Deliberative Argumentation for Service Provision in Smart Environments

Juan Carlos Nieves, Helena Lindgren

Department of Computing Science
Umeå University
SE-901 87, Umeå, Sweden
{jcnieves,helena}@cs.umu.se

Abstract. In this paper, we introduce an inquiry dialogue approach for supporting decision making in a smart environment setting. These inquiry dialogues have as topic either *agreement atoms* or *agreement rules*, which capture services in a smart environment. These services are provided and supported by three rational agents with different roles: *Environment Agent*, *Activity Agent* and *Coach Agent*. These three agents have different capabilities and represent different data sources; however, they have to collaborate in order to deliver services in a smart environment.
The knowledge base of each agent is captured by extended logic programs. Therefore, the construction of arguments is supported by the *Well-Founded Semantics* (WFS). The outcome of the inquiry dialogues is supported by well-known argumentation semantics.

1 Introduction

In this paper the cooperative layer of a multi-agent system is presented and exemplified by a scenario of an older adult who has needs and wishes for the support in the conduction of some daily activities in a smart home environment. The older adult also has *preferences* about *how* to be supported and *when*. However, the needs and wishes may change over a day and between days. Therefore, the agents need to find the *optimal actions* in the presence of partial and inconsistent information in a particular situation. Consequently, providing supportive services by synthesizing the relevant sources of data, possibly represented using a variety of formats, represents a fundamental challenge in the information management.

This challenge is addressed by a *formal dialogue-based approach* in a multi-agent setting. Formal argumentation dialogues have been intensively explored on the last years [1, 3, 6, 8, 10] in the community of formal argumentation theory. Most of these approaches have been suggested as general frameworks for setting up different kinds of dialogues. By having in mind these frameworks, we introduce an argumentation dialogue approach for supporting decision making in a smart environment setting in terms of *agreement rules*.

From the structure point of view, our argumentation dialogues follow the dialogue style suggested by [3]. However, since we support our specification language on default theories (*i.e.*, *extended logic programs*) and default theories can be mapped into Assumption-Based Argumentation (ABA) [4], our approach is close to ABA-dialogue inference [6]. Indeed, the inferences of our argumentation dialogues in terms of *x-committed agreement rules*

(Definition 9) are based on argumentation semantics as it is done on ABA-dialogues [6]. Moreover, we want to point out that both our arguments (Definition 3) and attack relations (Definition 4) can be regarded as particular definitions of arguments and attacks in ABA. In this sense, our dialogues can be seen as a specialization of ABA dialogues.

The article is organized as follows. Section 2 describes a multi-agent approach designed to deliver personalized services in a smart environment. Section 3 introduces our argumentation-based deliberative method; moreover, we show some relevant properties of our approach. In the last section, conclusions and future work are presented.

2 A Multi-Agent System for Providing Intelligent Services

In [9], we introduced a multi-agent approach designed to deliver personalized services in a smart environment. To this end, three agents were designed: the *Environment Agent*, *Activity Agent* and *Coach Agent*. In [9], these agents were motivated from an activity-theoretical point of view. In this section, these three agent are instantiated from the point of view of a particular intelligent infrastructure called *As-A-Pal*.

Kitchen As-A-Pal is a smart environment, which serves as a living laboratory environment for designing and developing a range of different knowledge-based applications intended to be deployed as part of a holistic approach to ambient assisted living. Kitchen As-A-Pal is augmented with sensors and passively tagged objects. The physical and ambient interfaces provide access to information and services.

Three agents have been partially implemented in the As-A-Pal environment: *Environment Agent*, *Activity Agent* and *Coach Agent*. These three agents have different roles and needs therefore to collaborate on providing support to the human actor in conducting activities in the As-A-Pal smart environment.

The **Environment Agent** is responsible for facilitating interaction in smart environments. Since the human actor is mobile, the context and conditions for interaction is changing with the human actor's and objects' physical position, the Environment Agent is expected to handle the dynamic availability of environmental resources. In an activity-theoretical perspective, the Environment Agent organizes and provides the *tools* for activity execution, e.g., the *mediators* when smart services are provided by the actor.

