Constraints-based Negotiation using Argumentation

Mohamed Mbarki
mohamed.mbarki@ift.ulaval.ca
Laval University

Jamal Bentahar
bentahar@ciise.concordia.ca
Concordia University

Bernard Moulin
bernard.moulin@ift.ulaval.ca
Laval University

Ahmad Moazin
a_moazi@encs.concordia.ca
Concordia University

Abstract

In this paper, we propose a framework for constraints-based negotiation using argumentation. *Agents' constraints* such as budget and time constraints play a key role in determining the set of offers agents can make. An offer in our setting consists of assigning values to a set of *negotiation constraints* such as prices of goods and services and delivery time. To reach an agreement, agents' constraints must be satisfied. Each offer must be supported by arguments and each agent tries to achieve an agreement using arguments to persuade the other agent to make a *concession*, which is a fundamental notion in negotiation. We propose a negotiation policy specifying when a concession can be made and showing when an agreement is reached. We also propose an algorithm specifying this policy and discuss its formal properties and implementation.

1. Introduction

In multi-agent settings, it could be the case that autonomous agents have conflicts with one another. Negotiation is a process that aims to solve these conflicts by reaching an agreement on certain issues, taking into account a set of attributes or variables that reflect the capabilities and preferences of the negotiating agents. We consider that each variable is associated with a constraint about which a conflict exists. For example, when buying a car, the price and warranty duration are two negotiation variables. In this paper, we call *negotiation constraints* the constraints associated with the negotiation variables.

Different approaches to modeling agent negotiation have been proposed in the literature (for example, (Rahwan et al. 2004; 2009; Amgoud, Dimopoulos, and Moraitis 2007; Faratin, Sierra, and Jennings 2002; Ros and Sierra 2006; Hindriks et al. 2009)). Furthermore, some projects on negotiation have recently been launched to study the dynamics of negotiation in different contexts, such as man-machine interaction context[1]. However, these approaches do not take into account the factors that influence the evolution of the negotiation, and more particularly the agents' constraints.

This paper proposes a framework for constraints-based negotiation where offers and counter-offers generation is based on constraints and arguments. An agent's constraints play a key role in determining the set of offers this agent can make. In our setting, an offer consists of an assignment of values to the negotiation constraints. In addition, agents have reasoning capabilities, which enable them to decide about the appropriate moves to perform during the negotiation in order to achieve their goals.

In our model, each offer is supported by a finite set of arguments. We realistically assume that two arguments supporting the same offer might, but not necessarily, attack each other, and there is not necessarily an attack relation between two arguments supporting two different offers. Moreover, we assume that a given argument cannot support two distinct offers. We also consider that an agent's preferences over arguments are not necessarily identical to his preferences over offers, which are supported by these arguments. Indeed, an agent's best offer can be supported by an argument that is not the most preferable for this agent. However, we consider that the offer an agent can make at a given negotiation step is the one that is supported by the most relevant argument at this step. To compute the most relevant argument, we use the approach presented in (Mbarki, Bentahar, and Moulin 2007), according to which the argument's relevancy is proportional to its probability to be accepted by the addressee given the beliefs the agent has about this addressee.

In this paper, we focus on the description of negotiating agents' reasoning and negotiation dynamics. Section 2. introduces the negotiating agent's theory. Section 3. addresses constraints issues in negotiation. Section 4. presents the argumentation-based negotiation. Section 5. reports on some implementation details. Finally, a discussion and some directions for future work are outlined in Sections 6. and 7. respectively.

2. Theory of Negotiating Agents

In our framework, negotiations take place between two agents s (for example the seller) and b (for example the buyer). In this framework, the following elements are supposed to be described in a formal language $\mathcal{L}$:

- Knowledge bases, KB_s for s and KB_b for b.
- A finite set of constraints, $\mathcal{C}tr$ (for example, budget constraint, time constraint, etc.).
- A finite set of offers, $\mathcal{O}r$.

[1] see for example the Delft project on negotiation: http://mmi.tudelft.nl/negotiation

- The negotiation commitment store, CS. $CS = CS_b \cup CS_s$, where CS_b and CS_s are the commitment stores of agents b and s respectively. The content of CS is updated as usual after each negotiation turn.
- A set of arguments, $\mathcal{A}r \subseteq Arg(\mathcal{L})$, where elements may be conflicting and $Arg(\mathcal{L})$ is the set of all arguments built from $\mathcal{L}$. In what follows, these conflicts will be captured by an attack relation $\mathcal{R}$.

We extend the agent theory proposed in (Amgoud, Dimopoulos, and Moraitis 2007) by considering the agents' constraints. The theory of a negotiating agent i is then defined as follows:

Definition 1 (Negotiating Agent Theory) *The theory of a negotiating agent i is a tuple $\langle \mathcal{C}^i, \mathcal{A}^i, \mathcal{O}^i, \preceq_o^i, \mathcal{F}^i, \mathcal{R}\rangle$ such that:*

- *$\mathcal{C}^i \subseteq \mathcal{C}tr$ is the set of agent's constraints.*
- *$\mathcal{A}^i \subseteq \mathcal{A}r$ is the set of agent's arguments.*
- *$\mathcal{O}^i \subseteq \mathcal{O}r$ is the set of agent's offers.*
- *$\preceq_o^i \subseteq \mathcal{O}^i \times \mathcal{O}^i$ is a partial preorder denoting a preference between agent's offers.*
- *$\mathcal{F}^i : \mathcal{O}^i \rightarrow 2^{\mathcal{A}^i}$ is a function giving all the arguments that support an offer. $\forall k, l$ with $k \neq l$, we have: $\mathcal{F}^i(o_k^i) \cap \mathcal{F}^i(o_l^i) = \varnothing$.*
- *$\mathcal{R} \subseteq \mathcal{A}r \times \mathcal{A}r$ is a binary attack relation between arguments.*

Agent's reasoning model is based on his argumentation system from which the acceptable arguments are computed. In Definition 1, arguments and attack relation are defined in an abstract way. In what follows, we will give them a concrete definition, which we use in our framework and its implementation. Particularly, this will allow us to explain how an argument can support an offer. We use assumption-based argumentation (Dung, Kowalski, and Toni 2006; Toni 2007) in which arguments are built from a set of rules and assumptions using backward deduction. In our approach, an agent generates his arguments jointly from his knowledge base and the commitment store of the addressee.

