



Ackermannian Completion of Separators

Jérôme Leroux^(✉) 

Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, F-33400 Talence, France
`jerome.leroux@labri.fr`

Abstract. Vector addition systems (VAS for short), or equivalently vector addition systems with states, or Petri nets are a long established model of concurrency with extensive applications in modeling and analysis of hardware, software and database systems, as well as chemical, biological and business processes. The central algorithmic problem is reachability: whether from a given initial configuration there exists a sequence of valid execution steps that reaches a given final configuration. The complexity of the problem has remained unsettled since the 1960s, and was recently proved to be Ackermannian-complete.

In 2009, we proved that the reachability problem can be decided with a simple algorithm by observing that negative instances of the reachability problem can be witnessed by partitioning the set configurations into semilinear sets called *complete separators*. Since we can decide in elementary time if a pair of semilinear sets denotes a complete separator, the size of such a witness is Ackermannian in the worst case.

In this paper, we show how recent results about the reachability problem can be combined to derive a matching upper-bound, i.e. for every negative instance of the reachability problem, we can effectively compute in Ackermannian time a complete separator witnessing that property.

1 Introduction

Vector addition systems [8] (VAS for short), or equivalently vector addition systems with states [7], or Petri nets are one of the most popular formal methods for the representation and the analysis of parallel processes [3]. The central algorithmic problem is reachability: whether from a given initial configuration there exists a sequence of valid execution steps that reaches a given final configuration. Many important computational problems in logic and complexity reduce or are even equivalent to this problem [22,6].

After an incomplete proof by Sacerdote and Tenney [20], decidability of the problem was established by Mayr [17,19], whose proof was then simplified by Kosaraju [9]. Building on the further refinements made by Lambert in the 1990s [10], in 2015, a first complexity upper-bound of the reachability problem was provided [12] more than thirty years after the presentation of the algorithm introduced by Mayr [9,10]. The upper-bound given in that paper is “cubic Ackermannian”, i.e. in $\mathbb{F}_{\omega,3}$ (see [21]). This complexity bound was obtained by analyzing the Mayr algorithm. With a refined algorithm and a new ranking

function for proving termination, an Ackermannian complexity upper-bound was obtained in [15]. This means that the reachability problem can be solved in time bounded by $F_\omega(p(n))$ where p is a primitive recursive function and where F_ω is an Ackermann function. Very recently, this complexity bound was proved to be optimal [14,2].

While the complexity of the reachability problem is settled, its parameterized version, in fixed dimension d , is still open with a large complexity gap between the lower-bound and the upper-bound. Some recent results provided ways to decrease that gap (see for instance [1,11]) but the problem remains open. Since there exists d -dimensional VAS with finite but very large reachability sets [18], any reachability algorithm directly based on the Mayr algorithm will necessarily fail in providing a better complexity upper-bound. In fact that algorithm enumerates in some way each possible reachable configurations when the reachability set is finite.

There is another algorithm for deciding the reachability problem independent of the Mayr algorithm. In fact, in [13], we introduced a simple enumerating algorithm for deciding the reachability problem by observing that negative instances of the reachability problem can be witnessed by partitioning the set of configurations into semilinear sets called *complete separators*. Since we can decide in elementary time if a pair of semilinear sets denotes a complete separator, and the reachability problem is Ackermannian-hard, the size of such a witness is necessarily Ackermannian in the worst case.

In this paper, we take the opportunity to show how to combine papers [15] and [13] to prove that from any negative instance of the reachability problem, we can effectively compute in Ackermannian time a complete separator witnessing that property. This result prove the optimality of algorithms based on complete separators for deciding the general reachability problem. Since this paper is an invited paper at FOSSACS'24, so without any reviewing process, no new proof are given in this paper. If a proof is given, it just to be self-content. But in any case, those proofs are copy-past from [15] and [13].

Even if our result does not provide a better understanding of the complexity of the parameterized reachability problem, it shows that algorithms based on complete separators are optimal in general dimension.

2 Basic Notions

In this section, we introduce basic notions and notation.