A *rule-based knowledge base* has been defined as the knowledge base of the Environment Agent.This knowledge base contains a set of predicates, which are turned grounded by considering readings from sensors embedded in the As-A-Pal environment.

The **Activity Agent** is responsible for supporting and enhancing the ongoing activities and the activities predicted to be performed in the near future. The Activity Agent recognizes activities (which have an objective) and actions (which are goal-oriented) performed in smart environments. The Activity Agent filters the available services to the ones that impact and enhance the ongoing activity.

Like the Environment Agent, the Activity Agent has a rule-based knowledge base. The Activity Agent has an extended knowledge base whose predicates are turned grounded by different activity recognition processes.

The **Coach Agent** enhances the human actor's ability to perform the activities perceived as important to the human, with assistance from the other agents. It is the Coach Agent's responsibility to guard the human actor's interests, so that the smart environment provides the desired support and services. It is responsible for maximizing the quality of activity execution, consequently, it needs to evaluate the performance, the human actor's satisfaction with her performance, and her satisfaction with how the ambient support is supporting her in activities. The quality of interaction service and satisfaction with activity performance can be obtained by continuously keeping track of the human actor's emotions and experiences.

Like the Environment and Activity Agents, Coach Agent has a rule-based knowledge base. The predicates of the extensional knowledge base of the Coach Agent are turned grounded by different emotions processes, *e.g.*, *emotions recognition* and questionnaires.

3 A Deliberative Argumentation Approach

In this section, an argumentation approach will be presented in order to manage agreements between the As-A-Pal architecture's agents. This argumentation approach will be basically an operational implementation of *deliberation dialogues*. A deliberation dialogue is characterized as a dialogue occurring when two or more parties attempt to agree on an action to be performed in some situation.

In a deliberation dialogue, all the participants use their knowledge to inform their contributions. A procedural approach for reaching agreements between the parties, which are taking part of a deliberation dialogue is by the considering *agreement rules* [9]. An agreement rule is basically *a consensus* in which the different participant of a deliberation dialogue agree.

3.1 Knowledge bases of the agents

We start defining the components of the knowledge base of each agent. To implement deliberation dialogues between the As-A-Pal agents, we provide each agent with a set of *agreement rules*. Agreement rules will be associated to specific *goals* related to *the services*, which As-A-Pal may provide. Hence, an agreement rule is defined as follows:

Definition 1. *An agreement rule*[1] *is of the form:* $\alpha : a_0 \leftarrow a_1, \ldots, a_j, not\ a_{j+1}, \ldots, not\ a_n$ *in which* $\alpha \in \mathbb{N}$*,* $a_i (0 \leq i \leq n)$ *is an atom such that for each* $a_i (1 \leq i \leq n)$ *either exists an agent* Ag *such that its logic-based knowledge base is* Σ *and* $a_i \in \mathcal{L}_{\Sigma}$[2] *or* $a_i \in \mathcal{L}_{AR}$ *such that* AR *is a set of agreement rules, and* $a_0 \neq a_i (1 \leq i \leq n)$*.*

Observing Definition 1, we can see that the atoms, which appear in the body of an agreement rule, *a.i.* $a_i (1 \leq i \leq n)$, are either beliefs, which belong to different agents, or atoms, which appear in other agreement rules. As we will see in Definition 2, the knowledge base of each agent is private. This means that an agent itself cannot know if an agreement rule holds true in a given moment. Hence, for knowing the trueness of agreement rules, the collaboration of all the agents whose knowledge is part of an given agreement rule is required.

The head of an agreement rule, *a.i.* a_o, will be associated to a particular belief which will be held by an As-A-Pal agent. For instance, this believe can be a service for the end

[1] This definition of an agreement rule extends our previous definition of agreement rules introduced in [9].

[2] By $\mathcal{L}_P$, we denote the set of atoms in the language of P.

user. This means that by considering the trueness of an agreement rule, different agents will agree on a particular service for a user.

