The semantics of behavior

Nico Roos[1]

Abstract. The BDI architecture is one of the most popular architectures for agents with symbolic reasoning capabilities. To formally define the notions of Beliefs, Desires and Intentions, different formal logics have been proposed in the literature. Although these proposals often refer to the work of Bratman [2], none, however, correctly capture the form of practical reasoning that Bratman describes. What is lacking, is a proper characterization of the agent's behavior. The formal logics proposed so far, do not allow for an adequate characterization of the refinement of behaviors that Bratman describes.

This paper focuses on describing an agent's behavior. The proposed behavioral descriptions allow for the specification of abstract behaviors, which can subsequently be refined. The approach enables a more accurate characterization of Bratman's BDI model.

1 Introduction

The Beliefs, Desires and Intentions (BDI) model was introduced by Bratman [2] as a basis for realizing practical reasoning. His ideas have subsequently been used in the specification of agent architectures [8, 3, 7, 6], agent programming language [16, 10, 23, 1, 5], and agent communication languages [15]. These applications of the BDI model have played an important role within the area of autonomous agents and multi-agent systems. For this reason, a clear understanding of the notions of beliefs, desires and intentions and their mutual connections is essential. Several formal logics have been proposed in order to provide this clarity [4, 9, 13, 12, 14, 17, 18, 21, 20, 22, 24]. Unfortunately, these logics do not always provide the clarity that we desire.

An important difference between the logics that have been proposed in order to describe the notions of beliefs, desires and intentions, and the work of Bratman [2], concerns the refinement of intentions. In Bratman's view, an intention is described by some abstract action such as: "going to Rome" instead of a state property: "being in Rome", which is the effect of the intended action. This abstract action is subsequently refined by (*i*) identifying options that can realize the intention, (*ii*) deliberating over these options within the context of the chosen intention; e.g., "going to Rome", and (*iii*) choosing a more refined intention; e.g., "going to Rome in the first week of May". Further refinements of the latter intention, over which the agent may deliberate could be: "going to Rome in the first week of May by train". These stepwise refinements of the intention realize the means-ends analysis.

In order to truly describe the refinement process proposed by Bratman, we need to be able to describe (*i*) an agent's (intended) behavior, and (*ii*) the refinement of abstract behavior specifications. Here a behavior is an (abstract) action the agent intends, desires or is going to execute. The approaches proposed in the literature do not enable us to describe the refinement of an abstract behavior.[2] Therefore, this paper will focus on the semantics of behaviors, the refinement of behaviors, and reasoning about behaviors. Since behaviors will be described by actions, dynamic logic is used as a starting point.

We start in the next section with a summary of dynamic logic. in Section 3, we discuss the semantics of a behavior specification. Section 4 discusses the specification of subsumption relations between actions. These relations will form the basis of the refinement process. Section 5 provides a proof theory for the proposed extensions of dynamic logic. The related work is briefly discussed in Section 6 and Section 7 concludes the paper.

2 Preliminaries

This paper makes use of dynamic logic in order to specify the semantics of behaviors. We therefore assume interpretations that are Kripke structures. An interpretation I contains a set of states S describing the possible states of the agent's environment, an interpretation function π, and several relations. The interpretation of atomic actions in the set A is described by the relation $R^A : A \to 2^{S\times S}$.

Dynamic logic allows us to formulate composite actions (plans) using regular expressions. The following operators are used to construct composite actions: an is-followed-by operator: ; , a non-deterministic choice operator: $+$, an iteration operator: * , and a test operator: ? . The composite action $a;a'$ denotes that a' is executed after a, the composite action $a+a'$ denotes that a non-deterministic choice is made between executing a and a', a^* denotes that a is executed 0 or more times; i.e., $a^* = \varepsilon + a + (a;a) + (a;a;a) + (a;a;a;a) + \ldots$ where ε is a special action representing the absence of an action, and φ? denotes a test whether the proposition φ holds. The set of all composite actions is denoted by A^*. We define an extended relation $R^{A^*} : A^* \to 2^{S\times S}$ interpreting composite actions in A^*, in the usual way. The possible and necessary effect of an action a described by the proposition φ, can be specified by the propositions $\langle a\rangle\varphi$ and $[a]\varphi$, respectively. The semantics of these propositions is given by:

- $(I,s) \models \langle a\rangle\varphi$ iff there is an $s' \in S$ such that $(s,s') \in R^{A^*}(a)$ and $(I,s') \models \varphi$.
- $(I,s) \models [a]\varphi$ iff for every $s' \in S$, if $(s,s') \in R^{A^*}(a)$, then $(I,s') \models \varphi$.

