2 is justified, and (3) neitherArg_1 nor Arg_2 is justified.

Definition 20. (**Knowledge acquisition induced by difference of recognition**) *Let $KBs = \{KB_1, ..., KB_n\}$ be a set of EALPs, $Args_{KB_i}$ and $Args_{KB_k}$ $(1 \leq i, k \leq n)$ be the sets of all arguments in KB_i and KB_k respectively, and $Arg_i \in Args_{KB_i}$ and $Arg_k \in Args_{KB_k}$ in which there exist $A{:}\mu_1 \in concl(Arg_i)$ and $A : \mu_2 \in concl(Arg_k)$ such that $\mu_1 \neq \mu_2$. JA denotes the set of justified arguments. Then,*

1. *if $Arg_i \in JA$ and $Arg_k \in JA$, agent i updates KB_i to $KB'_i = KB_i \cup Arg_{KB_k}$, and agent k updates KB_k to $KB'_k = KB_k \cup Arg_{KB_i}$;*
2. *if $Arg_{KB_i} \in JA$ and $Arg_{KB_k} \notin JA$, then agent k updates KB_k to $KB'_k = KB_k \cup Arg_{KB_i}$;*
3. *if $Arg_{KB_i} \notin JA$ and $Arg_{KB_k} \notin JA$, then agent i and k do not learn anything, resulting in no updates on their knowledge bases.*

Under this definition, agents or agents' attitude toward update are supposed to be credulous in the sense that they update their knowledge bases as far as arguments are justified.

Corollary 3. *The new knowledge base of KB' after knowledge acquisition induced by difference of recognition can be inconsistent in d-inconsistent or o-inconsistent, that is, the interpretation I such that $\forall B \in KB\ I \models KB'$ is d-inconsistent or o-inconsistent.*

Again we do not need to have a fear of such an inconsistency since our EALP is advantageously paraconsistent [14]. If the underlying complete lattice of truth values is 1-wide, then we have

Corollary 4. *JA is preserved by knowledge acquisition induced by difference of recognition.*

Example 3. Let $\mathcal{T} = \mathcal{FOUR}$, and $MAS = \{KB_A, KB_B\}$, where Agent A and B have the following knowledge bases on the evaluation of a movie.

$KB_A = \{\ recommend(movie){:}\mathbf{t} \leftarrow famous(actor){:}\mathbf{t}\ \&\ famous(story){:}\mathbf{t},$
$famous(actor){:}\mathbf{t} \leftarrow,\quad famous(story){:}\mathbf{t} \leftarrow\ \},$
$KB_B = \{\ recommend(movie){:}\mathbf{f} \leftarrow poor(actor){:}\mathbf{t}\ \&\ see(movie){:}\mathbf{t},$
$poor(actor){:}\mathbf{t} \leftarrow, see(movie){:}\mathbf{t} \leftarrow\ \}.$

Suppose they put forward the arguments Arg_A and Arg_B respectively.

$Arg_A = [\ recommend(movie){:}\mathbf{t} \leftarrow famous(actor){:}\mathbf{t}\ \&\ famous(story){:}\mathbf{t},$
$famous(actor){:}\mathbf{t} \leftarrow, famous(story){:}\mathbf{t} \leftarrow],$
$Arg_B = [\ recommend(movie){:}\mathbf{f} \leftarrow poor(actor){:}\mathbf{t}\ \&\ see(movie){:}\mathbf{t},$
$poor(actor){:}\mathbf{t} \leftarrow, see(movie){:}\mathbf{t} \leftarrow\ ].$

Then, there is no attack relation between them, so all arguments made from MAS are justified ($JA = Args_{KB_A} \cup Args_{KB_B}$). However, agent A and B have difference of recognition about $recommend(movie)$. So they go into the learning process of and get the new knowledge base KB'_A and KB'_B respectively.

$KB'_A = KB'_B = \{\ recommend(movie){:}\mathbf{t} \leftarrow famous(actor){:}\mathbf{t}\ \&\ famous(story){:}\mathbf{t},$
$recommend(movie){:}\mathbf{f} \leftarrow poor(actor){:}\mathbf{t}\ \&\ see(movie){:}\mathbf{t},$
$poor(actor){:}\mathbf{t} \leftarrow, see(movie){:}\mathbf{t} \leftarrow\ \ famous(actor){:}\mathbf{t} \leftarrow,\ \ famous(story){:}\mathbf{t} \leftarrow,\ \}$

The new set of justified argument *JA'* constructed from these new KB'_A and KB'_B includes the additional argument: [$recommend(movie):\top \leftarrow famous(actor):\mathbf{t}\,\&\,famous(story):\mathbf{t}\,\&\,poor(actor):\mathbf{t}\,\&\,see(movie):\mathbf{t}$, $famous(actor):\mathbf{t}\leftarrow$, $famous(story):\mathbf{t}\leftarrow, poor(actor):\mathbf{t}\leftarrow, see(movie):\mathbf{t}\leftarrow$]. This is due to the reductant constructed from two contrary propositions: $recommend(movie):\mathbf{t}$ and $recommend(movie):\mathbf{f}$. This fact also exemplifies the failure Corollary 4 since $\mathcal{T}$ is not 1-wide. In argumentation, both agents A and B only got on their soapbox, but they do not intend to exclude the other's argument. What they get to know through learning is that the movie has good and wrong points: $recommend(movie):\top$. In EALP, this does not mean a contradiction but a way of recognizing things. Agents now is in such an epistemic state.

4.3 Knowledge acquisition induced by rebut

In this subsection, we formally consider the third learning scheme that we have seen in an argument example on which is correct, the Copernican theory or Ptolemaic theory. In terms of LMA, it is a scheme induced by rebut since these two theories rebut each other. Then, we consider it by three cases similarly to Definition 20. First, we introduce a preliminary notion of *Agreement rule and Agreed composite argument*

Definition 21. (**Agreement rule and Agreed composite argument**). *Let $MAS = \{KB_1, ..., KB_n\}$ be a set of EALPs, $Args_{KB_i}$ $(1 \leq i \leq n)$ be the set of all arguments in KB_i, Arg_i and Arg_k be in $Args_{KB_i}$ and $Args_{KB_k}$ respectively, and JA be the set of justified argument.*

Suppose Arg_i=$[r_1^i, \ldots, r_n^i] \notin JA$ such that $r_1^i = A:\mu_1 \leftarrow L_1^i\,\&\,\ldots\,\&\,L_{n_i}^i$, and Arg_k =$[r_1^k \ldots, r_m^k] \in JA$ such that $r_1^k = \sim A:\mu_2 \leftarrow L_1^k\,\&\,\ldots\,\&\,L_{n_k}^k$, and $A:\mu_1$ and $\sim A:\mu_2$ rebut each other. Then, we call the following synthetic rule an agreement rule:
$A:\rho \leftarrow L_1^i\,\&\,\ldots\,\&\,L_{n_i}^i\,\&\,L_1^k\,\&\,\ldots\,\&\,L_{n_k}^k$ *for some ρ such that $\rho < \mu_2$. And the following is called an agreed composite argument:*
$[A:\rho \leftarrow L_1^i\,\&\,\ldots\,\&\,L_{n_i}^i\,\&\,L_1^k\,\&\,\ldots\,\&\,L_{n_k}^k; (Arg_1 \setminus r_1^i); (Arg_2 \setminus r_1^k)]$,
where the semicolon denotes the list concatenation.

This definition is given relying upon the notion of justified arguments like the previous definitions of learning. Let us see an intuitive meaning of the agreement rule. Suppose $[\sim A:\mathbf{t}] \in JA$ and $[A:\mathbf{t}] \notin JA$ under the complete lattice of ideals of $\mathcal{FOUR}$. The regions of ideals as truth values [14] for two conflicting literals $A:\mathbf{t}$ and $\sim A:\mathbf{t}$ and the agreement region for both $A:\mathbf{t}$ and $\sim A:\mathbf{t}$ is seen in Figure 1. That is, the agent who asserts $[A:\mathbf{t}]$ could modifies its rule $A:\mathbf{t}$ down to $A:\bot$. The ρ may be arbitrary as far as it is less than μ_2.

Example 4. Let $\mathcal{T} = \mathcal{R}[0,\ 1]$, $KB_1 = \{p:0.8 \leftarrow q:0.4\,\&\,\mathbf{not}\,r:0.1, q:0.5\}$, and $KB_2 = \{\sim p:0.6 \leftarrow r:0.1, r:0.5\}$. Then, the arguments are $Arg_1 = [p:0.8 \leftarrow q:0.4\,\&\,\mathbf{not}\,r:0.1, q:0.5]$ and $Arg_2 = [\sim p:0.6 \leftarrow r:0.1, r:0.5]$. After argumentation, we have $Arg_1 \notin JA$, $Arg_2 \in JA$, and hence an agreed composite argument, $[p:\rho \leftarrow q:0.4\,\&\,r:0.1, q:0.5, r:0.5]$ for $\rho < 0.6$.

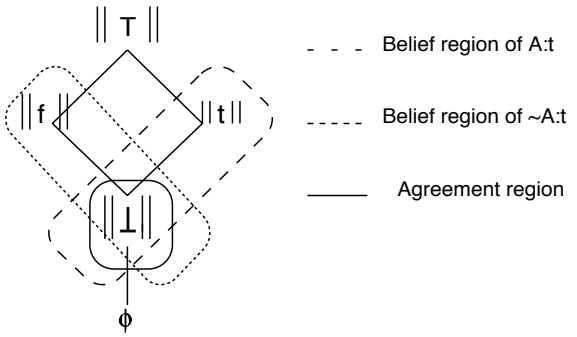


Fig. 1. Relation of belief regions, where $\|\mu\| = \{\rho \in \mathcal{T} \mid \rho \leq \mu\}$)

Using the notions of ACA, we give a method of knowledge acquisition induced by rebut.

Definition 22. (**Knowledge acquisition induced by rebut**) *Let* $MAS = \{KB_1, ..., KB_n\}$ *be a set of EALPs,* $Args_{KB_i}$ $(1 \leq i \leq n)$ *be the set of all arguments in* KB_i*,* Arg_i *and* Arg_k *be in* $Args_{KB_i}$ *and* $Args_{KB_k}$ *respectively, and JA be the set of justified argument. Suppose* $Arg_i = [r_1^i, \ldots, r_n^i]$ *such that* $r_1^i = A{:}\mu_1 \leftarrow L_1^i \,\&\, \ldots \,\&\, L_{n_i}^i$*, and* $Arg_k = [r_1^k \ldots, r_m^k]$ *such that* $r_1^k = {\sim} A{:}\mu_2 \leftarrow L_1^k \,\&\, \ldots \,\&\, L_{n_k}^k$*, and* $A{:}\mu_1$ *and* ${\sim} A{:}\mu_2$ *rebut each other. Then,*

1. *if* $Arg_i \in JA$ *rebuts* $Arg_k \notin JA$*, then* $KB'_k = KB_k \cup Arg_i \setminus r_1^k$*;*
2. *if* $Arg_i \notin JA$ *rebuts* $Arg_k \in JA$*, then agent i makes an agreed composite argument ACA from* Arg_i *and* Arg_k*, and* $KB'_i = KB_i \cup ACA \setminus \{r_1^i\}$*;*
3. *if* $Arg_i \notin JA$ *rebuts* $Arg_k \notin JA$*, agents i and k do not learn anything, resulting in no change in their knowledge bases.*

Example 5. Consider $\mathcal{T} = \{1, 2, ..., 10\}$, and $MAS = \{KB_A, KB_B, KB_C\}$, where $KB_A = \{\, recommend(movie){:}8 \leftarrow good\_story{:}9 \,\&\, \mathbf{not}\, expensive(movie){:}7,\, good\_story{:}9 \leftarrow \,\}$, $KB_B = \{{\sim} recommend(movie){:}2 \leftarrow skilled\_actor{:}3,\, skilled\_actor{:}3 \leftarrow \,\}$ $KB_C = \{recommend(movie){:}1 \leftarrow expensive(movie){:}8,\, expensive(movie){:}8 \leftarrow \,\}$. When these agents argue about the issue $recommend(movie){:}8$, agent B's argument and agent C's argument are justified. Following Definition 20, agent A obtains the following new knowledge:
$KB'_A = \{\, recommend(movie){:}1 \leftarrow actor{:}3 \,\&\, story{:}9 \,\&\, \mathbf{not}\, expensive{:}7,\, actor{:}3 \leftarrow ,\, story{:}9 \leftarrow \,\}$.

Without simply renouncing his belief, agent A still has his belief *recommend*(*movie*) but with a less truth value 1 than 2 of agent B and his original value 8 since the part of premises of the original rule, $story{:}\mathbf{t}$, is justified (in fact there is no objection to it). Furthermore, at the beginning of the argument, agent A has no information about the actor, but through the argumentation, he got to know about the actor. As the result, he degraded his belief *recommend*(*movie*), but still keeps it with a different truth value and an newly added premise. This type of learning looks very natural in our daily life as well.

5 Illustrative Examples of ABL

In this section, we describe two overall argument examples to show the effectiveness and usefulness of the agent learning methods based on argumentation.

Example 6. Agents 1, 2 and 3 are arguing about the next conference venue. As an unconventional but necessary complete lattice of truth values, we take the power set of a set consisting of symposium venues A, B, and C as elements, with set inclusion order. Their knowledge bases are as follows. $MAS = \{KB_1, KB_2, KB_3, \}$, where
$KB_1 = \{\ symposium:(A) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),\ symposium:(B) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),\ symposium:(C) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$scenic:(A,B) \leftarrow,\quad safe:(A,C) \leftarrow\quad \},$
$KB_2 = \{\ symposium:(B) \leftarrow easy\_access:(A,B),\ \sim symposium:(A) \leftarrow last\_venue:(A) \,\&\, \mathbf{not}\, tasty\_food:(A),\ easy\_access:(A,B) \leftarrow, last\_venue:(A) \leftarrow\quad \},$
$KB_3 = \{\ symposium:(A) \leftarrow tasty\_food:(A),\ \sim symposium:(C) \leftarrow \mathbf{not}\, easy\_access:(C),\ \sim symposium:(B) \leftarrow \mathbf{not}\, safe:(B),\ tasty\_food:(A) \leftarrow\quad \}.$

The literal, for example, $symposium:(A)$, reads "A is a venue candidate for the symposium", and $safe:(A,C)$ "A and C are safe places". The set of all possible arguments in this MAS is as follows.
$Arg_{11} = [\ symposium:(A) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),\ scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{12} = [\ symposium:(B) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{13} = [\ symposium:(C) \leftarrow scenic:(A,B) \,\&\, safe:(A,C).,$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{14} = [\ symposium:(A,B) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{15} = [\ symposium:(A,C) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{16} = [\ symposium:(B,C) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{17} = [\ symposium:(A,B,C) \leftarrow scenic:(A,B) \,\&\, safe:(A,C),$
$\quad scenic:(A,B) \leftarrow, safe:(A,C) \leftarrow\ ],$
$Arg_{18} = [\ scenic:(A,B) \leftarrow\ ],\qquad Arg_{19} = [\ safe:(A,C) \leftarrow\ ]$
$Arg_{21} = [\ symposium:(B) \leftarrow easy\_access:(A,B),\ easy\_access:(A,B) \leftarrow\ ],$
$Arg_{22} = [\ \sim symposium:(A) \leftarrow last\_venue:(A) \,\&\, \mathbf{not}\, tasty\_food:(A),$
$\quad last\_venue:(A) \leftarrow\ ],$
$Arg_{23} = [\ easy\_access:(A,B) \leftarrow\ ],\qquad Arg_{24} = [\ last\_venue:(A) \leftarrow\ ],$
$Arg_{31} = [\ symposium:(A) \leftarrow tasty\_food:(A), tasty\_food:(A) \leftarrow\ ],$
$Arg_{32} = [\sim symposium:(C) \leftarrow \mathbf{not}\, easy\_access:(C)\ ],$
$Arg_{33} = [\ \sim symposium:(B) \leftarrow \mathbf{not}\, safe:(B)\ ],\ Arg_{34} = [\ tasty\_food:(A) \leftarrow\ ].$

The attack relation is shown in Figure 2, and from it we can construct the set of justified arguments, $\{Arg_{11}, Arg_{23}, Arg_{24}, Arg_{31}, Arg_{34}\}$. Overally, this says "There is no guarantee that B is safe, and C is not an easy to access place. The next symposium should be held at A which was the last symposium venue". We could see this

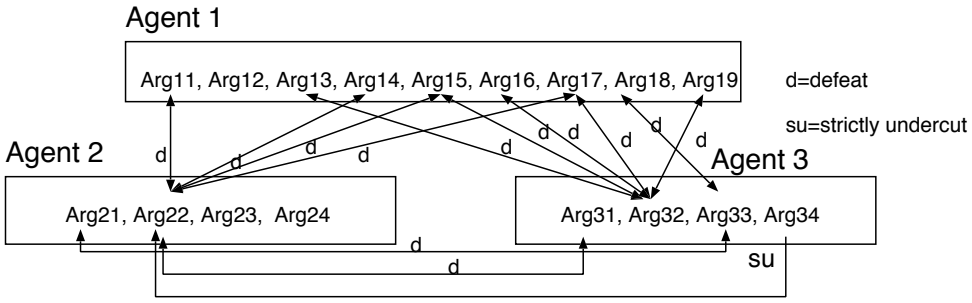


Fig. 2. Attack relation

as a current common belief (agreement) among the agents. Then, each agent goes into the phase of learning according to Definition 18, 20 and 22. Agent 1 gets two arguments(Arg_{33} and Arg_{32}) and loses two rules($symposium : (B) \leftarrow scenic : (A, B) \,\&\, safe : (A, C)$ and $symposium : (C) \leftarrow scenic : (A, B) \,\&\, safe : (A, C)$) based on Definition 22, and gets an argument Arg_{31} based on Definition 20. In the same way, Agents 2 and 3 revise their knowledge bases. The new knowledge bases KB'_1, KB'_2 and KB'_3 are as follows;
$KB'_1 = \{\ symposium : (A) \leftarrow scenic : (A, B) \,\&\, safe : (A, C),\ scenic : (A, B) \leftarrow,\ safe : (A, C) \leftarrow,\ tasty\_food : (A) \leftarrow,\ symposium : (A) \leftarrow tasty\_food : (A),\ \sim symposium : (C) \leftarrow \mathbf{not}\ easy\_access : (C),\ \sim symposium : (B) \leftarrow \mathbf{not}\ safe : (B)\ \}$,
$KB'_2 = \{\ symposium : (A) \leftarrow scenic : (A, B) \,\&\, safe : (A, C),\ \sim symposium : (B) \leftarrow \mathbf{not}\ safe : (B),\ scenic : (A, B) \leftarrow,\ safe : (A, C) \leftarrow,\ easy\_access : (A, B) \leftarrow,\ last\_venue : (A) \leftarrow,\ tasty\_food : (A) \leftarrow\ \}$,
$KB'_3 = \{\ symposium : (A) \leftarrow tasty\_food : (A),\ \sim symposium : (C) \leftarrow \mathbf{not}\ easy\_access : (C),\ \sim symposium : (B) \leftarrow \mathbf{not}\ safe : (B),\ tasty\_food : (A) \leftarrow,\ symposium : (A) \leftarrow scenic : (A, B) \,\&\, safe : (A, C),\ scenic : (A, B) \leftarrow,\ safe : (A, C) \leftarrow,\ tasty\_food : (A) \leftarrow\ \}$.

As the result of knowledge acquisition, any conflicting arguments disappear among Arg_{KB_A}, Arg_{KB_B} and Arg_{KB_C} as far as they put forward only arguments justified within their knowledge bases. This represents a stable and calm state of the agent society to which they reached after both argumentation and learning, and hence in which they have no need to argue and make decisions for the moment.

Next, we describe a dynamically changing argument example in which agents are involved in not only a single argument at a time but a process of consecutive arguments over time, and agents gradually become wise through them. This suggests an interesting and important direction to which argumentation studies head from now since acquisition not only ends once and for all, but also it continues repeatedly every time new information are found and added, and new agents appear. Similar observation can be seen in dialectic development of thought, society and so on in philosophy, and social processes of scientific development in philosophy of science.

Example 7. Let $\mathcal{T} = \mathcal{R}[0\ 1]$ be a complete lattice of the unit interval of real numbers. Consider the following multi-agents systems $MAS = \{KB_{Child}, KB_{Ptolemy}, KB_{Copernicus}\}$, where
$KB_{Child} = \{\ agree(Ptolemaic\_theory) : 0.0 \leftarrow, agree(Copernican\_theory) : 0.0 \leftarrow\ \}$,
$KB_{Ptolemy} = \{\ \sim agree(Copernican\_theory) : 1.0 \leftarrow bible : 1.0 \,\&\, \mathbf{not}\, move(Earth) : 1.0, agree(Ptolemaic\_theory) : 1.0 \leftarrow move(Sun) : 1.0, stay(Earth) : 0.2 \leftarrow \mathbf{not}\, move(Earth) : 0.0, \sim move(Earth) : 1.0 \leftarrow bible : 1.0, bible : 1.0 \leftarrow, move(Earth) : 0.0 \leftarrow, move(Sun) : 1.0 \leftarrow see(moving\_Sun) : 1.0, see(moving\_Sun) : 1.0 \leftarrow\ \}$,
$KB_{Copernics} = \{\sim agree(Ptolemaic\_theory) : 1.0 \leftarrow, agree(Copernican\_theory) : 1.0 \leftarrow move(Earth) : 1.0, move(Earth) : 1.0 \leftarrow observation : 0.8, move(Earth) : 1.0, move(Sun) : 1.0 \leftarrow, observation : 0.8 \leftarrow\}$.

First, consider the situation in which the child agent meets Ptolemy agent and argues about astronomy. Then, all arguments should be justified since as can be seen from its knowledge base, the child agent knows neither Ptolemaic theory nor Copernican theory. According to Definition 20, child agent's knowledge is simply increased as follows.
$KB'_{Child} = \{\ agree(Ptolemaic\_theory) : 0.0 \leftarrow, agree(Copernican\_theory) : 0.0 \leftarrow, \sim agree(Copernican\_theory) : 1.0 \leftarrow bible : 1.0 \,\&\, \mathbf{not}\, move(Earth) : 1.0, agree(Ptolemaic\_theory) : 1.0 \leftarrow move(Sun) : 1.0, stay(Earth) : 0.2 \leftarrow \mathbf{not}\, move(Earth) : 0.0, \sim move(Earth) : 1.0 \leftarrow bible : 1.0, bible : 1.0 \leftarrow, move(Earth) : 0.0 \leftarrow, move(Sun) : 1.0 \leftarrow see(moving\_Sun) : 1.0, see(moving\_Sun) : 1.0 \leftarrow\ \}$

We omit the change in Ptolemy agent's knowledge since we now are not concerned with it. The child agent becomes agreeable to the Ptolemaic theory through the argument with Ptolemy agent. Next, consider the situation in which the child agent meets Copernicus agent and argues about astronomy again with the new knowledge base above. As the result of argumentation, we have a new JA as follows.
$JA = \{\ [\ agree(Ptolemaic\_theory) : 0.0 \leftarrow\ ],\ [\ agree(Copernican\_theory) : 0.0 \leftarrow\ ],\ [\ agree(Copernican\_theory) : 1.0 \leftarrow move(Earth) : 1.0, move(Earth) : 1.0 \leftarrow\ ],\ [\ move(Earth) : 1.0 \leftarrow observation : 0.8, observation : 0.8 \leftarrow\ ],\ [\ move(Sun) : 1.0 \leftarrow see(moveing_Sun) : 1.0, see(moving\_Sun) : 1.0 \leftarrow\ ],\ [move(Sun) : 1.0 \leftarrow\ ],\ [bible : 1.0 \leftarrow\ ],\ [observation : 0.8 \leftarrow\ ],\ [move(Earth) : 0.0 \leftarrow\ ],\ [see(moving\_Sun) : 1.0 \leftarrow\ ]\ \}$.

Acceding to Definition 18 and 20, the child agent acquires the following new knowledge.
$KB''_{Child} = \{\ \ agree(Ptolemaic\_theory) : 0.0 \leftarrow, agree(Copernican\_theory) : 0.0 \leftarrow, agree(Ptolemaic\_theory) : 1.0 \leftarrow move(Sun) : 1.0, agree(Copernican\_theory) : 1.0 \leftarrow move(Earth) : 1.0, bible : 1.0 \leftarrow, move(Earth) : 0.0 \leftarrow, see(moving\_Sun) : 1.0 \leftarrow, move(Sun) : 1.0 \leftarrow see(moving\_Sun) : 1.0, move(Earth) : 1.0 \leftarrow observation : 0.8, observation : 0.8 \leftarrow\ \}$.

The child agent gets to believe both Ptolemaic theory and Copernican theory, that is, it possesses believable aspects in them. What we have presented here is said to be unsupervised learning, that is learning without teachers. We would say argumentation, in a sense, plays a role of teachers in a changing information space

over time. The order of argumentation and learning might bring us a different outcome in general, resulting in non-confluent property. For this example, the outcome coincides before and after the change of order in argumentation and learning.

6 Related Work

So far, much work has been devoted towards generic methods to update or revise knowledge bases avoiding contradictions caused by merging them or accommodating new information. However, there is few work with which we share our purpose of this paper in relation to argumentation, except [1] [2] [6]. In [1], Amgoud and Parsons propose a method to merge conflicting knowledge bases based on their preference-based argumentation framework. It allows arguments to be built not from a union of knowledge bases but from separate knowledge bases, and the arguments to then be merged. For example, supports of justified arguments can be safely merged without drawing inconsistency. In [2], Capobianco et al. think that the beliefs of agents are warranted goals computed by argumentation. They design the agents with ability to sense the changes in the environment and integrate them into their existing beliefs. Then, new perceptions always supersede old ones. This is a simple updating method, but in doing so, they introduce dialectical databases that is for storing arguments as precompiled knowledge to speed up argument construction when making arguments and responding in the future. In [6], Gómez et al. attempt to integrate their defeasible argumentation and the machine learning technique of neural networks. The latter is used to generate contradictory information that in turn is to be resolved by the former. This, however, is a work on a combination of existing learning techniques with argumentation, not an amalgamation of both. In the area of legal reasoning, we can find some works on argument construction from the past cases in legal data and knowledge base. Such a case-based legal reasoning shows another possibility of synergy of argumentation and machine learning. But it just have started.

