A Primitive Recursive Algorithm for the General Petri Net Reachability Problem

Zakariae Bouziane
IRISA / INRIA de Rennes, France
bouziane@irisa.fr

Abstract

In 1981, Mayr and Kosaraju proved the decidability of the general Petri net reachability problem. However, their algorithms are non primitive recursive. Since then the primitive recursiveness of this problem was stated as an open problem. In this paper, we give a double exponential space algorithm for the general Petri net reachability problem.

1 Introduction

Many aspects of the fundamental nature of computation are often studied via formal models, such as Turing machines, finite-state machines, and push-down automata. One formalism that has been used to model parallel computations is the Petri net [7, 8]. As a means of gaining a better understanding of the Petri net model, the decidability and computational complexity of typical automata theoretic problems concerning Petri nets have been examined. Examples of such problems include deadlock freedom, and liveness of the system. Solutions to these example problems proclaim, in a sense, the absence of "difficulties" for all states that are reachable in the system.

Another question is whether some arbitrary state can be reached from a fixed initial state. The later, the so-called general reachability problem, is of basic importance for many others. It is recursively equivalent to the liveness problem [2]. Moreover, a number of other problems in the representation of parallel and concurrent systems, in language generating systems, in algebra and in number theory can be shown to be effectively reducible or equivalent to the reachability problem.

Cardoza, Lipton, Mayr and Meyer [1, 6] have shown exponential space lower bound for the reachability problem, but the only known algorithm is non-primitive recursive [5, 4]. Even the decidability of this problem was an open question for many years.

In this paper, we give a double exponential space algorithm for the general Petri net reachability problem. Our proof is based on the reduction of this later problem to the problem of solving a polynomial equation (Section 2). Then we decompose the polynomial equation into a system of two simple equations (Lemma 3.1). We generate the set of solutions of the first equation (Lemma 3.2) and we extract from it those that verify the second one (Lemma 3.3). Finally, we prove that minimal solutions are at most of double exponential size (Lemma 3.4). This leads to a double exponential bound for the general Petri net reachability problem (Section 4).

2 Petri nets and polynomial equations

As usual, $\mathbb{N}$ denotes the set of nonnegative integers and $\mathbb{N}^m$ denotes the m-dimensional column vectors of natural numbers. For any integers $a, b \in \mathbb{N}$ such that $a \leq b$, the interval $[a, b]$ denotes the set $\{a, a+1, \ldots, b\}$. For any vectors $V_1, V_2 \in \mathbb{N}^m$ we write:

- $V_1 = V_2$ if and only if $V_1(i) = V_2(i)$, for every $i \in \{1, 2, \ldots, m\}$,
- $V_1 + V_2$ to denote the vector of $\mathbb{N}^m$ whose i^{th} component is $V_1(i) + V_2(i)$,
- $\sup(V_1, V_2)$ to denote the vector of $\mathbb{N}^m$ whose i^{th} component is $\max(V_1(i), V_2(i))$ and
- $\|V_1\|_\infty = \max\{V_1(i) \mid i \in [1, m]\}$.

A *Petri net* is a tuple $N = (P, T, F, M_0)$ where, $P = \{p_1, p_2, \ldots, p_m\}$ is a finite set of *places*, $T = \{t_1, t_2, \ldots, t_n\}$ is a finite set of *transitions*, $F : (P \times T) \cup (T \times P) \longrightarrow \mathbb{N}$ is a *flow function* [1], and $M_0 \in \mathbb{N}^m$ is an *initial marking*. In all this paper m will designate the number of places and n the number of transitions of the Petri net.

[1] We write $F(., t)$ (resp. $F(t, .)$) to denote the m-dimensional column vector whose i^{th} component is $F(p_i, t)$ (resp. $F(t, p_i)$).

Let $N = (P, T, F, M_0)$ be a Petri net and $M_f \in \mathbb{N}^m$ be a marking. M_f is said to be *reachable* in N, and we note it $M_f \in \mathcal{RS}(N)$, if there exist a finite set of vectors $H_1, H_2, \ldots, H_k \in \mathbb{N}^m$ and a sequence of transitions $t_{i_1}, t_{i_2}, \ldots, t_{i_k} \in T$ such that

$$\begin{cases} M_0 = H_1 + F(., t_{i_1}) \\ \\ H_j + F(t_{i_j}, .) = \\ \quad H_{j+1} + F(., t_{i_{j+1}}), \quad \forall j \in [1, k-1] \\ \\ H_k + F(t_{i_k}, .) = M_f \end{cases} \tag{1}$$

Note that system (1) has a solution if and only if it has a solution such that, for every $j, j' \in [1, k]$, $j \neq j'$, we have $M_0 \neq H_j + F(t_{i_j}, .) \neq H_{j'} + F(t_{i_{j'}}, .)$.

