Algorithmic Analysis of Termination and Counter Complexity in Vector Addition Systems with States: A Survey of Recent Results

Antonín Kučera, Masaryk Univerzity, Czechia

We give an overview of recently established results about the effective asymptotic analysis of termination and counter complexity of VASS computations. In contrast to "classical" problems such as reachability, boundedness, liveness, coverability, etc., that are **EXPSPACE**-hard, the decision problems related to VASS asymptotic analysis tend to have low complexity and many important variants are even decidable in polynomial time. We also present selected concepts and techniques used to achieve these results.

1. INTRODUCTION

Vector addition systems with states (VASS) [Hopcroft and Pansiot 1979] are a generic formalism expressively equivalent to Petri nets. In formal verification, VASS are traditionally used to model parameterized systems, programs operating over unbounded integer variables, etc. Various questions about the original systems can thus be reduced to the corresponding problems for VASS. Unfortunately, the scalability of this approach is limited by the high computational complexity of the relevant VASS problems (see, e.g., [Czerwiński et al. 2019; Lipton 1976; Mayr and Meyer 1981]).

In automated asymptotic analysis of computer programs, VASS-based abstractions are used for evaluating the dependency of the running time (and other complexity measures) on the size of the program input (see, e.g., [Sinn et al. 2013; 2017]). This motivates the study of *asymptotic complexity* of VASS computations. More concretely, the *termination complexity* of a given VASS is a function $\mathcal{L} : \mathbb{N} \to \mathbb{N}_\infty$ assigning to a given n the maximal length of a computation initiated in a configuration $p\mathbf{v}$ where $\mathbf{v} \leq \vec{n}$. Similarly, a *counter complexity* of a given counter c is a function $\mathcal{C}[c] : \mathbb{N} \to \mathbb{N}_\infty$ such that $\mathcal{C}[c](n)$ is the supremum of the counter values assigned to c along a computation initiated in $p\mathbf{v}$ where $\mathbf{v} \leq \vec{n}$. The termination/counter complexity can be seen as variants of the standard time/space complexity adjusted to the VASS computational model.

The non-deterministic choice in VASS can be resolved in favor of increasing or decreasing the termination/counter complexity. In program analysis, both options are considered sensible [Broy and Wirsing 1981]. If the choice corresponds to actions taken by the environment or overapproximates branching constructs such as `if-then-else`, it is interpreted as *demonic*, i.e., the least convenient option maximizing the complexity is taken. If the choice is under control of the program, it is *angelic*, and the most convenient option is taken. This naturally leads to the model of *VASS games*, where the control states are split into two subsets of *demonic* and *angelic* states, and the choice is resolved by two players, *Demon* and *Angel*, aiming at increasing and decreasing the complexity, respectively. VASS games where all states are demonic are called *demonic VASS*.

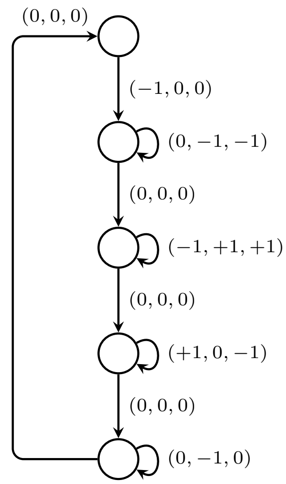
To get some intuition how the results about asymptotic termination/counter complexity of VASS computations help to analyze the complexity of imperative programs, consider the example of Fig. 1. The program of Fig. 1a inputs a non-negative integer `i` and then it executes two nested **while**-loops. Since the assignment `j = i` cannot be modeled by a VASS directly, an auxiliary variable `Aux` is introduced, and the assignment is emulated by two extra loops. Thus, an equivalent program of Fig. 1b is obtained. For this program, a demonic VASS model is constructed simply by replacing deterministic branching with demonic choice. The termination complexity of the demonic VASS of Fig. 1c is $\mathcal{O}(n^2)$. This bound carries over to the original program of Fig. 1a, because the constructed VASS over-approximates its behavior. A closer look re-

```
input i
while(i >0)
    i = i-1
    j = i
    while(j >0)
        j = j-1
```

(a) An imperative program.

```
input i
while(i >0)
    i--
    j = 0
    Aux = 0
    while(i >0)
        i--
        j++
        Aux--
    while(Aux>0)
        i++
        Aux--
    while(j >0)
        j--
```

(b) An equivalent program.



(c) A demonic VASS model.

Fig. 1. A demonic VASS model of a simple imperative program.

```
input i;
j:=0; k:=0; z:=0;

if (i mod 2) = 0    // demonic choice //
    then j := i*i
    else k := i*i

choose:             // angelic choice //
    z:=j;
or: z:=k;
```

Fig. 2. A program with demonic and angelic non-determinism.

veals that the program Fig. 1a of needs $\Omega(n^2)$ time to terminate, and hence the bound is asymptotically optimal.

An example of a program with angelic choice is given in Fig. 2. Here, the controller strives to keep the value of z as small as possible. A VASS model of this program includes gadgets weakly computing the multiplication[1] and it has both demonic and angelic control states. It turns out that $\mathcal{C}[c]$ is *linear* in this VASS game, where c is the counter modeling the variable z. Hence, the controller can keep the value of z linear in the input size.

This paper gives an overview of recent results about decision problems related to the asymptotic analysis of VASS computations. Surprisingly, these problems tend to have *low complexity*, and many important variants are even solvable in *polynomial time*. This contrasts sharply with the complexity of "classical" problems such as reachability, boundedness, coverability, or liveness, where the **EXPSPACE** lower bound of [Lipton 1976] applies. Selected concepts and proof techniques are sketched along with the presented results.

[1]The concept of weak Petri computer is due to Rabin; see [Leroux and Schnoebelen 2014] for more details.

2. DEFINITIONS

We use $\mathbb{Z}$, $\mathbb{N}$, and $\mathbb{R}$ to denote the sets of integers, non-negative integers, and the reals, respectively. We put $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$ where ∞ is treated according to the standard conventions. The vectors of $\mathbb{Z}^d$ where $d \geq 1$ are denoted by $\mathbf{v}, \mathbf{u}, \ldots$, and the vector $(n, \ldots, n)$ is denoted by $\vec{n}$. A vector $\mathbf{v}$ is *positive* if every component of $\mathbf{v}$ is positive.

2.1. Vector Addition Systems with States (VASS)

Let $d \geq 1$. A *d-dimensional vector addition system with states (VASS)* is a pair $\mathcal{V} = (Q, \mathit{Tran})$, where $Q \neq \emptyset$ is a finite set of *states* and $\mathit{Tran} \subseteq Q \times \mathbb{Z}^d \times Q$ is a finite set of *transitions* such that for every $q \in Q$ there exist $p \in Q$ and $\mathbf{u} \in \mathbb{Z}^d$ where $(q, \mathbf{u}, p) \in \mathit{Tran}$.

A *configuration* of $\mathcal{A}$ is a pair $p\mathbf{v} \in Q \times \mathbb{N}^d$, where $\mathbf{v}$ is the vector of counter values. To simplify our notation, we identify counters with their indexes ranging over $\{1, \ldots, d\}$. Hence, the value of a counter $c \in \{1, \ldots, d\}$ in a configuration $p\mathbf{v}$ is $\mathbf{v}(c)$.

A *finite path* in $\mathcal{V}$ of length m is a finite sequence $\varrho = p_1, \mathbf{u}_1, p_2, \mathbf{u}_2, \ldots, p_m$ such that $(p_i, \mathbf{u}_i, p_{i+1}) \in \mathit{Tran}$ for all $1 \leq i < m$. We use $\Delta(\varrho) = \mathbf{u}_1 + \cdots + \mathbf{u}_{m-1}$ to denote the *effect* of ϱ. We say that ϱ is a *cycle* if $p_1 = p_m$, and a cycle ϱ is *simple* if all control states visited by ϱ are pairwise different except for p_1 and p_m. An *infinite path* in $\mathcal{V}$ is an infinite sequence $p_1, \mathbf{u}_1, p_2, \mathbf{u}_2, \ldots$ such that the prefix $p_1, \mathbf{u}_1, \ldots, p_m$ is a finite path in $\mathcal{V}$ for every $m \geq 1$.

A *computation* of $\mathcal{V}$ is a sequence of configurations $p_1\mathbf{v}_1, p_2\mathbf{v}_2, \ldots$ of length $m \in \mathbb{N}_\infty$ such that for every $1 \leq i < m$ there is a transition $(p_i, \mathbf{u}_i, p_{i+1})$ satisfying $\mathbf{v}_{i+1} = \mathbf{v}_i + \mathbf{u}_i$. Note that every computation determines its associated path in $\mathcal{V}$ in the natural way.

The underlying directed graph of $\mathcal{V}$ can be split into strongly connected components (SCCs) in the standard way. Every SCC $\mathcal{B}$ can be seen as a VASS where the set of control states is restricted to $\mathcal{B}$ and all ingoing/outgoing transitions from/to the other SCCs are deleted. If *all* outgoing transitions of some $p \in \mathcal{B}$ are deleted, a new transition $(p, \vec{0}, p)$ is created to satisfy the requirements of VASS definition. We say that $\mathcal{V}$ is *strongly connected* if it has only one SCC.