Our formal language $\mathcal{L}$ consists of countably many sentences (or wffs). The language is associated with an abstract *mapping* for the *contrary* relation among sentences (Dung, Kowalski, and Toni 2006; Toni 2007) (the negation is an example of this mapping, so the contrary of p is $\neg p$). We do not assume this mapping to be necessarily symmetric. $!x$ denotes an arbitrary contrary of a wff x.

Definition 2 (Argumentation Framework) *An assumption-based argumentation framework is a tuple $\langle \mathcal{L}, \mathcal{I}r, \mathcal{A}s, \Delta \rangle$ where:*

- *$(\mathcal{L}, \mathcal{I}r)$ is a deductive system, with a countable set $\mathcal{I}r$ of inference rules,*
- *$\mathcal{A}s \subseteq \mathcal{L}$ is a (non-empty) set of assumptions,*
- *Δ is a total mapping from $\mathcal{L}$ into $2^{\mathcal{L}} - \emptyset$, where Δp is the non-empty set of contraries of p and $!p$ is an arbitrary contrary of p ($!p \in \Delta p$).*

We will assume that the inference rules in $\mathcal{R}$ have the form: $c_1, \ldots, c_n \rightarrow c_0$ with $n > 0$ or the form c_0 where each $c_i \in \mathcal{L}$ $(i = 0, \ldots, n)$.
c_0 is referred to as the head and $c_1, \ldots, c_n$ as the body of a rule $c_1, \ldots, c_n \rightarrow c_0$. The body of a rule c_0 is considered to be empty. We will restrict attention to *flat assumption-based frameworks*, such that if $c \in \mathcal{A}s$, then there exists no inference rule of the form $c_1, \ldots, c_n \rightarrow c \in \mathcal{I}r$. Before defining the notions of argument and attack relation, we give here a formal definition of the backward deduction that is used in this framework.

The backward deduction can be represented by a top-down proof tree linking the conclusion to the assumptions. The root of the tree is labelled by the conclusion and the terminal nodes are labelled by the assumptions. For every non-terminal node in the tree, there is an inference rule whose head matches the sentence labelling the node. The children of the node are labelled by the body of the inference rule. Consequently, the backward deduction can be represented as a set of steps $S_1, \ldots, S_m$ and in each step we have a set of sentences to which we can apply inference rules because each sentence matches the head of a rule. From each step to the next one, the procedure consists of selecting one sentence and replacing it, if the sentence is not an assumption, by the body of the corresponding inference rule. The selection strategy is represented by the following function:

$$SS : Step \rightarrow \mathcal{L}$$

where $Step = \{S_1, \ldots, S_m\}$.

Definition 3 (Backward Deduction) *Given a deduction system $(\mathcal{L}, \mathcal{I}r)$ and a selection strategy function SS, a backward deduction of a conclusion c from a set of assumptions X is a finite sequence of sets $S_1, \ldots, S_m$, where $S_1 = \{c\}$, $S_m = X$, and for every $1 \leq i < m$:*
***1.** If $SS(S_i) \notin X$ and $SS(S_i) \in S_i$ then $S_{i+1} = S_i - \{SS(S_i)\} \cup B$ for some inference rules of the form $B \rightarrow SS(S_i)$.*
***2.** Otherwise, $S_{i+1} = S_i$.*

Definition 4 (Argument) *Let $X \subseteq \mathcal{A}s$ be a consistent subset of assumptions (i.e. X does not include a formula and one of its contraries), and let c be a sentence in $\mathcal{L}$. An argument in favor of c is a pair (X, c) such that c is obtained by the backward deduction from X. c is called the conclusion of the argument.*

Definition 5 (Attack Relation) *Let $\mathcal{A}r \subseteq Arg(\mathcal{L})$ be a set of arguments over the argumentation framework. The attack relation between arguments $\mathcal{R} \subseteq \mathcal{A}r \times \mathcal{A}r$ is a binary relation over $\mathcal{A}r$ that is not necessarily symmetric. An argument (X, c) attacks another argument (X', c') denoted by $(X, c)\mathcal{R}(X', c')$ iff c is a contrary of c' or c is a contrary of a sentence $c'' \in X'$.*

According to (Dung 1995), an agent can have multiple sets of acceptable arguments, called *extensions*. Each extension must be *conflict-free*. In (Dung 1995), different acceptability semantics have been defined.

Definition 6 (Conflict-free) *Let $\mathcal{A}r \subseteq Arg(\mathcal{L})$ be a set of arguments over the argumentation framework. $S \subseteq \mathcal{A}r$ is conflict-free iff there is no $a, a' \in S$ such that $a\mathcal{R}a'$.*

Definition 7 (Defense) *Let $\mathcal{A}r \subseteq Arg(\mathcal{L})$ be a set of arguments over the argumentation framework, and let $S \subseteq \mathcal{A}r$. An argument a is defended by S iff $\forall\ b \in \mathcal{A}r$ if $b\mathcal{R}a$, then $\exists\ c \in S : c\mathcal{R}b$.*

Definition 8 (Acceptability Semantics) *Let S be a conflict-free set of arguments and $\mathcal{T} : 2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$ be a function such that $\mathcal{T}(S) = \{a|a$ is defended by $S\}$. S is a complete extension iff $S = \mathcal{T}(S)$. S is a preferred extension iff S is a maximal (w.r.t set $\subseteq$) complete extension. S is a grounded extension iff S is the smallest (w.r.t set $\subseteq$) complete extension.*

After defining the acceptability semantics, we can define the arguments that can be used by negotiating agents to support their offers. We call these arguments *potential arguments*. In fact, an argument a supports an offer o, iff this offer is the conclusion of a, i.e. $a = (X, o)$ for some given assumptions X.

Definition 9 (Argument Status) *Let $\mathcal{A}r \subseteq Arg(\mathcal{L})$ be a set of arguments over the argumentation framework, and $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ its extensions under a given semantics. Let $a \in \mathcal{A}r$, we have: (i) a is accepted iff $a \in \varepsilon_j$, $\forall\varepsilon_j$ with $j = 1, \ldots, m$; (ii) a is rejected iff $\nexists\varepsilon_j$ such that $a \in \varepsilon_j$; and (iii) a is undecided iff a is neither accepted nor rejected.*

Definition 10 (Potential Arguments) *Let $\mathcal{A}r \subseteq Arg(\mathcal{L})$ be a set of arguments over the argumentation framework, and $a \in \mathcal{A}r$. The set of potential arguments under a given semantics is $\mathcal{PA} = \{a|a$ is accepted$\} \cup \{a|a$ is undecided$\}$.*

In our model, each negotiating agent uses his potential arguments to generate and support his offers. The generation of the agent's offers is supported mainly by his negotiation constraints and potential arguments. An agent does not need to present the strongest arguments according to his beliefs, but he is supposed to use the most relevant arguments, the ones that, according to the agent's beliefs, have a better chance of being accepted by the addressee.

