DOI:
10.1007/978-3-642-29184-5_5

An Implemented Dialogue System for Inquiry and Persuasion

Luke Riley[1], Katie Atkinson[1], Terry Payne[1] and Elizabeth Black[2]

[1] Department of Computer Science,
University of Liverpool,
L69 3BX, UK
{L.J.Riley,katie,trp@liverpool.ac.uk}
[2] Department of Information and Computer Sciences,
Utrecht University,
80.089, NL
{lizblack@cs.uu.nl}

Abstract. In this paper, we present an implemented system that enables autonomous agents to engage in dialogues that involve inquiries embedded within a process of practical reasoning. The implementation builds upon an existing formal model of value-based argumentation, which has itself been extended to permit a wider range of arguments to be expressed. We present extensions to the formal underlying theory used for the dialogue system, as well as the implementation itself. We demonstrate the use of the system through a particular case study. We discuss a number of interesting issues that have arisen from the implementation and the experimental avenues that this test-bed will enable us to pursue.

1 Introduction

Communication through argumentation is one of the key strands of work on computational argumentation. Work on agent-based dialogue systems has been greatly influenced by the dialogue typology of Walton and Krabbe [12]. A number of proposals have been set out for dialogue systems that encompass the main dialogue categories, for example see: [4] for inquiry dialogues; [10] for negotiation; [9] for persuasion; [8] for deliberation. However, very little work has been done on specifying and implementing systems that combine two or more dialogue types. In [3] a formal framework was set out for multi-agent dialogues over actions in which inquiry dialogues over beliefs are combined with persuasion dialogues over actions. The dialogue system allows agents with heterogeneous knowledge to each give input into a decision about how to act to achieve a shared goal. The underlying representation of an argument is in terms of a formal version of an argumentation scheme for practical reasoning, and critical questions that agents can employ to challenge assertions made by their peers. Although this dialogue system has been set out in a formal specification [3], it has not previously been validated through an implementation. In this paper we present the details of an implementation of this dialogue system for inquiry and persuasion over action. For a full implementation to be realised it was necessary to extend the formalism presented in [3] to enable a richer set of arguments to be put forward, which we describe. The implemented system we present provides not only a proof-of-concept in terms of an application of a formal specification, but we also note a number of issues that have been identified through this the exercise. Furthermore, we consider this implementation to be a starting point for further investigations into agent argumentation dialogues, in particular, with respect to coalition formation.

The paper is structured as follows. In Section 2 we recapitulate from [3] the background material about the dialogue system we have implemented. In Section 3 we present new material that extends the formalism of [3] by providing the full list of critical questions associated with the argumentation scheme that is used in the dialogue system. In Section 4 we describe the implementation and demonstrate it with an example. In Section 5 we discuss issues that have arisen from the implementation, future avenues this work will allow us to explore and we conclude the paper.

2 Background

Our dialogue system allows agents to inquire about beliefs (to determine the state of the world) and collectively perform practical reasoning over the an action to perform in a given situation. To do this, agents in our system may have epistemic knowledge (beliefs), represented by Garcia and Simari's Defeasible Logic Programming [6], as well as normative knowledge about the effects of actions.

The following definitions provide the formal framework for modeling beliefs.

Definition 1: *A* **defeasible rule** λ *is denoted* $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \alpha_0$ *where* α_i *is a literal for* $0 \leq i \leq n$*. A* **defeasible fact** *is denoted* α *where* α *is a literal. A* **belief** *is either a defeasible rule or a defeasible fact. We define the following functions* $\mathsf{DefeasibleSection}(\lambda) = \{\alpha_1, \ldots, \alpha_n\}$*;* $\mathsf{DefeasibleProp}(\lambda) = \alpha_0$.

The definition of a defeasible derivation is adapted from [6] to work with our assumption that all beliefs are defeasible.

Definition 2: *Let* Ψ *be a set of beliefs and* α *a literal. A* **defeasible derivation** *of* α *from* Ψ*, denoted* $\Psi \mid\sim \alpha$*, is a finite sequence* $\alpha_1, \alpha_2, \ldots, \alpha_n$ *of literals s.t.:* α_n *is* α*; and each literal* α_m $(1 \leq m \leq n)$ *is in the sequence because either* α_m *is a defeasible fact in* Ψ*, or there exists a defeasible rule* $\beta_1 \wedge \ldots \wedge \beta_j \rightarrow \alpha_m$ *in* Ψ *s.t. every literal* β_i $(1 \leq i \leq j)$ *is an element* α_k *preceding* α_m *in the sequence* $(k < m)$*.*

A b-argument is a minimally consistent set of beliefs from which a claim can be defeasibly derived.

Definition 3: *A* **b-argument** *is denoted* $B = \langle \Phi, \phi \rangle$ *where* ϕ *is a defeasible fact and* Φ *is a set of beliefs s.t.: 1)* $\Phi \mid\sim \phi$*; 2)* $\forall \phi, \phi'$ *s.t.* $\Phi \mid\sim \phi$ *and* $\Phi \mid\sim \phi'$*, it is not the case that* $\phi \cup \phi' \vdash \perp$ *(where* $\vdash$ *represents classical implication); and there is no subset of* Φ *satisfying (1 and 2).* Φ *is called the* **support** *of the b-argument and* ϕ *is called the* **claim**.

Each agent has a unique id x taken from a set $\mathcal{I}$ of agent identifiers. Each agent's belief base could be inconsistent.

Definition 4: *A* **belief base** *of an agent* x *is a finite set of beliefs, denoted* Σ^x.

For handling reasoning about the effects of actions, the following argumentation scheme for practical reasoning is used, taken from [1]:

In the current circumstances R, we should perform action A, which will realise goal G, which will result in the new circumstances S, which will promote some value V.

This scheme uses 'values' to describe a social interest an agent has, which will be promoted by moving to a state in which goal G becomes true [2]. An agent may propose an action including its justification by instantiating this scheme. Other agents can then challenge instantiations by posing critical questions (CQ) associated with the scheme. Seventeen critical questions are associated with the above scheme [1] which raise potential issues with: the validity of the elements instantiated in the scheme; the connections between the elements of the scheme; the side effects of actions; and the possible alternatives.

In [3] an agent's knowledge about the effects of actions is represented as a Value-based Alternating Transition System (VATS), a modified version of an Action-Based Transition System (AATS) [13], which has been extended to enable the inclusion of values.

