University of Udine

Department of Mathematics and Computer Science

PREPRINT

Model Checking the Logic of Allen's Relations Meets and Started-by is PNP-Complete

Laura Bozzelli, Alberto Molinari, Angelo Montanari, Adriano Peron, Pietro Sala

Preprint nr.: 2/2016

Reports available from: https://www.dimi.uniud.it/preprints/

Model Checking the Logic of Allen's Relations *Meets* and *Started-by* is $\mathbf{P}^{\mathbf{NP}}$-Complete

Laura Bozzelli
Technical University of Madrid (UPM), Madrid, Spain
laura.bozzelli@fi.upm.es

Alberto Molinari Angelo Montanari
University of Udine, Udine, Italy
molinari.alberto@gmail.com angelo.montanari@uniud.it

Adriano Peron
University of Napoli "Federico II", Napoli, Italy
adrperon@unina.it

Pietro Sala
University of Verona, Verona, Italy
pietro.sala@univr.it

In the plethora of fragments of Halpern and Shoham's modal logic of time intervals (HS), the logic AB of Allen's relations *Meets* and *Started-by* is at a central position. Statements that may be true at certain intervals, but at no sub-interval of them, such as accomplishments, as well as metric constraints about the length of intervals, that force, for instance, an interval to be at least (resp., at most, exactly) k points long, can be expressed in AB. Moreover, over the linear order of the natural numbers $\mathbb{N}$, it subsumes the (point-based) logic LTL, as it can easily encode the next and until modalities. Finally, it is expressive enough to capture the ω-regular languages, that is, for each ω-regular expression R there exists an AB formula φ such that the language defined by R coincides with the set of models of φ over $\mathbb{N}$. It has been shown that the satisfiability problem for AB over $\mathbb{N}$ is **EXPSPACE**-complete. Here we prove that, under the homogeneity assumption, its model checking problem is $\Delta_2^p = \mathbf{P}^{\mathbf{NP}}$-complete (for the sake of comparison, the model checking problem for full HS is **EXPSPACE**-hard, and the only known decision procedure is nonelementary). Moreover, we show that the modality for the Allen relation *Met-by* can be added to AB at no extra cost ($\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ is $\mathbf{P}^{\mathbf{NP}}$-complete as well).

1 Introduction

In this paper, we investigate the model checking problem for the interval logic of Allen's Relations *Meets* and *Started-by*. Given a model of a system (generally, a Kripke structure) and a temporal logic formula, which specifies the expected properties of the system, model checking algorithms verify, in fully automatic way, whether the model satisfies the formula; if this is not the case, they provide a counterexample, that is, a computation of the system failing to satisfy some property. Model checking has been successfully employed in formal verification as well as in various areas of AI, ranging from planning to configuration and multi-agent systems [7, 14].

Standard point-based temporal logics, such as LTL, CTL, and CTL* [6, 20], are commonly used as specification languages. Even though they turn out to be well-suited for a variety of application domains, there are relevant system properties, involving, for instance, actions with duration, accomplishments, and temporal aggregations, which are inherently "interval-based" and thus cannot be properly dealt with by temporal logics that allow one to predicate over computation states only. To overcome these limitations, one can resort to *interval temporal logics* (ITLs), that take intervals—instead of points—as their primitive entities [9], which have been successfully applied in various areas of computer science and AI, including hardware and software verification, computational linguistics, and planning [11, 19, 21, 24].

ITL model checking is the context of this paper. In order to check interval properties of computations, one needs to collect information about states into computation stretches: each finite path of a Kripke

Submitted to:
GandALF 2016

© L. Bozzelli, A. Molinari, A. Montanari, A. Peron and P. Sala
This work is licensed under the Creative Commons
Attribution-Noncommercial-No Derivative Works License.

structure is interpreted as an interval, whose labelling is defined on the basis of the labelling of the component states. Among ITLs, Halpern and Shoham's modal logic of time intervals HS [9] is the main reference. It features one modality for each possible ordering relation between a pair of intervals apart from equality (the so-called Allen's relations [1]). The *satisfiability problem* for HS has been thoroughly studied, and it turns out to be highly undecidable for all relevant (classes of) linear orders [9]. The same holds for most HS fragments [3]; however, some meaningful exceptions exist, including the logic of temporal neighbourhood $\mathsf{A\overline{A}}$ and the logic of sub-intervals D [4, 5]. The *model checking problem* for HS has entered the research agenda only recently [2, 11, 12, 13, 15, 16, 17, 18]. In [15], Molinari et al. deal with model checking for full HS over Kripke structures under the homogeneity assumption [22], showing its non-elementary decidability by means of a suitable small model theorem (**EXPSPACE**-hardness has been proved in [2]). Since then, the attention was brought to HS fragments, which, similarly to what happens with satisfiability, are often computationally better.

In this paper we first prove that model checking for the logic $\mathsf{A\overline{A}B}$ (resp., $\mathsf{A\overline{A}E}$) of Allen's relations *Meets*, *Met-by*, and *Started-by* (resp., *Finished-by*) is in $\mathbf{P^{NP}}$; then we prove that its fragment AB (resp., $\mathsf{\overline{A}E}$) is $\mathbf{P^{NP}}$-hard; finally we show that its fragment $\mathsf{\overline{A}B}$ (resp., AE) belongs to $\mathbf{P^{NP}}[O(\log^2 n)]$ and it is $\mathbf{P^{NP}}[O(\log n)]$-hard. $\mathbf{P^{NP}}$ (also denoted as Δ_2^p) is the class of problems decided by a deterministic polynomial time Turing machine that queries an NP oracle. The classes $\mathbf{P^{NP}}[O(\log n)]$ and $\mathbf{P^{NP}}[O(\log^2 n)]$ are analogous, but the number of queries is bounded by $O(\log n)$ and $O(\log^2 n)$, respectively, being n the input size [8, 23]. These three classes are higher than both **NP** and **co-NP** in the *polynomial time hierarchy*, and closed under complement. It is worth noticing that, whereas we know many natural problems which are complete for Σ_2^p or Π_2^p (in general, for Σ_k^p and Π_k^p, with $k \geq 2$), the classes $\mathbf{P^{NP}}$, $\mathbf{P^{NP}}[O(\log n)]$, and $\mathbf{P^{NP}}[O(\log^2 n)]$ are not so "populated" (and neither are the classes Δ_k^p, for $k > 2$). Among the few natural problems complete for $\mathbf{P^{NP}}$, we would like to mention model checking for several fragments of CTL*, including CTL$^+$, ECTL$^+$, and FCTL [10]. As for the other two classes, very recently Molinari et al. have shown that model checking A, $\mathsf{\overline{A}}$, or $\mathsf{A\overline{A}}$ formulas is in $\mathbf{P^{NP}}[O(\log^2 n)]$ and hard for $\mathbf{P^{NP}}[O(\log n)]$ [18].

Related work. In [11, 12], Lomuscio and Michaliszyn address the model checking problem for some fragments of HS extended with epistemic modalities. Their semantic assumptions considerably differ from those made in [15], making it difficult to compare the outcomes of the two research lines. Moreover, they consider a restricted form of model checking, which verifies a specification against a single (finite) initial computation interval (this is in general a limitation, unless some operators of HS are available, such as $\langle \mathsf{A} \rangle$): their goal is indeed to reason about a given computation of a multi-agent system, rather than on all its admissible computations. Recently they have shown how to exploit regular expressions in order to specify the way in which the intervals of a Kripke structure get labelled [13]. Such an extension leads to a significant increase in the expressiveness of HS formulas.

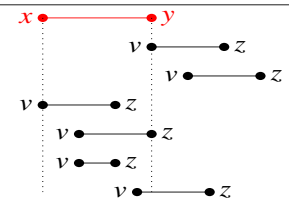
Organization of the paper. In the next section we introduce the fundamental elements of the model checking problem for HS and its fragments. Then, in Section 3, we provide a $\mathbf{P^{NP}}$ model checking algorithm for $\mathsf{A\overline{A}B}$ (and $\mathsf{A\overline{A}E}$) formulas. In Section 4 we prove the $\mathbf{P^{NP}}$-hardness of model checking for AB (and $\mathsf{\overline{A}E}$). $\mathbf{P^{NP}}$-completeness of AB, $\mathsf{\overline{A}E}$, $\mathsf{A\overline{A}B}$ and $\mathsf{A\overline{A}E}$ follows. In Appendix A we show that the problem for formulas of $\mathsf{\overline{A}B}$ and AE is in $\mathbf{P^{NP}}[O(\log^2 n)]$ and hard for $\mathbf{P^{NP}}[O(\log n)]$.

2 Preliminaries

The interval temporal logic HS. An interval algebra to reason about intervals and their relative order was proposed by Allen in [1], while a systematic logical study of interval representation and reasoning

Table 1: Allen's relations and corresponding HS modalities.

Allen relation	HS	Definition w.r.t. interval structures	Example
MEETS	$\langle \mathrm{A} \rangle$	$[x,y]\mathcal{R}_A[v,z] \iff y = v$	
BEFORE	$\langle \mathrm{L} \rangle$	$[x,y]\mathcal{R}_L[v,z] \iff y < v$	
STARTED-BY	$\langle \mathrm{B} \rangle$	$[x,y]\mathcal{R}_B[v,z] \iff x = v \wedge z < y$	
FINISHED-BY	$\langle \mathrm{E} \rangle$	$[x,y]\mathcal{R}_E[v,z] \iff y = z \wedge x < v$	
CONTAINS	$\langle \mathrm{D} \rangle$	$[x,y]\mathcal{R}_D[v,z] \iff x < v \wedge z < y$	
OVERLAPS	$\langle \mathrm{O} \rangle$	$[x,y]\mathcal{R}_O[v,z] \iff x < v < y < z$	



was done a few years later by Halpern and Shoham, who introduced the interval temporal logic HS featuring one modality for each Allen relation, but equality [9]. Table 1 depicts 6 of the 13 Allen's relations, together with the corresponding HS (existential) modalities. The other 7 relations are the 6 inverses (given a binary relation $\mathcal{R}$, the inverse $\overline{\mathcal{R}}$ is such that $b\overline{\mathcal{R}}a$ if and only if $a\mathcal{R}b$) and equality.

The HS language consists of a set of proposition letters $\mathcal{AP}$, the Boolean connectives $\neg$ and $\wedge$, and a temporal modality for each of the (non trivial) Allen's relations, i.e., $\langle \mathrm{A} \rangle$, $\langle \mathrm{L} \rangle$, $\langle \mathrm{B} \rangle$, $\langle \mathrm{E} \rangle$, $\langle \mathrm{D} \rangle$, $\langle \mathrm{O} \rangle$, $\langle \overline{\mathrm{A}} \rangle$, $\langle \overline{\mathrm{L}} \rangle$, $\langle \overline{\mathrm{B}} \rangle$, $\langle \overline{\mathrm{E}} \rangle$, $\langle \overline{\mathrm{D}} \rangle$, and $\langle \overline{\mathrm{O}} \rangle$. HS formulas are defined by the grammar $\psi ::= p \mid \neg\psi \mid \psi \wedge \psi \mid \langle X \rangle \psi \mid \langle \overline{X} \rangle \psi$, where $p \in \mathcal{AP}$ and $X \in \{A, L, B, E, D, O\}$. In the following, we shall also exploit as abbreviations the standard logical connectives for disjunction $\vee$, implication $\rightarrow$, and double implication $\leftrightarrow$. Furthermore, for any modality X, the dual universal modalities $[X]\psi$ and $[\overline{X}]\psi$ are defined as $\neg\langle X \rangle \neg\psi$ and $\neg\langle \overline{X} \rangle \neg\psi$, respectively. Finally, given any subset of Allen's relations $\{X_1, \cdots, X_n\}$, we denote by $\mathsf{X_1 \cdots X_n}$ the HS fragment featuring existential (and universal) modalities for $X_1, \ldots, X_n$ only.

W.l.o.g., we assume the *non-strict semantics of HS*, which admits intervals consisting of a single point[1]. Under such an assumption, all HS modalities can be expressed in terms of modalities $\langle \mathrm{B} \rangle, \langle \mathrm{E} \rangle, \langle \overline{\mathrm{B}} \rangle$, and $\langle \overline{\mathrm{E}} \rangle$ [9]. HS can thus be seen as a multi-modal logic with these 4 primitive modalities and its semantics can be defined over a multi-modal Kripke structure, called *abstract interval model*, where intervals are treated as atomic objects and Allen's relations as binary relations between pairs of intervals. Since later we will focus on some HS fragments excluding $\langle \overline{\mathrm{B}} \rangle$ and $\langle \overline{\mathrm{E}} \rangle$, we add both $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$ to the considered set of HS modalities.