Parsons, Wooldridge and Amgoud explore how the kinds of dialogue in which agents engage depend upon features of the agents themselves and then introduced assertion attitudes such as *confident, careful* and *thoughtful* and acceptance attitudes such as *credulous, cautious* and *skeptical* [10], to examine the effects of those features on the way in which agents determine what locutions can be made in the progress of a dialogue. Our agents are *confident* in their argumentation on the basis of the dialectical proof theory, but our learning policies at the end of an argument in this paper is similar to their notions of *thoughtful* and *skeptical* in the sense that ours are based on the set of justified arguments, JA. However, it does not mean that our learning agents should accept or acquire JA that include the knowledge of the other party in an unprincipled way even if they are part of JA. Otherwise, every agent engaged in an argument would become identical, resulting in the same knowledge base and hence the loss of its personality. This situation is not desirable in our view. In fact, we have intended to give three knowledge acquisition methods in such a way that knowledge base

after learning does not always coincide with JA. Put it differently, we would say that our learning agents are much more *deliberative* rather than thoughtful or skeptical. Paglieri and Casterfranchi claim that belief revision and argumentation should be grounded in cognitive processing of epistemic states and dynamics of agents [9]. This is an important direction to learning agents, but we think that the underlying framework EALP and LMA for ABL are comprehensive enough to take into consideration cognitive aspects of belief revision and argumentation. In fact, the second knowledge acquisition induced by difference of recognition shows one evidence to direct our work to such an attempt.

7 Concluding Remarks and Future Work

We provided three basic methods of learning towards argument-based learning (ABL). We think that they are unique in two senses. One is that they are not concerned with learning in a single agent framework but with learning in a multi-agents one where agents need to interact with other agents. The multi-agents learning naturally becomes more complex. The other is that they are built on the notions of attack relations in LMA and multiple-valuedness of knowledge in EALP, such as undercutting of assumptions, difference of recognition, and rebuts. Multiple-valued learning is more crucial and fruitful than two-valued case for uncertain environments in particular.

We also pointed out a dynamic nature of argumentation and learning, and showed a progressive argument example where the environment is dynamically changing, and hence arguments and learning have to be done every time new information are found, and new agents appear.

EALP is a very generic knowledge representation language for uncertain arguments, and LMA built on top of it also yields a generic argumentation framework so that it allows agents to construct uncertain arguments under truth values specified depending on application domains. For example, it includes Prakken and Sartor's ELP-based argumentation framework [11] that is now considered standard and well accepted, as a very simple special case of LMA. Therefore, our learning methods of this paper could have extensive applicability to many argumentation models [3]. Furthermore, we think that the learning methods under uncertain knowledge bases based on multiple-valuedness of LMA is a novel attempt worthy of special mention since they turn to include unique ones proper to LMA as well.

A prototype implementation of an argument-based learning system is now going on in such a way that it is incorporated into the existing automated argument system based on EALP and LMA (*http://www.cs.ie.niigata-u.ac.jp/˜sawamura/DEMO/aamas_demo.html*).

Finally, we just mention worthy to pursuit future research directions. Our knowledge acquisition approaches are not intended to be used in any situation. The application of each of them is related to the type of the dialogue [15] occurring among agents. The detailed analysis, however, will be left to the future work. Learning argument structures or strategies is naturally done by us in the

daily life and an important aspect of learning related to argumentation as well. This, in general, is called topica, a set of topos, which dates back to ancient Greek and can be seen in Aristotle's Rhetoric, turning our eyes to philosophy. Argumentation is a special apparatus of dialogue. In the next stage, we will address to learning through dialogue from a broader angle.

References

1. L. Amgoud and S. Parsons. An argumentation framework for merging conflicting knowledge base. In *Proc. of the 8th European Conference on Logics in Artificial Intelligence, JELIA'2002, In LNCS*, volume 2424, 2002.
2. M. Capobianco, C. I. Chesñevar, and G. R. Simari. An argument-based framework to model an agent's beliefs in a dynamic environment. In *Proceedings of the First International Workshop on Argumentation in Multiagent Systems, AAMAS2004 Conference*, pages 163–178, 2004.
3. C. I. Chesñevar, G. Maguitman, and R. P. Loui. Logical models of argument. *ACM Computing Surveys*, 32:337–383, 2000.
4. P. M. Dung. An argumentation semantics for logic programming with explicit negation. In *Proc. of 10th Int. Conference on Logic Programming*, pages 616–630, 1993.
5. S. A. Gómez and C. I. Chesñevar. Integrating defeasible argumentation and machine learning techniques. In *Proc. of WICC*. ACM, 2003.
6. S. A. Gómez and C. I. Chesñevar. A hybrid approach to pattern classification using neural networks anddefeasible argumentation. In *Proc. of the International FLAIRS 2004 Conference*, pages 393–398. AAAI press, 2004.
7. M. Kifer and V. S. Subrahmanian. Theory of generalized annotated logic programming and its applications. *J. of Logic Programming*, 12:335–397, 1992.
8. M. Možina, J. Žabkar, T. Bench-Capon, and I. Bratko. Application of argument based machine learning to law. In *Proc. of the 10th International Conference on AI and Law*, pages 248–249. ACM press, 2005.
9. F. Paglieri and C. Castelfranchi. Revising beliefs through arguments: Bridging the gap between argumentation and belief revision in mas. In *Proc. of First International Workshop on Argumentation in Multi-Agent Systems*, volume 1. Springer-Verlag, 2004.
10. S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *J. Logic Computat*, 13(3), 2003.
11. H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *J. of Applied Non-Classical Logics*, 7(1):25–75, 1997.
12. H. Prakken and G. Vreeswijk. Logical systems for defeasible argumentation. In *In D. Gabbay and F. Guenther, editors, Handbook of Philosophical Logic*, pages 219–318. Kluwer, 2002.
13. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 1995.
14. T. Takahashi and H. Sawamura. A logic of multiple-valued argumentation. In *Proceedings of the third international joint conference on Autonomous Agents and Multi Agent Systems (AAMAS'2004)*, pages 800–807. ACM, 2004.
15. D. Walton. *The New Dialectic: Conversational Contexts of Argument*. Univ. of Toronto Press, 1998.

Arguments and Couterexamples in Case-based Joint Deliberation

Santiago Ontañón[1] and Enric Plaza[2]

[1] MAIA, Department of Applied Mathematics and Analysis
UB, University of Barcelona, Gran Via de les Corts Catalanes, 585
08007 Barcelona, Catalonia, Spain.
santi@maia.ub.es
[2] IIIA, Artificial Intelligence Research Institute
CSIC, Spanish Council for Scientific Research
Campus UAB, 08193 Bellaterra, Catalonia, Spain.
enric@iiia.csic.es

Abstract. Multiagent learning can be seen as applying ML techniques to the core issues of multiagent systems, like communication, coordination, and competition. In this paper, we address the issue of learning from communication among agents circumscribed to a scenario with two agents that (1) work in the same domain using a shared ontology, (2) are capable of learning from examples, and (3) communicate using an argumentative framework. We will present a two fold approach consisting of (1) an argumentation framework for learning agents, and (2) an individual policy for agents to generate arguments and counterarguments (including counterexamples). We focus on argumentation between two agents, presenting (1) an interaction protocol (AMAL2) that allows agents to learn from counterexamples and (2) a preference relation to determine the joint outcome when individual predictions are in contradiction. We present several experiment to asses how joint predictions based on argumentation improve over individual agent's prediction.

1 Introduction

Argumentation frameworks for multiagent systems can be used for different purposes like joint deliberation, persuasion, negotiation, and conflict resolution. In this paper, we focus on argumentation-based joint deliberation among learning agents. Argumentation-based joint deliberation involves discussion over the outcome of a particular situation or the appropriate course of action for a particular situation. Learning agents are capable of learning from experience, in the sense that past examples (situations and their outcomes) are used to predict the outcome for the situation at hand. However, since individual agents experience may be limited, individual knowledge and prediction accuracy is also limited. Thus, learning agents that are capable of arguing their individual predictions with other agents may reach better prediction accuracy after such an argumentation process.

In this paper we address the issue of joint deliberation among two learning agents using an argumentation framework. Our assumptions are that these two agents work in the same domain using a shared ontology, they are capable of learning from examples, and they interact following a specific interaction protocol. In this paper, we will propose an argumentation framework for learning agents, and an individual policy for agents to generate arguments and counterarguments.

Existing argumentation frameworks for multiagent systems are based on deductive logic. An argument is seen as a logical statement, while a counterargument is an argument offered in opposition to another argument [5, 17]. However, these argumentation frameworks are not designed for learning agents, since they assume a fixed knowledge base. Learning agents, however may induce several generalizations that are consistent with the examples seen at a particular moment in time; the *bias* of the generalization technique used determines which of the valid generalizations is effectively hold by a learning agent.

Having learning capabilities allows agents a new form of counterargument, namely the use of *counterexamples*. Counterexamples offer the possibility of agents learning *during* the argumentation process, and thus improving their performance (both individual and joint performance). Moreover, learning agents will allow us to design individual agent policies to generate adequate arguments and counterarguments. Existing argumentation frameworks mostly focus on how to deal with contradicting arguments, while few address the problem of how to generate adequate arguments (but see [17]). Thus, they focus on the issue defining a preference relation over two contradicting arguments; however for learning agents we will need to address two issues: (1) how to define a preference relation over two conflicting arguments, and (2) how to define a policy to generate arguments and counterarguments.

In this paper we present a case-based approach to address both issues. The agents use case-based reasoning (CBR) to learn from past cases (where a case is a situation and its outcome) in order to predict the outcome of a new situation; moreover, the reasoning needed to support the argumentation process will also be based on cases. In particular, both the preference relation among arguments and the policy for generating arguments and counterarguments will be based on cases. Finally, we propose an interaction protocol called AMAL2 to support the argumentation process among two agents to reach a joint prediction over a specific situation or problem.

In the remainder of this paper we are going to introduce the multiagent CBR ($\mathcal{M}$AC) framework in which we perform our research (Section 2). In this framework, Section 2.1 introduces the idea of justified predictions. After that, Section 3 provides a specific definition of arguments and counterarguments that we will use in the rest of the paper. Then, Section 4 defines a preference relation between contradicting arguments. Section 5 presents specific policies to generate both arguments and counterarguments. Using the previous definitions, Section 6 presents a protocol called AMAL2 to allow two agents to solve a problem in a collaborative way using argumentation. Finally, Section 7 presents an empir-

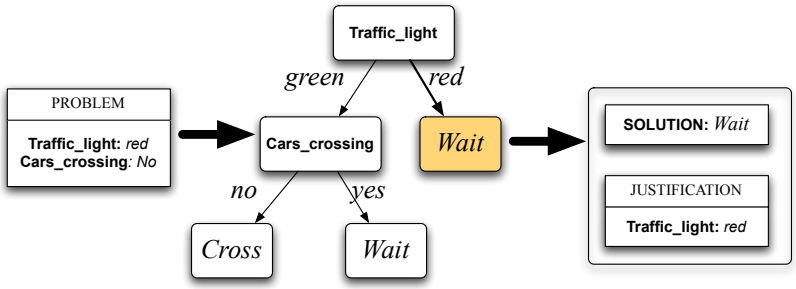


Fig. 1. An example of justification generation using a decision tree.

ical evaluation of the argumentation protocol presented. The paper closes with related work and conclusions sections.

2 Case-Based Multiagent Learning

In this section we are going to define the multiagent learning framework in which our research is performed [14].

Definition 1. *A* MultiAgent Case Based Reasoning System *(*$\mathcal{M}$AC*)* $\mathcal{M} = \{(A_1, C_1), ..., (A_n, C_n)\}$ *is a multiagent system composed of* $\mathcal{A} = \{A_i, ..., A_n\}$*, a set of CBR agents, where each agent* $A_i \in \mathcal{A}$ *possesses an individual case base* C_i*.*

Each individual agent A_i in a $\mathcal{M}$AC is completely autonomous and each agent A_i has access only to its individual and private case base C_i. A case base $C_i = \{c_1, ..., c_m\}$ is a collection of cases. Agents in a $\mathcal{M}$AC system are able to individually solve problems, but they can also collaborate with other agents to solve problem in a collaborative way.

In this framework, we will restrict ourselves to analytical tasks, i.e. tasks, like classification, where the solution of a problem is achieved by selecting a solution class from an enumerated set of solution classes. In the following we will note the set of all the solution classes by $\mathcal{S} = \{S_1, ..., S_K\}$. Therefore, a case is a tuple $c = \langle P, S \rangle$ containing a case description P and a solution class $S \in \mathcal{S}$. In the following, we will use the terms *problem* and *case description* indistinctly. Moreover, we will use the dot notation to refer to elements inside a tuple. e.g., to refer to the solution class of a case c, we will write $c.S$.

2.1 Justified Predictions

Many expert and CBR systems have an explanation component [18]. The explanation component is in charge of justifying why the system has provided a specific answer to the user. The line of reasoning of the system can then be examined by a human expert, thus increasing the reliability of the system.

Most of the existing work on explanation generation focuses on generating explanations to be provided to the user. However, in our approach we use explanations (or justifications) as a tool for improving communication and coordination among agents. We are interested in justifications to be used as arguments. For that purpose, we take benefit from the ability of some learning systems to provide justifications.

Definition 2. *A* justification *built by a CBR system to solve a problem P that has been classified into a solution class S_k is a description that contains the relevant information that the problem P and the retrieved cases $C_1, ..., C_n$ (all belonging to class S_k) have in common.*

For example, Figure 1 shows a justification build by a decision tree for a toy problem. In the figure, a problem has two attributes (traffic_light, and cars_crossing), after solving it using the decision tree shown, the predicted solution class is Wait. Notice that to obtain the solution class, the decision tree has just used the value of one attribute, traffic_light. Therefore, the justification must contain only the attribute/value pair shown in the figure. The values of the rest of attributes are irrelevant, since whatever their value the solution class would have been the same.

In general, the meaning of a justification is that all (or most of) the cases in the case base of an agent that satisfy the justification (i.e. all the cases that are *subsumed* by the justification) belong to the predicted solution class. In the rest of the paper, we will use $\sqsubseteq$ to denote the subsumption relation. In our work, we use LID [3], a CBR method capable of building symbolic justifications. LID uses the formalism of feature terms or ψ-terms) to represent cases [2].

We call *justified prediction* the justification for a prediction provided by a learning agent :

Definition 3. *A* justified prediction *is a tuple $\langle A, P, S, D\rangle$ containing the problem P, the solution class S found by the agent A for the problem P, and the justification D that endorses S as the correct solution for P.*

Justifications can have many uses for CBR systems [10, 13]. In this paper, we are going to use justifications as arguments, in order to allow agents to engage learning based argumentation processes.

3 Argumentation in Multiagent Learning

Let us start by presenting a definition of argument, that we will use in the rest of the paper:

Definition 4. *An* argument *α generated by an agent A is composed of a statement S and some evidence D supporting that S is correct.*

In the remainder of this section we will see how this general definition of argument can be instantiated in specific kind of arguments that the agents can generate. In the context of $\mathcal{M}$AC systems, agents argue about the correct solution of new problems and can provide information in two forms:

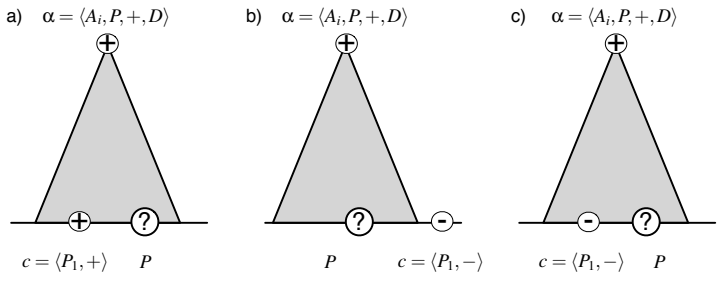


Fig. 2. Relation between cases and justified predictions. The case c is a counterexample of the justified prediction α in c), while it is not in a) and b).

- A specific case: $\langle P, S \rangle$,
- A justified prediction: $\langle A, P, S, D \rangle$.

In other words, agents can provide specific cases or generalizations learnt from cases. Using this information, and having in mind that agents will only argue about the correct solution of a given problem, we can define three types of arguments: justified predictions, counterarguments, and counterexamples.

- A *justified prediction* α is generated by an agent A_i to argue that A_i believes that the correct solution for a given problem P is $\alpha.S$, and the evidence provided is the justification $\alpha.D$. In the example depicted in Figure 1, an agent A_i may generate the argument $\alpha = \langle A_i, P, \mathsf{Wait}, (\mathsf{Traffic\_light} = \mathsf{red}) \rangle$, meaning that the agent A_i believes that the correct solution for P is Wait because the attribute Traffic_light equals red.
- A *counterargument* β is an argument offered in opposition to another argument α. In our framework, a counterargument consists of a justified prediction $\langle A_j, P, S', D' \rangle$ generated by an agent A_j with the intention to rebut an argument α generated by another agent A_i, that endorses a different solution class than α for the problem at hand and justifies this with a justification D'. In the example depicted in Figure 1, if an agent generates the argument $\alpha = \langle A_i, P, \mathsf{Walk}, (\mathsf{Cars\_crossing} = \mathsf{no}) \rangle$, an agent that thinks that the correct solution is Stop might answer with the counterargument $\beta = \langle A_j, P, \mathsf{Stop}, (\mathsf{Cars\_crossing} = \mathsf{no} \wedge \mathsf{Traffic\_light} = \mathsf{red}) \rangle$, meaning that while it is true that there are no cars crossing, the traffic light is red, and thus the street cannot be crossed.
- A *counterexample* c is a case that contradicts an argument α. Specifically, for a case c to be a counterexample of an argument α, the following conditions have to be met: $\alpha.D \sqsubseteq c$ and $\alpha.S \neq c.S$. Figure 2 illustrates the concept of a counterexample: justified predictions are shown above the triangles while the specific cases subsumed by the justified predictions are at the bottom of the triangles. Figure 2 presents three situations: In a) c is not a counterexample of α since the solution of c is the solution predicted by α; in b) c is not a

counterexample of α since c is not subsumed by the justification $\alpha.D$; finally, in c) c is a counterexample of α).

By exchanging arguments, counterarguments (including counterexamples), agents can argue about the correct solution of a given problem. However, in order to do so, they need a specific interaction protocol, a preference relation between contradicting arguments, and a decision policy to generate counterarguments (including counterexamples). In the following sections we will present these three elements.

4 Preference Relation

The argument that an agent provides might not be consistent with the information known to other agents (or even to some of the information known by the agent that has generated the justification due to noise in training data). For that reason, we are going to define a preference relation over contradicting justified predictions based on cases. Basically, we will define a *confidence* measure for each justified prediction (that takes into account the cases known by each agent), and the justified prediction with the highest confidence is the preferred one.

The confidence of justified predictions is assessed by the agents via an process of *examination of justifications*. The idea behind examination of justifications is to count how many of the cases in an individual case base *endorse* the justified prediction, and how many of them are counterexamples of that justified prediction. The more endorsing cases, the higher the confidence; and the more the counterexamples, the lower the confidence.

Specifically, to examine a justified prediction α, an agent obtains the set of cases contained in its individual case base that are subsumed by $\alpha.D$. The more of these cases that belong to the same solution class $\alpha.S$ predicted by α, the higher the confidence will be. After examining a justified prediction α, an agent A_i obtains the *aye* and *nay* values:

- $Y_\alpha^{A_i} = |\{c \in C_i |\ \alpha.D \sqsubseteq c.P \wedge \alpha.S = c.S\}|$ is the number of cases in the agent's case base *subsumed* by the justification $\alpha.D$ that belong to the solution class $\alpha.S$ proposed by α,
- $N_\alpha^{A_i} = |\{c \in C_i |\ \alpha.D \sqsubseteq c.P \wedge \alpha.S \neq c.S\}|$ is the number of cases in the agent's case base *subsumed* by justification $\alpha.D$ that *do not* belong to that solution class.

When two agents A_1 and A_2 want to assess the confidence on a justified prediction α made by one of them, each of them examines the arguments and sends the *aye* and *nay* values obtained to the other agent. Then, both agents have the same information and can assess the confidence value for the justified prediction as follows:

$$\mathsf{C}(\alpha) = \frac{Y_\alpha^{A_1} + Y_\alpha^{A_2} + 1}{Y_\alpha^{A_1} + Y_\alpha^{A_2} + N_\alpha^{A_1} + N_\alpha^{A_2} + 2}$$

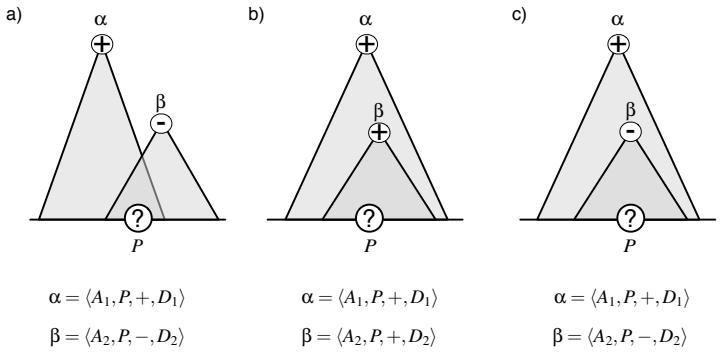


Fig. 3. Relation between arguments.

i.e. the confidence on a justified prediction is the number of endorsing cases divided by the number of endorsing cases plus counterexamples found by each of the two agents. Notice that we add 1 to the denominator, the reason is to avoid excessively high confidences to justified predictions whose confidence has been computed using a small number of cases (in this way, a prediction endorsed by 2 cases and with no counterexamples has a lower confidence than a prediction endorsed by 10 cases with no counterexamples). Notice that this correction follows the same idea than the Laplace correction to estimate probabilities (only that we are just interested on preventing overestimation of the confidence).

Thus, the preference relation used in our framework is the following one: a justified prediction α is preferred over another one β if $C(\alpha) \geq C(\beta)$.

5 Generation of Arguments

In our framework, arguments are generated by the agents using learning algorithms. Any learning method able to provide a justified prediction can be used to generate arguments. For instance, decision trees and LID [3] are suitable learning methods.

Thus, when an agent wants to generate an argument endorsing that a specific solution class is the correct solution for a given problem P, it generates a justified prediction as explained in Section 2.1.

When an agent A_i generates a counterargument β to rebut an argument α, A_i expects that β is preferred over α. With that purpose, in this section we are going to present a specific policy to generate counterarguments based on the *specificity* criterion [15].

The specificity criterion is widely used in deductive frameworks for argumentation, and states that between two conflicting arguments, the one using most specific information should be preferred. In our approach, counterarguments are generated based on the specificity criterion. However, there is no guarantee such

counterarguments will always win, since we use a preference relation based on confidence. Notice that we use the specificity criterion instead of the preference criterion defined in Section 4 because specificity can be evaluated by an agent alone, while the preference criterion requires the collaboration of both agents to be evaluated. Thus, it would require to the agent to engage in collaboration with the other one at each step of the search process, which is clearly unfeasible.

Therefore, when an agent wants to generate a counterargument β to an argument α, it will generate a counterargument that it is more specific than α. Figure 3 illustrates this idea. In Figure 3.c) β is a counterargument of α, and is more specific than α. However in Figure 3.a) β is not more specific than α and in Figure 3.c) both arguments endorse the same solution, and thus β is not a counterargument of α.

The generation of counterarguments using the specificity criterion imposes some restrictions over the learning method, although LID or ID3 can be easily adapted to generate counterarguments. For instance, to adapt LID we can do the following: LID is an algorithm that generates a description starting by the empty term and heuristically adding features to that term. Thus, at every step, the description is more specific, and the number of cases that are subsumed by that description is reduced. When the description only covers cases of a single solution class, LID terminates and predicts that solution class. To generate a counterargument to an argument α LID just has to use as starting point the description $\alpha.D$ instead of the empty term. In this way, the justification provided by LID will always be subsumed by $\alpha.D$, and thus the resulting counterargument will be more specific than α. However, notice that LID may sometimes not be able to generate counterarguments, since the description $\alpha.D$ may not be able to be specialized any further, or because the agent does not contain any cases subsumed by $\alpha.D$ to run LID.

Moreover, notice that agents can also try to rebut the other agent's arguments using counterexamples. Specifically, in our experiments, when an agent A_i wants to rebut an argument α, uses the following policy:

1. The agent A_i tries to generate a counterargument β more specific than α (in our experiments agents use LID). If the A_i succeeds, β is sent to the other agent as a counterargument of α.
2. If not, then A_i searches for a counterexample $c \in C_i$ of α in its individual case base C_i. If a case c is found, then c is sent to the other agent as a counterexample of α.
3. If no counterexamples are found, then A_i cannot rebut the argument α.

Notice that agents only send specific cases to each other if a counterargument cannot be found. To understand why have we done that, we must have in mind a known result in ensemble learning stating that when aggregating the predictions of several classifiers (i.e. agents) correlation between their predictions must be low in order to have a good classification accuracy [12]. Therefore, since when a counterexample is sent to the other agent the degree of correlation between the two agents' case bases increases, agents prefer to send a counterargument whenever possible, and only send a counterexample only when it is not.

The next section presents the interaction protocol we propose to perform argumentation in our learning framework.

6 Argumentation-based MultiAgent Learning

In this section we will present the Argumentation-based MultiAgent Learning Protocol for 2 agents (AMAL2). The idea behind AMAL2 is to allow a pair of agents to argue about the correct solution of a problem, arriving at a join solution that is based on their past learning and the information they exchange during argumentation.

At the beginning of the protocol, both agents will make their individual predictions for the problem at hand. Then, the protocol establishes rules allowing one of the agents in disagreement with the prediction of the other to provide a counterargument. Then, the other agent can respond with another counterargument, and so on.

In the remaining of this section we will present all the elements of the AMAL2 protocol. First, we will formally present the specific performatives that the individual agents will use in the AMAL2 protocol, that will allow them to *state* a prediction, to *rebut* an argument, and to *withdraw* an argument that the other agent's arguments have rendered invalid. Then, we will present the AMAL2 protocol.

6.1 Protocol Performatives

During the AMAL2 protocol, each agent will propose arguments and counterarguments to argue about which is the correct solution for a specific problem P. The AMAL2 protocol consists on a series of rounds. In the initial round, both agents state which are their individual predictions for P. Then, at each iteration an agent can try to rebut the prediction made by the other agent, or change its own prediction. Therefore, at each iteration, each of the two agents holds a prediction that it believes is the correct one.

We will use $H_t = \langle \alpha_1^t, \alpha_2^t \rangle$ to note the pair of predictions that each agent holds at a round t. When at a certain iteration an agent changes its mind and changes the prediction it is holding (because it has been convinced by the counterarguments of the other agent), it has to inform the other agent using the withdraw performative.

At each iteration, agents can send the following performatives to the other agent:

- $assert(\alpha)$: meaning that the justified prediction that the agent is holding for the next round will be α.
- $rebut(\alpha, \beta)$: meaning that the agent has found a counterargument or a counterexample α to the prediction β.
- $withdraw(\alpha)$: meaning that the agent is removing a justified prediction α, since the counterarguments presented by the other agent have rendered it invalid.

In the next section the AMAL2 protocol is presented that uses the performatives presented in this section.

6.2 Argumentation Protocol

The AMAL2 protocol among two agents A_1 and A_2 to solve a problem P works in a series of rounds. We will use t to denote the current round (initially $t = 0$). The idea behind protocol is the following one: initially, each agent makes its individual prediction. Then, the confidence of each prediction is assessed, and the prediction with the highest confidence is considered the winner. However, if the agent that has provided the prediction with lower confidence doesn't agree, it has the opportunity to provide a counterargument. Agents keep exchanging arguments and counterarguments until they reach an agreement or until no agent is able to generate more counterarguments. At the end of the argumentation, if the agents have not reached an agreement, then the prediction with the highest confidence is considered the joint prediction.

