A logic of situated resource-bounded agents

Natasha Alechina and Brian Logan

University of Nottingham
School of Computer Science
Nottingham NG8 1BB, UK
{nza,bsl}@cs.nott.ac.uk

Abstract. We propose a framework for modelling situated resource-bounded agents. The framework is based on an objective ascription of intentional modalities and can be easily tailored to the system we want to model and the properties we wish to specify. As an elaboration of the framework, we introduce a logic, OBA, for describing the observations, beliefs, intentions and actions of simple agents, and show that OBA is complete, decidable and has an efficient model checking procedure, allowing properties of agents specified in OBA to be verified using standard theorem proving or model checking techniques.

1 Introduction

A major goal in intelligent agent research is the formal modelling of agent-environment systems. Such an account is key both in deepening our understanding of the notion of agency, e.g., the relationships between agent architectures, environments and behaviour, and for the principled design of agent systems. A common approach is to model the agent and its environment in some logic and prove theorems about the agent's behaviour in that logic. It is perhaps most natural to reason about the behaviour of the agent in an epistemic logic; epistemic notions such as knowledge and belief provide a compact and powerful way of reasoning about the structure and behaviour of agents [13], and there has been a considerable amount of work in this area, for example, [12, 11, 15, 18, 9, 14, 21, 20, 22].

Much of this work has focused on postulating properties of agents in general, for example the relationships between beliefs, desires and intentions investigated in [18]. While useful, such results can only provide very general guidance to the agent designer, since they abstract away from the specifics of particular agent-environment systems. However, many interesting logical properties of agents depend on the agent's architecture and program and the environment in which it is embedded. In addition, from a technical point of view, existing work often makes strong assumptions which can limit its applicability when considering feasible (i.e., implementable) agent designs, e.g., the assumption that agents are logically omniscient.

In this paper, we propose a logical framework for modelling agent-environment systems. We adopt an explicitly design-oriented view in the sense that our framework makes only minimal assumptions about agents in general, and those assumptions we do make are motivated by consideration of feasible agent designs. For example, we assume that all feasible agents will have a finite state and will require time to perform inferences (i.e., they are not logically omniscient).

The remainder of the paper is organised as follows. In sections 2 and 3 we present our modelling framework and develop a model of a simple agent-environment system within the framework. In section 4 we introduce a new logic based on these ideas, OBA, which can be used to model a resource-bounded agent's observations, beliefs, goals and actions, and state some complexity results for OBA. In section 5 we illustrate our approach with a simple example based on the well known Tileworld domain and show how OBA can be used to specify properties of the Tileworld agent. In section 6 we discuss related work, and in section 7 we conclude with some ideas for future work.

2 The Agent-Environment Model

In this section we present a logical framework for modelling agent-environment systems based on state transition systems. We first describe how to ascribe beliefs to agents on the basis of the contents of their state and then outline how to specify properties of agent-environment systems on the basis of transitions between states.

The state of an agent-environment system, w, consists of two parts: the environment state $e(w)$ and the agent's state $s(w)$ [1]. The environment state is a description of the (physical or computational) environment "inhabited" by the agent. The agent's state contains all the internal representations which determine the behaviour of the agent. We assume that some parts of the agent's state can be interpreted as referring to (real or hypothetical) objects or events in the environment, e.g., that there is an obstacle dead ahead, or to properties of the agent itself, e.g., the level of the agent's battery. Other parts of the agent's state are not objectively correlated with the outside world and are needed only for the agent's internal functioning, such as the agent's goals, or things like program counters and stack. The combined agent-environment states are assumed to be finite and we assume that there are finitely many of them.

We describe the properties of the environment in a language built from a set of propositional variables $\mathcal{P}$. A finite subset of $\mathcal{P}$, $\mathcal{P}_b = \{(\neg)p_1, ..., (\neg)p_m\}$ corresponds to the beliefs ascribable to the agent. We assume that the agent's state consists of finitely many locations $\mathbf{l_1}$,...,$\mathbf{l_n}$, and that each location $\mathbf{l_i}$ can contain (exactly) one of finitely many values, $v_{i_1}, \ldots, v_{i_k}$. For example, we could have a location $\mathbf{l_t}$ for the output of a temperature sensor which may take an integer value between -50 and 50. Based on those values, we can ascribe beliefs about the external world to the agent: for example, based on $\mathbf{l_t}$ = 20 we ascribe to the agent a belief that the outside temperature is 20 C. Each proposition $(\neg)p_i \in \mathcal{P}_b$ corresponds to a set of values for a given location $\mathbf{l_i}$ (or set of locations), but 'translates' this into a statement about the world. We assume a mapping Bel assigning to each state $s(w)$ of the agent a set of propositional variables and their negations which form beliefs of the agent in state $s(w)$. Note that this 'translation' is fixed and does not depend on the truth or falsity of the propositions in the real world. In general, there is no requirement that $Bel(s(w))$ be consistent; if a propositional variable and its negation are associated with two different locations (e.g., in an agent which

[1] For simplicity, in this paper we consider only a single agent.

has two temperature sensors) then the agent may simultaneously believe that p and $\neg p$. $Bel(s(w))$ does not have to map a single value to a single belief, for example, all values of $\mathbf{l_t} > 20$ could be mapped to a single belief that it's "warm". Conversely, we do not assume that for every propositional variable p, either p or $\neg p$ belong to $Bel(s(w))$; if a location $\mathbf{l_i}$ has no value (e.g., if a sensor fails) or has a value that does not correspond to any proposition, then the agent may have no beliefs about the outside world at all.