2.2. VASS Games

A *d-dimensional VASS game* is a tuple $\mathcal{A} = (Q, \mathit{Tran}, Q_A, Q_D)$ where (Q, Tran) is a VASS and (Q_A, Q_D) is a partition of Q into (possibly empty) subsets of *angelic* and *demonic* control states. The encoding size of $\mathcal{A}$, where the counter update vectors in transitions are written in binary, is denoted by $\|\mathcal{A}\|$.

All notions introduced for VASS are applicable also to VASS games. Furthermore, a configuration $p\mathbf{v}$ of $\mathcal{A}$ is angelic/demonic depending on whether p is angelic/demonic. We say that $\mathcal{A}$ is a *demonic VASS* if $Q_A = \emptyset$.

A computation in $\mathcal{A}$ is determined by two players, *Angel* and *Demon*, responsible for selecting transitions in angelic and demonic configurations, respectively. In general, the decision may depend on the whole computational history. Formally, a *strategy* for player Angel (or Demon) is a function η assigning to every finite computation $p_1\mathbf{v}_1, \ldots, p_m\mathbf{v}_m$ where $p_m \in Q_A$ (or $p_m \in Q_D$) a transition $(p_m, \mathbf{u}, q)$. A strategy is *positional* if it depends only on the configuration $p_m\mathbf{v}_m$, and *counterless* if it depends only on the control state p_m.

Every pair of strategies (σ, π) for Angel and Demon and every initial configuration $p\mathbf{v}$ determine the unique *maximal* computation $\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})$ initiated in $p\mathbf{v}$. The maximality means that the computation cannot be prolonged by performing the transition selected by σ or π without making some counter negative. Observe that $\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})$ can be finite or infinite.

2.3. Termination and Counter Complexity

For a given computation $\alpha = p_1\mathbf{v}_1, p_2\mathbf{v}_2, \ldots$, we use $\mathit{len}(\alpha)$ to denote the length of α and $\mathit{max}[c](\alpha)$ to denote $\sup\{\mathbf{v}_i(c) \mid i \in \mathbb{N}\}$ (where c is a counter). Note that both $\mathit{len}(\alpha)$ and $\mathit{max}[c](\alpha)$ can be infinite.

Intuitively, $\mathit{len}(\alpha)$ corresponds to the computational time of α, and $\mathit{max}[c](\alpha)$ models the required space in the counter c. The aim of Angel and Demon is to minimize and maximize the use of these resources, i.e., to minimize and maximize the *len* and $\mathit{max}[c]$ objective functions. By applying standard game-theoretic arguments (see, e.g., [Ajdarów and Kučera 2021]), we obtain the *determinacy* of VASS games with the *len* and $\mathit{max}[c]$ objectives, i.e., the equalities

$$\sup_{\pi} \inf_{\sigma} \mathit{len}(\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})) = \inf_{\sigma} \sup_{\pi} \mathit{len}(\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})) \tag{1}$$

$$\sup_{\pi} \inf_{\sigma} \mathit{max}[c](\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})) = \inf_{\sigma} \sup_{\pi} \mathit{max}[c](\mathit{Comp}^{\sigma,\pi}(p\mathbf{v})) \tag{2}$$

where σ and π range over all strategies for Angel and Demon, respectively. Hence, there exist unique *termination* and *maximal counter c values* of $p\mathbf{v}$, denoted by $\mathit{Tval}(p\mathbf{v})$ and $\mathit{Cval}[c](p\mathbf{v})$, defined by (1) and (2), respectively. Furthermore, both players have *optimal* positional strategies σ^* and π^* achieving the outcome specified by the equilibrium value or better in every configuration $p\mathbf{v}$ against an arbitrary strategy of the opponent.

Intuitively, $\mathit{Tval}(p\mathbf{v})$ and $\mathit{Cval}[c](p\mathbf{v})$ are $\mathit{len}(\alpha)$ and $\mathit{max}[c](\alpha)$ where α is a computation initiated in $p\mathbf{v}$ obtained when both players make optimal decisions. Observe that $\mathit{Tval}(p\mathbf{v})$ and $\mathit{Cval}[c](p\mathbf{v})$ are monotonic in $\mathbf{v}$, i.e., $\mathbf{v} \le \mathbf{u}$ implies $\mathit{Tval}(p\mathbf{v}) \le \mathit{Tval}(p\mathbf{u})$ and $\mathit{Cval}[c](p\mathbf{v}) \le \mathit{Cval}[c](p\mathbf{u})$. The *termination complexity* and *counter c complexity* of $\mathcal{A}$ are the functions $\mathcal{L}, \mathcal{C}[c] : \mathbb{N} \to \mathbb{N}_\infty$ defined by

$$\mathcal{L}(n) = \max\{\mathit{Tval}(p\vec{n}) \mid p \in Q\}\,,$$
$$\mathcal{C}[c](n) = \max\{\mathit{Cval}[c](p\vec{n}) \mid p \in Q\}\,.$$

When the underlying $\mathcal{A}$ is not clear, we write $\mathcal{L}_{\mathcal{A}}$ and $\mathcal{C}_{\mathcal{A}}[c]$ instead of $\mathcal{L}$ and $\mathcal{C}[c]$.

Remark 2.1. The asymptotic analysis of termination complexity is trivially reducible to the asymptotic analysis of counter complexity. More specifically, for every VASS game $\mathcal{A}$, we can construct a VASS game $\mathcal{B}$ by adding a fresh "step counter" *sc* incremented by every transition. Clearly, $\mathcal{L}_{\mathcal{A}} \in \Theta(\mathcal{C}_{\mathcal{B}}[\mathit{sc}])$. Therefore, the lower and upper complexity bounds for the problems of asymptotic analysis carry over from $\mathcal{L}$ to $\mathcal{C}[c]$ and from $\mathcal{C}[c]$ to $\mathcal{L}$, respectively.

Remark 2.2. In *strongly connected demonic* VASS games, the counters can be increased to the obtained asymptotic lower bounds *simultaneously*. Consider a d-dimensional VASS game $\mathcal{A}$. The initial vector $(n, \ldots, n)$ of counter values can be "split" into d copies of a smaller vector $(\lfloor n/d \rfloor, \ldots, \lfloor n/d \rfloor)$, and the d computations pumping the individual counters can be run from this smaller vector and concatenated. The concatenation may require traversals to control state suitable for pumping the next counter, and here we need the assumption that $\mathcal{A}$ is strongly connected and demonic. Hence, for all sufficiently large n, there exists a computation initiated in a configuration $p\vec{n}$ such *every* counter c is increased at least to $\mathcal{C}[c](\lfloor n/d \rfloor - |Q| \cdot \kappa)$ along the computation, where κ is the maximal absolute value of a counter update in a transition of $\mathcal{A}$.

Observe that according to our definitions, Angel may terminate a computation in every angelic configuration $q\mathbf{u}$ with *at least one* outgoing transition making some counter negative. Another possibility, perhaps more convenient for modeling purposes, is to consider an angelic qu terminating only if *all* outgoing transitions make some counter negative. Unfortunately, this modification breaks the monotonicity of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$, and the asymptotic analysis of their growth is no longer sensible. Alternatively, one may consider decreasing a *given* counter below zero as a termination condition, but this leads to similar issues.

3. MEASURING THE VASS COMPLEXITY

Measuring the asymptotic growth of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$ for a given VASS game makes sense only if these functions take only finite values. We say that $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) is *bounded* if $\mathcal{L}(n) < \infty$ (or $\mathcal{C}[c](n) < \infty$) for all $n \in \mathbb{N}$. The complexity of the boundedness problem is relatively low.

THEOREM 3.1. *The boundedness of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$ is decidable in polynomial time for demonic VASS, and* $\mathbf{NP}$*-complete for general VASS games.*

The results for demonic VASS follows by observing that

— $\mathcal{L}(n)$ in unbounded iff there exists a cycle ϱ such that $\Delta(\varrho) \geq \vec{0}$;
— $\mathcal{C}[c](n)$ in unbounded iff there exists a cycle ϱ such that $\Delta(\varrho) \geq \vec{0}$ and $\Delta(\varrho)(c) > 0$.

The existence of ϱ can be decided in polynomial time by a slight modification of the algorithm of [Kosaraju and Sullivan 1988].

The $\mathbf{NP}$ upper bound for general VASS games is obtained by showing that if Angel has *some* strategy σ such that *len*$(\mathit{Comp}^{\sigma,\pi}(p\vec{n})) < \infty$ (or *max*$[c](\mathit{Comp}^{\sigma,\pi}(p\vec{n})) < \infty$) for all π and n, then Angel also has a *counterless* strategy with this property. A proof for *len* can be found in [Brázdil et al. 2010], and the same technique applies also to *max*$[c]$. Hence, it suffices to guess an appropriate counterless strategy σ for Angel, apply σ to a given VASS game $\mathcal{A}$ by preserving only the transition selected by σ, and then check the boundedness of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$ in the resulting demonic VASS $\mathcal{A}^\sigma$. The $\mathbf{NP}$ lower bound for $\mathcal{L}(n)$ follows from the $\mathbf{coNP}$ lower bound for the *unknown initial credit problem* for generalized energy games [Chatterjee et al. 2010]. By Remark 2.1, the $\mathbf{NP}$ lower bound for $\mathcal{L}(n)$ carries over to $\mathcal{C}[c](n)$.

