Bounds on the Automata Size for Presburger Arithmetic

FELIX KLAEDTKE
ETH Zurich

Automata provide a decision procedure for Presburger arithmetic. However, until now only crude lower and upper bounds were known on the sizes of the automata produced by the automata-based approach for Presburger arithmetic. In this paper, we give an upper bound on the number of states of the minimal deterministic automaton for a Presburger arithmetic formula. This bound depends on the length of the formula and the quantifiers occurring in it. We establish the upper bound by comparing the automata for Presburger arithmetic formulas with the formulas produced by a quantifier-elimination method. We show that our bound is tight, also for nondeterministic automata. Moreover, we provide automata constructions for atomic formulas and establish lower bounds for the automata for linear equations and inequations.

Categories and Subject Descriptors: F.1.1 [**Computation by Abstract Devices**]: Models of Computation—*automata*; F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*computational logic*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Automata-based Decision Procedures, Presburger Arithmetic, Quantifier Elimination, Complexity

1. INTRODUCTION

Presburger arithmetic (PA) is the first-order theory with addition and the ordering relation over the integers. A number of decision problems can be expressed in it, such as solvability of systems of linear Diophantine equations, integer programming, and various problems in system verification. The decidability of PA was established around 1930 independently by Presburger [1930; 1984] and Skolem [1931; 1970] using the method of quantifier elimination.

Due to the applicability of PA in various domains, its complexity and the complexity of decision problems for fragments of it have been investigated intensively. For example, Fischer and Rabin [1974; 1998] gave a double exponential nondeterministic time lower bound on any decision procedure for PA. Later, Berman [1980] showed that the decision problem for PA is complete in the complexity class $LATIME(2^{2^{O(n)}})$, i. e., the class of problems solvable by alternating Turing ma-

This work was partially supported by the German Research Foundation (DFG) and the Swiss National Science Foundation (SNF).
Author's address: Felix Klaedtke, ETH Zurich, Department of Computer Science, Haldeneggsteig 4/Weinbergstraße, 8092 Zurich, Switzerland; email: `felixkl@inf.ethz.ch`.

chines in time $2^{2^{O(n)}}$ with a linear number of alternations. The upper bound for PA is established by a result from Ferrante and Rackoff [1979] showing that quantified variables need only to range over a restricted finite domain of integers. Grädel [1988] and Schöning [1997] investigated the complexity of decision problems of fragments of PA.

The complexity of different decision procedures for PA has also been studied, e. g., in [Oppen 1978; Reddy and Loveland 1978; Ferrante and Rackoff 1975; 1979]. For instance, Oppen [1978] showed that Cooper's quantifier-elimination decision procedure for PA [Cooper 1972] has a triple exponential worst case complexity in deterministic time. Reddy and Loveland [1978] improved Cooper's quantifier elimination and used it for obtaining space and deterministic time upper bounds for checking the satisfiability of PA formulas in which the number of quantifier alternations is bounded.

Another approach for deciding PA or fragments of it that has recently become popular is to use automata; a point that was already made by Büchi [1960]. The idea is simple: Integers are represented as words, e. g., using the 2's complement representation, and the word automaton (WA) for a formula accepts precisely the words that represent the integers making the formula true. The WA can be recursively constructed from the formula, where automata constructions handle the logical connectives and quantifiers. This automata-based approach for PA led to deep theoretical insights, e. g., the languages that are regular in any base are exactly the sets definable in PA [Cobham 1969; Semenov 1977; Bruyère et al. 1994]. More recently, the use of automata has been proposed for mechanizing decision procedures for PA and for manipulating sets definable in PA [Boudet and Comon 1996; Wolper and Boigelot 1995]. Roughly speaking, this applied use of WAs for PA is similar to the use of binary decision diagrams (BDDs) for propositional logic. For example, the automata library LASH [LASH] provides tool support for manipulating PA definable sets using automata to represented these sets, and it has been successfully used to verify systems with variables ranging over the integers. Other model checkers that use WAs for computing the potential infinite sets of reachable states of systems with integer variables are, e. g., FAST [Bardin et al. 2003] and ALV [Yavuz-Kahveci et al. 2005].

A crude complexity analysis of automata-based decision procedures for PA leads to a non-elementary worst case complexity. Namely, for every quantifier alternation there is a potential exponential blow-up. However, experimental comparisons [Shiple et al. 1998; Bartzis and Bultan 2003; Ganesh et al. 2002] illustrate that automata-based decision procedures for PA often perform well in comparison with other methods. In [Boudet and Comon 1996], the authors claimed that the minimal deterministic WA for a PA formula has at most a triple exponential number of states in the length of the formula. Unfortunately, as explained by Wolper and Boigelot [2000], the argument used in [Boudet and Comon 1996] to substantiate this claim is incorrect. Wolper and Boigelot [2000] gave an argument why there must be an elementary upper bound on the size of the minimal deterministic WA for a PA formula. However, their argumentation is rather sketchy and only indicates that there has to be an elementary upper bound.

In this paper, we rigorously prove an upper bound on the size of the minimal deterministic WA for PA formulas and thus, answer a long open question. Namely, for a PA formula in prenex normal form, we show that the minimal deterministic WA has at most $2^{n^{(b+1)^{a+4}}}$ states, where n is the formula length, a is the number of quantifier alternations, and b is the maximal length of the quantifier blocks. A similar upper bound holds for arbitrary PA formulas. This bound on the automata size for PA contrasts with the upper bound on the automata size for the monadic second-order logic WS1S, or even WS1S with the ordering relation "$<$" as a primitive but without quantification over monadic second-order variables. There, the number of states of the minimal WA for a formula can be non-elementarily larger than the formula's length [Stockmeyer 1974; Reinhardt 2002]. In order to establish the upper bound on the automata size for PA, we give a detailed analysis of the deterministic WAs for formulas by comparing the constructed WAs with the quantifier-free formulas produced by using Reddy and Loveland's quantifier-elimination method. From this analysis, we obtain the upper bound on the size of the minimal deterministic WA for PA formulas.

We also show that the upper bound on the size of deterministic WAs for formulas is tight. In fact, we show a stronger result. Namely, we give a family of Presburger arithmetic formulas for which even a nondeterministic WA has at least triple exponentially many states.

Furthermore, we investigate the automata constructed from atomic formulas. Specific algorithms for constructing WAs for linear (in)equations have been developed in [Boudet and Comon 1996; Boigelot 1999; Wolper and Boigelot 2000; Bartzis and Bultan 2003; Ganesh et al. 2002]. We give upper and lower bounds on the automata size for linear (in)equations and we improve some of the automata constructions in [Boigelot 1999; Wolper and Boigelot 2000; Ganesh et al. 2002] for linear (in)equations. We prove that our automata constructions are optimal in the sense that the constructed deterministic WAs are minimal.

We proceed as follows. In §2, we give background. In §3, we investigate the WAs for quantifier-free formulas. In §4, we prove the upper bound on the size of the minimal deterministic WA for PA formulas and in §5, we give a worst case example. Finally, in §6, we draw conclusions.

2. PRELIMINARIES

2.1 Presburger Arithmetic

Presburger arithmetic (PA) is the first-order logic over the structure $\mathfrak{Z} := (\mathbb{Z}, <, +)$. We use standard notation. For instance, for a term $t(x_1, \dots, x_r)$ and $a_1, \dots, a_r \in \mathbb{Z}$, $t[a_1, \dots, a_r]$ is the integer when the binary function symbol $+$ is interpreted as integer addition and the variable x_i is interpreted as the integer a_i, for $1 \leq i \leq r$. Analogously, we write $\mathfrak{Z} \models \varphi[a_1, \dots, a_r]$ for a formula $\varphi(x_1, \dots, x_r)$ and $a_1, \dots, a_r \in \mathbb{Z}$ if φ is true in $\mathfrak{Z}$ when the variable x_i is interpreted as the integer a_i, for $1 \leq i \leq r$. For a formula $\varphi(x_1, \dots, x_r)$, we define $[\![\varphi]\!] := \{(a_1, \dots, a_r) \in \mathbb{Z}^r : \mathfrak{Z} \models \varphi[a_1, \dots, a_r]\}$.

2.1.1 *Extended Logical Language.* We extend the logical language of PA by (i) constants for the integers 0 and 1, (ii) the unary operation "$-$" for integer negation, and (iii) the unary predicates "$d|$" for the relation "divisible by d," for each $d \geq 2$. These constructs are definable in PA, e. g., the formula $\exists x(x+\cdots+x=t)$ defines $d|t$, where x occurs d times in the term $x+\cdots+x$ and x does not appear in the term t. The reason for the extended logical language, where (i), (ii), and (iii) are treated as primitives, is that it admits quantifier elimination, i. e., for a formula $\exists x\varphi(x,\overline{y})$, where φ is quantifier-free, we can construct a logically equivalent quantifier-free formula $\psi(\overline{y})$.

Additionally, we allow the relation symbols $\leq, >, \geq$, and $\neq$ with their standard meanings. In the following, we assume that terms and formulas are defined in terms of the extended logical language for PA. We denote by PA the set of all Presburger arithmetic formulas over the extended logical language and QF denotes the set of quantifier-free formulas.

For convenience, we use standard symbols when writing terms. For instance, c stands for $1+\cdots+1$ (repeated c times) if $c>0$, and $-(1+\cdots+1)$ if $c<0$. We call the term c a *constant* and identify the term c with the integer that it represents. Analogously, we write $k\cdot x$ for $x+\cdots+x$ (repeated k times) if $k>0$, and $-(x+\cdots+x)$ if $k<0$. Moreover, if $k=0$ then $k\cdot x$ abbreviates $x+(-x)$. We say that k is a *coefficient*. For a term t and $k\in\mathbb{Z}$, $k\cdot t$ denotes the term where the constant and the coefficients in t are multiplied by k.

A term t is *homogeneous* if it is either 0 or of the form $k_1\cdot x_1+\cdots+k_r\cdot x_r$, for some $r\geq 1$, where the variables $x_1,\ldots,x_r$ are pairwise distinct and $k_1,\ldots,k_r\in\mathbb{Z}\setminus\{0\}$. The *normalized form* of $t_1 \lesseqgtr t_2$, with $\lesseqgtr\in\{=,\neq,<,\leq,>,\geq\}$, is the logically equivalent (in)equation $t\lesseqgtr c$, where summands of the form $k\cdot x$ in t_1 and t_2 are collected on the left-hand side t and constants in t_1 and t_2 are collected on the right-hand side c according to standard calculation rules. The *normalized form* of $d|t$ is the formula $d|t'+c$, where $c\in\mathbb{Z}$ is the sum of the constants in t and t' is the homogeneous term in which the coefficients of the summands of the form $k\cdot x$ in t are collected. We use $\mathsf{A}(\varphi)$ to denote the set of atomic formulas occurring in $\varphi\in\mathsf{PA}$ in their normalized forms.

2.1.2 *Formula Length.* The *length* of a formula is the number of letters used in writing the formula. Note that the length of a formula depends significantly on how we define the length of coefficients and constants. For instance, $x=10\cdot y$ contains 6 letters, namely, x, $=$, 1, 0, $\cdot$, and y. The "expanded version" has $2+19$ letters since $10\cdot y$ abbreviates the term $y+y+y+y+y+y+y+y+y+y$. We use the same definition of the length of a formula as in [Oppen 1978; Fischer and Rabin 1974; Reddy and Loveland 1978]. In particular, the length of a coefficient or constant is the number of letters of the expanded version. However, it is possible to express $k\cdot x$ by a formula of length $\mathrm{O}(\log|k|)$. The idea is illustrated by $x=10\cdot y$: the formula is logically equivalent to $\exists z(x=z+z\wedge\exists x(z=x+x+y\wedge x=y+y))$. Note that we only need a fixed number of variables for any k (see [Fischer and Rabin 1974]). For the sake of uniformity, we define the length of the formula $d|t$ as the length of the term t plus $d+1$. Again, there is a logically equivalent formula of length $\mathrm{O}(\log d)$ plus the length of t. For the results in this paper it does not matter if we define the length of an integer k as $\mathrm{O}(\log|k|)$ or as $\mathrm{O}(|k|)$.

2.1.3 *Nesting of Quantifiers.* It is well-known that we obtain coarse complexity bounds for checking satisfiability if we only take into account the formula length.

We obtain more precise complexity bounds when we additionally account for the number of quantifiers and the number of quantifier alternations.

The *quantifier number* of $\varphi \in \mathsf{PA}$ is the number of quantifiers occurring in φ, i. e.,

$$\mathrm{qn}(\varphi) := \begin{cases} \mathrm{qn}(\psi) & \text{if } \varphi = \neg\psi, \\ \mathrm{qn}(\psi_1) + \mathrm{qn}(\psi_2) & \text{if } \varphi = \psi_1 \oplus \psi_2 \text{ with } \oplus \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}, \\ 1 + \mathrm{qn}(\psi) & \text{if } \varphi = Qx\psi \text{ with } Q \in \{\exists, \forall\}, \\ 0 & \text{otherwise.} \end{cases}$$

For a quantifier $Q \in \{\exists, \forall\}$, $\overline{Q}$ denotes its dual, i. e., $\overline{Q} := \forall$ if $Q = \exists$, and $\overline{Q} := \exists$ if $Q = \forall$. The number of *quantifier alternations* of $\varphi \in \mathsf{PA}$ is

$$\mathrm{qa}(\varphi) := \min\{\mathrm{qa}_\exists(\varphi), \mathrm{qa}_\forall(\varphi)\}\,,$$

where

$$\mathrm{qa}_Q(\varphi) := \begin{cases} \mathrm{qa}_{\overline{Q}}(\psi) & \text{if } \varphi = \neg\psi, \\ \max\{\mathrm{qa}_Q(\psi_1), \mathrm{qa}_Q(\psi_2)\} & \text{if } \varphi = \psi_1 \oplus \psi_2 \text{ with } \oplus \in \{\vee, \wedge\}, \\ \mathrm{qa}_Q(\neg\psi_1 \vee \psi_2) & \text{if } \varphi = \psi_1 \rightarrow \psi_2, \\ \mathrm{qa}_Q((\psi_1 \rightarrow \psi_2) \wedge (\psi_2 \rightarrow \psi_1)) & \text{if } \varphi = \psi_1 \leftrightarrow \psi_2, \\ 1 + \mathrm{qa}_{\overline{Q}}(\psi) & \text{if } \varphi = \overline{Q}x\psi, \\ \max\{1, \mathrm{qa}_Q(\psi)\} & \text{if } \varphi = Qx\psi, \\ 0 & \text{otherwise,} \end{cases}$$

for $Q \in \{\exists, \forall\}$.

2.2 Automata over Finite Words

The set of all words over an alphabet Σ is denoted by Σ^*, Σ^+ denotes the set of all non-empty words over Σ^*, and λ denotes the *empty word*. The *length* of the word $w \in \Sigma^*$ is denoted by $|w|$.

A *deterministic word automaton* (DWA) is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_\mathrm{I}, F)$, where Q is a finite set of states, Σ is a finite alphabet, $\delta : Q \times \Sigma \to Q$ is the transition function, $q_\mathrm{I} \in Q$ is the initial state, and $F \subseteq Q$ is the set of accepting states. The *size* of $\mathcal{A}$ is the cardinality of Q. The *language* of $\mathcal{A}$ is $L(\mathcal{A}) := \{w \in \Sigma^* : \widehat{\delta}(q_\mathrm{I}, w) \in F\}$, where $\widehat{\delta}(q, \lambda) := q$ and $\widehat{\delta}(q, wb) := \delta(\widehat{\delta}(q, w), b)$, for $q \in Q$, $b \in \Sigma$, and $w \in \Sigma^*$. A state $q \in Q$ is *reachable* from $p \in Q$ if there is a word $w \in \Sigma^*$ such that $\widehat{\delta}(p, w) = q$.

Let $\mathcal{A} = (Q, \Sigma, \delta, q_\mathrm{I}, F)$ be a DWA. The states $p, q \in Q$ are *equivalent*, $p \sim_\mathcal{A} q$ for short, if for all $w \in \Sigma^*$, we have that $\widehat{\delta}(p, w) \in F$ iff $\widehat{\delta}(q, w) \in F$. We omit the subscript of the relation $\sim_\mathcal{A}$ if $\mathcal{A}$ is clear from the context. Note that $\sim \subseteq Q \times Q$ is an equivalence relation. We denote the equivalence class of $q \in Q$ by $\widetilde{q}$. By merging equivalent states, we obtain the DWA $\widetilde{\mathcal{A}} := (\{\widetilde{q} : q \in Q\}, \Sigma, \widetilde{\delta}, \widetilde{q}_\mathrm{I}, \{\widetilde{q} : q \in F\})$ with $\widetilde{\delta}(\widetilde{q}, b) := \widetilde{\delta(q, b)}$, for $q \in Q$ and $b \in \Sigma$. Obviously, we have that $L(\widetilde{\mathcal{A}}) = L(\mathcal{A})$.

