A Three-value Abstraction Technique for the Verification of Epistemic Properties in Multi-agent Systems

Francesco Belardinelli and Alessio Lomuscio

Department of Computing, Imperial College London, UK, EU

Abstract. We put forward an abstraction technique, based on a three-value semantics, for the verification of epistemic properties of agents participating in a multi-agent system. First we introduce a three-value interpretation of epistemic logic, based on a notion of order defined on the information content of the local states of each agent. Then, we use the three-value semantics to introduce an abstraction technique to verify epistemic properties of agents in infinite-state multi-agent systems.

Keywords: Logics in multi-agent systems; Epistemic Logic; Formal verification by model checking.

1 Introduction

Modal logics for knowledge representation and reasoning, including epistemic logics, have been proved to be a valuable formal tool for the modelling and analysis of multi-agent systems [15, 21, 29]. These logical languages typically include an operator K_i to represent the knowledge of an agent i, as well as possibly modalities for collective, common and distributed, knowledge. In combination with techniques for automated verification by model checking, epistemic logics have been used to model and verify complex multi-agents scenarios [26], among which communication and security protocols [6], auction-based mechanisms [5], business process workflows [4, 20].

The application of methods from knowledge representation and reasoning to the verification of multi-agent systems (MAS) depends crucially on the development of efficient model checking methodologies and algorithms. In particular, abstraction techniques are key to tackle the state-space explosion problem [9, 23]. Moreover, whenever agents manipulate infinite data types (e.g., natural numbers, integers, reals, lists, arrays, etc.), finite abstractions are often the only chance to obtain a decidable model checking problem [1, 4, 13].

Inspired by the considerations above, in this paper we put forward an abstraction technique, based on a three-value semantics, for the verification of epistemic properties of agents participating in a MAS. Specifically, the contribution of the paper is twofold. Firstly, we introduce a three-value interpretation of epistemic logic, which is based on a partial order $\leq$ defined on the information content of the local states of each agent. According to this intuition, agent i considers epistemically possible not just states that are indistinguishable to her, i.e., in which i's local state is identical, but also states comparable by order $\leq$. We illustrate the formal machinery with examples from agent-based systems, particularly infinite-state systems that are not directly amenable to standard model checking techniques. Secondly, we use the three-value semantics to introduce an abstraction technique to model check epistemic properties of agents in infinite-state MAS. As a result, our contribution is meant to advance the state-of-the-art both in the theory of epistemic logic and the verification of MAS.

Related works. The area of epistemic logics has reached such a level of maturity nowadays that it is extremely difficult to provide an exhaustive account. Here we only mention the contributions most closely related to the verification of multi-agent systems by abstraction. Techniques to model check epistemic properties of agents in MAS have witnessed a growing interest in recent years, with a number of tools made publicly available [18, 22, 27]. This work pursues the same research direction, but we target explicitly infinite-state MAS, for which the verification task is considerably more complex. Abstraction techniques for epistemic properties of MAS have appeared in [11, 10], but the underlying logic is two-valued, and therefore only its "universal" fragment is preserved by the abstraction procedure. Instead, here we adopt the abstraction method via under- and over-approximations, which has been applied mainly to the verification of simple transitions systems against temporal properties [2, 7, 19, 28]. Previous contributions on three-value abstractions for epistemic logics have appeared in [14, 20, 24, 25]. However, the settings and the three-value semantics are different w.r.t. the account here put forward. Specifically, in [14] there is no notion of under- and over-approximation, as the three-value semantics follows [16, 17]. This implies that only (universal and existential) fragments of the original language are preserved. Hence, the class of verifiable specifications is somewhat limited, while here we are able to verify the full language in principle. Further, in [24, 25] the proposed three-value semantics is not a conservative extension of the standard two-value semantics for epistemic logic, in particular no analogue to Proposition 1 below can be proved. As a consequence, verification results available for the three-value semantics do not immediately transfer to the two-value semantics. Finally, differently from [20], we ground under- and over-approximations on a relation $\leq$ of order between local states, which provides guidance as to the definition of the abstract system, while making the abstraction process more transparent in our opinion.

Scheme of the paper. In Section 2 we introduce the multi-agent epistemic logic $\mathbf{K}$ that includes operators for distributed and common knowledge, and we provide $\mathbf{K}$ with a three-value semantics based on an order $\leq$ on the local states of each agent. We illustrate the formal machinery with examples of (infinite-state) multi-agent systems. In Section 3 we develop an agent-based abstraction technique that we prove to preserve the three-value interpretation of formulas in $\mathbf{K}$. We conclude by discussing applications of these results to the verification of epistemic properties of infinite-state MAS.

2 Preliminaries

In this section we introduce the formalism of three-value epistemic logic. First, we present the language of multi-agent epistemic logic, including modalities for collective knowledge. Then, we provide this logic with a Kripke-style semantics, which allows to compare the local information possessed by agents, thus inducing a natural three-value semantics suitable for abstractions.

In the following $Ag = \{1, \ldots, m\}$ is a set of indexes for agents and AP is a set of atomic propositions. Also, we denote the $i+1$-th element of a tuple v as v_i.

The Language. To reason about multi-agent systems and to describe properties pertaining to the agents' knowledge, we make use of the multi-modal epistemic logic **K** defined by the following BNF:

$$\varphi ::= q \mid \neg\varphi \mid \varphi \to \varphi \mid C_\Gamma \varphi \mid D_\Gamma \varphi$$

where $q \in AP$ and $\Gamma \subseteq Ag$.

The informal meaning of formulas $C_\Gamma \varphi$ is that "φ is *common* knowledge in group Γ"; while $D_\Gamma \varphi$ is read as "φ is *distributed* knowledge in group Γ". As customary, we can introduce individual knowledge formulas $K_i \varphi$ as shorthands for either $C_{\{i\}} \varphi$ or $D_{\{i\}} \varphi$. Also, we omit group Γ whenever $\Gamma = Ag$. Notice that **K** is not to be confused with the homonymous normal modal logic.

The Models. To provide a formal interpretation to the epistemic formulas in **K**, we introduce a notion of agent and interpreted systems.

Definition 1 (Agent). *Given a set Ag of agent indexes, an* agent *is a tuple $i = \langle L, Act, Pr, \tau \rangle$ such that*

- *L is the (possibly infinite) set of* local states *with a partial order $\leq$ on L;*
- *Act is the set of* actions*;*
- *$Pr : L \to (2^{Act} \setminus \{\emptyset\})$ is the* protocol function*;*
- *$\tau : L \times ACT \to 2^L$ is the* local transition function*, where $ACT = Act_1 \times \cdots \times Act_{|Ag|}$ is the set of* joint actions*, such that $\tau(l, a)$ is defined iff $a_i \in Pr(l)$.*

The notion of agent in Def. 1 is typical of the literature on interpreted systems [15, 27]: each agent is assumed to be situated in some local state, and to perform the actions in Act according to protocol Pr. The evolution of her local state is determined by the transition function τ. Differently from the state-of-the-art, we also consider a partial order $\leq$ on local states, i.e., a reflexive, antisymmetric, and transitive relation on L. Intuitively, $l \leq l'$ means that in local state l' agent i has at least as much information as in l. The partial order $\leq$ is key to approximate the knowledge of agent i, whenever computing the exact information of i is too costly computationally, not dissimilarly to the use of over- and under-approximations in system verification [2, 28]. Further, the standard notion of agent appearing in the literature can be seen as a particular case of Def. 1, in which the partial order $\leq$ is the identity. If this is the case, we say that the agent is *standard*.

