Argumentation Schemes for Collaborative Planning

Alice Toniolo[1], Timothy J. Norman[1], and Katia Sycara[2]

[1] Department of Computing Science, University of Aberdeen, Scotland, UK
{a.toniolo,t.j.norman}@abdn.ac.uk
[2] Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, US
katia@cs.cmu.edu

Abstract. We address the collaborative planning problem in teams of agents where partners have different objectives and norms. In this context, agreeing on the best course of action to adopt represents a significant challenge. Concurrent actions and causal plan-constraints may lead to conflicts of opinion on what to do. Moreover, individual norms can constrain agent behaviour. We propose an argumentation-based model for deliberative dialogues based on argumentation schemes. This model facilitates agreements about joint plans by enriching the quality of the dialogue through the exchange of relevant information about plan commitments and norms.

Keywords: Argumentation schemes, Practical reasoning, Planning

1 Introduction

In collaborative planning, distributed intelligent agents work in cooperation to create an agreed plan of activities to fulfil requirements that are unachievable by single individuals. For effective teamwork, mechanisms that enable agreements to be reached regarding a shared plan are essential. Identifying the best course of action is a complex task when agents of the team represent independent organisations with their own objectives, activities to perform and regulations to follow. Recent work has shown that the use of argumentation models for deliberative dialogues is a promising approach for generating consistent collaborative plans [1, 8]. Argumentation-based dialogues provide mechanisms to facilitate agreements through the exchange of information about collaborative tasks.

In this paper, we present a model of arguments, based on argumentation schemes, that can be used for deliberative dialogues among a team of agents in preparing a collaborative plan where agents have different objectives. This model deals with issues involved in planning, considering a wide set of conflicts among actions and norm constraints. Previous work has considered norms and plan-constraints in separate contexts for solving conflicts in practical reasoning [1, 8] and for norm adoption [6], but in this paper we bring together these issues within a single coherent model. Existing research on argumentation for practical reasoning has mainly focussed on collaboration among agents in the creation of

a common plan when agents have different beliefs or preferences. For example, in the model presented in [1] the arguments allow agents to explore what is possible and justified at the level of a single plan among a group of agents.

In contrast, our model explores what is possible when agents elaborate individual plans for achieving different objectives where only some activities require cooperation. When an agent requires collaboration for performing an action, our model allows agents to argue about possible conflicts with their individual plans such as concurrency, causality and legality of actions. Agents can also justify the need to adopt certain actions according to their plan rules and norms. Our model of arguments allows agent to exploit these conflicts, understand the reasons that have caused them and facilitate the establishment of agreements.

This paper is organised as follows. Section 2 introduces the deliberative dialogue. Section 3 describes a language for plans. In Section 4 we describe the structure of the argumentation framework followed by the different kinds of arguments. In Section 5 we discuss related work and our conclusions.

2   Deliberative Dialogue

In our scenario agents are heterogeneous and may have different objectives that are not necessarily known to others. Agents prepare individual plans and, then, engage in deliberative dialogue regarding collaborative actions. The debate commences when one agent informs the team about its intention to perform an action or requests an action to be performed by others. The proponent may seek collaboration for different reasons; for example, the agent needs to obtain a permission from others to perform an action. The proponent engages in a discussion with other agents describing the action with preconditions, effects and the goal that this action will help to achieve. The opponents, receiving more information about the action, can select new arguments according to conflicts with their commitments. The agents involved will then exchange arguments attacking others' opinions. The discussion ends when the parties agree on which course of action to perform, or when there are no other new arguments to exchange. If agents agree, the action is included in the agents' individual plans and in the shared plan. If agents disagree, the proponent withdraws the proposal and re-plans the action with an alternative that would not conflict with the new information gathered. The protocol of a dispute between two agents x, y about a proposal φ for an action is proposed in Fig.1 following similar protocols presented in the literature (e.g. [5]). In this paper we propose three schemes for arguments that agents can use during the discussion according to issues of practical reasoning such as concurrent actions, causality among actions and norms.

3   A Model of Plans

The language for plans that underpins our model is based on *situation calculus* [9]. In this section we introduce the planning domain using the foundational axioms [9] extended for temporal applications [7] and norms [4].
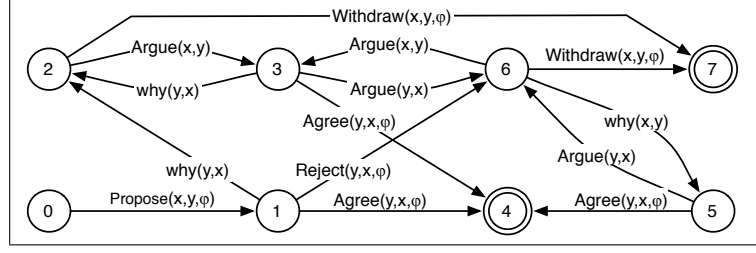
Fig. 1. Deliberative protocol for proponent x and opponent y about a proposal φ.

Planning Domain. The language includes the following sorts: $\mathcal{A}$ for actions, $\mathcal{S}$ for situations, $\mathcal{O}$ for domain objects and $\mathcal{T}$ for time stamps ranging over the integers. Lower case letters refer to variables, $a \in \mathcal{A}, s \in \mathcal{S}, v \in \mathcal{O}, t \in \mathcal{T}$ and upper case letters to constants. The operators are $\land, \lor, \supset, \equiv, \forall, \exists, =, \neq$ and where not specified free variables are universally quantified. We refer to the set of agents as $Agt \subset \mathcal{O}$ where $Agt = \{x, y, z, \dots\}$. A fluent $r(v_1, \dots, v_n, s)$ is a predicate that represents the feature of the world in situation s; $R(s)$ refers to the ground predicate r in s. $S_0 \in \mathcal{S}$ is the initial situation. Other elements are: a predicate $do(a, s)$ to indicate a situation resulting from performing action a in s; a relation $s < s'$ to order situations, where s occurs before s'; and a predicate $Poss(a, s)$ to indicate that action a can be performed in s. Intuitively, through instantiations of $Poss(a, s)$, situations are structured as a tree where the root is S_0, nodes are situations representing the world while arcs are possible actions that modify the state of the world. Each situation s is the result of performing a sequence of actions from root S_0, represented as $s = do(a_n, do(a_{n-1}, \dots do(a_1, S_0)))$ and abbreviated $s = do(\langle a_1; \dots; a_n \rangle, S_0)$. This tree represents the planning domain. A Basic Action Theory $\mathcal{D}$ [9] defines the axioms for the the planning domain. Our model is based on an extended version, $\mathcal{D}_{ext}$, for norms and durative actions.