Other intentional notions such as goals can be modelled analogously to beliefs, i.e., by introducing an explicit translation from the contents of the agent's state into the set of goals. We may also ascribe arbitrary formulas to the agent as beliefs or goals, so long as we do not ascribe to it any a priori ability to infer consequences or otherwise manipulate those formulas.

Our aim is to model the transitions of the agent-environment system as a kind of Kripke structure and express properties of the agent in some modal logic. The choice of the set of transitions depends on the system we want to model and the properties we wish to verify.

The most basic structure which we could consider is similar to, for example, the interpreted systems of [9] except that the agent's beliefs are modelled as a local property of the agent's state. To be more precise, the structure consists of a set of agent-environment states W, each state $w \in W$ has an agent part $s(w)$ and an environment part $e(w)$. The state of the agent $s(w)$ comes equipped with a set of formulas $Bel(s(w))$ corresponding to agent's beliefs; the state of the environment $e(w)$ corresponds to a classical possible world, or a complete truth assignment to propositional variables in $\mathcal{P}$. The agent believes that p in state w, $w \models Bp$, if $p \in Bel(s(w))$. Note that this truth definition for B does not give rise to any interesting logical properties of B, e.g. to KD45 axioms. This is intentional: we do not want our agents to be logically omniscient and the logical properties of agent's beliefs should be determined by the agent's architecture and program.

If we restrict ourselves to this basic set up, we can express some properties of the agent in a suitable logic, such as CTL ([4]), and use a standard model checker such as SMV to check whether the properties are true of the agent-environment system. For example, we may want to check whether the agent always has true beliefs about a property p, i.e., whether $AG(Bp \rightarrow p)$. The set of states is generated by the agent's program together with appropriate environment responses. Note that the check for beliefs is done locally in state w and consists of checking whether a given formula is in the set $Bel(s(w))$, so the problem can be easily solved by a standard model checking techniques (e.g., with beliefs encoded as propositional variables).

However such an approach is too coarse grained even for analysing simple agents. For example, if p is a property observable by the agent and observation of p is always reliable, then the agent will have true beliefs about p immediately after it has performed an observation. However, this does not guarantee that the agent will always have true beliefs about p in every state—if the agent doesn't sense its environment continuously, it may have false beliefs about p in some states, for example, in states resulting from a change in the environment. We want to be able to express the fact that in all states resulting from the agent performing an observation, Bp is true only if p is true. The most natural way to express this is to label transitions of the system by 'moves' of the agent and the environment.

In the remainder of this paper we work out this idea for a simple agent-environment system in which the agent senses the environment, updates its state, and sends an action to the environment, which in turn changes according to (one of the outcomes of) the action. Note that we could have easily considered a different set of transitions, for example, we could have introduced finer grain in the agent's state update transition to model the agent's deliberation more precisely.

3 A Simple Agent-Environment System

In what follows we view the agent and its environment as a pair of interacting automata (cf [19]) (see Figure 1). At each cycle, the environment generates an input or 'percept' to the agent followed by the agent generating an output or 'action' to the environment. We assume that the environment "computes" its response to the agent's action instantaneously and that the agent produces its response to the percept from the environment in bounded time. We model perception and action as two non-deterministic transitions, obs and act. An obs transition takes the state of the environment and updates the agent's state with a percept corresponding to the information returned by the agent's sensors. We assume that perception either returns accurate information about the environment or one of finitely many outcomes of failed perception. The action transition act computes the new state of the environment given the current state of the environment and the action selected by the agent. An action either causes the environment to change in the desired way (the action succeeds), or results in one of finitely many outcomes for a failed action. If the response generated by the environment depends on both the agent's action and the amount of time it took the agent to produce it, this approximates an asynchronous interaction between the agent and a dynamic environment. Intuitively, if the agent spends too much time selecting an action, then performing it does not produce the expected result.

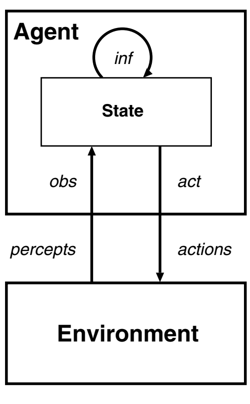

Fig. 1. Simple agent-environment system

The agent consists of some state and an internal transition inf which models the computation the agent uses to update its state (e.g. update its beliefs and select an action to perform). It is assumed to depend on the agent's percept at this cycle and its state from the previous cycle and to terminate in bounded time.

As above, we describe the properties of the environment in a language built from a set of propositional variables $\mathcal{P}$. In addition to the mapping Bel introduced in the previous section, we define a mapping Obs which takes the agent's state $s(w)$ and returns a set of formulas, as a restriction of Bel to the locations holding percepts. Since observational beliefs are a subset of all beliefs, for every agent state $s(w)$ we have $Obs(s(w)) \subseteq Bel(s(w))$. Analogously we ascribe a set of goals $Goal(s(w))$ to the agent based on designer-specified translation of the agent's state.