Notice that the protocol starts because one of the two agents receives a problem to be solved, and that agent sends that problem to the other agent requesting requesting to engage in an argumentation process. Thus, after both agents know the problem P to solve, round $t = 0$ of the protocol starts:

1. Initially, each one of the agents individually solves P, and builds a justified prediction (A_1 builds α_1^0, and A_2 builds α_1^0). Then, each agent A_i sends the performative $assert(\alpha_i^0)$ to the other agent. Thus, both agents know $H_0 = \langle \alpha_1^0, \alpha_2^0 \rangle$.
2. At each round t, the agents check whether their arguments in H_t agree. If they do the protocol moves to step 4, otherwise the agents compute the confidence for each argument and use the preference relation (presented in Section 4) to determine which argument in H_t is preferred. After that, the agent that has provided the non preferred argument may try to rebut the other agent's argument. Each individual agent uses its own policy to rebut arguments:
   - If an agent A_i generates a counterargument α_i^{t+1}, then it sends the following performatives to the other agent, A_j, in a single message: $rebut(\alpha_i^{t+1}, \alpha_j^t)$, $withdraw(\alpha_i^t)$, $assert(\alpha_i^{t+1})$. This message starts a new round $t + 1$, and the protocol moves back to step 2.
   - If an agent A_i selects c as a counterexample of the other agent's justified prediction, then A_i sends the following performative to the other agent, A_j: $rebut(c, \alpha_j^t)$. The protocol moves to step 3.
   - If no agent provides any argument the protocol moves to step 4.
3. The agent A_j that has received the counterexample c retains it and generates a new argument α_j^{t+1} that takes into account c. To inform A_i of the new argument, A_j sends A_i the following performatives: $withdraw(\alpha_j^t)$, $assert(\alpha_j^{t+1})$. This message starts a new round $t + 1$, and the protocol moves back to step 2.

4. The protocol ends yielding a joint prediction, as follows: if both arguments in H_t agree then their prediction is the joint prediction, otherwise the prediction in H_t with the higher confidence is considered the joint prediction.

Moreover, in order to avoid infinite iterations, if an agent sends twice the same argument or counterargument, the protocol also terminates.

Finally notice that when an agent A_i submits a counterargument α that defeats the other agent's argument, then α becomes A_i's argument, and thus the other agent may try to rebut it using another counterexample.

7 Experimental Evaluation

In this section we empirically evaluate the AMAL2 argumentation protocol. We have made experiments in two different data sets: *sponge*, and *soybean*. The sponge data set is a marine sponge classification problem, contains 280 marine sponges represented in a relational way and pertaining to three different orders of the Demospongiae class. The soybean data set is a standard data sets from the UCI machine learning repository, with 307 examples pertaining to 19 different solution classes.

In an experimental run, training cases are distributed among the agents without replication, i.e. there is no case shared by two agents. In the testing stage problems arrive randomly to one of the agents. The goal of the agent receiving a problem is to identify the correct solution class of the problem received.

Each experiment consists of a 10-fold cross validation run. An experiment consists of training and test phases as usual; during the training phase the training cases are distributed among the two agents in different ways, as we will see later. During the test phase learning is disabled, i.e. the agents cannot learn from one test case to the next (in order to evaluate all test cases uniformly). This is relevant here because the agents solving a test case can also learn from *other* cases (the counterexamples in the argumentation process). To keep test case uniformity the agents discard the cases learnt during the argumentation of a test case before moving to argue about the next test case.

Moreover, we have made experiments in four different scenarios: in the first scenario, a 100% of the cases of the training set are distributed among the agents; in the second scenario, the agents only receive a 75% of the training cases; in the third scenario, they only receive a 50%; finally in the fourth scenario agents only receive a 25% of the training cases. We have made those experiments to see how the argumentation protocol (and how the argument generation policies) work when the agents have different amount of data.

Figures 4.a and 4.b show the classification accuracy achieved by agents using the AMAL2 argumentation protocol in the sponge and soybean data sets. For each of the 4 scenarios (100%, 75%, 50% and 25%) three bars are shown: *individual*, *maxconf* and *AMAL2*. The *individual* bar represents the classification accuracy achieved by agents solving problems individually, the *maxconf* bar represents classification accuracy of the two agents using the following simple

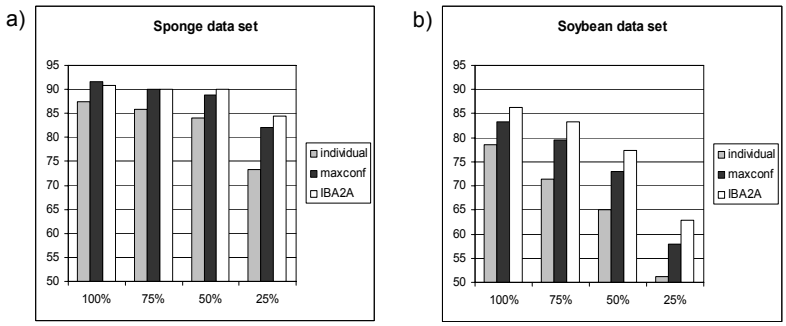


Fig. 4. Classification accuracy results in the Sponge and Soybean domains.

strategy: both agents solve the problem individually, then they evaluate the confidence of both predictions, and the prediction with the highest confidence is selected (notice that this is equivalent to using the AMAL2 protocol without any agent providing any counterargument). Finally, the AMAL2 bar represents the classification accuracy of the two agents using the AMAL2 protocol.

Figures 4.a and 4.b show several things. First, that using collaboration is always beneficial, since both *maxconf* and AMAL2 systematically outperform the individual agents in terms of accuracy. Moreover, both figures also show that the accuracy achieved by AMAL2 is higher than that of *maxconf* (in fact, AMAL2 is better or equal than *maxconf* in all the experiments except in the 100% scenario of the sponge data set). Moreover, the less data the individual agents have the greater the benefits of AMAL2 are. When each individual agent has enough data, then predictions and confidence estimations are reliable, and thus little or nothing is gained form the argumentation. However, when agents have access to limited data, the argumentation process helps them finding predictions that take into account more information, thus making the joint prediction more accurate.

To show that our approach is proficient we can compare our results with that of a single agent owning all the cases. In this "centralized" scenario the accuracy is 89.64% for the sponge data set, and 89.12% for the soybean data set. These results should be compared with the 100% scenarios, where individual agents achieve a much lower accuracy but using AMAL2 they achieve a comparable performance to that of the centralized approach. Specifically, in the sponges data set the accuracy of 89.64% goes down to 87.43% for individual agents, and using AMAL2 the accuracy is 90.86%, that recovers and even surpasses the centralized accuracy. In the soybean data set the accuracy of 89.12% goes down drastically to 78.63% for individual agents, and using AMAL2 the accuracy is 86.25%, that significantly recovers but not surpasses the centralized accuracy. The difference between these two data sets is that the soybean data set has a large number of classes and thus performance drastically diminishes when dividing the data set among two agents (since the likelihood of an agent having cases of each specific class diminishes). In practical terms this accuracy can be recovered by adding

redundancy to the case bases of the agents, i.e. allowing some duplicated cases (cases that are present in both case bases) [11].

Summarizing, collaborating agents (either using argumentation or the simple *maxconf* method) always increase their performance with respect to their individual performance. Similarly, using argumentation generally improves with respect to just using the simple *maxconf* aggregation function. However, when each individual agent has enough data, little is gained form the argumentation with respect to using *maxconf* aggregation function. Finally, when agents have access to limited data, there is ample opportunity for them to learn from communicating with another agent; the experiments reflect this hypothesis by the fact that argumentation in this situations increases performance to a larger degree.

8 Related Work

Research on MAS argumentation focus on several issues like a) logics, protocols and languages that support argumentation, b) argument selection and c) argument interpretation. Approaches for logic and languages that support argumentation include defeasible logic [5] and BDI models [17]. An overview of logical models of reasoning can be found at [6]. Moreover, the most related area of research is case-based argumentation. Combining cases and generalizations for argumentation has been already used in the HYPO system [4], where an argument can contain both specific cases or generalizations. Moreover, generalization in HYPO was limited to selecting a set of predefined dimensions in the system while our framework presents a more flexible way of providing generalizations. Furthermore, HYPO was designed to provide arguments to human users, while we focus on agent to agent argumentation. Case-based argumentation has also been implemented in the CATO system [1], that models ways in which experts compare and contrast cases to generate multi-case arguments to be presented to law students. Moreover, the goal of CATO differs from the goal of our work, since it is designed to allow law students to learn basic case-based argumentation law skills.

Concerning CBR in a multiagent setting, the first research was on "negotiated case retrieval" [16] among groups of agents. Our work on multiagent case-based learning started in 1999 [8]; later Mc Ginty and Smyth [9] presented a multiagent collaborative CBR approach (CCBR) for planning. Finally, another interesting approach is *multi-case-base reasoning* (MCBR) [7], that deals with distributed systems where there are several case bases available for the same task and addresses the problems of cross-case base adaptation. The main difference is that our $\mathcal{MAC}$ approach is a way to distribute the *Reuse* process of CBR (using a voting system) while *Retrieve* is performed individually by each agent; the other multiagent CBR approaches, however, focus on distributing the *Retrieve* process.

9 Conclusions and Future Work

In this paper we have presented a learning framework for argumentation. Specifically, we have presented AMAL2, a protocol that allows two agents to argue about the solution of a given problem. Finally, we have empirically evaluated it showing that the increased amount of information that the agents use to solve problems thanks to the argumentation process increases their problem solving performance, and specially when the individual agents have access to a limited amount of information. Clearly, an agent that knows all it needs does not need external help (nor, by the way, needs to continue learning if there is no room for improvement).

The main contributions of this work are: a) an argumentation framework for learning agents; b) a case based preference relation over arguments, based on computing a joint confidence estimation of arguments (this preference relation has sense in this learning framework since arguments are learnt from examples); c) a specific and efficient policy to generate arguments and counterarguments based on the specificity relation (commonly used in argumentation frameworks); d) a principled usage of counterexamples in the argumentation process, and e) a specific argumentation protocol for pairs of agents that collaborate to decide the joint solution of a given problem.

Moreover, the work presented in this paper concerns only pairs of agents. However, as future work we plan to generalize the AMAL2 protocol to work with a larger number of agents. A possibility to do that is a token based protocol where the agent owner of the token engages in a 1-to-1 argumentation dialog with every other agent that disagrees with its prediction. When all these 1-to-1 argumentation dialogs have finished, the token passes to the next agent. This process continues until no agent engages in any new 1-to-1 argumentation. Then, from the outcome of all the 1-to-1 argumentation processes, a joint prediction will be achieved just as now on step 4 of the AMAL2 protocol: either the agreed prediction or the one with higher conficence.

Acknowledgments. This research was partially supported by the CBR-ProMusic project TIC2003-07776-C02-02.

References

[1] Vincent Aleven. *Teaching Case-Based Argumentation Trough a Model and Examples*. PhD thesis, University of Pittsburgh, 1997.

[2] E. Armengol and E. Plaza. Bottom-up induction of feature terms. *Machine Learning Journal*, 41(1):259–294, 2000.

[3] E. Armengol and E. Plaza. Lazy induction of descriptions for relational case-based learning. In *Proceedings of the 10th European Conference on Machine Learning, ECML'2001*, pages 13–24, 2001.

[4] Kevin Ashley. Reasoning with cases and hypotheticals in hypo. *International Journal of Man-Machine Studies*, 34:753–796, 1991.

[5] Carlos I. Chesñevar and Guillermo R. Simari. Formalizing Defeasible Argumentation using Labelled Deductive Systems. *Journal of Computer Science & Technology*, 1(4):18–33, 2000.

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

[7] D. Leake and R. Sooriamurthi. Automatically selecting strategies for multi-case-base reasoning. In S. Craw and A. Preece, editors, *Advances in Case-Based Reasoning: Proceedings of the Fifth European Conference on Case-Based Reasoning*, pages 204–219, Berlin, 2002. Springer Verlag.

[8] Francisco J. Martín, Enric Plaza, and Josep-Lluis Arcos. Knowledge and experience reuse through communications among competent (peer) agents. *International Journal of Software Engineering and Knowledge Engineering*, 9(3):319–341, 1999.

[9] Lorraine McGinty and Barry smyth. Collaborative case-based reasoning: Applications in personalized route planning. In I. Watson and Q. Yang, editors, *ICCBR*, number 2080 in LNAI, pages 362–376. Springer-Verlag, 2001.

[10] Santi Ontañón and Enric Plaza. Justification-based multiagent learning. In *Int. Conf. Machine Learning (ICML 2003)*, pages 576–583. Morgan Kaufmann, 2003.

[11] Santi Ontañón and Enric Plaza. Justification-based case retention. In *European Conference on Case Based Reasoning (ECCBR 2004)*, number 3155 in LNAI, pages 346–360. Springer-Verlag, 2004.

[12] M. P. Perrone and L. N. Cooper. When networks disagree: Ensemble methods for hydrid neural networks. In *Artificial Neural Networks for Speech and Vision*. Chapman-Hall, 1993.

[13] Enric Plaza, Eva Armengol, and Santiago Ontañón. The explanatory power of symbolic similarity in case-based reasoning. *Artificial Intelligence Review*, 24(2):145–161, 2005.

[14] Enric Plaza and Santiago Ontañón. Ensemble case-based reasoning: Collaboration policies for multiagent cooperative cbr. In I. Watson and Q. Yang, editors, *In Case-Based Reasoning Research and Development: ICCBR-2001*, number 2080 in LNAI, pages 437–451. Springer-Verlag, 2001.

[15] David Poole. On the comparison of theories: Preferring the most specific explanation. In *IJCAI-85*, pages 144–147, 1985.

[16] M V Nagendra Prassad, Victor R Lesser, and Susan Lander. Retrieval and reasoning in distributed case bases. Technical report, UMass Computer Science Department, 1995.

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

[18] Bruce A. Wooley. Explanation component of software systems. *ACM CrossRoads*, 1998.

Argumentation and Persuasion in the Cognitive Coherence Theory: Preliminary Report

Philippe Pasquier[1], Iyad Rahwan[2,4], Frank Dignum[3], and Liz Sonenberg[1]

[1] University of Melbourne, Australia
[2] British University of Dubai, UAE
[3] Utrecht University, The Netherlands
[4] (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 proposition justifying or refuting the proposition expressed in the standpoint." [26, 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 has a way to refine multiagent communication [17, 11, 4, 3]. 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 [14, 15, 12] has been put forward as a way to model the cognitive aspects of agent communication pragmatics (section 2). Inspired from 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 cognitive

response to persuasive arguments using a single set of principles (section 4). Finally, the coverage of the proposed approach is discussed (section 5).

While at the beginning of this ongoing research work, this paper 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 [7] and the coherence theory from Thagard [23]. 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 [18], neural network or activation network based [20], probabilistic network [24], decision-theoretic, ...), we present one that is constraint satisfaction based resulting in a simple symbolic-connexionist hybrid formalism (we refer the reader to [22] 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 [7], 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[5]. 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:

[5] This is a way of prioritizing some cognitive constraints as it is done in the BOID architecture [1].

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 maximize the cognitive coherence degree of the considered set of elements.*

It is a constraint optimization problem shown to be NP-complete in [25]. 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, 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 $\mathcal{P} \cup \mathcal{B} \cup \mathcal{I} \cup SC$ 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 $\mathcal{A} \cup \mathcal{R} = \mathcal{P} \cup \mathcal{B} \cup \mathcal{I} \cup SC$.*

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 [16]). Social commitment has proven to be a powerful concept to capture the interdependencies between agents [21]. 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 [5].

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[6].

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 dialogue moves (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 Figure 1 and consists of four phases:

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[7]. 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 [12] for full justification and discussion of this algorithm. Traces of execution will be provided along with the examples in this paper.

[6] 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 behavior. Notice that integrating algorithms to progressively learn such probabilities is an obvious perspective of the presented model.

[7] n coherence calculus (sum over m constraints) for each level and a maximum of n levels to be searched.

Function LocalSearch(W)

Inputs: $W = \{\mathcal{P}, \mathcal{B}, \mathcal{I}, SC, \mathcal{C}^+, \mathcal{C}^-, \mathcal{A}, \mathcal{R}\}$; // current agent state
Outputs: List, $Change$; // ordered list of elements (change(s) to attempt).
Global:
Local:
Float, G, $Gval$, C, $Cval$; // Expected utility value of the best move;
Elements set, A', R';
Elements, y, x;
Agent, J; // Agent state buffer
Body:
for all $x \in \mathcal{P} \cup \mathcal{B} \cup \mathcal{I} \cup SC$ **do**
  if $x \in \mathcal{A}$ **then**
    $\mathcal{A}' := \mathcal{A} - \{x\}$; $\mathcal{R}' := \mathcal{R} \cup \{x\}$;
  else
    $\mathcal{R}' := \mathcal{R} - \{x\}$; $\mathcal{A}' := \mathcal{A} \cup \{x\}$;
  end if
  $W' := \{\mathcal{P}, \mathcal{B}, \mathcal{I}, SC, \mathcal{C}^+, \mathcal{C}^-, \mathcal{A}', \mathcal{R}'\}$;
  $G := C(W') - C(W) - Res(x)$; // Expected utility of flipping x
  $C := C(W') - C(W)$; // Pure coherence gain
  if $G > Gval$ **then**
    $J := W'$; $y := x$; $Gval := G$; $Cval := C$;
  end if
end for// Ends when (coherence is not raising anymore and the expected utility is not positive) or a social commitment need to be changed.
if ($Cval < 0$ and $Gval < 0$) or $y \in SC$ **then**
  Return $Change$;
else
  Update ($Res(y)$); Add (J,$Change$);
  LocalSearch(J);
end if

Fig. 1. Recursive specification of the local search algorithm.

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. The cognitive coherence framework provides the necessary mechanisms to answer (even partially) the following questions which are usually poorly treated in the AI and MAS literature:

1. *Why and when should agents converse?* Agents dialogue in order to try reducing incoherences they cannot reduce alone.
2. *When should an agent take a dialogue initiative, on which subject and with whom?* An agent engages in a dialogue when an incoherence appears that he cannot reduce alone. Whether because it is an external incoherence and he cannot accept or reject external cognitions on his own, or because it is an internal incoherence he fails to reduce alone. The subject of this dialogue

should thus focus on the elements which constitute the incoherence. The dialogue partners are the other agents involved in the incoherence if it is an external one or an agent he thinks could help him in the case of a merely internal incoherence.

3. *By which type of dialogue?* Even if we gave a general mapping of incoherence types toward dialogue types using Walton and Krabble typology in [14], the theory is generic enough to be applied to any conventional communicational framework. In [15], we gave the procedural scheme for this choice using DIAGAL [2] dialogue games as primitive dialogue types.
4. *How to define and measure the utility of a conversation?* As defined in section 2.2, the utility of a dialogue is the difference between the incoherence before and after this dialogue minus the cost of the dialogue moves.
5. *When to stop dialogue or, how to pursue it?* The dialogue stops when the incoherence is reduced[8] or, either it continues with a structuration according to the incoherence reductions chain. As dialogues are attempts to reduce incoherence, expected utility is used to choose between different competing dialogues moves (including dialogue initiative and dialogue ending).
6. *What are the impacts of the dialogue on agents' private cognitions?* In cases where dialogue, considered as an attempt to reduce an incoherence by working on the external world, definitively fails, the agent reduces the incoherence by changing his own mental attitudes in order to recover coherence (this is the attitude change process to be described in section 3).
7. *Which intensity to give to illocutionary forces of dialogue acts?* Evidently, the intensities of the illocutionary forces of dialogue/speech acts generated are influenced[9] by the incoherence magnitude. The more important the incoherence magnitude is, the more intense the illocutionary forces are.
8. *What are the impacts of the dialogue on agents' moods?* The general scheme is that: following the coherence principle, coherence is a source of satisfaction and incoherence is a source of dissatisfaction. We deduce emotional attitudes from internal coherence dynamic (happiness arises from successful reduction, sadness from failed attempt of reduction, fear from a future important reduction attempt, stress and anxiety from an incoherence persistence,...).
9. *What are the consequences of the dialogue on social relations between agents?* Since agents can compute and store dialogue utility, they can build and modify their relations with other agents in regard to their past dialogues. For example, they can strengthen relations with agents with whom past dialogues were useful, ...

All those dimensions of our theory - except 7, 8 and 9 - have been implemented and exemplified as presented and discussed in [13] and [15]. The presented practical framework relies on our dialogue games based agent communication language (DIAGAL) and our dialogue game simulator toolbox (DGS)[2].

[8] Note that this ending criterium is to be tempered with other external factors like time, resources and social norms. Those resources can be taken into account in the update of the resistance to change of various discussed elements.

[9] Actually, this is not the only factor, other factors could also matter: social role, hierarchical positions,...

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 behavior.

For contemporary psychologists, attitudes are the main components of cognition. These are the subjective preliminary to rational action [6]. Theoretically, an agent's behavior 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 behaviors or dialogical attempts to get the corresponding social commitments depending on their nature.

From another point of view, it could happen (due to hierarchies, power relations, value-based negotiation, argumentation,...) 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.

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 type of information disclosure. While in cooperative systems this information might be useful to help solving conflicts, or by making the negotiation and the convergence to a deal more efficient, it has been shown in [10] that argumentation and full cooperation is not necessarily always the best strategy for negotiation convergence. More generally, it is unclear if such information disclosure is worth in open system where heterogeneous and competitive (even malicious) agents can use this information to endorse non-cooperative behavior. In this paper, we won't address strategic issues related to argumentation.

In our framework, argumentation can be achieved by constraint propagation by introducing a syntactic facility that will allow the agents to send to one another parts of their elements and constraints networks. Previous work has been done around that idea in the field of distributed constraint satisfaction [9, 10].

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 $(H \cup \{h\}) \subset Arg_{\mathcal{I} \cup SC} \wedge (H \cup \{h\}) \cap SC \neq \varnothing$;*

In the cognitive coherence framework, argumentation will be used 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. Let's take an example to illustrate this argument generation systematics and illustrate previous definitions.

Two agents W and J are driving a car (it is a joint activity and the agents have complementary access to the necessary resources). The car is at a stop and the agents have to decide which way to go. Suppose that the initial states of agents W and J are the ones presented by Figure 2. Since W wants to go left (he has the corresponding intention accepted), he wants the corresponding social commitment to be accepted (see Figure 3). W will thus make an offer to J[10]:

W: I would turn left.

[10] More precisely, he will propose to enter an offer game (see [2] for details about the DIAGAL agent language) which is the only game which entry and success conditions unify with the current and wanted state respectively. Using the current framework and algorithms this will result automatically from the situation described by Figure 2 as described in [12]. This is what the cognitive coherence framework is made for: automatizing agent communications.

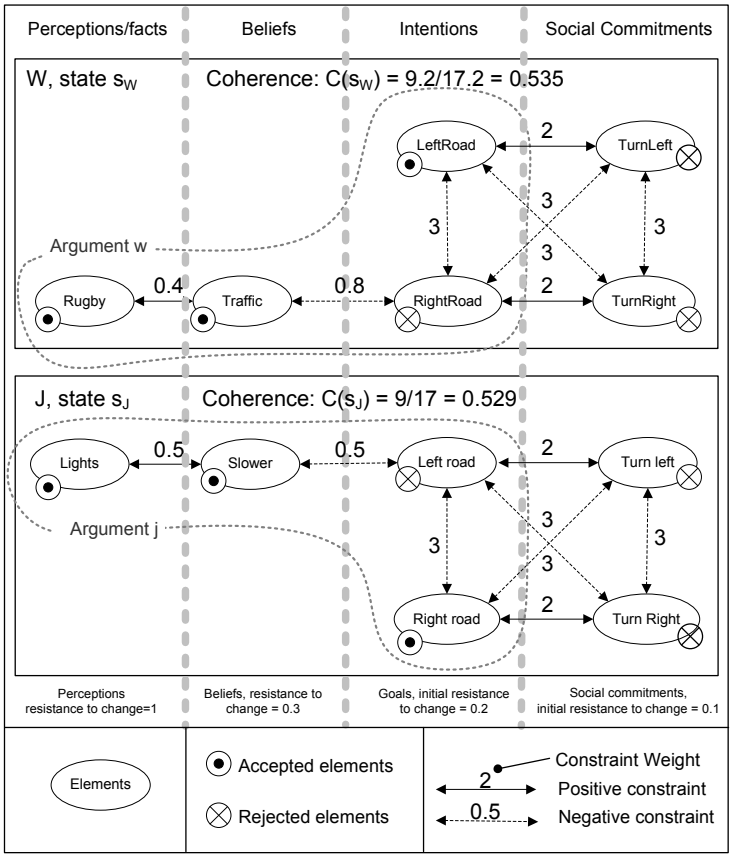


Fig. 2. Initial states s_W and s_J for W and J. Here, all the resistances to change are initialized as shown in order to indicate that perceptions are more resistant than beliefs, that are more resistant than intentions that are more resistant than social commitments. Other choices may be made.

If agent J also would had wanted to turn left (W's proposal would have been coherent with her views), she would have then accepted the proposal and the corresponding social commitment would have been accepted:

J: Ok.

However, as depicted by Figure 2 agent J wants to turn right (i.e. the corresponding intention is accepted), W's proposal acceptance would entail a loss in coherence for J (see Figure 3). J will then embed a counter-proposal[11] as at-

[11] In the form of a DIAGAL request game.

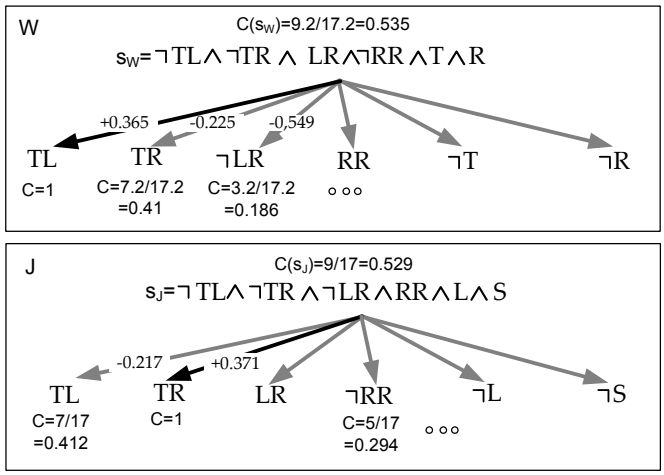


Fig. 3. Reasoning as computed by the local search algorithm from the initial states s_W and s_J for W and J. Here the perceptions/beliefs that "there is a rugby match", "there is a lot of traffic", "there are a lot of lights", "traffic is slower" are noted R, T, L, S respectively, the intentions to turn left and to turn right are noted LR and RR respectfully and the social commitments to turn left and right are noted TR and TL. Rejected elements are noted with a negation sign and only the root of the search tree indicates the full state of the agent, the others nodes just indicate the change they imply. Arcs are labelled with the value of the expected utility function (presented section 2.2). The black path indicates the change(s) returned by the local search algorithm.

tempt to get a result that would be more coherent with her view. Her argument for this choice (j) will be attached to her proposal:

> *J: There 's a lot of lights on the left road, that will slow us down. Can't we turn right instead?*

Notice that, this makes the external incoherence explicit for W[12]. In order to complete the argumentation dialogue initiated by J, W will disclose his own argument (w).