According to Definition 1, each agreement rule has a natural number attached. This number will be used for attaching a preference level to each agreement rule. We will assume that smaller number capture high preferences. In the As-A-Pal smart environment, these preferences levels will initially be set up based on user-studies. However, we will expect that the As-A-Pal architecture will update these preference levels by considering the user-satisfiability, which is managed by the *Coach Agent*.

In the As-A-Pal architecture, each of the agents which belong to the As-A-Pal architecture is supported by a knowledge base, which is split mainly in three components.

Definition 2. *An As-A-Pal agent Ag is defined by the following structure $Ag = \langle \Sigma, AR, CS \rangle$ in which Σ is an extended logic program which denotes the knowledge base of agent Ag, AR is a set of agreement rules and CS is a set of* normal clauses *which is called a commitment store.*

We will assume that Σ and AR keep private information for each agent. In other words, other agents do not have access to Σ and AR. On the other hand, the commitment store of each agent keeps public information that other agents could access. AR and CS will be relevant structures for dealing with the dialogues between the As-A-Pal's agents.

In order to identify the atoms which only appear in agreement rules, let $Ag = \langle \Sigma, AR, CS \rangle$ and $\mathcal{L}_{Agreement} = \mathcal{L}_{AR} \setminus (\mathcal{L}_{\Sigma})$. The atoms which appears in $\mathcal{L}_{Agreement}$ are called *agreement-atoms*.

3.2 Arguments

Now that we have defined the structure of the knowledge base of each agent in the As-A-Pal architecture, we will move on how to come up with agreements between the different agents which take part of the As-A-Pal architecture. To this end, we will introduce a basic definition of an argument.

WFS provides a reasoning engine for inferring information from a logic programs. In the context of the As-A-Pal's agents, WFS will support the construction of arguments from the knowledge bases. The definition of an argument is as follows:

Definition 3 (Argument). *Let Σ be a logic program. $A_D = \langle S, c \rangle$ is an argument if the following conditions holds: 1.- $WFS(S) = \langle T, F \rangle$ and $c \in T$; 2.- $S \subseteq \Sigma$ such that S is a minimal set* (w.r.t. *set inclusion) of Σ satisfying* 1*; 3.- $WFS(S) = \langle T, F \rangle$ such that $\nexists a \in \mathcal{L}_P$ and $\{a, \neg a\} \subseteq T$. $\mathcal{A}_\Sigma$ denotes the set of arguments built from Σ.*

As we can observe in Definition 3, an argument $\langle S, c \rangle$ is composed by two components a *support* S and a *conclusion* a. An argumentation can be regarded as an explanation of a particular claim. We have implemented an argumentation engine which constructs arguments from a logic program according to Definition 3 [7][3]. Moreover, we have showed that the arguments constructed according to Definition 3 satisfy basic principles of consistency, see [7][4]. Now, let us define an attack relationship between arguments as follows:

[3] This argumentation engine can be download from: http://esteban-guerrero.tumblr.com/argengine

[4] The paper is on-line on www8.cs.umu.se/ esteban/doc/paperclean2.pdf.

Definition 4 (Attack relationship between arguments). *Let* $A = \langle S_A, g_A \rangle$*,* $B = \langle S_B, g_B \rangle$ *be two arguments,* $WFS(S_A) = \langle T_A, F_A \rangle$ *and* $WFS(S_B) = \langle T_B, F_B \rangle$*. We say that* A *attacks* B *if one of the following conditions holds: -* $a \in T_A$ *and* $\neg a \in T_B$*; and -* $a \in T_A$ *and* $a \in F_B$*.* $At(\mathcal{A}rg)$ *denotes the set of attack relations between the arguments which belong to the set of arguments* $\mathcal{A}rg$*.*

3.3 Inquiry Dialogues

Now that we have defined how the knowledge base of each agent is structured, our dialogue approach will be presented. The general idea of our approach is to apply *inquiry dialogues* in order to validate the trueness of either an agreement atom or an agreement rule. For instance, if an agreement atom holds true in an given state of the As-A-Pal architecture, then the given agreement atom holds the trueness of a particular belief in the whole As-A-Pal system.