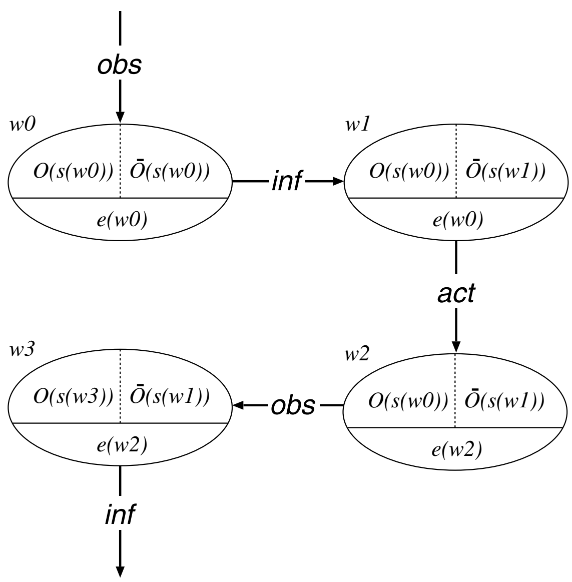

Fig. 2. World Transitions (where $O(s(w)) = Obs(s(w))$ and $\bar{O}(s(w)) = Bel(s(w)) \setminus Obs(s(w)))$

The combined agent-environment automaton cycles through the three transitions: obs, inf and act (see Figure 2). At each cycle the environment updates the agent's state $s(w)$ with its percept for this cycle. The agent then derives the consequences of its new beliefs (if any) and performs an action on the basis of this new state. The agent's observational beliefs, $Obs(s(w))$, only change after the obs transition and the rest of the agent's beliefs and goals only change after the inf transition. We assume that inf does not change the agent's observational beliefs $Obs(s(w))$, as these correspond to the agent's "sensor readings", and, as such, should not be 'overwritten' by inference.[2] We further assume that the environment does not change while the agent executes its program, so $e(w)$ only changes after the act transition (which doesn't change the agent's state).

Introducing three separate transitions rather than collapsing them into one single step (corresponding to a full obs, inf and act cycle) allows more precise modelling of changes in the agent's beliefs and intentions. We want to be able to talk about the state of the agent after it has performed the observation but before it completed all the planning necessary to choose an action, or after it has selected the action but before it has performed it. This is essential in analysing a resource-bounded agent which may be capable of selecting the "right" action to perform but is unable to do so before the environment changes so as to invalidate the action's preconditions.

4 A Logic for Agent-Environment Systems

In this section, we give a formal definition of the transition graphs of the agent-environment system described in the previous section, and a logical language to reason about the agent's observations, beliefs and goals.

We describe transition graphs in a logic which we call OBA (for 'Observation, Belief and Action'). OBA includes PDL (Propositional Dynamic Logic, see [17]) with three atomic transition labels o, i and a corresponding to obs, inf and act, and in addition contains observation, belief and goal operators. We define a transition term x as $o|i|a|x;y|x^*$, where if x and y are transition terms then $x;y$ is their sequential composition (transition x followed by y) and x^* stands for 0 or finitely many iterations of x.

In addition to the set of propositional variables $\mathcal{P}$ and boolean connectives, the language of OBA contains:

- for each transition term x, a unary modality $\langle x \rangle$. $\langle x \rangle \phi$, where ϕ is any formula, stands for 'from here we can make an x transition after which ϕ holds'. A dual modality $[x]\phi$ is defined as $\neg\langle x \rangle \neg\phi$ and stands for 'after all x transitions we can make from here, ϕ holds'.
- unary modalities O, O' and G, which can only be applied to propositional variables or their negations. $O\phi$ stands for 'the agent observes that ϕ', $O'\phi$ stands for 'the agent has a non-observational belief ϕ', and $G\phi$ stands for 'the agent intends that ϕ' (in all three cases, ϕ is a propositional variable or its negation).

The belief modality B is defined as follows: $B\phi =_{df} O\phi \vee O'(\phi)$.

In the models below we only allow legal states of the system as possible worlds, i.e., states which are possible given the characteristics of environment and the agent's program.

Definition 1. *An OBA model M is a tuple* $\langle W, V, Obs, Bel, Goal, obs, inf, act \rangle$ *where*

[2] This limitation is not overly restrictive: the obs transition is non-deterministic (perception is not guaranteed to veridical) and the agent may still derive incorrect beliefs about the world on the basis of its percepts.

W is a set of worlds. Each world w is a pair $(s(w), e(w))$ where $s(w)$ is the state of the agent and $e(w)$ the state of the environment.

$Bel(s(w))$ is the finite set of beliefs (elements of $\mathcal{P}$ or their negations) associated with $s(w)$;

$Obs(s(w)) \subseteq Bel(s(w))$ is the set of observational beliefs;

$Goal(s(w))$ is the finite set of goals (elements of $\mathcal{P}$ or their negations) associated with $s(w)$;

V assigns true *or* false *to pairs $(e(w), p)$ where $e(w)$ is a state of the environment and $p \in \mathcal{P}$.*

obs, inf, act are binary relations on W. To say that a world v is reachable from a world w by an atomic transition x (e.g. obs) we will use notation $w \xrightarrow{x} v$ (e.g. $w \xrightarrow{obs} v$).

