Argumentative Alternating Offers

Nabila Hadidi
LIPADE
Paris Descartes University
45 rue des Saints-Pères
75270 Paris 06, France
hadidi@mi.parisdescartes.fr

Yannis Dimopoulos
Dept. of Computer Science
University of Cyprus
75 Kallipoleos Str.
Nicosia, Cyprus
yannis@cs.ucy.ac.cy

Pavlos Moraitis
LIPADE
Paris Descartes University
45 rue des Saints-Pères
75270 Paris 06, France
pavlos@mi.parisdescartes.fr

ABSTRACT

This paper presents an argumentative version of the well known alternating offers negotiation protocol. The negotiation mechanism is based on an abstract preference based argumentation framework where both epistemic and practical arguments are taken into consideration in order to decide about different strategic issues. Such issues are the offer that is proposed at each round, acceptance or refusal of an offer, concession or withdrawal from the negotiation. The argumentation framework shows clearly how offers are linked to practical arguments that support them, as well as how the latter are influenced by epistemic arguments. Moreover it illustrates how agents' argumentative theories evolution, due to the exchange of arguments, influences the negotiation outcome. Finally, a generic algorithm that implements a concession based negotiation strategy is presented.

Categories and Subject Descriptors

I.2.11 [**Computing Methodologies**]: Distributed Artificial Intelligence—*Multiagent systems*

General Terms

Theory

Keywords

Argumentation, Negotiation

1. INTRODUCTION

Negotiation is the process of looking for an agreement between two or several agents on one or more issues. There exist three main approaches to negotiation, namely game theory (see e.g. [11]), heuristics (see e.g. [6]) and argumentation (see e.g. [10],[13]).

In the last years there is a plethora of works on argumentation based negotiation (see e.g. [1], [7], [2], [8]), testifying the increasing importance that is attached to the role of argumentation in negotiation. Although a precise and formal account of the added value of argumentation in negotiation is still missing, it is at least clear that exchanging arguments revealing (at least some of) the reasons for which a

negotiator is proposing an offer may release several blocked situations. Such an example is a situation where the conflict is due to different perceptions of the world, which may have further repercussions on the behavior of a negotiator, including even parameters like his own preferences. Indeed, arguments received by the opponent on some issue might provide a piece of missing information to the proponent who could suddenly discover that the proposed offer is not optimal for himself, or that there is an objective constraint that forbids his opponent to accept his offer.

It is, therefore, evident that trying to "influence", in one way or another, the opponent's beliefs about the world may be a meaningful way to defend or attack an offer. This situation can be handled through the simultaneous consideration of both *practical* and *epistemic* arguments in the reasoning process and by deciding in which situation each type of argument must prevail. This may be part of the strategy of the agent. We remind that practical arguments support offers while epistemic arguments represent what the agent believes about the world.

The above intuitions define the perspective that is taken in this paper. To capture these intuition, we propose an original adaptation of the well known *alternating offers protocol* [12]in the argumentation context. Then, we adapt a reasoning mechanism combining practical and epistemic arguments proposed in [3] in the negotiation context and we exploit the possibilities that our *argumentative alternating offers protocol* provides for alternating practical and epistemic arguments depending on the evolution of the negotiation. Finally, we present a generic algorithm, which, building on the above reasoning mechanism and the possibilities that the argumentative alternating offers protocol provides, implements a parameterized concession based negotiation strategy. The algorithm is generic in the sense it can operate regardless of whether there is a time constraint or not (which is the case in this paper), or of the tactics (or heuristics) the agents might use in several situations where a choice has to be made (e.g. accept or reject an offer, choose the best offer to propose). Thus, it can be parameterized to capture the previous issues without further modification.

To the best of our knowledge, it is the first time that the way that epistemic arguments interfere with practical arguments in a negotiation process is presented along with a generic algorithm that incorporates this mechanism in the service of strategic considerations. This seriously differentiates our work from other important works in the domain such as [2], [7], [1], [9], etc. A similar combination of epistemic and practical arguments is proposed in [4] but in a

deliberation dialogue.

2. NEGOTIATION FRAMEWORK

The negotiation framework we propose is based on the abstract preference-based argumentation framework of [2].

We assume two agents, ag_i and ag_j, $i \neq j$, that are involved in a bilateral negotiation over a set of offers (options) $O = \{o_1, o_2, ..., o_n\}$ which are identified from a logical language $\mathcal{L}$. We further assume that there is an option $o_D \in O$ that represents disagreement. The options are mutually exclusive, which means that each agent can choose only one of them at once.

2.1 Arguments

From the language $\mathcal{L}$ a set of arguments $Args(\mathcal{L})$ are constructed. By argument we mean a *reason* for *believing* or *doing something*. We assume that an agent is aware of all the arguments of the set $Args(\mathcal{L})$. It encodes the fact that when an agent receives an argument from another agent, it can interpret it correctly and it can also compare it with its own arguments.

2.1.1 Types of arguments

Unlike [2], we distinguish between *epistemic* and *practical* arguments, that are both taken into account, as in [3], in the reasoning mechanism used by the agents. Thus, we have:

1. *Practical arguments* A_p support offers (or decisions) by trying to justify those offers.

2. *Epistemic arguments* A_e represent what the agent believes about the world

In what follows, we are not interested in the construction of these arguments. We make the following assumptions:

- $Args(\mathcal{L}){=}A_e \cup A_p$,

- $A_e \cap A_p = \emptyset$,

- Arguments structure is unknown.