A DWA $\mathcal{A}$ is *minimal* if for every DWA $\mathcal{B}$ with $L(\mathcal{B}) = L(\mathcal{A})$, either $\mathcal{B}$ has more states than $\mathcal{A}$ or $\mathcal{B}$ is isomorphic to $\mathcal{A}$. By the Myhill-Nerode theorem (see [Hopcroft and Ullman 1979]), a DWA is minimal iff every state is reachable from the initial state and there are no two distinct states that are equivalent. It follows that if $\mathcal{A}$

is a DWA, where every state is reachable from the initial state then $\widetilde{\mathcal{A}}$ is minimal. Note that we can assume without loss of generality that all states in a DWA are reachable from the initial state, since the states that are not reachable from the initial state do not affect the language of the DWA and hence, we can eliminate them.

3. AUTOMATA CONSTRUCTIONS

In this section, we investigate automata for quantifier-free PA formulas. In §3.1, we define how DWAs recognize sets of integers. In §3.2, we provide automata constructions for linear (in)equations and prove that the constructed automata are minimal, and in §3.3, we give an automata construction for the divisibility relation. Finally, in 3.4, we give an upper bound on the size of the minimal DWA for a quantifier-free formula.

3.1 Representing Sets of Integers with Automata

We use an idea that goes back at least to Büchi [1960] for using automata to recognize tuples of numbers by mapping words to tuples of numbers. There are many possibilities to represent integers as words. We use an encoding similar to [Boigelot 1999; Wolper and Boigelot 2000], which is based on the ϱ's complement representation of integers, where $\varrho \geq 2$ and the most significant bit is the first digit. For the remainder of the paper, we fix $\varrho \geq 2$ and let Σ be the alphabet $\{0, \dots, \varrho - 1\}$.

Definition 3.1. For $b_{n-1} \dots b_0 \in \Sigma^*$, we define $\langle b_{n-1} \dots b_0 \rangle_{\mathbb{N}} := \sum_{0 \leq i < n} \varrho^i b_i$. We generalize this encoding to integers as follows. For $b_n b_{n-1} \dots b_0 \in \Sigma^+$, we define

$$\langle b_n b_{n-1} \dots b_0 \rangle_{\mathbb{Z}} := \langle b_{n-1} \dots b_0 \rangle_{\mathbb{N}} - \begin{cases} 0 & \text{if } b_n = 0, \\ \varrho^n & \text{if } b_n \neq 0. \end{cases}$$

We call the first letter b_n the *sign letter*, since it determines whether the word represents a positive or a negative number.

Note that the empty word λ does not represent an integer. This requirement saves us from considering some special cases in §3.2.2 and §3.2.3 where we optimize the automata constructions for (in)equations. However, for the natural numbers, it holds that $\langle \lambda \rangle_{\mathbb{N}} = 0$. Furthermore, note that the encoding of an integer is not unique. First, we have that $\langle bu \rangle_{\mathbb{Z}} = \langle bcu \rangle_{\mathbb{Z}}$, where $b, c \in \Sigma$ and $u \in \Sigma^*$ with $c = 0$ if $b = 0$ and $c = \varrho - 1$, otherwise. Second, it holds that $\langle bu \rangle_{\mathbb{Z}} = \langle b'u \rangle_{\mathbb{Z}}$, for all $u \in \Sigma^*$ and $b, b' \in \Sigma \setminus \{0\}$, i.e., the sign letter $b \neq 0$ can be replaced by any other letter $b' \neq 0$. The motivation for allowing any letter to be the sign letter is that we do not have to deal with words in Σ^+ that do not represent an integer. This eliminates case distinctions of the automata constructions in the next subsections.

We extend the encoding to tuples of natural numbers and integers as follows: A word $w := \overline{b}_{n-1} \dots \overline{b}_0 \in (\Sigma^r)^*$ *represents* the tuple $\overline{a} := (a_1, \dots, a_r) \in \mathbb{N}^r$ of integers, where the ith "track" of the word w encodes the natural number a_i. That is, for all $1 \leq i \leq r$, we have that $a_i = \langle b_{n-1,i} \dots b_{0,i} \rangle_{\mathbb{N}}$, where $\overline{b}_j = (b_{j,1}, \dots, b_{j,r})$ for $0 \leq j < n$. Analogously, we can use a word $w = \overline{b}_n \overline{b}_{n-1} \dots \overline{b}_0 \in (\Sigma^r)^+$ to represent an integer tuple $\overline{z} = (z_1, \dots, z_r) \in \mathbb{Z}^r$. The first letter $\overline{b}_n$ of w is the *sign letter* since it determines the signs of the integers $z_1, \dots, z_r$. We define $\sigma(\overline{b}_n) := (c_1, \dots, c_r)$,

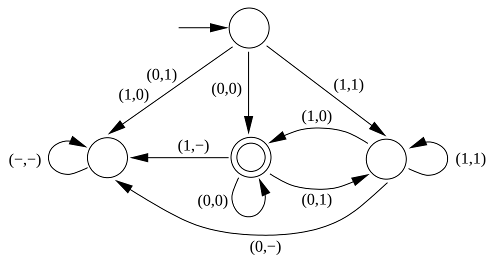

Fig. 1. DWA over the alphabet $\{0,1\}^2$ representing the set $\{(x,y) \in \mathbb{Z}^2 : y = 2x\}$.

where $c_i = 0$ if the ith coordinate of $\overline{b}_n$ is 0 and $c_i = -1$, otherwise, for each $1 \leq i \leq r$. We abuse notation and write $\langle w \rangle_{\mathbb{N}}$ to denote the tuple $\overline{a} \in \mathbb{N}^r$ and $\langle w \rangle_{\mathbb{Z}}$ to denote the integer tuple $\overline{z}$.

Moreover, we write $\langle\!\langle \overline{a} \rangle\!\rangle_{\mathbb{N}}$ for the shortest word in $(\Sigma^r)^*$ that represents $\overline{a} \in \mathbb{N}^r$. Note that $\langle\!\langle \overline{a} \rangle\!\rangle_{\mathbb{N}}$ is well-defined since (1) there is a word $w \in (\Sigma^r)^*$ with $\langle w \rangle_{\mathbb{N}} = \overline{a}$, and (2) if $\langle v \rangle_{\mathbb{N}} = \langle v' \rangle_{\mathbb{N}}$ for $v, v' \in (\Sigma^r)^*$, then v and v' have a common suffix $u \in (\Sigma^r)^*$ with $\langle u \rangle_{\mathbb{N}} = \langle v \rangle_{\mathbb{N}}$. Similar to $\langle\!\langle \overline{a} \rangle\!\rangle_{\mathbb{N}}$ for $\overline{a} \in \mathbb{N}^r$, we define $\langle\!\langle \overline{z} \rangle\!\rangle_{\mathbb{Z}}$, for $\overline{z} \in \mathbb{Z}^r$, as the shortest word $w \in (\Sigma^r)^+$ with $\overline{z} = \langle w \rangle_{\mathbb{Z}}$ and the first letter of w is in $\{0, \varrho - 1\}^r$.

Definition 3.2. Let $U \subseteq \mathbb{Z}^r$. The language $L \subseteq (\Sigma^r)^*$ *represents* U if $L = \{w \in (\Sigma^r)^+ : \langle w \rangle_{\mathbb{Z}} \in U\}$. A DWA $\mathcal{A}$ *represents* U if $L(\mathcal{A})$ represents U.

Note that by this definition not every language over Σ^r represents a set of tuples of integers, and not every DWA with alphabet Σ^r represents a subset of $\mathbb{Z}^r$.

Example 3.3. The set of pairs $(x, y) \in \mathbb{Z}^2$ where y equals $2x$ is represented by the DWA depicted in Figure 1 by using the base $\varrho = 2$ for representing integers as words, i. e., the alphabet of the DWA is $\{0,1\}^2$. In the figure, we use abbreviations like $(0,-)$ to denote the letters $(0,0)$ and $(0,1)$.

3.2 Linear Equations and Inequations

In this subsection, we first recall the automata constructions given in [Boigelot et al. 1998; Boigelot 1999; Wolper and Boigelot 2000; Ganesh et al. 2002] for linear (in)equations. Then, we improve these constructions such that they are optimal, i. e., the constructed DWAs are minimal. Assume that the (in)equation $t \lessgtr c$ is given in normalized form, i. e., $t(x_1, \ldots, x_r)$ is a homogeneous term, $\lessgtr \in \{=, \neq, <, \leq, >, \geq\}$, and $c \in \mathbb{Z}$.

First, we make the following observation for a word $u \in (\Sigma^r)^*$ and $\overline{b} \in \Sigma^r$. If $u \neq \lambda$ then $\langle u\overline{b} \rangle_{\mathbb{Z}} = \varrho \langle u \rangle_{\mathbb{Z}} + \overline{b}$. For $u = \lambda$, we have that $\langle \overline{b} \rangle_{\mathbb{Z}} = \sigma(\overline{b})$. Given this, it is relatively straightforward to obtain an analog of a DWA with *infinitely* many states for $t \lessgtr c$. The set of states is $\{q_{\mathrm{I}}\} \cup \mathbb{Z}$, where q_{I} is the initial state. Note that we identify integers with states. The idea is to keep track of the value of t as successive bits are read. Thus, except for the special initial state, a state in $\mathbb{Z}$ represents the current value of t. Lemma 3.4 below justifies this intuition. The transition function $\eta : (\{q_{\mathrm{I}}\} \cup \mathbb{Z}) \times \Sigma^r \to (\{q_{\mathrm{I}}\} \cup \mathbb{Z})$ is defined as follows for a letter $\overline{b} \in \Sigma^r$. For the initial state, we define $\eta(q_{\mathrm{I}}, \overline{b}) := t[\sigma(\overline{b})]$. For $q \in \mathbb{Z}$, we define $\eta(q, \overline{b}) := \varrho q + t[\overline{b}]$.

LEMMA 3.4. *For $u \in (\Sigma^r)^*$ of length $n \geq 0$ we have that*

(a) $\widehat{\eta}(q, u) = \varrho^n q + t\big[\langle u\rangle_{\mathbb{N}}\big]$, *for* $q \in \mathbb{Z}$, *and*

(b) $\widehat{\eta}(q_{\mathrm{I}}, \bar{b}u) = t\big[\langle \bar{b}u\rangle_{\mathbb{Z}}\big]$, *for* $\bar{b} \in \Sigma^r$.

PROOF. (a) is easily proved by induction over n, and (b) follows from (a) and the definition of η. □

Later we make use of the following lemma, which translates the question whether $q \in \mathbb{Z}$ is reachable from $p \in \mathbb{Z}$ via $\widehat{\eta}$ to a number-theoretic problem.

LEMMA 3.5. *Let $p, q \in \mathbb{Z}$. There are $N, a_1, \dots, a_r \geq 0$ such that $N \geq \lceil \log_\varrho(1 + \max\{a_1, \dots, a_r\})\rceil$ and $\varrho^N p + t[a_1, \dots, a_r] = q$ iff there is a word $w \in (\Sigma^r)^*$ such that $\widehat{\eta}(p, w) = q$.*

PROOF. ($\Rightarrow$) Assume that $\langle\!\langle a_1, \dots, a_r\rangle\!\rangle_{\mathbb{N}}$ has length ℓ. Note that $\ell \leq N$. This follows from the fact that for every $a \in \mathbb{N}$, there is a word $u \in \Sigma^*$ of length $\lceil \log_\varrho(1 + a)\rceil$ such that $\langle u\rangle_{\mathbb{N}} = a$. By Lemma 3.4(a), we have that

$$\widehat{\eta}\big(p, \bar{0}^{N-\ell}\langle\!\langle a_1, \dots, a_r\rangle\!\rangle_{\mathbb{N}}\big) = \varrho^N p + t[a_1, \dots, a_r] = q\,.$$

($\Leftarrow$) Assume that $\widehat{\eta}(p, w) = q$, for some $w \in (\Sigma^r)^*$. Let N be the length of w. We have that $N \geq \lceil \log_\varrho(1 + a)\rceil$, where a is the largest number in the tuple $\langle w\rangle_{\mathbb{N}}$. It follows from Lemma 3.4(a) that $\widehat{\eta}(p, w) = \varrho^N p + t[\langle w\rangle_{\mathbb{N}}]$. □

The automata constructions in [Wolper and Boigelot 2000; Ganesh et al. 2002] are based on the observation that the states $q, q' \in \mathbb{Z}$ can be merged if, intuitively speaking, q and q' are both small or both large. Here, the meaning of "small" and "large" depends on the coefficients of t and on the constant c. More precisely, we say that $q \in \mathbb{Z}$ is *small* if $q < \min\{c, -\|t\|_+\}$, and *large* if $q > \max\{c, \|t\|_-\}$, where

$$\|t\|_- := \sum_{\substack{1\leq j\leq r\\ \text{and } k_j<0}} |k_j| \qquad \text{and} \qquad \|t\|_+ := \sum_{\substack{1\leq j\leq r\\ \text{and } k_j>0}} k_j$$

assuming that t is of the form $k_1{\cdot}x_1 + \dots + k_r{\cdot}x_r$. Note that from a small value we can only obtain smaller values and from a large value we can only obtain larger values by η, i. e., for all $\bar{b} \in \Sigma^r$, if $q > \|t\|_-$ then $\eta(q, \bar{b}) = \varrho q + t[\bar{b}] > q$, and if $q < -\|t\|_+$ then $\eta(q, \bar{b}) = \varrho q + t[\bar{b}] < q$. A difference between the constructions in [Wolper and Boigelot 2000] and [Ganesh et al. 2002] are the bounds that determine the meaning of "small" and "large".

For $m < n$, we define the DWA $\mathcal{A}^{t \lessgtr c}_{(m,n)} := (Q, \Sigma^r, \delta, q_{\mathrm{I}}, F)$, where $Q := \{q_{\mathrm{I}}\} \cup \{q \in \mathbb{Z} : m \leq q \leq n\}$, $F := \{q \in Q \cap \mathbb{Z} : q \lessgtr c\}$, and

$$\delta(q, \bar{b}) := \begin{cases} m & \text{if } \eta(q, \bar{b}) \leq m,\\ n & \text{if } \eta(q, \bar{b}) \geq n,\\ \eta(q, \bar{b}) & \text{otherwise,} \end{cases}$$

for $q \in Q$ and $\bar{b} \in \Sigma^r$.

LEMMA 3.6. *The DWA $\mathcal{A}^{t \lessgtr c}_{(m,n)}$ represents $[\![t \lessgtr c]\!]$ if m is small and n is large. Moreover, $\mathcal{A}^{t \lessgtr c}_{(m,n)}$ has $2 + n - m$ states.*

Proof. From the definition of the state set Q it immediately follows that $\mathcal{A}^{t \lessgtr c}_{(m,n)}$ has $2 + n - m$ states.

For $w \in (\Sigma^r)^*$, assume that $t[\langle w \rangle_{\mathbb{Z}}] \lessgtr c$. Note that due to our encoding of integers as words the length of w is greater than 0. By Lemma 3.4, we have that $\widehat{\eta}(q_{\mathrm{I}}, w) = t[\langle w \rangle_{\mathbb{Z}}]$. By induction over the length of w, it is straightforward to show that

$$\widehat{\delta}(q_{\mathrm{I}}, w) = \begin{cases} m & \text{if } \widehat{\eta}(q_{\mathrm{I}}, w) \leq m, \\ n & \text{if } \widehat{\eta}(q_{\mathrm{I}}, w) \geq n, \\ \widehat{\eta}(q_{\mathrm{I}}, w) & \text{otherwise.} \end{cases}$$

Note that m is small and n is large by assumption. By the definition of the set F of accepting states, we have that $w \in L(\mathcal{A}^{t \lessgtr c}_{(m,n)})$. Using similar arguments, we can prove that $t[\langle w \rangle_{\mathbb{Z}}] \lessgtr c$, for every $w \in L(\mathcal{A}^{t \lessgtr c}_{(m,n)})$. We omit it. □

In the following, we optimize the constructions such that the produced DWA for an (in)equation is minimal. Moreover, we give lower bounds on the minimal DWAs for (in)equations. However, these results are not needed for the upper bound on the minimal DWA for a PA formula, which we establish in §4.

In the remainder of this subsection, let $\mathcal{A}^{t \lessgtr c}_{(m,n)} = (Q, \Sigma^r, \delta, q_{\mathrm{I}}, F)$ be the DWA for the (in)equation $t \lessgtr c$ with $m = \max\{q \in \mathbb{Z} : q \text{ is small}\}$ and $n = \min\{q \in \mathbb{Z} : q \text{ is large}\}$. We restrict ourselves to the cases where $\lessgtr \in \{=, <, >\}$. The cases with $\lessgtr \in \{\neq, \leq, \geq\}$ reduce to the cases for $=$, $<$, $>$ and complementation of DWAs, since $t \neq c$ is logically equivalent to $\neg t = c$, $t \leq c$ is logically equivalent to $\neg t > c$, and $t \geq c$ is logically equivalent to $\neg t < c$. Note that complementation of a DWA can be done by flipping accepting and non-accepting states. After complementation we have to make the initial state of the DWA non-accepting since the empty word does not represent any integer tuple. The resulting DWA is minimal iff the original DWA is minimal.

3.2.1 *Eliminating Unreachable States.* An obvious optimization is to eliminate the states in $\mathcal{A}^{t \lessgtr c}_{(m,n)}$ that are not reachable from q_{I}. These states are characterized as follows. We define the *greatest common divisor* of the term $t(x_1, \ldots, x_r)$ as $\gcd(t) := \gcd(|k_1|, \ldots, |k_r|)$, where k_i is the coefficient of the variable x_i, for $1 \leq i \leq r$.