Definition 1. *[15] An* abstract interval model *is a tuple $\mathcal{A} = (\mathcal{AP}, \mathbb{I}, A_{\mathbb{I}}, B_{\mathbb{I}}, E_{\mathbb{I}}, \sigma)$, where $\mathcal{AP}$ is a set of proposition letters, $\mathbb{I}$ is a possibly infinite set of atomic objects (worlds), $A_{\mathbb{I}}$, $B_{\mathbb{I}}$, and $E_{\mathbb{I}}$ are three binary relations over $\mathbb{I}$, and $\sigma : \mathbb{I} \mapsto 2^{\mathcal{AP}}$ is a (total) labeling function, which assigns a set of proposition letters to each world.*

In the interval setting, $\mathbb{I}$ is interpreted as a set of intervals and $A_{\mathbb{I}}$, $B_{\mathbb{I}}$, and $E_{\mathbb{I}}$ as Allen's relations A (*meets*), B (*started-by*), and E (*finished-by*), respectively; σ assigns to each interval in $\mathbb{I}$ the set of proposition letters that hold over it.

Given an abstract interval model $\mathcal{A} = (\mathcal{AP}, \mathbb{I}, A_{\mathbb{I}}, B_{\mathbb{I}}, E_{\mathbb{I}}, \sigma)$ and an interval $I \in \mathbb{I}$, the truth of an HS formula over I is inductively defined as follows:

- $\mathcal{A}, I \models p$ iff $p \in \sigma(I)$, for any $p \in \mathcal{AP}$;
- $\mathcal{A}, I \models \neg\psi$ iff it is not true that $\mathcal{A}, I \models \psi$ (also denoted as $\mathcal{A}, I \not\models \psi$);
- $\mathcal{A}, I \models \psi \wedge \phi$ iff $\mathcal{A}, I \models \psi$ and $\mathcal{A}, I \models \phi$;
- $\mathcal{A}, I \models \langle X \rangle \psi$, for $X \in \{A, B, E\}$, iff there exists $J \in \mathbb{I}$ such that $I X_{\mathbb{I}} J$ and $\mathcal{A}, J \models \psi$;
- $\mathcal{A}, I \models \langle \overline{X} \rangle \psi$, for $\overline{X} \in \{\overline{A}, \overline{B}, \overline{E}\}$, iff there exists $J \in \mathbb{I}$ such that $J X_{\mathbb{I}} I$ and $\mathcal{A}, J \models \psi$.

[1] All the results we prove in the paper hold for the strict semantics as well.

Kripke structures and abstract interval models. In model checking, finite state systems are usually modelled as Kripke structures. In [15], the authors define a mapping from Kripke structures to abstract interval models, that allows one to specify interval properties of computations by means of HS formulas.

Definition 2. *A* finite Kripke structure *is a tuple* $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$*, where* $\mathcal{AP}$ *is a set of proposition letters,* W *is a finite set of states,* $\delta \subseteq W \times W$ *is a left-total relation between pairs of states,* $\mu : W \mapsto 2^{\mathcal{AP}}$ *is a total labelling function, and* $w_0 \in W$ *is the initial state.*

For all $w \in W$, $\mu(w)$ is the set of proposition letters that hold at w, while δ is the transition relation that describes the evolution of the system over time.

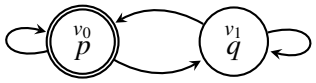


Figure 1: The Kripke structure $\mathcal{K}_2$.

Figure 1 depicts the finite Kripke structure $\mathcal{K}_2 = (\{p,q\}, \{v_0, v_1\}, \delta, \mu, v_0)$, where $\delta = \{(v_0,v_0),(v_0,v_1),(v_1,v_0),(v_1,v_1)\}$, $\mu(v_0) = \{p\}$, and $\mu(v_1) = \{q\}$. The initial state v_0 is identified by a double circle.

Definition 3. *A* track ρ *over a finite Kripke structure* $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ *is a finite sequence of states* $v_1 \cdots v_n$*, with* $n \geq 1$*, such that* $(v_i, v_{i+1}) \in \delta$ *for* $i = 1, \ldots, n-1$.

Let $\mathrm{Trk}_{\mathcal{K}}$ be the (infinite) set of all tracks over a finite Kripke structure $\mathcal{K}$. For any track $\rho = v_1 \cdots v_n \in \mathrm{Trk}_{\mathcal{K}}$, we define:

- $|\rho| = n$, $\mathrm{fst}(\rho) = v_1$, $\mathrm{lst}(\rho) = v_n$, and for $1 \leq i \leq |\rho|$, $\rho(i) = v_i$;
- $\mathrm{states}(\rho) = \{v_1, \cdots, v_n\} \subseteq W$;
- $\rho(i,j) = v_i \cdots v_j$, with $1 \leq i \leq j \leq |\rho|$, is the subtrack of ρ bounded by i and j;
- $\mathrm{Pref}(\rho) = \{\rho(1,i) \mid 1 \leq i \leq |\rho|-1\}$ and $\mathrm{Suff}(\rho) = \{\rho(i,|\rho|) \mid 2 \leq i \leq |\rho|\}$ are the sets of all proper prefixes and suffixes of ρ, respectively.

Finally, if $\mathrm{fst}(\rho) = w_0$ (the initial state of $\mathcal{K}$), ρ is called an *initial track*.

An abstract interval model (over $\mathrm{Trk}_{\mathcal{K}}$) can be naturally associated with a finite Kripke structure $\mathcal{K}$ by considering the set of intervals as the set of tracks of $\mathcal{K}$. Since $\mathcal{K}$ has loops (δ is left-total), the number of tracks in $\mathrm{Trk}_{\mathcal{K}}$, and thus the number of intervals, is infinite.

Definition 4. *The* abstract interval model induced by a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ *is* $\mathcal{A}_{\mathcal{K}} = (\mathcal{AP}, \mathbb{I}, A_{\mathbb{I}}, B_{\mathbb{I}}, E_{\mathbb{I}}, \sigma)$*, where* $\mathbb{I} = \mathrm{Trk}_{\mathcal{K}}$*,* $A_{\mathbb{I}} = \{(\rho,\rho') \in \mathbb{I} \times \mathbb{I} \mid \mathrm{lst}(\rho) = \mathrm{fst}(\rho')\}$*,* $B_{\mathbb{I}} = \{(\rho,\rho') \in \mathbb{I} \times \mathbb{I} \mid \rho' \in \mathrm{Pref}(\rho)\}$*,* $E_{\mathbb{I}} = \{(\rho,\rho') \in \mathbb{I} \times \mathbb{I} \mid \rho' \in \mathrm{Suff}(\rho)\}$*, and* $\sigma : \mathbb{I} \mapsto 2^{\mathcal{AP}}$ *is such that* $\sigma(\rho) = \bigcap_{w \in \mathrm{states}(\rho)} \mu(w)$*, for all* $\rho \in \mathbb{I}$.

Relations $A_{\mathbb{I}}, B_{\mathbb{I}}$, and $E_{\mathbb{I}}$ are interpreted as the Allen's relations A, B, and E, respectively. Moreover, according to the definition of σ, $p \in \mathcal{AP}$ holds over $\rho = v_1 \cdots v_n$ iff it holds over all the states $v_1, \cdots, v_n$ of ρ. This conforms to the *homogeneity principle*, according to which a proposition letter holds over an interval if and only if it holds over all its subintervals [22].

Definition 5. *Let* $\mathcal{K}$ *be a finite Kripke structure and* ψ *be an HS formula; we say that a track* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *satisfies* ψ*, denoted as* $\mathcal{K}, \rho \models \psi$*, iff it holds that* $\mathcal{A}_{\mathcal{K}}, \rho \models \psi$*. Moreover, we say that* $\mathcal{K}$ *models* ψ*, denoted as* $\mathcal{K} \models \psi$*, iff for all* initial *tracks* $\rho' \in \mathrm{Trk}_{\mathcal{K}}$ *it holds that* $\mathcal{K}, \rho' \models \psi$*. The* model checking problem *for HS over finite Kripke structures is the problem of deciding whether* $\mathcal{K} \models \psi$.

We conclude with a simple example (a simplified version of the one given in [15]), showing that the fragments investigated in this paper can express meaningful properties of state transition systems.

In Figure 2, we provide an example of a finite Kripke structure $\mathcal{K}_{Sched}$ that models the behaviour of a scheduler serving three processes which are continuously requesting the use of a common resource. The initial state is v_0: no process is served in that state. In the states v_i and $\overline{v}_i$, with $i \in \{1,2,3\}$, the i-th process is served (this is denoted by the fact that p_i holds in those states). For the sake of readability,

edges are marked either by r_i, for $request(i)$, or by u_i, for $unlock(i)$. Edge labels do not have a semantic value, that is, they are neither part of the structure definition, nor proposition letters; they are simply used to ease reference to edges. Process i is served in state v_i, then, after "some time", a transition u_i from v_i to $\overline{v}_i$ is taken; subsequently, process i cannot be served again immediately, as v_i is not directly reachable from $\overline{v}_i$ (the scheduler cannot serve the same process twice in two successive rounds). A transition r_j, with $j \neq i$, from $\overline{v}_i$ to v_j is then taken and process j is served. This structure can easily be generalised to a higher number of processes.

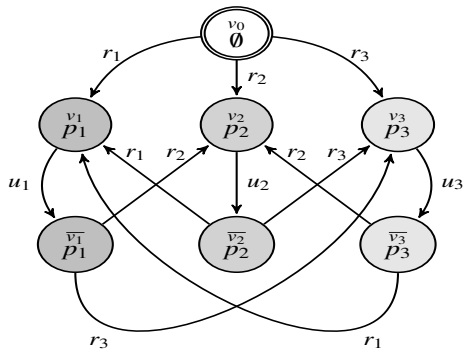


Figure 2: The Kripke structure $\mathcal{K}_{Sched}$.

We now show how some meaningful properties to be checked over $\mathcal{K}_{Sched}$ can be expressed in the HS fragment $\overline{\mathsf{A}}\mathsf{E}$. In all the following formulas, we force the validity of the considered properties over all legal computation sub-intervals by using the modality $[E]$ (all computation sub-intervals are suffixes of at least one initial track of the Kripke structure). The first formula requires that at least 2 proposition letters are witnessed in any suffix of length at least 4 of an initial track. Since a process cannot be executed twice in a row, it is satisfied by $\mathcal{K}_{Sched}$: $\mathcal{K}_{Sched} \models [E]\big(\langle \mathrm{E}\rangle^3 \top \rightarrow (\chi(p_1,p_2) \vee \chi(p_1,p_3) \vee \chi(p_2,p_3))\big)$ where $\chi(p,q) = \langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, p \wedge \langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, q$. The second formula requires that, in any suffix of length at least 11 of an initial track, process 3 is executed at least once in some internal states (*non starvation*). $\mathcal{K}_{Sched}$ does not satisfy it, because the scheduler can postpone the execution of a process ad libitum: $\mathcal{K}_{Sched} \not\models [E](\langle \mathrm{E}\rangle^{10} \top \rightarrow \langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, p_3)$. The third formula requires that, in any suffix of length at least 6 of an initial track, p_1, p_2, and p_3 are all witnessed. The only way to satisfy this property would be to force the scheduler to execute the three processes in a strictly periodic manner (*strict alternation*), that is, $p_i p_j p_k p_i p_j p_k p_i p_j p_k \cdots$, for $i,j,k \in \{1,2,3\}$ and $i \neq j \neq k \neq i$, but $\mathcal{K}_{Sched}$ does not meet such a requirement: $\mathcal{K}_{Sched} \not\models [E](\langle \mathrm{E}\rangle^5 \rightarrow (\langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, p_1 \wedge \langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, p_2 \wedge \langle \mathrm{E}\rangle \langle \overline{\mathrm{A}}\rangle\, p_3))$.

The general picture. We now describe known and new complexity results about the model checking problem for HS fragments (see Figure 3 for a graphical account).

In [15], Molinari et al. have shown that, given a Kripke structure $\mathcal{K}$ and a bound k on the structural complexity of HS formulas, i.e., on the nesting depth of $\langle \mathrm{E}\rangle$ and $\langle \mathrm{B}\rangle$ modalities, it is possible to obtain a *finite* representation for $\mathcal{A}_{\mathcal{K}}$, which is equivalent to $\mathcal{A}_{\mathcal{K}}$ with respect to satisfiability of HS formulas with structural complexity less than or equal to k. Then, by exploiting such a representation, they proved that the model checking problem for (full) HS is decidable, providing an algorithm with non-elementary complexity. In [2], **EXPSPACE**-hardness of the fragment BE, and thus of full HS, has been shown.

