Model Checking Epistemic Halpern-Shoham Logic Extended with Regular Expressions

Alessio Lomuscio and Jakub Michaliszyn
Imperial College London, UK

Abstract. The Epistemic Halpern-Shoham logic (EHS) is a temporal-epistemic logic that combines the interval operators of the Halpern-Shoham logic with epistemic modalities. The semantics of EHS is based on interpreted systems whose labelling function is defined on the endpoints of intervals. We show that this definition can be generalised by allowing the labelling function to be based on the whole interval by means of regular expressions. We prove that all the positive results known for EHS, notably the attractive complexity of its model checking problem for some of its fragments, still hold for its generalisation. We also propose the new logic EHS\textsuperscript{RE} which operates on standard Kripke structures and has expressive power equivalent to that of EHS with regular expressions. We compare the expressive power of EHS\textsuperscript{RE} with standard temporal logics.

1 Introduction

Model checking is a leading technique in automatic verification. The model checking problem consists of establishing whether a property, expressed as a logical formula, holds on a system, represented as a model [11]. Model checking has recently been studied in the context of interval temporal logic [22, 24]. In this context temporal specifications consist of formulas expressing properties of intervals rather than states as it is traditionally the case in temporal logic.

Interval temporal logic has a long and successful tradition in Logic in Computer Science. The logics ITL [31], defined by Moszkowski, and HS [17], defined by Halpern and Shoham, are the most commonly used formalisms. ITL suffers from the high-complexity of its model checking problem which is non-elementary-complete [21]. In this paper we focus on HS as the basic underlying framework. HS is a modal temporal logic in which the elements of a model are pairs of points in time, or intervals. For an interval \([p, q]\) it is assumed that \(q\) happens no earlier than \(p\), but no assumption is made on the underlying order, which can be discrete, continuous, linear, branching, etc.

Traditionally, twelve modal operators acting on intervals are defined in HS. They are: \(A\) (“after/ meets”), \(B\) (“begins”), \(D\) (“during”), \(E\) (“ends”), \(L\) (“later”), \(O\) (“overlaps”) and their duals: \(\overline{A}\), \(\overline{B}\), \(\overline{D}\), \(\overline{E}\), \(\overline{L}\), \(\overline{O}\). Some of them are redundant; for example, \(B\) and \(E\) can define \(D\) (a prefix of a suffix is an infix) [14][15].

The analysis of HS and its fragments is traditionally limited to its satisfiability problem. This is known to be undecidable in general [8][13][17], even when
HS is restricted to its unimodal fragments \cite{9}. Notable decidable fragments are the $AA$ fragment with length constraints \cite{7}, the $ABBL$ fragment \cite{20}, and the recently identified Horn fragment \cite{4}. Some fragments are decidable only over some particular classes of orderings. For example, the $BBD\overline{DL}$ fragment was shown to be decidable over the class of all dense orders \cite{30}, while the $D$ fragment is undecidable over discrete orders \cite{25}. The same logic becomes decidable if one assume that an interval is its own infix \cite{29}. While a wealth of results have been put forward, open questions remain. For example, the decidability of the $D$ fragments over the class of all orders is currently open.

**The logic EHS.** In applications, temporal logics often appear in combination with other modalities expressing other aspects of the system or its components. A notable example is temporal-epistemic logic \cite{16} where the knowledge of the components, or agents, is assessed from an information-theoretic point of view. Temporal-epistemic logic is widely explored in applications, including security; dedicated model checkers have been released \cite{1, 2, 23}.

In the traditional approach, the underlying temporal logic is state-based, either in its linear or branching variants. A notable exception to this is the Epistemic HS logic (EHS) \cite{22}, which consists of a combination of epistemic modalities with the interval-based temporal logic HS. EHS combines all the HS interval-temporal modalities with standard epistemic modalities: $K_i$ (“agent $i$ knows that”) and $C_\Gamma$ (“it is a common knowledge in group of agents $\Gamma$ that”). The logic EIT, a simple fragment of EHS where only epistemic modalities are allowed, but modalities are interpreted on intervals rather than points, has been shown to be PSPACE-hard. Model checking of the $BDE$-fragment of EHS with epistemic operators is PSPACE-complete. Finally, in \cite{24} it was shown that the $ABL$ fragment of EHS has a decidable model checking problem.

The labelling function in the structures considered in \cite{22} is defined on the endpoints of the intervals. This corresponds to the intuitive representation of intervals as pairs and is often adopted in the literature. However, other choices are possible. For example, \cite{28} considers the labelling for an interval as the intersection of the labelling of all its elements. We argue that even more expressive setups are required.

Assume, for example, that we need to label a whole process of printing by means of the propositional variable printing. By adopting \cite{28}, by labelling the process with printing, it would follow that every subinterval would need to be labelled with printing too. This may not correspond to our intuition.

Similarly, if we were to adopt a labelling based on endpoints, and $S$ ($E$) is the state where printing starts (ends, respectively), it would follow that all the intervals starting in $S$ and ending in $E$ have to be labelled with printing. But if more than one process is present, it follows that the interval starting at the beginning of the first process and ending at the end of the second one is also labelled with printing, which, again, may be against our intuition.

This is just a simple example (we explore more significant ones in Section \ref{sec:4}); but it suggests that more liberal labellings imposing no such constraints are called for in this context. From a theoretical standpoint, it is of interest to
generalise previous labelling approaches and assess the impact these have on the decidability of the model checking problem. We are not aware of any previous attempt in this direction in the context of any HS logic.

Contribution. We put forward a generalisation of the labelling functions independently proposed in [22] and [28]. The novel labelling is defined by using regular expressions based on the states of the whole interval. For example, the process of printing from the example above can now be modelled by using the regular expression \( S \mathrel{\text{∧}} E^*E \). The models that result from this labelling are here called interpreted systems with regular labelling, ISRL for short. We study the logic \( \text{EHS}^+ \), sharing the syntax of EHS, but interpreted over ISRL, and show that it enjoys all the positive results known for EHS.

In order to be able to express properties of standard point-based models, and formally characterise the expressive power of \( \text{EHS}^+ \), we also define and study the logic \( \text{EHS}^\text{RE} \). Intuitively, \( \text{EHS}^\text{RE} \) can be seen as the result of moving the regular expressions from the labelling function to the atomic propositions. We show polynomial time reductions between the model checking problems for \( \text{EHS}^\text{RE} \) and \( \text{EHS}^+ \) and characterise the expressive power of the former.

Related work. Initial results for the model checking of HS and some of its variants have appeared recently [22, 24, 28]. The results of this paper generalise those presented in [22, 24]. Our setting is more expressive than [22] and further benefits from the fact that many properties become easier to express.

Note that ITL does allow for regular expressions to be used. Unlike \( \text{EHS}^\text{RE} \), where regular expressions can be used only for propositions, in ITL they can be used for any subformula. However, ITL expresses properties of a single interval, while \( \text{EHS}^\text{RE} \) can express properties of different branches. Furthermore, HS enjoys several fragments, such as the BDE one, with a computationally attractive model checking problem. This may be of particular use in applications.

Two further formalisms that are related to \( \text{EHS}^\text{RE} \) are PDL [18] and its linear counterpart LDL [12]. An epistemic version of PDL, E-PDL, was proposed in [5]. However, epistemic modalities in E-PDL are interpreted on points, not intervals as in EHS and \( \text{EHS}^\text{RE} \). This is largely the reason why \( \text{EHS}^\text{RE} \) is more expressive than E-PDL and the model checking problem for E-PDL is decidable in polynomial time [20], whereas the model checking problem for EIT is already PSPACE-hard. Notice also that E-PDL does not have backward modalities and can express properties of actions, unlike \( \text{EHS}^\text{RE} \).

Results on the correspondence between regular expressions and HS were presented in [27], where it was shown that each \( \omega \)-regular language can be encoded in the \( \text{ABB} \) fragment of HS. The encoding, however, uses additional proposi-
tional variables to label interval, and therefore cannot be used for the model checking problem.