Definition 1. *An Extended Action Theory is $\mathcal{D}_{ext} = \mathcal{D} \cup \Omega_d \cup \Omega_n$ where:*

- $\mathcal{D} = \Sigma \cup \mathcal{D}_{S_0} \cup \mathcal{D}_{una} \cup \mathcal{D}_{ss} \cup \mathcal{D}_{ap}$ *is a Basic Action Theory where:*
    - *Σ: set of domain independent axioms for situations.*
    - *$\mathcal{D}_{S_0}$: set of sentences representing the initial state of the world.*
    - *$\mathcal{D}_{una}$: set of unique name axioms for actions.*
    - *$\mathcal{D}_{ap}$: set of action precondition axioms in the form $\Pi_A(s) \equiv Poss(A, s)$.*
    - *$\mathcal{D}_{ss}$: set of successor state axioms in the form:*

$$Poss(a, s) \supset [R(do(a, s)) \equiv \gamma_R^+(a, s) \lor (R(s) \land \neg\gamma_R^-(a, s))]$$

   where γ_R^+, γ_R^- represent the add and delete conditions for fluent R.
- *Ω_d: set of axioms that handle actions with duration.*
- *Ω_n: set of axioms that handle norms.*

The set Ω_d includes axioms that ensure consistency for actions with duration $a_d \in \mathcal{A}$, following a work and notation of Pinto et al. [7]. Intuitively, action a_d is formed by: an instantaneous action $begin(a_d)$ for the beginning of a_d; an action

$end(a_d)$ for the end of a_d; and a fluent $exec(a_d, s)$ for the execution of a_d during
s. In this approach different actions can be concurrent as long as their beginning
and ending points do not coincide. The extension Ω_n (based on Demolombe et al.
[4]) includes axioms that define what an agent is obliged, permitted or forbidden
to do in terms of actions and features of the world under certain conditions. We
assume that everything is permitted when not explicitly prohibited and that all
individual norms are logically consistent. Active norms branch nodes or arcs of
the possible situation tree. Two operators, O and F, combined to a fluent express
obligations and prohibitions. A special fluent $occ(a, s)$, denoting an action which
occurs at the end of s, is used to represent norms that regulate actions.

Definition 2. *The following fluents define norm constraints: $\boldsymbol{O_R(s)}$ is an obligation fluent which holds in situation s iff a norm asserts that fluent R must hold
in s. $\boldsymbol{F_R(s)}$ is a prohibition fluent which holds in s iff a norm asserts that R must
not hold in s. $F_R(s) \wedge R(s)$ and $O_R(s) \wedge \neg R(s)$ are violations for these norms.
$\boldsymbol{O_{occ(A)}(s)}$ indicates that an action A must be the next action occurring after s.
$\boldsymbol{F_{occ(A)}(s)}$ indicates that action A is forbidden to occur as the next action. Violations are $Poss(A, s) \wedge [(\exists a).(a \neq A) \wedge O_{occ(a)}(s)]$ and $F_{occ(A)}(s) \wedge Poss(A, s)$.*

When an action a_d is executed in an interval s_1 to s_2, if a_d is forbidden beginning,
end and execution of a_d are forbidden for the whole interval, $F_{occ(A_d)}(s_1, s_2)$.
If a_d is obliged, the condition enforces a_d to start in situation s_1 and end in
s_2, $O_{occ(A_d)}(s_1, s_2)$. Each norm fluent is accompanied by a successor state axiom in the form of Def.1; e.g. $O_R(do(a, s)) \equiv [\xi^+_{O_R}(a, s) \vee (O_R(s) \wedge \neg \xi^-_{O_R}(a, s))]$,
where ξ^+, ξ^- are conditions of activation and expiration of norms. A situation s is legal when all the actions in the history of s are possible defined by
the predicate $legal(s) \equiv [s = S_0 \vee (\forall a, s').do(a, s') \leq s \supset Poss(a, s') \wedge \neg \eta(a, s')]$
and compliant with the norms ensured by the predicate $\eta(a, s)$ defined as:
$\eta(a, s) \equiv [(\exists r).r(s) \wedge F_r(s) \vee \neg r(s) \wedge O_r(s)] \vee [F_{occ(a)}(s) \vee (\exists a' \neq a) \wedge O_{occ(a')}(s)]$.

Agent Plans. An agent x maintains a description of the planning domain as a
subset of the extended action theory, $\mathcal{D}^x_{ext} \subseteq \mathcal{D}_{ext}$. Agent x's individual plan P^x
is a sequence of actions that identifies a path of possible and legal situations in
the situation tree that goes from the root to a situation where the overall goal
is satisfied. The goals of an agent x are a set of sentences ψ_k. The overall goal
Ψ is defined by their conjunction through a sentence $\Psi \equiv \psi_1 \wedge \cdots \wedge \psi_m$. Since
actions of an agent may not be known by other agents, situations represent only
internal states and they have no meaning outside that agent's subjective frame
of reference. We denote S^x as a situation on the path identified by x's plan.

Definition 3. *An individual plan P^x is a solution for a planning problem identified by a domain $\mathcal{D}^x_{ext}$ and a goal Ψ. $P^x = \langle A_1; \ldots; A_n \rangle$ is a sequence of actions
that identifies a path of legal situations from S^x_0 to a situation S^x_n where $\Psi(S^x_n)$
is satisfied. $\mathcal{D}^x_{ext} \models legal(S^x_n) \wedge \Psi(S^x_n)$ and $S^x_n = do(\langle A_1; \ldots; A_n \rangle, S^x_0)$.*

In order for agents to engage in dialogue about their plans, they rely on a common
time line where the path of situations identified by the plan is grounded to obtain
a temporal plan, $\vec{P}^x$. Two functions are used for an agent x to identify the plan

Table 1. Elements of $\vec{P}^x$.