Definition 5: *The VATS formalism is as follows: A* **VATS** *for an agent x, denoted S^x, is a 9-tuple $\langle Q^x, q_0^x, Ac^x, Av^x, \rho^x, \tau^x, \Phi^x, \pi^x, \delta^x \rangle$ s.t.:*

- Q^x *is a finite set of* states*;*
- $q_0^x \in Q^x$ *is the designated* initial state*;*
- Ac^x *is a finite set of* actions*;*
- Av^x *is a finite set of* values*;*
- $\rho^x : Ac^x \mapsto 2^{Q^x}$ *is an* action precondition function*, which for each action $a \in Ac^x$ defines the set of states $\rho(a)$ from which a may be executed;*
- $\tau^x : Q^x \times Ac^x \mapsto Q^x$ *is a partial* system transition function*, which defines the state $\tau^x(q, a)$ that would result by the performance of a from state q. As this function is partial, not all actions are possible in all states;*
- Φ^x *is a finite set of* atomic propositions*;*
- $\pi^x : Q^x \mapsto 2^{\Phi^x}$ *is an interpretation function, which gives the set of primitive propositions satisfied in each state: if $p \in \pi^x(q)$, then this means that the propositional variable p is satisfied (equivalently, true) in state q; and*
- $\delta^x : Q^x \times Q^x \times Av^x \mapsto \{+, -, =\}$ *is a* valuation function *which defines the* status *(promoted ($+$), demoted ($-$), or neutral ($=$)) of a value $v \in Av^x$ ascribed by the agent to the transition between two states: $\delta^x(q, q', v)$ labels the transition between q and q' with respect to the value $v \in Av^x$.*

Note, $Q^x = \emptyset \leftrightarrow Ac^x = \emptyset \leftrightarrow Av^x = \emptyset \leftrightarrow \Phi^x = \emptyset$.

With its VATS an agent can construct a-arguments for and against actions. Together a-arguments and CQs are referred to as arguments over actions (AOAs).

Definition 6: *An* **a-argument** *constructed by an agent x from its VATS S^x is a 6-tuple $A = \langle q_x, a, q_y, p, v, s \rangle$ s.t.: $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $p \in \pi^x(q_y)$; $v \in Av^x$; $\delta^x(q_x, q_y, v) = s$ where $s \in \{+, -, =\}$.*
We define the following functions: $\mathsf{Action}(A) = a$*;* $\mathsf{Goal}(A) = p$*;* $\mathsf{Value}(A) = v$*;* $\mathsf{EndState}(A) = q_y$*;* $\mathsf{Polarity}(A) = s$*.*
If $\mathsf{Polarity}(A)$ *has the value $+(-resp.)$, then we say A is an a-argument for (against resp.) action a to achieve goal p. If* $\mathsf{Polarity}(A)$ *has the value "$=$", then we say A is an a-argument that is neutral with regards to action a.*

Our framework assumes a closed cooperative multi-agent system. Agents collaborate to find the best action to achieve the dialogue initiator's goal by entering a *persuasion over action* (*pAct*) dialogue, which provides the agents with an opportunity to persuade the others by putting forward AOAs for the known possible actions. However, before the AOAs can be asserted, each agent x must inquire over its known propositions so that its initial state can be found. Once this has occurred, the correct AOAs for the current system state can be uttered. To find its initial state each agent x participating in the dialogue first opens an inquiry (inq) sub-dialogue (i.e. a dialogue that is embedded within a top-level dialogue) [3] with the other agents in the system. The result is a truth value for all of agents x's propositions that have not already been discussed in another inq sub-dialogue.

Our dialogue is defined as a dialogue game. Dialogue games typically consist of a set of communicative acts (called moves) and a set of rules that firstly state which moves are legal for any point of the dialogue (the protocol), secondly define the effect of making a move and a lastly determine when a dialogue terminates [7, 11]. Within our framework, a dialogue denoted $\mathcal{D}_r^t$, is a sequence of moves $m_r, \ldots, m_t$ where $r, \ldots, t \in \mathbb{N}$ represents the time-point at which each move was made, with r being the starting point of the dialogue and t the end point. If $r = 1$, then this dialogue is considered a top level dialogue whose type is $pAct$, which is opened by the dialogue initiator. If the top level dialogue is closed then the dialogue game is over [4]. If $r \neq 1$ then this is a sub dialogue whose type is inq. The following functions operate over a dialogue[5]:

- $\mathsf{Current}(\mathcal{D}_1^t)$ returns the most recently opened dialogue that has not been closed.
- $\mathsf{Type}(\mathcal{D}_r^t)$ returns the type of the dialogue $\mathcal{D}_r^t$ (i.e. $pAct$ or inq).
- $\mathsf{Initiator}(\mathcal{D}_r^t)$ returns the agent who opened dialogue D_r^t.
- $\mathsf{Participants}(\mathcal{D}_r^t)$ returns the set of agents in the dialogue D_r^t.
- $\mathsf{Topic}(\mathcal{D}_r^t)$ returns the goal the agents are trying to achieve **iff** $\mathsf{Type}(D_r^t) = pAct$.
- $\mathsf{Topic}(\mathcal{D}_r^t)$ returns the set of propositions which the agents are jointly trying to find the truth value of **iff** $\mathsf{Type}(D_r^t) = inq$.
- $\mathsf{Turn}(\mathcal{D}_r^t)$ returns the identifer of the agent whose turn it is.

The moves that the agents can perform are presented in Table 1. Agents take it in turns to perform one move at a time. All agents' assertions are stored in their commitment stores (CS) that grow monotonically over time, as follows:

Definition 7: **Commitment store update.**
For a pAct dialogue with participants $\{x_1, \ldots, x_n\}$, $\forall x \in \{x_1, \ldots, x_n\}$ *and a commitment store of agent* x *at time-point t denoted* CS_x^t,

$$CS_x^t = \begin{cases} \emptyset & \textit{iff } t = 0, \\ CS_x^{t-1} \cup \Upsilon & \textit{iff } m_t = \langle x, assert, \Upsilon \rangle, \\ CS_x^{t-1} & \textit{otherwise.} \end{cases}$$

[3] Further details of the inquiry sub-dialogues are discussed in Section 3.4.

[4] For future work we will look into opening more than one $pAct$ dialogue before the game is over

[5] Further dialogue details are given in [3].

Move	Format
$open$	$\langle x, open, dialogue(\theta, \gamma, \Lambda)\rangle$
$assert$	$\langle x, assert, \Upsilon\rangle$
$close$	$\langle x, close, dialogue(\theta, \gamma, \Lambda)\rangle$

Table 1. The format for moves used in this dialogue game, where x represents the agent making the move and either $\theta = pAct$ and γ is a proposition (representing the dialogue goal), or $\theta = inq$ and γ is a set of propositions (that the agent is inquiring over); Λ is a list of agents ($\Lambda = [x_1, \ldots, x_n]$, $\{x_1, \ldots, n_n\} \subseteq \mathcal{I}$); Υ is either a set of a-arguments and critical questions (if $\theta = pAct$) or Υ is a set of b-arguments and beliefs (if $\theta = inq$); and x is an agent ($x \in \mathcal{I}$).