Epistemic arguments are denoted by variables $\alpha_1, \alpha_2,...$, while practical arguments by variables $\delta_1, \delta_2,...$ When no distinction is necessary between arguments, we use variables $a, b, c...$

Let F be a function that maps each option to the arguments that support it, i.e., $\forall o \in O$, $F(o) \subseteq A_p$. Each argument can support only one option, thus $\forall o_y, o_z \in O, o_y \neq o_z, F(o_y) \cap F(o_z) = \emptyset$. When $\delta \in F(o)$, we say that o is the conclusion of δ, noted Conc(δ)=o.

2.1.2 Comparison between arguments

As in [3], we assume three binary preference relations on arguments.

- $\succeq_e$: Partial preorder on the set A_e,

- $\succeq_p$: Partial preorder on the set A_p,

- $\succeq_m$: defined on the sets A_e and A_p, such that $\forall \alpha \in A_e$, $\forall \delta \in A_p$, $(\alpha, \delta) \in \succeq_m$ and $(\delta, \alpha) \notin \succeq_m$. That means that any epistemic argument is stronger (preferred) than any practical argument (m stands for mixed relation).

In what follows $\succ_x$ with $x \in \{e, p, m\}$ denotes the strict relation associated with $\succeq_x$. It is defined as $(a, b) \in \succ_x$ iff $(a, b) \in \succeq_x$ and $(b, a) \notin \succeq_x$. Moreover when $(a, b) \in \succeq_x$ and $(b, a) \in \succeq_x$ we will say that the arguments a and b are *indifferent*, denoted by $a \sim b$.

2.1.3 Conflict between arguments

Conflicts between arguments in $\mathcal{A} = A_p \cup A_e$ are captured by the binary relation R ([3]).

- R_e: Represents the conflicts between arguments in A_e.

- R_p: Represents the conflict between practical arguments, such that $R_p = \{(\delta, \delta') \mid \delta, \delta' \in A_p, \delta \neq \delta'$ and $Conc(\delta) \neq Conc(\delta')\}$. This relation is symmetric.

- R_m: Represents the conflicts between epistemic and practical arguments s.t. $(\alpha, \delta) \in R_m$, $\alpha \in A_e$ and $\delta \in A_p$.

Thus we have $R = R_e \cup R_p \cup R_m$.

We assume that practical arguments supporting different offers are in conflict. Thus for any two offers o_y, o_z, $\forall a \in F(o_y)$ and $\forall a' \in F(o_z)$, it holds that $(a, a') \in R_p$ and $(a', a) \in R_p$.

2.1.4 Attacks between arguments (Defeat)

Each preference relation $\succeq_x$ (with $x \in \{e, p, m\}$) is combined with the relation of conflict R_x (with $x \in \{e, p, m\}$), to give a *defeat relation* between arguments, noted Def_x (with $x \in \{e, p, m\}$).

Definition 1. (Defeat) Let $A \subseteq Args(\mathcal{L})$ be a set of arguments and $a, b \in A$. Then $(a, b) \in Def_x$ iff $(a, b) \in R_x$, and $(b, a) \notin \succ_x$.

We have $Def_{global} = Def_e \cup Def_p \cup Def_m$. In the following sections we will need two particular notions of defeat namely *rebuttal* and *undercutting*. For explaining those notions we will consider here a particular structure of arguments based on a propositional language $\mathcal{L}'$ although our negotiation framework is independent of the structure of the arguments. $\vdash$ stands for classical inference and $\equiv$ for logical equivalence.

Definition 2. (Argument Structure) An argument is a pair $a = (S, q)$ where q is a formula in $\mathcal{L}'$ and S a set of formulae in $\mathcal{L}'$ s.t.

- S is consistent

- $S \vdash q$

- S is a minimal set of propositions that satisfies the two previous conditions

Here S is called the support of the argument a and it is written $S = Support(a)$ and q its conclusion and it is written $q = Conclusion(a)$.

Definition 3. (Undercutting) Let a and b be two arguments. Argument a *undercuts* b iff $\exists p \in Support(b)$ s.t. $p \equiv \neg Conclusion(a)$.

Definition 4. (Rebuttal) Let a and b be two arguments. Argument a *rebuts* b iff $Conclusion(a) \equiv \neg Conclusion(b)$.

In the context of a negotiation, *practical* arguments *rebut* practical arguments, *epistemic* arguments *undercut* practical arguments, whereas *epistemic* arguments can both *undercut* and *rebut* other epistemic arguments. Recall that practical arguments cannot attack epistemic arguments.

2.2 Extensions of arguments

In [5], different acceptability semantics have been introduced for computing the status of arguments. These are based on two basic concepts, *defence* and *conflict-freeness*, defined as follows:

Definition 5. (Defence/Conflict-free) Let $\mathcal{T} =< A, Def >$ be an argumentation system with $A \subseteq Args(\mathcal{L})$. Let $A' \subseteq A$.

- A' is conflict free iff $\nexists\ a, b \in A'$ s.t $(a, b) \in Def$.

- A' defends $a \in A$ iff $\forall\ b \in A$, if$(b, a) \in Def$, then $\exists\ c \in A'$ s.t $(c, b) \in Def$.

Definition 6. (Acceptability semantics) Let $\mathcal{T} =< A, Def >$ be an argumentation system with $A \subseteq Args(\mathcal{L})$ and A' a conflict free set of arguments.

- A' is an admissible extension iff A' defends any element in A'.

- A' is a preferred extension iff A' is a maximal(w.r.t set $\subseteq$) admissible set.

- A' is a stable extension iff it is a preferred extension that defeats any argument in $A \backslash A'$.

Definition 7. (Argument status) Let $\mathcal{T} =< A, Def >$ be an argumentation system with $A \subseteq Args(\mathcal{L})$ and $E_1, E_2, ..., E_n$ its extensions under a given semantics. Let an argument $a \in A$.

- a is skeptically accepted iff $\forall E_q$, $1 \leq q \leq n$, $a \in E_q$.

