A Methodology for the Generation of Multi-Agent Argumentation Dialogue Scenarios

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

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

Abstract. Increasingly research into the uses of argumentation in multi-agent dialogues takes an experimental approach. Such studies explore how agents can successfully employ argumentation besides the best and worst case situations of formal analysis. While a vital part in these experiments is influenced by the scenarios from which dialogues are generated, there is very little research on how these can be generated in a meaningful way, respecting the characteristics of the underlying dialogue problem. This paper proposes, by means of an example system for deliberation dialogues, a methodology for the construction and evaluation of a scenario generation process. It is shown how scenarios can accommodate argumentation with structured arguments and how it is tested whether the generated scenarios are interesting for experimentation.

1 Introduction

One of the key components of multi-agent systems is communication and in recent years systems have been developed to model agent communication using argumentation-enabled dialogue systems. Such systems promise to be more efficient in bringing agents to a decision and yield better agreements. [10]

While formal studies have indeed provided insights into theoretical reachability of ideal and intuitive conclusions, more and more research is looking to empirically investigate benefits to overcome the often strong limitations that are introduced in formal studies and to find results besides best and worst case situations. Through a software experiment unique situations, called scenarios, are generated. Agents are run to construct a dialogue from the scenario, after which the efficiency and effectiveness of the dialogue is measured. Inherently this specific scenario directly influences the results and it is therefore crucial that it is generated in a meaningful fashion. However, very little research has focussed on how scenarios can be generated while respecting the characteristics of the underlying dialogue problem.

This paper, being part of a project taking the experimental approach to finding the use of argumentation for agents, will provide a methodology to generate scenarios in a structured way. An example dialogue system with BDI-based agents is introduced for which scenarios are generated. Several desirable metrics

are introduced by which it is shown how the most influential input parameters of the system can be found.

2 Background on experimentation in argumentation-enabled dialogues

Computational argumentation is roughly divided in the areas of argumentation logics and dialogical argumentation. The former pertains to the evaluation of an argumentation system as to identify the justified arguments while the latter usually makes use of such logics but studies the effects of arguments in agent dialogues instead. Agents engage in various types of dialogues where argumentation is useful; notably, as classified by Walton and Krabbe [12], negotiation, persuasion and deliberation.

2.1 Existing work

Several works explored the practical benefits of argumentation in dialogues. In both Karunatillake et al. [3] and Paquier et al. [8] argumentation-based negotiation is studied. While the focus of the systems is different (social agent societies and exploring the negotiation space respectively) the argumentative parts of agents are modelled alike. Within a dialogue agents may ask for and supply a motivation behind proposals. However, the language does not allow agents to build structured arguments, which severely limit the expressivity of agents.

Black and Bentley [1] empirically investigated the use of argumentation in two-party deliberation dialogues. Agents are initialized with a set of abstract 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 number of arguments, values and actions.

Ontañón and Plaza [6] experiment with two-agent inductive learning dialogues using structured arguments. Evaluation is performed using examples from an existing repository, in contrast to generating new scenarios for the agents.

2.2 Characteristics of deliberation dialogues

While the existing work already points towards useful applications, there has not yet been a project that uses argumentation with structured arguments, formed from an agents beliefs and goals. This paper shows a methodology to generate scenarios that accommodate for argumentation with structured arguments while strongly reflecting the characteristics of deliberation dialogue type as identified from the existing literature on argumentation-based dialogues [1, 4, 5]:

- Mutual deliberation goal
- Unequal roles between agents
- Not all options (possible actions) are known by all agents

- Both compatible and conflicting goals between agents
- Incomplete information and from different sources

The scenario generation process that is now introduced is grounded in and supports these characteristics. A similar analysis of the characteristics of the underlying problem structure should be performed for any dialogue type that is under experimentation.

2.3 Generating interesting scenarios for experimentation

Scenarios generated using the methodology proposed in this paper will reflect the unique features of the deliberation dialogue as just identified. However, we also need to evaluate whether the generated scenarios will be interesting enough to use in argumentation dialogue experiments. That is, the scenarios need to provide incentive to the deliberation process. When testing argumentation in deliberation dialogues it is desired that agents can indeed make proposals for some action, forward (counter-)arguments and question statements. Scenarios can be interesting in two ways: to which degree it allows for the justification of doing some action and to which degree proposals for these actions can be attacked. How to test scenarios on these qualities will be discussed in section 5.

