Bounded Epistemic Situation Calculus Theories

Giuseppe De Giacomo
DIAG
Sapienza Università di Roma
Roma, Italy
degiacomo@dis.uniroma1.it

Yves Lespérance
Dept. of Computer Sci. & Eng.
York University
Toronto, Canada
lesperan@cse.yorku.ca

Fabio Patrizi
DIAG
Sapienza Università di Roma
Roma, Italy
patrizi@dis.uniroma1.it

Abstract

We define the class of *e-bounded* theories in the epistemic situation calculus, where the number of fluent atoms that the agent thinks may be true is bounded by a constant. Such theories can still have an infinite domain and an infinite set of states. We show that for them verification of an expressive class of first-order μ-calculus temporal epistemic properties is *decidable*. We also show that if the agent's knowledge in the initial situation is e-bounded and the objective part of an action theory maintains boundedness, then the entire epistemic theory is e-bounded.

Introduction

The Situation Calculus [McCarthy and Hayes, 1969; Reiter, 2001] is a widely used and expressive first-order logical framework for reasoning about action in which many issues have been addressed, e.g., the frame problem, time, continuous change, complex actions and processes, uncertainty, etc. In [De Giacomo *et al.*, 2012], it was shown that for an important class of so called *bounded action theories* in the situation calculus, verification of a very expressive class of first-order μ-calculus temporal properties is *decidable*. Bounded action theories are basic action theories [Reiter, 2001], where it is entailed that in all situations, the number of fluent atoms that are true is bounded by a constant. In such theories, the object domain remains nonetheless infinite, as is the domain of situations. Boundedness may seem like a restrictive condition, but it can be argued that in real domains facts rarely persist indefinitely as everything decays and changes. Moreover, agents often forget facts either because they are not used or because they cannot be confirmed by sensing. [De Giacomo *et al.*, 2012] give many examples of domains that can be modeled as bounded action thories. They also identify various ways in which one can obtain bounded action thories: (1) by strenghtening preconditions to block actions where the bound would be exceeded, (2) by ensuring that actions are *effect bounded* and never make more fluents true than they make false, and (3) by using *fading fluents* whose strength fades over time unless they are reconfirmed.

In this paper, we devise analogous results in the *epistemic situation calculus* [Scherl and Levesque, 2003], where one can refer in the language to what the agent knows and does not know after performing some actions, as well as what is true, and also deal with sensing/knowledge producing actions. It is natural to look at boundedness in this setting, as the intuition for bounded action theories is partly epistemic. We define a notion of *e-bounded* theories where it is entailed that in all situations, the number of fluent atoms that the agent thinks may be true is bounded by a constant. The object domain remains nonetheless infinite, as is the domain of situations. There may be an infinite set of different fluent atoms that the agent thinks may be true over the course of an infinite run, even though in any given situation, the set of fluent atoms that the agent thinks may be true is bounded. We discuss how one can obtain e-bounded theories in practice. Our main result is that for e-bounded theories, verification of a very expressive class of first-order μ-calculus temporal epistemic properties is *decidable*. We also show that e-boundedness of an epistemic action theory can be related to the boundedness of the action theory that the agent uses to model change in the world. We prove that if the agent's knowledge in the initial situation is e-bounded and the objective part of an action theory maintains boundedness (i.e., if the number of fluent atoms that are true initially is bounded it remains bounded in later situations), then the entire epistemic theory is e-bounded.

Our notion of e-bounded theory is based on bounding the number of fluent atoms that the agent considers possible in a situation. This yields a very simple and general notion of epistemic boundedness. It is defined relative to the standard possible worlds semantics for incomplete knowledge, which refers to the possibilities that the agent considers plausible. This accomodates very general forms of partial knowledge, e.g., disjunctive knowledge, existential knowledge, etc. We briefly discuss alternative approaches at the end of the paper. While our study is set in the situation calculus, the insights and results we obtained could be applied to other settings where there is an interplay between knowledge and action.

Epistemic Situation Calculus

The *situation calculus* [McCarthy and Hayes, 1969; Reiter, 2001] is a sorted predicate logic language for representing and reasoning about dynamically changing worlds. All changes to the world are the result of *actions*, which are terms in the language. We denote action variables by lower case letters a and action types by capital letters A. A possible world history is represented by a term called a *situation*. The constant S_0 is used to denote the actual initial situation where no actions have yet been performed. Sequences of actions are built using the function symbol do, where $do(a, s)$ denotes the successor situation resulting from performing action a in situation s. Besides actions and situations, there is also the sort of *objects* for all other entities. Predicates whose value varies from situation to situation are called *fluents*, and are denoted by symbols taking a situation term as their last argument (e.g., $Holding(x, s)$). We denote fluents by F and the finite set of primitive fluents by $\mathcal{F}$. The arguments of fluents (apart from the last which is of sort situation) are assumed to be of sort object. For simplicity, and w.l.o.g., we assume that there are no functions and no non-fluent predicates.

In this setting [Reiter, 2001], one can write successor state axioms that succinctly characterize how the world changes. These imply both effect axioms and frame axioms and provide a solution to the frame problem. A special predicate $Poss(a, s)$ is used to state that action a is executable in situation s, and one can write preconditions axioms to characterize this predicate. The abbreviation $Executable(s)$ means that every action performed in reaching situation s was possible in the situation in which it occured.

To capture how the agent's knowledge changes when she performs sensing, (as well as introspection), we need to model knowledge explicitly in the language. A popular approach for this is to add a possible world account of knowledge to the situation calculus as proposed in [Moore, 1985; Scherl and Levesque, 2003]. This can be done by introducing a special fluent $K(s', s)$ read as situation s' is epistemically accessible fron situation s, i.e., the agent thinks that s' is a possible way the world may be in situation s. Then we can introduce the following definition:

$$\mathbf{Knows}(\phi, s) \doteq \forall s'.K(s', s) \supset \phi[s']$$

i.e., the agent knows that ϕ in situation s iff ϕ holds in all situations s' the agent considers possible in s. Here ϕ is a *situation suppressed* (i.e., with the situation argument in fluents suppressed) epistemic situation calculus formula uniform in its (suppressed) situation argument, where **Knows** may occur. We recall that in essence, a situation calculus formula is *uniform* if it refers only to a single situation [Reiter, 2001]. We denote by $\phi[s]$ the corresponding formula with the situation argument reintroduced and assigned to situation s. To specify the information provided by boolean/binary sensing actions, [Levesque, 1996] introduces a special fluent $SF(a, s)$, which holds if action a returns the binary sensing result 1 in situation s. Using this we can write sensed fluent axioms like

