Image Computation in Infinite State Model Checking

Alain Finkel[1] and Jérôme Leroux[2]

[1] Laboratoire Spécification et Vérification, CNRS UMR 8643 & ENS de Cachan
61 av. du Président Wilson, 94235 Cachan cedex, France
finkel@lsv.ens-cachan.fr

[2] Departement d'Informatique et de Recherche Opérationnelle
Université de Montréal, Pavillon André-Aisenstadt
CP 6128 succ Centre Ville, H3C 3J7, Montréal, QC Canada
leroujer@iro.umontreal.ca

Abstract. The model checking of a counters system S often reduces to the effective computation of the set of predecessors $\mathrm{Pre}_S^*(X)$ of a set of integer vectors X. Because the exact computation of this set is not possible in general, we are interested in characterizing the minimal Number Decision Diagrams (NDD) [WB00] that represents the set $\mathrm{Pre}^{\leq k}(X)$. In particular, its size is proved to be just polynomially bounded in k when S is a counters system with a finite monoïd [FL02], explaining why there is no exponential blow up in k.

1 Introduction

Model checking infinite-state transition systems S often reduces to the effective computation of the potentially infinite set of predecessors Pre_S^*. More precisely, the safety model checking can be expressed as the following problem:

- Given as inputs an infinite-state transition system S and two possibly infinite sets X_0 and X of respectively initial states and non-safe states, decide if $X_0 \cap \mathrm{Pre}_S^*(X)$ is empty.

Infinite Sets. In order to effectively compute $\mathrm{Pre}_S^*(X)$, one generally needs to find a class of infinite sets which has the following properties: closure under union, closure under Pre_S, membership and inclusion are decidable with a good complexity, and there exists a canonical representation. We are considering the Number Decision Diagrams (NDD) that provides an automata-based symbolic representation of some subsets of $\mathbb{N}^m$.

Infinite-State Transition Systems. We will focus on systems S with m integer variables and more precisely on counters systems with a finite monoïd (also known as finite linear systems [FL02]), a class of systems that contains the reset/transfer Petri Nets [DJS99], generalized broadcast protocols [EN98, Del00], and all the counters automata. As this model is very general and powerful, the

R. Alur and D.A. Peled (Eds.): CAV 2004, LNCS 3114, pp. 361–371, 2004.

price to pay is the undecidability of reachability properties and in particular the sequence $(\mathrm{Pre}_S^{\leq k}(X))_k$ does not converge in general.

Our Image Computation Problem. The characterization of the NDD structure that represents $\mathrm{Pre}_S^{\leq k}(X)$ in function of k is an important problem in order to effectively compute the exact limit $\mathrm{Pre}_S^{*}(X)$ or a "good" over-approximation:

- When there exists an integer k_0 such that $\mathrm{Pre}_S^{*}(X) = \mathrm{Pre}_S^{\leq k_0}(X)$, the characterization can be useful in order to design an efficient algorithm that incrementally computes these sets. Recall that even if the convergence of $(\mathrm{Pre}_S^{\leq k}(X))_k$ is not guaranteed by the theory, in practice we observe that often, this sequence converges [Del00, BB03] and it often converges quickly [Bra, Bab]. Moreover, as soon as the set X is upward closed and S is a Well Structured Transition System [FS01], the convergence is insured.
- When the sequence $(\mathrm{Pre}_S^{\leq k}(X))_k$ diverges, the characterization can be useful in order to design NDD specialized acceleration operators that computes the exact limit $\mathrm{Pre}_S^{*}(X)$ [BLW03, FL02] or an over-approximation [BGP99].

Related Works. We use the approach called the *regular model checking*: for channel systems, Semi-Linear Regular Expressions [FPS00] and Constrained Queue-content Decision Diagrams [BH99] have been proposed; for lossy channel systems [ABJ98], the tools Lcs (in the more general tool TreX [ABS01] [Tre]) uses the downward-closed regular languages and the corresponding subset of Simple Regular Expressions for sets and it represents them by finite automata to compute Post*; for stack automata, regular expressions or finite automata are sufficient to represent Pre* and Post* [BEF$^+$00]; for Petri nets and parameterized rings, [FO97] uses regular languages and Presburger arithmetics (and acceleration) for sets. For Transfer and Reset Petri nets [DFS98], the tool Babylon [Bab] utilizes the upward closed sets and represents them by Covering Sharing Trees [DRV01], a variant of BDD; for counters automata, the tool Brain [Bra] uses linear sets and represent them by their linear bases and periods; Mona [Mon] [KMS02] and FMona [BF00] use formula in WS1S to represent sets; the tool CSL-ALV [BGP97] [Alv] uses linear arithmetic constraints for sets and manipulates formula with the Omega solver and the automata library of Lash.

