BDI logic with probabilistic transition and fixed-point operator

NIDE, Naoyuki[1], Shiro Takata[2], and Megumi Fujita[1]

[1] Nara Women's University,
Kita-Uoya Nishimachi, Nara-shi, Nara, 630-8506 Japan
nide@ics.nara-wu.ac.jp, saboten@ics.nara-wu.ac.jp
[2] Kinki University,
Kowakae 3–4–1, Higashi-Osaka-shi, Osaka, 577-8502 Japan
shiro@info.kindai.ac.jp

Abstract. One of the advantages of the BDI (Belief-Desire-Intention) model is that we can formally discuss and prove properties about the mental states (beliefs, desires and intentions) and behaviors of rational agents using a modal logic called BDI logic. However, various extensions, such as probabilistic state transitions in reinforcement learning and cooperative acts in multi-agent environments, have been attempted in the BDI model. Since those notions are difficult to treat precisely in traditional BDI logic, the advantage of formalization in BDI logic is diminished. In this paper, we propose an extension of BDI logic, called $\mathfrak{TOMATO}$, which introduces probabilistic state transitions and a fixed-point operator. We can strictly describe and infer various properties of rational agents with those extended notions by using $\mathfrak{TOMATO}$.

1 Introduction

The BDI (Belief-Desire-Intention) model [1] is a model of rational agents based on Bratman's "theory of intention" [2, 3]. There have been many studies and applications on this model, which have proved its usefulness [4].

In the BDI model, a rational agent has three kinds of mental states, which are belief, desire and intention, and the agent determines its action to achieve its goal by maintaining and updating these states of mind. One of the features of the BDI model is that it has a modal logic system called "BDI logic". BDI logic explicitly describes those mental states and their temporal changes, so we can formally prove and discuss rational agents' mental states and their behaviors. For example, a blind commitment strategy [5], well-known one of the commitment maintenance strategies which is stated as 'once an agent intends to achieve ϕ necessarily in the future, then she maintains that intention until she believes that she has achieved ϕ', can be written as $\mathsf{INTEND}(\mathsf{AF}\,\phi) \supset \mathsf{A}(\mathsf{INTEND}(\mathsf{AF}\,\phi)\ \mathsf{U}\ \mathsf{BEL}(\phi))$. As another example, a property of rational agent that "if an agent intends to achieve p at the next time point, and believes that p and q are mutually excluded forever, then she does not intend to achieve q at that time", one of the consistencies of mental states [2], can be shown by proving $\mathsf{INTEND}(\mathsf{AX}\,p) \wedge \mathsf{BEL}(\mathsf{AG}(p \supset \neg q)) \supset \neg\,\mathsf{INTEND}(\mathsf{AX}\,q)$. This point is considered to be a major advantage to designing rational agents, and that's why the BDI model has been generally accepted.

However, in the advancement of research of rational agents, various extensions to BDI logic have been proposed. If there are mismatches between notions appearing in these extensions and the ones in traditional BDI logic, we may have difficulties in formalizing them appropriately. Therefore, one of the advantages of the BDI model that we can strictly discuss properties about rational agents can be diminished. Examples of such extensions are, as described in Section 2, "probabilistic state transitions" which are used in the reinforcement learning task and "cooperative actions" which are used in multi-agent system. In particular, these notions are considered important for realization of rational agents in the real world. Based on this standpoint, we propose a logic system called $\mathfrak{TOMATO}$ (Theory about Observations of Multi-Agents with Tense and Odds) which introduces probabilistic state transitions and a fixed-point operator by extending traditional BDI logic.

We have constructed sound and complete deduction systems of traditional BDI logic using sequence calculi [6–8]. Therefore, we also aim to construct one for $\mathfrak{TOMATO}$. In this paper, we show the soundness of the deduction system of $\mathfrak{TOMATO}$, and in addition, the completeness which is restricted to propositional logic. Our future work includes studying the completeness of $\mathfrak{TOMATO}$ in predicate logic.

With a deduction system, we can formally discuss properties of rational agents syntactically rather than semantically, and automatic proof checking also becomes possible. We also intend to construct a decision algorithm using the tableau method [9] in the future, though restricted to propositional logic.

One of the advantages of $\mathfrak{TOMATO}$ is that, using probabilistic state transition operators, we can describe state transitions in MDPs (Markoff decision processes), which is a basis of the reinforcement learning task. In addition, using a fixed-point operator, we can finitely describe notions, such as mutual belief and cooperative intentions, in multi-agent systems, which cannot be described in $\mathcal{LORA}$ [10] without using infinite conjunctions/disjunctions. Moreover, inferences about these properties using sequent calculus are possible. These points are discussed in detail in Section 4.

In this paper, we first describe the mismatches between the traditional BDI model and the above-mentioned new notions in Section 2, and we introduce $\mathfrak{TOMATO}$ in Section 3. In Section 4, we show examples of descriptions and proofs in $\mathfrak{TOMATO}$ concerning probabilistic state transitions and cooperative actions. In Section 5, we present discussions and describe our future work, and conclude in Section 6.

2 Divergence from BDI Model

2.1 Treatment of probabilistic state transition

As described in Section 1, one of the notions that is difficult to treat strictly in traditional BDI logic is the idea of probabilistic state transitions, which is mandatory to incorporate machine learning techniques into the BDI model.

We propose the integration of a BDI agent and reinforcement learning, in which an agent combines deliberation and reflexive actions according to the situation [11].

For example, when we are passing a familiar road, we can select the route in response to our surroundings without the need for practical reasoning. As another example, a soccer player instantaneously performs an appropriate action according to the skills acquired by intensive training. Our idea is, similar to these situations, to import reactive action acquired by learning into a BDI agent to enable more human-like behaviors.

We attempted, within the BDI model, to describe state transitions used in MDP [12], which is a basis for the reinforcement learning task [13]. However, MDP is basically based on probabilistic transitions, and within traditional BDI logic, which does not have probabilistic transition operators, we can only describe agent movement as "moves one of the accessible states".

For instance, if we try to write a situation "if an agent at state s_1 executes an action e_1, then it transfers to state s_2 and receives reward 3 with probability 0.7, or transfers to state s_3 and receives reward 5 with probability 0.3" in traditional BDI logic, we have to eliminate the probabilities and only write as "transfers to either one".

PCTL [14] is known as a logical system that extends CTL to treat a probabilistic transition. However, since it describes probability per path (a line of time points) as described in Section 5.2, describing the probability for each action (event) may be difficult in this logic.

2.2 Treatment of cooperative action

Another example is the difficulty in the treatment of cooperative actions in multi-agent environments. Even though this is an important issue, the original BDI logic can treat only a single agent's mental state.

There is a logical system $\mathcal{LORA}$ [10], which is extended to describe the mental states of multiple agents in multi-agent environments. It treats various concepts required for handling agents' cooperative actions, such as mutual belief, recognition of the potential for cooperative action, and generation and execution of joint intension. However, $\mathcal{LORA}$ is a complicated logical system with various components, including action expressions corresponding to dynamic logic and operators such as Agt for judging whether an agent can execute an action. Nevertheless, it is still necessary to introduce new operators, by using infinite conjunctions/disjunctions of formulas, to describe cooperative actions,

If a logical system is complicated, it will be intractable and difficult to construct its deduction system. Then the advantage of formalization in the logic is diminished. In fact, the deduction system of $\mathcal{LORA}$ has not been given.

As an example, for an agent group g, to form a joint intention for achieving a mutual goal (ϕ) of lifting a 1-ton object, it is necessary that agents in g can achieve this only cooperatively, and they mutually believe this fact. To describe this situation in $\mathcal{LORA}$, we introduce the formula $(\mathsf{J\text{-}Can}^0\ g\ \phi)$ using pre-existing operators, which states that g can first achieve ϕ in a single step, as an abbreviation of a formula signifying that "g can execute some action α and ϕ is achieved by this action. Also, g mutually believes this fact". Next, a formula $(\mathsf{J\text{-}Can}\ g\ \phi)$ which states that an agent group g can achieve the goal ϕ, is introduced as an abbreviation of the infinite disjunction $(\mathsf{J\text{-}Can}^0\ g\ \phi) \vee (\mathsf{J\text{-}Can}^0\ g\ (\mathsf{J\text{-}Can}^0\ g\ \phi)) \vee (\mathsf{J\text{-}Can}^0\ g\ (\mathsf{J\text{-}Can}^0\ g\ (\mathsf{J\text{-}Can}^0\ g\ \phi))) \vee \cdots$. Subsequently, the process of forming a joint intention of achieving ϕ is described using $\mathsf{J\text{-}Can}$.