$$SF(senseOpen(door), s) \equiv Open(door, s)$$

i.e., the action $senseOpen(door)$ returns the result 1 in situation s iff $door$ is open in s. (For the non-binary sensing action case, see [Scherl and Levesque, 2003].) One can then specify how the agent's knowledge changes using the following successor state axiom for the K fluent [Scherl and Levesque, 1993; 2003]:

$$\begin{aligned}&K(s'', do(a, s)) \equiv \\ &\quad \exists s'.s'' = do(a, s') \wedge K(s', s) \wedge Poss(a, s') \ \wedge \\ &\qquad [SF(a, s') \equiv SF(a, s)]\end{aligned}$$

i.e., situation s'' is accessible after action a is performed in s iff it is the result of performing a in a situation that was accessible in s and where a was executable and the sensing result agrees with that produced in s.

The above assumes that actions are fully observable (every s' such that $K(s', s)$ involves the same action history as s). [Bacchus *et al.*, 1999] generalizes the framework to accomodate partial observability of actions, noisy sensing, and non-deterministic actions. Instead of sensed fluent axioms, they use observation-indistinguishability axioms of the form:

$$Oi(A(\vec{x}), a', s) \equiv \phi_A(A(\vec{x}), a', s),$$

one for each action type A, where $\phi_A(A(\vec{x}), a', s)$ characterizes the conditions under which actions $A(\vec{x})$ and a' are indistinguishable in situation s. E.g., we could use

$$\begin{aligned}&Oi(pickup(x, r), a, s) \equiv \\ &\quad \exists r' a = pickup(a, r') \wedge (r' = Ok \vee r' = Fail)\end{aligned}$$

to specify that the agent cannot tell whether he succeeds or fails when he attempts to pick up block x. With this, they use the folllowing successor state axiom for the K fluent:

$$\begin{aligned}&K(s'', do(a, s)) \equiv \exists a' \exists s'.s'' = do(a', s') \wedge \\ &\qquad Poss(a', s') \wedge Oi(a, a', s) \wedge K(s', s)]\end{aligned}$$

i.e., situation s'' is accessible after action a is performed in s iff it is the result of performing an action a' that is observation-indistinguishable in s from a in a situation s' that was K-accessible in s where a' is executable.[1] Precondition axioms are then used to relate the result of a possibly noisy sensing action, which must be one of its parameters, to the sensed condition in the actual situation, e.g.

$$\begin{aligned}&Poss(senseIfOpen(door, r), s) \equiv At(door) \wedge \\ &\quad (Open(door, s) \wedge r = 1) \vee (\neg Open(door, s) \wedge r = 0)\end{aligned}$$

Given this language, one can formulate *epistemic basic action theories* [Levesque *et al.*, 1998; Reiter, 2001] that describe how the world state and what the agent knows changes as the result of the available actions. Here, we assume that there is a *finite number of action types*. Moreover, we assume that the terms of object sort are in fact a countably infinite set $\mathcal{N}$ of *standard names* for which we have the unique name assumption and domain closure [Levesque and Lakemeyer, 2001]. We also assume that the accessibility relation K is initially reflexive, transitive and euclidean, and the successor state axiom for K preserves this for all situations. We assume that precondition axioms do not mention K. We also assume that the condition on the right hand side of observation-indistinguishability axioms is a *domain-independent* formula. Domain independent formulas are those for which the evaluation depends only on the active domain: essentially negation is only used to express difference of extensions and not complement. An example of domain-independent formula is $\exists x.Vehicle(x) \wedge \neg Car(x)$, and a domain-dependent variant is $\exists x.\neg Car(x)$. These are standard in the Database literature, see [Abiteboul *et al.*, 1995]. We make use of domain independence in the observation-indistinguishability axioms

[1] In [Bacchus *et al.*, 1999] this need only hold if action a is executable in s; we drop this restriction for simplicity.

to ensure that no new objects are introduced other than action parameters.[2]

Bounded Epistemic Action Theories

Let b be some natural number. We can use the notation $|\{\vec{x} \mid \phi(\vec{x})\}| \geq b$ to stand for the FO formula:

$$\exists \vec{x}_1, \ldots, \vec{x}_b . \phi(\vec{x}_1) \wedge \cdots \wedge \phi(\vec{x}_b) \wedge \bigwedge_{i,j \in \{1,\ldots,b\}, i \neq j} \vec{x}_i \neq \vec{x}_j .$$

We can also define $(|\{\vec{x} \mid \phi(\vec{x})\}| < b) \doteq \neg(|\{\vec{x} \mid \phi(\vec{x})\}| \geq b)$. Using this, [De Giacomo *et al.*, 2012] define the notion of a fluent $F(\vec{x}, s)$ in situation s being *bounded* by a natural number b as $Bounded_{F,b}(s) \doteq |\{\vec{x} \mid F(\vec{x}, s)\}| < b$ and the notion of situation s being bounded by b:

$$Bounded_b(s) \doteq \bigwedge_{F \in \mathcal{F}} Bounded_{F,b}(s).$$

An action theory $\mathcal{D}$ is *bounded* by b if

$$\mathcal{D} \models \forall s. Executable(s) \supset Bounded_b(s).$$

[De Giacomo *et al.*, 2012] show that that for bounded action theories, verification of sophisticated temporal properties is decidable. It also identifies interesting classes of bounded action theories.

Here we devise analogous notions of boundedness for epistemic action theories. We define the notion of a fluent $F(\vec{x}, s)$ in situation s being *e-bounded* by a natural number b as:

$$EBounded_{F,b}(s) \doteq |\{\vec{x} \mid \exists s' K(s', s) \wedge F(\vec{x}, s')\}| < b.$$

Observe that:

$$\{\vec{x} \mid \exists s' K(s', s) \wedge F(\vec{x}, s')\} = \{\vec{x} \mid \neg \textbf{Knows}\ \neg F(\vec{x}, s')\}.$$

Thus fluent $F(\vec{x}, s)$ is e-bounded by b in situation s if the number of distinct $\vec{x}$ such that the agent considers it possible that $\vec{x}$ is in F in s is bounded by b. We define the notion of situation s being e-bounded by a natural number b as follows:

$$EBounded_b(s) \doteq \bigwedge_{F \in \mathcal{F}} EBounded_{F,b}(s).$$

We say that an epistemic action theory $\mathcal{D}$ is *ebounded* by b if

$$\mathcal{D} \models \forall s. Executable(s) \supset EBounded_b(s).$$

We shall see that for e-bounded epistemic action theories, verification of sophisticated properties is decidable.

