On the Existential Theories of Büchi Arithmetic and Linear p-adic Fields

Florent Guépin
ENS Lyon, France
Email: florent.guepin@ens-lyon.fr

Christoph Haase
University of Oxford, UK
Email: christoph.haase@cs.ox.ac.uk

James Worrell
University of Oxford, UK
Email: jbw@cs.ox.ac.uk

***Abstract*—We consider the complexity of the satisfiability problems for the existential fragment of Büchi arithmetic and for the existential fragment of linear arithmetic over p-adic fields. Our main results are that both problems are NP-complete. The NP upper bound for existential linear arithmetic over p-adic fields resolves an open question posed by Weispfenning [*J. Symb. Comput.*, 5(1/2) (1988)] and holds despite the fact that satisfying assignments in both theories may have bit-size super-polynomial in the description of the formula. A key technical contribution is to show that the existence of a path between two states of a finite-state automaton whose language encodes the set of solutions of a given system of linear Diophantine equations can be witnessed in NP.**

I. Introduction

The computational complexity of first-order linear arithmetic over the integers (also called Presburger arithmetic) and of first-order linear arithmetic over the rational numbers has been thoroughly studied. In this paper we are concerned with extensions of linear arithmetic that encode divisibility information. One such extension is *Büchi arithmetic*. Given an integer $p \geq 2$, Büchi arithmetic of base p is the first-order theory of the structure $\langle \mathbb{N}, +, V_p \rangle$, where V_p maps every non-zero integer to its greatest divisor that is a power of p.

A basic result about Büchi arithmetic is that a subset $X \subseteq \mathbb{N}^n$ is first-order definable over $\langle \mathbb{N}, +, V_p \rangle$ if and only if X is p-*automatic*, that is, recognisable by an automaton under a base-p encoding of natural numbers. This result was first stated by Büchi [1] (in an incorrect form) and later reformulated and proved by Bruyère [2]. A consequence of this fact is that the first-order theory of $\langle \mathbb{N}, +, V_p \rangle$ is decidable for every $p \geq 2$. The decidability frontier cannot be pushed any further: the first-order theory of the structure $\langle \mathbb{N}, +, V_p, V_\ell \rangle$ is undecidable for multiplicatively independent $p, \ell \in \mathbb{N}$ [3]. The celebrated Cobham–Semënov theorem states that if $X \subseteq \mathbb{N}^n$ is separately definable in Büchi arithmetic over two multiplicatively independent bases p and ℓ, then X is definable in Presburger arithmetic [4], [5], [6], [7].

The second extension of linear arithmetic that we consider is linear arithmetic over the p-adic numbers $\mathbb{Q}_p$. Given a prime p and a non-zero rational number x, the p-adic valuation $v_p(x)$ is defined to be the unique integer $d \in \mathbb{Z}$ such that $x = p^d \cdot \frac{a}{b}$ with $a, b \in \mathbb{Z}$ and $p \nmid a, b$. Intuitively $v_p(x)$ is the exponent of the greatest power of p that divides x. There is a clear connection between the p-adic valuation v_p and the function V_p of Büchi arithmetic: namely for a natural number x we have $V_p(x) = p^{v_p(x)}$. Thus we could view $v_p(x)$ as a succinct representation of $V_p(x)$. Note, though, that we only consider the p-adic valuation in the case of prime p, unlike in Büchi arithmetic.

The field $\mathbb{Q}_p$ of p-adic numbers is obtained as the Cauchy completion of the field of rational numbers under an ultrametric obtained from the valuation v_p. Alongside their established importance in number theory, p-adic numbers have more recent applications in computer arithmetic [8], and many other areas such as physics. Decidability of the first-order theory of the valued field $\mathbb{Q}_p$ was shown by Ax and Kochen [9], [10] and Ershov [11]. These works used model-theoretic techniques that do not yield primitive recursive bounds. Later Cohen [12] gave a primitive recursive decision procedure using quantifier elimination in a two-sorted language, with one sort for elements of the field $\mathbb{Q}_p$ and another sort for the codomain $\mathbb{Z} \cup \{\infty\}$ of the valuation function (see also Macintyre et al. [13] and Weispfenning [14] for alternative approaches to quantifier elimination).

The first-order theory of *linear arithmetic* over $\mathbb{Q}_p$ (as well as general valued fields) has been studied by Weispfenning [15] and Sturm [16]. Both these works use a single-sorted formalism in which the valuation function is not explicitly mentioned, but rather the binary divisibility relation $v_p(a) \leq v_p(b)$ is taken as primitive. This relation can directly be expressed in the two-sorted language. It is shown in [15, Theorem 3.4] and [16, Corollary 11.1] that for every prime p the decision problem for the full first-order theory of linear arithmetic over $\mathbb{Q}_p$ is complete for the Berman complexity class $\mathrm{STA}(*, 2^{O(n)}, n)$, and hence can be solved in exponential space. Furthermore [15, Theorem 6.2] shows that the truth in $\mathbb{Q}_p$ of an existential sentence φ with m variables can be decided in time $\langle \varphi \rangle^{O(m)}$, where $\langle \varphi \rangle$ denotes the length of φ (i.e., the decision problem lies in EXPTIME). This upper bound has recently been improved to the counting hierarchy [17], an analogue of the polynomial hierarchy that contains the latter and is believed to be strictly contained in PSPACE. The concluding remarks of [15, Section 6] pose the question of whether the existential fragment of linear arithmetic over $\mathbb{Q}_p$ lies in NP.

The first main result of this paper is to show that the decision problem for the existential fragment of Büchi arithmetic is in NP. Here we regard the base p, given in binary, as part of the

978-1-7281-3608-0/19/$31.00

input. The second main result shows that the decision problem for the existential fragment of linear arithmetic over the p-adic numbers is in NP. Again, we consider the prime p in binary as part of the input. This last result resolves the above-mentioned problem of Weispfenning positively.

Unlike the case of existential Presburger arithmetic, the NP upper bound for existential Büchi arithmetic cannot be directly obtained by guessing and checking satisfying assignments. For example, it is known that for infinitely many primes q the multiplicative order $\mathrm{ord}_q(2)$ of 2 modulo q is at least $\sqrt{q}$ [18]. For such a prime the predicate *x is a strictly positive power of 2 that is congruent to 1 modulo q* can easily be expressed as a formula of existential Büchi arithmetic of base 2 that has length linear in the bit-length of q, while the smallest satisfying assignment is $x = 2^{\mathrm{ord}_q(2)}$. Thus satisfying assignments in existential Büchi arithmetic may have super-polynomial bit-length in the formula size, even for a fixed base.

Both main results rely on a key technical lemma, described in Section III, involving well-known automata-theoretic representations of solution sets of linear equations over nonnegative integers and over p-adic integers. We show that while the obtained automata have size exponential in the length of the equation systems, there is a nondeterministic procedure that decides state-to-state reachability in the automata and runs in time polynomial in the length of the underlying equation system.

II. Definitions and Main Results

We recall some basic definitions and results that will be used throughout. All numbers are assumed to be encoded in binary, unless otherwise stated. The L^2-norm of a vector $\boldsymbol{v} \in \mathbb{R}^n$ is $\|\boldsymbol{v}\|_2 := \sqrt{\sum_{i=1}^n |v_i|^2}$, while the L^∞-norm is $\|\boldsymbol{v}\|_\infty := \max_{i=1}^n |v_i|$. Given a matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ with components $a_{ij} \in \mathbb{Z}$, $1 \le i \le m$, $1 \le j \le n$, the $(1,\infty)$-norm of $\mathbf{A}$ is $\|\mathbf{A}\|_{1,\infty} := \max_{i=1}^m \sum_{j=1}^n |a_{ij}|$.

A. Büchi arithmetic

Given $p \ge 2$, let V_p denote the partial function on $\mathbb{N}$ with domain the set of all strictly positive integers and such that for all $a \neq 0$, $V_p(a) = b$ if and only if there exists $j \ge 0$ with $b = p^j$, $p^j \mid a$, and $p^{j+1} \nmid a$.[1] Büchi arithmetic of base p is the first-order theory of the structure $\langle \mathbb{N}, 0, 1, +, V_p\rangle$. Formally, we consider V_p as a binary relation, but we write $V_p(x) = y$ as shorthand for $(x, y) \in V_p$. Without loss of generality, we can assume that atomic formulas of Büchi arithmetic are either linear Diophantine equations $\boldsymbol{a}\cdot\boldsymbol{x} = c$ or assertions $V_p(x) = y$.

By application of the automata-based method described below, one can derive a polynomial-space upper bound for the satisfiability problem for the existential fragment of Büchi arithmetic. The first main result of this paper provides a tight complexity bound on this problem (where the base p, encoded in binary, is part of the input of the decision problem).

[1] An alternative formulation would be to include a special value ∞ in our structure, and define $V_p(0) = \infty$. However following this route does not change the class of definable subsets of $\mathbb{N}$.

Theorem 1. The satisfiability problem for existential Büchi arithmetic is NP-complete.

B. Linear arithmetic over p-adic fields

Fix an integer prime p. The *p-adic valuation* $v_p\colon \mathbb{Q} \to \mathbb{Z} \cup \{\infty\}$ is defined as follows. For every $a \in \mathbb{Z} \setminus \{0\}$, we define $v_p(a) := \max\{k : p^k \mid a\}$. If $a, b \in \mathbb{Z} \setminus \{0\}$ then we furthermore write $v_p(a/b) = v_p(a) - v_p(b)$. Finally we define $v_p(0) = \infty$. Note the relation between the valuation function v_p and the Büchi function V_p, namely $V_p(a) = p^{v_p(a)}$ for every $a \in \mathbb{Z} \setminus \{0\}$.

