Branching-Time Model Checking of Parametric One-Counter Automata

Stefan Göller[1], Christoph Haase[2], Joël Ouaknine[2], and James Worrell[2]

[1] Institut für Informatik, Universität Bremen, Germany
[2] Department of Computer Science, University of Oxford, UK

Abstract. We study the computational complexity of model checking EF logic and modal logic on parametric one-counter automata (POCA). A POCA is a one-counter automaton whose counter updates are either integer values encoded in binary or integer-valued parameters. Given a formula and a configuration of a POCA, the model-checking problem asks whether the formula is true in this configuration for all possible valuations of the parameters. We show that this problem is undecidable for EF logic via reduction from Hilbert's tenth problem, however for modal logic we prove PSPACE-completeness. Obtaining the PSPACE upper bound involves analysing systems of linear Diophantine inequalities of exponential size that admit solutions of polynomial size. Finally, we show that model checking EF logic on POCA without parameters is PSPACE-complete.

1 Introduction

Counter automata, a fundamental and widely-studied model of computation, consist of a finite-state controller which manipulates a finite set of counters ranging over the naturals. A classic result by Minsky states that Turing completeness can already be obtained when restricting to two counters [17]. Due to this fact, research has subsequently focused on studying restricted classes of counter automata and related formalisms. Among others, we note the use of restrictions to a single counter (one-counter automata or *OCA*, for short), restrictions on the underlying structure of the controller (such as flatness [5,15]), restrictions on the kinds of allowable tests on the counters, and on the types of computations considered (such as reversal-boundedness [10,11]). Counter automata are also closely related to Petri nets and pushdown automata. In recent years, motivated by complexity-theoretic considerations on the one hand and practical applications on the other, researchers have investigated decision problems for counter automata with additional primitive operations on counters, such as additive updates encoded in *binary* [1,15] or even in *parametric* form, *i.e.*, updates whose precise values depend on a finite set of parameters [3,12]. We refer to such counter automata as *succinct* and *parametric* respectively, the former being a subclass of the latter. Natural applications of such counter automata include the modeling of resource-bounded processes, numeric data types, programs with lists, recursive or multi-threaded programs, and XML query evaluation; see, *e.g.*, [4,11,10,1].

The two most prominent decision problems for counter automata are *reachability* and *model checking*. Reachability asks whether there is path between two configurations in the potentially infinite transition system generated by a counter automaton. For counter

L. Birkedal (Ed.): FOSSACS 2012, LNCS 7213, pp. 406–420, 2012.

automata with parameters, this problem generalises to asking whether there exists a valuation of the parameters such that reachability holds between two configurations in the concrete transition system induced through the valuation. Model checking is the problem of deciding whether a formula given in some temporal logic holds in a configuration of the transition system induced by a counter automaton, and when parameters are present whether the formula holds in a configuration in *all* transition systems induced by all possible valuations. Due to Minsky's result, the restriction to a *single* counter is a natural way to potentially obtain decidability for reachability and model checking problems. Consequently, in this paper we restrict our attention to this class of counter automata, and in particular investigate model checking problems for *succinct one-counter automata (SOCA)* and *parametric one-counter automata (POCA)*.

State of the art. Reachability is known to be NL-complete for OCA and has recently been shown to be NP-complete for SOCA and decidable for POCA [9]. The complexity of model-checking problems for various temporal logics including LTL, CTL and fragments thereof has been studied for OCA, SOCA and POCA in a number of recent works [20,8,7,6,22]. When comparing OCA with SOCA, an exponential complexity jump for the model checking problem may arise: both CTL and μ-calculus model checking on OCA are PSPACE-complete [20,7], whereas for SOCA these problems are EXPSPACE-complete [20,6]. However, this jump is not inherent, since for example model checking LTL is PSPACE-complete for both OCA and SOCA. When parameters come into play, model checking LTL on POCA is NEXP-complete and becomes undecidable for CTL [6]. In [8], model checking the fragment EF of CTL on OCA, which can be seen as an extension of modal logic with a reachability predicate, is shown to be complete for P^{NP}. Despite its relatively limited expressiveness, EF is a useful specification language, and in particular bisimilarity checking of arbitrary systems against finite systems is polynomial-time reducible to EF model checking [13].

Our contribution. In this paper, we investigate the decidability and complexity of EF and modal logic (ML) model checking on transition systems generated by SOCA and POCA. As mentioned above, CTL model checking of POCA is undecidable [6], which is shown by reduction from the reachability problem for two-counter automata. In [6], we conjectured that EF model checking on POCA could be decidable, which is not unreasonable for two reasons. First, the undecidability proof for CTL on POCA in [6] heavily relies on the use of the *until* operator. Second, reachability for POCA is decidable [9], which is shown via a translation into the quantifier-free fragment of Presburger arithmetic with divisibility. Since there exist extensions of the latter theory that allow for universal quantification, see *e.g.* [2], and since EF primarily allows for reasoning about reachability relations, it seemed plausible that an instance of an EF model-checking problem on POCA could be translated into a sentence in such an extended theory. Nevertheless, we show in this paper that model checking EF logic on POCA is undecidable via a different reduction, namely from Hilbert's tenth problem, which Matiyasevich showed to be undecidable [16]. On the positive side, we establish tight complexity bounds for model checking POCA and SOCA against large fragments of EF. First, by dropping the reachability modality and thus restricting EF to ML, we show that the model-checking problem for POCA becomes PSPACE-complete. Obtaining the PSPACE upper bound involves a careful analysis of the size of the solution sets

Table 1. Complexity of model checking EF, ML and CTL/modal μ-calculus on OCA, SOCA and POCA

	OCA	**SOCA**	**POCA**
CTL, μ-cal.	PSPACE-complete [7,20]	EXPSPACE-complete [6,20]	Π^0_1-complete[6]
EF	P^{NP}-complete [7,8]	**PSPACE-complete**	$\mathbf{\Pi^0_1}$**-complete**
ML	P-complete [14]		

of certain systems of linear Diophantine inequalities of potentially exponential size. Second, when no parameters are present, we show that EF model checking for SOCA is PSPACE-complete. The main technical challenge is to develop an "exponential periodicity property" that characterizes those counter values at which an EF formula holds. Our results are summarized in **bold font** in Table 1, which also summarizes known results from the literature.

Structure of this paper. We introduce basic definitions and notations in Section 2 and present results on model checking POCA in Section 3. Section 4 deals with model checking SOCA before we conclude in Section 5. Due to space limitations, details of some proofs are deferred to a full version of this paper.

2 Preliminaries

Throughout this paper, we denote by $\mathbb{N} = \{0, 1, \ldots\}$ the *non-negative integers* and by $\mathbb{Z}$ the *integers*. We define $[i, j] \stackrel{\text{def}}{=} \{i, i + 1, \ldots, j\}$ and introduce $[i]$ as an abbreviation for $[1, i]$. For any $n \in \mathbb{N}$, we denote by $\lg n$ the smallest $i \in \mathbb{N}$ such that $n \leq 2^i$. Given a function $f : \mathbb{N} \to \mathbb{N}$, we write $f(n) = \mathsf{poly}(n)$ (resp. $f(n) = \mathsf{exp}(n)$) if there is some polynomial $p(n)$ such that $f(n) \leq p(n)$ (resp. $f(n) \leq 2^{p(n)}$) for each $n \in \mathbb{N}$.

The Branching-Time Logic EF: Formulas of EF over a finite set $\mathbb{P}$ of *atomic propositions* are inductively defined by the following grammar, where p ranges over $\mathbb{P}$:

$$\varphi ::= p \mid \varphi \wedge \varphi \mid \neg\varphi \mid \mathsf{EX}\varphi \mid \mathsf{EF}\varphi.$$

We define the standard Boolean abbreviations $\varphi_1 \vee \varphi_2 \stackrel{\text{def}}{=} \neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $\varphi_1 \to \varphi_2 \stackrel{\text{def}}{=} \neg\varphi_1 \vee \varphi_2$ and $\varphi_1 \leftrightarrow \varphi_2 \stackrel{\text{def}}{=} \varphi_1 \to \varphi_2 \wedge \varphi_2 \to \varphi_1$. Moreover, we define the additional modalities $\mathsf{AX}\varphi \stackrel{\text{def}}{=} \neg\mathsf{EX}\neg\varphi$ and $\mathsf{AG}\varphi \stackrel{\text{def}}{=} \neg\mathsf{EF}\neg\varphi$. *Modal Logic* ($\mathsf{ML}$) is obtained from EF by disallowing the EF operator. An EF formula φ is in *negation normal form* if all negation symbols occur only in front of atomic propositions. The *size* $|\varphi|$ of EF formulas φ is defined as usual.