Given a set Ag of agents, a *global state* is a tuple $s = \langle l_1, \ldots, l_{|Ag|} \rangle$ of local states, one for each agent in the system. We denote the set $L_1 \times \ldots \times L_{|Ag|}$ of all global states as $\mathcal{G}$. We now introduce interpreted systems to describe formally the interactions of agents in a multi-agent environment.

Definition 2 (IS). *An* interpreted system *is a tuple* $M = \langle Ag, I, \tau, \Pi \rangle$ *where*

- *every* $i \in Ag$ *is an agent;*
- $I \subseteq \mathcal{G}$ *is the set of* (global) initial states*;*
- $\tau : \mathcal{G} \times ACT \to 2^{\mathcal{G}}$ *is the* global transition function *such that* $\tau(s, a) = \tau_1(s_1, a) \times \ldots \times \tau_{|Ag|}(s_{|Ag|}, a)$*;*
- $\Pi : \mathcal{G} \times AP \to \{\mathrm{tt}, \mathrm{ff}, \mathrm{uu}\}$ *is the* labelling function*.*

According to Def. 2, an interpreted system describes the evolution of a group Ag of agents from any initial state in I, according to the global transition function τ. By the constraint on each τ_i, $\tau(s, a)$ is defined iff $a_i \in Pr(s_i)$ for every $i \in Ag$. In the following we also make use of the local transition relation $\to$ such that $l \to l'$ iff $l' \in \tau(l, a)$ for some $a \in ACT$, as well as its reflexive and transitive closure $\to^*$. A global transition relation $\to$ and its reflexive and transitive closure $\to^*$ are defined similarly on global states in $\mathcal{G}$. Then, the set $\mathcal{S}$ of *reachable states* is introduced as the closure of I under $\to^*$, that is, $s \in \mathcal{S}$ iff $s_0 \to^* s$ for some initial $s_0 \in I$. Hereafter we assume that only reachable states count as epistemic alternatives for the agents in the interpreted system. That is, states that are not reachable in the system are not considered epistemically possible by the agents. This is in line with current accounts of IS [15, 27].

Atomic propositions in AP can be assigned value true (tt), false (ff), or undefined (uu). This last value can be used to describe situations in which the truth of an atom is not set, or it is unknown, or underspecified. We will see examples of these instances at the end of the section. We say that the truth value t is *defined* whenever $t \in \{\mathrm{tt}, \mathrm{ff}\}$. If all agents in Ag are standard and the truth value of all atoms is defined, then we say that the IS is *standard* as well.

In the two-value semantics for epistemic logic the interpretation of knowledge formulas is normally given by means of an individual indistinguishability relation $\sim_i$ on global states, which is defined by the identity of local states, that is, $s \sim_i s'$ iff $s_i = s'_i$ [15]. Here we define over-approximation R_i^{may} and under-approximation R_i^{must} of relation $\sim_i$ by leveraging on the fact that we consider the partial order $\leq$ on local states, rather than simply their identity. Specifically, for each agent $i \in Ag$, we define relation R_i^{may} on global states such that $R_i^{may}(s, s')$ iff for some reachable $s'' \in \mathcal{S}$, $s''_i \geq s_i$ and $s''_i \geq s'_i$. Further, $R_i^{must}(s, s')$ iff $s'_i \leq s_i$. Notice that in particular $R_i^{must}(s, s')$ implies $R_i^{may}(s, s')$. Intuitively, R_i^{may} can be thought of as over-approximating the knowledge of agent i. Indeed, states s and s' are related by R_i^{may} if the information of agent i in s and s' can be consistently combined in some reachable state s''_i (which is indeed an over-approximation of both s_i and s'_i); while $R_i^{must}(s, s')$ holds iff s'_i under-approximates the information contained in s_i. We remark that the use of over- and under-approximations R_i^{may} and R_i^{must} is customary in multi-valued logics and abstraction for transition systems [2, 28]. Here we apply approximations to epistemic logic by grounding them on an order defined on information states.

To interpret common and distributed knowledge, for $x \in \{may, must\}$, we consider the intersection $R_\Gamma^{Dx} = \bigcap_{i \in \Gamma} R_i^x$ and the transitive closure $R_\Gamma^{Cx} = (\bigcup_{i \in \Gamma} R_i^x)^+$ of the union of accessibility relations. Then, $R_\Gamma^{Dx}(s, s')$ holds iff $R_i^x(s, s')$ holds for all $i \in \Gamma$; while $R_\Gamma^{Cx}(s, s')$ is the case iff for some sequence $s_0, \ldots, s_n$ of states, (i) $s_0 = s$ and $s_n = s'$; and (ii) for every $k < n$, $R_i^x(s_k, s_{k+1})$ for some $i \in \Gamma$. Finally, notice that R_i^{may} and R_i^{must} are both reflexive and R_i^{may} is also symmetric. However, they are not transitive in general, and therefore they are not equivalence relations. As a result, relations R_i^{may} and R_i^{must} do not define an S5-modality. This is to be expected and not really an issue in the present context, as we are interested in truth of formulas in a model as opposed to validity in a class of models. In particular, if the interpreted system is standard, then $R_i^{may} = R_i^{must}$ is an equivalence relation and we are back to the standard indistinguishability relation $\sim_i$ of the two-value semantics for epistemic logic.

Finally, we introduce a three-value interpretation of epistemic formulas in the logic **K**.