Lemma 3.7. *A state $q \in \{m < i < n : i \in \mathbb{Z}\}$ is reachable from the initial state q_{I} iff q is a multiple of* $\gcd(t)$.

Proof. ($\Rightarrow$) This direction is easy to prove by induction on the length of $w \in (\Sigma^r)^*$ with $\widehat{\delta}(q_{\mathrm{I}}, w) \in \mathbb{Z}$: for all $\bar{b} \in \Sigma^r$, it holds that (i) $\delta(q_{\mathrm{I}}, \bar{b}) = t[\sigma(\bar{b})]$ is a multiple of $\gcd(t)$, and (ii) if $\widehat{\delta}(q_{\mathrm{I}}, w) \in \{m < i < n : i \in \mathbb{Z}\}$ is a multiple of $\gcd(t)$ then $\varrho\widehat{\delta}(q_{\mathrm{I}}, w) + t[\bar{b}]$ is a multiple of $\gcd(t)$.

($\Leftarrow$) Assume that q is a multiple of $\gcd(t)$. There are $v_1, \ldots, v_r \in \mathbb{Z}$ such that $t[v_1, \ldots, v_r] = q$. With Lemma 3.4(b) we conclude that $\widehat{\delta}\big(q_{\mathrm{I}}, \langle\!\langle v_1, \ldots, v_r \rangle\!\rangle_{\mathbb{Z}}\big) = t[v_1, \ldots, v_r]$. □

Trivially, q_{I} is reachable from q_{I}. Analogously, as in the direction from left to right in the above proof of Lemma 3.7, we obtain that the state m is reachable from q_{I}.

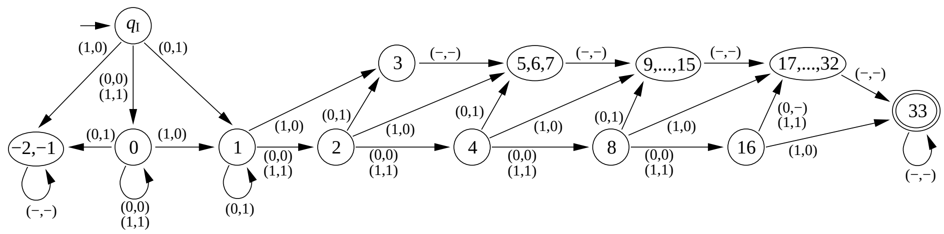

Fig. 2. Minimal DWA over the alphabet $\{0,1\}^2$ for the inequation $x - y > 32$.

Note that there is an $m' \leq m$ that is a multiple of $\gcd(t)$. Similar, we have that n is reachable from q_I. Thus, by Lemma 3.7, the states that are not reachable from q_I are precisely the states in $\{m < i < n : i \in \mathbb{Z}\}$ that are not a multiple of the greatest common divisor of the absolute values of the coefficients occurring in the term t.

Alternatively, instead of filtering out the states $q \in \mathbb{Z}$ that are not a multiple of $\gcd(t)$ we can rewrite the (in)equation $t \gtreqless c$ into the logically equivalent atomic formula α and then construct the DWA for α, where α is defined as

$$\alpha := \begin{cases} t' \gtreqless \left\lceil \frac{c}{\gcd(t)} \right\rceil & \text{if } \gtreqless \text{ is } <, \\ t' \gtreqless \left\lfloor \frac{c}{\gcd(t)} \right\rfloor & \text{if } \gtreqless \text{ is } >, \\ t' \gtreqless \frac{c}{\gcd(t)} & \text{if } \gtreqless \text{ is } = \text{ and } c \text{ is a multiple of } \gcd(t), \\ 1 < 0 & \text{otherwise,} \end{cases}$$

where the coefficients in t' are the coefficients of t divided by $\gcd(t)$. In the remainder of this subsection, we assume that $\gcd(t) = 1$.

3.2.2 *Optimal Construction for Inequations.* In the following, we assume that the inequation is of the form $t > c$ with $c \geq 0$. The cases where $\gtreqless$ is $<$ or $c \geq 0$ are analogous. The following example illustrates that many states of $\mathcal{A}^{t>c}_{(m,n)}$ can be merged if c is significantly larger than $\|t\|_-$.

Example 3.8. The automata construction described above for the inequation $x - y > 32$ produces a DWA with the set of states $Q = \{q_I, -2, -1, 0, \dots, 32, 33\}$; but the minimal DWA (see Figure 2) for $x - y > 32$ has only 13 states when we choose the base $\varrho = 2$.

The reason for this gap is that several states can be merged. First, we merge the states -2 and -1 since from both states only non-accepting states are reachable. Second, we can merge the states in $Q' := \{q \in Q \cap \mathbb{Z} : 2q + a - b > c, \text{ for all } a, b \in \{0,1\}\} = \{17, \dots, 32\}$ to a single state since all states in Q' are non-accepting and all their transitions go to state 33. The state 16 cannot be merged with any other state since if we read the letter $(1,0)$, we end up in the accepting state 33, and if we read the letters $(0,0)$, $(1,1)$, or $(0,1)$ we end up in the non-accepting states 32 or 31. The states in $\{9, \dots, 15\}$ can again be merged to a single state since with every transition we reach a state in Q'. Analogously, we can merge the states in $\{5, 6, 7\}$.

In the following, we determine the equivalent states in $\mathcal{A}^{t>c}_{(m,n)}$. Note that from Lemma 3.7 it follows that all states are reachable from q_{I} since we assume that $\gcd(t) = 1$. We use the notation $[d, d')$ for the set $\{d, \dots, d'-1\}$ if $d, d' \in \mathbb{Z}$, and if $d \in \mathbb{Z}$ and $d' = \infty$ then $[d, d') := \{z \in \mathbb{Z} : z \geq d\}$. In order to identify the equivalent states, we define the following strictly monotonically decreasing sequence $d_0 > d_1 > \cdots > d_\ell$, for some $\ell \geq 1$. Let $d_0 := \infty$ and $d_1 := \max\{c+1, \|t\|_-\}$. Assume that $d_0 > d_1 > \cdots > d_i$ are already defined, for some $i \geq 1$.

—If $d_i = \|t\|_-$ then we are done, i. e., $\ell = i$.

—If $d_i > \|t\|_-$ then we define d_{i+1} as the smallest element in the set S that consists of the integers $z \geq \|t\|_-$ such that for all $\bar{b} \in \Sigma^r$, there is an index $j \in \{1, \dots, i\}$ such that $\varrho z + t[\bar{b}]$ and $\varrho(d_i - 1) + t[\bar{b}]$ are in $[d_j, d_{j-1})$. Note that the smallest element in S always exists since $d_i - 1 \in S$ and all elements in S are greater than or equal to $\|t\|_-$.

The following lemma characterizes the equivalent states in the DWA $\mathcal{A}^{t>c}_{(m,n)}$. In particular, it shows that we can merge the states in $R := \{-\|t\|_+ - 1, -\|t\|_+\}$, and for each $1 \leq i \leq \ell$, the states in $[d_i, d_{i-1})$ can be collapsed into one state.

LEMMA 3.9. *For all $p, q \in Q$, it holds that $p \sim q$ iff $p = q$ or $p, q \in R$ or $p, q \in [d_i, d_{i-1})$, for $1 \leq i \leq \ell$.*

PROOF. ($\Leftarrow$) If $p = q$ then it is obvious that $p \sim q$. If $p, q \in R$ then we also have that $p \sim q$, since both states are non-accepting and all transitions from these states either go to $-\|t\|_+$ or to $-\|t\|_+ - 1$.

It remains to prove that for $1 \leq i \leq \ell$, if $p, q \in [d_i, d_{i-1})$ then $p \sim q$. We prove this claim by induction over i.

For the base case $i = 1$, we make a case distinction. If $c \geq \|t\|_-$, there is nothing to prove since $[d_1, d_0) \cap Q$ is a singleton. If $c < \|t\|_-$, we have that $[d_1, d_0) \cap Q = \{\|t\|_-, \|t\|_- + 1\}$. The states $\|t\|_-$ and $\|t\|_- + 1$ can be merged since both states are accepting and all transitions from these states either go to $\|t\|_-$ or to $\|t\|_- + 1$.

For the step case, assume that $i > 1$ and let $p, q \in [d_i, d_{i-1})$. Without loss of generality we assume that $p \leq q$. By the definition of the transition function δ and the sequence $d_0 > d_1 > \cdots > d_\ell$, we have that

$$\varrho d_i + t[\bar{b}] \leq \delta(p, \bar{b}) \leq \delta(q, \bar{b}) \leq \varrho(d_{i-1} - 1) + t[\bar{b}]\,,$$

for all $\bar{b} \in \Sigma^r$. Since there is a $1 \leq j < i$ with $\varrho d_i + t[\bar{b}], \varrho d_{i-1} + t[\bar{b}] \in [d_j, d_{j-1})$ we conclude that $\delta(p, \bar{b}), \delta(q, \bar{b}) \in [d_j, d_{j-1})$. The claim now follows from the induction hypothesis.

($\Rightarrow$) We prove the claim by contraposition, i. e., $p \not\sim q$ is implied by the three conditions (i) $p \neq q$, (ii) $p \in R \Rightarrow q \notin R$, and (iii) for all $1 \leq i \leq \ell$, $p \in [d_i, d_{i-1}) \Rightarrow q \notin [d_i, d_{i-1})$. Assume $p \neq q$. It suffices to distinguish the following three cases.

Case 1: $p \in R$ and $q \notin R$. Since we can reach an accepting state from q, we have that $p \not\sim q$.

Case 2: $p \in [d_i, d_{i-1})$ and $q \notin [d_i, d_{i-1})$, for some $1 \leq i \leq \ell$. It is straightforward to prove by induction over i that $p \not\sim q$.

Case 3: $p \notin R \cup \bigcup_{1 \leq i \leq \ell}[d_i, d_{i-1})$. Note that the conditions (ii) and (iii) are satisfied. We have that either $p = q_{\rm I}$ or $p \in S$, where $S := \{s \in Q \cap \mathbb{Z} : -\|t\|_+ < s < \|t\|_-\}$.

If $p = q_{\rm I}$ and $q \in R$ then we conclude similar to Case 1 that $p \not\sim q$. Assume that $p = q_{\rm I}$ and $q \notin R$. Let $\bar{b} \in \Sigma^r$ be the letter that has a 0 in its ith coordinate iff the ith coefficient of t is negative, and otherwise the ith coordinate is $\varrho - 1$. It holds that $q_{\rm I} \not\sim q$, since $\delta(q_{\rm I}, \bar{b}) = -t[\bar{b}] \in R$ and $\delta(q, \bar{b}) = \varrho q + (\varrho - 1)\|t\|_+ \geq q$. From Case 1, it follows that $p \not\sim q$.

Assume that $p \in S$. Note that for every $s \in S$ there is a $\bar{b} \in \Sigma^r$ such that $\delta(s, \bar{b}) \in S$. It follows that for every $n \geq 0$ there is a word $u \in (\Sigma^r)^*$ of length n such that $\widehat{\delta}(p, u) \in S$. We conclude that there is a word $u \in (\Sigma^r)^*$ such that $\widehat{\delta}(p, u) \in S$ and $\widehat{\delta}(q, u) \in R \cup \bigcup_{1 \leq i \leq \ell}[d_i, d_{i-1})$, since $\delta(s, \bar{b}) - \delta(s', \bar{b}) = \varrho(s - s')$, for all $s, s' \in S$ and all $\bar{b} \in \Sigma^r$. Analogously to the Cases 1 and 2 we conclude that $p \not\sim q$. □

From Lemma 3.9, it follows that the minimal DWA representing $[\![t > c]\!]$ has at least $\|t\|_- + \|t\|_+$ states. Note that this is in contrast to the number of symbols we need to write the inequation $t > c$ if coefficients are represented as binary numbers. For instance, we need $22 + 7$ letters for $1025 \cdot x - 1024 \cdot y > 0$, since each of the two coefficients can be represented with 11 digits. The same lower bound on the minimal DWA size holds for $t < c$. In the following, we show that a similar lower bound holds for DWAs representing $[\![t = c]\!]$.

3.2.3 *Optimal Construction for Equations.* For an equation $t{=}c$, we can collapse the states in $\mathcal{A}^{t=c}_{(m,n)}$ from which we cannot reach the accepting state $c \in Q$ to a single non-accepting sink state. These optimizations produce the minimal DWA for $t{=}c$. For instance, the case for $p \in Q \cap \mathbb{Z}$ is proved as follows. Assume that we can reach the state c from $p \in Q \cap \mathbb{Z}$, i.e., there is a $u \in (\Sigma^r)^*$, with $\widehat{\delta}(p, u) = c$. Any other states $q \in Q \cap \mathbb{Z}$ with $q \neq p$ from which we can reach c cannot be merged with p, since

$$c = \widehat{\delta}(p, u) \overset{\text{Lemma 3.4(a)}}{=} \varrho^{|u|}p + t\big[\langle u \rangle_{\mathbb{N}}\big] \neq \varrho^{|u|}q + t\big[\langle u \rangle_{\mathbb{N}}\big] \overset{\text{Lemma 3.4(a)}}{=} \widehat{\delta}(q, u)\,.$$

The other cases are proved similarly.

A lower bound for the minimal DWA representing $[\![t{=}c]\!]$ is based on the following lemma about the states of the DWA $\mathcal{A}^{t \lessgtr c}_{(m,n)} = (Q, \Sigma^r, \delta, q_{\rm I}, F)$, where $\lessgtr \in \{=, \neq, <, \leq, >, \geq\}$. Let $S := \{s \in Q \cap \mathbb{Z} : -\|t\|_+ < s < \|t\|_-\}$ and $[n] := \{0, \ldots, n-1\}$, for $n \geq 0$.

LEMMA 3.10. *Every $q \in Q \cap \mathbb{Z}$ is reachable from every $p \in S$.*

PROOF. We need a result from number theory. Let $\gamma > 0$ and let $c_1, \ldots, c_\gamma$ be integers with $0 < c_1 < \cdots < c_\gamma$ and $\gcd(c_1, \ldots, c_\gamma) = 1$. The *Frobenius number* $G(c_1, \ldots, c_\gamma)$ is the greatest integer z for which the linear equation $c_1 \cdot x_1 + \cdots + c_\gamma \cdot x_\gamma = z$ has *no* solution in the natural numbers. For $\gamma = 1$, it trivially holds that $G(c_1) = -1$. For $\gamma > 1$, the upper bound $G(c_1, \ldots, c_\gamma) \leq \frac{c_\gamma^2}{\gamma - 1}$ was proved by Dixmier [1990]. We will make use of the following bound on the Frobenius numbers: for all $\gamma > 0$, we have that

$$G(c_1, \ldots, c_\gamma) < \varrho^{c_1+\cdots+c_\gamma} - (c_1 + \cdots + c_\gamma)\,. \tag{1}$$

The inequality (1) is proved as follows, where we assume without loss of generality that $\varrho = 2$. For $c_\gamma \leq 3$, there are only finitely many cases, which are easy to check. Assume that $c_\gamma > 3$. Note that $\gamma > 1$. We have that

$$2^{c_1+\cdots+c_\gamma} - (c_1 + \cdots + c_\gamma) > 2^{c_\gamma} 2^{c_1+\cdots+c_{\gamma-1}} - \gamma \cdot c_\gamma \geq 2^{c_\gamma} 2^\gamma - c_\gamma^2\,.$$

By the result from Dixmier [1990], it suffices to check that $2^{c_\gamma} 2^\gamma - c_\gamma^2 \geq \frac{c_\gamma^2}{\gamma-1}$. This inequality is equivalent to $2^{\gamma-1} \geq \frac{\gamma \cdot c_\gamma^2}{2^{c_\gamma}(\gamma-1)}$. The inequality can be further simplified to $2^{\gamma-1} \geq \frac{\gamma}{\gamma-1}$, since $\frac{c_\gamma^2}{2^{c_\gamma}} \leq 1$, for all $c_\gamma > 3$. We are done, since the inequality $2^{\gamma-1} \geq \frac{\gamma}{\gamma-1}$ holds, for all $\gamma > 1$.

In the following, we will prove the lemma, i. e., for all $p \in S$ and $q \in Q \cap \mathbb{Z}$, there is a word $u \in (\Sigma^r)^*$ such that $\widehat{\delta}(p, u) = q$. Note that if $r = 0$ and $r = 1$ then $S = \emptyset$ and the claim is trivially true. Assume that $r \geq 2$. By Lemma 3.5, it suffices to show that the equation

$$\varrho^N p + t(x_1, \ldots, x_r) = q \tag{2}$$

has a solution $a_1, \ldots, a_r \geq 0$ with $N \geq \lceil \log_\varrho(1+\max\{a_1, \ldots, a_r\})\rceil$. We distinguish four cases depending on p and q.

Case 1: $p = 0$. Equation (2) simplifies to

$$t(x_1, \ldots, x_r) = q\,. \tag{3}$$

There are positive and negative coefficients in t, since $p \in S$. It follows that equation (3) has infinitely many solutions in the natural numbers. Recall that we assume that $\gcd(t) = 1$. We are done, since equation (2) is satisfied for any of these solutions $a_1, \ldots, a_r \geq 0$ and any $N \geq \lceil \log_\varrho(1 + \max\{a_1, \ldots, a_r\})\rceil$.