The semantics of an EF formula is given in terms of transition systems. A *transition system* T is a tuple $T = (S, \mathbb{P}, \lambda, \longrightarrow)$, where S is the set of *states*, $\mathbb{P}$ is a finite set of *atomic propositions*, $\lambda : S \to 2^{\mathbb{P}}$ is the *state-labeling function* and $\longrightarrow \subseteq S \times S$ is the *transition relation*. We use infix notation for $\longrightarrow$ and write $s \longrightarrow s'$ whenever $(s, s') \in \longrightarrow$. An s-s' *path* ϱ in a transition system T is a finite sequence of states $\varrho : s_1 \cdots s_n$ such that $s = s_1$, $s' = s_n$ and $s_i \longrightarrow s_{i+1}$ for all $i \in [n - 1]$, and we write $\varrho : s \longrightarrow^* s'$ to express that ϱ is an s-s' path. Table 2 presents the semantics

Table 2. Semantics of EF

$$\begin{array}{lll lll}
(T,s) \models p & \iff & p \in \lambda(s) & (T,s) \models \varphi_1 \wedge \varphi_2 & \iff & (T,s) \models \varphi_1 \text{ and } (T,s) \models \varphi_2 \\
(T,s) \models \neg\varphi & \iff & (T,s) \not\models \varphi & (T,s) \models \mathsf{EX}\varphi & \iff & \exists s' \in S.(T,s') \models \varphi \text{ and } s \longrightarrow s' \\
 & & & (T,s) \models \mathsf{EF}\varphi & \iff & \exists s' \in S.(T,s') \models \varphi \text{ and } s \longrightarrow^* s'
\end{array}$$

of EF formulas. Given an EF formula φ, a transition system T and a state $s \in S$, the satisfaction relation $(T,s) \models \varphi$ is defined by induction on the structure of φ, and we say *φ holds at s in T* if $(T,s) \models \varphi$.

Parametric One-Counter Automata: Let $X = \{x_1, \ldots, x_n\}$ denote a finite set of *parameters*, and let $\mathsf{Op} \overset{\text{def}}{=} \{\mathsf{add}(z), \mathsf{add}(x) : z \in \mathbb{Z}, x \in X\} \cup \{\mathsf{zero}\}$ be a set of *operations*. A *parametric one-counter automaton (POCA)* is a tuple $\mathcal{A} = (Q, X, \mathbb{P}, \lambda, \Delta)$, where Q is a finite set of *control locations*, $\mathbb{P}$ is a finite set of *atomic propositions*, $\lambda : Q \to 2^{\mathbb{P}}$ is the *location-labeling function*, and $\Delta \subseteq Q \times \mathsf{Op} \times Q$ is the *transition relation*. A *succinct one-counter automaton (SOCA)* is a POCA with $X = \emptyset$. We write $q \xrightarrow{\mathsf{op}} q'$ whenever $(q, \mathsf{op}, q') \in \Delta$. By $n_{max}(\mathcal{A})$ we denote the largest absolute value of all integers occurring in the operations of $\mathcal{A}$. The *size $|\mathcal{A}|$ of a POCA $\mathcal{A}$* is defined as $|\mathcal{A}| \overset{\text{def}}{=} |\Delta| + \lg n_{max}(\mathcal{A})$. A *valuation* $\nu : X \to \mathbb{Z}$ is a function assigning an integer to each parameter. Given a POCA $\mathcal{A}$, a valuation induces a SOCA $\mathcal{A}^\nu$ which is obtained by replacing each transition $q \xrightarrow{\mathsf{add}(x_i)} q'$ with $q \xrightarrow{\mathsf{add}(\nu(x_i))} q'$. For a SOCA $\mathcal{A}$, we denote by $T(\mathcal{A}) \overset{\text{def}}{=} (S_{\mathcal{A}}, \mathbb{P}, \lambda_{\mathcal{A}}, \longrightarrow_{\mathcal{A}})$ the *transition system induced by $\mathcal{A}$*, where $S_{\mathcal{A}} \overset{\text{def}}{=} Q \times \mathbb{N}$, $\lambda_{\mathcal{A}} \overset{\text{def}}{=} (q,n) \mapsto \lambda(q)$, and $(q,n) \longrightarrow_{\mathcal{A}} (q', n')$ if, and only if, either $q \xrightarrow{\mathsf{add}(z)} q'$ and $n' = n + z$, or $q \xrightarrow{\mathsf{zero}} q' \in \Delta$ and $n = n' = 0$. For convenience, we write $q(n)$ instead of (q,n) for states in $S_{\mathcal{A}}$. Given two states $q(n)$ and $q'(n')$, *reachability* is to decide whether there exists a $q(n)$-$q'(n')$ path in $T(\mathcal{A})$.

Proposition 1 ([9]). *Reachability in SOCA is* NP*-complete.*

The *model-checking problem* for POCA, and thus for SOCA, is defined as follows:
<u>ML/EF MODEL CHECKING ON POCA</u>

INPUT: A POCA $\mathcal{A} = (Q, X, \mathbb{P}, \lambda, \Delta)$, $q \in Q$ and an ML/EF formula φ.
QUESTION: Does $(T(\mathcal{A}^\nu), q(0)) \models \varphi$ hold for each assignment $\nu : X \to \mathbb{Z}$?

We note that deciding whether $(T(\mathcal{A}^\nu), q(0)) \models \varphi$ holds for each assignment ν is the complement of deciding if $(T(\mathcal{A}^\nu), q(0)) \models \neg\varphi$ holds for some assignment ν.

We close this section with an example of a model-checking problem. Figure 1 depicts a SOCA $\mathcal{A}_i$ with $i \in [0, m]$ for some $m \in \mathbb{N}$. Starting in state $q_i(n)$ with $n \in [0, 2^{m+1} - 1]$, it is easily verified that the state $q_z(0)$, which is labeled with p_i, is reachable from $q_i(n)$ if, and only if, the coefficient of 2^i in the binary expansion of n is 1, which is the case if, and only if, $(T(\mathcal{A}), q_i(n)) \models \mathsf{EF}p_i$ or alternatively $(T(\mathcal{A}), q_i(n)) \models \mathsf{EX}^{m+2}p_i$. Here, EX^{m+2} is an abbreviation for the $m+2$-fold application of the EX operator.

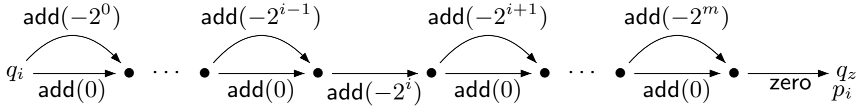


Fig. 1. SOCA $\mathcal{A}_i$ used for testing a bit of a number $n \in [2^{m+1} - 1]$

3 Model Checking POCA

In this section, we prove that model checking EF on POCA is undecidable (Section 3.1). We show that for ML model checking on POCA is decidable and in PSPACE (Section 3.2).

3.1 Model Checking EF on POCA

We now consider model checking EF on POCA and show that this problem is Π_1^0-complete. With EF being a notational fragment of CTL, membership in Π_1^0 follows from the fact that CTL model checking on POCA is Π_1^0-complete [6]. Thus, we concentrate in this section on a matching Π_1^0-lower bound by giving a reduction from Hilbert's Tenth Problem to the complement of the model checking problem.

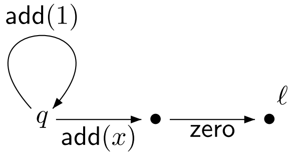
HILBERT'S TENTH PROBLEM (HTP)

INPUT: A polynomial p with coefficients ranging over the integers.
QUESTION: Do there exist $a_1, \ldots, a_n \in \mathbb{Z}$ such that $p(a_1, \ldots, a_n) = 0$?

HTP was shown to be Σ_1^0-complete by Matiyasevich [16]. Note that HTP remains Σ_1^0-hard if we restrict the a_i to range over $\mathbb{N}$: A Diophantine equation $p(x_1, x_2, .., x_n) = 0$ is solvable in the integers if, and only if, one of the 2^n equations $p(\pm x_1, \ldots, \pm x_n) = 0$ has a solution in the naturals. Replacing every unknown with the sum of squares of four unknowns gives, by Lagrange's Theorem, the reduction in the other direction.