Notation for Vectors of Integers. By $\mathbb{Z}$ we denote the set of integers, and by $\mathbb{N}$ the set $\{0, 1, 2, \dots\}$ of non-negative integers. Given $d \in \mathbb{N}$, the elements of $\mathbb{Z}^d$ are called (d -dim) *vectors*; they are denoted in bold face, and for $\mathbf{x} \in \mathbb{Z}^d$ we put $\mathbf{x} = (\mathbf{x}(1), \dots, \mathbf{x}(d))$ so that we can refer to the vector components. In this context, d is called the *dimension* of $\mathbf{x}$. We use the component-wise sum

$\mathbf{x} + \mathbf{y}$ of vectors, and their component-wise order $\mathbf{x} \leq \mathbf{y}$. For $c \in \mathbb{N}$, we put $c \cdot \mathbf{x} = (c \cdot \mathbf{x}(1), \dots, c \cdot \mathbf{x}(d))$.

Linear and Semilinear Sets. A set $\mathbf{L} \subseteq \mathbb{N}^d$ is *linear* if there are d -dim vectors $\mathbf{b}$, the *basis*, and $\mathbf{p}_1, \dots, \mathbf{p}_k$, the *periods* (for $k \in \mathbb{N}$), such that $\mathbf{L} = \{\mathbf{x} \in \mathbb{N}^d \mid \mathbf{x} = \mathbf{b} + \mathbf{u}(1) \cdot \mathbf{p}_1 + \dots + \mathbf{u}(k) \cdot \mathbf{p}_k \text{ for some } \mathbf{u} \in \mathbb{N}^k\}$. In this case, by a *presentation* of $\mathbf{L}$ we mean the tuple $(\mathbf{b}, \mathbf{p}_1, \dots, \mathbf{p}_k)$.

A set $\mathbf{S} \subseteq \mathbb{N}^d$ is *semilinear* if it is a finite union of linear sets, i.e. $\mathbf{S} = \mathbf{L}_1 \cup \dots \cup \mathbf{L}_k$ where $\mathbf{L}_j$ are linear sets for all j . In this case, by a *presentation* of $\mathbf{S}$ we mean the sequence of presentations of $\mathbf{L}_1, \dots, \mathbf{L}_k$. When we say that a *semilinear set* $\mathbf{S}$ is given, we mean that we are given a presentation of $\mathbf{S}$; when we say that $\mathbf{S}$ is *effectively constructible* in some context, we mean that there is an algorithm computing its presentation (in the respective context).

We recall that a set $\mathbf{S} \subseteq \mathbb{N}^d$ is semilinear if, and only if, it is expressible in Presburger arithmetic [4]; the respective transformations between presentations and formulas are effective and elementary. Hence if $\mathbf{S} \subseteq \mathbb{N}^d$ is semilinear, then also its complement, denoted as $\overline{\mathbf{S}}$, is semilinear, and $\overline{\mathbf{S}}$ is effectively constructible when (a presentation of) $\mathbf{S}$ is given.

Fast Growing Functions. The Grzegorzcz hierarchy [5,16] is defined thanks to a family $(F_d)_{d \in \mathbb{N}}$ of functions $F_d : \mathbb{N} \rightarrow \mathbb{N}$ such that every primitive recursive function is asymptotically bounded by some function F_d . This family is defined by $F_0(n) \stackrel{\text{def}}{=} n + 1$ and inductively by $F_{d+1}(n) \stackrel{\text{def}}{=} F_d^{n+1}(n)$ for every $n, d \in \mathbb{N}$. Observe that $F_1(n) = 2n + 1$, $F_2(n) = 2^{n+1}(n + 1) - 1$, and $F_3(n)$ grows as a tower of n exponentials. It follows that F_3 is a non elementary function since it eventually exceeds any fixed iteration of the exponential function. An *Ackermannian function*, denoted as F_ω is defined thanks to the diagonal extraction $F_\omega(n) \stackrel{\text{def}}{=} F_{n+1}(n)$ for every $n \in \mathbb{N}$. This function is non primitive recursive.