The **general Petri net reachability problem** is the problem of deciding for a given a Petri net $N = (P, T, F, M_0)$ and a marking $M_f \in \mathbb{N}^m$ whether $M_f \in \mathcal{RS}(N)$.

Our computation of an upper bound for the reachability problem is based on a reduction of this later to the problem of solving a polynomial equation. For that we need to define the polynomials associated to a Petri net. We first code the set of vectors into integers by the help of the *Gödel coding injective function* φ defined by $\varphi(V) = 2^{V(1)}.3^{V(2)}.5^{V(3)} \ldots q_m^{V(m)}$ where q_m is the m^{th} prime number. Then, in order to simplify the notations, we put $\alpha_0 = \varphi(M_0)$, $\beta_0 = \varphi(M_f)$ and, $\beta_i = \varphi(F(., t_i))$ and $\alpha_i = \varphi(F(t_i, .))$, for $i \in [1, n]$. Finally, let $\mathbf{I}[X]$ be the set of polynomials over one variable X and whose coefficients are in $\{0, 1\}$. All the polynomials we will deal with in this paper are in $\mathbf{I}[X]$.

The reduction from the reachability problem to the problem of solving a polynomial system is given by the following lemma.

Lemma 2.1 *$M_f \in \mathcal{RS}(N)$ if and only if there exists a finite set of integers $h_{i,j} \in \mathbb{N}$ such that the following polynomial equation is verified*

$$\begin{cases} X^{\alpha_0} + \sum_{i=1}^{n} \sum_{j=1}^{k_i} X^{h_{i,j}.\alpha_i} = \\ \\ \quad X^{\beta_0} + \sum_{i=1}^{n} \sum_{j=1}^{k_i} X^{h_{i,j}.\beta_i} \quad \in \mathbf{I}[X] \end{cases} \tag{2}$$

Proof. For every vectors $V_1, V_2, V_3 \in \mathbb{N}^m$, we have

$$[V_1 = V_2 + V_3] \Longleftrightarrow [\varphi(V_1) = \varphi(V_2).\varphi(V_3)]$$

Thus, for every vectors $V_1, V_2, V_3 \in \mathbb{N}^m$, we have

$$[V_1 = V_2 + V_3] \Longleftrightarrow [X^{\varphi(V_1)} = X^{\varphi(V_2).\varphi(V_3)}]$$

If we put $l_i = \varphi(H_i)$, for $i \in [1, k]$, then, from the reachability definition, we deduce that $M_f \in \mathcal{RS}(N)$ if and only if we have

$$\begin{cases} \alpha_0 = l_1.\beta_{i_1} \\ \\ l_j.\alpha_{i_j} = l_{j+1}.\beta_{i_{j+1}} \quad \forall j \in [1, k-1] \\ \\ l_k.\alpha_{i_k} = \beta_0 \end{cases}$$

Hence, $M_f \in \mathcal{RS}(N)$ if and only if we have

$$\begin{cases} X^{\alpha_0} = X^{l_1.\beta_{i_1}} \\ \\ X^{l_j.\alpha_{i_j}} = X^{l_{j+1}.\beta_{i_{j+1}}} \quad \forall j \in [1, k-1] \\ \\ X^{l_k.\alpha_{i_k}} = X^{\beta_0} \end{cases}$$

We can choose $\alpha_0 \neq l_j.\alpha_{i_j} \neq l_{j'}.\alpha_{i_{j'}}$, for every $j, j' \in [1, k]$ (This is equivalent to $M_0 \neq H_j + F(t_{i_j}, .) \neq H_{j'} + F(t_{i_{j'}}, .)$).

Thus, $M_f \in \mathcal{RS}(N)$ if and only if we have

$$X^{\alpha_0} + \sum_{j=1}^{k} X^{l_j.\alpha_{i_j}} = X^{\beta_0} + \sum_{j=1}^{k} X^{l_j.\beta_{i_j}} \qquad \in \mathbf{I}[X]$$

■

The general Petri net reachability problem is then equivalent to decide whether the polynomial system (2) has a solution. Solving this later problem is the subject of the next section.