2 Interpreted systems with regular labelling

We begin by recalling the notions of regular expressions. Given a set $X$, the set of regular expressions over $X$, denoted by $RE_X$, is defined by the following BNF:

\[ e ::= \emptyset | s | e | e + e | e^* \]

where $s \in X$. We allow parentheses for grouping and often omit the concatenation symbol "\$."

For each regular expression $e$, let $\text{Lang}(e)$ stand for the language denoted by $e$. Formally, $\text{Lang}(\emptyset) = \emptyset$, $\text{Lang}(s) = \{s\}$, $\text{Lang}(e_1; e_2) = \{w_1w_2 | w_1 \in \text{Lang}(e_1) \wedge w_2 \in \text{Lang}(e_2)\}$, $\text{Lang}(e_1 + e_2) = \text{Lang}(e_1) \cup \text{Lang}(e_2)$, and $\text{Lang}(e^*)$ is the smallest set containing $\emptyset$ such that for all $w_1 \in L(e)$ and $w_2 \in \text{Lang}(e^*)$, $w_1w_2 \in \text{Lang}(e^*)$.

Now we generalise interval-based interpreted systems [22] to systems with labelling based on regular expressions.

**Definition 1.** Given a set of agents $A = \{0, 1, \ldots, m\}$, an interpreted system with labelling on regular expressions, ISRL for short, is a tuple $\text{IS} = (\{L_i, l_0^i, \text{ACT}_i, P_i, t_i\}_{i \in A}, \lambda)$, where:

- $L_i$ is a finite set of local states for agent $i$,
- $l_0^i \in L_i$ is the initial state for agent $i$,
- $\text{ACT}_i$ is a finite set of local actions available to agent $i$,
- $P_i : L_i \to 2^{\text{ACT}_i}$ is a local protocol function for agent $i$, returning the set of possible local actions in a given local state,
- $t_i : L_i \times \text{ACT} \times L_i$, where $\text{ACT} = \text{ACT}_0 \times \cdots \times \text{ACT}_m$, is a local transition relation returning the next local state when a joint action is performed by all agents on a given local state,
- $\lambda : \text{Var} \to \text{RE}_G$ is a labelling function, where $G = L_0 \times L_1 \times \cdots \times L_m$ is the set of global configurations and $\text{Var}$ is a finite set of propositional variables.

Agent 0 is often called the environment.

We now define models of an IS on sets of paths from its initial configuration. Let $t^G \subseteq G^2$ be a relation such that $t^G((l_0, \ldots, l_m), (l_0', \ldots, l_m'))$ iff there exists a joint action $(a_0, \ldots, a_m) \in \text{ACT}$ such that for all $i$ we have $a_i \in P_i(l_i)$ and $t_i(l_i, (a_0, \ldots, a_m), l_i')$.

**Definition 2.** Given an ISRL $\text{IS} = (\{L_i, l_0^i, \text{ACT}_i, P_i, t_i\}_{i \in A}, \lambda)$ over a set of agents $A = \{0, \ldots, m\}$, the model of the IS is a tuple $M = (S, s_0, t, \{\sim_i\}_{i \in A}, \lambda)$, where

- $S \subseteq G^+$ is the set of global states, i.e., non-empty sequences $g_0 \ldots g_k$ such that $g_0 = (l_0^0, \ldots, l_0^m)$ and for each $i < k$ we have $t^G(g_i, g_{i+1})$. 

4
- $s_0 = g_0 = (l_0, \ldots, l_m)$ is the initial state of the system,
- $t \subseteq S^2$ is the global transition relation such that $t(g_0 \ldots g_k, g'_0 \ldots g'_l)$ iff $l = k + 1$ and for all $i \leq k$ we have $g_i = g'_i$,
- $\sim_i \subseteq S^2$ is the equivalence relation such that $g_0 \ldots g_k \sim_i g'_0 \ldots g'_l$ iff $g_k = (l_0, \ldots, l_m)$, $g'_k = (l'_0, \ldots, l'_m)$ and $l_i = l'_i$, and
- $\lambda$ is the labelling function.

Intuitively, $S$ denotes the set of global configurations of the ISRL equipped with information about all their predecessors. This is the standard construction used for defining unravelling in temporal logic (see, e.g., Definition 4.51 in [6]). We need to keep the information regarding the predecessors for the semantics of backward modalities; the semantics of the epistemic modalities is defined only on the current state.

Given a model $M$, an interval in $M$ is a finite path on $M$, i.e., a sequence of states $I = s_1, s_2, \ldots, s_n$ such that $t(s_i, s_{i+1})$, for $1 \leq i \leq (n - 1)$. A point interval is an interval that consists of exactly one state. We assume $I = \lambda = \top$ for a point interval $I$ and $p(I) = \lambda$ for all the other intervals.

For each state of $s = g_0, \ldots, g_k \in S$, we assume $g(s) = g_k$. So $g(s)$ denotes the actual states of $s$, not its history. We extend $g$ to intervals by assuming $g(I) = g(s_0) \ldots g(s_k)$ for every interval $I = s_0, \ldots, s_k$.

We say that an ISRL is point-based if $\lambda$ only labels the point intervals, i.e., for each $v \in \text{Var}$ we have $\lambda(v) = \sum_{s \in S} s$ for some $S' \subseteq S$. An ISRL is endpoint-based if $\lambda$ is defined on the endpoints of the intervals, i.e., for each $v \in \text{Var}$ we have $\lambda(v) = \sum_{s \in S'} (s + s S') + \sum_{(s, s') \in P} s S' s'$ for some $S' \subseteq S$, $P \subseteq S^2 \setminus \{(s, s) \mid s \in S\}$. Notice that the models of the point-based ISRL can be seen as standard Kripke structures; the models of the endpoint-based ISRL can be seen as the generalised Kripke structures of [22].

For $g = (l_0, l_1, \ldots, l_m)$ we denote by $l_i(g)$ the local state $l_i \in L_i$ of agent $i \in A$ in $g$. For a global state $s = g_0, \ldots, g_k$, we assume $l_i(s) = l_i(g_k)$.

Now we give an example of an interpreted system and of its model. We will use this example in the following sections to illustrate other constructions.

Example 1. Consider an ISRL $IS_{ex} = ((L_i, l_i^0, ACT_i, P_i, l_i)_{i \in A}, \lambda)$ over a set of agents $A = \{0, 1\}$ and a set of propositional variables $\text{Var} = \{p\}$, where
- $L_0 = \{l_0\}$, $L_1 = \{l_1, l_2, l_3\}$,
- $l_0^0 = l_0$, $l_1^0 = l_1$,
- $ACT_0 = \{a_1, a_2\}$, $ACT_1 = \{e\}$,
- $P_0(l_0) = ACT_0$, $P_1(l_1) = P_1(l_2) = P_1(l_3) = ACT_1$,
- $t_0 = \{(l_0, (a_1, e), l_0), (l_0, (a_2, e), l_0)\}$, $t_1 = \{(l_1, (a_1, e), l_2), (l_1, (a_2, e), l_2), (l_2, (a_2, e), l_3), (l_2, (a_1, e), l_1), (l_3, (a_1, e), l_1), (l_3, (a_2, e), l_1)\}$,
- $\lambda(p) = g_1(g_2 + g_3)^* g_3$, where $g_i = (l_0, l_i)$.

Figure 1 depicts the agents of $IS$. We have $G = \{g_1, g_2, g_3\}$ and $t^G = \{((l_0, l_1), (l_0, l_2)), ((l_0, l_2), (l_0, l_3)), ((l_0, l_2), (l_0, l_1)), ((l_0, l_3), (l_0, l_1))\}$. The model $M_{ex}$ of $IS_{ex}$ is infinite. Its fragment is depicted in Figure 2.
The agents from Example 1, where * stands for any action.

