Autonomous Agents and Multi-Agent Systems manuscript No.
(will be inserted by the editor)

An Inquiry Dialogue System

Elizabeth Black · Anthony Hunter

Received: date / Accepted: date

Abstract The majority of existing work on agent dialogues considers negotiation, persuasion or deliberation dialogues; we focus on inquiry dialogues, which allow agents to collaborate in order to find new knowledge. We present a general framework for representing dialogues and give the details necessary to generate two subtypes of inquiry dialogue that we define: argument inquiry dialogues allow two agents to share knowledge to jointly construct arguments; warrant inquiry dialogues allow two agents to share knowledge to jointly construct dialectical trees (essentially a tree with an argument at each node in which a child node is a counter argument to its parent). Existing inquiry dialogue systems only model dialogues, meaning they provide a protocol which dictates what the possible legal next moves are but not which of these moves to make. Our system not only includes a dialogue-game style protocol for each subtype of inquiry dialogue that we present, but also a strategy that selects exactly one of the legal moves to make. We propose a benchmark against which we compare our dialogues, being the arguments that can be constructed from the union of the agents' beliefs, and use this to define soundness and completeness properties that we show hold for all inquiry dialogues generated by our system.

Keywords agent interaction · argumentation · inquiry · dialogue · cooperation

1 Introduction

Dialogue games are now a common approach to characterizing argumentation-based agent dialogues (e.g. [33,39,42]). Dialogue games are normally made up of a set of communicative acts called moves, and sets of rules stating: which moves it is legal to make at any point in a dialogue (the *protocol*); the effect of making a move; and when a dialogue terminates. One attraction of dialogue games is that it is possible

E. Black
COSSAC: IRC in Cognitive Science and Systems Engineering (www.cossac.org),
Department of Engineering Science, University of Oxford, Oxford, UK
E-mail: lizblack@robots.ox.ac.uk

A. Hunter
Department of Computer Science, University College London, London, UK

to embed games within games, allowing complex conversations made up of nested dialogues of more than one type (e.g. [34,45]). Most of the work so far has looked only at *modelling* different types of dialogue from the influential Walton and Krabbe typology [49], meaning that they provide a protocol which dictates what the possible legal next moves are but not which one of these legal moves to make. Here we present a generative system, as we not only provide a protocol but also provide a strategy for selecting exactly one of the legal moves to make.

Examples of dialogue systems which model each of the five main Walton and Krabbe dialogue types are: *information-seeking* [30,40] (where participants aim to share knowledge); *inquiry* [32,40] (where participants aim to jointly discover new knowledge); *persuasion* [4,17] (where participants aim to resolve conflicts of opinion); *negotiation* [3,30, 35,47] (where participants who need to cooperate aim to agree on a method for doing this that resolves their conflicting interests); and *deliberation* [29] (where participants aim to jointly decide on a plan of action).

Walton and Krabbe classify their dialogue types according to three characteristics: the *initial situation* from which the dialogue arises; the *main goal* of the dialogue, to which all the participating agents subscribe; and the *personal aims* of each individual agent. This article focuses on two subtypes of inquiry dialogue that we define. Walton and Krabbe define an inquiry dialogue as arising from an initial situation of "general ignorance" and as having the main goal to achieve the "growth of knowledge and agreement". Each individual participating in an inquiry dialogue has the goal to "find a 'proof' or destroy one" [49, page 66].

We have previously proposed a dialogue system [12] for generating a subtype of inquiry dialogue that we call *argument inquiry.* In an argument inquiry dialogue, the 'proof' that the participating agents are jointly searching for takes the form of an argument for the topic of the dialogue. In this article we adapt the system proposed in [12] to generate a second subtype of inquiry dialogue that we call *warrant inquiry.* In a warrant inquiry dialogue, the 'proof' that the participating agents are jointly searching for takes the form of a dialectical tree (essentially a tree with an argument at each node, where each child node is a counter argument to its parent and that has at its root an argument whose claim is the topic of the dialogue). Warrant inquiry dialogues are so called as the dialectical tree produced during the dialogue may act as a warrant for the argument at its root.

The goal, then, of the participants in an *argument inquiry dialogue* is to *share beliefs* in order to *jointly construct arguments* for a specific claim that none of the individual participants may construct from their own personal beliefs alone; the goal of agents taking part in a *warrant inquiry dialogue* is to *share arguments* in order to *jointly construct a dialectical tree* that none of the individual participants may construct from their own personal beliefs alone. In an argument inquiry dialogue, the agents wish to exchange beliefs in order to jointly construct arguments for a particular claim; however, an argument inquiry dialogue does not allow the agents to determine the acceptability of the arguments constructed (i.e. whether the arguments are ultimately defeated by any other conflicting arguments). In a warrant inquiry dialogue, the agents are interested in determining the acceptability of a particular argument; they do this by jointly constructing a dialectical tree that collects all the arguments that may be relevant to the acceptability of the argument in question.

Argument inquiry dialogues are often embedded within warrant inquiry dialogues. Without embedded argument inquiry dialogues, the arguments that can be exchanged within a warrant inquiry dialogue potentially miss out on useful arguments that involve unexpressed beliefs of the other agent. (We have presented the system for generating argument inquiry dialogues previously [12]; we present it again here as it is necessary for generating warrant inquiry dialogues.) The main contribution of this article is a protocol and strategy sufficient to generate sound and complete warrant inquiry dialogues.

As far as we are aware, there are only two other groups that have proposed inquiry protocols. Amgoud, Maudet, Parsons and Wooldridge proposed a protocol for general argument inquiry dialogues (e.g. [3,39]), however this protocol can lead to unsuccessful dialogues in which no argument for the topic is found even when such an argument does exist in the union of the two agents' beliefs. In [32], McBurney and Parsons present an inquiry protocol that is similar in spirit to our warrant inquiry protocol in that it allows the agents involved to dialectically reason about the acceptability of an argument given a set of arguments and the counter argument relations between them. Although their protocol allows agents to exchange arguments in order to carry out the dialectical reasoning, it does not allow the agents to jointly construct arguments and so the reasoning that they participate in may be incomplete in the sense that it may miss important arguments that can only be constructed jointly by two or more agents. Neither of these groups have proposed a strategy for use with their inquiry protocol, i.e. their systems model inquiry dialogues but are not sufficient to generate them.

A key contribution of this work is that we not only provide a protocol for modelling inquiry dialogues but we also provide a specific strategy to be followed, making this system sufficient to also *generate* inquiry dialogues. Other works have also considered the generation of dialogues. For example, [44] gives an account of the different factors which must be considered when designing a dialogue strategy. Parsons *et al.* [39] explore the effect of different agent attitudes, which reduce the set of legal moves from which an agent must choose a move but do not select exactly one of the legal moves to make. Pasquier *et al.*'s cognitive coherence theory [41] addresses the pragmatic issue of dialogue generation, but it is not clear what behaviour this would produce. Both [2] and [31] propose a formalism for representing the private strategy of an agent to which argumentation is then applied to determine the move to be made at a point in a dialogue; however, neither give a specific strategy for inquiry dialogues.

Whilst much of the work on argumentation has been intended for use in adversarial domains such as law (e.g. [5,7,10,28,43]), we have been inspired by the *cooperative* medical domain. In adversarial domains, agents participating in a dialogue are typically concerned with defending their own arguments and defeating the arguments of their opponents; in cooperative domains, agents instead aim to arrive at the best joint outcome, even if this means accepting the defeat of their own arguments. Medical knowledge is typically uncertain and often incomplete and inconsistent, making argumentation an attractive approach for carrying out reasoning and decision making in the medical domain [24]. Inquiry dialogues are a type of dialogue that are of particular use in the medical domain, where it is often the case that people have distinct types of knowledge and so need to interact with others in order to have all the information necessary to make a decision.

Another important characteristic of the medical domain is that is safety-critical [23]; if our dialogue system is to be used in such a domain, it is essential that the dialogues our system produces arrive at the appropriate outcome. We wish the outcome of our dialogues to be predetermined by the fixed protocol, the strategy being followed and the belief bases of the participating agents; i.e. given the agents' beliefs, we want to know what outcome they will arrive at and that this will be the appropriate outcome.

As discussed in [39], this can be be viewed as a positive or negative feature of a dialogue system depending on the application. In a more competitive environment it may well be the case that one would wish it to be possible for agents to behave in an intelligent manner in order to influence the outcome of a dialogue. However, we want our dialogues to always lead to the 'ideal' outcome. That is to say, we want the dialogues generated by our system to be sound and complete, in relation to some standard benchmark. We compare the outcome of our dialogues with the outcome that would be arrived at by a single agent that has as its beliefs the union of both the agents participating in the dialogue's beliefs. This is, in a sense, the 'ideal' situation, where there are clearly no constraints on the sharing of beliefs.

As the dialogue outcome we are aiming for is the same as the outcome we would arrive at if reasoning with the union of the agents' beliefs, a natural question to ask is why not simply pool the agents' beliefs and then reason with this set? In some situations, it may indeed be more appropriate to pool the agents' beliefs (e.g. as part of computer supported collaborative learning, [27]); however, in many real world scenarios, such as within the medical domain, there are often privacy issues that would restrict the agents from simply pooling all beliefs; what we provide here can be viewed as a mechanism for a joint directed search that ensures the agents only share beliefs that could be relevant to the topic of the dialogue. It could also be the case that the belief bases of the agents are so vast that the communication costs involved in pooling all beliefs would be prohibitive.

The main contribution of this article is a system for generating sound and complete warrant inquiry dialogues. We build on the system we proposed in [12] in the sense that we use the same underlying formalism for modelling and generating dialogues. However, whilst in [12] we provided only a protocol and strategy for generating sound and complete argument inquiry dialogues, here we also include a protocol and strategy for generating warrant inquiry dialogues and give soundness and completeness results for all inquiry dialogues generated by our system. We have presented the details relating to argument inquiry dialogues (that were previously given in [12]) again in this article as they are necessary for generating warrant inquiry dialogues.

The rest of this article proceeds as follows. In Section 2 we present the knowledge representation used by the agents in our system and define how this knowledge can be used to construct arguments, and in Section 3 we present a method for constructing a dialectical tree in order to carry out a dialectical analysis of a set of arguments. Sections 2 and 3 thus present the argumentation system on which this dialogue system operates, which is based on García and Simari's Defeasible Logic Programming (DeLP) [25]. The presentation of the argumentation system here differs only slightly from that in [25] and does not represent a contribution of this work.

In Section 4 we define the general framework used to represent dialogues. In Section 5 we give the protocols for modelling both argument inquiry and warrant inquiry dialogues, and also give a strategy for use with these protocols that allows agents to generate the dialogues (completing our inquiry dialogue system). In Section 6 we define soundness and completeness properties for both argument inquiry and warrant inquiry dialogues and show that these properties hold for all well formed inquiry dialogues generated by our system. Finally, in Section 7 we discuss other work related to this and in Section 8 we summarise our conclusions.

2 Knowledge representation and arguments

We adapt García and Simari's Defeasible Logic Programming (DeLP) [25] for representing each agent's beliefs. DeLP is a formalism that combines logic programming with defeasible argumentation. It is intended to allow a single agent to reason internally with inconsistent and incomplete knowledge that may change dynamically over time and has been shown to be applicable in different real-world contexts (e.g. [14, 26]). It provides a warrant procedure, which we will present in Section 3, that applies a dialectical reasoning mechanism to a set of arguments in order to decide whether a particular argument from that set is warranted.

The presentation here differs only slightly from that in [25]. García and Simari assume that, as well as a set of defeasible rules, there is a set of strict rules. They also assume that facts are non-defeasible. As we are inspired by the medical domain (in which we know knowledge to often be incomplete, unreliable and inconsistent), we wish *all* knowledge to be treated as defeasible. We deal with this by assuming the sets of strict rules and facts are empty and by defining a defeasible fact (essentially a defeasible rule with an empty body). We use a restricted set of propositional logic and assume and that a literal is either an atom α or a negated atom $\neg\alpha$. We use the notation $\vdash$ to represent the classical consequence relation; we use $\bot$ to represent classical contradiction; we use $\overline{\alpha}$ to represent the complement of α, i.e. $\alpha = a$ if and only if $\overline{\alpha} = \neg a$, $\alpha = \neg a$ if and only if $\overline{\alpha} = a$.

Definition 1 A **defeasible rule** 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.

The warrant procedure defined in [25] assumes that a formal criterion exists for comparing two arguments. We use a preference ordering across all knowledge, from which a preference ordering across arguments is derived. We imagine that the preference ordering on medical knowledge would depend on the knowledge source [50]; we might assume, for example, that knowledge from an established clinical guideline is preferred to knowledge that has resulted from a small clinical trial. We associate a preference level with a defeasible rule or defeasible fact to form a belief; the lower the preference level the *more* preferred the belief.

Definition 2 A **belief** is a pair (ϕ, L) where ϕ is either a defeasible fact or a defeasible rule, and $L \in \{1, 2, 3, ...\}$ is a label that denotes the **preference level** of the belief. The function pLevel returns the preference level of the belief: $\mathsf{pLevel}((\phi, L)) = L$. The set of all beliefs is denoted $\mathcal{B}$.

We make a distinction between beliefs in defeasible facts (called *state beliefs*, as these are beliefs about the state of the world) and beliefs in defeasible rules (called *domain beliefs*, as these are beliefs about how the domain is expected to behave). We also consider the set of defeasible facts and the set of defeasible rules, and the union of these two sets.

Definition 3 A **state belief** is a belief (ϕ, L) where ϕ is a defeasible fact. The set of all state beliefs is denoted $\mathcal{S}$. A **domain belief** is a belief (ϕ, L) where ϕ is a defeasible rule. The set of all domain beliefs is denoted $\mathcal{R}$. The set of all defeasible facts is denoted $\mathcal{S}^\star = \{\phi \mid (\phi, L) \in \mathcal{S}\}$. The set of all defeasible rules is denoted $\mathcal{R}^\star = \{\phi \mid (\phi, L) \in \mathcal{R}\}$. The set of all defeasible rules and all defeasible facts is denoted $\mathcal{B}^\star = \{\phi \mid (\phi, L) \in \mathcal{B}\} = \mathcal{S}^\star \cup \mathcal{R}^\star$.

We assume that there are always exactly two agents (*participants*) taking part in a dialogue, each with its own identifier taken from the set $\mathcal{I} = \{1, 2\}$. Although we have restricted the number of participants to two here for the sake of simplicity, we believe it is straightforward to adapt the system to allow multiple participants. Many of the difficult issues associated with multi-party dialogues (e.g. What are the agents' roles? How to manage turn taking? Who should be addressed with each move? [18]) can be easily overcome here due to the collaborative and exhaustive nature of the dialogues we are considering (e.g. all agents have the same role; each agent is assigned a place in a sequence and that sequence is followed for turn taking; each move made is broadcast to all the participants). We are currently working on multi-party dialogues that use the framework presented here.

Each agent has a, possibly inconsistent, belief base.

Definition 4 A **belief base** associated with an agent x is a finite set, denoted Σ^x, such that $\Sigma^x \subseteq \mathcal{B}$ and $x \in \mathcal{I} = \{1, 2\}$.

Example 1 Consider the following belief base associated with agent 1.

$$\Sigma^1 = \left\{ \begin{array}{l} (a, 1), (\neg a, 1), (b, 2), (d, 1), \\ (a \rightarrow c, 3), (b \rightarrow \neg c, 2), (d \rightarrow \neg b, 1), (\neg a \rightarrow \neg b, 1) \end{array} \right\}$$

The top four elements are state beliefs and we can see that the agent conflictingly believes strongly in both a and $\neg a$. The bottom four elements are all domain beliefs. $\mathsf{pLevel}((a, 1)) = 1$. $\mathsf{pLevel}((\neg a, 1)) = 1$. $\mathsf{pLevel}((b, 2)) = 2$. $\mathsf{pLevel}((d, 1)) = 1$. $\mathsf{pLevel}((a \rightarrow c, 3)) = 3$. $\mathsf{pLevel}((b \rightarrow \neg c, 2)) = 2$. $\mathsf{pLevel}((d \rightarrow \neg b, 1)) = 1$. $\mathsf{pLevel}((\neg a \rightarrow \neg b, 1)) = 1$. Recall that the lower the pLevel value, the more preferred the belief.

We now define what constitutes a defeasible derivation. This has been adapted slightly from [25] in order to deal with our definition of a belief.

Definition 5 Let Ψ be a set of beliefs and α a defeasible fact. 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 such that α_n is the defeasible fact α and each literal α_m ($1 \leq m \leq n$) is in the sequence because:

- (α_m, L) is a state belief in Ψ, or
- $\exists(\beta_1 \wedge \ldots \wedge \beta_j \rightarrow \alpha_m, L') \in \Psi$ s.t. every literal β_i ($1 \leq i \leq j$) is an element α_k preceding α_m in the sequence ($k < m$).

The function $\mathsf{DefDerivations} : \wp(\mathcal{B}) \mapsto \mathcal{S}^\star$ returns the set of literals that can be defeasibly derived from a set of beliefs Ψ such that $\mathsf{DefDerivations}(\Psi) = \{\phi \mid$ there exists $\Phi \subseteq \Psi$ such that $\Phi \mid\!\sim \phi\}$.

Example 2 If we continue with the running example started in Example 1 we see that the following defeasible derivations exist from Σ^1.
a. $\neg a$. b. d. a, c. $b, \neg c$. $d, \neg b$. $\neg a, \neg b$.

We now define an argument as being a minimally consistent set from which the claim can be defeasibly derived.

