An Epistemic Halpern–Shoham Logic

Alessio R. Lomuscio, Jakub Michaliszyn
Department of Computing, Imperial College London, UK
{a.lomuscio,j.michaliszyn}@imperial.ac.uk

Abstract

We define a family of epistemic extensions of Halpern–Shoham logic for reasoning about temporal-epistemic properties of multi-agent systems. We exemplify their use and study the complexity of their model checking problem. We show a range of results ranging from PTIME to PSPACE–hard depending on the logic considered.

1 Introduction

In the past ten years there has been considerable work aimed at verifying specifications of multi-agent systems (MAS) by means of model checking [Clarke *et al.*, 1999]. Several techniques have been developed including binary decision diagrams [Raimondi and Lomuscio, 2005; Gammie and van der Meyden, 2004], symmetry reduction [Cohen *et al.*, 2009] and bounded model checking [Lomuscio *et al.*, 2007]. While in reactive systems plain temporal logic is normally used, several agent-based logical languages have been employed to specify MAS. These include BDI logic [Rao, 1996], ATL [Alur *et al.*, 1996], and temporal-epistemic logic [Fagin *et al.*, 1995]. The latter has received considerable attention due to a number of agent-based applications that benefit from a knowledge-based approach.

Model checking approaches for temporal-epistemic specifications have covered different assumptions on the flow of time. Linear Temporal Logic (LTL) was assumed in some lines [van der Hoek and Wooldridge, 2003], Computational Tree Logic (CTL) was employed in others [Raimondi and Lomuscio, 2005]. These variants share a single underlying principle: formulas are evaluated at a state. However, there is a long and successful tradition in temporal reasoning that explores the implications of interpreting formulas on *intervals*, not at states. Two most popular approaches are the Halpern-Shoham Logic [Halpern and Shoham, 1991] and Interval Temporal Logic [Moszkowski, 1983].

Reasoning about intervals is useful when one needs to express naturally properties that hold continuously between two instances of time. For example, in the interval-based Halpern-Shoham logic (HS) one can naturally state whether in a given interval I, every occurrence of "*request*" (a propositional variable that labels point intervals) during the interval is followed by an occurrence of "*fulfil*", represented by the formula $[E](\langle B\rangle request \Rightarrow \langle D\rangle fulfil \vee \langle E\rangle fulfil)$. HS logic includes twelve modal operators expressing various conditions expressing temporal order between intervals, e.g., "begins", "overlaps", etc. Since the seminal work in the area showed that the satisfiability problem for HS logic is undecidable [Halpern and Shoham, 1991], much of the recent literature has focused on the identification of decidable fragments of HS logic and the assessment of their computational complexity. There are 2^{12} different sub-logics of HS logic; the complexity of their satisfiability problem is now known for the great majority of them [Monica *et al.*, 2011].

Against this background, our long term goals is to develop and use interval-based specifications for reasoning about MAS. We have two objectives for this paper. Firstly, as we are interested in providing powerful languages for the specification of MAS, we put forward a family of epistemic logics based on HS. Secondly, since we are ultimately interested in their use in practice, we define and study their model checking problem. Our results show that natural, knowledge-based concepts can be defined on intervals and that there are fragments of epistemic variants of fragments of HS with a PSPACE-complete model checking problem. Our experience shows that most real-life specifications do not use a deep nesting of knowledge operators. As we show, fragments with bounded knowledge depth are even easier as they are NP-complete, or even in PTIME.

The rest of the paper is organised as follows. In Section 2 we define interval-based interpreted systems, the semantics we introduce for reasoning about intervals and knowledge in a MAS, and define their model checking problem. In Section 3 we introduce an epistemic logic, whose satisfaction is defined on intervals and explore its model checking problem. We extend these investigations in Section 4 by adding a restricted number of interval operators and combine them with epistemic modalities. Section 5 is dedicated to some noteworthy special cases with very attractive computational complexity. We conclude in Section 6, where we also discuss related work.

2 Interval-Based Interpreted Systems

In this section we introduce interval-based interpreted systems, a novel semantics to interpret a variety of temporal and epistemic logics on temporal intervals, and give a general definition of the model checking problem on them.

To begin, we say that an order $(S, \leq)$ is *tree-like* if it is a transitive closure of a tree, i.e., it satisfies the following properties:

- The order is strongly discrete, i.e., the are only finitely many points between any two points.
- The order contains the least element.
- For any $a, b, c \in S$, if $a \leq c$ and $b \leq c$, then either $a \leq b$ or $b \leq a$.

In line with the literature on interpreted systems [Fagin *et al.*, 1995] in the following we assume a finite set of agents $A = \{1, \ldots, m\}$. Each agent is endowed with a set of local states $L_i, i \in A$. We also consider an environment e and its corresponding set of local states L_e. As standard in interpreted systems we assume each agent is equipped with a set of local actions and a local protocol function, which, as usual, through joint-actions, produces a transition relation t. We define systems purely on the basis of the transition relation.