3 Deliberation model

To facilitate scenarios with the complexity of the deliberation dialogues a model is now introduced that combines a structured argumentation logic with a multi-agent deliberation dialogue system. This forms the example system for which a scenario generation process is modelled and evaluated.

3.1 Argumentation logic

Arguments in the deliberation dialogues are formed using a simple instantiation of the abstract ASPIC framework for argumentation with structured arguments [9], which is an instance of the Dung [2] abstract argumentation model. It allows agents to create structured arguments, 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 on the arguments the status of an argument can be evaluated.

In this paper a simple instantiation of the ASPIC 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 $A \hspace{1pt}\mid\hspace{-3pt}\sim p$ where A is the set of used premises and inferences and p is the conclusion. Extended versions of this simple instantiation will be studied in later work.

3.2 Dialogue context

The deliberation dialogues consist of a series of moves that, except for proposals, always reply to a previous move. In these moves the agents make proposals, question statements, provide arguments or surrender to a previous statement. Although the full dynamics of an argumentation-based deliberation dialogue model are not needed for this paper, it is still good to briefly cover the interplay between agents, arguments and the proposals. The multi-agent deliberation model here is a simplification from that of Kok et al. [4].

First of all the dialogue takes place in a deliberation dialogue context.

Definition 1. A deliberation dialogue context consists of:

- An ASPIC *argumentation system* $\mathcal{L}$
- A *topic language* L_t consisting of
  - options $L_o \subseteq L_t$
  - goals $L_g \subseteq L_t$
  - beliefs $L_b \subseteq L_t$
- A mutual deliberation goal $g_d \in L_g$

Agents engage in a dialogue and make proposals for action respecting a mutual goal describing their shared interest, like making profit. Reasons for submitting proposals can in turn be requested to which arguments can be forwarded showing how some goal is achieved by performing the proposed action. From there the agents question or surrender to premises and provide arguments.

Agents take turns and every turn they may submit multiple moves to the dialogue. Except the *propose* move every move has a specific target that it attacks or to which it surrenders. In this way multiple distinct proposal trees can be constructed by the agents. When no agent makes any more moves the dialogue ends and the winning proposal can be determined according to some heuristic. Figure 1 show an example in which a proposed option o is supported by two arguments with a goal as conclusion, one of which is has a counter-argument causing agent b to retract his earlier claim.
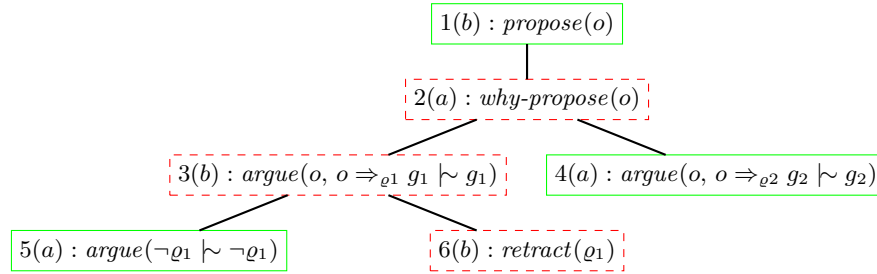

Fig. 1: Example of a proposal tree with two arguments from option o

3.3 Agent model

Every participating agent has a certain role in the system, such as a topic expert or nancial agent. This role describes the duties and desires of an agent as being a part of its context.

Definition 2. A set of roles $\mathcal{R}$ is defined where every role $r \in \mathcal{R}$ in a deliberation context with mutual goal g_d and set sizes n_{O_r} and n_{G_r} 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 $|G_r| = n_{G_r}$

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 vary the number of options and goals associated with a role.

Internally the agents are modelled using the influential BDI architecture [11], which is also a natural way of designing argumentation-enabled agents. [7] An agent maintains a set of goals that it wants to bring about and a set of beliefs consisting of facts and rules. For deliberation systems this is extended with a set of options, which are the actions that an agent knows it can propose to solve the deliberation problem at hand.