Vector Addition Systems. A (d -dim) *vector addition system* (VAS for short) is a finite set $\mathbf{A}$ of vectors in $\mathbb{Z}^d$ called *actions*. Vectors $\mathbf{x} \in \mathbb{N}^d$ are called *configurations*, and with an action $\mathbf{a}$ we associate the binary relation $\xrightarrow{\mathbf{a}}$ on the configurations in $\mathbb{N}^d$ by putting $\mathbf{x} \xrightarrow{\mathbf{a}} \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{N}^d$ such that $\mathbf{y} - \mathbf{x} = \mathbf{a}$. The relations $\xrightarrow{\mathbf{a}}$ are naturally extended to the relations $\xrightarrow{\sigma}$ for finite sequences $\sigma = \mathbf{a}_1 \dots \mathbf{a}_k$ of actions by $\mathbf{x} \xrightarrow{\sigma} \mathbf{y}$ if $\mathbf{x} \xrightarrow{\mathbf{a}_1} \dots \xrightarrow{\mathbf{a}_k} \mathbf{y}$ for all $\mathbf{x}, \mathbf{y} \in \mathbb{N}^d$.

On the set $\mathbb{N}^d$ of configurations we define the *reachability relation* $\xrightarrow{\mathbf{A}^*}$: we put $\mathbf{x} \xrightarrow{\mathbf{A}^*} \mathbf{y}$ if there is $\sigma \in \mathbf{A}^*$ such that $\mathbf{x} \xrightarrow{\sigma} \mathbf{y}$. For $\mathbf{x} \in \mathbb{N}^d$ and $\mathbf{X} \subseteq \mathbb{N}^d$ we put $\text{POST}_{\mathbf{A}}^*(\mathbf{x}) \stackrel{\text{def}}{=} \{\mathbf{y} \in \mathbb{N}^d \mid \mathbf{x} \xrightarrow{\mathbf{A}^*} \mathbf{y}\}$, and $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \stackrel{\text{def}}{=} \bigcup_{\mathbf{x} \in \mathbf{X}} \text{POST}_{\mathbf{A}}^*(\mathbf{x})$. Symmetrically, for $\mathbf{y} \in \mathbb{N}^d$ and $\mathbf{Y} \subseteq \mathbb{N}^d$ we put $\text{PRE}_{\mathbf{A}}^*(\mathbf{y}) \stackrel{\text{def}}{=} \{\mathbf{x} \in \mathbb{N}^d \mid \mathbf{x} \xrightarrow{\mathbf{A}^*} \mathbf{y}\}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}) \stackrel{\text{def}}{=} \bigcup_{\mathbf{y} \in \mathbf{Y}} \text{PRE}_{\mathbf{A}}^*(\mathbf{y})$. By $\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{Y}$ we denote that $\mathbf{x} \xrightarrow{\mathbf{A}^*} \mathbf{y}$ for some $\mathbf{x} \in \mathbf{X}$ and $\mathbf{y} \in \mathbf{Y}$.

The semilinear *reachability problem* takes as input a triple $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ where $\mathbf{X}, \mathbf{Y}$ are (presentations of) semi-linear sets of configurations of a VAS $\mathbf{A}$, and checks if $\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{Y}$ hold. In the standard definition of the reachability problem the sets $\mathbf{X}, \mathbf{Y}$ are singletons; the problem is decidable [19], and it has been recently

shown to be Ackermann-complete [15,14,2]. It is well-known (and easy to show) that the above more general version (the semilinear reachability problem) is tightly related to the standard version, and has thus the same complexity.

3 Separators

A *separator* is a negative instance of the semilinear reachability problem, i.e. a triple $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ where $\mathbf{X}, \mathbf{Y}$ are semilinear sets of configurations of a VAS $\mathbf{A}$ such that $\neg(\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{Y})$. The *domain* $\mathbf{D}$ of a separator $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ is the semilinear set $\overline{\mathbf{X} \cup \mathbf{Y}}$. Notice that $\mathbf{X}, \mathbf{D}$, and $\mathbf{Y}$ forms a partition of $\mathbb{N}^d$. When the domain is empty, the separator is said to be *complete*. Notice that a triple $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ is a *complete separator* if, and only if, $(\mathbf{X}, \mathbf{Y})$ is a partition of $\mathbb{N}^d$ into semilinear sets such that $\mathbf{y} - \mathbf{x} \neq \mathbf{a}$ for every $\mathbf{x} \in \mathbf{X}$, $\mathbf{y} \in \mathbf{Y}$, and $\mathbf{a} \in \mathbf{A}$. In particular this property is decidable in elementary time by encoding it as the satisfiability of a Presburger formula. A separator $(\mathbf{X}', \mathbf{Y}')$ is called a *completion* of a separator $(\mathbf{X}, \mathbf{Y})$ if $(\mathbf{X}', \mathbf{Y}')$ is complete, $\mathbf{X} \subseteq \mathbf{X}'$ and $\mathbf{Y} \subseteq \mathbf{Y}'$.

