University of Udine

Department of Mathematics and Computer Science

PREPRINT

On the Complexity of Model Checking for Syntactically Maximal Fragments of the Interval Temporal Logic HS with Regular Expressions

Laura Bozzelli, Alberto Molinari, Angelo Montanari, Adriano Peron

Preprint nr.: 3/2017

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

On the Complexity of Model Checking for Syntactically Maximal Fragments of the Interval Temporal Logic HS with Regular Expressions

Laura Bozzelli Adriano Peron
University of Napoli "Federico II", Napoli, Italy
lr.bozzelli@gmail.com adrperon@unina.it
Alberto Molinari Angelo Montanari
University of Udine, Udine, Italy
molinari.alberto@gmail.com angelo.montanari@uniud.it

In this paper, we investigate the model checking (MC) problem for Halpern and Shoham's interval temporal logic HS. In the last years, interval temporal logic MC has received an increasing attention as a viable alternative to the traditional (point-based) temporal logic MC, which can be recovered as a special case. Most results have been obtained under the homogeneity assumption, that constrains a proposition letter to hold over an interval if and only if it holds over each component state. Recently, Lomuscio and Michaliszyn proposed a way to relax such an assumption by exploiting regular expressions to define the behaviour of proposition letters over intervals in terms of their component states. When homogeneity is assumed, the exact complexity of MC is a difficult open question for full HS and for its two syntactically maximal fragments $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}E\overline{B}\overline{E}}$. In this paper, we provide an asymptotically optimal bound to the complexity of these two fragments under the more expressive semantic variant based on regular expressions by showing that their MC problem is $\mathbf{AEXP_{pol}}$-complete, where $\mathbf{AEXP_{pol}}$ denotes the complexity class of problems decided by exponential-time bounded alternating Turing Machines making a polynomially bounded number of alternations.

1 Introduction

Model checking (MC), which allows one to automatically check whether a model of a given system satisfies a desired behavioural property, is commonly recognized as one of the most effective techniques in automatic system verification. Besides in formal verification, it has been successfully used also in more general contexts (e.g., databases, planning, configuration systems, multi-agent systems [12, 11, 18]). The actual possibility of exploiting MC relies on a good balance of expressiveness and complexity in the choice of the system model and of the language for specifying behavioural properties. Systems are usually modeled as finite state-transition graphs (finite Kripke structures), while properties are commonly expressed by formulas of point-based temporal logics, such as LTL, CTL, and CTL* [25, 9].

In this paper, we focus on MC with interval temporal logic (ITL) as the specification language. ITL features intervals, instead of points, as its primitive temporal entities [13, 24, 28]. ITL allows one to deal with relevant temporal properties, such as actions with duration, accomplishments, and temporal aggregations, which are inherently "interval-based" and cannot be properly expressed by point-based temporal logics. ITL has been fruitfully applied in various areas of computer science, including formal verification, computational linguistics, planning, and multi-agent systems [24, 26, 15].

Among ITLs, the landmark is Halpern and Shoham's modal logic of time intervals HS [13], which features one modality for each of the 13 ordering relations between pairs of intervals (the so-called

Submitted to:
GandALF 2017

Allen's relations [1]), apart from equality. (Actually, the three Allen's modalities *meets* A, *started-by* B, and *finished-by* E, together with the corresponding inverse modalities $\overline{\mathsf{A}}$, $\overline{\mathsf{B}}$, and $\overline{\mathsf{E}}$, suffice for expressing the entire set of relations.) The satisfiability problem for HS is undecidable over all relevant classes of linear orders [13], and most of its fragments (with meaningful exceptions) are undecidable as well [7, 19].

The MC problem for HS and its fragments consists in the verification of the correctness of the behaviour of a given system with respect to interval properties expressed in HS. Each finite computation path is interpreted as an interval, and its labelling is defined on the basis of the labelling of the states occurring in the path. Most results have been obtained by imposing suitable restrictions on proposition letters labeling intervals: either a proposition letter can be constrained to hold over an interval if and only if it holds over each component state (*homogeneity assumption* [27]), or interval labeling can be defined in terms of the labeling of interval endpoints. An almost complete picture of the MC problem for full HS and its fragments has been recently depicted with the contribution of many works by Molinari et al. [20, 21, 22, 4, 6, 20, 23], which all consider MC over finite Kripke structures for HS endowed with a state-based semantics (allowing branching both in the past and in the future) enforcing the homogeneity assumption. The summary of these results is depicted in the second column of Table 1 (the first column reports the fragments of HS denoted by the list of the featured modalities). The complexity classes shown in red represent new (upper/lower) bounds to the complexity of the problem deriving from the results of this paper, while the other classes (in black) are known bounds. Only few, hard issues are left open in this picture, mostly regarding the precise complexity of the full logic and its maximal fragments. A comparison of different semantics solutions (i.e., state-based semantics, linear semantics and computation-tree-based semantics), together with an expressiveness comparison with standard point-based temporal logics LTL, CTL, and CTL* can be found in [5].

Different assumptions have been done by Lomuscio and Michaliszyn in [15, 16] for some HS fragments extended with epistemic operators (*KC*). They assume a computation-tree-based semantics (formulae are interpreted over the unwinding of the Kripke structure) and interval labeling takes into account only the endpoints of intervals. The different semantic assumptions prevent any immediate comparison with respect to the former approach. The decidability status of MC for full epistemic HS is still unknown. (A summary of the results by Lomuscio and Michaliszyn is depicted in the last column of Table 1.)

The first meaningful attempt to relax the homogeneity assumption can be found in [17], where Lomuscio and Michaliszyn propose to use regular expressions to define the labeling of proposition letters over intervals in terms of the component states. Note that the homogeneity assumption can be trivially encoded by regular expressions. In that work, the authors prove the decidability of MC with regular expressions for some very restricted fragments of epistemic HS, giving some rough upper bounds to its computational complexity. A deeper insight into the problem of MC for HS with regular expressions can be found in [3] where, under the assumption of a state-based semantics, it is proved that MC with regular expressions for full HS is decidable, and that a large class of HS fragments can be checked in polynomial working space (see the third column of Table 1).

In this paper, we study the problems of MC for the two (syntactically) maximal (symmetric) fragments $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}}\overline{\mathsf{E}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{B}}\overline{\mathsf{E}}$ with regular expressions, which are not covered by [3], proving that the complexity of both problems is $\mathbf{AEXP_{pol}}$-complete. $\mathbf{AEXP_{pol}}$ denotes the complexity class of problems decided by exponential-time bounded alternating Turing Machines with a polynomially bounded number of alternations. Such a class captures the precise complexity of some relevant problems [2, 10] (e.g., the first-order theory of real addition with order [10]). First, we note that settling the exact complexity of these fragments under the homogeneity assumption (which can be encoded by regular expressions) is a difficult open question [22]. Moreover, considering that $\mathbf{AEXP_{pol}} \subseteq \mathbf{EXPSPACE}$ and that HS under homogeneity is subsumed by HS with regular expressions, the results proved in this paper improve the

Table 1: Complexity of MC for HS and its fragments (†local MC).

	Homogeneity	Regular expressions	[15] – [17]
Full HS, BE	non-elem. **EXPSPACE**-hard	non-elem. **EXPSPACE**-hard	BE+$KC^\dagger$: **PSPACE** BE†: **P**
$A\overline{A}B\overline{B}E, A\overline{A}E\overline{B}E$	$\in$ **EXPSPACE** [$\in$ **AEXP$_{pol}$**] **PSPACE**-hard	non-elem **PSPACE**-hard [**AEXP$_{pol}$**-complete]	
$A\overline{A}B\overline{E}$	**PSPACE**-complete	non-elem [$\in$ **AEXP$_{pol}$**] **PSPACE**-hard	
$A\overline{A}B\overline{B}, B\overline{B}, \overline{B}$, $A\overline{A}E\overline{E}, E\overline{E}, \overline{E}$	**PSPACE**-complete	**PSPACE**-complete	$A\overline{B}$+KC: non-elem.
$A\overline{A}B, A\overline{A}E, AB, \overline{A}E$	**P^{NP}**-complete	**PSPACE**-complete	
$A\overline{A}, \overline{A}B, AE, A, \overline{A}$	$\in$ **P$^{NP[O(\log^2 n)]}$** **P$^{NP[O(\log n)]}$**-hard	**PSPACE**-complete	
Prop, B, E	**co-NP**-complete	**PSPACE**-complete	

upper bounds for the fragments $A\overline{A}B\overline{B}E$ and $A\overline{A}E\overline{B}E$ given in [22].

Such results are obtained by preliminarily establishing an exponential-size model-trace property: for each interval, it is possible to find an interval of bounded exponential length that is indistinguishable with respect to the fulfillment of $A\overline{A}B\overline{B}E$ formulas (resp., $A\overline{A}E\overline{B}E$). Such a property allows us to devise a MC procedure belonging to the class **AEXP$_{pol}$**. Finally, the matching lower bounds are obtained by polynomial-time reductions from the so-called alternating multi-tiling problem, and they already hold for the fragments $B\overline{E}$ and $E\overline{B}$ of $A\overline{A}B\overline{B}E$ and $A\overline{A}E\overline{B}E$, respectively.

The paper is structured as follows. In Section 2, we introduce the logic HS and provide some background knowledge. In Section 3 we prove the exponential-size model-trace property for $A\overline{A}B\overline{B}E$. In Section 4, we provide an **AEXP$_{pol}$** upper bound to the MC problem for $A\overline{A}B\overline{B}E$. Finally, in Section 5, we prove the hardness of the fragment $B\overline{E}$. Similar proofs can be given for establishing the **AEXP$_{pol}$**-completeness of $A\overline{A}E\overline{B}E$, and the **AEXP$_{pol}$**-hardness of $E\overline{B}$. Due to space constraints, most of the proofs are reported in the appendix.

2 Preliminaries

We introduce some preliminary notation. Let $\mathbb{N}$ be the set of natural numbers. For all $i, j \in \mathbb{N}$, with $i \leq j$, $[i, j]$ denotes the set of natural numbers h such that $i \leq h \leq j$. Let Σ be an alphabet and w be a finite word over Σ. We denote by $|w|$ the length of w. By ε we denote the empty word. For all $1 \leq i \leq j \leq |w|$, $w(i)$ denotes the i-th letter of w, while $w(i, j)$ denotes the finite subword of w given by $w(i)w(i+1)\cdots w(j)$. For $|w| = n$, we define $\mathrm{fst}(w) = w(1)$ and $\mathrm{lst}(w) = w(n)$. The sets of all proper prefixes and suffixes of w are $\mathrm{Pref}(w) = \{w(1,i) \mid 1 \leq i \leq n-1\}$ and $\mathrm{Suff}(w) = \{w(i,n) \mid 2 \leq i \leq n\}$, respectively. The concatenation of two words w and w' is denoted as usual by $w \cdot w'$. Moreover, if $\mathrm{lst}(w) = \mathrm{fst}(w')$, $w \star w'$ represents $w(1, n-1) \cdot w'$, where $n = |w|$ ($\star$-concatenation).

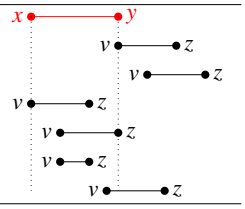
2.1 Kripke structures, regular expressions, and finite automata

Finite state systems are usually modelled as finite Kripke structures. Let $\mathcal{AP}$ be a finite set of proposition letters, which represent predicates decorating the states of the given system.

Definition 1 (Kripke structure)**.** A *Kripke structure* over $\mathcal{AP}$ is a tuple $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$, where S is a set of states, $R \subseteq S \times S$ is a transition relation, $\mu : S \mapsto 2^{\mathcal{AP}}$ is a total labelling function assigning to each

Table 2: Allen's relations and corresponding HS modalities.

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



state s the set of propositions that hold over it, and $s_0 \in S$ is the initial state. $\mathcal{K}$ is said finite if S is finite.

Let $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ be a Kripke structure. A *trace* (or finite path) of $\mathcal{K}$ is a non-empty finite word ρ over S such that $(\rho(i), \rho(i+1)) \in R$ for all $i \in [1, |\rho| - 1]$. A trace is *initial* if it starts from the initial state s_0. A trace ρ induces the finite word $\mu(\rho)$ over $2^{\mathcal{AP}}$ given by $\mu(\rho(1)) \cdots \mu(\rho(n))$ with $n = |\rho|$. We call $\mu(\rho)$ the *labeling sequence induced by* ρ.

Let us recall now the class of regular expressions over finite words. Since we are interested in expressing requirements over the labeling sequences induced by the traces of Kripke structures, which are finite words over $2^{\mathcal{AP}}$, here we consider *propositional-based* regular expressions (RE), where the atomic expressions are propositional formulas over $\mathcal{AP}$ instead of letters over an alphabet. Formally, the set of RE r over $\mathcal{AP}$ is defined as $r ::= \varepsilon \mid \phi \mid r \cup r \mid r \cdot r \mid r^*$, where ϕ is a propositional formula over $\mathcal{AP}$. The size $|r|$ of an RE r is the number of subexpressions of r. An RE r denotes a language $\mathscr{L}(r)$ of finite words over $2^{\mathcal{AP}}$ defined as:

- $\mathscr{L}(\varepsilon) = \{\varepsilon\}$ and $\mathscr{L}(\phi) = \{A \in 2^{\mathcal{AP}} \mid A \text{ satisfies } \phi\}$;
- $\mathscr{L}(r_1 \cup r_2) = \mathscr{L}(r_1) \cup \mathscr{L}(r_2)$, $\mathscr{L}(r_1 \cdot r_2) = \mathscr{L}(r_1) \cdot \mathscr{L}(r_2)$, and $\mathscr{L}(r^*) = (\mathscr{L}(r))^*$.

We also recall the class of nondeterministic finite automata over finite words (NFA). An NFA is a tuple $\mathscr{A} = (\Sigma, Q, Q_0, \Delta, F)$, where Σ is a finite alphabet, Q is a finite set of states, $Q_0 \subseteq Q$ is the set of initial states, $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation, and $F \subseteq Q$ is the set of accepting states. An NFA $\mathscr{A}$ is *complete* if, for all $(q, \sigma) \in Q \times \Sigma$, $(q, \sigma, q') \in \Delta$ for some $q' \in Q$. Given a finite word w over Σ with $|w| = n$ and two states $q, q' \in Q$, a run of $\mathscr{A}$ from q to q' over w is a sequence of states $q_1, \ldots, q_{n+1}$ such that $q_1 = q$, $q_{n+1} = q'$, and for all $i \in [1, n]$, $(q_i, w(i), q_{i+1}) \in \Delta$. The language $\mathscr{L}(\mathscr{A})$ accepted by $\mathscr{A}$ is the set of finite words w on Σ s.t. there is a run from some initial state to some accepting state over w.

Remark 2. Given a RE r, by a standard construction [14], one can compositionally construct a complete NFA $\mathscr{A}_r$ with alphabet $2^{\mathcal{AP}}$, whose number of states is linear in the size of r. We call $\mathscr{A}_r$ the *canonical* NFA associated with r.

2.2 The interval temporal logic HS

A systematic logical study of interval representation and reasoning was proposed by J. Y. Halpern and Y. Shoham, who introduced the interval temporal logic HS [13] featuring one modality for each Allen relation [1], but equality. Table 2 depicts 6 of the 13 Allen's relations, together with the corresponding HS (existential) modalities. The other 7 relations are the 6 inverse relations (given a binary relation $\mathcal{R}$, its inverse $\overline{\mathcal{R}}$ is such that $b\overline{\mathcal{R}}a$ iff $a\mathcal{R}b$) and equality.

Given a finite set $\mathcal{P}_u$ of *uninterpreted interval properties*, the HS language over $\mathcal{P}_u$ consists of propositions from $\mathcal{P}_u$, the Boolean connectives $\neg$ and $\wedge$, and a temporal modality for each of the (non trivial) Allen's relations, i.e., $\langle A \rangle$, $\langle L \rangle$, $\langle B \rangle$, $\langle E \rangle$, $\langle D \rangle$, $\langle O \rangle$, $\langle \overline{A} \rangle$, $\langle \overline{L} \rangle$, $\langle \overline{B} \rangle$, $\langle \overline{E} \rangle$, $\langle \overline{D} \rangle$, and $\langle \overline{O} \rangle$. HS formulas are defined by the grammar $\psi ::= p_u \mid \neg\psi \mid \psi \wedge \psi \mid \langle X \rangle \psi$, where $p_u \in \mathcal{P}_u$ and $X \in \{A, L, B, E, D, O, \overline{A}, \overline{L}, \overline{B}, \overline{E},$

$\overline{D}, \overline{O}\}$. We also exploit the standard logical connectives (disjunction $\vee$ and implication $\rightarrow$) as abbreviations. Furthermore, for any existential modality $\langle X \rangle$, the dual universal modality $[X]\psi$ is defined as $\neg\langle X \rangle \neg\psi$. An HS formula φ is in *positive normal form* (*PNF*) if negation is applied only to atomic formulas in $\mathcal{P}_u$. By using De Morgan's laws and for any existential modality $\langle X \rangle$, the dual universal modality $[X]$, we can convert in linear-time an HS formula φ into an equivalent formula in *PNF*, called the *PNF* of φ. For a formula φ in *PNF*, the *dual* $\widetilde{\varphi}$ of φ is the *PNF* of $\neg\varphi$.

Given any subset of Allen's relations $\{X_1, \ldots, X_n\}$, we denote by $\mathsf{X_1 \cdots X_n}$ the HS fragment closed under Boolean connectives that features (existential and universal) modalities for $X_1, \ldots, X_n$ only.

Without loss of generality, we assume the *non-strict semantics of HS*, which admits intervals consisting of a single point. (All the results we prove in the paper hold for the strict semantics as well.) 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 $\langle \overline{\mathrm{E}} \rangle$ [28]. HS can, thus, be viewed as a multi-modal logic with 4 primitive modalities. However, since we focus on the HS fragments $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ and $\mathsf{A\overline{A}B\overline{B}\overline{E}}$, that do not feature $\langle \mathrm{B} \rangle$ and $\langle \mathrm{E} \rangle$ respectively, we also consider the modalities $\langle \mathrm{A} \rangle$ and $\langle \overline{\mathrm{A}} \rangle$. Note that the modalities $\langle \mathrm{L} \rangle$ and $\langle \mathrm{O} \rangle$ (resp., $\langle \overline{\mathrm{L}} \rangle$ and $\langle \overline{\mathrm{O}} \rangle$) can be expressed in the fragment $\mathsf{A\overline{A}E\overline{B}\overline{E}}$ (resp., $\mathsf{A\overline{A}B\overline{B}\overline{E}}$). As for the semantics of HS, in this paper we follow the approach of [3], where the intervals correspond to the traces of a finite Kripke structure $\mathcal{K}$ (*state-based semantics*) and each abstract interval proposition $p_u \in \mathcal{P}_u$ denotes a regular language of finite words over $2^{\mathcal{AP}}$. More specifically, every abstract interval proposition p_u is a (propositional-based) regular expression over $\mathcal{AP}$. Thus, in the following, for the sake of simplicity, by an HS formula over $\mathcal{AP}$ we mean an HS formula whose abstract interval propositions (or atomic formulas) are RE over $\mathcal{AP}$.