3.1. Function hierarchies

The asymptotic growth of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$ has so far been classified with respect to the *polynomial* and *Grzegorczyk* hierarchies.

3.1.1. Polynomial hierarchy. The polynomial hierarchy is determined by the functions $n, n^2, n^3, \ldots$. Basic questions related to the polynomial hierarchy include the following:

— *Membership to a given level of the hierarchy.* For a given VASS game $\mathcal{A}$ and $k \geq 1$, is it decidable whether $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) belongs to $\mathcal{O}(n^k)$, $\Omega(n^k)$, or $\Theta(n^k)$? If so, what is the complexity of the problem?
— *Membership to the hierarchy.* For a given VASS game $\mathcal{A}$, is it decidable whether $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) belongs to $\mathcal{O}(n^k)$ for some $k \geq 1$?
— *Computing the least level in the hierarchy.* For a given VASS game $\mathcal{A}$, can we compute the *least* k (if it exists) such that $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) belongs to $\mathcal{O}(n^k)$? What is the complexity of this problem? If $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) does *not* belong to the polynomial hierarchy, is there any lower bound on its asymptotic growth?
— *Density of the hierarchy.* If $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) does *not* belong to $\mathcal{O}(n^k)$, does it necessarily belong to $\Omega(n^{k+1})$?

As we shall see in Section 4, answers to the above questions are mostly positive, and the complexity of the considered problems is well understood and relatively low.

3.1.2. Grzegorczyk hierarchy. Let $F_1, F_2, F_3, \ldots$ be a family of fast growing functions, where $F_i : \mathbb{N} \to \mathbb{N}$ is defined inductively as follows:

— $F_1(n) = 2n + 1, \quad F_2(n) = n^2, \quad F_3(n) = 2^n$,
— $F_{k+1}(n) = F_k^{(n)}(n)$ for $k \geq 3$.

Here, $F^{(\ell)}(n)$ denotes the ℓ-th iterate of F, i.e., $F(\cdots(F(n))\cdots)$ composed ℓ times, where $F^{(0)}(n) = n$. Observe that F_4 is already non-elementary.

For every $k \geq 1$, the class of functions $\mathcal{G}_k$ is defined by

$$\mathcal{G}_k = \{f : \mathbb{N} \to \mathbb{N} \mid f \leq F_k^{(\mu)} \text{ for some } \mu \in \mathbb{N}\}\,.$$

For every $k \geq 4$, the class $\mathcal{G}_k$ exactly captures the growth of functions in the class $\mathcal{E}_k$ of the standard Grzegorczyk hierarchy [Grzegorczyk 1953] (each function of $\mathcal{E}_k$ is bounded by some function of $\mathcal{G}_k$ for all sufficiently large arguments, and vice versa). Hence, we refer to $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3, \ldots$ as the *Grzegorczyk hierarchy*. Note that $\mathcal{G}_1$, $\mathcal{G}_2$, and $\mathcal{G}_3$ contain all linear, polynomial, and elementary functions, respectively.

Consider a function $f : \mathbb{N} \to \mathbb{N}$. The membership $f \in \mathcal{G}_k$ bounds the growth of f from above, similarly as the membership $f \in \mathcal{O}(n^k)$ for the polynomial hierarchy. It is perhaps not immediately clear what is an appropriate analogue of $f \in \Omega(n^k)$. Intuitively, f achieves the growth of $\mathcal{G}_k$ if it grows at least as fast as the generator F_k of $\mathcal{G}_k$. Technically[2], we say that f is *beyond* F_k if there is a constant $\lambda \in \mathbb{N}$ such that $f(n) \geq F_k(\lfloor n/\lambda \rfloor)$ for all $n \in \mathbb{N}$.

The problems studied for the Grzegorczyk hierarchy are analogous to the ones for the polynomial hierarchy. Let us note that the membership to the Grzegorczyk hierarchy problem is trivial, because if the termination/counter complexity of a d-dimensional VASS game $\mathcal{A}$ is bounded, then it belongs to $\mathcal{G}_{d+1}$. To see this, recall that Angel has a counterless strategy σ such that the demonic VASS $\mathcal{A}^\sigma$ obtained by applying σ has bounded termination/counter complexity. Hence, it suffices to show that the termination/counter complexity of $\mathcal{A}^\sigma$ belongs to $\mathcal{G}_{d+1}$, which follows from [Schmitz 2019].

— *Membership to a given level of the hierarchy.* For a given VASS game $\mathcal{A}$ and $k \geq 1$, is it decidable whether $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) belongs to $\mathcal{G}_k$? If so, what is the complexity of the problem?
— *Computing the least level in the hierarchy.* For a given VASS game $\mathcal{A}$, can we compute the *least* k such that $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) belongs to $\mathcal{G}_k$? What is the complexity of this problem?
— *Density of the hierarchy.* If $\mathcal{L}(n)$ (or $\mathcal{C}[c](n)$) does *not* belong to $\mathcal{G}_k$, is it necessarily beyond F_{k+1}?

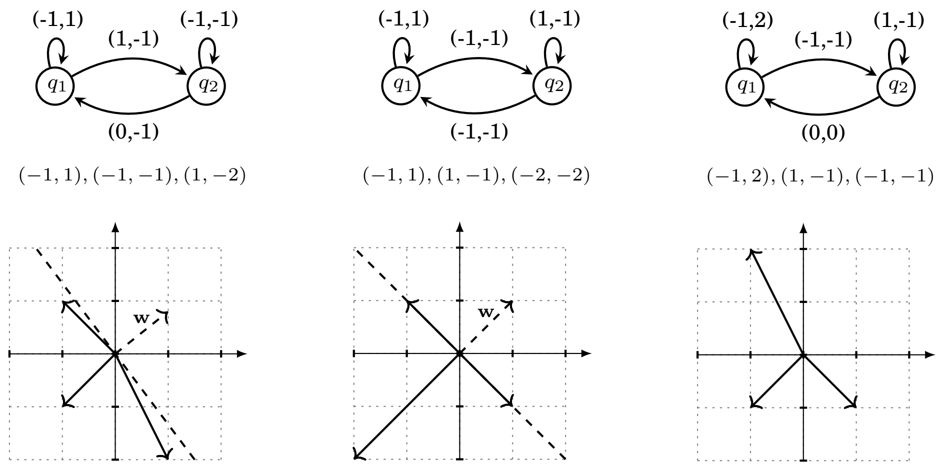
4. RESULTS ABOUT THE POLYNOMIAL HIERARCHY

The problems for the polynomial hierarchy formulated in Section 3.1.1 are mostly resolved and the picture is essentially complete. In the following sections, we give an overview of these results together with some remarks about the underlying proof techniques.

4.1. Linearity of termination complexity for demonic VASS

The linearity of $\mathcal{L}(n)$ for demonic VASS, i.e., the question whether $\mathcal{L} \in \mathcal{O}(n)$, is studied in [Brázdil et al. 2018] where it is shown that the problem is decidable in polynomial

[2]Note that the condition is not trivial, because there exist functions $g \notin \mathcal{G}_k$ that are not beyond F_{k+1}; one example is a function g defined by $g(n) = F_k^{(\lfloor \log n \rfloor)}(n)$.



(a) Positive normal, Inc in open half-space.

(b) Positive normal, Inc in closed half-space.

(c) No positive normal.

Fig. 3. Deciding the linearity of termination complexity.

time. Furhermore, if $\mathcal{L} \notin \mathcal{O}(n)$, then $\mathcal{L} \in \Omega(n^2)$. To get some intuition behind these results, consider a *strongly connected* demonic VASS $\mathcal{A}$ and the set of *increments* defined by

$$Inc = \{\Delta(\varrho) \mid \varrho \text{ is a simple cycle of } \mathcal{A}\}\,.$$

Clearly, the set Inc is finite. Now, we distinguish three cases.

(a) There exists a positive $\mathbf{w} \in \mathbb{R}^d$ such that $\mathbf{u} \cdot \mathbf{w} < 0$ for every $\mathbf{u} \in Inc$. In other words, all increments belong to the open half-space determined by $\mathbf{w}$.
(b) Case (a) does not hold, but there exists a positive $\mathbf{w} \in \mathbb{R}^d$ such that $\mathbf{u} \cdot \mathbf{w} \leq 0$ for every $\mathbf{u} \in Inc$. Hence, all increments belong to the closed half-space determined by $\mathbf{w}$.
(c) Cases (a) and (b) do not hold.

We claim that in Case (a), (b), and (c), we have that $\mathcal{L}(n) \in \mathcal{O}(n)$, $\mathcal{L}(n) \in \Omega(n^2)$, and $\mathcal{L}(n)$ is unbounded, respectively.