3 Solution of the polynomial equation

From lemma 2.1 we deduce that the reachability problem is equivalent to the problem of deciding whether system (2) has a solution. In this section we show that this later is decidable. Our proof consists in reducing system (2) into an equivalent system made of two equations (lemma 3.1). Then we show that the set of solutions of the first one can be effectively constructed (lemma 3.2). Finally, we construct this set and extract from it the solutions that also verify the second equation (lemma 3.3 and lemma 3.4).

Lemma 3.1 *Equation (2) has a solution if and only if there exist two finite sets of integers $f_{i,j}, g_{i,j} \in \mathbb{N}$ such that the two following polynomial equations are verified*

$$\left\{ \begin{array}{l} X^{\alpha_0} + \sum_{i=1}^{n} \sum_{j=1}^{k_i} X^{f_{i,j}.\alpha_i} = \\ \quad X^{\beta_0} + \sum_{i=1}^{n} \sum_{j=1}^{k'_i} X^{g_{i,j}.\beta_i} \qquad \in \mathbf{I}[X] \end{array} \right. \tag{3}$$

and

$$\sum_{j=1}^{k_i} (X^{\beta_i})^{f_{i,j}.\alpha_i} = \sum_{j=1}^{k'_i} (X^{\alpha_i})^{g_{i,j}.\beta_i} \quad \forall i \in [1,n] \tag{4}$$

Proof. As for fixed i, the $f_{i,j}$ (resp. the $g_{i,j}$) are all distinct, equation (4) is equivalent to the proposition:

$$\forall i \in [1,n] \; ; \; \left\{ \begin{array}{l} k_i = k'_i \quad \text{and} \\ \\ \{f_{i,1}, \ldots, f_{i,k_i}\} = \{g_{i,1}, \ldots, g_{i,k_i}\} \end{array} \right.$$

Thus, solving the system made of equation (3) and equation (4) is equivalent to solving equation (2). ■

In order to compute the set of solutions of equation (3) we first define a set of generator for it:

Let E_0 be the set of tuples $(P_1, \ldots, P_{2.n})$ such that, for every $i \in [1,n]$, $P_i \in \mathbf{I}[X^{\alpha_i}]$ and $P_{n+i} \in \mathbf{I}[X^{\beta_i}]$, and $X^{\alpha_0} + \sum_{i=1}^{n} P_i(X) = X^{\beta_0} + \sum_{i=n+1}^{2.n} P_i(X) \in \mathbf{I}[X]$ [2]. E_0 is then the set of solutions of equation (3).

Let $l_0 = lcm\{\alpha_i, \beta_i \mid i \in [1,n]\}$ [3] and $l_1 = \min\{\lambda.l_0 \mid \lambda \in \mathbb{N} \, , \, \max(\alpha_0, \beta_0) < \lambda.l_0\}$.

Let E_1 be the set of tuples $(Q_1, \ldots, Q_{2.n}) \in E_0$ such that, for every $i \in [1, 2.n]$, $\deg(Q_i) < l_1$. Note that E_1 is finite.

Let E_2 be the set of tuples $(R_1, \ldots, R_{2.n})$ such that for every $i \in [1,n]$, $R_i \in \mathbf{I}[X^{\alpha_i}]$, $R_{n+i} \in \mathbf{I}[X^{\beta_i}]$ and $\max\{\deg(R_i), \deg(R_{n+i})\} < l_0$, and $\sum_{i=1}^{n} R_i(X) = \sum_{i=n+1}^{2.n} R_i(X)$. E_2 is also finite. Let $k \in \mathbb{N}$ be the number of its elements. We can write $E_2 = \{(R_1^1, \ldots, R_{2.n}^1), \ldots, (R_1^k, \ldots, R_{2.n}^k)\}$.

Lemma 3.2 shows that the infinite set E can be completely determined by giving the two finite and easy to compute sets E_1 and E_2.

Lemma 3.2 *E_0 is equal to the set of tuples $(P_1, \ldots, P_{2.n})$ such that*

$$\left\{ \begin{array}{l} (P_1, \ldots, P_{2.n}) = (Q_1, \ldots, Q_{2.n}) + \\ \quad X^{l_1}. \sum_{i=1}^{k} .S_i(X).(R_1^i, \ldots, R_{2.n}^i) \end{array} \right.$$

for some $(Q_1, \ldots, Q_{2.n}) \in E_1$ and $S_1, \ldots, S_k \in \mathbf{I}[X]$ such that $\sum_{i=1}^{k} .S_i(X) = \sum_{i=0}^{a} X^{i.l_0}$, for some $a \in \mathbb{N}$.