- $\boldsymbol{P^x(\psi_k)}$: sequence of actions in P^x to achieve a goal ψ_k in S_k^x where $S_k^x \leq S_n^x$.
- $\boldsymbol{\mathcal{P}_{A_k} = \{R_1, \ldots, R_n\}}$: set of preconditions of action A_k. If A_k is in P^x, Π_{A_k} must hold in a situation S_{k-1}^x since $\Pi_{A_k}(S_{k-1}^x) \equiv Poss(A_k, S_{k-1}^x)$. $\mathcal{P}_{A_k}$ is the minimal set of R_i holding in S_{k-1}^x that satisfies the formula $\Pi_{A_k}(S_{k-1}^x)$.
- $\boldsymbol{\mathcal{E}_{A_k} = \{R_1, \ldots, R_m\}}$: set of effects of action A_k. A fluent $R_i(S_k^x)$ is an effect of A_k if the transition causes a change of its truth value $R_i(S_{k-1}^x) \equiv \neg R_i(S_k^x)$.
- $\boldsymbol{\langle \mathcal{P}_{A_k}, \mathcal{E}_{A_k} \rangle}$: tuple representing the execution of an action. When A_{dk} has duration, $\mathcal{P}_{A_{dk}} = \mathcal{P}_{begin(A_{dk})}$ and $\mathcal{E}_{A_{dk}} = \mathcal{E}_{end(A_{dk})}$ are related to the global action.
- $\boldsymbol{cLink(R, A_h, A_k)}$: causal link which represents a relation between A_h and A_k in P^x, with $S_k^x = do(A_k, S_{k-1}^x)$ and $S_h^x = do(A_h, S_{h-1}^x)$, where a precondition $R(S_{k-1}^x) \in \mathcal{P}_{A_k}$ not true in S_0, is provided by an effect $R(S_h^x) \in \mathcal{E}_{A_h}$ of a previous action A_h and $S_h^x \leq S_{k-1}^x$. $R(s^x)$ holds in all the situations $S_h^x \leq s^x \leq S_{k-1}^x$.
- $\boldsymbol{occBet(a, s_1^x, s_2^x)}$: an action a occurs in the path from s_1^x to s_2^x defined as $occBet(a, s_1^x, s_2^x) \equiv [(\exists s_p^x, s_q^x).s_q^x = do(a, s_p^x) \wedge s_1^x \leq s_p^x < s_q^x \leq s_2^x]$.
- $\boldsymbol{occBet_\mathcal{T}(a, t_1, t_2)}$:] represents the previous relation in terms of time obtained from $occBet_\mathcal{T}(a, t_1, t_2) \equiv [occBet(a, s_1^x, s_2^x) \wedge start(s_1^x) = t_1 \wedge start(s_2^x) = t_2]$.
- $\boldsymbol{occOver(a_d, s_1^x, s_2^x)}$: the execution of a_d overlaps the path s_1^x to s_2^x, where $occOver(a_d, s_1^x, s_2^x) \equiv [occBet(begin(a_d), s_1^x, s_2^x) \vee occBet(end(a_d), s_1^x, s_2^x)]$.
- $\boldsymbol{occOver_\mathcal{T}(a_d, t_1, t_2)}$: the corresponding time relation where $occOver_\mathcal{T}(a_d, t_1, t_2) \equiv [occBet_\mathcal{T}(begin(a_d), t_1, t_2) \vee occBet_\mathcal{T}(end(a_d), t_1, t_2)]$.
- **Norm Premises**: A norm is activated if the formula $\xi_{F/O_R}^+(A_p, S_p^x)$ holds in $S_p^x = do(A_p, S_{p-1}^x)$. The norm premises are identified by $\boldsymbol{\mathcal{N}_{Prem} = \{R_1, \ldots, R_n\}}$ as the minimal set of $R_i(S_{p-1}^x)$ which satisfy ξ^+ and $\boldsymbol{A_{dp}}$ causing the transition.

temporally grounded. A function $start(s^x) = t$ indicates the time t when the situation s^x begins ($start(S_0) = 0$). A function $sit(x, t)$ returns the ongoing situation in x's plan at time t. In fact, s^x is returned if $(\exists s^x, a).start(s^x) \leq t < start(do(a, s^x)) \supset sit(x, t) = s^x$. When action A_k in P^x causes the transition from S_{k-1}^x to $S_k^x = do(A_k, S_{k-1}^x)$ and the temporal axioms are $start(S_{k-1}^x) = T_{k-1}$ and $start(S_k^x) = T_k$, we refer to the temporal action A_k as $[T_k]A_k$. Hence, the plan $\vec{P}^x$ is composed by $\vec{P}^x = \langle [T_1]A_1; \ldots; [T_n]A_n \rangle$. Moreover, agents engaging in dialogue need to deal with new knowledge introduced by other team members. We use the K-accessibility relation in [9] where $K(s', s)$ indicates that in s an agent believes that the world is in situation s'. This relation associates situations that the agent considers indistinguishable. An agent x knows φ in s^x, $Know(\varphi, s^x) \stackrel{\text{def}}{=} \forall s_q^x (K(s_q^x, s^x) \supset \varphi(s_q^x))$, if φ holds in all situations K-accessible from s^x. Table 1 summarises further elements of plan that are used in the paper.

4 Argumentation

The arguments that agents use during the deliberative dialogue are structured as argumentation schemes [10]. These schemes are composed by premises and conclusions (identified by "$\Rightarrow$"). Each scheme also includes a set of critical questions CQs that provide a structured way of challenging an argument. Thus, argumen-

tation schemes are defeasible rules of inference in the norm-governed practical reasoning context that provide heuristics to guide the deliberative dialogue. We introduce general critical questions which identify the type of argument that can be formulated and then we specify what each argument can capture with more specific attacks (ATKs). The following locutions are used in the arguments:

Definition 4. *Given plan $\vec{P}^x$ of agent x:* $\boldsymbol{Perform(x, A_{dk}, T_{k1}, T_{k2})}$ *indicates that x intends to perform action A_{dk} through $[T_{k1}]begin(A_{dk})$, $[T_{k2}]end(A_{dk})$;* $\boldsymbol{Hold(x, R, T_k)}$ *indicates that for x a feature R holds at T_k, $Hold(x, R, T_k) \stackrel{def}{=} [R(S_k^x) \wedge start(S_k^x) = T_k]$ and $\neg Hold(x, R, T_k) \stackrel{def}{=} [\neg R(S_k^x) \wedge start(S_k^x) = T_k]$; $Hold(x, \{R_1, \ldots, R_n\}, T_k) \stackrel{def}{=} [Hold(x, R_1, T_k) \wedge \cdots \wedge Hold(x, R_n, T_k)]$ is used for a set of features;* $\boldsymbol{achieve(x, \psi_k)}$ *indicates that x intends to achieve goal ψ_k.*

In our description the debate is between agent x and agent y where x is informing y about the intention of performing action A_{dk}. Hence, the claim φ is $Perform(x, A_{dk}, T_{k1}, T_{k2})$. The individual plans are $\vec{P}^x$ and $\vec{P}^y$ where A_{dk} is in $\vec{P}^x$ and if agents agree, A_{dk} will be included in the set of agreed actions $\vec{P}^{xy}$.