Definition 3. A set of agents $\mathcal{A}$ is defined where every agent $a \in \mathcal{A}$ has:

- A role $r \in \mathcal{R}$
- A set of *options* $O_a \subseteq L_o$
- A set of *goals* $G_a \subseteq L_g$
- A set of *beliefs* $B_a \subseteq L_b$

4 Scenario generation

As established above, generated scenarios will need to support full structured argumentation using the agents' beliefs and goals. Options in a deliberation problem typically exist because, through some line of reasoning, they promote the achievement of a goal, as supported by an argument for the option. This property forms the idea behind the five-step process for the generations of scenarios.

To illustrate the generation process a running example is used. Three agents will participate in a deliberation dialogue with mutual goal g_d. These agents are split into two different roles, both of which are assigned two options and two goals, as visualized in Table 1.

4.1 Rule chaining

The first step in generating scenarios is to create a body of knowledge that, as is typical in deliberation problems, forms lines of reasoning between options and the goals that they promote. These are called rule chains and will 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.

Table 1: Example dialogue with three agents

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

Definition 4. 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 consequence to all other rules is an atom from the chaining seed set and in turn is the antecedent for the follow-up rule. Although chains of rules with only one positive antecedent may seem restricted, it will already support a sufficiently complex scenario as will be shown in Section 5.

Definition 5. 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 goal g_d to its option o_1. With $l = 3$ and the seed set S 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. Agents that know about all rules in this chain as well as the option o_1 will be able to construct a single argument for g_2.

4.2 Conflict generation

Scenarios do not only contain reasons why an option will lead to some goal being achieved. An important part of deliberation problems is that there are conflicts between what is known and what the rule chains proclaim. Therefore, the second generation 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 that 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 over the truth status of relevant facts.

Definition 6. 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 the attack between two arguments as formed from these rules is always be symmetric, i.e. they are equally acceptable. Although this is a simplification of the complex knowledge of real world deliberation problems it does already allow for structured arguments and counter arguments and, as demonstrated later, can be sufficiently complex to generate interesting dialogues.

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

4.3 Completing the knowledge pool

As a third step the pool of knowledge is completed before being allocated to the agents. The options and goals in the pool are aggregated from the roles in the system.

Definition 7. A knowledge pool K is assigned:

- a set of pool options $O_K = \bigcup_{r \in \mathcal{R}} O_r$
- a set of pool goals $G_K = \bigcup_{r \in \mathcal{R}} G_r$

Beliefs are derived from roles as well, by, for every option in the pool, generating either a chain or a set of conflicts, depending on whether the role was assigned the specific option.

Definition 8. For every option $o \in O_K$, given some role $r \in \mathcal{R}$ and a set size $n_{B_{\bar{r}}^o}$ a set of *role-option beliefs* B_r^o is any set such that:

- if $o \in O_r$ then $B_r^o = C_{g_d,o} \cup C_{g,o}$ for some goal $g \in G_r$
- if $o \notin O_r$ then $B_r^o \subseteq \bar{C}_{g,o}$ for an arbitrary goal $g \in G_r$ such that $|B_r^o| = n_{B_{\bar{r}}^o}$

The knowledge pool now contains both full paths from some option to some goal as well as negated facts to form counter arguments. The variable $n_{B_{\bar{r}}^o}$ can be used to tweak the number of generated negated beliefs for options not assigned to a role. Table 2 shows the rule chains that are generated for the roles in the running example as well as the selected negated beliefs for every chain.

Table 2: Belief assignment in the example's knowledge pool

$l = 3$	(chains with length 3)
$n_{B_{\bar{r}}^o} = 2$	(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
$B_{r_2}^{o_1}$	$\neg p_2, \neg \varrho 3$
$B_{r_2}^{o_2}$	$o_2 \Rightarrow_{\varrho 9} p_9,\ p_9 \Rightarrow_{\varrho 12} p_8,\ p_8 \Rightarrow_{\varrho 13} g_4,$
	$o_2 \Rightarrow_{\varrho 14} p_1,\ p_1 \Rightarrow_{\varrho 15} p_9,\ p_9 \Rightarrow_{\varrho 16} g_d$
$B_{r_2}^{o_3}$	$o_3 \Rightarrow_{\varrho 17} p_7,\ p_7 \Rightarrow_{\varrho 18} p_3,\ p_3 \Rightarrow_{\varrho 19} g_4,$
	$o_3 \Rightarrow_{\varrho 17} p_7,\ p_7 \Rightarrow_{\varrho 21} p_8,\ p_8 \Rightarrow_{\varrho 22} g_d$

4.4 Option and goal allocation

Following the characteristics of deliberation problems, agents inherit options and goals associated with their roles. Additionally, agents have personal goals that they do not necessarily share with other agents. Examples of such goals are those originating from personal values or goals transcending an agent's current role.