Inspired by [3], we will consider a combination between *argument inquiry dialogues* and *warrant inquiry dialogues*. Hence, our inquiry dialogues are based on three basic moves: open - $\langle x, open, dialogue(\theta, \gamma) \rangle$; assert - $\langle x, assert, \langle S, a \rangle \rangle$ and close - $\langle x, close, dialogue(\theta, \gamma) \rangle$ in which x denotes an agent, $\langle S, a \rangle$ is an argument, $\theta \in \{wi, ai\}$, if $\theta = wi$ then γ is an agreement atom and if $\theta = ai$ then γ is an agreement rule. wi means "*warrant inquiry dialogue*" and ai means "*argument inquiry dialogue*". $\mathcal{M}$ denotes set of moves defined above. Let us observe that the format of these moves are not exactly the same as the ones introduced by [3]. Our moves are personalized in terms of agreement atoms and agreement rule. Moreover, the arguments asserted by assert-modes will be constructed according to Definition 3. According to Black and Hunter [3] a dialogue is defined as follows:

Definition 5. *A dialogue, denoted* D_r^t*, is a sequence of moves* $[m_r, \ldots, m_t]$ *involving a set of participants* $\mathcal{I}$*, 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.-* $Sender(m_s) \in \mathcal{I}$ $(r \leq s \leq t)$*; 3.-* $Sender(m_{s1}) \neq Sender(m_{s2})$ *such that* $(p \leq s1 < s2 \leq q)$*,* $(q - p) + 1 = |A|$[5] *and* $(r \leq p < q \leq t)$*. in which* $Sender : \mathcal{M} \longmapsto \mathcal{I}$ *is a function such that* $Sender(\langle Agent, Act, Content \rangle) = Agent$*.*

The only difference between Definition 5 and the original definition presented in [3] is that the set of participants is not restricted to two participants. In the As-A-Pal architecture, we have identified three main agents; hence, these agents will take part of the dialogue.

As in [3], a dialogue terminates whenever all the participants of a dialogue have made a close move, *w.r.t.* the topic of the dialogue, in a consecutive form. A dialogue allows us to manage multi nested dialogues; hence, the nested dialogues terminate before the outermost dialogue terminates.

Whenever an agent takes part of a dialogue, its commitment store will be updated. The update of the commitment stores of each agent is done as follows:

Definition 6. *Let* D_r^t *be the current dialogue and* $\mathcal{I}$ *be the set of participants. For all agent* $\in \mathcal{I}$*: 1.-* $CS^t_{agent} = \emptyset$ *iff* $t = 0$*; 2.-* $CS^t_{agent} = CS^{t-1}_{agent} \cup S$ *iff* $m_t = \langle agent, assert, \langle S, a \rangle \rangle$*, 3.-* $CS^t_{agent} = CS^{t-1}_{agent}$ *if the previous cases do not hold.*

[5] We are assuming that A has at least two participants.

According to Definition 6, the commitment store of each agent is updated whenever it performs an *assert* move; moreover, the information, which is added to the commitment store, is the support of the argument which is asserted. An important consequence of this update is that the information, which is added to the commitment store, is turned public; hence, the other agents which are taking part of the dialogue have access to this information. Therefore, this information can be used by other agents in order to construct their own arguments. It is worth mentioning that the commitment store of each agent basically is *an extended normal logic program*.

In order to deal with argument inquiry dialogues, a query store is attached to a dialogue. A query store is basically a set of atoms.

Definition 7. *Let D_r^t be the current dialogue and $\mathcal{I}$ be the set of participants such that $agent \in \mathcal{I}$. A query store QS_r is a finite set of positive literals such that:*

$$QS_r = \begin{cases} \mathcal{B}^+ \cup \mathcal{B}^- & \textit{iff } m_t = \langle agent, open, dialogue(ai, a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^-) \rangle, \\ \emptyset & \textit{otherwise.} \end{cases}$$

Let us observe that although the topic of an argument inquiry dialogue can have negative literal, these literals are updated into the query store as positive literals.

