1 Introduction

There are two main semantic approaches to formalizing agent systems via modal logics, the possible worlds semantics [Hintikka, 1962] and the interpreted system model [Fagin et al., 1995]. The first approach includes the well-known theory of intension [Cohen and Levesque, 1990] and the formalism of the belief-desire-intension paradigm [Rao and Georgeff, 1998]. The second approach, offers a natural interpretation, in terms of the states of computer processes, to S5 epistemic logic. The significance of the second approach is that we are able to associate the system with a computer program, and formulas can be understood as properties of program computations. In this sense, the interpreted system model is computationally grounded [Wooldridge, 2000].

2 Computationally Grounded BDI Logic

We introduce a multimodal logic of belief, desire and intention, called OBDI logic, where the changes and computation of agents’ beliefs, desires, and desires are based on agents’ observations (i.e., local states). We assume the reader is familiar with the notion of interpreted system model, and we will follow the terminology of [Fagin et al., 1995].

2.1 Syntax

Given a set \( \Phi \) of propositional atoms, the language of OBDI logic is defined by the following BNF notations:

\[
\langle \text{wff} \rangle ::= \langle \text{wff} \rangle \land \langle \text{wff} \rangle | \langle \text{wff} \rangle \lor \langle \text{wff} \rangle | \neg \langle \text{wff} \rangle | \langle \text{wff} \rangle \rightarrow \langle \text{wff} \rangle \\
B_i \langle \text{wff} \rangle \mid D_i \langle \text{wff} \rangle \mid I_i \langle \text{wff} \rangle \\
\]

Informally, \( B_i \varphi \) and \( D_i \varphi \) means that agent \( i \) believes and desires \( \varphi \), respectively. While \( I_i \varphi \) denotes that \( \varphi \) holds under the assumption that agent \( i \) acts based on his intention.

2.2 The BDI-system Model

Given a set \( G \) of global states and a system \( K \) over \( G \), an agent’s mental state over system \( K \) is a tuple \( (B, D, I) \), where \( B, D \) and \( I \) are systems (sets of runs over \( G \)) such that \( I \subseteq K \) and \( D \subseteq B \subseteq K \). A BDI-system is a structure \( (K, M_1, \ldots, M_n) \), where \( K \) is a system and for every \( i, M_i \) is agent \( i \)'s mental state over \( K \).

Assume that we have a set \( \Phi \) of primitive propositions. An interpreted BDI-system \( I \) consists of a pair \( (S, \pi) \), where \( S \) is a BDI-system and \( \pi \) is a valuation function, which gives the set of primitive propositions true at each point in \( G \).

2.3 Semantics

In what follows, we inductively define the satisfaction relation \( \models_{OBDI} \) between a formula \( \varphi \) and a pair of interpreted BDI-system and a point. Given an interpreted BDI-system \( I = (S, \pi) \), suppose that \( S = (K, M_1, \ldots, M_n) \) and for every \( i, M_i = (B_i, D_i, I_i) \). Let \( r \) be a run in \( K \) and \( u \) a natural number, then we have that:

- \( (I, r, u) \models_{OBDI} B_i \varphi \iff (I, r', v) \models_{OBDI} \varphi \) for those \( (r', v) \in B_i \) such that \( (r, u) \sim_i (r', v) \);
- \( (I, r, u) \models_{OBDI} D_i \varphi \iff (I, r', v) \models_{OBDI} \varphi \) for those \( (r', v) \in D_i \) such that \( (r, u) \sim_i (r', v) \);
- \( (I, r, u) \models_{OBDI} I_i \varphi \iff (I, r', v) \models_{OBDI} \varphi \) for those \( (r', v) \in I_i \) such that \( (r, u) \sim_i (r', v) \);

1 This work was supported by the Australian Research Council grant DP0452628, and partially by the NSFC grants 60496327, 10410638 and 60473004.
Proposition 1 The following formulas are valid:

- $X(\varphi \Rightarrow \psi) \Rightarrow (X\varphi \Rightarrow X\psi)$
- $X\varphi \Rightarrow YX\varphi$
- $\neg X\varphi \Rightarrow Y\neg X\varphi$

where $X$ and $Y$ stand for $B_1$, $D_1$ or $I_1$ (for the same i).

- Relationship between belief and desire $B_i\varphi \Rightarrow D_i\varphi$

- Temporal operators $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$
- $\Box(\neg \varphi) \Rightarrow \neg \Box \varphi$
- $\diamond U\psi \Leftrightarrow \psi \vee (\varphi \land \Box(\varphi U\psi))$

2.4 Axiomatization

We give a proof system, called OBDI proof system, for those BDI-agents with perfect recall and a global clock, which contains the axioms of propositional calculus plus those formulas in Propositions 1. The proof system is closed under the propositional inference rules plus: $\frac{\Box\varphi}{\Box D_i\varphi}$, and $\frac{\Box \varphi}{\Box I_i \varphi}$ for every agent $i$.

Theorem 2 The OBDI proof system is sound and complete with respect to interpreted BDI-systems with satisfaction relation $\models_{OBDI}$.