Given a Kripke structure $\mathcal{K} = (\mathcal{AP}, S, E, \mu, s_0)$ over $\mathcal{AP}$, a trace ρ of $\mathcal{K}$, and an HS formula φ over $\mathcal{AP}$, the satisfaction relation $\mathcal{K}, \rho \models \varphi$ is inductively defined as follows (we omit the standard clauses for the Boolean connectives):

$$
\begin{array}{ll}
\mathcal{K}, \rho \models r & \Leftrightarrow \mu(\rho) \in \mathscr{L}(r) \text{ for each RE } r \text{ over } \mathcal{AP}, \\
\mathcal{K}, \rho \models \langle \mathrm{B} \rangle \varphi & \Leftrightarrow \text{there exists } \rho' \in \mathrm{Pref}(\rho) \text{ such that } \mathcal{K}, \rho' \models \varphi, \\
\mathcal{K}, \rho \models \langle \mathrm{E} \rangle \varphi & \Leftrightarrow \text{there exists } \rho' \in \mathrm{Suff}(\rho) \text{ such that } \mathcal{K}, \rho' \models \varphi, \\
\mathcal{K}, \rho \models \langle \overline{\mathrm{B}} \rangle \varphi & \Leftrightarrow \mathcal{K}, \rho' \models \varphi \text{ for some trace } \rho' \text{ such that } \rho \in \mathrm{Pref}(\rho'), \\
\mathcal{K}, \rho \models \langle \overline{\mathrm{E}} \rangle \varphi & \Leftrightarrow \mathcal{K}, \rho' \models \varphi \text{ for some trace } \rho' \text{ such that } \rho \in \mathrm{Suff}(\rho'), \\
\mathcal{K}, \rho \models \langle \mathrm{A} \rangle \varphi & \Leftrightarrow \mathcal{K}, \rho' \models \varphi \text{ for some trace } \rho' \text{ such that } \mathrm{fst}(\rho') = \mathrm{lst}(\rho), \\
\mathcal{K}, \rho \models \langle \overline{\mathrm{A}} \rangle \varphi & \Leftrightarrow \mathcal{K}, \rho' \models \varphi \text{ for some trace } \rho' \text{ such that } \mathrm{lst}(\rho') = \mathrm{fst}(\rho).
\end{array}
$$

$\mathcal{K}$ is a *model* of φ, denoted $\mathcal{K} \models \varphi$, if for all initial traces ρ of $\mathcal{K}$, it holds that $\mathcal{K}, \rho \models \varphi$. The MC problem for HS is checking, for a finite Kripke structure $\mathcal{K}$ and an HS formula φ, whether $\mathcal{K} \models \varphi$ or not.

Note that the state-based semantics provides a branching-time setting both in the past and in the future. In particular, while the modalities for B and E are linear-time (they allow us to select prefixes and suffixes of the current trace), the modalities for A and $\overline{B}$ (resp., $\overline{A}$ and $\overline{E}$) are branching-time in the future (resp., in the past) since they allow us to nondeterministically extend a trace in the future (resp., in the past). As shown in [5], for the considered semantics, the logics HS and CTL* are expressively incomparable already under the homogeneity assumption. However, under the homogeneity assumption, the use of the past branching-time modalities $\overline{A}$ and $\overline{E}$ is necessary for capturing requirements which cannot be expressed in CTL*. For instance, the requirement "*each state reachable from the initial one where p holds has a predecessor where p holds as well*" cannot be expressed in CTL*, but can be easily expressed in the fragment $\mathsf{\overline{A}E}$ [5]. In the more expressive setting based on regular expressions, the future branching-time modalities A and $\overline{B}$ are already sufficient for capturing requirements which cannot be expressed in CTL*, such as the following branching-time bounded response property: "*for each state reachable from the initial one where a request req occurs, there is a computation from this state such*

that the request is followed by a response res within an even number *of steps*". This requirement can be expressed in the fragment $\mathsf{A\overline{B}}$ as follows: $[A](req \rightarrow \langle \overline{\mathrm{B}} \rangle (req \cdot (\top \cdot \top)^* \cdot res))$.

In the rest of the paper, we focus on the fragment $\mathsf{A\overline{A}B\overline{BE}}$. Analogous constructions and results can be symmetrically given for the fragment $\mathsf{A\overline{A}E\overline{BE}}$ as well.

3 Exponential-size model-trace property of $\mathsf{A\overline{A}B\overline{BE}}$

In this section, we show an *exponential-size model-trace property* for $\mathsf{A\overline{A}B\overline{BE}}$, which will be used as the basic step to prove that the MC problem for $\mathsf{A\overline{A}B\overline{BE}}$ belongs to $\mathbf{AEXP_{pol}}$. Fix a Kripke structure $\mathcal{K} = (\mathcal{AP}, S, R, \mu, s_0)$ and a finite set $\mathsf{spec} = \{r_1, \ldots, r_H\}$ of (propositional-based) regular expressions over $\mathcal{AP}$: such a property ensures that for each $h \geq 0$ and trace ρ of $\mathcal{K}$, it is possible to build another trace ρ' of $\mathcal{K}$, of bounded exponential length, which is indistinguishable from ρ with respect to the fulfilment of any $\mathsf{A\overline{A}B\overline{BE}}$ formula φ having atomic formulas in spec and nesting depth of the modality $\langle \mathrm{B} \rangle$ at most h (written $\mathrm{d_B}(\varphi) \leq h$). Formally, $\mathrm{d_B}(\varphi)$ is inductively defined as follows (i) $\mathrm{d_B}(r) = 0$, for any RE r over $\mathcal{AP}$; (ii) $\mathrm{d_B}(\neg\psi) = \mathrm{d_B}(\psi)$; (iii) $\mathrm{d_B}(\psi \wedge \phi) = \max\{\mathrm{d_B}(\psi), \mathrm{d_B}(\phi)\}$; (iv) $\mathrm{d_B}(\langle \mathrm{B} \rangle \psi) = 1 + \mathrm{d_B}(\psi)$; (v) $\mathrm{d_B}(\langle \mathrm{X} \rangle \psi) = \mathrm{d_B}(\psi)$, for $X \in \{A, \overline{A}, \overline{B}, \overline{E}\}$.

In order to state the result, we first introduce the notion of *h-prefix bisimilarity* between a pair of traces ρ and ρ' of $\mathcal{K}$. As proved by Proposition 8 below, h-prefix bisimilarity is a sufficient condition for two traces ρ and ρ' to be indistinguishable with respect to the fulfillment of any $\mathsf{A\overline{A}B\overline{BE}}$ formula φ over spec with $\mathrm{d_B}(\varphi) \leq h$. Then, for a given trace ρ, we show how to determine a subset of positions of ρ, called the *h-prefix sampling* of ρ, that allows us to build another trace ρ' with singly exponential length (both in h and $|\mathsf{spec}|$, where $|\mathsf{spec}|$ is defined as $\sum_{r \in \mathsf{spec}} |r|$) such that ρ and ρ' are h-prefix bisimilar.

For any regular expression r_ℓ in spec with $\ell \in [1, H]$, let $\mathscr{A}_\ell = (2^{\mathcal{AP}}, Q_\ell, Q_\ell^0, \Delta_\ell, F_\ell)$ be the *canonical (complete)* NFA accepting $\mathscr{L}(r_\ell)$ (recall that $|Q_\ell| \leq 2|r_\ell|$). Without loss of generality, we assume that the sets of states of these automata are pairwise disjoint.

The notion of prefix bisimilarity exploits the notion of *summary* of a trace ρ of $\mathcal{K}$, namely a tuple "recording" the initial and final states of ρ, and, for each automaton $\mathscr{A}_\ell$ with $\ell \in [1, H]$, the pairs of states $q, q' \in Q_\ell$ such that some run of $\mathscr{A}_\ell$ over $\mu(\rho)$ takes from q to q'.

Definition 3 (Summary of a trace)**.** Let ρ be a trace of $\mathcal{K}$ with $|\rho| = n$. The summary $\mathscr{S}(\rho)$ of ρ (w.r.t. spec) is the triple $(\rho(1), \Pi, \rho(n))$, where Π is the set of pairs (q, q') such that there is $\ell \in [1, H]$ so that $q, q' \in Q_\ell$ and there is a run of $\mathscr{A}_\ell$ from q to q' over $\mu(\rho)$.

Note that the number of summaries is at most $|S|^2 \cdot 2^{(2|\mathsf{spec}|)^2}$. Evidently, the following holds.

Proposition 4. *Let $h \geq 0$, and ρ and ρ' be two traces of $\mathcal{K}$ such that $\mathscr{S}(\rho) = \mathscr{S}(\rho')$. Then, for all regular expressions $r \in$ spec and traces ρ_L and ρ_R of $\mathcal{K}$ such that $\rho_L \star \rho$ and $\rho \star \rho_R$ are defined, the following conditions hold:*
(1) $\mu(\rho) \in \mathscr{L}(r)$ iff $\mu(\rho') \in \mathscr{L}(r)$; (2) $\mathscr{S}(\rho_L \star \rho) = \mathscr{S}(\rho_L \star \rho')$; (3) $\mathscr{S}(\rho \star \rho_R) = \mathscr{S}(\rho' \star \rho_R)$.

We now introduce the notion of *prefix bisimilarity* between a pair of traces ρ and ρ' of $\mathcal{K}$.

Definition 5 (Prefix bisimilarity)**.** Let $h \geq 0$. Two traces ρ and ρ' of $\mathcal{K}$ are *h-prefix bisimilar* (w.r.t. spec) if the following conditions inductively hold:

- for $h = 0$: $\mathscr{S}(\rho) = \mathscr{S}(\rho')$;
- for $h > 0$: $\mathscr{S}(\rho) = \mathscr{S}(\rho')$ and for each proper prefix ν of ρ (resp., proper prefix ν' of ρ'), there exists a proper prefix ν' of ρ' (resp., proper prefix ν of ρ) such that ν and ν' are $(h-1)$-prefix bisimilar.

Property 6. For all $h \geq 0$, h-prefix bisimilarity is an equivalence relation over traces of $\mathcal{K}$.

The h-prefix bisimilarity of two traces ρ and ρ' is preserved by right (resp., left) $\star$-concatenation with another trace of $\mathcal{K}$ (the proof is reported in Appendix A.1).

Proposition 7. *Let $h \geq 0$, and ρ and ρ' be two h-prefix bisimilar traces of $\mathcal{K}$. Then, for all traces ρ_L and ρ_R of $\mathcal{K}$ such that $\rho_L \star \rho$ and $\rho \star \rho_R$ are defined, the following holds:*
(1) $\rho_L \star \rho$ and $\rho_L \star \rho'$ are h-prefix bisimilar; (2) $\rho \star \rho_R$ and $\rho' \star \rho_R$ are h-prefix bisimilar.

By exploiting Propositions 4 and 7, we can prove that h-prefix bisimilarity preserves the fulfillment of $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formulas over spec having nesting depth of modality $\langle \mathrm{B} \rangle$ at most h.

Proposition 8. *Let $h \geq 0$, and ρ and ρ' be two h-prefix bisimilar traces of $\mathcal{K}$. Then, for each $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula ψ over spec with $\mathrm{d_B}(\psi) \leq h$, $\mathcal{K}, \rho \models \psi$ iff $\mathcal{K}, \rho' \models \psi$.*

Proof. We prove the proposition by a nested induction on the structure of the formula ψ and on the nesting depth $\mathrm{d_B}(\psi)$. For the base case, ψ is a regular expression in spec. Since $\mathscr{S}(\rho) = \mathscr{S}(\rho')$ (ρ and ρ' are h-prefix bisimilar) the result follows by Proposition 4. Now, let us consider the inductive case. The cases where the root modality of ψ is a Boolean connective directly follow by the inductive hypothesis. As for the cases where the root modality is either $\langle \mathrm{A} \rangle$ or $\langle \overline{\mathrm{A}} \rangle$, the result follows from the fact that, being ρ and ρ' h-prefix bisimilar, $\mathrm{fst}(\rho) = \mathrm{fst}(\rho')$ and $\mathrm{lst}(\rho) = \mathrm{lst}(\rho')$. It remains to consider the cases where the root modality is in $\{\langle \mathrm{B} \rangle, \langle \overline{\mathrm{B}} \rangle, \langle \overline{\mathrm{E}} \rangle\}$. We prove the implication $\mathcal{K}, \rho \models \psi \Rightarrow \mathcal{K}, \rho' \models \psi$ (the converse implication being similar). Let $\mathcal{K}, \rho \models \psi$.

- $\psi = \langle \mathrm{B} \rangle \varphi$: since $0 < \mathrm{d_B}(\psi) \leq h$, it holds that $h > 0$. Since $\mathcal{K}, \rho \models \langle \mathrm{B} \rangle \varphi$, there is a proper prefix ν of ρ such that $\mathcal{K}, \nu \models \varphi$. Since ρ and ρ' are h-prefix bisimilar, there is a proper prefix ν' of ρ' such that ν and ν' are $(h-1)$-prefix bisimilar. Being $\mathrm{d_B}(\varphi) \leq h-1$, by the inductive hypothesis we obtain that $\mathcal{K}, \nu' \models \varphi$. Hence, $\mathcal{K}, \rho' \models \langle \mathrm{B} \rangle \varphi$: the thesis follows.
- $\psi = \langle \overline{\mathrm{B}} \rangle \varphi$: since $\mathcal{K}, \rho \models \langle \overline{\mathrm{B}} \rangle \varphi$, there is a trace ρ_R such that $|\rho_R| > 1$ and $\mathcal{K}, \rho \star \rho_R \models \varphi$. By Proposition 7, $\rho \star \rho_R$ and $\rho' \star \rho_R$ are h-prefix bisimilar. By the inductive hypothesis on the structure of the formula, we obtain that $\mathcal{K}, \rho' \star \rho_R \models \varphi$, hence, $\mathcal{K}, \rho' \models \langle \overline{\mathrm{B}} \rangle \varphi$.
- $\psi = \langle \overline{\mathrm{E}} \rangle \varphi$: this case is similar to the previous one. □

In the following, we show how a trace ρ, whose length exceeds a suitable exponential bound—precisely, $(|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$—can be contracted preserving h-prefix bisimilarity and, consequently, fulfillment of formulas φ with $\mathrm{d_B}(\varphi) \leq h$. The basic contraction step of ρ is performed by choosing a subset of ρ positions called *h-prefix sampling* (PS_h). A contraction can be performed whenever there are two positions $\ell < \ell'$ satisfying $\mathscr{S}(\rho(1,\ell)) = \mathscr{S}(\rho(1,\ell'))$ in between two consecutive positions in the linear ordering of PS_h. We prove that by taking the contraction $\rho' = \rho(1,\ell) \cdot \rho(\ell'+1, |\rho|)$, we obtain a trace of $\mathcal{K}$ which is h-prefix bisimilar to ρ. The basic contraction step can then be iterated over ρ' until the length bound is reached.

The notion of h-prefix sampling is inductively defined using the notion of *prefix-skeleton sampling*. For a set I of natural numbers, by "two consecutive elements of I" we refer to a pair of elements $i, j \in I$ such that $i < j$ and $I \cap [i,j] = \{i, j\}$.

Definition 9 (Prefix-skeleton sampling)**.** Let ρ be a trace of $\mathcal{K}$. Given two ρ-positions i and j, with $i \leq j$, the *prefix-skeleton sampling of ρ in the interval $[i,j]$* is the *minimal* set $Pos \supseteq \{i, j\}$ of ρ-positions in the interval $[i,j]$ satisfying:

- for each $k \in [i+1, j-1]$, the minimal position $k' \in [i+1, j-1]$ such that $\mathscr{S}(\rho(1,k')) = \mathscr{S}(\rho(1,k))$ is in Pos.

It immediately follows from Definition 9 that the prefix-skeleton sampling Pos of (any) trace ρ in an interval $[i,j]$ of ρ-positions is such that $|Pos| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2}) + 2$.

Definition 10 (h-prefix sampling)**.** Let $h \geq 0$. The *h-prefix sampling of a trace ρ* of $\mathcal{K}$ is the *minimal* set PS_h of ρ-positions inductively satisfying the following conditions:

- Base case: $h = 0$. $PS_0 = \{1, |\rho|\}$;
- Inductive step: $h > 0$. (i) $PS_h \supseteq PS_{h-1}$ and (ii) for all pairs of consecutive positions i, j in PS_{h-1}, the prefix-skeleton sampling of ρ in the interval $[i, j]$ is in PS_h.

Let $i_1 < \ldots < i_N$ be the ordered sequence of positions in PS_h (note that $i_1 = 1$ and $i_N = |\rho|$). The *h-sampling word of* ρ is the sequence of summaries $\mathscr{S}(\rho(1,i_1)) \cdots \mathscr{S}(\rho(1,i_N))$.

The following upper bound to the cardinality of prefix samplings holds.

Property 11. The h-prefix sampling PS_h of a trace ρ of $\mathcal{K}$ is such that $|PS_h| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+1}$.

The following lemma (proved in Appendix A.2) shows that for two traces, the property of having the same h-sampling word, is a sufficient condition to be h-prefix bisimilar.

Lemma 12. *For $h \geq 0$, two traces having the same h-sampling word are h-prefix bisimilar.*

By exploiting the sufficient condition of Lemma 12, we can finally state the exponential-size model-trace property for $\mathsf{A\overline{A}B\overline{BE}}$. In the proof of Theorem 14 below, it is shown how to derive from any trace ρ of $\mathcal{K}$, an h-prefix bisimilar trace ρ' *induced by* ρ (in the sense that ρ' is obtained by contracting ρ, i.e., by concatenating subtraces of ρ in an ordered way) such that $|\rho'| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$. By Proposition 8, ρ' is indistinguishable from ρ w.r.t. the fulfilment of any $\mathsf{A\overline{A}B\overline{BE}}$ formula φ over the set of atomic formulas in spec such that $\mathrm{d_B}(\varphi) \leq h$. We preliminarily define the notion of *induced trace* (note that if π is induced by ρ, then $\mathrm{fst}(\pi) = \mathrm{fst}(\rho)$, $\mathrm{lst}(\pi) = \mathrm{lst}(\rho)$, $|\pi| \leq |\rho|$, and $|\pi| = |\rho|$ iff $\pi = \rho$).

Definition 13 (Induced trace)**.** Let ρ be a trace of $\mathcal{K}$ of length n. A *trace induced by* ρ is a trace π of $\mathcal{K}$ such that there exists an increasing sequence of ρ-positions $i_1 < \ldots < i_k$, with $i_1 = 1$, $i_k = n$, and $\pi = \rho(i_1) \cdots \rho(i_k)$.

