A Polynomial Time Presburger Criterion and Synthesis for Number Decision Diagrams*

Jérôme Leroux†
IRISA/INRIA Rennes, Campus de Beaulieu, Rennes, France.
jleroux@irisa.fr

Abstract

Number Decision Diagrams (NDD) are the automata-based symbolic representation for manipulating sets of integer vectors encoded as strings of digit vectors (least or most significant digit first). Since 1969 [8, 29], we know that any Presburger-definable set [26] (a set of integer vectors satisfying a formula in the first-order additive theory of the integers) can be represented by a NDD, and efficient algorithm for manipulating these sets have been recently developed [31, 4]. However, the problem of deciding if a NDD represents such a set, is a well-known hard problem first solved by Muchnik in 1991 [23, 24, 5] with a quadruply-exponential time *algorithm. In this paper, we show how to determine in* polynomial time *whether a NDD represents a Presburger-definable set, and we provide in this positive case a* polynomial time *algorithm that constructs from the NDD a Presburger-formula that defines the same set.*

1. Introduction.

Presburger arithmetic [26] is a decidable logic used in a large range of applications. As described in [17], this logic is central in many areas including integer programming problems [28], compiler optimization techniques [25], program analysis tools [7, 11, 10] and model-checking [1, 9, 16]. Different techniques [12] and tools have been developed for manipulating *the Presburger-definable sets* (the sets of integer vectors satisfying a Presburger formula): by working directly on the Presburger-formulas [14] (implemented in OMEGA [25]), by using semi-linear sets [13] (implemented in BRAIN [27]), or by using NDD (an automaton that represents the sets of integer vectors encoded as strings of digit vectors (least or most significant digit first)) [30, 4] (implemented in FAST [2], LASH [16] and MONA [15]). Presburger-formulas and semi-linear sets lack canonicity. As a direct consequence, a set that possesses a simple representation could unfortunately be represented in an unduly complicated way. Moreover, deciding if a given vector of integers is in a given set, is at least *NP-hard* [3, 13]. On the other hand, a minimization procedure for automata provides a canonical representation for *NDD-definable sets* (a set represented by a NDD). That means, the NDD that represents a given set only depends on this set and not on the way we compute it. For these reasons, NDD are well adapted for applications that require a lot of boolean manipulations such as model-checking.

Whereas there exist efficient algorithms for computing a NDD that represents the set defined by a given Presburger formula [14, 31, 4], the inverse problem of computing a Presburger-formula from a Presburger-definable set represented by a NDD, called the *Presburger synthesis problem*, was first studied in [18] and only *partially solved in exponential time* (resp. *doubly exponential time*) for *convex integer polyhedrons* [17] (resp. for *semi-linear sets with the same set of periods* [22]). Presburger-synthesis has many applications. For example, in software verification, we are interested in computing the set of reachable states of an infinite state system by using NDD representations and in analyzing the structure of these sets with a tool such as [25] which manipulates Presburger-formulas. The Presburger-synthesis problem is also central to a new generation of constraint solvers for Presburger arithmetic that manipulate both NDD and Presburger-formulas [17, 14].

The Presburger-synthesis problem is naturally related to the problem of deciding whether a NDD represents a Presburger-definable set, a well-known hard problem first solved by Muchnik in 1991 [23] with a quadruply exponential time algorithm. To the best of our knowledge no better algorithm for the full class of Presburger-definable sets has been proposed since 1991.

In this paper, we prove that we can decide in *polynomial*

*Research funded by the Faculté des arts et des sciences of the Université de Montréal and by the Natural Sciences and Engineering Research Council of Canada through a discovery grant held by Pierre McKenzie.

†This work was carried out during the author's postdoctoral studies at Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, Montréal, QC Canada

time whether a NDD (least significant digit first) represents a Presburger-definable set. Moreover, for a NDD that represents such a set, we provide an algorithm that computes in *polynomial time* a Presburger-formula that defines the set represented by the NDD. These results rely on a deeper analysis of the structure of a NDD that represents a Presburger-definable set, and on a new geometric point of view on the Presburger-definable sets (whence the length of section 3).

In section 3 we recall some geometrical objects used in the sequel. In section 4, we describe the size of the structures manipulated in this paper for complexity issue. Section 5 contains the definition of NDD and introduces the notion of *detectable sets* that corresponds to sets obtained by modifying the set of final states of a NDD. In section 6, we provide our new geometric point of view of the Presburger-definable sets. Section 7 shows that this geometrical point of view can be "used in polynomial time" from a Presburger-definable NDD. Finally, in section 8, we prove the main results of this paper.

Proofs:
Some proofs had to be omitted due to space constraints. A self-contained long version of this paper (with detailed proofs for all results) can be obtained from the author or as a technical report [21].

2. Preliminaries

Throughout this paper, intersection, union, difference and symmetric difference of two sets A and B are written $A \cap B$, $A \cup B$, $A \backslash B$, and $A \Delta B = (A \backslash B) \cup (B \backslash A)$. We denote by $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ respectively the set of non-negative integers, integers, and rational numbers. The *cardinality* of a finite set X is written $|X| \in \mathbb{N}$. The set of *functions* from a set X to a set Y, also called *sequences* of elements in Y indexed by X is written Y^X. A function $f \in Y^X$ is also denoted by $f : X \to Y$. For such a function and for any $A \subseteq X$ and $B \subseteq Y$, we define $f(A) = \{f(a);\ a \in A\}$ and $f^{-1}(B) = \{x \in X;\ f(x) \in B\}$.

The set X^m is called the set of vectors with $m \in \mathbb{N}$ components in a set X. Given an integer $i \in \{1, \dots, m\}$ and a vector $x \in X^m$, the i-th component of x is written $x[i] \in X$. Vector $\mathbf{e}_j \in \mathbb{Q}^m$ is defined by $\mathbf{e}_j[j] = 1$ and $\mathbf{e}_j[i] = 0$ for any $i \in \{1, \dots, m\} \backslash \{j\}$. Vector $(0, \dots, 0) \in \mathbb{Q}^m$ is denoted by 0. Vectors $x + y$ and $t.x$ are defined by $(x + y)[i] = (x[i]) + (y[i])$ and $(t.x)[i] = t.(x[i])$ for any $i \in \{1, \dots, m\}$, $x, y \in \mathbb{Q}^m$, $t \in \mathbb{Q}$. For any $x, y \in \mathbb{Q}^m$, let $\langle x, y\rangle = \sum_{i=1}^{m} x[i].y[i]$ be the *dot product*. For any subset $X \subseteq \mathbb{Q}^m$, we denote by $X^\perp = \{y \in \mathbb{Q}^m;\ \forall x \in X\ \langle x, y\rangle = 0\}$. For any $x \in \mathbb{Q}^m$, let us consider the norm $||x||_\infty = \max_i |x[i]|$ where $|x[i]|$ is the absolute value of $x[i]$. We naturally define $A + B = \{a + b;\ (a, b) \in A \times B\}$ and $T.A = \{t.a;\ (t, a) \in T \times A\}$ for any $A, B \subseteq \mathbb{Q}^m$ and $T \subseteq \mathbb{Q}$. For any $a, b \in \mathbb{Q}^m$ and $t \in \mathbb{Q}$, let us define $a + B = \{a\} + B$, $A + b = A + \{b\}$, $t.A = \{t\}.A$ and $T.a = T.\{a\}$.

The set of words over a non-empty finite alphabet Σ is written Σ^*. The *length* of a word σ is written $|\sigma| \in \mathbb{N}$. The word of length 0 is written ϵ and we denote by Σ^+ the set $\Sigma^+ = \Sigma^* \backslash \{\epsilon\}$. The concatenation of two words σ and σ' in Σ^* is written $\sigma.\sigma'$. Such a word σ is called a *prefix* of $\sigma.\sigma'$ (respectively a *strict prefix* if $\sigma' \neq \epsilon$).