We take as starting point an adaptation of Atkinson's argumentation scheme for practical reasoning [1]. This scheme involves a single action warranted by preconditions, where the effects allow the agent to achieve a goal which promotes or demotes a value. An argument Arg_I initiates the debate about the action, i.e. $Perform(x, A_{dk}, T_{k1}, T_{k2})$. The proponent specifies $\mathcal{P}_{A_{dk}}$, $\mathcal{E}_{A_{dk}}$ for A_{dk} in $\vec{P}^x$ and a partial goal ψ_k where $A_{dk} \in P^x(\psi_k)$. The initial argument Arg_I results:

$\boldsymbol{Arg_I}$: - Given preconditions $Hold(x, \mathcal{P}_{A_{dk}}, T_{k1})$
- $Perform(x, A_{dk}, T_{k1}, T_{k2})$
- brings about $Hold(x, \mathcal{E}_{A_{dk}}, T_{k2})$
- that will contribute to $achieve(x, \psi_k)$ [where $A_{dk} \in P^x(\psi_k)$],
$\Rightarrow$ therefore $Perform(x, A_{dk}, T_{k1}, T_{k2})$

The opponent y receiving Arg_I can select new arguments according to possible conflicts with commitments and norms following the critical questions:

- **CQ1:** Is the action possible according to concurrent actions in the plan?
- **CQ2:** Is the action possible according to causal plan constraints?
- **CQ3:** Is there any norm which regulates actions or states of the world?

A description of the argument structures is presented in the following sections.

Arguments for Concurrent Actions. Here, we explore the critical question *CQ1* concerned with concurrent actions. The argumentation scheme Arg_c is an extension of the initial argument where action A_{dk} is considered in the context of other actions already scheduled in an individual plan. Although in an individual plan actions can be concurrent only if starting or ending points do not coincide, actions from different plans can entirely overlap. Two concurrent actions A_k and A_h are executable if their preconditions hold and their effects are consistent. Furthermore, the effects of A_k should not contradict the preconditions of A_h and vice-versa. For a durative action A_{dk} there should not be direct interference on its preconditions and effects with other actions A_{dh} throughout its execution.

When agent x discusses with y about A_{dk} with specifications $\langle \mathcal{P}_{A_{dk}}, \mathcal{E}_{A_{dk}} \rangle$ (see Table 1), y needs to identify its internal situations that correspond to the execution of A_{dk}. Action A_{dk} is represented by $[T_{k1}]begin(A_{dk})$ and $[T_{k2}]end(A_{dk})$ in $\vec{P}^x$. The situations that delimit A_{dk} from y's point of view are $sit(y, T_{k1})$ and $sit(y, T_{k2})$. Suppose that in y's plan there is an action A_{dh} overlapping the execution of A_{dk}, $(\exists s^y).sit(y, T_{k1}) \leq s^y \leq sit(y, T_{k2})$ where $exec(A_h, s^y)$ holds. Action A_{dh} is formed by $[T_{h1}]begin(A_{dh}), [T_{h2}]end(A_{dh})$ in $\vec{P}^y$. The concurrency for y is $occOver(A_{dk}, S^y_{h1}, S^y_{h2})$ where $S^y_{h1-1} < S^y_{h1} \leq S^y_{h2-1} < S^y_{h2}$ and $S^y_{h1} = do(begin(A_{dh}), S^y_{h1-1}) \land S^y_{h2} = do(end(A_{dh}), S^y_{h2-1}) \land start(S^y_{h1}) = T_{h1} \land start(S^y_{h2}) = T_{h2}$. Action A_{dk} is possible for agent y only if there is no interference on preconditions and effects between A_{dk} and A_{dh} throughout their execution; i.e. $(\forall s^y, r).r \in \{\mathcal{P}_{A_{dk}} \cup \mathcal{E}_{A_{dk}}\} \land S^y_{h1} \leq s^y \leq S^y_{h2} \land occOver(A_{dk}, S^y_{h1}, S^y_{h2}) \supset Know(r, s^y) \lor \neg Know(\neg r, s^y) \lor \neg Know(r, s^y)$. The conflicting condition is $Know(\neg R, s^y)$. Action A_{dk} cannot be adopted by y when in its plan there is a concurrent action A_{dh} and: $i) A_{dk}$ has an effect $R \in \mathcal{E}_{A_{dk}}$ that contradicts an effect $\neg R \in \mathcal{E}_{A_{dh}}$ of A_{dh}; $ii) A_{dk}$ has an effect $R \in \mathcal{E}_{A_{dk}}$ that negates a precondition $\neg R \in \mathcal{P}_{A_{dh}}$ of A_{dh}; $iii) A_{dk}$ has a precondition $R \in \mathcal{P}_{A_{dk}}$ negated by an effect $\neg R \in \mathcal{E}_{A_{dh}}$ of A_{dh}; $iv) A_{dk}$ has a precondition $R \in \mathcal{P}_{A_{dk}}$ that contradicts a precondition $\neg R \in \mathcal{P}_{A_{dh}}$.

Argument Arg_c involves the two conflicting actions A_{dk}, A_{dh} with their specifications. The conflict is caused by A_{dk}, which $occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$, since $R \in \{\mathcal{P}_{A_{dk}} \cup \mathcal{E}_{A_{dk}}\}$ and $\neg R \in \{\mathcal{P}_{A_{dh}} \cup \mathcal{E}_{A_{dh}}\}$. Arg_c has the structure:

Arg_c: - Given preconditions $Hold(x, \mathcal{P}_{A_{dk}}, T_{k1})$,
  - $Perform(x, A_{dk}, T_{k1}, T_{k2})$,
  - brings about $Hold(x, \mathcal{E}_{A_{dk}}, T_{k2})$ [and $R \in \{\mathcal{P}_{A_{dk}} \cup \mathcal{E}_{A_{dk}}\}$],
  - and given preconditions $Hold(y, \mathcal{P}_{A_{dh}}, T_{h1})$,
  - $\neg Perform(y, A_{dh}, T_{h1}, T_{h2})$,
  - brings about $\neg Hold(y, \mathcal{E}_{A_{dh}}, T_{h2})$ [and $\neg R \in \{\mathcal{P}_{A_{dh}} \cup \mathcal{E}_{A_{dh}}\}$]
  - that will contribute $\neg achieve(y, \psi_h)$, [and $A_{dh} \in P^y(\psi_h)$]
  - and $occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$
  $\Rightarrow$ therefore $\neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \land Perform(y, A_{dh}, T_{h1}, T_{h2})$.