Moreover, we may assume with no loss of generality that $a_i > 0$ for each $i \in [n]$. If some a_i were to be zero in a solution, we can obtain a new polynomial p' in $n-1$ variables by replacing a_i with 0 in p.

Let us fix some polynomial p with coefficients ranging over $\mathbb{Z}$. We will subsequently show how we can compute from p a POCA $\mathcal{A}_p$ with a control state q_p and an EF formula φ_p such that p has a solution over the naturals if, and only if, $(T(\mathcal{A}_p^\nu), q_p(0)) \models \varphi_p$ for *some* valuation ν of the parameters of $\mathcal{A}$. Recall that the valuation of the parameters of $\mathcal{A}_p$ ranges over $\mathbb{Z}$. However, we can easily ensure with a simple EF formula that a parameter x is positive. For the following SOCA $\mathcal{A}_{x\geq 1}$.



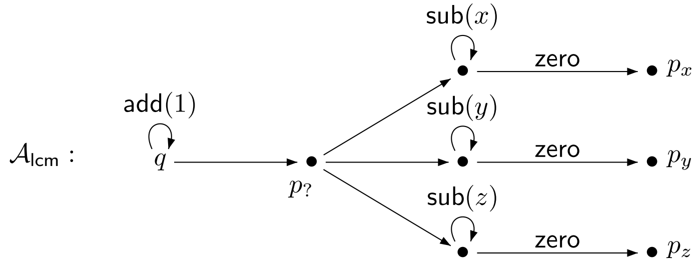
we have $(T(\mathcal{A}_{x\geq 1}^\nu), q(0)) \models \neg\mathsf{EF}\ell$ if, and only if, $\nu(x) \geq 1$.

More challenging than testing if a parameter is positive when reducing from HTP is that we need to be able to express a multiplication relation over the parameters in the POCA.

In order to do that, we employ a trick that became popular by the work of Robinson [18] which allows us to define multiplication in terms of the least common multiple. In fact given $x, y \in \mathbb{N}$, we have

$$\begin{aligned} &\mathsf{lcm}(x+y, x+y+1) - \mathsf{lcm}(x, x+1) - \mathsf{lcm}(y, y+1) \\ =&(x^2 + x + 2xy + y^2 + y) - (x^2 + x) - (y^2 + y) \quad = \quad 2xy \end{aligned}$$

We note that addition and subtraction of the parameters can easily be realized by introducing additional *slack parameters* in the POCA. Thus, we can enhance our POCA by transitions of the kind $\mathsf{sub}(x)$, meaning that $\nu(x)$ is subtracted from the counter, provided the counter is at least $\nu(x)$. We now demonstrate that for parameters x, y, z of some POCA that each assume positive values, which we can check as seen above, we can "express" in EF that $z = \mathsf{lcm}(x, y)$. Consider the following POCA $\mathcal{A}_{\mathsf{lcm}}$, where unlabeled transitions are assumed to be labeled with "$\mathsf{add}(0)$":



The idea is to express that for all $n \in \mathbb{N}$, we have that both x and y divide n if, and only if, z divides n. We note that for each $\nu : \{x, y, z\} \to \mathbb{Z}$ with $\nu(x), \nu(y), \nu(y) \geq 1$ we have that $(T(\mathcal{A}^{\nu}_{\mathsf{lcm}}), q(0))) \models \mathsf{AG}(p_? \to ((\mathsf{EF}p_x \wedge \mathsf{EF}p_y) \leftrightarrow \mathsf{EF}p_z))$ if, and only if, $\nu(z) = \mathsf{lcm}(\nu(x), \nu(y))$.

Thus, by introducing a sufficient number of slack variables, we can express multiplication, addition and subtraction, which allows us to solve HTP for any arbitrary polynomial. Thus, we obtain the following theorem.

Theorem 2. *Model checking* EF *logic on POCA is Π^0_1-complete.*

We note that by [16] there exists a *fixed universal* polynomial $p_u(n, k, x_1, \ldots, x_m)$ such that for each recursively enumerable set $S \subseteq \mathbb{N}$, there is some $k_0 \in \mathbb{N}$ such that $S = \{n \in \mathbb{N} \mid \exists n_1, \ldots, n_m \in \mathbb{N} : p_u(n, k_0, n_1, \ldots, n_m) = 0\}$. This allows us to strengthen our result insofar as there exists a *fixed* EF formula φ and a *fixed* POCA $\mathcal{A} = (Q, X, \mathbb{P}, \lambda, \Delta)$ with a transition $q \xrightarrow{\mathsf{add}(y)} q' \in \Delta$ and a control state $q_0 \in Q$ such that it is Π^0_1-complete to decide for a given $n \in \mathbb{N}$ whether by replacing y with n, $(T(\mathcal{A}^{\nu}), q_0(0)) \models \varphi$ holds for all $\nu : X \to \mathbb{Z}$.

3.2 Model Checking ML on POCA

This section will be devoted to proving a PSPACE upper bound for model checking ML on POCA. Let us fix some POCA $\mathcal{A} = (Q, X, \mathbb{P}, \lambda, \Delta)$ with $X = \{x_1, \ldots, x_\ell\}$, some control state $q_0 \in Q$ and some ML formula α. *Provided* that ML model checking of SOCA is in PSPACE (we show that even model checking EF on SOCA is in PSPACE in Section 4.2), in order to obtain a PSPACE upper bound, it is sufficient to show that if $(T(\mathcal{A}^\nu), q_0(0)) \models \alpha$ holds for some $\nu : X \to \mathbb{Z}$ then there is some $\mu : X \to \mathbb{Z}$ such that $(T(\mathcal{A}^\mu), q(0)) \models \alpha$ and $|\mu(x)|$ can be represented with polynomially many bits in $|\mathcal{A}| + |\alpha|$ for each $x \in X$, since such an assignment can be guessed in PSPACE.

For each $q \in Q$ and each subformula φ of α, let us define $\mathcal{M}(q, \varphi) \subseteq \mathbb{Z}^\ell \times \mathbb{N} \subseteq \mathbb{Z}^{\ell+1}$ as follows:

$$\mathcal{M}(q, \varphi) \stackrel{\text{def}}{=} \{(z_1, \ldots, z_\ell, n) \mid (T(\mathcal{A}^\nu), q(n)) \models \varphi \text{ and } \nu(x_i) = z_i, i \in [1, \ell]\}.$$

Before we proceed with the proof of the upper bound, we need to introduce some additional notation. For an integer matrix $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, we denote by $\|A\| = \max_i \{\sum_j |a_{ij}|\}$ *the norm of* A. For an integer vector $\boldsymbol{b} = (b_i)$, we denote by $\|\boldsymbol{b}\| = \sum_i |b_i|$ *the norm of* $\boldsymbol{b}$. A *system of linear Diophantine inequalities (SLDI)* is a system of the form $\mathcal{S} = (A\boldsymbol{x} \geq \boldsymbol{b})$, where $A \in \mathbb{Z}^{m \times n}$ is an $m \times n$ matrix, $\boldsymbol{b} \in \mathbb{Z}^m$ is an m-vector and $\boldsymbol{x}$ is an n-vector of indeterminates all ranging over the integers. By $\mathsf{Sol}(\mathcal{S})$, we denote the set of *integer solutions* to the SLDI $\mathcal{S} = (A\boldsymbol{x} \geq \boldsymbol{b})$. Finally, we define $\|\mathcal{S}\|_{\text{mat}} \stackrel{\text{def}}{=} \|A\|$ and $\|\mathcal{S}\|_{\text{vec}} \stackrel{\text{def}}{=} \|\boldsymbol{b}\|$.

Recall that $x_1, \ldots, x_\ell$ are the parameters of $\mathcal{A}$. Our overall goal is to express $\mathcal{M}(q, \varphi)$ by a *union* of solutions to SLDIs, each of the form

$$\mathcal{S} = (A\boldsymbol{x} \geq \boldsymbol{b}), \qquad \text{where } A \in \mathbb{Z}^{m \times (\ell+1)} \text{ and } \boldsymbol{b} \in \mathbb{Z}^m \text{ for some } m \geq 1.$$

In the remainder of this section, we will assume for any $(A\boldsymbol{x} \geq \boldsymbol{b})$ that A is some $m \times (\ell + 1)$ matrix and $\boldsymbol{b}$ is some m-vector for some $m \geq 1$. The intuition is that the i^{th} component of $\boldsymbol{x}$ with $i \in [\ell]$ is going to correspond to the parameter x_i of $\mathcal{A}$ and the $(\ell + 1)^{\text{th}}$ component of $\boldsymbol{x}$ is going to correspond to the counter value where the ML formula is evaluated. In case $A = (a_{ij})$ we define $\|A\|_{\ell+1} \stackrel{\text{def}}{=} \max\{|a_{i(\ell+1)}| : i \in [m]\}$ and lift this definition to $\|\mathcal{S}\|_{\ell+1} \stackrel{\text{def}}{=} \|A\|_{\ell+1}$.