For counters systems with a finite monoïd, tools Fast [Fas], [FL02], [BFLP03] and Lash [Las] utilize semi-linear sets and represents them by NDD, moreover, these two tools are able to accelerate loops [Boi03] [FL02]. In [FL04], the NDD $\mathrm{Pre}_S(X)$ is proved to be computable in polynomial time in function of the NDD X for a large class of systems S that contains all the counters system with a finite (or infinite) monoïd. Moreover, the size of the NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ is proved to be polynomial in k when S and X are "defined in the interval-logic", a restrictive class compared to the sets that can be represented by NDD.

Our Results.

1. We prove that the asymptotic number of states of the minimal NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ is polynomial in k for any counters systems S with a finite monoïd and for any set X represented by a NDD.

2. We show that the structure of the minimal NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ is similar to a BDD. That provides a new way for implementing a NDD library using all the BDD techniques for speeding-up the computation like cache-computation, minimization in linear time, Strong canonical form, well-known in the field of BDD [Bry92].

Plan of the Paper. The structure of the minimal NDD that represents a set X is given in section 3. In the next one, the definition of counters systems with a finite monoïd is recall. The structure of the minimal NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ in function of k, is studied in the last section 5.

2 Preliminaries

The cardinal of a finite set X is written card(X). The set of rational numbers, non negative rational numbers, integers and positive integers are respectively written $\mathbb{Q}$, $\mathbb{Q}^+$, $\mathbb{Z}$ and $\mathbb{N}$. The set of vectors with $m \geq 1$ components in a set X is written X^m. The i-th component of a vector $x \in X^m$ is written $x_i \in X$; we have $x = (x_1, \dots, x_m)$. For any vector $v, v' \in \mathbb{Q}^m$ and for any $t \in \mathbb{Q}$, we define $t.v$ and $v + v'$ in $\mathbb{Q}^m$ by $(t.v)_i = t.v_i$ and $(v + v')_i = v_i + v'_i$. For any $x \in \mathbb{Q}^m$, we define $||x||_\infty = \max_i(|x_i|)$.

The set of square matrices of size m in $K \subseteq \mathbb{Q}$ is written $\mathcal{M}_m(K)$. The element $M_{ij} \in K$ is the i-th raw and j-th column of a matrix $M \in \mathcal{M}_m(K)$. The identity matrix is written I_m. The vector $M.x \in \mathbb{Q}^m$ is naturally defined by $(M.x)_i = \sum_{j=1}^m M_{ij} x_j$. The subset $M^{-1}Y \subseteq \mathbb{Q}^m$ is defined by $M^{-1}Y = \{x \in \mathbb{Q}^m;\ M.x \in Y\}$ for any $M \in \mathcal{M}(\mathbb{Q})$ and $X \subseteq \mathbb{Q}^m$. For any $M \in \mathcal{M}_m(\mathbb{Q})$, we define $||M||_\infty = \max_{i,j}(|M_{ij}|)$.

The set of words over a finite alphabet Σ is written Σ^*. The concatenation of two words σ and σ' in Σ^* is written $\sigma.\sigma'$. The empty word in Σ^* is written ϵ. The residue $\sigma^{-1}.\mathcal{L}$ of a language $\mathcal{L} \subseteq \Sigma^*$ by a word $\sigma \in \Sigma^*$ is defined by $\sigma^{-1}.\mathcal{L} = \{w \in \Sigma^*;\ \sigma.w \in \mathcal{L}\}$

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 relation, $Q_0 \subseteq Q$ is the set of initial states and $F \subseteq Q$ is the set of final states. As usual, we extends 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 $\mathcal{L}(\mathcal{A}) = \{\sigma \in \Sigma^*;\ \delta(q_0, \sigma) \in F\}$.

3 Number Decision Diagrams

Recall that there exist two natural ways in order to associate to a word σ a vector in $\mathbb{N}^m$ following that the first letter of σ is considered as an "high bit" or a "low bit". In this article, we consider the "low bit" representation (even if the other one, just seems to be symmetrical, results proved in the paper cannot be easily extended to the other one).

Let us consider an integer $r \geq 2$ called the *basis of the decomposition* and an integer $m \geq 1$ called the *dimension of the represented vectors.* A *digit vector* b is an element of the finite alphabet $\Sigma_{r^m} = \{0, \dots, r-1\}^m$. The vector $\rho(\sigma) \in \mathbb{N}^m$ associated to a word $\sigma = b_1 \dots b_n$ of $n \geq 1$ digit vectors $b_i \in \Sigma_{r^m}$ is defined by $\rho(\sigma) = \sum_{i=1}^{n} r^{i-1}.b_i$. We naturally define $\rho(\epsilon) = (0, \dots, 0)$.