Definition 8: *The union of all the commitment stores is defined as*

$$CSs = \bigcup_{\forall x_i \in \{x_1, \ldots, x_n\}} CS_{x_i}.$$

Dialogues commence when an event triggers one agent to open a *pAct* dialogue through its *pAct* strategy (see Section 3.5) to identify the best action to achieve a given proposition p, where $p = \mathsf{Topic}(D_1^t)$. The other agents that have been included in the open dialogue move then initiate their individual *pAct* strategies, which are guaranteed to find all the arguments related to the dialogue topic, via the use of the *pAct* protocol (Section 3.3) and the *inq* protocol (Section 3.4), before terminating. Both dialogue types will not complete until all agents have made a close move one after another without a different move separating them, as this ensures that the dialogue does not terminate until none of the agents have anything more they want to say.

Once the *pAct* dialogue has terminated, the system evaluates the arguments to determine the maximally consistent acceptable set. This is achieved within our framework using a *Value-Based Argumentation Framework* (VAF) [2], A VAF is an extension of the argumentation framework (AF) of Dung [5]. In an AF an argument is admissible with respect to a set of arguments S if all of its attackers are attacked by some argument in S, and no argument in S attacks an argument in S. In a VAF, an argument succeeds in defeating an argument it attacks only if its value is ranked as high, or higher, than the value of the argument attacked; a particular ordering of the values is characterised as an *audience*. Arguments in a VAF are admissible with respect to an audience A and a set of arguments S if they are admissible with respect to S in the AF which results from removing all the attacks that are unsuccessful given the audience A. A maximal admissible set of a VAF is known as a *preferred extension*.

The output of evaluating a VAF is a recommended action (or non-action) that should be performed to achieve the agents' shared goal. An action can only be recommended by the system if it is present in an AOA that is present in the *preferred extension* and the AOA states that the action promotes a value. In the event that there is more than one acceptable action the choice is offered to the dialogue initiator.

3 Extending the Formalisation of Critical Questions

The dialogue system set out in [3] handled only three of the possible seventeen critical questions associated with the practical reasoning argumentation scheme. Here we extend the dialogue system by specifying all the necessary critical questions and show their use within the dialogue system. The CQs formalised in Section 3.2 that follow Definition 2 are as a-arguments also. All CQs can be asserted by any agent to challenge an assertion of any other agent (including itself). If all agents follow the $pAct$ protocol then AOAs can only be asserted if they have not previously been asserted.

3.1 The State Comparison Definition

One particular issue that arose when implementing the dialogue system was the need for a mechanism to clarify how agents who may be using different propositions to represent the state of the world can accurately compare states (since agents' VATS reflect only an individual's representation of the world). As such, we define that two agents, m and n can compare their respective states $q_m \approx q_n$ iff $\pi(q_m) \cap \Phi^n = \pi(q_n) \cap \Phi^m$, otherwise $q_m \not\approx q_n$. The intersection is used to eliminate propositions that reside in only one of the agent's beliefs.

When the above approximation holds, the two states q_n and q_m cannot reasonably be said to be different, as both states will agree for each shared proposition. However, these two states may not be identical as the same conclusion can be reached whatever the truth assignments of the distinct propositions. If the comparison does not hold then the states are different due to both agents holding inconsistent truth assignments for their shared propositions.

This comparison requires either; each agent to have an internal model of the other agent's beliefs, or an instantiated state in an assertion should make explicit all the propositions that the agent holds to be true or false. Both will allow an agent to access the beliefs of another. This paper chooses the latter option due to the ease of implementation of such a representation. There are no privacy issues for this framework as it is designed for a closed and cooperative system. The following shows how the state comparison definition works, when:

$$\Phi^m = \{p, q, r, t\},\ \Phi^n = \{p, r, v\},\ q_m = [p, \neg q, \neg r, t],\ q_n = [p, \neg r, v]$$

The state comparison definition:

$$\pi(q_m) \cap \Phi^n = \pi(q_n) \cap \Phi^m$$

The substitution:

$$\{p, t\} \cap \{p, r, v\} = \{p, v\} \cap \{p, q, r, t\}$$

$$\{p\} = \{p\}$$

$Conclusion$: No evidence to suggest the states are necessarily different.

3.2 The Additional Critical Questions

We now define arguments that instantiate the remaining critical questions given in [1] that are applicable to our framework (those that are not applicable are discussed subsequent to the presentation of the definitions). Within the formal definitions given below we also give the natural language representation of the questions. Accompanying the definitions are figures that illustrate a situation where each CQ could be posed. All illustrations usually assume (unless otherwise stated) that both agents have in the initial state $\neg p$ (via the interpretation function), the pAct dialogue topic is to achieve p and agent 1 takes the first turn.

Definition 9: **A cq2-argument** *Answers the question* 'Assuming the circumstances, does the action have the stated consequences?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$, $\tau^x(q_x, a) = q_y$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ or an AOA $\langle q'_x, a', q'_y, v', s' \rangle$ iff $q_x \approx q'_x$, $a = a'$, $q_y \not\approx q'_y$.*

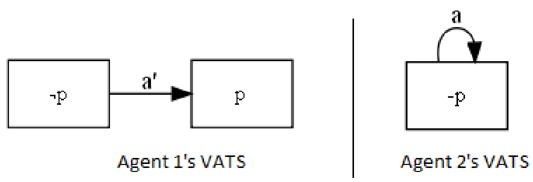

Fig. 1: Illustration of a cq2-argument (Definition 9) and a cq15-argument (Definition 21). Agent 2 will assert a cq-argument when agent 1 asserts an AOA to achieve p. The cq-argument that agent 2 chooses depends on which formal conditions are met. No value has been included for this example as values are not part of the definition of a cq2-argument or a cq15-arguments.

Definition 10: **A cq3-argument** *Answers the question* 'Assuming the circumstances, and the action has the stated consequences, will the action bring about the desired goal?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, \neg p \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $p \notin (q_y)$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ or an AOA $\langle q'_x, a', q'_y, v', s' \rangle$ iff $q_x \approx q'_x$, $a = a'$, $q_y \approx q'_y$.*

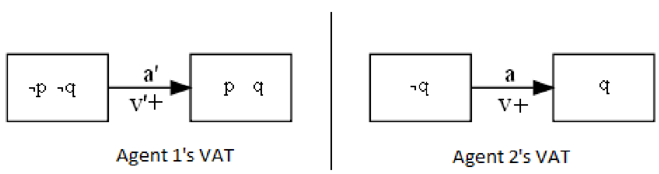

Fig. 2: Illustration of a cq3-argument (Definition 10). Agent 2 will assert a cq3-argument when agent 1 asserts an AOA to achieve p. The initial state of Agent 1 is $[\neg p, \neg q]$ and the initial state of Agent 2 is $[\neg q]$.

