Reachability in Fixed-Dimensional Continuous VASS

Michal Ajdarów
University of Liverpool, UK

A. R. Balasubramanian
Max Planck Institute for Software Systems (MPI-SWS), Kaiserslautern, Germany

Łukasz Orlikowski
University of Warsaw, Poland

Abstract

Vector Addition System with States (VASS) are a ubiquitous model of infinite-state systems consisting of a set of non-negative counters which can be incremented and decremented. It is known that the reachability problem for VASS is Ackermann-complete. Because of this huge complexity, various over-approximations of VASS have been studied in the literature. One such over-approximation is continuous VASS (CVASS), in which the counters are (non-negative) rational numbers and whenever a vector is added to the current counter values, it is first scaled with an arbitrarily chosen rational factor between zero and one. It is known that the reachability problem for CVASS is NP-complete.

In this paper, we initiate the study of fixed-dimensional CVASS, i.e., CVASS with a fixed number of counters. We study both the reachability and coverability problems, under both unary and binary encodings as well as over both the non-negative and the rational semantics. This gives rise to a collection of eight different problems. As our main result, we prove a complexity dichotomy for all of these eight problems when the transition vectors are over the rationals: For dimension 1, all of the eight problems are in AC^1, and so within P, whereas for any dimension at least 2, all of the eight problems are NP-complete. Furthermore, the hardness holds even when the underlying automaton is acyclic. To achieve this hardness result, we present a new technique called the Egyptian prime fractions technique.

Finally, we also study these problems when the transition vectors are over the integers. Except for dimension 2, we classify the complexity of these problems over the non-negative semantics: For dimension 1, all of the problems are in AC^1, whereas for dimensions 3 and above, all of the problems are NP-complete.

2012 ACM Subject Classification Theory of computation → Models of computation; Theory of computation → Automata over infinite objects

Keywords and phrases Continuous Counters, Vector Addition Systems, Complexity, Reachability

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2026.7

Related Version *Full Version*: https://arxiv.org/abs/2606.30042

Funding *Michal Ajdarów*: Supported by the EPSRC project EP/X042596/1.
A. R. Balasubramanian: This research was sponsored in part by the Deutsche Forschungsgemeinschaft project 389792660 TRR 248–CPEC.
Łukasz Orlikowski: Supported by the ERC grant INFSYS, agreement no. 950398.

Acknowledgements The research for this work was carried out in part at the Autobóz Research Workshop in 2025 in Grenaa, Denmark.

1 Introduction

Vector Addition System with States (VASS) (or equivalently Petri nets) are one of the most well-studied models of infinite-state systems with applications in logic, automata, verification and modeling concurrent and business process [23, 12, 14, 26]. A VASS consists of finitely many states along with finitely many counters ranging over the natural numbers. The

transitions of a VASS allow it to move from one state to another, whilst incremeting or decrementing the values of the counters (provided the new values are also natural numbers). One of the most important problems studied for VASS is the *reachability* problem, which asks if a given source configuration $(p, \mathbf{u})$ can reach a given target configuration $(q, \mathbf{v})$. After decades of research, the complexity of the reachability problem for VASS was finally settled to be Ackermann-complete [20, 10].

Due to this huge complexity of the reachability problem, various *over-approximations* of the reachability relation for VASS have been studied in the literature (See [4] for a survey). Since usually the target configuration in the reachability problem is an error configuration that we do not want the source configuration to reach, if we show that this error is not reachable in an over-approximation, then the same applies for the original VASS as well. If the over-approximation is tractable, then this provides a method to prove safety that trades completeness for efficient analysability.

One particular over-approximation that has proven to be quite useful in recent years is the *continuous semantics* (equivalently also called as continuous VASS or CVASS) [11, 7]. In this semantics, the transitions of a VASS are allowed to be fired *fractionally*. This means that, to fire a transition t, we first pick a fraction $\alpha \in (0, 1]$ and then execute the updates of t by the fraction α, i.e., if the transition t originally incremented (resp. decremented) some counter by a value c, it will now be incremented (resp. decremented) by the value $\alpha \cdot c$. (Hence picking $\alpha = 1$ is the same as performing the original transition). Because of this *fractional firing*, counter values can now be (non-negative) rational numbers. Depending on whether we allow the counters to only be non-negative or allow them to go below zero, we get two types of continuous semantics, namely the $\mathbb{Q}_+$-semantics and $\mathbb{Q}$-semantics respectively.