Definition 1 ([WB00], [BC96]). *A* Number Decision Diagram (NDD) $\mathcal{A}$ *is a finite deterministic and complete automaton over the alphabet* Σ_{r^m} *such that:*

$$\rho^{-1}(\rho(\mathcal{L}(\mathcal{A}))) = \mathcal{L}(\mathcal{A})$$

The subset $X = \rho(\mathcal{L}(\mathcal{A})) \subseteq \mathbb{N}^m$ is called *the set represented* by the NDD $\mathcal{A}$. Such a subset X is said *NDD-definable.*

Remark 1. Thanks to the condition $\rho^{-1}(\rho(\mathcal{L}(\mathcal{A}))) = \mathcal{L}(\mathcal{A})$, the set $\mathbb{N}^m \backslash X$ is represented by the NDD $\mathcal{A} = (Q, \Sigma_{r^m}, \delta, q_0, Q \backslash F)$. Recall that from any deterministic and complete binary automaton $\mathcal{A}$, we can efficiently computes a NDD $\mathcal{A}'$ such that $\rho(\mathcal{L}(\mathcal{A})) = \rho(\mathcal{L}(\mathcal{A}'))$ [KMS02, Ler03].

Remark 2. Any Presburger definable set (a set defined by a formula in the first order logic $\langle \mathbb{N}, +, \leq \rangle$) [BC96, WB00], or any semi-linear set (a set equal to a finite union of sets of the form $x_0 + \sum_{p \in P} \mathbb{N}.p$ where $x_0 \in \mathbb{N}^m$ and P is a finite subset of $\mathbb{N}^m$) [GS66, Reu89] can be effectively represented by a NDD. Moreover, recall that a set is NDD-definable if and only if it is definable by a formula in the first order logic $\langle \mathbb{N}, +, \leq, V_r \rangle$ where V_r is the valuation function in base r defined by $y = V_r(x)$ if and only if y is the greatest power of r that divides x [BHMV94].

In the remaining of this section, we characterize the minimal NDD that represents a subset X.

The equality $\rho(\sigma.\sigma') = \rho(\sigma) + r^{|\sigma|}.\rho(\sigma')$ shows that the function $\gamma_\sigma : \mathbb{N}^m \rightarrow \mathbb{N}^m$ defined by $\gamma_\sigma(x) = \rho(\sigma) + r^{|\sigma|}.x$ plays an important role. We are going to prove that X is NDD-definable if and only if the following set $Q(X)$ is finite:

$$Q(X) = \{\gamma_\sigma^{-1}(X);\ \sigma \in \Sigma_{r^m}^*\}$$

Remark that for any digit vector $b \in \Sigma_{r^m}$ and for any $q \in Q(X)$, the set $\gamma_b^{-1}(q)$ remains in $Q(X)$. Hence, if $Q(X)$ is finite, we can easily associate to a set X a deterministic and complete automaton $\mathcal{A}(X)$.

Definition 2. *Let* $X \subseteq \mathbb{N}^m$ *be such that* $Q(X) = \{\gamma_\sigma^{-1}(X);\ \sigma \in \Sigma_{r^m}^*\}$ *is finite. The deterministic and complete automaton* $\mathcal{A}(X)$ *is defined by:*

$$\begin{cases} \mathcal{A}(X) = (Q(X), \Sigma_{r^m}, \delta, q_0, F) \\ \delta(q, b) = \gamma_b^{-1}(q) \\ q_0 = X \\ F = \{q \in Q(X);\ (0, \dots, 0) \in q\} \end{cases}$$

We are going to prove that $\mathcal{A}(X)$ is the unique minimal NDD that represents X.

Lemma 1. *For any $X \subseteq \mathbb{N}^m$ and $\sigma \in \Sigma_{rm}^*$, we have $\sigma^{-1}.\rho^{-1}(X) = \rho^{-1}(\gamma_\sigma^{-1}(X))$.*

Proof. We have $w \in \sigma^{-1}.\rho^{-1}(X)$ iff $\sigma.w \in \rho^{-1}(X)$ iff $\rho(\sigma.w) \in X$ iff $\gamma_\sigma(\rho(w)) \in X$ iff $\rho(w) \in \gamma_\sigma^{-1}(X)$ iff $w \in \rho^{-1}(\gamma_\sigma^{-1}(X))$. □

The following theorem is really important because it proves that the structure of the minimal NDD that represents a set X can be obtained just by studying the sets $\gamma_\sigma^{-1}(X)$.

Theorem 1. *A set $X \subseteq \mathbb{N}^m$ is NDD-definable if and only if $Q(X)$ is finite. Moreover, in this case, $\mathcal{A}(X)$ is the unique minimal NDD that represents X.*

Proof. Assume that $Q(X)$ is a finite set. We are going to show that $\mathcal{A}(X)$ is a NDD that represents X by proving $\mathcal{L}(\mathcal{A}(X)) = \rho^{-1}(X)$. By definition of $\mathcal{A}(X)$, we have $\sigma \in \mathcal{L}(\mathcal{A}(X))$ iff $(0, \ldots, 0) \in \gamma_\sigma^{-1}(X)$. Therefore $\sigma \in \mathcal{L}(\mathcal{A}(X))$ iff $\rho(\sigma) = \gamma_\sigma((0, \ldots, 0)) \in X$. Hence, we have proved that $\mathcal{L}(\mathcal{A}(X)) = \rho^{-1}(X)$. In particular $\rho(\mathcal{L}(\mathcal{A}(X))) = X$ and $\rho^{-1}(\rho(\mathcal{L}(\mathcal{A}(X)))) = \mathcal{L}(\mathcal{A}(X))$. We have proved that $\mathcal{A}(X)$ is an NDD that represents X.