A *deterministic and complete automaton* $\mathcal{A}$ is a tuple $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$; Q is the finite set of states, Σ is the finite alphabet, $\delta : Q \times \Sigma \to Q$ is the transition function, $q_0 \in Q$ is the initial state and $F \subseteq Q$ is the set of final states. The *Cartesian product* $\mathcal{A}_1 \times_F \mathcal{A}_2$ of two automata $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{0,1}, F_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma, \delta_2, q_{0,2}, F_2)$, where $F \subseteq Q_1 \times Q_2$, is the deterministic and complete automaton $\mathcal{A}_1 \times_F \mathcal{A}_2 = (Q, \Sigma, \delta, q_0, F)$ defined by $Q = Q_1 \times Q_2$, $\delta((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$, and $q_0 = (q_{0,1}, q_{0,2})$. As usual, we extend δ over $Q \times \Sigma^*$ such that $\delta(q, \sigma.\sigma') = \delta(\delta(q, \sigma), \sigma')$. The language $\mathcal{L}(\mathcal{A}) \subseteq \Sigma^*$ accepted by a deterministic and complete automaton $\mathcal{A}$ is defined by $\mathcal{L}(\mathcal{A}) = \{\sigma \in \Sigma^*;\ \delta(q_0, \sigma) \in F\}$. A tuple (q, σ, q') such that $\delta(q, \sigma) = q'$ is called a *path from* q *to* q' *labeled by* σ and it is written $q \xrightarrow{\sigma} q'$ or just $q \to q'$. In this case, q' is said reachable from q. A subset $Q' \subseteq Q$ is said *reachable* from a subset $Q_0 \subseteq Q$ if there exists a path from a state in Q_0 to a state in Q'. A *strongly connected component* Q' of an automaton $\mathcal{A}$ is an equivalence class for the equivalence relation $\rightleftarrows$ defined over Q by $q \rightleftarrows q'$ if and only if $q \to q'$ and $q' \to q$.

3. Geometric sets

In this paper, we use a large range of geometric sets. Section 3.1 recalls the notion of *integral dimension*. The *vector space definition* is given in the next section 3.2. Section 3.3 recalls some properties satisfied by finite unions of *affine spaces*, called *semi-affine spaces* [20]. Section 3.4 gives the definition of a *patterns* and a *modular spaces*, and the final one provides the definition of *polyhedrons* and *boundaries*.

3.1. Integral dimension and V-degenerate subsets

Given a subset $X \subseteq \mathbb{Z}^m$, there exists a minimal integer $\dim(X) \in \{-1, \dots, m\}$ (for $\leq$), called the *integral dimension*, satisfying the following inequality:

$$\sup_{n \in \mathbb{N} \backslash \{0\}} \left(\frac{|\{x \in X;\ ||x||_\infty \leq n\}|}{n^{\dim(X)}} \right) < +\infty$$

Remark 3.1 *Let $X \subseteq \mathbb{Z}^m$. We have $\dim(X) = -1$ if and only if X is empty, $\dim(X) = 0$ if and only if X is a non-empty finite set, and $\dim(X) \geq 1$ if and only if X is infinite.*

Without ambiguity, given a subset $X \subseteq \mathbb{Q}^m$, the integer $\dim(\mathbb{Z}^m \cap X)$ is also denoted by $\dim(X)$, and called the *integral dimension* of X.

A subset $X \subseteq V$ where $V \subseteq \mathbb{Q}^m$ is said *V-degenerate* if $\dim(X) < \dim(V)$. Let us consider the equivalence relation $\sim_V$ defined over the subsets of V by $X_1 \sim_V X_2$ if and only if $X_1 \Delta X_2$ is V-degenerate. The equivalence class $[X]_V$ of a subset $X \subseteq V$ satisfies $[X]_V = \{X' \subseteq V;\ \dim(X \Delta X') < \dim(V)\}$.

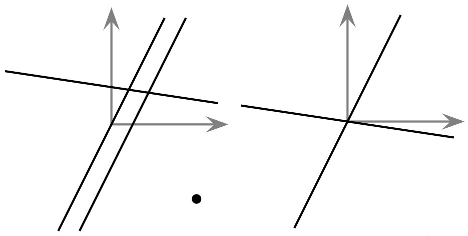
3.2. Vector spaces

Recall that a *vector space* V of $\mathbb{Q}^m$ is a subset of $\mathbb{Q}^m$ such that there exists a finite subset $V_0 \subseteq \mathbb{Q}^m$ satisfying $V = \sum_{v_0 \in V_0} \mathbb{Q}.v_0$ (when $V_0 = \emptyset$ then $V = \{0\}$). Such a vector space V is said *generated by* V_0. The *dimension* of a vector space V is defined as the minimal integer $n \in \mathbb{N}$ (for $\leq$) such that there exists a finite subset V_0 of n vectors in $\mathbb{Q}^m$ that generates V.

Lemma 3.2 *For any vector space, the integral dimension and the dimension are equal.*

3.3. Semi-affine spaces

An *affine space* A of $\mathbb{Q}^m$ is either the empty set or a set of the form $A = a + V$ where $a \in \mathbb{Q}^m$ and V is a vector space of $\mathbb{Q}^m$. In this case the vector space V is unique, denoted by $\overrightarrow{A}$ and called the *direction* of A. A finite union of affine spaces $S = \bigcup_{A \in \mathcal{C}} A$ is called a *semi-affine space* [20] (see figure 1 for an example).



A semi-affine space S Its direction $\overrightarrow{S}$

Figure 1. Direction of a semi-affine space

Recall that a finite or infinite intersection of affine spaces remains an affine space. In particular, for any subset $X \subseteq \mathbb{Q}^m$, there exists a unique minimal (for $\subseteq$) affine space $\mathrm{aff}(X)$ that contains X, called the *affine hull of* X. As proved by lemma 3.3, a finite or infinite intersection of semi-affine spaces remains a semi-affine space. Hence, there also exists a unique minimal (for $\subseteq$) semi-affine space $\mathrm{saff}(X)$ that contains X, called the *semi-affine hull of* X.

Lemma 3.3 ([20]) *The class of semi-affine spaces is stable by any infinite intersection.*

Example 3.4 *The semi-affine hull of a* finite *subset $X \subseteq \mathbb{Q}^m$ is equal to X because X is the finite union over $x \in X$ of the affine space $\{x\} = x + \{0\}$. The semi-affine hull of an* infinite *subset $X \subseteq \mathbb{Q}$ is equal to $\mathbb{Q}$ (remark that $m = 1$). In fact, the class of affine spaces of $\mathbb{Q}$ is equal to $\{\mathbb{Q}, \emptyset\} \cup \{\{x\};\ x \in \mathbb{Q}\}$.*

Example 3.5 *As $\mathrm{aff}(X)$ is an affine space and in particular a semi-affine space that contains X, we deduce that $\mathrm{saff}(X) \subseteq \mathrm{aff}(X)$. This last inclusion can be strict as shown by the example $X = \{(0,0), (1,0), (0,1)\}$. In fact, in this case, we have $\mathrm{saff}(X) = X$ and $\mathrm{aff}(X) = \mathbb{Q}^2$.*

A maximal (for $\subseteq$) non-empty affine space $A \subseteq S$, is called an *affine component* of S. The set of affine components of S is written $\mathrm{comp}(S)$. As proved by the following proposition 3.6, a semi-affine space can be canonically represented by its set of affine components. This is an important property for *implementation issues* of a *semi-affine library*.

Proposition 3.6 ([20]) *The set of affine components $\mathrm{comp}(S)$ of a semi-affine space S is finite and S is equal to the finite union of its affine components $S = \bigcup_{A \in \mathrm{comp}(S)} A$.*

The *direction* $\overrightarrow{S}$ of a semi-affine space S is defined by $\overrightarrow{S} = \bigcup_{A \in \mathrm{comp}(S)} \overrightarrow{A}$. Remark that the semi-affine space direction definition extends the affine spaces direction definition because if S is a non-empty affine space then $\mathrm{comp}(S) = \{S\}$.