The p-adic valuation v_p induces a non-Archimedean absolute value $|\cdot|_p$ on $\mathbb{Q}$ that is defined by writing $|x|_p = p^{-v_p(x)}$ for all $x \in \mathbb{Q}$. The field $\mathbb{Q}_p$ of p-adic numbers is the Cauchy completion of $\mathbb{Q}$ with respect to the absolute value $|\cdot|_p$. Any p-adic number $x \in \mathbb{Q}_p \setminus \{0\}$ can be expressed as a *p-adic expansion*, i.e., as an infinite power series $x = \sum_{i=k}^\infty a_i p^i$ (that converges with respect to $|\cdot|_p$), where $k \in \mathbb{Z}$, $a_i \in \{0, \ldots, p-1\}$ for each i, and $a_k \neq 0$. The p-adic valuation extends to a map $v_p\colon \mathbb{Q}_p \to \mathbb{Z} \cup \{\infty\}$ with $v_p(x) = k$ for $x = \sum_{i=k}^\infty a_i p^i$ with $a_k \neq 0$.

A p-adic number x such that $v_p(x) \ge 0$ is called a *p-adic integer*. The p-adic expansion of such an x has the form $x = \sum_{i=0}^\infty a_i p^i$, where $a_i \in \{0, \ldots, p-1\}$ for each i. The p-adic expansion of every positive integer is simply its base-p expansion with least significant digit first. Note, however, that negative integers have infinite p-adic expansions, e.g., $-1 = \sum_{i=0}^\infty (p-1)p^i$ for all primes p.

Following Cohen [12], we work with a two-sorted first-order language L_{LVF} for linear arithmetic over valued fields. This language has a sort for the set of p-adic numbers $\mathbb{Q}_p$ and a sort for the set of *values* $\mathbb{Z}$. We are interested in linear arithmetic over $\mathbb{Q}_p$, so L_{LVF} includes a constant symbol of p-adic sort for each element of $\mathbb{Q}$, a constant symbol of value sort for each element of $\mathbb{Z}$, a binary function symbol $+$ on both the p-adic and value sorts, and a binary relation symbol denoting the p-adic valuation v_p. We will informally consider v_p as a partial function with domain $\mathbb{Q}_p \setminus \{0\}$, i.e., we write $v_p(x) = y$ instead of $(x, y) \in v_p$.

As remarked in the introduction, the papers [15], [16] study linear arithmetic over $\mathbb{Q}_p$ using a single-sorted formalism in which the valuation function is not directly mentioned, but rather the binary divisibility relation $v_p(a) \le v_p(b)$ is taken as primitive. The language of [15], [16] also has a constant symbol denoting p and further allows constants to be built using both product and sum. Thus, e.g., the atomic L_{LVF}-formula $v_p(x) = 2$ would be translated as $v_p(p\cdot p) \le v_p(x) \wedge v_p(x) \le v_p(p\cdot p)$ in the one-sorted language. Since the terms that can be built from the constant p using product and sum denote integers of bit-length polynomial in the size of the term, formulas in the one-sorted language can be translated into the language L_{LVF} with only a polynomial blow-up in formula size.

The concluding remarks of [15, Section 6] pose the question of whether the existential fragment of linear arithmetic over $\mathbb{Q}_p$ admits a decision procedure running in nondeterministic

polynomial time. The second main result of this paper resolves this question positively:

Theorem 2. The decision problem for existential sentences of linear arithmetic over $\mathbb{Q}_p$ (where p, given in binary, is regarded as part of the input) is NP-complete.

C. Automata and integer solutions of linear equations

Let $p \geq 2$ be an integer. A central concept in this paper is that of a p-automaton: a deterministic automaton whose language encodes a set of integers in base p. In this section, we define the notion of p-automaton and recall from [19] the construction that gives a representation of the set of nonnegative integer solutions of a system of linear equations by a finite-state p-automaton.

A system S of *linear Diophantine equations* has the form $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$, where $\mathbf{A}$ is an $m \times n$ matrix with integer coefficients, $\boldsymbol{c} \in \mathbb{Z}^m$, and $\boldsymbol{x} = (x_1, \ldots, x_n)^\top$ is a vector of variables taking values in the nonnegative integers. We write $[\![S]\!] := \{\boldsymbol{u} \in \mathbb{N}^n : \mathbf{A}\boldsymbol{u} = \boldsymbol{c}\}$ for the set of all nonnegative integer solutions of S. We denote by $\langle S \rangle$ the size of the encoding of S, i.e., the number of symbols required to represent S assuming binary encoding of all numbers.

Definition 3. A *deterministic automaton* is a tuple $A = (Q, \Sigma, \delta, q_0, F)$, where

- Q is a set of *states*,
- Σ is a finite alphabet,
- $\delta\colon Q \times \Sigma \to Q \cup \{\bot\}$, where $\bot \notin Q$, is the *transition function*,
- $q_0 \in Q$ is the *initial state*, and
- $F \subseteq Q$ is the set of *final states*.

Note that we allow automata to have infinitely many states and to have partially defined transition functions (due to the presence of $\bot$ in the codomain of δ).

For states $q, r \in Q$ and $u \in \Sigma$, we write $q \xrightarrow{u} r$ if $\delta(q, u) = r$, and extend $\to$ inductively to words by stipulating, for $w \in \Sigma^*$ and $u \in \Sigma$, that $q \xrightarrow{w \cdot u} r$ if there is $s \in Q$ such that $q \xrightarrow{w} s \xrightarrow{u} r$. We write $q \xrightarrow{*} r$ if there is some $w \in \Sigma^*$ such that $q \xrightarrow{w} r$. The *language of* A is defined as $L(A) = \{w \in \Sigma^* : q_0 \xrightarrow{w} q_f, q_f \in F\}$.

Given an integer $p \geq 2$, a *p-automaton* is a deterministic automaton over an alphabet $\Sigma_p^n := \{0, 1, \ldots, p-1\}^n$ for some nonnegative integer n. A finite word over alphabet Σ_p^n can naturally be seen as encoding an n-tuple of nonnegative integers in base p. In fact we consider two such encodings: the *lsd-first encoding*, in which the least significant digit is on the left, and the *msd-first encoding*, in which the most significant digit is on the left. Formally, given a word $w = \boldsymbol{u}_0 \cdots \boldsymbol{u}_k \in (\Sigma_p^n)^*$, we define $[\![w]\!]_l \in \mathbb{N}^n$ and $[\![w]\!]_m \in \mathbb{N}^n$ by

$$[\![w]\!]_l := \sum_{j=0}^{k} p^j \cdot \boldsymbol{u}_j \qquad [\![w]\!]_m := \sum_{j=0}^{k} p^{k-j} \cdot \boldsymbol{u}_j \,.$$

The subscripts l and m in the above definition indicate whether we consider the lsd-first or msd-first interpretation. Note also that for $w = \varepsilon$, the empty word, we have $[\![w]\!]_l = [\![w]\!]_m = \mathbf{0}$.

Following Wolper and Boigelot [19], we define a p-automaton whose language is the msd-first encoding all nonnegative integer solutions of systems of linear equations.

Definition 4. Let $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$ be a system of linear equations with integer coefficients such that $\mathbf{A}$ has dimension $m \times n$. Corresponding to S, we define a p-automaton $A_{msd}(S) := (Q, \Sigma_p^n, \delta, \boldsymbol{q}_0, F)$ such that

- $Q = \mathbb{Z}^m$,
- $\delta(\boldsymbol{q}, \boldsymbol{u}) = p \cdot \boldsymbol{q} + \mathbf{A}\boldsymbol{u}$ for all $\boldsymbol{q} \in Q$ and $\boldsymbol{u} \in \Sigma_p^n$,
- $\boldsymbol{q}_0 = \mathbf{0}$, and
- $F = \{\boldsymbol{c}\}$.

Although the automaton $A_{msd}(S)$ has infinitely many states, it defines a regular language since only finitely many states can reach the set F of accepting states; we call such states *live states* subsequently.

Proposition 5. Given automaton $A_{msd}(S)$, no state $\boldsymbol{q} \in Q$ such that $\|\boldsymbol{q}\|_\infty > \|\mathbf{A}\|_{1,\infty}$ and $\|\boldsymbol{q}\|_\infty > \|\boldsymbol{c}\|_\infty$ can reach an accepting state.

Proof. Suppose that state $\boldsymbol{q} \in Q$ is such that $\|\boldsymbol{q}\|_\infty > \|A\|_{1,\infty}$ and $\|\boldsymbol{q}\|_\infty > \|c\|_\infty$. Then for all $\boldsymbol{u} \in \Sigma_p^n$ we have

$$\begin{aligned} \|\delta(\boldsymbol{q}, \boldsymbol{u})\|_\infty &= \|p \cdot \boldsymbol{q} + \mathbf{A}\boldsymbol{u}\|_\infty \\ &\geq p \cdot \|\boldsymbol{q}\|_\infty - \|\mathbf{A}\boldsymbol{u}\|_\infty \\ &\geq p \cdot \|\boldsymbol{q}\|_\infty - \|\mathbf{A}\|_{1,\infty} \cdot \|\boldsymbol{u}\|_\infty \\ &> p \cdot \|\boldsymbol{q}\|_\infty - \|\boldsymbol{q}\|_\infty \cdot (p-1) \\ &= \|\boldsymbol{q}\|_\infty \,. \end{aligned}$$

In other words, the set of states $\{\boldsymbol{q} \in Q : \|\boldsymbol{q}\|_\infty > \max(\|\mathbf{A}\|_{1,\infty}, \|\boldsymbol{c}\|_\infty)\}$ is invariant under the transition relation of $A_{msd}(S)$ and excludes the accepting state $\boldsymbol{c}$. Hence no state in this set can reach an accepting state. □

It follows from Proposition 5 that a rough upper bound on the number $\#Q$ of states of $A_{msd}(S)$ is

$$\#Q \leq 2^m \cdot \max(\|\mathbf{A}\|_{1,\infty}, \|\boldsymbol{c}\|_\infty)^m \,, \tag{1}$$

where m is the number of equations in the system S.