Descriptions of the effects of actions should hold in all state of an agent's environment, not just the current state. Therefore, descriptions of actions are often given in terms of axioms. Here, however, we prefer to use one set of propositions to describe all information. We will therefore use the modal operator $\Box$, to specify that something *always* holds. For instance: $\Box(\varphi \to (\langle a\rangle\top \wedge [a]\psi))$. Since an agent

[1] Department of Knowledge Engineering, Maastricht University, email: roos@maastrichtuniversity.nl

[2] The refinement of abstract behaviors is related to hierarchical planning, such as HTNs [19].

may, incorrectly, believe that action a always results in the effect ψ, we will not interpret $\Box$ as referring to all states in S, but to all states reachable from the current state given some sequence of actions. The latter choice makes it possible to use doxastic logic to describe an agent's possibly incorrect beliefs about the effect of actions. The semantics of $\Box$ denoting always and its dual $\Diamond$ denoting sometimes, is given by:

- $(I, s) \models \Box\kappa$ iff for every $s' \in S$, if $(s, s') \in \left(\bigcup_{a \in A} R^A(a)\right)^*$, then $(I, s') \models \kappa$.
- $(I, s) \models \Diamond\kappa$ iff for some $s' \in S$, $(s, s') \in \left(\bigcup_{a \in A} R^A(a)\right)^*$ and $(I, s') \models \kappa$.

We will use $\Sigma \subseteq \mathcal{L}$ to describe all available information. The entailment relation between the information $\Sigma \subseteq \mathcal{L}$ and a conclusion φ is given by:

$\Sigma \models \varphi$ iff for every interpretation I and for every state $s \in S$ of that interpretation, $(I, s) \models \Sigma$ implies $(I, s) \models \varphi$.

3 The semantics of behaviors

An agent that needs to realize an intention, must choose a behavior that realizes the intention. A behavior is the result of executing a composite action that causes a transition from the current state to one of the intended states. Dynamic logic can be used to identify possible composite actions that realize a transition to an intended state. However, dynamic logic cannot be used to specify which action is actually chosen since dynamic logic only allows for *what if* analysis. That is, the agent can only derive what may and will hold *if* a (composite) action a is chosen.

To describe an agent's behavior; i.e., the action a the agent *chooses* to execute, we introduce a special predicate $\mathsf{do}(a)$. We could interpret $\mathsf{do}(\cdot)$ as a predicate in first order logics; i.e., "$(I, s) \models \mathsf{do}(a)$ iff $\pi(a) \in \pi(s)(\mathsf{do})$". This interpretation does not specify the *transition* from state s to a next state s' caused by the choice of executing a. Moreover, the semantics does not enable us to evaluate relations between chosen actions, such as:

do('goto the station'; 'take the train to Rome') $\to \mathsf{do}$('go to Rome')

To formalize the transition specified by an action and to enable the description of relations between (composite) actions, which are important for describing Bratman's BDI model, we propose a different approach.

To express the semantics of the predicate do in terms of a transition to a next state, we introduce a relation over states: $R^{\mathsf{do}} \subseteq S \times S$. This relation specifies a transition of the (current) state s to a next state s'. This transition need not be unique because of uncertainty about the effects of actions, especially in case of abstract actions.

Note that the relation $R^{\mathsf{do}} \subseteq S \times S$ does not specify which action is executed, though the predicate $\mathsf{do}(a)$ does specify that the action a is executed. If the relation R^{do} were a function of the executed action, it would specify a transition for every action. However, we only need transitions that can be the result of the action a that is actually executed. To identify in the semantics which action is actually executed, we will make use of dynamic logic. First, however, we will look at the properties of the relation R^{do}.

The relation R^{do} introduces a *notion of time* in the semantics. Here, time progresses in discrete steps by the actions the agent is executing. Of course, the agent cannot travel back in time. Therefore, we require that R^{do} is an *acyclic* relation. R^{do} should also enable the choice for composite actions. Moreover, we require that each state has its own unique history. This implies that the relation R^{do} should represent a tree in the direction of the future. Finally, we do not require the tree to contain infinitely long paths toward the future. When representing an intention for instance, we do not care what happens after the intention has been realized. So the relation R^{do} must possess the following properties:

- *irreflexive*: $\forall s\, (s, s) \notin R^{\mathsf{do}}$
- *transitive*: $\forall s, t, u[\{(s, t), (t, u)\} \subseteq R^{\mathsf{do}} \to (s, u) \in R^{\mathsf{do}}]$
- *tree*: $\forall s, t, u[\{(t, s), (u, s)\} \subseteq R^{\mathsf{do}} \to [(t, u) \in R^{\mathsf{do}} \veebar (u, t) \in R^{\mathsf{do}}]]$

The relation R^{do} describes the actual, the believed, the desired or the intended behavior of an agent depending on the preceding modal operator. Multiple paths indicate that there is uncertainty about the behavior. We *assume* that the behavior specified by R^{do} is always caused by actions the agent is executing. To determine these actions, we will use the semantics of composite actions of dynamic logic. The semantics of a composite action a is specified by the relation $R^{A^*}(a) \subseteq S \times S$. The agent is executing the action a in state s if all possible behaviors (s, s') that may be realized according to relation R^{do} are behaviors that are possible according to the relation $R^{A^*}(a)$.

The reason for the following somewhat complex specification of the chosen action is because an abstract action may be realized by a sequence of actions. So, we need to be able to identify whether a sequence of actions realize an abstract action. To make this identification, we need to consider the paths to the future specified by R^{do}.

$(s_0, s_1, \ldots)$ is a path determined by R^{do} and the state s_0 iff

- for every index $i \geq 0$, $(s_i, s_{i+1}) \in R^{\mathsf{do}}$, and
- for every index $i \geq 0$ there is no $s' \in S$ such that $(s_i, s') \in R^{\mathsf{do}}$ and $(s', s_{i+1}) \in R^{\mathsf{do}}$.