Definition 11: **A cq4-argument** *Answers the question* 'Does the goal realise the value stated?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, p, v, s \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $p \in \pi(q_y)$; $v \in Av^x$; $\delta^x(q_x, q_y, v) \neq +$; $s \in =, -$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ iff $q_x \approx q'_x$, $p = p'$, $v = v'$, $s' = +$.*

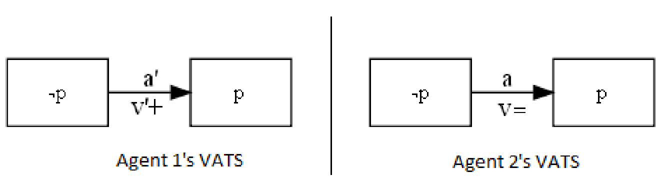

Fig. 3: Illustration of a cq4-argument (Definition 11). Agent 2 will assert a cq4-argument when agent 1 asserts an AOA to achieve p.

Definition 12: **A cq5-argument** *Answers the question* 'Are there alternative ways of realising the same consequences?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$. It challenges an AOA $\langle q'_x, a', q'_y, v', s' \rangle$ or an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ iff $q_x \approx q'_x$, $a \neq a'$, $q_y \approx q'_y$.*

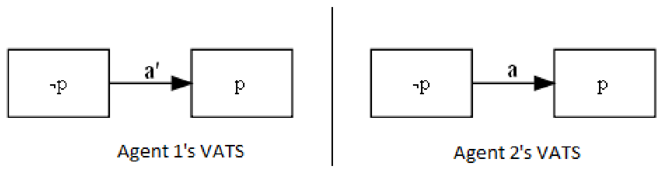

Fig. 4: Illustration of a cq5-argument (Definition 12). Agent 2 will assert a cq5-argument when agent 1 asserts an AOA to achieve p. No value has been included for this example as values are not part of the definition of a cq5-argument.

Definition 13: **A cq6-argument** *Answers the question* 'Are there alternative ways of realising the same goal?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, p, v, + \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $p \in \pi(q_y)$; $v \in Av^x$; $\delta(q_x, q_y, v) = +$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ iff $q_x \approx q'_x$, $a \neq a'$, $p = p'$, $s' = +$.*

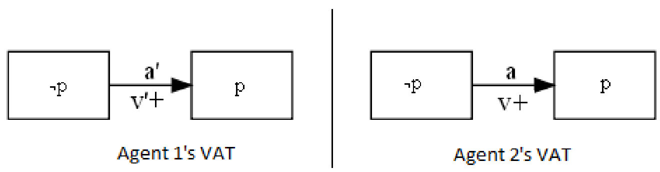

Fig. 5: Illustration of cq6-argument (Definition 13), a cq7 argument (Definition 14), a cq10 argument (Definition 17) and a cq-11 argument (Definition 18). Agent 2 will assert a cq-argument when agent 1 asserts an AOA to achieve p. The cq-argument that agent 2 chooses depends on which formal conditions are met. The main difference between CQ6 and CQ7 is whether the agent knows the new action achieves the goal (CQ6) or not (CQ7).

Definition 14: **A cq7-argument** *Answers the question* 'Are there alternative ways of promoting the same value?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, v, + \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $v \in Av^x$; $\delta(q_x, q_y, v) = +$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ or an AOA $\langle q'_x, a', q'_y, v', s' \rangle$ iff $q_x \approx q'_x$, $a \neq a'$, $v = v'$, $s' = +$.*

See Fig. 5 for an illustration of example VATSs that could produce a cq7-argument.

Definition 15: **A cq8-argument** *Answers the question* 'Does doing the action have a side effect which demotes the value?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, v, - \rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $v \in Av^x$; $\delta^x(q_x, q_y, v) = -$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s' \rangle$ or an AOA $\langle q'_x, a', q'_y, v', s' \rangle$ iff $q_x \approx q'_x$, $a = a'$, $v = v'$, $s' = +$.*

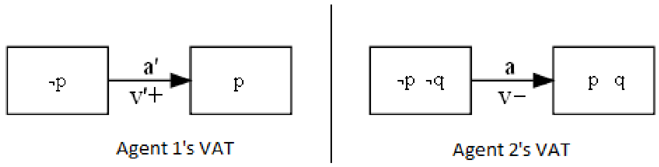

Fig. 6: Illustration of a cq8-argument (Definition 15) and a cq9-argument (Definition 16). Agent 2 will assert a cq-argument when agent 1 asserts an AOA to achieve p. The initial state of agent 2 is $[\neg p, \neg q]$ and the side effect is q. The cq-argument that agent 2 chooses depends on which formal conditions are met.

Definition 16: *A* **cq9-argument** *Answers the question* 'Does doing the action have a side effect which demotes some other value?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, v, -\rangle$ s.t.: $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $v \in Av^x$; $\delta^x(q_x, q_y, v) = -$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', +\rangle$ or an AOA $\langle q'_x, a', q'_y, v', s'\rangle$ iff $q_x \approx q'_x$, $a = a'$, $v \neq v'$, $s' = +$.*

See Fig. 6 for an illustration of example VATSs that could produce a cq9-argument.

Definition 17: *A* **cq10-argument** *Answers the question* 'Does doing the action have a side effect which promotes some other value?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, v, +\rangle$ s.t.: $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $v \in Av^x$; $\delta^x(q_x, q_y, v) = +$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s'\rangle$ or an AOA $\langle q'_x, a', q'_y, v', s'\rangle$ iff $q_x \approx q'_x$, $a = a'$, $v \neq v'$, $s' = +$.*

See Fig. 5 for an illustration of example VATSs that could produce a cq10-argument.

Definition 18: **A cq11-argument** *Answers the question* 'Does doing the action preclude some other action which would promote some other value?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a, q_y, v, +\rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) = q_y$; $v \in Av^x$; $\delta(q_x, q_y, v) = +$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s'\rangle$ or an AOA $\langle q'_x, a', q'_y, v', s'\rangle$ iff $q_x \approx q'_x$, $a \neq a'$, $v \neq v'$, $s' = +$.*

See Fig. 5 for an illustration of example VATS that could produce a cq11-argument.

Definition 19: **A cq13-argument** *Answers the question* 'Is the action possible?'. *It is constructed from a VATS S^x and denoted $\langle a\rangle$ s.t. $a \notin Ac^x$. It challenges any AOA that includes q'_x, a', q'_y in its definition, iff $a = a'$.*

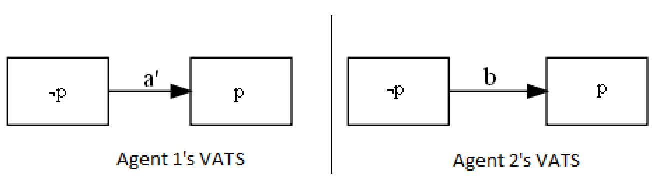

Fig. 7: Illustration of a cq13-argument (Definition 19). Agent 2 will assert a cq13-argument when agent 1 asserts an AOA to achieve p. No values are shown as no values occur in the definition of a cq13-argument.