However, to be accurate, we have to introduce $\mathsf{J\text{-}Can}$ as a new operator rather than as an abbreviation, because the infinite disjunctive cannot be originally written as a proper formula[1]. Moreover, because infinite conjunctions are used in the definition of mutual belief[2], this part of J-Can cannot be written in $\mathcal{LORA}$ either.

Consequently, we consider treating infinite conjunctions and disjunctions uniformly by introducing a fixed-point operator to reduce complication of the syntax.

3 Extension of BDI logic

In this section, based on the discussions so far, we propose a modal logic system $\mathfrak{TOMATO}$ for easily handling the notions described in Section 2. $\mathfrak{TOMATO}$ is a branching-time temporal logic with a fixed-point operator and mental state operators for each agent in multi-agent environments.

3.1 Formulas

Syntax We give the definition of formulas in $\mathfrak{TOMATO}$ here. Hereinafter, the word 'formula' means that of $\mathfrak{TOMATO}$ unless expressly stated otherwise. Symbols like x and y are used as usual variable symbols in first-order predicate logic, and symbols such as X and Y are variable symbols, each of which expresses a formula. We call the latter 'formula variables'[3]. Typically, they are used with fixed-point operators.

Suppose that we fix a first-order language $\mathcal{L}$, a set of formula variables $\mathcal{V}$, a set of event constant symbols $\mathcal{E}$, and a set of agent constant symbols $\mathcal{A}$, where $\mathcal{E}$ and $\mathcal{A}$ are finite and $\mathcal{V}$ is infinite. Hereafter, we write $\{p \mid p \in \mathbb{R}, 0 \leq p \leq 1\}$ as $[0,1]$. Then,

- Any atomic formula in $\mathcal{L}$ is a formula (in $\mathfrak{TOMATO}$).
- If ϕ, ψ are formulas, then $\phi \vee \psi$ and $\neg\phi$ are also formulas.
- If ϕ is a formula and x is a variable symbol, then $\forall x \phi$ is also a formula.
- If $e \in \mathcal{E}$, n is a positive integer, and for $i = 1, 2, \cdots, n$, ϕ_i is a formula, $p_i \in [0,1]$ and $r_i \in \{\geq, >\}$, then $\mathsf{X}^e({}_{r_1 p_1}\,\phi_1 \mid \cdots \mid {}_{r_n p_n}\,\phi_n)$ is a formula. In particular, when $n = 1$, we write $\mathsf{X}^e_{r_1 p_1}\,\phi_1$ instead of $\mathsf{X}^e({}_{r_1 p_1}\,\phi_1)$.
- If ϕ is a formula and $a \in \mathcal{A}$, then $\mathsf{BEL}^a\,\phi$, $\mathsf{DESIRE}^a\,\phi$, $\mathsf{INTEND}^a\,\phi$ are formulas.
- If $X \in \mathcal{V}$, then X is a formula.
- If ϕ is a formula, $X \in \mathcal{V}$, and X does not occur negatively (i.e. does not occur in odd number of nesting of '$\neg$') in ϕ, then $\mu X.\phi$ is a formula. However, X may occur only inside the scope of any modal operator (X^e, BEL^a, DESIRE^a, INTEND^a); for example, $\mu X.p \wedge X$ is not a formula.

We introduce $\wedge$, $\supset$, $\Leftrightarrow$, $\exists$ as abbreviations in the usual manner. In addition, $\nu X.\phi$ is an abbreviation of $\neg\mu X.\neg\phi[X := \neg X]$. Here μ is the so-called least fixed-point operator [15], and ν is the greatest fixed-point operator. We also introduce notations $\mathsf{X}^e_{<p}\,\phi$,

[1] In other words, no finite formula in $\mathcal{LORA}$ can be semantically equivalent to $(\mathsf{J\text{-}Can}\ g\ \phi)$, without introducing a new operator.

[2] See [10] for the need of infiniteness.

[3] The name 'formula variables' may be slightly irrelevant, because they don't range over formulas. However, their main use is to form fixed-points, which can be regarded as new formulas. In this sense, we call them 'formula variables'.

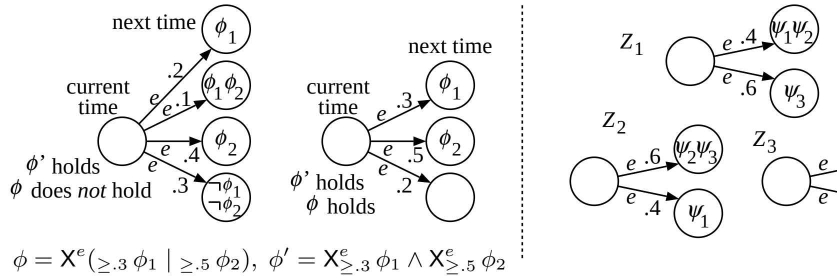

Fig. 1. Intuitive explanation of X^e operator

$\mathsf{X}^e_{\leq p}\,\phi$, $\mathsf{X}^e_{=p}\,\phi$ as abbreviations of $\neg\,\mathsf{X}^e_{\geq p}\,\phi$, $\neg\,\mathsf{X}^e_{>p}\,\phi$, $(\mathsf{X}^e_{\geq p}\,\phi)\wedge(\neg\,\mathsf{X}^e_{>p}\,\phi)$, respectively.

When needed, we eliminate ambiguities using parenthesis. Without parenthesis, operators associate in the following order: unary operators (including fixed-point operators), $\wedge$, $\vee$, $\supset$, $\Leftrightarrow$. Moreover, $\supset$ is right-associative, while other binary operators are left-associative.

Informal explanation of operators X^e is an extension of the next-time operator AX in CTL with an event e and transition probabilities. For example, $\mathsf{X}^e(_{\geq .3}\,\phi_1 \mid {}_{\geq .5}\,\phi_2)$ intuitively means that if an event e occurs, then at the next time point, ϕ_1 holds with probability of at least 0.3, and aside from that case, ϕ_2 holds with probability of at least 0.5. Note the difference between that formula and $\mathsf{X}^e_{\geq .3}\,\phi_1 \wedge \mathsf{X}^e_{\geq .5}\,\phi_2$; the former ensures that the case in which ϕ_1 holds and the one in which ϕ_2 holds does not overlap, but the latter does not (the left half of Fig. 1, where at each state ϕ_1 and ϕ_2 may or may not hold unless expressly stated).

$\mathsf{BEL}^a\,\phi$, $\mathsf{DESIRE}^a\,\phi$ and $\mathsf{INTEND}^a\,\phi$ mean that an agent a has a belief, desire or intention ϕ, respectively. For simplicity, we currently do not introduce probabilities into these mental state operators. However, it is thought to be possible to do so in the same way as for X^e operator. It can be useful for modeling agents, which have functions of some sort of statistical estimations such as pattern recognition.

Expressiveness compared to traditional BDI logics It is known that branching-time temporal logics with AX and the fixed-point operators have strictly stronger expressive power than CTL* [16, 17].

Since $\mathcal{TOMATO}$ has an individual next-time operator for each event, we have to write $\bigwedge_{e\in\mathcal{E}} \mathsf{AX}^e\,\phi$ (where, and hereafter, $\mathsf{AX}^e\,\phi$ is an abbreviation of $\mathsf{X}^e_{\geq 1}\,\phi$) to express what is equivalent to $\mathsf{AX}\,\phi$ in CTL. Formulas using other CTL or CTL* operators can also be written in $\mathcal{TOMATO}$ in a similar manner. Moreover, with event-wise next-time operators, we can write formulas such as $\mu X.(\psi \vee \phi \wedge \mathsf{AX}^e\,X)$, which means that "if an event e continuously occurs, then ϕ holds until ψ holds" and cannot be handled by CTL*.