Definition 3 (Satisfaction). *The three-value satisfaction relation $\models^3$ for an IS M, state $s \in \mathcal{S}$, and formula ϕ is inductively defined as follows:*

$$
\begin{array}{lll}
((M,s) \models^3 q) = t & \textit{iff} & \Pi(s,q) = t, \textit{ for } t \in \{\mathrm{tt}, \mathrm{ff}\} \\
((M,s) \models^3 \neg\phi) = \mathrm{tt} & \textit{iff} & ((M,s) \models^3 \phi) = \mathrm{ff} \\
((M,s) \models^3 \neg\phi) = \mathrm{ff} & \textit{iff} & ((M,s) \models^3 \phi) = \mathrm{tt} \\
((M,s) \models^3 \phi \to \phi') = \mathrm{tt} & \textit{iff} & ((M,s) \models^3 \phi) = \mathrm{ff} \textit{ or } ((M,s) \models^3 \phi') = \mathrm{tt} \\
((M,s) \models^3 \phi \to \phi') = \mathrm{ff} & \textit{iff} & ((M,s) \models^3 \phi) = \mathrm{tt} \textit{ and } ((M,s) \models^3 \phi') = \mathrm{ff} \\
((M,s) \models^3 C_\Gamma\varphi) = \mathrm{tt} & \textit{iff} & \textit{for all } s' \in \mathcal{S},\ R_\Gamma^{Cmay}(s,s') \textit{ implies } ((M,s') \models^3 \varphi) = \mathrm{tt} \\
((M,s) \models^3 C_\Gamma\varphi) = \mathrm{ff} & \textit{iff} & \textit{for some } s' \in \mathcal{S},\ R_\Gamma^{Cmust}(s,s') \textit{ and } ((M,s') \models^3 \varphi) = \mathrm{ff} \\
((M,s) \models^3 D_\Gamma\varphi) = \mathrm{tt} & \textit{iff} & \textit{for all } s' \in \mathcal{S},\ R_\Gamma^{Dmay}(s,s') \textit{ implies } ((M,s') \models^3 \varphi) = \mathrm{tt} \\
((M,s) \models^3 D_\Gamma\varphi) = \mathrm{ff} & \textit{iff} & \textit{for some } s' \in \mathcal{S},\ R_\Gamma^{Dmust}(s,s') \textit{ and } ((M,s') \models^3 \varphi) = \mathrm{ff}
\end{array}
$$

*In all other cases, the value of ϕ is undefined (*uu*).*

By Def. 3 we can derive the satisfaction clauses for individual knowledge formulas as follows:

$$
\begin{array}{lll}
((M,s) \models^3 K_i\varphi) = \mathrm{tt} & \text{iff} & \text{for all } s' \in \mathcal{S},\ R_i^{may}(s,s') \text{ implies } ((M,s') \models^3 \varphi) = \mathrm{tt} \\
((M,s) \models^3 K_i\varphi) = \mathrm{ff} & \text{iff} & \text{for some } s' \in \mathcal{S},\ R_i^{must}(s,s') \text{ and } ((M,s') \models^3 \varphi) = \mathrm{ff}
\end{array}
$$

Intuitively, agent i knows ϕ at state s iff in all states s' that are epistemically compatible with s (in the sense that the information of s and s' can be consistently combined in a third reachable state s''), ϕ holds at s'. This can be seen as a conservative notion of knowledge, as ϕ has to be true in all such states s', in which i might have strictly more information than in s. Symmetrically, for $K_i\phi$ to be false at s, ϕ has to be false in some state s' in which i has at most as much information as in s.

We remark that the logic **K** does not contain temporal operators, and therefore in **K** we cannot describe notions pertaining to the evolution of knowledge, nor the knowledge of temporal facts. Nonetheless, we provided a dynamic account of agents and interpreted systems, which is apparent in Def. 3 as the interpretation of epistemic formulas is restricted to the set $\mathcal{S}$ of reachable states.

Indeed, in line with the standard semantics of interpreted systems [15, 27], we assume that agents consider epistemically possible only the reachable states in $\mathcal{S}$, and therefore the dynamics of IS is accounted for also in the semantics of static epistemic properties. In Section 3 we will see that this has a major impact on the definition of abstractions.

The two-value satisfaction relation $\models^2$ for standard IS can be derived from $\models^3$ by considering clauses for tt only, as well as identity of local states and classic negation (clauses for propositional connectives are immediate and thus omitted):

$$
\begin{array}{lll}
(M, s) \models^2 q & \text{iff} & \Pi(s, q) = \text{tt} \\
(M, s) \models^2 C_\Gamma \varphi & \text{iff} & \text{for all } s' \in \mathcal{S},\ s \sim_\Gamma^C s' \text{ implies } (M, s') \models^2 \varphi \\
(M, s) \models^2 D_\Gamma \varphi & \text{iff} & \text{for all } s' \in \mathcal{S},\ s \sim_\Gamma^D s' \text{ implies } (M, s') \models^2 \varphi
\end{array}
$$

An IS M satisfies a formula φ, or $M \models^2 \varphi$, iff for all states $s \in \mathcal{S}$, $(M, s) \models^2 \varphi$. Similarly, $(M \models^3 \varphi) = \text{tt}$ (resp. ff) iff for all (resp. some) $s \in \mathcal{S}$, $((M, s) \models^3 \varphi) = \text{tt}$ (resp. ff). In all other cases, $(M \models^3 \varphi) = \text{uu}$.

We now state the model checking problem for this setting.

Definition 4 (Model Checking Problem). *Given an IS M and a formula ϕ in $\mathbf{K}$, determine whether $M \models \phi$.*

Since we defined agents on possibly infinite sets of local states, interpreted systems are really infinite-state systems and the model checking problem is undecidable in general. In Section 3 we develop abstraction techniques to tackle the model checking problem. For the time being, we prove the following auxiliary result, which shows that for standard IS the two-value and three-value semantics for $\mathbf{K}$ coincide.

Proposition 1. *In every standard IS M, for every state s and formula ϕ in $\mathbf{K}$,*

$$
\begin{array}{c}
((M, s) \models^3 \phi) = \text{tt} \ \ \textit{iff} \ \ (M, s) \models^2 \phi \\
((M, s) \models^3 \phi) = \text{ff} \ \ \textit{iff} \ \ (M, s) \not\models^2 \phi
\end{array}
$$

Proof. The proof is by induction on ϕ, the interesting cases concern the knowledge formulas. We prove the case for $\phi = K_i \varphi$. We remarked above that in standard IS the distinction between over- and under-approximations collapse, and $R_i^{may} = R_i^{must} = \sim_i$. Hence, $((M, s) \models^3 \phi) = \text{tt}$ iff for all $s' \in \mathcal{S}$, $R_i^{may}(s, s')$ implies $((M, s') \models^3 \varphi) = \text{tt}$. Since $R_i^{may}(s, s')$ iff $s \sim_i s'$ and by induction hypothesis, the above is equivalent to $s \sim_i s'$ implies $(M, s') \models^2 \varphi$, for all $s' \in \mathcal{S}$, that is, $(M, s) \models^2 \phi$. The case for $((M, s) \models^3 K_i \varphi) = \text{ff}$ is symmetric; while the inductive cases for $\phi = C_\Gamma \varphi$ and $\phi = D_\Gamma \varphi$ are proved similarly.

By Proposition 1 on standard IS the three-value semantics for $\mathbf{K}$ is a conservative extension of the typical two-value semantics. This result has a major impact on the abstraction procedure put forward in Section 3.