> *W: Yes, but there is a rugby match today, so there will be a lot of traffic on the right road, we should avoid going this way and turn left.*

During that process, the agents eventually communicate each other the entire connected component attached to the discussed issues. However, this doesn't tell anything about the way they will evaluate and integrate the exchanged arguments. Next sections discuss and propose a modelling of those dimensions.

[12] See [15] and [12] for a discussion about the importance of the explicitation phase of dialogue that is usually neglected.

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 [6]. 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 [8]. 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 [18] (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 [19], 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 agents integrate the other's point of view in their own cognitive model and act accordingly (which would be very cooperative) or will they compare their point of view with the other's and then substitute those two if their is weaker or 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.

4.3 Argument evaluation

Argument evaluation will be done by comparing (using a shared measure) the strengths of the arguments provided by both sides in order to decide whose standpoint will be chosen as the more rational one. We use the following argument evaluation measure:

Definition 9. (Strength of an argument)
The strength of a given argument $\langle H, h\rangle$ is the sum of the weights of the satisfied constraints minus the sum of the weights of the non-satisfied ones. Formally:

$$Strengh(\langle H, h\rangle) = 2 * \sum_{(x,y)\in Sat(H\cup h)} Weight(x,y) - \sum_{(x,y)\in Con(H\cup h)} Weight(x,y)$$

The issue of the dispute will depend fully on the comparison between the strength of the considered arguments. In our example, that means that because the strength of W's argument ($Weight(w) = 4.2$) for going through the left road is stronger than the strength of J's argument ($Weight(j) = 4$) for going by the right road, J will concede. The social commitment proposed by W will be accepted and the one advocated by J rejected.

J: Ok, we will go through the left way.[13]

4.4 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 to their strength. This corresponds to a fully cooperative and trustful cognitive behavior. 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

[13] Concretely, this means that J's embedded request will be refused by W and W's offer finally accepted by J. All the opened games will thus be closed.

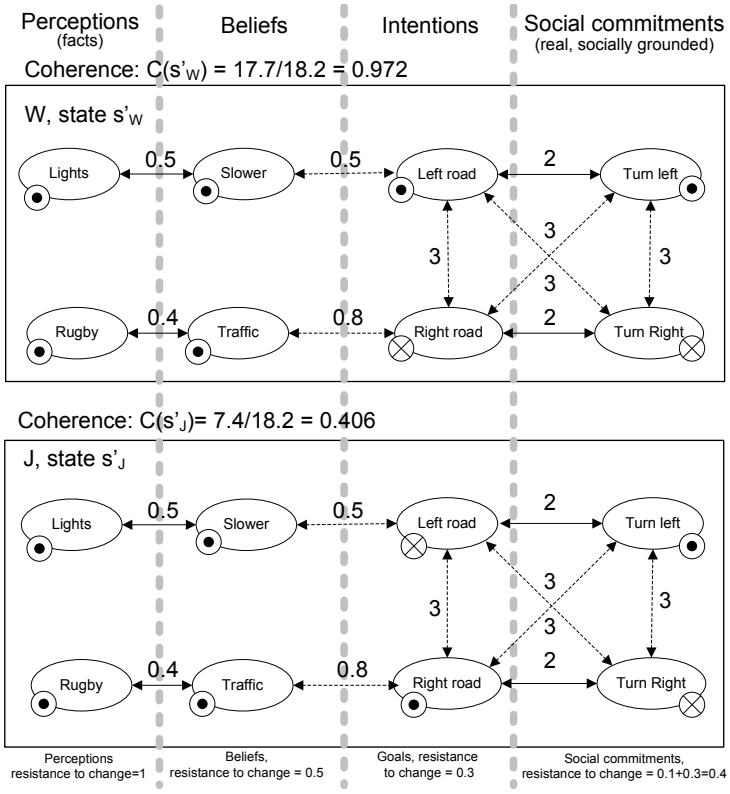


Fig. 4. W and J states after their argumentation dialogue.

integration procedure is fully cooperative since the others' arguments will be fully taken into account in future reasoning. In the current model integration is done after the argument evaluation, thus being a post-evaluation memorization of arguments. Note that different choices may have been possible that will be inquired in future work.

In our example, argument evaluation and integration result in the cognitive models depicted by Figure 4. While W cannot improve his cognitive coherence anymore, Figure 5 shows J's reasoning which embeds an attitude change. Figure 6 presents the final state of the agents which is an equilibrium (no element acceptance change can improve cognitive coherence). Notice that the agent coherence is not maximal (i.e. 1) because of the integration of J's argument which is against the chosen issue (and is valuable).

Finally, it is probable that W will turn left in order to fulfill the corresponding social commitment and advance the state of the environment...

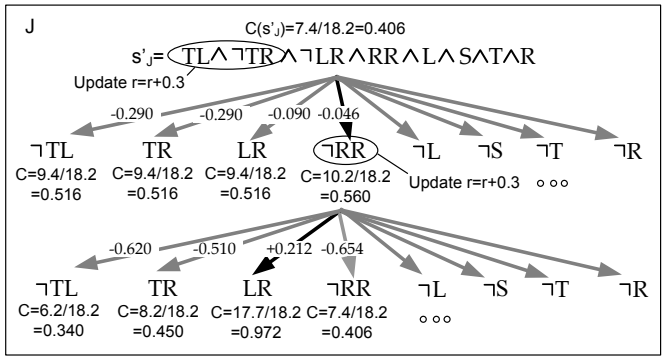


Fig. 5. J's reasoning from the state s'_J, resulting from the argumentation dialogue. Notice the attitude change.

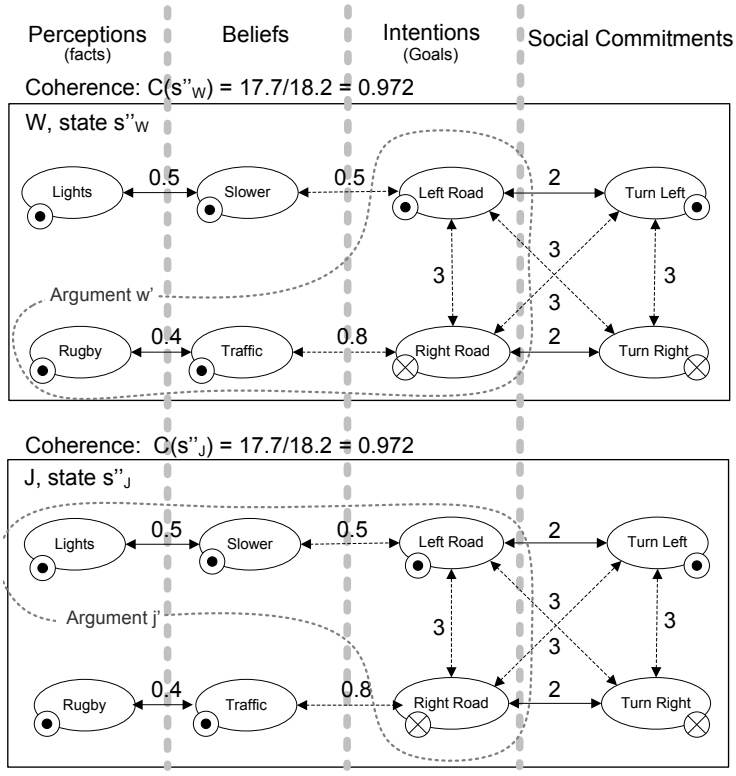


Fig. 6. Final states (after integration) for W and J.

5 Coverage of the presented approach

Our approach allows to cover a variety of argumentation dialogues. For example, argumentations that rely on element types (cognitions types and their related resistance to change). For example, the following dialogue involves perception as an argument:

W: Google can answer a request in less than 2 seconds and gives you pertinent pages out of several millions ones.
J: No!
W: Yes.
J: How do you know?
W: I have seen it.

Also, while *social arguments* have not been considered in the literature yet, we think they are crucial in multi-agents settings. Here is an example, that can be captured by our approach, where J justifies his decision using a social argument:

Q: Do you want to go to the cinema tonight?
J: No, I can't.
Q: Why?
J: I promised my boss to finish a paper tonight.

More generally, the treatment of the cognitive aspects of pragmatics models the persuasion process that allow to capture a variety of persuasive dialogues including those that do not involve argumentation. Here is an example of such dialogue:

Boss: You have to finish that paper tonight.
J: Yes.

In DIAGAL [2], an order given by an agent that has authority over his interlocutor results in a social commitment being accepted by definition. However, J's behavior will still be guided by his coherence calculus and J will either enter an attitude change and accept the corresponding intention or cancel or violate this social commitment while coping the sanctions (which are taken into account in the agent reasoning through the resistance to change of the accepted commitment).

This shows how our approach integrates argumentation with other agent communication behavior through the modelling of the cognitive aspect of pragmatics that emphasizes the persuasive dimension of every communication. The limit case of argumentation dialogue being the one in which each argument consists of a single element, our approach can be seen as an attempt to unify argumentation-based frameworks with previous agent communication frameworks (specifically social commitment based communication) through some higher level concepts from cognitive sciences.

6 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 behavioral 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. While promising, this alternative approach to argumentation requires more work. In particular, studying how this framework differs from and complements previous (dialectic based) proposals is in our future work list.

References

1. 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.
2. 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)*.
3. 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.
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. 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.
6. P. Erwin. *Attitudes and Persuasion*. Psychology Press, 2001.
7. L. Festinger. *A Theory of Cognitive Dissonance*. Stanford University Press, 1957.
8. 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.
9. H. Jung and M. Tambe. Toward argumentation as distributed constraint satisfaction. In *Proceedings of the AAAI Fall Symposium on Negotiation Methods fos Autonomous Cooperative Systems*, 2001.

10. 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.
11. B. Moulin, H. Irandoust, M. Blanger, and G. Desbordes. Explanation and argumentation capabilities: Towards the creation of more persuasive agents. *Artificial Intelligence Review*, 17(3):169–222, 2002.
12. P. Pasquier. *Aspects cognitifs des dialogues entre agents artificiels : l'approche par la cohérence cognitive*. PhD thesis, Laval University, Quebec, Canada, August 2005.
13. 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.
14. 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.
15. P. Pasquier and B. Chaib-draa. Agent communication pragmatics: The cognitive coherence approach. *Cognitive Systems*, 6(4):364–395, 2005.
16. 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.
17. I. Rahwan, S. Ramchurn, N. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation based negotiation. *Knowledge Engineering Review*, 18(4):343–375, 2003.
18. 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.
19. M Sherif and C.I. Hovland. *Social Judgement*. Yale University Press, 1961.
20. 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.
21. M. P. Singh. An ontology for commitments in multiagent systems: Toward a unification of normative concepts. *Artificial Intelligence and Law*, 7:97–113, 1999.
22. R. Sun. *Connectionist-Symbolic Integration*, chapter An introduction to hybrid connectionist-symbolic models. Lawrence Erlbaum Associates., 1997.
23. P. Thagard. *Coherence in Thought and Action*. The MIT Press, 2000.
24. P. Thagard. Probabilistic network and explanatory coherence. *Cognitive science Quaterly*, (1):91–114, 2000.
25. P. Thagard and K. Verbeurgt. Coherence as constraint satisfaction. *Cognitive Science*, 22:1–24, 1998.
26. F. H. van Eemeren and R. Grootendorst. *A Systematic Theory of Argumentation: the Pragma-Dialectical Approach*. Cambridge University Press, 2004.

An argumentation-based approach for dialog move selection

Leila Amgoud Nabil Hameurlain

Institut de Recherche en Informatique de Toulouse
Université Paul Sabatier, 118 route de Narbonne,
31062 Toulouse Cedex 4, France
amgoud@irit.fr
LIUPPA, Université de Pau
Avenue de l'université
BP 1155 64012 Pau Cedex, France
nabil.hameurlain@univ-pau.fr

Abstract. Modeling different types of dialog between autonomous agents is becoming an important research issue. Several proposals exist with a clear definition of the dialog protocol, which is the set of rules governing the high level behavior of the dialog. However, things seem different with the notion of strategy. There is no consensus on the definition of a strategy and on the parameters necessary for its definition. Consequently, there are no methodology and no formal models for strategies.
This paper argues that a strategy is a *decision* problem that consists of: i) selecting the type of act to utter at a given step of a dialog, and ii) selecting the content that will accompany the act. The first kind of decision amounts to select among all the acts allowed by the protocol, the best option which according to some *strategic beliefs* of the agent will at least satisfy the most important *strategic goals* of the agent. The second kind of decision consists of selecting among different alternatives (eg. different offers), the best one that, according to some *basic beliefs* of the agent, will satisfy the *functional goals* of the agent. The paper proposes then a formal model based on argumentation for computing on the basis of the above kinds of mental states, the best move (act + content) to play at a given step of the dialog. The model is illustrated through an example of auctions.

1 Introduction

An increasing number of software applications are being conceived, designed, and implemented using the notion of autonomous agents. These applications vary from email filtering [10], through electronic commerce [12, 16], to large industrial applications [6]. In all of these disparate cases, however, the agents are *autonomous* in the sense that they have the ability to decide for themselves which goals they should adopt and how these goals should be achieved [17]. In most agent applications, the autonomous components need to interact with one another because of the inherent interdependencies which exist between them. They need to communicate in order to resolve differences of opinion and conflicts of interest that result from differences in preferences, work together to find solutions to dilemmas and to construct proofs that they cannot manage alone, or simply to

inform each other of pertinent facts. Many of these communication requirements cannot be fulfilled by the exchange of single messages. Instead, the agents concerned need to be able to exchange a *sequence of messages* which all bear upon the same subject. In other words they need the ability to engage in *dialogs*. In [15] different categories of dialogs have been distinguished including persuasion and negotiation. Work in the literature has focused on defining formal models for these dialog types. Generally, a dialog system contains the following three components: the agents involved in the dialog (i.e their *mental states*), a dialog *protocol* and a set of *strategies*. The dialog protocol is the set of *rules of encounter* governing the high-level behavior of interacting agents. A protocol defines among other things:

- the set of permissible acts (eg. asking questions, making offers, presenting arguments, etc.);
- the legal replies for each act.

A dialog protocol identifies the different possible replies after a given act. However, the exact act to utter at a given step of the dialog is a *strategy* matter. While the protocol is a public notion, strategy is crucially an individualistic matter. A strategy can be seen as a two steps *decision process*:

1. among all the possible replies allowed by the protocol, to choose the move to play. For instance, in a negotiation dialog, the protocol may allow after an offer act the following moves: accepting/rejecting the offer or making a new offer.
2. to choose the content of the move if any. In the above example, if the agent chooses to make a new offer, it may decide among different alternatives the best one to propose.

In most works on modeling dialogs, the definition of a protocol poses no problems. However, the situation is different for dialog strategies. There is no methodology and no formal models for defining them. There is even no consensus on the different ingredients involved when defining a strategy. Regarding persuasion dialogs, there are very few works devoted to the notion of strategy in the literature if we except the work done in [2, 7]. In these works, different criteria have been proposed for the argument selection. As for negotiation dialogs, it has been argued that the game-theoretic approaches characterize correctly optimal strategies [8, 13]. However, another line of research [5, 9, 11, 14] has emphasized the limits of game-theoretic approaches for negotiation, and has shown the interest of arguing during a negotiation. Consequently, the optimal strategies given by game theory are no longer valid and not suitable. In [3], the authors have studied the problem of choosing the best offer to propose during a dialog and several criteria have been suggested. However, in that framework, the act offer is supposed to be chosen by the agent. Thus, this work has focused only on the second step of the decision process.

This paper argues that the strategy is a decision problem in which an agent tries to choose among different *alternatives* the best option, which according to its beliefs, will satisfy at least its most important goals. Two kinds of goals (resp. of beliefs) are distinguished: the *strategic* and the *functional* goals (resp. the *strategic* and *basic* beliefs).

The strategic goals are the meta level goals of the agent. Such goals will help an agent, on the basis of the strategic beliefs, to select the type of act to utter. Regarding functional goals, they will help an agent to select, on the basis of the basic beliefs to select the content of a move.

We propose a formal model for defining strategies. The model takes as input two sets of goals: the strategic and the functional goals together with the strategic and basic beliefs and returns among the possible replies allowed by the protocol after a given act, the next move (act + its content) to play. The model is an extension of the argument-based decision framework proposed in [1]. The basic idea behind this model is to construct for each alternative the different arguments (reasons) supporting it, then to compare pairs of alternatives on the basis of the quality of their supporting arguments.

The paper is organized as follows: Section 2 presents the different classes of goals and beliefs maintained by an agent. Section 3 introduces the logical language which will be used throughout the paper. Section 4 introduces an abstract argumentation-based decision model which forms the backbone of our approach. Section 5 presents an instantiation of that abstract model for computing the best move to play among the different replies allowed by the protocol. Section 6 introduces a second instantiation of the abstract model for computing the content of the move selected by the first instantiation. The whole framework is then illustrated in section 8. Section 9 is devoted to some concluding remarks and some perspectives.

2 Agents' mental states

During a dialog, an agent makes two decisions: it first selects the type of act to utter, for instance making a new offer, asking a question or arguing. Once the act chosen, the agent should select the content of the act if necessary. We say if necessary because some acts such as "withdrawal" from a dialog does not need a content. However, for an act "offer", it is important to accompany the act with an appropriate content. If the agents are negotiating the "price" of a car, then the act offer should contain a given price. The two above decision problems involve two different kinds of goals:

Strategic goals: For choosing the type of act to utter, an agent refers to what we call *strategic goals*. By strategic goals we mean the meta-level goals of the agent such as "minimizing the dialog time", "selling at the end of the dialog", etc. Suppose that at a given step of a negotiation dialog, an agent has to choose between making an offer and asking a question. If the agent wants to minimize the dialog time then it would choose to make an offer instead of spending more time in questions. However, if the agent wants to get a maximum of information about the wishes of the other agent, then the agent would decide to ask a question.
Strategic goals are generally independent of the *subject* of the dialog. If the agents are negotiating the place of a next meeting, then those goals are not related to the place.

Functional goals: The goals of the agent which are directly related to the subject of the dialog are called *functional goals*. They represent what an agent wants to achieve or to get regarding the subject of the dialog. Let us take the example of the agent

negotiation the place of a meeting. The agent may prefer a place which is not warm and not expensive. The agent may also prefer a place with an international airport. These functional goals are involved when selecting the content of a move. In a negotiation, an agent proposes offers that satisfy such goals.

As for goals, the beliefs involved in the two decision problems are also of different nature:

Strategic beliefs that are the meta-level beliefs of the agent. They may represent the beliefs of the agent about the dialog, and about the other agents involved in the dialog. In negotiation dialogs where agents are trying to find a common agreement, agents may intend to simulate the reasoning of the other agents. Thus it is important for each agent to consider the beliefs that it has on the other agents'goals and beliefs. Indeed, a common agreement can be more easily reached if the agents check that their offers may be consistent with what they believe are the goals of the others.

Basic beliefs represent the beliefs of the agent about the environment and the subject of the dialog. Let us consider again the example of the agent negotiating the place of a meeting. Basic beliefs of the agent may include for instance the fact that "London is not warm", "Tunisia is hot", "London is very expensive", etc. This base may also contain some integrity constraints related to the dialog subject such as "the meeting cannot be at the same time in London and in Algeria".

3 The logical language

Let $\mathcal{L}$ be a propositional language, and $Wff(\mathcal{L})$ be the set of well-formed formulas built from $\mathcal{L}$. Each agent has the following bases:

- $\mathcal{B}_b = \{(k_p, \rho_p), p = 1, \ldots, s\}$, where $k_p \in Wff(\mathcal{L})$, is the basic beliefs base. The beliefs can be less or more certain. They are associated with certainty levels ρ_p. A pair (k_p, ρ_p) means that k_p is at least certain at a degree ρ_p.
- $\mathcal{B}_s = \{(l_j, \delta_j), j = 1, \ldots, m\}$, where $l_j \in Wff(\mathcal{L})$, is the strategic beliefs base. Each of these beliefs has a certainty level δ_j.
- $\mathcal{G}_s = \{(g_q, \lambda_q), q = 1, \ldots, t\}$, where $g_q \in Wff(\mathcal{L})$, is a base of strategic goals. The strategic goals can have different priority degrees, represented by λ_q. A pair (g_q, λ_q) means that the goal g_q is important for the agent at least to a degree λ_q.
- $\mathcal{G}_f = \{(go_r, \gamma_r), r = 1, \ldots, v\}$, where $go_r \in Wff(\mathcal{L})$, is the base of the functional goals of the agent. Each functional goal has a degree of importance denoted by γ_r.

The different certainty levels and priority degrees are assumed to belong to a unique linearly ordered scale T with maximal element denoted by 1 (corresponding to total certainty and full priority) and a minimal element denoted by 0 corresponding to the complete absence of certainty or priority.
We shall denote by $\mathcal{B}_b^*$, $\mathcal{B}_s^*$, $\mathcal{G}_s^*$ and $\mathcal{G}_f^*$ the corresponding sets of propositional formulas when weights are ignored.

Let $\mathcal{S}$ be the set of speech acts allowed by the protocol. $\mathcal{S}$ may contain acts such as

"'Offer' for making offers in negotiation dialogs, "Question" for asking questions, "Assert" for asserting information such as "the weather is beautiful", "Argue" for presenting arguments in persuasion dialogs, etc. The protocol precises for each act the possible replies to it. Let us suppose that the function `Replies` returns for each act, all the legal replies to it.

$$\texttt{Replies}: \mathcal{S} \longmapsto 2^{\mathcal{S}}$$

Some acts may have a content. For instance, an act "Offer" should be accompanied with a content such as a price, a town, etc. However, the act "Withdraw" does not need any content. Such acts will then have an empty content, denoted by the symbol "?". In what follows, the function `Content` returns for a given act, the set of its possible contents. Formally:

$$\texttt{Content}: \mathcal{S} \longmapsto 2^{Wff(\mathcal{L}) \cup \{?\}}$$

For instance, $\texttt{Content}(Withdraw) = \{?\}$, $\texttt{Content}(Offer) = \{London, Algeria\}$.

During a dialog, agents exchange *moves* which are pairs: a *speech act* and its *content*. Formally:

Definition 1 (Moves). *A* move *is a pair* (a, x)*, where* $a \in \mathcal{S}$ *and* $x \in \texttt{Content}(a)$.

The strategy problem is formalized as follows:

Definition 2 (The strategy problem). *Let* (a, x) *be the current move in a dialog. What is the next move* (a', x') *to utter such that* $a' \in \texttt{Replies}(a)$ *?*

To answer this question, one should find both a' and x'. Indeed, a' is the "best" element in $\texttt{Replies}(a)$ that satisfies $\mathcal{G}_s^*$ according to $\mathcal{B}_s^*$. This will be denoted by: $\mathcal{B}_s^*, a' \rightarrow \mathcal{G}_s^*$. Here by "best" we mean the act that satisfies as much important goals as possible.

Concerning x', this is also the "best" element among $X \subseteq Wff(\mathcal{L})$ that satisfies $\mathcal{G}_f^*$ according to $\mathcal{B}_b^*$. This will be denoted by: $\mathcal{B}_b^*, x' \rightarrow \mathcal{G}_f^*$. Here the set X is exactly the set of different alternatives concerning the content of a move. This set may contain different offers (eg. different town) if we have to choose the content of the act "offer", it may contain a set of formulas if we have to choose the content of the act "Assert", it may also contain a set of arguments if one has to select the content of the move "Argue", etc.

The solution to the strategy problem is the pair (a', x') such that $(\mathcal{B}_s^*, a' \rightarrow \mathcal{G}_s^*) \wedge (\mathcal{B}_b^*, x' \rightarrow \mathcal{G}_f^*)$.

4 The abstract argumentation-based decision model

Recently, Amgoud [1] has proposed a formal framework for making decisions under uncertainty on the basis of arguments that can be built in favor of and against a possible choice. Such an approach has two obvious merits. First, decisions can be more easily explained. Moreover, argumentation-based decision is maybe closer to the way humans make decisions than approaches requiring explicit utility functions and uncertainty distributions.

Solving a decision problem amounts to defining a pre-ordering, usually a complete one, on a set $\mathcal{X}$ of possible choices (or decisions), on the basis of the different consequences of each decision. In our case, the set $\mathcal{X}$ may be either the set $\texttt{Replies}(a)$ of the possible replies to a move, or the set $\texttt{Content}(a)$. The basic idea behind an argumentation-based model is to construct arguments in favor of and against each decision, to evaluate such arguments, and finally to apply some principle for comparing the decisions on the basis of the arguments and their quality or strengths. Thus, an argumentation-based decision process can be decomposed into the following steps:

1. Constructing arguments in *favor* of */against* each decision in $\mathcal{X}$.
2. Evaluating the strength of each argument.
3. Comparing decisions on the basis of their arguments.
4. Defining a pre-ordering on $\mathcal{X}$.

Definition 3 (Argumentation-based decision framework). *An* argumentation-based decision framework *is a tuple* $<\mathcal{X}, \mathcal{A}, \succeq, \triangleright_{Princ}>$ *where:*

- $\mathcal{X}$ *is a set of all possible decisions.*
- $\mathcal{A}$ *is a set of arguments.*
- $\succeq$ *is a (partial or complete) pre-ordering on* $\mathcal{A}$*.*
- $\triangleright_{Princ}$ *(for principle for comparing decisions), defines a (partial or complete) pre-ordering on* $\mathcal{X}$*, defined on the basis of arguments.*

The output *of the framework is a (complete or partial) pre-ordering* $\triangleright_{Princ}$*, on* $\mathcal{X}$*.* x_1 $\triangleright_{Princ}$ x_2 *means that the decision* x_1 *is* at least as preferred as *the decision* x_2 *w.r.t. the* principle $Princ$*.*

Notation: Let A, B be two arguments of $\mathcal{A}$. If $\succeq$ is a pre-order, then $A \succeq B$ means that A is at least as 'strong' as B.
$\succ$ and $\approx$ will denote respectively the strict ordering and the relation of equivalence associated with the preference between arguments. Hence, $A \succ B$ means that A is strictly preferred to B. $A \approx B$ means that A is preferred to B and B is preferred to A.

Different definitions of $\succeq$ or different definitions of $\triangleright_{Princ}$ may lead to different decision frameworks which may not return the same results.