Now, assume that X is NDD-definable and let us prove that $Q(X)$ is finite. There exists a NDD $\mathcal{A}$ such that X is represented by $\mathcal{A}$. Let $\mathcal{L}$ be the regular language accepted by $\mathcal{A}$. As $\mathcal{A}$ is an NDD that represents X, we have $\rho^{-1}(\rho(\mathcal{L})) = \mathcal{L}$ and $\rho(\mathcal{L}) = X$. We deduce $\mathcal{L} = \rho^{-1}(X)$. As the minimal deterministic and complete automaton that recognizes $\mathcal{L}$ is unique, there exists a unique minimal automaton that represents X. Recall that the set of states of this minimal automaton is given by $\{\sigma^{-1}.\mathcal{L};\ \sigma \in \Sigma_{rm}^*\}$. From lemma 1, we deduce that $Q(X) = \{\rho(\sigma^{-1}.\mathcal{L});\ \sigma \in \Sigma_{rm}^*\}$. Therefore, $Q(X)$ is finite and by uniqueness of the minimal automaton, $\mathcal{A}(X)$ is the unique minimal NDD that represents X. □

4 Counters Automata with Finite Monoïd

The class of counters automata with finite monoïd finite (a.k.a finite linear systems [FL02]) is a natural extension of some classes of models like Reset/Transfer Petri Nets [DJS99], counter automata or broadcast protocols [EN98, Del00]. Recall that this class is also used as the input model of the accelerated symbolic model checker Fast [BFLP03].

Let us first provide the definition of a counters system.

Definition 3. *A NDD-linear function f is a tuple $f = (D, M, v)$ such that $D \subseteq \mathbb{N}^m$ is NDD-definable, $M \in \mathcal{M}_m(\mathbb{Z})$ and $v \in \mathbb{Z}^m$.*

Without any ambiguity, we also denote by f the function $f : D \to \mathbb{N}^m$ defined by $f(x) = M.x + v$ for any $x \in D$. The composition of two NDD-linear functions is naturally defined by $(D_1, M_1, v_1) \circ (D_2, M_2, v_2) = (D_2 \cap M_2^{-1}(D_1 - v_2), M_1.M_2, M_1.v_2 + v_1)$.

Definition 4. *A* counters system *S (a.k.a linear system [FL02]) is a tuple $S = (\Sigma, f_\Sigma)$ where Σ is a finite set of* actions *and $f_\Sigma = \{f_a;\ a \in \Sigma\}$ is a finite set of NDD-linear functions.*

For any word $\sigma = b_1 \dots b_n$ of $n \geq 1$ actions $b_i \in \Sigma$, the NDD-linear function f_σ is defined as $f_\sigma = f_{b_n} \circ \cdots f_{b_1}$. The NDD-linear function f_ϵ is defined by $f_\epsilon = (\mathbb{N}^m, I, (0, \dots, 0))$. We denote by $(D_\sigma, M_\sigma, v_\sigma)$ the NDD-linear function f_σ.

Like in [FL02], we define the monoïd of S.

Definition 5 ([FL02]). *The monoïd multiplicatively generated by the square matrices M_a is called the monoïd of S and written $\mathcal{M}_S = \{M_\sigma;\ \sigma \in \Sigma^*\}$.*

Definition 6. *A counters system S such that $\mathcal{M}_S$ is finite is called a counter system with a finite monoïd (a.k.a finite linear system [FL02]).*

Remark 3. The class of counters systems with a finite monoïd enjoys good properties that allow to easily *accelerate* the computation of the reachability set [FL02, Boi03, Ler03].

Finally, let us recall the definition of the set of immediate predecessors.

Definition 7. *Let S be a counters system. The set* $\mathrm{Pre}_S(X)$ *of immediate predecessors of a set X is defined by* $\mathrm{Pre}_S(X) = \bigcup_{a \in \Sigma} f_a^{-1}(X)$.

5 Structure of the Minimal NDD $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$

In [FL04], we have proved that for any counters system S, we can effectively computes in polynomial time a NDD that represents $\mathrm{Pre}_S(X)$ from any NDD that represents X. This result provides an exponential time algorithm for computing the minimal NDD $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$ in function of k. In fact, assume that each step of the computation multiplies the number of states of the NDD just by 2, then after k steps, the number of states of the NDD is multiplied by 2^k. However, in practice, such an exponential blow up does not appear.