Definition 9. 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* $G_a^{\bar{r}}$ where for every $g \in G_a^{\bar{r}}$ it holds that $g \in G_K \setminus G_r$ and such that $|G_a^{\bar{r}}| = n_{G_a^{\bar{r}}}$
- The combined set of *goals* $G_a = G_r \cup G_a^{\bar{r}} \cup \{g_d\}$

Non-role originating goals are only goals that already exist in the knowledge pool. This is not for simplicity but ensures partial overlap between the goals of agents with different roles, as is typical for deliberation situations. The variable $n_{G_a^{\bar{r}}}$ will be used to set the number of non-role goals allocated to an agent.

4.5 Role and non-role belief allocation

The final step is to allocate the beliefs that an agent will have in the deliberation scenario. We will again use an agent's role to allocate knowledge from the pool to an agent. Since we have seen that agents usually have incomplete knowledge we will not assign all of a role's rule chains or negated facts.

Definition 10. An agent $a \in \mathcal{A}$ with some role r is assigned a set of $n_{B_a^r}$ *role-originating beliefs*

$$B_a^r \subseteq \bigcup_{o \in O_K} B_r^o \text{ such that } |B_a^r| = n_{B_a^r}$$

The variable $n_{B_a^r}$ is used to set the number of role-option beliefs, which for simplicity is equal for all agents. Since no agent will be assigned full knowledge, it is likely to miss some rule needed to construct a full argument for a chain, or that it cannot construct a counter-argument.

Agents also have beliefs that not necessarily originate from the role they have in the system. Such beliefs can come from various sources, such as an agent's expertise or prior encounters. It will be modelled as a set of rules taken from newly generated chains for some of the agent's options and goals, not different than how chains are generated for roles.

Definition 11. An agent $a \in \mathcal{A}$ is assigned a set of $n_{B_a^{\bar{r}}}$ *non-role originating beliefs*

$$B_a^{\bar{r}} \subseteq \bigcup_{o \in O_a} C_{g,o} \text{ for an arbitrary goal } g \in G_a \text{ such that } |B_a^r| = n_{B_a^{\bar{r}}}$$

Table 3: Information allocation to the agents for the example

	(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} $\neg \varrho 4,$
	$o_2 \Rightarrow_{\varrho 9} p_9, \ p_9 \Rightarrow_{\varrho 12} p_8, \ p_8 \Rightarrow_{\varrho 13} g_4,$
	$o_2 \Rightarrow_{\varrho 14} p_1, \ p_1 \Rightarrow_{\varrho 15} p_9, \ p_9 \Rightarrow_{\varrho 16} g_d,$
	$o_3 \Rightarrow_{\varrho 17} p_7, \ p_7 \Rightarrow_{\varrho 18} p_3, \ p_3 \Rightarrow_{\varrho 19} g_4,$
	$o_3 \Rightarrow_{\varrho 17} p_7, \ p_7 \Rightarrow_{\varrho 21} p_8, \ p_8 \Rightarrow_{\varrho 22} g_d,$
	$p_8 \Rightarrow_{\varrho 26} p_7, \ p_2 \Rightarrow_{\varrho 3} g_2$

The variable $n_{B_a^{\bar{r}}}$ is used to set the number of non-role originating beliefs known to the agent. The additional chains, if fully assigned to the agent, can be used to create supplementary arguments from some option to an agent's goal. In addition, the individual rules in the chain can solve the problem of missing rules in the chains it was assigned from its role. For instance if, in our running example, agent a_2 was not assigned the rule ϱ_4 it no longer knows all rules in the chain and cannot construct an argument for g_1 any more. However, its non-role originating beliefs set may include a supplementary chain $\{o_2 \Rightarrow p_1,\ p_1 \Rightarrow p_2,\ p_2 \Rightarrow g_d\}$ which again allows him to construct an argument for g_1 from o_2.