Definition 6 An **argument** A constructed from a set of, possibly inconsistent, beliefs Ψ ($\Psi \subseteq \mathcal{B}$) is a tuple $\langle \Phi, \phi \rangle$ where ϕ is a defeasible fact and Φ is a set of beliefs such that:

1. $\Phi \subseteq \Psi$,
2. $\Phi \mid\sim \phi$,
3. $\forall \phi, \phi'$ s.t. $\Phi \mid\sim \phi$ and $\Phi \mid\sim \phi'$, it is not the case that $\phi \cup \phi' \vdash \perp$,
4. there is no subset of Φ that satisfies (1-3).

Φ is called the **support** of the argument and is denoted $\mathsf{Support}(A)$; ϕ is called the **claim** of the argument and is denoted $\mathsf{Claim}(A)$. For two arguments A_1 and A_2, A_1 is a **subargument** of A_2 iff $\mathsf{Support}(A_1) \subseteq \mathsf{Support}(A_2)$. The set of all arguments that can be constructed from a set of beliefs Ψ is denoted $\mathcal{A}(\Psi)$.

Example 3 Continuing the running example, the following arguments can be constructed by the agent.

$$\mathcal{A}(\Sigma^1) = \left\{ \begin{array}{ll} a_1 = \langle\{(a,1)\}, a\rangle, & a_5 = \langle\{(a,1),(a \to c,3)\}, c\rangle, \\ a_2 = \langle\{(\neg a,1)\}, \neg a\rangle, & a_6 = \langle\{(b,2),(b \to \neg c,2)\}, \neg c\rangle, \\ a_3 = \langle\{(b,2)\}, b\rangle, & a_7 = \langle\{(d,1),(d \to \neg b,1)\}, \neg b\rangle, \\ a_4 = \langle\{(d,1)\}, d\rangle, & a_8 = \langle\{(\neg a,1),(\neg a \to \neg b,1)\}, \neg b\rangle \end{array} \right\}$$

Note, a_1 is a subargument of a_5, a_2 is a subargument of a_8, a_3 is a subargument of a_6 and a_4 is a subargument of a_7. Every argument is a subargument of itself.

As Ψ may be inconsistent, there may be conflicts between arguments within $\mathcal{A}(\Psi)$.

Definition 7 Let A_1 and A_2 be two arguments. A_1 is in **conflict** with A_2 iff $\mathsf{Claim}(A_1) \cup \mathsf{Claim}(A_2) \vdash \perp$ (i.e. $\mathsf{Claim}(A_1) = \overline{\mathsf{Claim}(A_2)}$, as the claim of an argument is always a literal).

Example 4 Continuing the running example, a_1 is in conflict with a_2, a_3 is in conflict with a_7, a_3 is in conflict with a_8, and a_5 is in conflict with a_6.

Note that, as we are using $\vdash$ to represent classical implication, two arguments are in conflict with one another if and only if their claims are the complement of one another. For certain applications, we may need a different definition of conflict between two arguments, depending on the meaning of the arguments' claims and the purpose of the argumentation process; for example, if the purpose of the argumentation is to decide between alternative actions to try to achieve a goal, we may want to define two arguments as being in conflict if their claims are two distinct actions.

We now define the attack relationship between arguments.

Definition 8 Let A_1, A_2 and A_3 be arguments such that A_3 is a subargument of A_2. A_1 **attacks** A_2 at subargument A_3 iff A_1 is in conflict with A_3.

Note that in the previous definition the subargument A_3 is unique, as we will now show. A_1 and A_3 conflict, and so the claim of A_1 is the negation of the claim of A_3. Let us assume that there is another disagreement sub-argument such that A_1 attacks A_2 at A_4, A_1 and A_4 conflict, and so the claim of A_1 is the negation of the claim of A_4. As A_1 is a literal, this means that the claim of A_4 is the same as the claim of A_3. As an argument is minimal, this means that A_3 and A_4 must be the same arguments.

Example 5 Continuing the running example: a_1 attacks a_2 at a_2, a_1 attacks a_8 at a_2, a_2 attacks a_1 at a_1, a_2 attacks a_5 at a_1, a_3 attacks a_7 at a_7, a_3 attacks a_8 at a_8, a_5 attacks a_6 at a_6, a_6 attacks a_5 at a_5, a_7 attacks a_3 at a_3, a_7 attacks a_6 at a_3, a_8 attacks a_3 at a_3, and a_8 attacks a_6 at a_3.

Given that we know one argument attacks another, we need a mechanism for deciding whether the attacking argument successfully defeats the argument being attacked or not. We base this on the preference level of the argument, which is equal to that of the least preferred belief used in its support.

Definition 9 Let A be an argument. The **preference level** of A, denoted $\mathsf{pLevel}(A)$, is equal to $\mathsf{pLevel}(\phi)$ such that:

1. $\phi \in \mathsf{Support}(A)$,
2. $\forall \phi' \in \mathsf{Support}(A), \mathsf{pLevel}(\phi') \leq \mathsf{pLevel}(\phi)$.

Given that we have an argument A_1 that attacks an argument A_2 at subargument A_3, we say that A_1 *defeats* A_2 if the preference level of A_1 is the same as or less (meaning *more* preferred) than the preference level of A_3. If it is the same, then A_1 is a *blocking defeater* for A_2; if it is less, then A_1 is a *proper defeater* for A_2.

Definition 10 Let A_1, A_2 and A_3 be arguments such that A_3 is a subargument of A_2 and A_1 attacks A_2 at subargument A_3. A_1 is a **proper defeater** for A_2 iff $\mathsf{pLevel}(A_1) < \mathsf{pLevel}(A_3)$. A_1 is a **blocking defeater** for A_2 iff $\mathsf{pLevel}(A_1) = \mathsf{pLevel}(A_3)$.

Example 6 Continuing the running example: a_1 is a blocking defeater for a_2, a_1 is a blocking defeater for a_8, a_2 is a blocking defeater for a_1, a_2 is a blocking defeater for a_5, a_6 is a proper defeater for a_5, a_7 is a proper defeater for a_3, a_7 is a proper defeater for a_6, a_8 is a proper defeater for a_3 and a_8 is a proper defeater for a_6.

In this section we have proposed a criterion for deciding whether an argument A_1 that attacks an argument A_2 defeats it or not. In the next section we introduce the warrant procedure from [25], which allows an agent to decide whether, given a set of interacting arguments, a particular argument from this set is ultimately defeated or undefeated.

3 Dialectical analysis of arguments

Given a set of beliefs Ψ and an argument $A_1 \in \mathcal{A}(\Psi)$, in order to know whether A_1 is defeated or not, an agent has to consider each argument from $\mathcal{A}(\Psi)$ that attacks A_1 and decide whether or not it defeats it. However, a defeater of A_1 may itself be defeated by another argument $A_2 \in \mathcal{A}(\Psi)$. Defeaters may also exist for A_2, themselves of which may also have defeaters. Therefore, in order to decide whether A_1 is defeated, an agent has to consider all defeaters for A_1, all of the defeaters for those defeaters, and so on. Following [25], it does so by constructing a dialectical tree, where each node is labelled with an argument, the root node is labelled A_1, and the arcs represent the defeat relation between arguments. Each path through the dialectical tree from root node to a leaf represents an argumentation line, where each argument in such a path defeats its predecessor.

García and Simari impose some extra constraints on what is an acceptable argumentation line. This is because they wish to ensure that their system avoids such things as circular argumentation and that it imposes properties such as concordance between supporting or interfering arguments. For more information on acceptable argumentation lines and their motivation the reader should refer to [25].

Definition 11 If $\Lambda = [\langle \Phi_0, \phi_0 \rangle, \langle \Phi_1, \phi_1 \rangle, \langle \Phi_2, \phi_2 \rangle, \ldots]$ is a sequence of arguments such that each element of the sequence $\langle \Phi_i, \phi_i \rangle$ is a defeater (proper or blocking) of its predecessor $\langle \Phi_{i-1}, \phi_{i-1} \rangle$, then Λ is an **argumentation line**. Λ is an **acceptable argumentation line** iff

1. Λ is a finite sequence,
2. $\Phi_0 \cup \Phi_2 \cup \Phi_4 \cup \ldots \nvdash \bot$ and $\Phi_1 \cup \Phi_3 \cup \Phi_5 \cup \ldots \nvdash \bot$,
3. no argument $\langle \Phi_k, \phi_k \rangle$ appearing in Λ is a subargument of an argument A_j that appears earlier in Λ $(j < k)$,
4. $\forall i$ s.t. $\langle \Phi_i, \phi_i \rangle$ is a blocking defeater for $\langle \Phi_{i-1}, \phi_{i-1} \rangle$, if $\langle \Phi_{i+1}, \phi_{i+1} \rangle$ exists, then $\langle \Phi_{i+1}, \phi_{i+1} \rangle$ is a proper defeater for $\langle \Phi_i, \phi_i \rangle$.

Example 7 Continuing the running example, the following are all examples of argumentation lines, however only Λ_2 is an *acceptable* argumentation line. Λ_1 is not acceptable as it breaks constraints (3) and (4), whereas Λ_3 is not acceptable as it breaks constraint (2).

$$\Lambda_1 = [a_5, a_2, a_1, a_2, a_1, a_2]$$

$$\Lambda_2 = [a_5, a_6, a_7]$$

$$\Lambda_3 = [a_5, a_6, a_8]$$

In order to determine whether the claim of an argument A_0 is *warranted* given a set of beliefs Ψ, the agent must consider every acceptable argumentation line that it can construct from $\mathcal{A}(\Psi)$ which starts with A_0. It does this by constructing a dialectical tree.

Definition 12 Let Ψ be a, possibly inconsistent, belief base and A_0 be an argument such that $A_0 \in \mathcal{A}(\Psi)$. A **dialectical tree** for A_0 constructed from Ψ, denoted $\mathsf{T}(A_0, \Psi)$, is defined as follows.

1. The root of the tree is labelled with A_0.
2. Let N be a node of the tree labelled A_n and let $\Lambda_i = [A_0, \ldots, A_n]$ be the sequence of labels on the path from the root to node N. Let arguments $B_1, B_2, \ldots, B_k$ be all the defeaters for A_n that can be formed from Ψ.
   For each defeater B_j $(1 \leq j \leq k)$, if the argumentation line $\Lambda_i' = [A_0, \ldots, A_n, B_j]$ is an acceptable argumentation line, then the node N has a child N_j that is labelled B_j.
   If there is no defeater for A_n or there is no B_j such that Λ_i' is acceptable, then N is a leaf node.

To ensure that the construction of a dialectical tree is a finite process we must show that it is not possible to construct an infinite argumentation line that meets conditions 2-4 of an acceptable argumentation line (Definition 11). Fortunately, we can show that if condition 3 holds (subarguments of arguments that appear earlier in an argumentation line cannot be repeated) and the arguments in the argumentation line come from a finite set (as they do in the construction of a dialectical tree), then the argumentation line is finite.

Proposition 1 *Let $\Lambda = [A_0, A_1, A_2, \ldots]$ be an argumentation line such that, for any A_i appearing in Λ, $A_i \in \mathcal{A}(\Phi)$ where Φ is a finite set of beliefs. If no argument A_k appearing in Λ is a subargument of an argument A_j that appears earlier in Λ ($j < k$), then Λ is a finite sequence of arguments.*

Proof: *Since Φ is a finite set, it follows from the definition of an argument (Def. 6) that $\mathcal{A}(\Phi)$ is also a finite set. Since the arguments from the sequence Λ come from this finite set, and since we are constrained that no argument in Λ can be repeated, it follows that Λ is a finite sequence.*

From the above proposition we see that, although we may go on constructing a dialectical tree indefinitely and never know if a branch violated condition 1 of an acceptable argumentation line, by meeting condition 3 of an acceptable argumentation line we ensure that this will never be the case.

Note that the root node of a dialectical tree T is denoted $\mathsf{Root}(\mathsf{T})$. Also note, we define two dialectical trees as being equal if and only if, if a sequence of labels appears from the root node to a node in one tree then it also appears as a sequence of labels from the root node to a node in the other. Our definition of dialectical tree equality takes into account the labels of all nodes that appear in the trees (as opposed to, for example, Chesñevar *et al.*'s definition of isomorphic dialectical trees [15], where blocking and proper defeaters are distinguished but no general constraint is made on the arguments that appear at the nodes of the trees). We have taken this approach as we will later show that the dialectical tree constructed by two agents during a warrant inquiry dialogue is equal to the dialectical tree that could be constructed from the union of their beliefs.

Definition 13 The dialectical trees T_1 and T_2 are **equal** to one another iff

1. the root of T_1 is labelled with A_0 iff the root of T_2 is labelled with A_0,
2. if N_1 is a node in T_1 and $[A_0, \ldots, A_n]$ is the sequence of labels on the path from the root of T_1 to N_1, then there is an N_2 s.t. N_2 is a node in T_2 and $[A_0, \ldots, A_n]$ is the sequence of labels on the path from the root of T_2 to N_2,
3. if N_2 is a node in T_2 and $[A_0, \ldots, A_n]$ is the sequence of labels on the path from the root of T_2 to N_2, then there is an N_1 s.t. N_1 is a node in T_1 and $[A_0, \ldots, A_n]$ is the sequence of labels on the path from the root of T_1 to N_1.

Following from [25], in order to determine whether the root of a dialectical tree is undefeated or not, we have to recursively mark each node in the tree as D (defeated) or U (undefeated), dependent on whether it has any undefeated child nodes that are able to defeat it.

Definition 14 Let $\mathsf{T}(A, \Psi)$ be a dialectical tree. The corresponding **marked dialectical tree** of $\mathsf{T}(A, \Psi)$ is obtained by marking every node in $\mathsf{T}(A, \Psi)$ as follows.

1. All leaves in $\mathsf{T}(A, \Psi)$ are marked U.
2. If N is a node of $\mathsf{T}(A, \Psi)$ and N is not a leaf node, then N will be marked U iff every child of N is marked D. The node N will be marked D iff it has at least one child marked U.

Example 8 Following the running example, the corresponding marked dialectical tree of $\mathsf{T}(a_5, \Sigma^1)$ is shown in Figure 1. Note that the arguments a_1 and a_8 do not appear in

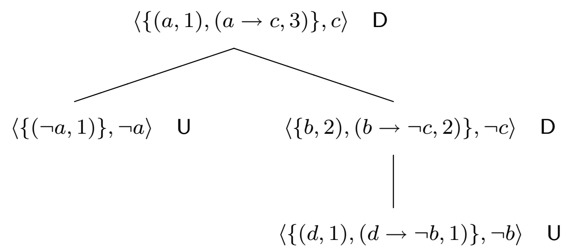


Fig. 1 A marked dialectical tree, $\mathsf{T}(a_5, \Sigma^1)$.

the tree even though they are defeaters of arguments that do appear in the tree, this is because their inclusion would break conditions of an acceptable argumentation line (Definition 11); the inclusion of a_1 would break conditions 3 and 4, the inclusion of a_8 would break condition 2.

The function Status takes an argument A and a set of beliefs Ψ. If the root node of the dialectical tree that has A at its root and is constructed from Ψ is marked U, then Status returns U; if it is marked D, then Status returns D.

Definition 15 The **status** of an argument A given a set of beliefs Ψ is returned by the function $\mathsf{Status} : \mathcal{A}(\mathcal{B}) \times \wp(\mathcal{B}) \mapsto \{\mathsf{U}, \mathsf{D}\}$ such that $\mathsf{Status}(A, \Psi) = \mathsf{U}$ iff $\mathsf{Root}(\mathsf{T}(A, \Psi))$ is marked U in the corresponding marked dialectical tree of $\mathsf{T}(A, \Psi)$, and $\mathsf{Status}(A, \Psi) = \mathsf{D}$ iff $\mathsf{Root}(\mathsf{T}(A, \Psi))$ is marked with D in the corresponding marked dialectical tree of $\mathsf{T}(A, \Psi)$.

The claim of an argument is warranted by the belief base if and only if the status of the root of the associated dialectical tree is U.

Example 9 Following the running example, as the root node of the the tree $\mathsf{T}(a_5, \Sigma^1)$ is marked D (shown in Figure 1), and hence $\mathsf{Status}(a_5, \Sigma^1) = \mathsf{D}$, we see that the argument a_5 is not warranted by the belief base Σ^1.

This warrant procedure makes it possible for a single agent to reason with incomplete, inconsistent and uncertain knowledge. In the following two sections we propose a dialogue system that allows two agents to use this warrant procedure to jointly reason with their beliefs.

4 Representing dialogues

In this section we provide a general framework for representing dialogues. We define how dialogues may be nested one within another, what it means for a dialogue to terminate, and what we mean when we refer to the current dialogue. The framework given in this section is not specific to warrant inquiry and argument inquiry dialogues (the details of which will not come until the following section) and it can be used to represent dialogues of other types (e.g. [13]).

The communicative acts in a dialogue are called *moves*. We assume that there are always exactly two agents (*participants*) taking part in a dialogue, each with its own

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

Table 1 The format for moves used in warrant inquiry and argument inquiry dialogues, where $x \in \mathcal{I}$, $\langle \Phi, \phi\rangle$ is an argument, and either $\theta = wi$ (for warrant inquiry) and $\gamma \in \mathcal{S}^\star$ (i.e. γ is a defeasible fact), or $\theta = ai$ (for argument inquiry) and $\gamma \in \mathcal{R}^\star$ (i.e. γ is a defeasible rule).

identifier taken from the set $\mathcal{I} = \{1, 2\}$. Each participant takes it in turn to make a move to the other participant. For a dialogue involving participants $1, 2 \in \mathcal{I}$, we also refer to participants using the variables x and $\hat{x}$ such that if x is 1 then $\hat{x}$ is 2 and if x is 2 then $\hat{x}$ is 1.