First, suppose the condition of Case (a) holds. Then every simple cycle shifts the vector of current counter values in the direction opposite to $\mathbf{w}$ by some amount bounded away from zero. Since a computation of length m can be decomposed into $\Omega(m)$ simple cycles, we obtain that a computation initiated in a configuration $p\vec{n}$ inevitably "hits some axis" (i.e., decreases some counter below zero) after $\mathcal{O}(n)$ transitions. This case is illustrated in Fig. 3a, where $Inc = \{(-1,1),(-1,-1),(1,-2)\}$ and $\mathbf{w} = (1, 0.8)$.

In Case (b), one can show that there always exist increments $\mathbf{u}_1, \ldots, \mathbf{u}_m$ and positive $b_1, \ldots, b_m \in \mathbb{N}$ such that $\sum_{i=1}^{m} b_i \cdot \mathbf{u}_i \geq 0$. The argument is purely geometric and applies not only to Inc but to an *arbitrary* finite set of vectors satisfying the condition of Case (b). In the example of Fig. 3b, we can put $\mathbf{u}_1 = (-1,1)$, $\mathbf{u}_2 = (1,-1)$, and $b_1 = b_2 = 1$. Furthermore, $\mathbf{u}_1$ and $\mathbf{u}_2$ can be used to produce a computation of quadratic length in the way shown in Fig. 4. In a configuration $q_1\vec{n}$, we start by performing $\lfloor n/4 \rfloor$ transitions $\mathbf{u}_1$, then "switch" to q_2, perform $\lfloor n/4 \rfloor$ transitions $\mathbf{u}_2$, switch back to q_1, and so on. The number of repetitions is $\Omega(n)$, and hence the total length of the computation is $\Omega(n^2)$. This construction can be generalized to arbitrary $\mathbf{u}_1, \ldots, \mathbf{u}_m$ and $b_1, \ldots, b_m$.

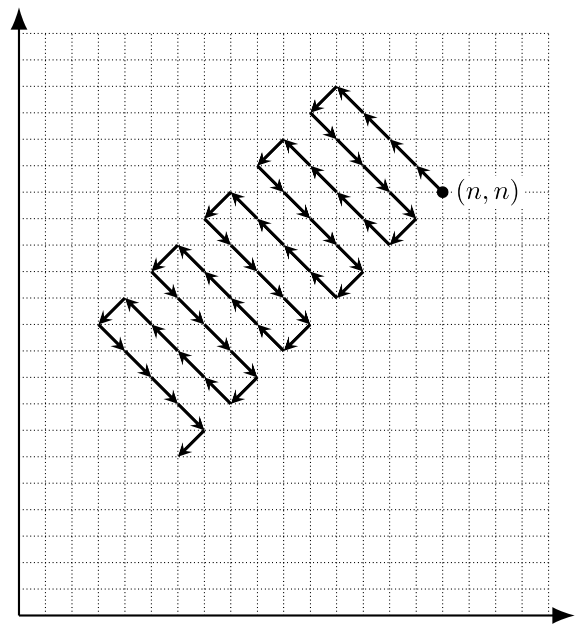


Fig. 4. Constructing a computation of quadratic length for the demonic VASS of Fig. 3b.

In Case (c), there are increments $\mathbf{u}_1, \ldots, \mathbf{u}_m$ and $b_1, \ldots, b_m \in \mathbb{N}$ such that $\sum_{i=1}^{m} b_i \cdot \mathbf{u}_i$ is *positive*. This suffices for constructing an infinite computation from a configuration $p\vec{n}$ for a sufficiently large n, because a (possibly negative) effect of the "switching paths" among the control states can now by compensated by iterating the simple cycles associated to $\mathbf{u}_1, \ldots, \mathbf{u}_m$. In the example of Fig. 3c, we can put $\mathbf{u}_1 = (-1, 2)$, $\mathbf{u}_2 = (1, -1)$, $b_1 = 2$, and $b_2 = 3$.

Although the above case analysis gives a clear intuition when and why the termination complexity of a demonic VASS is asymptoticaly linear, it does not immediatelly lead to a polynomial-time decision algorithm, because the size of Inc is generally exponential in $\|\mathcal{A}\|$. The decision algorithm of [Brázdil et al. 2018] is based on constructing a linear program in the style of [Kosaraju and Sullivan 1988]. An interesting byproduct of this construction is a result saying that $\mathcal{L}(n) \in \mathcal{O}(n)$ iff there exist a positive $\mathbf{c} \in \mathbb{R}^d$, $h : Q \to \mathbb{R}$, and $\varepsilon > 0$ such that the function $f : Q \times \mathbb{N}^d \to \mathbb{R}$ defined by $f(p\mathbf{v}) = \mathbf{c} \cdot \mathbf{v} + h(p)$ decreases at least by ε by performing an arbitrary transition. Hence, the linearity of $\mathcal{L}(n)$ is always witnessed by a *linear weighted ranking function*.

Finally, let us note that if $\mathcal{A}$ is not strongly connected, then the linearity of $\mathcal{L}(n)$ can be decided by analyzing the strongly connected components of $\mathcal{A}$ (see Proposition 5.1 and recall $\mathcal{G}_1 = \mathcal{O}(n)$).

4.2. Deciding polynomial termination and counter complexity for demonic VASS

Polynomial asymptotic growth of the counter complexity for strongly connected demonic VASS is also decidable in polynomial time [Zuleger 2020]. More precisely, we have the following:

THEOREM 4.1 ([ZULEGER 2020]). *Let $\mathcal{A}$ be a d-dimensional strongly connected demonic VASS. For every counter c, we have that either $\mathcal{C}[c](n) \in \Theta(n^k)$ for some $1 \leq k \leq 2^d$, or $\mathcal{C}[c](n) \in 2^{\Omega(n)}$. It is decidable in polynomial time which of the two possibilities holds. In the first case, the k is computable in polynomial time.*

According to Remark 2.1, the results of Theorem 4.1 apply also to the termination complexity.

The algorithm of [Zuleger 2020] iteratively computes counters with higher and higher polynomial complexity by a linear program constructed for a suitably pruned $\mathcal{A}$. By Remark 2.2, the counters of $\mathcal{A}$ are pumpable to their asymptotic values *simultaneously*. For example, if $\mathcal{A}$ has three counters such that $\mathcal{C}[c](n)$ is in $\Theta(n^2)$, $\Theta(n)$, and $2^{\Omega(n)}$ for $c = 1, 2, 3$, the achievable simultaneous growth of the counters can be represented by the "box" of Fig. 5a. Note that the bottom vector $(n^2, n, 2^{\Omega(n)})$ corresponds to the *asymptotic* growth, and hence the value of the second counter can actually be *smaller* than n (say, $\lfloor n/3 \rfloor$) after completing the pumping computation.

A natural idea for extending the decidability result of Theorem 4.1 to general demonic VASS is to decompose the underlying directed graph of $\mathcal{A}$ into strongly connected components (SCCs) and analyze them individually. This approach is elaborated in [Ajdarów and Kučera 2021]. First, Theorem 4.1 is generalized so that the counters can be initialized to arbitrary "polynomial values" such as n^2 or n^3. This is necessary for incorporating the growth achieved in previous SCCs (see Fig. 5b). The counters pumped to $2^{\Omega(n)}$ can subsequently be deleted from $\mathcal{A}$, because they can be made suffcently large to remain $2^{\Omega(n)}$ even after performing pumping computations in all successor SCCs. The deleted counters are indicated by a '$*$' in Fig. 5b.

Observe that every path in the DAG of SCCs determines a unique vector describing the associated asymptotic growth of the counters. For example, there are two paths from the top SCC to the bottom SCC in Fig. 5b, and the two associated vectors $(n^5, n^5, *)$ and $(n^2, *, *)$ are incomparable. In general, the number of such paths in *exponential* in $\|\mathcal{A}\|$. Still, the membership $\mathcal{C}[c](n) \in \Omega(n^k)$ is witnessed by a single path where the c-component of the associated vector is either n^ℓ for some $\ell \geq k$, or $2^{\Omega(n)}$, or $*$. Hence, the problem whether $\mathcal{C}[c](n) \in \Omega(n^k)$ is in **NP**. Similarly, the problems whether $\mathcal{C}[c](n) \in \mathcal{O}(n^k)$ and $\mathcal{C}[c](n) \in \Theta(n^k)$ are in **coNP** and **DP**, respectively[3]. In [Ajdarów and Kučera 2021], the matching lower bounds are provided, which yeilds the following:

THEOREM 4.2 ([AJDARÓW AND KUČERA 2021]). *Let $k \geq 1$. For every demonic VASS $\mathcal{A}$ and a counter c of $\mathcal{A}$, we have that $\mathcal{C}[c](n)$ is either in $\mathcal{O}(n^k)$ or in $\Omega(n^{k+1})$. Furthermore, the problem whether*

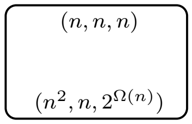
— $\mathcal{C}[c](n) \in \mathcal{O}(n^k)$ *is* **coNP**-*complete for* $k \geq 1$*;*
— $\mathcal{C}[c](n) \in \Omega(n^k)$ *is in* **P** *for* $k = 1$ *and* **NP**-*complete for* $k \geq 2$*;*
— $\mathcal{C}[c](n) \in \Theta(n^k)$ *is* **coNP**-*complete for* $k = 1$ *and* **DP**-*complete for* $k \geq 2$.