3. Constraints Consideration

The notion of constraints allows us to abstract various criteria that influence the negotiation process. In this paper, we model a negotiation by considering independent constraints that negotiating agents try to satisfy. Considering dependencies between constraints is in our plan for future work. We consider that the variables associated with these constraints represent conflicts between the two negotiating agents. For simplification and illustration reasons, but without loss of generality, we consider in the rest of the paper only one constraint, which is represented by a given variable.

In a given negotiation, each negotiating agent tries to maximize (respectively minimize) the value of the negotiation constraint (i.e. the value of the variable associated with this constraint), whereas the other agent tries to minimize (respectively maximize) this value. The price of a service is an example of a typical constraint in a negotiation of services. The customer wants the lowest price for the service; while the service provider tries to get the highest possible price.

Negotiation can range over a quantitative variable (e.g. the service price or the delivery date) or qualitative variable (e.g. quality of service). In (Zadeh 1965), it has been argued that qualitative variables can be defined over continuous domains as in quantitative models. As in (Faratin, Sierra, and Jennings 1998), we define the quantitative variable associated to a given agent's constraint over a real domain. Such a domain designates all the values of this constraint that are acceptable by the agent.

The negotiation constraint c for an agent i can take values over the range $[min_c^i, max_c^i]$ where $i \in \{b, s\}$. This range represents what we call the *constraint agreement space* of agent i, denoted by $\mathcal{EA}_c^i$. The two boundaries of the constraint agreement space are determined by the agent before starting the negotiation. One of these two boundaries represents the limit that the agent cannot overtake while making concessions, and the other represents what the agent considers as his best offer. The former is generally fixed by the agent's constraint, while the later is variable and determined at each negotiation step by the agent's argumentation process. For instance, for a negotiation of price, the value max_c^s represents the seller's preferred offer and the value min_c^s indicates the minimum price he can accept. During the negotiation, each negotiating agent recalculates, at each step, his set of potential arguments. Then, if the agent needs to make a concession, he determines his new preferred offer (i.e. the new max_c^s for the seller or the new min_c^b for the buyer). We will specify later on when an agent will be forced to make a concession.

For a given negotiation, the union of the agreement spaces of the agents s and b for a constraint c is the *negotiation space*, denoted by $\mathcal{EN}_c^{b\leftrightarrow s}$ and the intersection of these two agreement spaces is the agreement space for this constraint, denoted by $\mathcal{EA}_c^{b\leftrightarrow s}$. Formally we have:

$$\begin{aligned}
\mathcal{EN}_c^{b\leftrightarrow s} &= \bigcup_{i\in\{b,s\}} \mathcal{EA}_c^i \\
\mathcal{EA}_c^{b\leftrightarrow s} &= \bigcap_{i\in\{b,s\}} \mathcal{EA}_c^i
\end{aligned}$$

Example 1 *During the negotiation of a product price, the buyer agent b wants to get the lowest price, while the seller agent s tries to get the highest price. Let c_1 be the constraint associated with the price of the product. Let us assume that $\mathcal{EA}_{c_1}^b = [5, 10]$ and $\mathcal{EA}_{c_1}^s = [7, 12]$. We can define:*

$$\begin{aligned}
\mathcal{EN}_{c_1}^{b\leftrightarrow s} &= \mathcal{EA}_{c_1}^b \cup \mathcal{EA}_{c_1}^s \\
&= [5, 10] \cup [7, 12] \\
&= [5, 12] \\
\mathcal{EA}_{c_1}^{b\leftrightarrow s} &= \mathcal{EA}_{c_1}^b \cap \mathcal{EA}_{c_1}^s \\
&= [5, 10] \cap [7, 12] \\
&= [7, 10]
\end{aligned}$$

The negotiation process between the agents s and b consists of a sequence of offers and counter-offers. This process goes on until an offer or counter-offer is accepted by both agents, or one of them terminates the negotiation without achieving an agreement.

4. Argumentation-based Negotiation

Each agent uses arguments to attempt to change the other agent's attitudes toward his offers so that an agreement might be reached. In order to help the negotiating agents s and b to convince each other, we propose an argumentation-based negotiation process. The agents negotiate an object whose possible values belong to the negotiation space, $\mathcal{EN}_c^{b\leftrightarrow s}$.

Using their argumentation systems, the negotiating agents can determine the set of potential values for the negotiation constraint c, which belong to their agreement spaces $\mathcal{EA}_c^b$ and $\mathcal{EA}_c^s$. Each agent i can have then several offers at each negotiation step t called *potential offers* and denoted by $\mathcal{PO}_{c,t}^i$ ($\mathcal{PO}_{c,t}^i \subseteq \mathcal{O}^i$). Each potential offer is supported by a set of arguments. A potential argument at step t of the negotiation might turn into invalid at step $t+1$. Consequently, the set of the potential offers is subject to change at every negotiation step.

At each step t of a given negotiation, an agent i generates his potential arguments $\mathcal{PA}_t^i \subseteq \mathcal{A}^i$ from his knowledge base KB_i and the commitment store CS. However, his potential offers $\mathcal{PO}_{c,t}^i$ are generated from his agreement space $\mathcal{EA}_c^i$, based on his potential arguments generated at this step. Each agent tries to achieve his objective by using his potential arguments and a concession process, which is described in the next section.

4.1 The Notions of Concession and Agreement

We assume that at step t of a given negotiation, the agent i ($i \in \{b,s\}$) has a set of potential offers $\mathcal{PO}_{c,t}^i = \{o_1, \ldots, o_n\}$ such that: $\forall o_k \in \mathcal{PO}_{c,t}^i, \exists A_{t,k}^i \subseteq \mathcal{PA}_t^i$: $A_{t,k}^i = \mathcal{F}^i(o_k)$, i.e. each offer o_k is supported by a set $A_{t,k}^i \subseteq \mathcal{PA}_t^i$ of arguments ($1 \leq k \leq n$). This simply means the offers are computed from the set of potential arguments (offers are conclusions of arguments). We also assume that the agent's offers are ordered in the following way: $o_1 \preccurlyeq_o^i \ldots \preccurlyeq_o^i o_n$, so that the agent i proposes his most preferred offer o_n supported by the most relevant argument of the set $A_{t,n}^i$ (details about computing arguments' relevancy are given in (Mbarki, Bentahar, and Moulin 2007)). In his turn, the addressee (denoted here by $j \in \{b,s\}$) has two possibilities: (1) accept the offer o_n if no counter-argument can be generated; or (2) reject the offer by presenting a counter-argument justifying the rejection and so attacking the argument supporting o_n. This counter-argument may or may not support a counter-offer. In case of acceptance, the negotiation stops because an agreement is reached.