Example 3.7 *Let us consider the semi-affine space $S = A_1 \cup A_2 \cup A_3 \cup A_4$ where $A_1 = \mathbb{Q}.(1,2)$, $A_2 = (2,0) + \mathbb{Q}.(1,2)$, $A_3 = (0,-3.5) + \mathbb{Q}.(20,-3)$ and $A_4 = \{(8,-7)\}$ given in figure 1. We have $\overrightarrow{S} = V_1 \cup V_3$ where $V_1 = \mathbb{Q}.(1,2)$ and $V_3 = \mathbb{Q}.(20,-3)$. Remark that S owns 4 affine components $\mathrm{comp}(S) = \{A_1, A_2, A_3, A_4\}$, the set $\{\overrightarrow{A};\ A \in \mathrm{comp}(S)\} = \{V_1, V_3, \{0\}\}$ owns 3 vector spaces and $\overrightarrow{S}$ owns only 2 affine components $\mathrm{comp}(\overrightarrow{S}) = \{V_1, V_3\}$. In fact, in general, we have $\mathrm{comp}(\overrightarrow{S}) \subseteq \{\overrightarrow{A};\ A \in \mathrm{comp}(S)\}$ for any semi-affine space S.*

Following lemma proves that if S is equal to $S = \bigcup_{A \in \mathcal{C}} A$ where $\mathcal{C}$ is a finite class of affine spaces not necessarily equal to $\mathrm{comp}(S)$, then $\overrightarrow{S} = \bigcup_{A \in \mathcal{C}} \overrightarrow{A}$.

Lemma 3.8 *For any finite class $\mathcal{C}$ of affine spaces, the direction of the semi-affine space $S = \bigcup_{A \in \mathcal{C}} A$ is equal to $\overrightarrow{S} = \bigcup_{A \in \mathcal{C}} \overrightarrow{A}$.*

The semi-affine space $\overrightarrow{\mathrm{saff}(X)}$, is written $\overrightarrow{\mathrm{saff}}(X)$.

3.4. Patterns and V-modular spaces

A *pattern* P of $\mathbb{Z}^m$ is a subset of $\mathbb{Z}^m$ such that there exists $n \in \mathbb{N}\backslash\{0\}$ and a subset $B \subseteq \mathbb{Z}^m$ such that $P = B + n.\mathbb{Z}^m$ (see figure 2 and example 3.9). Intuitively, a pattern is a subset of $\mathbb{Z}^m$ obtained from a "motif B repeated in all directions". Remark that a subset $P \subseteq \mathbb{Z}^m$ is a pattern if and only if there exists $n \in \mathbb{N}\backslash\{0\}$ such that $P = P + n.\mathbb{Z}^m$, and in this case $P = B + n.\mathbb{Z}^m$ where $B = P \cap \{0, \ldots, n-1\}^m$.

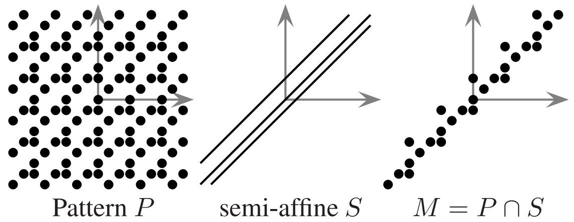


Figure 2. A $\mathbb{Q}.(1,1)$-modular space M

A *V-modular space M*, where V is a vector space, is a subset of the form $M = P \cap S$ where P is a pattern and S is a semi-affine space obtained as a finite union of affine spaces A satisfying $\overrightarrow{A} = V$.

Example 3.9 *Let us consider the pattern $P = B + n.\mathbb{Z}^2$ where $n = 3$ and $B = \{(0,0),(1,1),(2,2),(0,2)\}$, the vector space $V = \mathbb{Q}.(1,1)$, the semi-affine space $S = V \cup ((0,2)+V) \cup ((0,-1)+V)$ and the V-modular space $M = P \cap S$. Sets P, S and M are given in figure 2.*

3.5. V-polyhedrons

Here, we define *V-polyhedrons*, we characterize *V-degenerate V-polyhedrons*, and we introduce the notion of *V-boundary* and *possible V-boundary* of a V-polyhedron.

The *V-half space* $H_{V,\alpha,\#,c}$ where V is a vector space, $\alpha \in V\backslash\{0\}$, $\# \in \{\leq, <, >, \geq\}$ and $c \in \mathbb{Q}$ is defined by $H_{V,\alpha,\#,c} = \{x \in V;\ \langle\alpha, x\rangle \#c\}$. We also denote by $H_{V,\alpha,=,c}$, the affine space $H_{V,\alpha,=,c} = \{x \in V;\ \langle\alpha, x\rangle = c\}$ (even equal to the vector space $V \cap (\mathbb{Q}.\alpha)^\perp$ if $c = 0$).

A *V-polyhedron* C is a subset of V defined as a boolean combination of V-half spaces $H_{V,\alpha,\#,c}$ (see figure 3 and example 3.10). A *possible boundary* S of such a V-polyhedron C is the semi-affine space $S = \bigcup_{\alpha\in D} V \cap (\mathbb{Q}.\alpha)^\perp$ where D is the finite set of $\alpha \in V\backslash\{0\}$ used to define C as a boolean combination of V-half spaces $H_{V,\alpha,\#,c}$.

Example 3.10 *Let us consider the vector space $V = \mathbb{Q}^2$, and the V-polyhedrons C_0, C_1 and C_2 given in figure 3 and*

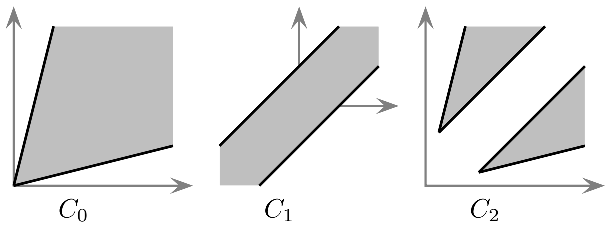


Figure 3. $\mathbb{Q}^2$-polyhedrons

defined by $C_0 = \{x \in \mathbb{Q}^2;\ (x[1] \leq 4.x[2]) \wedge (x[2] \leq 4.x[1])\}$, $C_1 = \{x \in \mathbb{Q}^2;\ -1 \leq x[1] - x[2] \leq 1\}$, and $C_2 = C_0\backslash C_1$. Remark that C_0 and C_2 are non V-degenerate, and C_1 is V-degenerate. Moreover, $S_0 = \{x \in \mathbb{Q}^2;\ (x[1] = 4.x[2]) \vee (x[2] = 4.x[1])\}$ is a possible V-boundary of C_0, $S_1 = \{x \in \mathbb{Q}^2;\ x[1] = x[2]\}$ is a possible V-boundary of C_1, and $S_2 = S_0 \cup S_1$ is a possible V-boundary of C_2.

Remark 3.11 *A V-polyhedron is equal to a finite union of* convex V-polyhedrons $\bigcap_{\alpha\in D} H_{V,\alpha,\#_\alpha,c_\alpha}$, *where D is a finite subset of $V\backslash\{0\}$, $(\#_\alpha)_{\alpha\in D}$ is a sequence in $\{\leq, <, >, \geq\}^D$ and $(c_\alpha)_{\alpha\in D}$ is a sequence in $\mathbb{Q}^D$.*

The following proposition 3.12 provides a geometrical characterization of *V-degenerate V-polyhedrons*.

Proposition 3.12 *A V-polyhedron C is V-degenerate if and only if there exists a finite subset $D \subseteq V\backslash\{0\}$ such that $C \subseteq \bigcup_{\alpha\in D}\{x \in V;\ -1 \leq \langle\alpha, x\rangle \leq 1\}$.*

The following lemma 3.13 shows that any V-polyhedron C owns a minimal (for $\subseteq$) possible V-boundary (up to V-degenerate sets) called the *V- boundary* of C and written $\mathrm{bound}_V(C)$.

Lemma 3.13 *Let C be a V-polyhedron. There exists a unique minimal (for $\subseteq$) semi-affine space $\mathrm{bound}_V(C)$ called the V-boundary of C such that $\mathrm{bound}_V(C)$ is a possible V-boundary of a V-polyhedron in $[C]_V$ (see section 3.1).*

Example 3.14 *Let us consider the V-polyhedrons C_0, C_1 and C_2 defined in example 3.10, and given in figure 3. We have $\mathrm{bound}_V(C_1) = \emptyset$ and $\mathrm{bound}_V(C_0) = \mathrm{bound}_V(C_2) = \{x \in \mathbb{Q}^2;\ (x[1] = 4.x[2]) \vee (x[2] = 4.x[1])\}$. Remark in particular that $\mathrm{bound}_V(C_2)$ is* not a possible V-boundary of C_2.