Theorem 14 (Exponential-size model-trace property for $\mathsf{A\overline{A}B\overline{BE}}$)**.** *Let ρ be a trace of $\mathcal{K}$ and $h \geq 0$. Then there exists a trace ρ' induced by ρ, whose length is at most $(|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$, which is h-prefix bisimilar to ρ. In particular, for every $\mathsf{A\overline{A}B\overline{BE}}$ formula ψ with atomic formulas in spec and such that $\mathrm{d_B}(\psi) \leq h$, it holds that $\mathcal{K}, \rho \models \psi$ iff $\mathcal{K}, \rho' \models \psi$.*

Proof. We show that if $|\rho| > (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$, then there exists a trace ρ' induced by ρ such that $|\rho'| < |\rho|$ and ρ and ρ' have the same h-sampling word. Hence, by iterating the reasoning and applying Proposition 8 and Lemma 12, the thesis follows. Assume that $|\rho| > (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$. Let $PS_h : 1 = i_1 < \ldots < i_N = |\rho|$ be the h-prefix sampling of ρ. By Property 11, $|PS_h| \leq (|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+1}$. Since the number of distinct summaries (w.r.t. spec) associated with the prefixes of ρ is at most $|S| \cdot 2^{(2|\mathsf{spec}|)^2}$, there must be two consecutive positions i_j and i_{j+1} in PS_h such that for some $\ell, \ell' \in [i_j + 1, i_{j+1} - 1]$ with $\ell < \ell'$, $\mathscr{S}(\rho(1,\ell)) = \mathscr{S}(\rho(1,\ell'))$. It easily follows that the sequence ρ' given by $\rho' := \rho(1,\ell) \cdot \rho(\ell' + 1, |\rho|)$ is a trace induced by ρ such that $|\rho'| < |\rho|$ and ρ and ρ' have the same h-sampling word. □

4 $\mathbf{AEXP_{pol}}$-membership of MC for $\mathsf{A\overline{A}B\overline{BE}}$

In this section, we exploit the exponential-size model-trace property of $\mathsf{A\overline{A}B\overline{BE}}$ to design a MC algorithm for $\mathsf{A\overline{A}B\overline{BE}}$ belonging to the class $\mathbf{AEXP_{pol}}$, namely, the class of problems solvable by singly exponential-time bounded Alternating Turing Machines (ATMs, for short) with a polynomial-bounded number of alternations. More formally, given an ATM $\mathscr{M}$ (we refer to [8] or Appendix B.1 for standard syntax and semantics of ATMs), $\mathscr{M}$ is *singly exponential-time bounded* if there is an integer constant

check$(\mathcal{K},\varphi)$ [$\mathcal{K}$ *is a finite Kripke structure and* φ *is an* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ *in PNF*]

`existentially choose` an $\mathsf{A\overline{A}}$-labeling *Lab* for $(\mathcal{K},\varphi)$;
`for each` state s and $\psi \in Lab(s)$ `do`
  `case` $\psi = \langle \mathrm{A} \rangle \psi'$ (resp., $\psi = \langle \overline{\mathrm{A}} \rangle \psi'$): `existentially choose` a certificate ρ with
    $\mathrm{fst}(\rho) = s$ (resp., $\mathrm{lst}(\rho) = s$) and `call` $checkTrue_{(\mathcal{K},\varphi,Lab)}(\{(\psi',\rho)\})$;
  `case` $\psi = [A]\psi'$ (resp., $\psi = [\overline{A}]\psi'$): `universally choose` a certificate ρ with
    $\mathrm{fst}(\rho) = s$ (resp., $\mathrm{lst}(\rho) = s$) and `call` $checkTrue_{(\mathcal{K},\varphi,Lab)}(\{(\psi',\rho)\})$;
`end for`
`universally choose` a certificate ρ for $(\mathcal{K},\varphi)$ with $\mathrm{fst}(\rho) = s_0$ (s_0 is the initial state of $\mathcal{K}$)
  and `call` $checkTrue_{(\mathcal{K},\varphi,Lab)}(\{(\varphi,\rho)\})$;

Figure 1: Procedure *check*

$c \geq 1$ such that for each input α, any computation starting on α halts after at most $2^{|\alpha|^c}$ steps. The ATM $\mathscr{M}$ has a *polynomial-bounded number of alternations* if there is an integer constant $c \geq 1$ such that, for all inputs α and computations π starting from α, the number of alternations of existential and universal configurations along π is at most $|\alpha|^c$.

In the sequel, we assume that $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formulas are in *PNF*. For a formula φ, let spec be the set of regular expressions occurring in φ. The size $|\varphi|$ of φ is given by the number of non-atomic subformulas of φ plus $|\mathsf{spec}|$. As another complexity measure of an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ, we consider the standard *alternation depth*, denoted by $\Upsilon(\varphi)$, between the existential $\langle X \rangle$ and universal modalities $[X]$ (and vice versa) occurring in the *PNF* of φ for $X \in \{\overline{B},\overline{E}\}$. Note that the definition does not consider the modalities associated with the Allen's relations in $\{A,\overline{A},B\}$. Moreover, let FMC be the set of pairs $(\mathcal{K},\varphi)$ consisting of a Kripke structure $\mathcal{K}$ and an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ s.t. $\mathcal{K} \models \varphi$. The complexity upper bound is as follows.

Theorem 15. *One can construct a singly exponential-time bounded ATM accepting* FMC *whose number of alternations on an input* $(\mathcal{K},\varphi)$ *is at most* $\Upsilon(\varphi)+2$.

In the rest of the section, we define a procedure (which can be easily translated into an ATM) proving the assertion of Theorem 15. We start with some auxiliary notation. Fix a finite Kripke structure $\mathcal{K}$ with set of states S and an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ in *PNF*. Let $h = \mathrm{d_B}(\varphi)$, and spec be the set of regular expressions occurring in φ.

A *certificate* of $(\mathcal{K},\varphi)$ is a trace ρ of $\mathcal{K}$ whose length is less than $(|S| \cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$ (the bound for the exponential trace property in Theorem 14). A $\overline{B}$*-witness* (resp., $\overline{E}$*-witness*) of a certificate ρ for $(\mathcal{K},\varphi)$, is a certificate ρ' of $(\mathcal{K},\varphi)$ such that ρ' is h-prefix bisimilar to a trace of the form $\rho \star \rho''$ (resp., $\rho'' \star \rho$) for some *certificate* ρ'' of $(\mathcal{K},\varphi)$ with $|\rho''| > 1$. By $\mathsf{SD}(\varphi)$ we denote the set consisting of the subformulas ψ of φ and the *duals* $\widetilde{\psi}$. By the results of Section 3, we deduce the following (see Appendix B.2):

Proposition 16. *Let* $\mathcal{K}$ *be a finite Kripke structure,* φ *be an* $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ *formula in PNF, and* ρ *be a certificate for* $(\mathcal{K},\varphi)$. *The following properties hold:*

1. *for each* $\langle X \rangle \psi \in SD(\varphi)$ *with* $X \in \{\overline{B},\overline{E}\}$, $\mathcal{K},\rho \models \langle X \rangle \psi$ *iff there exists an* X*-witness* ρ' *of* ρ *for* $(\mathcal{K},\varphi)$ *such that* $\mathcal{K},\rho' \models \psi$;
2. *for each trace of the form* $\rho \star \rho'$ *(resp.,* $\rho' \star \rho$*) such that* ρ' *is a certificate for* $(\mathcal{K},\varphi)$, *one can construct in time singly exponential in the size of* $(\mathcal{K},\varphi)$, *a certificate* ρ'' *which is h-prefix bisimilar to* $\rho \star \rho'$ *(resp.,* $\rho' \star \rho$*), with* $h = \mathrm{d_B}(\varphi)$.

The set $\mathsf{A\overline{A}}(\varphi)$ is the set of formulas in $\mathsf{SD}(\varphi)$ of the form $\langle X \rangle \psi'$ or $[X]\psi'$ with $X \in \{A,\overline{A}\}$. An $\mathsf{A\overline{A}}$-labeling *Lab* for $(\mathcal{K},\varphi)$ is a mapping associating to each state s of $\mathcal{K}$ a maximally consistent set of subformulas of $\mathsf{A\overline{A}}(\varphi)$. More precisely, for all $s \in S$, $Lab(s)$ is such that for all $\psi, \widetilde{\psi} \in \mathsf{A\overline{A}}(\varphi)$,

checkTrue$_{(\mathcal{K},\varphi,Lab)}(\mathscr{W})$ [$\mathscr{W}$ *is a well-formed set and Lab is an* $\mathsf{A}\overline{\mathsf{A}}$*-labeling for* $(\mathcal{K},\varphi)$]

`while` $\mathscr{W}$ is *not* universal `do`
  `deterministically select` $(\psi,\rho)\in\mathscr{W}$ such that ψ is not of the form $[\overline{E}]\psi'$ and $[\overline{B}]\psi'$
  `update` $\mathscr{W}\leftarrow\mathscr{W}\setminus\{(\psi,\rho)\}$;
  `case` $\psi=r$ with $r\in\mathsf{RE}$: `if` $\rho\notin\mathscr{L}(r)$ `then` *reject the input*;
  `case` $\psi=\neg r$ with $r\in\mathsf{RE}$: `if` $\rho\in\mathscr{L}(r)$ `then` *reject the input*;
  `case` $\psi=\langle\mathrm{A}\rangle\psi'$ or $\psi=[A]\psi'$: `if` $\psi\notin Lab(\mathrm{lst}(\rho))$ `then` *reject the input*;
  `case` $\psi=\langle\overline{\mathrm{A}}\rangle\psi'$ or $\psi=[\overline{A}]\psi'$: `if` $\psi\notin Lab(\mathrm{fst}(\rho))$ `then` *reject the input*;
  `case` $\psi=\psi_1\vee\psi_2$: `existentially choose` $i=1,2$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi_i,\rho)\}$;
  `case` $\psi=\psi_1\wedge\psi_2$: `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi_1,\rho),(\psi_2,\rho)\}$;
  `case` $\psi=\langle\mathrm{B}\rangle\psi'$: `existentially choose` $\rho'\in\mathrm{Pref}(\rho)$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\}$;
  `case` $\psi=[B]\psi'$: `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\mid\rho'\in\mathrm{Pref}(\rho)\}$;
  `case` $\psi=\langle X\rangle\psi'$ with $X\in\{\overline{E},\overline{B}\}$: `existentially choose` an X*-witness* ρ' of ρ for $(\mathcal{K},\varphi)$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\}$;
`end while`
`if` $\mathscr{W}=\emptyset$ `then` *accept* `else universally choose` $(\psi,\rho)\in\widetilde{\mathscr{W}}$ and `call` *checkFalse*$_{(\mathcal{K},\varphi,Lab)}(\{(\psi,\rho)\})$

Figure 2: Procedure *checkTrue*

$Lab(s)\cap\{\psi,\widetilde{\psi}\}$ is a singleton. We say that *Lab* is *valid* if for all states $s\in S$ ad $\psi\in Lab(s)$, $\mathcal{K},s\models\psi$ (we consider s as a length-1 trace). Finally, a *well-formed set for* $(\mathcal{K},\varphi)$ is a finite set $\mathscr{W}$ consisting of pairs (ψ,ρ) such that $\psi\in\mathsf{SD}(\varphi)$ and ρ is a certificate of $(\mathcal{K},\varphi)$. We say that $\mathscr{W}$ is *universal* if each formula occurring in $\mathscr{W}$ is of the form $[X]\psi$ with $X\in\{\overline{B},\overline{E}\}$. The *dual* $\widetilde{\mathscr{W}}$ of $\mathscr{W}$ is the well-formed set obtained by replacing each pair $(\psi,\rho)\in\mathscr{W}$ with $(\widetilde{\psi},\rho)$. A well-formed set $\mathscr{W}$ is *valid* if for each $(\psi,\rho)\in\mathscr{W}$, $\mathcal{K},\rho\models\psi$.

The procedure *check*, reported in Figure 1, defines the ATM required to prove the assertion of Theorem 15. The procedure *check* takes a pair $(\mathcal{K},\varphi)$ as input and: (1) it guesses an $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* for $(\mathcal{K},\varphi)$; (2) it checks that the guessed labeling *Lab* is valid; (3) for every certificate ρ starting from the initial state, it checks that $\mathcal{K},\rho\models\varphi$. To perform steps (2)–(3), it exploits the auxiliary ATM procedure *checkTrue* reported in Figure 2. The procedure *checkTrue* takes as input a well-formed set $\mathscr{W}$ for $(\mathcal{K},\varphi)$ and, assuming that the current $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* is valid, checks whether $\mathscr{W}$ is valid. For each pair $(\psi,\rho)\in\mathscr{W}$ such that ψ is not of the form $[X]\psi'$ with $X\in\{\overline{B},\overline{E}\}$, *checkTrue* directly checks whether $\mathcal{K},\rho\models\psi$. In order to allow a deterministic choice of the current element of the iteration, we assume that the set $\mathscr{W}$ is implemented as an ordered data structure. At each iteration of the while loop in *checkTrue*, the current pair $(\psi,\rho)\in\mathscr{W}$ is processed according to the semantics of HS, exploiting the guessed $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* and Proposition 16. The processing is either deterministic or based on an existential choice, and the currently processed pair (ψ,ρ) is either removed from $\mathscr{W}$, or replaced with pairs (ψ',ρ') such that ψ' is a strict subformula of ψ.

At the end of the while loop, the resulting well formed set $\mathscr{W}$ is either empty or universal. In the former case, the procedure accepts. In the latter case, there is a switch in the current operation mode. For each element (ψ,ρ) in the dual of $\mathscr{W}$ (note that the root modality of ψ is either $\langle\overline{\mathrm{E}}\rangle$ or $\langle\overline{\mathrm{B}}\rangle$), the auxiliary ATM procedure *checkFalse* is invoked, which accepts the input $\{(\psi,\rho)\}$ iff $\mathcal{K},\rho\not\models\psi$. The procedure *checkFalse* is the *dual* of *checkTrue*: it is simply obtained from *checkTrue* by switching *accept* and *reject*, by switching existential choices and universal choices, and by converting the last call to *checkFalse* into *checkTrue* (see Figure 3 in Appendix B.3). Thus *checkFalse* accepts an input $\mathscr{W}$ iff $\mathscr{W}$ is *not* valid.

Recall that the length of a certificate is singly exponential in the size of the input $(\mathcal{K}, \varphi)$. Thus, since the number of alternations of the ATM *check* between existential and universal choices is evidently the number of switches between the calls to the procedures *checkTrue* and *checkFalse* plus two, by Theorem 14 and Proposition 16, we state the following result that directly implies Theorem 15 (Appendix B.3).

Proposition 17. *The ATM check is a singly exponential-time bounded ATM accepting* FMC *whose number of alternations on an input* $(\mathcal{K}, \varphi)$ *is at most* $\Upsilon(\varphi)+2$.

5 $\mathbf{AEXP_{pol}}$-hardness of the fragment $\mathsf{B\overline{E}}$

In this section, we show that the MC problem for the fragment $\mathsf{B\overline{E}}$ is $\mathbf{AEXP_{pol}}$-hard (implying the $\mathbf{AEXP_{pol}}$-hardness of $\mathsf{A\overline{A}B\overline{B}\overline{E}}$). The result is obtained by a polynomial-time reduction from a variant of the domino-tiling problem for grids with rows and columns of exponential length called *alternating multi-tiling problem*.

An instance of this problem is a tuple $\mathscr{I} = (n, D, D_0, H, V, M, D_{acc})$, where: n is a positive *even* natural number encoded in unary; D is a non-empty finite set of *domino types*; $D_0 \subseteq D$ is a set of *initial domino types*; $H \subseteq D \times D$ and $V \subseteq D \times D$ are the *horizontal* and *vertical matching relations*, respectively; $M \subseteq D \times D$ is the *multi-tiling matching relation*; $D_{acc} \subseteq D$ is a set of *accepting domino types*. A *tiling of* $\mathscr{I}$ is a map assigning a domino type to each cell of a $2^n \times 2^n$ squared grid coherently with the horizontal and vertical matching relations. Formally, a tiling of $\mathscr{I}$ is a mapping $f : [0, 2^n-1] \times [0, 2^n-1] \to D$ s.t.:

- for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ with $j < 2^n-1$, $(f(i,j), f(i,j+1)) \in H$;
- for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ with $i < 2^n-1$, $(f(i,j), f(i+1,j)) \in V$.

The *initial condition* $Init(f)$ of the tiling f is the content of the first row of f, namely $Init(f) := f(0,0)f(0,1)\dots f(0,2^n-1)$. A *multi-tiling of* $\mathscr{I}$ is a tuple $(f_1, \dots, f_n)$ of n tilings which are coherent w.r.t. the multi-tiling matching relation M, namely, such that:

- (i) for all $i, j \in [0, 2^n-1] \times [0, 2^n-1]$ and $\ell \in [1, n-1]$, $(f_\ell(i,j), f_{\ell+1}(i,j)) \in M$ (*multi-cell requirement*), and (ii) $f_n(2^n-1, j) \in D_{acc}$ for some $j \in [0, 2^n-1]$ (*acceptance*).

The *alternating multi-tiling problem* for an instance $\mathscr{I}$ is checking whether

- $\forall w_1 \in (D_0)^{2^n}, \exists w_2 \in (D_0)^{2^n}, \dots, \forall w_{n-1} \in (D_0)^{2^n}, \exists w_n \in (D_0)^{2^n}$ such that there exists a multi-tiling $(f_1, \dots, f_n)$ where for all $i \in [1, n]$, $Init(f_i) = w_i$.

Theorem 18. *The alternating multi-tiling problem is* $\mathbf{AEXP_{pol}}$*-complete.*

See Appendix C.1 for a proof of Theorem 18. The fact that the MC problem for the fragment $\mathsf{B\overline{E}}$ is $\mathbf{AEXP_{pol}}$-hard is an immediate corollary of the following result.

Theorem 19. *One can construct, in time polynomial in the size of* $\mathscr{I}$*, a finite Kripke structure* $\mathcal{K}_\mathscr{I}$ *and a* $\mathsf{B\overline{E}}$ *formula* $\varphi_\mathscr{I}$ *over the set of propositions* $\mathcal{AP} = D \cup (\{r,c\} \times \{0,1\}) \cup \{\bot, end\}$ *such that* $\mathcal{K}_\mathscr{I} \models \varphi_\mathscr{I}$ *iff* $\mathscr{I}$ *is a* positive *instance of the alternating multi-tiling problem.*

The rest of this section is devoted to the construction of the Kripke structure $\mathcal{K}_\mathscr{I}$ and the $\mathsf{B\overline{E}}$ formula $\varphi_\mathscr{I}$ proving Theorem 19. Let $\mathcal{AP}$ be as in the statement of Theorem 19. The Kripke structure $\mathcal{K}_\mathscr{I}$ is given by $\mathcal{K}_\mathscr{I} = (\mathcal{AP}, S, R, \mu, s_0)$, where $S = \mathcal{AP}$, $s_0 = end$, μ is the identity mapping (we identify a singleton set $\{p\}$ with p), and $R = \{(s, s') : s \in \mathcal{AP} \setminus \{end\}, s' \in \mathcal{AP}\}$. Note that the initial state *end* has no successor, and that a trace of $\mathcal{K}_\mathscr{I}$ can be identified with its induced labeling sequence. The construction of the $\mathsf{B\overline{E}}$ formula $\varphi_\mathscr{I}$ is based on a suitable encoding of multi-tilings which is described in the following. The symbols $\{r\} \times \{0,1\}$ and $\{c\} \times \{0,1\}$ in $\mathcal{AP}$ are used to encode the values of two n-bits counters numbering the 2^n rows and columns, respectively, of a tiling. For a multi-tiling $F = (f_1, \dots, f_n)$ and for all $i, j \in [0, 2^n-1]$, the (i,j)-th *multi-cell* $(f_1(i,j), \dots, f_n(i,j))$ of F is encoded by the word C of length

$3n$ over $\mathcal{AP}$, called *multi-cell code*, given by $d_1 \cdots d_n(r,b_1)\cdots(r,b_n)(c,b'_1)\cdots(c,b'_n)$ where $b_1\cdots b_n$ and $b'_1\cdots b'_n$ are the binary encodings of the row number i and column number j, respectively, and for all $\ell \in [1,n]$, $d_\ell = f_\ell(i,j)$ (i.e., the content of the (i,j)-th cell of component f_ℓ). The *content* of C is $d_1\cdots d_n$. Since F is a multi-tiling, the following well-formedness requirement must be satisfied by the encoding C: for all $\ell \in [1,n-1]$, $(d_\ell, d_{\ell+1}) \in M$. We call such words *well-formed multi-cell codes*.