Definition 20: **A cq14-argument** *Answers the question* 'Are the consequences as described possible?'. *It is constructed from a VATS S^x and denoted $\langle q_x, a\rangle$ s.t. $q_x = q_0^x$; $a \in Ac^x$; $\tau^x(q_x, a) \notin Q^x$. It challenges any AOA that includes q'_x, a', q'_y in its definition, iff $q_x \approx q'_x$, $a = a'$.*

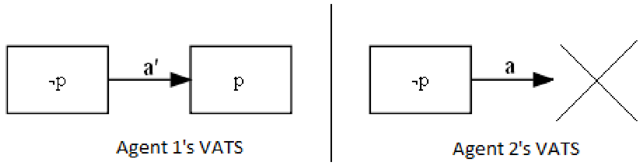

Fig. 8: Illustration of a cq14-argument (Definition 21). Agent 2 will assert a cq14-argument when agent 1 asserts an AOA to achieve p. No values are shown as no values occur in the definition of a cq14-argument.

Definition 21: **A cq15-argument** *Answers the question* 'Can the desired goal be realised?'. *It is constructed from a VATS S^x and denoted $\langle \neg p\rangle$ s.t. $p \in \Phi^x$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s'\rangle$ iff $p = p'$ and $(\forall q \in Q^x)(p \notin \pi(q))$.*

See Fig. 1 for an illustration of example VATS that could produce a cq15-argument.

Definition 22: **A cq16-argument** *Answers the question* 'Is the value indeed a legitimate value?'. *It is constructed from a VATS S^x and denoted $\langle v, -\rangle$ s.t. $v \notin Av^x$. It challenges an AOA $\langle q'_x, a', q'_y, p', v', s'\rangle$ or an AOA $\langle q'_x, a', q'_y, v', s'\rangle$ iff $v = v'$.*

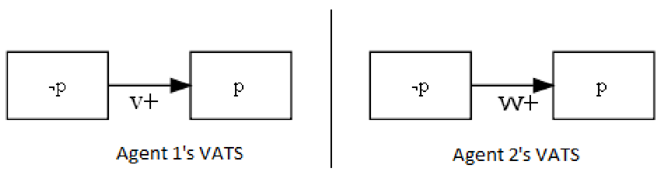

Fig. 9: Illustration of a cq16-Argument (Definition 22). Agent 2 will assert a cq16-argument when agent 1 asserts an AOA to achieve p. No actions are shown as an action does not occur in the definition of a cq16-argument.

Missing from the above list are CQ1 (are the believed circumstances true?), CQ12 (are the circumstances as described possible?) and CQ17 (is the other agent guaranteed to execute its part of the desired joint action?); their omission is explained here.

We assume that cooperative agents all accept the outcome of the inquiry dialogues and hence the representation issues concerning conflicting views of the initial state (as raised by CQ1 and CQ12) will be resolved. Note that under this assumption the outcome of the inquiry will be accepted by all agents even though it may be possible for an agent to construct a relevant counter argument.

The actual reasons for agents accepting one b-argument over another should be application dependant. For example in a safety critical system, the presence of one b-argument for a safety critical proposition maybe enough to convince the agents to accept that proposition over its negation. In other applications a simple majority vote could be sufficient. Lastly CQ17 is omitted as the system is not currently concerned with joint actions, though the use of an AATS lends itself to this, as we will explore in future work.

3.3 Extending the pAct Protocol

The protocol this implementation uses, named the $pAct$ protocol extends the one presented in [3] by including the extra critical questions so that the agents can use them in a dialogue move. It returns the set of possible moves that are legal for each agent in the dialogue when the current dialogue is of the type $pAct$.

The $pAct$ protocol takes the top level dialogue from the set of all dialogues $\mathcal{D}$, the identifier of the agent from the set of all identifiers $\mathcal{I}$ and returns the set of legal moves which is an element of the set of all subsets of the set of all moves $\mathcal{M}$. These definitions are the same for the inquiry protocol. Possible moves for an agent in the $pAct$ dialogue are: an assertion of an a-argument to achieve the dialogue goal; an assertion of a cq-argument; a move to close the $pAct$ dialogue or a move to open or close a nested inq dialogue.

3.4 Defining the Inquiry Protocol

An Inquiry Protocol needs to be formally defined as the details were left out of [3]. This protocol returns the set of possible moves that are legal for each agent in the dialogue when the current dialogue is of the type Inq.

This protocol will not allow any proposition to become a claim of a b-argument without supporting evidence. Supporting evidence takes the form of defeasible facts or a fully supported defeasible rule. A defeasible rule λ is fully supported when there is a defeasible derivation for the head of the rule that includes the rule and can be constructed from the union of all the commitment stores.

The protocol works by firstly allowing each agent to assert all its relevant beliefs that are not already present in the commitment store ($\Xi_{\mathsf{a}}(D_1^t, x)$). A belief can be either a defeasible rule or a defeasible fact. A defeasible fact is relevant if it is an element of the dialogue topic (Ξ_{a} (2)(ii,a)) or an element of a defeasible rule in the combined commitment store of all the agents (Ξ_{a} (2)(ii,b)). A defeasible rule is relevant if its consequent returns a defeasible fact that is an element of the dialogue topic (Ξ_{a} (3)(ii,a)) or an element of another defeasible rule in the CSs (Ξ_{a}(3)(ii,b)). Secondly in $\Xi_{\mathsf{b}}(D_1^t, x)$ the agent checks to see if any of its asserted beliefs are now fully supported ($\Phi \subseteq CSs$). If they are, these beliefs get asserted as b-arguments in the form $B = \langle \Phi, \phi \rangle$ if they have not been already ($B \notin CSs$). Each agent only asserts a b-argument with claim ϕ if it asserted the belief that included ϕ ($\phi \in CS_x^t$). This is to eliminate multiple assertions of b-arguments. Lastly if the agents cannot assert anything new then the only move that will be returned is the 'close dialogue' move.

Definition 23: *The* **Inquiry protocol** *is a function* $\Xi : \mathcal{D} \times \mathcal{I} \mapsto \wp(\mathcal{M})$. *If* D_1^t *is a top-level dialogue s.t.* $\mathsf{Current}(D_1^t) = D_r^t$, $\mathsf{Turn}(D_r^t) = x$, $\mathsf{Participants}(D_r^t) = \Lambda = [x_1, \ldots, x_n]$, $CSs = \bigcup_{\forall x_i \in \{x_1, \ldots, x_n\}} CS_{x_i}^t$, $\mathsf{Type}(D_r^t) = Inq$, $\mathsf{Topic}(D_r^t) = \Phi^{\mathsf{Initiator}(D_r^t)}$ *and* $1 \leq t$, *then* $\Xi(D_1^t, x)$ *is*

$$\Xi_{\mathsf{a}}(D_1^t, x) \cup \Xi_{\mathsf{b}}(D_1^t, x) \cup \{\langle x, close, dialogue(Inq, \Phi^{\mathsf{Initiator}(D_r^t)}, \Lambda)\rangle\}$$