- a is credulously accepted iff $\exists E_q$, $1 \leq q \leq n$, s.t $a \in E_q$ and $\exists E_w$, $1 \leq w \leq n$, s.t $a \notin E_w$.

- a is rejected iff $\nexists E_q$, $1 \leq q \leq n$, such that $a \in E_q$.

2.3 Negotiating Agents Theories

As in [2], we assume that each agent involved in a negotiation has a negotiation theory that contains arguments $\mathcal{A}$ that can be exchanged during the negotiation. However, in our work we distinguish two types of arguments, i.e $\mathcal{A} = A_p \cup A_e$. This, as it will become evident in the following, has several effects on the reasoning process of the agents and consequently the negotiation process. Formally, a negotiation theory is defined as follows.

Definition 8. (Negotiation theory) Let O be a set of options, $ag \in Ag$ an agent and Ag the set of negotiating agents. The negotiation theory $\mathcal{T}^{ag}$ of agent ag is a tuple $\mathcal{T}^{ag} =< \mathcal{A}^{ag}, F^{ag}, Def_{global}^{ag} >$ where $Def_{global}^{ag} = Def_e \cup Def_p \cup Def_m$ and $\mathcal{A}^{ag} = A_p^{ag} \cup A_e^{ag}$ such that:

- $\mathcal{A}^{ag} \subseteq Args(\mathcal{L})$. This set represents all the arguments that the agent can built from his beliefs and all the arguments that support each option in O.

- $F^{ag} : O \rightarrow 2^{A_p^{ag}}$ associates practical arguments to offers. We have $\bigcup_{1 \leq y \leq n} F^{ag}(o_y) = A_p^{ag}$.

- $Def_{global}^{ag} \subseteq \mathcal{A}^{ag} \times \mathcal{A}^{ag}$

2.4 Offer status and preferences between offers

In [3], five statuses are defined for the options/offers. In this work, we use only two of them. A *skeptical offer* which has a supporting argument that is skeptically accepted, and a *credulous offer* which has a supporting argument that is credulously accepted.

The *Effective Supporting Arguments* of an offer, defined formally below, are all arguments, either *skeptically* or *credulously* accepted, that support the offer.

Definition 9. (Effective Supporting Arguments-ESA) Let O be a set of offers, $E_1, ..., E_n$ the extensions under a given semantics of the theory $\mathcal{T} =< \mathcal{A}, F, Def_{global} >$ and $o_y \in O$ an offer. Then the set of *effective supporting arguments* for offer o_y is $ESA(o_y) = \{a | a \in F(o_y)$ and $a \in E_1 \cup ... \cup E_n\}$.

In simple words, $ESA(o_y)$ is the set of arguments that support o_y and are included in at least one extension. The cardinalities of the ESA of the offers can be used to define a *preference relation* on these offers.

Definition 10. Let O be a set of offers, $\mathcal{T} =< \mathcal{A}, F, Def_{global} >$ a negotiation theory, and $o_x, o_y \in O$. Then $o_x \trianglerighteq o_y$ iff $\forall a \in ESA(o_x)$ and $\forall b \in ESA(o_y)$ it holds that $a \sim b$ (i.e. they are indifferent) and $| ESA(o_x) | \geq | ESA(o_y) |$.

Therefore, $\trianglerighteq$ favors options that are supported by more arguments. Although this is a simple preference relation, and possibly more sophisticated methods for ranking offers exist, it suffices for the purposes of this work.

3. ARGUMENTATION-BASED ALTERNATING OFFERS PROTOCOL

In [12], Rubinstein introduced the *Alternating Offers protocol* for bargaining between agents. It is a bilateral protocol between the *proposer* who initiates the process, and the *responder* who replies to the proposal. The proposer starts the negotiation process by presenting a *proposal* using a SUBMITPROPOSAL message. The responder can *accept* or *reject* the offer in its entirety by sending an ACCEPT or REJECT message as a reply. The responder can also propose a counter-offer by sending the COUNTER reply accompanied by the counter proposal. In this case, the proposer has the same options and therefore can accept, reject or reply with a counter proposal of its own. If one of the agents is satisfied with the current iteration of the proposal, he can send an ACCEPT message to the other. He can also signal his dissatisfaction and abort the negotiation session by sending a REJECT message. To seal the agreement, the other agent has to send a CONFIRM message and receive a CONFIRM-ACCEPTANCE message in reply.

The protocol, as described above, is generic, with no time limits and no central coordinator to manage the negotiations, and either of the parties can leave the process at any time.

In this work we adapt the classical alternating offers protocol to the case of argumentation-based negotiation. To do so we extend the concept of *round* used in the classical protocol to include, besides the classical *propose*, *accept* or *reject* messages, the possibility to *argue* in order to defend or attack an offer. In addition, *propose* and *argue* are accompanied by *supporting (practical or epistemic) arguments*.

3.1 Moves

Arguments and offers are conveyed through *dialogue moves* (or simply *moves*). A move is denoted by $m_{r,g}$, whereas $r \geq 1$ identifies the round (and therefore the offer which is currently discussed), and $g \geq 1$ the number (order) of the move in that round. In the argumentative alternating offers protocol the following moves are used. In all moves ag_i and ag_j are the participating agents and $o_y \in O$.

- $Propose(ag_i, ag_j, o_y, \delta)$, where $\delta \in F^{ag_i}(o_y)$. This move allows agent ag_i to propose an offer o_y to agent ag_j, along with a practical argument δ that supports it.

- $Argue(ag_i, ag_j, a, Target)$, where $a \in A^{ag_i}$ and $Target$ is the move the argument of which is attacked by a or nil. This move allows agent ag_i to argue by defending his own offer o_y or to counter-attack an offer sent by ag_j. The arguments used in this move satisfy the following conditions

  - If $Target =$ nil then $a \in F^{ag_i}(o_y)$, i.e., a is a practical argument that support the offer o_y.