Note that the second item guarantees that a path contains all intermediate states.

The paths defined by R^{do} enable us to specify the chosen actions of a behavior. An agent is doing an action a in state s if every path specifying a behavior starting in s contains a state s' that is a possible effect of the action a; i.e., $(s, s') \in R^{A^*}(a)$. If a is a composite action, then doing a should imply that the agent also does the sequence of atomic actions determined by the composite action a. Since the relation $R^{A^*} : A^* \to S \times S$ summarizes the effects of a composite action a, we need to identify the paths determined by the sequence of atomic actions underlying the composite action.

A path $(s_0, s_1, \ldots, s_n)$ determined by a composite action a and the state s_0, is recursively defined as:

- If a is an atomic action ($a \in A$) and $(s_0, s_1) \in R^A(a)$, then (s_0, s_1) is a path of a.
- If $a = b; c$, $(s_0, \ldots, s_k)$ is a path of b and $(s_k, \ldots, s_l)$ is a path of c, then $(s_0, \ldots, s_k, \ldots, s_l)$ is a path of a.
- If $a = b + c$, $(s_0, \ldots, s_k)$ is a path of b and $(s_0, s'_1, \ldots, s'_l)$ is a path of c, then both $(s_0, \ldots, s_k)$ and $(s_0, s'_1, \ldots, s'_l)$ are paths of a.
- If $a = b^*$ and $(s_0, \ldots, s_k)$ is a path of $(\varepsilon + b; b^*)$, then $(s_0, \ldots, s_k)$ is a path of a.
- If $a = \varepsilon$, then (s_0) is a path of a.
- If $a = \varphi?$ and $(I, s_0) \models \varphi$, then (s_0) is a path of a.

- Nothing else is a path of a.

The paths defined by R^{do} and by the actions, enable us to specify the chosen composite actions of a behavior.

Let $P(s)$ be the set of all paths determined by R^{do} and the state s. $(I, s) \models \mathsf{do}(a)$ iff $a \in A^*$ and for every $(s_0, s_1, \ldots) \in P(s)$ there is a path $(s_0, s'_1, \ldots, s'_n)$ of a such that for every s'_i there is an s_{j_i} in $(s_0, s_1, \ldots)$ such that $s'_i = s_{j_i}$ and $j_{i-1} < j_i$.

Figure 1 gives an illustration of semantics. The relation R^{do} contains the couples (s_0, s_1), (s_0, s_2) and (s_1, s_2). So, we have one path in P, namely: (s_0, s_1, s_2). Since $(s_0, s_1) \in R^A(a)$, $(s_1, s_2) \in R^A(b)$ and $(s_0, s_2) \in R^{A^*}(a; b)$, $\mathsf{do}(a)$ holds in state s_0, $\mathsf{do}(b)$ holds in state s_1, and $\mathsf{do}(a; b)$ also holds in state s_0, respectively. If we would ignore the path that realizes the composite action $a; b$, then $\mathsf{do}(a; b)$ still holds in state s_0 while $\mathsf{do}(a)$ need not hold in state s_0 and $\mathsf{do}(b)$ need not hold in state s_1.

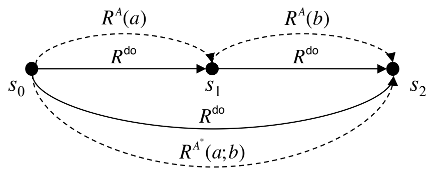

Figure 1. Behavior specification.

Since we distinguish abstract actions, it is useful to introduce a special predicate $\mathsf{dea}(a)$ that identifies a directly executable action a. Moreover, to distinguish atomic actions from composite actions, it is useful to introduce a special predicate $\mathsf{action}(a)$ that specifies whether the term a denotes an atomic action.

4 Relations between actions

In the previous section we have formalized the semantic relation between an agent's behavior and the specification of actions. Dynamic logic can be used to describe the effects of actions. It is, however, not possible to specify relations between (composite) actions in dynamic logic. So, we cannot specify that an action refines some abstract action. In the remainder of this section, we will consider two possibilities of specifying relations between actions. The first approach consists of explicitly specifying subsumption relations between (composite) actions. The second approach is based on the assumption that actions are fully characterized by their effects.

4.1 Subsumption relations between actions

The idea behind defining subsumption relations between actions is the following: if an action a that enables an agent to reach a subset of the states that can be reached by an action b, then the action b subsumes the action a. We denote this by $a \sqsubseteq b$ where a and b are (composite) actions. We also allow this relation to depend on the current state s. So, if we wish to denote that the subsumption relation always holds, we will have to specify; $\Box(a \sqsubseteq b)$.

The semantics of the subsumption relation over actions given a state s, is specified by:

Let $a, b \in A^*$ be two actions.
$(I, s) \models a \sqsubseteq b$ iff for every $t \in S$, if $(s, t) \in R^{A^*}(a)$, then $(s, t) \in R^{A^*}(b)$.

This definition does not imply that a subsumption relation holds in every state of the interpretation I. To illustrate this, consider a state s where φ holds. In the state s, the action a is subsumed by the action $(\varphi?; a) + (\neg\varphi?; b)$; i.e.: $(I, s) \models \varphi \wedge (a \sqsubseteq (\varphi?; a) + (\neg\varphi?; b))$, while this is not the case in a state t in which $\neg\varphi$ holds. When we require the subsumption relation to hold in every state of an interpretation, we can prove the following result:

Proposition 1 *Let $a, b \in A^*$ be two actions, and let I be an interpretation.*
For every $s \in S$, $(I, s) \models a \sqsubseteq b$ iff $R^{A^*}(a) \subseteq R^{A^*}(b)$.