Example 1 (Wumpus Hunter) Consider a version of the well known Wumpus World [Russell and Norvig, 2010] where there is an agent H whose only task is to hunt wumpus monsters that live in various dungeons (we assume that there are no pits to avoid or gold to pick up, but many dungeons and wumpuses). Each dungeon is laid out as an 8×8 grid. Initially, H is in dungeon $D1$ at location $\langle 0, 0\rangle$ facing $Right$. The agent and wumpus are both alive, and $\langle 0, 0\rangle$ is the only location the agent has visited. The agent knows all of this, which is specified by the following initial state axioms:

$$\begin{array}{l}
\textbf{Knows}\ (DungeonH(d) \equiv d = D1, S_0) \\
\textbf{Knows}\ (LocH(l) \equiv l = \langle 0,0\rangle, S_0) \\
\textbf{Knows}\ (DirH(d) \equiv d = Right, S_0) \\
\textbf{Knows}\ (AliveW, S_0) \qquad \textbf{Knows}\ (AliveH, S_0) \\
\textbf{Knows}\ (Visited(l) \equiv l = \langle 0,0\rangle, S_0)
\end{array}$$

H knows that the wumpus is at some unique location other than $\langle 0, 0\rangle$, but does not know which; in fact he is at $\langle 1, 1\rangle$:

$$\begin{array}{l}
\textbf{Knows}\ (\exists l\, LocW(l), S_0) \\
\textbf{Knows}\ (\forall l \forall l' (LocW(l) \wedge LocW(l') \supset l = l'), S_0) \\
\neg\textbf{Knows}\ (\neg LocW(l), S_0) \equiv ValidLoc(l) \wedge l \neq \langle 0,0\rangle \\
LocW(l, S_0) \equiv l = \langle 1,1\rangle \\
ValidLoc(l) \doteq \bigvee_{0 \leq n,m \leq 7} l = \langle n, m\rangle
\end{array}$$

Now let's specify the world dynamics. The agent can move around by going forward one square in the direction he is facing, provided he has not reached the grid edge, and by turning 90 degrees clockwise (always possible). The following successor state axioms and precondition axioms specify this:

$$\begin{array}{l}
LocH(l, do(a, s)) \equiv a = moveFwd\, \wedge \\
\quad \exists d \exists l' (DirH(d, s) \wedge LocH(l', s) \wedge Adjacent(l', d, l)) \\
\quad \vee LocH(l, s) \wedge a \neq moveFwd
\end{array}$$

$$\begin{array}{l}
DirH(d, do(a, s)) \equiv \\
\quad a = turn \wedge (DirH(Up, s) \wedge d = Right \vee \ldots) \\
\quad \vee DirH(d, s) \wedge a \neq turn
\end{array}$$

$$\begin{array}{l}
Poss(moveFwd, s) \equiv \\
\quad \exists l \exists d \exists l' (LocH(l, s) \wedge DirH(d, s) \wedge Adjacent(l, d, l'))
\end{array}$$

The location of the wumpus never changes except when the agent arrives in a new dungeon; in this case, the wumpus's location is determined by the partially observable exogenous action $arriveInDungeon(d, l)$ (discussed later):

$$\begin{array}{l}
LocW(l, do(a, s)) \equiv \\
\quad \exists d\, a = arriveInDungeon(d, l) \vee LocW(l, s) \\
\qquad \wedge \neg \exists d \exists l' (a = arriveInDungeon(d, l') \wedge l' \neq l)
\end{array}$$

The wumpus is alive when H arrives in a dungeon and remains so until H shoots an arrow forward towards him (always possible):

$$\begin{array}{l}
AliveW(do(a, s)) \equiv \\
\quad \exists d \exists l'\, a = arriveInDungeon(d, l') \\
\quad \vee AliveW(s) \wedge (a \neq shootFwd \vee \exists l' \exists d \exists l' (LocH(l, s) \\
\qquad \wedge DirH(d, s) \wedge LocW(l', s) \wedge \neg Aligned(l, d, l'))
\end{array}$$

Similarly we have an axiom stating that H remains alive unless he moves to a location where there is a live wumpus. $Visited$ records the locations the agent has visited so far; it is reset to $\{\langle 0, 0\rangle\}$ when the agent arrives in a new dungeon:

$$\begin{array}{l}
Visited(l, do(a, s)) \equiv \\
\quad \exists d \exists l'\, a = arriveInDungeon(d, l') \wedge l = \langle 0,0\rangle \vee \\
\quad a = moveFwd\, \wedge \\
\qquad \exists d \exists l' (DirH(d, s) \wedge LocH(l', s) \wedge Adjacent(l', d, l)) \\
\quad \vee Visited(l, s)\, \wedge \\
\quad \neg(\exists d \exists l'\, a = arriveInDungeon(d, l') \vee a = moveFwd)
\end{array}$$

[2] Note that domain independence is also used in the decidability proof, but the assumption is later dropped exploiting well-known results [Libkin, 2007].

H remains in the current dungeon until he exits it, after which he is outside, and then the exogenous action $\mathit{arriveInDungeon}(d, l)$ may occur, after which he is in dungeon d:

$$\begin{array}{l}\mathit{DungeonH}(d, \mathit{do}(a, s)) \equiv \\ \quad \exists l a = \mathit{arriveInDungeon}(d, l) \vee \\ \quad a = \mathit{exitCurDungeon} \wedge d = \mathit{Outside} \ \vee \ \mathit{DungeonH}(d, s) \wedge \\ \quad \neg(\exists l\, a = \mathit{arriveInDungeon}(d, l) \vee a = \mathit{exitCurDungeon})\end{array}$$