Definition 20 (Multi-tiling codes)**.** A *multi-tiling code* is a finite word w over $\mathcal{AP}$ obtained by concatenating well-formed multi-cell codes in such a way that the following conditions hold:

- for all $i,j \in [0,2^n-1]$, there is a multi-cell code in w with row number i and column number j *(completeness requirement)*;
- for all multi-cell codes C and C' occurring in w, if C and C' have the same row number and column number, then C and C' have the same content *(uniqueness requirement)*;
- for all multi-cell codes C and C' in w having the same row-number (resp., column number), column numbers (resp., row numbers) j and $j+1$, respectively, and contents $d_1\cdots d_n$ and $d'_1\cdots d'_n$, respectively, it holds that $(d_\ell, d'_\ell) \in H$ (resp. $(d_\ell, d'_\ell) \in V$) for all $\ell \in [1,n]$ *(row-adjacency requirement)* (resp., *(column-adjacency requirement)*);
- there is a multi-cell code in w with row-number 2^n-1 whose content is in $D^{n-1}\cdot d_{acc}$ for some $d_{acc} \in D_{acc}$ *(acceptance requirement)*.

Finally, we have to encode the initial conditions of the components of a multi-tiling. An *initial cell code* encodes a cell of the first row of a tiling and is a word w of length $n+1$ of the form $w = d(c,b_1)\cdots(c,b_n)$, where $d \in D_0$ and $b_1,\ldots,b_n \in \{0,1\}$. We say that d is the *content* of w and the integer in $[0,2^n-1]$ encoded by $b_1\cdots b_n$ is the *column number* of w.

Definition 21 (Multi-initialization codes)**.** An *initialization code* is a finite word w over $\mathcal{AP}$ which is the concatenation of initial cell codes such that:

- for all $i \in [0,2^n-1]$, there is an initial cell code in w with column number i.
- for all initial cell codes C and C' occurring in w, if C and C' have the same column number, then C and C' have the same content.

A *multi-initialization code* is a finite word over $\mathcal{AP}$ of the form $\bot\cdot w_n\cdots\bot\cdot w_1\cdot end$ such that for all $\ell \in [1,n]$, w_ℓ is an initialization code.

Definition 22 (Initialized multi-tiling codes)**.** An *initialized multi-tiling code* is a finite word over $\mathcal{AP}$ of the form $\bot\cdot w\cdot\bot\cdot w_n\cdots\bot\cdot w_1\cdot end$ such that w is a multi-tiling code, $\bot\cdot w_n\cdots\bot\cdot w_1\cdot end$ is a multi-initialization code, and the following requirement holds:

- for each multi-cell code in w having row number 0, column number i, and content $d_1\cdots d_n$ and for all $\ell \in [1,n]$, there is an initial cell code in w_ℓ having column number i and content d_ℓ *(initialization coherence requirement)*.

We sketch now the idea for the construction of the $\mathsf{B\overline{E}}$ formula $\varphi_{\mathscr{I}}$ ensuring that $\mathcal{K}_{\mathscr{I}} \models \varphi_{\mathscr{I}}$ iff $\mathscr{I}$ is a positive instance of the alternating multi-tiling problem. We preliminarily observe that since the initial state of $\mathcal{K}_{\mathscr{I}}$ has no successors, the only initial trace of $\mathcal{K}_{\mathscr{I}}$ is the trace *end* of length 1. To guess a trace corresponding to an initialized multi-tiling code, $\mathcal{K}_{\mathscr{I}}$ is unraveled backward starting from *end*, exploiting the modality $\overline{\mathsf{E}}$. The structure of the formula $\varphi_{\mathscr{I}}$ is $\varphi_{\mathscr{I}} := [\overline{E}](\varphi_1 \rightarrow \langle\overline{\mathrm{E}}\rangle(\varphi_2 \wedge (\ldots([\overline{E}](\varphi_{n-1} \rightarrow \langle\overline{\mathrm{E}}\rangle(\varphi_n \wedge \langle\overline{\mathrm{E}}\rangle\, \varphi_{IMT})))\ldots)))$. The formula $\varphi_{\mathscr{I}}$ features $n+1$ unravelling steps starting from the initial trace *end*. The first n steps are used to guess a sequence of n initialization codes. Intuitively, each formula φ_i is used to constrain the i-th unravelling to be an initialization code, in such a way that at depth n in the formula a multi-initialization code is under evaluation. The last unravelling step (the innermost in the formula) is used to guess the multi-tiling code. Intuitively, the innermost formula φ_{IMT} is evaluated over a trace corresponding to an initialized multi-tiling code, and checks its structure: multi-cell codes are "captured" by regular expressions (encoding in particular their row and column numbers

and contents); moreover the completeness, uniqueness, row- and column-adjacency requirements of Definition 20 are enforced by the joint use of $[\overline{E}]$ and regular expressions: intuitively, by means of $[\overline{E}]$, one or two multi-cell codes are generated "separately"; then, if they appear in the considered multi-tiling code, the aforementioned constraints are verified by means of auxiliary formulas, consisting of suitable regular expressions. The initialization coherence requirement of Definition 22 is guaranteed in an analogous way, by comparing initial cell codes and multi-cell codes. Note that the first $n-1$ occurrences of alternations between universal and existential modalities $[\overline{E}]$ and $\langle\overline{\mathrm{E}}\rangle$ correspond to the alternations of universal and existential quantifications in the definition of alternating multi-tiling problem. The correctness of the construction of $\varphi_{\mathscr{I}}$ is stated by the next proposition (the definition of $\varphi_1,\ldots,\varphi_n$, φ_{IMT} and a proof of Proposition 23 are in Appendix C.2).

Proposition 23. *One can build, in time polynomial in the size of $\mathscr{I}$, $n+1$ $\mathsf{B}\overline{\mathsf{E}}$ formulas $\varphi_{IMT},\varphi_1,\ldots,\varphi_n$ such that $\Upsilon(\varphi_{IMT})=\Upsilon(\varphi_1)=\ldots=\Upsilon(\varphi_n)=0$, and fulfilling the following conditions.*

- *For all finite words ρ over $\mathcal{AP}$ of the form $\rho=\rho'\cdot\bot\cdot w_n\cdots\bot\cdot w_1\cdot end$ such that $\rho'\neq\varepsilon$ and $\bot\cdot w_n\cdots\bot\cdot w_1\cdot end$ is a multi-initialization code, $\mathcal{K}_{\mathscr{I}},\rho\models\varphi_{IMT}$ if and only if ρ is an initialized multi-tiling code.*
- *For all $\ell\in[1,n]$ and words ρ of the form $\rho=\rho'\cdot\bot\cdot w_{\ell-1}\cdots\bot\cdot w_1\cdot end$ such that $\rho'\neq\varepsilon$ and $w_j\in(\mathcal{AP}\setminus\{\bot\})^*$ for all $j\in[1,\ell-1]$, $\mathcal{K}_{\mathscr{I}},\rho\models\varphi_\ell$ if and only if ρ' is of the form $\rho'=\bot\cdot w_\ell$, where w_ℓ is an initialization code.*

Since the initial state of $\mathcal{K}_{\mathscr{I}}$ has no successors and corresponds to the atomic proposition *end*, by Proposition 23 and Definitions 20–22, we obtain that $\mathcal{K}_{\mathscr{I}}\models\varphi_{\mathscr{I}}$ iff $\mathscr{I}$ is a positive instance of the alternating multi-tiling problem. This concludes the proof of Theorem 19.

6 Conclusions

In this paper, we have investigated the MC problem for two maximal fragments of HS, $\mathsf{A}\overline{\mathsf{A}}\mathsf{B}\overline{\mathsf{B}\mathsf{E}}$ and $\mathsf{A}\overline{\mathsf{A}}\mathsf{E}\overline{\mathsf{B}\mathsf{E}}$, endowed with interval labeling based on regular expressions, and we have proved that such a problem is $\mathbf{AEXP_{pol}}$-complete. The paper also settles, in the more general setting of the regular expression-based semantics, the open complexity question for the same fragments under the homogeneity assumption. Future work will focus on the problem of determining the exact complexity of MC for full HS, both under homogeneity and in the regular expression-based semantics. In addition, we will study the MC problem for HS over *visibly pushdown systems* (VPS), in order to deal with recursive programs and infinite state systems.

References

[1] J. F. Allen (1983): *Maintaining Knowledge about Temporal Intervals*. Communications of the ACM 26(11), pp. 832–843, doi:10.1145/182.358434.

[2] L. Bozzelli, H. van Ditmarsch & S. Pinchinat (2015): *The Complexity of One-agent Refinement Modal Logic*. Theoretical Computer Science 603(C), pp. 58–83, doi:10.1016/j.tcs.2015.07.015.

[3] L. Bozzelli, A. Molinari, A. Montanari & A. Peron (2017): *An in-Depth Investigation of Interval Temporal Logic Model Checking with Regular Expressions*. In: *SEFM*. Preprint available at `https://www.dimi.uniud.it/la-ricerca/pubblicazioni/preprints/2.2017/`.

[4] 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, pp. 389–405, doi:10.1007/978-3-319-40229-1_27.

[5] L. Bozzelli, A. Molinari, A. Montanari, A. Peron & P. Sala (2016): *Interval vs. Point Temporal Logic Model Checking: an Expressiveness Comparison*. In: *FSTTCS*, pp. 26:1–14, doi:10.4230/LIPIcs.FSTTCS.2016.26.

[6] L. Bozzelli, A. Molinari, A. Montanari, A. Peron & P. Sala (2016): *Model Checking the Logic of Allen's Relations Meets and Started-by is* $\mathbf{P}^{\mathbf{NP}}$*-Complete*. In: *GandALF*, pp. 76–90, doi:10.4204/EPTCS.226.6.

[7] 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, doi:10.1007/s10472-013-9376-4.

[8] A. K. Chandra, D. C. Kozen & L. J. Stockmeyer (1981): *Alternation*. *Journal of the ACM* 28(1), pp. 114–133, doi:10.1145/322234.322243.

[9] 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, doi:10.1145/4904.4999.

[10] J. Ferrante & C. Rackoff (1975): *A Decision Procedure for the First Order Theory of Real Addition with Order*. *SIAM Journal of Computation* 4(1), pp. 69–76, doi:10.1137/0204006.

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

[12] M. Gligoric & R. Majumdar (2013): *Model Checking Database Applications*. In: *TACAS*, pp. 549–564, doi:10.1007/978-3-642-36742-7_40.

[13] J. Y. Halpern & Y. Shoham (1991): *A Propositional Modal Logic of Time Intervals*. *Journal of the ACM* 38(4), pp. 935–962, doi:10.1145/115234.115351.

[14] S. C. Kleene (1956): *Representation of Events in Nerve Nets and Finite Automata*. In: *Automata Studies*, 34, Princeton University Press, pp. 3–41.

[15] A. Lomuscio & J. Michaliszyn (2013): *An Epistemic Halpern-Shoham Logic*. In: *IJCAI*, pp. 1010–1016. Available at `http://dl.acm.org/citation.cfm?id=2540128.2540274`.

[16] A. Lomuscio & J. Michaliszyn (2014): *Decidability of model checking multi-agent systems against a class of EHS specifications*. In: *ECAI*, pp. 543–548, doi:10.3233/978-1-61499-419-0-543.

[17] A. Lomuscio & J. Michaliszyn (2016): *Model Checking Multi-agent Systems Against Epistemic HS Specifications with Regular Expressions*. In: *KR*, AAAI Press, pp. 298–307.

[18] A. Lomuscio & F. Raimondi (2006): *MCMAS: A Model Checker for Multi-agent Systems*. In: *TACAS*, LNCS 3920, Springer, pp. 450–454, doi:10.1007/11691372_31.

[19] J. Marcinkowski & J. Michaliszyn (2014): *The Undecidability of the Logic of Subintervals*. *Fundamenta Informaticae* 131(2), pp. 217–240, doi:10.3233/FI-2014-1011.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] B. Moszkowski (1983): *Reasoning About Digital Circuits*. Ph.D. thesis, Dept. of Computer Science, Stanford University, Stanford, CA.

[25] A. Pnueli (1977): *The temporal logic of programs*. In: *FOCS*, IEEE, pp. 46–57, doi:10.1109/SFCS.1977.32.

[26] I. Pratt-Hartmann (2005): *Temporal prepositions and their logic*. *Artificial Intelligence* 166(1-2), pp. 1–36, doi:10.1016/j.artint.2005.04.003.

[27] P. Roeper (1980): *Intervals and Tenses*. *J. Philosophical Logic* 9, pp. 451–469, doi:10.1007/BF00262866.

[28] Y. Venema (1990): *Expressiveness and Completeness of an Interval Tense Logic*. *Notre Dame Journal of Formal Logic* 31(4), pp. 529–547, doi:10.1305/ndjfl/1093635589.

Appendix

A Proofs from Section 3

A.1 Proof of Proposition 7

Proposition 7. *Let $h \geq 0$, and ρ and ρ' be two h-prefix bisimilar traces of $\mathcal{K}$. Then, for all traces ρ_L and ρ_R of $\mathcal{K}$ such that $\rho_L \star \rho$ and $\rho \star \rho_R$ are defined, the following holds:*
(1) $\rho_L \star \rho$ and $\rho_L \star \rho'$ are h-prefix bisimilar; (2) $\rho \star \rho_R$ and $\rho' \star \rho_R$ are h-prefix bisimilar.

Proof. First note that since $\mathscr{S}(\rho) = \mathscr{S}(\rho')$, $\mathrm{fst}(\rho) = \mathrm{fst}(\rho')$ and $\mathrm{lst}(\rho) = \mathrm{lst}(\rho')$. Hence, $\rho_L \star \rho$ (resp., $\rho \star \rho_R$) is defined iff $\rho_L \star \rho'$ (resp., $\rho' \star \rho_R$) is defined. The proofs of Properties 1 and 2 are by induction on h.

Property 1: Since ρ and ρ' are h-prefix bisimilar, $\mathscr{S}(\rho) = \mathscr{S}(\rho')$. Thus, by Proposition 4, $\mathscr{S}(\rho_L \star \rho) = \mathscr{S}(\rho_L \star \rho')$. Thus, if $h = 0$ (base case), the result follows. Now, assume that $h > 0$ (induction step). Assume that ν is a proper prefix of $\rho_L \star \rho$ (the symmetric case, where we choose a proper prefix of $\rho_L \star \rho'$ is similar). We need to show that there exists a proper prefix ν' of $\rho_L \star \rho'$ such that ν and ν' are $(h-1)$-prefix bisimilar. If ν is a prefix of ρ_L, then we set $\nu' = \nu$ and the result trivially holds (note that since ρ and ρ' are h-prefix bisimilar, it holds that $|\rho| > 1$ iff $|\rho'| > 1$). Otherwise, there is a proper prefix ξ of ρ such that $\nu = \rho_L \star \xi$. Since ρ and ρ' are h-prefix bisimilar, there exists a proper prefix ξ' of ρ' such that ξ and ξ' are $(h-1)$-prefix bisimilar. Thus, by setting $\nu' = \rho_L \star \xi'$, by the inductive hypothesis, the result follows.

Property 2: By Proposition 4, $\mathscr{S}(\rho \star \rho_R) = \mathscr{S}(\rho' \star \rho_R)$. Thus, if $h = 0$, the result follows. Now, assume that $h > 0$. We proceed by a double induction on $|\rho_R|$. For the base case, where $|\rho_R| = 1$ the result is obvious. Now, assume that $|\rho_R| > 1$. Let ν be a proper prefix of $\rho \star \rho_R$ (the symmetric case, where we choose a proper prefix of $\rho' \star \rho_R$ is similar). We need to show that there exists a proper prefix ν' of $\rho' \star \rho_R$ such that ν and ν' are $(h-1)$-prefix bisimilar. If $\nu = \rho$ or ν is a proper prefix of ρ, then there exists a prefix ν' of ρ' such that ν and ν' are $(h-1)$-prefix bisimilar. Thus, since ν' is a proper prefix of $\rho' \star \rho_R$, the result follows. Otherwise, there exists a proper prefix ξ of ρ_R such that $\nu = \rho \star \xi$. By setting $\nu' = \rho' \star \xi$, considering the inductive hypothesis on $|\rho_R|$, we obtain that ν and ν' are h-prefix bisimilar, hence $(h-1)$-prefix bisimilar as well, concluding the proof. □

A.2 Proof of Lemma 12

We prove a stronger result.

Lemma 24. *Let $h \geq 0$, ρ and ρ' be two traces of $\mathcal{K}$, and PS_h and PS'_h be the two h-prefix samplings of ρ and ρ', respectively. Assume that ρ and ρ' have the same h-sampling word. Hence, there is $N \geq 1$ such that*

- PS_h: $i_1 < i_2 < \ldots < i_N$;
- PS'_h: $i'_1 < i'_2 < \ldots < i'_N$;
- *for all $j \in [1,N]$, $\mathscr{S}(\rho(1,i_j)) = \mathscr{S}(\rho'(1,i'_j))$.*

Then, for all $j \in [1,N-1]$, $n \in [i_j+1, i_{j+1}]$ and $n' \in [i'_j+1, i'_{j+1}]$ such that $\mathscr{S}(\rho[1,n]) = \mathscr{S}(\rho'[1,n'])$, it holds that $\rho(1,n)$ and $\rho'(1,n')$ are h-prefix bisimilar.

Proof. The proof is by induction on $h \geq 0$. For $h = 0$, the result is obvious. Now, assume that $h > 0$. If $N = 1$ (resp., $N = 2$), then $\rho = \rho'$ and $|\rho| = |\rho'| = N$. Hence, evidently, the result follows. Now, assume that $N > 2$. Since by hypothesis, $\mathscr{S}(\rho(1,n)) = \mathscr{S}(\rho'(1,n'))$, we need to show that

1. for each $m \in [1, n-1]$, there exists $m' \in [1, n'-1]$ such that $\rho(1,m)$ and $\rho'(1,m')$ are $(h-1)$-prefix bisimilar;
2. for each $m' \in [1, n'-1]$, there exists $m \in [1, n-1]$ such that $\rho(1,m)$ and $\rho'(1,m')$ are $(h-1)$-prefix bisimilar;

We prove Property 1 (the proof of Property 2 being symmetric). We use the following fact that can be easily proved.

Claim: let $k \in [0, h-1]$ and $1 = x_1 < \ldots < x_r = N$ be the subsequence of $1, \ldots, N$ such that $i_{x_1} < \ldots < i_{x_r}$ is the k-prefix sampling of ρ. Then, $i'_{x_1} < \ldots < i'_{x_r}$ is the k-prefix sampling of ρ'.