Proof For every $s \in S$, $(I, s) \models a \sqsubseteq b$ iff for every $s \in S$, $\{(s, u) \mid (s, u) \in R^{A^*}(a)\} \subseteq \{(s, u) \mid (s, u) \in R^{A^*}(b)\}$ iff $R^{A^*}(a) \subseteq R^{A^*}(b)$. $\Box$

In order to reason about subsumption relations, it would be useful if we could describe a subsumption relation in terms of dynamic logic. It turns out that this is possible if we introduce for each state a proposition ξ that uniquely characterize the state.

Proposition 2 *Let $a, b \in A^*$ be two actions, let I be an interpretation and let s be a state of the interpretation.*
$(I, s) \models a \sqsubseteq b$ iff for every proposition ξ that uniquely characterizes a state, $(I, s) \models \langle a \rangle \xi$ implies $(I, s) \models \langle b \rangle \xi$.

Proof $(I, s) \models a \sqsubseteq b$ iff for every $t \in S$, if $(s, t) \in R^{A^*}(a)$, then $(s, t) \in R^{A^*}(b)$ iff for every $t \in S$, if $(s, t) \in R^{A^*}(a)$ and $(I, t) \models \xi$, then $(s, t) \in R^{A^*}(b)$ and $(I, t) \models \xi$ iff for every ξ, $(I, s) \models \langle a \rangle \xi$ implies $(I, s) \models \langle b \rangle \xi$. $\Box$

Although in general it will not be feasible to construct a proposition ξ, the above result will be useful in defining a proof theory.

Corrolary 1 *If $(I, s) \models a \sqsubseteq b$, then for every proposition φ, $(I, s) \models \langle a \rangle \varphi$ implies $(I, s) \models \langle b \rangle \varphi$.*

We can also prove the following property of $a \sqsubseteq b$.

Proposition 3 *For every interpretation state pair (I, s), if $(I, s) \models a \sqsubseteq b$, then for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$.*

Proof Suppose that $(I, s) \models [b]\varphi$ but $(I, s) \not\models [a]\varphi$. Then there is a $t \in S$ such that $(s, t) \in R^{A^*}(a)$ and $(I, t) \not\models \varphi$. So, $(s, s') \in R^{A^*}(a)$ and $(I, s') \not\models \varphi$. Since $(I, s) \models a \sqsubseteq b$, for every $t \in S$, if $(s, t) \in R^{A^*}(a)$, then $(s, t) \in R^{A^*}(b)$. Therefore, $(s, s') \in R^{A^*}(b)$. Since $(I, s) \models [b]\varphi$, for every $t \in S$, if $(s, t) \in R^{A^*}(b)$, then $(I, t) \models \varphi$. Therefore, $(I, s') \models \varphi$. Contradiction. $\Box$

4.2 Characterizing actions by their effects

The second approach is based on the *assumption* that *actions are completely characterized by their effects*. This assumption implies that action b subsumes action a if the effects of action a realize at least all effects of action b. So, the action of 'going to Rome' subsumes the action of 'going to Rome by train' because both actions have as effect 'being in Rome' while the latter action also has the effect of 'arriving by train'.

$(I, s) \models a \sqsubseteq b$ iff for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$.

An important question is whether we can always make this assumption. That is, whether we can always determine an interpretation satisfying the assumption.

Proposition 4 *For every interpretation I with a set of states S there exists an interpretation I' with states S' and there exists a surjective function $f : S \to S'$ such that the following two conditions hold:*

1. *Our* assumption *holds.*
2. *For every $s \in S$ and $\varphi \in \mathcal{L}$, if $(I, s) \models \varphi$, then $(I', f(s)) \models \varphi$.*

Proof We prove the proposition by constructing the interpretation I'. We ensure in the construction of I' that the assumption holds. So, we have to prove for any φ that $(I, s) \models \varphi$ implies $(I', f(s)) \models \varphi$.

Note that each state s' can be uniquely characterized by by a proposition $\xi_{s'}$ except for duplicate states that are indistinguishable from s'. Hence, there is proposition that uniquely characterizes the states $\{s' \mid (s, s') \in R^{A^*}(a)\}$, namely $\omega = \bigvee_{s' \mid (s,s') \in R^{A^*}(a)} \xi_{s'}$.

Determine for every state s, every relation $a \sqsubseteq b$ that must hold in s according to our assumption. Clearly, the relations $a \sqsubseteq b$ that must hold in s form a partial ordering. Let $a_1, \ldots, a_k$ be a linearization such that for very a_i there is no a_j such that $j < i$ and $a_i \sqsubseteq a_j$. We construct the interpretation I' and the function f starting from a_1. For every a_i, and for every a_j such that $a_i \sqsubseteq a_j$ and such that for no a_l such that $a_i \sqsubseteq a_l \sqsubseteq a_j$, execute the following construction step.