Using the fixed-point operator, we can also handle notions which correspond to the action expressions in $\mathcal{LORA}$ [10]. In $\mathcal{LORA}$, concatenations, choices, and repetitions of actions, such as in dynamic logic, can be written as action expressions. For example,

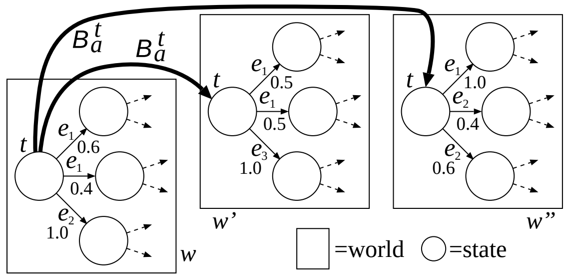

Fig. 2. Overview of BDI structure

a formula of $\mathcal{LORA}$ ($\mathsf{Nec}\,\alpha\,\phi$) means "just after executing an action α, ϕ holds". Supposing that $\alpha = ((\alpha_1;\alpha_2)*;\alpha_3)$, it means "if an action α_3 is executed soon after executing an action sequence α_1, α_2 for arbitrary times, then ϕ holds". In $\mathfrak{TOMATO}$, the equivalent of this can be written as $\nu X.(\mathsf{AX}^{e_3}\,\phi \wedge \mathsf{AX}^{e_1}\,\mathsf{AX}^{e_2}\,X)$, where e_1, e_2, e_3 are events corresponding to α_1, α_2, α_3, respectively.

Mutual mental states [8, 10] can also be handled by the fixed-point operator. When $g \subset \mathcal{A}$, we abbreviate $\bigwedge_{a\in g} \mathsf{BEL}^a\,\phi$ as $\mathsf{E\text{-}BEL}^g\,\phi$. Then, we abbreviate $\mathsf{E\text{-}BEL}^g\,\nu X.(\phi \wedge \mathsf{E\text{-}BEL}^g\,X)$ as $\mathsf{M\text{-}BEL}^g\,\phi$, which means that "a group of agents g has a mutual belief ϕ". Mutual desires and intentions can be written in the same manner.

3.2 Semantics

BDI structure First we fix the following:

- a set of possible worlds $W(\neq \emptyset)$
- for each $w \in W$, a set of states $St_w(\neq \emptyset)$ (may be different in different worlds)
- for each $w \in W$ and each $t \in St_w$, an interpretation (including variable assignment) $i_{w,t}$ of $\mathcal{L}$. In other words, a domain U and an interpretation of each constant, predicate, function, and variable symbol of $\mathcal{L}$. All components except the interpretation of predicate symbols must be the same for all states.
- for each $a \in \mathcal{A}$ and each $t \in \bigcup_{w\in W} St_w$, a serial, transitive and Euclidean binary relation $\mathcal{B}_a^t$ on the set $\{w \mid t \in St_w\}$, and serial binary relations $\mathcal{D}_a^t$, $\mathcal{I}_a^t$ on the same set.
- for each $w \in W$ and each $e \in \mathcal{E}$, a serial binary relation R_w^e on St_w, and a function $\mathcal{P}_w^e : R_w^e \to [0,1]$ where $\sum_{t'\in\{t' \mid tR_w^e t'\}} \mathcal{P}_w^e(t,t') = 1$ for any $t \in St_w$.

We call a tuple of the above-mentioned components a BDI-structure. Intuitively, a state corresponds to a time point in temporal logics, and a possible world is a time tree of states. $t\ R_w^e\ t'$ and $\mathcal{P}_w^e(t,t') = p$ mean that if an event e occurs at state t, then the next time is t' with probability p. $\mathcal{B}_a^t$, $\mathcal{D}_a^t$, and $\mathcal{I}_a^t$ are accessibility relations on possible worlds at time t, which represent the belief, desire and intention of agent a, respectively (an overview is shown in Fig. 2).

Since each R_w^e is defined to be serial, any event can occur at any state. However, in fact, usually only specific events can occur at a specific state. This property can be expressed by establishing a so-called dead-state d, at which a specific atomic formula *dead* holds, and creating state transitions from any state t to d with any non-executable event at t (in particular, state transition from d by any event goes to d itself). For example, a property that "if an event e can occur, then ϕ holds after e occurs" can be written as $\neg \mathsf{AX}^e$ *dead* $\supset \mathsf{AX}^e\, \phi$.

In this paper, for simplicity, we do not consider the mental state consistencies of the BDI model [7, 18]. Thus there are no special relationships among $\mathcal{B}_a^t$, $\mathcal{D}_a^t$ and $\mathcal{I}_a^t$. A brief discussion on this issue appears in Section 5.1.

Interpretation of formulas We write $\{(w,t) \mid w \in W, t \in \mathit{St}_w\}$ as *Swt* hereafter. Given a BDI structure M and a function $f_{\mathcal{V}} : \mathcal{V} \to 2^{\mathit{Swt}}$, we define the interpretation $[\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$ of a formula ϕ as follows (note that $[\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} \subset \mathit{Swt}$).

- If ϕ is an atomic formula, $[\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \{(w,t) \mid \phi$ is true w.r.t. $i_{w,t}\}$
- $[\![\phi \vee \psi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = [\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} \cup [\![\psi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$
- $[\![\neg\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \mathit{Swt} \setminus [\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$
- $[\![\forall x\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \bigcap_{u\in U} [\![\phi]\!]_{\langle M^u,f_{\mathcal{V}}\rangle}$ where M^u is a BDI structure obtained by replacing the interpretation of x in M with u.
- $[\![\mathsf{X}^e({}_{r_1p_1}\,\phi_1 \mid \cdots \mid {}_{r_np_n}\,\phi_n)]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \{(w,t) \mid$ there are some mutually disjoint subsets $T_1,\cdots,T_n$ of $\{t' \mid t\ R_w^e\ t'\}$ s.t. $T_i \subset \{t' \mid (w,t') \in [\![\phi_i]\!]_{\langle M,f_{\mathcal{V}}\rangle}\}$ and $\sum_{t'\in T_i} \mathcal{P}_w^e(t,t')\ r_i\ p_i$ for $i = 1,\cdots,n\}$ (note that each $r_1,\cdots,r_n$ is $\geq$ or $>$)
- $[\![\mathsf{BEL}^a\,\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \{(w,t) \mid$ for any w' s.t. $w\ \mathcal{B}_a^t\ w'$, $(w',t) \in [\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}\}$
- Similar for $[\![\mathsf{DESIRE}^a\,\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$ and $[\![\mathsf{INTEND}^a\,\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$
- $[\![X]\!]_{\langle M,f_{\mathcal{V}}\rangle} = f_{\mathcal{V}}(X)$ for $X \in \mathcal{V}$

Then, a formula ϕ, with (or without) free occurrences of a formula variable X, can be regarded as a function $f_\phi : \mathit{Swt} \to \mathit{Swt}$, which receives an interpretation of X as an argument and returns an interpretation of ϕ. Therefore, we define that

- $[\![\mu X.\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle}$ is the least fixed-point of f_ϕ.

Here, the least fixed-point is known to exist since f_ϕ in this case is monotonic by definition [19].

We say that ϕ holds at a state t of a world w when $[\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} \ni (w,t)$. If $[\![\phi]\!]_{\langle M,f_{\mathcal{V}}\rangle} = \mathit{Swt}$ for any M and $f_{\mathcal{V}}$, we say that ϕ is valid.

3.3 Deduction system

In this section we give a deduction system of $\mathcal{TOMATO}$ using sequent calculus.

We identify α-equivalent formulas. We regard the left-hand side of '$\to$' of a sequent as a (finite) multi-set of formulas, and likewise for the right-hand side (thus we do not have the exchange rule). Hereafter, we sometimes enclose a whole sequent into [ ] to clarify the range of the sequent in the text.

We use a capital Greek letter (Σ, Δ etc.; including a letter with a hash such as Σ', and Δ') to denote a multi-set of 0 or more formulas. As an exception, Θ contains only one or no formula.