Definition 1. *An* interval-based interpreted system*, or* IBIS*, is a tuple $IS = (S, s_0, t, L)$, where:*

- *The set $S \subseteq L_e \times L_1 \times \cdots \times L_m$ is a set of global states, reachable from s_0 via t;*
- *The state $s_0 \in S$ is the initial global state of the system;*
- *$t \subseteq S^2$ is a transition relation such that $\langle S, t \rangle$ is a tree-like order and s_0 is its least element,*
- *$L : S^2 \to 2^{Var}$ is a labelling function, where Var is a set of propositional variables.*

Given an IBIS IS, an *interval* in IS is a finite path on IS, i.e., a sequence of states $I = s_1 s_2 \ldots s_n$ such that $s_i t s_{i+1}, 1 \leq i \leq (n-1)$. A *point interval* is an interval that consists of exactly one state.

In general interval-based interpreted systems are infinite. To define the model checking problem, we introduce a compact representation of IBIS, called *generalised Kripke structures (GSK)*.

We consider the assumptions of Definition 1.

Definition 2. *A* generalised Kripke structure*, or a* model*, is a tuple $M = (S, s_0, t, L)$, where:*

- *The set $S \subseteq L_e \times L_1 \times \cdots \times L_m$ is the set of reachable global states,*
- *$s_0 \in S$ is an initial global state,*
- *$t \subseteq S^2$ is the transition relation,*
- *$L : S^2 \to 2^{Var}$ is a labelling function for a set of atoms Var.*

We extend the definition of L to intervals by considering $L([a, b]) = L(a, b)$. For a global state $s = (l_e, l_1, \ldots, l_m)$, we denote by $l_i(s)$ the local state $l_i \in L_i$ of agent $i \in A$ in s. Note that, differently from standard temporal-epistemic logic, the structures above admit the labelling of propositions at *pairs* of states and not at a single state.

To see that GKS are compact representations of IBIS, we define their unravelling as follows.

Definition 3. *Given a GKS $M = (S, s_0, t, L)$, its* unravelling *$IS = (\hat{S}, s_o, \hat{t}, \hat{L})$ is defined as follows:*

- *$\bar{S} = \bigcup_{i=0}^{\infty} \bar{S}_i$ is the set of all finite paths in M starting from s_0, i.e., all $\bar{S}_i$ are defined by the induction rules:*
  $\bar{S}_0 = \{s_0\}$;
  $\bar{S}_i = \{s_0 \ldots s_{i+1} \mid s_0 \ldots s_{i-1} \in \bar{S}_{i-1} \wedge (s_{i-1}, s_i) \in t\}$, for $i > 0$.
- *$\hat{t}$ is the transitive closure of the relation $\bar{t}$ defined by: $s \bar{t} s'$ iff $dpt(s') = dpt(s) + 1$ and $source(s)\, t\, source(s')$, where $source((s_0, s_1, \ldots, s_j),\ l_1, \ldots, l_m) = s_j$ and $dpt((s_0, s_1, \ldots, s_j), l_1, \ldots, l_m) = j$.*
- *$\hat{L}$ is defined as $\hat{L}(s, s') = L(source(s), source(s'))$.*

It is easy to see that the unravelling of any GKS is an IBIS. Notice that each IBIS is a GKS, and unravelling of an IBIS results in the same structure. We write IS_M to indicate that IS_M is the unravelling of the GKS M. In the following we assume that the unravelling of a finite path $s_1 \ldots s_n$ is the structure with the universe $\{s_1, \ldots, s_n\}$ and the transition relation $\{(s_i, s_j) | i < j \leq n\}$. We write $\hat{I}$ to denote the unravelling of a path I on M. Since there is one to one correspondence between paths in M and intervals in IS_M, we will often use I and $\hat{I}$ interchangeably.

In the rest of the paper we define a number of interval logics $\mathcal{L}$ of different expressibility. In this context we are interested in studying the complexity of their model checking problem.

Definition 4. *Given a logic $\mathcal{L}$, a GKS $M = (S, s_0, t, L)$, a path $I = s_0 s_1 \ldots s_l$, and a formula $\varphi \in \mathcal{L}$,* the model checking problem for *$\mathcal{L}$ amounts to checking whether or not $IS_M, \hat{I} \models \varphi$.*

The above defines the problem of model checking generalised Kripke structures against specifications in $\mathcal{L}$. Different logics will have a different syntax and may use a different satisfiability notion, but we will refer to the problem as stated above.

We show that our lower bounds hold even under a strong condition known in interval temporal logic literature as the *locality assumption*[Moszkowski, 1983]. We say a model $M = (S, s_0, t, L)$ satisfies the locality assumption if for all $s, s' \in S$, $L([s, s']) = L([s, s])$, that is, the labelling of intervals depends only on the initial point. GKS satisfying the locality assumption can be seen as standard Kripke structures, i.e., Kripke frames paired with an interpretation for the states.

3 Epistemic Interval Temporal Logic

In this section we put forward two epistemic logics for which satisfaction is defined on intervals and not on points. Epistemic Interval Temporal Logic (EIT) is a multi-modal epistemic logic where each operator K_i represents the knowledge of agent $i \in A$. The logic EIT_C is the extension of EIT to common knowledge over groups $G \subseteq A$.

3.1 The logics EIT and EIT_C

We start by defining the syntax for the logics EIT and EIT_C.

Definition 5. *The syntax of EIT is given by the following BNF:*

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi$$

The syntax of EIT$_C$ is given by the following BNF:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid C_G\varphi$$

where $p \in Var$ is a propositional variable, $i \in A$ is an agent, and $G \subseteq A$ is a set of agents.