In [15] we proved that every separator can be effectively completed. In this paper, we show how this result can be extended with optimal complexity bounds. More formally, we prove that any separator can be completed in Ackermannian time. The Ackermannian lower-bound is immediate since the reachability problem for VAS is Ackermannian-complete and as already mentioned, we can check in elementary time if a pair of semilinear sets is a completion of a separator. The most difficult part of the result is the Ackermannian upper-bound.

4 Semi-Pseudo-Linear Sets

Given two semilinear sets $\mathbf{X}, \mathbf{Y}$ of configurations of a VAS $\mathbf{A}$, the sets $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{Y}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}) \cap \mathbf{X}$ are not semilinear in general. However, we proved in [13] that those sets are semi-pseudo-linear, a class of sets that can be tightly over-approximated by semilinear sets called *linearizations*. Linearizations are obtained by solving several instances of the semilinear reachability problem. Since in [14,2], we provided an Ackermannian upper-bounds on that decision problem, we can reasonably think that the completion of separators can be done in Ackermannian time. To prove that result, in this section we provide complexity bounds on the size of linearizations. Those linearizations will be used in the next section for completing separators in Ackermannian time.

Let us recall some definitions. A monoid $\mathbf{M}$ is a set of configurations such that $\mathbf{0} \in \mathbf{M}$, and such that $\mathbf{M} + \mathbf{M} \subseteq \mathbf{M}$. The monoid *spanned* by a set $\mathbf{P} \subseteq \mathbb{N}^d$ is the set of finite sums of vectors in $\mathbf{P}$. It is denoted as $\Sigma \mathbf{P}$. A vector $\mathbf{a} \in \mathbb{N}^d$ is called an *interior vector* of a monoid $\mathbf{M}$, if for every $\mathbf{m} \in \mathbf{M}$, there exists a natural number $n \geq 1$ such that $n\mathbf{a} \in \mathbf{m} + \mathbf{M}$.

A *pseudo-linear set* is a set $\mathbf{X} \subseteq \mathbb{N}^d$ such that there exists a linear set $\mathbf{L} = \mathbf{b} + \mathbf{M}$ where $\mathbf{M}$ is the monoid spanned by the periods of $\mathbf{L}$, such that

$\mathbf{X} \subseteq \mathbf{L}$ and such that for every finite set $\mathbf{R}$ of interior vectors of $\mathbf{M}$, there exists $\mathbf{x} \in \mathbf{X}$ such that $\mathbf{x} + \Sigma \mathbf{R} \subseteq \mathbf{X}$. In that case, the linear set $\mathbf{L}$ is called a *linearization* of $\mathbf{X}$. A semi-pseudo-linear set $\mathbf{X}$ is a finite union of pseudo-linear sets $\mathbf{X} = \mathbf{X}_1 \cup \dots \cup \mathbf{X}_k$. In that case a semilinear set of the form $\mathbf{L}_1 \cup \dots \cup \mathbf{L}_k$ where $\mathbf{L}_j$ is a linearization of $\mathbf{X}_j$ is called a *linearization* of $\mathbf{X}$.

By combining the proof of [13, Theorem 6.4] with [15], we deduce the following theorem where f_d is a function of the form $F_{d+3}(Cn)$ for some constant C independent of d . In this theorem, the size in binary or in unary does not change the result and there is a lot of freedom in the definition of the size of presentations of semilinear sets and VAS.