The interpretation of a sequent $[\Sigma \to \Delta]$ is defined as that of the formula $\bigwedge \Sigma \supset \bigvee \Delta$. We define that $\bigwedge \emptyset = \mathit{true}$ and $\bigvee \emptyset = \mathit{false}$, where *true* is an abbreviation of a suitable tautology and *false* is an abbreviation of $\neg$*true*.

$$\frac{}{\phi \to \phi}\ \text{Initial} \quad \frac{\Sigma \to \Delta}{\Sigma, \Sigma' \to \Delta, \Delta'}\ \text{Weak} \quad \frac{\Sigma, \phi, \phi \to \Delta}{\Sigma, \phi \to \Delta}\ \text{CL} \quad \frac{\Sigma \to \Delta, \phi, \phi}{\Sigma \to \Delta, \phi}\ \text{CR}$$

$$\frac{\Sigma \to \Delta, \phi}{\Sigma, \neg\phi \to \Delta}\ \neg\text{L} \quad \frac{\Sigma, \phi \to \Delta}{\Sigma \to \Delta, \neg\phi}\ \neg\text{R} \quad \frac{\Sigma, \phi \to \Delta \quad \Sigma, \psi \to \Delta}{\Sigma, \phi \vee \psi \to \Delta}\ \vee\text{L} \quad \frac{\Sigma \to \Delta, \phi, \psi}{\Sigma \to \Delta, \phi \vee \psi}\ \vee\text{R}$$

$$\frac{\Sigma, \phi[x := t] \to \Delta}{\Sigma, \forall x\phi \to \Delta}\ \forall\text{L} \quad \frac{\Sigma \to \Delta, \phi[x := y]}{\Sigma \to \Delta, \forall x\phi}\ \forall\text{R} \quad \frac{\Gamma, \mathsf{X}^e_{>1-p}\, \neg\phi \to \Delta}{\Gamma \to \Delta, \mathsf{X}^e_{\geq p}\, \phi}\ \mathsf{X}_{\geq}\text{R}$$

$$\frac{\Gamma, \phi[X := \mu X.\phi] \to \Delta}{\Gamma, \mu X.\phi \to \Delta}\ \mu\text{L} \quad \frac{\Gamma \to \Delta, \phi[X := \mu X.\phi]}{\Gamma \to \Delta, \mu X.\phi}\ \mu\text{R} \quad \frac{\Gamma, \mathsf{X}^e_{\geq 1-p}\, \neg\phi \to \Delta}{\Gamma \to \Delta, \mathsf{X}^e_{>p}\, \phi}\ \mathsf{X}_{>}\text{R}$$

$$\frac{\Gamma, \mathsf{BEL}^a\, \Gamma \to \mathsf{BEL}^a\, \Delta, \Theta, \mathsf{BEL}^a\, \Theta}{\mathsf{BEL}^a\, \Gamma \to \mathsf{BEL}^a\, \Delta, \mathsf{BEL}^a\, \Theta}\ \mathsf{BEL\text{-}KD45} \quad \frac{\Gamma \to \Theta}{\mathsf{DESIRE}^a\, \Gamma \to \mathsf{DESIRE}^a\, \Theta}\ \mathsf{DESIRE\text{-}KD}$$

$$\frac{\Gamma \to \Theta}{\mathsf{INTEND}^a\, \Gamma \to \mathsf{INTEND}^a\, \Theta}\ \mathsf{INTEND\text{-}KD} \quad \frac{\cdots\ \mathsf{X}^e_{r_1\, p_1}(\phi_1 \wedge \psi_1) \wedge \cdots \wedge \mathsf{X}^e_{r_n\, p_n}(\phi_n \wedge \psi_n)\ \cdots}{\cdots\ \mathsf{X}^e({}_{r_1 p_1}\, \phi_1 \mid \cdots \mid {}_{r_n p_n}\, \phi_n)\ \cdots}\ \mathsf{X}_{\text{excl}}$$

Fig. 3. Inference rules of $\mathfrak{TOMATO}$ (excluding a rule described in Section 3.4)

Inference rules We enumerate the inference rules of $\mathfrak{TOMATO}$ in Fig. 3. However, note that there is an additional rule which is concerned with the X^e operator in the left-hand side of '$\to$' of a sequent. It is not shown in Fig. 3 but described in Section 3.4.

For a multi-set of formulas Γ and a unaly operator K, $K(\Gamma)$ stands for a multi-set of formulas obtained by applying K for each element of Γ.

In the $\forall$L rule, t is an arbitrary term. In the $\forall$R rule, y is a variable symbol which does not occur freely in the conclusion of the rule.

The X_{excl} rule means that any subformula of the form shown in the assumption anywhere in the sequent can be replaced by the formula shown in the conclusion. In this rule, $n \geq 2$, and for $i = 1, \cdots, n$, ψ_i is $\neg X_1 \wedge \cdots \wedge \neg X_{i-1} \wedge X_i \wedge \neg X_{i+1} \wedge \cdots \wedge \neg X_n$, where $X_1, \cdots, X_n$ are formula variables that does not occur freely in the conclusion of the rule. This rule is provided so that we can decompose formulas in the form of $\mathsf{X}^e(\cdots)$ into those in the form of $\mathsf{X}^e_{r_1 p_1}\, \phi_1$, by reversely applying it.

The BEL-KD45 rule, same as in [6, 20], is constructed so that the axiom schemas KD45 for the BEL operator are ensured to be held. The μL and μR rules are provided to enable proofs by loop (see Section 6 for example), such as in [6, 20, 21].

3.4 Additional inference rule for X^e

In this section, we describe an additional inference rule for the X^e operator, which is not included in Fig. 3.

Let $\Gamma = \{\mathsf{X}^e_{r_1 p_1}\, \psi_1, \cdots, \mathsf{X}^e_{r_n p_n}\, \psi_n\}$, where each $r_1, \cdots, r_n$ is $\geq$ or $>$, and $\mathfrak{Q} = \{\psi_1, \cdots, \psi_n\}$. If a function $v : 2^{\mathfrak{Q}} \to [0, 1]$ satisfies that $\sum_{Q \subset \mathfrak{Q}} v(Q) = 1$, and $(\sum_{Q \in \{T \mid T \subset \mathfrak{Q}, \psi_i \in T\}} v(Q))\, r_i\, p_i$ holds for each $i = 1, \cdots, n$, then we call v a *probability distribution function* (PrDF) of Γ. Intuitively, a PrDF determines probabilities of transitions from a state to next-time states, at each of which a subset Q of $\mathfrak{Q}$ holds, so that for each ψ_i, the probability that ψ_i holds satisfies $r_i p_i$.

For a PrDF v of Γ, we call $\{Q \subset \mathfrak{Q} \mid v(Q) > 0\}$ a satisfaction request set (SRS) of Γ on v, and write it as $\text{req}_v(\Gamma)$. Let Z be an SRS of Γ (on some PrDF v). If all elements of Z are satisfiable, we say that Z is satisfiable. In general, Γ is satisfiable iff there is a

$$\frac{\psi_1, \psi_2 \to \quad \psi_2, \psi_3 \to \quad \psi_3, \psi_1 \to}{\mathsf{X}^e_{\geq .3}\, \psi_1, \mathsf{X}^e_{\geq .4}\, \psi_2, \mathsf{X}^e_{\geq .6}\, \psi_3 \to} \qquad \frac{\psi_1 \to}{\mathsf{X}^e_{\geq .3}\, \psi_1, \mathsf{X}^e_{\geq .4}\, \psi_2, \mathsf{X}^e_{\geq .6}\, \psi_3 \to}$$

$$\frac{\psi_2 \to}{\mathsf{X}^e_{\geq .3}\, \psi_1, \mathsf{X}^e_{\geq .4}\, \psi_2, \mathsf{X}^e_{\geq .6}\, \psi_3 \to} \qquad \frac{\psi_3 \to}{\mathsf{X}^e_{\geq .3}\, \psi_1, \mathsf{X}^e_{\geq .4}\, \psi_2, \mathsf{X}^e_{\geq .6}\, \psi_3 \to}$$

Fig. 4. Example of inference rules about X^e operators

satisfiable SRS Z of Γ.