Formulas built according to Definition 5 are the usual epistemic logics in their static versions with and without common knowledge.

In the following we use the Boolean connectives $\vee$, $\Rightarrow$, $\Leftrightarrow$ and constants $\top, \bot$ introduced in the standard way. We also use the dual operator for epistemic possibility $\bar{K}_i$ defined as $\bar{K}_i\varphi = \neg K_i \neg\varphi$. As usual, let $E_G\varphi$ stand for $\bigwedge_{i\in G} K_i\varphi$ and define E^i_G as $E^0_G\varphi = \varphi$ and $E^{n+1}_G\varphi = E_G E^n_G\varphi$.

We now define when an epistemic formula is satisfied in an interval in an IBIS.

Definition 6 (Satisfaction)**.** *Given a formula $\varphi \in EIT_C$, an IBIS IS, and an interval I, we inductively define whether φ holds in the interval I, denoted $IS, I \models \varphi$, as follows:*

- *For all $p \in Var$, we have $IS, I \models p$ iff $p \in L(I)$.*
- *$IS, I \models \neg\varphi$ iff it is not the case that $IS, I \models \varphi$.*
- *$IS, I \models \varphi_1 \wedge \varphi_2$ iff $IS, I \models \varphi_1$ and $IS, I \models \varphi_2$.*
- *$IS, I \models K_i\varphi$, where $i \in A$, iff for all $I' \sim_i I$ we have $IS, I' \models \varphi$.*
- *$IS, I \models C_G\varphi$, where $G \subseteq A$, iff for all $n \in \mathbb{N}$ we have $IS, I \models E^n_G\varphi$.*

The satisfaction definition for EIT is obtained from Definition 6 by omitting the last clause. The key clause in Definition 6 is the one relating to the epistemic modality K_i. Intuitively, agent i knows that φ in an interval I if φ holds in all intervals I' which are epistemically indistinguishable to I, denoted $I \sim_i I'$. This is defined by considering all global states in the interval I and ensuring they are pointwise epistemically equivalent to the corresponding states in I'. Formally, define as usual [Fagin *et al.*, 1995] that two global states g, g' are such that $g \sim_i g'$ iff $l_i(g) = l_i(g')$, i.e., two global states are epistemically equivalent for agent i if its local states are the same in the two global states. We say that two intervals $I = s_1, \ldots s_k, I' = s'_1, \ldots s'_l$ are such that $I \sim_i I'$ iff $k = l$ and for all $j \le k$, $l_i(s_j) = l_i(s'_j)$. In other words an agent i cannot distinguish between the corresponding states in the intervals I, I'. This notion corresponds to other concepts in computer science, including the notion of observational equivalence in security [Honda and Yoshida, 1995]. Intuitively, it is the natural extension from local state-equivalence to local path-equivalence in an epistemic setting.

Example 7. *Consider the structure from Figure 1 and the interval $I = s_1s_2s_3$. A formula K_1p is not satisfied in I because there is an interval $s_3s_2s_3 \sim_1 I$ that is not labelled by p. A formula $K_1\neg K_1 p$ is satisfied in I because for all intervals $I' \sim_1 I$ we have that $I' \sim_1 s_3s_2s_3$, so I' does not satisfy K_1p.*

Example 8. *Consider a structure containing the states s and t, such that the pair (s, t) is the only one labelled by a propositional variable p. Consider an agent 1 that cannot distinguish any states, i.e., we have $u \sim_1 u'$ for all states u, u', and the formula $\varphi = \bar{K}_1 p$. Formula φ is true in an interval of length n iff there is a path from s to t of length n.*

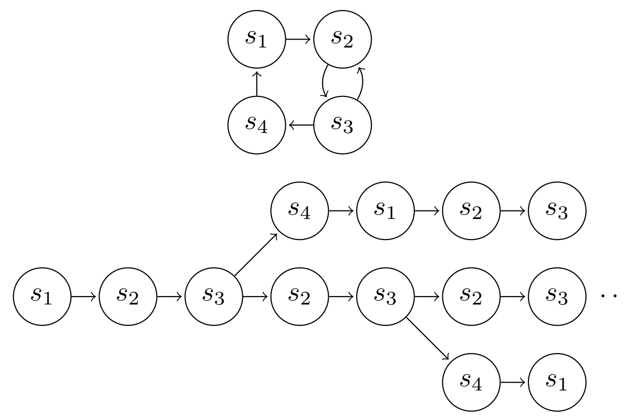

Figure 1: An example of a GKS M (top) and its unravelling IS_M (bottom). The edges that follow from the transitivity of the transition relation in case of IS_M are omitted for readability. The labelling L of M is such that $L(s_1, s_3) = \{p\}$ and $L(s, s') = \emptyset$ for all other pairs of states. Agent 1 cannot distinguish only between s_1 and s_3.

3.2 The complexity of model checking GKS against EIT and EIT$_C$

In this subsection we show that the model checking problem for EIT$_C$ can be solved in polynomial space. Then, we show that this bound is tight, proving that the model checking problem is PSPACE-hard, even for the logic EIT on structures satisfying the locality assumption. In Section 5 we discuss interesting fragments of the logic EIT$_C$ for which better complexity can be shown.

Theorem 9. *The model checking problem for the EIT logic is in* PSPACE.

Proof. We show an alternating algorithm that solves the problem in polynomial time. Since APTIME = PSPACE, this leads to the membership is PSPACE.