Proof.

- For every $(Q_1, \ldots, Q_{2.n}) \in E_1$ and for every $l \in [1,n]$ we have $Q_l \in \mathbf{I}[X^{\alpha_l}]$, $Q_{n+l} \in \mathbf{I}[X^{\beta_l}]$ and every monomial X^c in $Q_l(X)$ or in $Q_{n+l}(X)$ verifies $0 \leq c < l_1$.

  For every $i \in [1,k]$ and for every $l \in [1,n]$ we have $R_l^i \in \mathbf{I}[X^{\alpha_l}]$, $R_{n+l}^i \in \mathbf{I}[X^{\beta_l}]$ and every monomial X^c in $R_l^i(X)$ or in $R_{n+l}^i(X)$ verifies $0 \leq c < l_0$.

  However, from the definitions of l_0 and l_1 we deduce that, for every $l \in [1,n]$ there exist $c_1, c_2, c_3, c_4 \in \mathbb{N}$ such that $l_0 = c_1.\alpha_l = c_2.\beta_l$ and $l_1 = c_3.\alpha_l = c_4.\beta_l$.

  Hence, for every $i \in [1,k]$, for every $l \in [1,n]$ and for every $b \in \mathbb{N}$ we have $X^{l_1}.X^{b.l_0}.R_l^i(X) \in \mathbf{I}[X^{\alpha_l}]$, $X^{l_1}.X^{b.l_0}.R_{n+l}^i(X) \in \mathbf{I}[X^{\beta_l}]$ and every monomial X^c in $X^{l_1}.X^{b.l_0}.R_l^i(X)$ or in $X^{l_1}.X^{b.l_0}.R_{n+l}^i(X)$ verifies $l_1 + b.l_0 \leq c < l_1 + (b+1).l_0$.

  Consequently, for every $l \in [1,n]$ we have $Q_l(X) + X^{l_1}. \sum_{i=1}^{k} .S_i(X).R_l^i(X) \in \mathbf{I}[X^{\alpha_l}]$ and $Q_{n+l}(X) +$

[2] $\mathbf{I}[X^c]$ is the subset of $\mathbf{I}[X]$ whose every monomial X^a in every polynomial P is of degree $deg(X^a) = a = d.c$ for some constant d.

[3] lmc is the least common multiple

$X^{l_1}.\sum_{i=1}^{k}.S_i(X).R^i_{n+l}(X) \in \mathbf{I}[X^{\beta_l}]$. Thus the sum

$$(Q_1,\ldots,Q_{2.n}) + X^{l_1}.\sum_{i=1}^{k}.S_i(X).(R^i_1,\ldots,R^i_{2.n})$$

is in E_0.

- Let $(P_1,\ldots,P_{2.n}) \in E_0$ and let $d = \max\{\deg(P_l) \mid l \in [1,2.n]\}$. For every $l \in [1,2.n]$, let Q_l to be the largest sub-polynomial of P_l of degree strictly less than l_1, and let S^i_l to be the largest sub-polynomial of P_l such that any monomial X^c of S^i_l verifies $l_1 + (i-1).l_0 \le c < l_1 + i.l_0$ (i is such that $l_1 + (i-1).l_0 \le d$ and $i \ge 1$).

  The tuple $(Q_1,\ldots,Q_{2.n})$ is then in E_1.

  For $l \in [1,n]$ we have $P_l \in \mathbf{I}[X^{\alpha_l}]$. Hence every monomial X^c of S^i_l verifies

  $l_1+(i-1).l_0 \le c = \lambda.\alpha_l < l_1+i.l_0$, for some $\lambda \in \mathbb{N}$

  Moreover, from the definition of l_0 and l_1, we deduce that $l_0 = \lambda_0.\alpha_l$ and $l_1 = \lambda_1.\alpha_l$, for some $\lambda_0, \lambda_1 \in \mathbb{N}$. Thus

  $$(\lambda_1 + (i-1).\lambda_0).\alpha_l \le c = \lambda.\alpha_l < (\lambda_1 + i.\lambda_0).\alpha_l$$

  Which means that

  $$X^c = X^{l_1}.X^{(i-1).l_0}.X^{b.\alpha_l} \text{ for some } b.\alpha_l < l_0$$

  In the case where $l \in [n+1,2.n]$, we use the same proof by replacing α_l by β_{l-n}.