To explain this experimental result, we are going to study the structure of the minimal NDD $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$ in function of $k \geq 0$. We prove an *unexpected result*: the NDD has a “BDD-like” structure and its number of states is polynomial in k for any counter systems S with a finite monoïd and for any set X NDD-definable.

We first prove a technical lemma.

Lemma 2. *Let $X \subseteq \mathbb{N}^m$ be a NDD-definable set, $M \in \mathcal{M}(\mathbb{Z})$ and $\alpha \in \mathbb{Q}^+$. There exists a finite class $\mathcal{C}_{X,M,\alpha}$ such that for any $v \in \mathbb{Z}^m$ and for any $w \in \Sigma_{r^m}^*$, we have:*

$$||v||_\infty \leq \alpha.r^{|w|} \Longrightarrow \gamma_w^{-1}(\mathbb{N}^m \cap M^{-1}(X - v)) \in \mathcal{C}_{X,M,\alpha}$$

Proof. Let $v \in \mathbb{Z}^m$ and $w \in \Sigma_{r^m}^*$ such that $||v||_\infty \leq \alpha.r^{|w|}$. We have:

$$\begin{aligned}\gamma_w^{-1}(\mathbb{N}^m \cap M^{-1}(X - v)) &= \mathbb{N}^m \cap \left[\frac{1}{r^{|w|}}(\mathbb{N}^m \cap M^{-1}(X - v) - \rho(w))\right] \\ &= \mathbb{N}^m \cap \left[M^{-1}(\frac{1}{r^{|w|}}(X - v - M.\rho(w)))\right]\end{aligned}$$

From the equality $X = \bigcup_{w_0 \in \Sigma_{r^m}^{|w|}} \gamma_{w_0}(\gamma_{w_0}^{-1}(X))$, we deduce:

$$\begin{aligned}
&\gamma_w^{-1}(\mathbb{N}^m \cap M^{-1}(X - v)) \\
&= \bigcup_{w_0 \in \Sigma_{r^m}^{|w|}} \left(\mathbb{N}^m \cap M^{-1} \left(\gamma_{w_0}^{-1}(X) + \frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \right) \right)
\end{aligned}$$

Let $B = \{z \in \mathbb{Z}^m;\ ||z||_\infty \leq 1 + \alpha + m.\,||M||_\infty\}$ and let us prove that for any $w_0 \in \Sigma_{r^m}^{|w|}$, if $\mathbb{N}^m \cap M^{-1} \left(\gamma_{w_0}^{-1}(X) + \frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \right) \neq \emptyset$ then $\frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \in B$. In fact, in this case, there exists $x_0 \in \gamma_{w_0}^{-1}(X)$ such that $x_0 + \frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \in M.\mathbb{N}^m \subseteq \mathbb{Z}^m$. Therefore $\frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \in \mathbb{Z}^m$. Moreover, we have the following inequality:

$$\begin{aligned}
\left|\left| \frac{\rho(w_0) - v - M.\rho(w)}{r^{|w|}} \right|\right|_\infty &\leq \frac{(r^{|w|} - 1) + \alpha.r^{|w|} + m.\,||M||_\infty .(r^{|w|} - 1)}{r^{|w|}} \\
&\leq 1 + \alpha + m.\,||M||_\infty
\end{aligned}$$

Now, just remark that the following finite class $\mathcal{C}_{X,M,\alpha}$ satisfies the lemma:

$$\mathcal{C}_{X,M,\alpha} = \left\{ \bigcup_{(q,b) \in F} \mathbb{N}^m \cap M^{-1}(q + b);\ F \subseteq Q(X) \times B \right\}$$

□

Theorem 2. *Let S be a counters system with a finite monoïd and X be a NDD-definable set. There exists a finite class $\mathcal{C}_{S,X}$ of subsets of $\mathbb{N}^m$ such that for any $w \in \Sigma_{r^m}^*$ and for any $\mathcal{L} \subseteq \Sigma^{\leq r^{|w|}}$, we have:*

$$\gamma_w^{-1} \left(\bigcup_{\sigma \in \mathcal{L}} f_\sigma^{-1}(X) \right) \in \mathcal{C}_{S,X}$$

Proof. Let $\alpha = \max_{(M,a) \in \mathcal{M}_S \times \Sigma} ||M.v_a||_\infty$. We define the function $g_\sigma : \mathbb{Q}^m \to \mathbb{Q}^m$ by $g_\sigma(x) = M_\sigma.x + v_\sigma$ for any $x \in \mathbb{Q}^m$, $\sigma \in \Sigma^*$.