where

$\Xi_{\mathsf{a}}(D_1^t, x) = \{\langle x, assert, \Phi \rangle |$

(1) $\Phi \neq \emptyset$ *where* Φ *is a set of beliefs, and*

(2) $\forall \phi \in \Phi$ *where* ϕ *is a defeasible fact:*
(i) $\phi \notin CSs, \phi \in \Sigma^x$ *, and*
either (ii,a) $\phi \in \mathsf{Topic}(D^t_r)$ *,*
or (ii,b) $\exists \lambda \in CSs$ *s.t.* $\phi \in \mathsf{DefeasibleSection}(\lambda)$
(3) $\forall \lambda \in \Phi$ *where* λ *is a defeasible rule:*
(i) $\lambda \notin CSs, \lambda \in \Sigma^x$ *, and*
either (ii,a) $\mathsf{DefeasibleProp}(\lambda) \in \mathsf{Topic}(D^t_r)$ *,*
or (ii,b) $\exists \lambda' \in CSs$ *s.t.* $\mathsf{DefeasibleProp}(\lambda) \in \mathsf{DefeasibleSection}(\lambda')$

$\Xi_{\mathsf{b}}(D^t_1, x) = \{\langle x, assert, \Upsilon \rangle |$
(1) $\Upsilon \neq \emptyset, \Upsilon$ *is a set of b arguments , and*
(2) $\forall B \in \Upsilon$*:* $B = \langle \Phi, \phi \rangle$ *is a b-argument,* $\Phi \subseteq CSs$*,* $\phi \in CS^t_x$ *and* $B \notin CSs$

3.5 pAct Strategy

Agents of this system use the $pAct$ strategy. This strategy either opens a $pAct$ dialogue if the agent is the dialogue initiator or selects one move out of the set of legal moves returned from the correct protocol (the $pAct$ protocol if $\mathsf{Type}(\mathsf{Current}(\mathcal{D}^t_1)) == pAct$, else the $inquiry$ protocol). The strategy is honest as agents assert only arguments that can be constructed from their knowledge bases.

When using the $pAct$ strategy, agents prefer a move to open an inquiry dialogue over assert moves over close moves. This means that once the agents start asserting AOAs, they already know the truth value of all their propositions, due to performing an open inq dialogue move first and so all the AOAs presented in the subsequent $pAct$ dialogue relate to the actual world state. Also, as the close move is the least preferred no agent will attempt to close the dialogue until it has run out of other moves and so the dialogue is exhaustive.

4 Implementation

The implementation of the framework detailed in this paper uses the *Java Agent DEvelopment Framework (JADE)*[6] to facilitate the storage, modelling and use of the agents' epistemic and normative knowledge at runtime. The user can inspect: the initial Value-based Argumentation Framework (VAF) that is used to evaluate the arguments produced by the agents following the protocol; the preferred extension of the VAF; and the final recommended action. Other elements that can be inspected include: the VATS of all the agents in the dialogue; further details on both the b-arguments, a-arguments and critical questions; and lastly the ability to view the complete resulting dialogue. In addition, the user can modify the value order after the dialogue has terminated, which may result in different recommended actions being generated.

Agents are modelled within a closed environment, and communicate by broadcasting messages to other agents within the dialogue via a shared blackboard. The use of a blackboard to record communicative acts eliminates the need for agents to individually retain the dialogue history. Agents take turns to update the blackboard, to avoid the need for complex coordination strategies.

[6] http://jade.tilab.com/

All attacks between arguments in the VAF are computed after the dialogue has terminated. As we assume a non-strategic approach, this allows agents to individually assert all relevant arguments prior to the VAF determining the arguments acceptability status. In future work we intend to investigate strategies.

Several issues were identified whilst developing a model to select a final recommended action from the asserted AOAs. Since several critical questions concern *problem formulation* issues [1], not all arguments have an associated value (e.g. those derived from CQ13, CQ15, CQ16); instead these arguments have been implemented to automatically defeat any other argument that they attack, as they are assigned the value 'truth' which always ranks higher than any other value, according to [2].

Finally, scenarios can arise whereby b-arguments may claim logical contradictions (e.g. p and $\neg p$). The current implementation resolves this issue by assuming that the assertion holds (i.e. p) and the contradiction is ignored (i.e. $\neg p$). This conflict strategy was selected because of time constraints but a conflict strategy should be chosen that suits the particular application the dialogue system is deployed in.

4.1 Implementation example

Our system has so far been evaluated through the use of examples scenarios. One such scenario will now be detailed to show how our system works at runtime.

Consider three agents engaged in a dialogue about the recent UK tuition fee debate. These agents represent the student union SU (Figure 10), the University College Union UCU (Figure 11) and the government GOV (Figure 12). There are three values present in this simplified version of the debate: **E** representing the *equality* of future students when compared to previous ones; **JS** representing *job security* for public sector educational workers; and **NES** representing *national economic security*. Along with these values, there are two possible actions: **raiseTuitionFeesUpFront** representing raising the tuition fees for all new students; and **graduateTax** representing a tax for all workers who hold a degree.

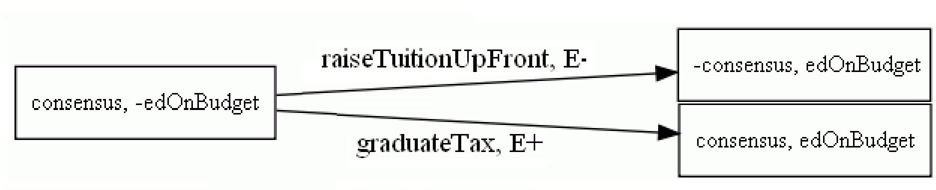

Fig. 10: The VATS for SU. **consensus** represents if the majority of the student union members are happy with the current fee level and **edOnBudget** represents if the educational system is currently on budget.

As shown by their respective VATS each agent has different views on this scenario. The SU agent thinks raising tuition fees up front would unfairly affect future students and thinks that a graduate tax would be a more appropriate alternative. SU's belief base

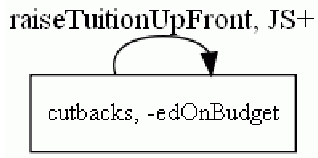

Fig. 11: The VATS for UCU. **cutbacks** represents if budget cutbacks are needed in the majority of universities. **edOnBudget** remains the same as in Fig. 11.

