



HAL
open science

Acceleration For Presburger Petri Nets

Jérôme Leroux

► **To cite this version:**

Jérôme Leroux. Acceleration For Presburger Petri Nets. VPT@CAV, 2013, Saint Petersburg, Russia. pp.10-12. hal-00959699

HAL Id: hal-00959699

<https://hal.science/hal-00959699v1>

Submitted on 15 Mar 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Acceleration For Petri Nets

Jérôme Leroux*

LaBRI, Université de Bordeaux, CNRS

Abstract

The reachability problem for Petri nets is a central problem of net theory. The problem is known to be decidable by inductive invariants definable in the Presburger arithmetic. When the reachability set is definable in the Presburger arithmetic, the existence of such an inductive invariant is immediate. However, in this case, the computation of a Presburger formula denoting the reachability set is an open problem. Recently this problem got closed by proving that if the reachability set of a Petri net is definable in the Presburger arithmetic, then the Petri net is flat, i.e. its reachability set can be obtained by runs labeled by words in a bounded language. As a direct consequence, classical algorithms based on acceleration techniques effectively compute a formula in the Presburger arithmetic denoting the reachability set.

1 Introduction

Petri Nets are one of the most popular formal methods for the representation and the analysis of parallel processes [1]. The reachability problem is central since many computational problems (even outside the realm of parallel processes) reduce to this problem. Sacerdote and Tenney provided in [2] a partial proof of decidability of this problem. The proof was completed in 1981 by Mayr [3] and simplified by Kosaraju [4] from [2, 3]. Ten years later [5], Lambert provided a further simplified version based on [4]. This last proof still remains difficult and the upper-bound complexity of the corresponding algorithm is just known to be non-primitive recursive. Nowadays, the exact complexity of the reachability problem for Petri nets is still an open-question. Even an Ackermannian upper bound is open (this bound holds for Petri nets with finite reachability sets [6]).

Basically, a Petri net is a pair $(T, \mathbf{c}_{\text{init}})$ where $T \subseteq \mathbb{N}^d \times \mathbb{N}^d$ is a finite set of *transitions*, and $\mathbf{c}_{\text{init}} \in \mathbb{N}^d$ is the *initial configuration*. A vector $\mathbf{c} \in \mathbb{N}^d$ is called a *configuration*. Given a transition $t = (\mathbf{a}, \mathbf{a}')$, we introduce the binary relations \xrightarrow{t} over the configurations defined by $\mathbf{x} \xrightarrow{t} \mathbf{x}'$ if there exists $\mathbf{v} \in \mathbb{N}^d$ such that $\mathbf{x} = \mathbf{a} + \mathbf{v}$ and $\mathbf{x}' = \mathbf{a}' + \mathbf{v}$. A *run* from a configuration \mathbf{x} to a configuration \mathbf{x}' labelled by a word $\sigma = t_1 \dots t_k$ of transitions $t_j \in T$ is a sequence $(\mathbf{c}_0, t_1, \mathbf{c}_1, \dots, t_k, \mathbf{c}_k)$ where $\mathbf{c}_0, \dots, \mathbf{c}_k$ are configurations such that $\mathbf{c}_0 = \mathbf{x}$, $\mathbf{c}_k = \mathbf{x}'$, and such that $\mathbf{c}_{j-1} \xrightarrow{t_j} \mathbf{c}_j$ for every $1 \leq j \leq k$. The *reachability set* is the set of configuration \mathbf{c} such that that there exists a run from the initial configuration \mathbf{c}_{init} to \mathbf{c} .

Example 1.1. *The Petri net depicted in Figure 1 was introduced in [7] as an example of Petri net having a reachability set which cannot be defined by a formula in the logic $\text{FO}(\mathbb{N}, +)$, called the Presburger arithmetic. In fact, the reachability set is equal to:*

$$\left\{ (p_1, p_2, p_3, p_4, p_5) \in \mathbb{N}^5 \mid \begin{array}{l} (p_1 = 1 \wedge p_4 = 0 \wedge 1 \leq p_2 + p_3 \leq 2^{p_5}) \vee \\ (p_1 = 0 \wedge p_4 = 1 \wedge 1 \leq p_2 + 2p_3 \leq 2^{p_5+1}) \end{array} \right\}$$

*Work funded by ANR grant REACHARD-ANR-11-BS02-001.

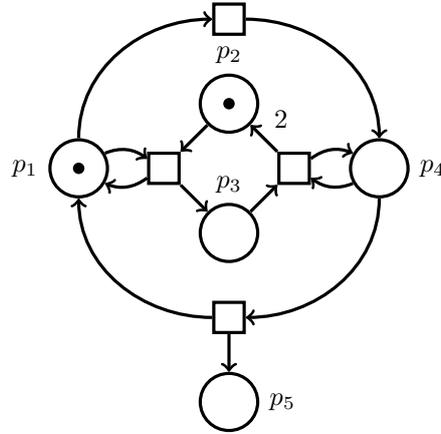


Figure 1: The Hopcroft and Pansiot net.