H may exit the current dungeon if he is at $\langle 0, 0\rangle$. $\mathit{arriveInDungeon}(d, l)$ may occur if the agent is outside any dungeon, where l is the location where the wumpus will hide; it must be a valid location different from $\langle 0, 0\rangle$:

$$\begin{array}{l}\mathit{Poss}(\mathit{exitCurDungeon}, s) \equiv \mathit{LocH}(\langle 0, 0\rangle) \\ \mathit{Poss}(\mathit{arriveInDungeon}(d, l), s) \equiv \\ \quad \mathit{DungeonH}(\mathit{Outside}, s) \wedge \mathit{ValidLoc}(l) \wedge l \neq \langle 0, 0\rangle\end{array}$$

The action $\mathit{arriveInDungeon}(d, l)$ is only partially observable; the agent knows that he has arrived in dungeon d, but does not know the location l where the wumpus is; i.e., $\mathit{arriveInDungeon}(d, l)$ is observation indistinguishable from $\mathit{arriveInDungeon}(d, l')$. All other actions are fully observable:

$$\begin{array}{l}\mathit{Oi}(\mathit{arriveInDungeon}(d, l), a, s) \equiv \\ \qquad \exists l' a = \mathit{arriveInDungeon}(d, l') \\ \mathit{Oi}(A(\vec{x}), a', s) \equiv a' = A(\vec{x})) \text{ for all } A \neq \mathit{arriveInDungeon}\end{array}$$

Finally, the agent can get information about the wumpus's location by performing the sensing action $\mathit{smell}(r)$; if the wumpus is located in an adjacent location, the agent will get a sensing result r of 1, otherwise the result will be 0:

$$\begin{array}{l}\mathit{Poss}(\mathit{smell}(r), s) \equiv \\ \quad \exists l \exists d \exists l'(\mathit{LocW}(l, s) \wedge \mathit{LocH}(l', s) \wedge \mathit{Adjacent}(l, d, l')) \\ \qquad \wedge r = 1 \vee \\ \quad \exists l \exists l'(\mathit{LocW}(l, s) \wedge \mathit{LocH}(l', s) \wedge \forall d \neg \mathit{Adjacent}(l, d, l')) \\ \qquad \wedge r = 0\end{array}$$

It is easy to show (by induction on executable situations) that this theory is e-bounded by the gridsize g. In S_0, locW is e-bounded by g, as the agent thinks that the wumpus may be in any location other than $\langle 0, 0\rangle$. All the other fluents are e-bounded by 1. Thus $\mathit{EBounded}_g(S_0)$. Suppose that $\mathit{EBounded}_g(s)$ for some executable s. Then it is easy to check that $\mathit{EBounded}_g(\mathit{do}(a, s))$ for any a such that $\mathit{Poss}(a, s)$. All fluents other than $\mathit{Visited}$ and locW remain bounded by 1. $\mathit{Visited}$ is always known and may only become true for grid locations. The set of l that the agent thinks may be locW remains the same if a is neither $\mathit{smell}(r)$ nor $\mathit{arriveInDungeon}(d, l)$. If it is the former, the set shrinks or remains the same. If it is the latter, it is reset to all grid locations other than $\langle 0, 0\rangle$. Thus $\mathit{do}(a, s)$ is e-bounded by g.

While the theory is e-bounded, it should be noted that the number of different dungeons d that the agent may visit and for which **Knows** $(\mathit{dungeonH}(d))$ may come to hold over a path is unbounded. This is not a propositional theory. Also, the agent initially has incomplete knowledge about the location of the wumpus, but can acquire more information about it. As well, note that the agent forgets everything that he knew about the previously visited dungeons when he arrives in a new one. The theory was specified using fluents that implicitly refer to the current dungeon and partially observable actions to achieve this. It would not be hard to make the dungeons that the agent arrives into have a dynamic graph-like layout with a bounded number of named locations. The dungeon structure could be defined using exogenous actions like $\mathit{addLoc}(l)$ and $\mathit{addConnection}(l, l')$. Then the set of locations that the agent comes to know about over a path would no longer be bounded, although we could still ensure that it is e-bounded in any situation as the number of locations in a dungeon remains bounded and the agent only thinks about the current dungeon. $\square$

Relating Boundedness and E-Boundedness

One question that naturally arises is what are interesting sufficient conditions to guarantee e-boundedness. Here we formulate one, a key result relating the boundedness of action theories that model change in the world with the e-boundedness of the entire epistemic theory. Specifically, we show that if an epistemic theory has e-bounded knowledge in the initial situation, and its objective part[3] which describes how the world changes, maintains boundedness [De Giacomo *et al.*, 2012], then the entire theory is e-bounded. We say that an objective theory $\mathcal{D}_{obj}$, with initial situation description $\mathcal{D}_0$, *maintains boundedness*, if for every bounded initial situation description $\mathcal{D}'_0$ we have that $(\mathcal{D}_{obj} - \mathcal{D}_0) \cup \mathcal{D}'_0$ is bounded. Thus:

Theorem 1 *Let $\mathcal{D}$ be an epistemic action theory with objective part $\mathcal{D}_{obj}$ and initial epistemic situation description $\mathcal{D}_{K0}$. If $\mathcal{D}_{obj}$ maintains boundedness and $\mathcal{D}_{K0}$ is e-bounded, then $\mathcal{D}$ is e-bounded.*

Proof (sketch). This holds because if the initial epistemic situation description is e-bounded it can be represented with a bounded number of isomorphic types (cf. Verification Section). From each of these types the successor situations remain bounded, as sensing actions can only reduce the number of epistemic alternatives, and if there are observationally indistinguishable actions, while they may increase the number of epistemic alternatives, given the domain independence condition, they do not introduce new elements into the active domain, but only reshuffle the ones already present, hence do not change the set of isomorphic types. So the entire theory is e-bounded. $\square$

Notice that the actual e-bound for the whole theory can be computed from the e-bound of $\mathcal{D}_{K0}$ and the bound maintained by the objective part $\mathcal{D}_{obj}$. This result, together with the various conditions for boundedness in [De Giacomo *et al.*, 2012], provides effective means for designing e-bounded epistemic theories. For example, if the initial epistemic situation description is e-bounded, and we assume bounded effects

[3] The objective part of epistemic theory is formed by all axioms that do not mention the K, SF and Oi fluents.

or fading fluents (both of which maintain boundedness), then the resulting theory is e-bounded.