The fragments $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{B}}\overline{\mathsf{E}}$ have been systematically studied in [17]. For each of them, an **EXPSPACE** model checking algorithm has been devised that, for any track of the Kripke structure, finds a satisfiability-preserving track of bounded length (*track representative*). In this way, the model checking algorithm needs to check only tracks with a bounded maximum length. **PSPACE**-hardness of the model checking problem for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{B}}\overline{\mathsf{E}}$ has been proved in [16] (if a succinct encoding of formulas is exploited, the algorithm remains in **EXPSPACE**, but a **NEXPTIME** lower bound can be given [17]). Finally, it has been shown that formulas satisfying a constant bound on the nesting depth of $\langle \mathrm{B}\rangle$ (respectively, $\langle \mathrm{E}\rangle$) can be checked in polynomial working space [17].

Some well-behaved HS fragments, namely, $\mathsf{A}\overline{\mathsf{A}\mathsf{B}\mathsf{E}}$, $\overline{\mathsf{B}}$, $\overline{\mathsf{E}}$, $\mathsf{A}\overline{\mathsf{A}}$, A, and $\overline{\mathsf{A}}$, which are still expressive enough to capture meaningful interval properties of state transition systems and whose model checking problem has a computational complexity markedly lower than that of full HS, have been identified in

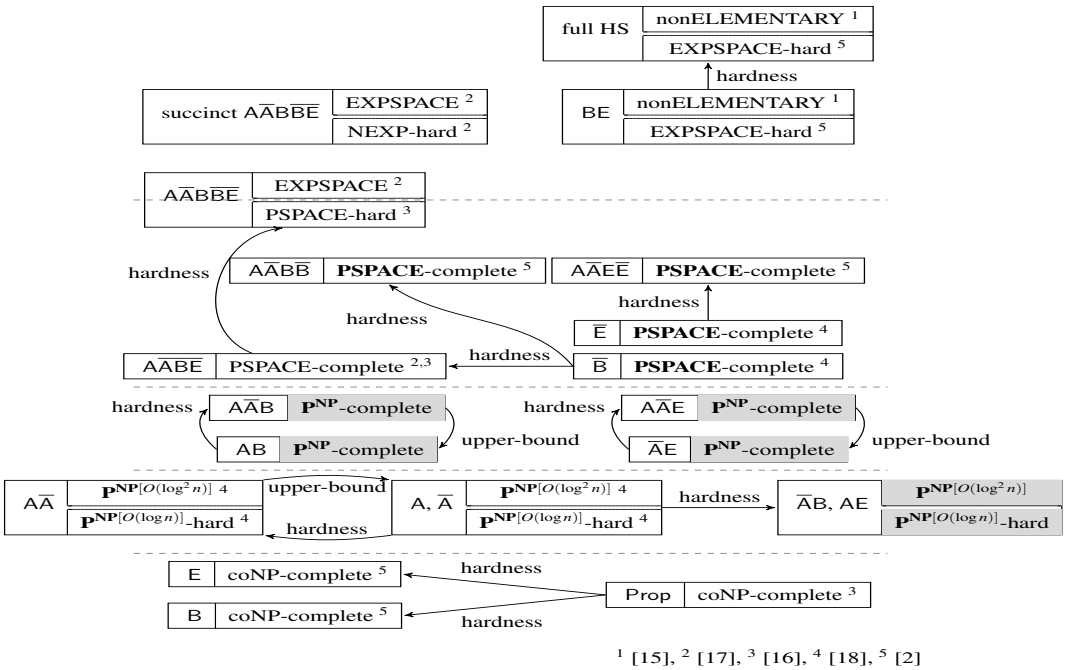


Figure 3: Complexity of the model checking problem for HS fragments: known results are depicted in white boxes, new ones in gray boxes.

[16, 18]. In particular the authors proved that the problem is **PSPACE**-complete for the fragments $\mathsf{A\overline{A}B\overline{E}}$, $\overline{\mathsf{B}}$, and $\overline{\mathsf{E}}$, and in between $\mathbf{P}^{\mathbf{NP}[O(\log n)]}$ and $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ [8, 23] for $\mathsf{A\overline{A}}$, A, and $\overline{\mathsf{A}}$. Two other well-behaved fragments, namely, $\mathsf{A\overline{A}B\overline{B}}$ and $\mathsf{A\overline{A}E\overline{E}}$, have been investigated in [2], showing that their model checking problem is **PSPACE**-complete. In addition, the authors showed that B and E are **co-NP**-complete (the same complexity as the model checking problem for the purely propositional HS fragment Prop [16]).

In this paper, we complete the analysis of the sub-fragments of $\mathsf{A\overline{A}B\overline{B}}$ (resp., $\mathsf{A\overline{A}E\overline{E}}$). In Section 3, we devise a $\mathbf{P}^{\mathbf{NP}}$ model checking algorithm for $\mathsf{A\overline{A}B}$ (resp., $\mathsf{A\overline{A}E}$). Then, in Section 4, we prove that AB (resp., $\overline{\mathsf{A}}\mathsf{E}$) is hard for $\mathbf{P}^{\mathbf{NP}}$. It immediately follows that model checking for AB and $\mathsf{A\overline{A}B}$ (resp., $\overline{\mathsf{A}}\mathsf{E}$ and $\mathsf{A\overline{A}E}$) formulas over finite Kripke structures is $\mathbf{P}^{\mathbf{NP}}$-complete. Finally, we show that $\overline{\mathsf{A}}\mathsf{B}$ (resp., AE) are in $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ (the proof is reported in the appendix) and hard for $\mathbf{P}^{\mathbf{NP}[O(\log n)]}$ (the hardness follows from that of $\overline{\mathsf{A}}$, resp., A [18]).

It is worth pointing out that the fragment $\overline{\mathsf{A}}\mathsf{B}$ belongs to a lower complexity class than the fragment AB (the same for the symmetric fragments AE and $\overline{\mathsf{A}}\mathsf{E}$). Such a difference can be explained as follows.

Let us consider a formula $\langle \mathrm{B} \rangle \langle \mathrm{A} \rangle \theta$, which belongs to AB. A track ρ satisfies $\langle \mathrm{B} \rangle \langle \mathrm{A} \rangle \theta$ if there exists a prefix $\tilde{\rho}$ of ρ from which a branch satisfying θ departs, i.e., a track starting from $\mathrm{lst}(\tilde{\rho})$. This amounts to say that AB allows one to impose specific constraints on the branches departing from a state occurring in a given path. Such an ability will be exploited in Section 4 to prove the $\mathbf{P}^{\mathbf{NP}}$-hardness of AB.

Conversely, the fragment $\overline{\mathsf{A}}\mathsf{B}$ cannot express constraints of this form. For any given track ρ, modality $\langle \overline{\mathrm{A}} \rangle$ only allows one to constrain tracks leading to the first state of ρ. As for modality $\langle \mathrm{B} \rangle$, if we consider a prefix $\tilde{\rho}$ of ρ, the set of tracks leading to its first state is exactly the same as the set of those leading to the first state of ρ, as $\mathrm{fst}(\tilde{\rho}) = \mathrm{fst}(\rho)$ (see Algorithm 3 in Appendix A). Therefore, pairing $\langle \overline{\mathrm{A}} \rangle$ and $\langle \mathrm{B} \rangle$ does not give any advantage in terms of expressiveness. Such a weakness of $\overline{\mathsf{A}}\mathsf{B}$ gives an intuitive account of the reason why $\overline{\mathsf{A}}\mathsf{B}$ formulas can be checked in time $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$, instead of time $\mathbf{P}^{\mathbf{NP}}$.

Algorithm 1 $\mathtt{MC}(\mathcal{K}, \psi, \text{DIRECTION})$

1: **for** all $\langle \mathrm{A} \rangle\, \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ **do**
2: $\quad \mathtt{MC}(\mathcal{K}, \phi, \text{FORWARD})$
3: **for** all $\langle \overline{\mathrm{A}} \rangle\, \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ **do**
4: $\quad \mathtt{MC}(\mathcal{K}, \phi, \text{BACKWARD})$
5: **for** all $v \in W$ **do**
6: $\quad$ **if** DIRECTION is FORWARD **then**
7: $\quad\quad V_{\mathsf{A}}(\psi, v) \leftarrow \mathit{Success}(\mathtt{Oracle}(\mathcal{K}, \psi, v, \text{FORWARD}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}}))$
8: $\quad$ **else if** DIRECTION is BACKWARD **then**
9: $\quad\quad V_{\overline{\mathsf{A}}}(\psi, v) \leftarrow \mathit{Success}(\mathtt{Oracle}(\mathcal{K}, \psi, v, \text{BACKWARD}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}}))$

3 A $\mathbf{P}^{\mathbf{NP}}$ model checking algorithm for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ formulas

In this section, we present a model checking algorithm for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ formulas (Algorithm 1) belonging to the complexity class $\mathbf{P}^{\mathbf{NP}}$. We recall that $\mathbf{P}^{\mathbf{NP}}$ is the class of problems solvable in (deterministic) polynomial time exploiting an oracle for an **NP**-complete problem. W.l.o.g., we restrict our attention to $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ formulas devoid of occurrences of conjunctions and universal modalities (definable, as usual, in terms of disjunctions, negations, and existential modalities).

Algorithm 1 presents the model checking procedure for a formula ψ against a Kripke structure $\mathcal{K}$. It exploits two global vectors, V_{A} and $V_{\overline{\mathsf{A}}}$, which can be seen as the tabular representations of two Boolean functions taking as arguments a subformula ϕ of ψ and a state v of $\mathcal{K}$. The intuition is that the function $V_{\mathsf{A}}(\phi, v)$ (resp., $V_{\overline{\mathsf{A}}}(\phi, v)$) returns $\top$ if and only if there exists a track $\rho \in \mathrm{Trk}_{\mathcal{K}}$ starting from the state v (resp., leading to the state v) such that $\mathcal{K}, \rho \models \phi$. The procedure MC is initially invoked with parameters $(\mathcal{K}, \neg\psi, \text{FORWARD})$. During the execution, it instantiates the entries of V_{A} and $V_{\overline{\mathsf{A}}}$, which are exploited in order to answer the model checking problem $\mathcal{K} \models \psi$; this is, in the end, equivalent to checking whether $V_{\mathsf{A}}(\neg\psi, w_0) = \bot$, where w_0 is the initial state of $\mathcal{K}$.

Let us consider the model checking procedure MC in more detail. Besides the Kripke structure $\mathcal{K}$ and the formula ψ, MC features a third parameter, DIRECTION, which can be assigned the value FORWARD (resp., BACKWARD), that is used in combination with the modality $\langle \mathrm{A} \rangle$ (resp., $\langle \overline{\mathrm{A}} \rangle$) for a forward (resp., backward) unravelling of $\mathcal{K}$. MC is applied recursively on the nesting of modalities $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$ in the formula ψ (in the base case, ψ features no occurrences of $\langle \mathrm{A} \rangle$ or $\langle \overline{\mathrm{A}} \rangle$). In order to instantiate the Boolean vectors V_{A} and $V_{\overline{\mathsf{A}}}$, an oracle is invoked (lines 5–9) for each state v of the Kripke structure. Such an invocation is syntactically represented by $\mathit{Success}(\mathtt{Oracle}(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}}))$, and it returns $\top$ whenever there exists a computation of the non-deterministic algorithm $\mathtt{Oracle}(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ returning $\top$, namely, whenever there is a suitable track starting from, or leading to v (depending on the value of the parameter DIRECTION), and satisfying ψ.

We now introduce the notion of $\mathsf{A}\overline{\mathsf{A}}$-*modal subformulas* of ψ; these subformulas "direct" the recursive calls of MC.

Definition 6. *The set of* $\mathsf{A}\overline{\mathsf{A}}$-modal subformulas *of an* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ *formula* ψ*, denoted by* $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$*, is the set of subformulas of* ψ *having either the form* $\langle \mathrm{A} \rangle\, \psi'$ *or the form* $\langle \overline{\mathrm{A}} \rangle\, \psi'$*, for some* ψ'*, which are* not in the scope of any $\langle \mathrm{A} \rangle$ or $\langle \overline{\mathrm{A}} \rangle$ modality.