  Consequently, if we let $R^i_l(X) = \frac{S^i_l(X)}{X^{l_1+(i-1).l_0}}$, we will have $(R^i_1,\ldots,R^i_{2.n}) \in E_2$. Thus $(P_1,\ldots,P_{2.n})$ can be written in the form of the equation in lemma 3.2. ∎

We are thus ready to compute the set of solutions of equation (2) as the set of solutions of equation (3) that also verify equation (4).

Lemma 3.3 *Equation (2) has a solution if and only if there exist $S_1,\ldots,S_k \in \mathbf{I}[X]$ such that the two following systems are verified*

$$\left\{\begin{array}{l} Q_l(2^{\beta_l}) + 2^{l_1.\beta_l}.\sum_{i=1}^{k} S_i(2^{\beta_l}).R^i_l(2^{\beta_l}) = \\ \qquad Q_{n+l}(2^{\alpha_l}) + 2^{l_1.\alpha_l}.\sum_{i=1}^{k} S_i(2^{\alpha_l}).R^i_{n+l}(2^{\alpha_l}) \end{array}\right. \tag{5}$$

for every $l \in [1,n]$, and

$$\sum_{i=1}^{k} S_i(X) = \sum_{i=0}^{a} X^{i.l_0} \tag{6}$$

for some $a \in \mathbb{N}$ and $(Q_1,\ldots,Q_{2.n}) \in E_1$.

Proof. From lemma 3.1 and lemma 3.2, we deduce that equation (2) has a solution if and only if there exist $S_1,\ldots,S_k \in \mathbf{I}[X]$ and $a \in \mathbb{N}$ such that $\sum_{i=1}^{k} S_i(X) = \sum_{i=0}^{a} X^{i.l_0}$, and for every $l \in [1,n]$,

$$\left\{\begin{array}{l} Q_l(X^{\beta_l}) + X^{l_1.\beta_l}.\sum_{i=1}^{k} S_i(X^{\beta_l}).R^i_l(X^{\beta_l}) = \\ \qquad Q_{n+l}(X^{\alpha_l}) + X^{l_1.\alpha_l}.\sum_{i=1}^{k} S_i(X^{\alpha_l}).R^i_{n+l}(X^{\alpha_l}) \end{array}\right.$$

However, for $l \in [1,n]$, the polynomials $Q_l(X^{\beta_l}) + X^{l_1.\beta_l}.\sum_{i=1}^{k} S_i(X^{\beta_l}).R^i_l(X^{\beta_l})$ and $Q_{n+l}(X^{\alpha_l}) + X^{l_1.\alpha_l}.\sum_{i=1}^{k} S_i(X^{\alpha_l}).R^i_{n+l}(X^{\alpha_l})$ are in $\mathbf{I}[X]$. Hence, these two polynomials are equal if and only if they have the same value on integer 2. This ends the proof. ∎

Lemma 3.4 gives a double exponential upper bound for the degree of a minimal solution of equation (2) (or equivalently the degree of minimal solution of the system made of equation (5) and equation (6)).

Lemma 3.4 *The system made of equation (5) and equation (6) has a solution if and only if it has a solution with $a \le 2^{2^{c.n.l_1}}$, for some constant $c \in \mathbb{N}$ independent of the system.*

Proof. For $i \in [1,k]$ and $l \in [1,n]$, let

- $A_{i,l} = 2^{l_1.\beta_l}.R^i_l(2^{\beta_l})$,
- $A_{i,n+l} = 2^{l_1.\alpha_l}.R^i_{n+l}(2^{\alpha_l})$,
- $A_{0,l} = Q_l(2^{\beta_l})$ and

- $A_{0,n+l} = Q_{n+l}(2^{\alpha_l})$.

Let $\Sigma = \{Z \in \mathbb{N}^k \mid \sum_{i=1}^{k} Z(i) = 1\}$.