  - If $Target \neq$ nil then $a \in A_e^{ag_i}$ is an argument presented against the argument of $Target$. Thus, an agent can't present an argument against his own arguments.

- $Reject(ag_i, ag_j, o_y)$. This move is sent by ag_i to inform ag_j that he has no arguments to present and he does not accept ag_j's offer.

- $Nothing(ag_i, ag_j)$. This move notifies ag_j that ag_i has no arguments to present and he either still considers his offer as a most preferred one for him (when he is the proposer), or believes that he has better options that the current offer (when he is the recipient of an offer sent by the other agent).

- $Accept(ag_i, ag_j, o_y)$. This move is used by agent ag_i to notify that he accepts the offer o_y made by ag_j.

- $Agree(ag_i, ag_j)$. This move means that ag_i now believes that his current offer is not optimal for himself and therefore accepts the arguments sent by ag_j. Agent ag_j starts a new round.

- $Withdraw(ag_i, ag_j)$. This move indicates that agent ag_i withdraws from negotiation.

- $final(ag_i, ag_j)$. This is a shorthand for $Propose(ag_i, ag_j, o_y, \emptyset)$ and is used during a final round of the negotiation. Its use and semantics will become apparent in the following.

The following functions retrieve the parameters of the moves.

- $Performative(m_{r,g})$ returns one of $Propose, Argue, Nothing, Reject, Accept, Withdraw, Agree$.

- $Agent(m_{r,g})$ returns the agent who sent the move.

- $Offer(m_{r,g})$ returns the offer sent in the round r.

- $Argument(m_{r,g})$ returns the argument sent to the other agent.

- $Targ(m_{r,g})$ returns the target of the move.

Finally, the following hold.

- If $Performative(m_{r,g})=Propose$ then $Argument(m_{r,g}) \in A_p^{ag_i}$ arguments

- If $Performative(m_{r,g})=Argue$ then $Argument(m_{r,g}) \in A_e^{ag_i} \cup A_p^{ag_i}$

3.2 Round

A round takes place in alternating way between two agents P (the proposer of the offer) and R (the recipient of the offer). The agent proposing an offer may send moves with performative from $\{Propose, Argue, Agree, Nothing, Withdraw\}$, whereas the agent that receives an offer may send moves with performative from $\{Argue, Reject, Accept, Nothing, Withdraw\}$. A round is defined formally as follows.

Definition 11. (Round) A round r between two agents P and R is a non empty sequence of moves $m_{r,1}, ..., m_{r,n}$, such that:

- $\forall i, k, i \neq k, \forall g, g', g \neq g'$ $Offer(m_{i,g}) \neq Offer(m_{k,g'})$.

- $\forall r, Agent(m_{r,g}) = P$ if $Odd(g)$, and $Agent(m_{r,g}) = R$ if $Even(g)$.

- $\forall m_{r,g}$, if $Odd(g)$ then $Performative(m_{r,g}) \in \{Propose, Argue, Agree, Nothing, Withdraw\}$.

- $\forall m_{r,g}$, if $Even(g)$ then $Performative(m_{r,g}) \in \{Argue, Reject, Accept, Nothing, Withdraw\}$.

- $\forall r, Performative(m_{r,1}) \in \{Propose, Withdraw\}$.

- $\forall r$, if $Performative(m_{r,g}) = Performative(m_{r,g+1}) = Withdraw$ then the dialogue ends with a disagreement.

- $\forall m_{r,g}$, if $Performative(m_{r,g})=Argue$ then:

  - If $Targ(m_{r,g}) \neq$ nil then $Targ(m_{r,g})=m_{r,g'}$ with $g' < g$, $Argument(m_{r,g}) Def_{global}^{Agent(m_{r,g})} Argument(m_{r,g'})$ and $Agent(m_{r,g}) \neq Agent(m_{r,g'})$. Here the agent sends an argument which attacks one presented previously by the other agent in the same round.

  - Else $Agent(m_{r,g})=Agent(m_{r,1})$ and $Argument(m_{r,g}) \in F^{Agent(m_{r,g})}(Offer(m_{r,1}))$. Here the agent sends a new argument to support his offer.

- If $Performative(m_{r,n}) = Accept$ then $Offer(m_{r,1})$ is the outcome of the dialogue which terminates with agreement.

- If $Performative(m_{r,n}) \in \{Agree, Reject\}$ then a new round $r+1$ starts with $Agent(m_{r+1,1}) \neq Agent(m_{r,1})$ i.e. with the other agent as proposer.

- $\forall m_{r,g}$, if $Performative(m_{r,g})=Nothing$ then $Argument(m_{r,g}) = \emptyset$ and $Offer(m_{r,1}) = \emptyset$.

- $\forall m_{r,1}, m_{r,g'}, g' > 1$ if $Offer(m_{r,1})=Offer(m_{r,g'})$ then $Agent(m_{r,1})= Agent(m_{r,g'})$ and $Argument(m_{r,1}) \neq Argument(m_{r,g'})$. In our protocol, unlike [2], an agent can propose the same offer more than once during a round provided that he supports it with an argument not used before.

Definition 12. (Argumentative alternating offers dialogue) An argumentative alternating offers dialogue d between two agents P, R is a non-empty sequence of rounds $d=\{r_1...r_\lambda\}$ between P and R.

In the alternating offers protocol ([12]), two outcomes are possible: (a) no agreement (disagreement), or (b) an agreement in some round. In the argumentative protocol the situation is similar.