4. Size and complexity

This section provides the size of the manipulated structures in this paper.

Naturally, the size of a rational number $x = \frac{n}{d}$ where n and $d \in \mathbb{N}\backslash\{0\}$ are relatively prime, a vector $v \in \mathbb{Q}^m$, a matrix $M \in \mathcal{M}_{m,n}(\mathbb{Q})$, a word $\sigma \in \Sigma^*$ are defined by $\text{size}(x) = \ln(1+|n|)+\ln(1+d)$, $\text{size}(v) = \sum_{i=1}^{m} \text{size}(v[i])$, $\text{size}(M) = \sum_{i=1}^{m}\sum_{j=1}^{n} \text{size}(M_{ij})$, $\text{size}(\sigma) = |\sigma|.\ln(1 + |\Sigma|)$.The size of an affine space A implicitly generated by a finite set $A_0 \subseteq \mathbb{Q}^m$ is defined by $\text{size}(A) = \sum_{a_0\in A_0} \text{size}(a_0)$. The size $\text{size}(S)$ of a semi-affine space S is given by $\text{size}(S) = \sum_{A\in\text{comp}(S)} \text{size}(A)$. The size of a finite set $\mathcal{C}$ of rational numbers, vectors, matrices, and so on, is given by $\text{size}(\mathcal{C}) = \sum_{Y\in\mathcal{C}} \text{size}(Y)$.

Recall that almost all the natural operations over affine spaces can be done in polynomial time (in the dimension $m \geq 1$).

The size of a deterministic and complete automaton $\mathcal{A}$ over an alphabet Σ is given by $\text{size}(\mathcal{A}) = |Q|.|\Sigma|$.

5. NDD and r-definable sets

Sets of integer vectors that can be represented by automata, called *Number decision Diagram (NDD)*, thanks to a *least or most significant digit first* decomposition, are related to the notion of r-definable [5] where $r \geq 2$ is an integer called *the basis of decomposition*. In this section this notion is recalled. Moreover, in section 5.1 and 5.2, the sets obtained by *modifying* respectively, the *initial state* and the *set of final states* of a NDD, are characterized. In the last section 5.3, we introduce the notion of *terminal components*, some particular strongly connected components of a NDD.

Given an integer $r \geq 2$, a subset $X \subseteq \mathbb{N}^m$, where $m \geq 1$ is called the *dimension*, is said r-definable if it is definable in the first order logic $\langle \mathbb{Z}, +, \leq, V_r\rangle$ where $V_r : \mathbb{Z} \to \mathbb{Z}$ is the *valuation function* defined by $V_r(0) = 1$ and $V_r(x)$ is the greatest power of r that divides $x \in \mathbb{Z}\backslash\{0\}$.

Example 5.1 *Let $r = 2$. For any $k_1, k_2 \in \mathbb{N}$, valuation $V_2(2^{k_1} + 2^{k_2})$ is equal to 2^k where $k = \min\{k_1, k_2\}$ if $k_1 \neq k_2$ and $k = k_1 + 1 = k_2 + 1$ otherwise.*

Recall that the first order logic $\langle \mathbb{Z}, +, \leq, V_r\rangle$ is *decidable*. The proof of this well known result is based on the decomposition of an integer vector into a *least or most significant word of digit vectors* over the alphabet $\Sigma_{r^m} = \{0, \dots, r-1\}^m$. Following notations introduced in [19], this *decomposition* can be provided thanks to the following function γ_b, where $b \in \Sigma_{r^m}$.

$$\gamma_b : \begin{array}{ccc} \mathbb{Z}^m & \longrightarrow & \mathbb{Z}^m \\ x & \longrightarrow & r.x + b \end{array}$$

Given a sequence b_1, ..., b_k of $k \geq 1$ digit vectors in Σ_{r^m}, we have the following equality also called the *least significant digit first decomposition*:

$$\gamma_{b_1} \circ \dots \circ \gamma_{b_k}(\{(0, \dots, 0)\}) = \sum_{i=1}^{k} r^{i-1} b_i$$

Hence, the vector $\rho(\sigma) = \gamma_\sigma(\{0, \dots, 0\}) \in \mathbb{N}^m$ can be naturally associated to the word $\sigma = b_1 \dots b_k$, where γ_σ is the function defined by the following equality (function γ_ϵ is equal to the identity function $\gamma_\epsilon(x) = x$):

$$\gamma_{b_1\dots b_k} = \gamma_{b_1} \circ \dots \circ \gamma_{b_k}$$

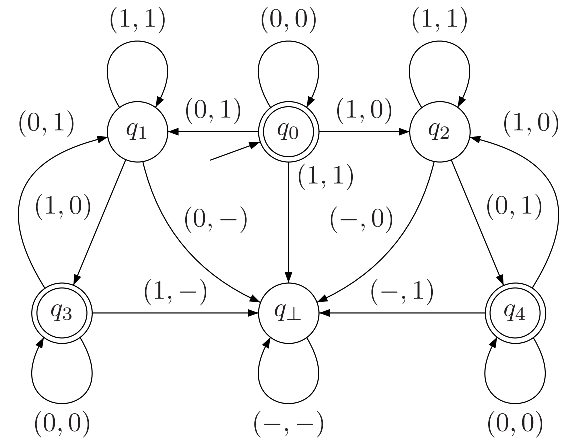


Figure 4. A NDD representing $\{x \in \mathbb{N}^2;\ (x[1] = 2.x[2]) \vee (x[2] = 2.x[1])\}$ **where** $- \in \{0, 1\}$

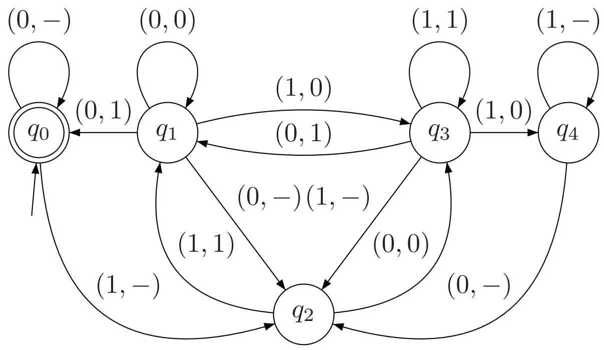


Figure 5. A NDD representing $\{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$ **where** $- \in \{0, 1\}$

Definition 5.2 ([31, 4]) *A* (least significant digit first) Number Decision Diagram (NDD) *is a deterministic and complete automaton over Σ_{r^m} such that for any state $q \in Q$, we have $q \in F$ if and only if $\delta(q, 0) \in F$.*

The set $X = \rho(\mathcal{L}(\mathcal{A}))$ is called the set represented by $\mathcal{A}$ and such a set is said *NDD-definable* (see figure 4 or 5 for an example of NDD). Recall that a set X is NDD-definable if and only if it is r-definable [5].

Remark 5.3 *There exists some deterministic and complete automata $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}_1)\cap\mathcal{L}(\mathcal{A}_2) = \mathcal{L}(\mathcal{A})$, but $X_1\cap X_2 \neq X$ where $X_1 = \rho(\mathcal{L}(\mathcal{A}_1))$, $X_2 = \rho(\mathcal{L}(\mathcal{A}_2))$ and $X = \rho(\mathcal{L}(\mathcal{A}))$. This side effect is no longer true for NDD thanks to the condition $q \in F$ if and only if $\delta(q, 0) \in F$ for every $q \in Q$. In fact, given two NDD $\mathcal{A}_1 = (Q_1, \Sigma_{r^m}, \delta_1, q_{0,1}, F_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_{r^m}, \delta_2, q_{0,2}, F_2)$ representing respectively two sets X_1 and X_2, the Cartesian product $\mathcal{A}_1 \times_{F_\#} \mathcal{A}_2$ where $\# \in \{\cup, \cap, \Delta\}$, $F_\cup = (Q_1 \times F_2) \cup (F_1 \times Q_2)$, $F_\cap = F_1 \times F_2$, and $F_\Delta = (F_1 \times (Q_2\backslash F_2)) \cup ((Q_1\backslash F_1) \times F_2)$, is a NDD representing $X_1\#X_2$.*

Remark 5.4 *A* (most significant digit first) Number Decision Diagram (NDD) *that represents a set $X \subseteq \mathbb{N}^m$ is a deterministic and complete automaton $\mathcal{A}$ over Σ_{r^m} that recognizes the mirror of $\rho^{-1}(X)$ and defined as $\mathcal{L}(\mathcal{A}) = \{a_1 \dots a_n;\ a_n \dots a_1 \in \rho^{-1}(X)\}$.*

5.1. Moving the initial state

The set represented by the NDD $\mathcal{A}_q$ when the initial state q_0 of a NDD $\mathcal{A}$ is replaced by another state $q \in Q$, can be easily characterized thanks to the function γ_σ.

