Abstract. A typical scenario for negotiation involves benevolent agents which cannot reach their goals by themselves because they do not have some resources or they do not know how to use them to reach their goals. Also, they may have incomplete or wrong information about the other agent’s goals and resources. This paper presents an approach for automated negotiation based on argumentation, where agents offer resources and knowledge for exchanging in order to achieve their goals. Such offers are formalized as arguments, whose generation and interpretation is based on belief revision techniques. This approach is presented through a high-level algorithm that has been implemented in logic programming. Different theoretical properties associated with this approach have been formalized and proved in Coq. As a case study we show how our approach can be used to solve a slightly modified version of the well-known Hammer-Nail Problem.

1 Introduction

Negotiation is a form of interaction in which two or more agents with different goals find some acceptable agreement. A typical scenario for negotiation involves two agents who have the need to collaborate for mutual benefit. At present, there is no agreed approach to characterising all negotiation frameworks. In [9] it has been argued that automated negotiation research can be considered to deal with three broad topics: a) Negotiation Protocols (the set of rules that govern the
interaction); b) Negotiation Objects (the range of issues over which agreement must be reached) and c) Agents’ Decision Making Model (which accounts for the decision making apparatus the participants employ to act in line with the negotiation protocol in order to achieve their objectives).

Moreover, different approaches can be used to model negotiation in a MAS setting. In [13], Rahwan et al. distinguish three different kinds of such approaches: those which are game-theoretic, those which are heuristic-based, and finally those based on argumentation (argumentation-based negotiation or ABN). Game-theoretic approaches are based on studying and developing strategic negotiation models based on game-theory precedents [14]. The interaction in the negotiation process is considered as a game in which each agent tries to maximize her utility. Given a protocol, most researchers in this line of work attempt to analyze the optimal strategy. While this approach is very powerful in terms of results analysis, it suffers from some drawbacks due to the assumptions upon which it is built. The most important ones are that agents are only allowed to exchange offers without any other information. On the other hand, heuristic-based approaches come to cope with some limitations of the game-theoretic approach. Some strong assumptions made in game-theoretic approach are relaxed using heuristics. Most of these assumptions concern the notion of rationality of agents as well as their resources. The support for particular heuristics is usually based on empirical testing and evaluation. In general, these methods offer approximations to the decisions made according to game-theoretic studies. Unfortunately, most of those approaches assume that agents have unbounded computational resources and that the space of outcomes is completely known. In most realistic environments, however, these assumptions fail.

Finally, argumentation-based negotiation has been proposed as an alternative to the two previous approaches (e.g. [11], [12], [17], [13], [4]). This approach allows the negotiating agents not only to exchange offers but also reasons that support these offers in order to mutually influence their preference relation on the set of offers, and consequently the outcome of the dialogue. Moreover, as the agents that negotiate usually have incomplete information about the others, the exchange of arguments gives them information that make it possible to update their beliefs.

In [13] Rahwan et al. depict an ABN framework in terms of the interaction between the negotiating agents and the environment. They outline those essential elements in the design of an ABN framework, classifying them in external and internal elements to the agent. External elements are those that define the environment in which ABN agents operate and interact: Communication language: the language that facilitates the negotiation, normally including basic locutions such as propose, accept and reject; Domain language: the language for referring to concepts of the environments, agents, resources, etc.; Negotiation Protocol: the conventions that govern the interaction among participants; and Information stores: stores that keep track of relevant information for the negotiation externally (e.g. past utterances, reputations of the participants, etc.). Internals elements are the main components needed by an agent in order to
be capable of engaging in negotiation. Figure 1 shows the sketch of an ABN agent. The **locution interpretation** component parses incoming messages. These locutions usually contain a proposal, or an acceptance or rejection message of a previous proposal. The **proposal database** component stores proposals for future reference. The **proposal evaluation and generation component** makes a decision about whether to accept, reject or generate a counterproposal, or even terminate the negotiation. The **locution generation** component sends the response to the relevant party. The **argument interpretation** component updates the agent’s mental state accordingly. Finally, the **argument generation** mechanism is responsible for deciding what response to actually send to the counterpart and what (if any) arguments should accompany the response.

![Fig. 1. Conceptual elements of an ABN agent (Rahwan et al. [13])](image)

In order to formalize their offers in a negotiation setting, ABN agents must be able to generate, select and evaluate arguments associated with such offers, updating their mental state accordingly. In this paper, we focus on such stage of an ABN agent, proposing an argumentation-based negotiation model between two cooperative agents. For our analysis we will assume that each agent is **benevolent** (he will always try to do what is asked for if he is able to do so) and **truthful** (i.e. he will not knowingly communicate false information). Besides, we will assume that bot agents cannot reach their respective goals by themselves, so that they have to ask for help from one another. The agents can thus exchange different resources, including the knowledge associated with possible plans to reach their goals. The resulting negotiation dialog is composed by an exchange of proposals which adopt the form of an argument whose claim is a possible exchange (which
are the resources the agent is asking for and what he is willing to offer in return). As the agents initially may have incomplete or wrong information about the other agent’s goals and resources, during the negotiation process they update their beliefs and consequently, their mental state, according the arguments exchanged. Thus, in the context of the ABN framework previously described, we will use a belief revision approach for both argument interpretation and argument generation.