Case 2: $p > 0$ and $q \geq 0$. Let $k_{i_1}, \ldots, k_{i_\mu}$ be the positive coefficients in t, and let $k_{j_1}, \ldots, k_{j_\nu}$ be the negative coefficients in t. Let N be the size of the DWA $\mathcal{A}^{t \lessgtr c}_{(m,n)}$, i. e., $N = 3 + \max\{|c|, \|t\|_+\} + \max\{c, \|t\|_-\}$. We rewrite equation (2) into

$$\varrho^N p - q + t_1(x_{i_1}, \ldots, x_{i_\mu}) = t_2(x_{j_1}, \ldots, x_{j_\nu})\,, \tag{4}$$

where t_1 is the term $k_{i_1} \cdot x_{i_1} + \cdots + k_{i_\mu} \cdot x_{i_\mu}$ and t_2 is the term $|k_{j_1}| \cdot x_{j_1} + \cdots + |k_{j_\nu}| \cdot x_{j_\nu}$. Note that $\varrho^N p - q \geq 0$ since $p > 0$ and $\varrho^N \geq q$. Let $D := \gcd(|k_{j_1}|, \ldots, |k_{j_\nu}|)$. In order to show the existence of a solution $a_1, \ldots, a_r \in [\varrho^N]$ of equation (4), we proceed in two steps:

Step 1: There are $a_{i_1}, \ldots, a_{i_\mu} \in [D]$ such that

$$D \;\Big|\; \varrho^N p - q + t_1[a_{i_1}, \ldots, a_{i_\mu}]\,.$$

Step 2: There are $a_{j_1}, \ldots, a_{j_\nu} \in [\varrho^N]$ such that

$$\varrho^N p - q + t_1[a_{i_1}, \ldots, a_{i_\mu}] = t_2[a_{j_1}, \ldots, a_{j_\nu}]\,.$$

Proof of Step 1: If $\mu = 0$ then there is nothing to prove. Assume that $\mu > 0$. There are $K, R \geq 0$ such that $\varrho^N p - q = DK + R$ with $R < D$. It suffices to show that there are $a_{i_1}, \dots, a_{i_\mu}$ with $0 \leq a_{i_1}, \dots, a_{i_\mu} < D$, and $K' \geq 0$, such that $DK' = R + t_1[a_{i_1}, \dots, a_{i_\mu}]$, since then

$$\begin{aligned}\varrho^N p - q + t_1[a_{i_1}, \dots, a_{i_\mu}] &= DK + R + t_1[a_{i_1}, \dots, a_{i_\mu}] = DK + DK' \\ &= D(K + K'),\end{aligned}$$

and thus, $D | \varrho^N p - q + t_1[a_{i_1}, \dots, a_{i_\mu}]$.

First, assume the existence of $a_{i_1}, \dots, a_{i_\mu} \geq 0$ with $D | R + t_1[a_{i_1}, \dots, a_{i_\mu}]$, where $a_{i_\xi} \geq D$, for some $1 \leq \xi \leq \mu$. To simplify matters, we assume without loss of generality that $\xi = 1$. There is an $a \geq 0$ with $a_{i_1} = D + a$. Further, assume that there is no $b < a_{i_1}$ with $D | R + t_1[b, a_{i_2}, \dots, a_{i_\mu}]$. For some $K' \geq 0$, we have that

$$DK' = R + t_1[a_{i_1}, \dots, a_{i_\mu}] = R + Dk_{i_1} + t_1[a, a_{i_2}, \dots, a_{i_\mu}].$$

Therefore, $D(K' - k_{i_1}) = R + t_1[a, a_{i_2}, \dots, a_{i_\mu}]$, i.e., $D | R + t_1[a, a_{i_2}, \dots, a_{i_\mu}]$. This contradicts the minimality of $D + a$.

It remains to show the existence of $a_{i_1}, \dots, a_{i_\mu} \geq 0$ with $D | R + t_1[a_{i_1}, \dots, a_{i_\mu}]$. The existence reduces to the problem of whether the equation

$$D \cdot y - k_{i_1} \cdot x_{i_1} - \dots - k_{i_\mu} \cdot x_{i_\mu} = R$$

has a solution in the natural numbers. This is the case since $\gcd(D, k_{i_1}, \dots, k_{i_\mu}) = 1$, by assumption.

Proof of Step 2: Assume that there are $\gamma \geq 1$ distinct coefficients in t_2 of equation (4). Without loss of generality, assume that $0 < |k_{j_1}| < \dots < |k_{j_\gamma}|$. Let $W := \frac{\varrho^N p - q + t_1[a_{i_1}, \dots, a_{i_\mu}]}{D}$ and $\ell_\xi := \frac{|k_{j_\xi}|}{D}$, for $1 \leq \xi \leq \nu$. Note that $\ell_1 < \dots < \ell_\gamma$ and that $\gcd(\ell_1, \dots, \ell_\gamma) = 1$. Equation (4) simplifies with the a_is from Step 1 to

$$W = \ell_1 \cdot x_{j_1} + \dots + \ell_\nu \cdot x_{j_\nu}. \tag{5}$$

An upper bound on W is

$$W \leq \tfrac{\varrho^N p - q + D\|t\|_+}{D} \leq \tfrac{\varrho^N (\|t\|_- - 1) + D\|t\|_+}{D} = \tfrac{\varrho^N \|t\|_-}{D} - \tfrac{\varrho^N}{D} + \|t\|_+ \tag{6}$$

and a lower bound on W is

$$\begin{aligned}W &\geq \tfrac{\varrho^N - q}{D} \geq \tfrac{\varrho^N - \max\{c, \|t\|_-\}}{D} \geq \tfrac{\varrho^{D(\ell_1 + \dots + \ell_\nu)} - D(\ell_1 + \dots + \ell_\nu)}{D} \\ &\geq \varrho^{\ell_1 + \dots + \ell_\gamma} - (\ell_1 + \dots + \ell_\gamma).\end{aligned}$$

From the lower bound on W and the upper bound on Frobenius numbers (1), it follows that equation (5) has a solution in the natural numbers. Let $\kappa \geq 0$ be maximal such that there are $a_1, \dots, a_\gamma \geq 0$ with

$$W = \ell_1 a_1 + \dots + \ell_\gamma a_\gamma + \kappa L, \tag{7}$$

where $L := \frac{\|t\|_-}{D}$. By contradiction, we obtain that $a_1, \dots, a_\gamma < L$: Assume that there is a ξ, $1 \leq \xi \leq \gamma$ with $a_\xi = L + a$, for some $a \geq 0$. Without loss of generality, assume that $\xi = 1$. This contradicts the assumption that κ is maximal:

$$\begin{aligned}W &= \kappa L + \ell_1 (L + a) + \ell_2 a_2 + \dots + \ell_\gamma a_\gamma \\ &= (\kappa + \ell_1) L + \ell_1 a + \ell_2 a_2 + \dots + \ell_\gamma a_\gamma.\end{aligned}$$

From κ and $a_1, \dots, a_\gamma$, we obtain a solution for equation (5) in the natural numbers, namely

$$\begin{aligned} W &= \kappa L + \ell_1 a_1 + \cdots + \ell_\gamma a_\gamma \\ &= \kappa(\ell_1 + \cdots + \ell_\nu) + \ell_1 a_1 + \cdots + \ell_\gamma a_\gamma \\ &= \ell_1(\kappa + a_1) + \cdots + \ell_\gamma(\kappa + a_\gamma) + \ell_{\gamma+1}\kappa + \cdots + \ell_\nu \kappa\,. \end{aligned}$$

It suffices to show that $\kappa < \varrho^N - \max\{a_1, \dots, a_\gamma\}$. An upper bound on κ is

$$\begin{aligned} \kappa &\overset{(7)}{=} \tfrac{W-(\ell_1 a_1+\cdots+\ell_\gamma a_\gamma)}{L} \\ &\le \tfrac{W}{L} - \tfrac{\max\{a_1,\dots,a_\gamma\}}{L} \\ &\overset{(6)}{\le} \tfrac{\varrho^N \|t\|_-}{DL} - \tfrac{\varrho^N}{DL} + \tfrac{\|t\|_+}{L} - \tfrac{\max\{a_1,\dots,a_\gamma\}}{L} \\ &\le \varrho^N - \tfrac{\varrho^N}{DL} + \tfrac{\|t\|_+ - \max\{a_1,\dots,a_\gamma\}}{L}\,. \end{aligned}$$

It remains to check whether the inequality

$$\varrho^N - \tfrac{\varrho^N}{DL} + \tfrac{\|t\|_+ - \max\{a_1,\dots,a_\gamma\}}{L} < \varrho^N - \max\{a_1, \dots, a_\gamma\}$$

is valid. The previous inequality simplifies to

$$\tfrac{\|t\|_+ + \max\{a_1,\dots,a_\gamma\}(L-1)}{L} < \tfrac{\varrho^N}{DL}\,.$$

Multiplying with the common denominator DL, the inequality simplifies further to

$$D\|t\|_+ + D\max\{a_1, \dots, a_\gamma\}(L-1) < \varrho^N\,.$$

Since $\max\{a_1, \dots, a_\gamma\} \le L - 1$ and $N \ge \|t\|_- + \|t\|_+ = DL + \|t\|_+$, it suffices to show the validity of the inequality

$$D\|t\|_+ + D(L-1)^2 < \varrho^{DL+\|t\|_+}\,. \tag{8}$$

It is straightforward to show that the inequality (8) is true for all $D, L \ge 1$ and $\|t\|_+ \ge 0$.

Case 3: $p < 0$ and $q \le 0$. It suffices to prove that there is a solution $a_1, \dots, a_r \in [\varrho^N]$ for the equation

$$t_1(x_{i_1}, \dots, x_{i_\mu}) = \varrho^N|p| - |q| + t_2(x_{j_1}, \dots, x_{j_\nu})\,,$$

where t_1 and t_2 are defined as in Case 2. This equation is similar to equation (4) except t_1 and t_2 are swapped. We can use a similar argumentation as in Case 2 for showing the existence of $a_1, \dots, a_r \in [\varrho^N]$.

Case 4: $p > 0$ and $q < 0$. This case can be solved with Case 1 and Case 2. Since $p > 0$ and $q < 0$, we have that $0 \in S$. By Case 2, the state 0 is reachable from p, and by Case 1, q is reachable from state 0.

Case 5: $p < 0$ and $q > 0$. Analogously, this case can be solved by Case 3 and Case 1. □

With Lemma 3.10 at hand, it is straightforward to prove for $\mathcal{A}^{t \lessgtr c}_{(m,n)}$ that $p \sim q$ iff $p = q$, for all $p, q \in S$. Therefore, we have that the minimal automaton representing $[\![t = c]\!]$ has at least $|S|$ states.

Another consequence of Lemma 3.10 is that S is a strongly connected component in $\mathcal{A}^{t \lessgtr c}_{(m,n)}$: By Lemma 3.10, every state $q \in S$ is reachable from every $p \in S$, and it is easy to show that the initial state q_{I} is not reachable from a state in S and that a state in S cannot be reached from any state that is not in $S \cup \{q_{\mathrm{I}}\}$.

3.2.4 *Implementation Issues.* We conclude this subsection by discussing implementation issues of the above automata constructions for (in)equations.

Wolper and Boigelot [2000] (see also [Boigelot and Wolper 2002]) propose an algorithm that constructs an automaton for an (in)equation $t \lessgtr c$ backward, i.e., the construction starts from the accepting states and iteratively computes the predecessor states of a state until no new states are generated. Additionally, we have to introduce a rejecting sink state, for making the transition function total. As pointed out by Wolper and Boigelot [2000], the advantage of the backward construction is that we only generate states from which we can reach an accepting state (except the sink state). Recall that we assume without loss of generality that $\gcd(t) = 1$ and that this assumption guarantees that we can reach the accepting state from the initial state (see §3.2.1). Furthermore, by Lemma 3.6 we can assume that we start the construction from the accepting states in $\{m, \dots, n\}$, where $m = \max\{q \in \mathbb{Z} : q \text{ is small}\}$ and $n = \min\{q \in \mathbb{Z} : q \text{ is large}\}$, and the predecessors of a state are determined by the transition function of the DWA $\mathcal{A}^{t \lessgtr c}_{(m,n)}$. We obtain a DWA with at most $2 + n - m$ states.

In the case of an equation $t = c$, it follows from the results in §3.2.3 that the backward construction in [Wolper and Boigelot 2000] yields the minimal DWA with at least $\|t\|_+ + \|t\|_-$ states. In the case of an inequation, like $t > c$, we also obtain by the backward construction a DWA with at least $\|t\|_+ + \|t\|_-$ states. However, in general, the DWA is not minimal. Recall that we assume that c is not negative. Lemma 3.9 in §3.2.2 characterizes the states that are equivalent. First, observe that we cannot reach an accepting state in the DWA $\mathcal{A}^{t>c}_{(m,n)}$ from the states $-\|t\|_+ - 1$ and $-\|t\|_+$. Thus, the backward construction does not generate them. Second, only some of the generated states in $\{\|t\|_-, \dots, 1 + \max\{c, \|t\|_-\}\}$ are equivalent. In particular, if $c < \|t\|_-$, we obtain by the backward construction only the equivalent states $\|t\|_-$ and $1 + \|t\|_-$. If $c > \|t\|_-$ there is a single accepting state, namely $1 + c$ from which we start the backward construction. If we generate the states in descending order, the backward construction can be extended such that it merges equivalent states: if all the outgoing transitions of the current state $p \in \{\|t\|_-, \dots, 1 + \max\{c, \|t\|_-\}\}$ lead to the same successor states as a state q, which was generated earlier, then p and q are equivalent. With this optimization of the backward construction, we obtain the minimal DWA for the inequation $t > c$. Alternatively, when $c > \|t\|_-$ and instead of complicating the backward construction, note that we can obtain the minimal DWA for $t > c$ by applying a standard minimization algorithm for DWAs, like the one by Hopcroft [1971].

3.3 Divisibility Relation

In this subsection, we give an upper bound of the size of the minimal DWA for a formula $d | t + c$, where $d \geq 2$, $t(x_1, \dots, x_r)$ is a homogeneous term, and $c \in \mathbb{Z}$.

Let $\mathcal{A}^{d|t+c}$ be the DWA with the set of states $Q := \{q_{\mathrm{I}}, 0, 1, \dots, d-1\}$. A state $q \in Q \cap \mathbb{Z}$ has an intuitive interpretation: if we reach the state q with a word

$w \in (\Sigma^r)^*$ then the remainder of the division of $t[\langle w\rangle_{\mathbb{Z}}]$ by d equals q. We denote by $\mathrm{rem}(q,d)$ the remainder of $q \in \mathbb{Z}$ divided by d. Let $\mathcal{A}^{d|t+c} := (Q, \Sigma^r, \delta, q_{\mathrm{I}}, F)$ be the DWA, where

$$\delta(q,\bar{b}) := \begin{cases} \mathrm{rem}\big(t[\sigma(\bar{b})], d\big) & \text{if } q = q_{\mathrm{I}},\\ \mathrm{rem}\big(\varrho q + t[\bar{b}], d\big) & \text{otherwise,}\end{cases}$$

for $q \in Q$ and $\bar{b} \in \Sigma^r$, and $F := \{q \in Q \cap \mathbb{Z} : d|q+c\}$. Note that there is exactly one $q \in Q \cap \mathbb{Z}$ with $d|q+c$.

The correctness of our construction follows from two facts:

(a) For $n \in \mathbb{Z}$, $d|n+c$ iff $d|\,\mathrm{rem}(n,d)+c$.

(b) For $w \in (\Sigma^r)^+$, $\widehat{\delta}(q_{\mathrm{I}}, w) = \mathrm{rem}\big(t[\langle w\rangle_{\mathbb{Z}}], d\big)$.

The proof of (a) is straightforward. There are $p, q \in \mathbb{Z}$ such that $pd + q = n$ and $0 \le q < d$. Note that $q = \mathrm{rem}(n,d)$. By definition, $d|n+c$ iff there is a $k \in \mathbb{Z}$ with $dk = n + c = pd + q + c$. The equality can be rewritten into $d(k-p) = q + c$, i. e., $d|\,\mathrm{rem}(n,d)+c$.