A move in our framework is of the form $\langle Agent, Act, Content\rangle$. *Agent* is the identifier of the agent generating the move, *Act* is the type of move, and the *Content* gives the details of the move. The format for moves used in warrant inquiry and argument inquiry dialogues is shown in Table 1, and the set of all moves meeting the format defined in Table 1 is denoted $\mathcal{M}$. Note that the framework allows for other types of dialogues to be generated and these might require the addition of extra moves (e.g. such as those suggested in [13]). Also, $\mathsf{Sender} : \mathcal{M} \mapsto \mathcal{I}$ is a function such that $\mathsf{Sender}(\langle Agent, Act, Content\rangle) = Agent$.

A dialogue is simply a sequence of moves, each of which is made from one participant to the other. As a dialogue progresses over time, we denote each timepoint by a natural number ($\mathbb{N} = \{1, 2, 3, \ldots\}$). Each move is indexed by the timepoint when the move was made. Exactly one move is made at each timepoint. The dialogue itself is indexed with two timepoints, indexing the first and last moves of the dialogue.

Definition 16 A **dialogue**, denoted D_r^t, is a sequence of moves $[m_r, \ldots, m_t]$ involving two participants in $\mathcal{I} = \{1, 2\}$, where $r, t \in \mathbb{N}$ and $r \leq t$, such that:

1. the first move of the dialogue, m_r, is a move of the form $\langle x, open, dialogue(\theta, \gamma)\rangle$,
2. $\mathsf{Sender}(m_s) \in \mathcal{I}$ $(r \leq s \leq t)$,
3. $\mathsf{Sender}(m_s) \neq \mathsf{Sender}(m_{s+1})$ $(r \leq s < t)$.

The **type** of the dialogue D_r^t is returned by $\mathsf{Type}(D_r^t)$ such that $\mathsf{Type}(D_r^t) = \theta$ (i.e. the type of the dialogue is determined by the content of the first move made). The **topic** of the dialogue D_r^t is returned by $\mathsf{Topic}(D_r^t)$ such that $\mathsf{Topic}(D_r^t) = \gamma$ (i.e. the topic of the dialogue is determined by the content of the first move made). The set of all dialogues is denoted $\mathcal{D}$.

The first move of a dialogue D_r^t must always be an open move (condition 1 of the previous definition), every move of the dialogue must be made to a participant of the dialogue (condition 2), and the agents take it in turns to make moves (condition 3). The *type* and the *topic* of a dialogue are determined by the content of the first move made; if the first move made in a dialogue is $\langle x, open, dialogue(\theta, \gamma)\rangle$, then the type of the dialogue is θ and the topic of the dialogue is γ. In this article, we consider two different types of dialogue (i.e. two different values for θ): wi (for *warrant inquiry*) and ai (for *argument inquiry*). If a dialogue is a warrant inquiry dialogue, then its topic must be a defeasible fact; if a dialogue is an argument inquiry dialogue, then its topic must be a defeasible rule; these are requirements of the format of open moves defined in Table 1. Although we consider only warrant inquiry and argument inquiry dialogues here, our definition of a dialogue is general so as to allow dialogues of other types to be considered within our framework.

We now define some terminology that allows us to talk about the relationship between two dialogues.

Definition 17 Let D_r^t and $D_{r_1}^{t_1}$ be two dialogues. $D_{r_1}^{t_1}$ is a **sub-dialogue** of D_r^t iff $D_{r_1}^{t_1}$ is a sub-sequence of D_r^t $(r < r_1 \leq t_1 \leq t)$. D_r^t is a **top-level** dialogue iff $r = 1$; the set of all top-level dialogues is denoted $\mathcal{D}_{top}$. D_1^t is a **top-dialogue of** D_r^t iff either the sequence D_1^t is the same as the sequence D_r^t or D_r^t is a sub-dialogue of D_1^t. If D_r^t is a sequence of n moves, $D_r^{t_2}$ **extends** D_r^t iff the first n moves of $D_r^{t_2}$ are the sequence D_r^t.

In order to terminate a dialogue, two close moves must appear next to each other in the sequence (called a *matched-close*); this means that each participating agent must agree to the termination of the dialogue. A close move is used to indicate that an agent wishes to terminate the dialogue; it may be the case, however, that the other agent still has something it wishes to say, which may in turn cause the original agent to change its mind about wishing to terminate the dialogue.

Definition 18 Let D_r^t be a dialogue of type $\theta \in \{wi, ai\}$ with participants $\mathcal{I} = \{1, 2\}$ such that $\mathsf{Topic}(D_r^t) = \gamma$. We say that m_s $(r < s \leq t)$ is a **matched-close for** D_r^t iff $m_{s-1} = \langle x, close, dialogue(\theta, \gamma)\rangle$ and $m_s = \langle \hat{x}, close, dialogue(\theta, \gamma)\rangle$.

So a matched-close will terminate a dialogue D_r^t but only if D_r^t has not already terminated and any sub-dialogues that are embedded within D_r^t have already terminated; this notion will be needed later on to define well-formed inquiry dialogues.

Definition 19 Let D_r^t be a dialogue. D_r^t **terminates at** t iff the following conditions hold:

1. m_t is a matched-close for D_r^t,
2. $\nexists D_r^{t_1}$ s.t. $D_r^{t_1}$ terminates at t_1 and D_r^t extends $D_r^{t_1}$,
3. $\forall D_{r_1}^{t_1}$ if $D_{r_1}^{t_1}$ is a sub-dialogue of D_r^t,
   then $\exists D_{r_1}^{t_2}$ s.t. $D_{r_1}^{t_2}$ terminates at t_2
   and either $D_{r_1}^{t_2}$ extends $D_{r_1}^{t_1}$ or $D_{r_1}^{t_1}$ extends $D_{r_1}^{t_2}$,
   and $D_{r_1}^{t_2}$ is a sub-dialogue of D_r^t.

As we are often dealing with multiple nested dialogues it is often useful to refer to the current dialogue, which is the innermost dialogue that has not yet terminated. As dialogues of one type may be nested within dialogues of another type, an agent must refer to the current dialogue in order to know which protocol to follow.

Definition 20 Let D_r^t be a dialogue. The **current dialogue** is given by $\mathsf{Current}(D_r^t)$ such that $\mathsf{Current}(D_r^t) = D_{r_1}^t$ $(1 \leq r \leq r_1 \leq t)$ where the following conditions hold:

1. $m_{r_1} = \langle x, open, dialogue(\theta, \gamma)\rangle$ for some $x \in \mathcal{I}$, some $\gamma \in \mathcal{B}^\star$ and some $\theta \in \{wi, ai\}$,
2. $\forall D_{r_2}^{t_1}$ if $D_{r_2}^{t_1}$ is a sub-dialogue of $D_{r_1}^t$,
   then $\exists D_{r_2}^{t_2}$ s.t. either $D_{r_2}^{t_2}$ extends $D_{r_2}^{t_1}$ or
   $D_{r_2}^{t_1}$ extends $D_{r_2}^{t_2}$,
   and $D_{r_2}^{t_2}$ is a sub-dialogue of $D_{r_1}^t$
   and $D_{r_2}^{t_2}$ terminates at t_2,

3. $\nexists D^{t_3}_{r_1}$ s.t. $D^t_{r_1}$ extends $D^{t_3}_{r_1}$ and $D^{t_3}_{r_1}$ terminates at t_3.

If the above conditions do not hold then $\mathsf{Current}(D^t_r) = null$.
The **topic of the current dialogue** is returned by the function $\mathsf{cTopic}(D^t_r)$ such that $\mathsf{cTopic}(D^t_r) = \mathsf{Topic}(\mathsf{Current}(D^t_r))$. The **type of the current dialogue** is returned by the function $\mathsf{cType}(D^t_r)$ such that $\mathsf{cType}(D^t_r) = \mathsf{Type}(\mathsf{Current}(D^t_r))$.

We now give a schematic example of nested dialogues.

Example 10 An example of nested dialogues is shown in Figure 2. In this example:

$\mathsf{Current}(D^t_1) = D^t_1$	$\mathsf{Current}(D^{t-1}_1) = D^{t-1}_i$	$\mathsf{Current}(D^k_1) = D^k_i$
$\mathsf{Current}(D^{k-1}_1) = D^{k-1}_j$	$\mathsf{Current}(D^t_i) = null$	$\mathsf{Current}(D^{t-1}_i) = D^{t-1}_i$
$\mathsf{Current}(D^k_i) = D^k_i$	$\mathsf{Current}(D^{k-1}_i) = D^{k-1}_j$	$\mathsf{Current}(D^k_j) = null$
	$\mathsf{Current}(D^{k-1}_j) = D^{k-1}_j$	

We have now defined our general framework for representing dialogues, in the following section we give the details needed to generate argument inquiry and warrant inquiry dialogues.

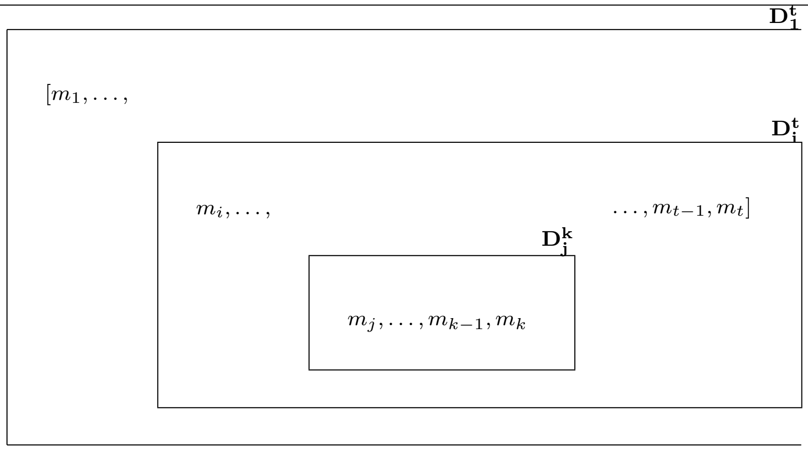
5 Generating dialogues

In this section we give the details specific to argument inquiry and warrant inquiry dialogues that, along with the general framework given in the previous section, comprise our inquiry dialogue system. In Sections 5.1 and 5.2 we give the protocols needed to model legal argument inquiry and warrant inquiry dialogues, we define what a well-formed argument inquiry dialogue is and what a well-formed warrant inquiry dialogue is (a dialogue that terminates and whose moves are legal according to the relevant protocol), and we define what the outcomes of the two dialogue types are. In Section 5.3 we give the details of a strategy that can be used to generate legal argument inquiry and warrant inquiry dialogues (i.e. that allows an agent to select exactly one of the legal moves to make at any point in the dialogue).

We adopt the common approach of associating a *commitment store* with each agent participating in a dialogue (e.g. [34, 39]). A commitment store is a set of beliefs that the agent is publicly committed to as the current point of the dialogue (i.e. that they have asserted). As a commitment store consists of things that the agent has already publicly declared, its contents are visible to the other agent participating in the dialogue. For this reason, when constructing an argument, an agent may make use of not only its own beliefs but also those from the other agent's commitment store.

Definition 21 A **commitment store** is a set of beliefs denoted CS^t_x (i.e. $CS^t_x \subseteq \mathcal{B}$), where $x \in \mathcal{I}$ is an agent and $t \in \mathbb{N}$ is a timepoint.

When an agent enters into a top-level dialogue of any kind a commitment store is created and persists until that dialogue has terminated (i.e. this same commitment store is used for any sub-dialogues of the top-level dialogue). If an agent makes a move asserting an argument, every element of the support is added to the agent's commitment store. This is the only time the commitment store is updated.



$$1 < i < j < k < t-1$$
$$D_1^t = [m_1, \ldots, m_i, \ldots, m_j, \ldots, m_k, m_{k+1}, \ldots, m_{t-1}, m_t]$$
$$m_1 = \langle P_1, open, dialogue(\theta_1, \phi_1)\rangle \qquad m_i = \langle P_i, open, dialogue(\theta_i, \phi_i)\rangle$$
$$m_j = \langle P_j, open, dialogue(\theta_j, \phi_j)\rangle \qquad m_{k-1} = \langle P_{k-1}, close, dialogue(\theta_j, \phi_j)\rangle$$
$$m_k = \langle P_k, close, dialogue(\theta_j, \phi_j)\rangle \qquad m_{t-1} = \langle P_{t-1}, close, dialogue(\theta_i, \phi_i)\rangle$$
$$m_t = \langle P_t, close, dialogue(\theta_i, \phi_i)\rangle$$

Fig. 2 Nested dialogues. D_1^t is a top level dialogue that has not yet terminated. D_i^t is a sub-dialogue of D_1^t that terminates at t. D_j^k is a sub-dialogue of both D_1^t and D_i^t, that terminates at k. D_1^t is a top-dialogue of D_1^t. D_1^k is a top-dialogue of D_j^k. D_1^t is a top-dialogue of D_i^t. D_1^k is a top-dialogue of D_1^k.

Definition 22 Commitment store update. Let the current dialogue be D_r^t with participants $\mathcal{I} = \{1, 2\}$.

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

The only move used in argument inquiry or warrant inquiry dialogues that affects an agent's commitment store is the assert move, which causes the support of the argument being asserted to be added to the commitment store; hence the commitment store of an agent participating in an argument inquiry or warrant inquiry dialogue grows monotonically over time. If we were to define more moves in order to allow our system to generate other dialogue types, it would be necessary to define the effect of those moves on an agent's commitment store. For example, a retract move (that causes a belief to be removed from the commitment store) may be necessary for a negotiation dialogue, in which case the commitment store of an agent participating in a negotiation dialogue will not necessarily grow monotonically over time.

In the following subsection we give the details needed to allow us to model well-formed argument inquiry dialogues.

5.1 Modelling argument inquiry dialogues

An argument inquiry dialogue is initiated when an agent wants to construct an argument for a certain claim, let us say ϕ, that it cannot construct alone. If the agent knows of a domain belief whose consequent is that claim, let us say $(\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \phi, L)$, then the agent will open an argument inquiry dialogue with $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \phi$ as its topic. If, between them, the two participating agents could provide arguments for each of the elements α_i $(1 \leq i \leq n)$ in the antecedent of the topic, then it would be possible for an argument for ϕ to be constructed. We define the *query store* as the set of literals that could help construct an argument for the consequent of the topic of an argument inquiry dialogue: when an argument inquiry dialogue with topic $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta$ is opened, a query store associated with that dialogue is created whose contents are $\{\alpha_1, \ldots, \alpha_n, \beta\}$. Throughout the dialogue the participating agents will both try to provide arguments for the literals in the query store. This may lead them to open further nested argument inquiry dialogues that have as a topic a rule whose consequent is a literal in the current query store.

Definition 23 For a dialogue D_r^t with participants $\mathcal{I} = \{1, 2\}$, a **query store**, denoted QS_r, is a finite set of literals such that

$$QS_r = \begin{cases} \{\alpha_1, \ldots, \alpha_n, \beta\} & \text{iff } m_r = \langle x, open, dialogue(ai, \alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \beta)\rangle \\ \emptyset & \text{otherwise.} \end{cases}$$

The **query store of the current dialogue** is given by $\mathsf{cQS}(D_r^t)$ such that $\mathsf{cQS}(D_r^t) = QS_{r_1}$ iff $\mathsf{Current}(D_r^t) = D_{r_1}^t$.

A query store is fixed and is determined by the open move that opens the associated argument inquiry dialogue. If the current dialogue is an argument inquiry dialogue, then the agents will consult the query store of the current dialogue in order to determine what arguments it might be helpful to try and construct (i.e. those whose claim is a member of the current query store).

A protocol is a function that returns the set of moves that are legal for an agent to make at a particular point in a particular type of dialogue. Here we give the specific protocol for argument inquiry dialogues. It takes the top-level dialogue that the agents are participating in and returns the set of legal moves that the agent may make.

Definition 24 The **argument inquiry protocol** is a function $\Pi_{\mathsf{ai}} : \mathcal{D}_{top} \mapsto \wp(\mathcal{M})$. If D_1^t is a top-level dialogue with participants $\mathcal{I} = \{1, 2\}$ such that $\mathsf{Sender}(m_t) = \hat{x}$, $1 \leq t$ and $\mathsf{cTopic}(D_1^t) = \gamma$, then $\Pi_{\mathsf{ai}}(D_1^t)$ is

$$\Pi_{\mathsf{ai}}^{\mathsf{assert}}(D_1^t) \cup \Pi_{\mathsf{ai}}^{\mathsf{open}}(D_1^t) \cup \{\langle x, close, dialogue(ai, \gamma)\rangle\}$$

where

$\Pi_{\mathsf{ai}}^{\mathsf{assert}}(D_1^t) = \{\langle x, assert, \langle \Phi, \phi\rangle\rangle |$
(1) $\phi \in \mathsf{cQS}(D_1^t)$,
(2) $\nexists t'$ s.t. $1 < t' \leq t$ and $m_{t'} = \langle x', assert, \langle \Phi, \phi\rangle\rangle$ and $x' \in \mathcal{I}\}$

$\Pi_{\mathsf{ai}}^{\mathsf{open}}(D_1^t) = \{\langle x, open, dialogue(ai, \beta_1 \wedge \ldots \wedge \beta_n \rightarrow \alpha)\rangle |$
(1) $\alpha \in \mathsf{cQS}(D_1^t)$,
(2) $\nexists t'$ s.t. $1 < t' \leq t$ and $m_{t'} = \langle x', open, \beta_1 \wedge \ldots \wedge \beta_n \rightarrow \alpha\rangle$ and $x' \in \mathcal{I}\}$

The argument inquiry protocol tells us that an agent may always legally make a close move, indicating that it wishes to terminate the dialogue (recall, however, that the dialogue will only terminate when both agents indicate they wish to terminate the dialogue, leading to a matched-close). An agent may legally assert an argument that has not previously been asserted, as long as its claim is in the current query store (and so may help the agents in finding an argument for the consequent of the topic of the dialogue). An agent may legally open a new, embedded argument inquiry dialogue providing an argument inquiry dialogue with the same topic has not previously been opened and the of the topic of the new argument inquiry dialogue is a defeasible rule that has a member of the current query store as its consequent (and so any arguments successfully found in this new argument inquiry dialogue may help the agents in finding an argument for the consequent of the topic of the current dialogue). It is straightforward to check conformance with the protocol as it only refers to public elements of the dialogue.