Let $\sigma = a_1 \dots a_n$ be a word of $n \geq 1$ actions in Σ and $w \in \Sigma_{r^m}^*$ be a word of vector digits such that $|\sigma| \leq r^{|w|}$. The sequence of prefixes $(\sigma_i)_{0 \leq i \leq n}$ of σ is defined by $\sigma_i = a_1 \dots a_i$. The set $I(M, a, \sigma) = \{i \in \{1, \dots, n\};\ (M_{\sigma_i}, v_{\sigma_i}) = (M, v)\}$ where $(M, a) \in \mathcal{M}_S \times \Sigma$ is useful to compute the set $\gamma_w^{-1}(f_\sigma^{-1}(X))$ as it is shown by the following equality:

$$\begin{aligned}
\gamma_w^{-1}(f_\sigma^{-1}(X)) =& \gamma_w^{-1} \left(g_{\sigma_0}^{-1}(D_{a_1}) \cap \dots \cap g_{\sigma_{n-1}}^{-1}(D_{a_n}) \cap g_{\sigma_n}^{-1}(X) \right) \\
=& \bigcap_{(M,a) \in \mathcal{M}_S \times \Sigma} \ \bigcap_{i \in I(M,a,\sigma)} \gamma_w^{-1} \left(\mathbb{N}^m \cap M^{-1}(D_a - v_{\sigma_i}) \right) \\
& \cap \gamma_w^{-1}(\mathbb{N}^m \cap M_{\sigma_n}^{-1}(X - v_{\sigma_n}))
\end{aligned}$$

Let $\mathcal{C}_{X,M,\alpha}$ and $\mathcal{C}_{D_a,M,\alpha}$ be some finite classes satisfying lemma 2 for any $(M,a) \in \mathcal{M}_S \times \Sigma$. From $v_{\sigma_i} = \sum_{j=1}^{i} M_{\sigma_j}.v_{a_j}$, we deduce $||v_{\sigma_i}||_\infty \leq \alpha.i \leq \alpha.|\sigma| \leq \alpha.r^{|w|}$. Therefore, we have proved that $\gamma_w^{-1}(f_\sigma^{-1}(X))$ is in the following finite class $\mathcal{C}_{S,X}^0$:

$$\mathcal{C}_{S,X}^0 = \left\{ \bigcap_{Y \in F} Y \cap X';\ X' \in \bigcup_{M \in \mathcal{M}_S} \mathcal{C}_{X,M,\alpha};\ F \subseteq \bigcup_{(M,a) \in \mathcal{M}_S \times \Sigma} \mathcal{C}_{D_a,M,\alpha} \right\}$$

Now, let $\mathcal{C}_{S,X}$ be the set of all finite unions of elements in $\mathcal{C}_{S,X}^0$. From the equality $\gamma_w^{-1}(\bigcup_{\sigma \in \mathcal{L}} f_\sigma^{-1}(X)) = \bigcup_{\sigma \in \mathcal{L}} \gamma_w^{-1}(f_\sigma^{-1}(X))$, we deduce that $\gamma_w^{-1}(\bigcup_{\sigma \in \mathcal{L}} f_\sigma^{-1}(X)) \in \mathcal{C}_{S,X}$ for any $\mathcal{L} \subseteq \Sigma^{\leq r^{|w|}}$. □

We can deduce many interesting results from the previous theorem. The first *unexpected one* is about the asymptotic number of states of the minimal NDD $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$ in function of k.

Corollary 1. *Let S be a counters system with a finite monoïd and X be a NDD-definable set. There exists a constant $c_{S,X}$ such that the number of states of the minimal NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ is bounded $k^m + c_{S,X}$.*

Proof. Let $\mathcal{C}_{S,X}$ be a class of finite subsets of $\mathbb{N}^m$ satisfying theorem 2 and let $X_k = \mathrm{Pre}_S^{\leq k}(X)$. From $X_k = \bigcup_{\sigma \in \Sigma^{\leq k}} f_\sigma^{-1}(X)$, we deduce that for any $w \in \Sigma_{r^m}^*$ such that $k \leq r^{|w|}$, we have $\gamma_w^{-1}(X_k) \in \mathcal{C}_{S,X}$. From theorem 1 we deduce that the set of states $Q(X_k)$ of the minimal NDD that represents X_k satisfies $Q(X_k) \subseteq \mathcal{C}_{S,X} \cup \{\gamma_w^{-1}(X_k);\ r^{|w|} < k\}$. From $\mathrm{card}(\{\gamma_w^{-1}(X_k);\ r^{|w|} < k\}) \leq k^m$, we deduce that the cardinal of $Q(X_k)$ is bounded by $k^m + \mathrm{card}(\mathcal{C}_{S,X})$. □

The previous corollary proves an experimental result : the number of counters m is an exponential limitation for the effective computation of $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$ for large value of k. However, it explains why there is no exponential blow up in k.

Now, let us study precisely the structure of $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$.