If, for $Z, Z' \subset 2^{\mathfrak{Q}}$, some $Q \in Z$ and $Z'' \subset 2^Q$ exist and $Z' = (Z \cup Z'') \setminus \{Q\}$ holds, we write $Z \succ Z'$. Let $\succ\!\succ$ be a non-reflective transitive closure of $\succ$. Note that if $Z \succ\!\succ Z'$ and Z is satisfiable, then Z' is also satisfiable. If Z is an SRS of Γ, and there is no SRS Z' of Γ which satisfies $Z \succ\!\succ Z'$, we say that Z is an essential SRS (eSRS). Since $\mathfrak{Q}$ is finite, there is no infinite sequence $Z_1, Z_2, \cdots$ s.t. $Z_1 \succ Z_2 \succ \cdots$. As a result, there exists a satisfiable SRS of Γ iff there exists a satisfiable eSRS of Γ.

Let $Z_1 = \{Q_{1,1}, \cdots, Q_{1,m_1}\}, \cdots, Z_k = \{Q_{k,1}, \cdots, Q_{k,m_k}\}$ be the enumeration of all eSRSs of Γ. Then for any sequence of positive integers $j_1, \cdots, j_k$, where $1 \leq j_1 \leq m_1, \cdots, 1 \leq j_k \leq m_k$, the following is an inference rule of $\mathfrak{TOMATO}$.

$$\frac{Q_{1,j_1} \to \quad \cdots \quad Q_{k,j_k} \to}{\Gamma \to}\ \mathsf{X\text{-}KD}$$

For example, Let $\Gamma = \{\mathsf{X}^e_{\geq 0.3}\, \psi_1, \mathsf{X}^e_{\geq 0.4}\, \psi_2, \mathsf{X}^e_{\geq 0.6}\, \psi_3\}$ and $\mathfrak{Q} = \{\psi_1, \psi_2, \psi_3\}$. Then a function $v : 2^{\mathfrak{Q}} \to [0, 1]$, where $v(\{\psi_1, \psi_2\}) = 0.4$, $v(\{\psi_3\}) = 0.6$, and $v(Q) = 0$ for all other $Q \subset \mathfrak{Q}$ is a PrDF of Γ, and $\mathrm{req}_v(\Gamma) = \{\{\psi_1, \psi_2\}, \{\psi_3\}\} = Z_1$ is an SRS of Γ. Z_1 is also an eSRS. A function v', where $v'(\{\psi_1, \psi_2\}) = 0.3$, $v'(\{\psi_2\}) = 0.1$, $v'(\{\psi_2, \psi_3\}) = 0.6$, and $v'(Q) = 0$ for all other $Q \subset \mathfrak{Q}$ is also a PrDF of Γ, but $\mathrm{req}_{v'}(\Gamma) = Z'_1$ is not an eSRS since $Z'_1 \succ\!\succ Z_1$.

In this example, Z_1 and $Z_2 = \{\{\psi_2, \psi_3\}, \{\psi_1\}\}$ and $Z_3 = \{\{\psi_3, \psi_1\}, \{\psi_2\}\}$ (see the right half of Fig. 1) are all eSRSs. In the tableau method, to show that $[\Gamma \to]$ is provable, we have to show that Γ is unsatisfiable. It is equivalent to show that there is no satisfiable eSRS of Γ, and also equivalent to show that any eSRS of Γ has at least one unsatisfiable element. The X-KD rule is constructed in this way.

Therefore, in this example, we have 8 $(= |Z_1| \cdot |Z_2| \cdot |Z_3|)$ rules, and one of them is the following.

$$\frac{\psi_1, \psi_2 \to \quad \psi_2, \psi_3 \to \quad \psi_2 \to}{\mathsf{X}^e_{\geq .3}\, \psi_1, \mathsf{X}^e_{\geq .4}\, \psi_2, \mathsf{X}^e_{\geq .6}\, \psi_3 \to}$$

However, the leftmost two sequents of the assumption of this rule are redundant. After removing similar redundancies from other rules, we need only 4 rules, as shown in Fig. 4, and the rest 4 can be omitted since the assumption of each of those includes another rule. (In addition, rules in Fig. 4 except the upper-left-most one can be aggregated into a single rule $\frac{\phi \to}{\mathsf{X}^e_{\geq p}\, \phi \to}$, where $0 < p \leq 1$.)

Definition of provability A sequent S is said to be derivable from a set L of sequents if one of the following conditions holds.

1. $S \in L$
2. There is an inference rule $\dfrac{S_1, \cdots, S_n}{S}$ ($n \geq 0$) and all of $S_1, \cdots, S_n$ are derivable from L

We say that a sequent S is provable if one of the following conditions is satisfied. Here $\phi^n(X)$ is defined as $\phi^0(X) = X$ and $\phi^n(X) = \phi[X := \phi^{n-1}(X)]$.

1. S is derivable from $\emptyset$.
2. $S = [\Sigma, \mu X.\phi \to \Delta]$ where X does not occur freely in Σ, Δ, and there is a positive integer n s.t. $[\Sigma, \phi^n(X) \to \Delta]$ is derivable from $\{[\Sigma, X \to \Delta]\}$.

A formula ϕ is defined to be provable if $[\to \phi]$ is provable.

Soundness and completeness In this section, we first show the soundness of $\mathfrak{TOMATO}$, and then we show a proof sketch to show the completeness of $\mathfrak{TOMATO}$ restricted to propositional logic. A study of the completeness of $\mathfrak{TOMATO}$ on predicate logic is for future work.

To show the soundness, it is enough to show that every inference rule preserves the validity of sequents, and that S in the item 2. of the provability definition is valid. The former is easy; therefore, we show the latter. For any ordinal α and the function f_ϕ in Section 3.2, we define a function f_ϕ^α : *Swt* $\to$ *Swt* as follows.

$$f_\phi^0(x) = x \qquad f_\phi^{\alpha+1}(x) = f_\phi(f_\phi^\alpha(x))$$
$$f_\phi^\lambda(x) = \bigcup_{\alpha<\lambda} \{f_\phi^\alpha(x)\} \text{ when } \lambda \text{ is a limit ordinal}$$

Then, if $[\Sigma, \phi^n(X) \to \Delta]$ is derivable from $\{[\Sigma, X \to \Delta]\}$, for any BDI structure and any infinite ordinal α, $[\Sigma \to \Delta]$ holds at any state in $f_\phi^\alpha(\emptyset)$. Also, an infinite ordinal α exists s.t. $f_\phi^\alpha(\emptyset) = [\![\mu X.\phi]\!]$. Thus $[\Sigma, \mu X.\phi \to \Delta]$ is valid.

Next we show the proof sketch of the completeness restricted to propositional logic. Without loss of generality, we can assume that any subformulas of the form $\mathsf{X}^e({}_{r_1 p_1}\,\phi_1 \mid \cdots \mid {}_{r_n p_n}\,\phi_n)$, where $n \geq 2$ do not occur anywhere in sequents, since we can omit them by reversely applying the X_{excl} rule (as described in Section 3.3).

Let *Nps* be a set of non-provable sequents that have only atomic formulas (i.e. atomic propositions) and formulas in the form of $\mu X.\phi$, $\mathsf{X}^e_{rp}\,\phi$, $\mathsf{BEL}^a\,\phi$, $\mathsf{DESIRE}^a\,\phi$, and $\mathsf{INTEND}^a\,\phi$ in the both sides of '$\to$', but do not have formulas in the form of $\mathsf{X}^e_{rp}\,\phi$ to the right of '$\to$'. For $S \in$ *Nps*, we define *dec*-$\mu(S)$ as a non-provable sequent obtained from S by reversely applying μL/R, $\vee$L/R, $\neg$L/R, $\mathsf{X}_{\geq}$R, and $\mathsf{X}_{>}$R rules as many times as possible. If there are more than one such sequents, choose an arbitrary one as *dec*-$\mu(S)$. Note that we cannot apply μL/R infinite times because in a formula $\mu X.\phi$ we do not have any X outside the scope of modal operators.

Regarding *Nps* as a set of states, we construct a 'flat' version of BDI structure (i.e. we do not take the set of worlds W into consideration, and all accessibility relations are binary relations on *Nps*) by the following procedure, which is based on Wang's algorithm [20, 22] for propositional modal logics.

First, we define binary relations $\mathcal{B}_a$, $\mathcal{D}_a$, and $\mathcal{I}_a$ on *Nps* for each $a \in \mathcal{A}$ as follows.