As motivational example we work on a slightly modified version of the well-known Hammer-Nail-Mirror (HNM) example [11]. We will assume two benevolent agents $A_{g1}$ and $A_{g2}$. Agent $A_{g1}$ has as goal hanging a picture, and it has a screw, a screwdriver and a hammer. Also, he knows how a hammer and a nail can be used to hang a picture and how a screw and a screwdriver can be used to hang mirrors. On the other hand, Agent $A_{g2}$ has as goal to hang a mirror, and it has a nail and the knowledge of how to hang a mirror using a hammer and a nail. Neither $A_{g1}$ nor $A_{g2}$ can reach their goals on the basis of their knowledge and resources. Then, they need to perform some exchanges in order to do so.

The remainder of this paper is structured as follows: in Sections 2 and 3 we define the agent architecture and the negotiation protocol, formalizing the notions of proposal, dialogue and deal. Then in Section 4 we show how to integrate belief revision operators in a high-level algorithm for solving negotiation problems between two argumentative agents. We also discuss some theoretical properties of our approach, which have been formalized and proved in Coq\(^4\). In Section 5 we show how the HNM problem can be solved in the context of our proposal. Finally, we discuss conclusions as well as related and future work in Section 6.

2 Agent Architecture

For our negotiation scenario, each agent will have in his mental state, resources, goals and plans, as well as beliefs on the other agent’s resources and goals. From the information provided in the mental state, an agent will decide which proposals he can offer the other agent in order to reach an agreement. In order to characterize the agent architecture, we will consider a propositional language $\mathcal{L}$, in which the following subsets are distinguished:

- $O_\mathcal{L}$: a set of atoms representing objects which are the resources an agent may have (e.g. nail, hammer).
- $G_\mathcal{L}$: a set of atoms representing goals (e.g. hangMirror, hangPicture).
- $P_\mathcal{L}$: a set of propositional formulae encoding plans, which may involve objects for achieving a goal. (e.g. nail $\land$ hammer $\rightarrow$ hangPicture).

\(^4\)Coq (http://coq.inria.fr/) is an interactive theorem prover. It provides a formal language to write mathematical definitions, executable algorithms and theorems together with an environment for semi-interactive development of machine-checked proofs.
In the literature, the term resources generally is considered in a broadest sense and can represent commodities, services, time, money, etc. In short, anything that is needed to achieve something. In this work, the set of resources, noted by $R_L$, will include also plans i.e. $R_L = O_L \cup P_L$. A plan represents the agent’s knowledge of how to use objects to reach goals.

**Definition 1 (Agent Mental State).** Let two agents: $Ag_i$, $Ag_j$ be involved in a negotiation. The mental state of an Agent $Ag_i$ is a tuple $M S_i = \langle R_i, G_i, B_iR_j, B_iG_j, H_i \rangle$, where: $R_i, B_iR_j \subset R_L; G_i, B_iG_j \subset G_L$ and $H_i$ is the history of the negotiation.

Namely, $Ag_i$ mental state has a set of available resources ($R_i$), a set of goals to achieve ($G_i$), as well as belief sets about which resources are available for the opponent agent $Ag_j$ ($B_iR_j$), and which goals he believes the agent $Ag_j$ has ($B_iG_j$). Also, its mental state includes the history of the dialogue with $Ag_j$ (see Def. 5 for the dialogue definition).

**Example 1.** Consider the $HN$ problem given in Section 1. In the beginning of the negotiation process, $Ag_1$’s mental state may be represented as $M S_1 = \langle R_1, G_1, B_1R_2, B_1G_2, H_1 \rangle$ where:

- $R_1 = \{\text{screw, screwdriver, hammer, hammer} \land \text{nail} \rightarrow \text{hangPicture}, \text{screw} \land \text{screwdriver} \rightarrow \text{hangMirror}\}$
- $G_1 = \{\text{hangPicture}\}$
- $B_1R_2 = \emptyset$
- $B_1G_2 = \emptyset$
- $H_1 = \emptyset$

The decision making apparatus the agents employ to act in order to achieve their objectives depends on their mental state. This apparatus will be in charge of computing those proposals the agent will make to the other agent.

As the first dialogue move associated with the initial proposal is a particular one, we will single it out by using a initialization function $Init$. Further proposals and counter-proposals are computed by another function $Answer$. Formally:

**Definition 2 (Decision Making Apparatus).** The decision making apparatus of an Agent $Ag_i$ is a tuple $DM_i = \langle Init_i, Answer_i \rangle$, where

$Init_i : MS_i \rightarrow MS_i \times Proposal$

$Answer_i : MS_i \times Proposal \rightarrow MS_i \times Proposal \cup \{accept, withdraw\}$

We will purposely leave unspecified the actual definitions of $Init_i$ and $Answer_i$ at this stage. Later on in Section 4 we will provide their specification through of high level algorithms. In summary, in our approach an ABN agent model will be composed by his mental state and his decision making apparatus. Formally:

**Definition 3 (Agent Model).** An agent $Ag_i$ is a tuple $< MS_i, DM_i >$, where $MS_i$ is its mental state and $DM_i$ its decision making apparatus.
3 Generating Proposals as Arguments to Reach Deals

Based on their mental states, the agents using its decision apparatus will generate proposals towards reaching their goals. In our formalization, a proposal is a statement that includes what the agent wants to receive and what the agent is willing to give in return, together with an explanation justifying why an agent needs what he is asking for. Thus, proposals will have the following intended meaning:

I propose that you provide me $R_{\text{get}}$ in exchange of $R_{\text{give}}$, because if I use $R_{\text{own}}$ then I can achieve $G$.

where $R_{\text{get}}$, $R_{\text{own}}$, and $R_{\text{give}}$ stand for resources, and $G$ is a set of goals.