In order to prove that small valuations $\nu : X \to \mathbb{Z}$ suffice for α, we are now going to prove that for each $q \in Q$ and each subformula φ of α, we have

$$\mathcal{M}(q, \alpha) = \bigcup_{i \in I} \mathsf{Sol}(\mathcal{S}_i)$$

for some index set I with $\|\mathcal{S}_i\|_{\text{mat}} = \mathsf{poly}(|\varphi|)$ and $\|\mathcal{S}_i\|_{\text{vec}} = \mathsf{poly}(|\varphi|) \cdot \mathsf{exp}(|\mathcal{A}|)$ for each $i \in I$. Once this fact has been established, we will show that each SLDI $\mathcal{S}_i$ admits solutions that can be represented using polynomially many bits in $|\mathcal{A}| + |\alpha|$, thus establishing the desired upper bound on necessary valuations of the parameters of $\mathcal{A}$.

We require some additional notation that, together with the subsequent lemma, will be useful for proving the existence of sets of SLDIs of "small" size for each $\mathcal{M}(q, \varphi)$. Let $H \subseteq \mathbb{Z}^{\ell+1}$. We define $H - x_k \stackrel{\text{def}}{=} \{(z_1, \ldots, z_\ell, z_{\ell+1} - z_k) \in \mathbb{Z}^{\ell+1} \mid (z_1, \ldots, z_{\ell+1}) \in H\}$ for each $k \in [\ell]$ and $H - z \stackrel{\text{def}}{=} \{(z_1, \ldots, z_\ell, z_{\ell+1} - z) \in \mathbb{Z}^{\ell+1} \mid (z_1, \ldots, z_{\ell+1}) \in H\}$ for each $z \in \mathbb{Z}$. The following lemma states that solutions to SLDIs are closed under the operations $-x_k$ and $-z$ and gives bounds on the blow-up of the introduced norms.

We remark that we do not require an effective variant of this lemma to establish our PSPACE upper bound.

Lemma 3. *Let $\mathcal{S} = (A\boldsymbol{x} \geq \boldsymbol{b})$ be an SLDI with $A = (a_{ij}) \in \mathbb{Z}^{m \times (\ell+1)}$. Then the following holds:*

(1) For each $k \in [\ell]$ there is some SLDI $\mathcal{S}'$ with $\mathsf{Sol}(\mathcal{S}') = \mathsf{Sol}(\mathcal{S}) - x_k$, $\|\mathcal{S}'\|_{mat} \leq \|\mathcal{S}\|_{mat} + \|\mathcal{S}\|_{\ell+1}$, $\|\mathcal{S}'\|_{\ell+1} = \|\mathcal{S}\|_{\ell+1}$, and $\|\mathcal{S}'\|_{vec} = \|\mathcal{S}\|_{vec}$.

(2) For each $z \in \mathbb{Z}$, there is some SLDI $\mathcal{S}'$ with $\mathsf{Sol}(\mathcal{S}') = \mathsf{Sol}(\mathcal{S}) - z$, $\|\mathcal{S}'\|_{mat} = \|\mathcal{S}\|_{mat}$, $\|\mathcal{S}'\|_{\ell+1} = \|\mathcal{S}\|_{\ell+1}$, and $\|\mathcal{S}'\|_{vec} \leq \|\mathcal{S}\|_{vec} + \|\mathcal{S}\|_{\ell+1} \cdot |z|$.

Proof. Let us assume $\boldsymbol{b} = (b_i)$. For Point (1), let $k \in [1, \ell]$. For each $(z_1, \ldots, z_{\ell+1}) \in \mathbb{Z}^{\ell+1}$ we have

$$\begin{aligned} & (z_1, \ldots, z_{\ell+1}) \in \mathsf{Sol}(\mathcal{S}) - x_k \\ \iff \quad & (z_1, \ldots, z_\ell, z_{\ell+1} + z_k) \in \mathsf{Sol}(\mathcal{S}) \\ \iff \quad & \forall i \in [1, m] : \left(\sum_{j \in [1,\ell]} a_{ij} \cdot z_j + a_{i(\ell+1)}(z_{\ell+1} + z_k) \geq b_i \right) \\ \iff \quad & \forall i \in [1, m] : \left((a_{ik} + a_{i(\ell+1)}) z_k + \sum_{\substack{j \in [1,\ell+1], \\ j \neq k}} a_{ij} \cdot z_j \geq b_i \right). \end{aligned}$$

We can thus define the matrix $A' = (a'_{ij})$, where $a'_{ij} = a_{ij}$ if $j \neq k$ and $a_{ij} = a_{ij} + a_{i(\ell+1)}$ if $j = k$, for each $i \in [1, m]$. We put $\mathcal{S}' = (A'\boldsymbol{x} \geq \boldsymbol{b})$ and we just proved $\mathsf{Sol}(\mathcal{S}') = \mathsf{Sol}(\mathcal{S}) - x_k$. Moreover, it holds $\|\mathcal{S}'\|_{\text{mat}} = \|A'\| \leq \|A\| + \|A\|_{\ell+1} = \|\mathcal{S}\|_{\text{mat}} + \|\mathcal{S}\|_{\ell+1}$, $\|\mathcal{S}'\|_{\ell+1} = \|A\|_{\ell+1} = \|\mathcal{S}\|_{\ell+1}$, and $\|\mathcal{S}'\|_{\text{vec}} = \|\boldsymbol{b}\| = \|\mathcal{S}\|_{\text{vec}}$.

Point (2) is shown analogously. □

We are now ready to prove the desired lemma.

Lemma 4. *For every $q \in Q$ and every subformula φ of α in negation normal form, we have $\mathcal{M}(q, \varphi) = \bigcup_{i \in I} \mathsf{Sol}(\mathcal{S}_i)$, where I is some index set and each $\mathcal{S}_i$ is some SLDI with $\|\mathcal{S}_i\|_{mat} \leq |\varphi|$, $\|\mathcal{S}_i\|_{\ell+1} \leq 1$, $\|\mathcal{S}_i\|_{vec} \leq (n_{max}(\mathcal{A}) + 1) \cdot |\varphi|$.*

Proof. We prove the lemma by structural induction on φ.

Case $\varphi = p$ for some $p \in \mathbb{P}$ (the case $\varphi = \neg p$ is dual).

First, let us assume $p \in \lambda(q)$. Then $\mathcal{M}(q, \varphi) = \mathbb{Z}^\ell \times \mathbb{N}$, which can be described by the solutions to the single SLDI $\mathcal{S} \overset{\text{def}}{=} (A\boldsymbol{x} \geq b)$ with $\boldsymbol{b} \overset{\text{def}}{=} \boldsymbol{0}$ and $A \overset{\text{def}}{=} (a_{ij}) \in \mathbb{Z}^{1 \times (\ell+1)}$ with $a_{1j} \overset{\text{def}}{=} 0$ for each $j \in [1, \ell]$ and $a_{1(\ell+1)} \overset{\text{def}}{=} 1$. Note that $\|\mathcal{S}\|_{\text{mat}} = \|A\| = 1 = |\varphi|$, $\|\mathcal{S}\|_{\ell+1} = \|A\|_{\ell+1} = 1$, and $\|\mathcal{S}\|_{\text{vec}} = \|\boldsymbol{b}\| = 0 \leq (n_{max}(\mathcal{A}) + 1) \cdot |\varphi|$.

In case $p \notin \lambda(q)$, we have $\mathcal{M}(q, \varphi) = \emptyset$, which we express as the solutions of the SLDI $\mathcal{S} = (A\boldsymbol{x} \geq \boldsymbol{b})$, where A is $1 \times (\ell + 1)$ zero matrix and $\boldsymbol{b} \overset{\text{def}}{=} 1$. We have $\|\mathcal{S}\|_{\text{mat}} = \|A\| = 0 \leq 1 = |\varphi|$, $\|\mathcal{S}\|_{\ell+1} = \|A\|_{\ell+1} = 0 \leq 1$, and $\|\mathcal{S}\|_{\text{vec}} = \|\boldsymbol{b}\| = 0 \leq (n_{max}(\mathcal{A}) + 1) \cdot |\varphi|$.