The following restrictions on obs, inf and act hold:

Availability *For every w, exactly one of the following is true: $\exists v(w \xrightarrow{obs} v)$, $\exists v(w \xrightarrow{inf} v)$, $\exists v(w \xrightarrow{act} v)$*

Order *On all paths in the transition system, atomic transitions succeed each other in the order $obs; inf; act$.*

Change(obs) *If $w \xrightarrow{obs} w'$ then $Goal(s(w)) = Goal(s(w'))$, $Bel(s(w)) \backslash Obs(s(w)) = Bel(s(w')) \backslash Obs(s(w'))$ and $e(w) = e(w')$. (Observation may change the percepts $Obs(s(w))$ but not the environment, non-observable beliefs or goals. Note that we are assuming that the same literal cannot be both an observational and a non-observational belief.)*

Change(inf) *If $w \xrightarrow{inf} w'$ then $Obs(s(w)) = Obs(s(w'))$ and $e(w) = e(w')$. (Inference does not change the environment or percepts.)*

Change(act) *If $w \xrightarrow{act} w'$ then $s(w)) = s(w')$. (The agent's state does not change as a consequence of action.)*

A formula ϕ is true at a world w in M ($M, w \models \phi$) if

- $M, w \models p$ *iff* $V(e(w), p) = true$;
- $M, w \models \neg\psi$ *iff* $M, w \not\models \psi$
- $M, w \models \psi \wedge \chi$ *iff* $M, w \models \psi$ *and* $M, w \models \chi$
- $M, w \models \langle o \rangle \psi$ *iff there exists a world w' such that $w \xrightarrow{obs} w'$ and $M, w' \models \psi$.*
- $M, w \models \langle i \rangle \psi$ *iff there exists a world w' such that $w \xrightarrow{inf} w'$ and $M, w' \models \psi$.*
- $M, w \models \langle a \rangle \psi$ *iff there exists a world w' such that $w \xrightarrow{act} w'$ and $M, w' \models \psi$.*
- $M, w \models \langle x; y \rangle \psi$ *iff there exists a world w' which is reachable from w by a transition x followed by a transition y and $M, w' \models \psi$.*
- $M, w \models \langle x^* \rangle \psi$ *iff there exists a world w' such that either $w = w'$ or there exists a path consisting of x transitions from w to w', and $M, w' \models \psi$.*
- $M, w \models O\psi$ *iff* $\psi \in Obs(s(w))$;
- $M, w \models O'\psi$ *iff* $\psi \in Bel(s(w)) \setminus Obs(s(w))$;
- $M, w \models G\psi$ *iff* $\psi \in Goal(s(w))$;

A formula ϕ is OBA-satisfiable if there is a OBA model M and a world w in M such that $M, w \models \phi$. A formula ϕ is OBA-valid ($\models_{OBA} \phi$) if its negation is not satisfiable.

Theorem 1. *The axiom system below is weakly complete and sound for OBA, that is, for every formula ϕ,*

$$\vdash_{OBA} \phi \text{ iff } \models_{OBA} \phi.$$

$\vdash_{OBA} \phi$ *stands for ϕ is either one of the axioms below or is obtained from axioms by application of inference rules given below:*

Classical propositional logic:
CL *Axioms of classical propositional logic;*
MP $\vdash \phi,\ \phi \rightarrow \psi \implies \vdash \psi$
Observational and non-observational beliefs are disjoint:
O $\neg(O\phi \wedge O'\phi)$
Normal modal logic axioms for $[x]$, where x is any transition
K $[x](\phi \rightarrow \psi) \rightarrow ([x]\phi \rightarrow [x]\psi)$
N $\vdash \phi \implies \vdash [x]\phi$
Axioms for composition and iterations
C $[x;y]\phi \leftrightarrow [x][y]\phi$
It1 $\langle x^*\rangle\phi \leftrightarrow (\phi \vee \langle x\rangle\langle x^*\rangle\phi)$
It2 $[x^*](\phi \rightarrow [x]\phi) \rightarrow (\phi \rightarrow [x^*]\phi)$
Availability and order of transitions
T1 $(\langle o\rangle\top \wedge \neg\langle i\rangle\top \wedge \neg\langle a\rangle\top) \vee (\langle i\rangle\top \wedge \neg\langle o\rangle\top \wedge \neg\langle a\rangle\top) \vee (\langle a\rangle\top \wedge \neg\langle i\rangle\top \wedge \neg\langle o\rangle\top)$
T2 $[o]\langle i\rangle\top \wedge [i]\langle a\rangle\top \wedge [a]\langle o\rangle\top$
Changes after transitions
C(obs) $\phi \rightarrow [o]\phi$*, where ϕ does not contain subformulas of the form $O\psi$*
C(inf) $\phi \rightarrow [i]\phi$*, where ϕ does not contain subformulas of the form $O'\psi$ or $G\psi$*
C(act) $X\phi \rightarrow [a]X\phi$*, where $X \in \{O, O', G\}$.*

Note that there are no axioms connecting beliefs and goals, in particular no requirement that the agent does not intend what it already believes (cf [18]). We will see in the next section that it is possible for a resource bounded agent to both believe and intend ϕ, e.g., while it is updating its goals.