The algorithm $solve\_eitc(M, I, \varphi)$ checks whether a formula φ is satisfied in an interval I of a model M recursively. In case of a formula of the form $K_i\varphi$, the algorithm calls itself universally for all (equivalent) intervals. In case of a formula of the form $C_G\varphi$ it first computes the equivalence relation corresponding to C_G using a standard fixed-point approach, and then proceeds as in the case of $K_i\varphi$.

Algorithm $solve\_eitc(M, I, \varphi)$ **Output:** $True/False$

1. If $\varphi = p$, then if I is labelled by p then return $True$, else return $False$.
2. If $\varphi = \neg\varphi'$, then return not $solve\_eitc(M, I, \varphi')$.
3. If $\varphi = \varphi_1 \wedge \varphi_1$, then return the logical and between the results of $solve\_eitc(M, I, \varphi_1)$ and $solve\_eitc(M, I, \varphi_2)$.
4. If $\varphi = K_i\varphi'$, then compute $solve\_eitc(M, I', \varphi')$ for all intervals I' such that $I \sim_i I'$. If $solve\_eitc(M, I', \varphi')$ returns $False$ for some of them, then return $False$, else return $True$.
5. If $\varphi = C_G\varphi'$, then compute the smallest equivalence relation $\sim_G$ containing all the relations $\sim_i$ for $i \in G$,

and then call $solve\_eitc(M, I', \varphi')$ analogously to the step above but by considering $\sim_G$ in place of $\sim_i$.

Clearly, the algorithm solves the model checking problem and requires only polynomial time. $\square$

We show the lower bound for the logic EIT.

Theorem 10. *The model checking problem for EIT is hard for* PSPACE, *even if we consider only structures that satisfy the locality assumption.*

Proof. We reduce the quantified Boolean formula (QBF) problem to the model checking for EIT with the locality assumption. For a formula $\Psi = Q_1 p_1 Q_2 p_2 \ldots Q_n p_n.\rho$, where $Q_i \in \{\forall, \exists\}$ and ρ is quantifier-free, we define a structure $M = (S, \$, t, L)$ with the set of states $S = \{T, \$\} \cup \{1, \ldots, n\} \times \{F, T\}$. We are interested in intervals of the form $\$(1, v_1)(2, v_2) \ldots (n, v_n)$, where each $v_i \in \{F, T\}$ represents a valuation of p_i. The only propositional variable in the model is T, and it labels only intervals starting from T, i.e., all the pairs (T, s). The relation t is such that from \$ there is a transition to every other state; from T there is a transition only to T; from (i, j) there is a transition to $(i+1, j')$ for any $j, j' \in \{F, T\}$.

We use $2n$ agents. For $i \leq n$, agent i cannot distinguish only between the states (i, T) and (i, F), and agent $n+i$ can distinguish two states only if one of them is (i, F).

Consider an interval $I = \$(1, v_1) \ldots (n, v_n)$. The formula $\bar{K}_{n+i} T$ is satisfied in I iff $I \sim_{n+i} T \ldots T$, that is, by the definition of agent $n+i$, iff $(i, v_i) \sim_{n+i} T$, that is iff $v_i = \top$. Clearly, $v_i = \top$ iff I represents an interval in which p_i is true.

Let φ_{n+1} be a result of replacing in ρ every p_i by $\bar{K}_{n+i} T$. By the above observation, an interval I satisfies φ_{n+1} iff it represents a valuation that satisfies ρ.

For quantifiers, consider the formula $K_i \varphi$. This formula is satisfied in an interval $I = \$(1, v_1) \ldots (n, v_n)$ iff for all $j \in \{F, T\}$, the formula φ is satisfied in $\$(1, v_1) \ldots (i-1, v_{i-1})\,(i, j)\,(i+1, v_{i+1}) \ldots (n, v_n)$, that is, iff φ is satisfied at all the intervals that represent the same valuation of propositional variables, possibly except for p_i. Similarly, the existential quantifier can be simulated by a formula $\bar{K}_i \varphi$.

To finish the reduction, we define formulas $\varphi_n, \ldots, \varphi_1$ recursively. If $Q_i = \forall$, then we set $\varphi_i = K_i \varphi_{i+1}$, and otherwise we set $\varphi_i = \bar{K}_i \varphi_{i+1}$. The above observations show that an interval $\$(1, F) \ldots (n, F)$ satisfies φ_1 iff Ψ is true. $\square$

We conclude this section by studying the complexity of the satisfiability problem, i.e., whether or not a given formula φ admits an IBIS IS and an interval I such that $IS, I \models \varphi$.

Theorem 11. *The satisfiability problem for EIT is* PSPACE*–complete; the satisfiability problem for EIT$_C$ is* EXPTIME*–complete.*

Proof. Let $\mathrm{S5}_*$ be a multimodal modal logic interpreted over structures with equivalence relations $S_1, S_2, \ldots$ associated to the modal operators $\Box_1, \Box_2, \ldots$, and $\mathrm{S5C}_*$ be an extension of $\mathrm{S5}_*$ by adding the common knowledge operator. Let f be a bijection that transforms the formulas of EIT$_C$ by replacing each K_i by $\Box_i$. We claim that any EIT (resp., EIT$_C$) formula φ is satisfiable iff $f(\varphi)$ is satisfiable in the logic $\mathrm{S5}_*$ (resp., $\mathrm{S5C}_*$), that is, that f and f^{-1} are polynomial-time reductions between the satisfiability problems for EIT (resp., EIT$_C$) and $\mathrm{S5}_*$ (resp., $\mathrm{S5C}_*$). The complexity of the satisfiability problem for $\mathrm{S5}_*$ and $\mathrm{S5C}_*$ follows from the results for $\mathrm{S5_m}$ and $\mathrm{S5C_m}$ [Halpern and Moses, 1992; Fagin *et al.*, 1995].