It turns out that both types of continuous semantics are remarkably well-behaved in terms of complexity. Reachability for both of them is NP-complete and is hence much lower than the usual semantics of VASS [7]. Besides this tractability result, the continuous semantics is quite useful in analysing practical benchmarks for problems related to reachability [13, 6], has applications in modeling client-server systems [21], biological networks and biochemical settings [16, 17, 18] and in parameterized verification [2, 1]. However, despite all of this wealth of positive results on continuous VASS, not much research has been devoted to studying them over fixed-dimensions, i.e., studying continuous VASS over a fixed number of counters. This situation is especially surprising, considering the fact that for VASS and many of its over-approximations, better complexity results are known for small dimensions in different settings [5, 9, 15, 3].

In this paper, we initiate the systematic study of the reachability as well as the coverability problem in fixed-dimensional continuous VASS. In the latter problem, we are given two configurations $(p, \mathbf{u})$ and $(q, \mathbf{v})$ and we want to check if $(p, \mathbf{u})$ can reach a configuration of the form $(q, \mathbf{v}')$ for some $\mathbf{v}' \geq \mathbf{v}$. We study both of these problems for both the $\mathbb{Q}_+$-semantics as well as the $\mathbb{Q}$-semantics, under both the unary and the binary encoding of numbers. (Hence, we get eight different problems, by picking one attribute in each of the three categories).

Our main result is a complete complexity classification of all of the eight problems for any fixed dimension, when the updates on the transitions are allowed to be rational numbers. Namely, first we show that for dimension 1, all of the above eight problems are in AC^1, and so within P. Then, we show that for any dimension $d \geq 2$, all of the above eight problems are NP-complete, and hence as hard as the case of an arbitrary number of counters. To achieve the desired NP-hardness result, we introduce a novel technique called the *Egyptian prime fractions* technique, where we use *sums of reciprocals of primes* to uniquely encode the assignments of a propositional formula into a single rational-valued counter.

Prior to our results, only an NC^2 upper bound was known for 1-dimensional CVASS [8, Theorem 1], which we improve to an AC^1 upper bound in this paper. Also, prior to our results, NP-hardness was only known for reachability for 2-dimensional CVASS over binary encoding [7, Lemma 4.13]. We strengthen this result to apply even for coverability over the unary encoding. Furthermore, our NP-hardness result, for all of the eight problems, already applies to *acyclic* continuous VASS, i.e., even when the underlying control structure of the continuous VASS is acyclic. This presents the first NP-hardness result for acyclic unary fixed-dimensional counter systems, as well as the first NP-hardness result for 2-dimensional unary VASS and its over-approximations.

Finally, we also study these problems for the $\mathbb{Q}_+$-semantics when the updates on the transitions are restricted to be integers. In this case, we present a complexity classification of these problems (reachability/coverability, unary/binary encoding) for any fixed dimension, except for the case of dimension 2: For dimension 1, all four problems are in AC^1, whereas for dimensions 3 and above, all four problems are NP-complete. Once again, the hardness results already hold for 3-dimensional acyclic unary continuous VASS. This is in stark contrast to VASS (and some of its other over-approximations and variants), where coverability over the unary encoding for any fixed dimension is in NL [15, 3, 22].

2 Preliminaries

Basic Notions. We use $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{Q}_+$ for the sets of non-negative integers, integers, rational numbers and non-negative rational numbers, respectively. For $a \geq 1$, let $[a] := \{1, 2, \ldots, a\}$. For a vector $\mathbf{v} \in \mathbb{Q}^d$, we denote its j-th coordinate by $\mathbf{v}[j]$. For $\mathbf{u}, \mathbf{v} \in \mathbb{Q}^d$, we write $\mathbf{u} \geq \mathbf{v}$ if $\mathbf{u}[i] \geq \mathbf{v}[i]$ for all $i \in [d]$.