Definition 13. (Outcome) Let $d=\{r_1, ..., r_n\} \in D$ be an argumentative alternating offers dialogue where D is the set of all the dialogues built from the argumentative alternating offers protocol and $r_n = \{m_{r_n,1}, ..., m_{r_n,k}\}$ be the last round of d. If $Performative(m_{r_n,k})=Accept$ then $Outcome(d) = Offer(m_{r_n,1})$ (Agreement). Else $Outcome(d) = nil$ (Disagreement).

4. NEGOTIATION STRATEGY

In this section, we present a strategy that can be used by the two agents involved in an argumentative alternating offers negotiation. The strategy is based on the theory of the agent T, his preference on the set of offers $\unrhd$, and the alternating offers protocol as defined in the previous section.

In order to improve presentation, some of the parameters of the messages of the negotiation dialogue are omitted from the algorithms that follow. These are mainly agent and target move names, and are easily derivable from context.

The main procedure of the strategy is described by procedure $negotiate(T, O, outcome)$, depicted in Algorithm 3. It accepts as parameters the agent theory T, and the set of possible offers O, and returns an *outcome* that can be either an offer, when an agreement is reached, or *nil* when the negotiation fails. As noted before, the set O contains an option o_D representing the possibility that the agent leaves the negotiation without an agreement, and therefore remains in the same state that he was initially. Therefore, offers that lead to situations that are less desirable than his current state are less preferred by the agent. This option o_D corresponds to what in classical negotiation theory is referred to as *reservation value*.

Compute the extensions $E_1, E_2, \ldots, E_n$ of $T_{r,g}$;
Compute $O^{cand} = \{o | o \in O \text{ s.t. } \exists a \in \cup_{i=1}^{n} E_i \text{ and } a \in F(o)\}$;
Compute $O^{best} = \{o | o \in O^{cand}, o_D \not\unrhd o, \text{ and } \neg\exists o' \in O^{cand} \text{ s.t. } o' \rhd o\}$;
return O^{best};

Algorithm 1: Procedure compute-best$(T_{r,g}, O, O^{best})$

Call compute-best$(T_{r,g}, O, O^{best})$;
if $O^{best} = \emptyset$ **then**
  | $o = nil; a = nil;$
else
  | Select an offer o from O^{best} and $a \in F(o)$ such that
  | a belongs to some extension of $T_{r,g}$;
end
return o, a;

Algorithm 2: Procedure proposal$(T_{r,g}, O, o, a)$

$r = 1; g = 1; own = false; T_{1,1} = T;$
$Received = \emptyset; Offered = \emptyset; UsedAtt = \emptyset;$
if *Agent proposes first* **then**
  | Call $proposal(T_{1,1}, O, o^{cur}, a^{curr})$;
  | Send $Propose(o^{cur}, a^{cur}); own = true;$
end
while *true* **do**
  | $g = g + 1$; Get $m_{r,g}$;
  | Incorporate $argument(m_{r,g})$ into $T_{r,g}$;
  | **switch** $Performative(m_{r,g})$ **do**
    | **case** *Argue*
      | Add $argument(m_{r,g})$ to $Received$;
      | **if** *own* **then**
        | Call
        | defend$(T_{r,g}, O, o^{cur}, Received, UsedAtt)$;
      | **else**
        | Call
        | check$(T_{r,g}, O, o^{cur}, Received, UsedAtt)$;
      | **end**
    | **case** *Propose*
      | Add $argument(m_{r,g})$ to $Received$;
      | $o^{cur} = Offer(m_{r,g})$;
      | Add o^{cur} to $Offered$;
      | $r = r + 1; g = 1;$
      | Call check$(T_{r,g}, O, o^{cur}, Received, UsedAtt)$;
    | **case** *agree*
      | Call $proposal(T_{r,g}, O, o^{cur}, a^{curr})$;
      | **if** $o^{cur} = nil$ **then**
        | Send *withdraw*; $g = g + 1$;
      | **else**
        | Send $Propose(o^{cur}, a^{cur})$;
        | $Received = \emptyset; UsedAtt = \emptyset;$
        | $r = r + 1; g = 1;$
        | $o = o^{cur}; own = true;$
      | **end**
    | **case** *nothing*
      | Call nothing-
      | reply$(T_{r,g}, O, own, o^{cur}, Received, UsedAtt)$;
    | **case** *reject*
      | $O = O - \{o^{cur}\}; own = false;$
      | Remove from $T_{r,g}$ all arguments of $F(o^{cur})$
    | **case** *withdraw*
      | Call withdrawal$(T_{r,g}, O, Offered, outcome)$;
      | return *outcome* and exit;
    | **case** *accept*
      | $outcome = o^{cur}$;
      | return *outcome* and exit;
    | **case** *final*
      | $outcome = Offer(m_{r,g})$;
      | return *outcome* and exit;
  | **end**
end

Algorithm 3: Procedure negotiate$(T, O, outcome)$

One of the agents initiates the negotiation by sending a proposal via a *propose* message. This proposal is selected by procedure proposal$(T_{r,g}, O, o, a)$ (Algorithm 2). This selection at some round r and step g, is based on the current theory of the agent $T_{r,g}$, and the current set of offers O. The offer o that is proposed must be supported by some argument a that belongs to some of the extensions of $T_{r,g}$. Among several possible such offers, the best wrt $\unrhd$ is selected. Note that an agent never proposes, accepts or de-

fends an offer that is worse wrt $\unrhd$ than o_D, as any such deal is considered by the agent worse than no deal.