Fig. 12: The VATS for GOV. **debate** represents that there is currently a parliamentary review on how the educational budget can be improved. **edOnBudget** remains the same as in Fig. 11.

is as follows:

$$\Sigma^{SU} = \{inFavourFees(X) > againstFees(Y) \rightarrow consensus,\\ inFavourFees(80\%), againstFees(20\%)\}$$

The UCU agent does not think that implementing a graduate tax is a workable alternative but on the other hand, it believes that no action will balance the educational budget. UCU's belief base is as follows:

$$\Sigma^{UCU} = \{\neg edOnBudget \rightarrow cutbacks\}$$

Lastly the GOV agent does not view a graduate tax as a possible action but realises that every value will be affected if it decides to raise the tuition fees. GOV's belief base is as follows:

$$\Sigma^{GOV} = \{budgetReview \wedge educationalReview \rightarrow debate,\\ budgetReview, educationalReview,\\ expenditure(X) \leq estimatedReturn(Y) + acceptableLosses(Z) \rightarrow edOnBudget,\\ expenditure(X) > estimatedReturn(Y) + acceptableLosses(Z) \rightarrow \neg\, edOnBudget,\\ expenditure(13.1), estimatedReturn(7), acceptableLosses(3)\}$$

The GOV agent tries to resolve the issue of what action (or non-action) should be recommended by starting a dialogue with the goal being to achieve $edOnBudget$ i.e. keeping education spending on budget. The value order that is used in this example is **NES** $\succ$ **JS** $\succ$ **E**. The dialogue order is SU followed by UCU and lastly GOV. For this example CQ16 is not asserted since it is assumed that all the values are recognised by all the agents.

Firstly the agents all open inquiry dialogues to find their correct state. Agent SU opens an inquiry dialogue to find the truth values of $consensus$ and $edOnBudget$.

The beliefs and b-arguments asserted are:
B1: (a defeasible rule asserted by GOV) $expenditure(X) \leq estimatedReturn(Y) + acceptableLosses(Z) \rightarrow edOnBudget$.
B2: (a defeasible rule asserted by SU) $inFavourFees(X) > againstFees(Y) \rightarrow consensus$.
B3: (a defeasible fact asserted by GOV) $expenditure(13.1)$.
B4: (a defeasible fact asserted by GOV) $estimatedReturn(7)$.
B5: (a defeasible fact asserted by GOV) $acceptableLosses(3)$.
B6: (a defeasible fact asserted by SU) $inFavourFees(80\%)$.
B7: (a defeasible fact asserted by SU) $againstFees(20\%)$.
B8: (a b-argument asserted by SU) $\langle \{againstFees(20\%), inFavourFees(80\%), inFavourFees(X) > againstFees(Y) \rightarrow consensus\}, consensus \rangle$.

With no other b-arguments possible, the conclusion of this inquiry dialogue is that only $consensus$ is true in the current state. As no b-arguments could be generated for $edOnBudget$ and our implementation operates under the closed world assumption then $edOnBudget$ is presumed to be false.

For the inquiry dialogues all shared propositions are only discussed once because the agents' protocols are exhaustive and so each shared proposition would always find the same truth value in each discussion.

The beliefs and b-arguments asserted in UCU's inquiry dialogue, which has been opened to find/confirm the truth values of $cutbacks$ and $edOnBudget$ are:
B9: (a defeasible rule asserted by UCU) $\neg edOnBudget \rightarrow cutbacks$.
B10: (a defeasible rule asserted by GOV) $expenditure(X) < estimatedReturn(Y) + acceptableLosses(Z) \rightarrow \neg\, edOnBudget$.
B11: (a b-argument asserted by GOV) $\langle \{acceptableLosses(3), estimatedReturn(7), expenditure(13.1), expenditure(X) > estimatedReturn(Y) + acceptableLosses(Z) \rightarrow \neg edOnBudget\}, \neg edOnBudget \rangle$.
B12: (a b-argument asserted by UCU) $\langle \{acceptableLosses(3), estimatedReturn(7), expenditure(13.1), expenditure(X) > estimatedReturn(Y) + acceptableLosses(Z) \rightarrow \neg edOnBudget, \neg edOnBudget \rightarrow cutbacks\}, cutbacks \rangle$.

The beliefs and b-arguments asserted in GOV's inquiry dialogue, which has been opened to find/confirm the truth values of $debate$ and $edOnBudget$ are:
B13: (a defeasible rule asserted by GOV) $budgetReview \wedge educationalReview \rightarrow debate$.
B14: (a defeasible fact asserted by GOV) $budgetReview$.
B15: (a defeasible fact asserted by GOV) $educationalReview$.
B16: (a b-argument asserted by GOV) $\langle \{educationalReview, budgetReview, budgetReview \wedge educationalReview \rightarrow debate\}, debate \rangle$.

After all the inquiry dialogues are complete, the agents will start to produce arguments over what actions to perform using the $pAct$ dialogue. The AOAs that are constructed

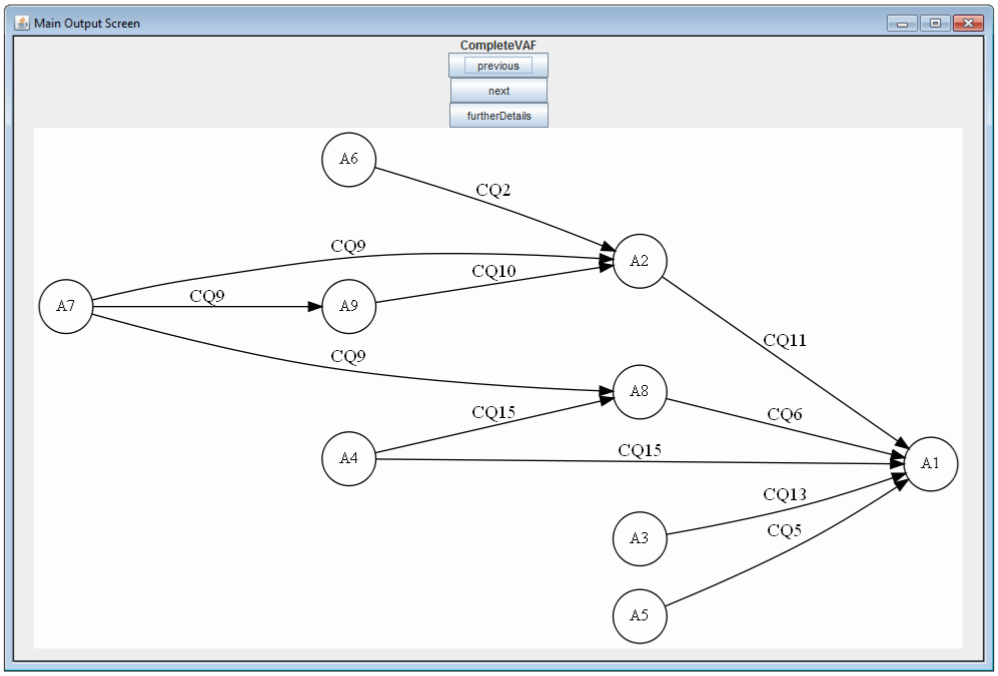

Fig. 13: VAF for the example dialogue. The nodes and edges represent the AOAs and attacks repectively. Each edge is labelled with its attack type.

in this example are [7]:

A1: (a-argument asserted by SU) As we are in state $[consensus, \neg edOnBudget]$, we should implement a **graduateTax**, which will achieve $edOnBudget$ and promote **equality**.