A key reachability property of the automaton $A_{msd}(S)$ is the following:

Lemma 6. Let $\boldsymbol{q}, \boldsymbol{r} \in \mathbb{Z}^m$ be states of $A_{msd}(S)$. Then for all $k \in \mathbb{N}$ and words $w \in (\Sigma_p^n)^k$ we have

$$\boldsymbol{q} \xrightarrow{w} \boldsymbol{r} \iff \boldsymbol{r} = p^k \cdot \boldsymbol{q} + \mathbf{A}\,[\![w]\!]_m$$

Proof. Straightforward induction on k. □

It follows from Lemma 6 that the language of $A_{msd}(S)$ is an msd-first encoding of the set of solutions of the system $\mathbf{A}\boldsymbol{x} = \boldsymbol{c}$. Indeed, applying the lemma to the initial state $\mathbf{0}$ and the final state $\boldsymbol{c}$ of $A_{msd}(S)$, we have that $\mathbf{0} \xrightarrow{w} \boldsymbol{c}$ if and only if $\mathbf{A}[\![w]\!]_m = \boldsymbol{c}$.

If we wish to emphasise the underlying system S of linear Diophantine equations of a p-automaton $A_{msd}(S)$ we annotate the transition relation with the subscript S and, e.g., write $\boldsymbol{q} \xrightarrow{*}_S \boldsymbol{r}$.

From the bound (1), it follows that p-automata can be used to obtain a PSPACE upper bound for deciding feasibility over the nonnegative integers of a system of linear Diophantine equations. However, von zur Gathen and Sieveking [20] have shown that any feasible system of linear Diophantine equations has a solution whose bit-size is polynomially bounded in the encoding of S, which yields an NP bound for the feasibility problem. Thus, measured in terms of the encoding size $\langle S \rangle$ of the underlying linear system of equations, while the automaton $A_{msd}(S)$ has exponentially many (live) states, it accepts a word of length polynomial in $\langle S \rangle$.

The main technical result of this paper gives bounds on the complexity of the *reachability problem for p-automata*, i.e., deciding whether $\boldsymbol{q} \xrightarrow{*} \boldsymbol{r}$ for states $\boldsymbol{q}, \boldsymbol{r} \in \mathbb{Z}^m$ of a p-automaton $A_{msd}(S)$. If $A_{msd}(S)$ is explicitly given, this problem is of course trivial, but constructing $A_{msd}(S)$ from S incurs an exponential blow-up. We will establish an NP-upper bound for this problem in case the input consists of the base p encoded in binary, the system of equations S, and the states $\boldsymbol{q}$ and $\boldsymbol{r}$ of the induced automaton (note that by (1) the encoding of the (live) states of $A_{msd}(S)$ is polynomial in the size of S):

Theorem 7. The state-to-state reachability problem for p-automata of the form $A_{msd}(S)$ is NP-complete.

D. Automata and p-adic solutions of linear equations

Given an integer $p \geq 2$ and a system S of linear Diophantine equations, we define an automaton $A_{lsd}(S)$ that represents the base-p lsd-first encoding of the set of nonnegative integer solutions of S. It is clear that the language of $A_{lsd}(S)$ is the reverse of the language of automaton $A_{msd}(S)$. In fact, since $A_{msd}(S)$ is reverse-deterministic, $A_{lsd}(S)$ can be obtained by simply reversing the direction of the transitions in $A_{msd}(S)$ and interchanging the initial and accepting states. In particular, $A_{lsd}(S)$ has the same set of states as $A_{msd}(S)$, although A_{lsd} has a partial transition function in general, while A_{msd} has a total transition function.

An explicit definition of automaton $A_{lsd}(S)$ is as follows:

Definition 8. Let $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$ be a system of linear equations with integer coefficients, where $\mathbf{A}$ has dimension $m \times n$. Define $A_{lsd}(S) := (Q, \Sigma_p^n, \delta, \boldsymbol{q}_0, F)$ such that

- $Q = \mathbb{Z}^m$,
- for all $\boldsymbol{q} \in Q$ and $\boldsymbol{u} \in \Sigma_p^n$, $\delta(\boldsymbol{q}, \boldsymbol{u}) = \frac{\boldsymbol{q} - \mathbf{A}\boldsymbol{u}}{p}$ if $\boldsymbol{q} \equiv \mathbf{A}\boldsymbol{u} \pmod{p}$ and $\delta(\boldsymbol{q}, \boldsymbol{u}) = \bot$ otherwise,
- $q_0 = \boldsymbol{c}$, and
- $F = \{\mathbf{0}\}$.

As we now explain, in the case of a prime base p, automaton $A_{lsd}(S)$ can also be used to encode the set of all p-adic integer solutions of the system S. Given an infinite word $w = \boldsymbol{u}_0 \boldsymbol{u}_1 \ldots \in (\Sigma_p^n)^\omega$, we define a corresponding n-tuple of p-adic integers $[\![w]\!]_l$ by

$$[\![w]\!]_l := \sum_{i=0}^{\infty} p^i \cdot \boldsymbol{u}_i \,.$$

We show that such a word w encodes a p-adic integer solution of the system S whenever the automaton $A_{lsd}(S)$ has an infinite run on w, starting from its initial state. (This requirement is non-vacuous since the the transition function of $A_{lsd}(S)$ is partially defined.)

Let $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$ be a system of linear equations, where $\mathbf{A} \in \mathbb{Z}^{m \times n}$ and $\boldsymbol{c} \in \mathbb{Z}^m$. Given an infinite word $w = \boldsymbol{u}_0 \boldsymbol{u}_1 \ldots \in (\Sigma_p^n)^\omega$, we have $\mathbf{A}[\![w]\!]_l = \boldsymbol{c}$ if and only if $\mathbf{A}[\![\boldsymbol{u}_0 \ldots \boldsymbol{u}_{k-1}]\!]_l \equiv \boldsymbol{c} \pmod{p^k}$ for all $k \in \mathbb{N}$. But for all $k \in \mathbb{N}$ we have

$$\begin{aligned} &\mathbf{A}[\![\boldsymbol{u}_0 \ldots \boldsymbol{u}_{k-1}]\!]_l \equiv \boldsymbol{c} \pmod{p^k} \\ \iff &\mathbf{A}[\![\boldsymbol{u}_0 \ldots \boldsymbol{u}_{k-1}]\!]_l + \boldsymbol{r} \cdot p^k = \boldsymbol{c} \text{ for some } \boldsymbol{r} \in \mathbb{Z}^m \\ \iff &\mathbf{A}[\![\boldsymbol{u}_{k-1} \ldots \boldsymbol{u}_0]\!]_m + \boldsymbol{r} \cdot p^k = \boldsymbol{c} \text{ for some } \boldsymbol{r} \in \mathbb{Z}^m \end{aligned}$$

By Lemma 6, the last line in the above chain of equivalences expresses that state $\boldsymbol{r}$ can reach state $\boldsymbol{c}$ in automaton $A_{msd}(S)$ by reading the word $\boldsymbol{u}_{k-1} \ldots \boldsymbol{u}_0$. Thus for $w = \boldsymbol{u}_0 \boldsymbol{u}_1 \ldots \in (\Sigma_p^n)^\omega$ we have $\mathbf{A}[\![w]\!]_l = \boldsymbol{c}$ if and only if for every $k \geq 0$ there exists a state $\boldsymbol{r}$ of $A_{lsd}(S)$ such that $\boldsymbol{c} \xrightarrow{\boldsymbol{u}_0 \ldots \boldsymbol{u}_{k-1}} \boldsymbol{r}$ in $A_{lsd}(S)$. Since $A_{lsd}(S)$ is deterministic, this last condition is equivalent to the existence of an infinite run in $A_{lsd}(S)$ over the word w starting from state $\boldsymbol{c}$.

E. Semi-linear sets

Given finite sets $B, P \subseteq \mathbb{N}^n$ of *base* and *period vectors*, define

$$L(B, P) := \left\{ \boldsymbol{b} + \sum_{i=1}^{m} \lambda_i \cdot \boldsymbol{p}_i : \boldsymbol{b} \in B, \boldsymbol{p}_i \in P, \lambda_i \in \mathbb{N} \right\}.$$

If $B = \{\boldsymbol{b}\}$ is a singleton then we simply write $L(\boldsymbol{b}, P)$. We call $L(\boldsymbol{b}, P)$ a *linear set* and we say that a subset of $\mathbb{N}^n$ is *semi-linear* if it can be written as a finite union of linear sets. In particular, each set $L(B, P)$ is semi-linear. It is well-known that the set of nonnegative integer solutions of a system of linear Diophantine equations is a semi-linear set [20], [21].