Now, we prove Property 1. Let $m \in [1, n-1]$. If $m = 1$, we set $m' = 1$, and the result follows. Now, assume that $m \geq 2$. Since $h > 0$, there must exist $x, y \in [1, N]$ such that $x < y$, $m \in [i_x + 1, i_y]$, and i_x and i_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ. By the claim above, i'_x and i'_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ'. We distinguish two cases:

- $m = i_y$. Since $n \in [i_j + 1, i_{j+1}]$ and $m < n$, it holds that $i_y \leq i_j$. Hence, $i'_y \leq i'_j$ as well. Moreover, since $n' > i'_j$, it holds that $i'_y < n'$. We set $m' = i'_y$. Since $\mathscr{S}(\rho(1, i_y)) = \mathscr{S}(\rho'(1, i'_y))$, $m = i_y$, $m' = i'_y$, and i_x and i_y (resp., i'_x and i'_y) are two consecutive positions in the $(h-1)$-prefix sampling of ρ (resp., ρ'), by the induction hypothesis on h, the result follows.
- $m \neq i_y$. Hence, $m \in [i_x + 1, i_y - 1]$. Since i_x and i_y are two consecutive positions in the $(h-1)$-prefix sampling of ρ, there must exist $z \in [x+1, y-1]$ such that $i_z \leq m$ and $\mathscr{S}(\rho(1,m)) = \mathscr{S}(\rho(1, i_z))$. Since $i_z \leq m$, $m < n$, and $n \in [i_j + 1, i_{j+1}]$, it holds that $i_z \leq i_j$. Hence, $i'_z \leq i'_j < n'$. We set $m' = i'_z$. Since $\mathscr{S}(\rho(1, i_z)) = \mathscr{S}(\rho'(1, i'_z))$, we obtain that $\mathscr{S}(\rho(1,m)) = \mathscr{S}(\rho'(1,m'))$, $m \in [i_x + 1, i_y]$ and $m' \in [i'_x + 1, i'_y]$. Thus, being i_x and i_y (resp., i'_x and i'_y) two consecutive positions in the $(h-1)$-prefix sampling of ρ (resp., ρ'), by the induction hypothesis on h, the result follows. □

B Proofs from Section 4

B.1 Alternating Turing Machines

We shortly recall the framework of Alternating Turing Machines (ATM, for short) [8]. Without loss of generality, we consider a model of alternation with a binary branching degree. Formally, an ATM is a tuple $\mathscr{M} = (\Sigma, Q, Q_\forall, Q_\exists, q_0, \delta, F)$, where Σ is the input alphabet, which contains the blank symbol #, Q is the finite set of states which is partitioned into $Q = Q_\forall \cup Q_\exists$, $Q_\exists$ (resp., $Q_\forall$) is the set of existential (resp., universal) states, q_0 is the initial state, $F \subseteq Q$ is the set of accepting states, and the transition function δ is a mapping $\delta : Q \times \Sigma \to (Q \times \Sigma \times \{L, R\})^2$. Configurations of $\mathscr{M}$ are words in $\Sigma^* \cdot (Q \times \Sigma) \cdot \Sigma^*$. A configuration $C = \eta \cdot (q, \sigma) \cdot \eta'$ denotes that the tape content is $\eta \cdot \sigma \cdot \eta'$, the current state (resp., input symbol) is q (resp., σ), and the reading head is at position $|\eta| + 1$. From configuration C, the machine $\mathscr{M}$ nondeterministically chooses a triple (q', σ', dir) in $\delta(q, \sigma) = ((q_l, \sigma_l, dir_l), (q_r, \sigma_r, dir_r))$, and then moves to state q', writes σ' in the current tape cell, and its reading head moves one cell to the left or to the right, according to dir. We denote by $succ_l(C)$ and $succ_r(C)$ the successors of C obtained by choosing respectively the left and the right triple in $((q_l, \sigma_l, dir_l), (q_r, \sigma_r, dir_r))$. The configuration C is accepting (resp., universal, resp., existential) if the associated state q is in F (resp., in $Q_\forall$, resp., in $Q_\exists$). Given an input $\alpha \in \Sigma^*$, a (finite) computation tree of $\mathscr{M}$ over α is a finite tree in which each node is labeled by a configuration. The root of the tree corresponds to the initial configuration associated with α, i.e., $(q_0, \alpha(1)) \cdot \alpha(2, |\alpha|)$. An *internal* node that is labeled by a universal configuration C has two children, corresponding to $succ_l(C)$ and $succ_r(C)$, while an internal node labeled by an existential configuration C has a single child, corresponding to either $succ_l(C)$ or $succ_r(C)$. The tree is accepting iff each of its

leaves is labeled by an accepting configuration. An input $\alpha \in \Sigma^*$ is *accepted* by $\mathscr{M}$ iff there exists an accepting computation tree of $\mathscr{M}$ over α.

The ATM $\mathscr{M}$ is singly exponential-time bounded if there is an integer constant $c \geq 1$ such that for each input α, when started on α—no matter what are the universal and existential choices—$\mathscr{M}$ halts in at most $2^{|\alpha|^c}$ steps. The ATM $\mathscr{M}$ has a polynomial-bounded number of alternations if there is an integer constant $c \geq 1$ such that for all inputs α and computation paths π from α, the number of alternations of existential and universal configurations along π is at most $|\alpha|^c$.

B.2 Proof of Proposition 16

Proposition 16. *Let $\mathcal{K}$ be a finite Kripke structure, φ an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula in PNF, and ρ a certificate for $(\mathcal{K},\varphi)$. The following properties hold:*

1. *for each $\langle X\rangle\psi \in \mathsf{SD}(\varphi)$ with $X \in \{\overline{B},\overline{E}\}$, $\mathcal{K},\rho \models \langle X\rangle\psi$ iff there exists an X-witness ρ' of ρ for $(\mathcal{K},\varphi)$ such that $\mathcal{K},\rho' \models \psi$;*
2. *for each trace of the form $\rho\star\rho'$ (resp., $\rho'\star\rho$) such that ρ' is a certificate for $(\mathcal{K},\varphi)$, one can construct in time singly exponential in the size of $(\mathcal{K},\varphi)$, a certificate ρ'' which is h-prefix bisimilar to $\rho\star\rho'$ (resp., $\rho'\star\rho$), with $h = \mathrm{d_B}(\varphi)$.*

Proof. First, we prove Property 1. Let $\langle X\rangle\psi \in \mathsf{SD}(\varphi)$ with $X \in \{\overline{B},\overline{E}\}$, $h = \mathrm{d_B}(\varphi)$, and ρ be a certificate for $(\mathcal{K},\varphi)$. Assume that $X = \overline{E}$ (the case where $X = \overline{B}$ being similar). First, assume that there exists an $\overline{E}$-witness ρ' of ρ for $(\mathcal{K},\varphi)$ such that $\mathcal{K},\rho' \models \psi$. Hence, ρ' is h-prefix bisimilar to a trace of the form $\rho''\star\rho$ with $|\rho''| > 1$. Since $\langle\overline{E}\rangle\psi \in \mathsf{SD}(\varphi)$, it holds that $\mathrm{d_B}(\langle\overline{E}\rangle\psi) \leq h$. By Proposition 8, it follows that $\mathcal{K},\rho \models \langle\overline{E}\rangle\psi$. For the converse implication, assume that $\mathcal{K},\rho \models \langle\overline{E}\rangle\psi$. Then, there exists a trace of the form $\rho''\star\rho$ with $|\rho''| > 1$ such that $\mathcal{K},\rho''\star\rho \models \psi$. By Theorem 14, there exists a certificate ν for $(\mathcal{K},\varphi)$ which is h-prefix bisimilar to ρ''. By Proposition 7, $\nu\star\rho$ is h-prefix bisimilar to $\rho''\star\rho$. By applying Proposition 8, we deduce that $\mathcal{K},\nu\star\rho \models \psi$. By applying again Theorem 14, there exists a certificate ρ' for $(\mathcal{K},\varphi)$ which is h-prefix bisimilar to $\nu\star\rho$ such that $\mathcal{K},\rho' \models \psi$. Thus, since ρ' is an $\overline{E}$-witness of ρ for $(\mathcal{K},\varphi)$, Property 1 follows.

For Property 2, from the trace $\rho\star\rho'$ (resp., $\rho'\star\rho$), where both ρ and ρ' are certificates for $(\mathcal{K},\varphi)$, we first compute the h-prefix sampling of $\rho\star\rho'$ (resp., $\rho'\star\rho$), where $h = \mathrm{d_B}(\varphi)$. Then, proceeding as in the proof of Theorem 14, we extract from $\rho\star\rho'$ (resp., $\rho'\star\rho$) a trace which is h-prefix bisimilar to $\rho\star\rho'$ (resp., $\rho'\star\rho$). Since the lengths of ρ and ρ' are singly exponential in the sizes of $(\mathcal{K},\varphi)$, Property 2 follows. □

B.3 Proof of Proposition 17

In order to prove Proposition 17, for technical convenience, for an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ, we consider a slight variant $\Upsilon_w(\varphi)$ of $\Upsilon(\varphi)$. Formally, $\Upsilon_w(\varphi)$ is given by $\Upsilon(\langle\overline{\mathrm{B}}\rangle\varphi)$ (or, equivalently, by $\Upsilon(\langle\overline{\mathrm{E}}\rangle\varphi)$). Note that for each $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula φ and $X \in \{\overline{E},\overline{B}\}$, $\Upsilon_w([X]\varphi) = \Upsilon_w(\widetilde{[X]\varphi}) + 1$.

Let $\mathcal{K}$ be a finite Kripke structure, φ be an $\mathsf{A\overline{A}B\overline{B}\overline{E}}$ formula in *PNF*, and $\mathscr{W}$ be a well-formed set for $(\mathcal{K},\varphi)$. We denote by $\Upsilon_w(\mathscr{W})$ the maximum over the alternation depths $\Upsilon_w(\psi)$, where ψ is a formula occurring in $\mathscr{W}$ (we set $\Upsilon_w(\mathscr{W}) = 0$ if $\mathscr{W} = \emptyset$). Note that for each non-empty *universal* well-formed set $\mathscr{W}$ for $(\mathcal{K},\varphi)$, $\Upsilon_w(\widetilde{\mathscr{W}}) = \Upsilon_w(\mathscr{W}) - 1$. Now, we prove Proposition 17.

Proposition 17. *The ATM check is a singly exponential-time bounded ATM accepting* FMC *whose number of alternations on an input $(\mathcal{K},\varphi)$ is at most $\Upsilon(\varphi)+2$.*

Proof. Fix an input $(\mathcal{K},\varphi)$, where φ is an $\mathsf{A\overline{A}B\overline{B}E}$ formula in *PNF*. Note that whenever there is a switch between the procedures *checkTrue* and *checkFalse*, e.g. from *checkTrue* to *checkFalse*, the input $\{(\psi,\rho)\}$ of the called procedure is contained in the dual $\widetilde{\mathscr{W}}$ of the currently processed well-formed set $\mathscr{W}$ for $(\mathcal{K},\varphi)$, and $\mathscr{W}$ is non-empty and universal: hence $\Upsilon_w(\{(\psi,\rho)\}) < \Upsilon_w(\mathscr{W})$. Moreover, a well-formed set $\mathscr{W}$ for $(\mathcal{K},\varphi)$ contains only formulas ψ such that $\psi \in \mathsf{SD}(\varphi)$. Additionally, in each iteration of the while loops of procedures *checkTrue* and *checkFalse*, the processed pair (ψ,ρ) in the current well-formed set $\mathscr{W}$ either is removed from $\mathscr{W}$ or is replaced with pairs (ψ',ρ') such that ψ' is a strict subformula of ψ. This ensures that the algorithm always terminates. Furthermore, since the number of alternations of the ATM *check* between existential choices and universal choices is evidently the number of switches between the calls to procedures *checkTrue* and *checkFalse* plus two, and the top calls to *checkTrue* take in input well-formed sets for $(\mathcal{K},\varphi)$ of the form $\{(\psi,\rho)\}$, where $\psi \in \mathsf{SD}(\varphi)$, we obtain the following result.

Claim 1. The number of alternations of the ATM *check* on an input $(\mathcal{K},\varphi)$ is at most $\Upsilon(\varphi)+2$.

Next, we show the following.

Claim 2. The ATM *check* runs in time singly exponential in the size of the input.

Proof of Claim 2. Fix an input $(\mathcal{K},\varphi)$. Let $T(\varphi)$ be the standard tree encoding of φ, where each node is labeled by some subformula of φ. Let $\psi \in \mathsf{SD}(\varphi)$. If ψ is a subformula of φ, we define d_ψ as the maximum over the distances from the root in $T(\varphi)$ of ψ-labeled nodes. If instead ψ is the dual of a subformula of φ, we let $d_\psi := d_{\widetilde{\psi}}$. Let us denote by $H(\mathcal{K},\varphi)$ the length of a certificate for $(\mathcal{K},\varphi)$. Recall that $H(\mathcal{K},\varphi) = (|S|\cdot 2^{(2|\mathsf{spec}|)^2})^{h+2}$, where S is the set of $\mathcal{K}$-states, spec is the set of atomic formulas (regular expressions) occurring in φ, and $h = \mathrm{d_B}(\varphi)$. By Proposition 16, it follows that each step in an iteration of the while loops in procedures *checkTrue* and *checkFalse* can be performed in time singly exponential in the size of $(\mathcal{K},\varphi)$. Then, in order to prove Claim 2, it suffices to show that for all computations π of the ATM *check* from input $(\mathcal{K},\varphi)$, the overall number N_ψ of iterations of the while loops (of procedures *checkTrue* and *checkFalse*) along π where the formula ψ is processed is at most $(2^{|\varphi|}\cdot H(\mathcal{K},\varphi))^{d_\psi}$. The proof is done by induction on d_ψ. For the base case, $d_\psi = 0$. Therefore, $\psi = \varphi$ or $\psi = \widetilde{\varphi}$, and by construction of the algorithm, N_φ and $N_{\widetilde{\varphi}}$ are at most equal to 1. Hence, the result holds. For the inductive step, assume that $d_\psi > 0$. We consider the case where ψ is a subformula of φ (the case where $\widetilde{\psi}$ is a subformula of φ is similar). Then, the result follows from the following chain of inequalities, where $P(\psi)$ denotes the set of nodes of $T(\varphi)$ which are parents of the nodes labeled by ψ, and for each node x, $fo(x)$ denotes the formula labeling x.

$$N_\psi \leq \sum_{x\in P(\psi)} N_{fo(x)}\cdot H(\mathcal{K},\varphi) \leq \sum_{x\in P(\psi)} (2^{|\varphi|}\cdot H(\mathcal{K},\varphi))^{d_{fo(x)}}\cdot H(\mathcal{K},\varphi) \leq \left(2^{|\varphi|}\cdot H(\mathcal{K},\varphi)\right)^{d_\psi}$$

The first inequality directly follows from the construction of the algorithm (note that if $fo(x) = [B]\psi$, then the processing of subformula $fo(x)$ in an iteration of the two while loops generates at most $H(\mathcal{K},\varphi)$ new "copies" of ψ). The second inequality follows from the inductive hypothesis and the last inequality follows from the fact that $|P(\psi)| \leq 2^{|\varphi|}$ and $d_{fo(x)} \leq d_\psi - 1$ for all $x \in P(\psi)$. This concludes the proof of Claim 2. □

It remains to show that the ATM *check* accepts FMC. Fix an input $(\mathcal{K},\varphi)$ and let *Lab* be the $\mathsf{A\overline{A}}$-labeling initially and existentially guessed by *check*. Evidently, after the top calls to *checkTrue*, each configuration of the procedure *check* can be described by a tuple $(\ell, Lab, \mathscr{W}, f)$, where: (i) $\mathscr{W}$ is a well-formed set for $(\mathcal{K},\varphi)$, (ii) $f = \mathtt{true}$ if $\mathscr{W}$ is processed within *checkTrue*, and $f = \mathtt{false}$ otherwise, and (iii) ℓ is an instruction label corresponding to one of the instructions of the procedures *checkTrue* and

checkFalse$_{(\mathcal{K},\varphi,Lab)}(\mathscr{W})$ [$\mathscr{W}$ *is a well-formed set and Lab is an* $\mathsf{A}\overline{\mathsf{A}}$*-labeling for* $(\mathcal{K},\varphi)$]

`while` $\mathscr{W}$ is *not* universal `do`
  `deterministically select` $(\psi,\rho)\in\mathscr{W}$ s.t. ψ is not of the form $[\overline{E}]\psi'$ and $[\overline{B}]\psi'$
  `update` $\mathscr{W}\leftarrow\mathscr{W}\setminus\{(\psi,\rho)\}$;
  `case` $\psi=r$ with $r\in\mathsf{RE}$: `if` $\rho\notin\mathscr{L}(r)$ `then` *accept the input*;
  `case` $\psi=\neg r$ with $r\in\mathsf{RE}$: `if` $\rho\in\mathscr{L}(r)$ `then` *accept the input*;
  `case` $\psi=\langle \mathrm{A}\rangle\psi'$ or $\psi=[A]\psi'$: `if` $\psi\notin Lab(\mathrm{lst}(\rho))$ `then` *accept the input*;
  `case` $\psi=\langle \overline{\mathrm{A}}\rangle\psi'$ or $\psi=[\overline{A}]\psi'$: `if` $\psi\notin Lab(\mathrm{fst}(\rho))$ `then` *accept the input*;
  `case` $\psi=\psi_1\vee\psi_2$: `universally choose` $i=1,2$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi_i,\rho)\}$;
  `case` $\psi=\psi_1\wedge\psi_2$: `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi_1,\rho),(\psi_2,\rho)\}$;
  `case` $\psi=\langle \mathrm{B}\rangle\psi'$: `universally choose` $\rho'\in\mathrm{Pref}(\rho)$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\}$;
  `case` $\psi=[B]\psi'$: `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\mid\rho'\in\mathrm{Pref}(\rho)\}$;
  `case` $\psi=\langle X\rangle\psi'$ with $X\in\{\overline{E},\overline{B}\}$: `universally choose` an *X-witness* ρ' of ρ for $(\mathcal{K},\varphi)$, `update` $\mathscr{W}\leftarrow\mathscr{W}\cup\{(\psi',\rho')\}$;
`end while`
`if` $\mathscr{W}=\emptyset$ `then` *reject the input*
`else existentially choose` $(\psi,\rho)\in\widetilde{\mathscr{W}}$ and `call` *checkTrue*$_{(\mathcal{K},\varphi,Lab)}(\{(\psi,\rho)\})$

Figure 3: Procedure *checkFalse*

checkFalse. We denote by ℓ_0 the label associated with the while instruction. A *main configuration* is a configuration associated with the label ℓ_0. Let $Lab_{\mathscr{W}}$ be the restriction of Lab to the set of formulas in $\mathsf{A}\overline{\mathsf{A}}(\varphi)$ which are subformulas of formulas occurring in $\mathscr{W}$. In other terms, for each state s, $Lab_{\mathscr{W}}(s)$ contains all and only the formulas $\psi\in Lab(s)$ such that either ψ or its dual $\widetilde{\psi}$ is a subformula of some formula occurring in $\mathscr{W}$. $Lab_{\mathscr{W}}$ is *valid* if for all states s and $\psi\in Lab_{\mathscr{W}}(s)$, $\mathcal{K},s\models\psi$.

Claim 3 Let $\mathscr{W}$ be a well-formed set for $(\mathcal{K},\varphi)$ and assume that $Lab_{\mathscr{W}}$ is valid. Then:

1. the main configuration $(\ell_0,Lab,\mathscr{W},\mathtt{true})$ leads to acceptance *iff* $\mathscr{W}$ is valid;
2. the main configuration $(\ell_0,Lab,\mathscr{W},\mathtt{false})$ leads to acceptance *iff* $\mathscr{W}$ is *not* valid.

Proof of Claim 3. We associate with $\mathscr{W}$ a natural number $\|\mathscr{W}\|$ defined as follows. Fix an ordering $\psi_1,\ldots,\psi_k$ of the formulas in $\mathsf{SD}(\varphi)$ such that for all $i\neq j$, $|\psi_i|>|\psi_j|$ implies $i<j$. First, we associate to $\mathscr{W}$ a $(k+1)$-tuple $(n_0,n_1,\ldots,n_k)$ of natural numbers defined as follows: the first component n_0 in the tuple is the alternation depth $\Upsilon_w(\mathscr{W})$ and for the other components n_i with $1\leq i\leq k$, n_i is the number of elements of $\mathscr{W}$ associated with the formula ψ_i (i.e., the number of elements of the form (ψ_i,ρ)). Then, $\|\mathscr{W}\|$ is the position of the tuple $(n_0,n_1,\ldots,n_k)$ along the total lexicographic ordering over $\mathbb{N}^{k+1}$. Note that if $\mathscr{W}$ is non-empty and universal, then since $\Upsilon_w(\widetilde{\mathscr{W}})<\Upsilon_w(\mathscr{W})$, it holds that $\|\widetilde{\mathscr{W}}\|<\|\mathscr{W}\|$. Moreover, note that $\|\mathscr{W}\|$ strictly decreases at each iteration of the while loop in the procedures *checkTrue* and *checkFalse* (this because at each iteration, $\Upsilon_w(\mathscr{W})$ does not increase and an element of $\mathscr{W}$ is replaced with elements associated with smaller formulas).