If a counter-argument has been presented by the agent j, the agent i uses his argumentation system to compute the set of his new potential arguments $\mathcal{PA}_{t+1}^i$ from his knowledge base and the new content of the commitment store. Then, he determines the next best offer from the new computed set $\mathcal{PO}_{c,t+1}^i$ of offers, which are supported by arguments in $\mathcal{PA}_{t+1}^i$. In fact, the offers, which are no longer supported by the new potential arguments are removed from the set of potential offers, and new offers can be added, since new arguments can emerge. Let us assume that the new offer is o_m. The gap between the values of the offers o_n and o_m represents what we call the *concession gap* made by the agent i at step $t+1$. Therefore, the strategy of concession is determined by the semantics used by the agent to compute his potential arguments. At each negotiation step, each negotiating agent uses his argumentation system to decide whether he will make a concession or not.

In a negotiation, the objective of the two negotiating agents is to reach an *agreement*, which satisfies the negotiation constraint of both agents in the best possible way. An agreement about an offer is reached between the two negotiating agents s and b iff (1) the value of the negotiation constraint (i.e. the value of the variable associated with this constraint), which is represented by the offer belongs to the new agreement space of both agents $\mathcal{EA}_c^{b\leftrightarrow s}$; (2) the addressee does not have any potential argument supporting a better offer; and (3) the addressee does not have any potential argument attacking the argument supporting this offer. If such an agreement is not reached, and no more offers can be made, then the negotiation is said to be failed.

4.2 Negotiation Progress

In our framework, at every negotiation step, agents can propose, accept, defend, or reject an offer. Before discussing the negotiation progress, let us first introduce the notions of *negotiable*, and *non-negotiable offers*

Definition 11 (Negotiable/Non-negotiable offer) *Let c be the negotiation constraint for a given negotiation, $o \in \mathcal{PO}_{c,t}^i$ ($i \in \{b,s\}$) an offer of the agent i for this constraint at step t, and v(o) the offer value.*

- *The offer o is negotiable iff:*

$$\begin{cases} v(o) > min_c^i & \textit{if } i = s \\ v(o) < max_c^i & \textit{if } i = b \end{cases}$$

- *The offer o is non-negotiable iff:*

$$\begin{cases} v(o) = min_c^i & \textit{if } i = s \\ v_(o) = max_c^i & \textit{if } i = b \end{cases}$$

When an agent makes a non-negotiable offer, no more offers can be made by the same agent as he reaches his limit. However, he can still support his offer by another argument, if the previous one is getting attacked. At each step $t > 0$ of the negotiation, the seller agent can make a new offer only if its value is less than or equal to the one of his previous offer, and the buyer agent can make a new offer only if its value is greater than or equal to the one of his previous offer. The negotiation succeeds when an offer is accepted. The negotiation stops if the best offer of the seller agent is less than his limit or if the best offer of the buyer agent exceeds his limit. The negotiation also stops if both negotiating agents perform in two consecutive steps the same act (an act is composed of an offer and an argument supporting this offer) where the offer is refused by the opponent. This simply means the agent reaches a non-negotiable offer, and the negotiation succeeds if the offer is accepted; otherwise, the negotiation fails.

After defining the elements that characterize an offer, we propose in the following an algorithm, which describes the negotiation progress (see Algorithm 1). This algorithm captures the dynamic behavior of agents participating in the negotiation. When he receives an argument from the agent j at step t, the agent i updates the set of potential arguments by computing the new set $\mathcal{PA}^i_{t+1}$ and consequently the new set $\mathcal{PO}^i_{c,t+1}$ of offers.

Proposition 1 (Completeness) *If $\mathcal{EA}^{b\leftrightarrow s}_c \neq \emptyset$, then the negotiation will end by achieving an agreement.*

***Proof* 1** *In a given negotiation, each negotiating agent has potential arguments supporting its offers. The offer values of each agent belong to his agreement space. We assume that $\mathcal{EA}^{b\leftrightarrow s}_c \neq \emptyset$ ($\mathcal{EA}^{b\leftrightarrow s}_c = \mathcal{EA}^b_c \cap \mathcal{EA}^s$). It is clear that any offer o proposed by an agent i ($i \in \{b, s\}$) whose value does not belong to $\mathcal{EA}^{b\leftrightarrow s}_c$ will be rejected by its opponent. Indeed, the other agent j ($j \in \{b, s\}$ and $j \neq i$) has only arguments that support potential offers where values are in his own agreement space. Therefore, the agent j will propose a counter-offer or will attack the argument that supports the offer o. Thereafter, even if the agent i will be able to defend his offer, he will have, after a finite number of defenses (depending on the number of arguments that support the offer o), to make concession because he will never change the attitudes of the agent j to accept an offer that is outside his agreement space. At a given moment, the values of the offers proposed by the two negotiating agents will be in $\mathcal{EA}^{b\leftrightarrow s}_c$. From that moment, each proposed offer will be acceptable to both negotiating agents, but may be not the best. By using our algorithm, the concession process will continue until one of the two gents accepts an offer that is equal to or better than his preferred offer. Hence, the negotiation will end with an agreement.*

Proposition 2 (Termination) *Using our concession theory, negotiation ends with or without agreement after a finite number of rounds.*

***Proof* 2** *In a given negotiation, each negotiating agent may make, accept, defend or reject an offer or an argument. At each step of the negotiation, each agent has a finite set of potential arguments and each argument cannot support two distinct offers. Therefore, each agent has at each negotiation step a finite set of potential offers. The seller agent must reject offers that are below his limit and the buyer agent must reject offers that exceed his limit. In addition, the seller agent must accept an offer that is greater than or equal to his best offer and the buyer agent should accept an offer that is equal to or less than his best offer. Each negotiating agent should defend his offer; it is the offer for which the agent has supporting arguments. Since the agent has a finite set of potential arguments, it is clear that the agent defends an offer a finite number of steps.*