Call compute-best($T_{r,g}, O, O^{best}$);
if $o \in O^{best}$ **then**
  | Send *accept*;
else
  Compute $\mathcal{E}^{best} = \{E | E$ is an extension of $T_{r,g}$ s.t. $\exists a \in E$ and $a \in \mathcal{F}(o)$ for $o \in O^{best}\}$;
  if *there is* $a \in E$, $E \in \mathcal{E}^{best}$ *s.t.* a *is an epistemic argument that attacks some argument* $b \in R$, *and* $(a, b) \notin UA$ **then**
    | Send *argue(a)*;
    | $UA := UA \cup \{(a, b)\}$; return UA;
  else
    | Send *nothing*;
  end
end
$g = g + 1$;

Algorithm 4: Procedure check($T_{r,g}, O, o, R, UA$)

Call compute-best($T_{r,g}, O, O^{best}$);
if $o \notin O^{best}$ **then**
  | Send *agree*;
else
  Compute $\mathcal{E}^{best} = \{E | E$ is an extension of $T_{r,g}$ s.t. $\exists a \in E$ and $a \in \mathcal{F}(o)$ for $o \in O^{best}\}$;
  Compute $\mathcal{E}^o = \{E | E \in \mathcal{E}^{best}$ and $\exists a \in E$ s.t. $a \in F(o)\}$;
  if *there is* $a \in E$, $E \in \mathcal{E}^o$ *s.t.* a *is an epistemic argument that attacks some argument* $b \in R$, *and* $(a, b) \notin UA$ **then**
    | Send *argue(a)*;
    | $UA := UA \cup \{(a, b)\}$;
    | return UA;
  else if *there is* $a \in E$, $E \in \mathcal{E}^o$ *s.t.* a *is a practical argument that has not been used before* and $a \in F(o^{cur})$ **then**
    | Send *argue(a)*;
  else Send *nothing*;
  end
end
$g = g + 1$;

Algorithm 5: Procedure defend($T_{r,g}, O, o, R, UA$)

Upon receiving a proposal in a move $m_{r,g}$, the agent incorporates the supporting argument in his theory, adds the argument to the set *Received* of arguments that have been sent by the other agent, and runs procedure check($T_{r,g}, O, o$, *Received*, *UsedAtt*) (Algorithm 4). If the proposed offer is one of the best (wrt $\unrhd$), he accepts the offer and the negotiation terminates. Otherwise, he attempts to find an epistemic argument a that belongs to one of the extensions of $T_{r,g}$, and counterattacks the argument supporting the offer. Note that a must not have been used before to attack the supporting argument of the other agent in the same round. This avoids loops in argumentation, and is achieved by recording the counterattacks in *UsedAtt*. If he is successful, he sends argument a with an *argue* to the other agent.

If he is unsuccessful, he is confronted with a situation where on the one hand he can not counterattack the proposal, but on the other hand there are offers that are more desirable than the proposal. In such a case he sends a *nothing* message to the other agent.

Therefore, the reply to a proposal can be any of *accept*,

argue, or *nothing*. The first case is straightforward. Whenever an agent receives an *argue* during a round during which he is the proposer, he runs procedure defend($T_{r,g}, O, o$, *Received*, *UsedAtt*) (Algorithm 5). If in the light of the last argument sent by the other agent his proposal is not one of the most preferred for himself, he replies with *agree*. Otherwise, he tries to defend his proposal against the attack by attacking one of the arguments sent by his counter-party during the current round. If no such attack exists, another argument supporting his offer is sent in a *argue* message. If no such argument exists, a *nothing* message is sent. This signifies that the agent insists that his current offer is one of the most preferred for himself. Upon receiving *nothing* the other agent sends a *reject* message, and becomes the proposer in the new round. This task is carried out by the part of procedure nothing-reply($T_{r,g}, O, own, o$, *Received*, *UsedAtt*) (Algorithm 6), which runs when parameter *own* is false, meaning that the offer currently discussed has been proposed by the other agent.

---

if *own* **then**
  Compute $\mathcal{E}^o = \{E | E$ is an extension of $T_{r,g}$ and $\exists a \in E$ s.t. $a \in F(o)\}$;
  if *there is* $a \in E$, $E \in \mathcal{E}^o$, $a \in \mathcal{F}(o)$ and a *has not been used before* **then**
    | Send *argue(a)*;
  else
    | Send *nothing*;
  end
  $g = g + 1$;
else
  Send *reject*; $g = g + 1$;
  Call *Proposal*($T_{r,g}, O, o^{cur}, a^{curr}$);
  if $o^{cur} = nil$ **then**
    | Send *withdraw*; $g = g + 1$;
  else
    | Send *Propose*(o^{cur}, a^{cur});
    | $R = \emptyset$; $UA = \emptyset$;
    | $r = r + 1$; $g = 1$;
    | $o = o^{cur}$; $own = true$;
    | return o, own, R, UA ;
  end
end

Algorithm 6: Procedure nothing-reply($T_{r,g}, O, own$, o, R, UA)

If an agent receives a *nothing* message in a round where he is the proposer, he is in a situation where he can not defend the argument that supports his offer. Therefore, he needs to find some other argument to support it, which he sends in an *argue* message. If no such argument exists, a *nothing* message is sent.