The protocol of an argument inquiry dialogue will be presented as a sequence of general steps. To this end, some notation is introduced: let $\mathcal{I}$ be the finite set of participants of a dialogue. We identify each agent from $\mathcal{I}$ by a natural number this means that $\mathcal{I} = \{1, \ldots, n\}$ and $i \in \mathcal{I}$ such that $i = \langle \Sigma^i, AR^i, CS^i \rangle$. Hence, an argument inquiry dialogue works as follows:

Step	**Argument Inquiry Dialogue**
1	One of the participant agents starts the argumentation inquiry dialogue with the move $\langle x, open, dialogue(ai, \gamma) \rangle$.
2	The query store QS is updated according to Definition 7.
3	Each participant agent i performs one of the following moves: 1. $\langle i, assert, \langle S, a \rangle \rangle$ if $\langle S, a \rangle \in \mathcal{A}_\Sigma, a \in QS$ in which $\Sigma = \Sigma^i \cup \bigcup_{j \in \mathcal{I} \text{ and } i \neq j} CS^j$ and none of the participants have asserted the argument $\langle S, a \rangle$ in the dialogue before. The commitment store of the agent i is updated according to Definition 6. 2. $\langle i, open, dialogue(ai, a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^-) \rangle$ if $a_0 \in QS, \alpha : a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^- \in AR^i$ and there is no previous open move in the dialogue with $a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^-$ as its topic. The dialogue go to Step 1 in a recursive way. 3. $\langle i, close, dialogue(ai, \gamma) \rangle$ if the agent i is unable to perform one of the previous steps.

There are formal conditions *w.r.t. well-formed argument inquiry dialogues*, which basically argue that all the moves *extend* an initial dialogue and all the participants of the dialogue have the opportunity to perform a move (see [3] for its definition).

In order to define the outcomes of dialogues, let us introduce the following notation: Given a dialogue D_r^t: $AR_{D_r^t} = \{\gamma | \langle x, open, dialogue(ai, \gamma) \rangle$ is a open-move that appears in $D_r^t\}$.

As we can observe, $AR_{D_r^t}$ contains basically the agreement rules which appear in the dialogue D_r^t. Considering $AR_{D_r^t}$, the outcome of an argument inquiry dialogue is defined as follows:

Definition 8. *Let D_r^t be a well-formed argument inquiry dialogue. The outcome of D_r^t is: $Outcome_{ai}(D_r^t) = \mathcal{A}_\Sigma$ such that $\Sigma = \bigcup_{i \in \mathcal{I}} CS^i \cup AR_{D_r^t}$.*

As we can see in Definition 8, the outcome of an argument inquiry dialogue is basically the set of arguments which we can build from the commitment stores of each of its participants and the agreement rules which appear in the dialogue D_r^t. Let us point out that $Outcome_{ai}(D_r^t)$ contains arguments which their conclusions can be agreement atoms. These arguments are the main outcomes of an argument inquiry dialogue since these arguments cannot be built by an agent itself.

Considering the arguments from $Outcome_{ai}(D_r^t)$ and the attack relation introduced by Definition 4, an argumentation framework *w.r.t.* an argument inquiry dialogue D_r^t is $AF_{D_r^t} = \langle Outcome_{ai}(D_r^t), At(Outcome_{ai}(D_r^t)) \rangle$.

