Testing the benefits of structured argumentation in multi-agent deliberation dialogues

Eric M. Kok, John-Jules Ch. Meyer, Henry Prakken, and Gerard A. W. Vreeswijk

Department of Information and Computing Sciences,
Utrecht University,
The Netherlands

Abstract. Work on argumentation-based dialogue systems often assumes that the adoption of argumentation leads to improved dialogue efficiency and effectiveness. Several studies have taken an experimental approach to prove these alleged benefits, but none has yet supported the expressiveness of a structured argumentation logic. This paper shows how the use of argumentation in deliberation style dialogues can be tested while supporting goal-based agents that use the ASPIC framework for structured argumentation. It is experimentally shown that employing an arguing strategy increases the effectiveness over a non-argumentative strategy.

Category I.2.11 [**Artificial Intelligence**] Distributed Artificial Intelligence—Languages and structures, multi-agent systems
General Terms Design, Experimentation
Keywords Argumentation, Multi-agent dialogues, Experimental evaluation, Strategies

1 Introduction

To improve communication and shared decision making in multi-agent systems it is often proposed to allow for argumentation in inter-agent dialogues. The idea is that, by providing arguments and giving counter-arguments, the effectiveness, in terms of the desirability of the dialogue result, as well as the efficiency, as the number of exchanged messages, will improve. These benefits are supposed to originate in the motivation that agents can give in their discourse. Instead of plain assertions and requesting information, agents can actually give a motivation, revealing their goals and thereby contributing to the general understanding. This increased understanding should result in a faster process and more considerate decision, raising the dialogue efficiency and effectiveness.

Throughout the years many argumentation-based dialogue frameworks and protocols have been developed and the theoretical reachability of ideal and intuitive outcomes has often been proved formally. However, not all properties can be studied formally [14]. Therefore a recent trend is to study the benefits of argumentation through experimentation.

At least three works have explored the practical benefits of argumentation in dialogues. In both Karunatillake et al. [7] and Paquier et al. [10] argumentation-based negotiation is studied. While the topics of their systems is different, that is, social agent societies and exploring the negotiation space, their argumentative parts are modelled alike. Within a dialogue agents may ask for and supply a motivation (reason) for proposals. However, the language does not allow agents to build structured arguments, which severely limits their expressivity in the dialogue.

Black and Bentley [4] experimentally evaluate the use of argumentation in two-party deliberation dialogues. Agents are initialized with a set of value-based arguments which are used in the dialogue to decide on some action. The argumentative strategy is shown to outperform a simple consensus-forming strategy in randomly generated dialogues with a wide variety in the number of arguments, values and actions. On the other hand, the arguments have very little structure, as there is merely a single inference step, that is, the application of the practical syllogism.

While this initial empirical research in argumentation-enabled dialogue systems has already provided interesting results, little experimental research has yet been done on dialogues in which agents exchange arguments with a more detailed internal structure. Where formal frameworks model expressive goal-based agents that construct arguments form their beliefs and goals, existing experimental work is limited to asking for motivation or providing arguments with very limited structure. This paper shows how, on the basis of the ASPIC framework for structured argumentation [13], deliberation scenarios can be generated, how strategies can use structured arguments to rationalize decision making and how the efficiency and effectiveness of dialogues can be measured. It is shown that an arguing strategy, even with self-interested agents, increases the shared utility over a non-arguing strategy.

2 Deliberation model

The agent dialogue type in which the use of argumentation will be tested is the deliberation dialogue. In deliberation, agents aim to reach agreement on a course of action to solve a problem. This type of dialogue is of particular interest because of the mix of competitive and cooperative elements. Foremost the agents are assumed to engaged in the dialogue because they share a mutual goal (to solve a given problem), which will need to be achieved by selecting some action, called an option. An example is what product a business should develop to increase profits. Various options are proposed and motivated with arguments for the mutual goal, which then can be attacked with counter-arguments. Every agent has a certain role in the dialogue, such as an engineer or salesman, which gives rise to most of the agent's goals and belief on the deliberation problem at hand. Originating in this knowledge, together with personal goals and beliefs, an agent acts in the dialogue in a self-interested way, i.e. it tries to influence the dialogue outcome to maximize its personal utility. These characteristics form the foundation for the deliberation model of this paper.

It is good to note that the deliberation as modelled in this paper does not address any planning towards the execution of agreed upon actions. It covers only the mutual process on deciding on a course of action. Although agents can discuss whether they believe a plan to be realisable, the actual planning should be handled in a subsequent (dialogue) phase.

2.1 Argumentation logic

Arguments are formed using an argumentation logic. We use a simple instantiation of the abstract ASPIC framework for argumentation with structured arguments [13], which is an instance of the Dung [5] abstract argumentation model. (For reasons of space we refer the reader to [13] for the full details.) It allows agents to create structured arguments from a knowledge base, modelled as inference trees of applied strict and defeasible rules. An argument can be attacked by rebutting a conclusion of a defeasible inference, by undermining one of its premises or by undercutting one of its defeasible inferences. From the resulting attack relation and a preference relation a defeat relation is defined. This relation then induces an abstract argumentation framework in the sense of [5], which can thus be used to evaluate the acceptability status of arguments.

In this paper a simple instantiation of the framework is assumed, with a simple logical language consisting of propositional literals, only defeasible rules and no preference ordering on arguments (such an instantiation is called an *ASPIC argumentation system*). Rules are written as $p \Rightarrow_{\varrho} q$, where the rule name ϱ is omitted for clarity when appropriate and where the premise p and conclusion q are literals in the topic language. Arguments are written as $A \mid\!\sim p$ where A is the set of used premises and inferences and p is the conclusion. The software experiment in this paper uses the ASPIC Java Components implementation [15] of the framework.

2.2 Dialogue model

The dialogue model used is a slightly simplified version of the framework for deliberation dialogues of Kok et al. [8], based on the work of Prakken. [11] The relevant details of the framework are now given. It models a dialogue between agents discussing a deliberation problem at hand using a topic language with options, goals and beliefs.

Definition 1. *A dialogue system consists of a set agents $\mathcal{A}$ discussing in a* topic language L_t*, which is a logical language closed under classical negation, containing disjoint sets of formulas for options $L_o \subseteq L_t$, beliefs $L_b \subseteq L_t$ and goals $L_g \subseteq L_t$, trying to reach a decision on a course of action, which is an option $o \in L_o$, given the* mutual goal $g_d \in L_g$.

Dialogues are modelled as a series of moves, each consisting of a locution from the communication language as listed in Table 1, and some content in the form of a proposition or argument.