For instance, $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\langle \mathrm{A} \rangle \langle \overline{\mathrm{A}} \rangle\, q) = \{\langle \mathrm{A} \rangle \langle \overline{\mathrm{A}} \rangle\, q\}$ and $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}\big(\big(\langle \mathrm{A} \rangle\, p \wedge \langle \mathrm{A} \rangle \langle \overline{\mathrm{A}} \rangle\, q\big) \rightarrow \langle \mathrm{A} \rangle\, p\big) = \{\langle \mathrm{A} \rangle\, p, \langle \mathrm{A} \rangle \langle \overline{\mathrm{A}} \rangle\, q\}$.

MC is recursively called on each formula ϕ such that $\langle \mathrm{A} \rangle\, \phi$ or $\langle \overline{\mathrm{A}} \rangle\, \phi$ belongs to $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ (lines 1–4). In this way, we can recursively gather in the Boolean vectors V_{A} and $V_{\overline{\mathsf{A}}}$, by increasing nesting

Algorithm 2 `Oracle`$(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$

1: $\tilde{\rho} \leftarrow$ `A_track`$(\mathcal{K}, v, |W| \cdot (2|\psi|+1)^2, \text{DIRECTION})$ ◁ *a track of* $\mathcal{K}$ *from/to* v *of length* $\leq |W| \cdot (2|\psi|+1)^2$
2: **for all** $\langle \mathrm{A} \rangle\, \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ **do**
3: **for** $i = 1, \cdots, |\tilde{\rho}|$ **do**
4: $T[\langle \mathrm{A} \rangle\, \phi, i] \leftarrow V_{\mathsf{A}}(\phi, \tilde{\rho}(i))$
5: **for all** $\langle \overline{\mathrm{A}} \rangle\, \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ **do**
6: **for** $i = 1, \cdots, |\tilde{\rho}|$ **do**
7: $T[\langle \overline{\mathrm{A}} \rangle\, \phi, i] \leftarrow V_{\overline{\mathsf{A}}}(\phi, \mathrm{fst}(\tilde{\rho}))$
8: **for** all subformulas φ of ψ, not contained in (or equal to) $\mathsf{A}\overline{\mathsf{A}}$-modal subformulas of ψ, by increasing length **do**
9: **if** $\varphi = p$, for $p \in \mathcal{AP}$ **then**
10: $T[p, 1] \leftarrow p \in \mu(\mathrm{fst}(\tilde{\rho}))$
11: **for** $i = 2, \cdots, |\tilde{\rho}|$ **do**
12: $T[p, i] \leftarrow T[p, i-1]$ and $p \in \mu(\tilde{\rho}(i))$
13: **else if** $\varphi = \neg\varphi_1$ **then**
14: **for** $i = 1, \cdots, |\tilde{\rho}|$ **do**
15: $T[\varphi, i] \leftarrow$ not $T[\varphi_1, i]$
16: **else if** $\varphi = \varphi_1 \vee \varphi_2$ **then**
17: **for** $i = 1, \cdots, |\tilde{\rho}|$ **do**
18: $T[\varphi, i] \leftarrow T[\varphi_1, i]$ or $T[\varphi_2, i]$
19: **else if** $\varphi = \langle \mathrm{B} \rangle\, \varphi_1$ **then**
20: $T[\varphi, 1] \leftarrow \bot$
21: **for** $i = 2, \cdots, |\tilde{\rho}|$ **do**
22: $T[\varphi, i] \leftarrow T[\varphi, i-1]$ or $T[\varphi_1, i-1]$
23: **return** $T[\psi, |\tilde{\rho}|]$

depth of the modalities $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$, the oracle answers for all the formulas ψ' such that $\langle \mathrm{A} \rangle\, \psi'$ or $\langle \overline{\mathrm{A}} \rangle\, \psi'$ is a subformula (be it maximal or not) of ψ.

Let us now consider the *non-deterministic polynomial time* procedure `Oracle`$(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ reported in Algorithm 2, which is used as the basic engine by the oracle in the aforementioned model checking Algorithm 1. The idea underlying Algorithm 2 is first to non-deterministically generate a track $\tilde{\rho}$ by unravelling the Kripke structure $\mathcal{K}$ according to the parameter DIRECTION, and then to verify ψ over $\tilde{\rho}$. Such a procedure actually exploits a result proved in [2] (see, in particular, Theorem 10) stating a so-called "polynomial-size model-track property" for formulas of the fragment $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$: if ρ is a track of $\mathcal{K}$, ϕ is an $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$ formula, and $\mathcal{K}, \rho \models \phi$, then there exists $\rho' \in \mathrm{Trk}_{\mathcal{K}}$ such that $|\rho'| \leq |W| \cdot (2|\phi|+1)^2$, $\mathrm{fst}(\rho) = \mathrm{fst}(\rho')$, $\mathrm{lst}(\rho) = \mathrm{lst}(\rho')$, and $\mathcal{K}, \rho' \models \phi$. This property guarantees that, in order to check the satisfiability of a formula ϕ, it is enough to consider tracks having a length bounded by $|W| \cdot (2|\phi|+1)^2$. Such a result holds by symmetry for formulas of the fragment $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ as well.

An execution of `Oracle`$(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ starts (line 1) by *non-deterministically* generating a track $\tilde{\rho}$ (having a length of at most $|W| \cdot (2|\psi|+1)^2$), with v as its first (resp., last) state if the DIRECTION parameter is FORWARD (resp., BACKWARD). The track is generated by visiting the unravelling of $\mathcal{K}$ (resp., of $\mathcal{K}$ with transposed edges). The remaining part of the algorithm checks *deterministically* whether $\mathcal{K}, \tilde{\rho} \models \psi$. Such a verification is performed in a bottom-up way: for all the subformulas ϕ of ψ (starting from the minimal ones) and for all the prefixes $\tilde{\rho}(1, i)$ of $\tilde{\rho}$, with $1 \leq i \leq |\tilde{\rho}|$ (starting from the shorter ones), the procedure establishes whether $\mathcal{K}, \tilde{\rho}(1, i) \models \phi$ or not, and this result is stored in the entry $T[\phi, i]$ of a Boolean table T. Note that if the considered subformula of ψ is an element of $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$, the algorithm does not need to perform any verification, since the result is already available in the Boolean vectors V_{A} and $V_{\overline{\mathsf{A}}}$ (as a consequence of the previously completed calls to the procedure `Oracle`), and the table T is updated accordingly (lines 2–7). For the remaining subformulas, the entries of T are computed, as we already said, in a bottom-up fashion (lines 8–22). The result of the overall verification is stored in $T[\psi, |\tilde{\rho}|]$ and returned (line 23).

The algorithm presented here for checking formulas of $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ can trivially be adapted to check formulas of the symmetric fragment $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}$.

The following lemma establishes the soundness and completeness of the procedure `Oracle`.

Lemma 1. *Let* $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ *be a finite Kripke structure,* ψ *be an* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ *formula, and* $V_{\mathsf{A}}(\bullet, \bullet)$ *and* $V_{\overline{\mathsf{A}}}(\bullet, \bullet)$ *be two Boolean arrays. Let us assume that*

1. *for each* $\langle \mathrm{A} \rangle\, \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ *and* $v' \in W$, $V_{\mathsf{A}}(\phi, v') = \top$ *iff there exists* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that*

$\mathrm{fst}(\rho) = v'$ *and* $\mathcal{K}, \rho \models \phi$, *and*

2. *for each* $\langle \overline{\mathsf{A}} \rangle \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ *and* $v' \in W$, $V_{\overline{\mathsf{A}}}(\phi, v') = \top$ *iff there exists* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that* $\mathrm{lst}(\rho) = v'$ *and* $\mathcal{K}, \rho \models \phi$.

Then, $\texttt{Oracle}(\mathcal{K}, \psi, v, \textsc{direction}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ *features a successful computation (returning* $\top$*) iff:*

- *there exists* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that* $\mathrm{fst}(\rho) = v$ *and* $\mathcal{K}, \rho \models \psi$, *in the case* DIRECTION *is* FORWARD*;*
- *there exists* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that* $\mathrm{lst}(\rho) = v$ *and* $\mathcal{K}, \rho \models \psi$, *in the case* DIRECTION *is* BACKWARD.

Proof. It is easy to check that if $\tilde{\rho}$ is the track non-deterministically generated by `A_track` at line 1, then, for $i = 1, \cdots, |\tilde{\rho}|$, it holds that $\mathcal{K}, \tilde{\rho}(1,i) \models \phi \iff T[\phi, i] = \top$, either by hypothesis, when ϕ occurs in $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ (lines 2–7), or by construction, when ϕ does not occur in $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ (lines 8–22).

Let us now assume that the value of the parameter DIRECTION is FORWARD (the proof for the other direction is analogous).

($\Rightarrow$) If $\texttt{Oracle}(\mathcal{K}, \psi, v, \textsc{forward}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ features a successful computation, it means that there exists a track $\tilde{\rho} \in \mathrm{Trk}_{\mathcal{K}}$ (generated at line 1) such that $\mathrm{fst}(\tilde{\rho}) = v$ and $T[\psi, |\tilde{\rho}|] = \top$. Hence $\mathcal{K}, \tilde{\rho} \models \psi$.

($\Leftarrow$) If there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{fst}(\rho) = v$ and $\mathcal{K}, \rho \models \psi$, as a result of Theorem 10 of [2], there exists $\tilde{\rho} \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathcal{K}, \tilde{\rho} \models \psi$, $\mathrm{fst}(\tilde{\rho}) = \mathrm{fst}(\rho)$, and $|\tilde{\rho}| \leq |W| \cdot (2|\psi|+1)^2$. It follows that in some non-deterministic instance of $\texttt{Oracle}(\mathcal{K}, \psi, v, \textsc{forward}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$, $\texttt{A_track}(\mathcal{K}, v, |W| \cdot (2|\psi|+1)^2, \textsc{forward})$ returns such $\tilde{\rho}$ (at line 1). Finally, we have that $T[\psi, |\tilde{\rho}|] = \top$ as $\mathcal{K}, \tilde{\rho} \models \psi$, hence the considered instance of $\texttt{Oracle}(\mathcal{K}, \psi, v, \textsc{forward}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$ is successful. □

The following theorem states soundness and completeness of the model checking procedure `MC`.

Theorem 1. *Let* $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ *be a finite Kripke structure,* ψ *be an* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ *formula, and* $V_{\mathsf{A}}(\bullet, \bullet)$ *and* $V_{\overline{\mathsf{A}}}(\bullet, \bullet)$ *be two Boolean arrays. If* $\texttt{MC}(\mathcal{K}, \psi, \textsc{direction})$ *is executed, then for all* $v \in W$*:*

- *if* DIRECTION *is* FORWARD, $V_{\mathsf{A}}(\psi, v) = \top$ *iff there is* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that* $\mathrm{fst}(\rho) = v$ *and* $\mathcal{K}, \rho \models \psi$*;*
- *if* DIRECTION *is* BACKWARD, $V_{\overline{\mathsf{A}}}(\psi, v) = \top$ *iff there is* $\rho \in \mathrm{Trk}_{\mathcal{K}}$ *such that* $\mathrm{lst}(\rho) = v$ *and* $\mathcal{K}, \rho \models \psi$.

Proof. The proof is by induction on the number n of occurrences of $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$ modalities in ψ.
(Base case: $n = 0$) Since $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi) = \emptyset$, conditions 1 and 2 of Lemma 1 are satisfied and the thesis trivially holds.
(Inductive case: $n > 0$) The formula ψ contains at least an $\langle \mathrm{A} \rangle$ or an $\langle \overline{\mathrm{A}} \rangle$ modality. Hence $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi) \neq \emptyset$. Since each recursive call to `MC` (either at line 2 or 4) is performed on a formula ϕ featuring a number of occurrences of $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$ which is strictly less than the number of their occurrences in ψ, we can apply the inductive hypothesis. As a consequence, when the control flow reaches line 5, it holds that:

1. for each $\langle \mathrm{A} \rangle \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ and $v' \in W$, $V_{\mathsf{A}}(\phi, v') = \top$ iff there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{fst}(\rho) = v'$ and $\mathcal{K}, \rho \models \phi$;
2. for each $\langle \overline{\mathrm{A}} \rangle \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ and $v' \in W$, $V_{\overline{\mathsf{A}}}(\phi, v') = \top$ iff there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{lst}(\rho) = v'$ and $\mathcal{K}, \rho \models \phi$.

This implies that conditions 1 and 2 of Lemma 1 are fulfilled. Hence (assuming that DIRECTION is FORWARD), it holds that, for $v \in W$, $V_{\mathsf{A}}(\psi, v) = \top$ iff there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{fst}(\rho) = v$ and $\mathcal{K}, \rho \models \psi$. The case for DIRECTION = BACKWARD is symmetric, and thus omitted. □

As an immediate consequence we have that the procedure `MC` solves the model checking problem for $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ with an algorithm belonging to the complexity class $\mathbf{P}^{\mathbf{NP}}$.