3 Model Checking BDI-Agents

In order to make our model checking algorithm practically useful, we must consider where our model, an interpreted BDI-system comes from. To make the things simpler, we may consider some abstract programs such as finite-state programs, which are expressible enough from the standpoint of theoretical computer science. Moreover, to make our model checking system practically efficient, we use symbolic model checking techniques. Thus, a finite-state program in our approach is represented in a symbolic way.

3.1 Symbolic representation of interpreted BDI-agents

We formally define a (symbolic) finite-state program with $n$ agents as a tuple $P = (x, \theta(x), \tau(x, x'), O_1, \ldots, O_n)$, where $x$ is a set of system variables; $\theta$ is a boolean formula over $x$, called the initial condition; $\tau$ is a boolean formula over $x \cup x'$, called the transition relation; and for each $i$, $O_i \subseteq x$, containing agent $i$’s local variables, or observable variables. Given a state $s$, we define agent $i$’s local state at state $s$ to be $s \cap O_i$. We may associate with $P$ an interpreted system $I_P = (\mathcal{R}, \pi)$ called the generated interpreted system of $P$.

For convenience, we may use $\mathcal{P}(\theta, \pi)$ to denote a finite-state program with $n$ agents $(x, \theta(x), \tau(x, x'), O_1, \ldots, O_n)$, if $x$ and $O_1, \ldots, O_n$ are clear from the context. Given a finite-state program $\mathcal{P}(\theta, \pi)$ with $n$ agents, we define an agent’s internal program (over $\mathcal{P}(\theta, \pi)$) as a tuple $\langle \mathcal{P}(\theta_1, \pi_1), \mathcal{P}(\theta_2, \pi_2), \mathcal{P}(\theta_3, \pi_3) \rangle$, where $\theta_j \Rightarrow \theta$ and $\pi_j \Rightarrow \pi$, for $j = 1, 2, 3$, and $\theta_2 \Rightarrow \theta_1$ and $\pi_2 \Rightarrow \pi_1$ are valid. Clearly, an agent’s internal program is exactly related with an agent’s mental state. Thus, we define a (symbolic) BDI-program with $n$ agents as a tuple $P_A = (\mathcal{P}_K, P_1, \ldots, P_n)$, where $\mathcal{P}_K$ is a finite-state program with $n$ agents and for each agent $i$, $P_i$ is agent $i$’s internal program over $\mathcal{P}$. We use $I_{P_A}$ to denote the corresponding interpreted BDI-system.

3.2 Model checking OBDI logic

Theorem 3 Given a BDI-program with $n$ agents $P_A = (\mathcal{P}_K, P_1, \ldots, P_n)$, suppose that $\mathcal{P}_K = (x, \theta(x), \tau(x, x'), O_1, \ldots, O_n)$, and for every $i$, $P_i = (\mathcal{P}(\theta_1, \pi_1), \mathcal{P}(\theta_2, \pi_2), \mathcal{P}(\theta_3, \pi_3))$. Then, for every LTL formula $\varphi$ and agent $i$, the following are valid in $I_{P_A}$:

1. $B_i\varphi \Rightarrow \forall (x - O_i)(G(\mathcal{P}(\theta_1, \pi_1)) \Rightarrow \Gamma(\varphi, \theta_1, \pi_1))$
2. $D_i\varphi \Rightarrow \forall (x - O_i)(G(\mathcal{P}(\theta_2, \pi_2)) \Rightarrow \Gamma(\varphi, \theta_2, \pi_2))$
3. $I_i\varphi \Rightarrow \forall (x - O_i)(G(\mathcal{P}(\theta_3, \pi_3)) \Rightarrow \Gamma(\varphi, \theta_3, \pi_3))$

where $\Gamma(\varphi, \theta, \pi)$ is a boolean formula built from $\varphi, \theta, \pi$ by using quantifications and fixed-point operations $\operatorname{lt}$ and $\operatorname{gt}$.

Remark that Theorem 3 provides a reduction of OBDI to LTL, while Proposition 9 in [Su, 2004] gives an OBDD-based method of model checking LTL formulas. The complexity of our reduction of logic OBDI to LTL is $PSPACE$-complete. However, because quantifications of boolean functions and fixed-point operators can be dealt with in any OBDD package, the reduction can be based on OBDDs. In fact, we implemented a prototype of the OBDI model checker using CUDD, a very influential OBDD package developed by Fabio Somenzi, and achieved some preliminary experimental results.

4 Concluding Remarks

In this work, we have explored computationally grounded modal logics that characterize the internal attitudes of an agent—its beliefs, desires, etc., beyond S5 axioms and carried out a methodology on symbolic model checking for general BDI-agents.

References

[Cohen and Levesque, 1990] P.R. Cohen and H.J. Levesque. Inten-
sion is choice with commitment. Artificial Intelligence, 42:23–
261, 1990.

[Fagin et al., 1995] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. Rea-


[Su, 2004] K. Su. Model checking temporal logics of knowledge in

[van der Hoek and Wooldridge, 2003] W. van der Hoek and M.
Wooldridge. Towards a logic of rational agency. L.J. of IG,

theories of agency. In E. Durfee, editor, Proceedings of ICMA-