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

Definition 12. An agent $a \in \mathcal{A}$ is assigned a set of *beliefs* $B_a = B_a^r \cup B_a^{\bar{r}}$

The running example's allocated options, goals and beliefs are listed in Table 3. Agents a_1 and a_2 share a role r_1 in the system, while agent a_3 has role r_2. Every agent inherits the options, goals and an arbitrary part of the beliefs from their role (as specified in Table 2), but also is assigned a supplementary non-role originating goal and a small arbitrary set of non-role originating beliefs.

5 Evaluation of generated scenarios

A method to structurally generate deliberation scenarios has been presented that models the dynamics of deliberation problems using goals, rule chains and (non-)role based belief allocation. Recall from 2.3 that generated scenarios should be tested on to what extent they have the potential to produce interesting dialogues, which is defined by to what degree an agent has an argument for its options and to what degree counter-arguments to those exist.

It is now shown how an experimental approach, similar to experimentation with a full dialogue system, can be used to evaluate the scenario generation process. Importantly, the generation process uses 10 input parameters, as collected in Table 4 for the system in this paper, which directly influence how a scenario looks like and therefore to what degree the scenario is interesting.

5.1 Metrics for interesting scenarios

For an agent to propose one of its options in a deliberation dialogue it needs to be able to generate an argument from the option to one of its goals. Consequently, the first metric for interesting scenarios is to test if agents can form such arguments, in which case the option is called justified.

Definition 13. An agent a's option $o \in O_a$ is a *justified option* if, on the basis of the beliefs $B_a \cup \{o\}$, an argument $A \mathrel{|\!\sim} g$ can be constructed for some goal $g \in G_a$ such that $o \in A$.

Note that depending on the exact deliberation dialogue system used the argument may be required to be a credulously or sceptically acceptable argument

Table 4: Input parameters used in the scenario generation process

		min	example	max
$n_\mathcal{A}$	The number of agents	1	3	6
$n_\mathcal{R}$	The number of roles	1	2	6
n_{O_r}	A role r's options set size	2	2	5
n_{G_r}	A role r's goals set size	2	2	5
n_S	The chaining seedset size	10	10	100
l	The length of rule chains	3	3	9
$n_{B_{\bar{r}}^o}$	An agent a's negated role-option beliefs set size	0	3	15
$n_{G_a^{\bar{r}}}$	An agent a's non-role originating goals set size	0	1	2
$n_{B_a^r}$	An agent a's role-originating beliefs set size	1	7	15
$n_{B_a^{\bar{r}}}$	An agent a's non-role originating beliefs set size	0	2	20

or even have the mutual deliberation goal as conclusion. For the system in this paper we will require an argument that is defensible under preferred semantics, matching the credulous nature of practical reasoning in deliberation dialogues.

It is now possible to define a metric that indicates to what extent a scenario is interesting, that is, which of the options of the agents are justified.

Definition 14. A generated scenario with a set of agents $\mathcal{A}$ has an *option justification percentage*

$$j_\mathcal{A} = \frac{|\bigcup_{a \in \mathcal{A}} \{o | o \in O_a \text{ where } o \text{ is a justified option}\}|}{n_\mathcal{A} \times n_{O_r}} \times 100$$

Consider again the generated example scenario. For each of the three agents their two options are tested. Table 5 shows the arguments that the agents can form. Agents a_1 and a_2 have arguments from both their options, while agent a_3 was able to construct an argument for o_2 but not for o_3 since it misses the rule $p_8 \Rightarrow_{\varrho 16} g_3$. Hence, the option justification percentage $j_\mathcal{A} = \frac{5}{6} \times 100 = 83\%$.

To see if scenarios also spur counter-arguments to the arguments used for justified options a second metric is defined. The idea is that for every justified option of an agent the other agents' knowledge is used to construct a counter-argument as so to allow attacks on proposals as made in a deliberation dialogue.

Definition 15. An agent a's justified option o, as supported by argument $A \mathrel{|\!\sim} g$, is also a *countered justified option* if some agent $a' \in \mathcal{A}$, where $a \neq a'$, can, on the basis of beliefs $B_{a'} \cup \{o\}$, construct a counter-argument $B \mathrel{|\!\sim} p$ that defeats $A \mathrel{|\!\sim} g$.