Once the agent cannot defend his offer or attack the arguments that support the offer proposed by the opponent, he is obliged to recalculate his set of potential arguments and the set of potential offers taking into account the new arguments advanced by the opponent. In this case, it is obvious that the best new offer of this agent is less preferred than the previous one, because of the concession. Since both agents have finite sets of potential arguments, each negotiating agent may

Algorithm 1: Negotiation Dynamics

Negotiation(s, b, KB_s, KB_b, c, min^s_c, max^b_c)

```
k := 1
CS := ∅
limit_b := max^b_c
limit_s := min^s_c
IndNeg := true
i := Random_Agent(s, b)
IF i=s
  THEN j := b
  ELSE j := s
A^i_{t_k} :=Generate_Arguments(KB_i ∪ CS)
PA^i_{t_k} :=Potential_Arguments(A^i_{t_k})
PO^i_{c,t_k} :=Potential_Offers(c, PA^i_{t_k})
o_{t_k} :=Preferred_Offer(PO^i_{c,t_k})
a_{t_k} := Relevance(F^i(o_{t_k}), KB_i ∪ CS) //relevant argument
IF ((i = b) and (v(o_{t_k}) > limit_i)) or ((i = s) and
   (v(o_{t_k}) < limit_i)) // the offer cannot be made
   THEN {display("No offer can be made")
         IndNeg := false
         return 0}
Act(i, o_{t_k}, a_{t_k}) // proposition of an offer + an argument
CS := CS ∪ {a_{t_k}}
WHILE (IndNeg) DO
{
  k := k + 1
  A^j_{t_k} :=Generate_Arguments(KB_j ∪ CS)
  PA^j_{t_k} :=Potential_Arguments(A^j_{t_k})
  PO^j_{c,t_k} :=Potential_Offers(c, PA^j_{t_k})
  o_{t_k} :=Preferred_Offer(PO^j_{c,t_k})
  IF (PO^j_{c,t_k} = ∅) or (v(o_{t_k}) < limit_j)
     THEN {display("Negotiation failed")
           IndNeg := false
           return 0}
  IF (i = b) and (o_{t_k} ≥ o_{t_{k-1}})
     THEN {display("Agreement reached")
           IndNeg := false
           Accept(j, o_{t_{k-1}})
           return o_{t_{k-1}} }
     ELSE IF (i = s) and (o_{t_{k-1}} ≤ o_{t_k})
          THEN {display("Agreement reached")
                IndNeg := false
                Accept(o_{i,t_{k-1}})
                return o_{t_{k-1}}}
  Att_{j,t_k} := Attack_Set(j, a_{t_{k-1}}) ∪ Support(o_{t_k})
  IF Att_{j,t_k} ≠ ∅
  // no argument attacking the opponent's offer
  // or supporting the agent's offer
    THEN a_{t_k} := Relevance(Att_{j,t_k}, KB_j ∪ CS)
     IF (k > 4) and (o_{t_{k-1}} = o_{t_{k-3}}) and (a_{t_{k-1}} = a_{t_{k-3}})
          and (o_{t_{k-2}} = o_{t_{k-4}}) and (a_{t_{k-2}} = a_{t_{k-4}})
            // The same act is performed
               THEN {display("Negotiation failed")
```

$IndNeg := false$
return 0}
$CS := CS \cup \{a_{t_k}\}$
IF *Offer*$(conc(a_{t_k}))$
THEN // *the argument supports an offer*
$Act(j, o_{t_k}, a_{t_k})$
ELSE // *the argument does not support an offer*
$Act(j, Null, a_{t_k})$
$j = i$
IF (*i*=*s*)
THEN $i := b$
ELSE $i := s$
}

---

make only a finite number of concessions before reaching his limit. Thereafter, the negotiation ends with an agreement if the offer is accepted by the opponent or without agreement if the offer is rejected and the act which contains the same offer and the same support will be performed by the same agent in the next step.

Proposition 3 (Soundness) *If an agreement is reached in a given negotiation, therefore this agreement is a satisfactory compromise for both negotiating agents.*

***Proof* 3** *If an offer o is an agreement, therefore the value of the negotiation constraint c is in $\mathcal{EA}_c^{b \leftrightarrow s} = \mathcal{EA}_c^b \cap \mathcal{EA}_c^s$, such that $i \in \{b, s\}$. This means the value of the constraint c is acceptable for both negotiating agents and both agents do not have arguments that support a better offer.*

Example 2 *Let c_1 be the negotiation constraint for a given negotiation. We assume that the set of the potential arguments of agent s at step t is $\mathcal{PA}_t^s = \{a_1, a_2, a_3, a_4, a_5\}$. We also assume that the set of potential offers generate by the agent s at step t is $\mathcal{PO}_{c,t}^s = \{o_1 =< price, 700 >, o_2 =< price, 1000 >\}$, such that $o_1 \preceq_o^s o_2$, $\mathcal{F}^s(o_1) = \{a_1\}$ and $\mathcal{F}^s(o_2) = \{a_2, a_3, a_4, a_5\}$.*

The offer o_2 is more preferred than the offer o_1 for the agent s. Therefore, it is rational that this agent proposes the offer o_2. To support this offer, s has many alternatives, he can use one argument of the set $\{a_2, a_3, a_4, a_5\}$. To have more chance to convince the opponent, s needs to calculate the most relevant argument over the set $\mathcal{F}^s(o_2)$. We assume that s advances a_3, which is the most relevant argument according to his beliefs and that the opponent attacks this argument by an argument a_6.

In the case where the received argument a_6 is not accepted by s, he has to defend the offer o_2 by using another argument among those remaining in the set $\mathcal{F}^s(o_2)$, which still have the potential status. Otherwise, (if a_6 is accepted), the agent s has to recalculate his set of potential arguments and the set of potential offers at step $t + 1$. We assume that $\mathcal{PA}_{t+1}^s = \{a_1, a_7, a_8\}$ and $\mathcal{PO}_{c,t+1}^s = \{o_1 = (< price, 700 >, o_3 =< price, 900 >\}$, such that $o_1 \preceq_o^s o_3$, $\mathcal{F}^s(o_1) = \{a_1\}$ and $\mathcal{F}^s(o_3) = \{a_7, a_8\}$.

Since the agent s has accepted the argument a_6, the offer o_3 must be less preferred for this agent than o_2 and his concession gap at step $t+1$ is $Gap_{t+1}^s = 1000 - 900 = 100$.