We will denote mixed systems of equations and inequations by the notation $S\colon \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}$, where $\sim$ is a vector of relation symbols, with each entry either "$=$" or "$<$". Given a finite set $B \subseteq \mathbb{N}^n$, we denote by $\|B\|_\infty$ the quantity $\max\{\|\boldsymbol{b}\|_\infty : \boldsymbol{b} \in B\}$. We moreover denote by $\|S\|_{1,\infty}$ the $(1, \infty)$-norm of the matrix $\begin{pmatrix} \mathbf{A} & -\boldsymbol{c} \end{pmatrix}$. We use the following bound on the magnitude of the generators B and P of the set of solutions of a system of linear Diophantine equations, which is derived from [22].

Proposition 9. [23, Prop. 4] Let $S\colon \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}$ be a system of m linear Diophantine equations and inequalities in n variables. Then there exist finite sets $B, P \subseteq \mathbb{N}^n$ such that $[\![S]\!] = L(B, P)$ and

- $\|B\|_\infty \leq (\|S\|_{1,\infty} + 1)^{O(m+n)}$
- $\|P\|_\infty \leq (\|\mathbf{A}\|_{1,\infty} + 1)^{O(m+n)}$

Proposition 9 shows that bit-size of the entries of the vectors in B and P are polynomially bounded in $\langle S \rangle$. The following proposition shows that we can decompose $L(B, P)$ as a union of linear sets whose sets of period vectors are linearly independent and hence have cardinality bounded by

n while only increasing the size of the constants appearing by a polynomial factor.

Proposition 10. [23, Prop. 5] Let $M = L(B, P) \subseteq \mathbb{N}^n$, where $B, P \subseteq \mathbb{N}^n$ are finite. Then we can write $M = \bigcup_{i \in I} L(B_i, P_i)$ such that for all $i \in I$,

- $\|B_i\|_\infty \leq \|B\|_\infty + (\#P \cdot \|P\|_\infty)^{O(n)}$, and
- each P_i is a linearly independent subset of P (and hence $\#P_i \leq n$).

By combining Propositions 9 and 10, we derive the following corollary:

Corollary 11. Let $S\colon \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}$ be a system of m linear Diophantine equations and inequalities in n variables. Then we can write $[\![S]\!] = \bigcup_{i \in I} L(B_i, P_i)$ for some finite set I such that for all $i \in I$,

- $\|B_i\|_\infty \leq (\|S\|_{1,\infty} + 1)^{O((mn)^3)}$,
- $\|P_i\|_\infty \leq (\|\mathbf{A}\|_{1,\infty} + 1)^{O(m+n)}$, and
- $\#P_i \leq n$.

Recall that a set $M \subseteq \mathbb{N}$ is *ultimately periodic* if there is a threshold $t \in \mathbb{N}$ and a period $\ell \in \mathbb{N}$ such that for all $a, b \in \mathbb{N}$ with $a, b \geq t$ and $a \equiv b \bmod \ell$ we have $a \in M$ if and only if $b \in M$.

Semi-linear sets in dimension one are ultimately periodic. We will use the following result (paraphrased from Wilf [24]) in to obtain bounds on the threshold and period of a given linear set.

Proposition 12 ([24])**.** Let $M \subseteq \mathbb{N}$ be a finite set such that $\gcd(M) = 1$. Then the linear set $L(0, M)$ contains every natural number a such that $(\max M)^2 \leq a$.

From the above proposition it follows that a linear set $L(c, Q) \subseteq \mathbb{N}$ is ultimately periodic with threshold $c + \max(Q)^2$ and period $\gcd(Q)$.

We will also need the the following result of Chrobak and Martinez [25], [26], corrected by To [27], which is related to Proposition 12.

Proposition 13 ([25], [26], [27])**.** The language of an NFA with n states over a unary alphabet (considered in the natural way as a subset of $\mathbb{N}$) can be written as a union of linear sets $L(a, b)$ such that $a = O(n^2)$ and $b = O(n)$.

III. Deciding reachability in p-automata

The goal of this section is to prove that the reachability problem for p-automata lies in NP (Theorem 7). Recall that an instance of this problem consists of a base $p \geq 2$, a system S of linear Diophantine equations, and two states $\boldsymbol{q}, \boldsymbol{r}$ of the corresponding p-automaton $A_{msd}(S)$; the question is whether $\boldsymbol{q}$ can reach $\boldsymbol{r}$. Since automaton $A_{lsd}(S)$ is gotten by reversing the transition function of $A_{msd}(S)$, we immediately get that state-to-state reachability in $A_{lsd}(S)$ also lies in NP. Membership of these reachability problems in NP is key to showing that the respective satisfiability problems for existential Büchi arithmetic and existential linear arithmetic over p-adic numbers lie in NP.

It is classical that deciding satisfiability over nonnegative integers of systems of linear equations is NP-complete [28]. Since a system S of linear equations is satisfiable over nonnegative integers if and only if the initial state of $A_{msd}(S)$ can reach the final state, it follows that the reachability problem for p-automata is NP-hard. The main goal in the rest of the section is obtain an NP upper bound for reachability between arbitrary pairs of states.

Let $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$, be a system of linear Diophantine equations in n variables, with corresponding p-automaton $A_{msd}(S) = (Q, \Sigma_p^n, \delta, \boldsymbol{q}_0, F)$. Suppose further that $\boldsymbol{q}, \boldsymbol{r} \in Q$ are states of $A_{msd}(S)$ and we wish to decide whether $\boldsymbol{q}$ can reach $\boldsymbol{r}$. Lemma 6 implies that deciding whether $\boldsymbol{q} \xrightarrow{*} \boldsymbol{r}$ is equivalent to deciding whether there exist $k \in \mathbb{N}$ and $w \in (\Sigma_p^n)^k$ such that

$$\boldsymbol{r} = p^k \cdot \boldsymbol{q} + \mathbf{A}[\![w]\!]_m \,.$$

This problem can equivalently be rephrased as finding some $\boldsymbol{x} \in \mathbb{N}^n$ and $k \in \mathbb{N}$ satisfying the following linear-exponential system of Diophantine inequalities:

$$\mathbf{A}\boldsymbol{x} = \boldsymbol{r} - p^k \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty < p^k \,. \tag{2}$$

It will thus suffice to prove the following proposition.

Proposition 14. Deciding feasibility of linear-exponential systems of the form (2) is in NP.

Proof. In order to provide a polynomial-size witness of the feasibility of (2), we introduce the following system T of linear Diophantine inequalities:

$$T\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{r} - y \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty < y \,. \tag{3}$$

Since the constraint $\|\boldsymbol{x}\|_\infty < y$ can be expressed as $\bigwedge_{1 \leq i \leq n} x_i < y$, (3) is a system of linear inequalities. The bounds from Corollary 11 imply that the set $[\![T]\!]$ admits a semi-linear decomposition $[\![T]\!] = \bigcup_{i \in I} L(B_i, P_i) \subseteq \mathbb{N}^{n+1}$ such that for all $i \in I$ we have $\#P_i \leq n + 1$ and the vectors in B_i and P_i have entries of absolute value at most

$$(\|T\|_{1,\infty} + 1)^{poly(\langle T \rangle)} \,. \tag{4}$$

As a certificate that the linear-exponential system (2) is satisfiable, we shall use a linear set $L(\boldsymbol{b}, P) \subseteq \mathbb{N}^{n+1}$ subject to the following conditions:

- C1 $\#P \leq n + 1$,
- C2 vector $\boldsymbol{b}$ and all vectors in P have entries of absolute value at most $(\|T\|_{1,\infty} + 1)^{poly(\langle T \rangle)}$,
- C3 $L(\boldsymbol{b}, P) \subseteq [\![T]\!]$,
- C4 the projection of $L(\boldsymbol{b}, P)$ on the y-coordinate contains a power of p.

We first show that (2) is satisfiable if and only if such a certificate exists. Suppose that (2) is satisfiable. Given a solution $(\boldsymbol{x}^*, k^*)$ of (2), clearly $(\boldsymbol{x}^*, p^{k^*})$ is a solution of (3). In particular, there exist $i \in I$ and $\boldsymbol{b} \in B_i$ such that $(\boldsymbol{x}^*, p^{k^*}) \in L(\boldsymbol{b}, P_i)$. Using the bounds in (4), we see that the linear set $L(\boldsymbol{b}, P_i)$ satisfies Items C1–C4 above.

Conversely, suppose that some linear set $L(\boldsymbol{b}, P) \subseteq \mathbb{N}^{n+1}$ satisfies Items C1–C4. By Items C3 and C4, T has a solution of the form $(\boldsymbol{x}^*, p^{k^*})$. But then $(\boldsymbol{x}^*, k^*)$ is a solution of the system (2).

Now to prove the proposition it suffices to give a nondeterministic polynomial-time procedure for guessing and checking certificates satisfying Items C1–C4. Algorithm 1 is such a procedure. The input is a linear exponential system as in (2). Line 2 guesses a linear set $L(\boldsymbol{b}, P)$ satisfying the bounds in Items C1 and C2, which thereby has size polynomial in the encoding length of the input. The rest of the algorithm verifies that $L(\boldsymbol{b}, P)$ satisfies Items C3 and C4.

Line 3 of the algorithm verifies Item C3 in the definition of a certificate, i.e., that the linear set $L(\boldsymbol{b}, P)$ comprises solutions of the system T. For this one need only check that $\boldsymbol{b}$ is a solution of T and all vectors in P are solutions of the corresponding homogeneous system

$$\mathbf{A}\boldsymbol{x} = -y \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty \leq y\,.$$

This can clearly be done in polynomial time.