The proof of Claim 3 is given by induction on $\|\mathscr{W}\|$. For the base case, $\|\mathscr{W}\|=0$, hence $\mathscr{W}$ is empty and evidently valid. By construction, *checkTrue* accepts the empty set, while *checkFalse* rejects the empty set. Hence, for the base case, the result holds. For the inductive step, let $\|\mathscr{W}\|>0$, hence $\mathscr{W}$ is not empty. First, assume that $\mathscr{W}$ is universal. Recall that $\|\widetilde{\mathscr{W}}\|<\|\mathscr{W}\|$. Then:

- Property 1: $\mathscr{W}$ is valid $\Longleftrightarrow$ for each $(\psi,\rho) \in \widetilde{\mathscr{W}}$, $\{(\psi,\rho)\}$ is not valid $\Longleftrightarrow$ (by the induction hypothesis) for each $(\psi,\rho) \in \widetilde{\mathscr{W}}$, the main configuration $(\ell_0, Lab, \{(\psi,\rho)\}, \texttt{false})$ leads to acceptance $\Longleftrightarrow$ (by construction of the algorithm and since $\mathscr{W}$ is universal) the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{true})$ leads to acceptance.
- Property 2: $\mathscr{W}$ is *not* valid $\Longleftrightarrow$ for some $(\psi,\rho) \in \widetilde{\mathscr{W}}$, $\{(\psi,\rho)\}$ is valid $\Longleftrightarrow$ (by the induction hypothesis) for some $(\psi,\rho) \in \widetilde{\mathscr{W}}$, the main configuration $(\ell_0, Lab, \{(\psi,\rho)\}, \texttt{true})$ leads to acceptance $\Longleftrightarrow$ (by construction of the algorithm and since $\mathscr{W}$ is universal) the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{false})$ leads to acceptance.

Hence, Properties 1 and 2 of Claim 3 hold if $\mathscr{W}$ is universal. Now, assume that the non-empty set $\mathscr{W}$ is not universal. We consider Property 2 of Claim 3 (the proof of Property 1 being dual). Let $(\psi,\rho) \in \mathscr{W}$ be the pair selected by the procedure *checkFalse* in the iteration of the while loop associated with the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{false})$. Here, we examine the cases where either $\psi = \langle \mathrm{A} \rangle \psi'$, or $\psi = [B]\psi'$ or $\psi = \langle X \rangle \psi'$ with $X \in \{\overline{B}, \overline{E}\}$ (the other cases being similar or simpler).

- Case $\psi = \langle \mathrm{A} \rangle \psi'$. We have that $\{(\langle \mathrm{A} \rangle \psi', \rho)\}$ is valid iff $\mathcal{K}, \mathrm{lst}(\rho) \models \langle \mathrm{A} \rangle \psi'$. By hypothesis, $Lab_{\mathscr{W}}$ is valid. Hence, $\{(\langle \mathrm{A} \rangle \psi', \rho)\}$ is not valid iff $\langle \mathrm{A} \rangle \psi' \notin Lab_{\mathscr{W}}(\mathrm{lst}(\rho))$. Let $\mathscr{W}' = \mathscr{W} \setminus \{(\psi,\rho)\}$. Note that $\|\mathscr{W}'\| < \|\mathscr{W}\|$. Then, we have that $\mathscr{W}$ is not valid $\Longleftrightarrow$ either $\langle \mathrm{A} \rangle \psi' \notin Lab_{\mathscr{W}}(\mathrm{lst}(\rho))$ or $\mathscr{W}'$ is not valid $\Longleftrightarrow$ (by the induction hypothesis) either $\langle \mathrm{A} \rangle \psi' \notin Lab_{\mathscr{W}}(\mathrm{lst}(\rho))$ or the main configuration $(\ell_0, Lab, \mathscr{W}', \texttt{false})$ leads to acceptance $\Longleftrightarrow$ (by construction of *checkFalse*) the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{false})$ leads to acceptance.
- Case $\psi = [B]\psi'$. Let $\mathscr{W}' = (\mathscr{W} \setminus \{(\psi,\rho)\}) \cup \{(\psi', \rho') \mid \rho' \in \mathrm{Pref}(\rho)\}$. Note that $\|\mathscr{W}'\| < \|\mathscr{W}\|$. Then,we have that $\mathscr{W}$ is not valid $\Longleftrightarrow$ $\mathscr{W}'$ is not valid $\Longleftrightarrow$ (by the induction hypothesis) the main configuration $(\ell_0, Lab, \mathscr{W}', \texttt{false})$ leads to acceptance $\Longleftrightarrow$ (by construction of *checkFalse*) the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{false})$ leads to acceptance.
- Case $\psi = \langle X \rangle \psi'$ with $X \in \{\overline{B}, \overline{E}\}$. By Proposition 16(1), $\mathcal{K}, \rho \models \langle X \rangle \psi'$ iff there exists an X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$ such that $\mathcal{K}, \rho' \models \psi'$. Then, Property 2 of Claim 3 directly follows from the following chain of equivalences: $\mathscr{W}$ is not valid $\Longleftrightarrow$ either $\mathscr{W} \setminus \{(\psi,\rho)\}$ is not valid, or for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, $\{(\psi', \rho')\}$ is not valid $\Longleftrightarrow$ for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, $(\mathscr{W} \setminus \{(\psi,\rho)\}) \cup \{(\psi', \rho')\}$ is not valid $\Longleftrightarrow$ (by the induction hypothesis) for each X-witness ρ' of ρ for $(\mathcal{K}, \varphi)$, the main configuration $(\ell_0, Lab, (\mathscr{W} \setminus \{(\psi,\rho)\}) \cup \{(\psi', \rho')\}, \texttt{false})$ leads to acceptance $\Longleftrightarrow$ (by construction of the procedure *checkFalse*) the main configuration $(\ell_0, Lab, \mathscr{W}, \texttt{false})$ leads to acceptance.

This concludes the proof of Claim 3. □

By exploiting Claim 3, we now prove the following result, which concludes the proof of Proposition 17.

Claim 4. The ATM *check* accepts an input $(\mathcal{K}, \varphi)$ iff $\mathcal{K} \models \varphi$.
Proof of Claim 4: fix an input $(\mathcal{K}, \varphi)$ and an $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* for $(\mathcal{K}, \varphi)$. A *Lab-guessing* for $(\mathcal{K}, \varphi)$ is a well-formed set $\mathscr{W}$ for $(\mathcal{K}, \varphi)$ which minimally satisfies the following conditions for all states s of $\mathcal{K}$:

- for all certificates ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s_0$, $(\varphi, \rho) \in \mathscr{W}$;
- for all $\langle A \rangle \psi \in Lab(s)$ (resp., $\langle \overline{A} \rangle \psi \in Lab(s)$), there is a certificate ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$ (resp., $\mathrm{lst}(\rho) = s$) such that $(\psi, \rho) \in \mathscr{W}$;
- for all $[A]\psi \in Lab(s)$ (resp., $[\overline{A}]\psi \in Lab(s)$) and for all certificates ρ for $(\mathcal{K}, \varphi)$ with $\mathrm{fst}(\rho) = s$ (resp., $\mathrm{lst}(\rho) = s$), $(\psi, \rho) \in \mathscr{W}$;

Evidently, by construction of the procedure *check*, for each input $(\mathcal{K}, \varphi)$, it holds that:

- (*) Procedure *check* accepts $(\mathcal{K},\varphi) \iff$ there exists an $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* and a *Lab*-guessing $\mathscr{W}$ for $(\mathcal{K},\varphi)$ such that for all $(\psi,\rho) \in \mathscr{W}$, the main configuration (ℓ_0, *Lab*, $\{(\psi,\rho)\}$,`true`) leads to acceptance.

Fix an input $(\mathcal{K},\varphi)$. First, assume that $\mathcal{K} \models \varphi$. Let *Lab* be the *valid* $\mathsf{A}\overline{\mathsf{A}}$-labeling defined as follows for all states s: for all $\psi \in \mathsf{A}\overline{\mathsf{A}}(\varphi)$, $\psi \in Lab(s)$ iff $\mathcal{K},s \models \psi$. By Theorem 14, there exists a *Lab*-guessing $\mathscr{W}$ for $(\mathcal{K},\varphi)$ such that for all $(\psi,\rho) \in \mathscr{W}$, $\mathcal{K},\rho \models \psi$. By Claim 3, for all $(\psi,\rho) \in \mathscr{W}$, the main configuration (ℓ_0, *Lab*,$\{(\psi,\rho)\}$,`true`) leads to the acceptance. Hence, by Condition (*), procedure *check* accepts $(\mathcal{K},\varphi)$.

For the converse direction, assume that procedure *check* accepts $(\mathcal{K},\varphi)$. By Condition (*), there exists an $\mathsf{A}\overline{\mathsf{A}}$-labeling *Lab* and a *Lab*-guessing $\mathscr{W}$ for $(\mathcal{K},\varphi)$ such that for all $(\psi,\rho) \in \mathscr{W}$, the main configuration (ℓ_0,*Lab*,$\{(\psi,\rho)\}$,`true`) leads to acceptance. First, we show that *Lab* is valid. Fix a state s and a formula $\psi \in Lab(s)$. We need to prove that $\mathcal{K},s \models \psi$. The proof is by induction on the nesting depth $d_{\mathsf{A}\overline{\mathsf{A}}}(\psi)$ of modalities $\langle \mathrm{A} \rangle$, $\langle \overline{\mathrm{A}} \rangle$, $[A]$, and $[\overline{A}]$ in ψ. Assume that $\psi = [A]\psi'$ for some ψ' (the other cases, where either $\psi = \langle \mathrm{A} \rangle \psi'$, or $\psi = \langle \overline{\mathrm{A}} \rangle \psi'$ or $\psi = [\overline{A}]\psi'$ being similar). By definition of *Lab*-guessing, for each certificate ρ for $(\mathcal{K},\varphi)$ with $\mathrm{fst}(\rho) = s$, $(\psi',\rho) \in \mathscr{W}$. Moreover, by the induction hypothesis, one can assume that $Lab_{\{(\psi',\rho)\}}$ is valid (note that for the base case, i.e. when ψ' does not contain occurrences of modalities $\langle \mathrm{A} \rangle$, $\langle \overline{\mathrm{A}} \rangle$, $[A]$, and $[\overline{A}]$, $Lab_{\{(\psi',\rho)\}}$ is trivially valid). By hypothesis, the main configuration (ℓ_0,*Lab*,$\{(\psi',\rho)\}$,`true`) leads to acceptance. By Claim 3, it follows that for each certificate ρ for $(\mathcal{K},\varphi)$ with $\mathrm{fst}(\rho) = s$, it holds that $\mathcal{K},\rho \models \psi'$. Thus, by Theorem 14, we obtain that $\mathcal{K},s \models \psi$. Hence, *Lab* is valid. By definition of *Lab*-guessing, for each certificate ρ for $(\mathcal{K},\varphi)$ with $\mathrm{fst}(\rho) = s_0$, $(\varphi,\rho) \in \mathscr{W}$. Thus, by hypothesis, Claim 3, and Theorem 14, we obtain that $\mathcal{K} \models \varphi$. This concludes the proof of Claim 4 and Proposition 17 as well. □

C Proofs from Section 5

C.1 Proof of Theorem 18

In this section, we show that the alternating multi-tiling problem is $\mathbf{AEXP_{pol}}$-complete. The membership in $\mathbf{AEXP_{pol}}$ can be easily proved. Thus, we focus on the hardness result which is the relevant one in this context. We establish the $\mathbf{AEXP_{pol}}$-hardness of the alternating multi-tiling problem in two steps. In the first step, we consider a variant of the considered problem, called *TM alternation problem*, which is defined in terms of multi-tape deterministic Turing machines, and we prove that this problem is $\mathbf{AEXP_{pol}}$-hard. Then, in the second step, we provide a polynomial-time reduction from the TM alternation problem to the alternating multi-tiling problem.

C.1.1 TM alternation problem

Let $n \geq 1$. An *n-ary deterministic Turing Machine* (TM, for short) is a deterministic Turing machine $\mathscr{M} = (n,I,A,Q,\{q_{acc},q_{rej}\},q_0,\delta)$ operating on n ordered semi-infinite tapes and having *only one* read/write head (shared by all tapes), where: I (resp., $A \supset I$) is the input (resp., work) alphabet, A contains the blank symbol $\# \notin I$, Q is the set of states, q_{acc} (resp., q_{rej}) is the terminal accepting (resp., rejecting) state, q_0 is the initial state, and $\delta : Q \times A \mapsto \{\bot\} \cup (Q \times A \times \{\leftarrow,\rightarrow\}) \cup (Q \times \{\mathsf{prev},\mathsf{next}\})$ is the transition function, where the symbol $\bot$ is for 'undefined' and for all $(q,a) \in Q \times A$, $\delta(q,a) = \bot$ iff $q \in \{q_{acc},q_{rej}\}$. In each non-terminal step, if the read/write head scans the kth cell from the left of the ℓth tape ($\ell \in [1,n]$ and $k \geq 1$) and $(q,a) \in (Q \setminus \{q_{acc},q_{rej}\}) \times A$ is the current pair state/scanned cell content, one of the following occurs:

- $\delta(q,a) \in Q \times A \times \{\leftarrow, \rightarrow\}$ (*ordinary moves*): $\mathscr{M}$ overwrites the tape cell being scanned, there is a change of state, and the read/write head moves one position to the left ($\leftarrow$) or right ($\rightarrow$) in accordance with $\delta(q,a)$.[1]
- $\delta(q,a) \in Q \times \{\mathsf{prev}, \mathsf{next}\}$ (*jump moves*): if $\delta(q,a) = (q', \mathsf{prev})$ (resp., $\delta(q,a) = (q', \mathsf{next})$) for some $q' \in Q$, then the read/write head jumps to the kth cell of the $(\ell-1)$th tape (resp., $(\ell+1)$th tape) and $\mathscr{M}$ moves to state q' if $\ell > 1$ (resp., $\ell < n$); otherwise, $\mathscr{M}$ moves to the rejecting state.

Initially, each tape contains a word in I^* and the read/write head points to the left-most cell of the first tape. Thus, an input of $\mathscr{M}$, called n-ary input, can be described by a tuple $(w_1, \ldots, w_n) \in (I^*)^n$, where for all $i \in [1,n]$, w_i represent the initial content of the ith tape. $\mathscr{M}$ *accepts* a n-ary input $(w_1, \ldots, w_n) \in (I^*)^n$, written $\mathscr{M}(w_1, \ldots, w_n)$, if the computation of $\mathscr{M}$ from $(w_1, \ldots, w_n)$ is accepting. We consider the following problem.

TM Alternation Problem. An instance of the problem is a tuple $(n, \mathscr{M})$, where $n > 1$ and $\mathscr{M}$ is a *polynomial-time bounded* n-ary deterministic Turing Machine with input alphabet I. The instance $(n, \mathscr{M})$ is *positive* iff the following holds, where $\mathsf{Q}_\ell = \exists$ if ℓ is odd, and $\mathsf{Q}_\ell = \forall$ otherwise (for all $\ell \in [1,n]$),

$$\mathsf{Q}_1 x_1 \in I^{2^n}. \mathsf{Q}_2 x_2 \in I^{2^n}. \ldots \mathsf{Q}_n x_n \in I^{2^n}. \mathscr{M}(x_1, \ldots, x_n)$$

Note that the quantifications Q_i are restricted to words over I of length 2^n. Thus, even if $\mathscr{M}$ is polynomial-time bounded, it operates on an input whose size is exponential in n.

Proposition 25. *The TM Alternation Problem is* $\mathbf{AEXP_{pol}}$*-complete.*

The proof of Proposition 25 is standard. However, for completeness, we give a proof of the hardness result in Proposition 25, which is the relevant one in this context. In particular, the lower bound in Proposition 25 directly follows from the following lemma.

Lemma 26. *Let $\mathscr{M}_\mathscr{A}$ be a singly exponential-time bounded ATM with a polynomial bounded number of alternations. Moreover, let $c \geq 1$ and $c_a \geq 1$ be integer constants such that for each input α, when started on α, $\mathscr{M}_\mathscr{A}$ has at most $|\alpha|^{c_a}$ alternations and $\mathscr{M}_\mathscr{A}$ reaches a terminal configuration in at most $2^{|\alpha|^c}$ steps. Then, given an input α, one can construct in time polynomial in α and in the size of $\mathscr{M}_\mathscr{A}$ an instance $(2|\alpha|^{max\{c,c_a\}}, \mathscr{M})$ of the* TM Alternation Problem *such that the instance is positive* iff *$\mathscr{M}_\mathscr{A}$ accepts α.*

Proof. Let $\mathscr{M}_\mathscr{A}$, c, and c_a be as in the statement of Lemma 26. Let $I_\mathscr{A}$ (resp., $A_\mathscr{A}$) be the input (resp., work) alphabet of $\mathscr{M}_\mathscr{A}$, where $I_\mathscr{A} \subset A_\mathscr{A}$, and Q be the set of $\mathscr{M}_\mathscr{A}$-states. Without loss of generality, we assume that the initial state of $\mathscr{M}_\mathscr{A}$ is existential. Fix an input $\alpha \in I_\mathscr{A}^*$. Define $k := \max\{c, c_a\}$ and $n := 2|\alpha|^k$. An α-configuration is a word in $A_\mathscr{A}^* \cdot (Q \times A_\mathscr{A}) \cdot A_\mathscr{A}^*$ of length exactly $2^{|\alpha|^k}$. Note that any configuration of $\mathscr{M}_\mathscr{A}$ reachable from the input α can be encoded by an α-configuration. We denote by C_α the initial (existential) α-configuration associated with the input α. A *partial computation* of $\mathscr{M}_\mathscr{A}$ is a finite sequence $\pi = C_1, \ldots, C_p$ of α-configurations such that $p \leq 2^{|\alpha|^k}$ and for each $1 \leq i < p$, C_{i+1} is a $\mathscr{M}_\mathscr{A}$-successor of C_i (note that a computation of $\mathscr{M}_\mathscr{A}$ over α is a partial computation). We say that π is *uniform* if additionally one of the following holds:

- C_p is terminal and π visits only existential n-configurations;
- C_p is terminal and π visits only universal n-configurations;
- $p > 1$, C_p is existential and for each $1 \leq h < p$, C_h is universal;

[1] If the read/write head points to the left-most cell of the ℓth tape and $\delta(q,a)$ is of the form $(q', a, \leftarrow)$, then $\mathscr{M}$ moves to the rejecting state.

- $p > 1$, C_p is universal and for each $1 \leq h < p$, C_h is existential.