Definition 2. *A dialogue d is a sequence of moves, where each move $m \in d$ is denoted by* $\mathtt{id}(m)$*, the move identifier,* $\mathtt{player}(m)$*, the agent that played the move,* $\mathtt{content}(m)$*, the content of the move and* $\mathtt{target}(m)$*, the move target. The set of all dialogues is denoted D.*

Consider two agents, a and b, who need to decide on where to go for dinner, while respecting their mutual goal g_d to enjoy the food. The moves of their dialogue are presented in Table 2

Every move, except *propose* and *skip*, explicitly replies to one previous move, so several disjoint proposal trees are formed, where the links between the nodes are reply relations between moves.

Table 1. The available speech acts in the communication language

speech act	attacking reply
propose(o)	*why-propose*(o)
	reject(o)
why-propose(o)	*argue*($A \mid\!\sim p$) where $o \in A$
argue($A \mid\!\sim p$)	*argue*($B \mid\!\sim p'$) where $B \mid\!\sim p'$ defeats $A \mid\!\sim p$
	why(p') where $p' \in A$ and $p \notin L_o$
why(p)	*argue*($A \mid\!\sim p$)
reject	
skip	

Table 2. An example dialogue showing the mapping of natural language to speech acts

agent	statement	logical form
a	I suggest we go to the pizzeria.	*propose*(o)
b	Why should we go there?	*why-propose*(o)
a	If we would go to the pizzeria, we could drink wine and that means we will enjoy our food.	*argue*($o, o \Rightarrow_{\varrho 1} p_1, p_1 \Rightarrow_{\varrho 2} g_d \mid\!\sim g_d$)
b	The pizzeria does serve tasty pizza's and having those means we will enjoy the food.	*argue*($o, o \Rightarrow_{\varrho 3} p_2, p_2 \Rightarrow_{\varrho 4} g_d \mid\!\sim g_d$)
b	We can not drink wine, though.	*argue*($\neg p_1 \mid\!\sim \neg p_1$)
b	And drinking wine does not mean we will enjoy the food.	*argue*($\neg \varrho_2 \mid\!\sim \neg \varrho_2$)
a		*skip*
b		*skip*
a		*skip*

Definition 3. *For each proposal move m_i in dialogue d a* proposal tree *P is defined as follows:*

1. *The root of P is m_i.*
2. *For each move m_j that is a node in P, its children are all moves m_k in d such that* $\texttt{target}(m_k) = m_j$.

This is a tree since every move in d has at most a single target.

Figure 1 shows the proposal tree of the example agents discussing where to go for dinner. The explicit reply structure is used to assign a dialogical status, *in* or *out*, to every move in the dialogue. By making proposals and replying to these the agents influence the status of the moves and ultimately of the dialogue outcome.

Definition 4. *The* move status *of a move m in a proposal tree P is* in *in dialogue d iff m has no attacking replies in d that are* in*; otherwise it is* out*.*

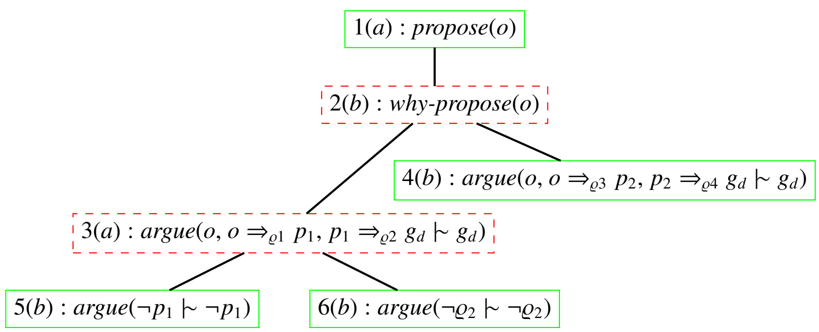

Fig. 1. Proposal tree of the example dialogue of Table 2

The proposal move in the example tree of Figure 1 is *in*, since it has no attacker that is *in*. The argue move $argue(o, o \Rightarrow_{\varrho 1} p_1, p_1 \Rightarrow_{\varrho 2} g_d \mathrel{|\!\sim} g_d)$ is *out* since that does have an attacker that is *in*.

Which agent may make the next move in a dialogue is determined by the turn taking rule. Agents take turns in sequence and end their turn explicitly with a *skip* move.

Definition 5. *For a dialogue $d = \langle m_1, \ldots, m_n \rangle$ the turntaker $\mathcal{T}(d) = \texttt{player}(m_n)$ unless $\texttt{content}(m_n) = \text{skip}$ in which case $\mathcal{T}(d) = a_i$ where $i = \texttt{id}(\texttt{player}(m_n)) + 1$ if $a_i \in \mathcal{A}$ or else $i = 0$.*

In their choice of moves the agents are bound by the following protocol:

Definition 6. *Protocol $\mathcal{P}$ restricts the legal moves on a dialogue d with mutual goal g_d, such that:*

1. *Agents can only reply to moves of others, i.e. for every attacking move $m \in d$ it holds that $player(m) \neq player(\texttt{target}(m))$.*
2. *Agents can only move when they have the turn, that is, if m is legal as a continuation of d, then $\mathcal{T}(d) = \texttt{player}(m_n)$.*
3. *A proposal must be unique in the dialogue, i.e. for every two proposal moves $m_i, m_j \in d$ if $\texttt{content}(m_i) = \texttt{content}(m_j)$ then $m_i = m_j$.*
4. *Moves may not be repeated in a* line of attack*, i.e. the path from the attacked move to the propose move in the tree.*
5. *Arguments supporting a proposal should show how the mutual goal is achieved, i.e. every move $m \in d$ where $\texttt{target}(m) = \text{why-propose}(q)$ is of the form $\text{argue}(A \mid\!\sim g_d)$ where $q \in A$.*

A dialogue terminates if all agents no longer make other moves than directly skipping.

Definition 7. *If in a dialogue $d = \langle m_0, \ldots, m_{n-|\mathcal{A}|+1}, \ldots, m_n \rangle$ every $m \in \langle m_{n-|\mathcal{A}|+1}, \ldots, m_n \rangle$ $\texttt{content}(m) = \text{skip}$ then $\mathcal{P}(d, g_d) = \emptyset$.*

The rationale behind the termination rule is that each agent should have the opportunity to make new moves when it still wants to. However, to prevent agents from endlessly skipping until some other agent makes a beneficial move or even a mistake, the number of skip moves is limited.

Finally, the dialogue outcome is determined by some selection process. This process may be a separate phase, like ordering proposals according to the preferences in the underlying argumentation system [1] or voting. For the experiment in this paper we have used a simple selection function that picks a random proposal, but only from those proposals that are *in*.