In what follows, $\texttt{Arg}(x)$ denotes the set of arguments in $\mathcal{A}$ which are in favor of x.
At the core of our framework is the use of a principle that allows for an argument-based comparison of decisions. Indeed, these principles capture different *profiles* of agents regarding decision making. Below we present one intuitive principle $Princ$, i.e agent profile. This principle, called *promotion focus* principle (Prom), prefers a choice that has at least one supporting argument which is preferred to (or stronger than) any supporting argument of the other choice. Formally:

Definition 4 (Promotion focus). *Let* $<\mathcal{X}, \mathcal{A}, \succeq, \triangleleft_{Prom}>$ *be an argumentation-based decision framework, and Let* $x_1, x_2 \in \mathcal{X}$*.*
$x_1 \triangleright_{Prom} x_2$ *w.r.t* $Prom$ *iff* $\exists\ A \in \texttt{Arg}(x_1)$ *such that* $\forall\ B \in \texttt{Arg}(x_2)$*,* $A \succeq B$*.*

Obviously, this is a sample of the many principles that we may consider. Human deciders may actually use more complicated principles.

5 The strategic decision model

This section presents an instantiation of the above model in order to select the next move to utter. Let us recall the strategy problem. Let (a, x) be the current move in a dialog. What is the next move (a', x') to utter such that $a' \in \texttt{Replies}(a)$ and $x' \in \texttt{Content}(a)$? The strategic decision model will select among $\texttt{Replies}(a)$ the best act to utter, say a'. Thus, the set $\texttt{Replies}(a)$ will play the role of $\mathcal{X}$.
Let us now define the arguments in favor of each $d \in \texttt{Replies}(a)$. Those arguments are built from the strategic beliefs base $\mathcal{B}_s$ of the agent and its strategic goals base $\mathcal{G}_s$.

The idea is that a decision is justified and supported if it leads to the satisfaction of at least the most important goals of the agent, taking into account the most certain part of knowledge. Formally:

Definition 5 (Argument). *An* argument in favor of *a choice d is a triple $A = <S, g, d>$ such that:*

- $d \in \texttt{Replies}(a)$
- $S \subseteq \mathcal{B}_s^*$ *and* $g \in \mathcal{G}_s^*$
- $S \cup \{d\}$ *is consistent*
- $S \cup \{d\} \vdash g$
- S *is minimal (for set inclusion) among the sets satisfying the above conditions.*

S *is the* support *of the argument,* g *is the* goal *which is reached by the choice* d*, and* d *is the* conclusion *of the argument. The set $\mathcal{A}_s$ gathers all the arguments which can be constructed from $<\mathcal{B}_s, \mathcal{G}_s, \texttt{Replies}(a)>$.*

Since the bases $\mathcal{B}_s$ and $\mathcal{G}_s$ are weighted, arguments in favor of a decision are more or less strong.

Definition 6 (Strength of an Argument). *Let $A = <S, g, d>$ be an argument in $\mathcal{A}_s$. The* strength *of A is a pair $<Level_s(A), Weight_s(A)>$ such that:*

- *The* certainty level *of the argument is $Level_s(A) = min\{\rho_i \mid k_i \in S$ and $(k_i, \rho_i) \in \mathcal{B}_s\}$. If $S = \emptyset$ then $Level_s(A) = 1$.*
- *The* degree of satisfaction *of the argument is $Weight_s(A) = \lambda$ with $(g, \lambda) \in \mathcal{G}_s$.*

Then, strengths of arguments make it possible to compare pairs of arguments as follows:

Definition 7. *Let A and B be two arguments in $\mathcal{A}_s$. A is* preferred *to B, denoted $A \succeq_s B$, iff min($Level_s(A)$, $Weight_s(A)$) $\geq$ min($Level_s(B)$, $Weight_s(B)$).*

Property 1. The relation $\succeq_s$ is a complete preorder ($\succeq_s$ is reflexive and transitive).

Now that the arguments defined, we are able to present the strategic decision model which will be used to return the best reply a' at each step of a dialog.

Definition 8 (Strategic decision model). *A* strategic decision model *is a tuple $<\texttt{Replies}(a)$, $\mathcal{A}_s$, $\succeq_s$, $\triangleright_{Princ}>$.*

According to the agent profile, a principle $\rhd_{Princ}$ will be chosen to compare decisions. If for instance, an agent is pessimistic then it will select the Prom priciple and thus the decisions are compared as follows:

Definition 9. *Let* a_1, $a_2 \in \texttt{Replies}(a)$. $a_1 \rhd_{Prom} a_2$ *w.r.t* $Prom$ *iff* $\exists\ A \in \texttt{Arg}(a_1)$ *such that* $\forall\ B \in \texttt{Arg}(a_2)$, $A \succeq_s B$.

Property 2. The relation $\rhd_{Prom}$ is a complete preorder.

Since the above relation is a complete preorder, it may be the case that several choices will be equally preferred. The most preferred ones will be returned by the function `Best`.

Definition 10 (Best decisions). *The set of best decisions is* $\texttt{Best}(\texttt{Replies}(a)) = \{a_i \in \texttt{Replies}(a), s.t. \forall\ a_j \in \texttt{Replies}(a),\ a_i \rhd_{Prom} a_j\}$.

Property 3. If $\mathcal{A}_s = \emptyset$, then $\texttt{Best}(\texttt{Replies}(a)) = \emptyset$.

Note that when the set of arguments is empty, then the set of best decisions is also empty. This means that all the decisions are equally preferred, and there is no way to choose between them. In such a situation, the decision maker chooses one randomly.

Definition 11 (Best move). *The best move to play (or the next reply in a dialog) is* $a' \in \texttt{Best}(\texttt{Replies}(a))$.

6 The functional decision model

Once the speech act to utter selected by the previous strategic decision model, say $a' \in \texttt{Best}(\texttt{Replies}(a))$, one should select its content if necessary among the elements of $\texttt{Content}(a')$. Here $\texttt{Content}(a')$ depends on the nature of the selected speech act. For instance, if the selected speech act is an "Offer", then $\texttt{Content}(a')$ will contain different objects such as prices if the agents are negotiating a price of a product, different towns if they are negotiating a place of the next holidays. Now, if the selected speech act is "Argue" which allows the exchange of arguments, then the content of this act should be an argument, thus $\texttt{Content}(a')$ will contain the possible arguments. In any case, we suppose that $\texttt{Content}(a')$ contains a set of propositional formulas. Even in the case of a set of arguments, every argument will be refered to it by a propositional formula.
Arguments in favor of each element in $\texttt{Content}(a')$ are built from the basic beliefs base and the functional goals base.

Definition 12 (Argument). *An* argument in favor of *a choice* d *is a triple* $A = <S, g, d>$ *such that:*

- $d \in \texttt{Content}(a')$
- $S \subseteq \mathcal{B}_b^*$ *and* $g \in \mathcal{G}_f^*$
- $S \cup \{d\}$ *is consistent*
- $S \cup \{d\} \vdash g$
- S *is minimal (for set inclusion) among the sets satisfying the above conditions.*

S = $Support(A)$ *is the* support *of the argument,* C = *Consequences(A) its* consequences *(the goals which are reached by the decision* d*) and* d = $Conclusion(A)$ *is the conclusion of the argument. The set* $\mathcal{A}_f$ *gathers all the arguments which can be constructed from* $<\mathcal{B}_b, \mathcal{G}_f, \mathcal{X}>$.

The strength of these arguments is defined exactly as in the previous section by replacing the corresponding bases.

Definition 13 (Strength of an Argument). *Let* $A = <S, g, d>$ *be an argument in* $\mathcal{A}_f$. *The* strength *of* A *is a pair* $<Level_f(A), Weight_f(A)>$ *such that:*

- *The* certainty level *of the argument is* $Level_f(A)$ = *min*$\{\rho_i \mid k_i \in S$ *and* $(k_i, \rho_i) \in \mathcal{B}_b\}$. *If* $S = \emptyset$ *then* $Level_f(A) = 1$.
- *The* degree of satisfaction *of the argument is* $Weight_f(A) = \lambda$ *with* $(g, \lambda) \in \mathcal{G}_f$.

Then, strengths of arguments make it possible to compare pairs of arguments as follows:

Definition 14. *Let* A *and* B *be two arguments in* $\mathcal{A}_f$. A *is* preferred *to* B, *denoted* $A \succeq_f B$, *iff min*$(Level_f(A), Weight_f(A)) \geq$ *min*$(Level_f(B), Weight_f(B))$.

The arguments against decisions in $\mathcal{X}$ are defined in the same way as in the previous section. We have just to replace the base $\mathcal{B}_s$ by $\mathcal{B}_b$, $\mathcal{G}_s$ by $\mathcal{G}_f$ and $\texttt{Replies}(a)$ by $\mathcal{X}$. The functional model which computes the best content of a move is defined as follows:

Definition 15 (Functional decision model). *A* functional decision model *is a tuple* $<\texttt{Content}(a'), \mathcal{A}_f, \succeq_f, \triangleright_{Princ}>$.

Again according to the agent profile, a principle $\triangleright_{Princ}$ will be chosen to compare decisions. If for instance, an agent is pessimistic then it will select the Prom principle and thus the decisions are compared as follows:

Definition 16. *Let* $x_1, x_2 \in \mathcal{X}$. $x_1 \triangleright_{Prom} x_2$ *w.r.t* $Prom$ *iff* $\exists\ A \in \texttt{Arg}(x_1)$ *such that* $\forall\ B \in \texttt{Arg}_P(x_2)$, $A \succeq_f B$.

Here again, the above relation is a complete preorder, and consequently several options may be equally preferred.

Definition 17 (Best decisions). *The set of best decisions is* $\texttt{Best}(\texttt{Content}(a')) = \{x_i \in \texttt{Content}(a'), s.t. \forall\ x_j \in \texttt{Content}(a'),\ x_i \triangleright_{Prom} x_j\}$.

The content x' to utter is an element of $\texttt{Best}(\texttt{Content}(a'))$ chosen randomly. Formally:

Definition 18 (Best move). *The best content is* x' *such that* $x' \in \texttt{Best}(\texttt{Content}(a'))$.

7 Computing the next move in a dialogue

In the previous section, we have presented a formal framework for explaining, oredering and making decisions. In what follows, we will show how that framework can be used for move selection. Let (a, x) be the current move of the dialogue, and an agent has to

choose the next one, say (a', x'). The act a' is returned as a best option by the framework $<\texttt{Replies}(a), \mathcal{A}_s, \succeq_s, \triangleright_{Prom}>$ (i.e $a' \in \texttt{Best}(\texttt{Replies}(a))$), whereas the content x' is among the best options returned by the framework $<\texttt{Content}(a'), \mathcal{A}_f, \succeq_f, \triangleright_{Prom}>$, i.e. $x' \in \texttt{Best}(\texttt{Content}(a'))$.

The basic idea is to look for the best replies for an act a. In case there is no solution, the answer will be (?, ?) meaning that there is no rational solution. This in fact corresponds either to the situation the set of strategic goals is empty, or the case where no alternative among the allowed replies satisfies the strategic goals of the agent.

In case there is at least one preferred solution, one should look for a possible content. If there is no possible content, then the chosen act is removed and the same process is repeated with the remaining acts. Note that the case of the existence of a preferred act but no its content is explained by the fact that the strategic goals of the agent are not compatible with its functional goals. Moreover, two forms of incompatibilities are distinguished: *strong* incompatibility in which there is no act which can be accompanied with a content, and a *weak* incompatibility in which only some acts can be associated with contents. The above idea of computing the next move is skeched in the following algorithm:

Function 1 Computing the best move

Parameters: a current move (a, x), a theory $\langle \mathcal{X}, \mathcal{B}_s, \mathcal{B}_f, \mathcal{G}_s, \mathcal{G}_f \rangle$

1: $\mathcal{X} \leftarrow \texttt{Replies}(a)$;
2: failure $\leftarrow \bot$;
3: **while** $\mathcal{X} \neq \emptyset$ and $\neg$ failure **do**
4: **if** $\texttt{Best}(\mathcal{X}) = \emptyset$ **then**
5: failure $\leftarrow \top$;
return (?, ?);
6: 7: **else** $a' \in \texttt{Best}(\texttt{Replies}(a))$ of the argumentation system $\langle \texttt{Replies}(a), \mathcal{A}_s, \succeq_s, \triangleleft_{Prom} \rangle$ (a' is chosen randomly);
8: **if** $\texttt{Content}(a') =?$ **then**
9: failure $\leftarrow \top$;
return $(a', ?)$;
10:11: **else**
12: **if** $\texttt{Best}(\texttt{Content}(a')) = \emptyset$ (best decisions of the argumentation system $\langle \texttt{Content}(a'), \mathcal{A}_f, \succeq_f, \triangleleft_{Prom} \rangle$); **then**
13: $\mathcal{X} \leftarrow \mathcal{X} - \{a'\}$;
14: **else**
15: failure $\leftarrow \top$;
16: **return** (a', x') with $x' \in \texttt{Content}(a')$;

The following properties can be shown:

Property 4. If $\mathcal{G}_s = \emptyset$, or $\mathcal{B}_s = \emptyset$, then the next move is (?, ?).

8 Illustrative example

To illustrate the formal model, we will present an example of auction protocols, the Dutch auction, which is used in the implementation of the fish market interaction protocol [4].

The idea here is that seller S wants to sell an item using an auction. A number of potential buyers $B_1, \ldots, B_n$, called also bidders, participate in rounds of auctions. There is at least one round for each item during, which the auctioneer counts down the price for the item and buyers simply send a signal to say if they want to bid at the current price or not.

In the context of fish market, the protocol is indeed, organized in terms of rounds. At each round, the seller proposes a price for the item. If there is no bidder then the price is lowered by a set amount until a bid is received. However, if the item reaches its reserve price the seller declares the item withdrawn and closes the round. If there is more than one bid, the item is not sold to any buyer, and the seller restarts the round at a higher price. Otherwise, if there is only one bid submitted at the current price, the seller atributes the item to that buyer. In this protocol, the set of allowed moves is then:

$$S = \{Offer, Accept, Pass, Attribute, Withdraw\}$$

The first move allows the seller to propose prices, the second move allows buyers to bid i.e to accept current price, the move $Pass$ allows also the buyers to pass their turn by saying nothing, the move $Attribute$ allows the seller to attibute the item to the selected buyer, and the last move $Withdraw$ allows the seller to withdraw the item from the auction. The following possible replies are also given by the protocol:

$Replies(Offer) \subseteq \{Accept, Pass\}$
$Replies(Accept) \subseteq \{Offer, Attribute\}$
$Replies(Pass) \subseteq \{Offer, Withdraw\}$
The dialog starts always by a move $Offer$ uttered by the seller.

The seller has a strategic goal which consists of minimizing the auction time. This goal is stored in the strategic goal base of the agent.

$$\mathcal{G}_s^S = \{(min\_time, 0.8)\}$$

This agent has some strategic beliefs such as: if the time spent in the round is higher than a certain bound $time\_bound$ then it should stop the auction.

$$\mathcal{B}_s^S = \{(time\_spent > time\_bound \wedge Withdraw \rightarrow min\_time, 1), (time\_spent < time\_bound \wedge Offer \rightarrow min\_time, 1), (time\_spent < time\_bound \wedge Attribute \rightarrow min\_time, 1)(time\_spent > time\_bound \wedge Offer \rightarrow \neg min\_time, 1)\}$$

The seller has also some functional goals. The first one consist of maximizing its gain $max - gain$. Moreover, a seller has a starting price and also a reserve price which represents the minimum amount that it will accept for the item. Thus a functional goal of this agent would be to have a price at least equal to the reserve price, $good - price$.

$$\mathcal{G}_f^S = \{(good\_price, 1)\}$$

The functional beliefs of the seller are given in its beliefs base:

$$\mathcal{B}_f^S = \{(current\_price > reserve\_price \wedge Offer(current\_price) \rightarrow good\_price, 1), (current\_price > reserve\_price \wedge Attribute(current\_price) \rightarrow good\_price, 1), (current\_price < reserve\_price \wedge Offer(current\_price) \rightarrow \neg good\_price, 1)\}$$

Regarding the buyers, the aim of B_1 is to get the item for the lowest possible price $cheap$ at most at $bound\_price$, and the aim of B_2 is to get the item for the lowest possible price max_profit at most at $bound\_price/2$, that is the agent B_2 bid for the current price only when he could make at least 100% profit on the item. These last are functional goals of the buyers since it concerns the subject of the negotiation. For the sake of simplicity, these agents do not have strategic beliefs and goals.

$$\mathcal{G}_f^{B_1} = \{(cheap, 0.8), (buy, 0.7)\}$$
$$\mathcal{G}_f^{B_2} = \{(max\_profit, 0.8), (buy, 0.7)\}$$

The buyers are supposed to have the following beliefs.

$$\mathcal{B}_f^{B_1} = \{(current\_price < bound\_price \wedge Accept(current\_price) \rightarrow cheap, 1), (current\_price < bound\_price \wedge Accept(current\_price) \rightarrow buy, 1), (current\_price > bound\_price \wedge Accept(current\_price) \rightarrow \neg buy, 1), (current\_price > bound\_price \wedge Accept(current\_price) \rightarrow \neg cheap, 1), (current\_price > bound\_price \wedge Pass \rightarrow \neg buy, 1)\}$$

$$\mathcal{B}_f^{B_2} = \{(current\_price < bound\_price/2 \wedge Accept(current\_price) \rightarrow max\_profit, 1), (current\_price < bound\_price/2 \wedge Accept(current\_price) \rightarrow buy, 1), (current\_price > bound\_price/2 \wedge Accept(current\_price) \rightarrow \neg buy, 1), (current\_price > bound\_price/2 \wedge Accept(current\_price) \rightarrow \neg max\_profit, 1), (current\_price > bound\_price/2 \wedge Pass \rightarrow \neg buy, 1)\}$$

Let us now consider the following dialog between the seller S and the two buyers B_1 and B_2:

$S : Offer(current\_price)$. In this case, the only possible move to the agent is $Offer$. Indeed, this is required by the protocol. An agent should select the content of that move. Here again, the agent has a starting price so it will present it. At this stage, the agent does not need its decision model in order to select the move.

$B_1 and B_2 : Accept(current\_price)$. In this case, the $current\_price$ is lower than $bound\_price/2$ for the agents. The agents have an argument in favor of $Accept$. In this case, they will choose $Accept$.

$S : Offer(current\_price)$. In this case, the item is not sold to any buyer since there is more than one bid. The seller restarts the round at a higher price. Indeed, this is required by the protocol. The only possible move to the agent is $Offer$. An agent should select the content of that move. Here again, the agent has a higher price so it will present it as the cuurent price. At this stage, the agent does not need its decision model in order to select the move. Let'us suppose that the $bound\_price/2 < current\_price < bound\_price$.

B_1 : $Accept(current\_price)$. In this case, the current price $current\_price$ is lower than the price bound of the agent. In this case the agent has an argument in favor of $Accept$ because this will support its important goal $cheap$. In this case, the agent will choose $Accept$.

B_2 : $Pass$. In this case, the current price $current\_price$ is higher than $bound\_price/2$, and then the agent could not make 100% profit on the item. In this case the agent has a counter argument again $Accept$ because this will violate its important goal max_profit, and no arguments in favor of it. However, it has an argument in favor of $Pass$ since it will not violate the important goal. In this case, the agent will choose $Pass$.

S : $Attribute(current\_price)$. The only possible move of the agent is $Attribute$. Inded this is required by the protocol since there is only one bidder submitted at the current price. Moreover, the current price is higher than the reserve price. In this case the seller has an argument in favor of the content $current\_price$ since this will support its important goal $good\_price$. The seller decides then to attribute the item to the bidder B_1 and closes the round.

9 Conclusion

A considerable amount of work has been devoted to the study of dialogs between autonomous agents and to development of formal models of dialog. In most works, the definition of a protocol poses no problems and several dialog protocols have been defined even for particular applications. However, the situation is different for dialog strategies. There are very few attemps for modeling strategies. Indeed, there is no methodology and no formal models for defining them. There is even no consensus on the different parameters involved when defining a strategy.

This paper claims that during a dialog, a strategy is used only for defining the next move to play at each step of the dialog. This amounts to define the speech act to utter and its content if necessary. The strategy is then regarded as a two steps *decision process*: among all the replies allowed by the protocol, an agent should select the best speech act to play, then it should select the best content for that speech act.

The idea behind a decision problem is to define an ordering on a set of choices on the basis of the beliefs and the goals of the agent. We have argued in this paper that selecting a speech act and selecting a content of a speech act involve two different kinds of goals and two different kinds of beliefs. Indeed, an agent may have strategic goals which represent the meta-level goals of the agents about the whole dialog. An agent may have also functional goals which are directly related to the subject of the dialog. Similarly, an agent may have strategic beliefs which are meta-level beliefs about the dialog, the other agents, etc. It may also have some basic beliefs about the subject of the dialog. We have shown that the choice of the next speech is based on the strategic beliefs and the strategic goals, whereas the choice of the content is based on the basic beliefs and the functional goals.

We have then proposed a formal framework for defining strategies. This framework can be regarded as two separate systems: one of them take as input the possible replies allowed by a protocol, a set of strategic beliefs and a set of strategic goals and returns

the best speech act, and the second system takes as input a set of alternatives, a set of basic beliefs and a set of functional goals and returns the best content of a speech act. The two systems are grounded on argumentation theory. The basic idea behind each system is to construct the arguments in favour and against each choice, to compute the strength of each argument and finally to compare pairs of choices on the basis of the quality of their supporting arguments. We have shown also the agents profiles play a key role in defining principles for comparing decisions. In this paper we have presented two examples: pessimistic agents which represent very cautious agents and optimistic agents which are adventurous ones.

An extension of this work would be to study more deeply the links between the strategic and the functional goals of an agent. In this paper, we suppose implicitly that there are coherent. However, in reality it may be the case that an agent has a strategic goal which is incompatible with a functional one. Let us take the example of an agent negotiating the price of a car. This agent may have as a strategic goal to sell at the end of the dialog. It may have also the goal of selling his car with highest price. These two goals are not compatible since if the agent wants really to sell at the end its car, it should reduce the price.

References

1. L. Amgoud. A general argumentation framework for inference and decision making. In *Proceedings of the 21th Conference on Uncertainty in Artificial Intelligence*, pages 26–33, 2005.
2. L. Amgoud and N. Maudet. Strategical considerations for argumentative agents. In *Proc. of the* 10^{th} *International Workshop on Non-Monotonic Reasoning, session "Argument, Dialogue, Decision", NMR'2002*, 2002.
3. L. Amgoud and K. Souhila. On the study of negotiation strategies. In *In AAMAS 2005 Workshop on Agent Communication (AC05),*, pages 3–16, 2005.
4. P. Noriega P. Garcia J.A. Rodriguez, F.J Martin and C. Sierra. Towards a test-bed for trading agents in electronic auction markets. *AI communications, IOS Press*, 1999.
5. N. R. Jennings, P. Faratin, A. R. Lumuscio, S. Parsons, and C. Sierra. *Automated negotiation: Prospects, methods and challenges*. International Journal of Group Decision and Negotiation, 2001.
6. N. R. Jennings, E. H. Mamdani, J. Corera, I. Laresgoiti, F. Perriolat, P. Skarek, and L. Z. Varga. Using archon to develop real-word dai applications part 1. *IEEE Expert*, 11:64–70, 1996.
7. A. Kakas, N. Maudet, and P. Moraitis. Layered strategies and protocols for argumentation based agent interaction. In *Proc. AAMAS'04 1st International Workshop on Argumentation in Multi-Agent Systems, (ArgMAS'04)*, 2004.
8. S. Kraus. Strategic negotiation in multi-agent environments. *MIT Press, USA*, 2001.
9. S. Kraus, K. Sycara, and A. Evenchik. *Reaching agreements through argumentation: a logical model and implementation*, volume 104. Journal of Artificial Intelligence, 1998.
10. P. Maes. Agents that reduce work and information overload. *Communication of the ACM*, 37(7):31–40, 1996.
11. 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.
12. J. A. Rodriguez, P. Noriega, C. Sierra, and J. Padget. A java-based electronic auction house. In *Proceedings of the 2nd International Conference on the Practical Application of intelligent Agents and Multi-Agent Technology*, pages 207–224, 1997.

13. J. Rosenschein and G. Zlotkin. Rules of encounter: Designing conventions for automated negotiation among computers. *MIT Press, USA*, 1994.
14. K. Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28:203–242, 1990.
15. D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, Albany, NY, 1995.
16. M. P. Wellman. A market-oriented programming environment and its application to distributed multicommodity flow problems. *Artificial Intelligence and Research*, 1:1–23, 1993.
17. M. J. Wooldridge and N.R. Jennings. Intelligent agents: theory and practice. *The Knowledge Engineering Review*, 10:115–152, 1995.

Loose Lips Sink Ships*: a Heuristic for Argumentation

Nir Oren, Timothy J. Norman, and Alun Preece

Department of Computing Science, University of Aberdeen, Aberdeen, AB24 3UE, Scotland
noren,tnorman,apreece@csd.abdn.ac.uk

Abstract. While researchers have looked at many aspects of argumentation, an area often neglected is that of argumentation strategies. That is, given multiple possible arguments that an agent can put forth, which should be selected in what circumstances. In this paper, we propose a heuristic that implements one such strategy, namely revealing as little information as possible to other dialogue participants. After formalising the concept and presenting a simple argumentation framework in which it can be used, we show a sample dialogue utilising the heuristic. We conclude by exploring ways in which this heuristic can be employed and a discussion of future work is made which will allow for the use of our approach in more complicated, realistic dialogues.

1 Introduction

Argumentation has emerged as a powerful reasoning mechanism in many domains. One common dialogue goal is to persuade, where one or more participants attempt to convince the others of their point of view. This type of dialogue can be found in many areas including distributed planning and conflict resolution, education and in models of legal argument.

At the same time that the breadth of applications of argumentation has expanded, so has the sophistication of formal models designed to capture the characteristics of the domain. In particular, Prakken [1] has focused on legal argumentation, and has identified four layers with which such an argumentation framework must concern itself. These are:

- The *logical layer*, which allows for the representation of basic concepts such as facts about the world. Most commonly, this layer consists of some form of non–monotonic logic.
- The *dialectic layer*, in which argument specific concepts such as the ability of an argument to defeat another are represented.
- The *procedural layer* governs the way in which argument takes place. Commonly, a dialogue game [2] is used to allow agents to interact with each other.

* This was a motto used in World War II to remind people not to inadvertently reveal possibly secret information.

– The *heuristic layer* contains the remaining parts of the system. Depending on the underlying layers, these may include methods for deciding which arguments to put forth and techniques for adjudicating arguments.

While many researchers have focused on the lowest two levels (excellent surveys can be found in [3, 1, 4]), and investigation into various aspects of the procedural layer is ongoing (for example, [5, 6]), many open questions remain at the heuristic level.

In this paper, we propose a decision heuristic for an agent allowing it to decide which argument to put forth. The basis for our idea is very simple; an agent should, while attempting to win a dispute, reveal as little of what it knows as possible. This heuristic has seen use in many real world situations. For example, it has long been speculated [7] that certain government spying organisations are easily able to break most forms of encryption. However, when required to present evidence in a court of law, these organisations first pose all possible arguments that avoid revealing this information, since, if it became public knowledge that current algorithms are vulnerable, stronger algorithms will be developed that they would be unable to break.