Corollary 1. *Let* $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ *be a finite Kripke structure and* ψ *be an* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ *formula. If* $\texttt{MC}(\mathcal{K}, \neg\psi, \textsc{forward})$ *is executed, then* $V_{\mathsf{A}}(\neg\psi, w_0) = \bot \iff \mathcal{K} \models \psi$.

Corollary 2. *The model checking problem for* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$ *formulas over finite Kripke structures is in* $\mathbf{P}^{\mathbf{NP}}$.

Proof. Given a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_0)$ and an $\mathsf{A\overline{A}B}$ formula ψ, the number of recursive calls performed by $\mathtt{MC}(\mathcal{K}, \neg\psi, \text{FORWARD})$ is at most $|\psi|$. Each one costs $O(|\psi| + |W| \cdot (|\mathcal{K}| + |\psi| + |\psi| \cdot |W|))$, where the first addend comes from searching ψ for its modal subformulas (lines 1–4), and the second one from the preparation of the input for the oracle call, for each $v \in W$ (lines 5–9). Therefore its (deterministic) complexity is $O(|\psi|^2 \cdot |\mathcal{K}|^2)$. As for $\mathtt{Oracle}(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\mathsf{A}} \cup V_{\overline{\mathsf{A}}})$, its (non-deterministic) complexity is $O(|\psi|^3 \cdot |\mathcal{K}|)$, where $|\psi|$ is a bound to the number of subformulas and $O(|\psi|^2 \cdot |\mathcal{K}|)$ is the number of steps necessary to generate and check $\tilde{\rho}$. □

Symmetrically, by easily adapting the procedure `Oracle`, it is straightforward to prove that the model checking problem for $\mathsf{A\overline{A}E}$ formulas is in $\mathbf{P^{NP}}$ as well.

4 $\mathbf{P^{NP}}$-hardness of model checking for AB formulas

In this section, we prove that model checking for AB (and $\overline{\mathsf{A}}\mathsf{E}$) formulas is hard for $\mathbf{P^{NP}}$ by reducing the $\mathbf{P^{NP}}$-complete problem SNSAT (Sequentially Nested SATisfiability), a logical problem with nested satisfiability questions [10], to it. SNSAT is defined as follows.

Definition 7. *An instance $\mathscr{I}$ of SNSAT consists of a set of Boolean variables $X = \{x_1, \cdots, x_n\}$ and a set of Boolean formulas $\{F_1(Z_1), F_2(x_1, Z_2), \cdots, F_n(x_1, \cdots, x_{n-1}, Z_n)\}$, where, for $i = 1, \cdots, n$, $F_i(x_1, \cdots, x_{i-1}, Z_i)$ features variables in $\{x_1, \cdots, x_{i-1}\}$ and in $Z_i = \{z_i^1, \cdots, z_i^{j_i}\}$, the latter being a set of variables local to F_i, that is, $Z_i \cap Z_j = \emptyset$, for $j \neq i$, and $X \cap Z_i = \emptyset$. We denote $|X| (= n)$ by $|\mathscr{I}|$. Let $v_{\mathscr{I}}$ be the valuation of the variables in X defined as follows: $v_{\mathscr{I}}(x_i) = \top \iff F_i(v_{\mathscr{I}}(x_1), \cdots, v_{\mathscr{I}}(x_{i-1}), Z_i)$ is satisfiable (by assigning suitable values to the local variables $z_i^1, \cdots, z_i^{j_i} \in Z_i$). SNSAT is the problem of deciding, given an instance $\mathscr{I}$, with $|\mathscr{I}| = n$, whether $v_{\mathscr{I}}(x_n) = \top$. In such a case, we say that $\mathscr{I}$ is a positive instance of SNSAT.*

Given an SNSAT instance $\mathscr{I}$, with $|\mathscr{I}| = n$, the valuation $v_{\mathscr{I}}$ is unique and it can be easily computed by a $\mathbf{P^{NP}}$ algorithm as follows. A first query to a SAT oracle determines whether $v_{\mathscr{I}}(x_1)$ is $\top$ or $\bot$, since $v_{\mathscr{I}}(x_1) = \top$ iff $F_1(Z_1)$ is satisfiable. Then, we replace x_1 by the value $v_{\mathscr{I}}(x_1)$ in $F_2(x_1, Z_2)$ and another query to the SAT oracle is performed to determine whether $F_2(v_{\mathscr{I}}(x_1), Z_2)$ is satisfiable, gaining the value of $v_{\mathscr{I}}(x_2)$. This step is iterated other $n-2$ times, until the value for $v_{\mathscr{I}}(x_n)$ is obtained.

Let $\mathscr{I}$ be an instance of SNSAT, with $|\mathscr{I}| = n$. We now show how to build a finite Kripke structure $\mathcal{K}_{\mathscr{I}}$ and an AB formula $\Phi_{\mathscr{I}}$, by using *logarithmic working space*, such that $\mathscr{I}$ is a positive instance of SNSAT if and only if $\mathcal{K}_{\mathscr{I}} \models \Phi_{\mathscr{I}}$. Such a reduction is inspired by similar constructions from [10].

Let $Z = \bigcup_{i=1}^{n} Z_i$ and let $R = \{r_i \mid i = 1, \cdots, n\}$ and $R_i = R \setminus \{r_i\}$ be $n+1$ sets of auxiliary variables. The Kripke structure $\mathcal{K}_{\mathscr{I}}$ consists of a suitable composition of n instances of a *gadget* (an instance for each variable $x_1, \cdots, x_n \in X$). The structure of the gadget for x_i, with $1 \leq i \leq n$, is shown in Figure 4, assuming that the labeling of states (nodes) is defined as follows:

- $\mu(w_{x_i}) = X \cup Z \cup \{s, t\} \cup R_i$, and $\mu(\overline{w_{x_i}}) = (X \setminus \{x_i\}) \cup Z \cup \{s, t\} \cup R_i \cup \{p_{\overline{x_i}}\}$;
- for $u_i = 1, \cdots, j_i$, $\mu(w_{z_i^{u_i}}) = X \cup Z \cup \{s, t\} \cup R_i$, and $\mu(\overline{w_{z_i^{u_i}}}) = X \cup (Z \setminus \{z_i^{u_i}\}) \cup \{s, t\} \cup R_i$;
- $\mu(\overline{s_i}) = X \cup Z \cup \{t\} \cup R_i$.

The Kripke structure $\mathcal{K}_{\mathscr{I}}$ is obtained by sequentializing (adding suitable arcs) the n instances of the gadget (in reverse order, from x_n to x_1), adding a collector terminal state s_0, with labeling $\mu(s_0) = X \cup Z \cup \{s\} \cup R$, and setting w_{x_n} as the initial state. The overall construction is reported in Figure 5. Formally, $\mathcal{K}_{\mathscr{I}} = (X \cup Z \cup \{s, t\} \cup R \cup \{p_{\overline{x_i}} \mid i = 1, \cdots, n\}, W, \delta, \mu, w_{x_n})$. $\mathcal{K}_{\mathscr{I}}$ enjoys the following properties: (i) any track satisfying s does not pass through any $\overline{s_i}$, for $1 \leq i \leq n$; (ii) any track *not* satisfying t has s_0 as its

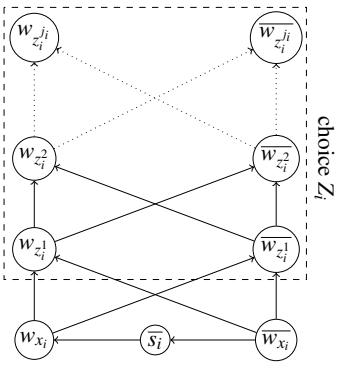


Figure 4: The gadget for x_i.

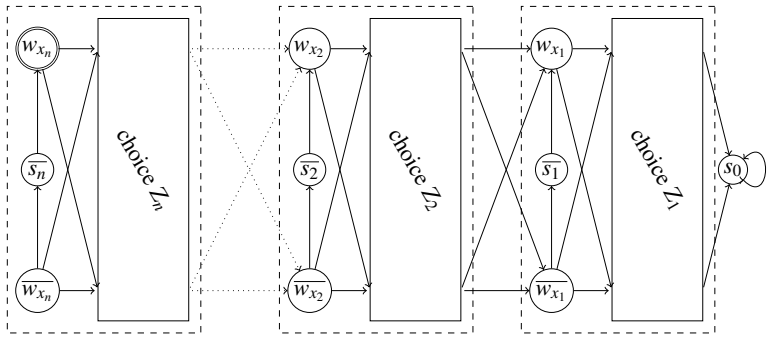


Figure 5: Kripke structure $\mathcal{K}_{\mathscr{I}}$ associated with an SNSAT instance $\mathscr{I}$, with $|\mathscr{I}| = n$. Notice that the states $\overline{s_n}$ and $\overline{w_{x_n}}$ are unreachable.

last state; (iii) any track *not* satisfying r_i passes through some state of the i-th gadget, for $1 \leq i \leq n$; (iv) the only track satisfying $p_{\overline{x_i}}$ is $\overline{w_{x_i}}$ (notice that $|\overline{w_{x_i}}| = 1$), for $1 \leq i \leq n$.

A track $\rho \in \mathrm{Trk}_{\mathcal{K}_{\mathscr{I}}}$ *induces* a truth assignment of all the proposition letters, denoted by ω_ρ, which is defined as $\omega_\rho(y) = \top \iff \mathcal{K}_{\mathscr{I}}, \rho \models y$, for any letter y. In the following, we will write $\omega_\rho(Z_i)$ for $\omega_\rho(z_i^1), \cdots, \omega_\rho(z_i^{j_i})$. In particular, if ρ starts from some state w_{x_i} or $\overline{w_{x_i}}$, and satisfies $s \wedge \neg t$ (that is, it reaches the collector state s_0 without visiting any node $\overline{s_j}$, for $1 \leq j \leq i$), ω_ρ fulfills the following conditions: for $1 \leq m \leq i$,

- if $w_{x_m} \in \mathrm{states}(\rho)$, then $\omega_\rho(x_m) = \top$, and if $\overline{w_{x_m}} \in \mathrm{states}(\rho)$, then $\omega_\rho(x_m) = \bot$;
- for $1 \leq u_m \leq j_m$, if $w_{z_m^{u_m}} \in \mathrm{states}(\rho)$, then $\omega_\rho(z_m^{u_m}) = \top$, and if $\overline{w_{z_m^{u_m}}} \in \mathrm{states}(\rho)$, then $\omega_\rho(z_m^{u_m}) = \bot$;

It immediately follows that $\mathcal{K}_{\mathscr{I}}, \rho \models F_m(x_1, \cdots, x_{m-1}, Z_m)$ iff $F_m(\omega_\rho(x_1), \cdots, \omega_\rho(x_{m-1}), \omega_\rho(Z_m)) = \top$. Finally, let $\mathscr{F}_{\mathscr{I}} = \{\psi_k \mid 0 \leq k \leq n+1\}$ be the set of formulas defined as: $\psi_0 = \bot$ and, for $k \geq 1$,

$$\psi_k = \langle \mathrm{A} \rangle \underbrace{\left[\begin{array}{c} (s \wedge \neg t) \wedge \bigwedge_{i=1}^{n} \Big((x_i \wedge \neg r_i) \rightarrow F_i(x_1, \cdots, x_{i-1}, Z_i) \Big) \\ \wedge \\ [B] \Big(\big(\bigvee_{i=1}^{n} \langle \mathrm{A} \rangle \, p_{\overline{x_i}} \big) \rightarrow \langle \mathrm{A} \rangle \left(\neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\ell_{=2} \wedge \neg \psi_{k-1}) \right) \Big) \end{array} \right]}_{\varphi_k},$$

where $\ell_{=2} = \langle \mathrm{B} \rangle \top \wedge [B][B]\bot$ is satisfied only by tracks of length 2. The first conjunct of φ_k $(s \wedge \neg t)$ forces the track to reach the collector state s_0, without visiting any state $\overline{s_j}$. The second conjunct checks that if the track assigns the truth value $\top$ to x_m passing through w_{x_m} (with $1 \leq m \leq n$), then $F_m(x_1, \cdots, x_{m-1}, Z_m)$ is satisfied by ω_ρ (which amounts to say that the SAT problem connected with Z_m has a positive answer, for the selected values of $x_1, \cdots, x_{m-1}$). Conversely, the third conjunct ensures that if the track assigns the truth value $\bot$ to some x_m by passing through $\overline{w_{x_m}}$, then, intuitively, the SAT problem connected with Z_m has no assignment satisfying $F_m(x_1, \cdots, x_{m-1}, Z_m)$. As a matter of fact, if ρ satisfies φ_k for some $k \geq 2$, and assigns $\bot$ to x_m, then there is a prefix $\tilde{\rho}$ of ρ ending in $\overline{w_{x_m}}$. Since $\bigvee_{i=1}^{n} \langle \mathrm{A} \rangle \, p_{\overline{x_i}}$ is satisfied by $\tilde{\rho}$, then $\langle \mathrm{A} \rangle \left(\neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\ell_{=2} \wedge \neg \psi_{k-1}) \right)$ must be satisfied as well. The only possibility is that the track $\overline{s_m} \cdot w_{x_m}$ does not model ψ_{k-1} (as $\overline{w_{x_m}} \cdot \overline{s_m}$ has to model $\langle \mathrm{A} \rangle (\ell_{=2} \wedge \neg \psi_{k-1})$). However, since $\psi_{k-1} = \langle \mathrm{A} \rangle \, \varphi_{k-1}$, this holds iff $\mathcal{K}, w_{x_m} \not\models \psi_{k-1}$.