Definition 8. *The dialogue outcome is a selection function $\mathcal{O} : D \times L_g \longrightarrow L_o$ matching a dialogue d and mutual goal g_d to a single option $o \in Q_d$, given the proposals set $Q_d = \{q | \text{propose}(q) \in d\}$.*

This paper uses a unique dialogue outcome $\mathcal{O}(d, g_d) = o_d$ being an arbitrary $o_d \in \{q | q \in Q_d$ and $propose(q) \in d$ that is *in* $\}$. In other words, it selects one of the proposed options that do not have any arguments against it that are *in*.

A possible addition to the presented dialogue model for deliberation is to make the reasoning over preferences between options explicit, for example by extending the used ASPIC argumentation system and adding the appropriate locution to the communication language. Definitions for move status need to be updated and the outcome selection process can take these preference-related moves into account. This is an interesting extension but the current model is sufficient for the goals of this study and is therefore left as future work.

2.3 Scenario generation

In our experiment, agents engage in a dialogue according to a scenario, which represents the underlying deliberation problem. It describes the mutual goal and the beliefs, goals and options known to the agents. Consequently, the structure of a scenario heavily influences the dialogue and the outcome. It is therefore important that the scenarios reflect a realistic situation, reflecting the characteristics of deliberation problems.

In the scenario generation process [9] variables are used that determine the sizes of the various defined sets. These are used to moderate the structure of the final scenario, for example the number of goals allocated to an agents. How the variables are set is explained at the end of this section.

Roles Every participating agent has a certain role in the system (for example, an engineer, a salesman or an account manager) which gives rise to most of the agent's goals and belief on the deliberation problem at hand. This describes the duties and desires of an agent as being a part of its context.

Definition 9. *A set of roles $\mathcal{R}$ is defined where, given set sizes n_{O_r} and n_{G_r}, every role $r \in \mathcal{R}$ in a deliberation context with mutual goal g_d is assigned:*

- *A set of options $O_r \subseteq L_o$ defined by $O_r = \{o_1, \ldots, o_i\}$ such that $|O_r| = n_{O_r}$*
- *A set of goals $G_r \subseteq L_g$ defined by $G_r = \{g_1, \ldots, g_j\}$ such that $|G_r| = n_{G_r}$*

Every agent $a \in \mathcal{A}$ is assigned a role $r \in \mathcal{R}$. The set of all scenario options is defined

$$O_K = \bigcup_{r \in \mathcal{R}} O_r.$$

The idea is that the role accounts for the basic set of options that the agent knows about and the goals the agent has. The variables n_{O_r} and n_{G_r} are used to control the number of options and goals associated with a role.

A running example is used to illustrate the scenario generation process. Three agents will each be assigned one of the two roles in the system, which in turn both have two options and two goals. This is visualized in Table 3.

Table 3. Example scenario with three agents

$n_{\mathcal{A}} = 3$	$\mathcal{A} = \{a_1, a_2, a_3\}$	
$n_{\mathcal{R}} = 2$	$\mathcal{R} = \{r_1, r_2\}$	
$n_{O_r} = 2$	$O_{r_1} = \{o_1, o_2\}$	$O_{r_2} = \{o_2, o_3\}$
$n_{G_r} = 2$	$G_{r_1} = \{g_1, g_2\}$	$G_{r_2} = \{g_3, g_4\}$
$n_S = 10$	$S = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}\}$	

Rule chaining The next step in generating scenarios is to create a body of knowledge for each role that gives rise to lines of reasoning between options and goals, as is typical for deliberation problems. These are called rule chains and connect a role's option to one of the role's goals. Rules in these chains are built from a limited set of atoms called the chaining seed set.

Definition 10. *A* chaining seed set *of atoms $S \subseteq L_b$ is defined as $S = \{p_1, \ldots . p_i\}$ such that $|S| = n_S$*

The variable n_S (the running example uses $n_S = 10$) is used to control the number of atoms that are used to generate rules for a chain. A chain starts with a rule with an option as premise and ends with a rule with a goal as conclusion. The conclusion of all other rules is an atom from the chaining seed set and in turn is the premise for the follow-up rule. Although chains of rules with only one positive premise may seem restricted, it will already support a sufficiently complex scenario, as will be shown in Section 3.

Definition 11. *Given a goal g, an option o and a chain length l a* rule chain *is a set of rules $C_{g,o}$ such that*

- *if $l = 1$ then $C_{g,o} = \{o \Rightarrow g\}$*
- *if $l > 1$ then $C_{g,o} = \{o \Rightarrow p_1, \ldots, p_i \Rightarrow p_j, \ldots, p_n \Rightarrow g\}$ where $n = l - 1$ and $\{p_1, \ldots, p_n\} \subseteq S$*

Intermediate atoms used to create rules are chosen arbitrarily from the chaining seed set. Note that only one chain is possible with chain length 1, but multiple paths with larger chain lengths, using different intermediate atoms. Also, the option o is the only required premise to generate a full argument in $\mathcal{L}$ for the goal g and every p_i is a sub-conclusion in such an argument.

When generating a chain for the running example, we may for instance chain role r_1's option o_1 to this agent's goal g_2. With $l = 3$ and the seed set S from Table 3 a chain

$$C_{g_2,o_1} = \{o_1 \Rightarrow_{\varrho 1} p_5, p_5 \Rightarrow_{\varrho 2} p_2, p_2 \Rightarrow_{\varrho 3} g_2\}$$

is constructed. With this chain the agent (if indeed assigned these rules later in the allocation process) can construct a single argument for g_2. Other chains, with different intermediate atoms, are of course possible.

Conflict generation Scenarios do not only contain reasons why an option will achieve some goal. An important part of deliberation problems is that there are conflicts between what is known and what the rule chains proclaim. Therefore, the next step is to extend the scenario with conflicting knowledge. This is modelled using negated facts, which are created based on a rule chain. A negated fact is generated for every way in which a rule in some chain can be attacked in $\mathcal{L}$, that is by undercutting, undermining or rebutting. These negated facts represent the contrary views in the deliberation problem about the truth status of relevant facts.

Definition 12. *A rule chain $C_{g,o}$ with length l linking some goal g and option o has a set of* possible conflicts *$\bar{C}_{g,o}$, containing for every rule $p \Rightarrow_{\varrho} q \in C_{g,o}$:*

- *a fact $\neg\varrho$ (an undercutter)*
- *a fact $\neg p$ (an underminer)*
- *a fact $\neg q$ (a rebuttal)*

A set of possible conflicts $\bar{C}_{g,o}$ thus contains facts that can be used to generate counter-arguments to arguments formed using $C_{g,o}$. Note that no rule weights are used in both chains and conflict set. Therefore an attack between two arguments as formed from these rules will always be result in defeat. Although this is a simplification of the complex knowledge of real world deliberation problems, it nevertheless allows for structured arguments and counter arguments and, as demonstrated later, can be sufficiently complex to generate interesting dialogues.