If at some point one of the agent has no offers, he sends a *withdraw* message, signifying that he is willing to leave the negotiation. This triggers a final round of negotiation that is carried out by procedure withdrawal($T_{r,g}, O, Offered, out$) (Algorithm 7) that selects one of the offers from the input set *Offered*. This set contains all the offers proposed during the negotiation by the agent who wishes to withdraw. The agent who receives the withdraw message, finds the best offer contained in *Offered*. If this is better than disagreement, he sends it in a *final* message. The negotiation terminates with agreement if this offer is still better than o_D for the other agent, otherwise it terminates with disagreement.

```
if Performative(m_{r,g-2}) = withdraw then
    out = nil;
else
    Select o ∈ Of s.t. o ▷ o_D and ¬∃o' ∈ Of s.t.
    o' ▷ o_D and o' ▷ o;
    if o exists then
        Send final(o); out = o;
    else
        Send withdraw; out = nil;
    end
end
return out;
```

Algorithm 7: Procedure withdrawal$(T_{r,g}, O, Of, out)$

It is worth noting that although the above algorithms implement a specific negotiation strategy, the overall process they describe is generic in the sense that it can easily adapted to accommodate other strategies. Consider for instance the case where one of the agents receives a *reject* message to an offer he made in some previous move. In the current version of procedure negotiate he removes his offer and the other agent takes turn. Moreover, in the next round he will *concede*, by sending his next best offer. All these are strategic decisions that can easily modified without altering in any way the working of the overall algorithms, and more importantly the role of argumentation in negotiation.

Moreover, the argumentive alternating offers protocol we propose has two useful properties. The first property is *soundness*. This property guaranties that any offer agreed by the two agents through the argumentative alternating offers protocol is better for both agents than the offer that corresponds to disagreement i.e. o_D. More formally:

PROPOSITION 1. *(Soundness) Let $d = \{r_1, ..., r_n\}$ be an argumentative alternating offers dialogue between two agents α and β. If $Outcome(d) = o$, $o \neq nil$ then $o \triangleright o_D^\alpha$ and $o \triangleright o_D^\beta$.*

Another interesting property of the argumentative alternating offers protocol is that any negotiation dialogue produced through this protocol terminates.

PROPOSITION 2. *$\forall d \in \mathcal{D}$ where $\mathcal{D}$ is the set of all the dialogues built from the alternating offers protocol, d terminates.*

5. EXAMPLE

For illustrating our negotiation algorithm we will use a simple scenario where a buyer (ag_b) and a seller(ag_s) negotiate over the price of a product. The set of options is $O = \{o_1, o_2, o_3, o_D\}$, where $o_1 = high$, $o_2 = medium$, $o_3 = low$ are referring to the price of the product, and o_D represents the options of not selling (buying) for the seller (buyer).

Assume that the seller prefers a high price to a medium price to not selling to a low price. Symmetrically, the buyer prefers a low price to a medium price to not buying to a high price. We also assume that we are in a high-season period, but the buyer agent is not aware of that before the negotiation. Both agents represent their knowledge in some propositional language $\mathcal{R}$.

Assume that the buyer has the following knowledge:
$regular\_customer$

$regular\_customer \rightarrow discount$
$discount \rightarrow buy2, buy2 \rightarrow o_3.$
$high\_season \rightarrow o_1$
$high\_season \rightarrow \neg discount$
$high\_season \land regular\_customer \rightarrow o_2$

From this knowledge base the agent can construct one practical argument $\delta_1 = (\{ regular\_customer, regular\_customer \rightarrow discount, discount \rightarrow buy2, buy2 \rightarrow o_3\}, o_3)$ that supports o_3. Two epistemic arguments can also be constructed: $\alpha_1 = (\{ regular\_customer, regular\_customer \rightarrow discount\}, discount)$, and $\alpha_2 = (\{ regular\_customer, regular\_customer \rightarrow discount, discount \rightarrow buy2\}, buy2)$.
We have therefore only one extension $E = \{\delta_1, \alpha_1, \alpha_2\}$. Thus, the option o_3 is skeptical and o_2, o_1 are rejected.

Assume now that the seller has the following knowledge:
$high\_season$
$high\_season \rightarrow \neg discount, high\_season \rightarrow o_1.$
$high\_season \land regular\_customer \rightarrow o_2.$
$sales\_season \land regular\_customer \rightarrow o_3.$

The seller agent has one practical argument $\delta_2 = (\{high\_season, high\_season \rightarrow o_1\}, o_1)$ which supports o_1. He has also one epistemic argument $\alpha_3 = (\{high\_season, high\_season \rightarrow \neg discount\}, \neg discount)$.

We have therefore only one extension $E = \{\delta_2, \alpha_3\}$. Thus, the offer o_1 is skeptical and o_2, o_3 are rejected.

Supposing that ag_b begins the negotiation, the dialogue between ag_b and ag_s will be as follows:

$m_{1,1}:Propose(ag_b, ag_s, o_3, \delta_1)$
$m_{1,2}:Argue(ag_s, ag_b, \alpha_3, (m_{1,1}))$
$m_{1,3}:Agree(ag_b, ag_s)$
$m_{2,1}:Propose(ag_s, ag_b, o_1, \delta_2)$
$m_{2,2}:Nothing(ag_b, ag_s)$
$m_{2,3}:Nothing(ag_s, ag_b)$
$m_{2,4}:Reject(ag_b, ag_s)$
$m_{3,1}:Propose(ag_b, ag_s, o_2, \delta_3)$
$m_{3,2}:Accept(ag_s, ag_b, o_2)$

The buyer agent proposes first his optimal offer which is o_3 with his supporting argument δ_1. The seller agent updates his theory which now contains the arguments δ_2, $\delta_3 = (\{high\_season, regular\_customer, high\_season \land regular\_customer \rightarrow o_2 \}, o_2)$, α_3, δ_1, α_1 and α_2. For the defeat relation of the seller agent we have the following situation:

- (δ_2, δ_3), (δ_2, δ_1) (δ_3, δ_1) because the conclusions of δ_2, δ_3 and δ_1 are not the same (and therefore conflicting) and also because the preferences of the agent are $\delta_2 \succ_p \delta_3$, $\delta_2 \succ_p \delta_1$ and $\delta_3 \succ_p \delta_1$.

- (α_3, δ_1) because there is undercutting and $\alpha_3 \succ_m \delta_1$.