Lines 4–11 of the algorithm verify Item C4 in the definition of a certificate. Line 4 computes the linear set $L(c, Q) \subseteq \mathbb{N}$ obtained from projecting $L(\boldsymbol{b}, P)$ onto the y-coordinate (which can be done simply by projecting the base vector $\boldsymbol{b}$ and set of period vectors P onto the y-coordinate). It follows from Proposition 12 that $L(c, Q)$ is ultimately periodic with threshold $t = c + (\max Q)^2$ and period $\ell = \gcd Q$ (see Line 5). Consequently, if $p^k \in L(c, Q)$ for some $k \geq 0$ then $p^{k^*} \in L(c, Q)$ for some

$$0 \leq k^* < \lceil \log_p t \rceil + \ell\,. \tag{5}$$

The exponent k^* in (5) has bit-size polynomial in the size of the certificate $L(\boldsymbol{b}, P)$ and the membership of p^{k^*} in $L(c, Q)$ can be checked in nondeterministic polynomial time. There are two cases. If $p^{k^*} \leq t$ then checking $p^{k^*} \in L(c, Q)$ is an instance of integer programming (which is in NP); see Line 8. On the other hand, if $p^{k^*} > t$ then by ultimate periodicity of $L(c, Q)$ we have that $p^{k^*} \in L(c, Q)$ if and only if there exists $a \in \{t, t+1, \ldots, t+\ell-1\}$ such that $a \in L(c, Q)$ and $p^{k^*} \equiv a \bmod \ell$ (see Lines 10–11). A nondeterministic polynomial-time procedure can guess such an a and check $a \in L(c, Q)$ (an instance of integer programming) and $p^{k^*} \equiv a \bmod \ell$ (by iterated squaring).

This completes the proof of correctness of Algorithm 1. □

As explained above, it follows from Proposition 14 that deciding state-to-state reachability in p-automata of the form $A_{msd}(S)$ is in NP. Thus we have proven Theorem 7.

Remark 15. From the estimate in (5), we can only derive a doubly-exponential upper bound on the magnitude of solutions of (2), and hence only an exponential bound on the length of the shortest word between two states $\boldsymbol{q}$ and $\boldsymbol{r}$ in a p-automaton $A_{msd}(S)$. It remains an open problem whether there exist words of polynomial length between *any* two states, or whether this exponential bound is tight.

Algorithm 1 Procedure for deciding satisfiability of linear-exponential system $\mathbf{A}\boldsymbol{x} = \boldsymbol{r} - p^k \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty < p^k$.

1: **let** T: $\mathbf{A}\boldsymbol{x} = \boldsymbol{r} - y \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty < y$
2: **guess** $L(\boldsymbol{b}, P) \subseteq \mathbb{N}^{n+1}$ satisfying Items C1 and C2
3: **check** $L(\boldsymbol{b}, P) \subseteq [\![T]\!]$
4: **let** $L(c, Q) \subseteq \mathbb{N}$ be $L(\boldsymbol{b}, P)$ projected onto y-coordinate
5: **let** $t := c + (\max Q)^2$ and $\ell := \gcd Q$
6: **guess** $k^* \in \{0, 1, \ldots, \lceil \log_p t \rceil + \ell\}$
7: **if** $k^* \leq \log_p t$ **then**
8: **check** $p^{k^*} \in L(c, Q)$
9: **else**
10: **guess** $a \in \{t, t+1, \ldots, t+\ell-1\}$
11: **check** $a \in L(c, Q)$ and $p^{k^*} \equiv a \bmod \ell$

From the proof of Theorem 7, we directly obtain the following corollary.

Corollary 16. Let $\boldsymbol{q}, \boldsymbol{r}$ be states of the p-automaton $A_{msd}(S)$ corresponding to a system of linear Diophantine equations S, and let $m \in \mathbb{N}$. Deciding whether $\boldsymbol{q} \xrightarrow{w} \boldsymbol{r}$ for some word w of length m is in NP.

Proof. Use Algorithm 1, and instead of guessing k^* in Line 6 set k^* to m. □

A. Reachability to zero in p-automata

In this section we consider a special case of the reachability problem for p-automata of the form $A_{msd}(S)$. Whereas this problem is NP-complete in general, and already NP-hard when the initial state is the zero vector, we show that the version of the problem in which the target is fixed to be the zero vector is solvable in polynomial time. (Recall again that the p-automaton $A_m(S)$ in this problem is represently implicitly by the linear system S.) The contents of this section are not needed for main results of the paper and the reader primarily interested in the NP upper bounds for Büchi arithmetic and linear arithmetic over p-adic fields may safely skip the material here.

In Section III we reduced the reachability problem for p-automata to determining satisfiability of systems of linear-exponential Diophantine inequalities of the form shown in (2) over the variables $\boldsymbol{x} \in \mathbb{N}^n$ and $k \in \mathbb{N}$. If the target state $\boldsymbol{r}$ in the reachability problem is set to $\mathbf{0}$ then (2) specialises to a system of the following form:

$$\mathbf{A}\boldsymbol{x} = -p^k \cdot \boldsymbol{q},\ \|\boldsymbol{x}\|_\infty < p^k\,. \tag{6}$$

We will show that systems of this form are solvable in polynomial time.

Dividing both sides in (6) by p^k, it follows that (6) can be turned into an instance of the following variant of linear programming. In the definition below, recall that $\sim$ denotes a vector of equality and strict inequality relations.

GRADED LINEAR PROGRAMMING (GRADED LP)

INPUT: A polyhedron $P = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}\} \subseteq \mathbb{R}^n$ and $p > 1$.

QUESTION: Is there some $k \geq 0$ and $\boldsymbol{x}^* \in \mathbb{Z}^n$ such that $\boldsymbol{x}^*/p^k \in P$?

We show that, just as for classical linear programming, Graded LP is also decidable in polynomial time.

Theorem 17. Graded LP is decidable in polynomial time. Moreover if a graded LP is feasible then it admits solutions $\boldsymbol{x}^*/p^k$ such that the bit size of $\boldsymbol{x}^*/p^k$ is polynomial in the description of the LP.

The remainder of this section is devoted to proving Theorem 17. We first prove two technical lemmas. The first shows that for any ε, any real interval of length ε contains a point x^*/p^k for "small" x^* and k.

Lemma 18. Let $y^* \in \mathbb{Q}$ and $\varepsilon > 0$. Then there are $k \geq 0$ and $x^* \in \mathbb{Z}$ such that

- $x^*/p^k \in [y^*, y^* + \varepsilon)$,
- $k \leq 2 - \log_p \varepsilon$, and
- $|x^*| \leq (|y^*| + \varepsilon) \cdot p^k$.

Proof. Choose $k \in \mathbb{N}$ such that $1/p^k < \varepsilon$, e.g., $k = \lfloor 2 - \log_p \varepsilon \rfloor$. If k defined this way is negative then set $k = 0$. Now pick $x^* \in \mathbb{Z}$ such that $y^* \leq x^*/p^k < y^* + \varepsilon$, and observe that $|x^*| \leq (|y^*| + \varepsilon) \cdot p^k$. □

Clearly, both k and y are computable in polynomial time. The next lemma shows that if a polyhedron P has full dimension then there always exists some point $\boldsymbol{x}^*/p^k \in P$, and this point can be computed in polynomial time.

Lemma 19. Let $P = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}\}$ be a full-dimensional polyhedron. Then there are $k > 0$ and $\boldsymbol{x}^* \in \mathbb{Z}^n$ such that $\boldsymbol{x}^*/p^k \in P$. Moreover, k and $\boldsymbol{x}^*$ are computable in polynomial time and the bit size of $\boldsymbol{x}^*/p^k$ is polynomial in the description of P.

Proof. Since P has full dimension, there is some $\boldsymbol{y}^* \in \operatorname{int}(P)$, the interior of P, that is computable in polynomial time and of size polynomial in the description of P, cf. [29, p. 170]. In particular, $\boldsymbol{y}^*$ has distance at least d to every supporting hyperplane of P for some rational $d > 0$ of bit-size polynomial in the description of P. The idea is to perturb $\boldsymbol{y}^*$ in every component using Lemma 18 while staying inside P.

To this end, set $\varepsilon = d/(n+1)$. For every $1 \leq i \leq n$, let k_i and x_i be obtained from Lemma 18 by applying it to y_i^* and ε. Set $k = \max\{k_1, \ldots, k_n\}$, and let $\boldsymbol{x}^* \in \mathbb{Z}$ be such that $x_i^* = p^{k-k_i} \cdot x_i$. We claim that $1/p^k \cdot \boldsymbol{x}^* \in P$. It suffices to show that $\|\boldsymbol{y}^* - \boldsymbol{x}^*\|_2 < d$. But this is the case since $\boldsymbol{x}^*$ differs from $\boldsymbol{y}^*$ in every component by at most ε, and

$$\|\boldsymbol{y}^* - \boldsymbol{x}^*\|_2 \leq \sqrt{n \cdot (d/(n+1))^2} < d.$$

All computations can be performed in polynomial time and the bit-size of $\boldsymbol{x}^*$ and magnitude of k are polynomial in the description of P. □

We are now ready to prove Theorem 17. Let $P = \{\boldsymbol{x} : \mathbf{A}\boldsymbol{x} \sim \boldsymbol{c}\}$ and $p > 1$ be an instance of Graded LP. We can check in polynomial time whether P has full dimension, cf. [29, p. 170]. If this is the case then the statement immediately follows from Lemma 19. Otherwise we reduce to the full-dimensional case as follows.