Assume that φ is satisfied in a model $IS = (S, s_0, t, L)$ and an interval I. Consider a Kripke structure $M = (W, s_0, S_1, \ldots, S_m, L)$ such that W consists of the intervals of IS, and for any worlds $\mathcal{J}, \mathcal{J}'$ of M, we have that $\mathcal{J} S_i \mathcal{J}'$ iff $\mathcal{J} \sim_i \mathcal{J}'$ in IS. By induction one can see that $f(\varphi)$ is satisfied in the world I in M.

Conversely, assume that a formula $f(\varphi)$ is satisfied in a world w of a model $M = (W, w_0, S_1, \ldots, S_m, \Pi)$. Assume that M contains only the relations in φ. We construct a model $IS = (W \cup \{s_0\}, s_0, t, L)$, where s_0 is a fresh initial state. The transition relation t contains an edge (s, t) iff $s = s_0$. For each i, the relation $\sim_i$ contains (s_0, s_0) and all the pairs from S_i. So we have $ss' \sim_i tt'$ iff $s = t = s_0$ and $s' S_i t'$. We set $L(s_0, w) = \pi(w)$. By induction we can see that φ is satisfied in the interval $s_0 w$ in IS. $\square$

The algorithm solving the model checking for EIT (resp., EIT$_C$) works as follows. For a given formula φ, replace all occurrences of K_i by $\Box_i$, and check the satisfiability of the resulting formula in logic $\mathrm{S5}_*$ (resp., $\mathrm{S5C}_*$).

4 Epistemic Halpern–Shoham Logic

As discussed, the logic EIT$_C$ contains no temporal operators. In this section, we enrich EIT$_C$ by means of interval-based Halpern–Shoham modalities.

The temporal operators in the Halpern–Shoham logic represent temporal relations between intervals as originally defined by Allen [Allen, 1983]. Six of these relations are presented in Figure 2: R_A ("**a**fter" or "meets"), R_B ("**b**egins" or "starts"), R_D ("**d**uring"), R_E ("**e**nds"), R_L ("**l**ater"), and R_O ("**o**verlaps"). Six additional operators can be defined corresponding to the six symmetric relations. Formally, for each $X \in \{A, B, D, E, L, O\}$, we also consider the relation $R_{\bar{X}}$, corresponding to ${R_X}^{-1}$.

$[x, y] R_A [x', y']$ iff $x' = y$ and $x' < y'$

$[x, y] R_B [x', y']$ iff $x' = x$ and $y' < y$

$[x, y] R_D [x', y']$ iff $x' < x$ and $y' < y$

$[x, y] R_E [x', y']$ iff $y' = y$ and $x' > x$

$[x, y] R_L [x', y']$ iff $x' > y$

$[x, y] R_O [x', y']$ iff $x < x' < y < y'$

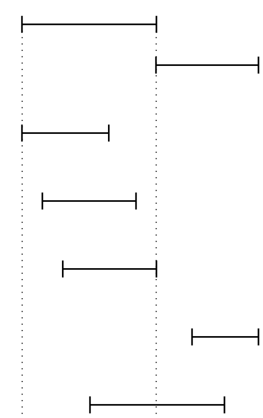

Figure 2: Basic Allen's relations.

Definition 12. *The syntax of the Epistemic Halpern–Shoham Logic (EHS) is defined by the following BNF.*

$$\begin{aligned}\varphi \quad ::= \quad & p \mid \neg\varphi \mid \varphi\wedge\varphi \mid K_i\varphi \mid C_G\varphi \mid \\ & \langle A\rangle\varphi \mid \langle \bar{A}\rangle\varphi \mid \langle B\rangle\varphi \mid \langle \bar{B}\rangle\varphi \mid \langle D\rangle\varphi \mid \langle \bar{D}\rangle\varphi \mid \\ & \langle E\rangle\varphi \mid \langle \bar{E}\rangle\varphi \mid \langle L\rangle\varphi \mid \langle \bar{L}\rangle\varphi \mid \langle O\rangle\varphi \mid \langle \bar{O}\rangle\varphi\end{aligned}$$

where $p \in Var$ is a propositional variable, $i \in A$ is an agent, and $G \subseteq A$ is a set of agents.

We write $[X]\varphi$ for $\neg\langle X\rangle\neg\varphi$.

Definition 13 (Satisfaction)**.** *Given a formula $\varphi \in$ EHS, an IBIS IS, and an interval I, we define whether φ holds in the interval I, denoted $IS, I \models \varphi$, by adding the following clause to those reported in Definition 6.*

6. *$M, I \models \langle X\rangle\varphi$ iff there exists an interval I' such that $IR_X I'$ and $M, I' \models \varphi$, where R_X is an Allen's relation as above.*

We often use the operator $[G]$ defined as

$$[G]\varphi = \varphi \wedge \bigwedge_{X\in\{A,\bar{A},B,\bar{L}\}} [X](\varphi \wedge [A]\varphi \wedge [L]\varphi)$$

which can be read as "in all the (reachable) intervals φ holds" and its existential version $\langle G\rangle\varphi = \neg[G]\neg\varphi$.