Let ω be the proposition characterizing the states $\{s' \mid (s, s') \in R^{A^*}(a_j)\}$. Then, $(I, s) \models [a_j]\omega$, and therefore $(I, s) \models [a_i]\omega$. Since $\omega = \xi_1 \vee \cdots \vee \xi_k$, and since every ξ_i completely characterizes a state, for every state $t \in \{s' \mid (s, s') \in R^{A^*}(a_i)\}$ there is a duplicate state $t' \in \{s' \mid (s, s') \in R^{A^*}(a_j)\}$. In the construction of the interpretation I', we replace each of these states t' by its duplicate state t. That is $f(t') = t$, and $(s, t') \in R^{A^*}(a_j)$ is replaced by $(s, t) \in R^{A^*}(a_i)$.

The construction guarantees that (1) our assumption holds, and (2) for every $s \in S$ and $\varphi \in \mathcal{L}$, if $(I, s) \models \varphi$, then $(I', f(s)) \models \varphi$.

The range of f is given by $S' = \{t \in S \mid s \in S, t = f(s)\}$. □

The assumption that actions are characterized by their effects, implies that, in order to distinguish two actions such as: 'walking to the train station' and 'taking the bus to the train station', the effect must be different. So, it is insufficient to only specify that the effect of both actions is: 'being at the train station'. Another requirement is that an agent's knowledge of the effect of actions must be complete.

We cannot specify in dynamic logic that all effects of an action have been specified. It is always possible to add another proposition $[a]\eta$. What we need, is a formalism which enables us to specify that 'being in Rome' is the *only* necessary effect of the action: 'going to Rome'. This is the only change in world caused by the action. The fact that my name did not change during the execution of the action: 'going to Rome', is not an effect of the action but a property of the world. So, we need a formalism to describe *all* necessary effects of an action. We will use the $\widehat{\ }$-operator, when applied to a modal *necessity* operator $\mathcal{N}$, to denote *all* effects of $\mathcal{N}$.

> $(I, s) \models \widehat{\mathcal{N}}\varphi$ iff $(I, s) \models \mathcal{N}\varphi$ and for every $\psi \in \mathcal{L}$, if $(I, s) \models \mathcal{N}\psi$, then $\models \varphi \to \psi$.

Note that this definition is related to the definition of *only-knowing* [11].

There always exists a proposition that characterize all the necessary effects of a modal operator. This proposition is unique in the sense that all propositions φ for which $\widehat{\mathcal{N}}\varphi$ holds, are equivalent.

Proposition 5 *Let $\mathcal{N}$ be a modal necessity-operator, let I be an interpretation and let s be a state of the interpretation.*

- *There exists a proposition $\varphi \in \mathcal{L}$ such that $(I, s) \models \widehat{\mathcal{N}}\varphi$.*
- *For every $\varphi, \psi \in \mathcal{L}$, if $(I, s) \models \widehat{\mathcal{N}}\varphi$ and $(I, s) \models \widehat{\mathcal{N}}\psi$, then $\models \varphi \leftrightarrow \psi$.*

Proof
- Let φ be the proposition characterizing the states $\{s' \mid (s, s') \in R^{A^*}(a_j)\}$. That is $\varphi = \xi_1 \vee \cdots \vee \xi_k$ and every ξ_i completely characterizes a state t_i with $\{t_1, \ldots, t_k\} = \{s' \mid (s, s') \in R^{\mathcal{N}}\}$. Suppose that $(I, s) \models \mathcal{N}\psi$. Then for every t_i, $(I, t_i) \models \psi$. Since ξ_i completely characterizes t_i, $\models \xi_i \to \psi$. So, $\models \varphi \to \psi$ since $\varphi = \xi_1 \vee \cdots \vee \xi_k$. Hence, $(I, s) \models \widehat{\mathcal{N}}\varphi$.
- Suppose that $(I, s) \models \widehat{\mathcal{N}}\varphi$ and $(I, s) \models \widehat{\mathcal{N}}\psi$. Since $(I, s) \models \widehat{\mathcal{N}}\varphi$, $(I, s) \models \mathcal{N}\varphi$ and therefore, $\models \psi \to \varphi$. Since $(I, s) \models \widehat{\mathcal{N}}\psi$, $(I, s) \models \mathcal{N}\psi$ and therefore, $\models \varphi \to \psi$. Hence, $\models \varphi \leftrightarrow \psi$.

Hence, the proposition holds. □

Applying the $\widehat{\ }$-operator to the necessity operator of dynamic logic using the above proposition, we get:

> $(I, s) \models \widehat{[a]}\varphi$ iff $(I, s) \models [a]\varphi$ and for every $\psi \in \mathcal{L}$, if $(I, s) \models [a]\psi$, then $\models \varphi \to \psi$.

Now we have introduced a modifier of modal-operator that enables us to denote all necessary effects of an action, we can address the problem of identifying subsumption relations between actions. The assumption that actions are completely characterized by their effects, is equivalent to a property that more useful to identify subsumption relations between actions.

Proposition 6 *The assumption:*

> $(I, s) \models a \sqsubseteq b$ *iff for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$.*

is equivalent to:

> $(I, s) \models a \sqsubseteq b$ *iff there is a $\varphi \in \mathcal{L}$ such that $(I, s) \models \widehat{[b]}\varphi$ and $(I, s) \models [a]\varphi$*

Proof It is sufficient to prove that "there is a $\varphi \in \mathcal{L}$ such that $(I, s) \models \widehat{[b]}\varphi$ and $(I, s) \models [a]\varphi$" is equivalent to "for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$".

