Computational Models for Argumentation in MAS

Carlos Iván Chesñevar  
Dept. of Computer Science  
UNIVERSITAT DE LLEIDA  
SPAIN

Guillermo R. Simari  
Dept. of Computer Science and Engineering  
UNIVERSIDAD NACIONAL DEL SUR  
ARGENTINA

Outline

• (Very brief) Introduction to Multiagent Systems
• What is argumentation? Fundamentals
• A Case Study: DeLP and its extensions as an argument-based approach to logic programming.
• Argumentation meets agents: argument-based negotiation
• Conclusions
### Types of Dialogues

<table>
<thead>
<tr>
<th>Type</th>
<th>Initial Situation</th>
<th>Main Goal</th>
<th>Participant’s aims</th>
<th>Subtypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information seeking</td>
<td>Personal ignorance</td>
<td>Spreading knowledge</td>
<td>Gain, pass on, show, or hide knowledge</td>
<td>Expert consultation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Interview</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Interrogation</td>
</tr>
<tr>
<td>Persuasion</td>
<td>Conflicting Beliefs</td>
<td>Resolution of conflict by verbal means</td>
<td>Persuade the other(s)</td>
<td>Dispute</td>
</tr>
<tr>
<td>Inquiry</td>
<td>General Ignorance</td>
<td>Growth of knowledge &amp; agreement</td>
<td>Find a proof or destroy one</td>
<td>Scientific Research</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Investigation</td>
</tr>
</tbody>
</table>

**Typology by Walton & Krabbe, 1995**

<table>
<thead>
<tr>
<th>Type</th>
<th>Initial Situation</th>
<th>Main Goal</th>
<th>Participant’s aims</th>
<th>Subtypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deliberation</td>
<td>Need for action</td>
<td>Reach a decision</td>
<td>Influence Outcome</td>
<td>Board meeting</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>War planning</td>
</tr>
<tr>
<td>Negotiation</td>
<td>Conflict of interests &amp; need for cooperation</td>
<td>Making a deal</td>
<td>Get the best for oneself</td>
<td>Bargaining</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Union negotiation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Land dispute</td>
</tr>
</tbody>
</table>
What is negotiation?

Negotiation is a form of **interaction** in which:

- a group of self-interested agents
- with conflicting interests
- and a desire to cooperate
- attempt to reach agreement
- on the division of scarce resources

**Scarce**: competing claims cannot be simultaneously satisfied

**Resources**: bandwidth, memory, money, processing power, fuel, equipment, …

---

**Negotiation Components**

- **Mechanism (or Protocol)**
  - Rules of the game
  - Who is allowed to say what, and when

- **Agent Strategies** within the rules of the protocol
  - *e.g.* what offer should I make?
  - *e.g.* what information should I provide?

- **Outcomes**
  - One of a set of possible deals (i.e. negotiation set), or
  - Conflict

*Mechanism + Participant Strategies = Outcome*
Approaches to Automated Negotiation

- Game-theoretic Approaches
- Heuristic Approaches
- Argumentation-based Approaches

Game Theory

- Branch of economics
- Study rational decisions (& outcomes) in multi-party strategic decision making
- Agents seen as **utility maximizers**
- Given a mechanism → analyze strategies & outcomes
- What mechanism to design?
What is Mechanism Design?

- Assuming **perfect rationality**
- Design rules of the game
- Such that rational agents would have to behave in a certain way
  - *e.g.*, Truth-telling is the dominant strategy
- And hence guarantee certain outcome properties

Problems with Game-Theoretic Approaches

- Assumption of **perfect rationality**:  
  - Each agent knows space of possible deals  
  - Each agent knows how to evaluate such deals  
  - Each agent knows space of possible strategies  
  - No time constraints (decisions computed instantly)  
  - No computation cost  
  - Optimal recursive modelling of opponents  
- Game theory says *nothing about how to program the agent*, *e.g.*,  
  - How to compute the utility function  
  - How to compute the space of possible strategies
Heuristic Approaches