An agreement rule γ will be called *x-committed* ($x \in \{s, p, c, g, i, ss, sg\}$) by a set of agents $\mathcal{I}$ as follows[6]:

Definition 9. *Let D_r^t be a well-formed argument inquiry dialogue involving a set of participant $\mathcal{I}$ and $m_r = \langle x, open, dialogue(ai, \gamma) \rangle$ such that $x \in \mathcal{I}$ and $\gamma = a_0 \leftarrow \mathcal{B}^+, \ not\ \mathcal{B}^-$ is an agreement rule. γ is s-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is a stable extension of $AF_{D_r^t}$. γ is p-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is a preferred extension of $AF_{D_r^t}$. γ is c-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is a complete extension of $AF_{D_r^t}$. γ is g-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is the grounded extension of $AF_{D_r^t}$. γ is i-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is the maximal (w.r.t. set inclusion) ideal extension of $AF_{D_r^t}$. γ is ss-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is a semi-stable extension of $AF_{D_r^t}$. γ is sg-committed by $\mathcal{I}$ w.r.t. D_r^t iff $\langle S, a_0 \rangle \in E$ and E is a stage extension of $AF_{D_r^t}$.*

It is straightforward to observe that by considering the subset relations between argumentation semantics, there are some relations that hold true between the different x-commitments ($x \in \{s, p, c, g, i, ss, sg\}$).

Proposition 1. *Let D_r^t be a well-formed argument inquiry dialogue involving a set of participant $\mathcal{I}$ and γ be an agreement rule. If γ is g-committed by $\mathcal{I}$ w.r.t. D_r^t then γ is {p,c}-committed by $\mathcal{I}$ w.r.t. D_r^t. If γ is s-committed by $\mathcal{I}$ w.r.t. D_r^t then γ is {ss,p,c,sg}-committed by $\mathcal{I}$ w.r.t. D_r^t. If γ is i-committed by $\mathcal{I}$ w.r.t. D_r^t then γ is {g,p,c}-committed by $\mathcal{I}$ w.r.t. D_r^t. If γ is ss-committed by $\mathcal{I}$ w.r.t. D_r^t then γ is {p,c}-committed by $\mathcal{I}$ w.r.t. D_r^t. If γ is p-committed by $\mathcal{I}$ w.r.t. D_r^t then γ is c-committed by $\mathcal{I}$ w.r.t. D_r^t.*

We observe that deciding whether an agreement rule is g-committed is decidable in polynomial time.

Proposition 2. *Let γ be an agreement rule, $\mathcal{I}$ be a set of agents and D_r^t be an argument inquiry dialogue. Deciding whether γ is g-committed agreement rule by $\mathcal{I}$ w.r.t. D_r^t is decidable in polynomial time .*

So far, we have introduced dialogues for committing agreement rules; however, it can be the case that a given agent knows a particular agreement atoms a_0 and wants to commit this given agreement atoms. This means to identify an agreement rule γ which has a_0 as its head atom and to validate weather γ is x-committed or not. To this end, we introduce *warrant inquiry dialogues*. Warrant inquiry dialogues will be introduced by a simple protocol. Like argument inquiry dialogues, we identify each agent from $\mathcal{I}$ by a natural number this means that $\mathcal{I} = 1, \ldots, n$ such that $i = \langle \Sigma^i, AR^i, CS^i \rangle$. Hence, a warrant inquiry dialogue works as follows:

[6] Due to lack of space, we omit the formal definition of the argumentation semantics. Please find their definitions in [2].

Step	**Warrant Inquiry Dialogue**
1	One of the participant agents starts the warrant inquiry dialogue with the move $\langle x, open, dialogue(wi, a_0)\rangle$.
2	Each participant agent i performs one of the following moves: 1. $\langle i, open, dialogue(ai, a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^-)\rangle$ if $\alpha : a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^- \in AR^i$ and there is no previous open move in the dialogue with $a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^-$ as its topic. 2. $\langle i, close, dialogue(wi, a_0)\rangle$ if the agent i is unable to perform the previous step.

Let us observe that a warrant inquiry dialogue basically allow the participant to suggest agreement rules which could infer the topic of the warrant inquiry dialogue. Hence, the outcome will be, like argument inquiry dialogues, a set of arguments and the commitment of the topic will depend on this set of arguments.