Consider again the example chain $C_{g_2,o_1} = \{o_1 \Rightarrow_{\varrho 1} p_5, p_5 \Rightarrow_{\varrho 2} p_2, p_2 \Rightarrow_{\varrho 3} g_2\}$ then there is a set of possible conflicts

$$\bar{C}_{g_2,o_1} = \{\neg\varrho_1, \neg p_5, \neg\varrho_2, \neg p_2, \neg\varrho_3\}.$$

Finally, the beliefs of a role are defined by considering the role's options and generating either a chain or set of conflicts for each of them.

Definition 13. *To every pair of an option $o \in \bigcup_{r\in\mathcal{R}} O_r$ and role $r \in \mathcal{R}$, where $o \in O_r$, a unique set of role-option beliefs $B_r^o = C_{g,o} \cup Cg_d, o$ is assigned by selecting a goal $g \in G_r$.*
To every pair of an option $o \in \bigcup_{r\in\mathcal{R}} O_r$ and role $r \in \mathcal{R}$, where $o \notin O_r$, a unique set of role-option beliefs $B_r^o \subseteq \bar{C}_{g,o}$ is assigned by selecting a goal $g \in L_g$ such that $|B_r^o| = n_{B_r^o}$.

Table 4. Belief assignment for the example roles

$l = 3$	(chains with length 3)	
$n_{B_r^o} = 2$	(full chain or 2 negated beliefs)	
		r_1
	$B_{r_1}^{o_1}$	$o_1 \Rightarrow_{\varrho 1} p_5, p_5 \Rightarrow_{\varrho 2} p_2, p_2 \Rightarrow_{\varrho 3} g_2,$
		$o_1 \Rightarrow_{\varrho 4} p_6, p_6 \Rightarrow_{\varrho 5} p_4, p_4 \Rightarrow_{\varrho 6} g_d$
	$B_{r_1}^{o_2}$	$o_2 \Rightarrow_{\varrho 7} p_5, p_5 \Rightarrow_{\varrho 2} p_2, p_2 \Rightarrow_{\varrho 8} g_1,$
		$o_2 \Rightarrow_{\varrho 9} p_9, p_9 \Rightarrow_{\varrho 10} p_1, p_1 \Rightarrow_{\varrho 11} g_d$
	$B_{r_1}^{o_3}$	$\neg\varrho_{17}, \neg p_3$
		r_2
	$B_{r_2}^{o_1}$	$\neg p_2, \neg\varrho_3$
	$B_{r_2}^{o_2}$	$o_2 \Rightarrow_{\varrho 9} p_9, p_9 \Rightarrow_{\varrho 12} p_8, p_8 \Rightarrow_{\varrho 13} g_4,$
		$o_2 \Rightarrow_{\varrho 14} p_1, p_1 \Rightarrow_{\varrho 15} p_9, p_9 \Rightarrow_{\varrho 16} g_d$
	$B_{r_2}^{o_3}$	$o_3 \Rightarrow_{\varrho 17} p_7, p_7 \Rightarrow_{\varrho 18} p_3, p_3 \Rightarrow_{\varrho 19} g_4,$
		$o_3 \Rightarrow_{\varrho 17} p_7, p_7 \Rightarrow_{\varrho 21} p_8, p_8 \Rightarrow_{\varrho 22} g_d$

Hence, in case a certain option was in the role's option set, it is assigned a chain to the mutual goal as well as a rule chain to one of its personal goals G_r. For the purpose of this paper an arbitrary goal g is selected from G_r. In case a certain option was not in the role's option set it is not given a rule chain but is instead assigned a set of possible conflicts. For this paper, this is the conflict set of a chain as created for some other agent that did have the option. In other words, it assigns (a subset of) the negated facts that are associated with a chain of another agent. This ensures that possible conflicts relate directly to the rules as generated in the other agents' chains, which makes it possible to construct counter arguments later in the dialogue.

The variable $n_{B^o_{\bar{r}}}$ can be used to control the number of generated negated beliefs. Table 4 shows all the role-option beliefs for the roles in the running example, as selected from generated rule chains.

Agent knowledge allocation In this paper each agent is assumed to have its own set of ASPIC inference rules; for notational convenience these are below regarded as part of the agents' beliefs. Agents inherit options, goals and beliefs from their respective roles. In addition, as is typical for deliberation problems, agents also have beliefs and personal goals not affiliated with their role. Additionally, agents may miss some part of the full body of knowledge as coming from their roles.

Assignment of agent knowledge starts by taking the options and goals from the agent's role, plus a set of additional personal goals. The variable $n_{G^{\bar{r}}_a}$ will be used to set the number of non-role goals allocated to an agent.

Definition 14. *An agent $a \in \mathcal{A}$ with role r and a set size $n_{G_{\bar{r}}}$ has:*

- *A set of* options $O_a = O_r$
- *A set of* non-role originating goals $G^{\bar{r}}_a$*, where for every $g \in G^{\bar{r}}_a$ it holds that $g \in L_g \setminus G_r$ and such that $|G^{\bar{r}}_a| = n_{G^{\bar{r}}_a}$*
- *The combined set of* goals $G_a = G_r \cup G^{\bar{r}}_a$

The beliefs of an agent are both role-originating and non-role originating. The role-originating beliefs set is a subset of all role-option beliefs as generated for each of the options in the scenario, where variable $n_{B^r_a}$ controls the size of the part that is allocated.

Definition 15. *An agent $a \in \mathcal{A}$ with some role r is assigned a set of $n_{B^r_a}$* role-originating beliefs

$$B^r_a \subseteq \bigcup_{o \in O_K} B^o_r \textit{ such that } |B^r_a| = n_{B^r_a}$$

Since no agent is assigned full knowledge, an agent is likely to miss some rule needed to construct a full argument. This is a characteristic of deliberation problems. On the other hand, the missing knowledge will be partially undone by assigning additional, personal knowledge to the agent, the non-role originating beliefs. These beliefs can come from various sources, such as an agent's expertise or prior encounters. It is modelled here as a set of rules taken from newly generated chains for some of the agent's options and goals, in the same way as how chains are generated for roles. The variable $n_{B^{\bar{r}}_a}$ is used to set the number of non-role originating beliefs known to the agent.

Definition 16. *An agent $a \in \mathcal{A}$ is, given set size $n_{B^{\bar{r}}_a}$, assigned a unique set* non-role originating beliefs

$$B^{\bar{r}}_a \subseteq \bigcup_{o \in O_a} C_{g,o} \textit{ for some selected goal } g \in G_a \textit{ such that } |B^r_a| = n_{B^{\bar{r}}_a}$$

Which goal is selected is determined by some selection function, but for this paper an arbitrary goal is used form the set G_a.