The conditions for conflicts between two concurrent actions lead an agent to formulate the following attacks against an action A_{dk} (formalised in Table 2):

  - **ATK1.1:** A_{dk} has an effect R that contradicts an effect $\neg R$ of action A_{dh} in my plan and their execution overlaps, therefore A_{dk} should not be performed.
  - **ATK1.2:** A_{dk} has an effect R that negates a precondition $\neg R$ of A_{dh} in my plan and their execution overlaps, therefore A_{dk} should not be performed.
  - **ATK1.3:** A_{dk} has a precondition R that is negated by an effect $\neg R$ of A_{dh} in my plan and their execution overlaps, therefore A_{dk} should not be performed.
  - **ATK1.4:** A_{dk} has a precondition R that contradicts a precondition $\neg R$ of action A_{dh} in my plan and their execution overlaps, hence A_{dk} should not be performed.

Arguments for Plan Constraints. In this section we define the argument for causal plan constraints Arg_p following $CQ2$. The preconditions that allow to schedule an action in the plan must be satisfied by performing other actions when not true at the outset. The causal relations among these actions must be protected when new actions are included in the plan for ensuring consistency. When an agent y receives a proposal $Perform(x, A_{dk}, T_{k1}, T_{k2})$ from agent x with

$\langle \mathcal{P}_{A_{dk}}, \mathcal{E}_{A_{dk}} \rangle$, it should verify that A_{dk} does not threat any causal relation in $\vec{P}^y$. Assume that A_{dk} has a precondition or an effect R, but $\neg R$ is a causal link between two actions A_{da} and A_{db} in $\vec{P}^y$. A conflict occurs when action A_{dk} is concurrent to the execution of the sequence of the two actions A_{da} and A_{db}. The link is represented as $cLink(\neg R, A_{da}, A_{db})$ where $\neg R \in \mathcal{E}_{A_{da}}$ and $\neg R \in \mathcal{P}_{A_{db}}$ (see Table 1). Agent y believes that $Know(\neg R, s^y)$ in the path $(\forall s^y) S_{a1}^y \leq s^y \leq S_{b2}^y$ from the starting of A_{da} to the end of A_{db}, where $S_{a1}^y = do(begin(A_a), S_{a1-1}^y)$ and $S_{b2}^y = do(end(A_b), S_{b2-1}^y)$. Action A_{dk} where $R \in \{\mathcal{P}_{A_{dk}} \cup \mathcal{E}_{A_{dk}}\}$ is a threat to the causal link if $occOver(A_{dk}, S_{a1}^y, S_{b2}^y)$ holds. Hence, argument Arg_p results:

$\boldsymbol{Arg_p}$: - Given preconditions $Hold(x, \mathcal{P}_{A_{dk}}, T_{k1})$,
  - $Perform(x, A_{dk}, T_{k1}, T_{k2})$,
  - brings about $Hold(x, \mathcal{E}_{A_{dk}}, T_{k2})$ [and $R \in \{\mathcal{P}_{A_{dk}} \cup \mathcal{E}_{A_{dk}}\}$],
  - Given $cLink(\neg R, A_{da}, A_{db})$
  - $\neg Perform(y, A_{da}, T_{a1}, T_{a2})$ and $\neg Perform(y, A_{db}, T_{b1}, T_{b2})$,
  - and $occOver_{\mathcal{T}}(A_{dk}, T_{a1}, T_{b2})$
  $\Rightarrow$ therefore $\neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{da}, T_{a1}, T_{a2}) \wedge$
      $Perform(y, A_{db}, T_{b1}, T_{b2})$

An opponent agent can use an attack A_{dk} if threat to the causal link is caused by its preconditions or effects. Agent x can also justify the adoption of an action with a similar argument. If x's plan contains a causal link from A_{dk} to $A_{dj} \in \vec{P}^x$, $cLink(Q, A_{dk}, A_{dj})$ such that $Q \in \mathcal{E}_{A_{dk}}, Q \in \mathcal{P}_{A_{dj}}$, then A_{dk} is necessary to perform A_{dj}. Therefore, agent x can formulate the arguments (see Table 2):

- **ATK2.1:** A precondition of action A_{dk} is a threat to a causal link between two action A_{da}, A_{db} in my plan, therefore A_{dk} should not be performed.
- **ATK2.2:** An effect of action A_{dk} is a threat to a causal link between two action A_{da}, A_{db} in my plan, therefore A_{dk} should not be performed.
- **ATK2.3:** The effects of action A_{dk} are fundamental preconditions for executing action A_{dj} in my plan, therefore A_{dk} should be performed.

Arguments for Norms. Here, we expand upon the critical question *CQ3* for norms. We consider norms as external regulations about what the agent is forbidden or obliged to do in terms of actions and states of the world. Agents' plans are internally norm-consistent enforced by the legality of situation S_n where the overall individual goal is achieved. Conflicts may arise when new information about states of the world or actions introduced by others cause inconsistencies. An agent x can assert that a feature R_k holds, $Hold(x, R_k, T_k)$. From agent y's point of view this is possible only if R_k is not forbidden to hold, $\neg Know(F_{R_k}, sit(y, T_k)) \wedge \neg Know(O_{\neg R_k}, sit(y, T_k))$. When an agent x claims to $Perform(x, A_{dk}, T_{k1}, T_{k2})$, y's norms allow x to perform A_{dk} only if the interval of the execution of A_{dk} does not overlap an interval where a norm forbids the performance of the action. If $F_{occ(A_{dk})}(S_1^y, S_2^y)$ and $OccOver(A_{dk}, S_1^y, S_2^y)$, where the interval of situations $sit(y, T_{k1})$ to $sit(y, T_{k2})$ overlaps S_1^y to S_2^y, then performing A_{dk} is forbidden. The argument for norms is based on the normative reasoning scheme proposed by Oren *et al.* [6]. The structure involves the active norm with conclusions about actions and features of the world and its premises as $Hold(y, \mathcal{N}_{Prem}, T_p)$ and $Perform(y, A_{dp}, T_{p1}, T_{p2})$ (see Table 1). Hence, the argument for norms for a prohibition Arg_n has the following structure:

$\boldsymbol{Arg_n}$: - Given premises $Hold(y, \mathcal{N}_{Prem}, T_p)$ and $Perform(y, A_{dp}, T_{p1}, T_{p2})$,
      - a norm $Forbids$ to
      - $Perform(x, A_{dk}, T_{k1}, T_{k2})/Hold(x, R, T_k)$
    $\Rightarrow$ therefore $\neg Perform(x, A_{dk}, T_{k1}, T_{k2})/\neg Hold(x, R, T_k)$

An opponent y can argue against a proponent x according to the instantiated norms. Furthermore, those parts in x's plan committed to maintain norm consistency can be justified with a similar argument. Agent x can claim $Hold(x, R_k, T_k)$ because of a norm that enforces $O_{R_k}(S_k^x)$ or $F_{\neg R_k}(S_k^x)$ where $T_k = start(S_k^x)$. A norm can enforce an agent x to $Perform(x, A_{dk}, T_{k1}, T_{k2})$. If action A_{dk} is obliged to occur between S_1^x and S_2^x where $start(S_1^x) = T_{k1}$ and $start(S_2^x) = T_{k2}$ then,

Table 2. Formalisation of Arguments Arg_c, Arg_p, Arg_n

CQ1-ATK1.1: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}), Perform(x, A_{dk}, T_{k1}, T_{k2}), Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}) \wedge$
$\boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dk}}}, Hold(y, \mathcal{P}_{A_{dh}}, T_{h1}), \neg Perform(y, A_{dh}, T_{h1}, T_{h2}), \neg Hold(y, \mathcal{E}_{A_{dh}}, T_{h2}) \wedge$
$\neg \boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dh}}}, \neg achieve(y, \psi_h), occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$
$\Rightarrow \neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{dh}, T_{h1}, T_{h2}) \rangle$

CQ1-ATK1.2: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}), Perform(x, A_{dk}, T_{k1}, T_{k2}), Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}) \wedge$
$\boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dk}}}, \neg Hold(y, \mathcal{P}_{A_{dh}}, T_{h1}) \wedge \neg \boldsymbol{R} \in \boldsymbol{\mathcal{P}_{A_{dh}}}, \neg Perform(y, A_{dh}, T_{h1}, T_{h2}),$
$\neg Hold(y, \mathcal{E}_{A_{dh}}, T_{h2}), \neg achieve(y, \psi_h), occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$
$\Rightarrow \neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{dh}, T_{h1}, T_{h2}) \rangle$

CQ1-ATK1.3: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}) \wedge \boldsymbol{R} \in \boldsymbol{\mathcal{P}_{A_{dk}}}, Perform(x, A_{dk}, T_{k1}, T_{k2}),$
$Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}), Hold(y, \mathcal{P}_{A_{dh}}, T_{h1}), \neg Perform(y, A_{dh}, T_{h1}, T_{h2}),$
$\neg Hold(y, \mathcal{E}_{A_{dh}}, T_{h2}) \wedge \neg \boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dh}}}, \neg achieve(y, \psi_h), occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$
$\Rightarrow \neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{dh}, T_{h1}, T_{h2}) \rangle$

CQ1-ATK1.4: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}) \wedge \boldsymbol{R} \in \boldsymbol{\mathcal{P}_{A_{dk}}}, Perform(x, A_{dk}, T_{k1}, T_{k2}),$
$Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}), \neg Hold(y, \mathcal{P}_{A_{dh}}, T_{h1}) \wedge \neg \boldsymbol{R} \in \boldsymbol{\mathcal{P}_{A_{dh}}}, \neg Perform(y, A_{dh}, T_{h1}, T_{h2}),$
$\neg Hold(y, \mathcal{E}_{A_{dh}}, T_{h2}), \neg achieve(y, \psi_h), occOver_{\mathcal{T}}(A_{dk}, T_{h1}, T_{h2})$
$\Rightarrow \neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{dh}, T_{h1}, T_{h2}) \rangle$

CQ2-ATK2.1: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}) \wedge \boldsymbol{R} \in \boldsymbol{\mathcal{P}_{A_{dk}}}, Perform(x, A_{dk}, T_{k1}, T_{k2}),$
$Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}), cLink(\neg \boldsymbol{R}, A_{da}, A_{db}), \neg Perform(y, A_{da}, T_{a1}, T_{a2}),$
$\neg Perform(y, A_{db}, T_{b1}, T_{b2}), occOver_{\mathcal{T}}(A_{dk}, T_{a1}, T_{b2}) \Rightarrow$
$\neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{da}, T_{a1}, T_{a2}) \wedge Perform(y, A_{db}, T_{b1}, T_{b2}) \rangle$

CQ2-ATK2.2: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}), Perform(x, A_{dk}, T_{k1}, T_{k2}), Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}) \wedge$
$\boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dk}}}, cLink(\neg \boldsymbol{R}, A_{da}, A_{db}), \neg Perform(y, A_{da}, T_{a1}, T_{a2}),$
$\neg Perform(y, A_{db}, T_{b1}, T_{b2}), occOver_{\mathcal{T}}(A_{dk}, T_{a1}, T_{b2}) \Rightarrow$
$\neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(y, A_{da}, T_{a1}, T_{a2}) \wedge Perform(y, A_{db}, T_{b1}, T_{b2}) \rangle$

CQ2-ATK2.3: $\langle Hold(x, \mathcal{P}_{A_{dk}}, T_{k1}), Perform(x, A_{dk}, T_{k1}, T_{k2}), Hold(x, \mathcal{E}_{A_{dk}}, T_{k2}) \wedge$
$\boldsymbol{R} \in \boldsymbol{\mathcal{E}_{A_{dk}}}, cLink(\boldsymbol{R}, A_{dk}, A_{dj}), Perform(x, A_{dk}, T_{k1}, T_{k2}), Perform(x, A_{dj}, T_{j1}, T_{j2}),$
$occOver_{\mathcal{T}}(A_{dk}, T_{k1}, T_{j2}) \Rightarrow Perform(x, A_{dk}, T_{k1}, T_{k2}) \wedge Perform(x, A_{dj}, T_{j1}, T_{j2}) \rangle$