Similar results are obtained also for $\mathcal{L}(n)$, but here the **coNP**, **NP**, and **DP** hardness holds for $k \geq 2$, $k \geq 3$, and $k \geq 3$, respectively. We refer to Table I for details.

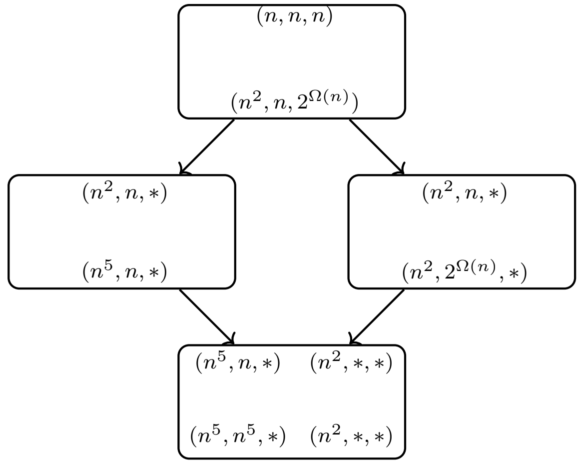
Remark 4.3. The crucial parameter influencing the complexity of the polynomial asymptotic analysis for demonic VASS is the number of different paths in the DAG of SCCs. For every subclass of demonic VASS where the number of different paths in the DAG of SCCs stays polynomial in $\|\mathcal{A}\|$, the problems of Theorem 4.2 are solvable in polynomial time (one example is the subclass of demonic VASS where the DAG of SCCs is a tree).

In program analysis, demonic VASS abstractions of imperative programs are not necessarily strongly connected because of branching constructs that are *not embedded within loops* (note that the demonic VASS of Fig. 1c is strongly connected despite the non-deterministic branches caused by inner loops of the program). If a program

[3]The class **DP** consists of problems that are intersections of one problem in **NP** and one problem in **coNP**. The **DP** class is expected to be somewhat larger than the union of **NP** and **coNP**.



(a) Counter growth in a strongly connected demonic VASS.



(b) Counter growth in a demonic VASS with multiple SCCs.

Fig. 5. Representing simultaneous counter growth in demonic VASS.

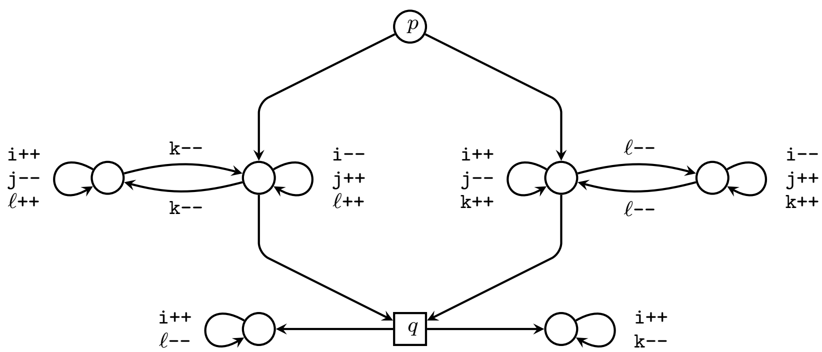


Fig. 6. Angel cannot commit to a counterless strategy when minimizing $\mathcal{C}[\mathtt{i}](n)$.

executes a long sequence of such constructs, the DAG of SCCs becomes a chain of diamond-like structures that is (computationally) hard to analyze. However, for subclasses of programs with a bounded length of such sequences, the associated subclasses of demonic VASS can be analyzed in polynomial time.

4.3. Deciding polynomial termination and counter complexity for VASS games

A crucial step towards the effective analysis of polynomial asymptotic growth of counter complexity in VASS games is identifying the type of strategies sufficient for player Angel. Consider the VASS game of Fig. 6 with four counters i, j, k, ℓ that are incremented or decremented by transitions in the indicated way. If Demon starts a computation in $p\vec{n}$, it may decide to pump either ℓ or k (but not both) to $\Theta(n^2)$. Observe that $\mathcal{C}[\mathtt{i}](n) \in \Theta(n)$, because Angel can decide to increase i either by ℓ or by k (by taking the left or the right outgoing transition of q, respectively). However, the decision depends on which of the two counters has previously been pumped to a quadratic value. In particular, this means that Angel cannot commit to a counterless strategy.

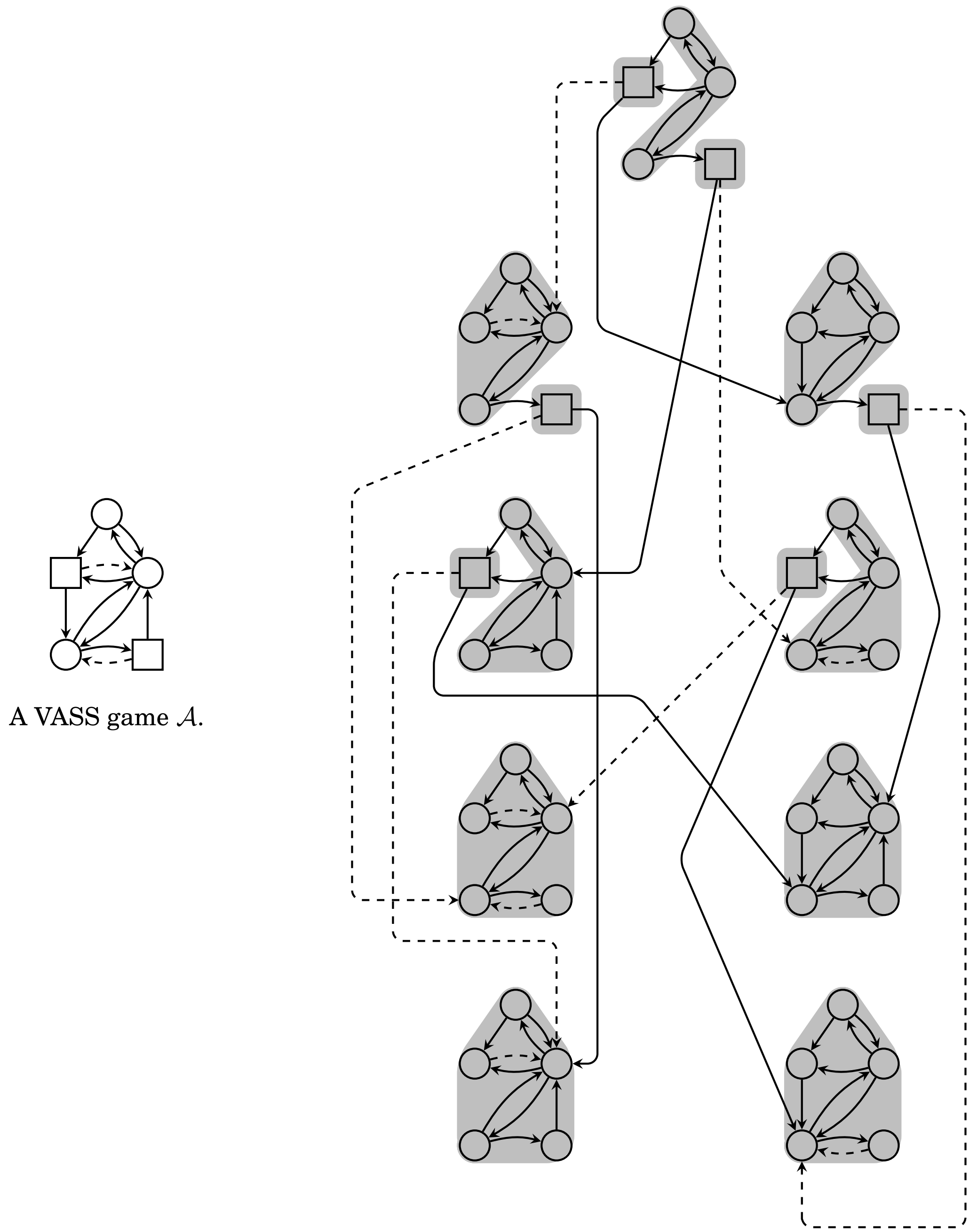


Fig. 7. A VASS game $\mathcal{A}$, where each of the two angelic states has two outgoing transitions indicated by solid/dashed arrows (counter updates are not shown). The corresponding VASS game $\mathcal{B}$ on the right has 45 control states, and the resulting locking decomposition of $\mathcal{A}$ has 15 vertices.

4.3.1. Locking strategy and locking decomposition. As observed in [Ajdarów and Kučera 2021], Angel can safely commit to *simple locking* strategy when minimizing the asymptotic growth of counter complexity. An Angel's strategy is *locking* if whenever an angelic control state p is visited for the first time, the strategy selects and "locks" an outgoing transition of p so that whenever p is revisited, the previously locked transition is taken.