Proof. To prove soundness, we need to show that all axioms are valid and rules preserve validity. The argument for standard axioms and rules such as **CL, MP, K, N, C, It1, It2** is omitted. The axiom **O** is valid because no belief ϕ can belong both to $O(s(w))$ and to $B(s(w)) \setminus Obs(s(w))$. The axiom **T1** is valid because from every state w, exactly one of the three transitions is possible. The axiom **T2** is valid because if it is possible to make an x transition from w then from the resulting state it is possible to make a y transition, provided $(x, y) \in \{(o, i), (i, a), (a, o)\}$. Validity of **C($obs$)** follows from **Change(obs)**. We can show by induction on subformulas χ of ϕ built from propositional variables and subformulas of the form $O'\psi$ and $G\psi$ that $M, w \models \chi$ iff $M, w' \models \chi$ where w' is accessible from w by obs. Analogously for **C(inf)** and **C(act)**.

Now we prove completeness. Let ϕ be an OBA-consistent formula. We will construct a finite satisfying model for ϕ in a standard way (see for example [3]). Let $Cl(\phi)$ be the Fisher-Ladner closure of the set of subformulas of ϕ together with three formulas $\langle o\rangle\top$, $\langle i\rangle\top$ and $\langle a\rangle\top$. The satisfying model for ϕ, M_{phi}, is defined as follows:

The set of worlds W_ϕ is the set of all maximal consistent subsets of $Cl(\phi)$

$V_\phi(e(w), p) = true$ iff $p \in w$
$Obs_\phi(s(w)) = \{\psi : O\psi \in w\}$
$Bel_\phi(s(w)) = Obs_\phi(s(w)) \cup \{\psi : O'\psi \in w\}$
$Goal_\phi(s(w)) = \{\psi : G\psi \in w\}$
$w \xrightarrow{obs} v$ in M_ϕ iff the conjunction of formulas in w, which we denote by $\hat{w}$, is consistent with $\langle o \rangle \hat{v}$, similarly for inf and act.

For every regular expression π composed from obs, inf and act using sequential composition and finite iteration, we define the corresponding accessibility relation as $S_\pi(w, v)$ iff the formula $\hat{w} \wedge \langle \pi \rangle \hat{v}$ is OBA-consistent. The proof that relations defined this way do indeed correspond to sequential composition and finite iteration, as well as the existence lemma and the truth lemma, are standard. Note that the restriction on the language ensures that $Bel_\phi(s(w))$ and $Goal_\phi(s(w))$ only contain propositional variables and their negations. The only slightly tricky clause in the truth lemma is showing that $O'\psi \in w$ iff $\psi \in Bel_\phi(s(w)) \setminus Obs_\phi(s(w))$. Let $O'\psi \in w$. Then $\psi \in Bel_\phi(s(w))$ by the definition of $Bel_\phi(s(w))$. By axiom **O**, $\psi \notin Obs_\phi(s(w))$. Hence, $\psi \in Bel_\phi(s(w)) \setminus Obs_\phi(s(w))$. The other direction: suppose $\psi \in Bel_\phi(s(w)) \setminus Obs_\phi(s(w))$. Then $\psi \notin Obs_\phi(s(w))$. However, ψ is in $Bel_\phi(s(w))$ hence $O'\psi \in w$ by the definition of $Bel_\phi(s(w))$.

We need to show that the special conditions on the OBA models hold: namely, that the transitions follow each other in the right way, and change the world in the way constrained by **Availability**, **Order**, **Change(**obs**)**, **Change(**inf**)** and **Change(**act**)**.

Let us first show that in each world, exactly one of the transitions obs, inf, act is possible. Note that the conjunction of formulas in each world is consistent with exactly one of $\langle o \rangle \top$, $\langle i \rangle \top$ and $\langle a \rangle \top$ (because the world is maximally consistent with respect to **T1** and contains some $\langle x \rangle \top$). This means that $\hat{w} \wedge \langle x \rangle \top$ for some x is consistent, and $\top$ can be expanded into a maximally consistent conjunction of formulas $\hat{v}$ from $Cl(\phi)$ by forcing choices; this gives us a world v accessible by x from w. For all other atomic transitions y, there is no such world v, because if $\hat{w} \wedge \langle y \rangle \hat{v}$ is consistent, then $\hat{w} \wedge \langle y \rangle \top$ is, and w inconsistent with **T1**.

Next, let us show that the **Order** property holds. Just as an example, we show that obs cannot be followed by act: proofs for the other conditions are similar. Suppose some $\hat{w}$ is consistent with $\langle o \rangle \hat{v}$ and $\hat{v}$ is consistent with $\langle a \rangle \hat{u}$. Note that in this case $\hat{v}$ contains $\langle a \rangle \top$. So it follows that $\langle o \rangle \langle a \rangle \top$ is consistent, but this contradicts **T2**.

Finally, let us show that **Change(**obs**)** holds (other conditions are analogous). We need to show that whenever $\hat{w} \wedge \langle o \rangle \hat{v}$ is consistent, the state of the environment is the same in w and v, that is for every propositional variable p, $p \in w$ iff $p \in v$. Suppose $p \in w$ but $p \notin v$. Then it follows that $p \wedge \langle o \rangle \neg p$ is consistent, but it contradicts **C(**obs**)**. Similarly for the case $p \notin w$ but $p \in v$. Non-observational beliefs are the same in w and v, that is $Bel_\phi(s(w)) \setminus Obs_\phi(s(w)) = Bel_\phi(s(v)) \setminus Obs_\phi(s(v))$: this follows from $O'\psi \in w$ iff $O'\psi \in v$, by **C(**obs**)**. Similarly for the set of goals in w and v.