Example 2 (Wumpus Hunter Cont.) We can use Theorem 1 to show that our example theory is e-bounded. As discussed earlier, the initial epistemic situation description is e-bounded by the grid size g. It is easy to see that the objective part maintains boundedness as all fluents are bounded by 1 except for $Visited$ which is bounded by g. □

Expressing Dynamic Properties

To express properties about epistemic action theories, we use a FO variant of the μ-calculus [Emerson, 1996; Stirling, 2001], one of the most powerful temporal logics, subsuming both linear time logics, such as LTL and PSL, and branching time logics such as CTL and CTL* [Baier *et al.*, 2008]. We introduce the logic $\mu\mathcal{L}_{\mathcal{K}}$, whose syntax is as follows:

$$\begin{array}{lcl} \varphi & ::= & p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \exists x.\varphi \mid \textbf{Knows}\ \varphi \\ \Phi & ::= & \varphi_c \mid \neg\Phi \mid \Phi_1 \wedge \Phi_2 \mid \langle - \rangle \Phi \mid \textbf{Knows}\ \Phi \mid Z \mid \mu Z.\Phi \end{array}$$

where p is an atomic *situation-suppressed* situation calculus FO formula, φ an arbitrary epistemic formula (we allow for quantifying in), φ_c an arbitrary closed epistemic formula, Z a SO (0-ary) predicate variable, and where in $\langle - \rangle \Phi$, Φ must be closed with respect to individual variables. Thus we do not allow quantification across dynamic operators.

We use the usual abbreviations for booleans, $[-]\Phi$ for $\neg\langle - \rangle\neg\Phi$, and $\nu Z.\Phi = \neg\mu Z.\neg\Phi[Z/\neg Z]$. As usual in the μ-calculus, formulae of the form $\mu Z.\Phi$ (and $\nu Z.\Phi$) must satisfy *syntactic monotonicity* of Φ wrt Z, which states that every occurrence of the variable Z in Φ must be within the scope of an even number of negation symbols.

The *fixpoint formulas* $\mu Z.\Phi$ and $\nu Z.\Phi$ denote respectively the *least* and the *greatest fixpoint* of the formula Φ seen as a predicate transformer $\lambda Z.\Phi$ (their existence is guaranteed by the syntactic monotonicity of Φ). Using these operators we can define sophisticated inductive and coinductive temporal/dynamic properties. For instance, we can use the least fixpoint formula $\mu Z.\varphi \vee \langle - \rangle Z$ to say that it is possible to achieve Φ, written in CTL notation as $E\Diamond\Phi$. Similarly, we can use the greatest fixpoint formula $\nu Z.\varphi \wedge [-]Z$ to say that Φ always holds, written in CTL notation as $A\Box\Phi$.

To define semantics, since $\mu\mathcal{L}_{\mathcal{K}}$ contains formulae with free predicate variables, given a model $\mathcal{M}$ of an action theory $\mathcal{D}$ with domain Δ for sort object and domain $\mathcal{S}$ for sort situation, we need to introduce an object variable valuation v, and a predicate variable valuation $\mathcal{V}$, i.e., a mapping from predicate variables Z to subsets of $\mathcal{S}$. Then we assign semantics to formulae by associating to $\mathcal{M}$, v and $\mathcal{V}$ two *extension functions* $(\cdot)_v^{\mathcal{M}}$ and $(\cdot)_{\mathcal{V}}^{\mathcal{M}}$, which together map $\mu\mathcal{L}_{\mathcal{K}}$ formulae to subsets of $\mathcal{S}$ as inductively defined as follows:

$$\begin{array}{lcl} (p)_v^{\mathcal{M}} & = & \{s \in \mathcal{S} \mid \mathcal{M}, v \models p[s]\} \\ (\neg\varphi)_v^{\mathcal{M}} & = & \mathcal{S} - (\varphi)_v^{\mathcal{M}} \\ (\varphi_1 \wedge \varphi_2)_v^{\mathcal{M}} & = & (\varphi_1)_v^{\mathcal{M}} \cap (\varphi_2)_v^{\mathcal{M}} \\ (\exists x.\varphi)_v^{\mathcal{M}} & = & \exists d \in \Delta.(\varphi)_{v[x/d]}^{\mathcal{M}} \\ (\textbf{Knows}\ \varphi)_v^{\mathcal{M}} & = & \{s \in \mathcal{S} \mid \forall s'.(s', s) \in K^{\mathcal{M}} \supset s' \in (\varphi)_v^{\mathcal{M}}\} \end{array}$$

$$\begin{array}{lcl} (\varphi_c)_{\mathcal{V}}^{\mathcal{M}} & = & (\varphi_c)_v^{\mathcal{M}} \qquad (\varphi_c \text{ closed}) \\ (\neg\Phi)_{\mathcal{V}}^{\mathcal{M}} & = & \mathcal{S} - (\Phi)_{\mathcal{V}}^{\mathcal{M}} \\ (\Phi_1 \wedge \Phi_2)_{\mathcal{V}}^{\mathcal{M}} & = & (\Phi_1)_{\mathcal{V}}^{\mathcal{M}} \cap (\Phi_2)_{\mathcal{V}}^{\mathcal{M}} \\ (\textbf{Knows}\ \Phi)_{\mathcal{V}}^{\mathcal{M}} & = & \{s \in \mathcal{S} \mid \forall s'.(s', s) \in K^{\mathcal{M}} \supset s' \in (\Phi)_{\mathcal{V}}^{\mathcal{M}}\} \\ (\langle - \rangle\Phi)_{\mathcal{V}}^{\mathcal{M}} & = & \{s \in \mathcal{S} \mid \exists a.(a, s) \in Poss^{\mathcal{M}} \wedge \\ & & \qquad\qquad do^{\mathcal{M}}(a, s) \in (\Phi)_{\mathcal{V}}^{\mathcal{M}}\} \\ (Z)_{\mathcal{V}}^{\mathcal{M}} & = & \mathcal{V}(Z) \\ (\mu Z.\Phi)_{\mathcal{V}}^{\mathcal{M}} & = & \bigcap\{\mathcal{E} \subseteq \mathcal{S} \mid (\Phi)_{\mathcal{V}[Z/\mathcal{E}]}^{\mathcal{M}} \subseteq \mathcal{E}\} \end{array}$$

Notice that the extension function $(\cdot)_v^{\mathcal{M}}$ is a standard epistemic FO interpretation function, while $(\cdot)_{\mathcal{V}}^{\mathcal{M}}$ deals with the temporal part. $(\cdot)_{\mathcal{V}}^{\mathcal{M}}$ assumes that epistemic FO formulas that constitute the atoms of the temporal formulas are closed, so it does not need to refer to the object variable valuation v. The only formulas of interest in verification are those that are closed (wrt predicate variables), and for them $(\cdot)_{\mathcal{V}}^{\mathcal{M}}$ does not depend on the predicate valuation $\mathcal{V}$. Thus, when Φ is closed, we simply write $(\Phi)^{\mathcal{M}}$. We say that a theory $\mathcal{D}$ verifies a $\mu\mathcal{L}_{\mathcal{K}}$ closed formula Φ, written $\mathcal{D} \models \Phi$, if $S_0 \in (\Phi)^{\mathcal{M}}$, for every model $\mathcal{M}$ of $\mathcal{D}$.