5. Proof of Concepts Prototype

In this section, we briefly describe the implementation of our negotiation approach using the Java language. As Java classes, *Conversational Agents* taking part in negotiations have inconsistent knowledge bases and argumentation systems. The argumentation systems are implemented as Java modules and arguments for or against propositions or offers are computed from agent's knowledge bases and the commitment store. Agents can have multiple *extensions*, which are implemented as Java modules using the agent's *Attack* relation that enables agents to calculate, for each extension, the *Attack_set* (arguments that attack the extension) and the *Defended_by* set (arguments defended by the extension). Potential arguments are generated from agent's extensions. Potential *Offers* are implemented using agent's potential arguments. An offer is a conclusion of a potential argument that consists of a value assigned to the negotiation constraint. Furthermore, the prototype includes a Java class called *Relevance* that enables each agent to calculate the most relevant argument among his set of arguments that support his offer and/or the set of arguments that attack the opponent's argument.

To illustrate better the prototype, let us consider the case of two negotiating agents S and B as shown in Example 3.

Example 3 *Let $price$ be the negotiation constraint,* 600 *the limit of the buyer,* 500 *the limit of the seller.* $KB_S = \{A, A \rightarrow Price200, C, C \rightarrow Price300, D, D \rightarrow Price400, !F, E, !F \wedge E \rightarrow Price500, M, M \rightarrow !F, N \rightarrow !G\}$ *is the knowledge base of the seller and* $KB_B = \{F, F \rightarrow Price300, N, G, N \wedge G \rightarrow Price400, H, H \rightarrow Price500, L, L \rightarrow Price600, !F \rightarrow !F\}$ *is the knowledge base of the buyer.*

Figure 1 depicts the knowledge bases of the two negotiating agents along with the possible arguments that can be computed from these knowledge bases for each agent. For instance, the agent S has an argument $(\{A, A \rightarrow Price200\}, Price200)$ supporting the offer $< Price, 200 >$ and the agent B has an argument $(\{F, F \rightarrow Price300\}, Price300)$ supporting the offer $< Price, 300 >$. The agents' potential arguments and resulting offers for the first round are illustrated in Figure 2. For example, the preferred offer for the buyer B is $< Price, 300 >$ and for the seller S is $< Price, 500 >$. Finally, the negotiation dialogue is illustrated in Figure 3 that shows the two possible dialogues depending on which agent starts the negotiation (i.e. which agent makes the first offer). For example, when the buyer starts, the first offer is $< Price, 300 >$; otherwise, the starting offer is $<Price, 500>$. In both cases, an agreement is reached, which is $< Price, 500 >$.

6. Discussion

In (Rahwan et al. 2004), the authors have proposed a negotiation-based argumentation framework stressing the importance of using and exchanging arguments in a negotiation setting to achieve better agreements. The authors have described the elements that an argumentation-based framework should contain. These elements are classified into external to the agents and internal. The external components

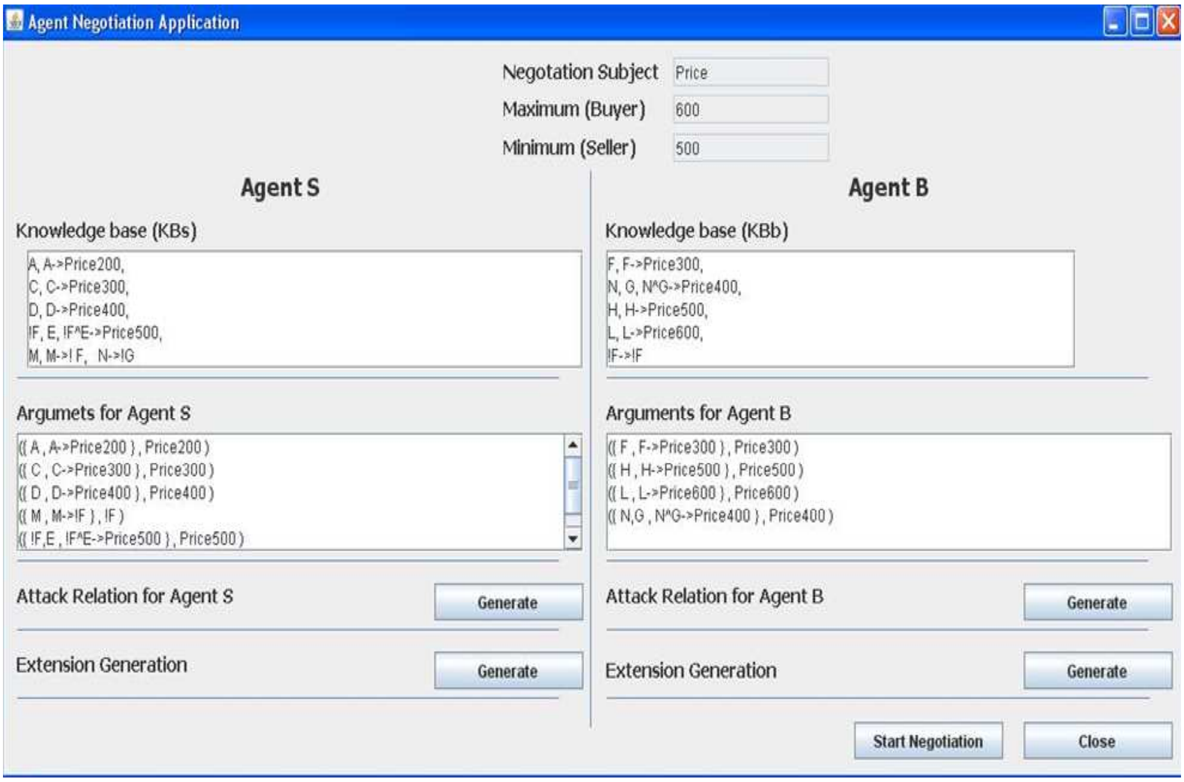

Figure 1: Agents' knowledge bases and arguments

are the communication and domain languages, the negotiation protocol, and various information stores. The internal elements are necessary to enable an agent to conduct argumentation-based negotiation. They are related to the processes of argument and proposal evaluation, argument and proposal generation, and argument selection. The paper identifies the different components and associated challenges, but does not present any concrete negotiation algorithm as we did in this paper. Our work focuses more on the internal aspect, namely the argumentation process (argument generation, selection and evaluation) for generating and evaluating offers and making concessions.