We prove (b) by induction over the length of w. For the base case, let $w = \bar{b} \in \Sigma^r$. Since we represent integers using ϱ's complement, we have that $t[\langle \bar{b}\rangle_{\mathbb{Z}}] = t[\sigma(\bar{b})]$. By definition, $\widehat{\delta}(q_{\mathrm{I}}, \bar{b}) = \mathrm{rem}\big(t[\langle \bar{b}\rangle_{\mathbb{Z}}], d\big)$. For the step case, assume $\widehat{\delta}(q_{\mathrm{I}}, w) = \mathrm{rem}\big(t[\langle w\rangle_{\mathbb{Z}}], d\big)$ and let $\bar{b} \in \Sigma^r$. There are $p, q \in \mathbb{Z}$ with $t[\langle w\rangle_{\mathbb{Z}}] = pd + q$ and $0 \le q < d$. Note that $q = \mathrm{rem}\big(t[\langle w\rangle_{\mathbb{Z}}], d\big)$ and $t[\langle w\bar{b}\rangle_{\mathbb{Z}}] = \varrho t[\langle w\rangle_{\mathbb{Z}}] + t[\bar{b}] = \varrho pd + \varrho q + t[\bar{b}]$. We have that

$$\begin{aligned}\mathrm{rem}\big(t[\langle w\bar{b}\rangle_{\mathbb{Z}}], d\big) &= \mathrm{rem}(\varrho pd + \varrho q + t[\bar{b}], d)\\ &= \mathrm{rem}(\varrho q + t[\bar{b}], d) = \delta(q, \bar{b})\\ &\overset{\mathrm{IH}}{=} \delta(\widehat{\delta}(q_{\mathrm{I}}, w), \bar{b}) = \widehat{\delta}(q_{\mathrm{I}}, w\bar{b})\,.\end{aligned}$$

LEMMA 3.11. *The DWA* $\mathcal{A}^{d|t+c}$ *represents* $[\![d|t+c]\!]$ *and has* $d+1$ *states.*

An optimization of the construction is to filter out the states that are not a multiple of $\gcd(\gcd(t), d)$. These states are not reachable from the initial state since $\mathrm{rem}(t[\bar{a}], d)$ is a multiple of $\gcd(\gcd(t), d)$, for every $\bar{a} \in \mathbb{Z}^r$.

3.4 Quantifier-free Formulas

In this subsection, we give an upper bound on the size of the minimal DWA for a quantifier-free PA formula. This upper bound depends on the maximal absolute value of the constants occurring in the (in)equations of the formula, the homogeneous terms, and the divisibility relations. The upper bound does *not* depend on the Boolean combination of the atomic formulas. This is not obvious since Boolean connectives are handled by the product construction if we construct the DWA recursively over the structure of the quantifier-free formula. The size of the resultant DWA using the product construction is in the worst case the product of the number of states of the given DWAs.

Let T be a finite nonempty set of homogeneous terms and let D be a finite set of atomic formulas of the form $d|t$, where $d \ge 1$ and t is a homogeneous term. Moreover, let $\ell > \max\big(\{\|t\|_+ : t \in \mathsf{T}\} \cup \{\|t\|_- : t \in \mathsf{T}\}\big)$ and $\ell' > \max\{d : d|t \in \mathsf{D}\}$.

THEOREM 3.12. *Let ψ be a Boolean combination of atomic formulas $t \lessgtr c$ and $d|t + c'$, with $t \in \mathsf{T}$, $d|t \in \mathsf{D}$, $-\ell < c < \ell$, $c' \in \mathbb{Z}$, and $\lessgtr \in \{=, \neq, <, \leq, >, \geq\}$. The size of the minimal DWA for ψ is at most $(2 + 2\ell)^{|\mathsf{T}|} \cdot \ell'^{|\mathsf{D}|}$.*

PROOF. Without loss of generality, we assume that the variables occurring in terms in T are $y_1, \ldots, y_r$. Let $\mathcal{C}$ be the product automaton of all the $\mathcal{A}^{t=0}_{(-\ell,\ell)}$s and $\mathcal{A}^{d|t'}$s, for $t \in \mathsf{T}$ and $d|t' \in \mathsf{D}$. To simplify notation we omit the subscripts $(-\ell, \ell)$ and we assume that $\mathsf{T} = \{t_1, \ldots, t_m\}$ and $\mathsf{D} = \{d_1|t'_1, \ldots, d_n|t'_n\}$. Note that the states of $\mathcal{C}$ are tuples $(p_1, \ldots, p_m, q_1, \ldots, q_n)$, where p_i is a state of $\mathcal{A}^{t_i=0}$ and q_j is a state of $\mathcal{A}^{d_j|t'_j}$. By Lemma 3.6, $\mathcal{A}^{t_i=0}$ has $2 + 2\ell$ states, and by Lemma 3.11, $\mathcal{A}^{d_j|t'_j}$ has $1 + d_j \leq \ell'$ states. It follows that the size of $\mathcal{C}$ is at most

$$\prod_{t \in \mathsf{T}} (2 + 2\ell) \cdot \prod_{d|t \in \mathsf{D}} (1 + d) \leq (2 + 2\ell)^{|\mathsf{T}|} \cdot \ell'^{|\mathsf{D}|} .$$

It remains to define the set of accepting states of $\mathcal{C}$ according to ψ. We define the DWA $\mathcal{D}$ as $\mathcal{C}$ except the set E of accepting states is defined as follows. A state $q = (p_1, \ldots, p_m, q_1, \ldots, q_n) \in \mathbb{Z}^{m+n}$ of $\mathcal{D}$ is in E iff $\mathfrak{Z} \models \psi_q$, where ψ_q is the formula obtained by substituting

—the integer p_i for the term t_i in the atomic formulas of the form $t_i \lessgtr c$, and

—the integer q_j for the term t'_j in the atomic formulas of the form $d_j|t'_j + c$.

Note that ψ_q is either true or false in $\mathfrak{Z}$ since it is a sentence.

It remains to prove that $\mathcal{D}$ represents $[\![\psi]\!]$. Let $w \in (\Sigma^r)^+$ be a word representing $\overline{a} \in \mathbb{Z}^r$. For a term $t \in \mathsf{T}$, the value $t[\overline{a}]$ can be replaced by ℓ if $t[\overline{a}] \geq \ell$ and by $-\ell$ if $t[\overline{a}] \leq -\ell$ in every atomic formula of the form $t \lessgtr c$ without changing its truth value since $-\ell < c < \ell$. This modified value corresponds to the state reached by $\mathcal{A}^{t=0}$ after reading the word w. For an atomic formula of the form $d|t + c$, with $d|t \in \mathsf{D}$, we can replace $t[\overline{a}] + c$ by $\mathrm{rem}(t[\overline{a}] + c, d)$ without changing the truth value. This adjusted value corresponds to the state reached by $\mathcal{A}^{d|t}$ after reading the word w. From the definition of E, it follows that $w \in L(\mathcal{D})$ iff $\mathfrak{Z} \models \psi[\overline{a}]$. □

4. AN UPPER BOUND ON THE AUTOMATA SIZE

In this section, we give an upper bound on the size of the minimal DWA for PA formulas. We obtain this bound by examining the quantifier-free formulas constructed by applying Reddy and Loveland's quantifier-elimination method [Reddy and Loveland 1978], which improves Cooper's quantifier-elimination method [Cooper 1972]. We use Reddy and Loveland's quantifier-elimination method since the produced formulas are "small" with respect to the following parameters on which the upper bound of the minimal DWA in Theorem 3.12 depends.

Definition 4.1. For $\varphi \in \mathsf{PA}$, we define

$$\mathsf{T}(\varphi) := \{t : t \lessgtr c \in \mathsf{A}(\varphi)\} ,$$
$$\mathsf{D}(\varphi) := \{d|t : d|t + c \in \mathsf{A}(\varphi)\} ,$$

and

$$\begin{aligned}\max_{\text{coef}}(\varphi) &:= \max\{1\} \cup \{|k| \,:\, k \text{ is a coefficient in } t \lessgtr c \in \mathsf{A}(\varphi)\}\,,\\ \max_{\text{const}}(\varphi) &:= \max\{1\} \cup \{|c| \,:\, t \lessgtr c \in \mathsf{A}(\varphi)\}\,,\\ \max_{\text{div}}(\varphi) &:= \max\{1\} \cup \{d \,:\, d|t + c \in \mathsf{A}(\varphi)\}\,.\end{aligned}$$

4.1 Eliminating a Quantifier

For the sake of completeness, we recall Reddy and Loveland's quantifier-elimination method. Consider the formula $\exists x\varphi$ with $\varphi(x, \overline{y}) \in \mathsf{QF}$. The construction of $\psi(\overline{y}) \in \mathsf{QF}$ proceeds in 2 steps.

Step 1: First, eliminate the connectives $\rightarrow$ and $\leftrightarrow$ in φ using standard rules, e. g., a subformula $\chi \rightarrow \chi'$ is replaced by $\neg\chi \vee \chi'$. Second, push all negation symbols in φ inward (using De Morgan's laws, etc.) until they only occur directly in front of the atomic formulas. Third, rewrite all atomic formulas and negated atomic formulas in which x occurs such that they are of one of the forms

$$k \cdot x < t(y_1, \dots, y_n)\,, \tag{A}$$

$$t(y_1, \dots, y_n) < k \cdot x\,, \tag{B}$$

or

$$d \,\big|\, t(x, y_1, \dots, y_n) \tag{C}$$

with $k > 0$. For instance, the negated inequation $\neg 2 \cdot x + 9 \cdot y < 5$ is rewritten into $-9 \cdot y + 5 - 1 < 2 \cdot x$, and the negated equation $\neg 2 \cdot x + 9 \cdot y = 5$ is replaced by the disjunction $-9 \cdot y + 5 < 2 \cdot x \vee 2 \cdot x < -9 \cdot y + 5$. Let $\varphi'(x, \overline{y})$ be the resulting formula.

Step 2: Let $\psi_{-\infty}$ be the formula, where all the atomic formulas of type (A) in φ' are replaced by "true", i. e., $0 < 1$, and all atomic formulas of type (B) are replaced by "false", i. e., $1 < 0$. We assume in the following, without loss of generality, that $0<1$ and $1<0$ do not occur as proper subformulas. Note that by propositional reasoning, we can always eliminate such subformulas, e. g., $\alpha \wedge 0 < 1$ can be simplified to α.

We define $\text{lcm}(x, \varphi)$ as the least common multiple of the ds in the atomic formulas of type (C) in the formula φ and of the coefficients of the variable x in the atomic formulas of type (B) in φ. Let B be the set of the atomic formulas in φ' of type (B). Let ψ be the formula

$$\bigvee_{1 \le j \le \text{lcm}(x,\varphi)} \psi_{-\infty}[j/x] \;\vee \bigvee_{t+c<k\cdot x \in \mathsf{B}} \; \bigvee_{1 \le j \le k\cdot \text{lcm}(x,\varphi)} \big(k \,\big|\, t + c + j \wedge \varphi'[t + c + j/k \cdot x]\big)\,,$$

where $\varphi'[t+c+j/k \cdot x]$ means that every atomic formula α in φ' in which x occurs is first multiplied by k and then $k \cdot x$ is substituted by $t + c + j$. Formally, for an atomic formula α, a term t, and $k \in \mathbb{Z} \setminus \{0\}$, we define

$$\alpha[t/k \cdot x] := \begin{cases} k' \cdot t < k \cdot t' & \text{if } \alpha = k' \cdot x < t',\\ k \cdot t' < k' \cdot t & \text{if } \alpha = t' < k' \cdot x,\\ kd|k' \cdot t + k \cdot t' & \text{if } \alpha = d|k' \cdot x + t',\\ \alpha & \text{otherwise.}\end{cases}$$

Oppen's [1978] correctness proof for Cooper's [1972] quantifier-elimination method can be adapted to the above described quantifier-elimination method by Reddy and Loveland [1978].

THEOREM 4.2. *The formula ψ is logically equivalent to $\exists x\varphi$.*

4.2 Analysis

We can construct from an arbitrary formula a logically equivalent quantifier-free formula by successively replacing subformulas of the form $Qx\varphi$, where $\varphi \in \mathsf{QF}$ and $Q \in \{\exists, \forall\}$, with the logically equivalent quantifier-free formulas that are produced by the quantifier-elimination method. Oppen [1978] analyzed the length of the formulas that are produced by iteratively applying Cooper's quantifier-elimination method. He proved a triple exponential upper bound on the formula length by relating the growth in the number of atomic formulas, the maximum of the absolute values of constants and coefficients appearing in these atomic formulas, and the number of distinct coefficients and divisibility predicates that may appear. Similar analysis of improved versions of Cooper's quantifier-elimination method are in [Reddy and Loveland 1978; Grädel 1988].

Reddy and Loveland [1978] observed that they obtain shorter formulas when pushing quantifiers inward before applying their quantifier-elimination method. For example, using the quantifier-elimination method to eliminate the quantified variable x_2 in $\exists x_1 \exists x_2 \varphi$ with $\varphi \in \mathsf{QF}$, we obtain a formula of the form $\exists x_1(\varphi_1 \vee \ldots \vee \varphi_n)$. Instead of applying the quantifier-elimination method to $\exists x_1(\varphi_1 \vee \ldots \vee \varphi_n)$, rewriting the formula first into $(\exists x_1 \varphi_1) \vee \ldots \vee (\exists x_1 \varphi_n)$ and then applying the quantifier-elimination method to each of the disjuncts separately produces shorter formulas due to the following reasons. First, we avoid using $\mathrm{lcm}(x_1, \varphi_1 \vee \ldots \vee \varphi_n)$ in Step 2 of the quantifier-elimination method; instead we determine $\mathrm{lcm}(x_1, \varphi_i)$, for each disjunct φ_i separately. Second, we use an inequation $t < k \cdot x_1$ of type (B) occurring in a disjunct φ_i only for eliminating x_1 in φ_i. We do not use this inequation $k \cdot x_1 < t$ for eliminating x_1 in disjuncts φ_j in which the inequation $k \cdot x_1 < t$ does not occur. However, if the variable x_1 is universally quantified, then we cannot push the quantifier inward. Note that in order to apply the quantifier-elimination method, we have to rewrite the formula $\forall x_1(\varphi_1 \vee \ldots \vee \varphi_n)$ into $\neg \exists x_1(\neg(\varphi_1 \vee \ldots \vee \varphi_n))$. To eliminate x_1, we have to use in Step 2 $\mathrm{lcm}(x_1, \neg(\varphi_1 \vee \ldots \vee \varphi_n))$ and the set B of the inequations of type (B) occurring in the formula produced by Step 1 normalizing $\neg(\varphi_1 \vee \ldots \vee \varphi_n)$.

Reddy and Loveland [1978] analyzed the quantifier-free formulas produced by successively applying their quantifier-elimination method to formulas in prenex normal form. We refine and extend their analysis to arbitrary formulas. However, before launching into the analysis, we need the following definitions. For $\varphi \in \mathsf{PA}$, we define

$$\mathsf{T}_+(\varphi) := \{t \in \mathsf{T}(\varphi) \,:\, t \text{ contains a variable that is bound in } \varphi\}$$

and

$$\mathsf{D}_+(\varphi) := \{d|t \in \mathsf{D}(\varphi) \,:\, t \text{ contains a variable that is bound in } \varphi\}\,.$$

Furthermore, let $\mathsf{T}_-(\varphi) := \mathsf{T}(\varphi) \setminus \mathsf{T}_+(\varphi)$ and $\mathsf{D}_-(\varphi) := \mathsf{D}(\varphi) \setminus \mathsf{D}_+(\varphi)$.

LEMMA 4.3. *For every formula $\varphi \in \mathsf{PA}$ of the form $Qx_1 \dots Qx_s\vartheta$ with $Q \in \{\exists, \forall\}$ and $\vartheta \in \mathsf{QF}$, there is a logically equivalent formula $\psi \in \mathsf{QF}$ such that*

$$|\mathsf{T}(\psi) \setminus \mathsf{T}_-(\varphi)| \leq |\mathsf{T}_+(\varphi)|^{s+1},$$
$$|\mathsf{D}(\psi) \setminus \mathsf{D}_-(\varphi)| \leq \big(|\mathsf{T}_+(\varphi)| + 1\big)^s \cdot \big(|\mathsf{D}_+(\varphi)| + s\big),$$

and

$$\max_{\mathrm{coef}}(\psi) < a^{2^{2s}},$$
$$\max_{\mathrm{div}}(\psi) < a^{2^{2s}},$$
$$\max_{\mathrm{const}}(\psi) < ba^{2^{2s}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+s)},$$

where $a > \max\{2, \max_{\mathrm{coef}}(\varphi), \max_{\mathrm{div}}(\varphi)\}$ and $b > \max\{2, \max_{\mathrm{const}}(\varphi)\}$.

PROOF. We first describe how we construct the quantifier-free formula ψ, where we assume that $Q = \exists$. For $Q = \forall$, we rewrite φ into $\neg\exists x_1 \dots \exists x_s\neg\vartheta$ and construct the quantifier-free formula for $\exists x_1 \dots \exists x_s\neg\vartheta$ as described below.

By a preprocessing step we rewrite ϑ into negation norm form (i. e., we eliminate the connectives $\rightarrow$ and $\leftrightarrow$, and we push the negation symbols inward such that the connective $\neg$ only occurs directly in front of atomic formulas) and we rewrite (in)equations so that we only have inequations of the form $t < t'$ or $t > t'$ and no negation occurs in front of an inequation. For instance, $t \leq t'$ is rewritten into $t < t' + 1$ and $\neg t \leq t'$ is rewritten into $t > t'$. Let ϑ_0 be the formula that we obtain by the rewriting. The only parameter that is changed by this rewriting is the maximal absolute value of a constant, which increases by at most 1. Observe that this special form of a formula is preserved when we apply the quantifier-elimination method: In Step 1 we only rewrite the inequations such that they are of type (A) or (B). Such rewriting does not alter the parameters. Step 2 also preserves this special form.