Such a heuristic can be useful in arguments between computer agents too. Revealing too much information in a current dialogue might damage an agent's chances of winning a future argument.

In the next section, we examine existing approaches to strategy selection, after which we provide the required theoretical foundations for our approach and informally describe it. Section 3 presents our heuristic in a more formal manner. After presenting an illustrative example, we conclude the paper by looking at possible directions in which this work can be extended.

2 Background and Related Research

Argumentation researchers have recognised the need for argument selection strategies for a long time. However, the field has only recently started receiving more attention. Moore, in his work with the DC dialectical system [8], suggested that an agent's argumentation strategy should take three things into account:

– Maintaining the focus of the dispute.
– Building its point of view or attacking the opponent's one.
– Selecting an argument that fulfils the previous two objectives.

The first two items correspond to the military concept of a strategy, i.e. a high level direction and goals for the argumentation process. The third item corresponds to an agent's tactics. Tactics allow an agent to select a concrete action that fulfils its higher level goals. While Moore's work focused on natural language argument, these requirements formed the basis of most other research into agent argumentation strategies.

In 2002, Amgoud and Maudet [9] proposed a computational system which would capture some of the heuristics for argumentation suggested by Moore.

Their system requires very little from the argumentation framework. A preference ordering is needed over all possible arguments, and a level of prudence is assigned to each agent. An argument is assigned a strength based on how convoluted a chain of arguments is required to defend it from attacks by other arguments. An agent can then have a "build" or "destroy" strategy. When using the build strategy, an agent asserts arguments with a strength below its prudence level. If it cannot build, it switches to a destroy strategy. In this mode, it attacks an opponent's arguments whenever it can. While the authors note other strategies are reasonable, they make no mention of them. Shortcomings of their approach include its basis on classical propositional logic and the assumption of unbounded rationality; computational limits may affect the arguments agents decide to put forth. Finally, no attempt is made to capture the intuition that a fact defended by multiple arguments is more acceptable than one defended by fewer (the so called "accrual of evidence" argument scheme [10]).

Using some ideas from Amgoud's work, Kakas et al. [11] proposed a three layer system for agent strategies in argumentation. The first layer contains "default" rules, of the form $utterance \leftarrow condition$, while the two higher layers provide preference orderings over the rules. Assuming certain restrictions on the rules, they show that only one utterance will be selected using their system, a trait they refer to as determinism. While their approach is able to represent strategies proposed by a number of other techniques, it does require hand crafting of the rules. No suggestions are made regarding what a "good" set of rules would be.

In [12], Amgoud and Prade examined negotiation dialogues in a possibilistic logic setting. An agent has a set of goals it attempts to pursue, a knowledge base representing its knowledge about the environment, and another knowledge base which is used to keep track of what it believes the other agent's goals are. The authors then present a framework in which these agents interact which incorporates heuristics for suggesting the form and contents of an utterance, a dialogue game allowing agents to undertake argumentation, and a decision procedure to determine the status of the dialogue. Their heuristics are of particular interest as they are somewhat similar to the work we investigate here. One of their heuristics, referred to as the criterion of partial size, uses as much of an opponent's knowledge as possible, while the heuristic referred to as the criterion of total size attempts to minimise the length of an argument. Apart from operating in a negotiation rather persuasion setting, their heuristics do not consider the amount of information revealed from one's own knowledge base.

Cayrol et al. [13] have investigated a heuristic which, in some respects, is similar to ours. In their work, an agent has two types of arguments in its knowledge base. The first, referred to as unrestricted arguments, is used as necessary. The second type, consisting of so called restricted arguments, is only used when necessary to defend unrestricted arguments. They provide an extension of Dung's argumentation framework which allows one to determine extensions in which a minimal amount of restricted knowledge is exposed, thus providing a reasoning procedure representing minimum information exposure. As we discuss in Section

5, argumentation frameworks based on Dung's work leave arguments as very abstract entities, making it difficult to apply the framework to some situations. Furthermore, unlike the work detailed in this paper, Cayrol et al. do not present a dialogical setting in which the heuristic can operate. Also, since their restricted arguments can only be used to defend unrestricted arguments, it is not clear how their heuristic will function in situations where all knowledge is restricted.

In [14], Bench-Capon describes a dialogue game based on Toulmin's work. He identifies a number of stages in the dialogue in which an agent might be faced with a choice, and provides some heuristics as to what argument should be advanced in each of these cases. Only an informal justification for his heuristics is provided.

Apart from guiding strategy, heuristics have seen other uses in dialogue games. Recent work by Chesñevar et al. [15] has seen heuristics being used to minimise the search space when analysing argument trees. Argument schemes [16] are well used tools in argumentation research, and can be viewed as a form of heuristic that guides the reasoning procedure.

3 The Framework and Heuristic

In many realms of argument, auxiliary considerations (apart from simply winning or losing the argument) come into play. In many scenarios, one such consideration is to minimise the information provided to other parties. For example, in a court case between a government and some alleged terrorists, the government might not be willing to reveal the sources of some of its evidence. We thus propose a simple heuristic to guide an agent in a dialogue: when faced with a number of possible arguments to put forth, the one that should be advanced is the one that exposes as little of the agent's internal knowledge as possible. Many extensions and refinements to this heuristic are possible, some of which are discussed in Section 5. However, in this paper we focus on the most simple form of the heuristic for the sake of perspicaciousness.

In formalising our heuristic, we borrow many ideas from other formal argumentation systems (e.g. [17–20]).

We formalise our system in two parts. First we specify the argumentation system itself, and then the heuristic is described, on the basis of this argumentation system.

3.1 The Argumentation Framework

Argumentation takes place over the language Σ, which contains propositional literals and their negation.

Definition 1. ***Argument*** *An argument is a pair (P, c), where $P \subseteq \Sigma \cup \{\top\}$ and $c \in \Sigma$ such that if $x \in P$ then $\neg x \notin P$. We define $Args(\Sigma)$ to be the set of all possible arguments in our language.*

P represents the premises of an argument (also referred to as an argument's support), while c stands for an argument's conclusion. Informally, we can read an argument as stating "if the conjunction of its premises holds, the conclusion holds". Facts can be represented using the form $(\top, a)$.

Arguments interact by supporting and attacking each other. Informally, when an argument attacks another, it renders the latter's conclusions invalid.

Definition 2. ***Attack*** *An argument $A = (P_a, c_a)$ attacks $B = (P_b, c_b)$ if $c_a = \neg c_b$ or $\exists f \in P_b$ such that $f \equiv \neg c_a$. For convenience, we write this as $attacks(A, B)$.*

An argument is only relevant to an instance of argumentation if its premises are true. We call such an argument *justified*. However, a simple definition of this concept can cause problems when it comes to self attacking (or self defending) arguments, as well as circular reasoning, and care must thus be taken when describing this concept. Before doing so, we must (informally) describe the proof theory used to determine which literals and arguments are in effect at any time.

The idea behind determining what arguments and literals are admissible at any time is as follows. We start by looking at the facts, and determining what knowledge can be derived from them by following chains of argument. Whenever a conflict occurs (i.e. we are able to derive both x and $\neg x$), we remove these literals from our derived set. Care must be taken to also get rid of any arguments (and further facts) derived from any conflicting literals. To do this, we keep track of the conflicting literals in a separate set, whenever a new conflict arises, we begin the knowledge determination process afresh, never adding any arguments whose conclusions are in the conflicting set to the knowledge set. The philosophical and practical ramifications of this approach will be discussed in Section 5.

More formally, an instance of the framework creates two sets $J \subseteq Args(\Sigma)$ and $C \subseteq \Sigma$ representing justified arguments and conflicts respectively.

Definition 3. ***Derivation*** *An argument $A = (P_a, c_a)$ is derivable from a set S given a conflict set C (written $S, C \vdash A$) iff $c_a \notin C$ and $(\forall p \in P_a : (\exists s \in S$ such that $s = (P_s, p)$ and $p \notin C)$ or $P_a = \{\top\})$.*

Clearly, we need to know what elements are in C. Given a knowledge base of arguments $\kappa \subseteq Args(\Sigma)$, this can be done with the following reasoning procedure:

$$J_0 = \{A | A \in \kappa \text{ such that } \{\}, \{\} \vdash A\}$$
$$C_0 = \{\}$$

Then, for $i > 0, j = 1 \ldots i$, we have:

$$C_i = C_{i-1} \cup \{c_A, \neg c_A | \exists A = (P_A, c_A), B = (P_B, \neg c_A) \in J_{i-1} \text{ such that } attacks(A, B)\}$$

$$X_{i0} = \{A | A \in \kappa \text{ and } \{\}, C_i \vdash A\}$$

$$X_{ij} = \{A | A \in \kappa \text{ and } X_{i(j-1)}, C_i \vdash A\}$$

$$J_i = X_{ii}$$

The set X allows us to recompute all derivable arguments from scratch after every increment of i[1]. Since i represents the length of a chain of arguments, when $i = j$ our set will be consistent to the depth of our reasoning, and we may assign all of these arguments to J. Eventually, $J_i = J_{i-1}$ (and $C_i = C_{i-1}$) which means there are no further arguments to find. We can thus define the conclusions reached by a knowledge base κ as $K = \{c | A = (P, c) \in J_i\}$, for the smallest i such that $J_i = J_{i+1}$. We will use the shorthand $K(\kappa)$ and $C(\kappa)$ to represent those literals which are respectively derivable from, or in conflict with a knowledge base κ.

We illustrate this algorithm with two examples (not all steps are shown):

Example 1. $\kappa = \{(\top, s), (s, t), (t, \neg s)\}$
$J_0 = \{(\top, s)\}$, $C_1 = \{\}$, $J_1 = X_{11} = \{(\top, s), (s, t)\}$
...
$J_2 = (\top, s), (s, t), (t, \neg s)$
$C_3 = \{s, \neg s\}$
$X_{30} = \{\} \ldots J_4 = J_3 = \{\}$

Example 2. $\kappa = \{(\top, a), (\top, b), (a, c), (b, d), (c, \neg d)\}$
$J_0 = \{(\top, a), (\top, b)\}$
$X_{10} = J_0$, $J_1 = X_{11} = \{(\top, a), (\top, b), (a, c), (b, d)\}$
...
$J_2 = X_{22} = \{(\top, a), (\top, b), (a, c), (b, d), (c, \neg d)\}$
...
$C_3 = \{(d, \neg d)\}$,
$J_4 = J_3 = X_{33} = X_{32} = \{(\top, a), (\top, b), (a, c)\}$

3.2 The Dialogue Game and Heuristic

Agents engage in a dialogue using the argumentation framework described above in an attempt to persuade each other of certain facts. An agent has a private knowledge base (KB) as well as a goal literal g. The environment, apart from containing agents, contains a public knowledge base which takes on a role similar to a global commitment store[2], and is thus referred to as CS.

Definition 4. ***Environment and agents*** *An Agent $\alpha \in Agents$ is a triple $(Name, KB, g)$ where $KB \subseteq Args(\Sigma)$ and $g \in \Sigma$. Name is a unique label assigned to the agent. Given n agents in the system, we assume they are labelled $Agent_0 \ldots Agent_{n-1}$.*

The environment is a pair $(Agents, CS)$ where Agents is the set of agents participating in the dialogue and $CS \subseteq Args(\Sigma)$

[1] This allows us to get rid of long invalid chains of arguments, as well as detect and eliminate arbitrary loops.

Agents take turns to put forward a line of argument consisting of a number of individual arguments. For example, an agent could make the utterance $\{(\top, a), (a, b)\}$. Alternatively, an agent may pass. The dialogue ends when CS has remained unchanged for n turns i.e. after all players have had a chance to modify it, but didn't (this is normally caused by all agents having passed consecutively). Once this has happened, the acceptable set of arguments is computed over the CS, and the status of each agent's goal can be determined, allowing one to compute the winners of the game.

Definition 5. ***Turns and utterances*** *The function*

$$turn : Environment \times Name \rightarrow Environment$$

takes an environment and an agent label, and returns a new environment containing the result of the utterance ($utterance : Environment \times Name \rightarrow 2^{Args(\Sigma)}$*) made by the labelled agent during its turn.*

$$turn(Environment, \alpha) = (Agents, \{CS \cup utterance(Environment, \alpha)\})$$

During turn i, we will set $\alpha = Agent_{i \bmod n}$, where n is the number of agents taking part in the dialogue. We will detail the *utterance* function for a rational agent below. Before doing so, we define the dialogue game itself. Each turn in the dialogue game results in a new public commitment store, which can be used by the agents in later turns.

Definition 6. ***Dialogue game*** *The dialogue game is defined as*
$turn_0 = turn((Agents, CS_0), Agent_0)$
$turn_i = turn(turn_{i-1}, Agent_{i \bmod n})$ *for* $i = 1, 2, \ldots$

The game ends when $turn_i \ldots turn_{i-n+1} = turn_{i-n}$.

CS_0 is dependent on the system, and contains any arguments that are deemed to be common knowledge. Also, note that the null utterance $\{\}$ is defined to be a pass.

By using the derivation procedure described in the previous section, agents can

- Determine, by looking at CS, what literals are in force and in conflict.
- Determine, by combining CS with parts of their own knowledge base, what literals they can prove (or cause to conflict).

By doing the latter, together with looking at the number of literals introduced into K and C, an agent can both determine how much information it reveals by putting forth an argument, and narrowing down the range of possible arguments it will submit (though possibly not to a unique argument).

An agent's first goal is to win the argument by proving its point. If it cannot do so, it will try to obtain a draw. Winning an argument requires that $g \in K(CS)$, while a draw results if no conclusions can be reached regarding the status of g, i.e. $g \in C(CS)$ or $\{g, \neg g\} \cap K(CS) = \{\}$.

Definition 7. ***Winning arguments*** *An agent* $\alpha = (Name, KB, g)$ *has a set of winning arguments defined as*
$Win = \{A \in 2^{KB} |\ g \in K(A \cup CS)$ *and if* $A \neq \{\}, \{\} \notin A\}$

Definition 8. ***Drawing arguments*** *An agent* $\alpha = (Name, KB, g)$ *has a set of drawing arguments defined as*
$Draw = \{A \in 2^{KB} |\ (g \in C(A \cup CS)$ *or* $\{g, \neg g\} \cap K(A \cup CS) = \{\})$ *and if* $A \neq \{\}, \{\} \notin A\}$

An *information aware* agent is one that attempts to win an argument while minimising the amount of information it exposes.

Definition 9. ***Information exposure*** *The information exposed by an agent* $\alpha = (Name, KB, g)$ *making an utterance* $A \in 2^{KB}$ *can be defined as follows:*

$$Inf = |K(A \cup CS) + C(A \cup CS)| - |K(CS) + C(CS)|$$

Where $K(X)$ *and* $C(X)$ *are the sets of literals obtained by running the reasoning process over the set of arguments* X*.*

An agent prefers a winning strategy over one which leads to a draw, and orders its winning strategies by the amount of information they reveal. This may still lead to multiple possible arguments, in which case other heuristics (such as choosing the shortest possible chain of arguments) may be employed to select a unique argument. We do not discuss these other heuristics in this paper. This preference over arguments can be captured in the following definition:

Definition 10. ***Possible arguments*** *The set of possible arguments an agent would utter is defined as*

$$PA = \begin{cases} A \in Win \text{ s.t. } Inf(A) = min(Inf(B)), B \in Win. & Win \neq \{\} \\ A \in Draw \text{ s.t. } Inf(A) = min(Inf(B)), B \in Draw & Win = \{\}, \\ & Draw \neq \{\} \\ \{\} & Win = \{\}, \\ & Draw = \{\} \end{cases}$$

The utterance an agent makes is one of these possible arguments: utterance $\in PA$

It should be noted that a "pass", i.e. {} might still be uttered as part of the Win or Draw strategy.

When the game is over, all that remains to be done is determine who (if anyone) won the argument:

Definition 11. ***Victory conditions*** *The set of winning agents is* $Agents_{win} = \{\alpha = (Name, KB, g) \in Agents |\ g \in K(CS)\}$*. Similarly, the set of drawing agents is* $Agents_{draw} = \{\alpha = (Name, KB, g) \in Agents |\ g \in C(CS)$ *or* $\neg g \notin K(CS)$ *and* $\alpha \notin Agents_{win}\}$*. All other agents are in the losing set:* $Agents_{lose} = \{\alpha \in Agents | \alpha \notin (Agents_{win} \cup Agents_{draw})\}$

Literals in $K(CS)$ at the end of the game are those agreed to be in force by all the agents.

In this section, we have defined an argument framework which allows an agent to determine which arguments are in force by performing forward chaining on a knowledge base of arguments, beginning with those arguments which have no premises. We then described a simple dialogue together with a reasoning procedure which allows an agent to put forth arguments revealing as little information as possible. During each move, an agent picks which arguments to reveal from its private knowledge base by computing what literals are in conflict (via $C(CS)$) and which literals would be deemed accepted (by using $K(CS)$) for the new CS containing the arguments it would put forth. If it determines that there are a number of possible arguments it could submit that would win (or, if no winning arguments exist, draw) it the game, it chooses to utter the set of arguments which minimise the amount of information it reveals[2].

Having defined our system, we can now look at its features. In the next section we provide a small example of a dialogue, after which we provide a more in-depth discussion of the framework, heuristic, and features that emerge by studying the example.

4 Example

To increase readability, we present our example in a somewhat informal manner. The argument focuses on the case for, or against, the possibility of weapons of mass destruction (WMDs) existing at some location.

We assume a two party dialogue (with $Agent_0 = \alpha$, $Agent_1 = \beta$), and describe only one agent's knowledge base. At the start of the game, our agent has the following facts in its private knowledge base KB:

$(\top, Chemicals)$	Chemicals exist
$(\top, Photo)$	Photos exist
$(\top, Newspaper)$	Newspaper articles exist
$(\top, Factory)$	Factories exist
$(\top, \neg Medicine)$	Medicine is not being produced
$(Newspaper, WMD)$	If newspapers say so, then WMDs exist
$(\{Photo, Factory\}, \neg WMD)$	Pictures of factories mean WMDs don't exist
$(Chemicals, \neg Medicine)$	Chemicals mean medicine isn't being produced
$(\{Chemicals, Factory\}, WMD)$	Chemicals and factories mean WMDs exist

[2] A Prolog implementation of the argumentation framework, dialogue game and heuristic is available at `http://www.csd.abdn.ac.uk/~noren`

Then the following dialogue takes place (α's goal is the literal WMD):

(α) $(\top, Newspaper), (Newspaper, WMD)$	1
(β) $(\top, \neg Newspaper), (\top, Factory), (Factory, Medicine),$ $(Medicine, \neg WMD)$	2
(α) $(\top, Chemicals), (Chemicals, \neg Medicine),$ $(\{Chemicals, Factory\}, WMD)$	3
(β) $\{\}$	4
(α) $\{\}$	5

Informally, agent α claims that since newspaper articles about the subject exist, WMDs must exist (as per the newspaper's claims). β responds by saying that it has not seen any articles, but that since he knows that factories exist, and that these factories produce medicines, WMDs are not present (possibly implying that any evidence found is due to these medicines). α counters that chemicals were found, and that the finding of these is incongruent with the presence of medicines, also stating that the presence of the factories and the chemicals is proof regarding the existence of WMDs. β has no response to this, and after α stays silent, the game ends with α successfully proving his goal.

Before examining the dialogue in detail, we can discuss a few interesting, global properties of the heuristic:

- An agent that knows it will lose an argument is still able to win it by assuming that its opponent does not have access to the same information it does. It could be argued that passing to draw (or win) a game when it has not revealed all its information is tantamount to lying.
- The heuristic is different to the "Occam's razor" heuristic that has often appeared in the literature. The latter proposes that the shortest argument be put forth first, while we are able to present longer arguments if they reveal less information. In many cases however, the two heuristics can coincide regarding what utterance should be made next.

Let us examine line 3 in more detail. Before this line, our public knowledge base, CS, contained the following arguments:

$$(\top, Newspaper)\ (Newspaper, WMD)\ (\top, \neg Newspaper)$$
$$(\top, Factory)\quad (Factory, Medicine)\ (Medicine, \neg WMD)$$

Clearly, apart from having an information exposure value (*Inf* in Definition 9) of one, an argument such as $(\top, Photo)$ will not be considered as it is not part of the winning or drawing set. The argument chosen has an information exposure value of two (as the literals $Chemicals$ and $\neg Medicine$ are added to CS), but was chosen as it is part of the winning set. Note that an argument such as

$$(Chemicals, \neg Medicine),\qquad (\top, Chemicals)$$
$$(\{Chemicals, Factory\}, WMD),\ (\top, Photo)$$

is also part of the winning set, but has a higher information exposure value.

The argument $(Chemicals, \neg Medicine), (\top, Chemicals)$ Belongs to the drawing set, and has an information exposure value of one.

The argument $(\top, Photo), (\{Photo, Factory\}, \neg WMD)$ has an information exposure value of 2, and, if suggested after line 3 of the dialogue, is part of the drawing set. It will thus not be selected as an utterance.

Since our winning set is non-empty, our agent was forced to pick an argument from there. By modifying Definition 10, we could define a number of different classes of agents with a range of preferences based on winning, drawing or losing an argument and revealing different amounts of information.

5 Discussion

This section examines the argumentation framework and the heuristic, tying it back to the concept of an argumentation strategy as proposed by Moore. We also examine some of the novel features of argument that emerge when dialogue takes place in the framework using the heuristic, and propose avenues for future research.

Our approach seems to share much in common with the "sceptical" approach to argumentation. When arguments conflict, we refuse to decide between them, instead ruling them both invalid. This means that our reasoning procedure is not complete, given the (rather convoluted) set of arguments

$$(\top, A), (\top, B), (A, \neg B), (B, \neg A), (A, C), (B, C), (\neg A, C), (\neg B, C)$$

we see that C should hold, but doesn't. Other argumentation systems (namely those utilising the unique–status–assignment approach [4]) are similarly incomplete, leaving this an open area for future research. Our sceptical approach does yield a consistent system, as no conflicting arguments will remain in the final set of arguments.

The simplicity of our approach means that only specific types of arguments can be represented (namely, those whose premises are a conjunction of literals, and whose conclusion is a single literal). However, as seen in the example, even with this limitation, useful arguments can still emerge.

We developed our own argumentation framework rather than using an existing one for a number of reasons, including:

- The abstract nature of many frameworks (e.g. [17]) makes arguments atomic concepts. We needed a finer level of granularity so as to be able to talk about which facts are exposed (allowing us to measure the amount of information revealed during the dialogue process). Less abstract frameworks (e.g. [21, 18]), while looking at concepts such as derivability of arguments still primarily focus on the interactions between arguments.
- Almost all other frameworks define higher level concepts in terms of arguments attacking, defeating and defending one another. For us, the concept of one argument justifying another is critical, together with the concept of attack.

– Other argumentation systems contain concepts which we do not require, such as a preference ordering over arguments.
– Approaches such as [18] divide their argument constructs into defeasible and indefeasible sets, with a consistency requirement on the indefeasible set, and then provide for default reasoning over the defeasible arguments. Our framework only takes the defeasible nature of arguments into account, ignoring default reasoning.

While representing the heuristic using one of the other approaches is (probably) not impossible, it appears to be more difficult than by using our own system.

Looking at Moore's three criteria for an agent argumentation strategy, we see that our heuristic fulfils its requirements. If the focus of the argument were not maintained, more information would be given than is strictly necessary to win, thus fulfilling the first requirement. Both the second and third requirements are clearly met by the decision procedure for which argument to advance described in Definition 10.

Investigating the use of the heuristic in more complex settings (by either increasing the representational power of the framework, or by representing the heuristic in another argumentation framework) is one possible direction of future work.

One disadvantage of our approach is that at each move, we evaluate possible arguments from the powerset of an agent's private knowledge. This leads to an exponential complexity in our algorithm. While simple techniques can be applied to shrink the size of the powerset, more complicated approaches which can further reduce the algorithm's running costs need to be examined.

Making the heuristic more realistic is another area we are investigating. For example, rather than treating all information equally, we could assign a numerical cost to each literal, and attempt to minimise this cost while winning the argument. Another avenue for future research involves determining how this heuristic can best be combined with techniques for resource bounded reasoning. Allowing agents to communicate with each other privately, rather than with all dialogue participants allows for a number of knowledge bases to exist. An agent might have certain information it is willing to reveal to some, but not all participants, and investigating strategies for such dialogues is another rich research area.

6 Conclusions

In this paper we proposed a heuristic for argumentation based on revealing as little information as possible to the other dialogue participants. While such an argumentation strategy arises in many real world situations, we are not familiar with any application that explicitly makes use of this technique. To study the heuristic, we proposed an argumentation framework that allowed us to focus on it in detail. Several novel features emerged from the interplay between the heuristic and the framework, including the the ability of an agent to win an argument that

it should (given all possible information) not be able to win. While we have only examined a very abstract model utilising the heuristic, we believe that many interesting extensions are possible, and many unanswered questions remain.

Acknowledgements

This work is partly funded by the DTI/EPSRC E-Science Core Program and British Telecom, via a grant for the CONOISE-G project (http://www.conoise.org), a multi-university collaboration between Aberdeen, Cardiff and Southampton universities, and BT. We would also like to thank Madalina Croitoru for her useful feedback on earlier versions of this paper.