Note that an agent’s proposal can be thought of as an argument whose claim is associated with what the agent needs to achieve his goals (namely $R_{\text{get}}$) and the resources that the agent offers in exchange ($R_{\text{give}}$), together with its support, i.e. the reasons given for requesting that resource from the other agent. The following definition formalize this concept.

**Definition 4 (Proposal).** Let $R_{\text{get}}$, $R_{\text{own}}$, and $R_{\text{give}}$ subsets of $R_L$, and $G$ subset of $G_L$. A Proposal is a tuple $\langle A, (R_{\text{get}}, R_{\text{give}}) \rangle$, where $(R_{\text{get}}, R_{\text{give}})$ corresponds to the claim of the argument, and $A = (R_{\text{own}}, G)$ provides the support associated with the claim, and the following conditions are hold:

\[
\begin{align*}
R_{\text{get}} \cup R_{\text{own}} &\vdash G \\
R_{\text{own}} &\not\vdash G \\
R_{\text{give}} \cap (R_{\text{get}} \cup R_{\text{own}}) &= \emptyset
\end{align*}
\]

Notice that (1) states that both sets of resources $R_{\text{own}}$ and $R_{\text{get}}$ are needed for the agent to reach the goal $G$; (2) means the agent cannot reach the goal using only $R_{\text{own}}$ and (3) states the set of resources $R_{\text{give}}$ is not needed by the agent to reach $G$.

**Example 2.** (Example 1 continued) Suppose that in this scenario $Ag_2$ begins the negotiation process making to $Ag_1$ the following proposal:

I propose that you provide me with a hammer in exchange for nothing, because if I use a nail and the knowledge about how to hang a mirror using a nail and a hammer, then I can hang a mirror.

Then this proposal is denoted by:

\[((\{\text{nail}, \text{nail} \land \text{hammer} \rightarrow \text{hangMirror}\}, \{\text{hangMirror}\}, \{\text{hammer}\}, \{\})\]

A dialogue between two agents will be defined as a finite sequence of proposals (which account for arguments in favor of some particular exchange), performed alternately by each of the agents involved in the dialogue, ending with accept (there is a deal) or withdraw (no deal is possible).

\footnote{We write $X \vdash G$ whenever $G \subseteq Cn(X)$, where $Cn$ is a logical consequence operator.}
Definition 5. Negotiation Dialogue A dialogue between agents $Ag_i$ and $Ag_j$ is a finite sequence of utterances $[u_1, ..., u_{n-1}, u_n]$ where for $r < n$, $u_r$ is a proposal and $u_n \in \{accept, withdraw\}$, such that: (1) there are no repeated utterances, i.e. $u_s \neq u_t$, with $t, s < n$; (2) utterance $u_k$ with $k > 1$ is performed by Agent $Ag_i$ only if utterance $u_{k-1}$ is performed by Agent $Ag_j$ (i.e. agents alternate moves). A dialogue will be initiated by $Ag_i$ iff $u_1$ is performed by $Ag_i$.

Note that dialogues can be warranted to be finite, as there is a finite set of possible combinations of proposals and utterance repetition is not allowed. From Defs. 2 and 5, we can see that a dialogue between agents $Ag_i$ and $Ag_j$ will be started by one of the agents with a proposal computed by $Init$, followed by a counter-proposal by the other agent computed by $Answer$, a counter-counter-proposal by the first agent, and so on. Without loss of generality we assume the agent $Ag_i$ is the one which starts the negotiation dialogue. Fig. 2 represents the negotiation dialogue flow initiated by $Ag_i$ as a finite-state machine.

![Diagram](image)

Fig. 2. Dialogue flow initiated by $Ag_i$.

3.1 Proposal Evaluation

As previously mentioned, we assume agents $Ag_i$ and $Ag_j$ cannot reach their goals by their own and therefore, the problem each agent faces is to find a suitable exchange of resources in the space of possible ones ($P(R_i) \times P(R_j)$) in order to reach his own goal. The proposals can be thought of as an argument $((R_{own}, G), (R_{get}, R_{give}))$ supporting an exchange of resources. By definition, the pair of resources $(R_{get}, R_{give})$ provides a solution to reach $Ag_i$’s goal and we define the function $\odot$ that assigns to each proposal $((R_{own}, G), (R_{get}, R_{give}))$ its associated solution.$^6$

Following [13], we assume that in our approach agents have an objective consideration when they evaluate proposals (i.e. they consider a proposal as a

$^6$ The function $\odot$ corresponds to the second component projection.
tentative proof to reach their goals, and they verify it by examining the validity of its underlying assumptions, such as resources availability. Since each agent is aware of his own resources and goals, he can determine first, in a selfish way, which are the exchanges that present solution for his problem. This is formalized in the following definition.