In real negotiations, agents need to make concessions. Recently, researchers proposed approaches in which agents can make concessions in different ways (Amgoud, Dimopoulos, and Moraitis 2007; Faratin, Sierra, and Jennings 1998; 2002; Ros and Sierra 2006; Rahwan et al. 2009). In (Faratin, Sierra, and Jennings 1998; 2002; Ros and Sierra 2006), agents can make concessions by using different tactics and/or a trade-off algorithm. These approaches do not allow negotiating agents to reason about their beliefs to justify their offers and to influence the behavior of their opponents. In our approach, each negotiating agent uses the negotiation constraint to ensure his satisfaction and to avoid the risk to concede everything to the other agent. In addition, by using their argumentation system, our agents are able to make concessions when it is necessary, which allows them to have more chances to reach an agreement. Recently, (Rahwan et al. 2009) have proposed an argumentation-based negotiation approach to maximize positive interaction among the goals of cooperative agents. This approach is proposed in the context of cooperative negotiation. However, our approach focuses on the connection between arguments and offers in the context of a non-cooperative negotiation.

In (Amgoud, Dimopoulos, and Moraitis 2007), by using the acceptability semantics proposed by Dung (Dung 1995), each negotiating agent is able to classify his arguments in three sub-sets: accepted, rejected and undecided arguments. According to the classification of his arguments, each negotiating agent can also define four sub-sets of offers: acceptable, rejected, negotiable and non-supported offers. These sub-sets of offers are respectively denoted by $\mathcal{O}_a$, $\mathcal{O}_r$, $\mathcal{O}_n$ and $\mathcal{O}_{ns}$. From this partition of the set of offers, a preference relation between offers, denoted by $\rhd$, is defined: $\mathcal{O}_a \rhd \mathcal{O}_r \rhd \mathcal{O}_n \rhd \mathcal{O}_{ns}$ (an offer of the sub-set $\mathcal{O}_a$ is more preferred than another one belonging to the others sub-sets). Amgoud and her colleagues (Amgoud, Dimopoulos, and Moraitis 2007) consider that an offer o is a concession iff $o \in \mathcal{O}_x$ such that $\exists \mathcal{O}_y \neq \emptyset$, and $\mathcal{O}_y \rhd \mathcal{O}_x$. The main drawback of this approach is that if a negotiating agent proposes a less preferred offer than his previous one and the two offers belong to the same sub-set ($\mathcal{O}_a$ or $\mathcal{O}_r$ or $\mathcal{O}_n$ or $\mathcal{O}_{ns}$), then the offer is not considered as a concession. However, in our framework, there is a preference relation between all the agent's potential offers. Hence, if a proposed offer is not accepted and if it cannot be supported, the negotiating

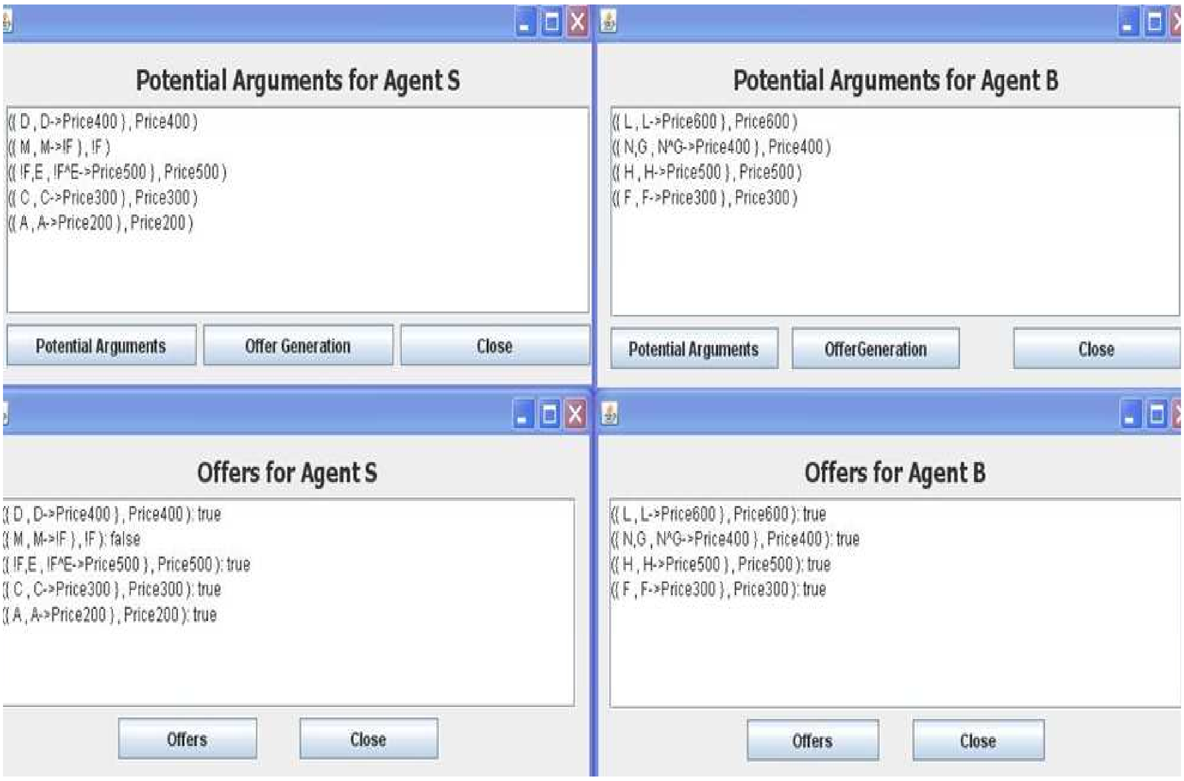

Figure 2: Agents' potential arguments and offers for round 1

agent needs to propose another offer, which is less preferred than the previous one. In addition, Amgoud and her colleagues consider that the set of potential offers is fixed during the negotiation. This does not reflect the dynamic aspect of negotiation. Whereas, in our approach, we use negotiation constraints and take into account the received arguments to enable negotiating agents to calculate their new potential offers (including new emergent offers) at every negotiation step.

7. Conclusion

In this paper, we have proposed a framework for constraints-based negotiation using argumentation. In this framework, each negotiating agent is equipped with a reasoning model that enables him to compute the potential arguments according to his beliefs and negotiation constraints. Using his potential arguments and constraint agreement space, each negotiating agent calculates his set of potential offers. The sets of potential arguments and potential offers of an agent are dynamic and may change at every negotiation step, which reflects the dynamic aspect of negotiation. Indeed, it is more natural to assume that agents may have different sets of offers, which will evolve during a negotiation. Furthermore, each negotiating agent uses his argumentation system to avoid the risk to concede everything to the opponent. In that way, the agent is able to reach a satisfactory agreement.