The following theorem states the correctness of the construction.

Theorem 2. *Let $\mathscr{I}$ be an instance of SNSAT with $|\mathscr{I}| = n$, and let $\mathcal{K}_{\mathscr{I}}$ and $\mathscr{F}_{\mathscr{I}}$ be defined as above. For all $0 \leq k \leq n+1$ and all $r = 1, \cdots, n$, it holds that:*

1. *if $k \geq r$, then $v_{\mathscr{I}}(x_r) = \top \iff \mathcal{K}_{\mathscr{I}}, w_{x_r} \models \psi_k$;*

2. *if* $k \geq r+1$, *then* $v_{\mathscr{I}}(x_r) = \bot \iff \mathcal{K}_{\mathscr{I}}, \overline{w_{x_r}} \models \psi_k$.

Proof. The proof is by induction on $k \geq 0$.
(Base case: $k = 0$). The thesis trivially holds.
(Inductive case: $k \geq 1$). We first prove the ($\Leftarrow$) implication for both item 1 and item 2.

- (Item 1) Assume that $k \geq r$ and $\mathcal{K}_{\mathscr{I}}, w_{x_r} \models \psi_k$. Thus, there exists $\rho \in \mathrm{Trk}_{\mathcal{K}_{\mathscr{I}}}$ such that $\rho = w_{x_r} \cdots s_0$ does not pass through any $\overline{s_m}$, $1 \leq m \leq r$ and $\mathcal{K}_{\mathscr{I}}, \rho \models \varphi_k$. We show by induction on $1 \leq m \leq r$ that $\omega_\rho(x_m) = v_{\mathscr{I}}(x_m)$.
  - Let us consider first the case where ρ passes through w_{x_m}, implying that $\omega_\rho(x_m) = \top$; thus $\mathcal{K}_{\mathscr{I}}, \rho \models x_m \wedge \neg r_m$ and $\mathcal{K}_{\mathscr{I}}, \rho \models F_m(x_1, \cdots, x_{m-1}, Z_m)$. If $m = 1$ (base case), since F_1 is satisfiable, then $v_{\mathscr{I}}(x_1) = \top$. If $m \geq 2$ (inductive case), by the inductive hypothesis, it holds that $\omega_\rho(x_1) = v_{\mathscr{I}}(x_1)$, ..., $\omega_\rho(x_{m-1}) = v_{\mathscr{I}}(x_{m-1})$. Since $\mathcal{K}_{\mathscr{I}}, \rho \models F_m(x_1, \cdots, x_{m-1}, Z_m)$ or, equivalently, $F_m(\omega_\rho(x_1), \cdots, \omega_\rho(x_{m-1}), \omega_\rho(Z_m)) = \top$, it holds that $F_m(v_{\mathscr{I}}(x_1), \cdots, v_{\mathscr{I}}(x_{m-1}), \omega_\rho(Z_m)) = \top$ and, by definition of $v_{\mathscr{I}}$, $v_{\mathscr{I}}(x_m) = \top$.
  - Conversely, let us consider the case where ρ passes through $\overline{w_{x_m}}$, implying that $\omega_\rho(x_m) = \bot$ and $m < r$, as we are assuming $\mathrm{fst}(\rho) = w_{x_r}$. In this case, the prefix $w_{x_r} \cdots \overline{w_{x_m}}$ of ρ satisfies both $\bigvee_{i=1}^{n} \langle \mathrm{A} \rangle\, p_{\overline{x_i}}$ and $\langle \mathrm{A} \rangle \left(\neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\ell_{=2} \wedge \neg \psi_{k-1})\right)$. Therefore, $\mathcal{K}_{\mathscr{I}}, \overline{w_{x_m}} \cdot \overline{s_m} \models \langle \mathrm{A} \rangle (\ell_{=2} \wedge \neg \psi_{k-1})$ and $\mathcal{K}_{\mathscr{I}}, \overline{s_m} \cdot w_{x_m} \not\models \psi_{k-1}$, with $\psi_{k-1} = \langle \mathrm{A} \rangle\, \varphi_{k-1}$. Hence $\mathcal{K}_{\mathscr{I}}, w_{x_m} \not\models \psi_{k-1}$. Since $1 \leq m < r$, we have $1 \leq m < r \leq k$, thus $k' = k-1 \geq m \geq 1$. By the inductive hypothesis (on $k' = k-1$), we get that $v_{\mathscr{I}}(x_m) = \bot$.

  Therefore $v_{\mathscr{I}}(x_r) = \omega_\rho(x_r)$ and, since $w_{x_r} \in \mathrm{states}(\rho)$, we have that $\omega_\rho(x_r) = \top$ and the thesis, that is, $v_{\mathscr{I}}(x_r) = \top$, follows.
- (Item 2) Assume that $k \geq r+1$ and $\mathcal{K}_{\mathscr{I}}, \overline{w_{x_r}} \models \psi_k$. The proof follows the same steps as the previous case and it is thus only sketched: there exists $\rho \in \mathrm{Trk}_{\mathcal{K}_{\mathscr{I}}}$ such that $\rho = \overline{w_{x_r}} \cdots s_0$ does not pass through any $\overline{s_m}$, for $1 \leq m \leq r$, and $\mathcal{K}_{\mathscr{I}}, \rho \models \varphi_k$. The only thing which changes is that the prefix $\overline{w_{x_r}}$ satisfies $\bigvee_{i=1}^{n} \langle \mathrm{A} \rangle\, p_{\overline{x_i}}$, thus as before we get $\mathcal{K}_{\mathscr{I}}, w_{x_r} \not\models \psi_{k-1}$. Now, $k' = k-1 \geq r \geq 1$ and, by the inductive hypothesis (on $k' = k-1$), it holds that $v_{\mathscr{I}}(x_r) = \bot$.

We prove now the converse implication ($\Rightarrow$) for both item 1 and item 2.

- (Item 1) Assume that $k \geq r$ and $v_{\mathscr{I}}(x_r) = \top$. Let us consider the track $\rho \in \mathrm{Trk}_{\mathcal{K}_{\mathscr{I}}}$, $\rho = w_{x_r} \cdots s_0$ never passing through any $\overline{s_m}$, for $1 \leq m \leq r$, such that $w_{x_m} \in \mathrm{states}(\rho)$ if $v_{\mathscr{I}}(x_m) = \top$, and $\overline{w_{x_m}} \in \mathrm{states}(\rho)$ if $v_{\mathscr{I}}(x_m) = \bot$, for $1 \leq m \leq r$. Such a choice of ρ ensures that $v_{\mathscr{I}}(x_m) = \omega_\rho(x_m)$. In addition, the choice of ρ has to induce also the proper evaluation of local variables, that is, if $v_{\mathscr{I}}(x_m) = \top$, then for $1 \leq u_m \leq j_m$, $w_{z_m^{u_m}} \in \mathrm{states}(\rho)$ if $F_m(v_{\mathscr{I}}(x_1), \cdots, v_{\mathscr{I}}(x_{m-1}), Z_m)$ is satisfied for $z_m^{u_m} = \top$, $\overline{w_{z_m^{u_m}}} \in \mathrm{states}(\rho)$ otherwise. Notice that such a choice of ρ is always possible. We have to show that $\mathcal{K}_{\mathscr{I}}, \rho \models \varphi_k$, hence $\mathcal{K}_{\mathscr{I}}, w_{x_r} \models \psi_k$.
  - For all $1 \leq m \leq r$ such that $v_{\mathscr{I}}(x_m) = \top$, it holds that $F_m(v_{\mathscr{I}}(x_1), \cdots, v_{\mathscr{I}}(x_{m-1}), Z_m)$ is satisfiable. Hence, by our choice of ρ, $F_m(\omega_\rho(x_1), \cdots, \omega_\rho(x_{m-1}), \omega_\rho(Z_m)) = \top$, or, equivalently, $\mathcal{K}_{\mathscr{I}}, \rho \models F_m(x_1, \cdots, x_{m-1}, Z_m)$. Therefore, $\mathcal{K}_{\mathscr{I}}, \rho \models \bigwedge_{i=1}^{n} \Big((x_i \wedge \neg r_i) \rightarrow F_i(x_1, \cdots, x_{i-1}, Z_i)\Big)$.
  - Conversely, for all $1 \leq m < r$ such that $v_{\mathscr{I}}(x_m) = \bot$ ($m \neq r$ as, by hypothesis, $v_{\mathscr{I}}(x_r) = \top$), it holds that $\overline{w_{x_m}} \in \mathrm{states}(\rho)$. Since $m < r$, it holds that $k \geq r > m$ and $k-1 \geq m \geq 1$. By the inductive hypothesis, we have that $\mathcal{K}_{\mathscr{I}}, w_{x_m} \not\models \psi_{k-1}$. It follows that $\mathcal{K}_{\mathscr{I}}, \overline{s_m} \cdot w_{x_m} \models \neg \psi_{k-1} \wedge \ell_{=2}$, $\mathcal{K}_{\mathscr{I}}, \overline{w_{x_m}} \cdot \overline{s_m} \models \neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\neg \psi_{k-1} \wedge \ell_{=2})$ and $\mathcal{K}_{\mathscr{I}}, \overline{w_{x_m}} \models \langle \mathrm{A} \rangle (\neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\neg \psi_{k-1} \wedge \ell_{=2}))$. Hence, $\mathcal{K}_{\mathscr{I}}, \rho \models [B]((\bigvee_{i=1}^{n} \langle \mathrm{A} \rangle\, p_{\overline{x_i}}) \rightarrow \langle \mathrm{A} \rangle (\neg s \wedge \ell_{=2} \wedge \langle \mathrm{A} \rangle (\neg \psi_{k-1} \wedge \ell_{=2})))$.

  Combining the two cases, we can conclude that $\mathcal{K}_{\mathscr{I}}, \rho \models \varphi_k$.
- (Item 2) Assume that $k \geq r+1$ and $v_{\mathscr{I}}(x_r) = \bot$. The proof is as before and it is sketched. In

this case, we choose a track $\rho = \overline{w_{x_r}} \cdots s_0$. Since $k' = k-1 \geq r$, by the inductive hypothesis, $\mathcal{K}_{\mathscr{I}}, w_{x_r} \not\models \psi_{k-1}$, and we can prove that $\mathcal{K}_{\mathscr{I}}, \overline{w_{x_r}} \models \langle \mathsf{A} \rangle (\neg s \wedge \ell_{=2} \wedge \langle \mathsf{A} \rangle (\neg \psi_{k-1} \wedge \ell_{=2}))$. □

The correctness of the reduction from SNSAT to model checking for AB follows as a corollary.

Corollary 3. *Let $\mathscr{I}$ be an instance of SNSAT, with $|\mathscr{I}| = n$, and let $\mathcal{K}_{\mathscr{I}}$ and $\mathscr{F}_{\mathscr{I}}$ be defined as above. Then, $v_{\mathscr{I}}(x_n) = \top \iff \mathcal{K}_{\mathscr{I}} \models [B]\bot \to \psi_n$.*

Proof. By Theorem 2, $v_{\mathscr{I}}(x_n) = \top \iff \mathcal{K}_{\mathscr{I}}, w_{x_n} \models \psi_n$. If $v_{\mathscr{I}}(x_n) = \top$, then $\mathcal{K}_{\mathscr{I}}, w_{x_n} \models \psi_n$ and, since w_{x_n} is the only initial track satisfying $[B]\bot$ (only satisfiable by tracks of length 1), $\mathcal{K}_{\mathscr{I}} \models [B]\bot \to \psi_n$. Conversely, if $\mathcal{K}_{\mathscr{I}} \models [B]\bot \to \psi_n$, then $\mathcal{K}_{\mathscr{I}}, w_{x_n} \models \psi_n$, allowing us to conclude that $v_{\mathscr{I}}(x_n) = \top$. □

Eventually we can state the complexity of the problem.