($\Rightarrow$) Suppose there is a $\varphi \in \mathcal{L}$ such that $(I, s) \models \widehat{[b]}\varphi$ and $(I, s) \models [a]\varphi$. Moreover, suppose that for some $\psi \in \mathcal{L}$, $(I, s) \models [b]\psi$ and $(I, s) \not\models [a]\psi$. Since $(I, s) \models \widehat{[b]}\varphi$, $\models \varphi \to \psi$. Therefore, $(I, s) \models [a]\psi$. Contradiction. Hence, for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$.

($\Leftarrow$) Suppose that for every $\varphi \in \mathcal{L}$, if $(I, s) \models [b]\varphi$, then $(I, s) \models [a]\varphi$. According Proposition 5, there is a $\psi \in \mathcal{L}$ such that $(I, s) \models \widehat{[b]}\psi$. Since $(I, s) \models [b]\psi$, also $(I, s) \models [a]\psi$ must hold. □

5 Proof theory

In the previous two sections we have addressed the semantics of an agent's behavior and the semantics of the subsumption relation between composite actions. We did not yet address how to reason with behavior specifications and subsumption relations between actions. Reasoning with these notions will be the focus of this section.

We will present a semantic tableaux method based on the construction of a prefix-tableaux. We choose a prefix tableaux because it enables an easy integration with semantic tableaux methods developed for other logics such as doxastic logic, which we may use to describe an agents beliefs. In a prefix tableaux, we can view a prefix as a representation of a state.

We start by giving the tableaux rules for the subsumption relation. In the left rule, φ must be an existing proposition, while in the right rule ξ must be a *new* atomic proposition *uniquely characterizing a state*. ξ is the name we choose for the new proposition.

$$\frac{x : a \sqsubseteq b}{x : \neg\langle a \rangle \varphi \mid x : \langle b \rangle \varphi} \qquad \frac{x : \neg(a \sqsubseteq b)}{x : \langle a \rangle \xi, x : \neg\langle b \rangle \xi}$$

The two rules relating the subsumption relation between actions to dynamic logic are valid tableaux rules.

Lemma 1 *Let Γ be the set of proposition of a node of the semantic tableaux, and let Γ' and possibly Γ'' be the directly succeeding nodes that are the result of applying one of the above rules.*

Then Γ is satisfiable iff Γ' or Γ'' is satisfiable

Proof Corollary 1 and Proposition 2 imply the correctness of the lemma for the left and the right rule, respectively. □

The next tableaux rules address the derivation of a subsumption relation assuming that actions are completely characterized by their effects. In the right rule, φ must be an existing proposition, while in the left rule ξ must be a *new* atomic proposition.

$$\frac{x : a \sqsubseteq b}{x : [a]\xi, x : [\widehat{b}]\xi} \qquad \frac{x : \neg(a \sqsubseteq b)}{x : \neg[a]\varphi \mid x : \neg[\widehat{b}]\varphi}$$

The two rules relating the subsumption relation to the effects of the actions involved are valid tableaux rules.

Lemma 2 *Let Γ be a the set of proposition of a node of the semantic tableaux, and let Γ' and possibly Γ'' be the directly succeeding nodes that are the result of applying one to the above rules.*

Then Γ is satisfiable iff Γ' or Γ'' is satisfiable

Proof Proposition 6 implies the correctness of the lemma for both rules. □

The last tableaux rules address reasoning about behavior. The first two rules describe the relation between doing an action and the subsumption relation. The remaining ten rules describe how (not) doing a composite action implies (not) doing sub-actions.

$$\frac{x : a \sqsubseteq b, x : \mathsf{do}(a)}{x : \mathsf{do}(b)} \qquad \frac{x : a \sqsubseteq b, x : \neg\mathsf{do}(b)}{x : \neg\mathsf{do}(a)}$$

$$\frac{x : \mathsf{do}(a)}{x : \langle a \rangle \top} \qquad \frac{x : \mathsf{do}(\varphi?)}{x : \varphi}$$

$$\frac{x : \mathsf{do}(\varepsilon)}{x : \top} \qquad \frac{x : \neg\mathsf{do}(\varepsilon)}{x : \bot}$$

$$\frac{x : \mathsf{do}(a; b)}{x : \mathsf{do}(a), x : \langle a \rangle \mathsf{do}(b)} \qquad \frac{x : \neg\mathsf{do}(a; b)}{x : \neg\mathsf{do}(a) \mid x : [a]\neg\mathsf{do}(b)}$$

$$\frac{x : \mathsf{do}(a + b)}{x : \mathsf{do}(a) \mid x : \mathsf{do}(b)} \qquad \frac{x : \neg\mathsf{do}(a + b)}{x : \neg\mathsf{do}(a), x : \neg\mathsf{do}(b)}$$

$$\frac{x : \mathsf{do}(a^*)}{x : \top \mid x : \mathsf{do}(a; a^*)} \qquad \frac{x : \neg\mathsf{do}(a^*)}{x : \bot}$$

The twelve rules concerning the behavioral predicate $\mathsf{do}(\cdot)$ are valid tableaux rules.