A2: (cq11-argument asserted by UCU) As we are in state $[cutbacks, \neg edOnBudget]$, we should **raiseTuitionFeesUpFront** which will promote **job security**.

A3: (cq13-argument asserted by UCU) A **graduateTax** is *not* a possible action.

A4: (cq15-argument asserted by UCU) Achieving $edOnBudget$ is *impossible*.

A5: (cq5-argument asserted by GOV) As we are in state $[debate, \neg edOnBudget]$, **raiseTuitionFeesUpFront**, would achieve the same state as **A1**.

A6: (cq2-argument asserted by GOV) As we are in state $[debate, \neg edOnBudget]$, **raiseTuitionFeesUpFront**, would achieve different state to **A2**.

A7: (cq9-argument asserted by GOV) As we are in state $[debate, \neg edOnBudget]$, **raiseTuitionFeesUpFront**, would demote **equality**.

A8: (cq-6 argument asserted by GOV) As we are in state $[debate, \neg edOnBudget]$, we should **raiseTuitionFeesUpFront** which will achieve $edOnBudget$ and promote **na-**

[7] The formal characterisation of the dialogue is ommitted solely for reasons of space. The semantic meaning of the arguments can be found in Section 3.2 and Definition 6. The full list of attacks is visualised in Fig. 14.

tional economic security.
A9: (cq10-argument asserted by gov) As we are in state $[debate, \neg edOnBudget]$, **raiseTuitionFeesUpFront** will promote **national economic security**.

Now after argument **A9** the dialogue closes as the only further arguments that can be asserted would be repetitions of information already present in the CS. As discussed earlier, the full set of arguments put forward during the dialogue can now be organised into a VAF that shows the attack relations between them, as can be seen in Figure 13.

Evaluating the VAF to determine which arguments are defeated yields the preferred extension and the final recommended action as can be seen in Figure 14.

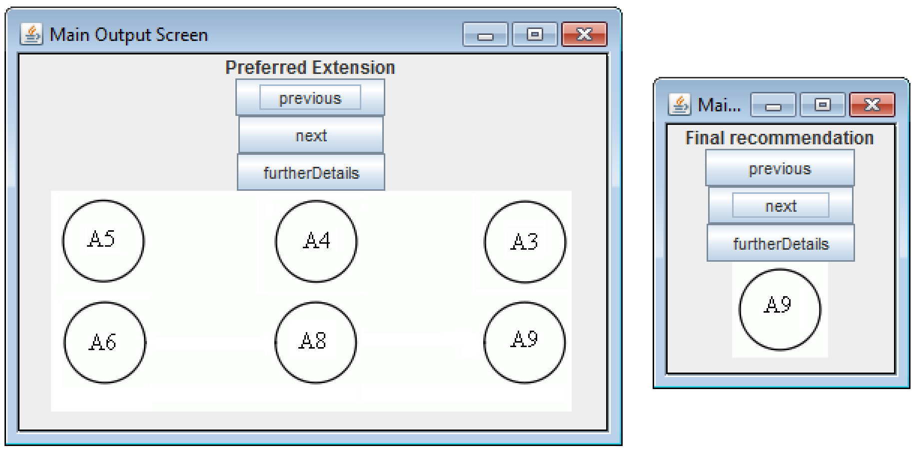

Fig. 14: The attack of **A7** on **A9** has not succeeded as **A9** promotes a higher value than **A7** according to the audience. **A9** is then recommended as it is the only argument that promotes a value.

5 Discussion and Concluding Remarks

In this paper we have taken a formal specification of a dialogue system for inquiry and persuasion over action, extended it, and subsequently implemented it to produce a working agent dialogue system. Our contribution is in terms of the extended formalism and the implemented system itself, with both providing grounds for future work.

The inclusion of additional critical questions enables agents to better identify ambiguities within their shared models, and thus construct additional arguments when looking to find some consensus. However, the existence of these additional critical questions could also undermine the ability of agents forming some consensus. This is in part due to the fact that in some contexts, all existing arguments could be defeated by a carefully selected question which, when posed, may result in no recommended action. Thus, it may be possible that the efforts of other agents to arrive at consensus may be undermined by an agent that possesses flawed normative and/or epistemic knowledge. Implementing the stages of practical reasoning from [1] may eliminate this problem, as the first stage (*problem formulation*) would resolve representation issues, the second stage (*epistemic reasoning*) would be represented by the inquiry dialogue and the third stage (*action selection*) would be represented by the persuasion over action dialogue.

An interesting future extension to this work would be to see how this system could be modified to allow for each agent to have its own preference order, instead of the currently implemented single global preference order. This modification could lead to a multi-agent system with self-interested agents, which could be further explored by introducing aspects of coalition formation.

Acknowledgements

Luke Riley is grateful for support from the EPSRC. Katie Atkinson was partially supported by the FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement No. 247228. Elizabeth Black is funded by the European Union Seventh Framework Programme (FP7/2007-2011), grant agreement 253911. The views expressed are those of the authors and are not necessarily representative of their project or funding body. We also thank the anonymous referees for their constructive comments.

References

1. K. Atkinson and T. J. M. Bench-Capon. Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artificial Intelligence*, 171(10–15):855–874, 2007.
2. T. J. M. Bench-Capon. Persuasion in practical argument using value based argumentation frameworks. *J. of Logic and Computation*, 13(3):429–48, 2003.
3. E. Black and K. Atkinson. Dialogues that account for different perspectives in collaborative argumentation. In *Proceedings of the Eighth International Conference on Autonomous Agents and Multi-Agent Systems*, pages 867–874, 2009.
4. E. Black and A. Hunter. An inquiry dialogue system. *JAAMAS*, 19(2):173–209, 2009.
5. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
6. A. J. García and G. R. Simari. Defeasible logic programming an argumentative approach. *Theory and Practice of Logic Prog.*, 4(1–2):95–138, 2004.
7. P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *J. of Logic, Language and Information*, 11(3):315–334, 2002.
8. D. H. P. McBurney and S. Parsons. The eightfold way of deliberation dialogue. *International Journal of Intelligent Systems*, 22(1):95–132, 2007.
9. H. Prakken. Formal systems for persuasion dialogue. *KER*, 21(2):163–188, 2006.
10. I. Rahwan, S. D. Ramchurn, N. R. Jennings, P. McBurney, S. Parsons, and L. Sonenberg. Argumentation-based negotiation. *KER*, 18(4):343–375, 2003.
11. C. Reed. Dialogue frames in agent communications. In *3rd Int. Conf. on Multi-Agent Systems*, pages 246–253, 1998.
12. D. Walton and E. Krabbe. *Commitment in dialogue: Basic concepts of interpersonal reasoning*. State University of New York Press, Albany NY, 1995.
13. M. Wooldridge and W. van der Hoek. On obligations and normative ability: Towards a logical analysis of the social contract. *J. of Applied Logic*, 3:396–420, 2005.