**Definition 6.** Let $Aq_i$ be an agent involved in a negotiation, where its mental state is $MS_i = \langle R_i, G_i, B_i, R_j, B_j, H_i \rangle$. A solution for $Aq_i$ is any pair $(X, Y)$, $X, Y \subseteq R_L$ such that:

1. $X \subseteq R_i$
2. $(R_i - X) \cup Y \vdash G_i$

We noted by $S_i$ the set of all the solutions for $Aq_i$.

Note that $X$ stands for those resources that $Aq_i$ is willing to give to $Aq_j$, whereas $Y$ is the set of resources that are given to $Aq_i$ to achieve his goal. In a similar way $S_j$ is defined. Then, a deal for $Aq_i$ and $Aq_j$ will be a solution which is applicable for both of them and it is defined as follows.

**Definition 7.** We will say that $(X, Y)$ where $X, Y \subseteq R_L$, is a deal for $Aq_i$ and $Aq_j$ iff $(X, Y) \in S_i \cap S_j$. We will denote with $D$ the set of all deals between $Aq_i$ and $Aq_j$.

With the definitions presented above, the agents evaluation process can be defined in a simple way as follows: If $prop = \langle (R_{own}, G_i), (R_{get}, R_{give}) \rangle$ is an $Aq_i$ proposal, then $prop$ will be accepted by $Aq_j$ if $(R_{give}, R_{get}) \in S_j$. Notice that a proposal $prop$ will be accepted only if it is a deal.

# 4 Integrating Belief revision in Argumentative Agents

In this section we show how belief revision can be used in an ABN agent to improve two important issues in a negotiation: the proposal interpretation and generation. We consider that the information contained in a proposal can be used by an agent to revise the beliefs he has about the other agent and then, having more accurate beliefs he can make proposals which are more likely to be accepted. In a first place, we summarize some notions of the belief change theory that will be applied in our approach.

## 4.1 Belief Revision operators

Classic belief change operations introduced in the AGM model [1] are known as expansions, contractions and revisions. An expansion incorporates a new belief without warranting the consistency of the resulting epistemic state. A contraction eliminates a belief $\alpha$ from the epistemic state as well as all those beliefs that make the inference of $\alpha$ possible. Finally, a revision incorporates a new belief $\alpha$
to the epistemic state warranting a consistent result, assuming that α itself is consistent.

As discussed before, in our setting we assume that the agents have their own beliefs about the other agent’s resources and goals. It must be noted that the sets of resources and objectives do not change during the negotiation. Only if a deal succeeds at the end of the negotiation process, the actual exchange of resources will take place and consequently the sets $R_{give}$ and $R_{get}$ will be changed. In order to model such a negotiation process in terms of belief revision we will use the notion of Choice kernel Set and Multiple Choice contraction proposed by Hansson [8] and followed by Fermé et al [7]. These notions will be useful for providing a practical approach to belief revision in our context. In order to make this paper self-contained, we provide below a brief review of the formal definitions involved.

**Definition 8 ([7]).** Let $L$ be a logical language, $Cn$ a consequence operator, $R \subseteq L$ and $G \in L$. Then $R \perp G$ is the set of all $X \subseteq R$ such that

1. $G \subseteq Cn(X)$
2. if $X' \subset X$ then $G \not\subseteq Cn(X')$

The set $R \perp G$ is called Choice kernel Set, and its elements are called $G$-kernels of $R$.

Informally, a Choice kernel Set is a minimal belief subset of the epistemic state from which $G$ can be deduced. An element in $R$ contributes to make $R$ imply $G$ if and only if it is an element of some $G$-kernels of $R$. Therefore, removing at least one element of each $G$-kernels of $R$, it is no longer possible to derive $G$. The function that selects sentences to be removed will be called an incision function since it makes an incision into every $G$-kernel.

**Definition 9 ([7]).** A function $\sigma$ is an incision function $\sigma$ for $R$, iff satisfies for all $G$:

1. $\sigma(R \perp G) \subseteq \bigcup(R \perp G)$
2. If $\emptyset \neq X \in R \perp G$, then $X \cap \sigma(R \perp G) \neq \emptyset$

Then, the multiple choice contraction operator removes the elements selected by an incision function as follows.

**Definition 10 ([7]).** Let $\sigma$ be an incision function for $R$ and $G \in L$. The multiple choice contraction $\approx$ for $R$ is defined as:

$$R \approx G = R - \sigma(R \perp G)$$

Next, a revision operator is expressed by using two suboperations: first a contraction and then an expansion (i.e. adding $G$ to the resulting set).

**Definition 11 ([8]).** Let $\approx$ be a global kernel contraction. Given a set of sentences $K$, we define for any set $A$ the revision operator $*$:

$$R * G = (R \approx \neg G) \cup \{G\}$$
4.2 Argument Generation

In a negotiation dialogue, the beliefs a particular agent has about the other agent’s resources and goals are significant for proposal generation, as they can help reaching a deal. From this information, an agent can infer which proposals he believes are more suitable for the other and consequently, more likely to be accepted. To formalize this notion we define the following concepts.

**Definition 12.** Let $A_g_i$ and $A_g_j$ be two agents and $X, Y \subseteq R_L$, we will say that $A_g_i$ believes $(X, Y)$ is a solution for $A_g_j$ whenever:

1. $Y \subseteq B_i R_j$ and
2. $(B_i R_j - Y) \cup X \vdash B_i G_j$.