Fig. 2: A fragment of the model of IS_{ex} from Example 1. I_1, I_2 and I_3 are labelled by p, as g(I_1) = g(I_2) = g_1g_2g_3 and g(I_3) = g_1g_2g_1g_2g_3 belong to \text{Lang}(\lambda(p)).

IR_A I' iff first(I') = last(I)
IR_B I' iff I = I'I_1 for some interval I_1
IR_D I' iff I = I_1'I_2 for some intervals I_1, I_2
IR_E I' iff I = I_1'I_1 for some interval I_1
IR_L I' iff there is a path from last(I) to first(I')
IR_O I' iff II_1 = I_2I_2' for some intervals I_1, I_2

Fig. 3: Basic Allen relations.

3 The logic EHS^+

We now define the syntax of the specification language we focus on in this paper. The temporal operators represent relations between intervals as originally defined by Allen [3]. Six of these relations are presented in Figure 3: R_A ("after" or "meets"), R_B ("begins" or "starts"), R_D ("during"), R_E ("ends"), R_L ("later"), and R_O ("overlaps"). Six additional operators can be defined corresponding to the six inverse relations. Formally, for each X \in \{A,B,D,E,L,O\}, we also consider the relation R_X^{-1}.

For convenience, we also consider the "next" relation R_N such that IR_N I' iff t(last(I), first(I')) [24]. Let HS = \{A,\hat{A},B,\hat{B},D,\hat{D},E,\hat{E},L,\hat{L},N,\tilde{N},O,\tilde{O}\}.

**Definition 3.** The syntax of the Epistemic Halpern–Shoham Logic (EHS^+), \mathcal{L}_{EHS^+} is defined by the following BNF.

\[
\varphi ::= p i | p | \neg \varphi | \varphi \land \varphi | K_i \varphi | C_i \varphi | \langle X \rangle \varphi
\]

where p \in \text{Var} is a propositional variable, i \in A is an agent, \Gamma \subseteq A is a set of agents, and X \in HS.
We use abbreviations including \([X]\varphi\) for \(\neg(X)\neg\varphi\) and the usual Boolean connectives \(\lor, \Rightarrow, \Leftrightarrow\) as well as the constants \(\top, \bot\) in the standard way.

Note that the modality \(\langle N \rangle\) is a counterpart of the \(EX\) operator of CTL. While \(\langle N \rangle\) is redundant in \(EHS^+\) since \(\langle N \rangle\varphi = (A)(\neg pi \land (B)\bot \land (A)\varphi)\), it is useful in fragments of \(EHS^+\) that do not contain \(B\) and \(E\).

In order to provide the semantics for the epistemic operators on an interval based semantics, we specify when two intervals are epistemically indistinguishable for an agent, i.e., an agent cannot distinguish between the two. We say that \(I \sim_i I'\), where \(I = s_1, \ldots, s_k, I' = s'_1, \ldots, s'_l\), iff \(k = l\) and for all \(j \leq k\) we have \(s_j \sim_i s'_j\). In other words, for two intervals to be indistinguishable to agent \(i\) the two intervals need to be of the same length and the agent cannot be able to distinguish any corresponding point in the interval. This appears the natural generalisation to intervals of the point-based knowledge modalities traditionally used in epistemic logic \([10]\). For example, in the model presented in Example \([11]\) we have \(I \sim_0 I'\) if and only if \(|I| = |I'|\) and \(I \sim_1 I'\) if and only if \(I = I'\); in general these relations may be more complicated. We extend this definition to the common knowledge case by considering \(\sim_{I^*} = (\bigcup_{\sim_i} \sim_i)^*\), for any group of agents \(\Gamma \subseteq A\), where \(*\) denotes the transitive closure. For further explanations we refer to \([22]\).

We now define when a formula is satisfied in an interval on an ISRL.

**Definition 4 (Satisfaction).** Given an \(EHS^+\) formula \(\varphi\), an ISRL \(IS\), its model \(M = (S, s_0, t, \{\sim_i\}_{i \in A}, \lambda)\) and an interval \(I\), we inductively define whether \(\varphi\) holds in the interval \(I\), denoted \(M, I \models \varphi\), as follows:

1. \(M, I \models p_i\) iff \(I\) is a point interval,
2. \(M, I \models p\) iff \(g(I) \in \text{Lang}(\lambda(p))\),
3. \(M, I \models \neg \varphi\) iff it is not the case that \(M, I \models \varphi\),
4. \(M, I \models \varphi_1 \land \varphi_2\) iff \(M, I \models \varphi_1\) and \(M, I \models \varphi_2\),
5. \(M, I \models K_i \varphi\), where \(i \in A\), iff for all \(I' \sim_i I\) we have \(M, I' \models \varphi\),
6. \(M, I \models C_\Gamma \varphi\), where \(\Gamma \subseteq A\), iff for all \(I' \sim_\Gamma I\) we have \(M, I' \models \varphi\),
7. \(M, I \models \langle X \rangle \varphi\) iff there exists an interval \(I'\) such that \(IR_X I'\) and \(M, I' \models \varphi\), where \(R_X\) is an Allen relation as above.

We write \(IS, I \models \varphi\) if \(M, I \models \varphi\), where \(M\) is the model of \(IS\), and \(IS \models \varphi\) if \(IS, s_0 \models \varphi\).

### 4 Expressive power

The expressivity of \(EHS^+\) is incomparable to that of traditional formalisms such as LTL, CTL, or EHS as \(EHS^+\) is defined on different semantics structures. To investigate its expressive power, we introduce \(EHS^{RE}\), a variant of \(EHS^+\) defined over point-based interpreted systems. We show that the model checking problems for \(EHS^{RE}\) and \(EHS^+\) admit a polynomial time reduction to one another on the corresponding semantics. We also observe that \(EHS^{RE}\) can represent properties not expressible by CTLK*, the epistemic version of CTL* (and therefore LTLK
EHS considered over point-based ISRL either. So over point-based ISRL we have that evaluating the EHS

\[ p \not\implies \phi \]

cannot be expressed in CTLK

\[ \text{IS} \mid \{ \phi \} \]

and thereby to that of some more widely known formalisms.

\[ \text{Lan} \]

is indeed more expressive than the usual temporal-epistemic logic interpreted on points.

For a labelling function \( \lambda \) and a regular expression \( r \), let \( \lambda \circ r \) be the regular expression obtained from \( r \) by replacing each propositional variable \( p \) by \( \sum_{g \in \lambda(p)} g \) (if \( \lambda(p) = \emptyset \), we put \( \emptyset \)).

**Definition 5.** The language of EHS\(^R\), \( \mathcal{L}_{EHS}^{RE} \), is defined as follows:

\[ \phi ::= pi \mid r \mid \neg \phi \mid \phi \land \phi \mid Ki \phi \mid C_i \phi \mid \langle X \rangle \phi \]

where \( r \in RE_{\text{Var}}, i \in A, \Gamma \subseteq A, \text{and } X \in \text{HS} \).

The semantics of EHS\(^R\) results from replacing the second rule in Definition 4 by (ii') \( M, I \models r \iff I = s_1, \ldots, s_k \text{ and } g(s_1) \ldots g(s_k) \in \text{Lang}(\lambda \circ r) \).

Intuitively, EHS\(^R\) is the result of adapting EHS\(^+\) by moving the regular expressions from the labelling function into the language.

For convenience, we will use \( p \) and \( \neg p \) in the regular expressions, by defining \( p = \sum_{X \subseteq \text{Var}, p \in X} X \) and \( \neg p = \sum_{X \subseteq \text{Var}, p \notin X} X \).

Let \( L_{\text{Var}} \) be the set of all possible labellings of interpreted systems with variables of \( \text{Var} \), and \( L_{\text{Var}}^{\text{point-based}} \subseteq L_{\text{Var}} \) be the set of all such labellings for point-based interpreted systems.