Example 14. *Consider the structure from Figure 1 and the interval $I = s_1 s_2 s_3$. We have $IS, I \models [G]\neg K_1 p$ because each interval labelled by p is of the form $s_1 t_1 \dots t_n s_3$ (where t_i stands for s_1, s_2, s_3 or s_4 as appropriate) and agent 1 cannot distinguish it from $s_1 t_1 \dots t_n s_1$, which does not satisfy p. By contrast, we have $IS, I \not\models [A]K_1\neg p$. Indeed the path $I' = s_3 s_2 s_3$ is such that $IR_A I'$, but I' does not satisfy $K_1\neg p$ since the interval $s_1 s_2 s_3 \sim_1 I'$ is labelled by p.*

Example 15. *Consider a structure containing intervals labelled by propositional variables s and t. The formula*

$$\langle G\rangle(\langle B\rangle(s \wedge pi) \wedge \langle E\rangle(t \wedge pi) \wedge ([D](pi \Rightarrow K_1 safe)))$$

where $pi = [B]\bot$ is a formula satisfied only in point intervals, states that there is a path from a point-interval satisfying s to a point-interval satisfying t through which Agent 1 knows that $safe$ holds.

Note that we consider tree-like orders, but not necessarily linear, and we allow for point intervals. It is known that over such orders, even the operator $\langle D\rangle$ can on its own represent successful runs of Turing machines [Marcinkowski and Michaliszyn, 2011].

4.1 BDE fragment of EHS logic

The undecidability of the satisfiability problem for EHS follows from the undecidability of HS [Halpern and Shoham, 1991]. It is however instructive to identify fragments of EHS that lead to comparatively attractive model checking problems. We here confine ourselves to consider the BDE fragment of EHS logic, i.e., the logic defined in Definition 12 by considering only the clauses for the modalities $\langle B\rangle$, $\langle D\rangle$, $\langle E\rangle$, and K_i and C_G. We show the following result.

Theorem 16. *The model checking problem for the BDE fragment of EHS is* PSPACE*–complete.*

Proof. The lower bound follows from Theorem 9; so we only show the upper bound. We extend the alternating algorithm from the proof of Theorem 9 by adding the relevant temporal cases. In the case of a formula of the form $\langle B\rangle\varphi$, $\langle D\rangle\varphi$, or $\langle E\rangle\varphi$, the revised algorithm guesses a new interval (resp., prefix, infix of suffix of the current one) and calls itself recursively.

Formally, the algorithm, called $solve\_ehs$, extends the algorithm $solve\_eitc$ by the following rules.

6. If $\varphi = \langle B\rangle\varphi'$, then for all k s.t. $1 \le k < l$ compute $solve\_ehs(\mathcal{M}, s_1 \dots s_k, \varphi')$. Return $True$ if at least one sub-process returned $True$.
7. If $\varphi = \langle D\rangle\varphi'$, then for all k, k' s.t. $1 < k < k' < l$ compute $solve\_ehs(\mathcal{M}, s_k \dots s_{k'}, \varphi')$. Return $True$ if at least one sub-process returned $True$.
8. If $\varphi = \langle E\rangle\varphi'$, then for all k s.t. $1 < k \le l$ compute $solve\_ehs(\mathcal{M}, s_k \dots s_l, \varphi')$. Return $True$ if at least one sub-process returned $True$.

Clearly, the algorithm requires only polynomial time. □

The undecidability of the satisfiability problem for the BDE fragment of EHS follows by the one of the logic containing $\langle B\rangle$, $\langle D\rangle$, and $\langle E\rangle$ only [Bresolin *et al.*, 2011].

5 The Complexity of Some Noteworthy Fragments

In this section we show that some interesting fragments of EHS logic are endowed with an easier model checking problem. We start by defining the *knowledge depth* of an EHS formula φ (denoted $KD(\varphi)$) as follows.

- $KD(p) = 0$ where p is a propositional variable.
- $KD(\varphi_1 \wedge \varphi_2) = max(KD(\varphi_1), KD(\varphi_2))$.
- $KD(\neg\varphi) = KD(\langle X\rangle\varphi) = KD(\varphi)$.
- $KD(K_i\varphi) = KD(C_G\varphi) = KD(\varphi) + 1$.

We assume that the reader is familiar with the concept of the polynomial hierarchy [Papadimitriou, 1994].

Theorem 17. *The model checking problem for EIT$_C$ specifications φ, such that $KD(\varphi) \le k$ for some $k \ge 1$, is Δ^P_{k-1}–complete.*

Proof. (Sketch). The lower bound simply follows by applying the reduction from Theorem 10 to the QBF$_{k-1}$ problem, that is, the QBF problem restricted to formulae with quantifier depth bounded by $k-1$.