Theorem 2. *The satisfiability problem for OBA is decidable.*

Proof. Decidability follows from the bounded model property for OBA which can be established as a by-product of Theorem 1: every satisfiable formula ϕ has a model of size at most $2^{|\phi|}$. This gives the following NEXPTIME decision procedure: guess a model of size $2^{|\phi|}$ and check whether ϕ is true there.

In fact, the complexity of the decision procedure can be improved to EXPTIME analogously to PDL (see [3]).

Theorem 3. *Given a formula ϕ and a model, state pair M, w there is an O($|M| \times |\phi|$) algorithm for checking whether $M, w \models_{OBA} \phi$.*

Proof. This follows from the result on complexity for model checking for PDL ([8]). In addition to PDL modalities we have modal operators O, O' and G in the language, but checking whether $O\psi$, $O'\psi$ or $G\psi$ is true in a world w can be done in constant time (subformulas of the form $O\psi$, $O'\psi$ and $G\psi$ can be treated as propositional variables).

In OBA, we can express properties of a particular agent design and verify whether they hold as expected. There are two possible ways of doing this. We can axiomatise the agent's program and the interaction between the agent and its environment, and prove that the property follows from those extra axioms and the OBA axioms. Alternatively, given the agent's program and all possible environment's responses to the agent's actions, we can generate the state transition graph and check whether the property is true in all states in the graph. Both approaches are feasible given the decidability and low cost of model checking of OBA.

5 An example: the Tileworld

In this section we illustrate our approach with an example based on the well-known Tileworld domain [16]. For reasons of brevity, our Tileworld is very simple. The environment consists of an unbounded rectangular grid. Each square in the grid contains either a tile or a hole or is empty, and tiles and holes are distributed randomly throughout the grid.[3] The environment contains a single agent which can move forward on rails laid over the squares and has a sensor which allows it to see whether the square directly underneath it contains a tile or a hole or is empty. If it sees a tile it can pick it up and carry it. If it is holding a tile and is located above a hole it can drop it in the hole. Holes are one tile deep and dropping a tile into a hole turns the hole into an empty square. For simplicity we have assumed that the agent's sensors are veridical in the sense that they either return correct information about the environment or an 'undefined' value indicating the sensor returned no data at this cycle. We also assume that the actions of picking up or dropping a tile can fail, leaving the world unchanged; moving forward always succeeds and leaves the agent above the next square. We stress that these assumptions are not essential, and OBA can be used to reason about more realistic agents with unreliable perception and more complex actions failures.

The goal of the agent is put a tile in a hole. Informally, the program of the Tileworld agent consists of the following simple rules:

- if holding a tile and above a hole, repeatedly try to drop the tile until the tile is in the hole;

[3] Unlike the original Tileworld [16], there are no obstacles and tiles and holes do not appear and disappear randomly, i.e., the environment is static.

– if not holding a tile and above a tile, repeatedly try to pick it up until the square below the agent is empty;
– otherwise, move forward.

The agent's state consists of four locations, $\mathbf{p}$, $\mathbf{g}$, $\mathbf{h}$, and $\mathbf{a}$:

$\mathbf{p}$ is for storing percepts, and can take one of four values: 0 for the agent is above a tile, 1 for the agent is above a hole, 2 for the agent is above an empty square and 3 for 'undefined'.
$\mathbf{h}$ is for storing information about whether the agent is holding a tile: 1 for the agent is holding a tile, and 0 for the agent is not holding a tile.
$\mathbf{g}$ is for storing goals, and can take one of four values: 0 for looking for a tile, 1 for picking up a tile, 2 for looking for a hole, and 3 for dropping a tile.
$\mathbf{a}$ is for storing selected actions, and can also take one of four values: 0 for pick up the tile beneath the agent, 1 for drop the tile the agent is holding, 2 for move forward one grid cell, and 3 for a special 'no-op' action which does not change the environment.

The execution cycle of the agent then becomes:

obs : sensing the environment to find out what is underneath the agent and updating the $\mathbf{p}$ location;
inf : updating the values of $\mathbf{h}$, $\mathbf{g}$ and $\mathbf{a}$ according to the agent's program; and
act : sending the value in $\mathbf{a}$ to the environment.

To ascribe beliefs and intentions to the Tileworld agent, we do not need a distinct propositional variable for every combination of location and value. The set $\mathcal{P}_{TL} = \{p_0, p_1, p_2, h, a_0, a_1, a_2, a_3\}$ is sufficient to capture those beliefs and goals which are relevant to the choice of the agent's actions. Due to lack of space we do not specify Obs, Bel and $Goal$ completely, but illustrate them via examples. Let p_0 mean 'the agent is above a tile'.

$p_0 \in Obs(s(w))$ iff in $s(w)$, $\mathbf{p}$ = 0 (the agent observes that it is above a tile); similarly for p_1 and p_2.
$\neg p_0 \in Obs(s(w))$ iff in $s(w)$, $\mathbf{p}$ = 1 or $\mathbf{p}$ = 2 (the agent observes that it is not above a tile); similarly for $\neg p_1$ and $\neg p_2$.
$p_0 \in Goal(s(w))$ iff in $s(w)$, $\mathbf{g}$ = 0 (the agent intends to be above a tile); similarly for p_1 and $\mathbf{g}$ = 2.