**Theorem 1.** There exist polynomial time computable functions \( f : L_{\text{Var}} \times \mathcal{L}_{EHS}^{RE} \to L_{\text{Var}}^{\text{point-based}} \times \mathcal{L}_{EHS}^{RE} \) and \( f' : L_{\text{Var}} \times \mathcal{L}_{EHS}^{RE} \to L_{\text{Var}} \times \mathcal{L}_{EHS}^{+} \) such that for any interpreted system \( IS = (\{L_i, t_i^0, \text{ACT}_i, P_i, t_i\}_{i \in A}, L) \), any formula \( \phi \) and any interval \( I \):

1. If \( IS, I \models \phi \) and \( f(L, \phi) = (L', \phi') \), then \( IS' = (\{L_i, t_i^0, \text{ACT}_i, P_i, t_i\}_{i \in A}, L') \) is point-based and such that \( IS', I \models \phi' \).

2. If \( IS \) is point-based, \( IS, I \models \phi \), and \( f'(L, \phi) = (L', \phi') \), then we have that \( (\{L_i, t_i^0, \text{ACT}_i, P_i, t_i\}_{i \in A}, L'), I \models \phi' \).

Given Theorem 1 we can say that the logics EHS\(^+\) and EHS\(^R\) can describe the same properties of corresponding interpreted systems. Since EHS\(^R\) expresses properties of point-based interpreted systems, whose models are standard Kripke structures, we can formally compare the expressive power of EHS\(^R\) to that of some more widely known formalisms.

**Definition 6.** Given two logics \( \mathcal{L}_1, \mathcal{L}_2 \), we write \( \mathcal{L}_1 \subseteq \mathcal{L}_2 \) if for each formula \( \phi_1 \) of \( \mathcal{L}_1 \) there is a formula \( \phi_2 \) of \( \mathcal{L}_2 \) such that for all point-based ISRL we have \( IS \models \phi_1 \iff IS \models \phi_2 \).

One can easily show that EHS\(^R\) \( \not\subseteq \) CTLK\(^*\). Consider the temporal property “all the paths starting in the initial state satisfy \( (p; \text{True})^\omega \)”. This property cannot be expressed in CTLK\(^*\) [22]. However, the property can be verified by evaluating the EHS\(^R\) formula \( p \land [A](\langle p; \top \rangle^* \implies [N](p; \top^*)) \).

Also observe that the property above cannot be expressed in the logic EHS considered over point-based ISRL either. So over point-based ISRL we have that EHS\(^R\) \( \not\subseteq \) EHS.
In terms of limitations, note that EHS\textsuperscript{RE} can only express properties of finite intervals. For example, the CTL property $\text{AF} p$ expressing the fact that each infinite path satisfies $p$ at some point cannot be encoded by any EHS\textsuperscript{RE} formula. Therefore $\text{CTLK} \not\subseteq \text{EHS}\textsuperscript{RE}$; similarly we have $\text{LTLK} \not\subseteq \text{EHS}\textsuperscript{RE}$.

Since EHS\textsuperscript{RE} does not allow us to name actions explicitly, we have that $\text{E-PDL} \not\subseteq \text{EHS}\textsuperscript{RE}$. It can also be shown that $\text{EHS}\textsuperscript{RE} \not\subseteq \text{E-PDL}$, since E-PDL cannot express the property $(\langle A \rangle (K_1 (pq^* r)))$ as the epistemic modalities in E-PDF is based on states rather than time-intervals.

5 The model checking problem

We now investigate the complexity of the model checking problem for fragments of the logics explored so far.

Definition 7. Given a formula $\varphi$ of a logic $L$, an ISRL $IS$ and an interval $I$, the model checking problem for $L$ amounts to checking whether or not $IS, I \models \varphi$.

In establishing the above, we say we have model checked $M$ against the specification $\varphi$ at an interval $I$. Notice that the formula is verified only in the given interval; however, one can easily check whether all the initial intervals satisfy a formula $\varphi$ by checking whether $M, s_0 \models [A] \varphi$.

The $ABL\Lambda$ fragment of EHS\textsuperscript{+}, denoted as $\text{EHS}^+_{ABL\Lambda}$, is the subset of $\text{EHS}^+$ where the BNF is restricted to the only modalities $K_i$, $C_\Gamma$, $\langle A \rangle$, $\langle \bar{B} \rangle$, $\langle L \rangle$ and $\langle N \rangle$. Similarly, the $BDE$ fragment of EHS\textsuperscript{+}, denoted as $\text{EHS}^+_{BDE}$, is the restriction of $\text{EHS}^+$ to the modalities $K_i$, $C_\Gamma$, $\langle B \rangle$, $\langle D \rangle$ and $\langle E \rangle$.

Theorem 2. Model checking ISRL against $\text{EHS}^+_{BDE}$ specifications is decidable and PSpace-complete.

The above follows from the fact that the satisfaction can be determined by examining only intervals of bounded length. The proof is in the appendix.

Theorem 3. Model checking ISRL against $\text{EHS}^+_{ABL\Lambda}$ specifications is decidable in non-elementary time.

We prove this by generalising the proof of Theorem 13 given in [24].

A top-level sub-formula of a formula $\varphi$ is a sub-formula of $\varphi$ of the form $X \varphi'$, for some modality $X$ of $\text{EHS}^+_{ABL\Lambda}$, that is not in scope of any modality. Assume an ISRL $IS$. Let $f^{IS}(\varphi)$ be defined recursively as

$$f^{IS}(\varphi) = (2|G| \prod_{q \in \text{Var}} 2^{|\lambda(q)|} \cdot 2^{f^{IS}(\varphi_1)} \cdot \ldots \cdot 2^{f^{IS}(\varphi_k)})$$

where $X_1 \varphi_1, \ldots, X_k \varphi_k$ are the top-level sub-formulas of $\varphi$. The idea is that $f^{IS}(\varphi)$ is an upper bound on the number of different interval types w.r.t. $\varphi$; an interval type consists of an information whether an interval is a point interval or not (hence 2), what are its endpoints (hence $|G|$), what are the states of the automata corresponding to the regular expressions after reading the interval.
(hence the product) and types of intervals related to the interval w.r.t. the top level sub-formulas of \( \varphi \) (hence the recursive part).

We define a bounded satisfaction relation \( \models_B \) for \( \text{EHS}^+_{A\bar{B}LN} \), for which the decidability of the model checking is straightforward. The rules (i’-vi’) of the definition of \( \models_B \) are the same as the rules (i-vi) from Definition 4 except that \( \models \) is replaced with \( \models_B \). The last rule, however, is different:

\[
\text{vii') } M, I \models_B \langle X \rangle \varphi \text{ if and only if there exists an interval } I' \text{ such that } |I'| \leq |I| + f^{IS}(\varphi), \text{ } IR_X I' \text{ and } M, I' \models_B \varphi, \text{ where } X \text{ is } A, \bar{B}, \text{ or } N.
\]

It is not hard to see that model checking is decidable for the bounded semantics. It turns out that in the \( \text{EHS}^+_{A\bar{B}LN} \) case, the relations \( \models \) and \( \models_B \) are the same, and therefore the model checking procedure for the bounded semantics solves the model checking problem for the unbounded semantics. All the details are in the appendix.

By employing the polynomial time reductions of Theorem 1, we can show that model checking point-based ISRL against \( BDE \) fragment of \( \text{EHS}^{RE} \) specifications is PSPACE-complete and that model checking point-based ISRL against \( A\bar{B}LN \) fragment of \( \text{EHS}^{RE} \) specifications is decidable.

6 Conclusions and Future Work

Temporal logic is one of the key foundational tools to reason about computing systems. Several variants of temporal logics have been studied, reflecting the underlying assumptions on the temporal flow, ranging from linear to branching and from discrete to continuous. Interval temporal logics \([17,31]\) are a relatively less explored variant of temporal logic. As is known, these are particularly appropriate to study the properties of continuous processes. However, while interval temporal logics could provide a formal basis for systems verification, little is known in terms of their model checking problem. Indeed, this was only recently explored in \([22,24,28]\) in the context of variants of the logic HS.