Example 3 (Wumpus Hunter Cont.) If the wumpus (in the current dungeon) is alive then it is possible for the hunter to eventually know that he has killed him: $A\Box(AliveW \supset E\Diamond\textbf{Knows}\ (\neg AliveW))$. It is also the case that the hunter knows that if the wumpus is alive then it is possible for him to eventually be killed by the wumpus: $A\Box\textbf{Knows}\ (AliveW \supset E\Diamond\neg AliveH)$. We can also show that the agent is aware of some of his memory limitations, e.g., even if the agent knows where the wumpus is, he may forget this if he moves to a different dungeon: $A\Box(\exists l\ \textbf{Knows}\ (locW(l)) \wedge locH = \langle 0, 0\rangle \supset \langle - \rangle\langle - \rangle\neg\exists l\ \textbf{Knows}\ (locW(l)))$. □

Our $\mu\mathcal{L}_{\mathcal{K}}$ language does not allow for quantification across situations. However the richness of the situation calculus mitigates this limitation. For instance we can introduce a finite number of "registers", i.e., fluents that store only one tuple, and then use them to store and refer to tuples across situations; see [De Giacomo *et al.*, 2012] for details. However, it remains of interest to generalize our framework to allow some forms of quantification across situations. Some encouraging results along these lines appear in [Hariri *et al.*, 2012a; Belardinelli *et al.*, 2012; Hariri *et al.*, 2012b].

Verification

We now show that verifying $\mu\mathcal{L}_{\mathcal{K}}$ temporal properties against bounded epistemic action theories is decidable. We first focus on epistemic action theories with a *complete specification of* S_0, i.e., s.t. $\mathcal{D}_0$ contains an axiom $F(\vec{x}, S_0) \equiv \phi_0^F(\vec{x})$ for each fluent $F \in \mathcal{F}$ and an axiom $K(s, S_0) \equiv \phi_0^K(s)$ specifying the initial uncertainty, as well as axioms specifying that K is an equivalence relation in S_0. At the end, we generalize our results to the incomplete specification case. Our first result is:

Theorem 2 *If $\mathcal{D}$ is an e-bounded epistemic action theory with complete specification of S_0, and Φ a closed $\mu\mathcal{L}_{\mathcal{K}}$ formula, then verifying if $\mathcal{D} \models \Phi$ is decidable.*

To explain this result, notice first that, as a consequence of complete knowledge, $\mathcal{D}$ has a unique model $\mathcal{M}$ with object sort $\Delta = \mathcal{N}$. Thus, we only need to check whether $S_0 \in (\Phi)^{\mathcal{M}}$. Let $\mathcal{L}_{\mathcal{K}}$ denote the set of uniform (in the situation argument) epistemic FO formulas of the situation calculus, and observe that the evaluation of any $\phi(\vec{x}, s) \in \mathcal{L}_{\mathcal{K}}$ does not depend on the situation s, but only on the corresponding *epistemic state* (e-state), i.e., the pair $\langle I_s, B_s \rangle$ s.t. $I_s = \langle \Delta, \cdot^{I_s} \rangle$ is the interpretation of $\mathcal{F}$ associated with s, and $B_s = \{I_{s'} | K(s', s)\}$ (observe that $I_s \in B_s$). It can be shown that for any two situations $s_1, s_2 \in \mathcal{S}$, $I_{s_1} = I_{s_2}$ and $B_{s_1} = B_{s_2}$ iff it is the case that for every $\phi(\vec{x}, s) \in \mathcal{L}_{\mathcal{K}}$ and every variable valuation v over $\mathcal{N}$, $\mathcal{M}, v \models \phi(\vec{x}, s_1)$ iff $\mathcal{M}, v \models \phi(\vec{x}, s_2)$. This applies in particular to successor state axioms, yielding that, for any action a, if $\langle I_{s_1}, B_{s_1} \rangle = \langle I_{s_2}, B_{s_2} \rangle$, then $\langle I_{s'_1}, B_{s'_1} \rangle = \langle I_{s'_2}, B_{s'_2} \rangle$, with $s'_1 = do(a, s_1)$ and $s'_2 = do(a, s_2)$.

Observe that the e-states $\langle I, B \rangle$ of $\mathcal{M}$ induce a partition of $\mathcal{S}$ s.t. two situations s, s' are in the same cell iff $I_s = I_{s'}$ and $B_s = B_{s'}$. By the above results, for any executable action a, we can generate the e-state $\langle I', B' \rangle$ of the successor situation of each situation in a cell H, by operating on $\langle I, B \rangle$. To do so, we evaluate on I the situation-suppressed successor state axioms instantiated on action $A(\vec{d})$ with $\vec{d}$ taken from Δ, obtaining I' (evaluating preconditions in the same way), and do the same for all action types $A'(\vec{d'})$ observationally-indistinguishable from $A(\vec{d})$ (these are FO-defined) on each $I_b \in B$, joining all the resulting states into B'. By considering all cells and actions, this defines a transition system (TS) $T = \langle \Delta, Q, q_0, \rightarrow \rangle$, s.t.: the object domain is $\Delta = \mathcal{N}$; Q is the set of states, corresponding to e-states; $q_0 = \langle I_{S_0}, B_{S_0} \rangle$ is the initial state; each transition $\langle I, B \rangle \rightarrow \langle I', B' \rangle$ is obtained by executing an action on $\langle I, B \rangle$, as above. T retains all the information necessary to evaluate Φ on $\mathcal{M}$. To show this, we transfer the semantics of $\mu\mathcal{L}_{\mathcal{K}}$ to TSs, using an inductive definition matching that of $(\cdot)^{\mathcal{M}}_{\mathcal{V}}$ with:

$$\begin{array}{lcl}
(p)^T_v & = & \{\langle I, B \rangle \in Q \mid I, v \models p\} \\
(\neg\varphi)^T_v & = & Q - (\varphi)^T_v \\
(\exists x.\varphi)^T_v & = & \exists d \in \Delta.(\varphi)^T_{v[x/d]} \\
(\mathbf{Knows}\ \varphi)^T_v & = & \{\langle I, B \rangle \in Q \mid \forall I_b \in B.\langle I_b, B \rangle \in (\varphi)^T_v\} \\
(\neg\Phi)^T_{\mathcal{V}} & = & Q - (\Phi)^T_{\mathcal{V}} \\
(\mathbf{Knows}\ \Phi)^T_{\mathcal{V}} & = & \{\langle I, B \rangle \in Q \mid \forall I_b \in B.\langle I_b, B \rangle \in (\Phi)^T_{\mathcal{V}}\} \\
(\langle - \rangle\Phi)^T_{\mathcal{V}} & = & \{\langle I, B \rangle \in Q \mid \exists \langle I', B' \rangle \in Q. \\
 & & \langle I, B \rangle \rightarrow \langle I', B' \rangle \wedge \langle I', B' \rangle \in (\Phi)^T_{\mathcal{V}}\}
\end{array}$$

Once $(\Phi)^T_{\mathcal{V}}$ is evaluated, it can be interpreted as a set of situations. The following result, consequence of the fact that $\mu\mathcal{L}_{\mathcal{K}}$ formulas are built from $\mathcal{L}_{\mathcal{K}}$ formulas, explains this, and states the equivalence of the $\mathcal{M}$- and T-based semantics.

Lemma 1 *If T is as above, then for any $\mu\mathcal{L}_{\mathcal{K}}$ formula Φ and valuation $\mathcal{V}$, $(\Phi)^{\mathcal{M}}_{\mathcal{V}} = \{s \in \mathcal{S} \mid \langle I_s, B_s \rangle \in (\Phi)^T_{\mathcal{V}}\}$.*

Thus, to check whether $S_0 \in (\Phi)^{\mathcal{M}}_{\mathcal{V}}$, instead of using $\mathcal{M}$, we can use T, and check whether $\langle I_0, B_0 \rangle \in (\Phi)^T_{\mathcal{V}}$. Notice however that, being infinite, T can neither be inspected nor even constructed. We next solve both problems.

Let $adom(I)$ denote the *active domain* of I, i.e., the set of all objects occurring in some fluent extension F^I, and let $adom(B) = \bigcup_{I_b \in B} adom(I_b)$. We denote by $C \subset \mathcal{N}$ the finite set of all constants, i.e, standard names, appearing in $\mathcal{D}$. Two e-interpretations $\langle I_1, B_1 \rangle \in \mathcal{I}^{\Delta_1}_{\mathcal{F}}$ and $\langle I_2, B_2 \rangle \in \mathcal{I}^{\Delta_2}_{\mathcal{F}}$ are *KA-isomorphic*, written $\langle I_1, B_1 \rangle \sim \langle I_2, B_2 \rangle$, if $C \subseteq \Delta_1$, $C \subseteq \Delta_2$, and there exists a bijection $h : adom(B_1) \cup C \rightarrow adom(B_2) \cup C$ that is the identity on C, and s.t.: $h(I_1) = I_2$ and $B_2 = \{h(I_{b1}) \mid I_{b1} \in B_1\}$, for $h(I)$ the interpretation obtained from I by renaming all of its objects o as $h(o)$. Intuitively, KA-isomorphic e-interpretations have matching relational structures. Next, we define a constructive procedure that, under e-boundedness, produces a finite-state TS $\hat{T}$ *KA-bisimilar* to T. The notion of KA-bisimilarity matches standard bisimilarity, except that the local condition in its definition requires two states to be KA-isomorphic. The following transfers a fundamental theorem for the μ-calculus to $\mu\mathcal{L}_{\mathcal{K}}$:

Lemma 2 *Given two KA-bisimilar TSs T and T', and a $\mu\mathcal{L}_{\mathcal{K}}$ formula Φ, $q_0 \in (\Phi)^T$ iff $q'_0 \in (\Phi')^{T'}$ where Φ' is a domain independent formula equivalent to Φ.*

The transformation of Φ into Φ' is required as Φ could be domain-dependent, and T, T' may have distinct interpretation domains. Through syntactic manipulations, Φ can be turned into a logically equivalent domain-independent formula Φ', see [Libkin, 2007]. To construct $\hat{T}$ we proceed exactly as done for T, except that we take values from a finite doman $\hat{\Delta} \subset \mathcal{N}$ (instead of Δ) s.t. $C \subseteq \hat{\Delta}$ and $|\hat{\Delta}| = |C| + b_e + N_{vars}$, where b_e is the bound (derived from the e-bound e) on the maximum number of distinct objects occurring in some e-state, and N_{vars} is the maximum among the number of distinct variables (assuming all quantified variables are renamed apart) occurring in some successor-state, observation-indistinguishability axioms, or the initial epistemic situation description $\mathcal{D}_0$. Such numbers are motivated by the fact that, to enforce KA-bisimilarity, a minimal requirement is that every state of T have a matching KA-isomorphic state in $\hat{T}$, thus b_e. Further, since $\hat{T}$ is obtained by executing actions on its states, a sufficient number of additional elements, N_{vars} is needed, to abstract all the possible combinations of the objects used in some action, and mentioned in the axioms.

Lemma 3 *T and $\hat{T}$ are KA-bisimilar.*

Thus, by Lemma 2, by checking whether $\hat{q}_0 \in (\Phi')^{\hat{T}}$, we can check whether $q_0 \in (\Phi)^T$. Since this can effectively be done by applying standard algorithms for model checking of the μ-calculus, suitably extended to evaluate the $\mathcal{L}_{\mathcal{K}}$ sentences occurring in Φ', we obtain an actual procedure for the verification of e-bounded epistemic theories against $\mu\mathcal{L}_{\mathcal{K}}$. This completes the proof of Theorem 2.

Finally, we consider the general case where we may have an incomplete specification of S_0.