Hence, equation (5) is equivalent to the linear diophantine system

$$\left\{ \begin{array}{l} A_{0,l} + \sum_{i=1}^{k} A_{i,l}.Y_{i,l} = \\ \quad A_{0,n+l} + \sum_{i=1}^{k} A_{i,n+l}.Y_{i,n+l} \quad \forall l \in [1,n] \end{array} \right. \tag{7}$$

and equation (6) is equivalent to the system

$$\left\{ \begin{array}{l} Y_{i,l} = Z_0(i) + 2^{l_0.\beta_l}.Z_1(i) + \ldots + 2^{a.l_0.\beta_l}.Z_a(i) \\ \qquad , \forall i \in [1,k] \, , \forall l \in [1,n] \\ \\ Y_{i,n+l} = Z_0(i) + 2^{l_0.\alpha_l}.Z_1(i) + \ldots + 2^{a.l_0.\alpha_l}.Z_a(i) \\ \qquad , \forall i \in [1,k] \, , \forall l \in [1,n] \end{array} \right. \tag{8}$$

for some $Z_0, \ldots, Z_a \in \Sigma$.

However, for every $l \in [1,n]$, the set of solutions

$$\left\{ \begin{array}{l} \mathcal{S}_l = \{(Y_{1,l}, \ldots, Y_{k,l}, Y_{1,n+l}, \ldots, Y_{k,n+l}) \in \mathbb{N}^{2.k} \mid \\ A_{0,l} + \sum_{i=1}^{k} A_{i,l}.Y_{i,l} = A_{0,n+l} + \sum_{i=1}^{k} A_{i,n+l}.Y_{i,n+l}\} \end{array} \right.$$

is semi-linear. Moreover, if we let $\mathcal{S}_{l,0} \subset \mathbb{N}^{2.k}$ to be set of *minimal solutions* of the l^{th} equation in system (7) and $\mathcal{S}_{l,1} \subset \mathbb{N}^{2.k}$ is the set of *minimal solutions* of its homogeneous part $\sum_{i=1}^{k} A_{i,l}.Y_{i,l} = \sum_{i=1}^{k} A_{i,n+l}.Y_{i,n+l}$ then we have

$$\left\{ \begin{array}{l} \mathcal{S}_l = \{V_l + \sum_{j=1}^{h} \lambda_j.W_{j,l} \mid V_l \in \mathcal{S}_{l,0} \\ \quad , W_{1,l}, \ldots, W_{h,l} \in \mathcal{S}_{l,1} \text{ and } \lambda_1, \ldots, \lambda_h \in \mathbb{N}\} \end{array} \right.$$

We only want the solutions $U_l = (Y_{1,l}, \ldots, Y_{k,l}, Y_{1,n+l}, \ldots, Y_{k,n+l}) \in \mathcal{S}_l$ that verify system (8). Let $U_{l,0} = V_{l,0} + \sum_{j=1}^{h} \lambda_j.W_{j,l}$ be such a solution. It can be written in the form

$$\left\{ \begin{array}{l} U_{l,0} = V_{l,0} + \sum_{j=1}^{h} \lambda_{j,0}.W_{j,l} + 2^{l_0.\gamma_l}.\sum_{j=1}^{h} \lambda_{j,1}.W_{j,l} + \\ \quad \cdots + 2^{a.l_0.\gamma_l}.\sum_{j=1}^{h} \lambda_{j,a}.W_{j,l} \end{array} \right.$$

where, $\gamma_l = \max\{\alpha_l, \beta_l\}$ and $0 \leq \lambda_{j,i} < 2^{l_0.\gamma_l}$.

Thus, for every $b \in [0, a-1]$ and for every $i \in [1,k]$, we have:

$$\left\{ \begin{array}{l} V_{l,0}(i) + \sum_{c=0}^{b} 2^{c.l_0.\gamma_l}.\sum_{j=1}^{h} \lambda_{j,c}.W_{j,l}(i) = \\ \quad Z_0(i) + 2^{l_0.\beta_l}.Z_1(i) + \cdots + 2^{b.l_0.\beta_l}.Z_b(i) + \\ \quad 2^{(b+1).l_0.\beta_l}.V_{l,b+1}(i) \\ \\ V_{l,0}(k+i) + \sum_{c=0}^{b} 2^{c.l_0.\gamma_l}.\sum_{j=1}^{h} \lambda_{j,c}.W_{j,l}(k+i) = \\ \quad Z_0(i) + 2^{l_0.\alpha_l}.Z_1(i) + \cdots + 2^{b.l_0.\alpha_l}.Z_b(i) + \\ \quad 2^{(b+1).l_0.\alpha_l}.V_{l,b+1}(k+i) \end{array} \right. \tag{9}$$

for some $V_{l,1}, \ldots, V_{l,a} \in \mathbb{N}^{2.k}$ (note that $V_{l,b}$ is not necessarily in $\mathcal{S}_{l,0}$).