Since the complexity of the model checking problem for HS fragments is typically high and the decidability of the full HS logic is not known, a compelling avenue of research involves establishing whether the expressivity of previously studied, well-behaved fragments of HS can be significantly enriched without losing the attractiveness of their model checking problem. The logic \( \text{EHS}^+ \), proposed in this paper, combines the interval temporal logic HS and epistemic logic. The logic can be see as a considerable generalisation of the logics proposed in \([22,24,28]\). Specifically, \( \text{EHS}^+ \) can express properties of complex processes consisting of many stages, even if the processes are repeating or overlapping. Regular expressions allow to express further properties not explored here.

We showed that the model checking for the \( BDE \) fragment of \( \text{EHS}^+ \) is decidable and PSPACE-complete, and that the model checking problem for the \( ABL \) fragment of the logic is decidable. While the complexity is the same as that for the EHS logic in \([24]\), \( \text{EHS}^+ \) is considerably more expressive.

Further ahead we intend to study more expressive fragments of \( \text{EHS}^+ \). We believe that the technique presented here can be extended to backward modalities, such as \( \langle A \rangle, \langle D \rangle, \langle E \rangle, \langle L \rangle \) and \( \langle N \rangle \). However, more investigations are
required, since in the case of backward modalities one cannot simply disregard the histories.

A further open problem is the decidability of any fragment involving the modality $O$. In a sense, $O$ is the hardest case of all operators. Indeed, is known that the satisfiability for the $O$ fragment of HS is undecidable [9]. Since $O$ can be expressed using $\bar{B}$ and $E$ [13], we cannot show the decidability of the join of the fragments studied in this paper ($\bar{A}BDELN$) without proving it for $O$.

Finally, we are interesting in implementing an efficient model checking toolkit for $\text{EHS}^{\text{RE}}$ specifications. We intend to develop more efficient algorithms on symbolic representations and a suitable predicate abstraction technique for $\text{EHS}^{\text{RE}}$.

**Acknowledgments.** The authors would like to thank Angelo Montanari whose comments on [21] lead to the present investigation. The second author was generously supported by Polish National Science Center based on the decision number DEC-2011/03/N/ST6/00415. This research was supported by the EPSRC under grant EP/I00520X.

**References**

1. MCK: Model checking knowledge. http://www.cse.unsw.edu.au/~mck
2. VerICS. http://www.ipipan.waw.pl/staff/w.Penczek/abmpw/index-ang.htm
3. Allen, J.F.: Maintaining knowledge about temporal intervals. Communications of the ACM 26(11), 832–843 (1983)
4. Artale, A., Kontchakov, R., Ryzhikov, V., Zakharyaschev, M.: Tractable interval temporal propositional and description logics. In: Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI15). pp. 1417–1423 (2015)
5. van Benthem, J., van Eijck, J., Kooi, B.: Logics of communication and change. Information and Computation 204(11), 1620–1662 (2006)
6. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic, Cambridge Tracts in Theoretical Computer Science, vol. 53. Cambridge University Press (2001)
7. Bresolin, D., Della Monica, D., Goranko, V., Montanari, A., Sciavicco, G.: Metric propositional neighborhood logics: Expressiveness, decidability, and undecidability. In: Proceedings of the 19th European Conference on Artificial Intelligence (ECAI10). pp. 695–700 (2010)
8. Bresolin, D., Della Monica, D., Montanari, A., Sala, P., Sciavicco, G.: Interval temporal logics over finite linear orders: the complete picture. In: Proc. of the 20th European Conference on Artificial Intelligence (ECAI12). pp. 199–204 (2012)
9. Bresolin, D., Monica, D., Goranko, V., Montanari, A., Sciavicco, G.: The dark side of interval temporal logic: Sharpening the undecidability border. In: Proceedings of the 18th International Symposium on Temporal Representation and Reasoning (TIME11). pp. 131–138 (2011)
10. Chandra, A., Kozen, D., Stockmeyer, L.: Alternation. Journal of ACM 28(1), 114–133 (1981)
11. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. The MIT Press, Cambridge, Massachusetts (1999)
12. De Giacomo, G., Vardi, M.Y.: Linear temporal logic and linear dynamic logic on finite traces. In: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence. pp. 854–860. IJCAI13, AAAI Press (2013)
13. Della Monica, D.: Expressiveness, decidability, and undecidability of interval temporal logic. Ph.D. thesis, University of Udine (2011)
14. Della Monica, D., Goranko, V., Montanari, A., Sciavicco, G.: Expressiveness of the interval logics of Allen’s relations on the class of all linear orders: complete classification. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI11). p. 845. AAAI Press (2011)
15. Della Monica, D., Goranko, V., Montanari, A., Sciavicco, G.: Interval temporal logics: a journey. Bulletin of EATCS 3(105) (2013)
16. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning about Knowledge. MIT Press, Cambridge (1995)
17. Halpern, J., Shoham, Y.: A propositional modal logic of time intervals. Journal of The ACM 38, 935–962 (1991)
18. Harel, D., Tiuryn, J., Kozen, D.: Dynamic Logic. MIT Press, Cambridge, MA, USA (2000)
19. Hopcroft, J., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation. Addison-Wesley Publishing Company (1979)
20. Lange, M.: Model checking propositional dynamic logic with all extras. Journal of Applied Logic 4(1), 39–49 (2006)
21. Lodaya, K.: A language-theoretic view of verification. In: Modern applications of automata theory, pp. 149–169. World Scientific, IISc research monographs (2012)
22. Lomuscio, A., Michaliszyn, J.: An epistemic Halpern-Shoham logic. In: Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI13). pp. 1010–1016. AAAI Press (2013)
23. Lomuscio, A., Qu, H., Raimondi, F.: MCMAS: A model checker for the verification of multi-agent systems. In: Proceedings of the 21th International Conference on Computer Aided Verification (CAV09). Lecture Notes in Computer Science, vol. 5643, pp. 682–688. Springer (2009)
24. Lomuscio, A., Michaliszyn, J.: Decidability of model checking multi-agent systems against a class of ehs specifications. In: Proceedings of the 21st European Conference on Artificial Intelligence (ECAI14). pp. 543–548 (2014)
25. Marcinkowski, J., Michaliszyn, J.: The ultimate undecidability result for the Halpern-Shoham logic. In: Proceedings of the 26th Annual IEEE Symposium on Logic in Computer Science (LICS11). pp. 377–386. IEEE Computer Society (2011)
26. Montanari, A., Puppis, G., Sala, P.: Maximal decidable fragments of Halpern and Shoham’s modal logic of intervals. In: Proceedings of 37th International Colloquium on Automata, Languages and Programming (ICALP10). Lecture Notes in Computer Science, vol. 6199, pp. 345–356 (2010)
27. Montanari, A., Sala, P.: Interval logics and $\omega B$-regular languages. In: Language and Automata Theory and Applications, Lecture Notes in Computer Science, vol. 7810, pp. 431–443. Springer (2013)
28. Montanari, A., Murano, A., Perelli, G., Peron, A.: Checking interval properties of computations. In: 21st International Symposium on Temporal Representation and Reasoning (TIME14). pp. 59–68. IEEE (2014)
29. Montanari, A., Pratt-Hartmann, I., Sala, P.: Decidability of the logics of the reflexive sub-interval and super-interval relations over finite linear orders. 17th Int. Symposium on Temporal Representation and Reasoning pp. 27–34 (2010)
30. Montanari, A., Puppis, G., Sala, P.: A decidable spatial logic with cone-shaped cardinal directions. In: Proceedings of the 23rd Conference on Computer Science and Logic (CSL09). pp. 394–408 (2009)
31. Moszkowski, B.C.: Reasoning about digital circuits. Ph.D. thesis, Stanford University, Stanford, CA, USA (1983)
32. Wolper, P.: Temporal logic can be more expressive. Information and control 56(1), 72–99 (1983)
A Sketch of the Proof of Theorem 1

Roughly speaking, functions $f$ and $f'$ just move the regular expressions from the labelling to the formula and the other way round. Function $f$ is such that $f(\lambda, \varphi) = (\lambda', \varphi')$, where $\lambda'(g) = g$ for all the states $s$ and $\varphi'$ is the result of replacing each propositional variable $q$ in $\varphi$ by $\sum_{g \lambda(q)} g$. Function $f'$ is such that $f'(\lambda', \varphi') = (\lambda, \varphi)$, where for each regular expression $r$ in $\varphi'$, we replace $r$ by an unique propositional variable $q$ and we put $\lambda(q') = \lambda' \circ r$. It is readily verifiable that both functions are as required.