Analogous to the degree of option justification we define a metric for the degree of justified options that are also countered by some argument.

Table 5: Arguments for option justification in the example scenario

		justification argument	counter-argument
a_1	o_1	$o_1, o_1 \Rightarrow_{\varrho 1} p_5$	$\neg o_1$ by a_3
		$p_5, p_5 \Rightarrow_{\varrho 2} p_2$	
		$p_2, p_2 \Rightarrow_{\varrho 3} g_d$	
	o_2	$o_2, o_2 \Rightarrow_{\varrho 4} p_2$	(no counter-argument)
		$p_2, p_2 \Rightarrow_{\varrho 5} p_9$	
		$p_9, p_9 \Rightarrow_{\varrho 6} g_1$	
a_2	o_1	$o_1, o_1 \Rightarrow_{\varrho 1} p_5$	$\neg o_1$ by a_3
		$p_5, p_5 \Rightarrow_{\varrho 2} p_2$	
		$p_2, p_2 \Rightarrow_{\varrho 3} g_d$	
	o_2	$o_2, o_2 \Rightarrow_{\varrho 18} p_6$	$\neg p_6$ by a_3
		$p_6, p_6 \Rightarrow_{\varrho 19} p_2$	
		$p_2, p_2 \Rightarrow_{\varrho 5} p_9$	
		$p_9, p_9 \Rightarrow_{\varrho 6} g_1$	
a_3	o_2	$o_2, o_2 \Rightarrow_{\varrho 12} p_3$	(no counter-argument)
		$p_3, p_3 \Rightarrow_{\varrho 13} p_7$	
		$p_7, p_7 \Rightarrow_{\varrho 14} g_3$	
	o_3	(no argument)	

Definition 16. A generated scenario with a set of agents $\mathcal{A}$ has an *option countered justification percentage*

$$\bar{j}_{\mathcal{A}} = \frac{|\bigcup_{a \in \mathcal{A}} \{o | o \in O_a \text{ where } o \text{ is a countered justified option}\}|}{|\bigcup_{a \in \mathcal{A}} \{o | o \in O_a \text{ where } o \text{ is a justified option}\}|} \times 100$$

The example's countered option justification percentage $\bar{j}_{\mathcal{A}} = \frac{3}{5} \times 100 = 60\%$ since agent a_3 can construct a counter-argument for three of the justified options: to both agent a_1's and a_2's argument for o_1 and to agent a_2's argument for o_2.

5.2 Experimental validation of scenario generation

The scenario generation method is controlled by a large number of input parameters. To investigate which input parameter settings within a reasonable range (see Table 4) produce interesting dialogues a software experiment was performed. Scenarios were generated repeatedly, with a total of a 1000, with random parameter settings and the two metrics $j_{\mathcal{A}}$ and $\bar{j}_{\mathcal{A}}$ are applied to see if the agent's options in the scenario are justified and countered justified.

To visualize the effect an input parameter can have Figure 2 shows the (countered) option justification for the number of role-originating beliefs $n_{B_a^r}$ allocated to an agent. Clearly the more beliefs are assigned to an agent the higher chances are that it can form arguments for its options (Figure 2a) and that other agents can form a counter-argument to that (Figure 2b).

While the effects on individual parameters can already be seen in graphs, this does not provide a full insight into the dynamics between the parameters. For
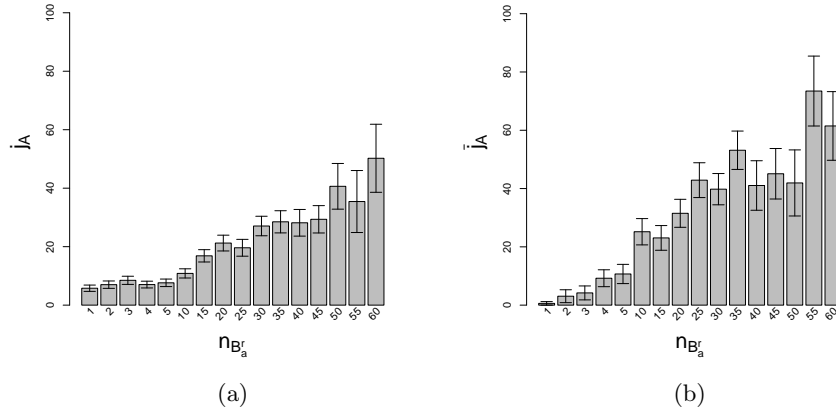
Fig. 2: Average (countered) option justification percentages $j_{\mathcal{A}}$ and $\bar{j}_{\mathcal{A}}$ (with standard errors of the mean) with $n_{B_a^r} \in \{1, \ldots, 25\}$