Observe that although a locking strategy eventually behaves like a counterless strategy, the choice of a locked transition may depend on a concrete computational history. The information about the history relevant for making the right choice is actually finite and can be obtained by tracing the history in the *locking decomposition of* $\mathcal{A}$, which is a finite DAG constructed as follows. A *locking set* is a finite set L of transitions such that

— for every $(p, \mathbf{u}, q) \in L$ we have that $p \in Q_A$,
— for every $p \in Q_A$, the set L contains at most one outgoing transition of p.

We use $\mathcal{L}$ to denote the set of all locking sets. For the VASS game $\mathcal{A}$ of Fig. 7 with two angelic control states, the set $\mathcal{L}$ has nine elements—the empty set, four singletons, and four sets with two transitions. The locking decomposition of $\mathcal{A}$ is obtained in two steps:

1. For every $L \in \mathcal{L}$, we take a fresh copy $\mathcal{A}[L]$ of $\mathcal{A}$ where
   (a) the outgoing transitions of all demonic states are the same as in $\mathcal{A}$,
   (b) every angelic state p such that L contains a transition $(p, \mathbf{u}, q)$ is declared as *demonic* in $\mathcal{A}[L]$ with the only outgoing transition $(p, \mathbf{u}, q)$,
   (c) for every angelic state p where (b) does not hold and every transition $(p, \mathbf{u}, q)$, we add a transition $(p, \mathbf{u}, \hat{q})$ where $\hat{q}$ is the copy of q in $\mathcal{A}[L \cup \{(p, \mathbf{u}, q)\}]$.

   Thus, we obtain a new VASS game $\mathcal{B}$. In Fig. 7, the construction of $\mathcal{B}$ is shown for the VASS game $\mathcal{A}$. The resulting VASS game $\mathcal{B}$ has 45 control states.
2. The locking decomposition $\mathcal{D}$ of $\mathcal{A}$ is the DAG of strongly connected components of $\mathcal{B}$. In Fig. 7, the locking decomposition of $\mathcal{A}$ has 15 vertices indicated by gray areas.

A locking strategy is *simple* if the choice of a locked transition depends only on the sequence of previously visited vertices in $\mathcal{D}$. Observe that every vertex of $\mathcal{D}$ is either an angelic control state left by the next transition, or a SCC consisting of only *demonic* states. In the first case, the counter values do not change substantially; in the second case, we can analyze their asymptotic growth by applying the results of Section 4.2. Thus, we evaluate the asymptotic growth of the counters for a given simple locking strategy against an optimal Demon's strategy.

4.3.2. Simple locking strategies are sufficient. The argument of [Ajdarów and Kučera 2021] showing that Angel can safely commit to a simple locking strategy is inductive. Roughly speaking, it is shown that Angel can safely lock an outgoing transition of the first angelic state visited along a computation. After the lock, the angelic state can be seen as demonic, and the induction hypothesis is thus applied to a VASS game with fewer angelic states.

To illustrate the idea, consider a VASS game $\mathcal{A}$ with only *one* angelic state u, where u has two outgoing transitions ℓ and r ("left" and "right", see Fig. 8). We construct a VASS game $\mathcal{B}$ in the way described in Section 4.3.1, i.e., we take the three copies $\mathcal{A}[\emptyset]$, $\mathcal{A}[\{\ell\}]$, $\mathcal{A}[\{r\}]$ so that performing ℓ or r in $\mathcal{A}[\emptyset]$ leads to the respective copy of $\mathcal{A}$ where the transition is locked.

Since Angel's moves are restricted in $\mathcal{B}$, we immediately obtain $\mathcal{C}_{\mathcal{A}}[c](n) \leq \mathcal{C}_{\mathcal{B}}[c](n)$. Now consider $\mathcal{B}$ with a modified semantics where performing an outgoing transition of u in $\mathcal{A}[\emptyset]$ not only locks the transition but also *decreases every counter by one half of its current value*. Let $\mathcal{C}'_{\mathcal{B}}[c](n)$ be the counter complexity of c in $\mathcal{B}$ under this modified semantics. Despite the modification, the asymptotic growth of $\mathcal{C}_{\mathcal{B}}[c](n)$ and $\mathcal{C}'_{\mathcal{B}}[c](n)$ is the same. Furthermore, we have that $\mathcal{C}_{\mathcal{A}}[c](n) \geq \mathcal{C}'_{\mathcal{B}}[c](n)$. To see this, realize that Demon can play in $\mathcal{A}$ in the following way: Before visiting the state u in $\mathcal{A}$, Demon

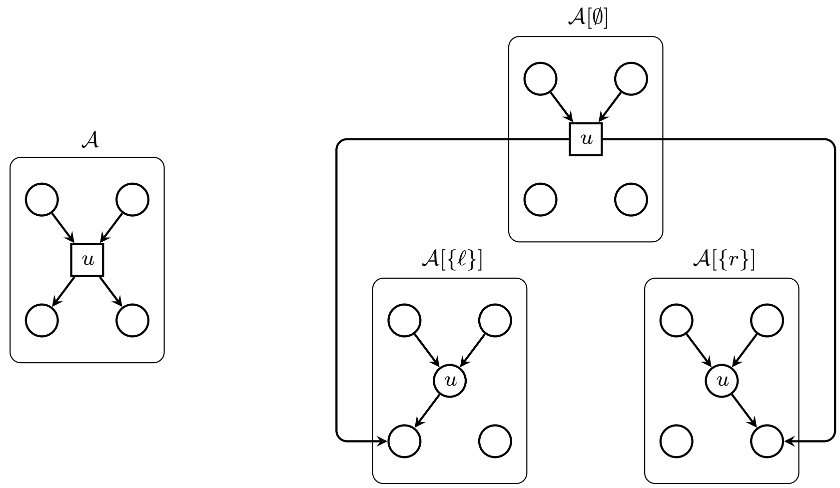


Fig. 8. A simple locking strategy is sufficient.

plays as in $\mathcal{A}[\emptyset]$. When the state u in $\mathcal{A}$ is visited *for the first time*, Demon virtually splits the counters into the *left* and the *right* halfs. Depending on whether Angel selects ℓ or r, Demon starts to simulate its strategy from $\mathcal{A}[\{\ell\}]$ or $\mathcal{A}[\{r\}]$ using the left or the right half of the counters, respectively. This goes on until the state u of $\mathcal{A}$ is revisited, and Demon adjusts the simulation mode according to the transition chosen by Angel. Hence, Demon "interleaves" the strategies borrowed from $\mathcal{A}[\{\ell\}]$ and $\mathcal{A}[\{r\}]$, and thus achieves the required growth of c.

4.3.3. Algorithms. To determine whether $\mathcal{C}[c](n) \in \mathcal{O}(n^k)$, it suffices to check the existence of a simple locking strategy for Angel achieving the $\mathcal{O}(n^k)$ asymptotic growth of c. An exhaustive search for such a strategy in the locking decomposition $\mathcal{D}$ can be implemented by an alternating polynomial-time algorithm (without ever constructing the whole $\mathcal{D}$). Similarly, we can decide whether $\mathcal{C}[c](n) \in \Omega(n^k)$. Consequently, these problems are solvable in polynomial space, and the matching lower bounds are given in [Ajdarów and Kučera 2021]. Thus, we obtain the following:

THEOREM 4.4 ([AJDARÓW AND KUČERA 2021]). *Let $k \geq 1$. For every VASS game $\mathcal{A}$ we have that $\mathcal{C}[c](n)$ is either in $\mathcal{O}(n^k)$ or in $\Omega(n^{k+1})$. Furthermore, the problem whether*

— $\mathcal{C}[c](n) \in \mathcal{O}(n^k)$ *is* **PSPACE**-*complete for* $k{\geq}1$*;*
— $\mathcal{C}[c](n) \in \Omega(n^k)$ *is in* **P** *for* $k{=}1$ *and* **PSPACE**-*complete for* $k{\geq}2$*;*
— $\mathcal{C}[c](n) \in \Theta(n^k)$ *is* **PSPACE**-*complete for* $k{\geq}1$.

Similar results hold also for $\mathcal{L}(n)$, but here the **PSPACE** hardness becomes valid for slightly higher values of k (see Table I).

Remark 4.5. The main structural parameters determining the complexity of the problems of Theorem 4.4 are the number of angelic states and number of different paths in the SCCs of $\mathcal{D}$ consisting of demonic states (see Remark 4.3). In particular, if the number of angelic states is bounded by a constant and the number of different paths in every SCC of $\mathcal{D}$ consisting of demonic states is bounded by a polynomial in $\|\mathcal{A}\|$, the problems of Theorem 4.4 are solvable in polynomial time.

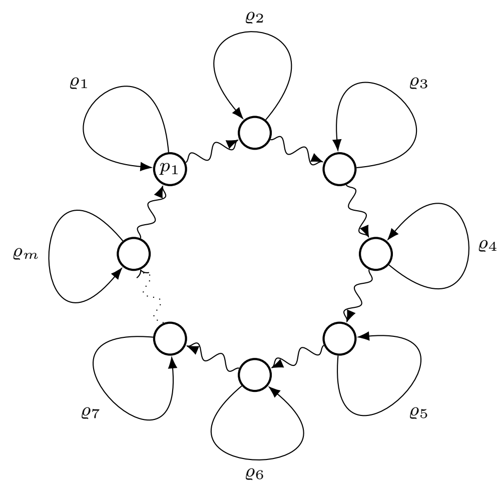


Fig. 9. The structure of iteration scheme $\varrho_1, \ldots, \varrho_m$.