- Agents do not necessarily know each other’s preferences
- Like a game of chess / tic-tac-toe
- Protocol does not prescribe an optimal strategy
- How to program strategies?
  - Time-dependent tactics (Fatima, 2001,2004)
  - Fuzzy similarities to generate counteroffer (Faratin, 2001)
  - Fuzzy modelling of market conditions (He et. al.)
  - Bayesian learning of opponents (Zeng & Sycara)
- Study strategy performance empirically

Heuristic Approaches: Limitations

- But...we still have similar problems as before!
  - Each agent knows space of possible deals
  - Each agent knows how to evaluate such deal
- Approximate notions of rationality → suboptimal outcomes: full space of possible outcomes not examined
- Needs extensive empirical evaluation: very difficult to predict how the system and constituent agents will behave.
Argumentation-based Approaches

In Game-Theoretic and Heuristic approaches → share some limitations

- E.g. agents exchange proposals (potential agreements or deals). Ags are not allowed to exchange any additional information (e.g.: network goods such as fax machines)
- Agents’ utilities or preferences are assumed to be completely characterised prior to the interaction.
- Agents’ preferences are assumed to be proper (ie, they reflect the true benefit the agent gets).
- Game theoretic and heuristic approaches assume that agents’ utilities or preferences are fixed (ie, agents cannot influence on other agents’ preference models or internal mental attitudes).

Argumentation-based approaches attempt to overcome these limitations…

Argumentation

⇒ Argument:
  - Reason/justification for some conclusion (belief, action, value, goal, etc.)

⇒ Argumentation:
  - Reasoning about arguments → decide on conclusion

⇒ Dialectical argumentation
  - Multi-party argumentation through dialogue
Argumentation-Based Negotiation

**Negotiation**
- Auctions
- Bargaining
- Mechanism Design

**Argumentation**
- Knowledge fusion
- Dialogue-Game argumentation
- Legal argumentation

**Approaches: summary**

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Game-theoretic</td>
<td>Strong analytical results</td>
<td>• Assumed fix, correct and complete preferences</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Says little about how to program agents</td>
</tr>
<tr>
<td>Heuristic-based</td>
<td>Concentrate on programming agents</td>
<td>• Assumed fix, correct and complete preferences</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Hard to validate strategies</td>
</tr>
<tr>
<td>Argumentation-based</td>
<td>Very expressive Preferences can change</td>
<td>• Validating strategies still at very early stages</td>
</tr>
</tbody>
</table>
Elements of ABN Frameworks

- Communication Languages & Domain Language
- Negotiation Protocol
- Information Stores
- Argument and Proposal Evaluation
- Argument and Proposal Generation
- Argument Selection

EXTERNAL FEATURES (ENVIRONMENT)

AGENT-BASED FEATURES

Comm. Language & Domain Language

- **Communication Language**: characterized by locutions, utterances or speech acts (*propose*, *accept*, *reject*, etc.)

- **Domain language**: used for referring to concepts of the environment. E.g. (Sierra et. Al, 1998)

  **Offer** \((a, b, \text{Price} = $200 \land \text{Item} = \text{palm130}, t_1)\)

  Agent a offers agent b a Palm130 for the price of $200 at time \(t_1\)

  **Reject** \((b, a, \text{Price} = $200 \land \text{Item} = \text{palm130}, t_2)\)

  Agent b rejects proposal from agent a at time \(t_2\)
Non-ABN vs. ABN Frameworks

<table>
<thead>
<tr>
<th></th>
<th>Non-ABN Frameworks</th>
<th>ABN Frameworks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain language</td>
<td>Expresses proposals only</td>
<td>Expresses proposals as well as metainformation about the world (beliefs, prefs., etc.)</td>
</tr>
<tr>
<td>Communication language</td>
<td>Locutions allow agents to pass call for bids, proposals, acceptance, etc.</td>
<td>In addition, locutions allow agents to pass meta-information</td>
</tr>
</tbody>
</table>