Theorem 1. *Given two semilinear sets $\mathbf{X}$ and $\mathbf{Y}$ of configurations of a d -dim VAS $\mathbf{A}$, the sets $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{Y}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}) \cap \mathbf{X}$ are semi-pseudo-linear. Moreover, we can effectively compute in time $f_d(n)$ where n is the size of the input, presentations of linearizations of those sets.*

The tightness of linearization approximations can be emphasized by introducing the notion of rank^1 given in [13]. Formally, the *rank* of a set $\mathbf{X} \subseteq \mathbb{N}^d$, denoted as $\text{rank } \mathbf{X}$ is the minimal $r \in \{-\infty, 0, \dots, d\}$ such that there exists a semi-linear set $\mathbf{S}$ that contains $\mathbf{X}$ of the form $\mathbf{b}_1 + \mathbf{M}_1 \cup \dots \cup \mathbf{b}_k + \mathbf{M}_k$ where $\mathbf{M}_1, \dots, \mathbf{M}_k$ are monoids spanned by at most r vectors. In [13], we prove that $\text{rank}(\mathbf{X}) = -\infty$ iff $\mathbf{X}$ is empty, $\text{rank}(\mathbf{X}) \leq \text{rank}(\mathbf{Y})$ if $\mathbf{X} \subseteq \mathbf{Y}$, and the following theorem.

Theorem 2 (Proposition 7.10 of [13]). *Let $\mathbf{S}_1, \mathbf{S}_2$ be linearizations of two non-empty semi-pseudo-linear sets $\mathbf{X}_1, \mathbf{X}_2$ with an empty intersection. We have:*

$$\text{rank}(\mathbf{S}_1 \cap \mathbf{S}_2) < \text{rank}(\mathbf{X}_1 \cup \mathbf{X}_2)$$

5 Ackermannian Completion

We show in this section how a separator $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ can be completed in Ackermannian time. We follow the algorithm introduced in [13] by first proving that if $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ is not complete, i.e. if the domain $\mathbf{D}$ is non empty, we can effectively compute a separator $(\mathbf{X}', \mathbf{A}, \mathbf{Y}')$ with a domain $\mathbf{D}'$ such that $\mathbf{X} \subseteq \mathbf{X}'$, $\mathbf{Y} \subseteq \mathbf{Y}'$, and such that $\text{rank}(\mathbf{D}') < \text{rank}(\mathbf{D})$. It follows that by applying at most d times this algorithm where d is the dimension of $\mathbf{A}$, we get a complete separator.

Let n be the size of the separator $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$.

The set $\mathbf{Y}'$ is obtained as follows. Since $\mathbf{D}$ is semilinear and effectively computable in elementary time, it follows from Theorem 1 that we can compute in time $f_d(E(n))$ where E is some fixed elementary function a linearization $\mathbf{U}$ of the semi-pseudo-linear set $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{D}$. We introduce $\mathbf{Y}' \stackrel{\text{def}}{=} \mathbf{Y} \cup (\mathbf{D} \setminus \mathbf{U})$.

Let us prove that $(\mathbf{X}, \mathbf{A}, \mathbf{Y}')$ is a separator. By contradiction, assume that $\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{Y}'$. Since $\neg(\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{Y})$, and $\mathbf{Y}' = \mathbf{Y} \cup (\mathbf{D} \setminus \mathbf{U})$, we deduce that $\mathbf{X} \xrightarrow{\mathbf{A}^*} \mathbf{D} \setminus \mathbf{U}$.

¹ In [13] this notion is called *dimension* but in our context, the dimension word is already used for the number of components of a vector.

$(\mathbf{D} \setminus \mathbf{U})$. However, since $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{D} \subseteq \mathbf{U}$ we get a contradiction. Hence $(\mathbf{X}, \mathbf{A}, \mathbf{Y}')$ is a separator and its domain is equal to $\mathbf{D} \cap \mathbf{U}$.

The set $\mathbf{X}'$ is obtained symmetrically. Since $\mathbf{D} \cap \mathbf{U}$ is semilinear and effectively computable in elementary time, it follows from Theorem 1 that we can compute in time $f_d(E'(f_d(E(n))))$ where E' is some fixed elementary function a linearization $\mathbf{V}$ of the semi-pseudo-linear set $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}') \cap \mathbf{D} \cap \mathbf{U}$. We introduce $\mathbf{X}' \stackrel{\text{def}}{=} \mathbf{X} \cup ((\mathbf{D} \cap \mathbf{U}) \setminus \mathbf{V})$.