References

1. Prakken, H., Sartor, G. In: Computational Logic: Logic Programming and Beyond. Essays In Honour of Robert A. Kowalski, Part II. Volume 2048 of Lecture Notes in Computer Science. Springer-Verlag, Berlin (2002) 342–380
2. Walton, D.N., Krabbe, E.C.W.: Commitment in Dialogue. State University of New York Press (1995)
3. Chesñevar, C.I., Maguitman, A.G., Loui, R.P.: Logical models of argument. ACM Computing Surveys **32**(4) (2000) 337–383
4. Prakken, H., Vreeswijk, G.: Logics for defeasible argumentation. In Gabbay, D., Guenthner, F., eds.: Handbook of philosophical logic, 2nd Edition. Volume 4. Kluwer Academic Publishers (2002) 218–319
5. Walton, D.N.: Legal argumentation and evidence. Penn State Press (2002)
6. McBurney, P., Parsons, S.: Risk agoras: Dialectical argumentation for scientific reasoning. In: Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, Stanford, USA (2000) 371–379
7. Schneier, B.: Applied Cryptography: Protocols, Algorithms, and Source Code in C. John Wiley & Sons, Inc., New York, NY, USA (1993)
8. Moore, D.: Dialogue game theory for intelligent tutoring systems. PhD thesis, Leeds Metropolitan University (1993)
9. Amgoud, L., Maudet, N.: Strategical considerations for argumentative agents (preliminary report). In: NMR. (2002) 399–407
10. Prakken, H.: A study of accrual of arguments, with applications to evidential reasoning. In: Proceedings of the tenth International Conference on Artificial Intelligence and Law. (2005) 85–94
11. Kakas, A.C., Maudet, N., Moraitis, P.: Layered strategies and protocols for argumentation-based agent interaction. In: ArgMAS. (2004) 64–77
12. Amgoud, L., Prade, H.: Reaching agreement through argumentation: a possiblistic approach. In: Proceedings of KR 2004. (2004)
13. Cayrol, C., Doutre, S., Lagasquie-Schiex, M.C., Mengin, J.: " minimal defence" : a refinement of the preferred semantics for argumentation frameworks. In: Proceedings of NMR-2002. (2002)
14. Bench-Capon, T.J.: Specification and implementation of Toulmin dialogue game. In: Proceedings of JURIX 98. (1998) 5–20

15. Chesñevar, C.I., Simari, G.R., Godo, L.: Computing dialectical trees efficiently in possibilistic defeasible logic programming. In Baral, C., Greco, G., Leone, N., Terracina, G., eds.: Logic Programming and Nonmonotonic Reasoning, 8th International Conference, LPNMR 2005, Diamante, Italy, September 5-8, 2005, Proceedings (LNCS 3662). Lecture Notes in Computer Science, Springer (2005) 158–171
16. Reed, C.A., Walton, D.N.: Applications of argumentation schemes. In Hansen, H.V., Tindale, C.W., Blair, J.A., Johnson, R.H., Pinto, R.C., eds.: Proceedings of the 4th Conference of the Ontario Society for the Study of Argument (OSSA2001), Windsor, Canada. (2001) CD ROM
17. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence **77**(2) (1995) 321–357
18. Prakken, H., Sartor, G.: A dialectical model of assessing conflicting arguments in legal reasoning. Artificial Intelligence and Law **4** (1996) 331–368
19. Pollock, J.L.: Perceiving and reasoning about a changing world. Computational Intelligence **14** (1998) 498–562
20. Verheij, B.: DefLog: On the logical interpretation of prima facie justified assumptions. Journal of Logic and Computation **13**(3) (2003) 319–416
21. Simari, G.R., Loui, R.P.: A mathematical treatment of defeasible reasoning and its implementation. Artif. Intell. **53**(2-3) (1992) 125–157

Strategic and Tactic Reasoning for Communicating Agents

Jamal Bentahar, Mohamed Mbarki, and Bernard Moulin

Laval University, Depart. of Computer Science and Software Engineering, Canada
{jamal.bentahar,mohamed.mbarki,bernard.moulin}@ift.ulaval.ca

Abstract. The purpose of this paper is to address the strategic and tactic issues in agent communication. Strategic reasoning enables agents to decide about the global communication plan in terms of the macro-actions to perform in order to achieve the main conversational goal. Tactic reasoning, on the other hand, allows agents to locally select, at each moment, the most appropriate argument according to the adopted strategy. Previous efforts at defining and formalizing strategies for argumentative agents have often neglected the tactic level and the relation between strategic and tactic levels. In this paper, we propose a formal framework for strategic and tactic reasoning for rational communicating agents and the relation between these two kinds of reasoning. This framework is based on our social commitment and argument approach for agent communication.

1 Introduction

Recent years have seen an increasing interest in agent communication. Using argumentation theories in this domain seems a promising way to develop more flexible and efficient agent communication mechanisms [1, 3, 4, 14, 16, 26]. The idea is to provide agents with reasoning capabilities allowing them to decide about the appropriate communicative acts to perform in order to achieve some conversational goals in different dialogue types [18, 19, 21, 22, 24].

In order to improve the agent communication efficiency, we propose in this paper a formal framework addressing strategic and tactic issues. A strategy is defined as *a global cognitive representation of the means of reaching some goals* [31]. Tactic is *basically the mean to reach the aims fixed at the strategic level* [20]. For example, according to Moore [20], maintaining focus of the dispute in a persuasive dialogue, and building a point of view or destroying the opponent's one refer to strategy, whereas selecting methods to fulfill these two objectives refers to tactic. In our framework, the agents' strategic and tactic reasoning is based upon their argumentative capabilities. Agents use this reasoning in order to achieve their conversational goals. Strategic reasoning allows agents to plan the global line of communication in terms of the sub-goals to achieve, whereas tactic reasoning allows them to locally select, at each moment, the most appropriate argument according to the adopted strategy. In other words, strategy is considered at the global level (in which direction the communication

can advance) and the tactics are considered at the local level (which move to be selected next).

In recent years, some significant proposals have explored the strategic reasoning of argumentative agents [2, 15, 25, 27]. However, the tactical reasoning has often been neglected or simplified to a private preference policy like in [15]. In addition, as outlined in [10], the problem of coming up with an optimal communication strategy that ensures beneficial interaction outcomes for the participating agents is still an open problem. We think that an efficient agent communication requires to address both the strategic and tactic levels and the relation between these two levels. The objective of this paper is to investigate this issue for argumentative-based agent communication. Our contribution starts by formalizing strategic and tactic reasoning and the relation between them using a management theory. At the tactical level, we develop a theory allowing agents to select the most relevant argument at each moment according to the adopted strategy. In addition, our approach enables agents to take into account the context of conversation and to be able to backtrack if some choices are not appropriate.

Paper overview. In Section 2, we introduce the fundamental ideas of our agent communication approach based on social commitments and arguments. In Section 3, we present the strategic level of our framework and its relation with the tactic level. In Section 4, we present the tactic reasoning. In Section 5, we illustrates our ideas by an example. In Section 6, we compare our framework to related work and conclude the paper.

2 Agent Communication Approach

Our agent communication approach is based on the philosophical notion of social commitments (SCs) [30]. A SC is an engagement made by an agent (called the *debtor*), that some fact is true or that some action will be performed. This commitment is directed to a set of agents (called *creditors*). A SC is an obligation in the sense that the debtor must respect and behave in accordance with this commitment. Commitments are social in the sense that they are expressed publicly and governed by some rules. This means that they are observable by all the participants. The main idea is that a speaker is committed to a statement when he made this statement or when he agreed upon this statement made by another participant and acts accordingly. For simplification reasons, we suppose that we have only one creditor. Thus, we denote a SC as follows: $SC(Ag_1, Ag_2, t, \varphi)$ where Ag_1 is the debtor, Ag_2 is the creditor, t is the time associated with the commitment, and φ its content. Logically speaking, a SC is a public propositional attitude. The content of a SC can be a proposition or an action. A detailed taxonomy of the SCs is presented in [5] and their logical semantics is developed in [6].

In order to model the dynamics of conversations in our framework, we interpret a *speech act* as an action performed on a SC or on a SC content. A speech act is an abstract act that an agent, the *speaker*, performs when producing an

utterance U and addressing it to another agent, the *addressee* [29]. According to speech act theory [29], the primary units of meaning in the use of language are not isolated propositions but rather speech acts of the type called *illocutionary acts*. Assertions, questions, orders and declarations are examples of these illocutionary acts. In our framework, a speech act can be defined using BNF notation as follows.

Definition 1 (Speech Acts). $SA(i_k, Ag_1, Ag_2, t_u, U) =_{def}$
$Act(Ag_1, t_u, SC(Ag_1, Ag_2, t, \varphi))$
$|Act-cont(Ag_1, t_u, SC(Ag_i, Ag_j, t, \varphi))$
$|Act(Ag_1, t_u, SC(Ag_1, Ag_2, t, \varphi))$ &
$Act-cont(Ag_1, t_u, SC(Ag_i, Ag_j, t, \varphi))$

where SA is the abbreviation of "Speech Act", i_k is the identifier of the speech act, Ag_1 is the speaker, Ag_2 is the addressee, t_u is the utterance time, U is the utterance, Act indicates the action performed by the speaker on the commitment: $Act \in \{Create, Withdraw, Violate, Satisfy\}$, $Act-cont$ indicates the action performed by the speaker on the commitment content: $Act\text{–}cont \in \{Accept\text{–}cont, Refuse\text{–}cont, Challenge\text{–}cont, Justify\text{–}cont, Defend\text{–}cont, Attack\text{–}cont\}$, $i, j \in \{1, 2\}$, $i \neq j$, the meta-symbol "&" indicates the logical conjunction between actions performed on social commitments and social commitment contents.

The definiendum $SA(i_k, Ag_1, Ag_2, t_u, U)$ is defined by the definiens $Act(Ag_1, t_u, SC(Ag_1, Ag_2, t, \varphi))$ as an action performed by the speaker on its SC. The definiendum is defined by the definiens $Act\text{–}cont(Ag_1, t_u, SC(Ag_i, Ag_j, t, \varphi))$ as an action performed by the speaker on the content of its SC ($i = 1, j = 2$) or on the content of the addressee's SC ($i = 2, j = 1$). Finally, the definiendum is defined as an action performed by the speaker on its SC and as an action performed by the speaker on the content of its SC or on the content of the addressee's SC. These actions are similar to the moves proposed in [28].

We notice here that using a social (public) approach as a theoretical foundation does not mean that agents do not reason on their private mental states or on the addressees' mental states (beliefs, intention, etc.). According to Definition 1, this public approach is used at the semantical level in order to interpret communicative acts as social commitments and not as mental states (see [6, 7] for more details about the public semantics). Public and mental (private) approaches are not contradictory, but rather, they are complementary. In our framework, agents reason on SCs and on their beliefs about the addressees' beliefs and preferences (see Section 4.2). These beliefs are not public, but they can, for example, be inferred from past interactions.

Our approach is also based on argumentation. Several argumentation theories and frameworks have been proposed in the literature (see for example [9, 17, 23]). An argumentation system essentially includes a logical language $\pounds$, a definition of the argument concept, a definition of the attack relation between arguments, and finally a definition of acceptability. We use the following definitions from [1]. Here Γ indicates a possibly inconsistent knowledge base with no deductive closure, and $\vdash$ stands for classical inference.

Definition 2 (Argument). *An argument is a pair (H, h) where h is a formula of $\pounds$ and H a subset of Γ such that: i) H is consistent, ii) $H \vdash h$ and iii) H is minimal, so that no subset of H satisfying both i and ii exists. H is called the support of the argument and h its conclusion.*

Definition 3 (Attack). *Let (H_1, h_1), (H_2, h_2) be two arguments. (H_1, h_1) attacks (H_2, h_2) iff $H_2 \vdash \neg h_1$. In other words, an argument is attacked if and only if there exists an argument for the negation of its conclusion.*

The link between commitments and arguments enables us to capture both the public and reasoning aspects of agent communication. This link is explained as follows. Before committing to some fact h being true (i.e. before creating a commitment whose content is h), the speaker agent must use its argumentation system to build an argument (H, h). On the other side, the addressee agent must use its own argumentation system to select the answer it will give (i.e. to decide about the appropriate manipulation of the content of an existing commitment). For example, an agent Ag_1 accepts the commitment content h proposed by another agent Ag_2 if it is able to build an argument supporting this content from its knowledge base. If Ag_1 has an argument $(H', \neg h)$, then it refuses the commitment content proposed by Ag_2. However, how agents can select the most appropriate argument at a given moment depends on its tactic. This aspect is detailed in Section 4. The social relationship that exists between agents, their reputations and trusts also influence the acceptance of the arguments by agents. However, this aspect will not be dealt with in this paper. The argumentation relations that we use in our model are thought of as actions applied to commitment contents. The set of these relations is: $\{Justify, Defend, Attack\}$.

In order to implement this communication model, we use an agent architecture composed of three layers: the mental layer, the social layer, and the reasoning layer. The mental layer includes beliefs, desires, goals, etc. The social layer captures social concepts such as SCs, conventions, roles, etc. Agents must use their reasoning capabilities to reason about their mental states before acting on SCs. The agent's reasoning capabilities are represented by the reasoning layer using an argumentation system. Our conversational agent architecture also involves general knowledge, such as knowledge about the conversation subject. Agents can also reason about their preferences in relation to beliefs. The idea is to capture the fact that some facts are more strongly believed. For this reason, we assume, like in [1], that any set of facts has a preference order over it. We suppose that this ordering derives from the fact that the agent's knowledge base denoted by Γ is stratified into non-overlapping sets $\Gamma_1, \ldots, \Gamma_n$ such that facts in Γ_i are all equally preferred and are more preferred than those in Γ_j where $i < j$. We can also define the preference level of a subset of Γ whose elements belong to different non-overlapping sets as follows.

Definition 4 (Preference Level). *The preference level of a nonempty subset γ of Γ denoted by level(γ) is the number of the highest numbered layer which has a member in γ.*

Example 1. Let $\Gamma = \Gamma_1 \cup \Gamma_2$ with $\Gamma_1 = \{a, b\}$ and $\Gamma_2 = \{c, d\}$ and $\gamma = \{a\}$ and $\gamma' = \{a, d\}$. We have: $level(\gamma) = 1$ and $level(\gamma') = 2$.

3 Strategic Reasoning

According to the *theory of constraints* proposed by Goldratt [13], the common view about strategy is that of *setting the high objectives of an initiative*. The strategy dictates the direction of all activities. Tactics, on the other hand, are *the chosen types of activities needed to achieve the objectives*. Indeed, tactics allow us to implement and accomplish the strategy. In management, a strategic plan defines the mission, vision and value statements of an enterprize. Once objectives are defined, alternative strategies can be evaluated. While a goal or an objective indicates "what" is to be achieved, a strategy indicates "how" that achievement will be realized. Strategies, therefore, depend on goals and objectives. Tactics are the steps involved in the execution of the strategy.

Our strategic and tactic framework for agent communication is based on this vision. In this framework, the dialogue strategy is defined in terms of the sub-goals to be achieved in order to achieve the final conversational goal. The sub-goals represents the macro-actions to be performed. This reflects the global vision and the direction of the dialogue. The strategy has a dynamic nature in the sense that the sub-goals can be elaborated while the dialogue advance. The strategy can also be *adjusted* when more information becomes available. The tactics represent the micro-actions to be performed in order to achieve each elaborate (elementary) sub-goal. This reflects the local vision of the dialogue. A tactic is succeeded when the sub-goal is achieved, and the strategy is succeeded when all the involved tactics are succeeded, which means that the final goal is achieved. Fig. 1 illustrates the strategic and tactic levels in our framework.

Indeed, in multi-agent systems, agents are designed to accomplish particular tasks. Each agent has its own domain and a certain goals to achieve. We call this kind of goals: *operational goals*. These agents often have to interact with each other in order to achieve some sub-goals of the operational goals. These sub-goals generate what we call *conversational goals*. In our framework, we distinguish between these two types of goals. In the same way, we distinguish between domain constraints, called *operational constraints*, and conversational constraints called *criterions*. Time and budget constraints are examples of operational constraints, and respecting the religious and ideological believes of the addressee is an example of criterions. In our framework, a dialogue strategy depends on the conversational goal, operational constraints and criterions. Operational constraints and criterions also reflect the factors that may influence the strategy design: goals, domain, agents' capabilities, agents' values, protocol, counterparts, agents' resources, and alternatives [25]. Domain, agents' capabilities, and agents' values are operational constraints. Protocol, counterparts, agents' resources, and alternatives are criterions.

The initiative agent must build a global and initial strategy before starting the conversation. *A strategy allows an agent to decide about the main sub-goals*

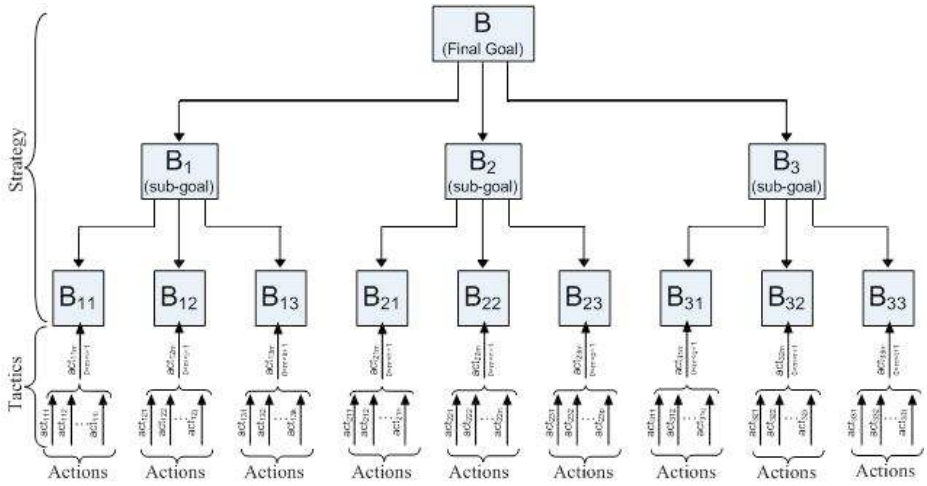


Fig. 1. Strategy and tactics in our framework

to be fixed in order to achieve the conversational goal according to a set of operational constraints and conversational criterions. To achieve the same conversational goal, an agent can have several alternative strategies depending on the sub-set of operational constraints and the sub-set of criterions the agent decide to satisfy. The conversational goal, sub-goals, operational constraints and criterions can be expressed in a logical language. The set of operational constraints and the set of criterions can be inconsistent. However, the sub-set of operational constraints and the sub-set of criterions the agent decide to satisfy should be consistent. We define a strategy as a function that associates to a goal and a sub-set of operational constraints and a sub-set of criterions a set of goals (sub-goals).

Definition 5 (Strategy). *Let $\mathcal{B}$ be a set of goals, Ctr be a set of operational constraints, and Cr be a set of conversational criterions. A strategy is a function:* $Str : \mathcal{B} \times 2^{Ctr} \times 2^{Cr} \rightarrow 2^{\mathcal{B}}$

Strategies are dynamic in nature. Agents should *adjust* the adopted dialogue strategy while the conversation progresses. This can be achieved by taking into account the new constraints and criterions that can appear during the conversation. In this case, the new constraints and criterions to be satisfied should be consistent with the initial sub-set of constraints and criterions selected to be satisfied. Thus, agents can apply the strategy function (Str) each time new constraints and criterions are added. This enables agents to decide about the sub-goals to be achieved of each already fixed sub-goal. In Fig. 1, this is illustrated by the different levels: from a level i to a level $i + 1$ (we suppose that the level in which we have the main or final goal is the lower one). We notice here that the set of criterions can progress with the dialogue, whereas the set of operational constraints is generally more stable.

Example 2. Let us suppose that: $Ctr = \{x_0, x_1, x_2\}$ and $Cr = \{y_0, y_1\}$. Let $B \in \mathcal{B}$ be the conversational goal, and $SCtr$ and SCr be two sub-sets of Ctr and Cr representing the constraints and criterions selected to be satisfied. We suppose that: $SCtr = \{x_0, x_1\}$ and $SCr = \{y_1\}$. We can have at a first time (level 0): $Str(B, SCtr, SCr) = \{B_1, B_2, B_3\}$. At a second time (level 1), we suppose that: $SCr = SCr \cup \{y_2\}$. Thus, by applying the Str function on B_1, we can obtain: $Str(B_1, SCtr, SCr) = \{B_{11}, B_{12}, B_{13}\}$.

This example illustrates how the strategy can influence the dialogue by deciding about the sub-goals to achieve in order to achieve the main conversational goal. The dialogue advance, on the other hand, influences the strategy by taking into account the new operational constraints and criterions. In the case where the new constraints and criterions are inconsistent with the initial selected ones, the adopted strategy should be *completely* or *partially changed*. The strategy should be completely changed if the main goal is changed. However, if only one of the sub-goals is changed, the strategy should be partially changed.

In our framework, agents start by using the strategic reasoning to build the general line of communication. This is reflected by applying the function Str on the main conversational goal. Thereafter, strategic reasoning and tactic reasoning are used in parallel. The link between strategy and tactics is that each tactic is related to a sub-goal fixed by the strategy. The execution of a tactic allows the execution, the evolution, and the adaptation of the strategy. For example, if the tactic does not allow the achievement of a sub-goal, the strategy should be adapted to fix another sub-goal.

4 Tactic Reasoning

In this section, we present our theory of the tactical reasoning for argumentation-based communicative agents. As illustrated in Fig. 1, tactics allow agents to select from a set of actions, one action in order to achieve a sub-goal fixed by the adopted strategy. The purpose of our theory is to guarantee that the selected action is the most appropriate one according to the current context. In the rest of this paper, the actions we consider are arguments that agents use to support their points of view or attack the opponent's point of view. The most appropriate action is then the most relevant argument. This enables agents to be more efficient in their argumentation. Our theory is based on the relevance of arguments.

4.1 Relevance of Arguments

The most significant attempts to formalize relevance have been done by van Rooy [32] and Fleger [12]. van Rooy supposes that the relevance of a communication act in purely competitive dialogues depends on its argumentative force in a given context. The argumentative force of a proposition with respect to a hypothesis is defined by a probability function, which assigns a value to a proposition. This

value represents the probability that this proposition is true. However, van Rooy does not specify how we can assign probabilities to different propositions. Fleger's proposal is based on the proof theory of minimality. It considers that an argument is irrelevant if it is not in relation to the conversation subject (or problem to be solved) or if it contains useless premises. This notion of relevance takes into account only the agent's knowledge base without considering the context of conversation. In addition, the minimality concept is not related to the notion of relevance, but it is a part of arguments definition.

In our framework, we define the relevance of an argument according to the context of conversation. Our objective is to allow agents to select the most relevant argument at a given moment by taking into account not only the last communicative act, but also the previous acts. The idea is to provide a solution allowing *backtracking*. This means that, an agent selects one among a set of possible arguments represented as a tree. If the choice proves to be incorrect because the selected argument is not accepted by the addressee agent and cannot be defended, the agent can backtrack or restart at the last point of choice and can try another argument, which is represented by trying another path in the tree. The arguments are ordered according to their relevance. We call this process *arguments selection mechanism*.

4.2 Arguments Selection Mechanism

Let L be a logical language. The context of conversation for an agent Ag_1 committed in a conversation with another agent Ag_2 is defined as follows.

Definition 6 (Context). *The context of conversation for an agent Ag_1 (the speaker) committed in a conversation with an agent Ag_2 (the addressee) is a 5-tuple $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD \rangle$ where:*

- *S is a formula of L representing the conversation subject that corresponds to the conversational goal,*
- *s is a formula of L representing the argument on which the speaker should act,*
- *$\mathcal{P}_{Ag_1,Ag_2}$ is the set of Ag_1's beliefs about Ag_2's beliefs $\mathcal{P}^{bel}_{Ag_1,Ag_2}$ and about Ag_2's preferences $\mathcal{P}^{pref}_{Ag_1,Ag_2}$. Thus $\mathcal{P}_{Ag_1,Ag_2} = \mathcal{P}^{bel}_{Ag_1,Ag_2} \cup \mathcal{P}^{pref}_{Ag_1,Ag_2}$,*
- *KD is the knowledge that the two agents share about the conversation.*

KD can contain results or laws related to the domain that are already proved. In addition, all information on which the two agents agree during the current conversation is added to KD. For example, the accepted arguments are added to KD. We also assume that $KD \cap \mathcal{P}_{Ag_1,Ag_2} = \varnothing$.

In the context C_{Ag_1,Ag_2}, formula s should be relevant for subject S in the sense that there is a logical relation between the two formulas. This relation represents the link between tactic and strategy. The idea is that the current action (at the tactic level) is related to a sub-goal, which is fixed by the strategy. The current argument can attack or support the formula representing the sub-goal. In order to define this logical relation between S and s, we introduce the notion of *argumentation tree* and the notion of *path* that we define as follows.

Definition 7 (Argumentation Tree). *Let A be the set of participating agents and AR be the set of arguments used by the agents in the dialogue. An argumentation tree T is a 2-tuple $T = \langle N, \rightarrow \rangle$ where:*

- $N = \{(Ag_i, (H, h)) | Ag_i \in A,\ (H, h) \in AR\}$ *is the set of nodes. Each node is described as a pair $(Ag_i, (H, h))$, which indicates that the argument (H, h) is used by the agent Ag_i,*
- $\rightarrow \subseteq N \times N$ *is a relation between nodes. We write $n_0 \rightarrow n_1$ instead of $(n_0, n_1) \in \rightarrow$ where $\{n_0, n_1\} \subseteq N$. The relation $\rightarrow$ is defined as follows: $(Ag_1, (H, h)) \rightarrow (Ag_2, (H', h'))$ iff $Ag_1 \neq Ag_2$ and (H', h') attacks (H, h) (see definition 3).*

This notion of argumentation tree is close to the notion of *argument tree* introduced in [8] and to the notion of *abstract dispute tree* used in [11]. The main difference between our argumentation tree notion and these two notions is that the first one is used to formalize the logical relation between the conversation subject S and the current argument s and not to illustrate the dialectical proof and the acceptance of arguments. In addition, our argumentation tree is used to illustrate the backtracking process which is not dealt with in [8] and in [11].

We associate each (argumentative) conversation to an argumentation tree. The root of such an argumentation tree is the initial node $n_0 = (Ag_i, (H, S))$ where Ag_i is the initiating agent ($Ag_i \in A$) and (H, S) is the argument supporting the conversation subject (or the conversation goal).

Definition 8 (Path). *Let $T = \langle N, \rightarrow \rangle$ be an argumentation tree. A path in T is a finite sequence of nodes $n_0, n_1, \ldots, n_m$ such that $\forall i\ 0 \leq i < m : n_i \rightarrow n_{i+1}$.*

Proposition 1. *Let $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD \rangle$ be a context of conversation and $A = \{Ag_1, Ag_2\}$ be the set of participating agents. There is a logical relation between S and s in the context C_{Ag_1,Ag_2} iff there is a path in the argumentation tree associated with the conversation between the root and the current node $n_m = (Ag_i, (H', s))$ where $i \in \{1, 2\}$ and (H', s) is the argument supporting s.*

The existence of a path in the tree between the root and the current argument means that this argument defends or attacks directly or indirectly the conversation subject. Thus, independently on the path, there is a logical relation between S and s.

In our approach, we first distinguish between *relevant* and *irrelevant* arguments in a given context. This distinction allows agents to eliminate at each argumentation step irrelevant arguments before ordering the relevant arguments in order to select the most relevant one.

Definition 9 (Irrelevant Argument). *Let $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD \rangle$ be a context of conversation, A be the set of participating agents, $T = \langle N, \rightarrow \rangle$ be the argumentation tree associated to the conversation, and $(Ag_i, (H, h))$ be a node in T where $i \in \{1, 2\}$. (H, h) is irrelevant in the context C_{Ag_1,Ag_2} iff:*

1. *There is no path between the node* $(Ag_i, (H, h))$ *and the root of* T *or;*
2. $\exists x : H \vdash x \wedge \neg x \in KD$.

The first clause states that the argument does not address the conversation subject. The second clause states that the argument contradicts the shared knowledge. We notice here that KD is a knowledge base that changes during the conversation. Thus, an argument built at a step t_i can become irrelevant at a later step t_j if it contradicts the new information accepted by the agent. In these two cases, the argument is irrelevant and the agent can not use it. Irrelevant arguments must be removed from the set of arguments that the agent can use at a given step of the conversation. This set, called the set of *potential arguments*, is denoted by PA.