Furthermore, we have proposed an algorithm that describes the negotiation evolution. We have proved that each agent always chooses the best preferred offer and the negotiation always ends with or without agreement after a finite number of rounds. If there is an agreement in a given negotiation, we have also proved that both negotiating agents reach a satisfactory compromise. An implementation of this algorithm has been discussed.

As extensions of this work, we plan to refine the algorithm in order to take into account dependent negotiation constraints. Indeed, negotiating agents may have different sets of offers and each proposed offer must be defined in terms of all constraints of both agents. We also plan to investigate the case where negotiating agents make concessions by reducing the values of certain negotiation constraints and increasing others. We are also interested in calculating metrics on different parameters, such as the competence of each agent relatively to the selection of his arguments and the number of his concessions that were made during the negotiation.

References

Amgoud, L.; Dimopoulos, Y.; and Moraitis, P. 2007. A unified and general framework for argumentation-based negotiation. *In 6th International Joint Conference on Autonomous Agents and Multi-Agents Systems, AAMAS* 1–8.

Dung, P.; Kowalski, R.; and Toni, F. 2006. Dialectic proof procedures for assumption- based, admissible argumentation. *Artificial Intelligence* 170:114–159.

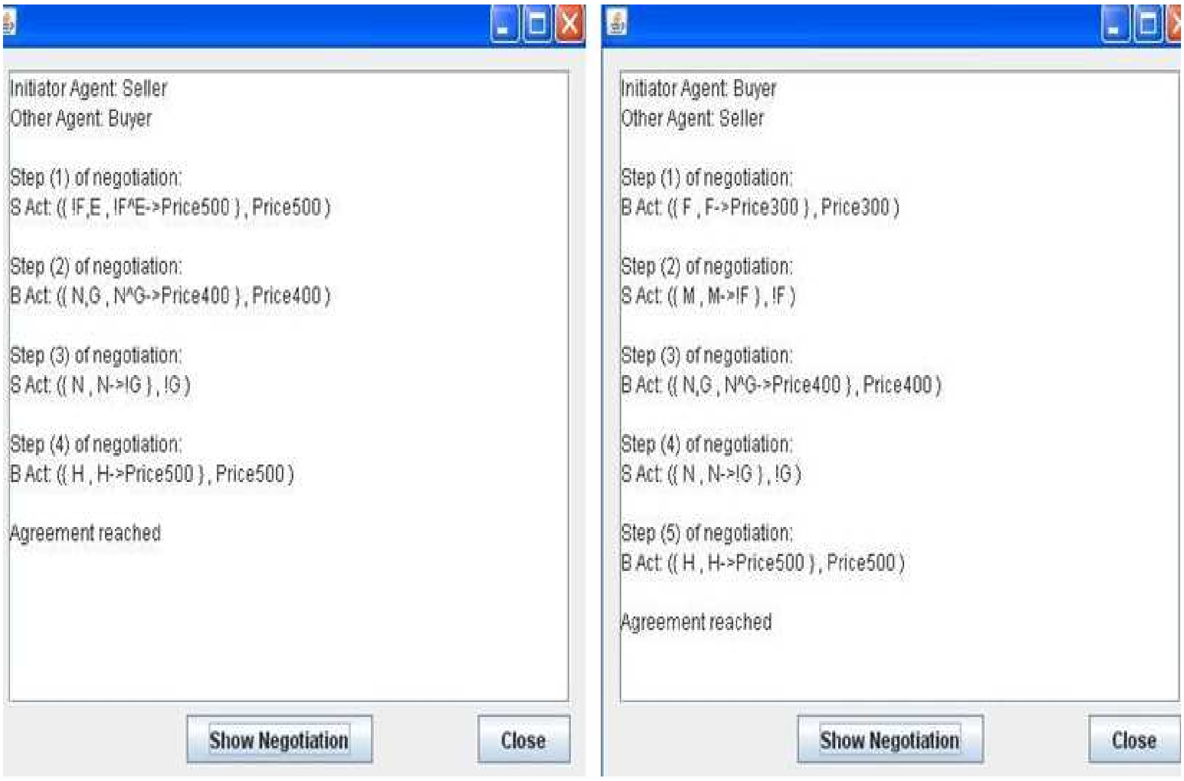

Figure 3: Negotiation dialogue: Seller starts (left) - Buyer starts (right)

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence* 77(2):321–358.

Faratin, P.; Sierra, C.; and Jennings, N. R. 1998. Negotiation decision functions for autonomous agents. *In Robotics and Autonomous Systems 24* 159–182.

Faratin, P.; Sierra, C.; and Jennings, N. R. 2002. Using similarity criteria to make issue trade-offs in automated negotiations. *In Artificial Intelligence 142* 205–237.

Hindriks, K.; Jonker, C. M.; Kraus, S.; Lin, R.; and Tykhonov, D. 2009. GENIUS - Negotiation Environment for Heterogeneous Agents. *Proceedings of the Eighth International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2009)* 1397–1398.

Mbarki, M.; Bentahar, J.; and Moulin, B. 2007. Specification and complexity of strategic-based reasoning using argumentation. *Argumentation in Multi-Agent Systems* LNAI 4766:142–160.

Rahwan, I.; Ramchurn, S. D.; Jennings, N. R.; McBurney, P.; Parsons, S.; and Sonenberg, L. 2004. Argumentation-based Negotiation. *The Knowledge Engineering Review* 18(4):343–375.

Rahwan, I.; Pasquier, P.; Sonenberg, L.; and Dignum, F. 2009. A Formal Analysis of Interest-based Negotiation. *Annals of Mathematics and Artificial Intelligence* 55(3-4):253–276.

Ros, R., and Sierra, C. 2006. A negotiation meta strategy combining trade-off and concession moves. *Autonomous Agents and Multi-Agent Systems. Springer Netherlands, 12(2)* 163–181.

Toni, F. 2007. Assumption-Based Argumentation for Selection and Composition of Services. *Computational Logic in Multi-Agent Systems (CLIMA VIII)* LNAI 5056:231–247.

Zadeh, L. 1965. Fuzzy sets. *Information Control, 8* 338–353.