CQ3-ATK3.1: $\langle Hold(y, \mathcal{N}_{Prem}, T_p) \wedge Perform(y, A_{dp}, T_{p1}, T_{p2}),$
$Forbids : Perform(x, A_{dk}, T_{k1}, T_{k2}) \Rightarrow \neg Perform(x, A_{dk}, T_{k1}, T_{k2}) \rangle$

CQ3-ATK3.2: $\langle Hold(y, \mathcal{N}_{Prem}, T_p) \wedge Perform(y, A_{dp}, T_{p1}, T_{p2}),$
$Forbids : Hold(x, R, T_k) \Rightarrow \neg Hold(x, R, T_k) \rangle$

CQ3-ATK3.3: $\langle Hold(x, \mathcal{N}_{Prem}, T_q) \wedge Perform(x, A_{dq}, T_{q1}, T_{q2}),$
$Obliges : Perform(x, A_{dk}, T_{k1}, T_{k2}) \Rightarrow Perform(x, A_{dk}, T_{k1}, T_{k2}) \rangle$

CQ3-ATK3.4: $\langle Hold(x, \mathcal{N}_{Prem}, T_q) \wedge Perform(x, A_{dq}, T_{q1}, T_{q2}),$
$Obliges : Hold(x, R, T_k) \Rightarrow Hold(x, R, T_k) \rangle$

$O_{occ(A_{dk})}(S_1^x, S_2^x)$, action A_{dk} must be performed between time T_{k1} and T_{k2}. The arguments for norms are identified as follows (formal structure in Table 2):

- **ATK3.1:** An active norm in my plan forbids an action A_{dk} to be performed, therefore A_{dk} should not be performed.
- **ATK3.2:** An active norm in my plan forbids a feature of the world R to hold, therefore R should not hold.
- **ATK3.3:** An active norm in my plan obliges an action A_{dk} to be performed, therefore A_{dk} should be performed.
- **ATK3.4:** An active norm in my plan obliges a feature of the world R to hold, therefore R should hold.

4.1   Example

In this section we illustrate some characteristics of our formal model using an example. Agent x represents a travel agency that targets a sustainable tourism through the collaboration with local transportation companies to create holiday tours. Agent x schedules a package to the mountains considering two destinations, C and D. C is the only possible because D is dangerous and an internal policy forbids to schedule trips towards unsafe destinations. The group of customers Gr leaves from location A by train Tr to a station B and, then, they will take bus Bs to reach C (See Fig.3). Travel agency x engages in a discussion with train company y for obtaining information about Tr from A to B. The initial proposal is $Perform(x, A_{d1}, 2, 8)$. Agent y is planning on the same period to do some maintenance on the railway path from A to B and it is obliged to divert the train Tr for B towards a location E in order to secure the railway path. Hence, the train company rejects the proposal. The agents will exchange few arguments following the protocol in Fig.1. At the end, since x cannot formulate any other argument it withdraws the action and re-plans avoiding the train from A to B. The arguments presented are formalised in Tab.4. The argument Arg_I explains the requirements of action A_{d1}. Using ATK2.3, x explains that the customers have to reach B for catching the bus from B to C. Action A_{d1} is justified with A_{d2} following the causal link $cLink(R_2, A_{d1}, A_{d2})$. Using ATK1.1 y explains that Tr leaving from A is diverted towards location E and it will not reach B at the scheduled time. Diverting Tr, A_{d4}, is necessary for the train company to secure the part of railway under maintenance, $\psi_2(S_4^y)$. The conflict occurs because A_{d1} is concurrent to A_{d4} where $occOver_\mathcal{T}(A_{d1}, 2, 7)$. Agent y justifies A_{d4} with ATK3.3 claiming that a norm obliges y to divert the train when railway works are scheduled, A_{d3}, and the path between A and B is a single track railway R_{13}.

5   Discussion and Conclusions

In this paper we have presented a model for arguments that contributes in deliberative dialogues, based on argumentation schemes for arguing about norms and actions in a multi-agent system. Argumentation for deliberative dialogues has been the topic of numerous research efforts over the last few years [1, 8]. Atkinson et al. [1] proposed an approach for practical reasoning based on argumentation

Table 3. Scenario of Example.

Agent x: Travel Agency

A_{d1}=Take(Gr,Tr,A,B) A_{d2}=Take(Gr,Bs,B,C)

Gr · Tr · Loc. A · Loc. B · Bs · Loc. C · Loc. D

Def. $R_1 = in(Gr, A)$
$R_2 = in(Gr, B)$
$R_3 = in(Gr, C)$
$R_4 = in(Gr, D)$
$R_5 = in(Tr, A)$
$R_6 = in(Tr, B)$
$R_7 = in(Bs, B)$
$R_8 = in(Bs, C)$
$R_9 = dest(Tr, B)$
$R_{10} = dest(Bs, C)$

Initial situation: $R_1(S_0^x), R_5(S_0^x), R_7(S_0^x), R_9(S_0^x), R_{10}(S_0^x)$
Norms: $F_{R_4}(S_0^x)$ **Goals**: $\Psi(s) = \psi_1(s)$ where $\psi_1 = R_3 \vee R_4$
Plan: $\vec{P}^x = \langle [2]begin(A_{d1}), [8]end(A_{d1}), [9]begin(A_{d2}), [12]end(A_{d2}) \rangle$
Situations: $S_0^x; S_1^x = do(begin(A_{d1}), S_0^x); S_2^x = do(end(A_{d1}), S_1^x);$
$S_3^x = do(begin(A_{d2}), S_2^x); S_4^x = do(end(A_{d2}), S_3^x)$
Time: $start(S_0^x) = 0, start(S_1^x) = 2, start(S_2^x) = 8, start(S_3^x) = 9, start(S_4^x) = 12$

Agent y: Train Company

A_{d3}=Work(A,B)

Loc. A · Loc. B · Tr · Loc. E

A_{d4}=Divert(Tr,A,B,E)

Def.
$R_{11} = altDest(Tr, E)$
$R_{12} = in(Tr, E)$
$R_{13} = singleTrack(A, B)$
$R_{14} = safeRail(A, B)$
$R_{15} = problems(A, B)$