Note that it is possible for example for neither p_0 nor $\neg p_0$ to be in $Obs(s(w))$ if the agent's sensors have failed ($\mathbf{p}$ = 3).

The agent's program is given by a series of rules which map from agent states to agent states. For example, the rule:

$$\mathbf{p} = 1, \mathbf{h} = 1, \mathbf{g} = 2, \mathbf{a} \neq 0 \Longrightarrow \mathbf{p} = 1, \mathbf{h} = 1, \mathbf{g} = 3, \mathbf{a} = 1$$

states that if the agent believes that it is above a hole, is holding a tile, and has a goal of looking for a hole, then the agent should drop the tile. Other rules detect the effects of successful actions, failed actions and sensor failures. For example, if the agent perceives an empty square after attempting to drop a tile, it updates its beliefs to record the fact that it is no longer holding a tile, and its goals to "looking for a tile", and selects a "move forward" action. If the agent perceives a hole after attempting to drop a tile, (i.e., the drop action failed and the agent is still holding the tile), then it does not update its beliefs and goals and repeats the drop action. If the agent's sensors return no data at this cycle, its beliefs and goals do not change and it selects the no-op action and so on.

OBA can be used to specify properties of the Tileworld agent that a designer may want to verify. For example, we can state that the agent will achieve a particular goal, e.g., finding a tile: $[(a; o; i)^*](Gp_0 \rightarrow \langle (a; o; i)^* \rangle p_0)$ ('if the agent intends to be above a tile then after finitely many cycles it will be above a tile'). This statement holds in all states provided the sensors are veridical, sensor failures only last for finitely many cycles and it is always possible to reach a tile after finitely many moves forward. The veridicality of sensors can be expressed as $[o](O\phi \rightarrow \phi)$ (note that $O\phi \rightarrow \phi$ is only guaranteed to hold after an obs transition). Another important property is commitment to goals: if the agent has a goal ϕ, it will not give it up until ϕ becomes true: $[(a; o; i)^*](G\phi \rightarrow [a; o; i](\phi \vee G\phi))$. As we mentioned in the previous section, it is possible for the agent both to believe and intend the same formula: for example, if the agent's goal is to be above a tile and it has performed a move forward action which brought it above a tile, then after a successful obs transition it will believe that it is above a tile (Op_0 hence Bp_0)[4], but still have the goal of being above the tile (the goals will be updated after the inf transition); hence $B\phi \rightarrow \neg G\phi$ is not valid.

6 Related Work

Our approach has some similarities with the situated automata work of Rosenschein and Kaelbling [19]. Rosenschein and Kaelbling model an agent using two functions, a state update function, f, which maps the input (percepts) and the value of the agent's internal state at the last cycle into a new value of the internal state, and an output function, g, that maps the input and the value of the agent's internal state at the last cycle into the output (actions). They ascribe belief (or rather knowledge) to the agent by associating with every combination of location/value pair ($\mathbf{l}$,v) the most informative proposition ϕ such that in all runs of the system when $\mathbf{l} = v$ in the agent's state, ϕ holds. This forces the agent to have true beliefs and its beliefs are closed under logical consequence. The resulting notion corresponds to implicit knowledge as defined in [9], or the information implicitly encoded in the state, and satisfies the properties of an S5 modality. van der Hoek, Linder and Meyer in [21] formalise all relevant aspects of the agent's architecture and use PDL to reason about state transitions as we do. However they model agents as logically omniscient, consider only deterministic actions, and model knowledge and beliefs using an additional Kripke structure associated with every global state. In contrast, our approach combines the modelling of belief change and action performance in a single structure and is much simpler, giving an efficient model-checking procedure.

A compact modular representation of agent-environment systems has been recently given in [10]; perhaps our logic can be interpreted on such systems rather than on a combined transition graphs in the future, especially for the multiagent case.

[4] Recall that $B\phi =_{df} O\phi \vee O'\phi$.

Step logic [7] provides a mechanism for reasoning about non-omniscient time-bounded reasoners; agent's beliefs are indexed by time points or steps, corresponding to stages in the agent's reasoning. Other recent approaches that avoid the problem of logical omniscience and model agent's beliefs as syntactic objects are Duc's dynamic epistemic logic [5, 6] and Ågotnes's logic of finite syntactic epistemic states [1].

In other work, we have applied a syntactic approach to modelling agent's beliefs to verifying properties of an agent implemented in a simplified version of the 3APL agent programming language [2]. However, in that work we made a simplifying assumption that the agent's beliefs are veridical and actions successful, to avoid explicitly modelling the agent's environment. The current paper provides a way to overcome this limitation.

7 Summary and Future Work

In this paper, we propose a new approach to the formal modelling of agent-environment systems which focuses on particular agent designs. Properties of the agent-environment system depend on the agent's architecture and program and the characteristics of its environment rather than on a priori assumptions about agents in general, e.g., that agents are rational or logically omniscient. We show how to ascribe beliefs and other intentional modalities based on designer-stipulated correlations between the values of locations in the state of the agent and the state of the environment, and how to model operations within the agent and interactions between agent and environment as state transitions. Our approach to belief ascription is local in the sense that belief is not defined in terms of all possible runs of the agent-environment system (cf. [9]) and the set of transitions can be tailored to the system we want to model and the properties we wish to specify.