Symmetrically, we deduce that $(\mathbf{X}', \mathbf{A}, \mathbf{Y}')$ is a separator and its domain $\mathbf{D}'$ is equal to $\mathbf{D} \cap \mathbf{U} \cap \mathbf{V}$.

Since $(\mathbf{X}, \mathbf{A}, \mathbf{Y}')$ is a separator, it follows that $\text{POST}_{\mathbf{A}}^*(\mathbf{X})$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}')$ have an empty intersection. In particular the semi-pseudo-linear sets $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{D}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}') \cap \mathbf{D} \cap \mathbf{U}$ have an empty intersection. If one of those semi-pseudo-linear sets is empty then $\mathbf{D}'$ is empty and in particular $\text{rank}(\mathbf{D}') < \text{rank}(\mathbf{D})$. Otherwise, from Theorem 2 we deduce that the rank of $\mathbf{U} \cap \mathbf{V}$ is strictly bounded by the rank of the union of $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{D}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}') \cap \mathbf{D} \cap \mathbf{U}$. Since this set is included in $\mathbf{D}$, and $\mathbf{D}'$ is included in $\mathbf{U} \cap \mathbf{V}$, we deduce that $\text{rank}(\mathbf{D}') < \text{rank}(\mathbf{D})$.

By replacing E and E' by $E + E'$, we can assume without loss of generality that $E = E'$. By iterating the previous construction at most d times, we deduce that from any separator $(\mathbf{X}, \mathbf{A}, \mathbf{Y})$ of size n , we can compute in time $(f_d \circ E)^{2d}(n)$ a completion of it. We deduce the main theorem of that paper.

Theorem 3. *Separators can be completed in Ackermannian time.*

6 Conclusion

In this paper, we have shown that separators can be completed in Ackermannian time. Our computation is based on a generic algorithm given in Section 5. This algorithm can be implemented as soon as we have an oracle computing semilinear sets over-approximating the sets $\text{POST}_{\mathbf{A}}^*(\mathbf{X}) \cap \mathbf{D}$ and $\text{PRE}_{\mathbf{A}}^*(\mathbf{Y}) \cap \mathbf{D}$. If those approximations are not linearizations, the termination of the algorithm is no longer true in general. However, since its correctness is maintained, it should be interesting to benchmark such an algorithm when using heuristics for implementing oracles computing reachability set over-approximations (based on abstract interpretation, acceleration techniques, parameterized invariant, and so on).

References

1. Czerwinski, W., Jecker, I., Lasota, S., Leroux, J., Orlikowski, L.: New lower bounds for reachability in vector addition systems. In: Bouyer, P., Srinivasan, S. (eds.) 43rd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2023, December 18-20, 2023, IIIT Hyderabad, Telangana, India. LIPIcs, vol. 284, pp. 35:1–35:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2023). <https://doi.org/10.4230/LIPICS.FSTTCS.2023.35>