Let, without loss of generalities, $\gamma_l = \beta_l$, then we have:

$$\left\{ \begin{array}{l} \max\{V_{l,b+1}(i) \mid i \in [1,k]\} \leq \|V_{l,0}\|_\infty + \\ \quad h.2^{l_0.\gamma_l}.\max\{\|W_{j,l}\|_\infty \mid j \in [1,h]\} \end{array} \right.$$

However, for every $i \in [1,k]$, we have $V_{l,a}(i) + \sum_{j=1}^{h} \lambda_{j,a}.W_{j,l}(i) = V_{l,a}(k+i) + \sum_{j=1}^{h} \lambda_{j,a}.W_{j,l}(k+i) = Z_a(i)$.

If we put m_1 to be the maximum number of digits in the representation of $V_{l,b}(i)$ in numeral base $2^{l_0.\beta_l}$ (for $b \in [0,a]$ and $i \in [1,k]$) and we put m_2 to be the maximum number of digits in the representation of $V_{l,b}(k+i)$ in numeral base $2^{l_0.\alpha_l}$ (for $b \in [0,a]$ and $i \in [1,k]$) then $m_1 = m_2$. Thus, for every $b \in [1,a]$, we have:

$$\|V_{l,b+1}\|_\infty \leq \|V_{l,0}\|_\infty + h.2^{l_0.\gamma_l}.\max\{\|W_{j,l}\|_\infty \mid j \in [1,h]\}$$

However, from [3], we deduce that, for every $W_{j,l} \in \mathcal{S}_{l,1}$ and for every $V_{l,0} \in \mathcal{S}_{l,0}$, we have

$$\begin{cases} \|W_{j,l}\|_\infty & \leq & \max\{A_{i,l}, A_{i,n+l} \mid i \in [1,k]\} \\ & \leq & 2^{\gamma_l.(l_0+l_1)} \\ \|V_{l,0}\|_\infty & \leq & 2.\max\{A_{i,l}, A_{i,n+l} \mid i \in [0,k]\} \\ & \leq & 2^{\gamma_l.(l_0+l_1)+1} \end{cases}$$

Hence, the number h of elements in $\mathcal{S}_{l,1}$ verifies

$$h \leq 2^{2.k.\gamma_l.(l_0+l_1)}$$

However, the number k of elements in E_2 verifies

$$k \leq 2^{n.l_0}$$

Consequently,

$$h \leq 2^{2.\gamma_l.(l_0+l_1).2^{n.l_0}} \leq 2^{2^{c_0.n.l_1}}$$

for some constant $c_0 \in \mathbb{N}$.

Which means that

$$\|V_{l,j}\|_\infty \leq 2^{2^{c_1.n.l_1}}$$

for some constant $c_1 \in \mathbb{N}$.

Thus, the elements of $\mathcal{S}_l$ that verify equation (8) can be generated by the union of $|\mathcal{S}_{l,0}|$ automaton $\mathcal{A}_l = \bigcup\{\mathcal{A}_l(V_{l,i}) \mid V_{l,i} \in \mathcal{S}_{l,0}\}$. Each automata $\mathcal{A}_l(V_{l,i})$ is over the alphabet Σ and has nodes in the set of vectors

$$\{V \in \mathbb{N}^k \mid \|V\|_\infty \leq 2^{2^{c_1.n.l_1}}\}$$

the initial marking is $V_{l,i}$ the final marking is **0** $= (0,\ldots,0)$ and the arcs are constructed in the following way: for any set of vectors $V_{l,0}, V_{l,1}, \ldots, V_{l,a} \in \mathbb{N}^{2.k}$ that verify equation (9), we put an arc labeled Z_i from $V_{l,i}$ to $V_{l,i+1}$, for $i \in [0, a-1]$, and an arc labeled Z_a from $V_{l,a} = Z_a$ to **0**.

Finally, the set of solutions of equation (7) that verify equation (8) is completely determined by the intersection

$$\mathcal{L} = \bigcap_{l=1}^{n} \mathcal{L}(\mathcal{A}_l)$$

where $\mathcal{L}(\mathcal{A}_l)$ is the language recognized by $\mathcal{A}_l$.