We will define $B_i S_j = \{ (X, Y) \mid A_g_i$ believes $(X, Y)$ is a solution for $A_g_j \}$.

**Definition 13.** Let $A_g_i$ and $A_g_j$ be two agents, we will say that $A_g_i$ believes $(X, Y)$ is a deal iff:

1. $X \subseteq R_i$,
2. $Y \subseteq B_i R_j$,
3. $(R_i - X) \cup Y \vdash G_i$ and
4. $(B_i R_j - Y) \cup X \vdash B_i G_j$.

We will define $B_i D = \{ (X, Y) \mid A_g_i$ believes $(X, Y)$ is a deal $\}$.

From definitions 12 and 13 the following propositions hold:\footnote{All the propositions and their proofs were formalized in Coq and are available at http://web.cifasis-conicet.gov.ar/~pilotti/Automated-Agent_Negotiation.v}

**Proposition 1.** $(X, Y) \in S_i$ and $(X, Y) \in B_i S_j \iff (X, Y) \in B_i D$.

**Proposition 2.** $(X, Y) \in B_i D$ and $(X, Y) \in S_j \Rightarrow (X, Y) \in D$.

**Proposition 3.** $(X, Y) \in B_i D$ and $(X, Y) \in B_j D \Rightarrow (X, Y) \in D$.

**Proposition 1** states that if a pair $(X, Y)$ is solution for $A_g_i$ and he believes that it is also a solution for $A_g_j$, then $A_g_i$ believes that $(X, Y)$ is a deal, and the reciprocal also is hold. Similarly, **Proposition 2** asserts that if the agent $A_g_i$ believes that $(X, Y)$ is a deal and $(X, Y)$ is also a solution for $A_g_j$, then $(X, Y)$ is a deal. Finally, **Proposition 3** states that if both agents believe that $(X, Y)$ is a deal, then it holds that $(X, Y)$ is a deal.

**Fig. 3** shows the set of solutions from the viewpoint of $A_g_i$. The dotted line represents that the agent does not know $S_i$. Because of this, $A_g_i$ can not be sure of making a proposal $prop$ such that $\triangleright (prop) \in D$. So, in order to entice the $A_g_j$ to accept some proposed agreement, the $A_g_i$ must choose a proposal $prop$ such that he believes that its associated solution is a deal, i.e. $\triangleright (prop) \in B_i D$.

The function $Gen$ is defined to compute the proposals that are solution to $A_g_i$ (i.e. $\triangleright (prop) \in S_i$) and to compute proposals that are potential solutions (i.e. $\triangleright (prop) \in B_i S_j$). The $Gen$ function is specified using belief revision operations and some properties that follow from its specification are given.
Definition 14. Let $R', R'' \subset R_L$ and $G' \subset G_L$, we define a function $\text{Gen}$ as follows:

$$\text{Gen}(R', R'', G') = \{ \langle (R_{own}, G), (R_{get}, R_{give}) \rangle : R_{get}, R_{own}, R_{give} \subset R_L, G \subset G_L, R_{get} \cap R' = \emptyset, R_{own} \subseteq R', (R_{own} \cup R_{get}) \in (R' \cup R'' \cup R_{get}) \perp G', R_{give} \subseteq R' - R_{own}, G = G' \}$$

In Def. 14 the $\text{Gen}$ function receives two sets of resources ($R'$ and $R''$) and a set of goals ($G'$). As an outcome it generates a set of proposals $\text{prop} = \{ \langle (R_{own}, G'), (R_{get}, R_{give}) \rangle \}$ where $R_{get}$ and the first set of resources ($R'$) are disjoint sets, but $R_{own}$ is a subset of it. The union of $R_{get}$ and $R_{own}$ is a minimal set from which $G$ can be deduced. The set $R_{give}$ corresponds to the unused resources of $R'$ to achieve $G$.

Proposition 4. Given an agent $A_{Qi}$, where his mental state is $\text{MS}_i = \langle R_i, G_i, B_i R_j, B_i G_j, H_i \rangle$, then the following holds:

1. If $\text{prop} \in \text{Gen}(R_i, B_i R_j, G_i)$ then:
   (a) $\text{prop} \in \text{Proposal}$
   (b) $\circ(\text{prop}) \in S_i$
2. If $\text{prop} \in \text{Gen}(B_i R_j, R_i, B_i G_j)$
   then $\circ(\text{prop}) \in B_i S_j$

Condition (1) in Prop. 4 establishes that the $\text{Gen}$ function computes the proposal that are solutions for $A_{Qi}$, from his point of view, namely, using as parameters his resources ($R_i$), his belief about the other agent’s resources ($B_i R_j$) and his goal ($G_i$). On the other hand, in (2) the $\text{Gen}$ function computes the proposals that $A_{Qi}$ thinks that are solutions for $A_{Qj}$, i.e. using as parameters his beliefs about the other agent’s resources ($B_i R_j$), his own resources ($R_i$) and his belief about the other agent’s goal ($B_i G_j$). In summary, Prop. 4 shows that the possible proposals that can be generated via an implementation of $\text{Gen}$ are potential solution for the negotiation problem between the agents involved.
4.3 Argument Interpretation