Corollary 4. *The model checking problem for* AB *formulas over finite Kripke structures is* $\mathbf{P^{NP}}$*-hard (under* **LOGSPACE** *reductions).*

Proof. The result follows from Corollary 3 considering that, for an instance of SNSAT $\mathscr{I}$, with $|\mathscr{I}| = n$, $\mathcal{K}_{\mathscr{I}}$ and $\psi_n \in \mathscr{F}_{\mathscr{I}}$ have a size polynomial in n and in the length of the formulas of $\mathscr{I}$. Moreover, their structures are repetitive, therefore they can be built by using logarithmic working space. □

We can prove the same complexity result for the symmetric fragment $\overline{\mathsf{A}}\mathsf{E}$, just by transposing the edges of $\mathcal{K}_{\mathscr{I}}$, and by replacing $[B]$ with $[E]$ and $\langle \mathsf{A} \rangle$ with $\langle \overline{\mathsf{A}} \rangle$ in the definition of ψ_n.

We summarize all the $\mathbf{P^{NP}}$-completeness results achieved in the following statement.

Corollary 5. *The model checking problem for* AB*,* $\overline{\mathsf{A}}\mathsf{E}$*,* $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$*, and* $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}$ *formulas over finite Kripke structures is* $\mathbf{P^{NP}}$*-complete.*

We conclude the paper by providing a complexity upper and lower bound for $\overline{\mathsf{A}}\mathsf{B}$ and the symmetric fragment AE. A $\mathbf{P^{NP}}[O(\log^2 n)]$ model checking algorithm for $\overline{\mathsf{A}}\mathsf{B}$ formulas can be obtained by a suitable adaptation of the one for $\mathsf{A}\overline{\mathsf{A}}$ we devised in [18]. Due to the lack of space, an outline of the construction is reported in Appendix A, while stating here only the result. As for the hardness, we can observe that the $\mathbf{P^{NP}}[O(\log n)]$-hardness of $\overline{\mathsf{A}}$ and A, proved in [18], immediately propagates to $\overline{\mathsf{A}}\mathsf{B}$ and AE, respectively.

Theorem 3. *The model checking problem for* $\overline{\mathsf{A}}\mathsf{B}$ *and* AE *formulas over finite Kripke structures is in* $\mathbf{P^{NP}}[O(\log^2 n)]$ *and it is hard for* $\mathbf{P^{NP}}[O(\log n)]$.

5 Conclusions and future work

In this paper, we have proved that the model checking problem for the HS fragments AB, $\overline{\mathsf{A}}\mathsf{E}$, $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$, and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}$ is $\mathbf{P^{NP}}$-complete. They are thus somehow "halfway" between $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$, $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{E}}$, and $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{E}}$, which are **PSPACE**-complete [2, 16, 17], and Prop, B, and E, which are **co-NP**-complete [2, 16], and A, $\overline{\mathsf{A}}$, and $\mathsf{A}\overline{\mathsf{A}}$, whose model checking is in $\mathbf{P^{NP}}[O(\log^2 n)]$ [18]. In addition, we have shown that model checking for the HS fragments $\overline{\mathsf{A}}\mathsf{B}$ and AE has a lower complexity (it is in between $\mathbf{P^{NP}}[O(\log n)]$ and $\mathbf{P^{NP}}[O(\log^2 n)]$).

As for future work, we are looking for possible improvements to known complexity results for (full) HS model checking. We know that it is **EXPSPACE**-hard (we proved **EXPSPACE**-hardness of its fragment BE), while the only available decision procedure is nonelementary. We also started a comparative study of the expressiveness of HS fragments (with the current semantics as well as with some variants of it, which limit past/future branching) and of standard temporal logics, such as LTL, CTL, and CTL*.

Acknowledgments. The work by Alberto Molinari, Angelo Montanari, and Pietro Sala has been supported by the GNCS project *Logic, Automata, and Games for Auto-Adaptive Systems*.

References

[1] J. F. Allen (1983): *Maintaining Knowledge about Temporal Intervals*. Communications of the ACM 26(11), pp. 832–843.

[2] L. Bozzelli, A. Molinari, A. Montanari, A. Peron & P. Sala (2016): *Interval Temporal Logic Model Checking: the Border Between Good and Bad HS Fragments*. In: *IJCAR*, LNAI 9706, Springer, pp. 389–405, doi:10.1007/978-3-319-40229-1_27.

[3] D. Bresolin, D. Della Monica, V. Goranko, A. Montanari & G. Sciavicco (2014): *The dark side of interval temporal logic: marking the undecidability border*. Annals of Mathematics and Artificial Intelligence 71(1-3), pp. 41–83.

[4] D. Bresolin, V. Goranko, A. Montanari & P. Sala (2010): *Tableau-based decision procedures for the logics of subinterval structures over dense orderings*. Journal of Logic and Computation 20(1), pp. 133–166.

[5] D. Bresolin, V. Goranko, A. Montanari & G. Sciavicco (2009): *Propositional interval neighborhood logics: Expressiveness, decidability, and undecidable extensions*. Annals of Pure and Applied Logic 161(3), pp. 289–304.

[6] E. A. Emerson & J. Y. Halpern (1986): *"Sometimes" and "not never" revisited: on branching versus linear time temporal logic*. Journal of the ACM 33(1), pp. 151–178.

[7] F. Giunchiglia & P. Traverso (1999): *Planning as Model Checking*. In: *ECP*, LNCS 1809, Springer, pp. 1–20.

[8] G. Gottlob (1995): *NP Trees and Carnap's Modal Logic*. Journal of the ACM 42(2), pp. 421–457.

[9] J. Y. Halpern & Y. Shoham (1991): *A Propositional Modal Logic of Time Intervals*. Journal of the ACM 38(4), pp. 935–962.

[10] F. Laroussinie, N. Markey & P. Schnoebelen (2001): *Model Checking CTL$^+$ and FCTL is Hard*. In: *FOSSACS*, pp. 318–331, doi:10.1007/3-540-45315-6_21.

[11] A. Lomuscio & J. Michaliszyn (2013): *An Epistemic Halpern-Shoham Logic*. In: *IJCAI*, pp. 1010–1016.

[12] A. Lomuscio & J. Michaliszyn (2014): *Decidability of model checking multi-agent systems against a class of EHS specifications*. In: *ECAI*, pp. 543–548.

[13] A. Lomuscio & J. Michaliszyn (2016): *Model Checking Multi-Agent Systems against Epistemic HS Specifications with Regular Expressions*. In: *KR*, pp. 298–308.

[14] A. Lomuscio & F. Raimondi (2006): *MCMAS: A Model Checker for Multi-agent Systems*. In: *TACAS*, LNCS 3920, Springer, pp. 450–454.

[15] A. Molinari, A. Montanari, A. Murano, G. Perelli & A. Peron (2016): *Checking interval properties of computations*. Acta Informatica, doi:10.1007/s00236-015-0250-1. Accepted for publication.

[16] A. Molinari, A. Montanari & A. Peron (2015): *Complexity of ITL model checking: some well-behaved fragments of the interval logic HS*. In: *TIME*, pp. 90–100, doi:10.1109/TIME.2015.12.

[17] A. Molinari, A. Montanari & A. Peron (2015): *A Model Checking Procedure for Interval Temporal Logics based on Track Representatives*. In: *CSL*, pp. 193–210, doi:10.4230/LIPIcs.CSL.2015.193.

[18] A. Molinari, A. Montanari, A. Peron & P. Sala (2016): *Model Checking Well-Behaved Fragments of HS: the (Almost) Final Picture*. In: *KR*, pp. 473–483.

[19] B. Moszkowski (1983): *Reasoning About Digital Circuits*. Ph.D. thesis, Stanford University, Stanford, CA.

[20] A. Pnueli (1977): *The temporal logic of programs*. In: *FOCS*, IEEE Computer Society, pp. 46–57.

[21] I. Pratt-Hartmann (2005): *Temporal prepositions and their logic*. Artificial Intelligence 166(1-2), pp. 1–36.

[22] P. Roeper (1980): *Intervals and Tenses*. Journal of Philosophical Logic 9, pp. 451–469.

[23] P. Schnoebelen (2003): *Oracle Circuits for Branching-Time Model Checking*. In: *ICALP*, LNCS 2719, Springer, pp. 790–801.

[24] C. Zhou & M. R. Hansen (2004): *Duration Calculus - A Formal Approach to Real-Time Systems*. Monographs in Theoretical Computer Science. An EATCS Series, Springer.

Algorithm 3 $\texttt{Oracle}(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\overline{\mathsf{A}}})$

```
1: ρ̃ ← A_track(K, v, |W| · (2|ψ| + 1)^2, DIRECTION)   ◁ a track of K from/to v of length ≤ |W| · (2|ψ| + 1)^2
2: for all ⟨Ā⟩ φ ∈ ModSubf_{AĀ}(ψ) do
3:     for i = 1, ···, |ρ̃| do
4:         T[⟨Ā⟩ φ, i] ← V_Ā(φ, fst(ρ̃))
5: for all subformulas φ of ψ, not contained in (or equal to) AĀ-modal subformulas of ψ, by increasing length do
6:     if φ = p, for p ∈ AP then
7:         T[p, 1] ← p ∈ μ(fst(ρ̃))
8:         for i = 2, ···, |ρ̃| do
9:             T[p, i] ← T[p, i − 1] and p ∈ μ(ρ̃(i))
10:    else if φ = ¬φ1 then
11:        for i = 1, ···, |ρ̃| do
12:            T[φ, i] ← not T[φ1, i]
13:    else if φ = φ1 ∨ φ2 then
14:        for i = 1, ···, |ρ̃| do
15:            T[φ, i] ← T[φ1, i] or T[φ2, i]
16:    else if φ = ⟨B⟩ φ1 then
17:        T[φ, 1] ← ⊥
18:        for i = 2, ···, |ρ̃| do
19:            T[φ, i] ← T[φ, i − 1] or T[φ1, i − 1]
20: return T[ψ, |ρ̃|]
```

A A $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ model checking algorithm for $\overline{\mathsf{A}}\mathsf{B}$ formulas

In this section, we outline a $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ model checking algorithm for $\overline{\mathsf{A}}\mathsf{B}$ formulas, which suitably adapts the ideas and techniques used in [18] to prove a similar result for the fragment $\mathsf{A}\overline{\mathsf{A}}$.

Since we are considering a sub-fragment of $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}$, if we restrict Algorithm 2 to Algorithm 3 by removing the case for the $\langle \mathrm{A} \rangle$ operator, we get a procedure for which Lemma 1 holds. Since Algorithm 3 is in **NP**, there must exist a reduction to SAT, i.e., given an instance $(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\overline{\mathsf{A}}})$ for `Oracle`, there is a Boolean formula $\Psi_{(\mathcal{K},\psi,v,\text{DIRECTION},V_{\overline{\mathsf{A}}})}$, which depends on $(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\overline{\mathsf{A}}})$, that is satisfiable if and only if $\texttt{Oracle}(\mathcal{K}, \psi, v, \text{DIRECTION}, V_{\overline{\mathsf{A}}})$ admits a successful computation for the given input. By Lemma 1, this is the case iff:

- there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{fst}(\rho) = v$ and $\mathcal{K}, \rho \models \psi$, in the case DIRECTION is FORWARD;
- there exists $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{lst}(\rho) = v$ and $\mathcal{K}, \rho \models \psi$, in the case DIRECTION is BACKWARD,

provided that for each $\langle \overline{\mathrm{A}} \rangle \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ and $v' \in W$, $V_{\overline{\mathsf{A}}}(\phi, v') = \top$ iff there exists $\rho' \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{lst}(\rho') = v'$ and $\mathcal{K}, \rho' \models \phi$.

We now sketch the construction of the formula $\Psi_{(\mathcal{K},\psi,v,\text{DIRECTION},V_{\overline{\mathsf{A}}})}$, which is built over the union of the following sets of Boolean variables:

- for any $i = 1, \cdots, |W|$ and $t = 1, \cdots, |W| \cdot (2|\psi| + 1)^2$, there exists a variable v_i^t such that v_i^t is assigned the truth value $\top$ iff the t-th state of a track ρ we are considering is $w_i \in W$, that is, $\rho(t) = w_i$;
- for each entry $T[\varphi, i]$ of T (the Boolean table of `Oracle`) there is a variable $x_{\varphi,i}$ such that $x_{\varphi,i}$ is assigned the truth value $\top$ iff $T[\varphi, i] = \top$.