The total set of beliefs is the union of role and non-role originating beliefs.

Definition 17. *An agent $a \in \mathcal{A}$ is assigned a set of* beliefs $B_a = B_a^r \cup B_a^{\bar{r}}$

Table 5. Knowledge allocation for example agents

	(Agents a_1 and a_2 have role r_1; agent a_3 has role r_2)			
$n_{G_a^{\bar{r}}} = 1$	(Agents inherit goals, g_d and get one non-role goal)			
	O_{a_1}	o_1, o_2	G_{a_1}	g_d, g_1, g_2, g_4
	O_{a_2}	o_1, o_2	G_{a_2}	g_d, g_1, g_2, g_3
	O_{a_3}	o_2, o_3	G_{a_3}	g_d, g_3, g_4, g_2
$n_{B_a^r} = 13$	(Agents inherit 13 of their 14 role beliefs)			
$n_{B_a^{\bar{r}}} = 2$	(And get 2 non-role beliefs)			
	B_{a_1}	$o_1 \Rightarrow_{\varrho 1} p_5,\ p_5 \Rightarrow_{\varrho 2} p_2,\ p_2 \Rightarrow_{\varrho 3} g_2,$		
		$o_1 \Rightarrow_{\varrho 4} p_6,\ p_4 \Rightarrow_{\varrho 6} g_d,$		
		$o_2 \Rightarrow_{\varrho 7} p_5,\ p_5 \Rightarrow_{\varrho 2} p_2,\ p_2 \Rightarrow_{\varrho 8} g_1,$		
		$o_2 \Rightarrow_{\varrho 9} p_9,\ p_9 \Rightarrow_{\varrho 10} p_1,\ p_1 \Rightarrow_{\varrho 11} g_d,$		
		$\neg\varrho_{17}, \neg p_3,$		
		$o_1 \Rightarrow_{\varrho 23} p_2,\ p_3 \Rightarrow_{\varrho 19} g_4$		
	B_{a_2}	$p_5 \Rightarrow_{\varrho 2} p_2,\ p_2 \Rightarrow_{\varrho 3} g_2,$		
		$o_1 \Rightarrow_{\varrho 4} p_6,\ p_6 \Rightarrow_{\varrho 5} p_4,\ p_4 \Rightarrow_{\varrho 6} g_d,$		
		$o_2 \Rightarrow_{\varrho 7} p_5,\ p_5 \Rightarrow_{\varrho 2} p_2,\ p_2 \Rightarrow_{\varrho 8} g_1,$		
		$o_2 \Rightarrow_{\varrho 9} p_9,\ p_9 \Rightarrow_{\varrho 10} p_1,\ p_1 \Rightarrow_{\varrho 11} g_d,$		
		$\neg\varrho_{17}, \neg p_7,$		
		$o_2 \Rightarrow_{\varrho 25} p_2,\ o_1 \Rightarrow_{\varrho 25} p_5$		
	B_{a_3}	$\neg\varrho_4,$		
		$o_2 \Rightarrow_{\varrho 9} p_9,\ p_9 \Rightarrow_{\varrho 12} p_8,\ p_8 \Rightarrow_{\varrho 13} g_4,$		
		$o_2 \Rightarrow_{\varrho 14} p_1,\ p_1 \Rightarrow_{\varrho 15} p_9,\ p_9 \Rightarrow_{\varrho 16} g_d,$		
		$o_3 \Rightarrow_{\varrho 17} p_7,\ p_7 \Rightarrow_{\varrho 18} p_3,\ p_3 \Rightarrow_{\varrho 19} g_4,$		
		$o_3 \Rightarrow_{\varrho 17} p_7,\ p_7 \Rightarrow_{\varrho 21} p_8,\ p_8 \Rightarrow_{\varrho 22} g_d,$		
		$p_8 \Rightarrow_{\varrho 26} p_7,\ p_2 \Rightarrow_{\varrho 3} g_2$		

Table 5 shows the final knowledge bases of the three agent in the running example. Note that the rules of several chains have been allocated fully to the agents, which now allows them to construct arguments with them. On the other hand, some rules that were part of a chain have not been assigned (see for example the second line in agent a_1's beliefs) while additional rules are added that do not originate in the role of the agent but which is personal knowledge (shown on the last line of every agent's beliefs).

Throughout the generation process a number of variables, listed in Table 6, are used to control the size of the different sets. Since the generation process is used to create scenarios reflecting typical deliberation problems, the variable settings should be chosen in such a way that the scenarios motivate agents to propose options, provide

Table 6. Input parameters used in the scenario generation process

		experiment setting
$n_{\mathcal{A}}$	The number of agents	4
$n_{\mathcal{R}}$	The number of roles	5
n_{O_r}	A role r's options set size	4
n_{G_r}	A role r's goals set size	5
n_S	The chaining seedset size	20
l	The length of rule chains	3
$n_{B^o_{\bar{r}}}$	An agent a's negated role-option beliefs set size	4
$n_{G^{\bar{r}}_a}$	An agent a's non-role originating goals set size	0
$n_{B^r_a}$	An agent a's role-originating beliefs set size	25
$n_{B^{\bar{r}}_a}$	An agent a's non-role originating beliefs set size	6

motivations and attack using counter-arguments. In [9] we used a software experiment to establish suitable settings by repeatedly generating scenarios and in turn measuring how many arguments and counter-arguments an agent can potentially create. Using multiple linear modelling a model was be created from which the ideal variable settings can be inferred. For the experiment of this paper such a model was created and the variables were set to maximize the potential arguments between agents. Table 6 list the parameter settings used in the experiment. Note that these differ slightly from [9] because of a corrected scenario generation model.

2.4 Comparing strategies

The final thing to specify is the dialogue strategy used by an agent, that is, how an agent selects from the moves allowed for by the protocol. The possibilities for strategies are almost endless and form a fascinating topic by itself, but for the purpose of this paper we can do with two fairly basic ones: one that uses argumentation in the dialogue to attack and defend claims and one that does not argue in the dialogue itself. What they share is the internal reasoning method, where both use the argumentation logic to evaluate options and claims. In order to determine which options are beneficial for it, an agent uses the notion of a defensible option. This is an option for which, considering its knowledge, the agent can construct a defensible argument. Defensible arguments ensure a credulous reasoning process, since you accept a claim if you have no stronger counter argument. Reasoning over action has in the literature been associated with credulous reasoning, while epistemic reasoning with facts has been linked to sceptical reasoning [12].