When an agent receive a proposal, an argument interpretation mechanism must be invoked in order to update the agent’s mental state accordingly. In our framework, the proposal interpretation is based on the following intuition: since agents are truthful, benevolent and aware of their own resources, when an agent $A_{ij}$ receives a proposal $\text{prop} = \langle (R_{\text{own}}, G), (R_{\text{get}}, R_{\text{give}}) \rangle$ from $A_{ij}$ then, $A_{ij}$ can infer the following information:

1. If $A_{ij}$ asks for $R_{\text{get}}$ then $A_{ij}$ believes $A_{ij}$ do not has $R_{\text{get}}$ as resource.
2. If $A_{ij}$ uses $R_{\text{own}}$ then $A_{ij}$ believes $A_{ij}$ has as resource $R_{\text{own}}$.
3. If $A_{ij}$ offers $R_{\text{give}}$ then $A_{ij}$ believes $A_{ij}$ has as resource $R_{\text{give}}$.
4. If $A_{ij}$ wants to reach $G$ then $A_{ij}$ believes $A_{ij}$ has $G$ as Goal.

Thus, $A_{ij}$ can change his believes accordingly, contracting his belief set as in (1) or revising it as in (2-4). In this way the computation of the beliefs set $B_{ij}$ may be closer to $S_j$ and consequently, the resulting set of possible deals $B_iD$ may be closer to $\mathcal{D}$ as well (illustrated in Fig. 3).

The agents will change their beliefs accordingly to the intuitions presented before, using belief revision operations. Lets $\text{contract}$ and $\text{revise}$ be implementations of the operators $\approx$ and $*$ respectively (see Def. 10 and Def. 11), and $\text{prop} = \langle (R_{\text{own}}, G), (R_{\text{get}}, R_{\text{give}}) \rangle$ an $A_i$ proposal received by $A_{ij}$. The following steps, which can be seen as variable assignments, implement the agent’s interpretation process:

1. $B_{j}R_{i} \leftarrow \text{contract}(B_{j}R_{i}, R_{\text{get}})$
2. $B_{j}R_{i} \leftarrow \text{revise}(B_{j}R_{i}, R_{\text{own}})$
3. $B_{j}R_{i} \leftarrow \text{revise}(B_{j}R_{i}, R_{\text{give}})$
4. $B_{j}G_{i} \leftarrow \text{revise}(B_{j}G_{i}, G)$

4.4 The Agent Decision Model: High-level algorithms

The agent’s decision making apparatus has been defined in Section 2 and has been implemented by using two algorithms $\text{Init}$ and $\text{Answer}$. The algorithm $\text{Init}$ is in charge of starting the negotiation. In a first place, it selects a proposal that the agent $A_{ij}$ believes is a deal $(B_i, D)$ that has not been proposed before. If such proposal does not exist, it tries to send a proposal associated with his own solutions $(S_i)$. If it fails, the agent sends a withdraw message. On its turn, $\text{Answer}$ receives the proposal generated from $\text{Init}$ and checks if it is an associated solution to the agents problem, and in that the proposal is accepted. If that is not the case, the agent’s beliefs are revised and $\text{Init}$ is called to generate a new proposal. High-level algorithms for $\text{Init}$ and $\text{Answer}$ are given next.

Algorithm 1: In line 1, the function $\text{Gen}_i$ (i.e. a suitable implementation of the $\text{Gen}$ function specified in Def. 14) is used to compute the set of proposals $\text{prop}S_i$ such that their associated solutions belong to $S_i$ (see Prop 4). Similarly, in line 2, $\text{Gen}_j$ is used to compute the set of proposals $\text{prop}B_iS_j$ that
Algorithm 1: *Init*$_i$
**Input:** $MS_i$
**Ensure:** Proposal
1: $propS_i \leftarrow \text{Gen}_i(R_i, B_i, R_j, G_i)$
2: $propS_j \leftarrow \text{Gen}_i(B_i, R_j, R_i, B, G_j)$
3: $propB_i \leftarrow propS_i \oplus propS_j$
4: $propSet \leftarrow propB_i \ominus \text{sent}(H)$
5: if $propSet \neq \emptyset$ then
6: $prop \leftarrow \text{select}(propSet)$
7: add($H, prop$)
8: return $prop$
9: else
10: $propSet \leftarrow propS_i \ominus \text{sent}(H)$
11: if $propSet \neq \emptyset$ then
12: $prop \leftarrow \text{select}(propSet)$
13: add($H, prop$)
14: return $prop$
15: else
16: return withdraw
17: end if
18: end if

Algorithm 2: *Answer*$_i$
**Input:** $MS_i, Proposals$
**Ensure:** $MS_i, Proposal$
1: $add(H, prop)$
2: $propS_i \leftarrow \text{Gen}_i(R_i, B_i, R_j, G_i)$
3: if $\ominus(prop) \in \ominus(propS_i)$ then
4: return accept
5: else
6: $B_i, R_j \leftarrow \text{contract}(B_i, R_j, R_{get})$
7: $B_i, R_j \leftarrow \text{revise}(B_i, R_j, R_{own})$
8: $B_i, R_j \leftarrow \text{revise}(B_i, R_j, R_{give})$
9: $B_i, G_j \leftarrow \text{revise}(B_i, G_j, G)$
10: $prop \leftarrow \text{Init}_i(MS_i)$
11: return $prop$
12: end if
the agent believes their associated solutions belong to $B_iS_j$ (see Prop 4). In line 3, the set $propS_iD$ is computed as those proposals in $propS_i$ such that their associated solutions are potential deals (see Prop 1). In line 4, those proposals that have been offered before are discarded. The select function chooses one proposal out of the set $propSet$ of possible candidate proposals.