Case $\varphi = \psi \vee \psi'$: By the induction hypothesis we have $\mathcal{M}(q,\psi) = \bigcup_{i \in I} \mathsf{Sol}(\mathcal{S}_i)$ for some index set I and for SLDI $\mathcal{S}_i$, for each $i \in I$ and $\mathcal{M}(q,\psi') = \bigcup_{i \in I'} \mathsf{Sol}(\mathcal{S}'_i)$ for some index set I' and for SLDI $\mathcal{S}'_i$, for each $i \in I'$. Obviously we can write $\mathcal{M}(q,\varphi)$ as $\bigcup_{i \in I} \mathsf{Sol}(\mathcal{S}_i) \cup \bigcup_{i \in I'} \mathsf{Sol}(\mathcal{S}'_i)$ and the bounds on the norms easily carry over from induction hypothesis.

Case $\varphi = \psi \wedge \psi'$: By induction the hypothesis we have $\mathcal{M}(q,\psi) = \bigcup_{i \in I} \mathsf{Sol}(\mathcal{S}_i)$ for some index set I and for SLDIs $\mathcal{S}_i$, for each $i \in I$ and $\mathcal{M}(q,\psi') = \bigcup_{i \in I'} \mathsf{Sol}(\mathcal{S}'_i)$ for some index set I' and for SLDIs $\mathcal{S}'_i$, for each $i \in I'$. Let us assume $\mathcal{S}_i = (A_i\boldsymbol{x} \geq \boldsymbol{b_i})$ for each $i \in I$ and $\mathcal{S}'_i = (A'_i\boldsymbol{x} \geq \boldsymbol{b'_i})$ for each $i \in I'$. We define the matrix $A_{ii'} \overset{\text{def}}{=} \binom{A_i}{A_{i'}}$ and the vector $b_{ii'} \overset{\text{def}}{=} \binom{b_i}{b_{i'}}$ for each $i \in I$ and each $i' \in I'$. Obviously, we have $\mathcal{M}(q,\varphi) = \mathcal{M}(q,\psi) \cap \mathcal{M}(q,\psi') = \bigcup_{i \in I, i' \in I'} \mathsf{Sol}(A_{ii'}\boldsymbol{x} \geq b_{ii'})$. Again, the bounds on the norms immediately carry over from induction hypothesis.

Case $\varphi = \mathsf{AX}\psi$: By the induction hypothesis, we have $\mathcal{M}(q',\psi) = \bigcup_{i \in I_{q'}} \mathsf{Sol}(\mathcal{S}_{i,q'})$ for some SLDIs $\mathcal{S}_{i,q'}$ for each $q' \in Q$. Let us assume that $\mathcal{S}_{i,q'} = (A_{i,q'}\boldsymbol{x} \geq \boldsymbol{b_{i,q'}})$ for each $i \in I_{q'}$ and each $q' \in Q$. Before giving the translation, we need to introduce some auxiliary SLDIs $\mathcal{S}_{\circ z}$ and $\mathcal{S}_{\circ x_k}$ for each $z \in \mathbb{Z}$, each $k \in [\ell]$ and each $\circ \in \{<,>,\leq,\geq\}$ such that

$$\mathsf{Sol}(\mathcal{S}_{\circ z}) = \{(z_1,\ldots,z_{\ell+1}) \in \mathbb{Z}^{\ell+1} \mid z_{\ell+1} \circ z\} \text{ and}$$
$$\mathsf{Sol}(\mathcal{S}_{\circ x_k}) = \{(z_1,\ldots,z_{\ell+1}) \in \mathbb{Z}^{\ell+1} \mid z_{\ell+1} \circ z_k\}.$$

For $z \in \mathbb{Z}$, we only give $\mathcal{S}_{\circ z}$ for $\circ =$"$<$", the remaining cases for $\circ$ can be defined analogously. We put $\mathcal{S}_{<z} \overset{\text{def}}{=} (A\boldsymbol{x} \geq \boldsymbol{b})$, where $A \overset{\text{def}}{=} (a_{1j}) \in \mathbb{Z}^{1\times(\ell+1)}$ with $a_{1j} \overset{\text{def}}{=} 0$ if $j \in [\ell]$ and $a_{1(\ell+1)} \overset{\text{def}}{=} -1$, and finally $\boldsymbol{b} \overset{\text{def}}{=} (-z+1)$ since over the integers we have $z_{\ell+1} < z$ if, and only if, $z_{\ell+1} \leq z-1$ if, and only if, $-z_{\ell+1} \geq -z+1$. Observe that $\|\mathcal{S}_{\circ z}\|_{\text{mat}} \leq 1$, $\|\mathcal{S}_{\circ z}\|_{\ell+1} \leq 1$, and $\|\mathcal{S}_{\circ z}\|_{\text{vec}} \leq |z|+1$ for each $\circ \in \{<,>,\leq,\geq\}$.

Likewise, we define $\mathcal{S}_{\circ x_k}$ for $\circ =$"$<$", the other cases for $\circ$ can be dealt with analogously. The reader easily verifies that one can define $\mathcal{S}_{<x_i} \overset{\text{def}}{=} (C\boldsymbol{x} \geq \boldsymbol{d})$ with $C \overset{\text{def}}{=} (c_{1j}) \in \mathbb{Z}^{1\times(\ell+1)}$ with $c_{1j} \overset{\text{def}}{=} 1$ if $j = i$, $c_{1j} \overset{\text{def}}{=} -1$ if $j = \ell+1$, and $c_{1j} \overset{\text{def}}{=} 0$ otherwise. Moreover, we put $\boldsymbol{d} \overset{\text{def}}{=} (1)$. Observe that $\|\mathcal{S}_{\circ x_k}\|_{\text{mat}} \leq 1$, $\|\mathcal{S}_{\circ x_k}\|_{\ell+1} \leq 1$, and $\|\mathcal{S}_{\circ x_k}\|_{\text{vec}} \leq 1$ for each $\circ \in \{<,>,\leq,\geq\}$. We now define

$$\mathcal{M}(q,\varphi) \overset{\text{def}}{=} \mathsf{Sol}(\mathcal{S}_{\geq 0}) \cap \bigcap_{\substack{q \xrightarrow{\mathsf{add}(y)} q' \in \Delta \\ y \in \mathbb{Z} \cup X}} \left(\mathsf{Sol}(\mathcal{S}_{<y}) \cup \bigcup_{i \in I_{q'}} (\mathsf{Sol}(\mathcal{S}_{i,q'}) - y) \right).$$

In the same fashion as for disjunction and conjunction, we can express the right-hand side of the latter equality as a union of SLDIs. Note that in this modification process the number of rows of the matrix may change, but *neither* do the norms of the matrices *nor* the norms of the vectors of the systems. The reader easily verifies that the $\|\cdot\|_{\text{mat}}$, $\|\cdot\|_{\ell+1}$, and $\|\cdot\|_{\text{vec}}$ norms of each auxiliary SLDI satisfy the bounds required by the lemma. Hence, in order to bound the norms of the SLDI that occur in the final union, it suffices to bound the norms of each SLDI $\mathcal{S}$ such that $\mathsf{Sol}(\mathcal{S}) = \mathsf{Sol}(\mathcal{S}_{i,q'}) - y$ for some $q' \in Q$, some $i \in I_{q'}$ and some $q \xrightarrow{\mathsf{add}(y)} q' \in \Delta$, where $y \in \mathbb{Z} \cup X$. To this end, we apply Lemma 3 by distinguishing between $y \in \mathbb{Z}$ and $y \in X$.