We conclude this section with two examples of interpreted systems. In particular, we consider two types of systems: (i) systems with a natural partial order defined on the local states of agents, and (ii) infinite-state IS for which we will define finite, three-value abstractions in Section 3.

Example 1. We first consider an example of an interpreted system with a partial order defined on each agent's local states. We introduce a variant of the muddy children puzzle [15], in which each child sees some of the other children, but she might not see all of them, and she does not know how many children are exactly taking part in the puzzle. Hence, we assume that the local state of child i is a tuple $\langle s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_{|Ag|} \rangle$ that registers whether any other child $j \neq i$ is either clean (0), muddy (1), or unknown $(-)$. We define an order $\leq$ on local states such that $l \leq l'$ iff $l'_j = l_j$ for every child $j \neq i$ with $l_j \neq -$.

Now consider a global state $s = (0, 1, -)$, in which child 1 sees that child 2 is muddy, while she has no information on 3. In particular, child 1 knows that, provided that child 2 is actually active in the puzzle (i.e., 2's local state is different from $-$), then she is not muddy, but 1 does not know this about child 3. Formally, we can check that $(s \models K_1(active_2 \rightarrow m_2)) = \mathrm{tt}$ as for all states s', $R_1^{may}(s, s')$ implies that $s'_2 \in \{1, -\}$, and therefore 2 is muddy whenever she is active. On the other hand, we have that $(s \models K_1(active_3 \rightarrow m_3)) \neq \mathrm{tt}$, as for state $s'' = (0, 1, 0)$, $R_1^{may}(s, s'')$ holds, that is, child 3 is active but clean. Also, $(s \models K_1(active_3 \rightarrow m_3)) \neq \mathrm{ff}$, as for every s', $R_1^{must}(s, s')$ implies $s'_3 = -$, i.e., $active_3 \rightarrow m_3$ is vacuously true. As a result, $(s \models K_1(active_3 \rightarrow m_3)) = \mathrm{uu}$. By reasoning similarly, we can check that $(s \models D(active_2 \rightarrow m_2)) = \mathrm{tt}$, while $(s \models C(active_3 \rightarrow m_3)) = \mathrm{uu}$.

Furthermore, we consider the impact of the system's evolution on the epistemic properties of agents. In the classic muddy children puzzle, the father announces that at least one child is muddy. As a consequence, no child considers state $(0, 0, 0)$ epistemically possible any longer. In particular, after the father's announcement, at state $v = (1, 0, 0)$ child 1 knows that she is muddy, as for all reachable states v', $R_1^{may}(v, v')$ implies $v_1 = 1$. Hence, it is the case that $(v \models K_1 m_1) = \mathrm{tt}$. On the other hand, for state $u = (1, 0, -)$, we have that $(u \models K_1 m_1) \neq \mathrm{tt}$, as $R_1^{may}(u, (0, 0, 1))$ and $((0, 0, 1) \models m_1) = \mathrm{ff}$. Further, $(u \models K_1 m_1) \neq \mathrm{ff}$ as, if $R_1^{must}(u, u')$, then $u'_1 = 1$ because at least one child has to be muddy, and therefore $(u' \models m_1) \neq \mathrm{ff}$. As a result, $(u \models K_1 m_1) = \mathrm{uu}$, that is, child 1 is not able to see any other muddy child, but she cannot infer that she is muddy, as she is unsure about 3. Most importantly, the epistemic properties of agents depends essentially on the states reachable in the system's execution.

Example 2. The second example we analyse hinges on a standard IS, but with an infinite number of states. In Section 3 we will show how a finite, three-value abstraction can be defined on such infinite-state IS, in order to make the model checking problem decidable.

In this scenario we consider agents 1 and 2, whose local states are represented by integer variables x and y respectively, taking values in $\mathbb{Z}$, together with the environment e. Agents 1 and 2 can increase or decrease the value of their integers at any time, but in selected cases the joint action takes effect only if they increase or decrease their values simultaneously. Formally, we define agents 1 and 2 so that (i) $L_1 = L_2 = \mathbb{Z}$; (ii) $Act_1 = Act_2 = \{inc, dec\}$; and (iii) $Pr_1(z) = Pr_2(z) = Act_1$ for all $z \in \mathbb{Z}$. Moreover, as regards the environment e, we have $L_e = \{(x, y) \mid x, y \in \mathbb{Z}\}$; and $Act_e = \{ok, no\}$ with $Pr_e((x, y)) = \{ok\}$ iff $x = -4 \Leftrightarrow y = -2$

or $x = 5 \Leftrightarrow y = 3$; $Pr_e((x, y)) = \{no\}$, otherwise. Then, the transition function τ_1 is given as follows. If $x \neq -4$, $x \neq 5$, and $a = (a_1, a_2, ok)$, then the updated value x' is obtained by applying the increase or decrease action a_1. Further, if $x = -4$ or $x = 5$, and $a_1 \neq a_2$ or $a_3 = no$, then the updated value x' is equal to x; else, if $a = (a_1, a_2, ok)$ for $a_1 = a_2$, then the updated value x' is obtained by applying the corresponding action a_1. The definition of τ_2 is symmetric for $y = -2$ and $y = 3$, and it is given as follows:

$$
\begin{array}{ll}
\tau_2(y, (a_1, a_2, ok)) = a_2(y) & \text{for } y \neq -2 \text{ and } y \neq 3; \\
\tau_2(y, (a_1, a_2, a_3)) = y & \text{for } y = -2 \text{ or } y = 3, \text{ and } a_3 = no \text{ or } a_1 \neq a_2; \\
\tau_2(y, (a_1, a_2, ok)) = a_2(y) & \text{for } y = -2 \text{ or } y = 3, \text{ and } a_1 = a_2.
\end{array}
$$

Intuitively, agents 1 and 2 freely increment and decrement their local variable, but synchronise in states $(-4, -2)$ and $(5, 3)$ to either increment or decrement simultaneously. Since each agent can only view her local variable, the environment e acts as to guarantee their synchronisation. In particular, the environment's transition function τ_e is given as follows:

$$
\begin{array}{ll}
\tau_e((x, y), (a_1, a_2, ok)) = (a_1(x), a_2(y)) & \text{for } (x, y) \neq (-4, -2) \text{ and } (x, y) \neq (5, 3); \\
\tau_e((x, y), (a_1, a_2, a_3)) = (x, y) & \text{for } (x, y) = (-4, -2) \text{ or } (x, y) = (5, 3), \\
 & \text{and } a_3 = no \text{ or } a_1 \neq a_2; \\
\tau_e((x, y), (a_1, a_2, ok)) = (a_1(x), a_2(y)) & \text{for } (x, y) = (-4, -2) \text{ or } (x, y) = (5, 3), \\
 & \text{and } a_1 = a_2.
\end{array}
$$

Finally, we introduce the interpreted system M on the set $Ag = \{1, 2, e\}$ of agents, starting from initial state $(0, 0)$.