5. RESULTS ABOUT THE GRZEGORCZYK HIERARCHY

One notable difference between the polynomial hierarchy and the Grzegorczyk hierarchy is that $\mathcal{G}_k$ classes are closed under function composition (clearly, the $\mathcal{O}(n^k)$ classes do not have this property for $k \geq 2$). Consequently, demonic VASS can be safely assumed strongly connected when studying the asymptotic growth of $\mathcal{L}(n)$ with respect to the Grzegorczyk hierarchy. More precisely, we have the following:

PROPOSITION 5.1. *Let $k \geq 1$. For every demonic VASS $\mathcal{A}$ we have that $\mathcal{L}_{\mathcal{A}}(n) \in \mathcal{G}_k$ iff $\mathcal{L}_{\mathcal{X}}(n) \in \mathcal{G}_k$ for every SCC $\mathcal{X}$ of $\mathcal{A}$.*

Recall that every SCC of $\mathcal{A}$ can be seen as a VASS (see Section 2.1). Roughly speaking, Proposition 5.1 holds because the counter values stay bounded by a function $F \in \mathcal{G}_k$ when entering an arbitrary SCC $\mathcal{Y}$, and hence the maximal length of the computation before leaving $\mathcal{Y}$ is bounded by $\mathcal{L}_{\mathcal{Y}} \circ F \in \mathcal{G}_k$. Also observe that an analogous proposition for $\mathcal{C}_{\mathcal{A}}[c](n)$ does *not* hold.

5.1. Class $\mathcal{G}_1$

The class $\mathcal{G}_1$ consists of all functions bounded by a linear function, i.e., $\mathcal{G}_1 = \mathcal{O}(n)$. Hence, this case is covered by the results presented in Section 4.

5.2. Class $\mathcal{G}_2$

The class $\mathcal{G}_2$ contains functions bounded by a polynomial. The membership of $\mathcal{L}(n)$ and $\mathcal{C}[c](n)$ to $\mathcal{G}_2$ for strongly connected demonic VASS is decidable in polynomial time by the results of [Leroux 2018].

Let $\mathcal{A}$ be a d-dimensional strongly connected demonic VASS. An *iteration scheme* of $\mathcal{A}$ is a sequence of (not necessarily simple) cycles $\varrho_1, \ldots, \varrho_m$ such that every counter c *decreased* by some ϱ_i is *strictly increased* by the total effect of all $\varrho_1, \ldots, \varrho_m$, i.e., if $\Delta(\varrho_i)(c) < 0$ for some $i \leq m$, then $\sum_{i=1}^{m} \Delta(\varrho_i)(c) > 0$.

THEOREM 5.2 ([LEROUX 2018]). *Let $\mathcal{A}$ be a strongly connected demonic VASS.*

— *$\mathcal{L}(n) \notin \mathcal{G}_2$ iff there is an iteration scheme of $\mathcal{A}$.*
— *$\mathcal{C}[c](n) \notin \mathcal{G}_2$ iff there is an iteration scheme $\varrho_1, \ldots, \varrho_m$ of $\mathcal{A}$ where $\sum_{i=1}^{m} \Delta(\varrho_i)(c) > 0$.*

The "if" direction is simple. We say that a counter is *critical* if it is decreased by some ϱ_i. Since every ϱ_i is a cycle on some control state and $\mathcal{A}$ is strongly connected, we can connect all control states employed in $\varrho_1, \ldots, \varrho_m$ by short paths and thus construct the structure of Fig. 9 (cycles on the same control state are concatenated). Suppose that all counters are set to n initially. Player Demon starts by walking around the dashed circle from p_1 and executes $\kappa \cdot n$ iterations of every ϱ_i for some constant κ. Note that every ϱ_i may decrease some critical counters, so $\kappa > 0$ needs to be suitably small. When Demon arrives back to p_1, the *total* effect of all ϱ_i *increases* every critical counter to at least $(\kappa + 1) \cdot n$. The connecting paths between the control states may potentially decrease the critical counters, but the value of every critical counter after arriving back to p_1 is at least $(1 + \kappa) \cdot n - \tau$, where τ is a constant independent of n. Then, Demon proceeds by another walk around the dashed circle, but since the initial values of all critical counters is higher, the ϱ_i cycles can be executed more often. After completing the second round, the values of the critical counters are thus increased to about $(1 + \kappa)^2 \cdot n$ (the actual increase is somewhat lower). Since Demon can complete $\Omega(n)$ dashed circles and always increase the number of executions of all ϱ_i, the total length of the whole computation is $2^{\Omega(n)}$, and all counters strictly increased by the total effect of $\varrho_1, \ldots, \varrho_m$ are pumped to exponentially large values. The "only if" direction of Theorem 5.2 is more elaborate.

Let T be the set of all transitions appearing in a cycle of some iteration scheme, and let I be the set of all counters strictly increased by the total effect of some iteration scheme. In [Leroux 2018], it is shown that there exists o monotone operator computable in polynomial time such that the pair (I, T) is the greatest fixed-point of this operator. Thus, the following theorem is obtained:

THEOREM 5.3 ([LEROUX 2018]). *Let $\mathcal{A}$ be a strongly connected demonic VASS. The problems whether $\mathcal{L}(n) \in \mathcal{G}_2$ and $\mathcal{C}[c](n) \in \mathcal{G}_2$ are decidable in polynomial time.*

The result for $\mathcal{L}(n)$ actually applies to *all* demonic VASS by Proposition 5.1.

5.3. The classes $\mathcal{G}_k$ where $k \geq 3$

The positive decidability results for $\mathcal{G}_1$ and $\mathcal{G}_2$ are extended to the whole Grzegorczyk hierarchy in [Kučera et al. 2020]. For demonic VASS, the following result is proven:

THEOREM 5.4 ([KUČERA ET AL. 2020]). *Let $\mathcal{A}$ be a strongly connected demonic VASS and $k \geq 1$. The problems whether $\mathcal{L}(n) \in \mathcal{G}_k$ and $\mathcal{C}[c](n) \in \mathcal{G}_k$ are decidable in polynomial time.*

Furthermore, it is shown that if $\mathcal{L}(n)$ or $\mathcal{C}[c](n)$ does *not* belong to $\mathcal{G}_k$, then it is beyond F_{k+1} (see Section 3.1.2). Again, the result for $\mathcal{L}(n)$ holds for all demonic VASS by Proposition 5.1.

For every transition t, the *transition complexity of* t is defined in the same way as the termination complexity; the only difference is that instead of *len*, the function $\#_t$ counting the number of occurrences of t along a computation is used. Clearly, $\mathcal{L}(n)$ is beyond F_k iff the transition complexity of some t is beyond F_k.

A proof of Theorem 5.4 is obtained by designing a polynomial-time recursive procedure computing the sets T_k and I_k of all transitions and counters with transition and counter complexity beyond F_k for a given $k \geq 3$. For $k = 3$, the procedure invokes the algorithm of [Leroux 2018]. For $k > 3$, the procedure starts by computing the sets T_{k-1} and I_{k-1} recursively, and proceeds by eliminating transitions and counters whose complexity is bounded by $F_{k-1}^{(\mu)}$ for some $\mu \in \mathbb{N}$. In this phase, additional recursive procedure calls are executed.

Table I. Algorithmic analysis of termination/counter complexity.

Problem	s.c. demonic VASS	demonic VASS	VASS games
$\mathcal{L}(n) \in \mathcal{O}(n)$	in $\mathbf{P}$	in $\mathbf{P}$	$\mathbf{NP}$-complete
$\mathcal{L}(n) \in \mathcal{O}(n^k), k \geq 2$	in $\mathbf{P}$	$\mathbf{coNP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{L}(n) \in \Omega(n^2)$	in $\mathbf{P}$	in $\mathbf{P}$	$\mathbf{coNP}$-complete
$\mathcal{L}(n) \in \Omega(n^k), k \geq 3$	in $\mathbf{P}$	$\mathbf{NP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{L}(n) \in \Theta(n)$	in $\mathbf{P}$	in $\mathbf{P}$	$\mathbf{NP}$-complete
$\mathcal{L}(n) \in \Theta(n^2)$	in $\mathbf{P}$	$\mathbf{coNP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{L}(n) \in \Theta(n^k), k \geq 3$	in $\mathbf{P}$	$\mathbf{DP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{C}[c](n) \in \mathcal{O}(n^k), k \geq 1$	in $\mathbf{P}$	$\mathbf{coNP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{C}[c](n) \in \Omega(n^k), k \geq 2$	in $\mathbf{P}$	$\mathbf{NP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{C}[c](n) \in \Theta(n)$	in $\mathbf{P}$	$\mathbf{coNP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{C}[c](n) \in \Theta(n^k), k \geq 2$	in $\mathbf{P}$	$\mathbf{DP}$-complete	$\mathbf{PSPACE}$-complete
$\mathcal{L}(n) \in \mathcal{G}_k, k \geq 2$	in $\mathbf{P}$	in $\mathbf{P}$	$\mathbf{NP}$-complete
$\mathcal{C}[c](n) \in \mathcal{G}_k, k \geq 2$	in $\mathbf{P}$	?	?