The algorithm for the upper bound proceeds as the algorithm for EIT$_C$, except for the case of the formulae of the form $K_i\varphi$ or $C_G\varphi$, where $KD(\varphi) = 0$. For the case of $K_i\varphi$ (resp., $C_G\varphi$) with $KD(\varphi) = 0$, the algorithm first generates the list of all pairs (t_1, t_l) such that there is an interval $I' = t_1 \dots t_l$ s.t. $I \sim_i I'$ (resp., $I \sim_G I'$, where $\sim_G$ is the smallest equivalence relation containing all the relations $\sim_i$ for $i \in G$). Then, for each pair (t_1, t_l) found, it checks whether the knowledge-free formula φ is true in an interval labelled by $L(t_1, t_l)$. The algorithm returns $True$ if at least one is and $False$ otherwise.

What remains to be explained is how to compute, for a given equivalence relation $\sim$ (which may stand for $\sim_i$ or $\sim_G$) and an interval $s_1 \ldots s_n$, the set of pairs (t_1, t_n) s.t. there is an interval $t_1 \ldots t_n \sim s_1 \ldots s_n$. This can be done in polynomial time by the function *find_all* defined as follows.

Algorithm *find_all*$(M, \sim, s_1 \ldots s_n)$ **Output**: Set of pairs

1. Let $R_1 := \{(t,t) | t \sim s_1\}$.
2. For each $i = 2, 3, \ldots, n$:
   - Set $R_i := \emptyset$.
   - For each $(t_1, t_{i-1}) \in R_{i-1}$:
     – For each state t_i such that $t_i \sim s_i$, if there is a transition from t_{i-1} to t_i, then add (t_1, t_i) to R_i.
3. Return R_n.

The algorithm runs in time $O(n \cdot |S|^3)$, where S is the set of states of M. □

The result above does not hold for EHS, as even for formulae of knowledge depth 1 we obtain a Δ_k^1–hard model checking problem. Below we define a variant whose model checking problem is Δ_k^P–complete for formulae of knowledge depth bounded by k.

Definition 18. *The syntax of the logic EHSP is defined by the following BNF expression*

$$\begin{array}{lll} \varphi & ::= & \psi \mid \neg\varphi \mid \varphi \wedge \varphi \mid K_i\varphi \mid C_G\varphi \\ \psi & ::= & p \mid \neg\psi \mid \psi \wedge \psi \mid \langle B\rangle\psi \mid \langle D\rangle\psi \mid \langle E\rangle\psi \end{array}$$

where $p \in Var$ is a propositional variable, $i \in A$ is an agent, and $G \subseteq A$ is a set of agents.

Theorem 19. *The model checking problem for EHSP specifications φ, such that $KD(\varphi) \le k$ for some $k \ge 0$, is Δ_k^P–complete.*

Proof. (Sketch). To obtain the lower bound, we modify the reduction given in the proof of Theorem 10 in the following way. We use additional propositional variables $1, \ldots, n$ and label each pair of states of the form $((i, T), s)$, where $s \in S$ by i; for each $i < n$ we also replace the formula $\bar{K}_{n+i}\top$ with $\langle D\rangle i$ and $\bar{K}_{2n}\top$ with $\langle E\rangle i$. It can be checked that the result then follows.

For the upper bound we extend the algorithm for EIT$_C$ by the following rule.

6. If $KD(\varphi) = 0$, then for all subintervals of I and all subformulae of φ, determine which of the formulae are satisfied in which subintervals (starting from formulae of smallest modal depth). Return $True$ if I satisfies φ and $False$ otherwise.

This step can be performed in time $|I|^2 \cdot |\varphi|$, which is clearly polynomial in size of the input. □

It follows that model checking specifications with no knowledge operators is in PTIME.

Finally, we present an NP–complete fragment of EHS.

Definition 20. *The syntax of the logic EHS$_\exists$ is defined by the following BNF expression.*

$$\begin{array}{lll} \varphi & ::= & p \mid \neg p \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \bar{K}_i\varphi \mid \\ & & \bar{C}_G \mid \langle B\rangle\varphi \mid \langle D\rangle\varphi \mid \langle E\rangle\varphi \end{array}$$

where $p \in Var$ is a propositional variable, $i \in A$ is an agent, and $G \subseteq A$ is a set of agents.

Satisfaction for EHS$_\exists$ is defined as in Definition 13 by assuming that $\bar{C}_G\varphi$ is an abbreviation for $\neg C_G \neg\varphi$.

Theorem 21. *The model checking problem for EHS$_\exists$ is* NP–*complete.*

Proof. (Sketch). The lower bound follows by applying the reduction given in the proof of Theorem 10 to existential formulae. We show the upper bound by describing the algorithm $solve\_ehs_\exists$ which consists of steps 1-3 from $solve\_eitc$ to which we add the following:

4. If $\varphi = \bar{K}_i\varphi'$, then guess I' such that $I \sim_i I'$ and return the value of $solve\_ehs_\exists(M, I', \varphi')$.
5. If $\varphi = \bar{C}_G\varphi'$, then compute the smallest equivalence relation $\sim_G$ containing all the relations $\sim_i$ for $i \in G$, guess I' such that $I \sim_G I'$ and return the value of $solve\_ehs_\exists(M, I', \varphi')$.
6. If $\varphi = \langle X\rangle\varphi'$, then guess an interval I' such that IR_XI' and return the value of $solve\_ehs_\exists(M, I', \varphi')$.