Lemma 3 *Let Γ be a the set of proposition of a node of the semantic tableaux, and let Γ' and possibly Γ'' be the directly succeeding nodes that are the result of applying one to the above rules.*

Then Γ is satisfiable iff Γ' or Γ'' is satisfiable

Proof *The first two rules*: Every path for a is a path for b. Therefore, if every path of R^{do} covers a path of a, then every path of R^{do} also covers a path of b.
The second two rules: One can only do an action a if there is at least one path for a. One can only do a test for φ successfully if φ actually holds.
The third two rules: Doing nothing is always possible.
The fourth two rules: Since $\mathsf{do}(a; b)$ holds if every path of R^{do} covers a path of $a; b$, the results immediately follow from the definition of the paths for $a; b$.
The fifth two rules: Since $\mathsf{do}(a + b)$ holds if every path of R^{do} covers a path of $a + b$, the results immediately follow from the definition of the paths for $a + b$.
The last two rules: Since $a^* = \varepsilon + a; a^*$, the results follow. □

The three lemmas together with similar results for the semantic tableaux method for dynamic logic and possibly other logics such as doxastic logic, enable us to prove the correctness and completeness of a semantic tableaux method.

Proposition 7 *The root of the tableaux is satisfiable iff there is a branch starting from the root and all nodes of this branch are satisfiable.*

Proof We prove the proposition using induction on the depth of the tableaux. It suffices to prove that there is a node of some branch at depth $i + 1$ with $i \geq 0$ that is satisfiable iff its parent at depth i is satisfiable. For the rules listed above, Lemmas 1, 2 and 3 imply the desired result. For tableaux rules of propositional logic, dynamic logic, doxastic logic, etc., there exist similar lemmas implying the desired results the corresponding rules. □

If a semantic tableaux is closed, then the proposition implies that the root is not satisfiable. If, however, the tableaux is open, we must show that the root is satisfiable. We do this by constructing an interpretation using the leaf of an open branch.

Proposition 8 *If there is a leaf of a branch is open, then there exists an interpretation I satisfying the leaf.*

Proof We use the leaf to construct an interpretation. For propositions belonging to propositional logic, dynamic logic, doxastic logic, etc., we use the standard construction process. For the subsumption relation between actions: $a \sqsubseteq b$, we can use the construction used in the proof of Proposition 4. For propositions describing atomic behaviors: $\mathsf{do}(a)$ with $a \in A$, if $(I, s) \models \mathsf{do}(a)$ must hold, we add (s, s') to R^{do} for every $(s, s') \in R^A(a)$. The resulting interpretation can be shown to satisfy the leaf of the open branch. □

The above two propositions enable us to prove the correctness and completeness of the semantic tableaux method.

Theorem 1 *The root of the semantic tableaux is satisfiable iff the semantic tableaux is open.*

Proof The theorem follows from Propositions 7 and 8. □

6 Related work

This section summarizes some of the main approaches to specify the semantics of Beliefs, Desires and Intentions and discusses how they deal with behaviors. Most semantics are based on temporal logics such as: LTL [4], CTL* [17, 18, 20, 21, 22, 24] and ATL [14]. The Observation-based BDI logic [22] is closely related on LTL. These temporal logics associate the execution of actions with the transition between discrete time points.

In LTL based logics, the linear timelines represent possible behaviors of the agent. In these logics, Bratman's refinement of a behavior can be viewed as selecting proper subset of a set of behaviors. However, behaviors are sequences of atomic directly executable actions. No abstract behaviors can be represented.

In CTL* based logics, trees toward the future are used to describe behaviors. A tree represents uncertainty about the effect of an action and different possible actions an agent can choose. Each path to the future in a tree represents a possible behavior. The refinement of a set of behaviors is realized by introducing duplicate trees in which some branches are eliminated. Also in these logics no abstract behaviors can be represented.

A rational agent should only select behaviors it considers to be possible. In LTL and CTL* based logics, the agent's beliefs specify all behaviors the agent considers possible. Therefore, to be rational the agent should only select subsets of the behaviors it believes to be possible. This is realized by the requirements of realism and strong realism in LTL and CTL* based logics, respectively.

In LTL based logics, realism has the following odd consequence. When an agent believes that it will rain tomorrow, independent of any action it chooses to execute, the (believed) fact that it will rain tomorrow will become the agent's desire and intention.

In CTL* based logics, strong realism has a different odd consequence because behaviors are selected by by introducing duplicate trees in which some branches are eliminated. When an agent at current time-point intends *exactly one* action, for instance switching off the light, then this action will be inevitable, and therefore the agent will desire and will believe that it is inevitable that it will switch off the light. To avoid this problem all actions and behaviors should be optional, so we cannot specify that the agent intends of execute one action at current time-point.

The approach presented in this paper does not consider behaviors describing what an agent will do for the rest of its life. Therefore, the agent cannot select a subset of the behaviors it considers possible. Of course, the agent should still behave rationally. The requirements proposed in [12] could be used to ensure rationality.

Dynamic logic has also been used to specify the semantics of Beliefs, Desires and Intentions [13, 12]. The execution of behaviors is not addressed. The focus is on giving a psychological plausible description of practical reasoning with an emphasis on the requirements choosing a desire to become a intention. The representation of abstract behaviors is not addressed.