example, it might be that the number of roles has no effect unless the number of agents is high (or low). It is important to capture the interplay of parameters as so to find the parameters that are most influential to the degree of (countered) option justification. When performing experiments that make use of the scenario generation process it is possible to adjust those important parameters.

The input parameters with the strongest influence on $j_{\mathcal{A}}$ and $\bar{j}_{\mathcal{A}}$ are found by performing a multiple linear regression analysis on the set of 2000 scenarios that were generated with random input parameter settings. This creates a model of the data and, in a stepwise fashion, determines the parameters that contributed significantly to the (countered) option justification metrics $j_{\mathcal{A}}$ and $\bar{j}_{\mathcal{A}}$.

Out of the 10 input parameters 7 have a statistically significant influence on $j_{\mathcal{A}}$, with $F(10, 1989) = 158.7$, $P < 0.001$ and adjusted R-squared 0.44. This means that 44% of the variance in $j_{\mathcal{A}}$ can be explained from the 7 input parameters. For $\bar{j}_{\mathcal{A}}$ 8 parameters have a statistically significant influence, with $F(10, 768) = 42.94$, $P < 0.001$ and adjusted R-squared 0.35, that is, explaining 35% of the variation in $\bar{j}_{\mathcal{A}}$. Table 6 shows the input parameters with significant influence ordered by the amount of influence (standard beta coefficient), t and P values (probability values, where non-significant parameters are labelled NP).

Different interesting results can be derived from the statistical analysis. Foremost, when experimenting with the deliberation system and scenario generation process of this paper the length of rule chains l is the first parameter that should be varied when a differing degree of option justification $j_{\mathcal{A}}$ is to be tested. The bigger l is, the smaller $j_{\mathcal{A}}$ will be, in line with the intuition that it increases belief disparity. On the other hand, if focus of an experiment is on countering

Table 6: Input parameters and their influence on $j_\mathcal{A}$ and $\bar{j}_\mathcal{A}$

	option justification $j_\mathcal{A}$				countered option justification $\bar{j}_\mathcal{A}$			
	β	t	P	$ideal$	β	t	P	$ideal$
l	−0.51	−28.66	< 0.001	5	−0.05	−1.65	< 0.1	4
$n_{B_a^r}$	0.43	23.99	< 0.001	60	0.53	14.60	< 0.001	45
$n_{B_a^{\bar{r}}}$	0.21	12.29	< 0.001	9	−0.03	−0.84	NS	3
n_{O_r}	−0.15	−8.71	< 0.001	3	−0.16	−5.31	< 0.001	2
n_S	−0.12	−6.72	< 0.001	75	−0.15	−4.67	< 0.001	35
$n_{B_{\bar{r}}^o}$	−0.04	−2.09	< 0.05	5	0.08	2.52	< 0.05	6
$n_\mathcal{R}$	−0.04	−2.54	< 0.05	5	0.10	3.32	< 0.001	5
$n_{G_a^{\bar{r}}}$	−0.02	−0.96	NS	2	0.05	1.67	< 0.1	3
$n_\mathcal{A}$	0.01	0.47	NS	4	0.37	12.26	< 0.001	6
n_{G_r}	−0.01	−0.43	NS	2	−0.04	−1.44	NS	3

proposals the number of agents $n_\mathcal{A}$ should be varied instead, since that will have most influence on $\bar{j}_\mathcal{A}$. Unsurprisingly the number of role-originating beliefs $n_{B_a^r}$ has a high influence on both $j_\mathcal{A}$ and $\bar{j}_\mathcal{A}$, since a larger number of beliefs from an agent's role will increase the chance of ending op with all rules for a chain or the right negated beliefs for a chain respectively. Moreover, it has an even bigger influence on $\bar{j}_\mathcal{A}$ than on $j_\mathcal{A}$, visualized by the steeper slope in Figure 2.