Proposition 5.5 *Let $\mathcal{A}$ be a NDD that represents a set X. For any path $q_0 \xrightarrow{\sigma} q$, the NDD $\mathcal{A}_q$ represents $\gamma_\sigma^{-1}(X)$.*

When a set $X \subseteq \mathbb{N}^m$ is implicitly represented by a NDD $\mathcal{A}$ (not necessary minimal), we denote by X_q the set represented by the NDD $\mathcal{A}_q$. Proposition 5.5 shows that for any path $q \xrightarrow{\sigma} q'$, we have $X_{q'} = \gamma_\sigma^{-1}(X_q)$.

Example 5.6 *Let us consider the NDD $\mathcal{A}$ presented in figure 4 that represents the set $X = \{x \in \mathbb{N}^2;\ (x[1] = 2.x[2]) \vee (x[2] = 2.x[1])\}$. We have $X_{q_0} = X$, $X_{q_\perp} = \emptyset$, $X_{q_1} = \{x \in \mathbb{N}^2; x[1] = 2.x[2] + 1\}$, $X_{q_2} = \{x \in \mathbb{N}^2; x[2] = 2.x[1] + 1\}$, $X_{q_3} = \{x \in \mathbb{N}^2; x[1] = 2.x[2]\}$, $X_{q_4} = \{x \in \mathbb{N}^2; x[2] = 2.x[1]\}$.*

Example 5.7 *Let us consider the NDD $\mathcal{A}$ presented in figure 5 that represents the set $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$. For any $c \in \{0, 1, 2, 3, 4\}$, we have $X_{q_c} = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1] + c\}$.*

5.2. Replacing the set of final states

In order to characterize the set represented by the NDD $\mathcal{A}^{F'}$ when the set of final states F of a NDD $\mathcal{A}$ is replaced by another set of states $F' \subseteq Q$, we introduce the notion of *semi-eyes* and *detectable* sets.

Let $\mathcal{A}$ be a NDD. We consider the binary relation $\sim$ over Q, defined by $q \sim q'$ if and only if $\delta(q, 0^*) \cap \delta(q', 0^*) \neq \emptyset$. As $\mathcal{A}$ is deterministic, $\sim$ is an equivalence relation. An equivalence class for this relation is called *an eye*. A finite union of eyes is called a *semi-eye*. Naturally, for any subset $F' \subseteq Q$, the automaton $\mathcal{A}^{F'}$ is a NDD if and only if F' is a semi-eye.

Example 5.8 *Let $\mathcal{A}$ be the NDD given in figure 4. The set of states Q can be partitioned into 4 eyes Y_1, Y_2, Y_3, Y_4 where $Y_1 = \{q_1, q_2, q_\perp\}$, $Y_2 = \{q_0\}$, $Y_3 = \{q_3\}$ and $Y_4 = \{q_4\}$.*

Example 5.9 *Let $\mathcal{A}$ be the NDD given in figure 5. The set of states Q can be partitioned into 3 eyes Y_1, Y_2, Y_3 where $Y_1 = \{q_0\}$ and $Y_2 = \{q_1\}$ and $Y_3 = \{q_2, q_3, q_4\}$.*

Let $\mathcal{A}$ be a NDD and remark that for any $X' \subseteq \mathbb{N}^m$, there exists a unique minimal (for $\subseteq$) semi-eye $F_\mathcal{A}(X')$ such that X' is included in the set represented by $\mathcal{A}^{F_\mathcal{A}(X')}$. In general, this inclusion is strict. However, for *detectable sets*, it becomes an equality.

A set $X' \subseteq \mathbb{Z}^m$ is said *detectable in a set $X \subseteq \mathbb{Z}^m$* if for any pair of words (σ_1, σ_2) such that $\gamma_{\sigma_1}^{-1}(X) = \gamma_{\sigma_2}^{-1}(X)$, we have $\gamma_{\sigma_1}^{-1}(X') = \gamma_{\sigma_2}^{-1}(X')$.

Remark 5.10 *When X and X' are respectively represented by two* minimal NDD *$\mathcal{A} = (Q, \Sigma_{r^m}, \delta, q_0, F)$ and $\mathcal{A}' = (Q', \Sigma_{r^m}, \delta', q'_0, F')$, we proved in [21] that X' is detectable in X if and only if for any pair of words (σ_1, σ_2) such that $\delta(q_0, \sigma_1) = \delta(q_0, \sigma_2)$, we have $\delta'(q'_0, \sigma_1) = \delta'(q'_0, \sigma_2)$.*

Proposition 5.11 *Let $X \subseteq \mathbb{N}^m$ be represented by a NDD $\mathcal{A}$. For any set $X' \subseteq \mathbb{N}^m$ detectable in X, the NDD $\mathcal{A}^{F_\mathcal{A}(X')}$ represents X'.*

The following proposition 5.12 will be useful to compute in polynomial time the set $F_\mathcal{A}(X')$ of a set $X' \subseteq \mathbb{N}^m$ with a *polynomial time membership problem*, and detectable in a set $X \subseteq \mathbb{N}^m$ represented by a NDD $\mathcal{A}$.

Proposition 5.12 *Let $X \subseteq \mathbb{N}^m$ represented by a NDD $\mathcal{A}$ and let $X' \subseteq \mathbb{N}^m$ detectable in X. In polynomial time, the computation of $F_\mathcal{A}(X')$ can be reduced to the membership problem for X'.*

Example 5.13 *Let $X = \{x \in \mathbb{N}^2;\ (x[1] = 2.x[2]) \vee (x[2] = 2.x[1])\}$ be represented by the NDD $\mathcal{A}$ given in figure 4. The sets $X_1 = \{x \in \mathbb{N}^2;\ (x[1] = 2.x[2])\}$ and $X_2 = \{x \in \mathbb{N}^2;\ (x[2] = 2.x[1])\}$ are both detectable in X. We have $F_\mathcal{A}(X_1) = \{q_3, q_0\}$ and $F_\mathcal{A}(X_2) = \{q_4, q_0\}$.*

5.3. Terminal components

The *strongly connected components* of a NDD play an important role in this paper. We call a *terminal component* T of a NDD $\mathcal{A}$, a strongly connected component reachable from the initial state, that contains at least one final state and such that any final state q' reachable from T is in T. Intuitively, a strongly connected component T is terminal if it is farthest from the initial state. The *set of terminal components* of $\mathcal{A}$ is denoted by $T(\mathcal{A})$.

Proposition 5.14 *Let $\mathcal{A}$ be a NDD. For any terminal component T of $\mathcal{A}$, there exists a unique vector space $V_T(\mathcal{A})$ such that* $\mathrm{saff}(\rho(\mathcal{L}(\mathcal{A}_q^{F'})))$ *is an affine space whose the direction is equal to $V_T(\mathcal{A})$, for any state $q \in T$ and for any semi-eye F' such that T remains a terminal component of $\mathcal{A}^{F'}$.*

Proposition 5.14 associates to any terminal component T of a NDD $\mathcal{A}$, a vector space $V_T(\mathcal{A})$. Moreover, as for any $q \in T \cap F$, we have $\mathrm{aff}(\rho(\mathcal{L}(\mathcal{A}_q))) = V_T(\mathcal{A})$, we deduce from [19] that we can compute in polynomial time this vector space. For any vector space V, we denote by $T_V(\mathcal{A})$ the set of terminal components $T \in T(\mathcal{A})$ such that $V_T(\mathcal{A}) = V$.