B Proof of Theorem 2

Proof. The lower bound follows from the lower bound for the endpoint-based variant of ISRL that was shown in [22] for the same syntax.

For the upper bound, we consider an alternating algorithm [10] working in polynomial time. Since $\text{APTime} = \text{PSpace}$, the theorem follows. Algorithm 1 reports the procedure ver-BDE that solves the model checking problem. Its complexity follows from the fact that each existentially or universally selected interval has the size bounded by the size of the initial interval. $\square$

Algorithm 1 The model checking procedure for $\text{EHS}^+_\text{BDE}$.

\begin{algorithm}
1: procedure ver-BDE($M$, $I$, $\varphi$)
2: if $\varphi = p$ then return $g(I) \in \text{Lang}(\lambda(p))$
3: if $\varphi = pi$ then return $pi(I)$
4: if $\varphi = \neg \varphi'$ then return $\neg \text{ver-BDE}(M, I, \varphi')$
5: if $\varphi = \varphi_1 \land \varphi_2$ then return $\text{ver-BDE}(M, I, \varphi_1) \land \text{ver-BDE}(M, I, \varphi_2)$
6: if $\varphi = K_i\varphi'$ where $i \in A$ then
7: universally select $J$ such that $J \sim_i I$
8: return $\text{ver-BDE}(M, J, \varphi')$
9: if $\varphi = C_G\varphi'$ where $G \subseteq A$ then
10: universally select $J$ such that $J \sim_G I$
11: return $\text{ver-BDE}(M, J, \varphi')$
12: if $\varphi = X\varphi'$ where $X \in \{\langle B \rangle, \langle D \rangle, \langle E \rangle\}$ then
13: existentially select $J$ such that $IR_X J$
14: return $\text{ver-BDE}(M, J, \varphi')$
\end{algorithm}

C Proof of Theorem 3

Observe that $\langle L \rangle$ can be defined in terms of $\langle A \rangle$: for any $\varphi$, $\langle L \rangle \varphi \equiv \langle A \rangle (\neg pi \land \langle A \rangle \varphi)$. Given this, in what follows we assume that the formulas do not contain $\langle L \rangle$. We now define some auxiliary notions.
For convenience, for each modality $X$ of $\text{EHS}^+_{\text{ABL}}$, we define a relation $R_X$ as follows: $R_{\langle A \rangle} = R_A$, $R_{\langle \bar{B} \rangle} = R_{\bar{B}}$, $R_{K_i} = \sim_i$ and $R_{C_G} = \sim_G$.

**Theorem 4.** Model checking ISRL under bounded semantics against $\text{EHS}^+_{\text{ABL}}$ specifications is decidable.

**Algorithm 2** The model checking procedure for $\text{EHS}^+_{\text{ABL}}$.

1: procedure verify($M$, $I$, $\varphi$)  
2: if $\varphi = p$ then return $I \in \text{Lang}(\lambda(p))$  
3: if $\varphi = p_i$ then return $pi(I)$  
4: if $\varphi = \neg \varphi'$ then return $\neg$verify($M$, $I$, $\varphi'$)  
5: if $\varphi = \varphi_1 \land \varphi_2$ then return verify($M$, $I$, $\varphi_1$) $\land$ verify($M$, $I$, $\varphi_2$)  
6: if $\varphi = K_i \varphi'$ where $i \in A$ then  
7: for all $J$ s.t. $I \sim_i J$ do  
8: if $\neg$verify($M$, $J$, $\varphi'$) then return false  
9: return true  
10: if $\varphi = C_G \varphi'$ where $G \subseteq A$ then  
11: for all $J$ s.t. $I \sim_G J$ do  
12: if $\neg$verify($M$, $J$, $\varphi'$) then return false  
13: return true  
14: if $\varphi = X \varphi'$ where $X \in \{\langle A \rangle, \langle \bar{B} \rangle\}$ then  
15: for all $J$ s.t. $IR_X J$ and $|J| \leq f(\varphi) + |I|$ do  
16: if verify($M$, $J$, $\varphi'$) then return true  
17: return false

**Proof.** The procedure verify() given in Algorithm 2 solves the model checking problem. Clearly, it always terminates and its computation time is non-elementary.

Our crucial theorem says that the bounded semantics is basically the same as the unbounded one.

**Theorem 5.** Given an $\text{EHS}^+_{\text{ABL}}$ formula $\varphi$, a model $M$, and an interval $I$, $M, I \models \varphi$ if and only if $M, I \models_B \varphi$.

**Proof.** Consider a model $M = (S, s_0, t, \{\sim_i\}_{i \in A}, \lambda)$. For each $p \in \text{Var}$ we denote by $A^p$ the minimal deterministic finite state automaton recognizing the language $\text{Lang}(\lambda(p))$. By $A^p_w(p)$, where $p \in \text{Var}$, we denote the state of $A^p$ after reading a word $w$; in the following, we treat $A^p_w$ as a function from $\text{Var}$ to automata states.
Definition 8 (Modal Context Tree). Given a model $M$, the modal context tree of an interval $I$ w.r.t. an $EHS^{+}_{A\bar{G}LN}$ formula $\varphi$, denoted by $MCT^\varphi_I$, is the minimal unordered tree with labelled nodes and edges defined recursively as follows.

- The root of the tree is labelled by the tuple $g(\text{first}(I)), g(\text{last}(I)), \pi(I), \mathcal{A}_I$.
- For each top-level sub-formula $X\psi$ of $\varphi$ and each interval $I'$ such that $IR_XI'$, the root of $MCT^\varphi_I$ has an $X\psi$-successor $MCT^\psi_{I'}$ (X indicates the labelling of an edge).

In other words $MCT^\varphi_I$ contains sufficient information about all the intervals that need to be considered to determine the value of $\varphi$ in $I$ as well as the states of the automata after reading $I$.

Example 2. Consider the ISRL $IS_{ex}$ from Example 1, the formula $\varphi = K_0\pi \land \neg \langle A \rangle p$, and an interval $I = g_1$.

To build the modal context tree, we use the automaton for $\lambda(p)$ presented in Figure 4.

![Fig. 4: A minimal automaton for $g_1(g_1 + g_2)^*g_3$. $z_3$ is the only accepting state.](image)

![Fig. 5: $MCT^\varphi_I$ from Example 2. The omitted $\langle A \rangle p$ successors are labelled by: $g_1, g_2, \bot, \{(p, z_2)\}; g_1, g_1, \bot, \{(p, z_2)\}; g_1, g_1, \bot, \{(p, z_3)\}; g_1, g_2, \bot, \{(p, z_3)\}; g_1, g_2, \bot, \{(p, z_3)\}; g_1, g_3, \bot, \{(p, z_3)\}.](image)

The top level sub-formulas of $\varphi$ are $K_1\pi$ and $\langle A \rangle p$. $MCT^\varphi_I$ (Figure 5) represents $I$. Notice that there are infinitely many $R_A$ successors of $I$, but $MCT^\varphi_I$ needs only 7 $\langle A \rangle p$-successors. For example, the successor labelled by $g_1, g_2, \bot, \{(p, z_2)\}$ represents all the intervals $I$ such that $g(I)$ is of the form $g_1(g_1 + g_2)^*$.