– $S\ \mathcal{D}_a\ S'$ iff we can obtain S' from *dec*-$\mu(S)$ by applying the following procedure:
  1. First, reversely apply Weak to *dec*-$\mu(S)$ to leave only all formulas in the form of $\mathsf{DESIRE}^a\,\phi$ to the left of '$\rightarrow$', and only one (iff there is any) of them in that form to the right of '$\rightarrow$'.
  2. Then, reversely apply DESIRE-KD once to remove outermost DESIRE^a.
  3. Last, reversely apply rules $\vee$L/R, $\neg$L/R, $\mathsf{X}_{\geq}$R, $\mathsf{X}_{>}$R as many times as possible.

– Similar for $\mathcal{I}_a$.

– To define $\mathcal{B}_a$, we first define $\mathcal{B}'_a$ in a similar manner to that for $\mathcal{D}_a$ and $\mathcal{I}_a$. Let $\mathsf{BEL}^{a+}(S)$ be the set of formulas of the form $\mathsf{BEL}^a\,\phi$ to the left of '$\rightarrow$' of sequent *dec*-$\mu(S)$, and $\mathsf{BEL}^{a-}(S)$ be a similar one for the right of '$\rightarrow$'. Assume that $S = S_0\ \mathcal{B}'_a\ S_1\ \mathcal{B}'_a\ S_2\ \mathcal{B}'_a\ \cdots$. Then $\mathsf{BEL}^{a+}(S_0), \mathsf{BEL}^{a+}(S_1), \cdots$, and $\mathsf{BEL}^{a-}(S_0), \mathsf{BEL}^{a-}(S_1), \cdots$ are both monotonically nondecreasing. Therefore, due to the finiteness of formulas and sequents, there is some S_k that satisfies that if $S_k\ \mathcal{B}'^{*}_a\ S'$ (here $\mathcal{B}'^{*}_a$ is the transitive closure of $\mathcal{B}'_a$), then $\mathsf{BEL}^{a+}(S_k) = \mathsf{BEL}^{a+}(S')$ and $\mathsf{BEL}^{a-}(S_k) = \mathsf{BEL}^{a-}(S')$. We define that $S\ \mathcal{B}_a\ S'$, $S'\ \mathcal{B}_a\ S''$ iff $S_k\ \mathcal{B}'^{*}_a\ S'$ and $S_k\ \mathcal{B}'^{*}_a\ S''$.

Next we define binary relations R^e on *Nps* and a function $\mathcal{P}^e : R^e \rightarrow [0,1]$ for each $e \in \mathcal{E}$ as follows. Let a sequent S be given.

1. First, we reversely apply Weak to *dec*-$\mu(S)$ to leave only all formulas in the form of $\mathsf{X}^e_{rp}\,\phi$ in the both sides of '$\rightarrow$'.
2. Then, reversely apply $\mathsf{X}_{\geq}$R and $\mathsf{X}_{>}$R as many times as possible to move all formulas in the form of $\mathsf{X}^e_{rp}\,\phi$ in the right-hand side of '$\rightarrow$' toward the left of '$\rightarrow$'. At this moment the sequent is in the form of $[\Gamma \rightarrow]$, where Γ is $\{\mathsf{X}^e_{r_1p_1}\,\psi_1, \cdots, \mathsf{X}^e_{r_np_n}\,\psi_n\}$.
3. Since $[\Gamma \rightarrow]$ is not provable, by the construction of X-KD rule, there is a PrDF v of Γ and an eSRS $\{Q_1, \cdots, Q_m\}$ of Γ on v, where none of sequents $[Q_1 \rightarrow], \cdots, [Q_m \rightarrow]$ are provable.
4. Now, we put $S\ R^e\ S'$ and $\mathcal{P}^e(S, S') = v(Q_j)$ iff S' can be obtained from some $[Q_j \rightarrow]$ above, by reversely applying rules $\vee$L/R, $\neg$L/R, $\mathsf{X}_{\geq}$R, and $\mathsf{X}_{>}$R as many times as possible.

Subsequently, for each state t in *Nps*, we choose an interpretation i_t of atomic propositions s.t. $i_t(p)$ is true iff p occurs to the left of '$\rightarrow$' of the sequent *dec*-$\mu(t)$. In addition, we also choose a function $f_{\mathcal{V}} : \mathcal{V} \rightarrow 2^{\mathit{Swt}}$ s.t. $f_{\mathcal{V}}(X) \ni t$ iff X occurs to the left of '$\rightarrow$' of the sequent *dec*-$\mu(t)$.

Now we have a 'flat' BDI structure. In addition, $\mathcal{B}_a$ satisfies KD45, and all other accessibility relations satisfy KD. We can easily convert it into a normal BDI structure M.

Then we show that for each state t in M, formulas to the left of '$\rightarrow$' of the sequent t are true at t, and ones to the right are false at t. We do so only for the formulas of the form $\mu X.\phi$ at both sides of '$\rightarrow$'.

Let F be a set of states (sequents) in M, which has $\mu X.\phi$ to the right of '$\rightarrow$'. By the construction method of M, for any ordinal α, we can show that $(f^{\alpha}_{\phi}(\emptyset))^c \supset F$ (here A^c denotes a complement set of A). Therefore, $\mu X.\phi$ is false at any state in F.

Let S be a state (sequent) in M, which has $\mu X.\phi$ to the left of '$\rightarrow$', and $S(n)$ be a state obtained from S by replacing $\mu X.\phi$ with $\phi^n(X)$. By the construction method of

M and the finiteness of formulas and sequents, there is a positive integer n s.t. for each sequence of states $S_0 \mathfrak{A} S_1 \mathfrak{A} \cdots$, where $S_0 = S(n)$ and $\mathfrak{A} = \bigcup_{a,t}(\mathcal{B}^t_a \cup \mathcal{D}^t_a \cup \mathcal{I}^t_a) \cup \bigcup_{e,w} R^e_w$, one of the followings holds[4].

i. X does not occur in some S_k.
ii. There are some k, ℓ s.t. $S_k = S_\ell$ and X occurs in S_k.

If all such sequences satisfy ii., then S is provable using the item 2. of the provability definition, and contradicts the assumption. Therefore, there is a sequence that satisfies i. above. By the construction of M, there is also a sequence $S'_0 \mathfrak{A} S'_1 \mathfrak{A} \cdots$, where $S'_0 = S$ and which satisfies i., and again by the construction of M, $\mu X.\phi$ is true at S.

A decision algorithm for propositional $\mathcal{TOMATO}$ can be directly derived from this proof of the completeness (if an algorithm to calculate eSRS is provided). We plan to mention this in the future.

4 Examples of description and proof

4.1 Modeling probabilistic state transitions

We can write the situation in the example of Section 2.1 as $\mathit{at}(s_1) \supset \mathsf{X}^{e_1}({}_{\geq .7}\, \mathit{at}(s_2) \wedge \mathit{reward}(3) \mid {}_{\geq .3}\, \mathit{at}(s_3) \wedge \mathit{reward}(5))$ using the probabilistic transition operator of $\mathcal{TOMATO}$.

Let ϕ be this formula. We can confirm that if we are at s_1, then after executing e_1, we can surely receive reward 3 or more by proving $\phi \wedge \mathit{at}(s_1) \supset \mathsf{AX}^{e_1} \exists x(\mathit{reward}(x) \wedge x \geq 3)$, provided that we can prove $3 \geq 3$ and $5 \geq 3$. The proof is shown in Fig. 5, where we abbreviate $\mathit{at}(s_2)$, $\mathit{reward}(3)$, $\mathit{at}(s_3)$, $\mathit{reward}(5)$, and $\mathit{reward}(x) \wedge x \geq 3$ as p_1, q_1, p_2, q_2, and ψ, respectively. An X-KD rule applied between the 3rd column from the bottom and a column right above it is derived from the fact that all eSRSs of $\{\mathsf{X}^{e_1}_{\geq .7}\, \xi_1, \mathsf{X}^{e_1}_{\geq .3}\, \xi_2, \mathsf{X}^{e_1}_{>0}\, \xi_3\}$ are $\{\{\xi_1, \xi_2\}, \{\xi_3\}\}$, $\{\{\xi_2, \xi_3\}, \{\xi_1\}\}$, and $\{\{\xi_3, \xi_1\}, \{\xi_2\}\}$ (where ξ_1, ξ_2, ξ_3 are arbitrary formulas).