We are now able to define a well-formed argument inquiry dialogue. This is a dialogue that starts with a move opening an argument inquiry dialogue (condition 1 of the following definition), that has a continuation which terminates (condition 2) and whose moves conform to the argument inquiry protocol (condition 3).

Definition 25 Let D_r^t be a dialogue with participants $\mathcal{I} = \{1, 2\}$. D_r^t is a **well-formed argument inquiry dialogue** iff the following conditions hold:

1. $m_r = \langle x, open, dialogue(ai, \gamma) \rangle$ where $x \in \mathcal{I}$ and $\gamma \in \mathcal{R}^\star$ (i.e. γ is a defeasible rule),
2. $\exists t'$ s.t. $t \leq t'$, $D_r^{t'}$ extends D_r^t, and $D_r^{t'}$ terminates at t',
3. $\forall s$ s.t. $r \leq s < t$ and D_r^t extends D_r^s,
   if D_1^t is a top-dialogue of D_r^t and
   D_1^s is a top-dialogue of D_r^s and
   D_1^t extends D_1^s and
   $\mathsf{Sender}(m_s) = x'$ (where $x' \in \mathcal{I}$),
   then $m_{s+1} \in \Pi_{\mathsf{ai}}(D_1^s, \hat{x}')$ (where $\hat{x}' \in \mathcal{I}, \hat{x}' \neq x'$).

The set of all well-formed argument inquiry dialogues is denoted $\mathcal{D}_{ai}$.

We define the outcome of an argument inquiry dialogue as the set of all arguments that can be constructed from the union of the commitment stores and whose claims are in the query store.

Definition 26 The **argument inquiry outcome** of a dialogue is given by a function $\mathsf{Outcome}_{\mathsf{ai}} : \mathcal{D}_{ai} \mapsto \wp(\mathcal{A}(\mathcal{B})) \cup \{\}$. If D_r^t is a well-formed argument inquiry dialogue with participants $\mathcal{I} = \{1, 2\}$, then
$\mathsf{Outcome}_{\mathsf{ai}}(D_r^t) = \{\langle \Phi, \phi \rangle \in \mathcal{A}(CS_1^t \cup CS_2^t) \mid \phi \in QS_r\}$

We are now able to model well-formed argument inquiry dialogues and determine their outcome. In the following subsection we give the details that we need to model well-formed warrant inquiry dialogues.

5.2 Modelling warrant inquiry dialogues

The goal of a warrant inquiry dialogue is to jointly construct a dialectical tree whose root is an argument for the topic of the dialogue; the topic of the dialogue is warranted if and only if the root of the dialectical tree is undefeated. The participants take it in turn to exchange arguments in order to construct this tree. The argument for the topic of the dialogue that is at the root of the dialectical tree is called the *root argument*. As it may be the case that more than one argument for the topic are asserted during the dialogue, the root argument is the first argument for the topic that gets asserted. If the root argument is *null* then this means that no argument for the topic has been asserted yet.

Definition 27 The function $\mathsf{RootArg} : \mathcal{D} \mapsto \mathcal{A}(\mathcal{B}) \cup \{null\}$ returns the **root argument** of a warrant inquiry dialogue. Let D_r^t be a warrant inquiry dialogue with participants $\mathcal{I} = \{1, 2\}$.

$$\mathsf{RootArg}(D_r^t) = \begin{cases} \langle \Gamma, \gamma \rangle & \text{if } \exists s \text{ s.t. } r < s \leq t \text{ and} \\ & \quad m_s = \langle x, assert, \langle \Gamma, \gamma \rangle \rangle \text{ and } \mathsf{Topic}(D_r^t) = \gamma \text{ and } x \in \mathcal{I} \text{ and} \\ & \quad \nexists s' \text{ such that } r < s' < s \text{ and} \\ & \qquad \exists \Gamma' \text{ s.t. } m_{s'} = \langle x', assert, \langle \Gamma', \gamma \rangle \rangle \\ & \qquad \text{and } x' \in \mathcal{I}, \\ null & \text{otherwise.} \end{cases}$$

As a warrant inquiry dialogue progresses, a dialectical tree is constructed from the union of the commitment stores whose root node is labelled with the root argument of the dialogue. This tree may, at the end of the dialogue, act as a warrant for the topic of the dialogue if and only if the status of the root node is U.

We now give the protocol for modelling legal warrant inquiry dialogues. As with the argument inquiry dialogue, this only refers to public elements of the dialogue, hence it is straightforward to check an agent's conformance to the protocol.

Definition 28 A **warrant inquiry protocol** is a function $\Pi_{\mathsf{wi}} : \mathcal{D}_{top} \mapsto \wp(\mathcal{M})$. If D_1^t is a top-level dialogue with participants $x, \hat{x} \in \mathcal{I}$ such that $\mathsf{Sender}(m_t) = \hat{x}$, $1 \leq t$ and $\mathsf{Topic}(D_1^t) = \gamma$, then $\Pi_{\mathsf{wi}}(D_1^t)$ is

$$\Pi_{\mathsf{wi}}^{\mathsf{assert}}(D_1^t) \cup \Pi_{\mathsf{wi}}^{\mathsf{open}}(D_1^t) \cup \{\langle x, close, dialogue(wi, \gamma) \rangle\}$$

where

$\Pi_{\mathsf{wi}}^{\mathsf{assert}}(D_1^t) = \{\langle x, assert, \langle \Phi, \phi \rangle \rangle |$
(1) either $\mathsf{RootArg}(D_1^t) = null$ and $\phi = \gamma$, or
$\mathsf{T}(\mathsf{RootArg}(D_1^t), CS_1^t \cup CS_2^t \cup \Phi) \neq \mathsf{T}(\mathsf{RootArg}(D_1^t), CS_1^t \cup CS_2^t)$
(2) $\nexists t'$ s.t. $1 < t' \leq t$ and $m_{t'} = \langle x', assert, \langle \Phi, \phi \rangle \rangle$ and $x' \in \mathcal{I}\}$

$\Pi_{\mathsf{wi}}^{\mathsf{open}}(D_1^t) = \{\langle x, open, dialogue(ai, \beta_1 \wedge \ldots \wedge \beta_n \rightarrow \alpha) \rangle |$
(1) either $\mathsf{RootArg}(D_1^t) = null$ and $\alpha = \gamma$,
or $\neg\alpha \in \mathsf{DefDerivations}(CS_1^t \cup CS_2^t)$
(2) $\nexists t'$ s.t. $1 < t' \leq t$ and $m_{t'} = \langle x', open, dialogue(ai, \beta_1 \wedge \ldots \wedge \beta_n \rightarrow \alpha) \rangle$
and $x' \in \mathcal{I}\}$

As with the argument inquiry dialogue, an agent can always legally make a move closing the current dialogue. An agent can legally assert an argument that has not previously been asserted as long as that argument is either the first to be asserted for the topic of the dialogue, or if asserting the argument causes the dialectical tree being constructed to change in some way (and so has the potential to affect the dialogue outcome). An agent can legally open an embedded *argument inquiry* dialogue with a defeasible rule as its topic as long as no such embedded argument inquiry dialogue has previously been opened and either no argument for the topic of the dialogue has yet been asserted and the consequent of the defeasible rule is the topic of the dialogue, or it is possible to defeasibly derive the negation of the consequent from the union of the commitment stores (and so any arguments successfully found may have some bearing on the outcome of the dialogue).

Note that, as the only type of open move that it is legal to make within a warrant inquiry dialogue is one that opens an embedded argument inquiry dialogue, the only type of dialogue that can be embedded within a warrant inquiry dialogue is an argument inquiry dialogue. Recall that argument inquiry dialogues may themselves have argument inquiry dialogues embedded within them, and so it is possible to have multiple nested argument inquiry dialogues embedded within a warrant inquiry dialogue. However, it is not possible to nest warrant inquiry dialogues within other warrant inquiry dialogues or within argument inquiry dialogues.

We now define a well-formed warrant inquiry dialogue. This is a dialogue that starts with a move opening an warrant inquiry dialogue (condition 1 of the following definition), that has a continuation that terminates (condition 2) and whose moves conform to the warrant inquiry protocol (condition 3).

Definition 29 Let D_r^t be a dialogue with participants $\mathcal{I} = \{1, 2\}$. D_r^t is a **well-formed warrant inquiry dialogue** iff the following conditions hold:

1. $m_r = \langle x, open, dialogue(wi, \gamma)\rangle$ where $x \in \mathcal{I}$ and $\gamma \in \mathcal{S}^\star$ (i.e. γ is a defeasible fact),
2. $\exists t'$ s.t. $t \leq t'$, $D_r^{t'}$ extends D_r^t, and $D_r^{t'}$ terminates at t',
3. $\forall s$ s.t. $r \leq s < t$ and D_r^t extends D_r^s,
   if D_1^t is a top-dialogue of D_r^t and
   D_1^s is a top-dialogue of D_r^s and
   D_1^t extends D_1^s and
   $\mathsf{Sender}(m_s) = x'$ (where $x' \in \mathcal{I}$),
   then $m_{s+1} \in \Pi_{\mathsf{wi}}(D_1^s, \hat{x}')$ (where $\hat{x}' \in \mathcal{I}, \hat{x}' \neq x'$).

The set of all well-formed warrant inquiry dialogues is denoted $\mathcal{D}_{wi}$.

Note that hereinafter we will use the term well-formed dialogue to refer to either a well-formed argument inquiry dialogue or a well-formed warrant inquiry dialogue.

The outcome of a warrant inquiry dialogue is determined by the dialectical tree that is constructed from the union of the commitment stores. If the root argument is undefeated in the dialectical tree then a warranted argument for the topic of the dialogue has successfully been found and the outcome of the dialogue is the root argument, otherwise the outcome of the dialogue is null.

Definition 30 The **warrant inquiry outcome** of a dialogue is a function $\mathsf{Outcome}_{\mathsf{wi}}$ such that $\mathsf{Outcome}_{\mathsf{wi}} : \mathcal{D}_{wi} \mapsto \mathcal{A}(\mathcal{B}) \cup \{\}$. Let D_r^t be a well formed warrant inquiry dialogue with participants $\mathcal{I} = \{1, 2\}$.

$$\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \begin{cases} \mathsf{RootArg}(D_r^t) & \text{if } \mathsf{Status}(\mathsf{RootArg}(D_r^t), CS_1^t \cup CS_2^t) \\ & = \mathsf{U}, \text{ else} \\ \{\} & \text{if } \mathsf{Status}(\mathsf{RootArg}(D_r^t), CS_1^t \cup CS_2^t) \\ & = \mathsf{D} \text{ or } \mathsf{RootArg}(D_r^t) = null. \end{cases}$$

We have now given protocols for both the argument inquiry and warrant inquiry dialogue. In the following subsection we provide a strategy that allows an agent to select exactly one of the legal moves returned by the relevant protocol.

5.3 Generating dialogues

We will shortly give the strategy function that allows an agent to select exactly one legal move to make at any point in either an argument inquiry or a warrant inquiry dialogue. It is this function that sets our system apart from many of the comparable existing systems, as it allows the actual generation of dialogues. Most dialogue systems only go so far as to provide something equivalent to our protocol function (e.g. [40, 42]). Such systems are intended for modelling legal dialogues, whilst our system allows generation of dialogues by providing a specific strategy function that allows agents to select exactly one legal move to make. A strategy function takes the top-level dialogue that an agent is participating in and returns exactly one move to be made.

A strategy is personal to an agent, as the move that it returns depends on the agent's private beliefs. The exhaustive strategy that we give here states that if there are any legal moves that assert an argument which can be constructed by the agent, then a single one of these moves is selected (according to a selection function that we define shortly, denoted $\mathsf{Pick}_{\mathsf{a}}$); else if there are any legal open moves with a defeasible rule from the agent's beliefs as their content, then a single one of these moves is selected (according to a selection function that we define shortly, denoted $\mathsf{Pick}_{\mathsf{o}}$); else a close move is made.

In order to select a single open move from a set of open moves, we assign a unique number to each move content and carry out a comparison of these numbers. Let us assume that $\mathcal{B}^\star$ is composed of a finite number Z of atoms. Let us also assume that there is a registration function μ over these atoms: so, for a literal α, $\mu(\alpha)$ returns a unique single digit number base Z (this number is only like an id number and can be arbitrarily assigned). For a rule $\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \alpha_{n+1}$, $\mu(\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \alpha_{n+1})$ is an $n+1$ digit number of the form $\mu(\alpha_1) \ldots \mu(\alpha_n)\mu(\alpha_{n+1})$. This gives a unique base Z number for each formula in $\mathcal{B}^\star$ and allows an agent to select a single open move using the natural ordering relation $<$ over base Z numbers.

Definition 31 Let $\Xi = \{\langle x, open, dialogue(\theta_1, \phi_1)\rangle, \ldots, \langle x, open, dialogue(\theta_k, \phi_k)\rangle\}$ be a set of legal open moves that could be made by agent x. The function $\mathsf{Pick}_{\mathsf{o}}$ returns the **selected open move** to make. $\mathsf{Pick}_{\mathsf{o}}(\Xi) = \langle x, open, dialogue(\theta_i, \phi_i)\rangle$ $(1 \leq i \leq k)$ such that for all j $(1 \leq j \leq k)$ if $i \neq j$, then $\mu(\phi_i) < \mu(\phi_j)$.

If the set Ξ taken by the function $\mathsf{Pick}_{\mathsf{o}}$ is not the empty set, then (as μ assigns a unique number to the content of each open move) $\mathsf{Pick}_{\mathsf{o}}$ deterministically returns a single open move.

Example 11 Let us assume that
$\Xi = \{\langle 1, open, dialogue(ai, a \wedge b \rightarrow c)\rangle, \langle 1, open, dialogue(ai, \neg a \rightarrow d)\rangle\}$
and that μ arbitrarily assigns a single digit base 5 number to the atoms that appear in Ξ as follows: $\mu(a) = 1$, $\mu(\neg a) = 2$, $\mu(b) = 3$, $\mu(c) = 4$, $\mu(d) = 5$.
This gives us the following unique base 5 numbers for the defeasible rules that appear in Ξ: $\mu(a \wedge b \rightarrow c) = 134$, $\mu(\neg a \rightarrow d) = 25$.
As $\mu(\neg a \rightarrow d) < \mu(a \wedge b \rightarrow c)$, we get $\mathsf{Pick_o}(\Xi) = \langle 1, open, dialogue(ai, \neg a \rightarrow d)\rangle$.

In order to select a single assert move from a set of assert moves, we similarly assign a unique tuple of numbers to each move content and carry out a comparison of these tuples. We assign a tuple of numbers to each argument in $\mathcal{A}(\mathcal{B})$ using a registration function λ together with μ. For an argument that takes the form

$$\langle\{(\phi_1, L_1), \ldots, (\phi_n, L_n)\}, \phi_{n+1}\rangle$$

we get

$$\lambda(\langle\{(\phi_1, L_1), \ldots, (\phi_n, L_n)\}, \phi_{n+1}\rangle) = \langle d_1, \ldots, d_n, d_{n+1}\rangle$$

where

$$d_1 < \ldots < d_n < d_{n+1}$$

and

$$\langle d_1, \ldots, d_n, d_{n+1}\rangle \text{ is a permutation of } \langle \mu(\phi_1), \ldots, \mu(\phi_n), \mu(\phi_{n+1})\rangle$$

(where μ is the registration function for $\mathcal{B}$). The function λ returns a unique tuple of base Z numbers for each argument. We use a standard lexicographical comparison, denoted $\prec_{lex}$, of these tuples of numbers to select a move to make (i.e. the one whose content is the maximum element in the lexicographical ordering).

Definition 32 Let $\Xi = \{\langle x, assert, \langle \Phi_1, \phi_1\rangle\rangle, \ldots, \langle x, assert, \langle \Phi_k, \phi_k\rangle\rangle\}$ be a set of legal assert moves that could be made by agent x. The function $\mathsf{Pick_a}$ returns the **chosen assert move** to make. $\mathsf{Pick_a}(\Xi) = \langle x, assert, \langle \Phi_i, \phi_i\rangle\rangle$ $(1 \leq i \leq k)$ such that for all j $(1 \leq j \leq k)$ if $i \neq j$, then $\lambda(\langle \Phi_i, \phi_i\rangle) \prec_{lex} \lambda(\langle \Phi_j, \phi_j\rangle)$.