By the definition of M, we can check informally that if $x \leq 3$, then agent 1 knows that $y \leq 3$, that is, the specification $(x \leq 3) \rightarrow K_1(y \leq 3)$ is true in M. Moreover, specification $(x \leq 3) \rightarrow D_{\{1,2\}}(y \leq 3)$ holds as well. However, to verify such formulas at some state $s = (x, y)$ such that $s \models x \leq 3$ we have in principle to check that $y \leq 3$ on an infinite number of states $s' = (x, y')$, for $y' \in \mathbb{Z}$, which are indistinguishable for agent 1. As a consequence, model checking epistemic specification on infinite-state IS is undecidable in principle. In the case in hand we can reason about the particular protocol and specification considered, and reach a conclusive answer. However, our aim is to develop an abstraction-based general-purpose verification procedure that does not rely on system-specific features and can be applied as generally as possible.

3 Abstraction

In this section we introduce an abstraction-based technique for the verification of epistemic properties on standard, possibly infinite interpreted systems. Specifically, for every agent $i \in Ag$ in a standard IS M, we define an abstract agent i^A and the corresponding abstract IS M^A. Then, we prove that any formula ϕ in $\mathbf{K}$ is preserved by the abstraction, that is, if ϕ receives a defined truth value in M^A, then this value is preserved in M. As a result, given an infinite-state standard IS M, we can define a verification procedure by model checking a suitable finite-state abstraction M^A. However, the abstraction M^A of a standard IS M is not necessarily standard, and some specification ϕ can receive an undefined truth value in M^A. Therefore, the outlined procedure defines a partial verification technique, which is to be expected given that model checking infinite-state systems is undecidable in the most general instance.

To define abstraction M^A we introduce some preliminary notions. Given a standard agent $i \in Ag$, we say that a set $\mathcal{U} = \{U_1, \ldots, U_k\} \subseteq 2^L \setminus \{\emptyset\}$ of non-empty subsets $U \subseteq L$ of local states is a *cover* of L iff for every $l \in L$, $l \in U$ for some $U \in \mathcal{U}$. Then, we define a partial order $\leq$ on sets U, U' in the cover $\mathcal{U}$ so that $U \leq U'$ iff $U' \subseteq U$. Intuitively, a set U' of local states contains more information than U iff U' is a subset of U. This is in line with the informal meaning of local states as epistemic alternatives: if agent i considers possible less epistemic alternatives, then she has more information about what the current state actually looks like. In the limit case, for U a singleton, i knows exactly the current state.

Given a cover $\mathcal{U}$ for a standard agent $i \in Ag$, we define the abstraction i^A.

Definition 5 (Abstract Agent). *Given a standard agent* $i = \langle L, Act, Pr, \tau \rangle$ *and a cover* $\mathcal{U}$*, we introduce the abstraction* $i^A = \langle L^A, Act^A, Pr^A, \tau^A \rangle$ *such that*

- $L^A = \mathcal{U}$ *with partial order* $\leq$ *such that* $U \leq U'$ *iff* $U' \subseteq U$;
- $Act = Act^A$;
- *for every* $U \in L^A$, $Pr^A(U) = \bigcup_{l \in U} Pr(l)$;
- $U' \in \tau^A(U, a)$ *iff for some* $l \in U$, $l' \in U'$, *we have* $l' \in \tau(l, a)$.

Notice that the size of an abstract agent i^A, given as the cardinality $|L^A|$ of the set L^A of her abstract local states, is finite although the set L of concrete local states might be infinite. Indeed, while in such a case cover L^A must contain at least one subset $U \subseteq L$ with infinitely many local states, the size $|L^A|$ of L^A given as its cardinality is finite. Further, an action a is enabled in abstract state U iff it is enabled in some local state in U; while a transition $U \to U'$ holds iff $l \to l'$ for some local states $l \in U$ and $l' \in U'$. Observe that the definition of the abstract transition function τ^A is in line with similar notions for the abstraction of simple transition systems [8]. As a consequence, it is also prone to some of the related issues. In particular, the abstract transition might generate reachable states for which there is no corresponding concrete transition and reachable states, that is, the abstract transition might be spurious. Hereafter, we impose constraints on our abstract agents and interpreted systems to avoid spurious transitions.

Next we define a kind of simulation relation between the global states built on the concrete and abstract agents respectively, and say that $s' \in \mathcal{G}^A$ *simulates* $s \in \mathcal{G}$, or $s \preceq s'$, iff for every $i \in Ag$, $s_i \in s'_i$. Notice that, since each $\mathcal{U}_i$ is a cover, for every $s \in \mathcal{G}$, $s \preceq s'$ for some $s' \in \mathcal{G}^A$.

We now introduce the abstraction M^A of a standard IS M, defined on abstract agents $i^A \in Ag^A$, as follows.

Definition 6 (Abstract IS). *Given a standard IS* $M = \langle Ag, I, \tau, \Pi \rangle$*, the abstract IS* $M^A = \langle Ag^A, I^A, \tau^A, \Pi^A \rangle$ *is such that*

– Ag^A *is the set of abstract agents* i^A*, for each agent* $i \in Ag$*;*
– $I^A = \{s' \in \mathcal{G}^A \mid s \preceq s'$ *for some* $s \in I\}$*;*
– τ^A *is defined as in Def. 2;*
– *for every* $s' \in \mathcal{G}^A$*, for* $t \in \{\mathrm{tt}, \mathrm{ff}\}$*,* $\Pi^A(s', p) = t$ *iff for all* $s \in \mathcal{G}$*,* $s \preceq s'$ *implies* $\Pi(s, p) = t$*; otherwise,* $\Pi^A(s', p) = \mathrm{uu}$*.*

Since each $\mathcal{U}_i$ is a cover, the set I^A of abstract initial states is non-empty whenever I is. Further, the global abstract transition function τ^A is indeed the composition of the various local τ_i^A, as per Def. 2; while an atom is either true or false at abstract state s' iff it is such in all concrete states simulated by s', otherwise it is undefined.

Above we mentioned that the abstract transition function can introduce reachable states in the abstract IS M^A, for which there is no corresponding concrete state reachable in M. In particular, in M^A an agent might consider reachable epistemic alternatives, that are not really such in M. This remark motivates the introduction of the following notion.

Definition 7 (Admissibility). *A set* Ag^A *of abstract agents is* admissible *iff for every* $s' \in \mathcal{G}^A$ *and* $s, t \in \mathcal{G}$*, if* $s \preceq s'$ *and* $t \preceq s'$ *then* $s \to^+ t$*.*

Intuitively, this condition on IS says that any state simulated by $s' \in \mathcal{G}^A$ is eventually reachable from any other state simulated by s'. Then, an abstraction M^A of an IS is *admissible* iff all its abstract agents in Ag^A are.