In Section 2, we emphasized the fact that agents can have *private* preferences about different knowledge (see definition 4). Therefore, they can have private preferences about arguments. This preference relation denoted by $(H, h) \ll_{pref}^{Ag_i} (H', h')$ means that agent Ag_i prefers the argument (H', h') to the argument (H, h). We define this relation as follows.

Definition 10 (Preference). *Let* (H, h) *and* (H', h') *be two arguments.* $(H, h) \ll_{pref}^{Ag_i} (H', h')$ *iff* $level(H') \leq level(H)$.

Because $\leq$ is an ordering relation, the preference relation $\ll_{pref}^{Ag_i}$ is reflexive, antisymmetric, and transitive. Agents may also have *favorites* among their arguments. How an agent favors an argument over others depends on the dialogue type. For example, in a persuasive dialogue, an agent can favor arguments having more chances to be accepted by the addressee. In order to characterize this notion, we introduce the notion of *weight of an argument*. The weight of an argument (H, h) compared to another argument (H', h') in the context $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD \rangle$ is denoted by $W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')}$ and is evaluated according to the following algorithm:

Algorithm 1 (Evaluation of an Argument compared to Another One).

Step 1: $W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')} = 0$.
Step 2: $(\forall x \in H), (\forall x' \in H') :$
$$(pref(x, x') \in \mathcal{P}^{pref}_{Ag_1,Ag_2}) \Rightarrow W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')} = W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')} + 1.$$

$pref(x, x') \in \mathcal{P}^{pref}_{Ag_1,Ag_2}$ *means that* Ag_1 *believes that* Ag_2 *prefers* x *to* x'.

According to this algorithm, the weight of an argument (H, h) compared to another argument (H', h') is incremented by 1 each time Ag_1 believes that Ag_2 prefers a knowledge in H to a knowledge in H'. Indeed, each element of H is compered once to each element of H'according to the preference relation. Consequently, the weight of an argument is finite because H and H' are finite sets.

The *favorite relation* is denoted by $\preceq_{fav}^{\mathcal{P}_{Ag_1,Ag_2}}$ and the *strict favorite relation* is denoted by $\prec_{fav}^{\mathcal{P}_{Ag_1,Ag_2}}$. $(H,h) \preceq_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H',h')$ means that agent Ag_1 favors the argument (H',h') over the argument (H,h) according to $\mathcal{P}_{Ag_1,Ag_2}$. This relation is defined as follows.

Definition 11 (Favorite Argument). *Let $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD\rangle$ be a context of conversation and (H,h) and (H',h') be two arguments in the context C_{Ag_1,Ag_2}. We have :*

$(H,h) \preceq_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H',h')$ *iff* $W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')} \leq W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H',h')/(H,h)}$,

$(H,h) \prec_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H',h')$ *iff* $W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H,h)/(H',h')} < W^{\mathcal{P}_{Ag_1,Ag_2}}_{(H',h')/(H,h)}$.

In order to allow agents to select the most relevant argument in a conversation context, we introduce an ordering relation between relevant arguments. This ordering relation depends on the adopted strategy and is based on the notion of the *risk of failure* of an argument. This notion of risk is subjective and there are several heuristics to evaluate the risk of an argument. In this paper we use a heuristic based on the fact that KD contains certain knowledge and $\mathcal{P}_{Ag_1,Ag_2}$ contains uncertain beliefs. We formally define this notion as follows.

Definition 12 (Risk of Failure of an Argument). *Let $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD\rangle$ be a context of conversation and (H,h) be a relevant argument in the context C_{Ag_1,Ag_2}. The risk of failure of (H,h) denoted by $risk((H,h))$ is the sum of the risks of failure of all the formulas included in H. The risk of failure of a formula q denoted by $risk(q)$ is defined as follows:*

- *if $q \in KD$ then $risk(q) = v_1$.*
- *if $q \in \mathcal{P}_{Ag_1,Ag_2}$ then $risk(q) = v_2$.*
- *otherwise $risk(q) = v_3$.*

Where $v_1 < v_2 < v_3$ and $v_1, v_2, v_3 \in \mathbb{R}$.

Values v_1, v_2 and v_3 should be instantiated according to the dialogue type and the confidence level of the beliefs included in $\mathcal{P}_{Ag_1,Ag_2}$. For example, in a persuasive dialogue and if we consider that KD contains certain knowledge, we may have $v_1 = 0$, $v_2 = 0.25$, $v_3 = 0.5$. If the confidence level of $\mathcal{P}_{Ag_1,Ag_2}$ is weak, it is possible to increase v_2. However, if this confidence level is high, it is possible to decrease v_2. In a persuasive dialogue, the idea behind the risk of failure is to promote arguments whose hypotheses have more chance to be accepted. Other approaches like those used in fuzzy systems to reason with uncertainty (using for example probabilities) can also be used to evaluate the risk of an argument. The advantage of our approach is that it is easy to implement and it reflects the intuitive idea that adding uncertain hypotheses increases the risk of failure of an argument.

The relevance ordering relation denoted by $\preceq_r$ can be defined as follows.

Definition 13 (Relevance Ordering Relation). *Let* $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD\rangle$ *be a conversation context and* (H,h) *and* (H',h') *be two relevant arguments in the context* C_{Ag_1,Ag_2}. (H',h') *is more relevant than* (H,h) *denoted by* $(H,h) \preceq_r (H',h')$ *iff:*

- $risk((H',h')) < risk((H',h'))$ *or*
- $risk((H',h')) = risk((H',h'))$ *and* $(H,h) \prec_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H',h')$ *or*
- $risk((H',h')) = risk((H',h'))$ *and* $(H,h) \preceq_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H',h')$ *and* $(H',h') \preceq_{fav}^{\mathcal{P}_{Ag_1,Ag_2}} (H,h)$ *and* $(H,h) \ll_{pref}^{Ag_1} (H',h')$.

According to this definition, (H',h') is more relevant than (H,h) if the risk of (H,h) is greater that the risk of (H',h'). If the two arguments have the same risk, the more relevant argument is the more favourable one according to the favourite relation $\prec_{fav}^{\mathcal{P}_{Ag_1,Ag_2}}$. If the two arguments have the same risk and they are equal according to the favourite relation, the more relevant argument is the more preferable one according to the preference relation $\ll_{pref}^{Ag_i}$ where $i \in \{1,2\}$. The two arguments have the same relevance if in addition they are equal according to the preference relation. The ordering relation $\preceq_r$ is reflexive, antisymmetric, and transitive. The proof is straightforward from the definition and from the fact that $\ll_{pref}^{Ag_i}$ is an ordering relation (see Definition 10).

Computationally speaking, the arguments selection mechanism is based on: (1) the elimination of irrelevant arguments; (2) the construction of new relevant arguments; (3) the ordering of the relevant arguments using the relevance ordering relation; and (4) the selection of one of the most relevant arguments. This process is executed by each participating agent at each argumentation step at the tactical level. The relevant arguments that are not selected at a step t_i, are recorded and added to the set of potential arguments PA because they can be used at a subsequent step. The set of potential arguments can be viewed as a stack in which the higher level argument is the most relevant one. A relevant argument constructed at a step t_i and used latter at a step t_j simulates the backtracking towards a previous node in the argumentation tree and the construction of a new path. The following example illustrates this idea.

5 Example

In this example, we present only a part of the argumentation tree, which is sufficient to illustrate the arguments selection mechanism. To simplify the notation, arguments are denoted by a_i and a'_i $(1 \leq i \leq n)$. We assume that the conversation subject is S, $A = \{Ag_1, Ag_2\}$, $KD = \{f, l, q\}$, and $\mathcal{P}_{Ag_2,Ag_1} = \{p, d, r\} \cup \{pref(q,p)\}$ where f, l, q, p, d and r are formulas of the language L. The part of the argumentation tree we are interested in starts from a node $n_i = (Ag_1, a_1)$ where $a_1 = (\{s, \neg s', s \wedge \neg s' \rightarrow u\}, u)$ and s, s', u are formulas of the language L. We also assume that from its knowledge base, agent Ag_2 produces four arguments taking into account the current context $C_{Ag_1,Ag_2} = \langle S, s, \mathcal{P}_{Ag_1,Ag_2}, KD\rangle$. These arguments are:

$a'_1 = (\{p, k, p \wedge k \rightarrow \neg s\}, \neg s)$, $a'_2 = (\{q, r, c, q \wedge r \wedge c \rightarrow \neg s\}, \neg s)$,
$a'_3 = (\{\neg d, m, \neg d \wedge m \rightarrow s'\}, s')$, and $a'_4 = (\{e, c, e \wedge c \rightarrow s'\}, s')$.

Where p, k, q, r, c, d, m and e are formulas of the language L. Hence: $PA(Ag_2) = \{a'_1, a'_2, a'_3, a'_4\}$ ($PA(Ag_2)$ is the set of Ag_2's potential arguments).

At this step (**step 1**), Ag_2 should select the most relevant argument using our relevance ordering relation. In order to do that, Ag_2 should evaluate the risk of failure of these arguments. We assume that $v_1 = 0$, $v_2 = 0.3$, $v_3 = 0.5$. Consequently: $risk(a'_1) = 0.3 + 0.5 = 0.8$, $risk(a'_2) = 0 + 0.3 + 0.5 = 0.8$, $risk(a'_3) = 0.7 + 0.5 = 1.2$, $risk(a'_4) = 0.5 + 0.5 = 1$.

The arguments a'_1 and a'_2 have the same risk of failure. However, because $pref(q, p) \in \mathcal{P}_{Ag_2,Ag_1}$ and according to our evaluation algorithm (algorithm 1), we obtain: $W^{\mathcal{P}_{Ag_2,Ag_1}}_{a'_1/a'_2} = 0$ and $W^{\mathcal{P}_{Ag_2,Ag_1}}_{a'_2/a'_1} = 1$.

Therefore, according to definitions 11 and 13, the four arguments are ordered as follows: $a'_1 \preceq_r a'_2 \preceq_r a'_3 \preceq_r a'_4$. Consequently, Ag_2 selects a'_2. Then (**step 2**), Ag_1 should take position on a'_2. For that we assume that Ag_1 has only one argument $a_2 = (\{f, l, f \wedge l \rightarrow \neg c\}, \neg c)$ attacking a'_2 in the new context $C_{Ag_1,Ag_2} = \langle S, \neg s, \mathcal{P}_{Ag_1,Ag_2}, KD\rangle$. Because $f, l \in KD$, Ag_1 accepts this argument. Thereafter, $\neg c$ is added to KD and according to definition 9, a'_4 becomes irrelevant. This argument is removed from the set of Ag_1's potential arguments. We then obtain $PA(Ag_2) = \{a'_1, a'_3\}$. According to the arguments selection mechanism, Ag_2 selects a'_1 (**step 3**). Selecting this argument at this step simulates a backtracking towards a lower level node (previous node) in the argumentation tree. This example is illustrated in Fig. 2.

6 Related Work and Conclusion

Recently, some interesting proposals have addressed the strategic reasoning of argumentative agents. In [25], Rahwan et al. propose a set of factors that may influence the strategy design. These factors are considered in our framework as operational constraints and criterions. In [2], Amgoud and Maudet define the strategy as a function allowing agents to select a communicative act from the permitted acts. This definition does not take into account the underlying factors and the operational selection mechanism. The more complete framework in the literature addressing tactic and strategic issues of agent communication was developed by Kakas et al. [15]. The authors propose an argumentation-based framework encompassing private tactics of the individual agents and strategies that reflect different classes of agent attitudes. This framework uses sceptical and credulous forms of argumentative reasoning. Private strategies specify the dialogue moves an agent is willing to utter, according to its own objectives and other personal characteristics. Unlike our proposal, this work does not specify the relation between strategy and tactic. In addition, strategies and tactics are mainly represented using a preference policy on the dialogue moves. However,

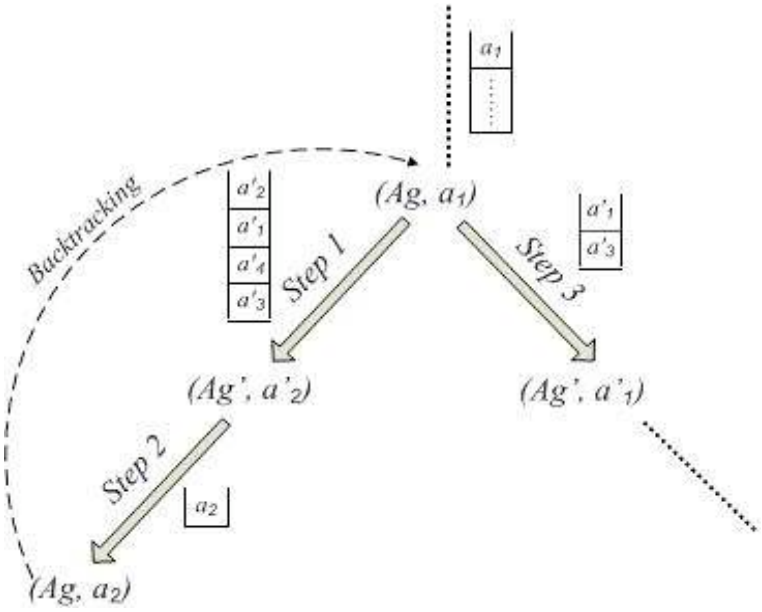


Fig. 2. A part of argumentation tree with the arguments selection mechanism

our strategy and tactic theory is based on the goals and sub-goals agents want to achieve. The context notion we use in our framework that reflects the conversational goal and the different agents' beliefs is different from the one used by the authors, which is generally defined on the basis of some priority rules.

The different proposals that have considered the strategic level, have neglected the important relation between strategy and tactics. The contribution of this paper is the proposition of an approach allowing agents to combine strategic and tactic reasoning in order to be more efficient in their communications. The link between strategic and tactic levels enables agents to have global and local visions of the dialogue. In addition, our tactic theory provides a strong mechanism to select the most appropriate argument depending on the strategy adopted by the agent. The mechanism uses our relevance principle that takes into account the context of conversation. This selection mechanism is implemented in the case of persuasion dialogues using logical programming and an agent-oriented platform (Jack Intelligent Agents). In addition, an important advantage of our approach is the fact that it allows backtracking.

The approach presented in this paper is general and can be implemented for other dialogue types. As future work, we plan to define in a systematic way the relevance ordering for each dialogue type. In addition, we intend to enhance protocols based on dialogue games with our strategic and tactic approach. This will allows us to develop more flexible and efficient argument-based agent conversations. We also intend to analyze and evaluate the behavior of the proposed

heuristics (e.g. the notion of risk of failure). On the other hand, our framework is operational in its design. Thus, if it is different from the one developed by Sadri et al. [28], which is more declarative. Considering the declarative meaning and investigating the formal properties of our argumentation setting is another key issue for future work.

Acknowledgements. We would like to thank the three anonymous reviewers for their interesting and helpful comments and suggestions. We also kindly thank the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Fonds Québécois de la Recherche sur la Nature et les Technologies (FQRNT) for their financial support.

References

1. Amgoud, L., Maudet, N., Parsons, S.: Modelling dialogues using argumentation. Proc. of the 4th Int. Conf. on Multi-Agent Systems. IEEE Press (2000) 31-38
2. Amgoud, L., Maudet, N.: Strategical considerations for argumentative agents (preliminary report). Proc. of the 9th Int. Workshop on Non-Monotonic Reasoning (2002) 409-417
3. Atkinson, K., Bench-Capon, T., McBurney, P.: A dialogue game protocol for multi-agent argument over proposals for action. Journal of AAMAS. Special issue on Argumentation in Multi-Agent Systems 11(2) (2005) 153–171.
4. Bentahar, J.: A pragmatic and semantic unified framework for agent communication. PhD Thesis, Laval University, Canada May (2005)
5. Bentahar, J., Moulin, B., Chaib-draa, B.: Commitment and argument network: a new formalism for agent communication. Advances in Agent Communication. LNAI 2922 (2004) 146–165
6. Bentahar, J., Moulin, B., Meyer, J-J.Ch., Chaib-draa, B.: A logical model for commitment and argument network for agent communication. Proc. of the 3rd Int. Joint Conf. on AAMAS (2004) 792–799
7. Bentahar, J., Moulin, B., Meyer, J-J.Ch., Lespérance, Y.: A new logical semantics for agent communication. Proc. of the 7th Int. Workshop on Computational Logic in Multi-Agent Systems (2006) Accepted
8. Besnard, P., Hunter, A.: A logic-based theory of deductive arguments. Artificial Intelligence 128 (2001) 203– 235
9. Chesñevar, C.I., Maguitman, A., Loui, R.: Logical models of argument. ACM Computing Surveys 32 (2000) 337–383
10. Dignum, V.: A model for organizational interaction: based on agents, founded in logic. PhD Thesis, Utrecht University, The Netherlands (2004)
11. Dung, P.M., Kowalski, R.A., Toni, F.: Dialectic proof procedures for assumption-based, admissible argumentation. Artificial Intelligence 170(2) (2006) 114–159
12. Flieger, J.C. Relevance and minimality in systems of defeasible argumentation. Internal Report. Imperial College of Science, Technology and Medecin (2002)
13. Goldratt, E.: Theory of Constraints. North River Press (1999)
14. Kakas, A., Moraitis, P.: Argumentation based decision making for autonomous agents. Proc. of the 2nd Int. Joint Conf. on AAMAS (2003) 883–890
15. Kakas, A., Maudet, N., Moraitis, P.: Layered strategies and protocols for argumentation-based agent interaction. Argumentation in Multi-Agent Systems. LNAI 3366 (2005) 64–77

16. Maudet, N., Chaib-draa, B.: Commitment-based and dialogue-game based protocols: new trends in agent communication languages. The Knowledge Engineering Review 17(2) (2002) 157-179
17. Moulin, B., Irandoust, I., Bélanger, M., Desbordes, G.: Explanation and argumentation capabilities: towards the creation of more persuasive agents. Artificial Intelligence Revue 17(3) (2002) 169–222
18. McBurney, P., Parsons, S., Wooldridge, M.: Desiderata for agent argumentation protocols. Proc. of the 1st Int. Joint Conf. on AAMAS (2002) 402–409
19. McBurney, P., van Eijk, R.M., Parsons, S., Amgoud. L.: A dialogue game protocol for agent purchase negotiations. Journal of AAMAS 7(3) (2003) 235–273
20. Moore, D.: Dialogue game theory for intelligent tutoring systems. PhD Thesis, Leeds Mitropolitan University, England (1993)
21. Parsons, S., Wooldridge, M., Amgoud, L.: On the outcomes of formal inter-agent dialogues. Proc. of the 2nd Int. Joint Conf. on AAMAS (2003) 616–623
22. Parsons, S., McBurney, P., Wooldridge, M.: Some preliminary steps towards a meta-theory for formal inter-agent dialogues. Argumentation in Multi-Agent Systems. LNAI 3366 (2005) 1–18
23. Prakken, H., Vreeswijk, G.: Logics for defeasible argumentation. Handbook of Philosophical Logic (Second Edition) (2000)
24. Rahwan, I., Ramchurn,S.D., Jennings, N.R., McBurney, P., Parsons, S., Sonenberg, L.: Argumentation-based negotiation. The Knowledge Engineering Review 18(4) 2003 343–375
25. Rahwan, I., McBurney, P., Sonenberg, L.: Towards a theory of negotiation strategy (a preliminary report). Proc. of the Workshop on Game Theoretic and Decision Theoretic Agents (2003)
26. Reed, C., Walton, D.: Towards a Formal and Implemented Model of argumentation schemes in Agent Communication. Argumentation in Multi-Agent Systems. LNAI 3366 (2005) 19–30
27. Rovatsos, M., Rahwan, I., Fischer, F., Weiss, G.: Adaptive strategies for practical argument-based negotiation. Proc. of the 2nd Int. Workshopon on Argumentation in Multi-Agent Systems (2005)
28. Sadri, F., Toni, F., Torroni, P.: Dialogues for negotiation: agent varieties and dialogue sequences. 8th Int. Workshop on Agent Theories, Architectures and Language. LNCS 2333 (2001) 405–421
29. Searle, J.R. Speech acts: an essay in the philosophy of languages. Cambridge University Press, England (1969)
30. Singh, M.P.: A social semantics for agent communication languages. Issues in Agent Communication. LNAI 1916 (2000) 31–45
31. van Dijk, T.A., Kintsch, W.: Strategies of Discourse Comprehension. New York Academic Press (1983)
32. van Rooy, R.: Relevance of communicative acts. Proc. of TARK VIII (2001) 83–96

A Framework for Learning Argumentation Strategies (Position Paper)

Chukwuemeka David Emele, Frank Guerin, Timothy J. Norman, and Pete Edwards

University of Aberdeen, Aberdeen, AB24 3UE, Scotland
demele,fguerin,tnorman,pedwards@csd.abdn.ac.uk

Abstract. Recent years have witnessed growing interest in the area of argumentation strategies but there is little work done in the area of learning how to argue. This paper aims to explore how agents can improve their argumentation strategies using machine learning. This includes both learning about other agent's strategies as well as learning general heuristics for argumentation strategies. This framework will allow agents to build flexible and adaptive strategies for arguing, based on previous experiences with other agents. We intend to apply this to agents in information seeking domains whereby agents seek to optimise their strategy by achieving their goals with minimal information revealed.

1 Interest

In a multi-agent system, comprising of agents representing different interests from different vendors and consumers, agents interact with other agents in order to complete a specified task. These interactions may result in conflicting views and disputes in which case argumentation can be employed as a mechanism for achieving cooperation and agreements. Argumentation is an iterative process emerging from exchanges among agents to persuade each other and bring about a change in intentions (Kraus et al. [7]).

Optimal strategies could be learned from a series of interactions with a view to winning future arguments, getting better results (e.g. getting the work done without revealing too much information), reducing the length of time and resources involved in future encounters, and so on. The framework proposed in this work will be applied to an information seeking domain where agents may have privacy policies and are reluctant to reveal information unless an acceptable justification is put forward.

Using this framework, an Agent could build a model of the argumentation strategy of other agents from a series of interactions and attempt to refine its own strategy as the argumentation progresses. In other words, the agent will be able to group other agents into categories based on behavioural characteristics (e.g. helpful, indifferent, etc) or organisational relationships (e.g. boss, subordinate, contemporary, etc) and seek to refine its strategy with a view to achieving

preferable outcomes (for instance, agent A could persuade another agent B to get some work done without revealing much information, which is a preferable outcome for A).

In our model, agents could adopt several strategies in the argumentation. An agent's strategy may be to continue to argue for as long as possible; another may adopt the strategy of always pushing the burden of proof onto the opponent, and so on.

The central idea of this work is captured in three stages:

1. During agent interactions, the behavioural characteristics and/or organisational relationship of the agents involved is/are evident, and a model of the privacy policy for those agents could be learnt. For instance, during the interactions between agent A and B, there may be some information that A requires from B in order to complete a task. Agent B may have a (conflicting) policy that does not give out such information unless an acceptable justification is put forward. In the course of the exchange (Kraus et al. [7]), agent A will have a better perception of agent B's policy.
2. Over time, agent A builds a model of the argumentation strategy of agent B based on previous encounters with B and can more easily predict what arguments are likely to get B to perform the task (in this case, to give up the information that agent A requires).
3. After agent A has built models of some agents (say, B, C, D) then agent A begins to generalise (and possibly categorise) other agents with respect to some peculiar (or related) characteristics (e.g. behavioural characteristics or organisational relationships). This means agent A begins to get better in the way it argues by learning over encounters with other agents and using this acquired intelligence to relate with similar agents. Based on the outcomes of these generalisations, agent A refines and adapts its argumentation strategy to enable it to argue better.

2 Questions

1. How do we represent an agent's strategy?
2. At what point should agent A change its argumentation strategy?
3. What is the yardstick for measuring the effectiveness of that strategy?
4. How should agent A select a strategy from a pool of strategies with a view to optimising the result?
5. What is the rationale for generalising?
6. How do we evaluate the outcomes of the generalisation?

3 Discussion

Dung [2] studied the fundamental mechanism humans use in argumentation, and explored ways to implement this mechanism on computers. The notion of acceptability of arguments is central to our work. Also Dung [2] proposed abstract

argumentation frameworks on which many works on argumentation are based. The argumentation framework proposed in our work uses the fundamentals of Dung's abstract argumentation framework (see Dung[2]).

Sycara's work [6] led to the subsequent research by Kraus et al. [7] which identified five different types of arguments that can be used in argumentation. We note that our work will utilize these five argument types (threats, promise of future rewards, appeals to precedent, appeals to prevailing practice, and appeals to self-interest) in determining the behavioural characteristics of agents in the domain (see Kraus et al. [7]).

The work done by Leila Amgoud and Nicolas Maudet [1] on the exploration of strategies for move selection in persuasive dialogue conducted by argumentative agents is very relevant to our work. The work presented a three-layered approach to strategy, and heuristics that are based on some human strategies issued from natural dialogues were proposed. Our work will leverage the three-layered approach to strategy and will build a framework that allows agents to learn and refine their argumentation strategies.

Prakken [5] holds that interactions between autonomous agents are not proof theory, therefore, the outcome of a dialogue is expectedly non-deterministic (dependent on the Commitment Store and the profile of the agent Amgoud and Parsons [4]), and so if an agent is not wise it might lose an argument which otherwise could have been won. We find this standpoint interesting, as it shows that there is much scope to explore different strategies and learning.

To the best of our knowledge, there is very little work done so far (except Rovatsos et al. [8]) that focuses on the intersection between learning and argumentation strategies. This work aims to explore this research gap.

References

1. Amgoud, L., Maudet, N.: Strategical considerations for argumentative agents (preliminary report). In Benferhat, S. & Giunchiglia, E. (eds.), Proceedings of the 9th International Workshop on Non-Monotonic Reasoning (NMR 2002): Special Session on Argument, Dialogue and Decision. (2002) 399-407
2. Dung, P. M.: On the acceptability of arguments and its fundamental role in nonmonotonic reason, logic programming, and n-person games. Artificial Intelligence. **77** (1995) 321–357
3. Moore, D.: Dialogue game theory for intelligent tutoring systems. PhD thesis, Leeds Metropolitan University, England. (1993)
4. Amgoud, L., Parsons, S.: Agent dialogue with conflicting preferences. In Proceedings of the International Workshop on Agent Theories, Architectures and Languages (ATAL01). (2001) 1–17
5. Prakken, H.: On dialogue systems with speech acts, arguments, and counterarguments. In Proceedings of the 7th European Workshop on Logic for Artificial Intelligence (JELIA), Lecture Notes in AI 1919. (2000) 239–253
6. Sycara, K.: Argumentation: Planning other agent's plans. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence. (1989) 517–523
7. Kraus, S., Nirkhe, M., Sycara, K.: Reaching agreements through argumentation: a logical model (preliminary report). In Proceedings of the Workshop on Distributed Artificial Intelligence. (1993)

8. Rovatsos, M., Rahwan, I., Fischer, F., Weiss, G.: Adaptive strategies for practical argument-based negotiation. Second International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2005), Utrecht, The Netherlands. (2005)