Theorem 3 *If $\mathcal{D}$ is an e-bounded epistemic action theory (with possibly an incomplete specification of S_0) and Φ a closed $\mu\mathcal{L}_{\mathcal{K}}$ formula, then verifying if $\mathcal{D} \models \Phi$ is decidable.*

This follows from the fact that, for fixed $\hat{\Delta}$, any two TSs $\hat{T}$ and $\hat{T}'$ as above, s.t. $q_0 \sim q'_0$, are KA-bisimilar, and that

KA-isomorphic e-states satisfy exactly the same $\mathcal{L}_{\mathcal{K}}$ formulas. Thus, by Lemma 2 and the above construction, to check all the models whose initial state satisfies $\mathcal{D}_0$, it is sufficient to construct and check one $\hat{T}$ per class of isomorphic e-states satisfying $\mathcal{D}_0$. Since by e-boundedness such classes are finite, this can be done by checking a finite number of TSs.

Conclusion

We have defined the class of *e-bounded* theories in the epistemic situation calculus and shown that for them verification of a very expressive class of first-order μ-calculus temporal epistemic properties is *decidable*. The proof is constructive and gives us an effective procedure for verification. A related result in the context of interpreted systems appeared in [Belardinelli *et al.*, 2012]. The main difference, apart from the setting and the verification logic (they use a first-order variant of CTL), is that our epistemic framework supports incomplete theories, while theirs is tailored towards single models.

Note that if we do not have a fixed bound for all situations, then one can represent Turing machine configurations (including the tape, which is at any moment finite) in situations, and use basic action theories to encode transitions between configurations. Thus the mu-calculus formula $\mu Z.Halt \vee \langle - \rangle Z$, where $Halt$ is just a propositional fluent representing the fact that the machine reached an acceptance state, is undecidable. The number of objects in each model remains infinite. But in each (epistemic) situation when new objects are inserted in the active domain, old objects must be removed so as to maintain the bound. In each infinite branch of the situation tree the number of objects mentioned may still be infinite. A sophisticated abstraction is needed to reduce this infinite set into a finite one. Of course a situation can mention fewer objects than the bound.

In e-bounded theories, the number of fluent atoms that the agent thinks may be true is bounded. But note that there may be an infinite number of x such that $\neg$**Knows** $(F(x)) \vee$ **Knows** $(\neg F(x))$. Properties F for which there is an infinite number of x for which it is unknown whether or not $F(x)$ holds cannot be represented and must be left out of the language. Alternative approaches that bound what is known by the agent as opposed to what is considered possible should also be investigated. But this would appear to require strong assumptions about what forms of knowledge are being considered. We could bound the number of fluent atoms that are known to be true or known to be false. Or we could bound the number of prime implicates that are known to be true or false. But not all of first order epistemic logic can be expressed in such forms, so the resulting account would be restricted in the kinds of knowledge specifications it handles. We leave exploring such alternative notions for future work. Another interesting avenue for future work is to introduce Golog and ConGolog programs to specify processes over e-bounded theories, and extend our verification techniques to this setting.

Acknowledgements

Authors acknowledge support of EU Projects FP7-ICT 257593 (ACSI) and FP7-ICT 318338 (OPTIQUE).

References

[Abiteboul *et al.*, 1995] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison Wesley, 1995.

[Bacchus *et al.*, 1999] Fahiem Bacchus, Joseph Y. Halpern, and Hector J. Levesque. Reasoning about noisy sensors and effectors in the situation calculus. *Artif. Intell.*, 111(1-2):171–208, 1999.

[Baier *et al.*, 2008] Christel Baier, Joost-Pieter Katoen, and Kim Guldstrand Larsen. *Principles of Model Checking*. MIT Press, 2008.

[Belardinelli *et al.*, 2012] Francesco Belardinelli, Alessio Lomuscio, and Fabio Patrizi. An abstraction technique for the verification of artifact-centric systems. In *KR*, 2012.

[De Giacomo *et al.*, 2012] Giuseppe De Giacomo, Yves Lespérance, and Fabio Patrizi. Bounded Situation Calculus Action Theories and Decidable Verification. In *KR*, 2012.

[Emerson, 1996] E. Allen Emerson. Model checking and the mu-calculus. In *Descriptive Complexity and Finite Models*, pages 185–214, 1996.

[Hariri *et al.*, 2012a] Babak Bagheri Hariri, Diego Calvanese, Giuseppe De Giacomo, Alin Deutsch, and Marco Montali. Verification of relational data-centric dynamic systems with external services. *CoRR*, abs/1203.0024, 2012.

[Hariri *et al.*, 2012b] Babak Bagheri Hariri, Diego Calvanese, Giuseppe De Giacomo, Riccardo De Masellis, Paolo Felli, and Marco Montali. Verification of description logic knowledge and action bases. In *ECAI*, pages 103–108, 2012.

[Levesque and Lakemeyer, 2001] Hector J. Levesque and Gerhard Lakemeyer. *The Logic of Knowledge Bases*. MIT Press, 2001.

[Levesque *et al.*, 1998] Hector Levesque, Fiora Pirri, and Ray Reiter. Foundations for a calculus of situations. *Electronic Transactions of AI (ETAI)*, 2(3–4):159–178, 1998.

[Levesque, 1996] Hector J. Levesque. What is planning in the presence of sensing? In *AAAI*, pages 1139–1146, 1996.

[Libkin, 2007] Leonid Libkin. Embedded finite models and constraint databases. In *Finite Model Theory and Its Applications*. Springer, 2007.

[McCarthy and Hayes, 1969] J. McCarthy and P. J. Hayes. Some Philosophical Problems From the StandPoint of Artificial Intelligence. *Machine Intelligence*, 4:463–502, 1969.

[Moore, 1985] Robert C. Moore. A formal theory of knowledge and action. In J. R. Hobbs and Robert C. Moore, editors, *Formal Theories of the Common Sense World*, pages 319–358. Ablex Publishing, Norwood, NJ, 1985.

[Reiter, 2001] Ray Reiter. *Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems*. MIT Press, 2001.

[Russell and Norvig, 2010] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach, 3rd ed.* Prentice Hall, 2010.

[Scherl and Levesque, 1993] Richard B. Scherl and Hector J. Levesque. The frame problem and knowledge-producing actions. In *AAAI*, pages 689–695, 1993.

[Scherl and Levesque, 2003] Richard B. Scherl and Hector J. Levesque. Knowledge, action, and the frame problem. *Artif. Intell.*, 144(1-2):1–39, 2003.

[Stirling, 2001] Colin Stirling. *Modal and Temporal Properties of Processes*. Springer, 2001.