Example 5.15 *Let us consider the NDD $\mathcal{A}$ given in figure 4 that represents $X = \{x \in \mathbb{N}^2;\ (x[1] = 2.x[2]) \vee (x[2] = 2.x[1])\}$. The set $T(\mathcal{A})$ contains two terminal components $T(\mathcal{A}) = \{T_0, T_1\}$ where $T_0 = \{q_2, q_4\}$ and $T_1 = \{q_1, q_3\}$. Moreover, we have $V_{T_0}(\mathcal{A}) = \mathbb{Q}.(1,2)$ and $V_{T_1}(\mathcal{A}) = \mathbb{Q}.(2,1)$. In particular we have $\overrightarrow{\mathrm{saff}}(X) = \bigcup_{T\in T(\mathcal{A})} V_T(\mathcal{A})$.*

Example 5.16 *Let us consider the NDD $\mathcal{A}$ given in figure 5 that represents $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$. The set $T(\mathcal{A})$ contains one terminal components $T(\mathcal{A}) = \{Q\}$. Moreover, we have $V_Q(\mathcal{A}) = \mathbb{Q}^2$. In particular, we also have $\overrightarrow{\mathrm{saff}}(X) = \bigcup_{T\in T(\mathcal{A})} V_T(\mathcal{A})$.*

6. Presburger-definable sets

A subset $X \subseteq \mathbb{Z}^m$ is said Presburger-definable if it can be defined by a formula in the first order theory $\langle \mathbb{Z}, +, \leq \rangle$. Naturally, any Presburger-definable set is r-definable and there exists some r-definable sets that are not Presburger-definable. In this section, we provide a "decomposition theorem" for the Presburger-definable sets that provides a new geometrical point of view of Presburger-definable sets.

Remark 6.1 *A* linear set *X of $\mathbb{Z}^m$ is a set of the from $X = b + \sum_{p\in P} \mathbb{N}.p$ where $b \in \mathbb{Z}^m$ is called the* basis *and $P \subseteq \mathbb{Z}^m$ is a finite subset of $\mathbb{Z}^m$ called the set of* periods*. A* semi-linear set *of $\mathbb{Z}^m$ is a finite union of linear sets of $\mathbb{Z}^m$. Recall that a set X is Presburger-definable if and only if it is* semi-linear *[13].*

Example 6.2 *The Presburger-definable set $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$ is represented in figure 6.*

Given a vector space V and a subset $X \subseteq \mathbb{Q}^m$, let us consider the following set $X_V \subseteq X$:

$$X_V = X \cap \left(\bigcup_{\substack{A\in\mathrm{comp}(\mathrm{saff}(X)) \\ \overrightarrow{A}=V}} A \right)$$

As X_V is non-empty if and only if V is in the finite set $\{\overrightarrow{A};\ A \in \mathrm{comp}(\overrightarrow{\mathrm{saff}}(X))\}$, we have a decomposition of X into a finite union of X_V. In [21], we proved that for any vector space V, the set $\mathrm{comp}(\mathrm{saff}(X_V))$ is a finite union of non-empty affine spaces A such that $\overrightarrow{A} = V$.

This decomposition of X is refined by the following theorem 6.3 when $X \subseteq \mathbb{Z}^m$ is Presburger-definable and V is an affine component of $\overrightarrow{\mathrm{saff}}(X)$ (see example 3.7 for the inclusion $\mathrm{comp}(\overrightarrow{\mathrm{saff}}(X)) \subseteq \{\overrightarrow{A};\ A \in \mathrm{comp}(\overrightarrow{\mathrm{saff}}(X))\}$).

Theorem 6.3 (decomposition theorem) *Let $X \subseteq \mathbb{Z}^m$ be a Presburger-definable set and $V \in \mathrm{comp}(\overrightarrow{\mathrm{saff}}(X))$. There exists a unique finite class $\mathcal{M}_V(X)$ of V-modular spaces such that there exists a sequence $(C_{V,M})_{M\in\mathcal{M}_V(X)}$ of V-polyhedrons satisfying the following two assertions:*

- *Sequence $(C_{V,M})_{M\in\mathcal{M}_V(X)}$ is a "kind of partition" of V: $C_{V,M}$ is non V-degenerate for any $M \in \mathcal{M}_V(X)$ whereas $V \backslash (\bigcup_{M\in\mathcal{M}_V(X)} C_{V,M})$ is V-degenerate and $C_{V,M} \cap C_{V,M'}$ is V-degenerate for any $M \neq M' \in \mathcal{M}_V(X)$.*
- *For any V-modular space $M \in \mathcal{M}_V(X)$, we have:*
$$\dim\left((X_V \Delta M) \cap (C_{V,M} + V^{\perp})\right) < \dim(X_V)$$

Moreover another sequence $(C'_{V,M})_{M\in\mathcal{M}_V(X)}$ of V-polyhedrons satisfies the previous two assertions if and only if $[C_{V,M}]_V = [C'_{V,M}]_V$ for any $M \in \mathcal{M}_V(X)$.

Recall that the V-boundary of a V-polyhedron is defined "up to V-degenerate V-polyhedrons". That means if C_1 and C_2 are two V-polyhedrons such that $[C_1]_V = [C_2]_V$ then $\mathrm{bound}_V(C_1) = \mathrm{bound}_V(C_2)$. Let X be a Presburger-definable set and V be an affine component of $\overrightarrow{\mathrm{saff}}(X)$. From the previous theorem 6.3 we deduce that $\bigcup_{M\in\mathcal{M}_V(X)} \mathrm{bound}_V(C_{V,M})$ only depends on X and V. This semi-affine space is written $\mathrm{bound}_V(X)$.

$$\mathrm{bound}_V(X) = \bigcup_{M\in\mathcal{M}_V(X)} \mathrm{bound}_V(C_{V,M})$$

Example 6.4 *Let us consider the Presburger-definable set $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$ given in figure 6. We have $\overrightarrow{\mathrm{saff}}(X) = V$ where $V = \mathbb{Q}^2$ and $X_V = X$. We also have $\mathcal{M}_V(X) = \{\emptyset, \mathbb{Z}^2\}$. Remark that the sequence of V-polyhedrons $(C_{V,M})_{M \in \mathcal{M}_V(X)}$ given by $C_{V,\mathbb{Z}^2} = \{x \in \mathbb{Q}^2;\ x[1] \geq 0 \wedge x[2] \geq 4.x[1]\}$ and $C_{V,\emptyset} = \mathbb{Q}^2 \backslash C_{V,\mathbb{Z}^2}$ satisfies the decomposition theorem. From $\mathrm{bound}_V(C_{V,\mathbb{Z}^2}) = \mathrm{bound}_V(C_{V,\emptyset}) = \mathbb{Q}.(0,1) \cup \mathbb{Q}.(1,4)$, we deduce that $\mathrm{bound}_V(X) = \mathbb{Q}.(1,0) \cup \mathbb{Q}.(1,4)$.*

7. Polynomial time decomposition

In this section, we show that $\overrightarrow{\mathrm{saff}}(X)$ and $\mathrm{bound}_V(X)$ are computable in polynomial time from a NDD that represents a Presburger-definable set X.