Let $\diamondsuit$ be a fresh symbol and $I = A_{\mathscr{A}} \cup \{\diamondsuit\}$. The *code* of a partial computation $\pi = C_1, \ldots, C_p$ is the word over I of length exactly 2^n (recall that $n = 2|\alpha|^k$) given by $C_1, \ldots, C_p, C^0_{p+1}, \ldots, C^0_{2^{|\alpha|^k}}$, where $C^0_i \in \{\diamondsuit\}^{2^{|\alpha|^k}}$ for all $p+1 \leq i \leq 2^{|\alpha|^k}$. We construct a polynomial-time bounded n-ary deterministic Turing Machine $\mathscr{M}$, which satisfies Lemma 26 for the given input α of $\mathscr{M}_{\mathscr{A}}$. The input alphabet of $\mathscr{M}$ is I. Given a n-ary input $(w_1, \ldots, w_n) \in (I^*)^n$, $\mathscr{M}$ operates in n-steps (macro steps). At step i ($i \in [1,n]$), the behavior of $\mathscr{M}$ is as follows, where for a partial computation $\pi = C_1, \ldots, C_p$, $\mathsf{first}(\pi) = C_1$ and $\mathsf{last}(\pi) = C_p$:

- *First step:* $i = 1$.
  1. If $w_1 \in I^{2^n}$ and w_1 encodes a uniform partial computation π_1 of $\mathscr{M}_{\mathscr{A}}$ *from* C_α, then the behavior is as follows. If $\mathsf{last}(\pi_1)$ is accepting (resp., rejecting), then $\mathscr{M}$ accepts (resp., rejects) the input. Conversely, if $\mathsf{last}(\pi_1)$ is not a terminal configuration, then $\mathscr{M}$ goes to step $i+1$.
  2. Otherwise, $\mathscr{M}$ *rejects* the input.
- $i > 1$.
  1. If $w_i \in I^{2^n}$ and w_i encodes a uniform partial computation π_i of $\mathscr{M}_{\mathscr{A}}$ such that $\mathsf{first}(\pi_i) = \mathsf{last}(\pi_{i-1})$, where π_{i-1} is the uniform partial computation encoded by w_{i-1}, then the behavior is as follows. If $\mathsf{last}(\pi_i)$ is accepting (resp., rejecting), then $\mathscr{M}$ accepts (resp., rejects) the input. If instead $\mathsf{last}(\pi_i)$ is not a terminal configuration, then $\mathscr{M}$ goes to step $i+1$, if $i+1 \leq n$, and rejects the input otherwise.
  2. Otherwise, if i is odd (resp., even), then $\mathscr{M}$ *rejects* (resp., *accepts*) the input.

Note that Conditions 1 in the steps above can be checked by $\mathscr{M}$ in polynomial time (in the size of the input) by using the transition function of $\mathscr{M}_{\mathscr{A}}$ and n-bit counters. Hence, $\mathscr{M}$ is a polynomial-time bounded n-ary deterministic Turing Machine which, evidently, can be constructed in time polynomial in n and in the size of $\mathscr{M}_{\mathscr{A}}$. Now, we prove that the construction is correct, i.e. $(n, \mathscr{M})$ is a positive instance of the TM Alternation Problem iff $\mathscr{M}_{\mathscr{A}}$ accepts α. For each $\ell \in [1,n]$, let $\mathsf{Q}_\ell = \exists$ if ℓ is odd, and $\mathsf{Q}_\ell = \forall$ otherwise. Since C_α is existential, $\mathscr{M}_{\mathscr{A}}$ accepts α iff there is a uniform partial computation π_1 of $\mathscr{M}_{\mathscr{A}}$ from C_α such that $\mathsf{last}(\pi_1)$ leads to acceptance. Moreover, for each $w_1 \in I^{2^n}$, $\mathscr{M}$ accepts an input of the form $(w_1, w'_2, \ldots, w'_k)$ *only if* w_1 encodes a non-rejecting uniform partial computation of $\mathscr{M}_{\mathscr{A}}$ from C_α. Thus, since $\mathsf{Q}_1 = \exists$, correctness of the construction directly follows from the following claim.

Claim. Let $\ell \in [1,n]$ and $\pi = \pi_1 \ldots \pi_\ell$ be a partial computation of $\mathscr{M}_{\mathscr{A}}$ from C_α such that π_ℓ is uniform and for each $1 \leq j < \ell$, π_j is non-empty and $\pi_j \cdot \mathsf{first}(\pi_{j+1})$ is uniform as well. Let $w_\ell \in I^{2^n}$ be the word encoding π_ℓ and for each $1 \leq j < \ell$, $w_j \in I^{2^n}$ be the word encoding $\pi_j \cdot \mathsf{first}(\pi_{j+1})$. Then, $\mathsf{last}(\pi_\ell)$ leads to acceptance in $\mathscr{M}_{\mathscr{A}}$ if and only if

$$\mathsf{Q}_{\ell+1} x_{\ell+1} \in I^{2^n} \ldots \mathsf{Q}_n x_n \in I^{2^n} . \mathscr{M}(w_1, \ldots, w_\ell, x_{\ell+1}, \ldots, x_n) \tag{1}$$

Proof of the claim. The proof is by induction on $n - \ell$.

Base Step: $\ell = n$. Note that in this case $\mathsf{last}(\pi_n)$ is a terminal configuration of $\mathscr{M}_{\mathscr{A}}$ (otherwise, the number of alternations of existential and universal configurations along π would be greater than $n - 1 \geq |\alpha|^{c_a}$). Thus, we need to show that $\mathsf{last}(\pi_n)$ is accepting iff $\mathscr{M}(w_1, \ldots, w_n)$. By construction, when started on the input $(w_1, \ldots, w_n)$, $\mathscr{M}$ reaches the nth step and Condition 1 in this step is satisfied. Moreover, either $\mathsf{last}(\pi_n)$ is accepting and $\mathscr{M}$ accepts the input $(w_1, \ldots, w_n)$, or $\mathsf{last}(\pi_n)$ is rejecting and $\mathscr{M}$ rejects the input $(w_1, \ldots, w_n)$. Hence, the result follows.

Induction Step: $\ell < n$. First, assume that $\mathsf{last}(\pi_\ell)$ is a terminal configuration. By construction on any input of the form $(w_1,\dots,w_\ell,w'_{\ell+1},\dots,w'_n)$, $\mathscr{M}$ reaches the ℓth step and Condition 1 in this step is satisfied. Moreover, either $\mathsf{last}(\pi_\ell)$ is accepting and $\mathscr{M}$ accepts the input $(w_1,\dots,w_\ell,w'_{\ell+1},\dots,w'_n)$, or $\mathsf{last}(\pi_\ell)$ is rejecting and $\mathscr{M}$ rejects the input $(w_1,\dots,w_\ell,w'_{\ell+1},\dots,w'_n)$. Hence, in this case the result holds. Now, assume that $\mathsf{last}(\pi_\ell)$ is not terminal. We consider the case where $\ell+1$ is even (the other case being similar). Then, $\mathsf{Q}_{\ell+1} = \forall$. Since C_α is existential and $\mathsf{last}(\pi_\ell)$ is not terminal, by hypothesis, $\mathsf{last}(\pi_\ell)$ must be a universal configuration. First, assume that $\mathsf{last}(\pi_\ell)$ leads to acceptance. Let $w_{\ell+1} \in I^{2^n}$. By construction on any input of the form $(w_1,\dots,w_\ell,w_{\ell+1},w'_{\ell+2}\dots,w'_n)$, $\mathscr{M}$ reaches the $(\ell+1)$th step. If $w_{\ell+1}$ satisfies Condition 2 in this step, then since $\ell+1$ is even, $\mathscr{M}$ accepts the input. Hence, $\mathsf{Q}_{\ell+2}x_{\ell+2} \in I^{2^n}.\dots\mathsf{Q}_n x_n \in I^{2^n}.\mathscr{M}(w_1,\dots,w_\ell,w_{\ell+1},x_{\ell+2},\dots,x_n)$. Otherwise, $w_{\ell+1}$ encodes a uniform partial computation $\pi_{\ell+1}$ of $\mathscr{M}_\mathscr{A}$ from $\mathsf{last}(\pi_\ell)$. Since $\mathsf{last}(\pi_\ell)$ leads to acceptance and $\mathsf{last}(\pi_\ell)$ is universal, $\mathsf{last}(\pi_{\ell+1})$ leads to acceptance as well. Thus, by applying the inductive hypothesis to the partial computation $\pi_1\dots\pi_{\ell-1}\pi'_\ell\pi_{\ell+1}$ (where π'_ℓ is obtained from π_ℓ by removing $\mathsf{last}(\pi_\ell)$), it follows that $\mathsf{Q}_{\ell+2}x_{\ell+2} \in I^{2^n}.\dots\mathsf{Q}_n x_n \in I^{2^n}.\mathscr{M}(w_1,\dots,w_\ell,w_{\ell+1},x_{\ell+2},\dots,x_n)$. Thus, the previous condition holds for each $w_{\ell+1} \in I^{2^n}$. Since $\mathsf{Q}_{\ell+1} = \forall$, it follows that Equation (1) holds. For the converse direction, assume that Equation (1) holds. Let $\pi_{\ell+1}$ be any uniform partial computation of $\mathscr{M}_\mathscr{A}$ from $\mathsf{last}(\pi_\ell)$. We need to show that $\mathsf{last}(\pi_{\ell+1})$ leads to acceptance. Since Equation (1) holds and $\mathsf{Q}_{\ell+1} = \forall$, we can apply the inductive hypothesis to the partial computation $\pi_1\dots\pi_{\ell-1}\pi'_\ell\pi_{\ell+1}$ (where π'_ℓ is obtained from π_ℓ by removing $\mathsf{last}(\pi_\ell)$). Hence, the result follows. This concludes the proof of the claim and Lemma 26 as well. □

C.1.2 $\mathbf{AEXP_{pol}}$-hardness of the alternating multi-tiling problem

We show the $\mathbf{AEXP_{pol}}$-hardness of the alternating multi-tiling problem by a polynomial time reduction from the TM alternation problem. Fix an instance $(n,\mathscr{M})$ of the TM alternation problem where $\mathscr{M} = (n,I,A,Q,\{q_{acc},q_{rej}\},q_0,\delta)$ is a polynomial-time bounded n-ary deterministic Turing Machine.

Remark 27 (Assumptions on $\mathscr{M}$). In order to simplify the reduction, without loss of generality, we can assume that $\mathscr{M}$ satisfies the following constraints:

- n is even;
- for each n-ary input $(w_1,\dots,w_n) \in I^{2^n} \times \dots \times I^{2^n}$, $\mathscr{M}$ reaches a terminal configuration in exactly 2^n-1 steps, and when $\mathscr{M}$ halts, the read/write head points to a cell of the nth tape;
- there is no move leading to the initial state q_0;
- for all $a \in A$, $\delta(q_0,a) \in Q \times A \times \{\rightarrow\}$;
- for all $(q,a),(q',a') \in Q \times A$, if $\delta(q,a) \in \{q'\} \times \{\mathsf{prev},\mathsf{next}\}$, then $\delta(q',a') \notin Q \times \{\mathsf{prev},\mathsf{next}\}$.

We construct in polynomial time in the size of $(n,\mathscr{M})$ an instance $\mathscr{I}$ of the alternating multi-tiling problem such that $(n,\mathscr{M})$ is a positive instance of the TM alternation problem iff $\mathscr{I}$ is a positive instance of the alternating multi-tiling problem. Evidently, by the definitions of the considered problems, it suffices to show the following.

Proposition 28. *One can construct in time polynomial in the size of $(n,\mathscr{M})$, an instance $\mathscr{I} = (n,D,D_0,H,V,M,D_{acc})$ of the alternating multi-tiling problem such that $D_0 = I$ and the following holds: for each n-ary input $(w_1,\dots,w_n) \in I^{2^n} \times \dots \times I^{2^n}$, $\mathscr{M}(w_1,\dots,w_n)$ iff there exists a multi-tiling $F = (f_1,\dots,f_n)$ of $\mathscr{I}$ such that for all $\ell \in [1,n]$, the initial condition $Init(f_i)$ of the tiling f_i is w_i.*

Proof. We adapt the well-known translation between time-space diagrams of computations (also known as computation tableau) of a nondeterministic TM and tilings for a set of domino types entirely determined by the given TM. In such a translation, adjacent rows in the tiled region encode successive configurations in a computation of the machine.

Let $\lceil A \rceil$ be a fresh copy of the work alphabet A of $\mathscr{M}$. For a word w over A, we denote by $\lceil w \rceil$ the associated word over $\lceil A \rceil$. Define $U := A \times [1,n]$ and $\lceil U \rceil := \lceil A \rceil \times [1,n]$. We adopt the following set D of domino types:

$$D = I \cup U \cup \lceil U \rceil \cup (Q \times U) \cup ((Q \times \{\leftarrow, \rightarrow, \mathsf{prev}, \mathsf{next}\}) \times U)$$

Intuitively, in the encoding, we keep trace of the tape-indexes of $\mathscr{M}$. Moreover, the domino types in $(Q \times \{\leftarrow, \rightarrow\}) \times U$ (resp., $(Q \times \{\mathsf{prev}, \mathsf{next}\}) \times U$) are used to encode the effects of the ordinary moves (resp., jump moves). For each $d \in D$, we denote by $\mathsf{symb}(d)$ the associated letter in A (recall that $I \subset A$). Moreover, if $d \in D \setminus I$, we write $\mathsf{tape}(d)$ to mean the associated tape index $\ell \in [1,n]$. Additionally, if $d \in (Q \times U) \cup ((Q \times \{\leftarrow, \rightarrow, \mathsf{prev}, \mathsf{next}\}) \times U)$, we denote by $\mathsf{state}(d)$, the state $q \in Q$ associated with d. If instead $d \in I \cup U \cup \lceil U \rceil$, we set $\mathsf{state}(d) = \perp$ ($\perp$ is for 'undefined').

Given $\ell \in [1,n]$ and a word v over the alphabet

$$A \cup \lceil A \rceil \cup (Q \times A) \cup ((Q \times \{\leftarrow, \rightarrow, \mathsf{prev}, \mathsf{next}\}) \times A)$$

we write $v \oplus \ell$ to denote the word over the alphabet $D \setminus I$ defined in the obvious way.

Fix an n-ary input $(w_1, \ldots, w_n) \in I^{2^n} \times \ldots \times I^{2^n}$ and a non-rejecting configuration C of $\mathscr{M}$ reachable from the input $(w_1, \ldots, w_n)$. Assume that in C, the read/write head points to the kth cell of the ℓth tape for some $k \geq 1$ and $\ell \in [1,n]$, and let $(q,a) \in Q \times A$ be the pair state/scanned cell content associated with C. Since on the input $(w_1, \ldots, w_n)$, $\mathscr{M}$ halts in $2^n - 1$ steps, it holds that $k \leq 2^n$ and C can be encoded by the tuples of words in D^{2^n} of the form $(w_1^C \oplus 1, \ldots, w_n^C \oplus n)$ defined as follows:

- *cases* $q = q_{acc}$ *or* $\delta(q,a) \in Q \times A \times \{\leftarrow, \rightarrow\}$: for each $j \in [1,n] \setminus \{\ell\}$, $w_j^C = w_j$ or $w_j^C = \lceil w_j \rceil$ where w_j is the content of the first 2^n cells of the jth tape, and one of the following holds:
  - $q = q_{acc}$: w_ℓ^C is of the form $w' \cdot (q,a) \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot w''$ is the content of the first 2^n cells of the ℓth tape and $|w'| = k-1$ (the read/write head points to the kth cell of the ℓth tape);
  - $\delta(q,a) \in \{q'\} \times A \times \{\rightarrow\}$ for some $q' \in Q$: w_ℓ^C is of the form $w' \cdot (q,a) \cdot ((q', \rightarrow), a') \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot a' \cdot w''$ is the content of the first 2^n cells of the ℓth tape and $|w'| = k-1$;
  - $\delta(q,a) \in \{q'\} \times A \times \{\leftarrow\}$ for some $q' \in Q$: w_ℓ^C is of the form $w' \cdot ((q', \leftarrow), a') \cdot (q,a) \cdot \lceil w'' \rceil$, where $w' \cdot a' \cdot a \cdot w''$ is the content of the first 2^n cells of the ℓth tape and $|w'| = k-1$;
- *case* $\delta(q,a) \in \{q'\} \times \{\mathit{prev}\}$ for some $q' \in Q$:
  - for each $j \in [1,n] \setminus \{\ell, \ell-1\}$, $w_j^C = w_j$ or $w_j^C = \lceil w_j \rceil$ where w_j is the content of the first 2^n cells of the jth tape;
  - w_ℓ^C is of the form $w' \cdot (q,a) \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot w''$ is the content of the first 2^n cells of the ℓth tape and $|w'| = k-1$;
  - if $\ell > 1$, $w_{\ell-1}^C$ is of the form $w' \cdot ((q', \mathsf{prev}), a) \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot w''$ is the content of the first 2^n cells of the $(\ell-1)$th tape and $|w'| = k-1$;
- *case* $\delta(q,a) \in \{q'\} \times \{\mathit{next}\}$ for some $q' \in Q$:
  - for each $j \in [1,n] \setminus \{\ell, \ell+1\}$, $w_j^C = w_j$ or $w_j^C = \lceil w_j \rceil$ where w_j is the content of the first 2^n cells of the jth tape;
  - w_ℓ^C is of the form $w' \cdot (q,a) \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot w''$ is the content of the first 2^n cells of the ℓth tape and $|w'| = k-1$;
  - if $\ell < n$, $w_{\ell+1}^C$ is of the form $w' \cdot ((q', \mathsf{next}), a) \cdot \lceil w'' \rceil$, where $w' \cdot a \cdot w''$ is the content of the first 2^n cells of the $(\ell+1)$th tape and $|w'| = k-1$;

We construct in polynomial-time an instance $\mathscr{I} = (n, D, D_0, H, V, M, D_{acc})$ of the alternating multi-tiling problem with $D_0 = I$ and $D_{acc} = \{q_{acc}\} \times U$ such that for each n-ary input $(w_1, \ldots, w_n) \in I^{2^n} \times \ldots \times I^{2^n}$, $\mathscr{M}(w_1, \ldots, w_n)$ iff there exists a multi-tiling $F = (f_1, \ldots, f_n)$ of $\mathscr{I}$ such that for all $\ell \in [1,n]$, the initial condition of the tiling f_i is w_i. Moreover, if $\mathscr{M}(w_1, \ldots, w_n)$, then the following holds:

- let $\pi = C_1 \dots C_{2^n-1}$ be the accepting computation of $\mathscr{M}$ over $(w_1, \dots, w_n)$ (by our assumptions, the write/read head in the accepting configuration C_{2^n-1} points to the nth tape). Then, there exist multi-tilings $F = (f_1, \dots, f_n)$ of $\mathscr{I}$ associated with the input $(w_1, \dots, w_n)$ such that for all $j \in [1, 2^n - 1]$, there is an encoding $cod(C_j)$ of configuration C_j so that for all $\ell \in [1, n]$, the row of index j of f_ℓ coincides with the ℓth component of $cod(C_j)$.

We define the matching relations H, V and M in order to ensure the above conditions. In particular, the horizontal matching relation H guarantees that the TM configurations are correctly encoded, while the vertical matching relation is used to encode the ordinary moves of $\mathscr{M}$. Finally, the multi-tiling matching relation M is used to encode the jump moves. Additionally, H, V and M also ensure that the rows of index 1 encode the initial configuration of $\mathscr{M}$ associated with the given n-ary input (the latter corresponds to the tuple of rows of index 0).