After the preprocessing step, we construct the quantifier-free formula ψ iteratively in s steps by constructing intermediate formulas $\varphi_0, \dots, \varphi_s$, where ψ will be φ_s. Let $\varphi_0 := \exists x_1 \dots \exists x_s\vartheta_0$. In the ℓth step we eliminate the variable $x_{s-\ell+1}$, where $1 \leq \ell \leq s$. This is done as follows. Assume that $\varphi_{\ell-1} = \exists x_1 \dots \exists x_{s-\ell+1}\vartheta_{\ell-1}$, where $\vartheta_{\ell-1} = \vartheta_{\ell-1,1} \vee \dots \vee \vartheta_{\ell-1,n_{\ell-1}}$. We push the existential quantification of $x_{s-\ell+1}$ inward in $\vartheta_{\ell-1}$ such that $x_{s-\ell+1}$ is quantified in front of each $\vartheta_{\ell-1,i}$. For every $1 \leq i \leq n_{\ell-1}$, we apply the quantifier-elimination method to $\exists x_{s-\ell+1}\vartheta_{\ell-1,i}$. After the $n_{\ell-1}$ applications of the quantifier-elimination method, we obtain for some $n_\ell \geq 1$, a formula $\vartheta_\ell := \vartheta_{\ell,1} \vee \dots \vee \vartheta_{\ell,n_\ell}$ that is logically equivalent to $\exists x_{s-\ell+1}\vartheta_{\ell-1}$. Let $\varphi_\ell := \exists x_1 \dots \exists x_{s-\ell}\vartheta_\ell$.

We now prove the upper bounds on the parameters of ψ. Let $n_0 := 1$ and $\vartheta_{0,1} := \vartheta_0$.

For proving the upper bounds on $|\mathsf{T}(\psi) \setminus \mathsf{T}_-(\varphi)|$ and $|\mathsf{D}(\psi) \setminus \mathsf{D}_-(\varphi)|$, we need the following two facts, which we prove by induction over $0 \leq \ell \leq s$:

(i) There are indices $1 \leq i_1, \dots, i_k \leq n_\ell$ with $k \leq |\mathsf{T}_+(\varphi)|^\ell$ such that for every index $1 \leq i \leq n_\ell$, there is an index $i' \in \{i_1, \dots, i_k\}$ with

$$\mathsf{T}(\vartheta_{\ell,i}) \subseteq \mathsf{T}(\vartheta_{\ell,i'}).$$

(ii) There are indices $1 \leq i_1, \dots, i_k \leq n_\ell$ with $k \leq (|\mathsf{T}_+(\varphi)| + 1)^\ell$ such that for every index $1 \leq i \leq n_\ell$, there is an index $i' \in \{i_1, \dots, i_k\}$ with

$$\mathsf{D}(\vartheta_{\ell,i}) \subseteq \mathsf{D}(\vartheta_{\ell,i'})\,.$$

From (i), we obtain the upper bounds on $|\mathsf{T}(\psi) \setminus \mathsf{T}_-(\varphi)|$: There are indices $1 \leq i_1, \dots, i_k$ with $k \leq |\mathsf{T}_+(\varphi)|^s$ and $\mathsf{T}(\psi) = \mathsf{T}(\varphi_\ell) = \mathsf{T}(\vartheta_{\ell,i_1}) \cup \dots \cup \mathsf{T}(\vartheta_{\ell,i_k})$. Since $|\mathsf{T}(\vartheta_{\ell,j}) \setminus \mathsf{T}_-(\varphi)| \leq |\mathsf{T}_+(\varphi)|$, for each $j \in \{i_1, \dots, i_k\}$, we conclude that $|\mathsf{T}(\psi) \setminus \mathsf{T}_-(\varphi)| \leq |\mathsf{T}_+(\varphi)| \cdot |\mathsf{T}_+(\varphi)|^s = |\mathsf{T}_+(\varphi)|^{s+1}$. Analogously, we obtain the upper bound on $|\mathsf{D}(\psi) \setminus \mathsf{D}_-(\varphi)|$ by using (ii).

We only carry out the induction for (i). We can use similar arguments in the induction for (ii). The base case for $\ell = 0$ is obvious. For the step case, let $\ell > 0$. Recall that the formula $\varphi_{\ell-1}$ has the form $\exists x_1 \dots \exists x_{s-\ell+1}(\vartheta_{\ell-1,1} \vee \dots \vee \vartheta_{\ell-1,n_{\ell-1}})$ and we obtain the formula $\varphi_\ell = \exists x_1 \dots \exists x_{s-\ell}(\vartheta_{\ell,1} \vee \dots \vee \vartheta_{\ell,n_\ell})$ by eliminating the variable $x_{s-\ell+1}$ in each disjunct $\vartheta_{\ell-1,i}$ separately. If $\xi_1 \vee \dots \vee \xi_m$ is the formula that we obtain by applying the quantifier-elimination method to a disjunct $\vartheta_{\ell-1,i}$ then there are indices $1 \leq \mu_1, \dots, \mu_h \leq m$ with $h \leq |\mathsf{T}_+(\varphi)|$ such that for every index $1 \leq \nu \leq m$, there is an index $\nu' \in \{\mu_1, \dots, \mu_h\}$ with $\mathsf{T}(\xi_\nu) \subseteq \mathsf{T}(\xi_{\nu'})$. This follows from the following observations about the quantifier-elimination method. First, the subformula $(\vartheta_{\ell-1,i})_{-\infty}$ used in Step 2 contains at most the (in)equations of $\vartheta_{\ell-1,i}$ in which $x_{s-\ell+1}$ does not occur. Second, the cardinality of the set B in Step 2 is at most $|\mathsf{T}_+(\varphi)|$. Third, for different values of j in Step 2, we obtain the same homogeneous terms.

Let $1 \leq i_1, \dots, i_k \leq n_{\ell-1}$ be the indices from the induction hypothesis. Note that $k \leq |\mathsf{T}_+(\varphi)|^{\ell-1}$. For $1 \leq i \leq n_{\ell-1}$, we have that the homogeneous terms of the (in)equations in a disjunct that is obtained by applying the quantifier-elimination method to $\vartheta_{\ell-1,i}$ occur also in a disjunct that is obtained by applying the quantifier-elimination method to $\vartheta_{\ell-1,j}$, for some $j \in \{i_1, \dots, i_k\}$. It follows that there are indices $1 \leq i'_1, \dots, i'_{k'} \leq n_\ell$ with $k' \leq |\mathsf{T}_+(\varphi)|^\ell$ such that for each $1 \leq \nu \leq n_\ell$, there is some $\nu' \in \{i'_1, \dots, i'_{k'}\}$ with $\mathsf{T}(\vartheta_{\ell,\nu}) \subseteq \mathsf{T}(\vartheta_{\ell,\nu'})$.

Now, we establish the upper bounds on $\max_{\text{coef}}(\psi)$, $\max_{\text{div}}(\psi)$, and $\max_{\text{const}}(\psi)$: we prove by induction over ℓ that

$$\max\nolimits_{\text{coef}}(\varphi_\ell), \max\nolimits_{\text{div}}(\varphi_\ell) < a^{2^{2\ell}} \quad \text{and} \quad \max\nolimits_{\text{const}}(\varphi_\ell) < ba^{2^{2\ell}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell)}\,.$$

For $\ell = 0$, these upper bounds obviously hold. Assume that $\ell > 0$. For $1 \leq i \leq n_{\ell-1}$, we examine the formula produced by the quantifier-elimination method applied to $\exists x_{s-\ell+1}\vartheta_{\ell-1,i}$. Because of our preprocessing step by rewriting ϑ to ϑ_0 note that Step 1 of the quantifier-elimination method does not alter the absolute values of the coefficients and constants, and the ds in the divisibility predicate. It suffices to look at the substitutions $\alpha[t + c + j/k \cdot x]$ carried out in Step 2, where α is an atomic formula in $\vartheta_{\ell-1,i}$, $t + c < k \cdot x$ is an inequation of type (B) in $\vartheta_{\ell-1,i}$, and $1 \leq j \leq k \cdot \operatorname{lcm}(x_{s-\ell+1}, \vartheta_{\ell-1,i})$.

—Assume that $\alpha = d|t$, for some $d \geq 1$ and some term t. By the induction hypothesis, we have that

$$kd < a^{2^{2(\ell-1)}} \cdot a^{2^{2(\ell-1)}} = a^{2^{2\ell-1}} \leq a^{2^{2\ell}}\,.$$

It follows that $\max_{\text{div}}(\varphi_\ell) < a^{2^{2\ell}}$.

—Assume that $\alpha = k' \cdot x < t'$ or $\alpha = t' < k' \cdot x$, for some $k' > 0$ and some term t'. By the induction hypothesis, we have that k, k', and the absolute values of the coefficients occurring in t and t' are smaller than $a^{2^{2(\ell-1)}}$. It follows that the absolute values of the coefficients in the normalized inequations of $k' \cdot (t + c + j) < k \cdot t'$ and $k \cdot t' < k' \cdot (t + c + j)$ are smaller than

$$a^{2^{2(\ell-1)}} \cdot a^{2^{2(\ell-1)}} + a^{2^{2(\ell-1)}} \cdot a^{2^{2(\ell-1)}} = 2a^{2^{2\ell-1}} \leq a^{2^{2\ell}} .$$

Hence, $\max_{\mathrm{coef}}(\varphi_\ell) < a^{2^{2\ell}}$.
The absolute values of the constants in the normalized inequations $k' \cdot (t + c + j) < k \cdot t'$ and $k \cdot t' < k' \cdot (t + c + j)$ are bounded by

$$\max_{\mathrm{coef}}(\varphi_{\ell-1}) \cdot \big(\max_{\mathrm{const}}(\varphi_{\ell-1}) + k \cdot \mathrm{lcm}(x_{s-\ell+1}, \vartheta_{\ell-1,i})\big) + \max_{\mathrm{coef}}(\varphi_{\ell-1}) \cdot \max_{\mathrm{const}}(\varphi_{\ell-1}) ,$$

which rewrites into

$$\max_{\mathrm{coef}}(\varphi_{\ell-1}) \cdot \big(2 \max_{\mathrm{const}}(\varphi_{\ell-1}) + k \cdot \mathrm{lcm}(x_{s-\ell+1}, \vartheta_{\ell-1,i})\big) . \tag{9}$$

An upper bound on $\mathrm{lcm}(x_{s-\ell+1}, \vartheta_{\ell-1,i})$ is

$$\big(a^{2^{2(\ell-1)}}\big)^{|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1} = a^{2^{2(\ell-1)} \cdot (|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)}$$

since we determine the least common multiple of at most $|\mathsf{T}_+(\varphi)| + |\mathsf{D}_+(\varphi)| + \ell - 1$ numbers and all these numbers are bounded by $a^{2^{2(\ell-1)}}$. By the induction hypothesis, we have that $|c|$ and the absolute value of the constant in t' are both smaller than $ba^{2^{2(\ell-1)}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)}$. Therefore, (9) is smaller than

$$a^{2^{2(\ell-1)}}\big(2ba^{2^{2(\ell-1)}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)} + a^{2^{2(\ell-1)}} \cdot a^{2^{2(\ell-1)}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)}\big) .$$

An upper bound is

$$a^{2^{2\ell-1}} \cdot ba^{2^{2(\ell-1)}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)} \leq ba^{2^{2\ell}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell-1)} .$$

It follows that $\max_{\mathrm{const}}(\varphi_\ell) < ba^{2^{2\ell}(|\mathsf{T}_+(\varphi)|+|\mathsf{D}_+(\varphi)|+\ell)}$. □

By iteratively applying Lemma 4.3 we obtain the following upper bounds for formulas in prenex normal form.

LEMMA 4.4. *For every formula $\varphi \in \mathsf{PA}$ of the form $Q_1x_1 \dots Q_rx_r\psi_0$ with $\psi_0 \in \mathsf{QF}$, there is a logically equivalent formula $\psi \in \mathsf{QF}$ such that*

$$|\mathsf{T}(\psi)| \leq T^{(\ell+1)^{\mathrm{qa}(\varphi)}} \qquad \textit{and} \qquad |\mathsf{D}(\psi)| \leq DT^{(\ell+1)^{\mathrm{qa}(\varphi)+2}} ,$$

where $T = \max\{2, |\mathsf{T}(\varphi)|\}$, $D = \max\{1, |\mathsf{D}(\varphi)|\}$, and ℓ is the maximal length of a quantifier block in φ. Furthermore, it holds that

$$\max_{\mathrm{coef}}(\psi) < a^{2^{2\,\mathrm{qn}(\varphi)}} ,$$
$$\max_{\mathrm{div}}(\psi) < a^{2^{2\,\mathrm{qn}(\varphi)}} ,$$

and

$$\max_{\mathrm{const}}(\psi) < ba^{2^{3\,\mathrm{qn}(\varphi)}DT^{(\ell+1)^{\mathrm{qa}(\varphi)+2}}} ,$$

where $a > \max\{2, \max_{\mathrm{coef}}(\varphi), \max_{\mathrm{div}}(\varphi)\}$ and $b > \max\{2, \max_{\mathrm{const}}(\varphi)\}$.

Proof. We construct the quantifier-free formula ψ by successively eliminating the quantifier blocks in φ, starting from the innermost one. Assume that after the kth step, where $0 \leq k < \mathrm{qa}(\varphi)$, we have produced the formula

$$Q_1 x_1 \dots Q_i x_i Q x_{i+1} \dots Q x_j \psi_k \,,$$

where $1 \leq i < j \leq r$, $Q_1, \dots, Q_i, Q \in \{\exists, \forall\}$ with $Q_i \neq Q$, and $\psi_k \in \mathsf{QF}$. Let $\psi_{k+1} \in \mathsf{QF}$ be the formula from Lemma 4.3 that is logically equivalent to $\varphi_k := Q x_{i+1} \dots Q x_j \psi_k$. We define $\psi := \psi_{\mathrm{qa}(\varphi)}$.

For $1 \leq i \leq \mathrm{qa}(\varphi)$, let ℓ_i be the length of the ith quantifier block. We prove by induction over $0 \leq k \leq \mathrm{qa}(\varphi)$ that

$$|\mathsf{T}(\psi_k)| \leq T^{(\ell+1)^k} \qquad \text{and} \qquad |\mathsf{D}(\psi_k)| \leq DT^{(\ell+1)^{k+2}} \,,$$
$$\max_{\mathrm{coef}}(\psi_k) < a^{2^{2(\ell_1+\dots+\ell_k)}} \qquad \text{and} \qquad \max_{\mathrm{div}}(\psi_k) < a^{2^{2(\ell_1+\dots+\ell_k)}} \,,$$

and

$$\max_{\mathrm{const}}(\psi_k) < ba^{2^{3(\ell_1+\dots+\ell_k)} DT^{(\ell+1)^{k+2}}} \,.$$

The base cases for $k = 0$ are trivial. For the step cases, let $k > 0$.

1. By Lemma 4.3, we have that

$$\begin{aligned} |\mathsf{T}(\psi_k) \setminus \mathsf{T}_-(\varphi_{k-1})| &\leq |\mathsf{T}_+(\varphi_{k-1})|^{\ell+1} \\ &\leq |\mathsf{T}(\psi_{k-1})|^{\ell+1} \overset{\mathrm{IH}}{\leq} \left(T^{(\ell+1)^{k-1}}\right)^{\ell+1} = T^{(\ell+1)^k} \end{aligned}$$

and

$$\begin{aligned} |\mathsf{D}(\psi_k) \setminus \mathsf{D}_-(\varphi_{k-1})| &\leq (|\mathsf{T}_+(\varphi_{k-1})| + 1)^{\ell} \cdot (|\mathsf{D}_+(\varphi_{k-1})| + \ell) \\ &\leq (|\mathsf{T}(\psi_{k-1})| + 1)^{\ell} \cdot (|\mathsf{D}(\psi_{k-1})| + \ell) \\ &\overset{\mathrm{IH}}{\leq} \left(T^{(\ell+1)^{k-1}} + 1\right)^{\ell} \cdot \left(DT^{(\ell+1)^{k+1}} + \ell\right) \\ &\leq 2^{\ell+1} DT^{(\ell+1)^k + (\ell+1)^{k+1}} \leq DT^{(\ell+1)+(\ell+1)^k+(\ell+1)^{k+1}} \\ &\leq DT^{(\ell+1)^{k+2}} \,. \end{aligned}$$

Note that $T \geq 2$ and $D \geq 1$.

2. By Lemma 4.3, we have that

$$\begin{aligned} \max_{\mathrm{coef}}(\psi_k) &\leq \left(\max\{2, \max_{\mathrm{coef}}(\psi_{k-1})\}\right)^{2^{2\ell_k}} \\ &\overset{\mathrm{IH}}{<} \left(a^{2^{2(\ell_1+\dots+\ell_{k-1})}}\right)^{2^{2\ell_k}} = a^{2^{2(\ell_1+\dots+\ell_k)}} \,. \end{aligned}$$

Analogously, we obtain the upper bound for $\max_{\mathrm{div}}(\psi_k)$.