By this notion of admissibility we are able to prove the following key result, which intuitively states that the epistemic relations in IS M and abstraction M^A commute with the simulation relation $\preceq$.

Lemma 1. *Let* M *be a standard IS with admissible abstraction* M^A*. If* $s \preceq s'$ *and* $R_i^{Amust}(s', t')$*, then* $s \sim_i t$ *for some* $t \in \mathcal{S}$ *such that* $t \preceq t'$*. Moreover, if* $s \preceq s'$ *and* $s \sim_i t$*, then* $R_i^{Amay}(s', t')$ *for some* t' *such that* $t \preceq t'$*.*

Proof. Suppose that $s \preceq s'$. Then, $R_i^{Amust}(s', t')$ iff $t'_i \leq s'_i$, iff $s'_i \subseteq t'_i$. In particular, for $l = s_i$, $s \preceq s'$ and $s'_i \subseteq t'_i$ imply $l \in t'_i$. Further, if $t' \in I^A$ is initial, then either $t \preceq t'$ for some initial $t \in I$ such that $t_i = l$, and therefore $s \sim_i t$ for $t \in \mathcal{S}$; or for some $t^0 \in I$, $t^0 \preceq t'$, but $t_i^0 \neq l$. However, we assumed that M^A is admissible, that is, $t \preceq t'$ is reachable from $t^0 \preceq t'$. Hence, we obtain a reachable state $t \in \mathcal{S}^A$ such that $t \preceq t'$ and $t_i = l = s_i$, i.e., $s \sim_i t$.

On the other hand, suppose that t' is reachable in M^A via execution $t'^0 \to \ldots \to t'^k$ such that $t'^0 \in I^A$ and $t'^k = t'$. By induction on k, we can prove that there exists an execution $t^0, \ldots, t^{k'}$ in M, for $k' \geq k$, and integers $j' \leq k'$ such that $t^j \preceq t^{j'}$ and $t_i^{k'} = l$. The case for $k = 0$, that is, $t' \in I^A$, goes as above. Hence suppose that the induction hypothesis holds for $k - 1$. Further, we have $t'^{k-1} \to t'^k$. In particular, $v \to v'$ for some $v \preceq t'^{k-1}$ and $v' \preceq t'^k$. Since M^A is admissible, v is reachable from $t^{k'-1}$ and $t^{k'}$ is reachable from v'. Hence, by reasoning similarly to the case for $k = 0$, we obtain an execution $t^0 \to \ldots \to t^{k'}$ in M such that $t^k \preceq t^{k'}$ and $t_i^{k'} = l$. In particular, for $t = t^{k'} \in \mathcal{S}$, we have $s \sim_i t$ and $t \preceq t'$.

Finally, suppose that $s \sim_i t$, that is, $s_i = t_i$. Hence, for $t'_i = s'_i$ we have $t'_i \subseteq t'_i$ and $t'_i \subseteq s'_i$. Moreover, for every $j \neq i$, there exists U_j such that $t_j \in U_j$. Define $t' = \langle U_0, \ldots, U_{i-1}, t_i, U_{i+1}, \ldots, U_{|Ag|} \rangle$. By Def. 6 abstract state t' is reachable in M^A by the same sequence of joint actions as t in M. Hence, $t' \in \mathcal{S}^A$ and $R_i^{Amay}(s', t')$ holds.

Notice that the need for admissibility stems from the presence of spurious executions in the abstract system. Various methodologies have been put forward for refining abstractions w.r.t. spurious behaviours [8]. Here we remark that our notion of admissibility is only meant to preserve reachability, and in general it is not sufficient to preserve more elaborate temporal properties. Nonetheless, it is enough to preserve epistemic properties, as shown by the next result.

Theorem 1. *Let M be a standard IS with admissible abstraction M^A, $s \preceq s'$, and $t \in \{\mathrm{tt}, \mathrm{ff}\}$. Then for every formula ϕ in* $\mathbf{K}$*,*

$$((M^A, s') \models^3 \phi) = t \;\; \textit{implies} \;\; ((M, s) \models^3 \phi) = t$$

Proof. The proof is by induction on the structure of ϕ. We only consider the cases for knowledge formulas, with $\phi = K_i\varphi$. If $((M, s) \models^3 \phi) \neq \mathrm{tt}$ then for some $t \in S$, $s \sim_i t$ and $((M, t) \models^3 \varphi) \neq \mathrm{tt}$. If $s \preceq s'$ and $s \sim_i t$, then by Lemma 1, for some $t' \in \mathcal{S}^A$, $R_i^{Amay}(s', t')$ and $t \preceq t'$. In particular, $((M, t) \models^3 \varphi) \neq \mathrm{tt}$ implies $((M^A, t') \models^3 \varphi) \neq \mathrm{tt}$ by induction hypothesis, that is, $((M^A, s') \models^3 \phi) \neq \mathrm{tt}$. As regards the case for $\phi = K_i\varphi$ being false. If $((M^A, s') \models^3 \phi) = \mathrm{ff}$ then for some $t' \in \mathcal{S}^A$, $R_i^{Amust}(s', t')$ and $((M^A, t') \models^3 \varphi) = \mathrm{ff}$. If $s \preceq s'$ and $R_i^{Amust}(s', t')$, then again by Lemma 1, for some $t \in \mathcal{S}$, $s \sim_i t$ and $t \preceq t'$. In particular, by induction hypothesis we obtain $((M, t) \models^3 \varphi) = \mathrm{ff}$, and therefore $((M, s) \models^3 \phi) = \mathrm{ff}$. The cases for the distributed and common knowledge formulas are proved similarly.

By Proposition 1 and Theorem 1 the next result follows immediately.

Corollary 1. *Let M be a standard IS with admissible abstraction M^A, and $s \preceq s'$. Then for every formula ϕ in* $\mathbf{K}$*,*

$$((M^A, s') \models^3 \phi) = \mathrm{tt} \;\; \textit{implies} \;\; (M, s) \models^2 \phi$$
$$((M^A, s') \models^3 \phi) = \mathrm{ff} \;\; \textit{implies} \;\; (M, s) \not\models^2 \phi$$

By Theorem 1 and Corollary 1 we obtain the following (partial) decision procedure to verify a multi-agent epistemic specification ϕ against infinite-state IS. Given a standard IS M we build an admissible abstraction M^A and then model check ϕ against M^A. If the outcome is either true tt or false ff, then by Corollary 1 we obtain that ϕ is true (resp. false) in M as well. In case that ϕ is undefined in M^A, then no conclusive answer can be drawn. As we mentioned, this limitation is to be expected, since the state-space of M is infinite, and the model checking problem for infinite-state systems is undecidable in general. Nonetheless, we may think of refinement procedures on the abstraction M^A, in order to obtain a refined abstraction M'^A that is able to decide ϕ. We leave abstraction refinement for future work, while here we observe that the abstract

IS M^A depends crucially on the cover $\mathcal{U}_i$ chosen for each agent $i \in Ag$. Here we did not provide details as to how such covers can be effectively found. In most cases of interest covers can be obtained by an analysis of the protocol and transition function of each agents, as well as the specification at hand. We consider an instance of such cases in the following example.