Formally, H is the set of pairs $(d, d') \in D \times D$ satisfying the following constraints:

- $d \in I$ iff $d' \in I$;
- if $d \in D \setminus I$, then $d' \in D \setminus I$ and $\mathsf{tape}(d) = \mathsf{tape}(d')$;
- if $d, d' \in U \cup \lceil U \rceil$, then either $d, d' \in U$, or $d, d' \in \lceil U \rceil$;
- $\mathsf{state}(d') \neq q_0$, and whenever $\mathsf{state}(d) = q_0$, then $d \in \{q_0\} \times U$ and $\mathsf{tape}(d) = 1$;
- if $d \in \lceil U \rceil$ or $d \in (Q \times \{\rightarrow, \mathsf{prev}, \mathsf{next}\}) \times U$, then $d' \in \lceil U \rceil$;
- if $d \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \notin Q \times A \times \{\rightarrow\}$, then $d' \in \lceil U \rceil$;
- if $d' \in U$ or $d' \in (Q \times \{\leftarrow, \mathsf{prev}, \mathsf{next}\}) \times U$, then $d \in U$;
- if $d' \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \notin Q \times A \times \{\leftarrow\}$, then $d \in U$;
- for $q' \in Q$, $d \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in \{q'\} \times A \times \{\rightarrow\}$ *iff* $d' \in \{(q', \rightarrow)\} \times U$;
- for $q' \in Q$, $d' \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in \{q'\} \times A \times \{\leftarrow\}$ *iff* $d \in \{(q', \leftarrow)\} \times U$.

By definition of H (independently of the form of V and M), we deduce the following:

Claim 1: let f be a tiling of $\mathscr{I}$ and row be the content of any row of f. Then, either $\mathsf{row} \in I^*$ or $\mathsf{row} = \mathsf{row}' \oplus \ell$ for some $\ell \in [1, n]$ and row' satisfies one of the following:

- $\mathsf{row}' = w \cdot (q, a) \cdot \lceil w' \rceil$ such that $w, w' \in A^*$, and $\delta(q, a) \notin Q \times A \times \{\leftarrow, \rightarrow\}$;
- $\mathsf{row}' = w \cdot d \cdot \lceil w' \rceil$ such that $w, w' \in A^*$, and $d \in (Q \times \{\mathsf{prev}, \mathsf{next}\}) \times A$;
- $\mathsf{row}' = w$ such that $w \in A^*$;
- $\mathsf{row}' = \lceil w \rceil$ such that $w \in A^*$;
- $\mathsf{row}' = w \cdot d \cdot ((q', \rightarrow), a') \cdot \lceil w' \rceil$ such that $w, w' \in A^*$, $d \in Q \times A$, and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in \{q'\} \times A \times \{\rightarrow\}$;
- $\mathsf{row}' = w \cdot ((q', \leftarrow), a') \cdot d \cdot \lceil w' \rceil$ such that $w, w' \in A^*$, $d \in Q \times A$, and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in \{q'\} \times A \times \{\leftarrow\}$;

Moreover, if for some i, $\mathsf{state}(\mathsf{row}(i)) = q_0$, then $i = 0$, $\mathsf{row}(0) \in \{q_0\} \times U$, and $\mathsf{tape}(\mathsf{row}(0)) = 1$.

Now, let us define the vertical matching relation V. V is the set of pairs $(d, d') \in D \times D$ satisfying the following constraints:

- if $d \in I$ then $d' \in (\{q_0\} \times U) \cup U \cup \lceil U \rceil \cup (Q \times \{\rightarrow\}) \times U$ and $\mathsf{symb}(d) = \mathsf{symb}(d')$;
- if $d \in D \setminus I$, then $d' \in D \setminus I$ and $\mathsf{tape}(d) = \mathsf{tape}(d')$;
- if $d \in U \cup \lceil U \rceil$, then $d' \in U \cup \lceil U \rceil \cup ((Q \times \{\mathsf{prev}, \mathsf{next}\}) \times U)$ and $\mathsf{symb}(d) = \mathsf{symb}(d')$;
- if $d \in (Q \times \{\leftarrow, \rightarrow, \mathsf{prev}, \mathsf{next}\}) \times U$, then $d' = (\mathsf{state}(d), \mathsf{symb}(d), \mathsf{tape}(d))$;
- if $d \in Q \times U$, then $\mathsf{state}(d) \neq q_{acc}$ and one of the following holds:
  - $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in Q \times \{\mathsf{symb}(d')\} \times \{\leftarrow, \rightarrow\}$ and $d' \in U \cup \lceil U \rceil$;
  - $\delta(\mathsf{state}(d), \mathsf{symb}(d)) \in Q \times \{\mathsf{prev}, \mathsf{next}\}$, $d' \in U$, and $\mathsf{symb}(d') = \mathsf{symb}(d)$.

Finally, we define the multi-tiling matching relation M. M is the set of pairs $(d, d') \in D \times D$ satisfying the following constraints:

- $d \in I$ iff $d' \in I$;
- if $d \in D \setminus I$, then $d' \in D \setminus I$ and $\mathsf{tape}(d') = \mathsf{tape}(d) + 1$;
- $\mathsf{state}(d) \neq q_{acc}$ and $\mathsf{state}(d') \neq q_0$;
- for each $q \in Q$, $d \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) = (q, \mathsf{next})$ *iff* $d' \in \{(q, \mathsf{next})\} \times U$;
- for each $q \in Q$, $d' \in Q \times U$ and $\delta(\mathsf{state}(d), \mathsf{symb}(d)) = (q, \mathsf{prev})$ *iff* $d \in \{(q, \mathsf{prev})\} \times U$;
- if $d \in (Q \times \{\mathsf{next}\}) \times U$, then $\mathsf{tape}(d) > 1$;
- if $d' \in (Q \times \{\mathsf{prev}\}) \times U$, then $\mathsf{tape}(d') < n$.

By Claim 1 and the definitions of the matching relations V and M, one can prove that if F is a multi-tiling of $\mathscr{I}$ with initial conditions $(w_1, \ldots, w_n) \in I^{2^n} \times \ldots \times I^{2^n}$, then F encodes an accepting computation of $\mathscr{M}$ over the n-ary input $(w_1, \ldots, w_n) \in I^{2^n} \times \ldots \times I^{2^n}$. Vice versa, by Remark 27, if $\mathscr{M}(w_1, \ldots, w_n)$, then it easily follows that there exists a multi-tiling encoding the accepting computation of $\mathscr{M}$ over $(w_1, \ldots, w_n)$. □

C.2 Proof of Proposition 23

Proposition 23. *One can build, in time polynomial in the size of $\mathscr{I}$, $n+1$ $\mathsf{B}\overline{\mathsf{E}}$ formulas $\varphi_{IMT}, \varphi_1, \ldots, \varphi_n$ such that $\Upsilon(\varphi_{IMT}) = \Upsilon(\varphi_1) = \ldots = \Upsilon(\varphi_n) = 0$, and fulfilling the following conditions.*

- *For all finite words ρ over $\mathcal{AP}$ of the form $\rho = \rho' \cdot \bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $\bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot end$ is a multi-initialization code, $\mathcal{K}_\mathscr{I}, \rho \models \varphi_{IMT}$ if and only if ρ is an initialized multi-tiling code.*
- *For all $\ell \in [1,n]$ and words ρ of the form $\rho = \rho' \cdot \bot \cdot w_{\ell-1} \cdot \ldots \cdot \bot \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $w_j \in (\mathcal{AP} \setminus \{\bot\})^*$ for all $j \in [1, \ell-1]$, $\mathcal{K}_\mathscr{I}, \rho \models \varphi_\ell$ if and only if ρ' is of the form $\rho' = \bot \cdot w_\ell$, where w_ℓ is an initialization code.*

Proof. Since each state of the Kripke structure $\mathcal{K}_\mathscr{I}$ is labeled by exactly one proposition in $\mathcal{AP}$, in the proof, we exploit standard regular expressions where atomic formulas are single letters in $\mathcal{AP}$. Evidently, a standard regular expression can be converted into a propositional-based regular expression where each letter $p \in \mathcal{AP}$ is replaced with the propositional formula $p \wedge \bigwedge_{p' \in \mathcal{AP} \setminus \{p\}} \neg p'$. Now, we prove Proposition 23. We focus on the construction of the $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{IMT} (the construction of formulas $\varphi_1, \ldots, \varphi_n$ being simpler). First, we define a $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{MT} ensuring the following:

- For all finite words ρ over $\mathcal{AP}$ of the form $\rho = \rho' \cdot \bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot end$ such that $\rho' \neq \varepsilon$ and $\bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot end$ is a multi-initialization code, $\mathcal{K}_\mathscr{I}, \rho \models \varphi_{MT}$ if and only if $\rho' = \bot \cdot w$ for some multi-tiling code w.

In order to construct the $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{MT}, we need some auxiliary formulas.

- A regular expression r_{mc} capturing the multi-cell codes:

$$r_{mc} := D^n \cdot (\{r\} \times \{0,1\})^n \cdot (\{c\} \times \{0,1\})^n$$

- A B formula ψ_{comp} requiring that for each word $C \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot$ such that $C, C_1, \ldots, C_N$ are multi-cell codes, there is $i \in [1,N]$ such that C and C_i have the same row number and column number.

$$\begin{aligned} \psi_{comp} := \quad & \langle \mathrm{B} \rangle \Big((r_{mc} \cdot \bot \cdot (r_{mc})^+) \wedge \\ & \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{n+i-1} \cdot (r,b) \cdot \mathcal{AP}^+ \cdot (r,b) \cdot \mathcal{AP}^{2n-i}) \wedge \\ & \bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c,b) \cdot \mathcal{AP}^+ \cdot (c,b) \cdot \mathcal{AP}^{n-i}) \Big) \end{aligned}$$

- A propositional formula $\psi_{=}$ requiring that for each word having a proper prefix of the form $C \cdot C'$ such that C and C' are multi-cell codes, then C and C' have the same row number and column number.

$$\psi_{=} := \bigwedge_{i\in[1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{n+i-1} \cdot (r,b) \cdot \mathcal{AP}^{3n-1} \cdot (r,b) \cdot \mathcal{AP}^{+}) \wedge \bigwedge_{i\in[1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c,b) \cdot \mathcal{AP}^{3n-1} \cdot (c,b) \cdot \mathcal{AP}^{+})$$

- A propositional formula $\psi_{r,inc}$ (resp., $\psi_{c,inc}$) requiring that for each word having a proper prefix of the form $C \cdot C'$ such that C and C' are multi-cell codes, then C and C' have the same column number (resp., the same row number) and there is $h \in [0, 2^n - 2]$ such that C and C' have row numbers (resp., column numbers) h and $h+1$, respectively. We consider the formula $\psi_{r,inc}$ (the definition of $\psi_{c,inc}$ being similar).

$$\begin{aligned}\psi_{r,inc} := \; & \bigwedge_{i\in[1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c,b) \cdot \mathcal{AP}^{3n-1} \cdot (c,b) \cdot \mathcal{AP}^{+}) \wedge \\ & \bigvee_{i\in[1,n]} \Big(\bigwedge_{j\in[1,i-1]} (\mathcal{AP}^{n+j-1} \cdot (r,1) \cdot \mathcal{AP}^{3n-1} \cdot (r,0) \cdot \mathcal{AP}^{+}) \wedge \\ & \qquad (\mathcal{AP}^{n+i-1} \cdot (r,0) \cdot \mathcal{AP}^{3n-1} \cdot (r,1) \cdot \mathcal{AP}^{+}) \wedge \\ & \qquad \bigwedge_{j\in[i+1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{n+j-1} \cdot (r,b) \cdot \mathcal{AP}^{3n-1} \cdot (r,b) \cdot \mathcal{AP}^{+}) \Big)\end{aligned}$$

- A B formula ψ_{double} requiring that for each word $C \cdot C' \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot$ such that $C, C', C_1, \ldots, C_N$ are multi-cell codes, there are $i, j \in [1,N]$ such that $C = C_i$ and $C' = C_j$. ψ_{double} is the conjunction of two B formulas θ and θ', where θ (resp., θ') requires that there is $i \in [1,N]$ such that $C_i = C$ (resp., $C_i = C'$). We consider formula θ' (the definition of θ being similar).

$$\begin{aligned}\theta' := \; & \langle \mathrm{B} \rangle \Big((r_{mc} \cdot r_{mc} \cdot \bot \cdot (r_{mc})^{+}) \wedge \\ & \bigwedge_{i\in[1,n]} \bigvee_{d\in D} (\mathcal{AP}^{3n+i-1} \cdot d \cdot \mathcal{AP}^{+} \cdot d \cdot \mathcal{AP}^{3n-i}) \wedge \\ & \bigwedge_{i\in[1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{4n+i-1} \cdot (r,b) \cdot \mathcal{AP}^{+} \cdot (r,b) \cdot \mathcal{AP}^{2n-i}) \wedge \\ & \bigwedge_{i\in[1,n]} \bigvee_{b\in\{0,1\}} (\mathcal{AP}^{5n+i-1} \cdot (c,b) \cdot \mathcal{AP}^{+} \cdot (c,b) \cdot \mathcal{AP}^{n-i}) \Big)\end{aligned}$$

- A B formula ψ_{not_unique} requiring that for each word $C \cdot C' \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot$ such that $C, C', C_1, \ldots, C_N$ are multi-cell codes, the following holds:
  - C and C' have the same row number and column number but different content;
  - there are $i, j \in [1,N]$ such that $C = C_i$ and $C' = C_j$.

  The construction of ψ_{not_unique} is based on the formulas ψ_{double} and $\psi_{=}$:

$$\psi_{not\_unique} := \psi_{double} \wedge \psi_{=} \wedge \bigvee_{i\in[1,n]} \bigvee_{d,d'\in D: d\neq d'} (\mathcal{AP}^{i-1} \cdot d \cdot \mathcal{AP}^{3n-1} \cdot d' \cdot \mathcal{AP}^{+})$$

- A B formula ψ_{row} (resp., ψ_{col}) requiring that for each word $C \cdot C' \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot$ such that $C, C', C_1, \ldots, C_N$ are multi-cell codes, the following condition holds.

– Let us denote by $d_1 \dots d_n$ the content of C and by $d'_1 \dots d'_n$ the content of C'. Whenever (1) there are $i, j \in [1,N]$ such that $C = C_i$ and $C' = C_j$, and (2) C and C' have the same row number and column numbers h and $h+1$, respectively (resp., C and C' have the same column number and row numbers h and $h+1$, respectively) for some $h \in [0, 2^n - 2]$, then it holds that $(d_\ell, d'_\ell) \in H$ (resp., $(d_\ell, d'_\ell) \in V$) for all $\ell \in [1,N]$.

We focus on the formula ψ_{row} (the definition of ψ_{col} being similar) whose construction is based on the formulas ψ_{double} and $\psi_{c,inc}$:

$$\psi_{row} := (\psi_{double} \wedge \psi_{c,inc}) \longrightarrow \bigwedge_{i \in [1,n]} \bigvee_{(d,d') \in H} (\mathcal{AP}^{i-1} \cdot d \cdot \mathcal{AP}^{3n-1} \cdot d' \cdot \mathcal{AP}^{+})$$

The $\mathsf{B\overline{E}}$ formula φ_{MT} is then defined as follows:

$$\neg(\mathcal{AP}^{*} \cdot \bot \cdot \mathcal{AP}^{*})^{n+2} \ \wedge \ \langle \mathrm{B} \rangle \Big(\underbrace{(\bot \cdot (r_{mc})^{+} \cdot \bot) \wedge \neg \bigvee_{(d,d') \in D^2 \setminus M} (\mathcal{AP}^{+} \cdot d \cdot d' \cdot \mathcal{AP}^{+})}_{\text{concatenation of well-formed multi-cell codes}} \ \wedge$$

$$\underbrace{[\overline{E}]((r_{mc} \cdot \bot \cdot (r_{mc})^{+} \cdot \bot) \longrightarrow \psi_{comp})}_{\text{Completeness requirement of Definition 20}} \wedge \underbrace{[\overline{E}]((r_{mc} \cdot r_{mc} \cdot \bot \cdot (r_{mc})^{+} \cdot \bot) \longrightarrow \neg \psi_{not\_unique})}_{\text{Uniqueness requirement of Definition 20}} \wedge$$

$$\underbrace{[\overline{E}]((r_{mc} \cdot r_{mc} \cdot \bot \cdot (r_{mc})^{+} \cdot \bot) \longrightarrow (\psi_{row} \wedge \psi_{col}))}_{\text{Row-adjacency and column-adjacency requirements of Definition 20}} \ \wedge \ \underbrace{\bigvee_{d_{acc} \in D_{acc}} (\mathcal{AP}^{+} \cdot d_{acc} \cdot (r,1)^{n} \cdot \mathcal{AP}^{+})}_{\text{Acceptance requirement of Definition 20}} \Big)$$

Finally, the $\mathsf{B\overline{E}}$ formula φ_{IMT} is given by

$$\varphi_{MT} \wedge \varphi_{coh}$$

where φ_{coh} is a $\mathsf{B\overline{E}}$ formula ensuring the initialization coherence requirement of Definition 22. In order to define φ_{coh}, we need some auxiliary formulas:

- A regular expression r_{ic} capturing the initial cell codes:

$$r_{ic} := D_0 \cdot (\{c\} \times \{0,1\})^{n}$$

- A B formula ψ_{single} requiring that for each word $C \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot$ such that $C, C_1, \ldots, C_N$ are multi-cell codes, there is $i \in [1,N]$ such that $C = C_i$ and the row number of C is 0. The definition of ψ_{single} is similar to the definition of the B formula ψ_{double}.
- A B formula ψ_{coh} requiring that for each word $C \cdot \bot \cdot C_1 \cdot \ldots \cdot C_N \cdot \bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot \mathit{end}$ such that $C, C_1, \ldots, C_N$ are multi-cell codes and $\bot \cdot w_n \cdot \ldots \cdot \bot \cdot w_1 \cdot \mathit{end}$ is a multi-initialization code, the following holds: whenever there is $i \in [1,N]$ such that $C = C_i$, the row number of C is 0 and the content of C is $d_1 \dots d_n$, then for all $\ell \in [1,n]$, there is an initial code in w_ℓ having the same column number as C and whose content is d_ℓ.

$$\psi_{coh} := \big(\langle \mathrm{B} \rangle([(\mathcal{AP} \setminus \{\bot\})^{+} \cdot \bot \cdot (\mathcal{AP} \setminus \{\bot\})^{+} \cdot \bot] \wedge \psi_{single})\big) \longrightarrow \bigwedge_{\ell \in [1,n]} \psi_\ell$$

$$\psi_\ell := \langle \mathrm{B} \rangle \Big(\big[(\mathcal{AP} \setminus \{\bot\})^{+} \cdot (\bot \cdot (\mathcal{AP} \setminus \{\bot\})^{+})^{n-\ell+1} \cdot \bot \cdot r_{ic}^{+}\big] \wedge$$

$$\bigwedge_{i \in [1,n]} \bigvee_{b \in \{0,1\}} (\mathcal{AP}^{2n+i-1} \cdot (c,b) \cdot \mathcal{AP}^{+} \cdot (c,b) \cdot \mathcal{AP}^{n-i}) \wedge \bigvee_{d \in D} (\mathcal{AP}^{\ell-1} \cdot d \cdot \mathcal{AP}^{+} \cdot d \cdot \mathcal{AP}^{n}) \Big)$$

The $\mathsf{B}\overline{\mathsf{E}}$ formula φ_{coh} is then defined as follows:

$$\varphi_{coh} := [\overline{E}]\big([r_{mc} \cdot (\bot \cdot (\mathcal{AP} \setminus \{\bot\})^+)^{n+1}] \to \psi_{coh}\big)$$

This concludes the proof of Proposition 23. □