If the set Ξ taken by the function $\mathsf{Pick_a}$ is not the empty set, then (as λ assigns a unique tuple to the content of each assert move) $\mathsf{Pick_a}$ deterministically returns a single assert move.

Example 12 Let us assume that
$\Xi = \{\langle 1, assert, \langle\{(a,1), (b,1), (a \wedge b \rightarrow c, 1)\}, c\rangle\rangle, \langle 1, assert, \langle\{(\neg a, 1), (\neg a \rightarrow d, 1)\}, d\rangle\rangle\}$
and that μ arbitrarily assigns a single digit base 5 number to the atoms that appear in Ξ as follows: $\mu(a) = 1$, $\mu(\neg a) = 2$, $\mu(b) = 3$, $\mu(c) = 4$, $\mu(d) = 5$.
This gives us the following unique tuples of base 5 numbers for the arguments that appear in Ξ: $\lambda(\langle\{(a,1), (b,1), (a \wedge b \rightarrow c, 1)\}, c\rangle) = \langle 1, 3, 134\rangle$, $\lambda(\langle\{(\neg a, 1), (\neg a \rightarrow d, 1)\}, d\rangle) = \langle 2, 25\rangle$.
As $\lambda(\langle\{(a,1), (b,1), (a \wedge b \rightarrow c, 1)\}, c\rangle) \prec_{lex} \lambda(\langle\{(\neg a, 1), (\neg a \rightarrow d, 1)\}, d\rangle)$, we get $\mathsf{Pick_a}(\Xi) = \langle 1, assert, \langle\{(a,1), (b,1), (a \wedge b \rightarrow c, 1)\}, c\rangle\rangle$.

We now define the exhaustive strategy, so called as it ensures that all moves which might have a bearing on the outcome of the dialogue will get made. This strategy defines a subset of the legal assert moves and a subset of the legal open moves. If the subset of legal assert moves is not empty, then $\mathsf{Pick_a}$ is used to select one of these moves to make; else if the subset of legal open moves is not empty, then $\mathsf{Pick_o}$ is used to select one of these moves to make; else a close move is made.

$$\Omega_x(D_1^t) = \begin{cases} \mathsf{Pick_a}(\mathsf{Asserts}_x(D_1^t)) & \text{iff } \mathsf{Asserts}_x(D_1^t) \neq \emptyset \\ \mathsf{Pick_o}(\mathsf{Opens}_x(D_1^t)) & \text{iff } \mathsf{Asserts}_x(D_1^t) = \emptyset \text{ and } \mathsf{Opens}_x(D_1^t) \neq \emptyset \\ \langle x, close, dialogue(\theta, \gamma)\rangle & \text{iff } \mathsf{Asserts}_x(D_1^t) = \emptyset \text{ and } \mathsf{Opens}_x(D_1^t) = \emptyset \end{cases}$$

where $\theta = \mathsf{cType}(D_1^t)$, $\gamma = \mathsf{cTopic}(D_1^t)$ and the choices for the moves are given by the following subsidiary functions.

$$\mathsf{Asserts}_x(D_1^t) = \{\langle x, assert, \langle \Phi, \phi\rangle\rangle \in \Pi_\theta(D_1^t) \mid \langle \Phi, \phi\rangle \in \mathcal{A}(\Sigma^x \cup CS_{\hat{x}}^t)\}$$
$$\mathsf{Opens}_x(D_1^t) = \{\langle x, open, dialogue(ai, \psi)\rangle \in \Pi_\theta(D_1^t) \mid (\psi, L) \in \Sigma^x\}$$

Fig. 3 The exhaustive strategy function uniquely selects a legal move to be made in either a warrant inquiry or an argument inquiry dialogue.

Definition 33 The **exhaustive strategy** for an agent x participating in a well-formed top-level dialogue D_1^t is a function $\Omega_x : \mathcal{D}_{top} \mapsto \mathcal{M}$ given in Figure 3.

Note the restrictions on the sets of legal moves from which an agent can pick a next move. As we are considering a cooperative domain, an agent will only assert an argument that it can construct from the union of its beliefs and the other agent's commitment store (and so will not make arguments up or deliberately deceive). Agents are restricted to only opening a new argument inquiry dialogue with topic ϕ if they have a belief (ϕ, L). This prevents an agent from opening a nested sub-dialogue unless it at least knows of a rule that might help construct the desired argument.

As the strategy we define here is exhaustive, in the sense that it ensures all legal moves get made, our simple selection functions $\mathsf{Pick_a}$ and $\mathsf{Pick_o}$ (that depend only on an arbitrary registration function) are sufficient. It would be interesting to consider more sophisticated selection functions that might, for example, aim to minimise the length of the dialogues produced whilst still arriving at the same dialogue outcome.

We now define a well-formed exhaustive dialogue. This is either a well-formed argument inquiry dialogue or a well-formed warrant inquiry dialogue that is generated by two agents who both follow the exhaustive strategy at all times.

Definition 34 Let D_r^t be a well-formed dialogue with participants $\mathcal{I} = \{1, 2\}$. D_r^t is a *well-formed exhaustive dialogue* if the following condition holds.

1. $\forall s$ s.t. $r \leq s < t$ and D_r^t extends D_r^s,
   if D_1^t is a top-dialogue of D_r^t and
   D_1^s is a top-dialogue of D_r^s and
   D_1^t extends D_1^s and
   $\mathsf{Sender}(m_s) = x$ (where $x \in \mathcal{I}$),
   then $m_{s+1} = \Omega_{\hat{x}}(D_1^s)$.

We have now provided a strategy for use with either the argument inquiry protocol or the warrant inquiry protocol, allowing agents to generate both well-formed warrant inquiry dialogues and well-formed argument inquiry dialogues. In the following section we give some examples of dialogues generated with the exhaustive strategy.

5.4 Dialogue examples

In this section we give examples of well-formed exhaustive dialogues. The first example we give is a well-formed exhaustive argument inquiry dialogue.

t	CS_1^t	m_t	CS_2^t	QS_t
1		$\langle 1, open, dialogue(ai, c \rightarrow d)\rangle$		$QS_1 = \{c, d\}$
2		$\langle 2, close, dialogue(ai, c \rightarrow d)\rangle$		
3		$\langle 1, open, dialogue(ai, b \rightarrow c)\rangle$		$QS_3 = \{b, c\}$
4		$\langle 2, assert, \langle\{(b, 1)\}, b\rangle\rangle$	$(b, 1)$	
5	$(b, 1)$ $(b \rightarrow c, 1)$	$\langle 1, assert, \langle\{(b, 1), (b \rightarrow c, 1)\}, c\rangle\rangle$		
6		$\langle 2, close, dialogue(ai, b \rightarrow c)\rangle$		
7		$\langle 1, open, dialogue(ai, a \rightarrow b)\rangle$		$QS_7 = \{a, b\}$
8		$\langle 2, assert, \langle\{(a, 1)\}, a\rangle\rangle$	$(a, 1)$	
9	$(a, 1)$ $(a \rightarrow b, 1)$	$\langle 1, assert, \langle\{(a, 1), (a \rightarrow b, 1)\}, b\rangle\rangle$		
10		$\langle 2, close, dialogue(ai, a \rightarrow b)\rangle$		
11		$\langle 1, close, dialogue(ai, a \rightarrow b)\rangle$		
12		$\langle 2, close, dialogue(ai, b \rightarrow c)\rangle$		
13		$\langle 1, close, dialogue(ai, b \rightarrow c)\rangle$		
14		$\langle 2, close, dialogue(ai, c \rightarrow d)\rangle$		
15	$(c \rightarrow d, 1)$	$\langle 1, assert, \langle\{(a, 1), (a \rightarrow b, 1), (b \rightarrow c, 1), (c \rightarrow d, 1)\}, d\rangle\rangle$		
16		$\langle 2, close, dialogue(ai, c \rightarrow d)\rangle$		
17		$\langle 1, close, dialogue(ai, c \rightarrow d)\rangle$		

Table 2 Argument inquiry dialogue example.

5.4.1 Argument inquiry dialogue example

Agent 1 wishes to enter into an argument inquiry dialogue with agent 2 in order to try to find an argument for d (where both agents are following the exhaustive strategy). We have

$$\Sigma^1 = \{(c \rightarrow d, 1), (b \rightarrow c, 1), (a \rightarrow b, 1)\} \qquad \Sigma^2 = \{(a, 1), (b, 1)\}$$

Agent 1 is aware of a defeasible rule whose consequent is d and so opens an argument inquiry dialogue with this defeasible rule as its topic. Assume $\mu(a) = 1$, $\mu(\neg a) = 2$, $\mu(b) = 3$, $\mu(\neg b) = 4$, $\mu(c) = 5$, $\mu(\neg c) = 6$, $\mu(d) = 7$ etc. The dialogue proceeds as in Table 2. The table represents the top-level dialogue, the first column gives the value of t, the second column gives the commitment store of agent 1, the third column gives the move m_t, the fourth column gives the commitment store of agent 2, and the fifth column gives the details of any query stores that are not equal to the empty set.

As there are two arguments for d that can be constructed from the union of the two commitment stores at the end of the top-level dialogue D_1^{17}, the outcome of this dialogue is the set of these two arguments.

$$\mathsf{Outcome}_{ai}(D_1^{17}) = \{\langle\{(a, 1), (a \rightarrow b, 1), (b \rightarrow c, 1), (c \rightarrow d, 1)\}, d\rangle, \\ \langle\{(b, 1), (b \rightarrow c, 1), (c \rightarrow d, 1)\}, d\rangle\}$$

There are two sub-dialogues of D_1^{17}: D_3^{13} that terminates at 13 and has topic $b \rightarrow c$; and D_7^{11} that terminates at 11 and has topic $a \rightarrow b$. D_7^{11} is also a sub-dialogue of D_3^{13}.

t	CS_1^t	m_t	CS_2^t	QS_t
1		$\langle 1, open, dialogue(wi, b)\rangle$		
2		$\langle 2, close, dialogue(wi, b)\rangle$		
3	$(a, 4)$ $(a \rightarrow b, 4)$	$\langle 1, assert, \langle\{(a, 4), (a \rightarrow b, 4)\}, b\rangle\rangle$		
4		$\langle 2, assert, \langle\{(d, 3), (d \rightarrow \neg a, 3)\}, \neg a\rangle\rangle$	$(d, 3)$ $(d \rightarrow \neg a, 3)$	
5	$(c, 3)$ $(c \rightarrow \neg b, 3)$	$\langle 1, assert, \langle\{(c, 3), (c \rightarrow \neg b, 3)\}, \neg b\rangle\rangle$		
6		$\langle 2, assert, \langle\{(\neg d, 1)\}, \neg d\rangle\rangle$	$(\neg d, 1)$	
7		$\langle 1, open, dialogue(ai, a \rightarrow b)\rangle$		$QS_7 = \{a, b\}$
8		$\langle 2, close, dialogue(ai, a \rightarrow b)\rangle$		
9		$\langle 1, close, dialogue(ai, a \rightarrow b)\rangle$		
10		$\langle 2, open, dialogue(ai, d \rightarrow \neg a)\rangle$		$QS_{10} = \{d, \neg a\}$
11		$\langle 1, close, dialogue(ai, d \rightarrow \neg a)\rangle$		
12		$\langle 2, close, dialogue(ai, d \rightarrow \neg a)\rangle$		
13		$\langle 1, open, dialogue(ai, c \rightarrow \neg b)\rangle$		$QS_{13} = \{c, \neg b\}$
14		$\langle 2, close, dialogue(ai, c \rightarrow \neg b)\rangle$		
15		$\langle 1, close, dialogue(ai, c \rightarrow \neg b)\rangle$		
16		$\langle 2, open, dialogue(ai, e \rightarrow \neg d)\rangle$		$QS_{16} = \{e, \neg d\}$
17	$(e, 2)$	$\langle 1, assert, \langle\{(e, 2)\}, e\rangle\rangle$		
18		$\langle 2, assert, \langle\{(e, 2), (e \rightarrow \neg d, 2)\}, \neg d\rangle\rangle$	$(e, 2)$ $(e \rightarrow \neg d, 2)$	
19		$\langle 1, close, dialogue(ai, e \rightarrow \neg d)\rangle$		
20		$\langle 2, close, dialogue(ai, e \rightarrow \neg d)\rangle$		
21		$\langle 1, close, dialogue(wi, b)\rangle$		
22		$\langle 2, assert, \langle\{(\neg e, 1)\}, \neg e\rangle\rangle$	$(\neg e, 1)$	
22		$\langle 1, close, dialogue(wi, b)\rangle$		
23		$\langle 2, close, dialogue(wi, b)\rangle$		

Table 3 Warrant inquiry dialogue example.

$$\mathsf{Outcome}_{\mathsf{ai}}(D_3^{13}) = \{\langle\{(a, 1), (a \rightarrow b, 1), (b \rightarrow c, 1)\}, c\rangle, \langle\{(b, 1), (b \rightarrow c, 1)\}, c\rangle\}$$

$$\mathsf{Outcome}_{\mathsf{ai}}(D_7^{11}) = \{\langle\{(a, 1), (a \rightarrow b, 1)\}, b\rangle, \langle\{(b, 1)\}, b\rangle\}$$

The following example is of a well-formed exhaustive warrant inquiry dialogue.

5.4.2 Warrant inquiry dialogue example

Agent 1 wishes to enter into a warrant inquiry dialogue with agent 2 in order to try to find a warrant for an argument for b (where both agents are following the exhaustive strategy). We have

$$\Sigma^1 = \{(a, 4), (a \rightarrow b, 4), (c, 3), (c \rightarrow \neg b, 3), (e, 2)\}$$

$$\Sigma^2 = \{(d, 3), (d \rightarrow \neg a, 3), (\neg d, 1), (e \rightarrow \neg d, 2), (\neg e, 1)\}$$

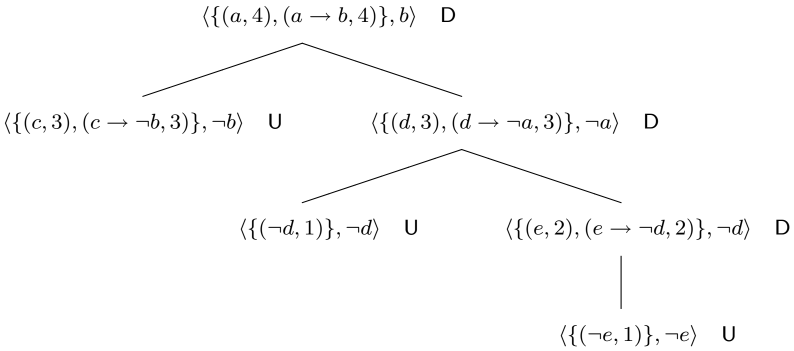


Fig. 4 The marked dialectical tree for warrant inquiry dialogue example.

Agent 1 opens a warrant inquiry dialogue with b as its topic. The dialogue proceeds as in Table 3. Again, we assume $\mu(a) = 1$, $\mu(\neg a) = 2$, $\mu(b) = 3$, $\mu(\neg b) = 4$, $\mu(c) = 5$, $\mu(\neg c) = 6$, $\mu(d) = 7$ etc.

The root argument of this dialogue is $\langle\{(a,4),(a \rightarrow b,4)\}, b\rangle$ (as this is the first argument asserted for the topic), and so the outcome of the top-level warrant inquiry dialogue D_1^{23} depends on the dialectical tree $\mathsf{T}(\langle\{(a,4),(a \rightarrow b,4)\}, b\rangle, CS_1^{23} \cup CS_2^{23})$. The marked version of this dialectical tree is shown in Figure 4.

As the root argument of the tree is defeated, the outcome of the dialogue is the empty set.

$$\mathsf{Outcome}_{\mathsf{wi}}(D_1^{23}) = \{\}$$

Note that there are four nested argument inquiry dialogues that appear as sub-dialogues of D_1^{23}: D_7^9, D_{10}^{12}, D_{13}^{15} and D_{16}^{20}.

$$\mathsf{Outcome}_{\mathsf{ai}}(D_7^9) = \{\}$$

$$\mathsf{Outcome}_{\mathsf{ai}}(D_{10}^{12}) = \{\}$$

$$\mathsf{Outcome}_{\mathsf{ai}}(D_{13}^{15}) = \{\}$$

$$\mathsf{Outcome}_{\mathsf{ai}}(D_{16}^{20}) = \{\langle\{(e,2),(e \rightarrow \neg d,2)\}, \neg d\rangle\}$$

The final example we give is an example of a well-formed exhaustive warrant inquiry dialogue from the medical domain.

t	CS_1^t	m_t	CS_2^t	QS_t
1		$\langle 2, open, dialogue(wi, mam)\rangle$		
2		$\langle 1, close, dialogue(wi, mam)\rangle$		
3		$\langle 2, open, dialogue(ai, pain \wedge nonCyc \rightarrow mam)\rangle$		$QS_3 = \{pain, nonCyc, mam\}$
4	$(pain, 1)$	$\langle 1, assert, \langle\{(pain, 1)\}, pain\rangle\rangle$		
5		$\langle 2, close, dialogue(ai, pain \wedge nonCyc \rightarrow mam)\rangle$		
6	$(nonCyc, 1)$	$\langle 1, assert, \langle\{(nonCyc, 1)\}, nonCyc\rangle\rangle$		
7		$\langle 2, assert, \langle\{(pain, 1), (nonCyc, 1), (pain \wedge nonCyc \rightarrow mam, 3)\}, mam\rangle\rangle$	$(pain, 1)$ $(nonCyc, 1)$ $(pain \wedge nonCyc \rightarrow mam, 3)$	
8		$\langle 1, close, dialogue(ai, pain \wedge nonCyc \rightarrow mam)\rangle$		
9		$\langle 2, close, dialogue(ai, pain \wedge nonCyc \rightarrow mam)\rangle$		
10		$\langle 1, close, dialogue(wi, mam)\rangle$		
11		$\langle 2, open, dialogue(ai, preg \rightarrow \neg mam)\rangle$		$QS_{11} = \{preg, \neg mam\}$
12	$(preg, 2)$	$\langle 1, assert, \langle\{(preg, 2)\}, preg\rangle\rangle$		
13		$\langle 2, assert, \langle\{(preg, 2), (preg \rightarrow \neg mam, 1)\}, \neg mam\rangle\rangle$	$(preg, 2)$ $(preg \rightarrow \neg mam, 1)$	
14		$\langle 1, close, dialogue(ai, preg \rightarrow \neg mam)\rangle$		
15		$\langle 2, close, dialogue(ai, preg \rightarrow \neg mam)\rangle$		
16		$\langle 1, close, dialogue(wi, mam)\rangle$		
17		$\langle 2, close, dialogue(wi, mam)\rangle$		

Table 4 Warrant inquiry dialogue example from medical domain.