Example 3. We reconsider Example 2. By an analysis of the protocols and transition functions of agents 1 and 2, we identify predicates $p_1 := (x < -4)$, $p_2 := (x = -4)$, $p_3 := (-4 < x < 5)$, $p_4 := (x = 5)$, and $p_5 := (x > 5)$ regarding agent 1, as well as predicates $q_1 := (y < -2)$, $q_2 := (y = -2)$, $q_3 := (-2 < y < 3)$, $q_4 := (y = 3)$, and $q_5 := (y > 3)$ for agent 2. With an abuse of notation, we identify a predicate p with the set of local states satisfying p, as it is customary, for instance, in predicate abstraction [12].

Consider again specification $(x \leq 3) \rightarrow K_1(y \leq 3)$, and a new predicate $p_6 := (-4 < x \leq 3)$. Then, condition $x \leq 3$ can be rewritten as $p_1 \vee p_2 \vee p_6$, and $y \leq 3$ is tantamount to $q_1 \vee q_2 \vee q_3 \vee q_4$. Further, observe that $\mathcal{U}_1 = \{p_1, \ldots, p_6\}$ is a cover of L_1, and $\mathcal{U}_2 = \{q_1, \ldots, q_5\}$ is a cover of L_2 (actually a partition). Then, we define abstract agents 1^A and 2^A such that

- $L_1^A = \mathcal{U}_1 = \{p_1, \ldots, p_6\}$ with order $p_3 \leq p_6$, and $L_2^A = \mathcal{U}_2 = \{q_1, \ldots, q_5\}$;
- $Act_1^A = Act_2^A = Act_1 = Act_2$;
- $Pr_1^A(p) = Act_1$, for all $p \in L_1^A$; and $Pr_2^A(q) = Act_1$, for all $q \in L_2^A$;
- the abstract transition function τ_1^A is such that, for $1 \leq j \leq 5$, $j \neq 3$,

  $$\tau_1^A(p_j, (dec, a_2, ok)) = \{p_j, p_{j-1}\} \qquad \tau_1^A(p_j, (inc, a_2, ok)) = \{p_j, p_{j+1}\}$$

  with the proviso that $p_{j-1} = p_j$ for $j = 1$, and $p_{j+1} = p_j$ for $j = 5$.
  Moreover,

  $$\tau_1^A(p_3, (dec, a_2, ok)) = \{p_2, p_3, p_6\} \qquad \tau_1^A(p_3, (inc, a_2, ok)) = \{p_3, p_4, p_6\}$$
  $$\tau_1^A(p_6, (dec, a_2, ok)) = \{p_6, p_2, p_3\} \qquad \tau_1^A(p_6, (inc, a_2, ok)) = \{p_6, p_3\}$$

  and for all $p \in L_1^A$, $\tau_1^A(p, (a_1, a_2, no)) = p$.
- the abstract transition function τ_2^A is defined similarly to τ_1^A.

Observe that the definitions of the abstract agents 1^A and 2^A are in accordance with Def. 5. Also, the abstract environment e^A is given as follows:

- $L_3^A = \{(p, q) \mid p \in L_1^A, q \in L_2^A\}$;
- $Act_e^A = Act_e$;
- $Pr_e^A((p_i, q_j)) = \{ok\}$ iff $i = 2 \Leftrightarrow j = 2$ or $i = 4 \Leftrightarrow j = 4$; otherwise, $Pr_e^A((p_i, q_j)) = \{no\}$
- we omit the detailed presentation of τ_e^A for reasons of space, but this can be obtained immediately by Def. 5.

Moreover, all agents 1^A, 2^A, and e^A are admissible, as in the concrete IS M, every state is reachable from any other state by an appropriate sequence of actions. The abstract IS M^A is defined on the set $Ag^A = \{1^A, 2^A, e^A\}$ of abstract agents as above, while the set I^A of abstract initial states contains pairs (p_3, q_3)

and (p_6, q_3) only. The abstract global transition function τ^A is defined as in Def. 6, and the labelling of abstract states is immediate. In particular, M^A is a finite-state system.

Now, we check specification $(x \leq 3) \rightarrow K_1(y \leq 3)$ on abstraction M^A. Specifically, if $((M^A, s) \models x \leq 3) = \mathrm{tt}$ then $s_1 = p_1$, $s_1 = p_2$, or $s_1 = p_6$. In the first case, $R_1^{Amay}(s, s')$ implies $s'_1 = p_1$ and $s'_2 = q_1$ or $s'_2 = q_2$. In both cases $((M^A, s') \models y \leq 3) = \mathrm{tt}$. Further, $s_1 = p_2$ and $R_1^{Amay}(s, s')$ imply $s'_1 = p_2$ and $s'_2 = q_2$, and again $((M^A, s') \models y \leq 3) = \mathrm{tt}$. Finally, $s_1 = p_6$ and $R_1^{Amay}(s, s')$ imply $s'_1 = p_6$ or $s'_1 = p_3$. In the former case we have that $s'_2 = q_2$ or $s'_2 = q_3$, and therefore $((M^A, s') \models y \leq 3) = \mathrm{tt}$. In the latter, $s'_2 = q_2$, $s'_2 = q_3$, or $s'_2 = q_4$. In all these cases we obtain $((M^A, s') \models y \leq 3) = \mathrm{tt}$. As a result, if $((M^A, s) \models x \leq 3) = \mathrm{tt}$, then for all $s' \in \mathcal{S}^A$, $R_1^{Amay}(s, s')$ implies $((M^A, s') \models y \leq 3) = \mathrm{tt}$, that is, $((M^A, s) \models K_1(y \leq 3)) = \mathrm{tt}$. Hence, the specification is true in the abstract model, and by the transfer result Theorem 1 we obtain that it holds in the concrete IS M as well.

4 Conclusions

In this paper we introduced a three-value semantics for the multi-agent epistemic logic **K**, based on a notion of order defined on the local states of each agent. Intuitively, in the standard, two-value interpretation of epistemic logic, a notion of i-indistinguishability is defined on global states by the identity of the local states for agent i. Here we generalised this idea by considering a partial order $\leq$ on local state, instead of the identity $=$. This semantic choice allows us to define an abstraction technique, in which local states are bundled together in sets that are then compared according to set-theoretic inclusion. Most importantly, we are able to model check an epistemic specification ϕ on a concrete, infinite-state IS M, by verifying the same formula on some suitable abstraction M^A, and then transfer the result to M by means of Theorem 1. We observe that the abstraction technique developed in Section 3 has a key advantage over similar contributions in [3, 25]. In fact, in [3, 25] abstract states are defined as satisfiable cubes of predicates, which are generated by means of an SMT solver with considerable computational cost. Nothing similar is needed in the present context, where predicates, seen as sets of states, can be arbitrary as long as they satisfy the admissibility condition.