Recently, in [8], the reachability sets of Petri nets were proved to be *almost semilinear*, a class of sets that extends the class of Presburger sets (the sets definable in $\text{FO}(\mathbb{N}, +)$) inspired by the *semilinear sets* [9]. Note that in general reachability sets are not definable in the Presburger arithmetic [7] (see Example 1.1). An application of the almost semilinear sets was provided; a final configuration is not reachable from an initial one if and only if there exists a forward inductive invariant definable in the Presburger arithmetic that contains the initial configuration but not the final one. Since we can decide if a Presburger formula denotes a forward inductive invariant, we deduce that there exist checkable certificates of non-reachability in the Presburger arithmetic. In particular, there exists a simple algorithm for deciding the general Petri net reachability problem based on two semi-algorithms. A first one that tries to prove the reachability by enumerating finite sequences of actions and a second one that tries to prove the non-reachability by enumerating Presburger formulas. Such an algorithm always terminates in theory but in practice an enumeration does not provide an efficient way for deciding the reachability problem. In particular the problem of deciding *efficiently* the reachability problem is still an *open question*.

When the reachability set is definable in the Presburger arithmetic, the existence of checkable certificates of non-reachability in the Presburger arithmetic is immediate since the reachability set is a forward inductive invariant (in fact the most precise one). The problem of deciding if the reachability set of a Petri net is definable in the Presburger arithmetic was studied twenty years ago independently by Dirk Hauschildt during his PhD [10] and Jean-Luc Lambert. Unfortunately, these two works were never published. Moreover, from these works, it is difficult to deduce a simple algorithm for computing a Presburger formula denoting the reachability set when such a formula exists.

For the class of *flat* Petri nets [11, 12], such a computation can be performed with *acceleration techniques*. Let us recall that a Petri net is said to be *flat* if there exist some words $\sigma_1, \dots, \sigma_k \in T^*$ such that every reachable configuration is the target of a run labeled by a word in $\sigma_1^* \dots \sigma_k^*$ from the initial configuration (A language included in $\sigma_1^* \dots \sigma_k^*$ is said to be *bounded* [13]). *Acceleration techniques* provide a framework for deciding reachability properties that works well in practice but without termination guaranty in theory. Intuitively, acceleration techniques consist in computing with some symbolic representations transitive closures

of sequences of actions. For Petri nets, the Presburger arithmetic is known to be expressive enough for this computation. As a direct consequence, when the reachability set of a Petri net is computable with acceleration techniques, this set is necessarily definable in the Presburger arithmetic. In [12], we proved that a Petri net is flat if, and only if, its reachability set is computable by acceleration.

Recently, we proved that many classes of Petri nets with known Presburger reachability sets are flat [12] and we conjectured that Petri nets with reachability sets definable in the Presburger arithmetic are flat. In [14] the conjecture got closed positively. As a direct consequence, classical acceleration techniques always terminate on the computation of Presburger formulas denoting reachability sets of Petri nets when such a formula exists.

References

- [1] J. Esparza and M. Nielsen, “Decidability issues for petri nets - a survey,” *Bulletin of the European Association for Theoretical Computer Science*, vol. 52, pp. 245–262, 1994.
- [2] G. S. Sacerdote and R. L. Tenney, “The decidability of the reachability problem for vector addition systems (preliminary version),” in *Proc. of STOC’77*. ACM, 1977, pp. 61–76.
- [3] E. W. Mayr, “An algorithm for the general petri net reachability problem,” in *Proc. of STOC’81*. ACM, 1981, pp. 238–246.
- [4] S. R. Kosaraju, “Decidability of reachability in vector addition systems (preliminary version),” in *Proc. of STOC’82*. ACM, 1982, pp. 267–281.
- [5] J. L. Lambert, “A structure to decide reachability in petri nets,” *Theoretical Computer Science*, vol. 99, no. 1, pp. 79–104, 1992.
- [6] D. Figueira, S. Figueira, S. Schmitz, and P. Schnoebelen, “Ackermannian and primitive-recursive bounds with dickson’s lemma,” in *Proc. of LICS 2011*. IEEE Computer Society, 2011, pp. 269–278.
- [7] J. E. Hopcroft and J.-J. Pansiot, “On the reachability problem for 5-dimensional vector addition systems,” *Theoretical Computer Science*, vol. 8, pp. 135–159, 1979.
- [8] J. Leroux, “The general vector addition system reachability problem by Presburger inductive invariants,” in *Proc. of LICS 2009*. IEEE Computer Society, 2009, pp. 4–13.
- [9] S. Ginsburg and E. H. Spanier, “Semigroups, Presburger formulas and languages,” *Pacific Journal of Mathematics*, vol. 16, no. 2, pp. 285–296, 1966.
- [10] D. Hauschildt, “Semilinearity of the reachability set is decidable for Petri nets.” Ph.D. dissertation, University of Hamburg, 1990.
- [11] L. Fribourg, “Petri nets, flat languages and linear arithmetic,” in *Proc. of WFLP’2000*, M. Alpuente, Ed., 2000, pp. 344–365.
- [12] J. Leroux and G. Sutre, “Flat counter automata almost everywhere!” in *Proc. of ATVA’05*, ser. LNCS, vol. 3707. Springer, 2005, pp. 489–503.
- [13] S. Ginsburg and E. Spanier, “Bounded regular sets,” *Proceedings of the American Mathematical Society*, vol. 17, no. 5, pp. 1043–1049, 1966.
- [14] J. Leroux, “Presburger vector addition systems,” in *Proc. LICS 2013*. IEEE Computer Society, 2013, to appear.