Definition 8. *A state q of a NDD $\mathcal{A}$ is said acyclic if the number of paths $q_0 \to q$ is finite.*

Corollary 2. *Let S be a counters system with a finite monoïd and let X be a NDD-definable set. The number of non acyclic states of the minimal NDD $\mathcal{A}(\mathrm{Pre}_S^{\leq k}(X))$ is bounded independently of k.*

Proof. Let $\mathcal{C}_{S,X}$ be a class of finite subsets of $\mathbb{N}^m$ satisfying theorem 2 and let $X_k = \mathrm{Pre}_S^{\leq k}(X)$. We are going to prove that for any non acyclic state q of $\mathcal{A}(X_k)$, we have $q \in \mathcal{C}_{S,X}$. As the number of paths $q_0 \to q$ is infinite, there exists a path $q_0 \xrightarrow{\sigma} q$ such that $r^{|\sigma|} \geq k$. In this case $\gamma_\sigma^{-1}(X_k) \in \mathcal{C}_{S,X}$. From $q = \gamma_\sigma^{-1}(X_k)$, we deduce $q \in \mathcal{C}_{S,X}$. Therefore, the number of non acyclic states of $\mathcal{A}(X_k)$ is bounded by $\mathrm{card}(\mathcal{C}_{S,X})$. □

Remark 4. The cardinal of the set $\mathcal{C}_{S,X}$ is used in the two corollaries. From the proof of lemma 2 and the proof of theorem 2, this cardinal can be easily bounded by an elementary function in the size of S, $\mathcal{A}(X)$ and in the size of the monoïd $\mathcal{M}_S$. From [MS77], we deduce that $\mathcal{M}_S$ has a size elementary in the size of S when matrices M_a are in $\mathcal{M}_m(\mathbb{N})$. Therefore, in this case, the cardinal of $\mathcal{C}_{S,X}$ is elementary in the size of S and $\mathcal{A}(X)$. When matrices M_a are in $\mathcal{M}_m(\mathbb{Z})$, the elementary size of the monoïd is an open problem to the best of our knowledge.

We can easily extend the previous corollary in order to show that the structure of the minimal NDD that represents $\mathrm{Pre}_S^{\leq k}(X)$ corresponds to a BDD [Bry92] "concatenated" with a NDD that does not depends on k. This final result shows a new way for implementing a NDD library using all the BDD techniques for speeding-up the computation like cache-computation, minimization in linear time and strong canonical form, well-known in the field of BDD. Our symbolic model-checker Fast [Fas], will be available with this new library as soon as possible.

References

[ABJ98] Parosh Aziz Abdulla, Ahmed Bouajjani, and Bengt Jonsson. On-the-fly analysis of systems with unbounded, lossy FIFO channels. In *Proc. 10th Int. Conf. Computer Aided Verification (CAV'98), Vancouver, BC, Canada, June-July 1998*, volume 1427 of *Lecture Notes in Computer Science*, pages 305–318. Springer, 1998.

[ABS01] Aurore Annichini, Ahmed Bouajjani, and Mihaela Sighireanu. TReX: A tool for reachability analysis of complex systems. In *Proc. 13th Int. Conf. Computer Aided Verification (CAV'2001), Paris, France, July 2001*, volume 2102 of *Lecture Notes in Computer Science*, pages 368–372. Springer, 2001.

[Alv] Alv homepage. http://www.cs.ucsb.edu/~bultan/composite/.

[Bab] Babylon homepage.
http://www.ulb.ac.be/di/ssd/lvbegin/CST/-index.html.

[BB03] Constantinos Bartziz and Tevfik Bultan. Efficient image computation in infinite state model checking. In *Proc. 15th Int. Conf. Computer Aided Verification (CAV'2003), Boulder, CO, USA, July 2003*, volume 2725 of *Lecture Notes in Computer Science*, pages 249–261. Springer, 2003.

[BC96] Alexandre Boudet and Hubert 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.

[BEF$^+$00] A. Bouajjani, J. Esparza, A. Finkel, O. Maler, P. Rossmanith, B. Willems, and P. Wolper. An efficient automata approach to some problems on context-free grammars. *Information Processing Letters*, 74(5–6):221–227, 2000.

[BF00] J.-P. Bodeveix and M. Filali. FMona: a tool for expressing validation techniques over infinite state systems. 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 204–219. Springer, 2000.

[BFLP03] Sébastien Bardin, Alain Finkel, Jérôme Leroux, and Laure 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.

[BGP97] Tevfik Bultan, Richard Gerber, and William Pugh. Symbolic model-checking of infinite state systems using Presburger arithmetic. In *Proc. 9th Int. Conf. Computer Aided Verification (CAV'97), Haifa, Israel, June 1997*, volume 1254 of *Lecture Notes in Computer Science*, pages 400–411. Springer, 1997.

[BGP99] Tevfik Bultan, Richard Gerber, and William 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.

[BH99] Ahmed Bouajjani and Peter Habermehl. Symbolic reachability analysis of FIFO-channel systems with nonregular sets of configurations. *Theoretical Computer Science*, 221(1–2):211–250, 1999.

[BHMV94] Véronique Bruyère, Georges Hansel, Christian Michaux, and Roger Villemaire. Logic and p-recognizable sets of integers. *Bull. Belg. Math. Soc.*, 1(2):191–238, March 1994.