Initial situation: $R_5(S_0^x), R_9(S_0^x), R_{11}(S_0^x), R_{13}(S_0^x), R_{15}(S_0^x)$
Norms: $O_{occ(A_{d4})}(S_2^y, S_3^y)$ **Goals**: $\Psi(s) = \psi_2(s)$ where $\psi_2 = R_{14}$
Plan: $\vec{P}^y = \langle [1]begin(A_{d3}), [2]begin(A_{d4}), [7]end(A_{d4}), [22]end(A_{d3}) \rangle$
Situations: $S_0^y; S_1^y = do(begin(A_{d3}), S_0^y); S_2^y = do(begin(A_{d4}), S_1^y);$
$S_3^y = do(end(A_{d4}), S_2^y); S_4^y = do(end(A_{d3}), S_3^y)$
Time: $start(S_0^y) = 0, start(S_1^y) = 1, start(S_2^y) = 2, start(S_3^y) = 7; start(S_4^y) = 22$

schemes, accompanied by a set of critical questions that allow agents to evaluate the outcomes on the basis of the social values highlighted by the arguments. More recent work [3] considers also temporal aspects regarding the consequences of performing an action on a subsequent action. Belesiotis et al. [2] have explored the use of situation calculus as a language to present arguments about a common plan in a multi-agent system. Existing research, however, does not adequately address the requirements of applications where agents are concerned with agreeing joint plans. It focuses on the choice of the best of a set of mutually exclusive actions to perform. In contrast we consider a team dialogue focussed upon what is the best course of action to adopt based on the integration of individual agent plans with different objectives. Furthermore, team members might have internal norms which guide the choice of the plan and they should be considered as part of the discussion about a course of action. Our aim is to construct a coherent model for referring to multi-agent plans which considers norm and plan-constraints.

In future research, the aim is to evaluate the strength of this model against the quality of the plan. We will also consider the integration of the model on a rigours formalised dialogue. In this paper, however, we have demonstrated through examples, how this model allows agents to clarify conflicts in different courses of action and facilitate the exchange of information about joint plans.

Table 4. Example of Argumentation-Based Dialogue

---

1. $Propose(x, y, Perform(x, A_{d1}, 2, 8))$ **2.** $Reject(y, x, Perform(x, A_{d1}, 2, 8))$
3. $Argue(x, y, \text{ArgI})$ - **ArgI**: $\langle Hold(x, \{R_1, R_5, R_9\}, 2), Perform(x, A_{d1}, 2, 8),$
$Hold(x, \{\neg R_1, \neg R_5, R_2, R_6\}, 8), achieve(x, \psi_1) \Rightarrow Perform(x, A_{d1}, 2, 8)\rangle$
4. $why(y, x, Perform(x, A_{d1}, 2, 8))$ **5.** $Argue(x, y, \text{ATK2.3})$ - **ATK2.3**:
$\langle Hold(x, \{R_1, R_5, R_9\}, 2) Perform(x, A_{d1}, 2, 6), Hold(x, \{\neg R_1, \neg R_5, R_2, R_6\}, 8),$
$cLink(R_2, A_{d1}, A_{d2}), Perform(y, A_{d1}, 2, 8), Perform(y, A_{d2}, 9, 12), occOver_{\mathcal{T}}(A_{d1}, 2, 12)$
$\Rightarrow Perform(y, A_{d1}, 2, 8) \wedge Perform(y, A_{d2}, 9, 12)\rangle$ **6.** $Argue(y, x, \text{ATK1.1})$ - **ATK1.1**:
$\langle Hold(x, \{R_1, R_5, R_9\}, 2), Perform(x, A_{d1}, 2, 8), Hold(x, \{\neg R_1, \neg R_5, R_2, R_6\}, 8),$
$Hold(y, \{R_5, R_9, R_{11}\}, 2), \neg Perform(y, A_{d4}, 2, 7), \neg Hold(y, \{\neg R_5, \neg R_6, R_{12}\}, 7),$
$\neg achieve(y, \psi_2), occOver_{\mathcal{T}}(A_{d1}, 2, 7) \Rightarrow \neg Perform(x, A_{d1}, 2, 8) \wedge Perform(y, A_{d4}, 2, 7)\rangle$
7. $why(y, x, Perform(y, A_{d4}, 2, 7))$ **8.** $Argue(y, x, \text{ATK3.3})$ - **ATK3.3**:
$\langle Hold(y, R_{13}, 1), Perform(y, A_{d3}, 1, 22), Obliges : Perform(x, A_{d4}, 2, 7) \Rightarrow$
$Perform(x, A_{d4}, 2, 7)\rangle$ **9.** $Withdraw(x, y, Perform(x, A_{d1}, 2, 8))$

References

1. Atkinson, K., Bench-Capon, T.: Practical reasoning as presumptive argumentation using action based alternating transition systems. Artificial Intelligence 171(10-15), 855–874 (2007)
2. Belesiotis, A., Rovatsos, M., Rahwan, I.: A generative dialogue system for arguing about plans in situation calculus. In: Argumentation in Multi-Agent Systems, Lecture Notes in Computer Science, vol. 6057, pp. 23–41. Springer Berlin (2010)
3. Bench-Capon, T., Atkinson, K.: Action-state semantics for practical reasoning. In: Proceeding of the Fall Symposium on the Uses of Computational Argument (2009)
4. Demolombe, R., Pozos-Parra, P.: The chisholm paradox and the situation calculus. In: Foundations of Intelligent Systems, Lecture Notes in Computer Science, vol. 3488, pp. 241–256. Springer Berlin / Heidelberg (2005)
5. McBurney, P., Hitchcock, D., Parsons, S.: The eightfold way of deliberation dialogue. International Journal of Intelligent Systems 22(1), 95–132 (January 2007)
6. Oren, N., Luck, M., Miles, S., Norman, T.J.: An argumentation inspired heuristic for resolving normative conflict. In: Proceedings of the Fifth Workshop on Coordination, Organizations, Institutions and Norms in Agent Systems (2008)
7. Pinto, J.A., Reiter, R.: Reasoning about time in the situation calculus. Annals of Mathematics and Artificial Intelligence 14, 251–268 (1995)
8. Rahwan, I., Amgoud, L.: An argumentation-based approach for practical reasoning. In: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems. pp. 347–354 (2007)
9. Reiter, R.: Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems. MIT Press (2001)
10. Walton, D.N.: Argumentation schemes for presumptive reasoning. Lawrence Erlbaum Associates (1996)