5.4.3 Warrant inquiry dialogue example from medical domain

Agent 1 represents a specific patient's agent, and keeps track of all data relevant to the patient (and so subsumes the role of electronic patient record). The patient is suffering from breast symptoms and is worried she may have breast cancer. She has been referred by her general practitioner to a breast surgeon for diagnosis of these symptoms. The breast surgeon's agent is agent 2. As part of the diagnostic procedure, the breast surgeon must decide whether the patient needs imaging and what form this imaging should take (mammogram or ultrasound), and so the surgeon must collect all acceptable arguments for and against these options in order to decide which is the best course of action. There are many things that the surgeon must take into account during the argumentation process and, as this example covers only a very small part of this space, the reader should not assume anything about the validity of the medical knowledge represented here. We have

$$\Sigma^1 = \{(pain, 1), (nonCyc, 1), (preg, 2)\}$$

$$\Sigma^2 = \{(pain \wedge nonCyc \rightarrow mam, 3), (preg \rightarrow \neg mam, 1)\}$$

$$\langle\{(pain, 1), (nonCyc, 1), (pain \wedge nonCyc \rightarrow mam, 3)\}, mam\rangle \quad \mathsf{D}$$

$$|$$

$$\langle\{(preg, 2), (preg \rightarrow \neg mam, 1)\}, \neg mam\rangle \quad \mathsf{U}$$

Fig. 5 The marked dialectical tree for warrant inquiry dialogue example from medical domain.

where *pain* represents that the patient has breast pain, *nonCyc* represents that the patient's breast pain is non-cyclical, *preg* represents that the patient is pregnant, and *mam* represents that the patient should have a mammogram.

Agent 2 opens a warrant inquiry dialogue with *mam* as its topic. The dialogue proceeds as in Table 4 (where both agents are following the exhaustive strategy). We assume $\mu(pain) = 1$, $\mu(nonCyc) = 2$, $\mu(preg) = 3$, $\mu(mam) = 4$, $\mu(\neg mam) = 5$.

The root argument of this dialogue is

$$\langle\{(pain, 1), (nonCyc, 1), (pain \wedge nonCyc \rightarrow mam, 3)\}, mam\rangle$$

as so the outcome of the top-level warrant inquiry dialogue D_1^{17} depends on the dialectical tree

$$\mathsf{T}(\langle\{(pain, 1), (nonCyc, 1), (pain \wedge nonCyc \rightarrow mam, 3)\}, mam\rangle, CS_1^{17} \cup CS_2^{17}).$$

The marked version of this dialectical tree is shown in Figure 5. As the root argument of the tree is defeated, the outcome of the dialogue is the empty set.

$$\mathsf{Outcome_{wi}}(D_1^{17}) = \{\}$$

Note that there are two nested argument inquiry dialogues that appear as subdialogues of D_1^{17}: D_3^9 and D_{11}^{15}.

$$\mathsf{Outcome_{ai}}(D_3^9) = \{\langle\{(pain, 1), (nonCyc, 1), (pain \wedge nonCyc \rightarrow mam, 3)\}, mam\rangle\}$$

$$\mathsf{Outcome_{ai}}(D_{11}^{15}) = \{\langle\{(preg, 2), (preg \rightarrow \neg mam, 1)\}, \neg mam\rangle\}$$

6 Properties of the dialogue system

We believe it is important to consider soundness and completeness properties if we are to understand the behaviour of our dialogues. This is particularly the case when dealing with a domain such as the medical domain, where we wish the appropriate outcome to be guaranteed given a particular situation. One can imagine that in a competitive domain we would want it to be possible for the agents to behave in an intelligent manner in order to be able to influence the outcome of the dialogue; in the safety-critical medical domain, we wish the agents to always arrive at the best possible outcome given their beliefs. When defining soundness and completeness properties, we compare the outcome of our dialogues with the benchmark that is the set of arguments that can be constructed from the union of the participating agents' beliefs.

We consider this benchmark to be the 'ideal' outcome, as there are clearly no constraints on the sharing of beliefs. However, it is only ideal if we accept that the agents each have the same level of expertise regarding the beliefs. Consider the situation in which a medical student is discussing a diagnosis with a consultant. In this situation, the ideal benchmark might be the outcome that the consultant would reach *without* taking into account any of the student's beliefs. Or there may be a situation in which we wish our argument inquiry dialogues not to produce every argument that the agents can jointly construct for a certain claim, but only those which are considered to be the 'best' in some sense. We would like to explore other possible benchmarks as future work.

In this section we define soundness and completeness properties for both argument inquiry and warrant inquiry dialogues and show that all dialogues generated by our system are sound and complete. We first deal with argument inquiry dialogues and then with warrant inquiry dialogues.

6.1 Argument inquiry dialogues

We say that an argument inquiry dialogue is sound if and only if, when the outcome of the dialogue includes an argument, then that same argument can be constructed from the union of the two participating agents' beliefs.

Definition 35 Let D_r^t be a well-formed argument inquiry dialogue. We say that D_r^t is **sound** iff, if $\langle \Phi, \phi \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^t)$, then $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$.

In order to show that all argument inquiry dialogues are sound we need to introduce some lemmas. The first states that if an agent asserts an argument, then it must be able to construct the argument from its beliefs and the other agent's commitment store. This is clear from the definition of the exhaustive strategy.

Lemma 1 *Let D_r^t be a well-formed argument inquiry dialogue. If D_1^t is a top-dialogue of D_r^t and $\Omega_x(D_1^t) = \langle x, assert, \langle \Phi, \phi \rangle \rangle$, then $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^x \cup CS_{\hat{x}}^t)$.*
Proof: *The exhaustive strategy (Def. 33) states that if $\Omega_x(D_1^t) = \langle x, assert, \langle \Phi, \phi \rangle \rangle$ then the condition $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^x \cup CS_{\hat{x}}^t)$ must hold.* □

From Lemma 1 and the fact that the commitment stores are only updated when an assert move is made, we get the lemma that a commitment store is always a subset of the union of the two agents' beliefs.

Lemma 2 *If D_r^t is a well-formed exhaustive argument inquiry dialogue, then $CS_1^t \cup CS_2^t \subseteq \Sigma^1 \cup \Sigma^2$.*
Proof: *The only time that a commitment store is changed is when an agent x makes the move $\langle x, assert, \langle \Phi, \phi \rangle \rangle$ (Def. 22). From Lem. 1, we see that for $\langle x, assert, \langle \Phi, \phi \rangle \rangle$ to be a move made at point $t+1$ in a dialogue, the condition $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^x \cup CS_{\hat{x}}^t)$ must hold, hence $\Phi \subseteq \Sigma^x \cup CS_{\hat{x}}^t$ (Def. 6). As a commitment store is empty when $t = 0$, any member of the union of the commitment stores must also be a member of the union of the agents' beliefs, hence $CS_1^t \cup CS_2^t \subseteq \Sigma^1 \cup \Sigma^2$.* □

The next lemma states that if we have a set Φ that is a subset of a set of beliefs Ψ, then the set of arguments that can be constructed from Φ is a subset of the set of arguments that can be constructed from Ψ.

Lemma 3 *Let $\Phi \subseteq \mathcal{B}$ and $\Psi \subseteq \mathcal{B}$ be two sets. If $\Phi \subseteq \Psi$, then $\mathcal{A}(\Phi) \subseteq \mathcal{A}(\Psi)$.*
Proof: *Assume that $\Phi \subseteq \Psi$ and $\langle \Pi, \pi \rangle$ is an argument s.t. $\langle \Pi, \pi \rangle \in \mathcal{A}(\Phi)$. From Def. 6, we see that $\Pi \subseteq \Phi$. As $\Pi \subseteq \Phi$, $\Phi \subseteq \Psi$ and the subset relationship is transitive, $\Pi \subseteq \Psi$. Hence, $\langle \Pi, \pi \rangle \in \mathcal{A}(\Psi)$ (Def. 6). Hence, if $\Phi \subseteq \Psi$ and $A \in \mathcal{A}(\Phi)$ then $A \in \mathcal{A}(\Psi)$. Hence, if $\Phi \subseteq \Psi$ then $\mathcal{A}(\Phi) \subseteq \mathcal{A}(\Psi)$.* □

We now show that argument inquiry dialogues generated with the exhaustive strategy are sound.

Proposition 2 *If D_r^t is a well-formed exhaustive argument inquiry dialogue, then D_r^t is sound.*
Proof: *Assume $\langle \Phi, \phi \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^t)$. From Def. 26, $\langle \Phi, \phi \rangle \in \mathcal{A}(CS_1^t \cup CS_2^t)$. From Lem. 2, $CS_1^t \cup CS_2^t \subseteq \Sigma^1 \cup \Sigma^2$. Hence, from Lem. 3, $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$. Hence, D_r^t is sound.* □

Similarly, an argument inquiry dialogue is complete if and only if, if the dialogue terminates at t and it is possible to construct an argument for a literal in the query store from the union of the two participating agents' beliefs, then that argument will be in the outcome of the dialogue at t.

Definition 36 Let D_r^t be a well-formed argument inquiry dialogue. We say that D_r^t is **complete** iff, if D_r^t terminates at t and $\phi \in QS_r$ and there exists Φ such that $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$, then $\langle \Phi, \phi \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^t)$.

In order to show that all exhaustive argument inquiry dialogues are complete we need to give some further lemmas. The first states that if a well-formed exhaustive dialogue (either argument inquiry or warrant inquiry) terminates at t, then the set of legal moves from which an agent must choose the move m_t does not include any open or assert moves. This is clear from the definition of the exhaustive strategy.

Lemma 4 *If D_r^t is a well-formed exhaustive dialogue that terminates at t such that $\mathsf{Sender}(m_{t-1}) = \hat{x}$ and D_r^t extends D_r^{t-1} and D_1^{t-1} is a top-dialogue of D_r^{t-1}, then (for $x \in \mathcal{I}$) $\mathsf{Asserts}_x(D_1^{t-1}) = \emptyset$ and $\mathsf{Opens}_x(D_1^{t-1}) = \emptyset$.*
Proof: *A dialogue is terminated with a matched-close (Def. 19). The exhaustive strategy (Def. 33) states that a close move will only be made by x if the sets $\mathsf{Asserts}_x(D_1^t)$ and $\mathsf{Opens}_x(D_1^t)$ are empty. Hence, $\mathsf{Asserts}_x(D_1^t) = \mathsf{Opens}_x(D_1^t) = \emptyset$.* □

The following lemma states that if two agents are in an argument inquiry dialogue that terminates at t and there exists some r $(1 \leq r < t)$ such that $\phi \in QS_r$ and there is an argument for ϕ of the form $\langle \{\phi\}, \phi \rangle$ that can be constructed from the union of the two agents' beliefs, then ϕ will be in the union of the commitment stores at timepoint t. We get this lemma from Lemma 4 and the definitions of the exhaustive strategy and the argument inquiry protocol.

Lemma 5 *For all r $(1 \leq r < t)$, if D_r^t is a well-formed exhaustive argument inquiry dialogue that terminates at t such that $\phi \in QS_r$ and there exists $\langle \{(\phi, L)\}, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$, then $\phi \in CS_1^t \cup CS_2^t$.*
Proof: *Let us assume that $\phi \in QS_r$ and $\langle \{(\phi, L)\}, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$, hence (for $x \in \mathcal{I}$) $\langle \{(\phi, L)\}, \phi \rangle \in \mathcal{A}(\Sigma^x \cup CS_{\hat{x}}^{t_2})$ for all values of t_2 s.t $1 \leq t_2 \leq t$ (from Def. 6). From Def. 24 and Def. 33, for all t_3 s.t. $r < t_3 < t$, $\langle x, assert, \langle \{(\phi, L)\}, \phi \rangle \rangle \in \mathsf{Asserts}_x(D_1^{t_3})$*

unless there exists a t_4, $1 < t_4 < t_3$, *s.t.* $m_{t_4} = \langle \_, assert, \langle \{(\phi, L)\}, \phi \rangle \rangle$. *As* D_1^t *terminates at t and from Lem. 4,* $\mathsf{Asserts}_x(D_1^t) = \mathsf{Opens}_x(D_1^t) = \emptyset$, *hence it must be true that there exists* t_4 $(1 < t_4 < t)$, *s.t.* $m_{t_4} = \langle \_, assert, \langle \{(\phi, L)\}, \phi \rangle \rangle$, *hence* $\phi \in CS_x^{t_4} \cup CS_{\hat{x}}^{t_4}$ $(1 < t_4 < t)$, *hence* $\phi \in CS_1^t \cup CS_2^t$. □

From Lemma 4 and the definitions of the exhaustive strategy and the argument inquiry protocol we also get the following lemma that if there is a defeasible rule whose consequent is present in the query store, then there will be a timepoint at which a query store will be created that contains all the literals of the defeasible rule.

Lemma 6 *For all r* $(1 \leq r < t)$, *if* D_r^t *is a well-formed exhaustive argument inquiry dialogue that terminates at t such that* $\phi \in QS_r$ *and there exists a domain belief* $(\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \phi, L) \in \Sigma^1 \cup \Sigma^2$, *then there exists* t_1 $(1 < t_1 < t)$ *such that* $QS_{t_1} = \{\alpha_1, \ldots, \alpha_n, \phi\}$ *and* D_r^t *extends* $D_r^{t_1}$.
Proof: *Assume* $(\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \phi, L) \in \Sigma^x$ *(where* $x \in \mathcal{I}$*),* $\phi \in QS_r$, $\{\alpha_1, \ldots \alpha_n, \phi\} \not\subseteq QS_r$ *and the dialogue* $D_r^{t_2}$ *terminates at* t_2. *From Def. 24 and Def. 33, we see that, for* t_3 *s.t.* $\mathsf{Current}(D_1^{t_3}) = D_r^{t_3}$, $\langle x, open, \alpha_1 \wedge \ldots \wedge \alpha \rightarrow \phi \rangle \in \mathsf{Opens}_x(D_1^{t_3})$ $(r \leq t_3 < t_2)$ *unless such a move has already been made at timepoint* t_4 $(1 < t_4 < t_3)$ *in which case* $QS_{t_4} = \{\alpha_1, \ldots, \alpha_n, \phi\}$. *As* $D_r^{t_2}$ *terminates at* t_2 *and from Lem. 4,* $\mathsf{Asserts}_x(D_1^{t_2}) = \mathsf{Opens}_x(D_1^{t_2}) = \emptyset$, *hence there exists* t_4 $(1 < t_4 < t_3 < t)$ *s.t.* $m_{t_4} = \langle x, open, \alpha_1 \wedge \ldots \wedge \alpha \rightarrow \phi \rangle$, *hence* $QS_{t_4} = \{\alpha_1, \ldots, \alpha_n, \phi\}$. □

We now show that argument inquiry dialogues generated with the exhaustive strategy are complete.

Proposition 3 *If* D_r^t *is a well-formed exhaustive argument inquiry dialogue, then* D_r^t *is complete.*
Proof: *If* D_r^t *does not terminate at t then* D_r^t *is complete. So assume* D_r^t *terminates at t,* $\phi \in QS_r$, *and* $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$. *By Def. 6,* $\Phi \subseteq \Sigma^1 \cup \Sigma^2$. *There are two cases. (Case 1)* $\Phi = \{(\phi, L)\}$. *Hence by Lem. 5,* $\phi \in CS_1^t \cup CS_2^t$. *From Def. 26,* $\langle \Phi, \phi \rangle \in \mathsf{Outcome_{ai}}(D_r^t)$. *(Case 2) There exists* $(\alpha_1 \wedge \ldots \wedge \alpha_n \rightarrow \phi, L) \in \Phi$. *By Def. 6, for all* α_i *there exists* Φ_i *s.t.* $\langle \Phi_i, \alpha_i \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$. *From Lem. 6, there exists* t_1 $(1 < t_1 \leq t)$ *s.t.* $QS_{t_1} = \{\alpha_1, \ldots, \alpha_n, \phi\}$. *Each* Φ_i *is either an example of case 1 or case 2, so, by recursion, there exists* r_2, t_2 $(r < r_2 < t_2 \leq t)$ *s.t.* $\langle \Phi_i, \alpha_i \rangle \in \mathsf{Outcome_{ai}}(D_{r_2}^{t_2})$. *Hence, from Def. 6,* $\langle \Phi, \phi \rangle \in \mathsf{Outcome_{ai}}(D_r^t)$. □

The soundness and completeness results we have given here are particularly interesting if we know that an argument inquiry dialogue terminates. Fortunately, we can show that all dialogues (both argument inquiry and warrant inquiry) generated with the exhaustive strategy terminate (as agents' belief bases are finite, hence there are only a finite number of assert and open moves that can be generated and agents cannot repeat these moves).