7 Conclusion

This paper addressed the semantics of an agent's behavior. Especially, the relation between abstract behavior and more specific behaviors was formalized. These aspects are important in order to give proper description of the refinement process described in Bratman's BDI model. Beside a discussion of the semantic relations and properties between behaviors, also a proof theory is given. Soundness and completeness of the proof theory is proved.

Future work will address application of the here presented model of behaviors in the description of an agent beliefs, desires and intentions. Moreover, the extension to multiple agents will be investigated. Finally, an implementation based on the specification is intended.

REFERENCES

[1] Davide Ancona and Viviana Mascardi, 'Coo-BDI: Extending the BDI model with cooperativity', in *DALT*, volume 2990 of *Lecture Notes in Computer Science*, pp. 109–134. Springer, (2003).

[2] Michael E. Bratman, *Intention, Plans, and Practical Reason*, Harvard University Press, Cambridge, MA, 1987.

[3] Michael E. Bratman, David J. Israel, and Martha E. Pollack, 'Plans and resource-bounded practical reasoning', *Computational Intelligence*, **4**, 349–355, (1988).

[4] Philip R. Cohen and Hector J. Levesque, 'Intention is choice with commitment', *Artificial Intelligence*, **42**(2-3), 213–261, (1990).

[5] Louise A. Dennis, Berndt Farwer, Rafael H. Bordini, Michael Fisher, and Michael Wooldridge, 'A common semantic basis for BDI languages', in *PROMAS*, volume 4908 of *Lecture Notes in Computer Science*, pp. 124–139. Springer, (2007).

[6] Mark d'Inverno, Michael Luck, Michael P. Georgeff, David Kinny, and Michael Wooldridge, 'The dmars architecture: A specification of the distributed multi-agent reasoning system', *Autonomous Agents and Multi-Agent Systems*, **9**(1-2), 5–53, (2004).

[7] Michael P. Georgeff and Franois Flix Ingrand, 'Decision-making in an embedded reasoning system', in *IJCAI*, (1989).

[8] Michael P. Georgeff and Amy L. Lansky, 'Reactive reasoning and planning', in *AAAI*, pp. 677–682, (1987).

[9] Guido Governatori, Vineet Padmanbhan, and Abdul Sattar, 'On fibring semantics for BDI logics', in *Logics in computer science*, pp. 198–210. Springer Verlag, (2002).

[10] Koen V. Hindriks, Frank S. de Boer, Wiebe van der Hoek, and John-Jules Ch. Meyer, 'Agent programming in 3APL', *Autonomous Agents and Multi-Agent Systems*, **2**(4), 357–401, (1999).

[11] Hector J. Levesque, 'All I know: A study in autoepistemic logic', *Artificial Intelligence*, **42**, 263–309, (1990).

[12] J.-J. Ch. Meyer, 'Intelligent agents: Issues and logics', in *Logics for Emerging Applications of Databases*, pp. 131–165. Springer, (2003).

[13] John-Jules Ch. Meyer Meyer, Wiebe van der Hoek, and Bernd van Linder, 'A logical approach to the dynamics of commitments', *Artificial Intelligence*, **113**(1-2), 1–40, (1999).

[14] Roberto Montagna, Giorgio Delzanno, Maurizio Martelli, and Viviana Mascardi, 'BDI^{ATL} : An alternating-time BDI logic for multiagent systems', in *EUMAS 2005*, pp. 214–223, (2005).

[15] Foundation of Intelligent Physical Agents, *Communicative act library specification, Technical Report FIPA00037*, http://www.fipa.org/specs/fipa00037/, 2002.

[16] Anand S. Rao, 'Agentspeak(l): BDI agents speak out in a logical computable language', in *MAAMAW*, volume 1038 of *Lecture Notes in Computer Science*, pp. 42–55. Springer, (1996).

[17] Anand S. Rao and Michael P. Georgeff, 'Modeling rational agents within a BDI-architecture', in *Knowledge Representation and Reasoning (KR)*, pp. 473–484, (1991).

[18] Anand S. Rao and Michael P. Georgeff, 'Decision procedures for BDI logics', *Journal of Logic and Computation*, **8**(3), 293–342, (1998).

[19] Earl D. Sacerdoti, *A structure for plans and behavior*, Elsevier-North Holland, 1977.

[20] Caroline Semmling and Heinrich Wansing, 'From BDI and stit to bdi-stit logic', *Logic and Logical Philosophy*, **17**(1-2), 185–207, (2008).

[21] Munindar P. Singh, 'Semantical considerations on some primitives for agent specification', in *Intelligent Agents II Agent Theories, Architectures, and Languages*, volume 1037 of *Lecture Notes in Computer Science*, pp. 49–64. Springer, (1996).

[22] Kaile Su, Abdul Sattar, Kewen Wang, Xiangyu Luo, Guido Governatori, and Vineet Nair, 'The observation-based model for bdi-agents', in *AAAI*, pp. 190–195, (2005).

[23] Birna van Riemsdijk, Wiebe van der Hoek, and John-Jules Ch. Meyer, 'Agent programming in dribble: from beliefs to goals using plans', in *AAMAS*, pp. 393–400. ACM, (2003).

[24] Michael Wooldridge, *Reasoning about Rational Agents*, The MIT Press, 2000.