If P does not have full dimension then P lies in a d-dimensional affine subspace of $\mathbb{R}^n$ for some $d < n$. This affine subspace is obtained as the intersection of all *implicitly defined equality constraints* of P, cf. [29, p. 100]. Here we say that P implicitly defines the equality $\boldsymbol{a} \cdot \boldsymbol{x} = c$ if there is an inequality $\boldsymbol{a} \cdot \boldsymbol{x} \geq c$ in the description of P such that $\boldsymbol{a} \cdot \boldsymbol{x}^* = c$ for all $\boldsymbol{x}^* \in P$. Such implicitly defined equalities can be identified in polynomial time using linear programming. Hence we can compute in polynomial time a submatrix $\mathbf{A}^=$ of $\mathbf{A}$, of row rank $d \leq n$, whose rows are the implicit equalities of P. Then P is an open subset of the hyperplane $P^= = \{\boldsymbol{x} : \mathbf{A}^=\boldsymbol{x} = \boldsymbol{c}^=\}$ and hence P is full dimensional in $P^=$.

We can compute in polynomial time the Hermite normal form $\mathbf{H}$ of $\mathbf{A}^=$ such that $\mathbf{H} = \mathbf{A}^=\mathbf{U} \in \mathbb{N}^{d \times n}$ for some unimodular $\mathbf{U} \in \mathbb{Z}^{n \times n}$ [29, p. 57]. Since $\mathbf{H} = \begin{pmatrix}\mathbf{B} & \mathbf{0}\end{pmatrix}$ for some lower triangular $\mathbf{B} \in \mathbb{N}^{d \times d}$, the system

$$Q = \{\boldsymbol{y} : \mathbf{H}\boldsymbol{y} = \boldsymbol{c}^=\} \subseteq \mathbb{R}^n$$

uniquely determines the first d components $y_1^*, \ldots, y_d^*$ of all $\boldsymbol{y}^* \in Q$. If there is a y_i^* such that $y_i^* \notin \mathbb{Z}/p^k$ for all $k \geq 0$ then (P, p) is a no-instance of Graded Linear Programming. Otherwise, consider the polyhedron

$$R = \{\boldsymbol{z} : \mathbf{A}\mathbf{U}(y_1^*, \ldots, y_d^*, z_1, \ldots, z_{n-d}) \sim \boldsymbol{c}\} \subseteq \mathbb{R}^{n-d}.$$

This polyhedron has full dimension, and as argued above, we can find some $\boldsymbol{z}^* \in R \cap \mathbb{Z}^{n-d}/p^j$ of polynomial bit size in polynomial time, from which we can derive the desired $\boldsymbol{x}^* = \mathbf{U}^{-1}(y_1^*, \ldots, y_d^*, z_1^*, \ldots, z_{n-d}^*) \in P \cap \mathbb{Z}^n/p^k$ for some $k \geq 0$. This completes the proof of Theorem 17.

IV. Deciding existential Büchi arithmetic

Based on the results in Section III, we now develop an NP upper bound for deciding satisfiability of formulas of existential Büchi arithmetic, thereby proving Theorem 1. Clearly it will suffice to show that the decision problem for the existential conjunctive fragment of Büchi arithmetic is in NP. Formulas in this fragment generalise classical integer programming and are of the form

$$\mathbf{A}\boldsymbol{x} = \boldsymbol{c} \wedge \bigwedge_{i \in I} V_p(x_i) = y_i \tag{7}$$

for an integer matrix $\mathbf{A}$ and integer vector $\boldsymbol{c}$.

It will first be useful to introduce a mild generalisation of the reachability relation for p-automata. Suppose we are given a linear system of equations $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$ and an additional system of constraints $T\colon \mathbf{B}\boldsymbol{x} = \boldsymbol{d}$. For all pairs of states $\boldsymbol{q}, \boldsymbol{r}$ of automaton $A_{msd}(S)$, write $\boldsymbol{q} \xrightarrow{w}_{S[T]} \boldsymbol{r}$ if $\boldsymbol{q} \xrightarrow{w}_S \boldsymbol{r}$ and $\mathbf{B}[\![w]\!]_m = \boldsymbol{d}$. Plainly $\boldsymbol{q} \xrightarrow{w}_{S[T]} \boldsymbol{r}$ if and only if

$$\begin{pmatrix}\boldsymbol{q} \\ \mathbf{0}\end{pmatrix} \xrightarrow{w}_{S \wedge T} \begin{pmatrix}\boldsymbol{r} \\ \boldsymbol{d}\end{pmatrix},$$

where $S \wedge T$ is the system of equations

$$S \wedge T\colon \begin{pmatrix}\mathbf{A} \\ \mathbf{B}\end{pmatrix} \boldsymbol{x} = \begin{pmatrix}\boldsymbol{c} \\ \boldsymbol{d}\end{pmatrix}.$$

It follows from Theorem 7 that it can be decided in nondeterministic polynomial time whether $\boldsymbol{q} \xrightarrow{w}_{S[T]} \boldsymbol{r}$ for some word $w \in (\Sigma_p^n)^*$, given as input the base $p \geq 2$, systems of equations S, T, and states $\boldsymbol{q}$, $\boldsymbol{r}$ of $A_{msd}(S)$.

It will suffice to prove an NP upper bound for the satisfiability problem for formulas in the existential conjunctive fragment of Büchi arithmetic. We first show an NP upper bound for the simple case in which there is a single valuation assertion $V_p(x) = y$ and then generalise to an arbitrary number of valuation assertions. Thus, consider the formula

$$\varphi\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c} \wedge V_p(x) = y\,,$$

built over the system of equations $S\colon \mathbf{A}\boldsymbol{x} = \boldsymbol{c}$, where x, y are particular variables in the vector $\boldsymbol{x}$. We only consider the case of satisfying assignments with $x \neq 0$; the case $x = 0$ can be dealt with separately in a trivial way.

The solutions of the system of equations $\mathbf{A}\boldsymbol{x} = \boldsymbol{c}$ are the values $[\![w]\!]_m$ for $w \in (\Sigma_p^n)^*$ such that $\mathbf{0} \xrightarrow{w}_S \boldsymbol{c}$. We now describe conditions on w that are equivalent to the additional requirement that $[\![w]\!]_m$ satisfy $V_p(x) = y$.

Observe that $V_p(x) = y$ constrains x and y such that in their base-p msd representation we have

- $x \in \{0, \ldots, p-1\}^k a 0^\ell$, and
- $y = 0^k 1 0^\ell$

for some $k, \ell \geq 0$ and $a \in \{1, \ldots, p-1\}$. Consequently, $[\![w]\!]_m$ satisfies $V_p(x) = y$ if and only if w admits a decomposition $w = s \cdot u \cdot t$ such that $s, t \in (\Sigma_p^n)^*$, $u \in \Sigma_p^n$ and

$$\mathbf{0} \xrightarrow{s}_{S[y=0]} \boldsymbol{d} \xrightarrow{u}_{S[x=a,y=1]} \boldsymbol{e} \xrightarrow{t}_{S[x=y=0]} \boldsymbol{c} \tag{8}$$

for some intermediate states $\boldsymbol{d}$ and $\boldsymbol{e}$ of automaton $A_{msd}(S)$. Now the intermediate states $\boldsymbol{d}$ and $\boldsymbol{e}$ can be guessed in polynomial time in $\langle\varphi\rangle$. Moreover, by Theorem 7, the reachability queries $\mathbf{0} \xrightarrow{*}_{S[y=0]} \boldsymbol{d}$ and $\boldsymbol{e} \xrightarrow{*}_{S[x=y=0]} \boldsymbol{c}$ can be checked in nondeterministic polynomial time, while checking $\boldsymbol{d} \to_{S[x=a,y=1]} \boldsymbol{e}$ is trivial. This gives the desired NP upper bound for this special case.

Generalising this approach to an arbitrary number of valuation assertions $\bigwedge_{i\in I} V(x_i) = y_i$ is straightforward: we nondeterministically guess an ordering among the y_i, which induces a decomposition as in (8) with at most $O(|I|)$ intermediate states that can also be guessed in polynomial time; validity of this decomposition can again be checked via Theorem 7. This completes the proof of NP-completeness of existential Büchi arithmetic.

V. Deciding existential linear arithmetic over p-adic fields

Recall from Section II-B the two-sorted language L_{LVF} for linear arithmetic over valued fields. Given an integer prime p, we are interested in deciding the truth of sentences of the form

$$\exists z_1 \ldots \exists z_m \exists u_1 \ldots \exists u_n \varphi\,, \tag{9}$$

where the variables z_i range over $\mathbb{Z}$, the variables u_i range over $\mathbb{Q}_p$, and φ is a boolean combination of atomic formulas. For proving that this problem lies in NP we can assume without loss of generality that φ has the form

$$\begin{aligned} &\mathbf{A}\boldsymbol{u} = \boldsymbol{c} \wedge \textstyle\bigwedge_{i=1}^{\ell} v_p(u_i) = z_i \\ &\wedge\, \mathbf{B}\boldsymbol{z} \sim \boldsymbol{d} \wedge \textstyle\bigwedge_{i=1}^{\ell-1} z_i < z_{i+1}\,, \end{aligned} \tag{10}$$

where matrices $\mathbf{A}, \mathbf{B}$ and vectors $\boldsymbol{c}, \boldsymbol{d}$ have integer coefficients. (In particular, one can guess a strict linear ordering on the integer variables in φ and replace disequalities $\boldsymbol{a} \cdot \boldsymbol{u} \neq 0$ with $\exists u' \exists z\, (\boldsymbol{a} \cdot \boldsymbol{u} = u' \wedge v_p(u') = z)$.)