- (α_3, α_2) and (α_3, α_1) because there is undercutting between α_3 and α_2, rebuttal between α_3, and α_1 and the preferences of the agent are $\alpha_3 \succ_e \alpha_2$ and $\alpha_3 \succ_e \alpha_1$.

The theory of the seller agent has one extension, $E = \{\alpha_3, \delta_2\}$, and therefore the seller tries to defeat with α_3 the argument he received in the last move.

When the buyer receives move $argue(\alpha_3)$, he first updates his theory. This theory now contains the arguments α_1, α_2, α_3, δ_1, δ_2, δ_3, whereas the defeat relation is as follows:

- (α_3, α_2) and (α_3, α_1) because there is undercutting between α_3 and α_2, rebuttal between α_3 and α_1. The preferences of the agent in this context are now $\alpha_3 \succ_e \alpha_2$ and $\alpha_3 \succ_e \alpha_1$.

- (δ_1, δ_2), (δ_1, δ_3) and (δ_3, δ_2) because the conclusions of δ_2, δ_3 and δ_1 are not the same and the preferences of the agent are $\delta_1 \succ_p \delta_2$, $\delta_1 \succ_p \delta_3$, $\delta_3 \succ_p \delta_2$.

- (α_3, δ_1) because there is an undercutting attack and $\alpha_3 \succ_m \delta_1$.

The theory of the buyer after $m_{1,2}$ has one extension, $E = \{\alpha_3, \delta_3\}$, and therefore o_3 is not the best offer for him. Consequently, he agrees with the seller.

This initiates a new round, where the seller proposes the offer o_1, which is a skeptical conclusion of his theory. The buyer updates his theory with argument δ_2, but there is no change. The offer o_2 remains the best, and thus he needs to defeat the argument he received. But, none of the acceptable arguments defeats δ_2 and then he sends *nothing* to indicate that he does not accept the offer.

When the seller agent receives the *nothing* message, he attempts to find another argument which supports his offer o_1. As this attempt fails, he sends a *nothing* message to signify that he has not change his preference on offer o_1 but he has no other argument for supporting it. The buyer ends the round with a *reject* and thus, a third round begins in which the buyer agent proposes his best offer o_2 with δ_3 to support it. Here we assume that the seller agent is willing to concede, and therefore he accepts o_2 because this is an acceptable offer for him. Thus the negotiation ends with an agreement.

6. RELATED WORK AND CONCLUSION

In the last years several works have appeared in the argumentation based negotiation literature. These works have focused on several aspects of negotiation such as the problem of decision making (see e.g. [7], [1]), the study of specific types of negotiation such as interest based negotiation [9], whereas the work of [2] proposed a general framework for argumentation based negotiation where several interesting issues have been studied. These issues include the link between the status of the arguments and the offers they support, the definition of important concepts such as the concession and its impact on the evolution of the negotiation, etc. This work is the most relevant to ours. Nevertheless, there are some important differences. One of them is that in our paper we make a clear distinction between epistemic and practical arguments and, by adapting the work presented in [3], we show how epistemic arguments interfere with practical arguments in the definition of the acceptable offers. Then, we show how this reasoning mechanism can be used by the agents in the context of an original adaptation of the well known alternating offers protocol [12]. Another difference is that in our work we are interested in strategic issues. More precisely, we propose a generic algorithm that implements a strategy that can be used by both agents. This algorithm can be parameterized in different ways in order to capture, for example, different conditions of concession, different methods for ranking offers or different tactics for deciding whether withdrawing or making a concession. Our future work will address several open issues. One such issue is the study of several tactics for choosing the best offer to propose, especially in the context of time constraint negotiations. Another issue is the investigation of different methods for ranking the offers, whereas a third issue is that of the formal properties of the argumentative alternating offers protocol, apart from the ones of soundness and termination that we have already presented.

7. ACKNOWLEDGMENTS

8. REFERENCES

[1] L. Amgoud, S. Belabbes, and H. Prade. Towards a formal framework for the search of a consensus between autonomous agents. In *4th International Joint Conference on Autonomous Agents and Multi-Agents systems*, pages 537–543, 2005.

[2] L. Amgoud, Y. Dimopoulos, and P. Moraitis. A unified and general framework for argumentation-based negotiation. In *6th Inter. Joint Conf. on Autonomous Agents and Multiagent Systems*, pages 113–124, 2007.

[3] L. Amgoud, Y. Dimopoulos, and P. Moraitis. Making decisions through preference-based argumentation. In *KR'08*, pages 963–970, 2008.

[4] E. Black and K. Atkinson. Dialogues that account for different perspectives in collaborative argumentation. In *8th International Conference on Autonomous Agents and Multi-Agents systems*, pages 867–874, 2009.

[5] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence Journal*, 77:321–357, 1995.

[6] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *Artif. Intell.*, 142(2):205–237, 2002.

[7] A. Kakas and P. Moraitis. Adaptive agent negotiation via argumentation. In *5th International Joint Conference on Autonomous Agents and Multi-Agents systems*, pages 384–391, 2006.

[8] S. Kraus, K. Sycara, and A. Evenchik. Reaching agreements through argumentation: a logical model and implementation. *Artif. Intell.*, 104:1–69, 1998.

[9] I. Rahwan, P. Pasquier, L. Sonenberg, and F. Dignum. On the benefits of exploiting underlying goals in argument-based negotiation. In *22nd Conference on Artificial Intelligence*, pages 116–121, 2007.

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

[11] J. Rosenschein and G. Zlotkin. *Rules of Encounter: Designing Conventions for Automated Negotiation Among Computers,*. MIT Press, Cambridge, Massachusetts, 1994., 1994.

[12] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.

[13] K. Sycara. Persuasive argumentation in negotiation. *Theory and Decision*, 28:203–242, 1990.