Finally, the selected $prop$ is added to $H$.

**Algorithm 2**: In lines 1-2, the $H$ is updated, and the set $propS_i$ is computed. In line 3, we check if the solution associated with the received proposal is a solution for $Ag_i$. For this purpose, we use $\odot$ to denote the associated solution of a given proposal and $\bigodot$ to denote the set of associated solutions of a set of proposals. Then, in lines 6 to 9, the agent updates his mental state following the steps presented in Section 4.

The proposed argumentation-based negotiation framework for two agents have been implemented using logic programming following the algorithms presented above. Then, concrete negotiating agent’s may be specified by instantiating their mental state and setting the selection function in charge to choose the proposal to negotiate. For this prototype the selection function chooses a random element may be easily change to represent different agent’s criteria. In the next section we show how it works in a case study.

5 **The Home Improvement Agents Problem revisited**

As already mentioned in Section 1, we consider a slightly modified version of the hammer-nail-mirror example [11] as a case study for our approach. Additionally to the agents’ beliefs in the original example we consider the following beliefs: $Ag_1$ believes that $Ag_2$ has a nail and that his goal is to have a screw and $Ag_2$ believes that $Ag_1$ has a nail. Therefore, $Ag_1$ has the following intial mental state:

\[
R_1 = \{ \text{screw, screwdriver, hammer, hammer} \land \text{nail} \rightarrow \text{hangPicture}, \text{screw} \land \text{screwdriver} \rightarrow \text{hangMirror} \}
\]

\[
G_1 = \{ \text{hangPicture} \}
\]

\[
B_1R_2 = \{ \text{nail} \}
\]

\[
B_1G_2 = \{ \text{screw} \}
\]

\[
H_1 = [ ]
\]

and $Ag_2$ has as initial mental state:

\[
R_2 = \{ \text{nail, hammer} \land \text{nail} \rightarrow \text{hangMirror} \}
\]

\[
G_2 = \{ \text{hangMirror} \}
\]

\[
B_2R_1 = \{ \text{nail} \}
\]

\[
B_2G_1 = \{ \}
\]

\[
H_2 = [ ]
\]

\footnote{We abstract away this selection function, which could be defined according to some valuation criterion (e.g. cost).}
Suppose that $Ag_1$ is the agent that starts the negotiation. Next we summarize the main steps in the first two moves in the negotiation process:

1. $Ag_1$ uses the algorithm $Init_1$ to compute the first proposal. The functions $\text{Gen}_1(R_1, \{\text{nail}\}, \{\text{hangPicture}\})$ and $\text{Gen}_1(\{\text{nail}\}, R_1, \{\text{screw}\})$ are computed, obtaining as a result:

$$\text{prop}_S = \{(\emptyset, \{\text{hangPicture}\}), (\{\text{hangPicture}\}, R_1), (\{\text{hammer}, \text{nail} \land \text{hammer} \rightarrow \text{hangPicture}\}, \{\text{hangPicture}\}), (\{\text{nail}\}, \emptyset), \ldots\}$$

Now $Ag_1$ can compute the potential deals from the set of his proposals (i.e. $\text{prop} \in \text{prop}_S$) considering those he believes are solutions for $Ag_2$ (i.e. $\circ (\text{prop}) \in \bigodot (\text{prop}_B \cap \text{prop}_S)$):

$$\text{prop}_B = \{(\emptyset, \{\text{screw}\}), (\{\text{screw}\}, \emptyset), (\emptyset, \{\text{screw}\})\}$$

Since this is the first move, $H_1$ is empty and thus $\text{prop}_S = \text{prop}_B$. Then, the $\text{select}$ function chooses the second proposal, adding it to $H_1$ and $Ag_1$ is ready to start the negotiation with the following proposal:

**I propose that you provide me nail, because if I use hammer and nail $\land$ hammer $\rightarrow$ hangPicture, then I can achieve hangPicture in exchange for screw.**

2. $Ag_2$ receives $Ag_1$ proposal, and invokes the $Answer_2$ algorithm. $Ag_2$ adds the proposal to $H_2$ and then, uses the $\text{Gen}_2$ function to compute $\text{prop}_S$.

$$\text{prop}_S = \text{Gen}_2(R_2, \emptyset, \{\text{hangMirror}\})$$

$$= \{(\{\text{hammer}\}, \emptyset, (R_2, \{\text{hangMirror}\}))\}$$

Since $\circ (\text{prop}) \notin \bigodot (\text{prop}_S)$ (i.e. $\{\text{screw}\}, \{\text{nail}\} \notin \{(\{\text{hammer}\}, \emptyset)\}$) $Ag_2$, do not accept uses the proposal information to update his beliefs, and his $Init_2$ function to generate a proposal to answer $Ag_1$. The current mental state of $Ag_2$ is now as follows:

$$\begin{align*}
R_2 &= \{\text{nail}, \text{hammer} \land \text{nail} \rightarrow \text{hangMirror}\} \\
G_2 &= \{\text{hangMirror}\} \\
B_2R_1 &= \{\text{screw}, \text{hammer}, \text{nail} \land \text{hammer} \rightarrow \text{hangPicture}\} \\
B_2G_1 &= \{\text{hangPicture}\} \\
H_2 &= \{((\{\text{hammer}, \text{nail} \land \text{hammer} \rightarrow \text{hangPicture}\}, \{\text{hangPicture}\}, \\
&\{\text{screw}\}), \{\text{nail}\})\}
\end{align*}$$
Notice that $Aq_2$ through his interpretation process, has revised his previous beliefs about $Ag_1$’s goal and now, he believes that his goal is to $hangPicture$.