Definition 18. *An agent a's option $o \in O_a$ is a B-g-*defensible option *from g if, on the basis of the beliefs $B_a \cup \{o\}$, an argument $A \mid\!\sim g$ can be constructed for some goal $g \in G_a$ such that $o \in A$.*

Consider the agent a_1 from the example scenario introduced above. Option o_2 is a B-g-defensible option from g_1 since it can construct a defensible argument $o_2, o_2 \Rightarrow_{\varrho 7} p_5, p_5 \Rightarrow_{\varrho 2} p_2, p_2 \Rightarrow_{\varrho 8} g_1 \mid\!\sim g_1$ based on $B_a \cup \{o_2\}$.

During the dialogue the agents maintain a model of the dialogue and evaluate the dialogical status of moves to determine the next move to play. From this it knows what options have been proposed, which together with its own options forms the set of options that it will play moves about. The agent will propose, attack and defend options depending on the extent to which they are beneficial for the agent. This is based in the utility that an agent has for its goals.

Definition 19. *Every goal $g \in G_a$ for agent a is assigned a* goal utility $U_a^g \in \{1, \ldots, |G_a|\}$.

For the purpose of this paper the agent assigns an increasing utility for every of its goals from a utility of 1 to the number of goals. Even this simple allocation causes enough variation in the utility of the agent's goals. This in turn is used to give a valuation to a specific option by summing the utilities of those goals for which a defensible argument could be constructed, i.e. those goals that make it a B-g-justified option.

Table 7. Goal utilities, arguments for B-g-justified options and heuristics for the example agents

	a_1	
$U_{a_1}^{g_d} = 3$	$B \cup \{o_1\} \mathrel{|\!\sim} g_2$	$H_{a1}^{o_1} =$ `build`
$U_{a_1}^{g_1} = 2$	$B \cup \{o_2\} \mathrel{|\!\sim} g_1$	$H_{a1}^{o_2} =$ `build`
$U_{a_1}^{g_2} = 4$	$B \cup \{o_2\} \mathrel{|\!\sim} g_d$	$H_{a1}^{o_3} =$ `destroy`
$U_{a_1}^{g_4} = 1$		
	a_2	
$U_{a_2}^{g_d} = 2$	$B \cup \{o_1\} \mathrel{|\!\sim} g_d$	$H_{a2}^{o_1} =$ `build`
$U_{a_2}^{g_1} = 1$	$B \cup \{o_2\} \mathrel{|\!\sim} g_1$	$H_{a2}^{o_2} =$ `build`
$U_{a_2}^{g_2} = 2$	$B \cup \{o_2\} \mathrel{|\!\sim} g_2$	$H_{a2}^{o_3} =$ `destroy`
$U_{a_2}^{g_4} = 3$	$B \cup \{o_2\} \mathrel{|\!\sim} g_d$	
	a_3	
$U_{a_3}^{g_d} = 4$	$B \cup \{o_2\} \mathrel{|\!\sim} g_4$	$H_{a3}^{o_1} =$ `destroy`
$U_{a_3}^{g_1} = 1$	$B \cup \{o_2\} \mathrel{|\!\sim} g_g$	$H_{a3}^{o_2} =$ `build`
$U_{a_3}^{g_2} = 3$	$B \cup \{o_3\} \mathrel{|\!\sim} g_4$	$H_{a3}^{o_3} =$ `build`
$U_{a_3}^{g_4} = 2$	$B \cup \{o_3\} \mathrel{|\!\sim} g_d$	

Definition 20. *For every option $o \in O_a \cup Q_d$ in dialogue d an agent a assigns an* option heuristic $H_{d,a}^o = \texttt{build}$ *iff*

$$(\sum_{g \in G_a} U_a^g \textit{ if } o \textit{ is a B-g-defensible option}) > 0;$$

otherwise $H_{d,a}^o = \texttt{destroy}$.

Table 7 shows the utilities assigned to each of the goals for an agent on the left. The arguments making B-g-justified options are shown in the middle, from which the option heuristic can be assigned.

The final step in the strategy is to select a move; one that is legal according to the protocol $\mathcal{P}$.

Definition 21. *An agent a in dialogue d had a* move generation function $\mathcal{G}_a : D \times L_g \times Pow(L_b) \times Pow(L_g) \times Pow(L_o) \Rightarrow L_c$ *mapping the current dialogue, the mutual goal* g_d *and the agent's beliefs, goals and options to a new move m.*

Arguing strategy The arguing agents of this paper use a simple but expressive strategy. Options for which it has a `build` heuristic can be proposed in a straightforward manner. Playing moves in existing proposal trees is done when it has a `build` or `destroy` heuristic for the option proposed in that tree. First the set of *active attackers* is established, which are the moves that, if attacked, will flip the status of the tree's proposal move from *in* to *out* or from *out* to *in*. The agent can then try to build a new reply to one of the active attackers by finding (counter-)arguments to the used claims. Moves with sceptically acceptable arguments to the attacked move's claim or questioned claim are called B-justified argue moves. Here, sceptical acceptance is used because the agent reasons about beliefs instead of actions. The same approach is used by Amgoud and Prade [1].

Definition 22. *Agent a in dialogue d has a* B-justified argue move argue$(A \mid\!\sim p)$ *for option* $o \in O_a$ *if, on the basis of beliefs* $B_a \cup \{o\}$*, a justified argument* $A \mid\!\sim p$ *can be constructed.*

Algorithm 1 defines the arguing agent's move generation strategy $\mathcal{G}_a$. Replying in an existing proposal tree is mostly straightforward when the active attackers are known. For every move the corresponding attacker from Table 1 can be used. In case of argue moves the agent prefers to play a counter-argument instead of questioning with a *why* move. Finally, although sometimes many different arguments can be played to attack some claim the agent will pick the first B-justified argue move it can form without considering whether other arguments might have been better. This is one of the many possible future research paths in strategy design for argumentative agents. Figure 2 show the proposal tree for option o_1 in our running example if the agents play the arguing strategy.