Proposition 4 *For any well-formed exhaustive dialogue* D_r^t, *there exists a* t_1 $(r < t \leq t_1)$ *such that* $D_r^{t_1}$ *terminates at* t_1 *and* $D_r^{t_1}$ *extends* D_r^t.
Proof: *An agent's belief base is assumed to be finite. The exhaustive strategy (Def. 33) states that the set of assert moves from which an agent may select a move to make depends on the arguments that an agent can construct from the union of its beliefs and the other agents commitment store. The other agent's commitment store is a subset of the union of the agents' beliefs (Lem. 2) and so is also finite, hence there can only be a finite number of assert moves that are available to an agent throughout the dialogue.*

Similarly, the exhaustive strategy states that an agent can only make an open move if the content of that move is a belief of the agent, as the beliefs are finite this means that there can only be a finite number of open moves available to the agent throughout the dialogue. As both the protocols (Def. 24, Def. 28) state that agents cannot repeat moves, each agent participating in a dialogue will, therefore, eventually exhaust the set of assert or open moves they may make. The exhaustive strategy states that when this happens the agents must each make a close move, hence giving us a matched close and terminating the dialogue. □

From combining our completeness proposition with the fact that exhaustive argument inquiry dialogues always terminate, we get the desired result that if an argument can be constructed from the union of the two participating agents' beliefs whose claim is a literal from the current query store, then there will come a timepoint at which that argument is in the outcome of the dialogue.

Proposition 5 *Let D_r^t be a well-formed exhaustive argument inquiry dialogue. If $\phi \in QS_r$ and there exists Φ such that $\langle \Phi, \phi \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$, then there exists t_1 ($1 < t_1$) such that $D_r^{t_1}$ extends D_r^t and $\langle \Phi, \phi \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^{t_1})$.*
Proof: *This follows from Prop. 3 and Prop. 4.* □

To summarise this section, each argument inquiry dialogue generated with the exhaustive strategy terminates such that the set of arguments that is its outcome is exactly the same as the set of all arguments that have as their claim a literal from the query store and that are constructed from the union of the participating agents' beliefs.

6.2 Warrant inquiry dialogues

The goal of a warrant inquiry dialogue is for two agents to share relevant parts of their knowledge in order to jointly construct (from the union of their commitment stores) a dialectical tree that has an argument for the topic of the dialogue at the root (henceforth referred to as the *dialogue tree*). This tree then acts as a warrant for the root argument if and only if the status of the root node is U (i.e. it is undefeated). When defining soundness and completeness properties, the benchmark that we compare this to is the dialectical tree that has the same root argument as the dialogue tree but is constructed from the union of the two agents' beliefs. Again, this benchmark is in a sense the 'ideal' situation, in which there are no constraints on the sharing of beliefs.

We say that a warrant inquiry dialogue is sound if and only if, if the outcome of the terminated dialogue is an argument $\langle \Phi, \phi \rangle$ and T is a dialectical tree that has $\langle \Phi, \phi \rangle$ at its root and is constructed from the union of the participating agents' beliefs, then the status of the root node of T is U.

Definition 37 Let D_r^t be a well-formed warrant inquiry dialogue. D_r^t is **sound** iff, if D_r^t terminates at t and $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$, then $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$.

Similarly, a warrant inquiry dialogue is complete if and only if, if the root argument of the dialogue is $\langle \Phi, \phi \rangle$ and the status of the root node of a dialectical tree that has $\langle \Phi, \phi \rangle$ at its root and is constructed from the union of the participating agents' beliefs is U, then the outcome of the dialogue when it is terminated is $\langle \Phi, \phi \rangle$.

Definition 38 Let D_r^t be a well-formed warrant inquiry dialogue. D_r^t is **complete** iff, if D_r^t terminates at t, $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$ and $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$, then $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$.

In order to show that warrant inquiry dialogues are sound and complete, we will show that the dialogue tree is in fact equal to the dialectical tree that has the same argument at its root but is constructed from the union of the participating agents' beliefs. As the outcome of the warrant inquiry dialogue is determined by the status of the root node of the dialogue tree, it is clear that if the outcome of the warrant inquiry dialogue is the argument $\langle \Phi, \phi \rangle$ then the status of the root node of the dialectical tree that is constructed from the union of the agents' beliefs and has $\langle \Phi, \phi \rangle$ at its root is U (given that the dialogue tree is equal to this dialectical tree). Similarly, if the root argument of the dialogue is $\langle \Phi, \phi \rangle$ and the status of the root node of the dialectical tree that is constructed from the union of the agents' beliefs and has $\langle \Phi, \phi \rangle$ at its root is U, then the outcome of the dialogue must be $\langle \Phi, \phi \rangle$.

To show that the dialogue tree is equal to the dialectical tree that has the root argument of the dialogue at its root and is constructed from the union of the agents' beliefs, we must show that if a path from the root node appears in one then it will also appear in the other. First, we will show that if we have an exhaustive warrant inquiry dialogue D_r^t that terminates at t, whose root argument is $\langle \Phi, \phi \rangle$, and there is a path from the root node $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ that appears in the dialogue tree constructed during D_r^t, then the same path from the root node appears in the dialectical tree that is constructed from the union of the two participating agents' beliefs and has $\langle \Phi, \phi \rangle$ at its root. This is due to the relationship between the commitment stores and the agents' beliefs (i.e. the union of the commitment stores is a subset of the union of the beliefs).

Lemma 7 *Let D_r^t be a well-formed exhaustive warrant inquiry dialogue that terminates at t such that $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$. If there exists a path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ in the dialogue tree $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$, then the same path exists in the dialectical tree $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$.*

Proof: *Assume the path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ appears in $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$. From Def. 12, $\forall i$ $(1 \leq i \leq n)$ $\langle \Phi_i, \phi_i \rangle \in \mathcal{A}(CS_1^t \cup CS_2^t)$, hence $\langle \Phi_i, \phi_i \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$ (from Lem. 2 and Lem. 3), hence the path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ can also be constructed from $\Sigma^1 \cup \Sigma^2$. As $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ is an acceptable argumentation line in $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$, it must also be an acceptable argumentation line in $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$ (from Def. 11). Hence, if there exists a path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ in $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$, then there exists a path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ in $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$.* $\square$

The next lemma is complementary to the previous one. It states that if we have an exhaustive warrant inquiry dialogue D_r^t that terminates at t, whose root argument is $\langle \Phi, \phi \rangle$, and there is a path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ that appears in the dialectical tree that is constructed from the union of the two participating agents' beliefs and has $\langle \Phi, \phi \rangle$ at its root, then the same path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ appears in the dialogue tree. This is due to the fact that the warrant inquiry protocol along with the exhaustive strategy ensures that all arguments that change the dialogue tree (i.e. cause a new node to be added to the tree) get asserted during the dialogue.

Lemma 8 *Let D_r^t be a well-formed exhaustive warrant inquiry dialogue that terminates at t such that $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$. If D_1^t is a top-dialogue of D_r^t and there* *exists a path* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ *in* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$*, then there exists a path* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ *in the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$.

Proof: *Assume the path* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ *appears in* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$*. Also assume there exists* t_1 *such that* $\Phi \subseteq CS_1^{t_1} \cup CS_2^{t_1}$ *and there does not exist* t' *such that* $1 < t' < t_1$ *and* $\Phi \subseteq CS_1^{t'} \cup CS_2^{t'}$ *(i.e.* t_1 *is the timepoint at which the root argument is asserted). From Def. 12,* $\langle \Phi_i, \phi_i \rangle \in \mathcal{A}(\Sigma^1 \cup \Sigma^2)$ $(1 \leq i \leq n)$*. There are two cases.*

(Case 1) $\langle \Phi_i, \phi_i \rangle \in \mathcal{A}(\Sigma^x)$ *where* $x \in \mathcal{I}$.

(Case 2) $\langle \Phi_i, \phi_i \rangle \notin \mathcal{A}(\Sigma^1)$ *and* $\langle \Phi_i, \phi_i \rangle \notin \mathcal{A}(\Sigma^2)$*, in which case there exists a defeasible rule* $(\alpha_1 \wedge \ldots \wedge \alpha_m \rightarrow \phi_i, L) \in \Phi_i$ *such that* $(\alpha_1 \wedge \ldots \wedge \alpha_m \rightarrow \phi_i, L) \in \Sigma^x$*, where* $x \in \mathcal{I}$.

Let us now consider $\langle \Phi_1, \phi_1 \rangle$*. It is either an instance of case 1, or of case 2.*

If it is case 1: From Def. 28 and Def. 33, for all t_2 *such that* $t_1 < t_2 \leq t$ *and* D_1^t *extends* $D_1^{t_2}$*,* $\langle x, assert, \langle \Phi_1, \phi_1 \rangle \rangle \in \mathsf{Asserts}_x(D_1^{t_2})$ *unless* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t_2} \cup CS_2^{t_2} \cup \Phi_1)$ *equals* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t_2} \cup CS_2^{t_2})$ *(i.e making the move to assert* $\langle \Phi_1, \phi_1 \rangle$ *does not change the dialogue tree).*

As the path $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *appears in* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$ *it must be the case (from Def. 12) that* $\langle \Phi_1, \phi_1 \rangle$ *is a defeater for* $\langle \Phi, \phi \rangle$ *and the argumentation line* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *is acceptable. Hence if* $\langle x, assert, \langle \Phi_1, \phi_1 \rangle \rangle \notin \mathsf{Asserts}_x(D_1^{t_2})$ *then the argumentation line* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *must already appear in the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t_2} \cup CS_2^{t_2})$*, otherwise asserting* $\langle \Phi, \phi \rangle$ *would certainly change the dialogue tree. As* $\mathsf{Asserts}_x(D_1^{t-1}) = \emptyset$ *(Lem. 4) it must be the case that* $\langle x, assert, \langle \Phi_1, \phi_1 \rangle \rangle \notin \mathsf{Asserts}_x(D_1^{t-1})$ *and so the argumentation line* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *must already appear in the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t-1} \cup CS_2^{t-1})$*. Hence, there must exist* t_3 *such that* $t_1 < t_3 < t$ *and* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *appears in* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t_3} \cup CS_2^{t_3})$.

If it is case 2: As $\langle \Phi_1, \phi_1 \rangle$ *is a defeater for* $\langle \Phi, \phi \rangle$*,* $\neg\phi_1 \in \mathsf{DefDerivations}(CS_1^{t_1} \cup CS_2^{t_1})$*. Hence, from Def. 28 and Def. 33, for all* t_4 *such that* $t_1 < t_4 \leq t$*,* $\langle x, open, dialogue(ai, \alpha_1 \wedge \ldots \wedge \alpha_m \rightarrow \phi_1) \rangle \in \mathsf{Opens}_x(D_1^{t_4})$ *unless there exists a* t_5 *such that* $1 < t_5 \leq t_4$ *and* $QS_{t_5} = \{\alpha_1, \ldots, \alpha_m, \phi_1\}$*, in which case, from Prop. 5, there exists* t_6 *such that* $t_5 \leq t_6 < t$ *and* $\langle \Phi_1, \phi_1 \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^{t_6})$*. As* $\mathsf{Opens}_x(D_1^{t-1}) = \emptyset$ *(Lem. 4), it must be the case that there exists* t_6 *such that* $t_5 \leq t_6 < t$ *and* $\langle \Phi_1, \phi_1 \rangle \in \mathsf{Outcome}_{\mathsf{ai}}(D_r^{t_6})$*. From Def. 26,* $\langle \Phi_1, \phi_1 \rangle \in CS_1^{t_6} \cup CS_2^{t_6}$*, hence (from Def. 12)* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle]$ *appears in the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^{t_6} \cup CS_2^{t_6})$.

Now let us consider $\langle \Phi_2, \phi_2 \rangle$*. We can apply the same reasoning again to show that there will exist a timepoint at which* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \langle \Phi_2, \phi_2 \rangle]$ *appears in the dialogue tree. If we now consider* $\langle \Phi_3, \phi_3 \rangle$ *we can apply the same reasoning to show us that there will exist a timepoint at which* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \langle \Phi_2, \phi_2 \rangle, \langle \Phi_3, \phi_3 \rangle]$ *appears in the dialogue tree. Hence, if we continued in this way, by recursion there exists a path* $[\langle \Phi, \phi \rangle, \langle \Phi_1, \phi_1 \rangle, \ldots, \langle \Phi_n, \phi_n \rangle]$ *in the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$. □

From the previous two lemmas we get the following proposition, which states that if we have a well-formed exhaustive warrant inquiry dialogue D_r^t that terminates at t, whose root argument is $\langle \Phi, \phi \rangle$, then the dialogue tree equals the dialectical tree that is constructed from the union of the two participating agents' beliefs and has $\langle \Phi, \phi \rangle$ as its root.

Proposition 6 *If* D_r^t *is a well-formed exhaustive warrant inquiry dialogue that terminates at* t *such that* $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$*, then the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$ *equals the dialectical tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t)$.

Proof: *This follows from Lem. 7 and Lem. 8.* □

Now we have shown that the dialogue tree produced in a warrant inquiry dialogue is equal to the dialectical tree that is constructed from the union of the agents' beliefs and has the same argument at its root, it follows that warrant inquiry dialogues are sound and complete.

Proposition 7 *If D_r^t is a well-formed exhaustive warrant inquiry dialogue, then D_r^t is sound.*
Proof: *If D_r^t does not terminate at t, then D_r^t is sound. So, assume D_r^t terminates at t and that* $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$. *If* $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$, *then* $\mathsf{Status}(\langle \Phi, \phi \rangle, CS_1^t \cup CS_2^t) = \mathsf{U}$ *(from Def. 30), hence if* $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$, *then* $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$ *(as* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^2 \cup CS_2^t)$ *equals* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$*, from Prop. 6). Hence, D_r^t is sound.* □

Proposition 8 *If D_r^t is a well-formed exhaustive warrant inquiry dialogue, then D_r^t is complete.*
Proof: *If D_r^t does not terminate at t, then D_r^t is complete. So, assume that D_r^t terminates at t and that* $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$. *Prop. 6 states that the dialogue tree* $\mathsf{T}(\langle \Phi, \phi \rangle, CS_1^2 \cup CS_2^t)$ *equals* $\mathsf{T}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2)$. *Hence if* $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$, *then* $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$ *(from Def. 30). Hence, D_r^t is complete.* □

We can combine these propositions with our result that all dialogues terminate to give us the following two propositions. The first states that all warrant inquiry dialogues have a continuation such that if the outcome of this continuation is $\langle \Phi, \phi \rangle$, then the status of the root node of the dialectical tree that is constructed from the union of the agents' beliefs with $\langle \Phi, \phi \rangle$ at its root is U.

Proposition 9 *Let D_r^t be a well-formed exhaustive warrant inquiry dialogue. There exists t' such that $r < t'$, $D_r^{t'}$ extends D_r^t, and if* $\mathsf{Outcome}_{\mathsf{wi}}(D_r^{t'}) = \langle \Phi, \phi \rangle$, *then* $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$.
Proof: *This follows from Prop. 4 and Prop. 7.* □

The next Proposition states that all warrant inquiry dialogues have a continuation such that if the root argument of the dialogue is $\langle \Phi, \phi \rangle$ and the status of the root node of the dialectical tree that is constructed from the union of the participating agents' beliefs and has $\langle \Phi, \phi \rangle$ at its root is U, then the outcome of the continuation of the dialogue is $\langle \Phi, \phi \rangle$.

Proposition 10 *Let D_r^t be a well-formed exhaustive warrant inquiry dialogue. There exists t' such that $r < t'$, $D_r^{t'}$ extends D_r^t, and if* $\mathsf{RootArg}(D_r^t) = \langle \Phi, \phi \rangle$ *and* $\mathsf{Status}(\langle \Phi, \phi \rangle, \Sigma^1 \cup \Sigma^2) = \mathsf{U}$, *then* $\mathsf{Outcome}_{\mathsf{wi}}(D_r^t) = \langle \Phi, \phi \rangle$.
Proof: *This follows from Prop. 4 and Prop. 8.* □

We have now shown that all warrant inquiry dialogues generated by the exhaustive strategy terminate such that the dialogue tree produced is equal to the equivalent dialectical tree constructed from the union of the participating agents' beliefs. The reader may be interested to know that we have defined another strategy (in the first author's PhD thesis [11]) that produces sound and complete warrant inquiry dialogues in which the dialogue tree is a pruned version of the equivalent dialectical tree constructed from the union of the participating agents' beliefs

7 Related work

There is another group who have done work on distributing García and Simari's Defeasible Logic Programming. They have presented a system [48] which allows a group of agents (including a moderator) to share arguments and jointly construct a dialectical tree. However, whilst they give a functional description of the argumentation process, they do not provide a protocol or strategy for the agents to follow in order to carry out the argumentation process. As we have done, they compare the behaviour of their system with that produced by a single agent reasoning with the union of the distributed agents' beliefs. They also give soundness and completeness results; however, these depend on the assumption that arguments are not split between agents, i.e. for each argument that can be constructed from the union of all agents' beliefs, it must be the case that a single agent can construct the argument from its beliefs alone. These results are weaker than our soundness and completeness results, which hold no matter how the beliefs are split across the agents.