Comm.Language: State of the Art

- Two major proposals for ACL have been advanced: KQML (Knowledge Query and Manipulation Lang., 1996) and FIPA ACL (Foundation for Intell. Physical Agents, 2001).
- FIPA ACL offers 22 locutions, and contents of messages can be in any domain language, e.g., "inform(a,b,φ,L)" stands for "Ag.a informs Ag.b about φ in language L".
- But... FIPA ACL fails to capture all utterances needed in a negotiation interaction!
  - There is no locution for expressing desire of enter/leave a negotiation interaction, request an argument for a claim, etc.
  - As a solution, new negotiation-specific locutions have been defined outside FIPA ACL (eg. threaten, promise, etc.)
Example of FIPA ACL Limitations

Consider the following locution:

\[ \text{Request}(\text{Ag2, Ag1, Do(Ag1, } \alpha), \text{Do(Ag1, } \alpha) \rightarrow \text{Do(Ag2, } \beta) ) \]

Agent2 request Agent1 to perform action \( \alpha \), and supports that request with an argument stating that “if Agent1 accepts, Agent2 will perform action \( \beta \) in return”.

But we don’t have information about if \( \beta \) is desirable for Agent1 or not! (Request locution does not convey this)

If it is, the above situation would represent a promise from Agent1 to Agent2. Otherwise, it could be a threat!

Domain Language: State of the Art

- In negotiation, the domain language must be capable of expressing the object of negotiation, as well as preferences of agents (Sierra, 1998), e.g.

\[ (\text{Price} = $200) \land (\text{Quality} = \text{high}) \land (\text{Penalty} = ?) \]

- ABN frameworks may need elements to express plans and resources. E.g. (Sadri, 2002)

\[ \text{plan}((\text{hit(nail)},\text{hang(picture)}),\{\text{picture,nail,hammer}\}) \]

Domain Language is important in ABN: the richer the domain language, the richer the arguments that can be exchanged between agents.
Given a communication and domain language, we need a *negotiation protocol* (which includes an *interaction protocol*) in order to constrain the use of the language.

Protocol: formal set of conventions governing the interaction among participants. (Jennings, 2001)

**Negotiation Protocol**

- **Interaction protocol**: specifies who is allowed to say what.
- **Negotiation protocol**: involves additional rules
  - Admission Rules: when an agent is allowed to participate in a negotiation? Under what conditions?
  - Rules for Participant Withdrawal: when a participant can withdraw from a negotiation?
  - Termination rules: when an encounter must end?
  - Rules for Proposal Validity: is proposal compliant?
  - Rules for Outcome Determination: what is the outcome?
  - Commitment Rules: which commitments are involved?
Protocols: State of the Art

Interaction protocols for ABN agents can be explicit or implicit.

Explicitly:
- by means of finite state machines (Sierra et al. 1998). Advantage: handy for a limited number of locutions. Disadv: complex for increased number of locutions.
- By means of dialogue games (Amgoud et al, 2001; McBurney et al., 2003). Advantage: they have public axiomatic semantics.

Implicitly:
- By means of logical constraints expressed as “if-then” rules (e.g. Kraus et al., 1998; Sadri et al. 2001, 2002). In this case, interaction protocol is “hardwired” in the agents’ internal specification.

Finite State Machines for Protocols

Negotiation protocol for two agents
Locutions in Dialogue Games – Example (McBurney et. al, 2003)

**Locution:** \texttt{willing\_to\_sell}(P_1,T,P_2,V), where \( P_1 \) is either an advisor or seller, \( T \) is the set of participants, \( P_2 \) is a seller, \( V \) is a set of sales options.

**Precond:** some participant \( P_3 \) must have previously uttered a locution \texttt{seek\_info}(P_3,S,p), where \( P_3 \in S \) (the set of sellers), and the options in \( V \) satisfy constraint \( p \)

**Meaning:** Speaker \( P_1 \) indicates audience \( T \) that agent \( P_2 \) is willing to supply the finite set \( V = \{ a_1, a_2, \ldots, a_k \} \) of purchase options to any buyer in \( T \). Each option satisfies constraint \( p \) in prior \texttt{seek\_info}(.) locution.