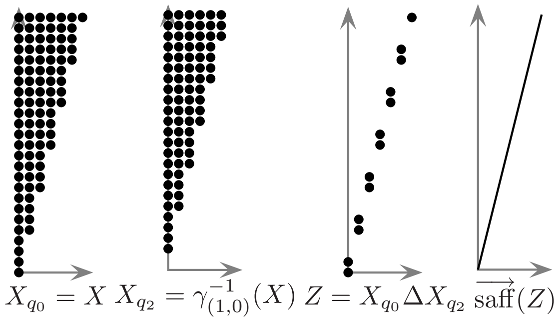


Figure 6. $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$

In order to illustrate the computation of $\overrightarrow{\mathrm{saff}}(X)$ from a NDD representing a Presburger-definable set X (see also example 5.15 and 5.16), assume that X is a non-empty set represented by a NDD $\mathcal{A}$. As any terminal component T is reachable from q_0, there exists at least one path $q_0 \xrightarrow{\sigma} q$ with $q \in T$. From proposition 5.5, we deduce that $X_q = \gamma_\sigma^{-1}(X)$ and in particular $\Gamma_\sigma(X_q) \subseteq X$ where $\Gamma_\sigma : \mathbb{Q}^m \to \mathbb{Q}^m$ is the *affine function that extends γ_σ* by $\Gamma_\sigma(x) = r^{|\sigma|}.x + \rho(\sigma)$. From this inclusion, a simple lemma shows that $\overrightarrow{\mathrm{saff}}(X_q) \subseteq \overrightarrow{\mathrm{saff}}(X)$. Now, just recall that $V_T(\mathcal{A}) = \overrightarrow{\mathrm{saff}}(X_q)$. Therefore, we have proved the inclusion $\bigcup_{T \in T(\mathcal{A})} V_T(\mathcal{A}) \subseteq \overrightarrow{\mathrm{saff}}(X)$. Even if the converse inclusion is not true in general, the following theorem shows that it holds for any Presburger-definable NDD. In particular we deduce that $\overrightarrow{\mathrm{saff}}(X)$ can be efficiently computed in polynomial time.

Theorem 7.1 *Let X be a Presburger-definable set represented by a NDD $\mathcal{A}$. We have the following equality:*

$$\overrightarrow{\mathrm{saff}}(X) = \bigcup_{T \in T(\mathcal{A})} V_T(\mathcal{A})$$

We illustrate the computation of $\mathrm{bound}_V(X)$ from the NDD $\mathcal{A}$ given in figure 5 that represents the set $X = \{x \in \mathbb{N}^2;\ x[2] \geq 4.x[1]\}$ given in figure 6. Remark that $\overrightarrow{\mathrm{saff}}(X) = V = \mathbb{Q}^2$, and $\mathrm{bound}_V(X) = \mathbb{Q}.(1,0) \cup \mathbb{Q}.(1,4)$. Proposition 5.5, shows that $X_{q_2} = \{x \in \mathbb{N}^m;\ x[2] \geq 4.x[1] + 2\}$ also given in figure 6. Remark that $\overrightarrow{\mathrm{saff}}(X_{q_0} \Delta X_{q_2}) = \mathbb{Q}.(1,4)$ which provides the affine component $\mathbb{Q}.(1,4)$ of $\mathrm{bound}_V(X)$.

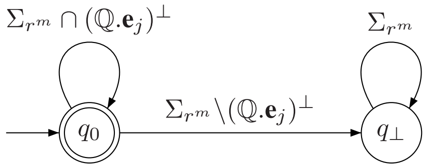


Figure 7. A NDD representing $\mathbb{N}^m \cap (\mathbb{Q}.\mathbf{e}_j)^\perp$

This technique is generalized to any Presburger-definable set X by the following theorem that proves in particular that we can efficiently compute $\mathrm{bound}_V(X)$ in polynomial time from a NDD that represents X (see also figure 7).

Theorem 7.2 *Let X be a Presburger-definable set represented by a NDD $\mathcal{A}$, and let V be an affine component of $\overrightarrow{\mathrm{saff}}(X)$.*

- *Consider $I_V(\mathcal{A})$, the set of states $(q_1, q_2) \in T \times T$ where $T \in T_V(\mathcal{A})$ and $\overrightarrow{\mathrm{saff}}(X_{q_1} \Delta X_{q_2})$ is strictly included in V.*
- *Consider $J_V(\mathcal{A})$, the set of $j \in \{1, \ldots, m\}$ such that $V \cap (\mathbb{Q}.\boldsymbol{e}_j)^\perp$ is strictly included in V and such that there exists $q \in F \cap T$ where $T \in T_V(\mathcal{A})$ and $\overrightarrow{\mathrm{saff}}(X_q \cap (\mathbb{Q}.\boldsymbol{e}_j)^\perp) = V \cap (\mathbb{Q}.\boldsymbol{e}_j)^\perp$.*

We have the following equality:

$$\mathrm{bound}_V(X) = \bigcup_{(q_1,q_2) \in I_V(\mathcal{A})} \overrightarrow{\mathrm{saff}}(X_{q_1} \Delta X_{q_2}) \cup \bigcup_{j \in J_V(\mathcal{A})} (V \cap (\mathbb{Q}.\boldsymbol{e}_j)^\perp)$$

8. Presburger synthesis

In this last section, we prove that we can decide in polynomial time if a NDD $\mathcal{A}$ represents a Presburger-definable set X. Moreover, in this case, we prove that we can compute in polynomial time a Presburger formula ϕ that defines X.

We only sketch the proof of this result. Assume that a Presburger-definable set X is represented by a NDD $\mathcal{A}$. We have proved that $\mathrm{comp}(\overrightarrow{\mathrm{saff}}(X))$ is computable in polynomial time and for any vector space V in this set, $\mathrm{bound}_V(X)$ is also computable in polynomial time. In technical report [21], we prove that we can also compute in polynomial time a sequence $(C_{V,M})_{M\in\mathcal{M}_V(X)}$ of V-polyhedrons satisfying decomposition theorem.

Remark that from decomposition theorem, we deduce the following corollary:

Corollary 8.1 *Let $X \subseteq \mathbb{N}^m$ be a non-empty Presburger-definable set and $(C_{V,M})_{M\in\mathcal{M}_V(X)}$ be a sequence of V-polyhedrons satisfying decomposition theorem. We have $\dim(X') < \dim(X)$ where X' is given by:*

$$X' = X\Delta\left(\bigcup_{\substack{V\in\mathrm{comp}(\overrightarrow{\mathrm{saff}}(X))\\ M\in\mathcal{M}_V(X)}} (M\cap(C_{V,M}+V^{\perp}))\right)$$

In technical report [21], we prove that we can choose $(C_{V,M})_{M\in\mathcal{M}_V(X)}$ correctly such that all the sets $M \cap (C_{V,M} + V^{\perp})$ are detectable in X. That means X' is detectable in X and in particular, by modifying the set of final states of the NDD $\mathcal{A}$, we obtain a NDD that represents X' with exactly the *same size*.

As $\dim(X') < \dim(X)$, an induction over the integral dimension provides the proof of the following theorem.

Theorem 8.2 *We can decide in polynomial time if a NDD $\mathcal{A}$ represents a Presburger-definable set X. Moreover, in this case, we can compute in polynomial time a Presburger formula ϕ that defines X.*

Remark 8.3 *Tools that manipulate NDD, represent the transition relation $\delta : Q \times \Sigma_{r^m} \to Q$ by a BDD [6] in order to avoid an exponential blow up due to the exponential size of Σ_{r^m}. Following [19], we deduce that all the results proved in this paper can be extended in polynomial time to this representation expect a technical one (see technical report [21]). In particular, we deduce that we can decide in non-deterministic polynomial time (non-deterministic polynomial time in the dimension m and polynomial time in the number of states $|Q|$) if such a NDD represents a Presburger-definable set. Moreover, in this case, we can compute in polynomial time a Presburger formula that defines the same set. We are not convinced that the problem of deciding if such a NDD represents a Presburger-definable set, can be done in polynomial time. Nevertheless, the problem remains open.*

Remark 8.4 *Our decision procedure can be used in order to decide in exponential time if a* most significant digit first *NDD $\mathcal{A}$ represents a Presburger-definable set X (a polynomial time procedure remains an open problem). In fact, by "flipping" the "direction" of the transitions and by determining the resulting automaton, we obtain a* least significant digit first *NDD $\bar{\mathcal{A}}$ that represents X. Even if from a theoretical point of view, an exponential blow up can appear, in practical examples, it is not the case (see [14] for the duality least/most significant digit first).*

Remark 8.5 *Our algorithms should be efficient on NDD with a large set of states. Assume that the dimension $m \geq 1$ is fixed. Recall that in [19], we proved that $\mathrm{aff}(X)$ is computable in linear time from a NDD that represents X. In particular $\overrightarrow{\mathrm{saff}}(X)$ is also computable in linear time. Moreover, even if theorem 7.2 seems to provide a $O(|Q|^4)$ time complexity algorithm for computing $\mathrm{bound}_V(X)$, we can just compute only one NDD for each $T \in T_V(\mathcal{A})$ whose the set of states is $T \times T$. Therefore $\mathrm{bound}_V(X)$ is computable in* quadratic *time. The exact complexity of our criterion will be detailed in a revue version of the paper.*

9. Conclusion and future work

We have described the precise structure of a NDD that represents a Presburger-definable set. We are currently working on the design of new efficient symbolic representations for Presburger-definable sets. In particular, we are interested in studying hybrid representations that use both NDD and constraint formulas. This is work in progress.