We introduce a particularly simple class of quantifier-free L_{LVF}-formulas that have no variables of integer sort. A system of *linear equations with valuation constraints* is a formula

$$\mathbf{A}\boldsymbol{u} = \boldsymbol{b} \wedge \bigwedge_{i\in I} v_p(u_i) = c_i, \tag{11}$$

where the entries of $\mathbf{A}$, $\boldsymbol{b}$, and the c_i are integers, and $\boldsymbol{u}$ is a vector of p-adic variables. To decide whether (11) has a solution, we may without loss of generality assume that $c_i \geq 0$ for all $i \in I$, i.e., that the u_i in (11) are all p-adic integers.

We will first give a non-deterministic polynomial-time reduction of the general decision problem for the existential linear theory of $\mathbb{Q}_p$ to the satisfiability problem for linear equations with valuation constraints. We then show that the latter problem is NP-complete.

Proposition 20. There is a nondeterministic polynomial-time reduction of the decision problem for existential L_{LVF}-sentences over $\mathbb{Q}_p$ to the satisfiability problem for systems of linear equations with valuation constraints over $\mathbb{Q}_p$.

Proof. Consider a quantifier-free L_{LVF}-formula φ, as specified in (10). At a high level the idea is to show that if φ is satisfiable then it admits a satisfying assignment in which the integer variables have bit-length bounded by a polynomial in $\langle\varphi\rangle$. Having established this, the values of the integer variables can be guessed in polynomial time, yielding a system of linear equations with valuation constraints.

Denote by $S\colon \mathbf{A}\boldsymbol{u} = \boldsymbol{c}$ the system of linear equations in p-adic variables appearing as a conjunct of φ. Furthermore, define

$$\begin{aligned} W := \{(v_p(u_1), v_p(u_2), \ldots, v_p(u_\ell)) \in \mathbb{N}^\ell \colon \\ \mathbf{A}\boldsymbol{u} = \boldsymbol{c} \wedge 0 \leq v_p(u_1) < \ldots < v_p(u_\ell)\} \end{aligned}$$

Since $A_{msd}(S)$ has a run over $w \in (\Sigma_p^n)^\omega$ if and only if $[\![w]\!]_l \in (\mathbb{Q}_p)^n$ is a solution of S, we have that $(k_1, \ldots, k_\ell) \in W$ if and only if there exist

- states $\boldsymbol{q}_1, \boldsymbol{r}_1, \ldots, \boldsymbol{q}_\ell, \boldsymbol{r}_\ell$ of $A_{msd}(S)$,
- words $w_1, \ldots, w_\ell \in (\Sigma_p^n)^*$,
- letters $a_1, \ldots, a_\ell \in \Sigma_p^n$,
- non-zero digits $d_1, \ldots, d_\ell \in \{1, \ldots, p-1\}$,

such that $k_1 = |w_1|$, $k_2 = |w_1| + |w_2| + 1$, etc.,

$$\begin{aligned} \boldsymbol{c} &\xrightarrow{w_1}_{S[u_1,\ldots,u_\ell=0]} \boldsymbol{q}_1 \xrightarrow{a_1}_{S[u_1=d_1,u_2,\ldots,u_\ell=0]} \boldsymbol{r}_1 \\ &\xrightarrow{w_2}_{S[u_2,\ldots,u_\ell=0]} \boldsymbol{q}_2 \xrightarrow{a_2}_{S[u_3=d_2,\ldots,u_\ell=0]} \boldsymbol{r}_2 \\ &\xrightarrow{w_3}_{S[u_2,\ldots,u_\ell=0]} \cdots \xrightarrow{a_\ell}_{S[u_\ell=d_\ell]} \boldsymbol{r}_\ell\,, \end{aligned} \tag{12}$$

and $A_{lsd}(S)$ has an infinite run on $w_{\ell+1}$ starting in state $\boldsymbol{r}_\ell$.

Combining Proposition 13 and the bound (1) on the number of live states of $A_{msd}(S)$, we get that for any two live states $\boldsymbol{q}$ and $\boldsymbol{r}$ of $A_{msd}(S)$, the set

$$\{k \in \mathbb{N} : \text{there is } w \in (\Sigma_p^n)^k \text{ such that } \boldsymbol{q} \xrightarrow{w}_S \boldsymbol{r}\}$$

can be written as a union of linear sets $L(\boldsymbol{b}, P)$ such that $\boldsymbol{b}$ and all vectors in P have entries of absolute value at most

$$2^{2m} \cdot \max(\|\mathbf{A}\|_{1,\infty}, \|\boldsymbol{c}\|_\infty)^{2m}\,.$$

It follows that there is a polynomial $P_1(\cdot)$ such that the set W can be written as union of semi-linear sets that each have description length at most $P_1(\langle\varphi\rangle)$.

Now $\boldsymbol{z}^* \in \mathbb{N}^m$ satisfies the formula $\exists u_1 \ldots \exists u_n \varphi$ if and only if $(z_1^*, \ldots, z_\ell^*) \in W$, $\mathbf{B}\boldsymbol{z}^* = \boldsymbol{d}$ and $z_1^* < \ldots < z_\ell^*$. Combining the description of W as a semi-linear set and the classical bounds of von zur Gathen and Sieveking [20] on integer solutions of systems of linear inequalites, it follows that there exists a polynomial $P_2(\cdot)$ such that if the formula (10) admits a satisfying assignment then there exists a satisfying assignment under which the values of the integer variables have bit-length at most $P_2(\langle\varphi\rangle)$. This completes the proof. □

Proposition 21. Satisfiability of systems of linear equations with valuation constraints over $\mathbb{Q}_p$ is in NP.

Proof. The proof parallels the approach to showing an NP upper bound for Büchi arithmetic.

Given a system of linear equations and valuation constraints of the form (11) above, write $I = \{i_1, \ldots, i_\ell\}$, and assume with no loss of generality that $c_{i_j} < c_{i_{j+1}}$ for all $1 \leq j \leq \ell$. Recall that the set of all p-adic integer solutions of the system of equations S in (11) is encoded by the p-automaton $A_{lsd}(S)$. A p-adic solution of (11) is encoded by a word $w \in (\Sigma_p^n)^\omega$ corresponding to some infinite run of $A_{lsd}(S)$ starting in state $\boldsymbol{b}$. As in the proof of Proposition 20, we can write

$$w = w_1 a_1 w_2 a_2 \cdots w_\ell a_\ell w_{\ell+1}\,,$$

where

- $w_1, \ldots, w_\ell \in (\Sigma_p^n)^*$ and $w_{\ell+1} \in (\Sigma_p^n)^\omega$,
- $a_1, \ldots, a_\ell \in \Sigma_p^n$,
- $c_1 = |w_1|$, $c_2 = |w_1| + |w_2| + 1$, etc.,
- there exist $d_1, \ldots, d_\ell \in \{1, \ldots, p-1\}$ with

$$\begin{aligned}\boldsymbol{b} &\xrightarrow{w_1}_{S[u_{i_1},\ldots,u_{i_\ell}=0]} \boldsymbol{q}_1 \xrightarrow{a_1}_{S[u_{i_1}=d_1,u_{i_2},\ldots,u_{i_\ell}=0]} \boldsymbol{r}_1 \\ &\xrightarrow{w_2}_{S[u_{i_2},\ldots,u_{i_\ell}=0]} \boldsymbol{q}_2 \xrightarrow{a_2}_{S[u_{i_2}=d_2,\ldots,u_{i_\ell}=0]} \boldsymbol{r}_2 \\ &\xrightarrow{w_3}_{S[u_{i_2},\ldots,u_{i_\ell}=0]} \cdots \xrightarrow{a_\ell}_{S[u_{i_\ell}=d_\ell]} \boldsymbol{r}_\ell \end{aligned} \quad (13)$$

- $A_{lsd}(S)$ has an infinite run on $w_{\ell+1}$ starting in $\boldsymbol{r}_\ell$.

Now $w_{\ell+1}$ being infinite, we additionally find $\boldsymbol{r}$ such that

$$\boldsymbol{r}_\ell \xrightarrow{*} \boldsymbol{r} \xrightarrow{+} \boldsymbol{r}. \quad (14)$$

To summarise, we can guess all intermediate states in the decomposition (13) and (14) in nondeterministic polynomial time and also check in nondeterministic polynomial time, via Corollary 16, that they are connected by some words w_i of appropriate length as required in (13). This completes the proof of the NP upper bound of Proposition 21. □

Proposition 22. For any fixed prime $p \geq 5$, determining satisfiability of systems of linear equations with valuation constraints over $\mathbb{Q}_p$ is NP-hard.

Proof. We reduce from the NP-complete 3-SAT problem: Given a Boolean formula $\psi \equiv \bigwedge_{1 \leq i \leq k} L_1^i \vee L_2^i \vee L_3^i$ in 3-CNF over Boolean variables $X_1, \ldots, X_n$, 3-SAT is to decide whether ψ has a satisfying assignment. We construct a system of linear equations with valuation constraints that is satisfiable if and only if ψ is satisfiable.

To this end, we introduce p-adic variables $x_1, \overline{x_1}, \ldots, x_n, \overline{x_n}$ with the intended meaning that $v_p(x_i) = 0$ means that X_i is set to true, and $v_p(\overline{x_i}) = 0$ means that X_i is set to false. Of course, we have to introduce constraints that ensure that only exactly one of those cases occurs. For every $1 \leq i \leq n$, consider

$$\bigwedge_{0<j<p} v_p(x_i + j \cdot \overline{x_i}) = 0. \quad (15)$$

Then any solution of (15) enforces that $v_p(x_i) \geq 0$ for all $1 \leq i \leq n$. To see this, suppose there is some x_i such that $v_p(x_i) = k < 0$. Then

$$x_i = a_k \cdot p^k + a_{k+1} \cdot p^{k+1} + \cdots$$

and since by (15) $v_p(x_i + \overline{x_i}) = 0$ we get

$$\overline{x_i} = (p - a_k) \cdot p^k + (p - a_{k+1}) \cdot p^{k+1} + \cdots\,.$$

But (15) also implies $v_p(x_i - \overline{x_i}) = 0$, since $p-1 \equiv -1 \bmod p$, from which we get

$$\overline{x_i} = (a_k - p) \cdot p^k + (a_{k+1} - p) \cdot p^{k+1} + \cdots$$

and which implies $p - a_k = a_k - p$ and hence $a_k = p$, a contradiction. Symmetrically, we obtain $v_p(\overline{x_i}) \geq 0$ for all $1 \leq i \leq n$.