**Response:** None required

**Information Store Updates:** For each \( a_i \in V \), the 3-tuple \((T, P_2, a)\) is inserted into \( IS(P_1) \).

**Commitment Store Updates:** None

---

Preconditions – Example (Sadri et. al, 2002)

Protocol rules are coded as part of the agents’ programs (usually abductive logic programs).

\[ P(t) \land C(t) \rightarrow P'(t+1) \]

“If agent receives performative (locution) \( P \) at time \( t \) and condition \( C \) was satisfied at that time, then the agent must use the performative \( P' \) at the next time point.”

E.g.: if an agent receives a performative including request of a resource, and it does not have the resource, then it must refuse the request.
Other external elements

- **Termination rules** can be specified in different ways:
  - E.g. in FSA, they are a set of links to a final state, usually after an agent utters `withdraw()` or `accept()`.
  - E.g. in (McBurney et.al, 2003), a rule specifies that the dialogue ends by the locution `withdraw_dialogue()`.

- **Outcome determination** also varies:
  - Some frameworks determine outcomes based on the logical structure of interacting arguments (i.e., outcome is *implicit* in the underlying argumentation logic).
  - In other frameworks, outcomes are reached through uttering a specific locution *explicitly* (e.g. `accept()`) usually on the basis of some internal utility evaluation.

Challenges (1)

- Protocols for ABN share challenges faced in design of argumentation protocols in general.
  - **Termination**: can termination be ensured?
    - E.g. Protocols proposed by Amgoud & Parsons (2001) do not allow to repeat the same locutions over and over again...
    - Torroni (2002) studied maximum length dialogues on the basis of properties of abductive logic programs representing agents.
  - **Guaranteed success**: under which conditions a particular protocol will end up with agreement?
    - Complexity results (Wooldridge & Parsons, 2000).
Challenges (2)

- Protocols for ABN share challenges faced in design of argumentation protocols in general.
  - **Conformance checking**: is a particular utterance acceptable given history & context of interaction?
    - Recently investigated applying model checking techniques (Huget and Wooldridge, 2003).
  - **Admission rules**: how to govern admission rules?
    - Relevant work in Electronic Institutions (Rodriguez-Aguilar, 2002).

Information Stores (IS)

- In some ABN frameworks, there is no centralised IS, and agents keep track of past utterances.
- **Commitment Stores (CS)**: a way of tracking the claims made by participants in dialogue games (Hamblin, 1970).
- CS not to be confused with “interaction history” (only passive storage).
- CS have specific *commitment rules* governing the addition/removal of statements.
Information Stores (IS)

- Information store manipulation rules have a direct effect on the types of utterances agent can make given
  
  - i. previous utterances;
  
  - ii. properties of the dialogue;
  
  - iii. the final outcome.

- Commitment to providing/requesting/exchanging resources may require different treatment from commitment in other types of dialogue (e.g., persuasion).

Elements of ABN Agents

- What constitutes a basic non-ABN negotiating agent? We can distinguish a number of components...
  
  - **Locution interpretation**: parses incoming messages
  
  - **Proposal Database**: stores proposals for future reference
  
  - **Proposal Evaluation/Generation**: ultimately makes a decision about whether to accept, reject, terminate negotiation, etc.
  
  - **Locution generation**: sends the response to the relevant party or parties.
Computational Models for Argumentation in Multiagent Systems – EASSS 2005

Classical Negotiating Agent

Basic Negotiator: elements
In addition to evaluating and generating proposals, an agent capable of participating in ABN must be equipped with mechanisms for

- **Evaluating arguments** (updating mental states accordingly)
- **Generating and Selecting arguments**
- **Evaluating and interpreting arguments**
Argument and Proposal Evaluation

- An ABN agent needs to evaluate potential agreements proposed by its counterparts.
- *Proposals* may be evaluated through comparison with some subjective preference criteria.
- *Argument* evaluation → less trivial...
  - **Objective considerations**: assessing “quality” of argument as a tentative proof (e.g. Specificity (Poole, 1984), acceptability classes (Dung, 1995), etc.).
  - **Subjective Considerations**: an agent may choose to consider its own preferences and motivations in making a judgement.