Acknowledgment: We thank Pierre McKenzie for his support and for his interesting remarks on so many versions of this paper.

References

[1] S. Bardin, A. Finkel, and J. Leroux. Faster acceleration of counter automata. In *Proc. 10th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2004) Barcelona, Spain, Mar. 2004*, volume 2988 of *Lecture Notes in Computer Science*, pages 576–590. Springer, 2004.

[2] S. Bardin, A. Finkel, J. Leroux, and L. Petrucci. FAST: Fast Acceleration of Symbolic Transition systems. In *Proc. 15th Int. Conf. Computer Aided Verification (CAV'2003), Boulder, CO, USA, July 2003*, volume 2725 of *Lecture Notes in Computer Science*, pages 118–121. Springer, 2003.

[3] L. Berman. Precise bounds for Presburger arithmetic and the reals with addition: Preliminary report. In *Proc. 18th IEEE Symp. Foundations of Computer Science (FOCS'77), Providence, RI, USA, Oct.-Nov. 1977*, pages 95–99, Providence, Rhode Island, 31 Oct.–2 Nov. 1977. IEEE.

[4] A. Boudet and H. Comon. Diophantine equations, Presburger arithmetic and finite automata. In *Proc. 21st Int. Coll. on Trees in Algebra and Programming (CAAP'96), Linköping, Sweden, Apr. 1996*, volume 1059 of *Lecture Notes in Computer Science*, pages 30–43. Springer, 1996.

[5] V. Bruyère, G. Hansel, C. Michaux, and R. Villemaire. Logic and p-recognizable sets of integers. *Bull. Belg. Math. Soc.*, 1(2):191–238, Mar. 1994.

[6] R. E. Bryant. Symbolic boolean manipulation with ordered binary-decision diagrams. *ACM Comput. Surv.*, 24(3):293–318, 1992.

[7] T. Bultan, R. Gerber, and W. Pugh. Model-checking concurrent systems with unbounded integer variables: symbolic representations, approximations, and experimental results. *ACM Transactions on Programming Languages and Systems*, 21(4):747–789, 1999.

[8] A. Cobham. On the base-dependance of sets of numbers recognizable by finite automata. *Mathematical Systems Theory*, 3:186–192, 1969.

[9] FAST homepage. `http://www.lsv.ens-cachan.fr/fast/`.

[10] L. Fribourg. Petri nets, flat languages and linear arithmetic. Invited lecture. In M. Alpuente, editor, *Proc. 9th Int. Workshop. on Functional and Logic Programming (WFLP'2000), Benicassim, Spain, Sept. 2000*, pages 344–365, 2000. Proceedings published as Ref. 2000.2039, Universidad Politécnica de Valencia, Spain.

[11] L. Fribourg and H. Olsén. Proving safety properties of infinite state systems by compilation into Presburger arithmetic. In *Proc. 8th Int. Conf. Concurrency Theory (CONCUR'97), Warsaw, Poland, Jul. 1997*, volume 1243 of *Lecture Notes in Computer Science*, pages 213–227. Springer, 1997.

[12] V. Ganesh, S. Berezin, and D. L. Dill. Deciding presburger arithmetic by model checking and comparisons with other methods. In *Proc. 4th Int. Conf. Formal Methods in Computer Aided Design (FMCAD'02), Portland, OR, USA, nov. 2002*, volume 2517 of *Lecture Notes in Computer Science*, pages 171–186. Springer, 2002.

[13] S. Ginsburg and E. H. Spanier. Semigroups, Presburger formulas and languages. *Pacific J. Math.*, 16(2):285–296, 1966.

[14] F. Klaedtke. On the automata size for presburger arithmetic. In *Proc. 19th Annual IEEE Symposium on Logic in Computer Science (LICS'04), Turku, Finland July 2004*, pages 110–119. IEEE Comp. Soc. Press, 2004.

[15] N. Klarlund, A. Møller, and M. I. Schwartzbach. MONA implementation secrets. *Int. J. of Foundations Computer Science*, 13(4):571–586, 2002.

[16] LASH homepage. `http://www.montefiore.ulg.ac.be/~boigelot/research/lash/`.

[17] L. Latour. From automata to formulas: Convex integer polyhedra. In *Proc. 19th Annual IEEE Symposium on Logic in Computer Science (LICS'04), Turku, Finland July 2004*, pages 120–129. IEEE Comp. Soc. Press, 2004.

[18] J. Leroux. *Algorithmique de la vérification des systèmes à compteurs. Approximation et accélération. Implémentation de l'outil Fast*. PhD thesis, Ecole Normale Supérieure de Cachan, Laboratoire Spécification et Vérification. CNRS UMR 8643, décembre 2003.

[19] J. Leroux. The affine hull of a binary automaton is computable in polynomial time. In *Proc. 5th Int. Workshop on Verification of Infinite State Systems (INFINITY 2003), Marseille, France, Sep. 2003*, volume 98 of *Electronic Notes in Theor. Comp. Sci.*, pages 89–104. Elsevier Science, 2004.

[20] J. Leroux. Disjunctive invariants for numerical systems. In *Proc. 2nd Int. Symp. Automated Technology for Verification and Analysis (ATVA 2004), Taipei, Taiwan, Nov. 2004*, volume 3299 of *Lecture Notes in Computer Science*, pages 99–107. Springer, 2004.

[21] J. Leroux. A polynomial time presburger criterion and synthesis for number decision diagrams. Technical Report 1251, Département d'Informatique et de Recherche Opérationnelle. Faculté des arts et des sciences. Université de Montréal. Montréal, Quebec, Canada, september 2004.

[22] D. Lugiez. From automata to semilinear sets: a solution for polyhedra and even more general sets. In *Proc. 9th. Int. Conf. on Implementation and Application of Automata (CIAA'04), Queen's University, Kingston, Ontario, Canada, Jul. 2004*, volume 3317 of *Lecture Notes in Computer Science*, pages 321–322. Springer, 2004.

[23] A. Muchnik. Definable criterion for definability in presburger arithmetic and its applications. (in russian), preprint, Institute of new technologies, 1991.

[24] A. Muchnik. The definable criterion for definability in presburger arithmetic and its applications. *Theoretical Computer Science*, 290:1433–1444, 2003.

[25] OMEGA homepage. `http://www.cs.umd.edu/projects/omega/`.

[26] M. Presburger. Uber die volstandigkeit eines gewissen systems der arithmetik ganzer zahlen, in welchem die addition als einzige operation hervortritt. In *C. R. 1er congres des Mathematiciens des pays slaves, Varsovie*, pages 92–101, 1929.

[27] T. Rybina and A. Voronkov. Brain: Backward reachability analysis with integers. In *Proc. 9th Int. Conf. Algebraic Methodology and Software Technology (AMAST'2002), Saint-Gilles-les-Bains, Reunion Island, France, Sep. 2002*, volume 2422 of *Lecture Notes in Computer Science*, pages 489–494. Springer, 2002.

[28] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley and Sons, New York, 1987.

[29] A. Semenov. Presburgerness of predicates regular in two number systems. *Siberian Mathematical Journal*, 18:289–299, 1977.

[30] P. Wolper and B. Boigelot. An automata-theoretic approach to Presburger arithmetic constraints. In *Proc. 2nd Int. Symp. Static Analysis (SAS'95), Glasgow, UK, Sep. 1995*, volume 983 of *Lecture Notes in Computer Science*, pages 21–32. Springer, 1995.

[31] P. Wolper and B. Boigelot. On the construction of automata from linear arithmetic constraints. In *Proc. 6th Int. Conf. Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2000), Berlin, Germany, Mar.-Apr. 2000*, volume 1785 of *Lecture Notes in Computer Science*, pages 1–19. Springer, 2000.