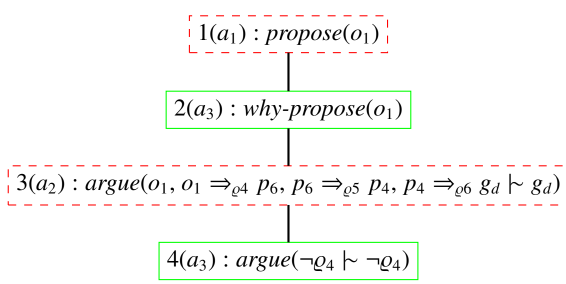

Fig. 2. Proposal tree of o_1 in the running example

Algorithm 1 The arguing agent move selection algorithm

Input: dialogue d, agent a

for all $o \in O_a \cup Q_d$ **do**
  if $o \notin Q_d$ **and** $H^q_{d,a} = build$ **then**
    return $propose(q)$
  else if $o \in Q_d$ **and** $H^q_{d,a} = build$ **or** $H^q_{d,a} = \texttt{destroy}$ **then**
    {Loop through all moves that are 'actively attacking' the proposal}
    for all $m \in getActiveAttackers(\emptyset, propose(q), \top, d)$ **do**
      if $m = propose(o)$, m is *in* **and**
      $why\text{-}propose(o) \notin d$ **then**
        return $why\text{-}propose(o)$s
      else if $m = argue(A \mid\!\sim p)$,
      B-justified argue move $argue(B \mid\!\sim p') \notin d$ **and**
      $B \mid\!\sim p'$ defeats $A \mid\!\sim p$ **then**
        return $argue(B \mid\!\sim p')$
      else if $m = why\text{-}propose(o)$ **and**
      B-justified argue move $argue(A \mid\!\sim g_d) \notin d$ where $o \in A$ **then**
        return $argue(A \mid\!\sim g_d)$
      else if $m = why(p)$ **and**
      B-justified argue move $argue(A \mid\!\sim p) \notin d$ **then**
        return $argue(A \mid\!\sim p)$
      end if
    end for
  end if
end for
return *skip*

Algorithm 2 The *getActiveAttackers* algorithm

Input: attackers set *att*, move m, if parent is attacker *par*, dialogue d

if $m = propose(q)$ or m is an attacking move **then**
  if m is *in* **then**
    {Include moves that are *in*}
    $att = att \cup \{m\}$
    for all $m' \in d$ where $\texttt{target}(m') = m$ **do**
      $getActiveAttackers(att, m', \top, d)$
    end for
  end if
else if *par* **then**
  {If this move's target was *in*, also look though its attackers}
  for all $m' \in d$ where $\texttt{target}(m') = m$ **do**
    $getActiveAttackers(att, m', \bot, d)$
  end for
end if
return *att*

Non-arguing strategy To compare the arguing strategy to one where the agents cannot argue in the dialogue, a simple strategy is introduced that can only propose and reject options. It still evaluates options in the dialogues using utilities for its goals, but it cannot question, attack and defend claims in the dialogue, effectively excluding the argumentation part of the deliberation dialogue. Algorithm 3 defines the non-arguing agent's move generation strategy $\mathcal{G}_a$.

Algorithm 3 The non-arguing agent move selection algorithm

Input: dialogue d, agent a

for all $o \in O_a \setminus Q_d$ **do**
    if $H^q_{d,a} = build$ **then**
        return $propose(q)$
    end if
end for
for all $o \in Q_d$ **do**
    if $H^q_{d,a} = \texttt{destroy}$ **and** $why\text{-}propose(o) \notin d$ **then**
        return $why\text{-}propose(o)$
    end if
end for
return $skip$

3 Experimental evaluation

In the preceding sections a deliberation model was presented that allows agents to engage in a deliberation dialogue as described by a generated scenario. To study the use of argumentation in such dialogues, an arguing and a non-arguing agent strategy was proposed. In the present section an experiment is described in which dialogues as generated by agents according to a scenario were evaluated of efficiency and effectiveness.

Efficiency pertains to how quickly the agents in the dialogue discuss all options. As in previous work [7] the metric used is to simply count the number of moves in the dialogue.

Definition 23. *The* move efficiency *of a dialogue d is measured by*

$$f_d = |d|$$

Effectiveness is measured by the shared utility of the dialogue outcome considering all agents. As in [7, 10] the utility that every agent itself assigns to the dialogue outcome is summed to form the utility.

Definition 24. *The* total utility *of dialogue d is measured by*

$$v_d = \sum_{a \in A} \sum_{g \in G_a} U^g_a \textit{ if } O_d \textit{ is a justified option for a}$$

Efficiency experiment A software experiment was conducted in which repeatedly scenarios were generated and played by one of the two strategies. The efficiency and effectiveness of every dialogue was then measured and the averages over all scenarios were compared. Every experiment consisted of 1000 played scenarios.

The average efficiency, the number of dialogue moves, of the arguing and non-arguing strategies is shown in Figure 3. Clearly the average number of moves when arguing ($f_d \approx 26$) is much higher than when the agents do not argue ($f_d \approx 14$). This is simply because all the non-arguing agents do is propose or reject options, while the arguing agents actually discuss claims. While argumentation in theory may prevent unnecessary moves, improving efficiency, this is clearly not true for this model of this paper.

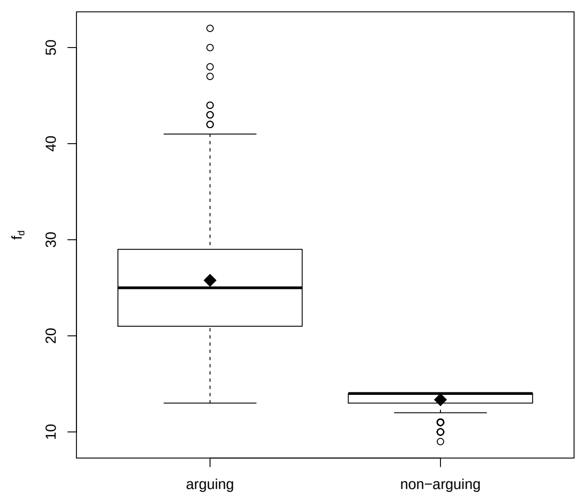

Fig. 3. Efficiency f_d of the arguing and non-arguing strategies, with average ◆

Effectiveness experiment The benefit of argumentation should become apparent by measuring the effectiveness of the dialogues with arguing agents against those with non-argumentative agents. The effectiveness, the total utility the agents have for the dialogue outcome, is shown in Figure 4. Clearly, the average effectiveness is much higher ($f_d \approx 10$) for the arguing strategy than for the non-arguing strategy ($f_d \approx 5$). With the non-arguing agents proposals will only go out if a reject move is played, but there is almost always some agent who does not agree with a proposal. In effect in a lot of dialogue the agents will reject all proposals, leaving no dialogue outcome and hence a utility of 0. Because the arguing agents can move arguments giving a motivation, they can make such attacked proposal *in* and available to select as dialogue outcome.