If $y = x_k$ for some $k \in [\ell]$, *i.e.* $y \in X$, we obtain the following bounds by Point (1) of Lemma 3:

- $\|\mathcal{S}\|_{\text{mat}} \overset{\text{Lemma 3 (1)}}{\leq} \|A_{i,q'}\| + \|A_{i,q'}\|_{\ell+1} \overset{\text{IH}}{\leq} |\psi| + 1 = |\varphi|$,
- $\|\mathcal{S}\|_{\ell+1} \overset{\text{Lemma 3 (1)}}{=} \|A_{i,q'}\|_{\ell+1} \overset{\text{IH}}{\leq} 1$, and
- $\|\mathcal{S}\|_{\text{vec}} \overset{\text{Lemma 3 (1)}}{=} \|\boldsymbol{b_{i,q'}}\| \overset{\text{IH}}{\leq} (n_{max}(\mathcal{A}) + 1) \cdot |\psi| \leq (n_{max}(\mathcal{A}) + 1) \cdot |\varphi|$

In case $y \in \mathbb{Z}$, we obtain the following by Point (2) of Lemma 3:

- $\|\mathcal{S}\|_{\text{mat}} \overset{\text{Lemma 3 (2)}}{=} \|A_{i,q'}\| \overset{\text{IH}}{\leq} |\psi| \leq |\varphi|$,
- $\|\mathcal{S}\|_{\ell+1} \overset{\text{Lemma 3 (2)}}{=} \|A_{i,q'}\|_{\ell+1} \overset{\text{IH}}{\leq} 1$, and
- $\|\mathcal{S}\|_{\text{vec}} \overset{\text{Lemma 3 (2)}}{\leq} \|\boldsymbol{b_{i,q'}}\| + \|A_{i,q'}\|_{\ell+1} \cdot |y| \overset{\text{IH}}{\leq} (n_{max}(\mathcal{A}) + 1) \cdot |\psi| + 1 \cdot n_{max}(\mathcal{A}) \leq (n_{max}(\mathcal{A}) + 1) \cdot |\varphi|$

Case $\varphi = \mathsf{EX}\psi$. By induction hypothesis, we have $\mathcal{M}(q', \psi) = \bigcup_{i \in I_{q'}} \mathsf{Sol}(\mathcal{S}_{i,q'})$ for some SLDIs $\mathcal{S}_{i,q'}$ for each $q' \in Q$. Let us assume that $\mathcal{S}_{i,q'} = (A_{i,q'}\boldsymbol{x} \geq \boldsymbol{b_{i,q'}})$ for each $i \in I_{q'}$ and each $q' \in Q$. We define

$$\mathcal{M}(q, \varphi) \overset{\text{def}}{=} \mathsf{Sol}(\mathcal{S}_{\geq 0}) \cap \left(\bigcup_{q \xrightarrow{\mathsf{add}(y)} q' \in \Delta} \bigcup_{i \in I_{q'}} (\mathsf{Sol}(\mathcal{S}_{i,q'}) - y) \right).$$

The analysis of the sizes of the norms can be proven analogously as for the case $\varphi = \mathsf{AX}\psi$. □

The following lemma from [19] states that solvable SLDIs have small solutions whose norm is independent on the number of rows of the SLDI.

Lemma 5 ([19], p. 239). *Each solvable SLDI $A\boldsymbol{x} \geq \boldsymbol{b}$ has a solution of norm at most* $\mathsf{poly}(\|A\| + \|\boldsymbol{b}\|)$.

Let us return to our original formula α. By Lemma 4, there exists some SLDI $\mathcal{S}_i$ such that $\mathcal{M}(q_0, \alpha) = \mathsf{Sol}(\mathcal{S}_i)$, and where $\|\mathcal{S}_i\|_{\text{mat}} \leq |\alpha|$ and $\|\mathcal{S}_i\|_{\text{vec}} \leq (n_{max}(\mathcal{A}) + 1) \cdot |\alpha|$. Since we are interested if $(T(\mathcal{A}^\nu), q_0(0)) \models \alpha$ for some $\nu : X \to \mathbb{Z}$, think of adding to each matrix that occurs in $\mathcal{S}_i$ two more rows expressing that $x_{\ell+1} = 0$. Let us call the resulting SLDI $\mathcal{S}_i'$. By Lemma 5, we know that if $\mathcal{S}_i'$ is solvable, then $\mathcal{S}_i'$ has a solution of norm at most $\mathsf{poly}(n_{max}(\mathcal{A}) + |\alpha|)$. In other words, if $(T(\mathcal{A}^\nu), q_0(0)) \models \alpha$ for some $\nu : X \to \mathbb{Z}$, then $(T(\mathcal{A}^\mu), q_0(0)) \models \alpha$ already holds for some $\mu : X \to \mathbb{Z}$ and $\mu(x)$ is polynomially bounded in $|\mathcal{A}| + |\alpha|$ for each $x \in X$.

Hence, we obtain the following theorem.

Theorem 6. ML *model checking for POCA is in* PSPACE.

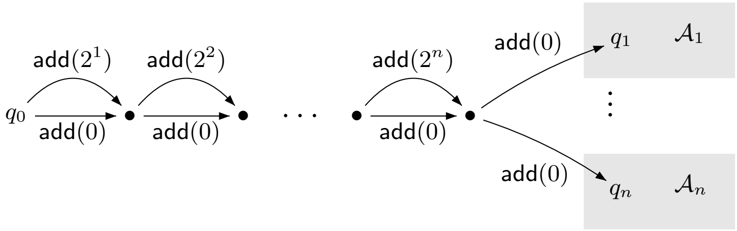


Fig. 2. SOCA $\mathcal{A}$ constructed for simulating the QBF formula α

4 Model Checking SOCA

In this section we prove that model checking ML on SOCA is PSPACE-hard (Section 4.1) and that model checking EF on SOCA is in PSPACE (Section 4.2).

4.1 Model Checking ML on SOCA

PSPACE-hardness of ML model checking on SOCA follows from a straight-forward reduction from QBF.

Proposition 7. *Model checking* ML *on SOCA is* PSPACE*-hard.*

Proof. We give a reduction from QBF. Let $\alpha = \exists x_1 \forall x_2 \cdots \exists x_n \beta(x_1, \ldots, x_n)$ be an instance of QBF. Without loss of generality, we can assume that β is in 3-CNF, *i.e.*, of the form $\beta = \bigwedge_{i \in [m]} \beta_i$, where each clause β_i consists of three literals, so $\beta_i = (\ell_{i_1} \vee \ell_{i_2} \vee \ell_{i_3})$. We construct in polynomial time a SOCA $\mathcal{A} = (Q, \mathbb{P}, \lambda, \Delta)$ and an ML formula φ such that for some $q_0 \in Q$ we have that α is valid if, and only if, $(T(\mathcal{A}), q_0(0)) \models \varphi$. We define $\mathbb{P} \overset{\text{def}}{=} \{p_i \mid i \in [n]\}$. The states and transitions of $\mathcal{A}$ are given in Figure 2, where the SOCA $\mathcal{A}_i$ is taken from Figure 1. Finally, we define φ to be the ML formula that is obtained by replacing each $\exists x_i$ from α with EX, each $\forall x_i$ with AX, and each literal ℓ_{i_j} with $\mathsf{EX}^{n+2} p_{i_j}$ if $\ell_{i_j} = x_{i_j}$ and $\neg\mathsf{EX}^{n+2} p_{i_j}$ if $\ell_{i_j} = \overline{x_{i_j}}$. It is easily verified that α is valid if, and only if, $(T(\mathcal{A}), q_0(0)) \models \varphi$. □

4.2 Model Checking EF on SOCA

In this section, we are going to show that EF model checking on SOCA is in PSPACE, and hence PSPACE-complete by Proposition 7. To this end, let us fix some SOCA $\mathcal{A} = (Q, \mathbb{P}, \lambda, \delta)$. Our result is based on the following lemma, which expresses periodicity properties of reachability relations in $\mathcal{A}$.

Lemma 8. *There are naturals* $\tau, \varepsilon, \delta = \exp(|\mathcal{A}|)$ *with* $\varepsilon \geq n_{max}(\mathcal{A})$ *such that for each* $n, n', m, m' > \tau$ *with* $n \equiv n'$ *mod* δ *and* $m \equiv m'$ *mod* δ *the following statements hold for each* $q, q' \in Q$*:*

(1) If $m + \varepsilon < n$ and $m' + \varepsilon < n'$, then $q(n) \longrightarrow^*_{\mathcal{A}} q'(m)$ if, and only if, $q(n') \longrightarrow^*_{\mathcal{A}} q'(m')$.

(2) If $m > n + \varepsilon$ and $m' > n' + \varepsilon$, then $q(n) \longrightarrow^*_{\mathcal{A}} q'(m)$ if, and only if, $q(n') \longrightarrow^*_{\mathcal{A}} q'(m')$.

Section 4.3 will be devoted to sketching a proof of Lemma 8. Assume the constants τ, ε and δ from Lemma 8 to be fixed for the rest of this section. Let us define $\mathcal{M}(q, \varphi) = \{n \in \mathbb{N} : (T(\mathcal{A}), q(n)) \models \varphi\}$ for each control state $q \in Q$ and each EF formula φ over $\mathbb{P}$. For the PSPACE upper bound, we will show that $\mathcal{M}(q, \varphi)$ is ultimately periodic with period δ.