Although the procedure runs in $\mathcal{O}(poly(n))$ time for every fixed k, the degree of the polynomial $poly(n)$ increases exponentially in k. This means that determining the *least* k such that $\mathcal{L}(n) \in \mathcal{G}_k$ or $\mathcal{C}[c](n) \in \mathcal{G}_k$ takes exponential time (recall that the least k is bounded by $d+1$ where d is the dimension of $\mathcal{A}$; see Section 3.1.2).

For VASS games, the problems whether $\mathcal{L}(n) \in \mathcal{G}_k$ and $\mathcal{C}[c](n) \in \mathcal{G}_k$ are computationally harder. In [Kučera et al. 2020], a full classification is given for $\mathcal{L}(n)$.

THEOREM 5.5 ([KUČERA ET AL. 2020]). *Let $\mathcal{A}$ be a VASS game and $k \geq 1$. The problem whether $\mathcal{L}(n) \in \mathcal{G}_k$ is* $\mathbf{NP}$*-complete.*

In particular, note that the linearity of termination complexity is $\mathbf{NP}$-complete for VASS games, while the linearity of counter complexity is $\mathbf{PSPACE}$-complete by Theorem 4.4. A proof of Theorem 5.5 reveals that Angel can safely commit to a *counterless* strategy when minimizing the complexity of $\mathcal{L}(n)$ with respect to the Grzegorczyk hierarchy, which does not hold for counter complexity.

6. SUMMARY, OPEN PROBLEMS, AND DIRECTIONS FOR FUTURE RESEARCH

The existing results about the termination/counter complexity are summarized in Table I. Since $\mathcal{G}_1 = \mathcal{O}(n)$, the results about $\mathcal{G}_1$ are covered by the first level of the polynomial hierarchy. The demonic VASS and VASS games constructed in the hardness proofs change every counter at most by one in a single transition. Consequently, the complexity bounds of Table I remain valid even if counter update vectors of $\mathcal{A}$ are encoded in *unary* (cf. Section 2.2).

The results of Table I are encouraging. Although the considered problems are not trivial, they are decidable with relatively low complexity. Intuitively, the reason why *all* of the considered problems are solvable in polynomial time for strongly connected demonic VASS is that they all reduce to the existence of cycles satisfying certain properties. The existence of such cycles can be determined by linear programming in the style of [Kosaraju and Sullivan 1988]. Furthermore, even solving the hard problems of Table I becomes computationally feasible for VASS abstractions of imperative programs where certain structural parameters stay small (see Remarks 4.3 and 4.5).

The decidability/complexity of the $\mathcal{C}[c](n) \in \mathcal{G}_k, k \geq 2$ problem is *open* for general demonic VASS and VASS games. We conjecture that the problem is decidable for both models, perhaps by adapting the techniques invented for the polynomial hierarchy [Ajdarów and Kučera 2021].

The polynomial hierarchy is *dense* in the sense that if the termination/counter complexity of a given VASS game is not in $\mathcal{O}(n^k)$, then it is in $\Omega(n^{k+1})$. Similarly, for the Grzegorczyk hierarchy we have that if the termination complexity is not in $\mathcal{G}_k$, then it is beyond F_{k+1}. We conjecture that the same holds for counter complexity, but so far the hypothesis has been proven only for strongly connected demonic VASS.

Counterless strategies are insufficient when Angel strives to achieve the $\mathcal{O}(n^k)$ termination/counter complexity. Nevertheless, Angel can safely commit to a simple locking strategy. On the other hand, counterless strategies are sufficient for achieving the termination complexity in $\mathcal{G}_k$, but this results does not hold for counter complexity. An interesting open question is whether simple locking strategies suffice for minimizing the counter complexity also in the Grzegorczyk hierarchy (we conjecture the answer is positive).

Apart solving the aforementioned open problems, a challenging line for future research is extending the above results to VASS with stochastic control states, i.e., to VASS Markov decision processes and VASS stochastic games. So far, the only existing result in this direction is [Brázdil et al. 2019] where the linearity of expected termination time is shown decidable in polynomial time for VASS MDPs with tree-like MEC decomposition. The analysis of stochastic VASS models is intricate, because the studied questions often subsume notoriously hard problems, and even solving constrained variants of these problems requires advanced tools and new ideas. Still, the challenge appears worth taking on, because efficient decision algorithms would allow for solving advanced questions of infinite-state probabilistic program analysis beyond the reach of existing methods.

ACKNOWLEDGMENTS

The work is supported by the Czech Science Foundation, Grant No. 21-24711S.

REFERENCES

M. Ajdarów and A. Kučera. 2021. Deciding Polynomial Termination Complexity for VASS Programs. In *Proceedings of CONCUR 2021 (Leibniz International Proceedings in Informatics)*, Vol. 203. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 30:1–30:15.

T. Brázdil, K. Chatterjee, A. Kučera, P. Novotný, and D. Velan. 2019. Deciding Fast Termination for Probabilistic VASS with Nondeterminism. In *Proceedings of ATVA 2019 (Lecture Notes in Computer Science)*, Vol. 11781. Springer, 462–478.

T. Brázdil, K. Chatterjee, A. Kučera, P. Novotný, D. Velan, and F. Zuleger. 2018. Efficient Algorithms for Asymptotic Bounds on Termination Time in VASS. In *Proceedings of LICS 2018*. ACM Press, 185–194.

T. Brázdil, P. Jančar, and A. Kučera. 2010. Reachability Games on Extended Vector Addition Systems with States. In *Proceedings of ICALP 2010, Part II (Lecture Notes in Computer Science)*, Vol. 6199. Springer, 478–489.

M. Broy and M. Wirsing. 1981. On the Algebraic Specification of Nondeterministic Programming Languages. In *Proceedings of CAAP'81 (Lecture Notes in Computer Science)*, Vol. 112. Springer, 162–179.

K. Chatterjee, L. Doyen, T. Henzinger, and J.-F. Raskin. 2010. Generalized Mean-payoff and Energy Games. In *Proceedings of FST&TCS 2010 (Leibniz International Proceedings in Informatics)*, Vol. 8. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 505–516.

W. Czerwiński, S. Lasota, R. Lazić, J. Leroux, and F. Mazowiecki. 2019. The Reachability Problem for Petri Nets Is Not Elementary. In *Proceedings of STOC 2019*. ACM Press, 24–33.

A. Grzegorczyk. 1953. Some Classes of Recursive Functions. *Rozprawy Matematyczne* 4 (1953), 1–45.

J.E. Hopcroft and J.-J. Pansiot. 1979. On the Reachability Problem for 5-dimensional Vector Addition Systems. *Theoretical Computer Science* 8 (1979), 135–159.

S.R. Kosaraju and G. Sullivan. 1988. Detecting Cycles in Dynamic Graphs in Polynomial Time. In *Proceedings of STOC'88*. ACM Press, 398–406.

A. Kučera, J. Leroux, and D. Velan. 2020. Efficient Analysis of VASS Termination Complexity. In *Proceedings of LICS 2020*. ACM Press, 676–688.

J. Leroux. 2018. Polynomial Vector Addition Systems With States. In *Proceedings of ICALP 2018 (Leibniz International Proceedings in Informatics)*, Vol. 107. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 134:1–134:13.

J. Leroux and Ph. Schnoebelen. 2014. On Functions Weakly Computable by Petri Nets and Vector Addition Systems. In *Reachability Problems (Lecture Notes in Computer Science)*, Vol. 8762. Springer, 190–202.

R. Lipton. 1976. *The Reachability Problem Requires Exponential Space*. Technical report 62. Yale University.

E.W. Mayr and A.R. Meyer. 1981. The Complexity of the Finite Containment Problem for Petri Nets. *Journal of the Association for Computing Machinery* 28, 3 (1981), 561–576.

S. Schmitz. 2019. The Parametric Complexity of Lossy Counter Machines. In *Proceedings of ICALP 2018 (Leibniz International Proceedings in Informatics)*, Vol. 132. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 129:1–129:15.

M. Sinn, F. Zuleger, and H. Veith. 2013. A Simple and Scalable Static Analysis for Bound Analysis and Amortized Complexity Analysis. In *Proceedings of CAV 2014 (Lecture Notes in Computer Science)*, Vol. 8559. Springer, 745–761.

M. Sinn, F. Zuleger, and H. Veith. 2017. Complexity and Resource Bound Analysis of Imperative Programs Using Difference Constraints. *Journal of Automated Reasoning* 59, 1 (2017), 3–45.

F. Zuleger. 2020. The Polynomial Complexity of Vector Addition Systems with States. In *Proceedings of FoSSaCS 2020 (Lecture Notes in Computer Science)*, Vol. 12077. Springer, 622–641.