The algorithm runs in non-deterministic polynomial time. □

6 Conclusions and Future Work

In the Introduction we pointed out that: i) while the satisfaction problem for interval temporal logic has been studied in considerable detail, no results are known for its model checking problem, and ii) while intervals have been used in various areas in computer science, they are still relatively unexplored in AI and MAS. In this paper we firstly put forward a semantics for epistemic modalities in the context of intervals and defined epistemic logics with and without interval operators. We argued that the notion we arrived to is the natural extension to intervals from the point-based epistemic interpretations commonly used in the literature.

Secondly, we introduced the model checking problem for these logics and studied it for a number of cases of interest. We proved that the powerful epistemic logic EIT$_C$ with individual and common knowledge as well as three interval operators has a PSPACE-complete model checking problem. While we see this as positive result given the expressiveness of EIT$_C$, we also identified fragments with better complexity, including PTIME, for logics with a limited number of nested epistemic operators. All the results hold as well for the logics without the common knowledge operator.

In the future we would like to identify the complexity of the model checking problem for the full logic EHS as well as the $AB\bar{B}DELO$ fragment, i.e., the fragment without past modalities. We are also interested in exploring how compact representations of the type considered in [Lomuscio and Raimondi, 2006] affect the results in this paper and how intervals can be related to clocks in the context of verification [Lomuscio *et al.*, 2007].

Acknowledgements. This research was supported by the EC STREP Project "ACSI" (grant no. 257593).

References

[Allen, 1983] J. Allen. Maintaining knowledge about temporal intervals. *Communications of ACM*, 26(11):832–843, November 1983.

[Alur *et al.*, 1996] R. Alur, T. Feder, and T. Henzinger. The benefits of relaxing punctuality. *Journal of the ACM*, 43(1):116–146, 1996.

[Bresolin *et al.*, 2011] D. Bresolin, D. Della Monica, V. Goranko, A. Montanari, and G. Sciavicco. The dark side of interval temporal logic: Sharpening the undecidability border. In *Proceedings of 18th International Symposium on Temporal Representation and Reasoning, TIME'11*, pages 131–138. IEEE, 2011.

[Clarke *et al.*, 1999] E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking*. The MIT Press, Cambridge, Massachusetts, 1999.

[Cohen *et al.*, 2009] M. Cohen, M. Dam, A. Lomuscio, and H. Qu. A symmetry reduction technique for model checking temporal-epistemic logic. In C. Boutilier, editor, *Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI'09*, pages 721–726, Pasadena, USA, 2009.

[Fagin *et al.*, 1995] R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge, 1995.

[Gammie and van der Meyden, 2004] P. Gammie and R. van der Meyden. MCK: Model checking the logic of knowledge. In *Proceedings of 16th International Conference on Computer Aided Verification, CAV'04*, volume 3114 of *LNCS*, pages 479–483. Springer-Verlag, 2004.

[Halpern and Moses, 1992] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[Halpern and Shoham, 1991] J. Halpern and Y. Shoham. A propositional modal logic of time intervals. *Journal of The ACM*, 38:935–962, 1991.

[Honda and Yoshida, 1995] K. Honda and N. Yoshida. On reduction-based process semantics. *Theoretical Computer Science*, 152:437–486, 1995.

[Lomuscio and Raimondi, 2006] A. Lomuscio and F. Raimondi. The complexity of model checking concurrent programs against ctlk specifications. In Matteo Baldoni and Ulle Endriss, editors, *DALT*, volume 4327 of *Lecture Notes in Computer Science*, pages 29–42. Springer, 2006.

[Lomuscio *et al.*, 2007] A. Lomuscio, B. Woźna, and W. Penczek. Bounded model checking knowledge and real time. *Artificial Intelligence*, 171(16-17):1011–1038, 2007.

[Marcinkowski and Michaliszyn, 2011] J. Marcinkowski and J. Michaliszyn. The ultimate undecidability result for the Halpern-Shoham logic. In *Proceedings of Twenty-Sixth Annual IEEE Symposium on logic in computer science, LICS'11*, pages 377–386. IEEE Computer Society, 2011.

[Monica *et al.*, 2011] D. Della Monica, V. Goranko, A. Montanari, and G. Sciavicco. Expressiveness of the interval logics of Allen's relations on the class of all linear orders: complete classification. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Two*, IJCAI'11, pages 845–850. AAAI Press, 2011.

[Moszkowski, 1983] B. Moszkowski. *Reasoning about digital circuits*. PhD thesis, Stanford University, Stanford, CA, USA, 1983.

[Papadimitriou, 1994] C. Papadimitriou. *Computational Complexity*, chapter 17, Polynomial hierarchy, pages 409–438. Addison-Wesley, 1994.

[Raimondi and Lomuscio, 2005] F. Raimondi and A. Lomuscio. Automatic verification of multi-agent systems by model checking via OBDDs. *Journal of Applied Logic*, Vol 5(2):235–251, 2005.

[Rao, 1996] A. Rao. Decision procedures for propositional linear-time Belief-Desire-Intention logics. In M. Wooldridge, J. P. Müller, and M. Tambe, editors, *Intelligent Agents II, LNAI 1037*, pages 33–48. Springer-Verlag: Heidelberg, Germany, 1996.

[van der Hoek and Wooldridge, 2003] W. van der Hoek and M. Wooldridge. Model checking cooperation, knowledge, and time - a case study. *Research In Economics*, 57(3):235–265, September 2003.