Lemma 9. *If $n \equiv n' \bmod \delta$, then $n \in \mathcal{M}(q, \varphi)$ if, and only if, $n' \in \mathcal{M}(q, \varphi)$, for each control state $q \in Q$, each* EF *formula φ over $\mathbb{P}$ and each $n, n' > \tau + |\varphi| \cdot \varepsilon + \delta$.*

Proof. Without loss of generality assume $n' > n$. We show $(T(\mathcal{A}), q(n)) \models \varphi$ if, and only if, $(T(\mathcal{A}), q(n + \delta)) \models \varphi$ by induction on $|\varphi|$, from which the statement will follow. We only consider the most interesting cases $\varphi = \mathsf{EX}\varphi'$ and $\varphi = \mathsf{EF}\varphi'$, the other cases are easy.

If $\varphi = \mathsf{EX}\varphi'$, we have $(T(\mathcal{A}), q(n)) \models \varphi$ if, and only if, there is some $q' \in Q$ and $z \in \mathbb{Z}$ such that $q \xrightarrow{\mathsf{add}(z)} q' \in \Delta$ and $(T(\mathcal{A}), q'(n+z)) \models \varphi'$. Since $n+z > \tau+|\varphi'|\cdot\varepsilon+\delta$, the induction hypothesis yields $(T(\mathcal{A}), q'(n + z)) \models \varphi'$ if, and only if, $(T(\mathcal{A}), q'(n + z + \delta)) \models \varphi'$. Hence $(T(\mathcal{A}), q(n)) \models \mathsf{EX}\varphi'$ if, and only if, $(T(\mathcal{A}), q(n + \delta)) \models \mathsf{EX}\varphi'$.

If $\varphi = \mathsf{EF}\varphi'$, we have $(T(\mathcal{A}), q(n)) \models \varphi$ if, and only if, there are $q' \in Q$, $m \in \mathbb{N}$ and ϱ such that $\varrho : q(n) \longrightarrow^*_{\mathcal{A}} q'(m)$ and $(T(\mathcal{A}), q(m)) \models \varphi'$. Suppose $m > \tau + |\varphi'| \cdot \varepsilon + \delta$ and no counter value less than δ occurs along ϱ, so in particular there is no zero test along ϱ. The induction hypothesis yields $(T(\mathcal{A}), q(m + \delta)) \models \varphi'$, and by shifting ϱ by δ the existence of a path $\varrho' : q(n + \delta) \longrightarrow^*_{\mathcal{A}} q(m + \delta)$ follows, hence $(T(\mathcal{A}), q(n + \delta)) \models \mathsf{EF}\varphi'$. Otherwise, if $m \leq \tau + |\varphi'| \cdot \varepsilon + \delta$ or a counter value less than δ occurs along ϱ, Lemma 8, Point (1) guarantees that $q(n) \rightarrow^*_{\mathcal{A}} q'(m)$ if, and only if, $q(n + \delta) \rightarrow^*_{\mathcal{A}} q'(m)$, which again allows us to conclude that $(T(\mathcal{A}), q(n)) \models \mathsf{EF}\varphi'$. The direction $(T(\mathcal{A}), q(n)) \models \varphi$ implies $(T(\mathcal{A}), q(n + \delta)) \models \varphi$ follows analogously. □

Theorem 10. EF *model checking of SOCA is* PSPACE*-complete.*

Proof. PSPACE-hardness has already been established in Section 4.1. For the upper bound, Algorithm 1 is an alternating algorithm that decides $(T(\mathcal{A}), q(n)) \models \varphi$ in PSPACE. For brevity, the cases $\varphi = \mathsf{AX}\varphi'$ and $\varphi' = \mathsf{AG}\varphi'$ have been left out, they are defined complementary to their EX respectively EF counterparts. We only sketch correctness of the case $\varphi = \mathsf{EF}\varphi'$ by induction on $|\varphi|$, all other cases are obviously correct. Let $m = max\{n + \varepsilon + \delta, \tau + |\varphi'| \cdot \varepsilon + \delta\}$. Suppose $T(\mathcal{A}), q(n)) \models \mathsf{EF}\varphi'$, there is some $q'(n')$ such that $q(n) \longrightarrow^*_{\mathcal{A}} q'(n')$ and $(T(\mathcal{A}), q'(n')) \models \varphi'$. If $n' > m$, Lemma 9 guarantees that there is $n'' \in [0, m]$ such that $T(\mathcal{A}), q'(n'')) \models \varphi'$, and Lemma 8, Point (2) yields $q(n) \longrightarrow^*_{\mathcal{A}} q'(n'')$, which by Proposition 1 can be checked in NP. By the induction hypothesis, Algorithm 1 returns *true* on input $q'(n'')$ and φ', which concludes the correctness proof. □

Algorithm 1. Fragment of the EF SOCA model checking algorithm

```
Input: EF formula $\varphi$, configuration $q(n)$ of $\mathcal{A}$
  case $\varphi = p$: return $p \in \lambda(q)$
  case $\varphi = \neg p$: return $p \notin \lambda(q)$
  case $\varphi = \varphi_1 \wedge \varphi_2$: return $(T(\mathcal{A}), q(n)) \models \varphi_1$ and $(T(\mathcal{A}), q(n)) \models \varphi_2$
  case $\varphi = \varphi_1 \vee \varphi_2$: return $(T(\mathcal{A}), q(n)) \models \varphi_1$ or $(T(\mathcal{A}), q(n)) \models \varphi_2$
  case $\varphi = \mathsf{EX}\varphi'$: existential move:
        choose $q \xrightarrow{\mathsf{op}} q' \in \Delta$
        case $\mathsf{op} = \mathsf{add}(z)$: return $(T(\mathcal{A}), q'(n+z)) \models \varphi'$
        case $\mathsf{op} = \mathsf{zero}$ and $n = 0$: return $(T(\mathcal{A}), q'(0)) \models \varphi'$
  case $\varphi = \mathsf{EF}\varphi'$: existential move:
        choose $q'(m)$ such that $q(n) \longrightarrow^*_{\mathcal{A}} q'(m)$ and $m \in [0, max\{n+\varepsilon+\delta, \tau+|\varphi'|\cdot\varepsilon+\delta\}]$
        return $(T(\mathcal{A}), q'(m)) \models \varphi'$
```

4.3 Proof Sketch of Lemma 8

In this section, we give a proof sketch of Lemma 8 which was left open in the previous section. The technical details are deferred to a full version of this paper.

On a technical level, it is helpful to view SOCA as *weighted graphs*, an approach also used in [9]. Given a SOCA $\mathcal{A}$, its corresponding weighted graph $G_{\mathcal{A}}$ is obtained by removing all zero-labeled edges from $\mathcal{A}$, and for every edge labeled with $\mathsf{add}(z)$, $G_{\mathcal{A}}$ has an edge labeled with z. Thus, we can assign any path π in $G_{\mathcal{A}}$ a *weight* $w(\pi)$ and a *drop* $d(\pi)$, which is the smallest weight of all prefixes of π. This allows us to relate runs in $T(\mathcal{A})$ with paths in $G_{\mathcal{A}}$: there is a zero-test free run $q(n) \longrightarrow^*_{\mathcal{A}} q'(n')$ if, and only if, there is a path π from q to q' in $G_{\mathcal{A}}$ with $w(\pi) = n' - n$ and $d(\pi) \geq -n$.

Let us fix a SOCA $\mathcal{A}$ and its corresponding graph G. In order to prove the periodicity properties expressed in Lemma 8, we will use cycles in G in order to construct paths whose weight is periodic for some period δ. For a start, let us concentrate on *cycles* in G with *negative weight*. Given a strongly connected component (SCC) S in G, we define $\gcd S$ as greatest common divisor of the set of all weights of all loop-free cycles in S. Note that $\gcd S = \mathsf{exp}(|\mathcal{A}|)$. It is easy to check that $\gcd S$ divides the weight of every cycle that runs through S, so $\gcd S$ could potentially serve as a period. However, if the weights of all cycles in S have the same sign, we cannot necessarily construct a cycle whose weight is an arbitrary multiple of $\gcd S$. For example, let $\{5, 7\}$ be the set of all weights of simple cycles in some SCC S with $S = \{q\}$ for some $q \in Q$. We have $\gcd S = 1$, however there is no cycle π in S with, say, $w(\pi) = 23$. This obstacle is related to the *Frobenius problem*, which is stated as follows [21]: given $x_1 < \ldots < x_n \in \mathbb{N}$ such that $\gcd\{x_1, \ldots, x_n\} = 1$, what is the *largest* $g \in \mathbb{N}$ such that g cannot be represented as non-negative integer linear combination of the x_i. It is shown in [21] that $g < x_n^2$. Thus in our example, this fact guarantees that there is a q-cycle π with $w(\pi) = m$ for every $m \geq 49$. The preceding observations allow us to conclude that once a certain threshold is crossed, we have periodicity of weights of cycles in an SCC.