3. By Lemma 4.3, we have that

$$
\begin{aligned}
\max_{\mathrm{const}}(\psi_k) &\leq \max_{\mathrm{const}}(\psi_{k-1}) \cdot \left(a^{2^{2(\ell_1+\dots+\ell_{k-1})}}\right)^{2^{2\ell_k}(|\mathsf{T}_+(\varphi_{k-1})|+|\mathsf{D}_+(\varphi_{k-1})|+\ell_k)} \\
&\leq \max_{\mathrm{const}}(\psi_{k-1}) a^{2^{2(\ell_1+\dots+\ell_k)}(|\mathsf{T}(\psi_{k-1})|+|\mathsf{D}(\psi_{k-1})|+\ell_k)} \\
&\leq \max_{\mathrm{const}}(\psi_{k-1}) a^{2^{2(\ell_1+\dots+\ell_k)}(T^{(\ell+1)^{k-1}}+DT^{(\ell+1)^{k+1}}+\ell_k)} \\
&\leq \max_{\mathrm{const}}(\psi_{k-1}) a^{2^{2(\ell_1+\dots+\ell_k)}(DT^{(\ell+1)^{k}}+DT^{(\ell+1)^{k+1}})} \\
&\leq \max_{\mathrm{const}}(\psi_{k-1}) a^{2^{2(\ell_1+\dots+\ell_k)}DT^{(\ell+1)^{k+2}}} \\
&\overset{\mathrm{IH}}{<} ba^{2^{3(\ell_1+\dots+\ell_{k-1})}DT^{(\ell+1)^{k+1}}} \cdot a^{2^{2(\ell_1+\dots+\ell_k)}DT^{(\ell+1)^{k+2}}} \\
&\leq ba^{(2^{3(\ell_1+\dots+s_{k-1})}+2^{2(\ell_1+\dots+\ell_k)})DT^{(\ell+1)^{k+2}}} \\
&\leq ba^{2^{3(\ell_1+\dots+\ell_k)}DT^{(\ell+1)^{k+2}}} . \quad \square
\end{aligned}
$$

Before we generalize Lemma 4.4 to arbitrary formulas, we want to point out that transforming a formula first into prenex normal form and then eliminating the quantifiers is not a good thing to do. The formula size can increase because of the following reasons.

First, a transformation into prenex normal form can increase the number of quantifier alternations. For instance, any transformation of $(\forall x\varphi) \land (\exists y\psi)$ into prenex normal form will introduce at least one additional alternation of quantifiers.

Second, when transforming a formula into prenex normal form we have to introduce fresh variables when pushing quantifiers to the front. As an example, consider the formula in prenex normal form

$$
\begin{aligned}
&\exists z_{n-1} \dots \exists z_2 \exists z_1 (x = z_{n-1} + z_{n-1} \land \\
&\qquad\qquad z_{n-1} = z_{n-2} + z_{n-2} \land \dots \land z_2 = z_1 + z_1 \land z_1 = y + y) ,
\end{aligned}
$$

for some $n \geq 1$. It consists of n distinct equations. A logically equivalent formula that consists of at most 4 distinct equations is

$$
\begin{aligned}
&\exists z\big(x = z + z \land \\
&\quad \exists z'(z = z' + z' \land \dots \land \exists z'(z = z' + z' \land \exists z(z' = z + z \land z = y + y)) \dots)\big) .
\end{aligned}
$$

Furthermore, the formula length decreases by a factor of $\mathrm{O}(\log n)$ since we use a fixed number of variables, i. e., we use x, y, z, z' instead of $x, y, z_1, \dots, z_{n-1}$.

The third reason why a transformation into prenex normal form is not a good idea is illustrated by the formula $(\forall x\varphi) \leftrightarrow \psi$. Quantifiers do not in general distribute over $\rightarrow$ and $\leftrightarrow$. Therefore, we eliminate the connective $\leftrightarrow$ and obtain $((\forall x\varphi) \rightarrow \psi) \land (\psi \rightarrow \forall x\varphi)$. Eliminating $\rightarrow$ yields $((\neg\forall x\varphi) \lor \psi) \land (\neg\psi \lor \forall x\varphi)$. To move the quantifiers to the front, we have to push the first negation inward. Finally, we obtain $\exists x\forall x'((\neg\varphi \lor \psi) \land (\neg\psi \lor \varphi[x'/x]))$ assuming that x does not occur free in ψ, and x' does not occur free in φ and ψ. We have not only doubled the length of the formula but we have also doubled the number of quantifiers. We want to *eliminate* quantifiers and have ended up doubling our work.

In analogy to the maximum of the lengths of the quantifier blocks of a formula in prenex normal form, we define the *quantifier block length* of the formula φ as

$$\mathrm{qbl}(\varphi) := \max\{\mathrm{qbl}_Q(\psi) \,:\, Q \in \{\exists,\forall\} \text{ and } \psi \text{ is a subformula of } \varphi\}\,,$$

where

$$\mathrm{qbl}_Q(\varphi) := \begin{cases} \mathrm{qbl}_{\overline{Q}}(\psi) & \text{if } \varphi = \neg\psi,\\ \mathrm{qbl}_Q(\psi_1) + \mathrm{qbl}_Q(\psi_2) & \text{if } \varphi = \psi_1 \oplus \psi_2 \text{ with } \oplus \in \{\wedge,\vee\},\\ \mathrm{qbl}_Q(\neg\psi_1 \vee \psi_2) & \text{if } \varphi = \psi_1 \to \psi_2,\\ \mathrm{qbl}_Q((\psi_1 \to \psi_2) \wedge (\psi_2 \to \psi_1)) & \text{if } \varphi = \psi_1 \leftrightarrow \psi_2,\\ 1 + \mathrm{qbl}_Q(\psi) & \text{if } \varphi = Qx\psi,\\ 0 & \text{otherwise,} \end{cases}$$

for $Q \in \{\exists,\forall\}$.

THEOREM 4.5. *For every formula $\varphi \in \mathsf{PA}$ of length n, there is a logically equivalent formula $\psi \in \mathsf{QF}$ such that*

$$|\mathsf{T}(\psi)| \le n^{(\mathrm{qbl}(\varphi)+1)^{\mathrm{qa}(\varphi)}} \qquad \text{and} \qquad |\mathsf{D}(\psi)| \le n^{1+(\mathrm{qbl}(\varphi)+1)^{\mathrm{qa}(\varphi)+2}}$$

$$\max_{\mathrm{coef}}(\psi) < a^{2^{2\,\mathrm{qn}(\varphi)}} \qquad \text{and} \qquad \max_{\mathrm{div}}(\psi) < a^{2^{2\,\mathrm{qn}(\varphi)}}\,,$$

and

$$\max_{\mathrm{const}}(\psi) < ba^{2^{3\,\mathrm{qn}(\varphi)}n^{1+(\mathrm{qbl}(\varphi)+1)^{\mathrm{qa}(\varphi)+2}}}\,,$$

where $a > \max\{2, \max_{\mathrm{coef}}(\varphi), \max_{\mathrm{div}}(\varphi)\}$ and $b > \max\{2, \max_{\mathrm{const}}(\varphi)\}$.

PROOF. We require that variables are not reused in φ, i. e., the set of free variables of φ is disjoint from the set of bound variables and the bound variables are pairwise distinct. Note that this can be achieved by replacing quantified variables by fresh variables. Such a variable renaming can increase the number of distinct atomic formulas. However, the number of atomic formulas after such a renaming still is less than or equal to the length of the original formula. Note that $n \ge \max\{2, |\mathsf{T}(\varphi)|, |\mathsf{D}(\varphi)|\}$.

We construct the formula $\psi \in \mathsf{QF}$ in $\mathrm{qa}(\varphi)$ steps. Let $\varphi_0 := \varphi$. Let $0 < k \le \mathrm{qa}(\varphi)$ and assume that after the $(k-1)$st step we have produced the formula φ_{k-1}. Let Φ be the set of maximal subformulas ϑ of φ_{k-1} in which variables are either only existentially quantified or universally quantified, and $\mathrm{qa}(\vartheta) \le 1$. We can assume without loss of generality that every formula in Φ is in prenex normal form and that $\Phi = \{\vartheta_1, \dots, \vartheta_m\}$. For $1 \le i \le m$, let $\xi_i \in \mathsf{QF}$ be the logically equivalent formula to ϑ_i from Lemma 4.3. We replace in φ_{k-1} every ϑ_i by ξ_i. We obtain the formula φ_k that is logically equivalent to φ and $\mathrm{qa}(\varphi_k) = \mathrm{qa}(\varphi) - k$. We define $\psi := \varphi_{\mathrm{qa}(\varphi)}$.

For the formula φ_k, we have that

$$\mathsf{T}(\varphi_k) \subseteq \mathsf{T}(\varphi_{k-1}) \setminus \Big(\bigcup_{1\le i\le m} \mathsf{T}_+(\vartheta_i)\Big) \cup \bigcup_{1\le i\le m} \big(\mathsf{T}(\xi_i) \setminus \mathsf{T}_-(\vartheta_i)\big)\,.$$

Since variables are not reused in φ, it follows that

$$|\mathsf{T}(\varphi_k)| \le |\mathsf{T}(\varphi_{k-1})| - \sum_{1\le i\le m} |\mathsf{T}_+(\vartheta_i)| + \sum_{1\le i\le m} |\mathsf{T}_+(\vartheta_i)|^{\mathrm{qn}(\vartheta_i)+1}\,.$$

It is straightforward to show that the right hand side has its maximum when $m = 1$ and $|\mathsf{T}_+(\vartheta_1)| = |\mathsf{T}(\varphi_{k-1})|$. Analogously to the step case in the proof of Lemma 4.4 for formulas in prenex normal form, it follows that $|\mathsf{T}(\varphi_k)| \leq n^{(\mathrm{qbl}(\varphi)+1)^{k+1}}$ under the assumption that $|\mathsf{T}(\varphi_{k-1})| \leq n^{(\mathrm{qbl}(\varphi)+1)^k}$.

We can argue similarly for $|\mathsf{D}(\varphi_k)|$. As in the proof of Lemma 4.4 for formulas in prenex normal form we obtain the upper bounds for $\max_{\mathrm{coef}}(\varphi_k)$, $\max_{\mathrm{div}}(\varphi_k)$, and $\max_{\mathrm{const}}(\varphi_k)$. □

4.3 Main Result

We now prove our main result: The upper bound on the automata size of the minimal DWA for Presburger arithmetic formulas.

THEOREM 4.6. *The size of the minimal DWA for a formula $\varphi \in \mathsf{PA}$ of length n is at most $2^{n^{(\mathrm{qbl}(\varphi)+1)^{\mathrm{qa}(\varphi)+4}}}$.*

PROOF. Since we measure the length of integers linearly, we have that the absolute value of every integer occurring in φ is bounded by n. It holds that $n > \max_{\mathrm{const}}(\varphi)$, $n > \max_{\mathrm{coef}}(\varphi)$, and $n > \max_{\mathrm{div}}(\varphi)$.

For $\mathrm{qn}(\varphi) = 0$, we have that the size of the minimal DWA is at most 2^n. For every atomic formula α_i of length n_i in φ, we can build a DWA of size at most n_i by using the constructions in §3.2 and §3.3. Applying the product construct yields a DWA of size at most $\prod_{1\leq i\leq m} n_i \leq 2^{\sum_{1\leq i\leq m} n_i} \leq 2^n$, where m is the number of atomic formulas in φ.

In the following, assume that $\mathrm{qn}(\varphi) \geq 1$ and, therefore, we have that $\mathrm{qa}(\varphi) \geq 1$ and $\mathrm{qbl}(\varphi) \geq 1$. For the sake of readability, we define $a := \mathrm{qa}(\varphi)$ and $\ell := \mathrm{qbl}(\varphi)$. From Theorem 4.5 it follows that there is a logically equivalent $\psi \in \mathsf{QF}$ with

$$|\mathsf{T}(\psi)| \leq n^{(\ell+1)^a} \qquad \text{and} \qquad |\mathsf{D}(\psi)| \leq n^{1+(\ell+1)^{a+2}} .$$

Upper bounds on $\max_{\mathrm{coef}}(\psi)$, $\max_{\mathrm{div}}(\psi)$, and $\max_{\mathrm{const}}(\psi)$ are

$$\max_{\mathrm{coef}}(\psi), \max_{\mathrm{div}}(\psi) < n^{2^{2\,\mathrm{qn}(\varphi)}} \leq 2^{2^{2a\ell}\log_2 n} \leq 2^{n^{1+2a\ell}}$$

and

$$\max_{\mathrm{const}}(\psi) < n^{1+2^{3\,\mathrm{qn}(\varphi)} n^{1+(\ell+1)^{a+2}}} \leq 2^{n^{3+3a\ell+(\ell+1)^{a+2}}} \leq 2^{n^{(\ell+1)^{a+1}+(\ell+1)^{a+2}}} .$$

Note that $n \geq 2$, $a\ell \geq \mathrm{qn}(\varphi)$, and $x^y = 2^{y\log_2 x}$, for $x \geq 1$ and $y \geq 0$.

Assume that there are $r \leq n$ free variables in φ. Since every term in ψ contains at most the free variables of φ, the sum of the absolute values of the coefficients in a term is bounded by $n \cdot n^{2^{2\,\mathrm{qn}(\varphi)}} \leq 2^{n^{2+2a\ell}} < 2^{n^{3+3a\ell}}$. With Theorem 3.12 at hand, we know that the size of the minimal DWA for ψ is at most

$$\left(2 + 2 \cdot 2^{n^{(\ell+1)^{a+1}+(\ell+1)^{a+2}}}\right)^{|\mathsf{T}(\psi)|} \cdot \max_{\mathrm{div}}(\psi)^{|\mathsf{D}(\psi)|} .$$

From

$$\left(2 + 2 \cdot 2^{n^{(\ell+1)^{a+1}+(\ell+1)^{a+2}}}\right)^{|\mathsf{T}(\psi)|} \leq 2^{n^{(\ell+1)^{a+1}+(\ell+1)^{a+2}+(\ell+1)^a}} \leq 2^{n^{(\ell+1)^{a+3}}}$$

and

$$\max_{\mathrm{div}}(\psi)^{|\mathsf{D}(\psi)|} \leq 2^{n^{2+2a\ell+(\ell+1)^{a+2}}} \leq 2^{n^{2(\ell+1)^a+(\ell+1)^{a+2}}} \leq 2^{n^{(\ell+1)^{a+3}}}$$

we conclude that the size of the minimal DWA for φ is at most $2^{n^{(\ell+1)^{a+4}}}$. □

Theorem 4.6 does not change if we measure the length of integers logarithmically and not linearly. The only change is that the maximal absolute integer in φ is now smaller than 2^n. We have to adjust the bounds on $\max_{\mathrm{coef}}(\psi)$, $\max_{\mathrm{div}}(\psi)$, and $\max_{\mathrm{const}}(\psi)$. For instance, we still have that

$$\max_{\mathrm{coef}}(\psi) < (2^n)^{2^{2\,\mathrm{qn}(\varphi)}} = 2^{n2^{2\,\mathrm{qn}(\varphi)}} \leq 2^{n^{1+2\,\mathrm{qa}(\varphi)\,\mathrm{qbl}(\varphi)}}.$$

We argue analogously for $\max_{\mathrm{div}}(\psi)$ and $\max_{\mathrm{const}}(\psi)$.

Corollary 4.7. *Let PA_c be the set of PA formulas with at most $c \geq 0$ quantifiers. The size of the minimal DWA for each $\varphi \in \mathsf{PA}_c$ is at most $2^{n^{\mathrm{O}(1)}}$, where n is the length of φ.*

Proof. If $\mathrm{qn}(\varphi) \leq c$ then $\mathrm{qa}(\varphi) \leq c$ and $\mathrm{qbl}(\varphi) \leq c$. Since c is fixed the claim follows directly from Theorem 4.6. □

We want to remark that Theorem 4.6 and Corollary 4.7 only give upper bounds on the sizes of the minimal DWAs for PA formulas. If the Boolean connectives and the quantifiers are handled by standard automata constructions, like complementation and subset construction, and the DWAs are minimized after every automata construction step, it may be the case that the whole construction uses one exponent more space. The reason is that an exponential blow-up can occur each time the subset construction is applied. It is an open question whether the standard automata constructions already suffice to construct a DWA in $2^{n^{(\mathrm{qbl}(\varphi)+1)^{\mathrm{qa}(\varphi)+4}}}$ space or time, for a given $\varphi \in \mathsf{PA}$ of length n. It is also open if there are more efficient automata constructions than the standard ones for constructing DWAs for PA formulas.

5. A WORST CASE EXAMPLE

We give a worst case example that shows that our upper bound on the automata size is tight. We use the formulas $\mathrm{Prod}_n(x,y,z)$ defined by Fischer and Rabin [1974], for $n \geq 0$. It holds that

$$[\![\mathrm{Prod}_n]\!] = \{(a,b,c) \in \mathbb{N} : ab = c \text{ and } a,b,c < \prod_{\substack{p \text{ is prime and} \\ p<f(n+2)}} p\},$$

where $f(n) := 2^{2^n}$. Note that it follows from the Prime Number Theorem that

$$\prod_{\substack{p \text{ is prime and} \\ p<f(n+2)}} p \geq 2^{f(n)^2} = 2^{f(n+1)}.$$

Fischer and Rabin looked at the structure $(\mathbb{N},+)$ and not at $\mathfrak{Z}$, but it is straightforward to adapt the definition of $\mathrm{Prod}_n(x,y,z)$ to $\mathfrak{Z}$. For $n \geq 0$, the length of Prod_n and the number of quantifier alternations is linear in n. The quantifier block length is constant, i. e., there is a $c \geq 0$ such that for all $n \geq 0$, $\mathrm{qbl}(\mathrm{Prod}_n) = c$. By Theorem 4.6 we know that the minimal DWA for Prod_n has at most $2^{2^{2^{O(n)}}}$ states.