The formula $\Psi_{(\mathcal{K},\psi,v,\text{DIRECTION},V_{\overline{\mathsf{A}}})}$ is the conjunction of the following formulas:

- a Boolean formula φ_{track} that, when satisfied, gives the encoding of a track ρ, whose length is at most $|W| \cdot (2|\psi| + 1)^2$, obtained by unravelling $\mathcal{K}$ starting from/up to v (depending on DIRECTION), and, for each $1 \le i \le |\tilde{\rho}|$, determines which are the proposition letters that hold on $\rho(1, i)$;
- for any $\langle \overline{\mathrm{A}} \rangle \phi \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$, we introduce the conjunct $\left(x_{\langle \overline{\mathrm{A}} \rangle \phi, 1} \leftrightarrow \bigvee_{j=1}^{|W|} (v_j^1 \wedge V_{\overline{\mathsf{A}}}(\phi, w_j)) \right)$, where $w_j \in W$, and the conjunct $\bigwedge_t (x_{\langle \overline{\mathrm{A}} \rangle \phi, 1} \leftrightarrow x_{\langle \overline{\mathrm{A}} \rangle \phi, t})$;
- for all subformulas φ of ψ, not contained in (or equal to) $\mathsf{A}\overline{\mathsf{A}}$-modal subformulas of ψ:
  - if $\varphi = \neg\varphi_1$, we introduce the conjuncts $(x_{\varphi,i} \leftrightarrow \neg x_{\varphi_1,i})$, for all $1 \le i \le |\rho|$;
  - $\varphi = \varphi_1 \vee \varphi_2$, we introduce the conjuncts $(x_{\varphi,i} \leftrightarrow x_{\varphi_1,i} \vee x_{\varphi_2,i})$, for all $1 \le i \le |\rho|$;
  - $\varphi = \langle \mathrm{B} \rangle \varphi_1$, we create the conjuncts $(x_{\varphi,i} \leftrightarrow \bigvee_{1 \le j < i} x_{\varphi_1,j})$, for all $1 \le i \le |\rho|$;

Notice that from the above construction, $\Psi_{(\mathcal{K},\psi,v,\text{DIRECTION},V_{\overline{\mathsf{A}}})}$ is polynomial in the size of $|\mathcal{K}|$ and of $|\psi|$.

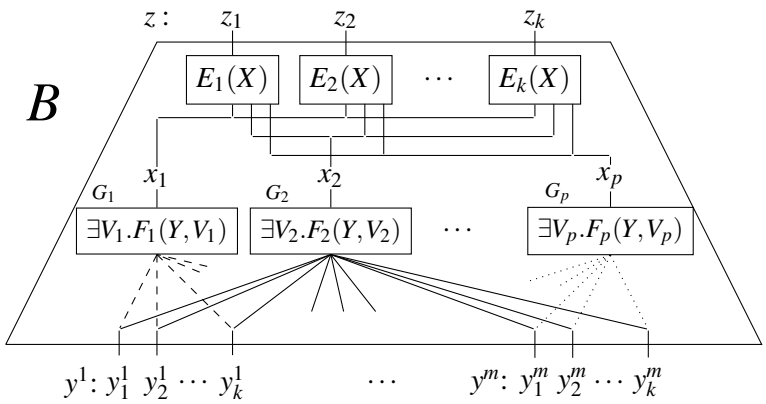


Figure 6: The general form of a block.

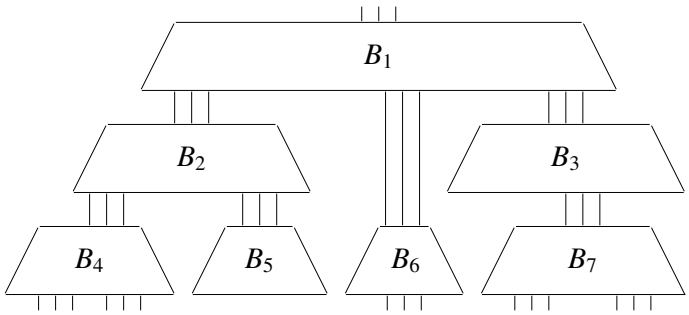


Figure 7: A tree of blocks (B_5 has degree $m = 0$).

Now following the approach proposed in [18], we reduce the model checking problem for $\overline{\mathsf{A}}\mathsf{B}$ to the problem TB(SAT)$_{1\times M}$, which is the first complete problem for the class $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$ that has been proposed in the literature [23].

For the purpose of self-containment, we briefly recall here the problem TB(SAT)$_{1\times M}$. As a preliminarily step, we need to introduce the notion of *block*. A block (see Figure 6) is a circuit B whose input lines are organized in m bit vectors y^1, ..., y^m, each one with k entries: $y^i = (y^i_1, \cdots, y^i_k)$. We denote $Y = \{y^j_s \mid j = 1, \cdots, m,\ s = 1, \cdots, k\}$. The values m and k are respectively called the *degree* and the *width* of B. The input lines are connected to p internal gates G_1, ..., G_p. Each gate G_i queries a SAT oracle to decide whether the associated Boolean formula $F_i(Y, V_i)$ is satisfiable for some truth assignment to the variables of V_i, where V_i is a set of private variables of F_i ($V_j \cap V_i = \emptyset$, for $j \neq i$, and $V_i \cap Y = \emptyset$). The output of G_i is conveyed by x_i, which evaluates to $\top$ *if and only if* $F_i(Y, V_i)$ is satisfiable. Finally, k classical circuits (without oracles) E_1, ..., E_k compute, from $X = \{x_1, \cdots, x_p\}$, the final k bits, output of the block B, conveyed by the output lines z_1, ..., z_k (thus the number of output lines equals the width of B). The size of B is the total number of gates, as usual, plus the sizes of all the formulas $F_i(Y, V_i)$.

Blocks of *the same width* can be combined together to form a tree-structured complex circuit, called *tree of blocks* (see Figure 7 for an example). Every block B_i in the tree has a level: blocks which are leaves of the tree are at level 1; a block B_i whose inputs depend on (at least) a block B_j at level $d-1$ and possibly on other blocks at some levels $\leq d-1$, is at level d. In Figure 7, B_4, B_5, B_6, B_7 are at level 1, B_2 and B_3 at level 2, and B_1 at level 3.

TB(SAT) is the problem of deciding whether a specific output z_i (of the root) of a tree of blocks T is $\top$ or $\bot$, given the values for the inputs (of the leaf blocks) of T. As proved in [23], TB(SAT) is $\mathbf{P}^{\mathbf{NP}}$-

complete. The problem $\mathrm{TB(SAT)}_{1\times M}$ is the problem TB(SAT) with a restriction: the Boolean formulas (SAT queries) of each block B of the tree of blocks must have the form $\exists \ell_1, \cdots, \ell_m \exists V_i' F_i(y^1_{\ell_1}, \cdots, y^m_{\ell_m}, \ell_1, \cdots, \ell_m, V_i')$, with $\ell_j \in \{1, \cdots, k\}$, where $1 \leq j \leq m$, and m and k are respectively the degree and the width of B. This amounts to say that F_i can use *only one bit from each input vector* of B (no matter which), hence "$1 \times M$". As a matter of fact, the existential quantification over the indexes $\ell_1, \cdots, \ell_m$ is an abuse of notation borrowed from [23]; however, $\exists \ell_j \in \{1, \cdots, k\}$ is just a shorthand for k bits (belonging to the set of private variables) "$\ell_j = 1$", ..., "$\ell_j = k$", among which exactly one is 1. In [23], the author proves that $\mathrm{TB(SAT)}_{1\times M}$ is $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$-complete.

Given an $\overline{\mathsf{A}}\mathsf{B}$ formula ψ (w.l.o.g., we assume that ψ contains only *existential* modalities and disjunctions) to be checked over a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_1)$, where $W = \{w_1, \cdots, w_{|W|}\}$, we consider its negation $\neg\psi$ and we build a tree of blocks $T_{\mathcal{K},\neg\psi}$ for $\neg\psi$. Each block of $T_{\mathcal{K},\neg\psi}$ has a type, FORWARD or BACKWARD, and it is associated with a subformula of $\neg\psi$. The root block B_{root} has type FORWARD and it is associated with $\neg\psi$. All the other blocks have type BACKWARD. Each block B has an output line z_i for each state $w_i \in W$, thus the *width* of all blocks is $k = |W|$.

Starting from B_{root}, $T_{\mathcal{K},\neg\psi}$ is built recursively as follows. If a block B is associated with a formula φ, then, for each $\langle\overline{\mathsf{A}}\rangle\,\xi' \in \mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\varphi)$, we create a BACKWARD child B' of B associated with ξ'. Notice that B has thus *degree* $m = |\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\varphi)|$. Then, the procedure is applied recursively to all the generated children of B, terminating when $\mathrm{ModSubf}_{\mathsf{A}\overline{\mathsf{A}}}(\varphi) = \emptyset$.

This procedure allows us to determine the general structure of $T_{\mathcal{K},\neg\psi}$. We now need to describe the internal structure of a (generic) block B in $T_{\mathcal{K},\neg\psi}$ for a formula φ. We refer again to Figure 6. First of all, each output line z_i is directly linked to the output x_i of the oracle gate G_i, avoiding circuits $E_1, \ldots, E_k$, which can be omitted from B. The formula of G_i is precisely $\Psi_{(\mathcal{K},\varphi,w_i,\mathrm{DIRECTION},V_{\overline{\mathsf{A}}})}$ as described above, where $V_{\overline{\mathsf{A}}}(\xi, w_j)$ is replaced by y^{ξ}_j—which denotes the input of B linked to the j-th output of the child B' associated with the formula ξ— and DIRECTION equals the type of B.

The basic fact used to prove the correctness of the construction, stated by the following Theorem 4, is the result proved in [2] (Theorem 10), already exploited in the previous sections: if some $\rho \in \mathrm{Trk}_{\mathcal{K}}$ is such that $\mathcal{K}, \rho \models \psi$, where ψ is an $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}$ formula, then there exists $\tilde{\rho} \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathcal{K}, \tilde{\rho} \models \psi$, $\mathrm{fst}(\tilde{\rho}) = \mathrm{fst}(\rho)$, $\mathrm{lst}(\tilde{\rho}) = \mathrm{lst}(\rho)$, and $|\tilde{\rho}| \leq |W| \cdot (2|\psi| + 1)^2$.

The proof of Theorem 4 is analogous to that given in [18], and it is therefore omitted.

Theorem 4. *Given an $\overline{\mathsf{A}}\mathsf{B}$ formula ψ and a finite Kripke structure $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_1)$, for each block B of $T_{\mathcal{K},\neg\psi}$, if B is associated with an $\overline{\mathsf{A}}\mathsf{B}$ formula φ, then*

- *if B is* FORWARD *(i.e., it is the root B_{root}), $\forall i \in \{1, \cdots, |W|\}$, $B(z_i) = \top$ iff there exists a track $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{fst}(\rho) = w_i$ and $\mathcal{K}, \rho \models \varphi$;*
- *if B is* BACKWARD*, $\forall i \in \{1, \cdots, |W|\}$, $B(z_i) = \top$ iff there exists a track $\rho \in \mathrm{Trk}_{\mathcal{K}}$ such that $\mathrm{lst}(\rho) = w_i$ and $\mathcal{K}, \rho \models \varphi$;*

The next two corollaries immediately follow.

Corollary 6. *Let ψ be an $\overline{\mathsf{A}}\mathsf{B}$ formula, $\mathcal{K} = (\mathcal{AP}, W, \delta, \mu, w_1)$ be a finite Kripke structure, and B_{root} be the root block of $T_{\mathcal{K},\neg\psi}$. It holds that $B_{root}(z_1) = \bot \iff \mathcal{K} \models \psi$.*

Corollary 7. *The model checking problem for formulas of the fragment $\overline{\mathsf{A}}\mathsf{B}$ over finite Kripke structures is in $\mathbf{P}^{\mathbf{NP}[O(\log^2 n)]}$.*

Proof. The result follows from Corollary 6 and the fact that the instance of $\mathrm{TB(SAT)}_{1\times M}$ generated from an $\overline{\mathsf{A}}\mathsf{B}$ formula ψ and a Kripke structure $\mathcal{K}$ is polynomial in $|\psi|$ and $|\mathcal{K}|$. □

The same complexity result holds for the symmetric fragment AE.