2. Czerwinski, W., Orlikowski, L.: Reachability in vector addition systems is Ackermann-complete. In: 62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022. pp. 1229–1240. IEEE (2021). <https://doi.org/10.1109/FOCS52979.2021.00120>
3. Esparza, J., Nielsen, M.: Decidability issues for Petri nets - a survey. *Bulletin of the European Association for Theoretical Computer Science* **52**, 245–262 (1994)
4. Ginsburg, S., Spanier, E.H.: Semigroups, Presburger formulas and languages. *Pacific Journal of Mathematics* **16**(2), 285–296 (1966). <https://doi.org/10.2140/pjm.1966.16.285>
5. Grzegorzczak, A.: Some classes of recursive functions. *Instytut Matematyczny Polskiej Akademii Nauk* (1953), <http://eudml.org/doc/219317>
6. Hack, M.: Decidability questions for Petri nets. Ph.D. thesis, MIT (1975), <http://publications.csail.mit.edu/lcs/pubs/pdf/MIT-LCS-TR-161.pdf>
7. Hopcroft, J.E., Pansiot, J.J.: On the reachability problem for 5-dimensional vector addition systems. *Theoretical Computer Science* **8**, 135–159 (1979)
8. Karp, R.M., Miller, R.E.: Parallel program schemata. *J. Comput. Syst. Sci.* **3**(2), 147–195 (1969). [https://doi.org/10.1016/S0022-0000\(69\)80011-5](https://doi.org/10.1016/S0022-0000(69)80011-5)
9. Kosaraju, S.R.: Decidability of reachability in vector addition systems (preliminary version). In: STOC. pp. 267–281. ACM (1982). <https://doi.org/10.1145/800070.802201>
10. Lambert, J.: A structure to decide reachability in Petri nets. *Theor. Comput. Sci.* **99**(1), 79–104 (1992). [https://doi.org/10.1016/0304-3975\(92\)90173-D](https://doi.org/10.1016/0304-3975(92)90173-D)
11. Lasota, S.: Improved Ackermannian lower bound for the Petri nets reachability problem. In: Berenbrink, P., Monmege, B. (eds.) 39th International Symposium on Theoretical Aspects of Computer Science, STACS 2022, March 15-18, 2022, Marseille, France (Virtual Conference). LIPIcs, vol. 219, pp. 46:1–46:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2022). <https://doi.org/10.4230/LIPIcs.STACS.2022.46>
12. Leroux, J., Schmitz, S.: Demystifying reachability in vector addition systems. In: 30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015. pp. 56–67. IEEE Computer Society (2015). <https://doi.org/10.1109/LICS.2015.16>
13. Leroux, J.: The general vector addition system reachability problem by presburger inductive invariants. In: Proceedings of the 24th Annual IEEE Symposium on Logic in Computer Science, LICS 2009, 11-14 August 2009, Los Angeles, CA, USA. pp. 4–13. IEEE Computer Society (2009). <https://doi.org/10.1109/LICS.2009.10>
14. Leroux, J.: The reachability problem for Petri nets is not primitive recursive. In: 62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022. pp. 1241–1252. IEEE (2021). <https://doi.org/10.1109/FOCS52979.2021.00121>
15. Leroux, J., Schmitz, S.: Reachability in vector addition systems is primitive-recursive in fixed dimension. In: 34th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2019, Vancouver, BC, Canada, June 24-27, 2019. pp. 1–13. IEEE (2019). <https://doi.org/10.1109/LICS.2019.8785796>
16. Löb, M.H., Wainer, S.S.: Hierarchies of number-theoretic functions. i. *Archiv für mathematische Logik und Grundlagenforschung* **13**(1), 39–51 (1970). <https://doi.org/10.1007/BF01967649>
17. Mayr, E.W.: An algorithm for the general petri net reachability problem. In: Proceedings of the 13th Annual ACM Symposium on Theory of Computing, May 11-13, 1981, Milwaukee, Wisconsin, USA. pp. 238–246. ACM (1981). <https://doi.org/10.1145/800076.802477>

18. Mayr, E.W., Meyer, A.R.: The complexity of the finite containment problem for petri nets. *J. ACM* **28**(3), 561–576 (1981). <https://doi.org/10.1145/322261.322271>
19. Mayr, E.W.: An algorithm for the general Petri net reachability problem. *SIAM J. Comput.* **13**(3), 441–460 (1984). <https://doi.org/10.1137/0213029>
20. Sacerdote, G.S., Tenney, R.L.: The decidability of the reachability problem for vector addition systems (preliminary version). In: *Proceedings of the 9th Annual ACM Symposium on Theory of Computing*, May 4-6, 1977, Boulder, Colorado, USA. pp. 61–76. ACM (1977). <https://doi.org/10.1145/800105.803396>
21. Schmitz, S.: Complexity hierarchies beyond elementary. *TOCT* **8**(1), 3:1–3:36 (2016). <https://doi.org/10.1145/2858784>
22. Schmitz, S.: The complexity of reachability in vector addition systems. *ACM SIGLOG News* **3**(1), 4–21 (2016). <https://doi.org/10.1145/2893582.2893585>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