The whole dialogue obtained in the negotiation program for this scenario is the following:

1 Says: I propose that you provide me [nail] because
   if I use [hammer, nail & hammer => hangPicture] then
   I can achieve [hangPicture] in exchange for [screw]
2 Says: I propose that you provide me [hangMirror] because
   if I use [] then
   I can achieve [hangMirror] in exchange for [nail]
1 Says: I propose that you provide me [nail] because
   if I use [hammer, nail & hammer => hangPicture] then
   I can achieve [hangPicture] in exchange for
   [screw, screwDriver, screwDriver & screw => hangMirror]
2 Says: Accept, I give you [nail] and you give me
   [screw, screwDriver, screwDriver & screw => hangMirror]

6 Discussion and Related Work

In this paper we have presented a novel approach to automated negotiation between two argumentative agents. An intentional architecture was given to each agent as to represent not only his own resources and goals but also, his beliefs on the other agent’s resources and goals. In our approach, the interpretation and generation of arguments are based on belief revision operators. In order to achieve their goals, agents engage in a benevolent dialogue, exchanging information that supports which resources they are willing to exchange. This results in a negotiation dialogue directed by the $Init$ and $Answer$ algorithms, principal components of the agents’ decision making apparatus. During the negotiation, the agents continuously update their mental states to generate proposals more likely to be accepted.

All the propositions presented in our approach were formalized in Coq and a prototype of this framework for two negotiating agents, with their associated algorithms were implemented in logic programming. Using this prototype, a revised version of the HNM was solved, showing how the agents can negotiate to solve this kind of cooperative problems, starting with incomplete (or wrong) information about the other agent’s beliefs.

In contrast with the original argumentative framework to solve the HNM problem in [11], our negotiation model allows the agents to gain and revise their beliefs as the dialogue takes place. Consequently, in our approach an agent does not need to have initial (or correct) beliefs about the other agent involved in the negotiation. In [12] a similar scenario is analyzed, but agents are aware of all the agents’ resources and the agents’ plans (or their knowledge about plans) are not consider negotiable. We think that our proposal is more flexible in this respect, as plans are also negotiation objects in our formalization.
We note that there have been previous approaches integrating belief revision and negotiation. In [19] a formal characterization of negotiation from a belief revision perspective is given, but no implementation issues are considered.

Tohmé [17] views negotiation as resource allocation with uncertainty caused by imperfect information about others. The agents exchange messages that correspond to updates of beliefs and consequently cause new messages to be generated. Tohmé showed that under certain conditions, beliefs converge in the long run, leading to successful negotiation (i.e. agreement).

In our proposal we assume that agents are benevolent, if agents do not need a resource, they should give it away when asked. This approach can also be found in several frameworks as for example: [11], [3]. In addition, in our work, agents are assumed to be truthful. Recent research has led to consider other situations such as negotiation among dishonest agents [15]), which is an interest scenario for future work.

Another relevant approach to argumentation-based negotiation can be seen in [2] where the proposed framework makes it possible to study the outcomes of a negotiation process. In contrast with this approach, our proposal rely on the characterization of belief revision operations to model the agent’s arguments generation, which their claims are the resources to be exchanged.

Formal models of belief change can be very helpful in providing suitable frameworks for rational agents [5], in which the information from interagent dialogues can be better exploited. In [16], Son et al. develop a generic model for negotiation in dynamic environments and apply it to generate joint-plans with negotiation for multiple agents, proposing a general scheme for one-to-one negotiations. Eventhough in our approach we do not deal with agent planning in an explicit way, some notion of plans are involved in the constructive implication we use to represent the agent’s know-how to reach a goal. These plans are considered part of the agent’s resources and are also negotiated. The model of negotiation proposed (based on answer-set planning) is instantiated to deal with dynamic knowledge of planning agents. In contrast, our approach is based on a high-level algorithm which relies on belief revision operators for analyzing candidate proposal and resources exchanges, considering that the agent’s resources do not change during the negotiation.

Part of our future work is focused on studying complexity issues related with our proposal, as done by Zhang in the context of belief-revision based bargaining and negotiation [18]. Furthermore, we are interested in extend our approach to a multiagent platform considering also the possibility that different agents may have different languages following the proposal of Son et al. [16]. We are also investigating the logical properties of our approach, particularly those concerning the characterization of different incision and selection functions. In this setting, we think that the role of argumentation and belief revision in the context of agent dialogues in a negotiation process (as considered in [6, 10]) deserves still further study.
Acknowledgements

This research work was partially supported by the Research Project PIP Conicet 112-200801-02798 (Conicet, Argentina), the LACCIR Project R1211LAC004, funded by Microsoft Research, CONACyT (Mexico) and Interamerican Development Bank (IDB), and the Projects Pict 2009-0015 and PID-UNR ING 308.

References