Definition 10. *Let D_r^t be a well-formed warrant inquiry dialogue involving a set of participant $\mathcal{I}$ and $m_r = \langle x, open, dialogue(wi, \gamma)\rangle$ such that $x \in \mathcal{I}$ and $\gamma = a_0$ is an agreement atom. $Outcome_{ai}(D_r^t) = \mathcal{A}_\Sigma$ such that $\Sigma = \bigcup_{i \in \mathcal{I}} CS^i \cup AR_{D_r^t}$.*

Since both warrant and argument inquiry dialogues induce an argumentation framework $AF_{D_r^T}$, let us abuse of Definition 9 and say that: given a well-formed warrant inquiry dialogue D_r^t and $x \in \{s, p, c, g, i, ss, sg\}$, a_0 is x-committed by $\mathcal{I}$ *w.r.t.* D_r^t iff $\delta = a_0 \leftarrow \mathcal{B}^+,\ not\ \mathcal{B}^- \in AR_{D_r^T}$ and δ is x-committed by $\mathcal{I}$ *w.r.t.* D_r^t. Due to lack of space, a whole example of the dialogues process is not presented.

4 Conclusions and future work

In the state of the art of formal argumentation dialogues, we can find different approaches for setting up different kinds of dialogues [1, 3, 6, 8, 10]. Since these approaches have been defined as general frameworks, they do not offer guidelines for splitting the knowledge base of each agent in order to identify the knowledge which is particularly managed at the level of dialogues. In this sense, we argue for identifying a particular vocabulary for capturing agreements. In our suggested approach, this particular vocabulary is materialized by the *agreement atoms*. We point out that all the commitments of our dialogues are expressed in terms of these agreement atoms (which also give place to agreement rules).

From the practical point of view, by identifying sets of agreement atoms (and their respective agreement rules), the design of dialogues in real application domains is guided by these agreement atoms and agreement rules.

From the technical point of view, the consideration of logic programs and logic programming semantics such as WFS has allowed us to have an efficient construction of arguments.Currently we are using implementations of WFS as the one suggested by XSB. However, our argumentation approach can take advantage of new approaches for inferring WFS in a setting of Big Data [11] in order to have a really faster argumentation builder. Moreover, since it is known that the grounded semantics is characterized by WFS [5], the implementation of a g-committed agreement solver can be implemented in a very efficient way. As we have observed, deciding whether an agreement rule is g-committed is decidable in polynomial time (Proposition 2). Part of our future work is to explore the characterization of WFS suggested by [11] in our argumentation setting.

References

1. L. Amgoud and H. Prade. Reaching agreement through argumentation: A possibilistic approach. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), Whistler, Canada, June 2-5, 2004*, pages 175–182. AAAI Press, 2004.
2. P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. *Knowledge Eng. Review*, 26(4):365–410, 2011.
3. E. Black and A. Hunter. An inquiry dialogue system. *Autonomous Agents and Multi-Agent Systems*, 19(2):173–209, 2009.
4. A. Bondarenko, P. M. Dung, R. A. Kowalski, and F. Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artificial Intelligence*, 93:63–101, 1997.
5. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
6. X. Fan and F. Toni. A general framework for sound assumption-based argumentation dialogues. *Artif. Intell.*, 216:20–54, 2014.
7. E. Guerrero, J. C. Nieves, and H. Lindgren. Arguing through the Well-founded Semantics: an Argumentation Engine. Submitted to a journal, 2014.
8. S. Kraus, K. P. Sycara, and A. Evenchik. Reaching agreements through argumentation: A logical model and implementation. *Artif. Intell.*, 104(1-2):1–69, 1998.
9. J. C. Nieves, E. Guerrero, J. Baskar, and H. Lindgren. Deliberative argumentation for smart environments. In *Principles and Practice of Multi-Agent Systems (PRIMA 2014)*, 2014.
10. H. Prakken. Formal systems for persuasion dialogue. *Knowledge Eng. Review*, 21(2):163–188, 2006.
11. I. Tachmazidis, G. Antoniou, and W. Faber. Efficient computation of the well-founded semantics over big data. *TPLP*, 14(4-5):445–459, 2014.