Drawing on these ideas, we define a new logic, OBA, which can be used to model a resource bounded agent's observation, beliefs, goals and actions. We prove completeness and decidability results for OBA and show that it has a reasonable model checking procedure ($O(|M| \times |\phi|)$, where $|M|$ is the size of the model and $|\phi|$ the size of the formula). OBA allows for both failed perception and failed actions, and the agent's beliefs are not required to be consistent or closed under logical consequence. We explicitly introduce the cycle of the agent in the logic and analyse the agent's beliefs at every point in the cycle, allowing us to distinguish the agent's beliefs, e.g., after a percept is received but before the rest of the state is updated. As an illustration, we model a simple Tileworld agent and express some properties of the resulting agent-environment system in OBA. Such properties can be efficiently checked using standard model checking or theorem proving techniques.

In future work we plan to develop the framework outlined above to analyse more complicated agent behaviours, for example, to model trial and error attempts to achieve a goal, or characterise robust behaviours.

References

1. T. Ågotnes. *A Logic of Finite Syntactic Epistemic States*. Ph.D. thesis, Department of Informatics, University of Bergen, Norway, 2004.

2. N. Alechina, M. Dastani, B. Logan, and J.-J. C. Meyer. A logic of agent programs. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence (AAAI 2007)*, pages 795–800. AAAI Press, 2007.
3. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2001.
4. E. M. Clarke and E. A. Emerson. Design and synthesis of synchronization skeletons using branching time temporal logic. In D. Kozen, editor, *IBM Logics of Programs Workshop*, number 131 in LNCS, pages 52–71. Springer-Verlag, May 1981.
5. H. N. Duc. Logical omniscience vs. logical ignorance on a dilemma of epistemic logic. In C. A. Pinto-Ferreira and N. J. Mamede, editors, *Progress in Artificial Intelligence, 7th Portuguese Conference on Artificial Intelligence, EPIA '95, Funchal, Madeira Island, Portugal, October 3-6, 1995, Proceedings*, volume 990 of *Lecture Notes in Computer Science*, pages 237–248. Springer, 1995.
6. H. N. Duc. Reasoning about rational, but not logically omniscient, agents. *Journal of Logic and Computation*, 7(5):633–648, 1997.
7. J. J. Elgot-Drapkin and D. Perlis. Reasoning situated in time I: Basic concepts. *Journal of Experimental and Theoretical Artificial Intelligence*, 2:75–98, 1990.
8. E. Emerson and C.-L. Lei. Efficient model checking in fragments of the propositional mu-calculus. In *Proceedings LICS'86*, pages 267–278, 1986.
9. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, Mass., 1995.
10. W. Jamroga and T. Ågotnes. Modular interpreted systems. In M. Huhns and O. Shehory, editors, *Proceedings of the Sixth International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2007)*, pages 892–899. IFAMAAS, May 2007.
11. G. Lakemeyer. Steps towards a first-order logic of explicit and implict belief. In J. Y. Halpern, editor, *Theoretical Aspects of Reasoning About Knowledge: Proceedings of the 1986 Conference*, pages 325–340, San Francisco, Calif., 1986. Morgan Kaufmann.
12. H. J. Levesque. A logic of implicit and explicit belief. In *Proceedings of the Fourth National Conference on Artificial Intelligence, AAAI-84*, pages 198–202. AAAI, 1984.
13. J. McCarthy. Ascribing mental qualities to machines. Technical report, Stanford AI Lab, 1978.
14. R. C. Moore. *Logic and Representation*. Number 39 in CSLI Lecture Notes. CSLI Publications, 1995.
15. R. Parikh. Knowledge and the problem of logical omniscience. In *Methodologies for Intelligent Systems, Proceedings of the Second International Symposium*, pages 432–439. North-Holland, 1987.
16. M. E. Pollack and M. Ringuette. Introducing the Tileworld: Experimentally evaluating agent architectures. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 183–189, Boston, MA, 1990. AAAI.
17. V. R. Pratt. Semantical considerations on Floyd-Hoare logic. In *Proceedings of the Seventeenth IEEE Symposium on Computer Science*, pages 109–121, 1976.
18. A. S. Rao and M. P. Georgeff. Modeling rational agents within a BDI-architecture. In *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR'91)*, pages 473–484, 1991.
19. S. J. Rosenschein and L. P. Kaelbling. A situated view of representation and control. *Artificial Intelligence*, 73:149–173, 1995.
20. M. P. Singh. Know-how. In M. Wooldridge and A. Rao, editors, *Foundations of Rational Agency*, pages 81–104. Kluwer Academic, Dordrecht, 1999.
21. W. van der Hoek, B. van Linder, and J.-J. C. Meyer. An integrated modal approach to rational agents. In M. Wooldridge and A. Rao, editors, *Foundations of Rational Agency*, pages 133–168. Kluwer Academic, Dordrecht, 1999.

22. M. Wooldridge and A. Lomuscio. A computationally grounded logic of visibility, perception, and knowledge. *Logic Journal of the IGPL*, 9(2):273–288, 2001.