We now show that the number of modal context trees for a given formula is bounded. We will use this later as a kind of pumping argument and show that if an interval is long enough, then some of its prefixes have the same modal context tree.
Lemma 1. Given a model \( M \) and a formula \( \varphi \), \(|\{MCT_I^\varphi \mid I \text{ is an interval in } M\}| < f^{IS}(\varphi)\).

Proof. We show the lemma by induction on \( \varphi \). Clearly, if a formula has no modalities, then \( \{MCT_I^\varphi \mid I \text{ is an interval in } M\} \) contains trees with only one node, that can be labelled with \( 2|G|^2 \prod_{q \in \text{Var}} 2^{\lambda(q)} \) different labels.

Consider a formula \( \varphi \) with the top-level sub-formulas \( X_1 \varphi_1, \ldots, X_k \varphi_k \). Each tree for \( \varphi \) consists of one of \( 2|G|^2 \prod_{q \in \text{Var}} 2^{\lambda(q)} \) possible roots and, for each \( i \), any subset of subtrees for \( \varphi_i \). Therefore, \(|\{MCT_I^\varphi \mid I \text{ is an interval in } M\}| < 2|G|^2 \prod_{q \in \text{Var}} 2^{\lambda(q)} \prod_{i} f^{IS}(\varphi_i) = f^{IS}(\varphi)\). \( \square \)

We show that the modal context tree does not depend on the histories.

Lemma 2. Consider a model \( M = (S, s_0, t, \{\sim_i\}_{i \in A}, \lambda) \) and a formula \( \varphi \). If \( I \) and \( I' \) are intervals such that \( g(I) = g(I') \), then \( MCT_I^\varphi = MCT_{I'}^\varphi \).

Proof. We show this by induction.

The roots of \( MCT_I^\varphi \) and \( MCT_{I'}^\varphi \) have the same labels, since \( g(\text{first}(I)) = g(\text{first}(I')) \), \( g(\text{last}(I)) = g(\text{last}(I')) \), \( pi(I) = pi(I') \) and the labelling is defined on \( g(I) \).

Consider a \( \langle X \rangle \varphi' \)-successor \( T \) of the root of \( MCT_I^\varphi \), where \( \langle X \rangle \varphi' \) is a top-level sub-formula of \( \varphi \) and \( X \in \{A, B, N\} \). There is an interval \( J \) such that \( IR_X J \) and \( MCT_J^{\varphi'} = T \). So there exists a \( J' \) such that \( I' IR_X J' \) and \( g(J) = g(J') \), because \( X \) is a “forward modality” so the \( R_X \) successors of \( I' \) do not depend on the history. By the inductive hypothesis, \( MCT_J^{\varphi'} = MCT_{J'}^{\varphi'} \), and therefore the roots of \( MCT_I^\varphi \) and \( MCT_{I'}^{\varphi'} \) have the same \( \langle X \rangle \varphi' \) successors.

As for the \( X \varphi' \) successors where \( X \) is an epistemic modality, it is enough to observe that \( IR_X I' \), and therefore \( I \) and \( I' \) are related to the same intervals by the equivalence relation \( R_X \). The lemma follows. \( \square \)

Now we argue that if two intervals have the same modal context tree w.r.t. \( \varphi \), then either both satisfy \( \varphi \) or none of them.

Lemma 3. Consider a model \( M = (S, s_0, t, \{\sim_i\}_{i \in A}, \lambda) \) and a formula \( \varphi \). If \( I \) and \( I' \) are intervals such that \( MCT_I^\varphi = MCT_{I'}^\varphi \), then \( M, I \models \varphi \) if and only if \( M, I' \models \varphi \).

Proof. We show it by induction on \( \varphi \).

Case 1. \( \varphi = p \) for some variable \( p \). The root of the \( MCT_I^\varphi \) is labelled by the state of an automaton corresponding to \( \lambda(p) \) after reading \( I \), and the root of the \( MCT_{I'}^\varphi \) is labelled by the state of an automaton corresponding to \( \lambda(p) \) after reading \( I' \). Since the two trees are equal, the automaton is in the same state in both cases, either accepting or rejecting, and therefore \( M, I \models \varphi \) if and only if \( M, I' \models \varphi \).

Case 2. \( \varphi = pi \). The root of the \( MCT_I^\varphi \) is labelled by \( pi(I) \), and so is the root of \( MCT_{I'}^\varphi \), and therefore \( pi(I) = pi(I') \).

Case 3. \( \varphi = \neg \varphi' \). By the inductive assumptions, \( M, I \models \varphi' \) if and only if \( M, I' \models \varphi' \), so \( M, I \models \varphi \) if and only if \( M, I' \models \varphi \).
Case 4. \( \varphi = \varphi_1 \land \varphi_2 \). By the induction assumption, \( M, I \models \varphi_1 \) if and only if \( M, I' \models \varphi_1 \) and \( M, I \models \varphi_2 \) if and only if \( M, I' \models \varphi_2 \), so \( M, I \models \varphi \) if and only if \( M, I' \models \varphi \).

Case 5. \( \varphi = K_i \varphi' \). Assume that \( M, I \models \varphi \). Consider any interval \( J' \) such that \( I' \sim_i J' \). By definition, in the tree \( MCT^\varphi_I \), the subtree \( MCT^\varphi_{J'} \) is a \( K_i \varphi' \)-successor of the root. It follows that in the tree \( MCT^\varphi_I (= MCT^\varphi_I) \), \( MCT^\varphi_{J'} \) is a \( K_i \varphi' \)-successor of the root. Let \( J \) be such that \( I \sim_i J \) and \( MCT^\varphi_J = MCT^\varphi_{J'} \). Clearly, since \( M, I \models \varphi, M, J \models \varphi' \), by the inductive assumptions, \( M, J' \models \varphi' \). Therefore \( M, I' \models \varphi \).

Case 6. \( \varphi = C_G \varphi' \). Assume that \( M, I \models \varphi \) and \( J' \) is such that \( I' \sim_G J' \). Again, in \( MCT^\varphi_I \) the subtree \( MCT^\varphi_{J'} \) is a \( C_G \varphi' \)-successor of the root. It follows that in the tree \( MCT^\varphi_I \), \( MCT^\varphi_{J'} \) is a \( C_G \varphi' \)-successor of the root. Let \( J \) be such that \( I \sim_G J \) and \( MCT^\varphi_J = MCT^\varphi_{J'} \), then \( M, J \models \varphi' \), and by the inductive assumptions, \( M, J' \models \varphi' \). Therefore \( M, I' \models \varphi \).

Case 7. \( \varphi = (A) \varphi' \). We have \( M, I \models (A) \varphi' \) if and only if there is an interval \( J \) starting in \( \text{last}(I) \) satisfying \( \varphi' \). Since \( g(\text{last}(I)) = g(\text{last}(I')) \), the intervals starting from \( \text{last}(I) \) and \( \text{last}(I') \) are the same (modulo histories), and therefore there exists an interval \( J' \) starting in \( \text{last}(I') \) such that \( g(J) = g(J') \). By Lemma it follows that \( MCT^\varphi_{J'} = MCT^\varphi_{J'} \).

Case 8. \( \varphi = (B) \varphi' \). Assume that there is an interval \( J \) such that \( IR_B J \) and \( M, J \models \varphi' \). Then, \( MCT^\varphi_{J'} \) is an \( (B) \varphi' \) successor of the root in \( MCT^\varphi_I \), and so in \( MCT^\varphi_I \). So there is an interval \( J' \) such that \( I'R_B J' \) and \( MCT^\varphi_{J'} = MCT^\varphi_{J'} \). By the inductive hypothesis, \( M, J' \models \varphi' \) and therefore \( M, I' \models \varphi \).