The number of nodes of every automaton $\mathcal{A}_l(V_{l,j})$ is less than $2^{2^{c_2.n.l_1}}$, for some constant c_2. Hence, $\mathcal{L} \neq \emptyset$ if and only if it contains a sequence of length no longer than $2^{n.2^{c_2.n.l_1}} \leq 2^{2^{c_3.n.l_1}}$, for some constant $c_3 \in \mathbb{N}$. Thus, the system made of system (7) and system (8) has a solution if and only if it has a solution with $a < 2^{2^{c.n.l_1}}$, for some constant $c \in \mathbb{N}$. ■

4 The general Petri net reachability problem is primitive recursive

Now, we are ready to compute a primitive recursive algorithm for the general Petri net reachability problem.

Theorem 4.1 *There exists a constant $c \in \mathbb{N}$ such that, for every Petri net $N = (P, T, F, M_0)$ and every marking $M_f \in \mathbb{N}^m$, we have $M_f \in \mathcal{RS}(N)$ if and only if there exist $H_1, H_2, \ldots, H_k \in \mathbb{N}^m$ and $t_{i_1}, t_{i_2}, \ldots, t_{i_k} \in T$ such that*

$$\begin{cases} M_0 = H_1 + F(.,t_{i_1}) \\ H_j + F(t_{i_j},.) = H_{j+1} + F(.,t_{i_{j+1}}) \quad \forall j \in [1, k-1] \\ M_f = H_k + F(t_{i_k},.) \end{cases}$$

and the the length k of the firing sequence verifies

$$k \leq 2^{2^{c.n.l}}$$

where $\alpha_0 = \varphi(M_0)$, $\beta_0 = \varphi(M_f)$ *and* $l = \min\{\lambda.l_0 \mid \lambda \in \mathbb{N}, \max\{\alpha_0, \beta_0\} < \lambda.l_0\}$ *for* $l_0 = \varphi(\sup\{F(t_i,.), F(.,t_i) \mid i \in [1,n]\})$.

Proof. This theorem is a consequence of the previous section. ■

As an immediate corollary of this theorem, we have:

Corollary 4.1 *The general Petri net reachability problem is decidable in double exponential space.*

5 Conclusions

The general Petri net reachability problem is shown to be primitive recursive. This result implies the primitive recursiveness of many other problems, such as the liveness and the deadlock freedom problems.

However, our bound is not proved optimal. Hence, the reduction of the gap between our upper bound and Cardoza, Lipton, Mayr and Meyer's lower bound [1, 6] is still an open problem.

Acknowledgment

Many thanks to Jean-Luc Lambert and Philippe Darondeau for their careful reading of the first versions of this work. They point me out to diferent reductions of the proofs.

References

[1] **Cardoza E., Lipton R., Meyer A.** *"Exponential space complete problems for Petri nets and commutative semi-groups"*, Proc. of the 8^{th} annual ACM Symposium on Theory of Computing, pp. 50-54, STOC'1976.

[2] **Hack M.** *"The recursive equivalence of the liveness problem and the reachability problem for Petri nets and vector addition systems"*, Proc. of the 15^{th} Annual Symposium on SWAT, New Orleans, LA, pp. 156-164, SWAT'1974.

[3] **Huet G.** *"An algorithm to generate the basis of solutions to homogeneous linear diophantine equations"*, Inf. Proc. Let. 7 (3), pp144-147, 1978.

[4] **Kosaraju R.** *"Decidability of reachability in Vector Addition Systems"*, Proc. of the 14^{th} Annual ACM Symposium on Theory of Computing pp. 267-280, STOC'1982.

[5] **Mayr E.** *"An algorithm for the general Petri net reachability problem"*, Proc. of the 13^{th} Annual ACM Symposium on Theory of Computing pp. 238-246, STOC'1981.
Also SIAM J. Compu. 13, 3, pp. 441-46-, 1984.

[6] **Mayr E., Meyer R.** *"The complexity of the word problems for commutative semi-groups and polynomial ideals"*, Advances in Mathematics 46, pp. 305-329, 1982.

[7] **Peterson J.** *"Petri net theory and the modeling of systems"*, Prentice-Hall, Inc.,1981.

[8] **Reisig W.** *"Petri nets : an introduction"*, Springer, Heidelberg, 1985.