[BLW03] Bernard Boigelot, Alexandre Legay, and Pierre Wolper. Iterating transducers in the large. In *Proc. 15th Int. Conf. Computer Aided Verification (CAV'2003), Boulder, CO, USA, July 2003*, volume 2725 of *Lecture Notes in Computer Science*, pages 223–235. Springer, 2003.

[Boi03] Bernard Boigelot. On iterating linear transformations over recognizable sets of integers. *Theoretical Computer Science*, 309(2):413–468, 2003.

[Bra] Brain homepage.
http://www.cs.man.ac.uk/˜voronkov/BRAIN/index.html.

[Bry92] Randal E. Bryant. Symbolic boolean manipulation with ordered binary-decision diagrams. *ACM Computing Surveys*, 24(3):293–318, 1992.

[Del00] Gorgio Delzanno. Automatic verification of parameterized cache coherence protocols. In *Proc. 12th Int. Conf. Computer Aided Verification (CAV'2000), Chicago, IL, USA, July 2000*, volume 1855 of *Lecture Notes in Computer Science*, pages 53–68. Springer, 2000.

[DFS98] Catherine Dufourd, Alain Finkel, and Philippe Schnoebelen. Reset nets between decidability and undecidability. In *Proc. 25th Int. Coll. Automata, Languages, and Programming (ICALP'98), Aalborg, Denmark, July 1998*, volume 1443 of *Lecture Notes in Computer Science*, pages 103–115. Springer, 1998.

[DJS99] Catherine Dufourd, Petr Jančar, and Philippe Schnoebelen. Boundedness of Reset P/T nets. In *Proc. 26th Int. Coll. Automata, Languages, and Programming (ICALP'99), Prague, Czech Republic, July 1999*, volume 1644 of *Lecture Notes in Computer Science*, pages 301–310. Springer, 1999.

[DRV01] Gorgio Delzanno, Jean-Francois Raskin, and Laurent Van Begin. Attacking symbolic state explosion. In *Proc. 13th Int. Conf. Computer Aided Verification (CAV'2001), Paris, France, July 2001*, volume 2102 of *Lecture Notes in Computer Science*, pages 298–310. Springer, 2001.

[EN98] E. Allen Emerson and Kedar S. Namjoshi. On model checking for nondeterministic infinite-state systems. In *Proc. 13th IEEE Symp. Logic in Computer Science (LICS'98), Indianapolis, IN, USA, June 1998*, pages 70–80. IEEE Comp. Soc. Press, 1998.

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

[FL02] Alain Finkel and Jérôme Leroux. How to compose Presburger-accelerations: Applications to broadcast protocols. In *Proc. 22nd Conf. Found. of Software Technology and Theor. Comp. Sci. (FST&TCS'2002), Kanpur, India, Dec. 2002*, volume 2556 of *Lecture Notes in Computer Science*, pages 145–156. Springer, 2002.

[FL04] Alain Finkel and Jérôme Leroux. Polynomial time image computation with interval-definable counters system. In *SPIN Model Checking and Software Verification, Proc. 11th Int. SPIN Workshop, Barcelona, Spain, Apr. 2004*, volume 2989 of *Lecture Notes in Computer Science*, pages 182–197. Springer, 2004.

[FO97] Laurent Fribourg and Hans 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.

[FPS00] Alain Finkel, S. Purushothaman Iyer, and Grégoire Sutre. Well-abstracted transition systems. In *Proc. 11th Int. Conf. Concurrency Theory (CONCUR'2000), University Park, PA, USA, Aug. 2000*, volume 1877 of *Lecture Notes in Computer Science*, pages 566–580. Springer, 2000.

[FS01] Alain Finkel and Phillipe Schnoebelen. Well structured transition systems everywhere! *Theoretical Computer Science*, 256(1–2):63–92, 2001.

[GS66] Seymour Ginsburg and Edwin H. Spanier. Semigroups, Presburger formulas and languages. *Pacific J. Math.*, 16(2):285–296, 1966.

[KMS02] Nils Klarlund, A. Møller, and M. I. Schwartzbach. MONA implementation secrets. *Int. J. of Foundations Computer Science*, 13(4):571–586, 2002.

[Las] LASH homepage.
http://www.montefiore.ulg.ac.be/˜boigelot/research/lash/.

[Ler03] Jérôme 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.

[Mon] MONA homepage. http://www.brics.dk/mona/index.html.

[MS77] Arnold Mandel and Imre Simon. On finite semigroups of matrices. *Theoretical Computer Science*, 5(2):101–111, October 1977.

[Reu89] Christophe Reutenauer. *Aspects Mathématiques des Réseaux de Petri*, chapter 3. Collection Études et Recherches en Informatique. Masson, Paris, 1989.

[Tre] TREX homepage. http://www.liafa.jussieu.fr/˜sighirea/trex/.

[WB00] Pierre Wolper and Bernard 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.