Let us now show that in any solution of (15) we have either $v_p(x_i) = 0$ or $v_p(\overline{x_i}) = 0$. To the contrary, assume $v_p(x_i) > 0$ and $v_p(\overline{x_i}) > 0$. Since $v_p(y + z) \geq \min(v_p(y), v_p(z))$ for any p-adic numbers y, z such that $y + z \neq 0$, and $v_p(0) = \infty$, we have that $v_p(x_i + \overline{x_i}) = 0$ asserted by (15) is violated. Thus, it remains to show that we cannot have $v_p(x_i) = 0$ and $v_p(\overline{x_i}) = 0$ simultaneously. To the contrary, assume $x_i = a_0 \cdot p^0 + a_1 \cdot p^1 + \cdots$ and $\overline{x_i} = b_0 \cdot p^0 + b_1 \cdot p^1 + \cdots$ for some $a_0, b_0 \neq 0$. But then $a_0 + j \cdot b_0 \equiv 0 \bmod p$ for $0 \neq j \equiv b_0^{-1} a_0 \bmod p$, and hence $v_p(x_i + j \cdot \overline{x_i}) > 0$ contradicting $v_p(x_i + j \cdot \overline{x_i}) = 0$ asserted in (15).

Finally, we provide an encoding of ψ. For a literal L, let $h(L)$ be such that $h(X_i) = x_i$ and $h(\neg X_i) = \overline{x_i}$. We additionally assert

$$\bigwedge_{1 \leq i \leq k} v_p(h(L_1^i) + h(L_2^i) + h(L_3^i)) = 0. \quad (16)$$

It is now not difficult to see that ψ is satisfiable whenever the conjunction of (15) and 16 is. If ψ has a satisfying assignment we set $x_i = 1$ and $\overline{x_i} = p$ if X_i is set to true, and

symmetrically $x_i = p$ and $\overline{x_i} = 1$ if X_i is set to false. Then the first digit of any $h(L_1^i) + h(L_2^i) + h(L_3^i)$ is at most $3 < p$ since $p \geq 5$, and thus $v_p(h(L_1^i) + h(L_2^i) + h(L_3^i)) = 0$. One easily checks that the remaining constraints in (15) and (16) are satisfied.

Conversely, if (15) and (16) are satisfied, we define a satisfying assignment of ψ by setting X_i to true if and only if $v_p(x_i) = 0$. By the arguments above, this assignment is well-defined. Moreover, since $v_p(h(L_1^i) + h(L_2^i) + h(L_3^i)) = 0$ for all $1 \leq i \leq k$, in every clause of ψ there is at least one literal that is set to true since for $v_p(h(L_1^i) + h(L_2^i) + h(L_3^i)) = 0$ to hold, at least one of $v_p(h(L_1^i)$, $v_p(h(L_2^i)$ or $v_p(h(L_3^i)$ equals zero. This completes the proof. □

Remark 23. While we believe it to be the case, it remains an open problem whether an NP lower bound can also be established for the cases $p = 2, 3$.

Theorem 2, stating that the decision problem for existential linear arithmetic over $\mathbb{Q}_p$ lies in NP, follows directly from Propositions 20 and 21.

VI. Conclusion

We have shown that the satisfiability problems for existential Büchi arithmetic and existential linear arithmetic over the p-adic numbers are both NP-complete. The latter result resolves an open problem posed by Weispfenning [15]. The key technical result was to show that reachability in the class of p-automata that respectively give msd-first and lsd-first encodings in base p of sets of solutions of systems of linear equations is in NP. Note that this NP bound does not involve explicitly constructing those automata. To the best of our knowledge, this is the first time that the automata-theoretic approach to deciding theories of linear arithmetic has contributed novel upper bounds for linear arithmetic decision problems.

References

[1] J. R. Büchi, "Weak second-order arithmetic and finite automata," *Math. Logic Quart.*, vol. 6, no. 16, pp. 66–92, 1960.

[2] V. Bruyère, "Entiers et automates finis," *Mémoire de fin détudes*, 1985.

[3] R. Villemaire, "The theory of $(\mathbb{N}, +, v_k, v_l)$ is undecidable," *Theor. Comput. Sci.*, vol. 106, no. 2, pp. 337–349, 1992.

[4] A. Cobham, "On the base-dependence of sets of numbers recognizable by finite automata," *Math. Syst. Theory*, vol. 3, no. 2, pp. 186–192, jun 1969.

[5] A. L. Semenov, "Presburgerness of predicates regular in two number systems," *Sib. Math. J.*, vol. 18, no. 2, pp. 289–300, 1977.

[6] A. A. Muchnik, "The definable criterion for definability in Presburger arithmetic and its applications," *Theor. Comput. Sci.*, vol. 290, no. 3, pp. 1433–1444, 2003.

[7] V. Bruyère, G. Hansel, C. Michaux, and R. Villemaire, "Logic and p-recognizable sets of integers," *Bull. Belg. Math. Soc. Simon Stevin*, vol. 1, no. 2, pp. 191–238, 1994.

[8] X. Caruso, "Computations with p-adic numbers," *arXiv preprint arXiv:1701.06794*, 2017.

[9] J. Ax and S. Kochen, "Diophantine problems over local fields i,ii," *Am. J. Math.*, vol. 87, pp. 605–648, 1965.

[10] ——, "Diophantine problems over local fields iii," *Ann. Math.*, vol. 83, pp. 437–456, 1966.

[11] J. L. Ershov, "On the elementary theory of maximal normed fields," *Doklady Akad. Nauk USSR*, vol. 165, pp. 21–23, 1965.

[12] P. J. Cohen, "Decision procedures for real and p-adic fields," *Commun. Pur. Appl. Math.*, vol. 22, no. 2, pp. 131–151, 1969.

[13] A. Macintyre, K. McKenna, and L. van den Dries, "Elimination of quantifiers in algebraic structures," *Adv. Math.*, vol. 47, no. 1, pp. 74–87, 1983.

[14] V. Weispfenning, "Quantifier elimination and decision procedures for valued fields," in *Models and Sets*, ser. Lect. Notes Math., vol. 1103. Springer, 1984, pp. 419–472.

[15] ——, "The complexity of linear problems in fields," *J. Symb. Comput.*, vol. 5, no. 1/2, pp. 3–27, 1988.

[16] T. Sturm, "Linear problems in valued fields," *J. Symb. Comput.*, vol. 30, no. 2, pp. 207–219, 2000.

[17] A. Lechner, J. Ouaknine, and J. Worrell, "On the complexity of linear arithmetic with divisibility," in *Logic in Computer Science (LICS)*, 2015, pp. 667–676.

[18] C. R. Matthews, "Counting Points Modulo p for some Finitely Generated Subgroups of Algebraic Groups," *Bull. Lond. Math. Soc.*, vol. 14, no. 2, pp. 149–154, 1982.

[19] P. Wolper and B. Boigelot, "On the construction of automata from linear arithmetic constraints," in *Tools and Algorithms for the Construction and Analysis of Systems, TACAS*, ser. Lect. Notes Comp. Sci., vol. 1785. Springer, 2000, pp. 1–19.

[20] J. von zur Gathen and M. Sieveking, "A bound on solutions of linear integer equalities and inequalities," *P. Am. Math. Soc.*, vol. 72, no. 1, pp. 155–158, 1978.

[21] S. Ginsburg and E. H. Spanier, "Bounded ALGOL-like languages," *T. Am. Math. Soc.*, pp. 333–368, 1964.

[22] L. Pottier, "Minimal solutions of linear Diophantine systems: Bounds and algorithms," in *Rewriting Techniques and Applications, RTA*, ser. Lect. Notes Comp. Sci., vol. 488. Springer, 1991, pp. 162–173.

[23] D. Chistikov and C. Haase, "The taming of the semi-linear set," in *International Colloquium on Automata, Languages, and Programming, ICALP*, ser. LIPIcs, vol. 55. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, pp. 128:1–128:13.

[24] H. S. Wilf, "A circle-of-lights algorithm for the "money-changing problem"," *Am. Math. Mon.*, vol. 85, no. 7, pp. 562–565, 1978.

[25] M. Chrobak, "Finite automata and unary languages," *Theor. Comput. Sci.*, vol. 47, no. 3, pp. 149–158, 1986.

[26] A. Martinez, "Efficient computation of regular expressions from unary NFAs," *Descriptional Complexity of Formal Systems, DCFS*, pp. 174–187, 2002, Report No. 586, Department of Computer Science, The University of Western Ontario, Canada.

[27] A. W. To, "Unary finite automata vs. arithmetic progressions," *Inf. Process. Lett.*, vol. 109, no. 17, pp. 1010–1014, 2009.

[28] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, ser. The IBM Research Symposia Series. Plenum Press, New York, 1972, pp. 85–103.

[29] A. Schrijver, *Theory of linear and integer programming*. John Wiley & Sons, 1986.