Case 9. \( \varphi = (N) \varphi' \). This can be shown similarly to Case 7 for \( (A) \varphi' \). □

As we remarked earlier, if an interval \( I \) is long enough, then \( I \) has two prefixes with the same modal context tree w.r.t. a formula \( \varphi \). Intuitively speaking, we would like to replace the longer prefix by the shorter one, thereby obtaining an interval \( I' \), and show that the modal context trees of \( I \) and \( I' \) are the same. By the above lemma, it would follow that they both satisfy the given formula. What remains to be proved is that if we have two prefixes with the same modal context tree, and we append the same interval to both, the results will also have the same modal context tree.

We use the following terminology. A partial state is a sequence of states \( g_1 \ldots g_k \) such that for all \( i < k \), we have \( t^G(g_i, g_{i+1}) \). Each state of the model is a partial state; but partial states are not required to start at \( g_0 \). A partial interval is a sequence \( s_1 \ldots s_k \) of partial states such that for each \( i < k \) we have that \( s_{i+1} = s_i g_i \) for some partial state \( g_i \). A partial interval \( I = s_1 \ldots s_k \) is clear if \( s_1 = g \) for some partial state \( g \). We extend the functions \( \text{first} \), \( \text{last} \), and \( g \) and the other notions to partial intervals in the obvious way.

We define the operation of adding context to partial intervals as follows. Given a partial interval \( I \) and a clear partial interval \( I' = s_1 \ldots s_k \) where \( t^G(g(\text{last}(I)), g(\text{first}(I'))) \), by \( I \oplus I' \) we denote the partial interval \( I s_1 \ldots s_k \) such that for each \( i \) we have that \( s_i = \text{last}(I)s_i \). So \( \oplus \) joins two intervals in a way that
accounts for the history of the partial states. Clearly, $I \oplus I'$ is an interval if and only if $I$ is an interval. We also define the operation $\circ$ such that $I \circ I' = s_1 \ldots s_k$, i.e., it only returns the adjusted partial states of $I'$.

**Lemma 4.** Consider a model $M$, a formula $\varphi$, two intervals $I, I'$, and a partial interval $J$. If $MCT_I^\varphi = MCT_J^\varphi$, and $t^G(g(\text{last}(I)), g(\text{first}(J)))$, then $MCT_{I \oplus J}^\varphi = MCT_{I' \oplus J}^\varphi$.

**Proof.** Consider a formula $\varphi$, a model $M$, two intervals $I, I'$ and a partial state $s = g$ such that $t^G(g(\text{last}(I)), g)$. We show that if $MCT_I^\varphi = MCT_J^\varphi$, then $MCT_{I \oplus s}^\varphi = MCT_{I' \oplus s}^\varphi$. The consideration above can be used to prove the lemma by induction.

Assume that the root of $MCT_I^\varphi$ is labelled by $f, l, p_i, A_I$. Then the roots of both $MCT_I^\varphi$ and $MCT_J^\varphi$ are labelled by $f, g, \bot, A_I$, where for each $p \in \text{Var}$ we put $A_I(p)$ equal to the state that the automaton for $p$ reaches from $A_I(p)$ after reading $g$.

Assume that $X_1 \varphi_1, \ldots, X_k \varphi_k$ are the top-level sub-formulas of $\varphi$ and $i \in \{1, \ldots, k\}$ (if there are no such formulas, then the result follows directly). We show that for each $i$, the roots of $MCT_J^\varphi$ and $MCT_{I \oplus s}^\varphi$ have the same $X_i \varphi_i$-successors.

**Case 1.** $X_i$ is an epistemic modality. Consider any interval $J$ such that $I \oplus sR_{X_i} J$. Let $J = J' \oplus s'$. By the definition, $J' R_{X_i} I$ and $sR_{X_i} s'$. By the former, we have that $MCT_J^\varphi$ is an $X_i \varphi_i$-successor of the root in $MCT_J^\varphi$, and so $MCT_{I \oplus s}^\varphi$ is $X_i \varphi_i$-successor of the root in $MCT_{I \oplus s}^\varphi$. Therefore, $J'' \oplus s' R_{X_i} J' \oplus s$, and thus $MCT_{I \oplus s}^\varphi$ is $X_i \varphi_i$-successors of the root of $MCT_{I \oplus s}^\varphi$.

**Case 2.** $X_i = \langle A \rangle$. Consider any interval $J$ such that $I \oplus sR_A J$. Then there is a clear partial interval $J$ such that $J = I \circ J$. Let $J' = I' \circ J$. It holds that $I' \circ sR_A J'$. By Lemma 2 we have $MCT_J^\varphi = MCT_{J'}^\varphi$.

Therefore, the $\langle A \rangle \varphi_i$-successors of the root in $MCT_J^\varphi$ are also $\langle A \rangle \varphi_i$-successors of the root in $MCT_{I \oplus s}^\varphi$. The other direction is similar.

**Case 3.** $X_i = \langle B \rangle$. Consider any interval $J$ such that $I \oplus sR_B J$. Then, there is a clear partial interval $J$ such that $J = (I \oplus s) \circ J$.

Let $J' = (I' \oplus s) \circ J$. It holds that $I' \circ sR_B J'$ By Lemma 2 we have $MCT_J^\varphi = MCT_{J'}^\varphi$.

Again, we conclude that the $\langle B \rangle \varphi_i$-successors of the root in $MCT_{I \oplus s}^\varphi$ are the same as $\langle B \rangle \varphi_i$-successors of the root in $MCT_{I \oplus s}^\varphi$.

**Case 4.** $X_i = \langle N \rangle$. The proof is similar to the one of Case 2 for $X_i = \langle A \rangle$.

By exploiting the Lemma above, we can now give the main result of this section.

The proof of Theorem [2] is by induction on the structure of $\varphi$.

The cases for $\varphi = p, \varphi = p_i, \varphi = \neg \varphi', \varphi = \varphi_1 \land \varphi_2, \varphi = K_i \varphi'$, and $\varphi = C_G \varphi'$ for some sub-formulas $\varphi', \varphi_1, \varphi_2$, follow from the fact that the semantic rules are the same in both semantics.
Assume that $\varphi = X\varphi'$ for some $\varphi'$, and $X \in \langle A \rangle, \langle B \rangle, \langle N \rangle$. If $M, I \models_B \varphi$, then there is an interval $I'$ of bounded size such that $M, I' \models_B \varphi'$ and $IR_X I'$. By the induction hypothesis, $M, I' \models \varphi'$ and therefore $M, I \models \varphi$.

If $M, I \models \varphi$, then there is an interval $I'$ such that $M, I' \models \varphi'$ and $IR_X I'$. Let $I'$ be the shortest possible interval with this property. We show that $|I'| \leq |I| + f^{IS}(\varphi)$.

Let $I' = s_1 \ldots s_t$ and $I'_k$ denote the prefix $s_1 \ldots s_k$ of $I'$. Assume that $|I'| > |I| + f^{IS}(\varphi')$. By Lemma 1 there are two prefixes $I'_k, I'_l$ such that $|I| < k < l$ and $MCT^\varphi_{I'_k} = MCT^\varphi_{I'_l}$. Let $J$ be a clear partial interval such that $I' = I'_l \oplus J$. By Lemma 4, we have that $MCT^\varphi_{I'_k \oplus J} = MCT^\varphi_{I'_l \oplus J}$. Clearly, $|I'_k \oplus J| < |I'|$ and, by Lemma 3, $M, I'_k \oplus J \models \varphi'$. Since $k > |I|$, it follows that $IR_X I'_k \oplus J$ (the condition $k > |I|$ is only required for $\langle B \rangle$ since $J$ has to contain $I$ as a prefix). But we assumed that $I'$ was the shortest interval; so this is a contradiction. It follows that $|I'| \leq |I| + f^{IS}(\varphi)$. $\Box$

Finally, the proof of Theorem 3 goes as follows. By Theorem 5, the bounded semantics and the unbounded semantics are equivalent. By Theorem 4, model checking the $ABL$ fragment of $EHS^+$ with bounded semantics is decidable. Therefore, model checking the $ABL$ fragment of $EHS^+$ with unbounded semantics is also decidable. Indeed, the procedure VERIFY given in Algorithm 2 solves the problem.