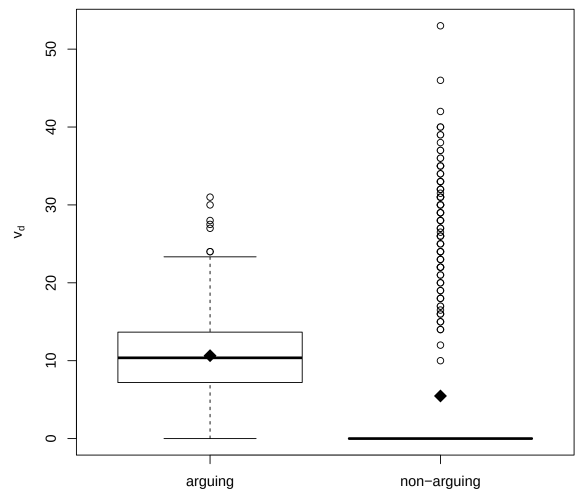

Fig. 4. Effectiveness v_d of the arguing and non-arguing strategies, with average ◆

Baseline effectiveness One might argue that non-arguing agents should just propose options and never reject any. With no rejections, all options are *in* and thereby preventing that no options can be selected as dialogue outcome at all. To study this, a comparison was made between the arguing agents and a baseline strategy in which a random option is selected from all that are available instead of engaging in a discussion. The result, as shown in Figure 5, shows that indeed this baseline strategy will return a similar (with no statistical difference) average effectiveness. It seems than that arguing in deliberation dialogues might not be beneficial after all, but there is still a difference in the way the results came to be. The arguing strategy empowers rational and self-interested agents and the dialogues these agents produce contain useful information like which proposals were clearly not the right choice and giving insight in what the right choice might be. Further research is needed to investigate how this additional information can best be utilized. One idea is to adjust the outcome selection process to take into account all arguments; another is to introduce belief revision which allows agents to adopt new knowledge during the game through argumentation. Interestingly, Karunatillake at al. [7] have found similar results where arguing about actions may not provide the clear benefits over unbounded obedience.

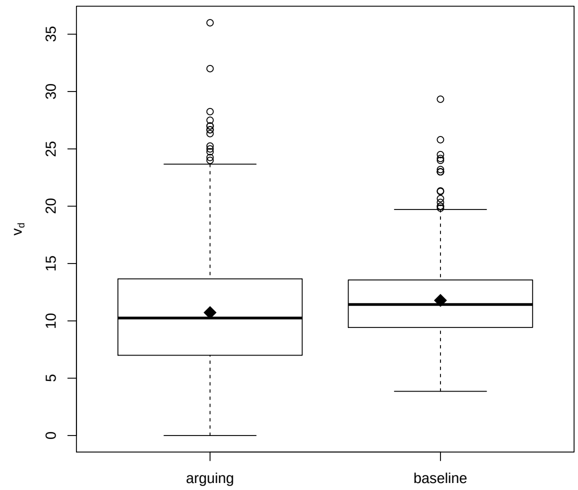

Fig. 5. Effectiveness v_d of the arguing strategy versus a random option selection, with average ◆

4 Conclusions

Existing work on the experimental evaluation of argumentation in agent dialogues makes use of very simple models of argumentation, in which arguments have no or little structure. This paper has advanced the state of the art by carrying out an experimental evaluation with arguments that have considerably more structure and that can be attacked in three ways. Agents can reason using options, goals and utilities and can attack dialogue arguments. Our study partly confirms findings of earlier work [7, 10, 4] that the use of argumentation in inter-agent dialogues may be beneficial to the agents. It was shown that arguing agents achieve a higher average shared utility than non-arguing ones. Further research is required to find whether a certain strategy can also outperform the baseline strategy.

A second contribution of our paper is a methodology for carrying out evaluation experiments using inter-agent dialogues with structured arguments. Since this kind of research is still rare a new method needed to be developed, which is based on the generation process for realistic scenarios (presented in further detail in [9]) and a strategy model for goal-directed agents, with the aim to support future experimental research. This may include other metrics for deliberation such as the amount of options covered and beliefs used or whether the outcome is Pareto optimal given the agents' preferences.

While the argumentation and deliberation model used in this paper is considerably more expressive than in previous evaluation experiments, it is still rather simple compared to the most sophisticated formal dialogue models of argumentation in the literature. [11, 1] We want to pursue different lines of research into more sophisticated models, such as the adoption in the topic language of argument schemes for practical reasoning [3, 2], improving the strategy model [6], enable agents to revise beliefs and adding an outcome selection process using preference-based argumentation.

Acknowledgments This research was supported by the Netherlands Organisation for Scientific Research (NWO) under project number 612.066.823.

References

1. L. Amgoud and H. Prade. Using arguments for making and explaining decisions. *Artificial Intelligence*, 173(3-4):413–436, 2009.
2. K. Atkinson, T. J. M. Bench-Capon, and P. McBurney. A dialogue game protocol for multi-agent argument over proposals for action. *Autonomous Agents and Multi-Agent Systems*, 11(2):153–171, 2005.
3. K. Atkinson, T. J. M. Bench-Capon, and P. McBurney. Arguing about cases as practical reasoning. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, pages 35–44, Bologna, Italy, 2005. ACM Press.
4. E. Black and K. Bentley. An empirical study of a deliberation dialogue system. In *Proceedings of the 1st International Workshop on the Theory and Applications of Formal Argumentation*, Barcelona, Spain, 2011.
5. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–357, 1995.
6. A. Kakas, N. Maudet, and P. Moraitis. Layered strategies and protocols for argumentation-based agent interaction. In *Proceedings of the 1st International Workshop on Argumentation in Multi-Agent Systems*, pages 64–77, New York, USA, 2004.
7. N. C. Karunatillake, N. R. Jennings, I. Rahwan, and P. McBurney. Dialogue games that agents play within a society. *Artificial Intelligence*, 173(9-10):935–981, 2009.
8. E. M. Kok, J.-J. C. Meyer, H. Prakken, and G. A. W. Vreeswijk. A Formal Argumentation Framework for Deliberation Dialogues. In P. McBurney, S. Parsons, and I. Rahwan, editors, *Proceedings of the 7th International Workshop on Argumentation in Multi-Agent Systems*, Berlin, Germany, 2010. Springer-Verlag.
9. E. M. Kok, J.-J. C. Meyer, H. van Oostendorp, H. Prakken, and G. A. W. Vreeswijk. A Methodology for the Generation of Multi-Agent Argumentation Dialogue Scenarios. In *9th European Workshop on Multi-agent Systems*, Maastricht, The Netherlands, 2011.
10. P. Pasquier, R. Hollands, I. Rahwan, F. Dignum, and L. Sonenberg. An empirical study of interest-based negotiation. *Autonomous Agents and Multi-Agent Systems*, 22(2):249–288, 2010.
11. H. Prakken. Coherence and Flexibility in Dialogue Games for Argumentation. *Journal of Logic and Computation*, 15(6):1009–1040, 2005.
12. H. Prakken. Combining sceptical epistemic reasoning with credulous practical reasoning. In P. Dunne and T. Bench-Capon, editors, *Proceedings of COMMA-06*, pages 311–322. IOS Press, 2006.
13. H. Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.
14. I. Rahwan. Guest editorial: Argumentation in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 11(2):115–125, 2005.
15. M. South and G. A. W. Vreeswijk. ASPIC Java Components. http://aspic.cossac.org/.