Machine learning cannot be performed only by describing in logic, and requires external tools to do so. However, after learning, we can describe the result as a property of an agent like the one above. Also, there is a possibility to implement a learning system within a frame of logic. In this sense, treating such properties in logic has a positive significance.

4.2 Modeling coordinated actions

J-Can described in Section 2.2 is necessary to describe coordinated actions. However, in $\mathcal{LORA}$, it can only be written using infinite disjunctions and conjunctions. It is expressible in $\mathcal{TOMATO}$ using the fixed-point operator.

To describe the first half of the description of $(\mathsf{J\text{-}Can}^0\ g\ \phi)$ in Section 2.2 (i.e. "g

[4] In other words, the process of reversely applying rules continuously will eventually stop by entering a loop. That is why our system can have a decision algorithm, despite the lack of subformula property.

$$\dfrac{\dfrac{\dfrac{\vdots}{p_1 \wedge q_1 \wedge X_1 \wedge \neg X_2, p_2 \wedge q_2 \wedge \neg X_1 \wedge X_2 \to} \quad \dfrac{\vdots}{\to 5 \geq 3} \quad \dfrac{\vdots}{\to 3 \geq 3}}{\dfrac{\vdots}{p_2 \wedge q_2 \wedge \neg X_1 \wedge X_2, \neg \exists x \psi \to}} \quad \dfrac{\vdots}{\neg \exists x \psi, p_1 \wedge q_1 \wedge X_1 \wedge \neg X_2 \to}}{\dfrac{\mathsf{X}^{e_1}_{\geq .7}(p_1 \wedge q_1 \wedge X_1 \wedge \neg X_2), \mathsf{X}^{e_1}_{\geq .3}(p_2 \wedge q_2 \wedge \neg X_1 \wedge X_2), \mathsf{X}^{e_1}_{>0} \neg \exists x \psi \to}{\dfrac{\vdots}{\to \phi \wedge \mathit{at}(s_1) \supset \mathsf{AX}^{e_1} \exists x \psi}}}$$

Fig. 5. Example of proof (1)

can execute some action α and ϕ is achieved by this action"), we introduce a predicate Agt s.t. $\mathsf{Agt}(e, a)$ holds iff an agent a can execute an event e. We use a list structure in first-order language to represent a group of agents, and introduce the 'member' predicate using its general definition in Prolog, i.e. a non-logical axiom $\forall x(\text{member}(x, \text{cons}(x, \text{nil})) \wedge \forall y \forall z(\text{member}(x, z) \supset \text{member}(x, \text{cons}(y, z))))$[5]. Then, we can represent the above-mentioned part as $\mu X.(\phi \vee \bigvee_{e \in \mathcal{E}, a \in \mathcal{A}}(\mathsf{Agt}(e, a) \wedge \text{member}(a, g) \wedge \mathsf{AX}^e\, X))$. (Note: $\mathcal{LORA}$ introduces equivalents for Agt and 'member' as primordial components of formulas, and enables the applying of $\forall$ for agents and actions. These reduce the length of formulas, but complicates syntax and semantics.)

Let ψ be this formula, and abbreviate $\nu X.(\xi \wedge \bigwedge_{a \in \mathcal{A}}(\text{member}(a, g) \supset \mathsf{BEL}^a\, X))$ as $\mathsf{E\text{-}Know}^g\, \xi$, which states that "$\xi$ holds and an agent group g mutually believes it". Then $\mathsf{E\text{-}Know}^g\, \psi$ is equivalent to $(\mathsf{J\text{-}Can}^0\ g\ \phi)$. Further, we can represent $(\mathsf{J\text{-}Can}\ g\ \phi)$ by $\mu X.((\mathsf{J\text{-}Can}^0\ g\ \phi) \vee (\mathsf{J\text{-}Can}^0\ g\ X))$. By proceeding in this way, we can construct further descriptions about coordinations of agents like in $\mathcal{LORA}$.

To prove various properties of coordinations is also possible. Fig. 6 is a proof of a property $(\mathsf{J\text{-}Can}^0\ g\ \phi) \supset \mathsf{E\text{-}Know}^g(\mathsf{J\text{-}Can}^0\ g\ \phi)$, whose equivalent is represented in $\mathcal{LORA}$ (we assume $\mathcal{A}$ in Section 3.1 be $\{a_1, \cdots, a_n\}$). Using the above-mentioned ψ, this formula can be rewritten as $\mathsf{E\text{-}Know}^g\, \psi \supset \mathsf{E\text{-}Know}^g\, \mathsf{E\text{-}Know}^g\, \psi$, so we give the proof of this formula. In that figure, we abbreviate $\bigwedge_{a \in \mathcal{A}}(\text{member}(a, g) \supset \mathsf{BEL}^a\, \xi)$ as $B_g\, \xi$. Hence, $\mathsf{E\text{-}Know}^g\, \xi$ is an abbreviation of $\neg \mu X.(\neg \xi \vee \neg B_g \neg X)$. Furthermore we abbreviate $\mu X.(\neg \xi \vee \neg B_g \neg X)$ as $\mathsf{nEk}^g\, \xi$. As a result, $\mathsf{E\text{-}Know}^g\, \xi$ is syntactically equivalent to $\neg\, \mathsf{nEk}^g\, \xi$. In Fig. 6, the topmost column of the rightward proof figure is derived from the leftward proof figure using the item 2. of the provability definition.

5 Discussions

We have given an extended BDI logic to handle notions required for formalizing realistic rational agents. However, there are more issues to consider, though we do not treat them in this paper. In this section we discuss some of them.

5.1 Treatment of mental state consistencies

As we described in Section 3.2, we have omitted discussions about mental state consistencies for simplicity. However, mental state consistencies are significant in Bratman's

[5] In fact, the proof in Fig. 6 does not depend on this definition.

$$\dfrac{\dfrac{\dfrac{\dfrac{X \to \mathsf{nEk}^g\,\psi}{\vdots}}{\neg\,\mathsf{nEk}^g\,\psi \to \neg X}}{\mathsf{BEL}^{a_1}\,\neg\,\mathsf{nEk}^g\,\psi \to \mathsf{BEL}^{a_1}\,\neg X}\quad \vdots \quad \dfrac{\dfrac{\dfrac{X \to \mathsf{nEk}^g\,\psi}{\vdots}}{\neg\,\mathsf{nEk}^g\,\psi \to \neg X}}{\mathsf{BEL}^{a_n}\,\neg\,\mathsf{nEk}^g\,\psi \to \mathsf{BEL}^{a_n}\,\neg X}}{\vdots \quad \cdots\ (n \text{ branches in total})\ \cdots \quad \vdots}$$

$$\dfrac{\dfrac{\vdots}{\neg\,\mathsf{E\text{-}Know}^g\,\psi \to \mathsf{nEk}^g\,\psi} \quad \dfrac{\dfrac{\dfrac{B_g\,\neg\,\mathsf{nEk}^g\,\psi \to B_g\,\neg X}{\vdots}}{\neg\,B_g\,\neg X \to \neg\psi \vee \neg\,B_g\,\neg\,\mathsf{nEk}^g\,\psi}}{\neg\,B_g\,\neg X \to \mathsf{nEk}^g\,\psi}}{\neg\,\mathsf{E\text{-}Know}^g\,\psi \vee \neg\,B_g\,\neg X \to \mathsf{nEk}^g\,\psi} \qquad \dfrac{\dfrac{\mathsf{nEk}^g\,\mathsf{E\text{-}Know}^g\,\psi \to \mathsf{nEk}^g\,\psi}{\vdots}}{\to \mathsf{E\text{-}Know}^g\,\psi \supset \mathsf{E\text{-}Know}^g\,\mathsf{E\text{-}Know}^g\,\psi}$$

Fig. 6. Example of proof (2)

intention principle and need to be handled to describe rational agents. For example, the property that "an agent will not form an intention if she cannot believe the possibility of achieving it" is said to be one of the required properties of rational agents. In traditional BDI logic, as in [7, 18], this is written as $\mathsf{INTEND}(\mathsf{EX}\,\phi) \supset \mathsf{BEL}(\mathsf{EX}\,\phi)$, and it presents the semantics that make it valid and the deduction system that can prove it. Currently $\mathcal{TOMATO}$ cannot treat such a property. This is for future work.