Before we prove the lower bound on the automata size for the formulas Prod_n, we need the following lemma.

LEMMA 5.1. *Let $\ell \geq 1$. For all $z \in \mathbb{N}$ with $\varrho^{\ell-1} \leq z \leq \varrho^\ell - 2$, there are $x, y, z' \in [\varrho^\ell]$ such that $xy = \varrho^\ell z + z'$.*

PROOF. Assume that $\varrho^{\ell-1} \leq z \leq \varrho^\ell - 2$. Let $x, y \in [\varrho^\ell]$ such that $xy \geq \varrho^\ell z$ and $xy - \varrho^\ell z$ is minimal. Note that it is always possible to find $x, y \in [\varrho^\ell]$ with $xy \geq \varrho^\ell z$ since for $x = y = \varrho^\ell - 1$, we have that

$$xy = (\varrho^\ell - 1)^2 = \varrho^{2\ell} - 2\varrho^\ell + 1 \geq \varrho^\ell(\varrho^\ell - 2) \geq \varrho^\ell z\,.$$

Let $z' := xy - \varrho^\ell z$. We have to show that $z' \in [\varrho^\ell]$. Since $xy \geq \varrho^\ell z$ we have that $z' \geq 0$. We prove $z' < \varrho^\ell$ by contradiction. Assume that $z' \geq \varrho^\ell$. It follows that

$$(x-1)y = xy - y = \varrho^\ell z + z' - y \geq \varrho^\ell z$$

since $y < \varrho^\ell$ and $z' \geq \varrho^\ell$. This contradicts the minimality of $xy - \varrho^\ell z$ since $xy > (x-1)y \geq \varrho^\ell z$. □

Our proof for the lower bound on the automata size for a formula Prod_n is based on the following lemma about the set

$$\mathrm{MULT}_m := \{(a, b, c) \in \mathbb{Z}^3 \,:\, a, b \in [\varrho^m] \text{ and } ab = c\}\,,$$

for $m \geq 0$.

LEMMA 5.2. *Let $m \geq 0$ and let $S \subseteq \mathbb{Z}^3$ be the graph of a partial function from $\mathbb{Z}^2$ to $\mathbb{Z}$ with $\mathrm{MULT}_m \subseteq S$. If S is definable in PA then every DWA representing S has at least ϱ^m states.*

PROOF. For $m = 0$, the claim is trivial since every DWA has at least 1 state. In the following, assume that $m > 0$ and that $\mathcal{A} = (Q, \Sigma^3, \delta, q_{\mathrm{I}}, F)$ is a DWA representing S.

Let K be the set of words of the form $(0,0,0)(0,0,b_{m-1})\dots(0,0,b_0) \in (\Sigma^3)^*$ with $b_{m-1} \neq 0$ and if $b_i = \varrho - 1$, for all $1 \leq i < m$, then $b_0 \leq \varrho - 2$. Let $w \in K$ and let z be the integer that is encoded by the third track of w. It holds that

$$\varrho^{m-1} \leq z \leq \varrho^m - 2\,.$$

From Lemma 5.1 it follows that there are $x, y, z' \in [\varrho^m]$ such that

$$xy = \varrho^m z + z'\,.$$

We conclude that for every prefix u of a word in K, there is a word $v \in (\Sigma^3)^*$ such that $\langle uv \rangle_{\mathbb{Z}} \in \mathrm{MULT}_m$.

Now, let L be the set of all prefixes of K. Let $u, u' \in L \setminus \{\lambda\}$ with $u \neq u'$. Moreover, let $v \in (\Sigma^3)^*$ with $\langle uv \rangle_{\mathbb{Z}} \in \mathrm{MULT}_m$. The first and second tracks of uv and $u'v$ encode both the pair (x, y). The third tracks of uv and $w'v$ are different. It follows that $\langle u'v \rangle_{\mathbb{Z}} \notin \mathrm{MULT}_m$. Since $\mathrm{MULT}_m \subseteq S$ and S is the graph of a partial function, we have that $\widehat{\delta}(q_{\mathrm{I}}, u) \neq \widehat{\delta}(q_{\mathrm{I}}, u')$. We conclude that the DWA $\mathcal{A}$ must have a distinct state for every word in L.

In the following, we determine the cardinality of L. For $0 \leq i \leq m+1$, let $L_i := \{w \in L : |w| = i\}$. We have that $L_0 = \{\lambda\}$, $L_1 = \{(0,0,0)\}$, $L_2 = \{(0,0,0)b : b \in \Sigma \setminus \{0\}\}$, $L_i = \{wb : w \in L_{i-1} \text{ and } b \in \Sigma\}$, for $3 \leq i \leq m$, and $L_{m+1} = K$. It holds that

$$\begin{aligned}|L| &= |L_0| + |L_1| + |L_2| + |L_3| + \cdots + |L_m| + |L_{m+1}| \\ &= 1 + 1 + (\varrho - 1) + (\varrho - 1)\varrho + \cdots + (\varrho - 1)\varrho^{m-2} + (\varrho - 1)\varrho^{m-1} - 2 \\ &= \varrho^m - 1\,.\end{aligned}$$

We conclude that $\mathcal{A}$ has at least ϱ^m states: for every word in L there is a distinct state and one rejecting sink state. □

THEOREM 5.3. *Let $n \geq 0$. The size of every DWA representing $[\![\mathrm{Prod}_n]\!]$ is at least* $2^{\left\lfloor \frac{f(n+1)}{2\log_2 \varrho} \right\rfloor}$.

PROOF. First, note that $[\![\mathrm{Prod}_n]\!]$ is the graph of a partial function from $\mathbb{Z}^2$ to $\mathbb{Z}$. Let $m := \left\lfloor \frac{f(n+1)}{2\log_2 \varrho} \right\rfloor$. It holds that $\mathrm{MULT}_m \subseteq [\![\mathrm{Prod}_n]\!]$ since $(\varrho^m - 1)^2 < \varrho^{2m} = 2^{2m\log_2 \varrho} \leq 2^{f(n+1)}$. The claim follows directly from Lemma 5.2. □

Remark 5.4. We make the following remarks on nondeterministic word automata and alternating word automata [Brzozowski and Leiss 1980; Chandra et al. 1981].

- (i) The proof of Theorem 5.3 carries over to nondeterministic word automata. That means, that we obtain the same lower bound for nondeterministic word automata as for DWAs although nondeterministic word automata can sometimes be exponentially more succinct than DWAs.
- (ii) A lower bound for the number of states of alternating word automata for the formula Prod_n is at least $\left\lfloor \frac{f(n+1)}{2\log_2 \varrho} \right\rfloor$. This lower bound follows by contradiction from the remark (i) above and the fact that an alternating word automaton can be translated to an equivalent nondeterministic word automaton with exponentially more states.

6. CONCLUSION

We analyzed the automata-theoretic approach for deciding Presburger arithmetic and established a tight upper bound on the automata size. Furthermore, we improved some of the automata constructions in [Boigelot 1999; Wolper and Boigelot 2000; Ganesh et al. 2002] for linear equations and inequations, proved that our constructions are optimal, and gave lower bounds for the automata for linear equations and inequations.

The main technique to prove the upper bound on the automata size was to relate deterministic word automata with the formulas constructed by a quantifier-elimination method. This technique can also be used to prove upper bounds on the sizes of minimal automata for other logics that admit quantifier elimination and where the structures are automata representable [Khoussainov and Nerode 1995; Blumensath and Grädel 2000; Rubin 2004], i. e., these structures are provided with automata for deciding equality on the domain and the atomic relations of the structure. Prominent examples are the mixed first-order theory over the structure $(\mathbb{R}, \mathbb{Z}, <, +)$ [Boigelot et al. 2005; Weispfenning 1999] and the first-order theory of queues [Rybina and Voronkov 2001; 2003].

REFERENCES

BARDIN, S., FINKEL, A., LEROUX, J., AND PETRUCCI, L. 2003. FAST: Fast accelereation of symbolic transition systems. In *Proc. of the 15th International Conference on Computer Aided Verification (CAV'03)*. Lecture Notes in Computer Science, vol. 2725. 118–121.

BARTZIS, C. AND BULTAN, T. 2003. Efficient symbolic representations for arithmetic constraints in verification. *Int. J. Found. Comput. Sci. 14,* 4, 605–624.

BERMAN, L. 1980. The complexity of logical theories. *Theor. Comput. Sci. 11*, 71–77.

BLUMENSATH, A. AND GRÄDEL, E. 2000. Automatic structures. In *Proc. of the 15th Annual IEEE Symposium on Logic in Computer Science (LICS'00)*. IEEE Computer Society Press, 51–62.

BOIGELOT, B. 1999. Symbolic methods for exploring infinite state spaces. Ph.D. thesis, Faculté des Sciences Appliquées de l'Université de Liège, Liège, Belgium.

BOIGELOT, B., JODOGNE, S., AND WOLPER, P. 2005. An effective decision procedure for linear arithmetic with integer and real variables. *ACM Trans. On Comp. Logic 6,* 3, 614–633.

BOIGELOT, B., RASSART, S., AND WOLPER, P. 1998. On the expressiveness of real and integer arithmetic automata (extended abstract). In *Proc. of the 25th International Colloquium on Automata, Languages and Programming (ICALP'98)*. Lecture Notes in Computer Science, vol. 1443. 152–163.

BOIGELOT, B. AND WOLPER, P. 2002. Representing arithmetic constraints with finite automata: An overview. In *Proc. of the 18th International Conference on Logic Programming (ICLP'02)*. Lecture Notes in Computer Science, vol. 2401. 1–19.

BOUDET, A. AND COMON, H. 1996. Diophantine equations, Presburger arithmetic and finite automata. In *Proc. of the 21st International Colloquium on Trees in Algebra and Programming (CAAP'96)*. Lecture Notes in Computer Science, vol. 1059. 30–43.

BRUYÈRE, V., HANSEL, G., MICHAUX, C., AND VILLEMAIRE, R. 1994. Logic and p-recognizable sets of integers. *Bull. Belg. Math. Soc. 1,* 2, 191–238.

BRZOZOWSKI, J. A. AND LEISS, E. L. 1980. On equations for regular languages, finite automata, and sequential networks. *Theor. Comput. Sci. 10,* 1, 19–35.

BÜCHI, J. 1960. Weak second-order arithmetic and finite automata. *Z. Math. Logik Grundlagen Math. 6*, 66–92.

CHANDRA, A. K., KOZEN, D., AND STOCKMEYER, L. J. 1981. Alternation. *J. ACM 28,* 1, 114–133.

COBHAM, A. 1969. On the base-dependence of sets of numbers recognizable by finite automata. *Math. Syst. Theory 3*, 186–192.

COOPER, D. 1972. Theorem proving in arithmetic without multiplication. *Machine Intelligence 7*, 91–99.

DIXMIER, J. 1990. Proof of a conjecture by Erdös and Graham concerning the problem of Frobenius. *J. Number Theory 34,* 2, 198–209.

FERRANTE, J. AND RACKOFF, C. W. 1975. A decision procedure for the first order theory of real addition with order. *SIAM J. Comput. 4,* 1, 69–76.

FERRANTE, J. AND RACKOFF, C. W. 1979. *The Computational Complexity of Logical Theories.* Lecture Notes in Mathematics, vol. 718. Springer-Verlag.

FISCHER, M. AND RABIN, M. 1974. Super-exponential complexity of Presburger arithmetic. In *Symposium on Applied Mathematics.* SIAM-AMS Proceedings, vol. VII. 27–41.

FISCHER, M. AND RABIN, M. 1998. Super-exponential complexity of Presburger arithmetic. In *Quantifier elimination and cylindrical algebraic decomposition*, B. Caviness and J. Johnson, Eds. Texts and Monographs in Symbolic Computation. Springer-Verlag, 122–135. Reprint of the article [Fischer and Rabin 1974].

GANESH, V., BEREZIN, S., AND DILL, D. L. 2002. Deciding Presburger arithmetic by model checking and comparisons with other methods. In *Proc. of the 4th International Conference on Formal Methods in Computer-Aided Design (FMCAD'02).* Lecture Notes in Computer Science, vol. 2517. 171–186.

GRÄDEL, E. 1988. Subclasses of Presburger arithmetic and the polynomial-time hierarchy. *Theor. Comput. Sci. 56*, 289–301.

Hopcroft, J. E. 1971. An $n \log n$ algorithm for minimizing the states in a finite automaton. In *Theory of Machines and Computations (Proc. of an International Symposium)*, Z. Kohavi and A. Paz, Eds. Academic Press, Technion (Israel Institute of Technology), Haifa, Israel, 189–196.

Hopcroft, J. E. and Ullman, J. D. 1979. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley.

Khoussainov, B. and Nerode, A. 1995. Automatic presentations of structures. In *Proc. of the International Workshop on Logical and Computational Complexity (LCC'94)*. Lecture Notes in Computer Science, vol. 960. 367–392.

LASH. The Liège Automata-based Symbolic Handler. See the web-page `http://www.montefiore.ulg.ac.be/~boigelot/research/lash/`.

Oppen, D. 1978. A $2^{2^{2^{pn}}}$ upper bound on the complexity of Presburger arithmetic. *J. Comput. Syst. Sci. 16*, 323–332.

Presburger, M. 1930. Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. In *Sprawozdanie z I Kongresu metematyków slowiańskich, Warszawa 1929*. 92–101 and 395.

Reddy, C. and Loveland, D. W. 1978. Presburger arithmetic with bounded quantifier alternation. In *Proc. of the 10th Annual ACM Symposium on Theory of Computing (STOC'78)*. ACM Press, 320–325.

Reinhardt, K. 2002. The complexity of translating logic to finite automata. In *Automata, Logics, and Infinite Games*, E. Grädel, W. Thomas, and T. Wilke, Eds. Lecture Notes in Computer Science, vol. 2500. Springer-Verlag, Chapter 13, 231–238.

Rubin, S. 2004. Automatic structures. Ph.D. thesis, University of Auckland, Auckland, New Zealand.

Rybina, T. and Voronkov, A. 2001. A decision procedure for term algebras with queues. *ACM Trans. On Comp. Logic 2,* 2, 155–181.

Rybina, T. and Voronkov, A. 2003. Upper bounds for a theory of queues. In *Proc. of the 30th International Colloquium on Automata, Languages and Programming (ICALP'03)*. Lecture Notes in Computer Science, vol. 2719. 714–724.

Schöning, U. 1997. Complexity of Presburger arithmetic with fixed quantifier dimension. *Theory Comput. Syst. 30,* 4, 423–428.

Semenov, A. 1977. Presburgerness of predicates regular in two number systems. *Sib. Math. J. 18*, 289–300.

Shiple, T. R., Kukula, J. H., and Ranjan, R. K. 1998. A comparison of Presburger engines for EFSM reachability. In *Proc. of the 10th International Conference on Computer Aided Verification (CAV'98)*. Lecture Notes in Computer Science, vol. 1427. 280–292.

Skolem, T. 1931. Über einige Satzfunktionen in der Arithmetik. In *Skrifter utgitt av Det Norske Videnskaps-Akademi i Oslo, I. Matematisk naturvidenskapelig klasse*. Vol. 7. Oslo, 1–28.

Skolem, T. 1970. Über einige Satzfunktionen in der Arithmetik. In *Selected Works in Logic*, J. Fenstad, Ed. Universitetsforlaget, Oslo, 281–306. Reprint of the article [Skolem 1931].

Stansifer, R. 1984. Presburger's article on integer arithmetic: Remarks and translation. Tech. Rep. TR84-639, Department of Computer Science, Cornell University, Ithaca, NY, USA.

Stockmeyer, L. 1974. The complexity of decision problems in automata theory and logic. Ph.D. thesis, Department of Electrical Engineering, MIT, Boston, MA, USA.

Weispfenning, V. 1999. Mixed real-integer linear quantifier elimination. In *Proc. of the International Symposium on Symbolic and Algebraic Computation (ISSAC'99)*. ACM Press, 129–136.

Wolper, P. and Boigelot, B. 1995. An automata-theoretic approach to Presburger arithmetic constraints (extended abstract). In *Proc. of the 2nd International Symposium on Static Analysis (SAS'95)*. Lecture Notes in Computer Science, vol. 983. 21–32.

Wolper, P. and Boigelot, B. 2000. On the construction of automata from linear arithmetic constraints. In *Proc. of the 6th International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS'00)*. Lecture Notes in Computer Science, vol. 1785. 1–19.

YAVUZ-KAHVECI, T., BARTZIS, C., AND BULTAN, T. 2005. Action language verifier, extended. In *Proc. of the 17th International Conference on Computer Aided Verification (CAV'05)*. Lecture Notes in Computer Science, vol. 3576. 413–417.

Received June 2005; revised August 2006; accepted October 2006