Different Types of Dialogues…

- **Theoretical Reasoning**: reasoning about what is true in the world should be rational and without biases of the participants.
- **Practical Reasoning**: a dialogue for deciding a course of action, or division of scarce resources. Agents not concerned in truth *per se*, but rather with the satisfaction of their needs.
- In negotiation dialogues agents are required to perform argument evaluation based on **objective** and **subjective** criteria.
Proposal & Argument Evaluation

- **“Benevolence” approach:** assume agents are benevolent, using the following rule: *If I do not need a resource, I should give it away when asked* (e.g. Parsons et. al, 1998; Amgoud et. al, 2000; Sadri et al, 2001).

- **Agents present arguments defending their intentions.** Two kinds of conflict may appear:
  - Agents have conflicting intentions (rebutting arguments)
  - An agent rejects one of the elements of the argument supporting the intention of another agent B (undercutting arguments).

---

The “Hang Mirror/Picture” Problem

(Credits to F.Dignum for original slide)
Proposal & Argument Evaluation

(Parsons et.al. 1998): agent A intends to hang a picture.
- After executing its planning procedure, produces intentions to acquire a nail, a hammer and a picture.
- Agent B gives A a nail, so he can hang the picture.
- Agent B does not want to give A the nail, because he needs it for his plan.
- Agent A says: I don't want the nail, but just to hang the picture. If I give you screw and screwdriver (I do not need them), you can have a plan for hanging the mirror, as you want.
- Agent B says: ok, let's proceed that way. I give you the nail, and both of us are happy!

Other Approaches

“Count on you” approach (Sadri et. al, 2001): not argue about beliefs e.g.:
- Agent A receives a request from Agent B for a resource that he needs it for achieving goal $G_A$ with a plan $P$.
- Agent A rejects the request, unless an alternative acceptable plan $P'$ for $G_A$ can be produced by B, with a promise to provide A with every needed resource for plan $P'$.

Agents are assumed to have some ordering in their plans.

In both approaches, we are relying on the (rather naive) assumption that agents accept any request on resources that they do not currently need. What if agents are purely self-interested, and want “something in return”? 
Utility in Proposal & Argument Evaluation

- **Idea**: the agent can calculate *expected utility* for cases of accepting/rejecting proposals. Comparing expected utilities, a decision is made.

- In (Kraus, 1998) *combined factors* are used to calculate utilities (*collision_flag*, *convincing_factor*, *acceptability_value*).

- (Ramchurn et al., 2003) take into account *trust* in the counterpart to calculate expected values.

- (Sierra et al., 1998) introduce *authority* as a criterion for evaluating arguments (authority graph). e.g.: introduction of conciliatory agent to solve appeal-to-authority arguments.

Challenges

- In most models, agents do not *voluntarily modify* their position, but rather *forcibly concede* as a result from pressure from counterparts. *Can objective evaluation of arguments be combined with subjective evaluation of its consequences?*

- Unifying argumentation frameworks and facilitate negotiation dialogues about goals, beliefs, plans, etc.

- (Rahwan et al., 2003) argue that *argumentation systems designed for arguing about beliefs are not readily suitable for allowing argumentation over goals.*
Example: Travel to Utrecht

Example: Travel to Utrecht

Some ways of attacking argument \( \langle A, goUtrecht \rangle \):

1) Present statement: \( \neg \text{achieve}(\{\text{buyTicket, arrangeAccom}\}, goUtrecht) \)

Counterpart attacks the relation between subgoals and the goal, arguing that buying a ticket and arranging accommodation are not enough for going to Utrecht.

Effect: \( \neg \text{achieve}(\{\text{buyTicket, arrangeAccom}\}, goUtrecht) \) is removed from the KB of the agent. If no alternative plan is found, goal is deemed unachievable.

2) Present statement: \( \text{Instrum}(goRotterdam, presentPaper) \)

Counterpart presents an alternative (go to Rotterdam, there is a similar conference there).

Effect: \( \text{Instrum}(goRotterdam, presentPaper) \) is added to the agent’s KB. Plans are compared and outcome is determined.