The most influential parameters have been established, but not yet the parameter setting that gives the most interesting dialogues, that is, that maximizes $j_\mathcal{A}$ and $\bar{j}_\mathcal{A}$. This often will be the starting point when experimenting since it offers the agents scenarios with most chances of proposing options and countering those. The parameter setting that maximizes one of the metrics will be called the ideal setting and can be found using the regression model defined above. As the model predicts precisely (with $P < 0.001$) the outcome of $j_\mathcal{A}$ or $\bar{j}_\mathcal{A}$ it can also be used to find the maximal value. A sufficiently large data set ($N = 2000$) is produced for both metrics and maximal values are found. The parameter values that produce these maximal values are shown in Table 6 and yield $j_\mathcal{A} = 66\%$ and $\bar{j}_\mathcal{A} = 99\%$. Hence, with the ideal settings for $j_\mathcal{A}$ the agents will be able to propose roughly half of their options while with the ideal values for $\bar{j}_\mathcal{A}$ the agents can counter essentially all options that are justified. Future work will study how a single ideal setting for combined measures of interestingness can be established.

6 Conclusion

A methodology has been proposed to model a scenario generation process for argumentation-enabled deliberation dialogues. The contribution is twofold. First, the state of the art in preparation of experiments for argumentation-based dialogues has been advanced. A scenario generation process for deliberation is proposed that is the first to the authors' knowledge that allows for structured

argumentation and through this supporting the full dynamics of the underlying argumentation problem. Secondly, it is shown how an experimental analysis helps to evaluate the quality of the scenario generation process by identifying the most influential input parameters and giving the ideal parameter settings that maximize the interestingness of scenarios.

The method as proposed in this paper, as well as the specific system described, is used in a larger project to make the uses of argumentation in multiagent systems concrete. It has allowed us to experimentally analyse the uses of argumentation in realistic scenarios by testing a variety of strategies for argumentative and non-arguing agents as well as identify specific scenarios in which argumentation is more (or less) suited.

References

1. Black, E., Bentley, K.: 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)
2. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. Artificial Intelligence 77(2), 321–357 (1995)
3. Karunatillake, N.C., Jennings, N.R., Rahwan, I., McBurney, P.: Dialogue games that agents play within a society. Artificial Intelligence 173(9-10), 935–981 (2009)
4. Kok, E.M., Meyer, J.J.C., Prakken, H., Vreeswijk, G.A.W.: A Formal Argumentation Framework for Deliberation Dialogues. In: McBurney, P., Parsons, S., Rahwan, I. (eds.) Proceedings of the 7th International Workshop on Argumentation in Multi-Agent Systems. Springer-Verlag, Berlin, Germany (2010)
5. McBurney, P., Hitchcock, D., Parsons, S.: The eightfold way of deliberation dialogue. International Journal of Intelligent Systems 22(1), 95–132 (2007)
6. Ontañón, S., Plaza, E.: Multiagent Inductive Learning: an Argumentation-based Approach. In: Proceedings of the 27th International Conference on Machine Learning. pp. 839–846. No. 2, Haifa, Israel (2010)
7. Paglieri, F., Castelfranchi, C.: Why argue? Towards a cost-benefit analysis of argumentation. Argument and Computation 1(1), 71–91 (2010)
8. Pasquier, P., Hollands, R., Rahwan, I., Dignum, F., Sonenberg, L.: An empirical study of interest-based negotiation. Autonomous Agents and Multi-Agent Systems 22(2), 249–288 (2010)
9. Prakken, H.: An abstract framework for argumentation with structured arguments. Argument and Computation 1(2), 93–124 (2010)
10. Rahwan, I.: Guest editorial: Argumentation in multi-agent systems. Autonomous Agents and Multi-Agent Systems 11(2), 115–125 (2005)
11. Rao, A., Georgeff, M.: Modeling rational agents within a BDI-architecture. In: Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning. pp. 473–484. Morgan Kaufmann (1991)
12. Walton, D.N., Krabbe, E.C.W.: Commitment in dialogue: Basic concepts of interpersonal reasoning. State University of New York Press, New York (1995)