Lemma 11. *There exists a* local threshold $\gamma \in \mathbb{N}$ *such that* $\gamma = \mathsf{exp}(|\mathcal{A}|)$ *and for all* $w, w' < -\gamma$ *and* $q \in Q$ *such that* $w \equiv w' \mod (\gcd S)$ *for some SCC* S *such that*

$q \in S$, whenever there exists a q-cycle π with $w(\pi) = w$ then there exists q-cycle π' with $w(\pi') = w'$ and $d(\pi') \geq w(\pi') - \gamma$.

Proving this lemma involves some tedious analysis of paths in G, but it is not too complicated. Note that the drop of π' does not get too large. We can now generalise Lemma 11 to *arbitrary* paths, and we define the *global period* δ as the least common multiple of $\gcd S$ of all SCCs in G. It is easily checked that $\delta = \mathsf{exp}(|\mathcal{A}|)$. Now consider an arbitrary q-q' path π in G with negative weight. If we find a q''-cycle π' along π with $w(\pi') < -\gamma$, we can invoke Lemma 11 in order to obtain a q''-cycle π'' with $w(\pi') \equiv w(\pi'') \mod \delta$. Thus, by using a counting argument on the number of control locations of $\mathcal{A}$, we can define a *global threshold* $\varepsilon = \mathsf{exp}(|\mathcal{A}|)$ that guarantees the existence of such a cycle. This allows us to state a variant of Lemma 11 for arbitrary paths:

Lemma 12. *For all $w, w' \in \mathbb{Z}$ such that $w, w' < -\varepsilon$ and $w \equiv w' \mod \delta$, whenever there exists a q-q' path π with $w(\pi) = w$ then there exists a q-q' path π' with $w(\pi') = w'$ and $d(\pi') \geq w(\pi') - \gamma$.*

We can now "re-import" the observations made for paths in weighted graphs to paths in $T(\mathcal{A})$ and sketch how to prove Lemma 8. To this end, we define $\tau \overset{\text{def}}{=} 2\varepsilon$. Regarding Point 1 of the lemma, we have that $min\{n, n'\} - min\{m, m'\} > \varepsilon$. Lemma 12 thus guarantees the existence of a path π with $w(\pi) = n - m$ if, and only if, there is a path π' with $w(\pi') = n' - m'$. Since $d(\pi) \geq w(\pi) - \tau$ and $m > \tau$, the existence of a run $q(n) \longrightarrow^*_{\mathcal{A}} q'(m)$ is guaranteed. The same argument yields a run $q(n') \longrightarrow^*_{\mathcal{A}} q'(m')$. Finally regarding Point 2, by using a symmetry argument, we can get a similar statement as in Lemma 12 for paths with positive weight that exceed ε. The existence of the desired runs then follows from an argument similar to Point 1.

5 Conclusion

We have strengthened our results from [6] and have proved that model checking the CTL fragment EF on POCA is undecidable via reduction from Hilbert's tenth problem. We showed that, when dropping the reachability modality, we regain decidability: Model checking ML on POCA is PSPACE-complete, which was proved by showing the existence of small solutions for a class of systems of linear Diophantine inequalities whose matrix norm is small. We showed that it is also PSPACE-complete to model check EF on SOCA by establishing an exponential periodicity property. It is interesting to mention that, in contrast to CTL, one can avoid an exponential complexity jump for EF and ML when model checking SOCA. More precisely, model checking EF (respectively ML) is P^{NP}-complete (respectively P-complete) on OCA, whereas it is PSPACE-complete for SOCA.

References

1. Bouajjani, A., Bozga, M., Habermehl, P., Iosif, R., Moro, P., Vojnar, T.: Programs with Lists are Counter Automata. In: Ball, T., Jones, R.B. (eds.) CAV 2006. LNCS, vol. 4144, pp. 517–531. Springer, Heidelberg (2006)

2. Bozga, M., Iosif, R.: On Decidability within the Arithmetic of Addition and Divisibility. In: Sassone, V. (ed.) FOSSACS 2005. LNCS, vol. 3441, pp. 425–439. Springer, Heidelberg (2005)
3. Bozga, M., Iosif, R., Lakhnech, Y.: Flat Parametric Counter Automata. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 577–588. Springer, Heidelberg (2006)
4. Chitic, C., Rosu, D.: On validation of xml streams using finite state machines. In: Proc. of WebDB, pp. 85–90. ACM, New York (2004)
5. Comon, H., Jurski, Y.: Multiple Counters Automata, Safety Analysis and Presburger Arithmetic. In: Vardi, M.Y. (ed.) CAV 1998. LNCS, vol. 1427, Springer, Heidelberg (1998)
6. Göller, S., Haase, C., Ouaknine, J., Worrell, J.: Model Checking Succinct and Parametric One-Counter Automata. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6199, pp. 575–586. Springer, Heidelberg (2010)
7. Göller, S., Lohrey, M.: Branching-time Model Checking of One-counter Processes. In: Proc. of STACS. LIPIcs, vol. 5, pp. 405–416. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2010)
8. Göller, S., Mayr, R., To, A.W.: On the Computational Complexity of Verifying One-Counter Processes. In: Proc. of LICS, pp. 235–244. IEEE Computer Society (2009)
9. Haase, C., Kreutzer, S., Ouaknine, J., Worrell, J.: Reachability in Succinct and Parametric One-Counter Automata. In: Bravetti, M., Zavattaro, G. (eds.) CONCUR 2009. LNCS, vol. 5710, pp. 369–383. Springer, Heidelberg (2009)
10. Hague, M., Lin, A.W.: Model Checking Recursive Programs with Numeric Data Types. In: Gopalakrishnan, G., Qadeer, S. (eds.) CAV 2011. LNCS, vol. 6806, pp. 743–759. Springer, Heidelberg (2011)
11. Ibarra, O.H., Dang, Z.: On the solvability of a class of diophantine equations and applications. Theor. Comput. Sci. 352(1), 342–346 (2006)
12. Ibarra, O.H., Jiang, T., Trân, N., Wang, H.: New decidability results concerning two-way counter machines and applications. In: Lingas, A., Carlsson, S., Karlsson, R. (eds.) ICALP 1993. LNCS, vol. 700, Springer, Heidelberg (1993)
13. Jančar, P., Kučera, A., Mayr, R.: Deciding bisimulation-like equivalences with finite-state processes. Theor. Comput. Sci. 258(1-2), 409–433 (2001)
14. Lange, M.: Model checking propositional dynamic logic with all extras. J. Applied Logic 4(1), 39–49 (2006)
15. Leroux, J., Sutre, G.: Flat Counter Automata Almost Everywhere! In: Peled, D.A., Tsay, Y.-K. (eds.) ATVA 2005. LNCS, vol. 3707, pp. 489–503. Springer, Heidelberg (2005)
16. Matiyasevich, Y.: Enumerable sets are Diophantine. Soviet Math. Dokl. 11, 354–357 (1970)
17. Minsky, M.L.: Recursive unsolvability of Post's problem of "tag" and other topics in theory of Turing machines. Annals of Mathematics. Second Series 74, 437–455 (1961)
18. Robinson, J.: Definability and Decision Problems in Arithmetic. J. Symbolic Logic 14(2), 98–114 (1949)
19. Schrijver, A.: Theory of linear and integer programming. John Wiley & Sons, Inc., New York (1986)
20. Serre, O.: Parity Games Played on Transition Graphs of One-Counter Processes. In: Aceto, L., Ingólfsdóttir, A. (eds.) FOSSACS 2006. LNCS, vol. 3921, pp. 337–351. Springer, Heidelberg (2006)
21. Shallit, J.: The Frobenius Problem and its Generalizations. In: Ito, M., Toyama, M. (eds.) DLT 2008. LNCS, vol. 5257, pp. 72–83. Springer, Heidelberg (2008)
22. To, A.W.: Model Checking FO(R) over One-Counter Processes and beyond. In: Grädel, E., Kahle, R. (eds.) CSL 2009. LNCS, vol. 5771, pp. 485–499. Springer, Heidelberg (2009)