When considering this, it is also interesting to consider consistency between probabilistic mental state operators mentioned in Section 3.1. For example, when the possibility of achievement of ϕ is believed with a probability 0.9, can we intend ϕ?

5.2 Treatment of probabilistic transitions

The temporal operator in $\mathcal{TOMATO}$ is an extension of the next-time operator in CTL with a probability. This is because we introduced this operator so that we can construct a proof system base on the tableau method. However, a disadvantage of this is that the description with the probability is restricted to the transition between current time and the next time.

In PCTL [14], we can describe the probability on the time sequence (path). In other words, the probability is given on path formulas. For example, a property "we can achieve ϕ not less than the probability of 0.9 in the future" can be written as $[\mathit{true}\,\mathcal{U}\,\phi]_{\geq 0.9}$. Currently $\mathcal{TOMATO}$ cannot describe such a property.

However, as described in Section 2.1, it is difficult to describe event-wise probabilities in PCTL, unlike in $\mathcal{TOMATO}$. Moreover, it is believed to be difficult for PCTL to create the proof system using the tableau method due to an excessive flexibility of probability descriptions in PCTL. Even for qualitative PCTL, in which probabilistic descriptions are restricted to 0 and 1, no deduction system is yet known [23]. To take the balance of construction of the proof system and flexibility of representation is an important issue.

5.3 Treatment of stability of mental states

We believe that there are more issues to be considered on BDI logic though we did not treat them in this paper. For example, mental states, such as belief, should generally be kept by default. However, there is no such concept in BDI logic in nature. The mental states in BDI logic are merely modal operators, and represented by accessibility relations on possible worlds, which vary at different times. Thus, there is no logical relation between the current belief and the one in the next time. If we want to keep the belief to some extent, we must explicitly introduce a non-logical axiom such as $\psi \supset \mathsf{A}(\mathsf{BEL}(\phi)\,\mathsf{U}\,\xi)$ (believes ϕ until ξ holds).

In the standard implementation of BDI agents, mental states, such as belief, are restricted to first-order formulas, and an agent adds or deletes its mental states in its database by an event such as `add-belief` and `del-belief`. The addition and deletion of her mental states occurs procedurally, so the consistency between it and the logic is not guaranteed. There are some trials, such as AgentSpeak(L) [24], for bridging this gap by offering a proof theory about the properties of such procedures. However, they do not dissolve the un-naturalness of the logic that the mental states are not maintained by default, nor eliminate the fact that mental states are restricted to first-order formulas in the implementations.

Mental states are not always expected to be kept; for example, if there is a belief $\mathsf{BEL}(\mathsf{AX}\,\phi)$ (believes that "ϕ in next time"), it would be natural that we have $\mathsf{BEL}(\phi)$ the next time. [25] is one of such studies, though it is based on non-branching temporal logic and lack of descriptive power is anticipated. It will be interesting to consider how we treat such things in modal logics.

Some studies treat the updating of mental states as an update of the model itself instead of time transition. Though such a method is difficult to apply to MDP because time path is restricted to be unique, it may be also a possibility to handle stability of mental states naturally.

6 Conclusion

In this paper, we proposed $\mathfrak{TOMATO}$, an extended BDI logic with probabilistic transitions and a fixed-point operator, to enable formal descriptions and discussions on rational agents with notions such as probabilistic state transitions in reinforcement learning and cooperative actions in multi-agent environments. We also showed some examples of descriptions and proofs associated with these notions. Our future work includes a study of the completeness of $\mathfrak{TOMATO}$ on predicate logic, construction of a proof algorithm in propositional logic and to introduce some of the notions described in Section 5, especially the consistency of mental states.

We expect $\mathfrak{TOMATO}$ to be a productive tool for modeling, designing and implementing rational agents.

References

1. Rao, A.S., Georgeff, M.P.: Modeling Rational Agents within a BDI-Architecture. In: Proc. of International Conference on Principles of Knowlegde Representation and Reasoning. (1991) 473–484
2. Bratman, M.E.: Intention, Plans, and Practical Reason. Harvard University Press (1987)

3. Bratman, M.E., Israel, D.J., Pollack, M.E.: Plans and resource-bounded practical reasoning. Computational Intelligence **4** (1988) 349–355
4. Wooldridge, M., Rao, A., eds.: Foundations of Rational Agency. Volume 14 of Applied Logic Series. Kluwer Academic Publishers (1999)
5. Rao, A.S., Georgeff, M.P.: Modeling Rational Agents within a BDI-Architecture. In Huhns, M.N., Singh, M.P., eds.: Reading in Agents. Morgan Kaufmann, San Francisco (1997) 317–328
6. Nide, N., Takata, S., Araragi, T.: Deduction Systems for BDI Logics Using Sequent Calculus. Computer Software **20**(1) (2003) 66–83 (In Japanese).
7. Nide, N., Takata, S., Araragi, T.: Reasoning About Mental State Compatibilities of Rational Agents and Its Applications. Transactions of the Institute of Electronics, Information and Communication Engineers **J86-D-I**(8) (2003) 514–523 (In Japanese).
8. Nide, N., Takata, S., Araragi, T.: A Deduction System of Extended BDI Logic to Handle Mutual Belief. Transactions of Information Processing Society of Japan **46**(SIG2 (TOM11)) (2005) 85–99 (In Japanese).
9. Wolper, P.: The tabuleau method for temporal logic: an overview. Logique et Anal. **28** (1985) 119–136
10. Wooldridge, M.: Reasoning about Rational Agents. The MIT Press (2000)
11. Takata, S., Nide, N., Yamakawa, H., Miyazaki, K., Ohta, M.: An achievement method of bdi agent who practically reasons about the skill acquired isung reinforcement learning. In: Proc. of JAWS2004. (2004) 517–524 (In Japanese).
12. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons (1994)
13. Nide, N., Takata, S., Yamakawa, H., Miyazaki, K., Ohta, M.: Correspondence between BDI model and world model in reinforcement learning. In: Proc. of JAWS2004. (2004) 378–385
14. Hansson, H., Jonsson, B.: A Logic for Reasoning about Time and Reliability. Formal Aspects of Computing **6**(5) (1994) 512–535
15. Kozen, D.: Results on the propositional μ-calculus. Theoretical Computer Science **27** (1983) 333–354
16. Dam, M.: Translating CTL into the modal μ-calculus. Technical Report ECS-LFCS-90-123, Laboratory for Foundations of Computer Science, Department of Computer Science, University of Edinburgh (1990)
17. Manolios, P.: Mu-calculus model-checking. In: Computer-Aided reasoning: ACL2 case studies. Kluwer Academic Publishers (2000) 93–111
18. Rao, A.S., Georgeff, M.P.: Decision Procedures for BDI Logics. Journal of Logic and Computation **8**(3) (1998) 292–343
19. Tarski, A.: A lattice-theoretical fixpoint theorem and its application. Pacific Journal of Mathematics **5** (1955) 285–309
20. Nide, N., Takata, S.: Deduction Systems for BDI Logics Using Sequent Calculus. In: Proc. of International Joint Conference on Autonomous Agents & Multiagent Systems (AAMAS 2002). (2002) 928–935
21. Stirling, C.: Modal and Temporal Properties of Processes. Springer Verlag (2001)
22. Wang, H.: Towards mechanical mathematics. IBM Journal of Research and Development **4** (1960) 224–268
23. Brázdil, T., Forejt, V., Křetínský, J., Kučera, A.: The satisfiability problem for probabilistic ctl. In: Proc. of 23rd Annual IEEE Symposium on Logic in Computer Science. (2008) 391–402
24. Rao, A.S.: AgentSpeak(L): BDI agents speak out in a logical computable language. In: Proc. of MAAMAW-96. Volume 1038 of LNAI., Springer-Verlag (1996) 42–55
25. Su, K., Sattar, A., Wang, K., Luo, X., Governatori, G., Padmanabhan, V.: Observation-based Model for BDI-Agents. In: Proc. of AAAI 2005. (2005) 190–195