Admittedly, powerful as it is, the proposed methodology has a number of limitations. We provided an heuristic for building the abstraction M^A, by using the predicates mentioned in the system description as well as the specification at hand, but did not provide any algorithmic procedure to build a suitable, finite M^A, nor any correctness proof of such a procedure. Further, we require our predicates, agents, and interpreted system to be admissible, that is, being closed under reachability. While we conjecture that in most cases of interest, this property hold, further investigations are needed on this point. These are all directions we aim to explore in future work, in order to develop a fully automated verification methodology for epistemic properties of infinite-state multi-agent systems. Finally, we plan to implement this verification procedure as an extension of the MCMAS model checker [27].

References

1. Bagheri, B., Calvanese, D., Montali, M., Giacomo, G., Deutsch, A.: Verification of relational data-centric dynamic systems with external services. In: Proceedings of the 32nd Symposium on Principles of Database Systems (PODS13). pp. 163–174. ACM (2013)
2. Ball, T., Kupferman, O.: An abstraction-refinement framework for multi-agent systems. In: Proceedings of the 21st Annual IEEE Symposium on Logic in Computer Science (LICS06). pp. 379–388. IEEE (2006)
3. Belardinelli, F., Lomuscio, A., Michaliszyn, J.: Agent-based refinement for predicate abstraction of multi-agent systems. In: Proceedings of the 22nd European Conference on Artificial Intelligence (ECAI16). pp. 286–294. IOS Press (2016)
4. Belardinelli, F., Lomuscio, A., Patrizi, F.: Verification of agent-based artifact systems. Journal of Artificial Intelligence Research 51, 333–376 (2014)
5. Belardinelli, F.: Model checking auctions as artifact systems: Decidability via finite abstraction. In: Proc. of the 21st European Conference on Artificial Intelligence (ECAI14). pp. 81–86 (2014)
6. Boureanu, I., Kouvaros, P., A.Lomuscio: Verifying security properties in unbounded multi-agent systems. In: Proceedings of the 15th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS16). pp. 1209–1218. IFAAMAS (2016)
7. Bruns, G., Godefroid, P.: Model checking with multi-valued logics. Tech. Rep. ITD-03-44535H, Bell Labs (2003)
8. Clarke, E.M., Grumberg, O., Jha, S., Lu, Y., Veith, H.: Counterexample-guided abstraction refinement. In: Proceedings of the 12th International Conference on Computer Aided Verification (CAV00). Lecture Notes in Computer Science, vol. 1855, pp. 154–169. Springer (2000)
9. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. The MIT Press, Cambridge, Massachusetts (1999)
10. Cohen, M., Dam, M., Lomuscio, A., Qu, H.: A data symmetry reduction technique for temporal-epistemic logic. In: Proceedings of the 7th International Symposium on Automated Technology for Verification and Analysis (ATVA09). Lecture Notes in Computer Science, vol. 5799, pp. 69–83. Springer (2009)
11. Cohen, M., Dam, M., Lomuscio, A., Russo, F.: Abstraction in model checking multi-agent systems. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS09). pp. 945–952. IFAAMAS Press (2009)
12. Das, S., Dill, D., Park, S.: Experience with predicate abstraction. In: Proceedings of the 11th International Conference on Computer Aided Verification (CAV99). pp. 160–171. Springer (1999)
13. De Giacomo, G., Lespérance, Y., Patrizi, F.: Bounded Situation Calculus Action Theories and Decidable Verification. In: Proc. of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR'12). pp. 467–477 (2012)
14. Enea, C., Dima, C.: Abstractions of multi-agent systems. In: Proceedings of 5th International Central and Eastern European Conference on Multi-Agent Systems

(CEEMAS07). Lecture Notes in Computer Science, vol. 4696, pp. 11–21. Springer (2007)
15. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning about Knowledge. MIT Press, Cambridge (1995)
16. Fitting, M.: Many-valued modal logics. Fundamenta Informaticae 15(3-4), 335–350 (1991)
17. Fitting, M.: Many-valued modal logics II. Fundamenta Informaticae 17, 55–73 (1992)
18. Gammie, P., van der Meyden, R.: MCK: Model checking the logic of knowledge. In: Proceedings of 16th International Conference on Computer Aided Verification (CAV04). Lecture Notes in Computer Science, vol. 3114, pp. 479–483. Springer (2004)
19. Godefroid, P., Jagadeesan, R.: On the expressiveness of 3-valued models. In: Proceedings of the 4th International Conference on Verification, Model Checkig, and Abstract Interpretation (VMCAI03). LNCS, vol. 2575, pp. 206–222. Springer-Verlag (2003)
20. Gonzalez, P., Griesmayer, A., Lomuscio, A.: Verification of GSM-based artifact-centric systems by predicate abstraction. In: Proceedings of the 13th International Conference on Service Oriented Computing (ICSOC15). Lecture Notes in Computer Science, vol. 9435, pp. 253–268. Springer (2015)
21. van der Hoek, W., Meyer, J.J.C.: Possible logics for belief. Logique et Analyse 127-128, 177–104 (1989)
22. Kacprzak, M., Nabialek, W., Niewiadomski, A., Penczek, W., Pólrola, A., Szreter, M., Woźna, B., Zbrzezny, A.: Verics 2007 - a model checker for knowledge and real-time. Fundamenta Informaticae 85(1), 313–328 (2008)
23. Katoen, C.B.J.P.: Principles of Model Checking (Representation and Mind Series). The MIT Press (2008)
24. Lomuscio, A., Michaliszyn, J.: Verifying multi-agent systems by model checking three-valued abstractions. In: Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS15). pp. 189–198 (2015)
25. Lomuscio, A., Michaliszyn, J.: Verification of multi-agent systems via predicate abstraction against ATLK specifications. In: Proc. of the 15th Int. Conference on Autonomous Agents and Multiagent Systems (AAMAS16). pp. 662–670 (2016)
26. Lomuscio, A., Penczek, W.: Handbook of Epistemic Logic, chap. Model Checking Temporal Epistemic Logic. College Publications (2015)
27. Lomuscio, A., Qu, H., Raimondi, F.: MCMAS: A model checker for the verification of multi-agent systems. Software Tools for Technology Transfer (2015), http://dx.doi.org/10.1007/s10009-015-0378-x, http://dx.doi.org/10.1007/s10009-015-0378-x
28. Shoham, S., Grumberg, O.: Monotonic abstraction-refinement for CTL. In: Proceedings of the 10th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS04). Lecture Notes in Computer Science, vol. 2988, pp. 546–560. Springer (2004)
29. Wooldridge, M.: An introduction to MultiAgent systems. Wiley, second edition edn. (2009)