Another work that has an equivalent aim to ours is [16]. The authors of [16] present a framework in which agents can exchange arguments to determine the acceptability of an argument in question. Notably, the agents are also able to jointly construct arguments. However, although they informally define a protocol and sketch an algorithm for multi-agent argumentation it is not clear that this is sufficient for agents to generate dialogues, nor do they give soundness and completeness results for their system.

The soundness and completeness results given here represent a key contribution of this work. As most existing dialogue systems provide a protocol but no strategy, it is hard to analyse the behaviour of the dialogues produced and hard to consider soundness and completeness results for such systems. There are some results on termination of dialogues, e.g.: Sadri *et al.* [47] show that a dialogue under their protocol always terminates in a finite number of steps; Parsons *et al.* [38,39] consider the termination properties of the protocols given in [3,4]. There are also some complexity results, e.g.: Parsons *et al.* [38,39], and Dunne *et al.* [21,22] consider questions such as "How many algorithm steps are required, for the most efficient algorithm, for a participant to decide what to utter in a dialogue under a given protocol?" and "How many dialogue utterances are required for normal termination of a dialogue under the protocol?".

Bentahar *et al.* [8] show soundness and completeness properties of their dialogue system that are equivalent to the soundness and completeness properties we define (i.e. they use the union of the agents' knowledge to define their benchmark). Their system, like ours, allows two agents to share knowledge to establish the acceptability of an argument; as they allow partial arguments to be exchanged, their dialogues can also take into account arguments that cannot be constructed by a single agent alone. However, their system (which is based on assumption-based argumentation) assumes that the each agent shares the same set of rules, whereas we make no assumptions about the division of beliefs between the agents. They also do not use different dialogue types to distinguish the joint construction of arguments from the process of determining their acceptability, as we do with our definitions of argument inquiry and warrant inquiry dialogues. Bentahar *et al.* classify their dialogues as merging persuasion with inquiry but we would argue that their dialogues are not persuasion as the agents are not aiming to convince the other to accept their position, but rather aim to arrive at the same outcome as would be achieved when reasoning with the union of the agents' knowledge.

We believe the only other similar works that consider soundness and completeness properties are [37,42,46]. Rather than consider a specific dialogue system, [37] defines different classes of protocol based on types of move relevance, and looks at completeness properties for these classes of protocol. As with [48], the notion of completeness considered in [37] is weaker than our notion of completeness in the sense that it does not consider arguments that can only be constructed jointly by two agents. [46] defines different agent programs for negotiation. If such an agent program is both exhaustive and deterministic then exactly one move is suggested by the program at a timepoint, making such a program generative and allowing consideration of soundness and completeness properties. As [46] deals with negotiation dialogues, however, their results are not comparable with ours. Prakken [42] presents a dialogue system for modelling persuasion dialogues, where he considers soundness and fairness properties of the system. However, Prakken is interested in showing that his definition of the outcome of the dialogue (based on a labelling of the moves made) does indeed produce the same outcome as the dialectical graph that is implicitly produced during the dialogue, rather than in comparing the outcome of his system with the outcome of another system.

It is interesting to note that, although Prakken's dialogue system models persuasion dialogues, the dialogues it produces appear very similar to our warrant inquiry dialogues, in that they both allow the exchange of arguments with the aim of jointly constructing a dialectical graph (although Prakken's system does not allow agents to jointly construct arguments). A main aim of the warrant inquiry protocol is to ensure that the dialogue stays on topic, which it achieves by only allowing the participants to assert arguments that will alter the dialogue tree; Prakken's protocol for liberal persuasion dialogues shares this aim, which it achieves by only allowing the participants to make moves that reply to a move that has previously been made. What makes one of these systems an inquiry dialogue system and the other a persuasion dialogue system is the strategical manoeuvring within the space of legal moves defined by the protocols. The exhaustive strategy was defined to ensure that cooperative agents participating in a warrant inquiry dialogue exchange *all* arguments that may have some bearing on the outcome of the dialogue, without any consideration of how an agent might manipulate the dialogue (e.g. by holding back certain knowledge) in order to convince the other to accept its position; this allows the agents to jointly arrive at some new knowledge that was not previously known be either agent alone (i.e. the status of the root argument given the union of the agents' beliefs), hence the term inquiry. When considering persuasion dialogues, it is assumed that agents are self-interested and each will make their choice of legal move in order to maximise the possibility of persuading the other to accept their position.

To conclude this discussion section we consider Dung's seminal piece of work on defining semantics for the acceptability of arguments [19]. Under Dung's semantics, one can construct an argument graph of all arguments under consideration and then assess the acceptability of arguments from this graph. As our protocol deals with dialectical trees, which are a special form of argument graph, we believe it would be possible to adapt our system so that it used Dung semantics rather than those of DeLP. The work we have presented in this article, whilst focussing on DeLP semantics, illustrates a general approach to jointly assessing the acceptability of an argument; the important point here is that our system ensures consideration of all arguments that may affect the acceptability of the argument in question. We believe that we could adapt our system to use the semantics and defeat relation of any argumentation system where the acceptability of an argument depends on the argument graph constructed around it, such as [1, 6, 9, 20, 36]. We intend to show this in future work.

8 Conclusions

We have presented a general framework for representing dialogues and given details of two specific protocols and a strategy for generating two subtypes of inquiry dialogue between two agents; together, this framework, the protocols and the strategy comprise our inquiry dialogue system. The argument inquiry dialogue allows two agents to share knowledge in order to jointly construct arguments for a specific claim that neither may construct from their own personal beliefs alone. The warrant inquiry dialogue allows two agents to share arguments in order to jointly construct a dialectical tree that neither may construct from their own personal beliefs alone, effectively allowing two agents to use García and Simari's Defeasible Logic Programming [25] (intended for internal reasoning by a single agent) to carry out *intra*-agent argumentation. As argument inquiry dialogues may be embedded within warrant inquiry dialogues, our system allows agents participating in a warrant inquiry dialogue to consider all arguments that can be constructed from the union of their beliefs when constructing the dialectical tree, something that is not possible in most other comparable systems (e.g. [32, 48]). The assumption-based system defined in [8] does allow joint construction of arguments; however, it assumes that each agent shares the same set of rules and so only the assumptions are distributed between the agents. The only system we are aware of that allows agents to jointly construct arguments without making any assumptions about the split of beliefs across the agents is [16]; however, they do not provide a precise protocol and strategy, nor do they provide soundness and completeness results for their system.

This system is intended for use in a cooperative, safety-critical domain (such as the medical domain) where we wish the results of the dialogue to be predetermined by the agents' beliefs and the protocol and strategy being used. Other groups have presented protocols capable of modelling inquiry dialogues (e.g. [32, 39]); however, none have provided the means to select exactly one legal move at each timepoint. We have addressed this problem by providing a strategy function that selects exactly one move from the set of legal moves returned by the protocol. We have proposed a benchmark against which to compare the outcome of our dialogues, being a single agent reasoning with the union of the participating agents' beliefs, and have shown that dialogues generated by our system are always sound and complete in relation to this benchmark; we have done this without imposing any restrictions on the division of beliefs between the agents. No other group has considered such properties of inquiry dialogues unless they have also made some assumption about the division of beliefs between the agents.

Acknowledgements First author funded by a studentship from Cancer Research UK. We are very grateful to the anonymous reviewers for their extremely helpful comments.

References

1. L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34(1-3):197–216, 2002.
2. L. Amgoud and N. Hameurlain. An argumentation-based approach for dialogue move selection. In *Third International Workshop on Argumentation in Multi-Agent Systems (ARGMAS 2006)*, pages 111–125, Hakodate, Japan, 8 - 12 May 2006.
3. L. Amgoud, N. Maudet, and S. Parsons. Arguments, dialogue and negotiation. In *Fourteenth European Conference on Artificial Intelligence (ECAI 2000)*, pages 338–342, Berlin, Germany, 2000. IOS Press.

4. L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Fourth International Conference on Multi-Agent Systems*, pages 31–38, Boston, USA, 2000. IEEE Press.
5. K. Atkinson and T. J. M. Bench-Capon. Argumentation and standards of proof. In *Eleventh International Conference on AI and Law (ICAIL 2007)*, pages 107–116, Palo Alto, CA, USA, 2007. ACM Press.
6. T. J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
7. T. J. M. Bench-Capon, T. Geldard, and P. H. Leng. A method for the computational modelling of dialectical argument with dialogue games. *Artificial Intelligence and Law*, 8:233–254, 2000.
8. J. Bentahar, R. Alam, and Z. Maamar. An argumentation-based protocol for conflict resolution. In *KR2008-workshop on Knowledge Representation for Agents and Multi-Agent Systems (KRAMAS 2008)*, pages 19–35, Sydney, Australia, 2008.
9. P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128:203–235, 2001.
10. F. Bex and H. Prakken. Reinterpreting arguments in dialogue: an application to evidential reasoning. In *Legal Knowledge and Information Systems. JURIX 2004: The Seventeenth Annual Conference*, pages 119–129. IOS Press, 2004.
11. E. Black. *A Generative Framework for Argumentation-Based Inquiry Dialogues*. PhD thesis, University College London, 2007.
12. E. Black and A. Hunter. A generative inquiry dialogue system. In *Sixth International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2007)*, pages 1010–1017, Honolulu, HI, USA, 2007.
13. E. Black and A. Hunter. Using enthymemes in an inquiry dialogue system. In *Seventh International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2008)*, pages 437 – 444, Estoril, Portugal, 2008.
14. R. F. Brena, J.-L. Aguirre, C. I. Chesñevar, E. H. Ramírez, and L. Garrido. Knowledge and information distribution leveraged by intelligent agents. *Knowl. Inf. Syst.*, 12(2):203–227, 2007.
15. C. I. Chesñevar, J. Dix, F. Stolzenburg, and G. R. Simari. Relating defeasible and normal logic programming through transformation properties. *Theoretical Computer Science*, 290(1):499–529, 2003.
16. I. de Almedia Móra, J. J. Alferes, and M. Schroeder. Argumentation and cooperation for distributed extended logic programs. In J. Dix and J. Lobo, editors, *Working notes of the workshop on Nonmonotonic Reasoning*, Trento, Italy, 1998.
17. F. Dignum, B. Dunin-Keplicz, and R. Verbrugge. Dialogue in team formation. In F. Dignum and M. Greaves, editors, *Issues in Agent Communication*, pages 264–280. Springer-Verlag, 2000.
18. F. Dignum and G. Vreeswijk. Towards a testbed for multi-party dialogues. In *AAMAS Int. Workshop on Agent Communication Languages and Conversation Policies*, pages 63–71, 2003.
19. 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.
20. P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170(2):114–159, 2006.
21. P. E. Dunne and T. J. M. Bench-Capon. Two party immediate response disputes: Properties and efficiency. *Artificial Intelligence*, 149(2):221–250, 2003.
22. P. E. Dunne and P. McBurney. Optimal utterances in dialogue protocols. In *Second International Conference on Autonomous Agents and Mutli-Agent Systems (AAMAS 2003)*, pages 608–615, New York, NY, USA, 2003. ACM Press.
23. J. Fox and S. Das. *Safe and Sound: Artificial Intelligence in Hazardous Applications*. AAAI Press and The MIT Press, 2000.
24. J. Fox, D. Glasspool, and J. Bury. Quantitative and qualitative approaches to reasoning under uncertainty in medical decision making. In *Eighth Conference on Artificial Intelligence in Medicine in Europe (AIME 2001)*, pages 272 – 282, Cascais, Portugal, 2001.
25. A. J. García and G. R. Simari. Defeasible logic programming an argumentative approach. *Theory and Practice of Logic Programming*, 4(1–2):95–138, 2004.
26. S. A. Gómez, C. I. Chesñevar, and G. R. Simari. Defeasible reasoning in web-based forms through argumentation. *International Journal of Information Technology and Decision Making*, 7(1):71–101, 2008.

27. M. P. González, C. I. Chesñevar, C. A. Collazos, and G. R. Simari. Modelling shared knowledge and shared knowledge awareness in cscl scenarios through automated argumentation systems. In J. M. Haake, S. F. Ochoa, and A. Cechich, editors, *CRIWG*, volume 4715 of *Lecture Notes in Computer Science*, pages 207–222. Springer, 2007.
28. T. F. Gordon. *The Pleadings Game; an Artificial Intelligence Model of Procedural Justice.* Kluwer Academinc Publishers, 1995.
29. D. Hitchcock, P. McBurney, and S. Parsons. A framework for deliberation dialogues. In H. V. H. *et al.*, editor, *Fourth Biennial Conference of the Ontario Society for the Study of Argumentation (OSSA 2001)*, Windsor, Ontario, Canada, 2001.
30. J. Hulstijn. *Dialogue Models for Inquiry and Transaction.* PhD thesis, Universiteit Twente, Enschede, The Netherlands, 2000.
31. A. Kakas, N. Maudet, and P. Moraitis. Layered strategies and protocols for argumentation-based agent interaction. In I. Rahwan, P. Moraitis, and C. Reed, editors, *First International Workshop on Argumentation in MultiAgent Systems (ARGMAS 2004)*, Lecture Notes in Artificial Intelligence (LNAI) 3366, pages 66–79, New York, 2004. Springer-Verlag.
32. P. McBurney and S. Parsons. Representing epistemic uncertainty by means of dialectical argumentation. *Annals of Mathematics and Artificial Intelligence*, 32(1-4):125–169, 2001.
33. P. McBurney and S. Parsons. Dialogue games in multi-agent systems. *Informal Logic*, 22(3):257–274, 2002.
34. P. McBurney and S. Parsons. Games that agents play: A formal framework for dialogues between autonomous agents. *Journal of Logic, Language and Information*, 11(3):315–334, 2002. Special issue on logic and games.
35. P. McBurney, R. van Eijk, S. Parsons, and L. Amgoud. A dialogue-game protocol for agent purchase negotiations. *Journal of Autonomous Agents and Multi-Agent Systems*, 7(3):235–273, 2003.
36. S. Modgil. An abstract theory of argumentation that accomodates defeasible reasoning about preferences. In *Ninth European Conference on Symbolic and Quantative Approaches to Reasoning with Uncertainty (ECSQARU 2007)*, pages 648–659, Hammamet, Tunisia, 2007.
37. S. Parsons, P. McBurney, E. Sklar, and M. Wooldridge. On the relevance of utterances in formal inter-agent dialogues. In *Sixth International Conference on Autonomous Agents and Mutli-Agent Systems (AAMAS 2007)*, pages 1002–1009, Honolulu, HI, USA, 2007.
38. S. Parsons, M. Wooldridge, and L. Amgoud. An analysis of formal inter-agent dialogues. In *First International Conference on Autonomous Agents and Mutli-Agent Systems (AAMAS 2002)*, pages 394– 401, Bologna, Italy, 2002. ACM Press.
39. S. Parsons, M. Wooldridge, and L. Amgoud. On the outcomes of formal inter-agent dialogues. In *Second International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2003)*, pages 616–623, Melbourne, Australia, 2003.
40. S. Parsons, M. Wooldridge, and L. Amgoud. Properties and complexity of some formal inter-agent dialogues. *Journal of Logic and Computation*, 13(3):347–376, 2003. Special issue on computational dialectics.
41. P. Pasquier, I. Rahwan, F. Dignum, and L. Sonenberg. Argumentation and persuasion in the cognitive coherence theory. In *First International Conference on Computational Models of Argument (COMMA 2006)*, pages 223–234, Liverpool, UK, 2006. IOS Press.
42. H. Prakken. Coherence and flexibility in dialogue games for argumentation. *Journal of Logic and Computation*, 15(6):1009–1040, 2005.
43. H. Prakken, C. Reed, and D. Walton. Dialogues about the burden of proof. In *Proceedings of the Tenth International Conference on AI and Law*, pages 115–124, New York, USA, 2005. ACM Press.
44. I. Rahwan, P. McBurney, and L. Sonenberg. Towards a theory of negotiation strategy (a preliminary report). In *Fifth Workshop on Game Theoretic and Decision Theoretic Agents (GTDT-2003)*, pages 73–80, 2003.
45. C. Reed. Dialogue frames in agent communications. In *Third International Conference on Multi-Agent Systems (ICMAS 1998)*, pages 246–253. IEE Press, 1998.
46. F. Sadri, F. Toni, and P. Torroni. Dialogues for negotiation: Agent varieties and dialogue sequences. In J.-J. Meyer and M. Tambe, editors, *Pre-proceedings of the Eighth International Workshop on Agent Theories, Architectures, and Languages (ATAL-2001)*, pages 69–84, 2001.
47. F. Sadri, F. Toni, and P. Torroni. Logic agents, dialogues and negotiation: an abductive approach. In M. Schroeder and K. Stathis, editors, *Symposium on Information Agents for E-Commerce (AISB 2001)*. AISB, 2001.

48. M. Thimm and G. Kern-Isberner. A distributed argumentation framework using defeasible logic programming. In *Second International Conference on Computational Models of Argument (COMMA 2008)*, pages 381–392, Toulouse, France, 2008. IOS Press.
49. D. N. Walton and E. C. W. Krabbe. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. SUNY Press, 1995.
50. M. Williams and A. Hunter. Harnessing ontologies for argument-based decision-making in breast cancer. In *19th IEEE International Conference on Tools with Artificial Intelligence - Vol.2 (ICTAI 2007)*, pages 254–261. IEEE, 2007.