Argument and Proposal Generation

- Problem: generating candidate arguments to present to a dialogue counterpart.
- Such arguments are sent to “entice” the counterpart to accept some proposed agreement.
- Some approaches:
  - (Sierra et. al, 1998) assume agents have a means of generating proposals that increase (or maximise) utilities.
  - (Kraus, Parsons, Sadri) assume that an underlying planner generates a set of actions or resources needed to achieve some intention.
- Proposals may be accompanied by arguments generated by explicit rules (e.g. Kraus et. Al, 1998)
Generating Candidate Arguments

IF

A request has been sent to agent B to perform action $\alpha$ &
Agent B rejected this request &
Agent B has goals $G_1$ and $G_2$ &
Agent B prefers $G_2$ to $G_1$ &
Agent B doing $\alpha$ achieves $\neg G_1$ and doing $\beta$ achieves $\neg G_2$
Agent A believes doing $\beta$ is credible and appropriate

THEN

Agent A request B to do action $\alpha$ with the following threat:
“if you don’t do $\alpha$, I will do $\beta$”

Agent A may generate other candidate arguments (e.g., promises or appeals) using other rules.

Argument and Proposal Generation

- Other frameworks take a planning approach to proposal generation:
  - agents justify requests by simply stating the truth about needs, plans, underlying assumptions, etc.
  - Note difference wrt utility-based approaches, where agents “create” arguments by exploiting their abilities to influence outcomes (e.g. threats, rewards).

- Authority can also be used in argument generation
  - (Sierra et. al, 1998) propose a simple “authoritarian” agent which always exploits its social power by threatening whenever possible.

**Challenge**: formal characterization of the “space” of possible arguments, and understand influence among factors (authority, expected utility, trust, honor, etc.)
Argument Selection

- Given a number of candidate arguments, which is the most suitable one?
- Argument selection may take part in conjunction with argument generation.
- (Kraus et. al, 1998) propose the following scale for argument strength:
  - Appeal to prevailing practice
  - A counter-example
  - An appeal to past promise
  - An appeal to self-interest
  - A promise of future reward
  - A threat
  - Weakest arguments
  - Strongest arguments

In (Ramchurn et. al, 2003), agents evaluate trust and utility to decide which candidate argument to send with a request:

If trust is low and utility of the proposal is high
(ie, I need to do X and I don’t trust you)
then send a strong argument

If trust is high and utility of the proposal is low
(ie, I don’t need to do X so much but I trust you)
then send a weak argument

Low and High are linguistic variables manipulated using fuzzy operators
Argument Selection: Challenges

- Argument Selection as a problem can be considered to be the essence of **strategy** in ABN dialogues.
- There is little existing work on strategies in MAS dialogues.
  - There has been research in **persuasion dialogues** (e.g. Amgoud & Maudet, 2002) and **inquiry and information seeking** dialogues (Parsons et. al, 2002, 2003).
  - (Rahwan et. al, 2003a) provide a first attempt at characterising strategic factors in negotiation dialogues.
- Information about negotiation counterpart should be taken into account → **learning** techniques required to find patterns.

Outline

- (Very brief) Introduction to Multiagent Systems
- What is argumentation? Fundamentals
- A Case Study: DeLP and its extensions as an argument-based approach to logic programming.
- Argumentation meets agents: argument-based negotiation
- Conclusions
Open issues and future directions

- Argumentation: state of the art.
- Multiagent systems (MAS): state of the art.
- Integration of argumentation and MAS
- Open issues and future directions:
  - Argumentation & communicative rationality in MAS
  - Argumentation & trust in MAS
  - Complexity of argument-based approaches in a MAS context
  - Argument-based programming languages for agents

Summary of This Tutorial

- Multiagent Systems: a brief overview
- Foundations of Argumentation Systems
- DeLP, O-DeLP, P-DeLP: argument-based approaches to logic programming
  - Main definitions. Using DeLP in Reasoning Modules in Agents.
  - DeLP, P-DeLP and O-DeLP in real-world applications.
- Argumentation meets agents
  - Important emerging field: argument-based negotiation.
- Conclusions
References


References

References


References

References


The End