Continuous Vector Addition System with States. Let $d \in \mathbb{N}$. A d-dimensional continuous vector addition system with states (d-CVASS) is a tuple $\mathcal{M} = (Q, T)$ where Q is a finite set of control states and $T \subseteq Q \times \mathbb{Q}^d \times Q$ is a finite set of transitions. $\mathcal{M}$ is called acyclic, if the directed graph induced by $\mathcal{M}$ with Q as vertices and edges corresponding to the transitions T, is an acyclic graph.

Intuitively, the d-CVASS $\mathcal{M}$ is a system with d counters, each of them storing a rational value. Its transitions update the counters by incrementing or decrementing them in a specific manner, as described below. A *configuration* of $\mathcal{M}$ is a pair $p(\mathbf{v})$ where $p \in Q$ is a state and $\mathbf{v} \in \mathbb{Q}^d$ is a vector representing the current values of the d counters. Given two configurations $p(\mathbf{u})$ and $q(\mathbf{v})$, we say that there is a *step* from $p(\mathbf{u})$ to $q(\mathbf{v})$ if there exists a transition $t = (p, \mathbf{w}, q)$ and a non-zero fraction $\alpha \in (0, 1]$ such that $\mathbf{v} = \mathbf{u} + \alpha \cdot \mathbf{w}$. In this case, we denote it by either $p(\mathbf{u}) \xrightarrow{\alpha t}_{\mathbb{Q}} q(\mathbf{v})$, or just as $p(\mathbf{u}) \to_{\mathbb{Q}} q(\mathbf{v})$. Moreover, we use $p(\mathbf{u}) \to_{\mathbb{Q}_+} q(\mathbf{v})$ to denote that $p(\mathbf{u}) \to_{\mathbb{Q}} q(\mathbf{v})$ and additionally both $\mathbf{u} \geq 0$ and $\mathbf{v} \geq 0$. These step relations are called the $\mathbb{Q}$-semantics ($\to_{\mathbb{Q}}$) and the $\mathbb{Q}_+$-semantics ($\to_{\mathbb{Q}_+}$), respectively.

A $\mathbb{Q}$-run (also called a run over the $\mathbb{Q}$-semantics) is a sequence of configurations $q_0(\mathbf{u}_0), q_1(\mathbf{u}_1), \ldots, q_l(\mathbf{u}_l)$ such that $q_{i-1}(\mathbf{u}_{i-1}) \to_{\mathbb{Q}} q_i(\mathbf{u}_i)$ for every $i \in [l]$. The notion of a $\mathbb{Q}_+$-run is defined analogously. We say that $p(\mathbf{u})$ can reach $q(\mathbf{v})$ under the $\mathbb{Q}$-semantics (resp. $\mathbb{Q}_+$-semantics) if there exists a $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $p(\mathbf{u})$ to $q(\mathbf{v})$. We use $\xrightarrow{*}_{\mathbb{Q}}$ (resp $\xrightarrow{*}_{\mathbb{Q}_+}$) to denote the $\mathbb{Q}$-reachability relation (resp. $\mathbb{Q}_+$-reachability relation) between configurations. Finally, we say that $p(\mathbf{u})$ can cover $q(\mathbf{v})$ over the $\mathbb{Q}$-semantics (resp. $\mathbb{Q}_+$-semantics) if there is a $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $p(\mathbf{u})$ to some $q(\mathbf{v}')$ with $\mathbf{v}' \geq \mathbf{v}$.

Reachability and Coverability. The main focus of this paper are the reachability and coverability problems for CVASS in fixed dimension, where the number of counters d is fixed and not part of the input. We now define these problems formally.

d-Reach$_{\mathbb{Q}}$

Input: A d-CVASS and two of its configurations $p(\mathbf{u}), q(\mathbf{v})$.

Question: Can $p(\mathbf{u})$ reach $q(\mathbf{v})$ over the $\mathbb{Q}$-semantics?

d-Cover$_{\mathbb{Q}}$

Input: A d-CVASS and two of its configurations $p(\mathbf{u}), q(\mathbf{v})$.

Question: Can $p(\mathbf{u})$ cover $q(\mathbf{v})$ over the $\mathbb{Q}$-semantics?

The problems d-Reach$_{\mathbb{Q}_+}$ and d-Cover$_{\mathbb{Q}_+}$ are defined analogously, with $\xrightarrow{*}_{\mathbb{Q}}$ replaced by $\xrightarrow{*}_{\mathbb{Q}_+}$. We further distinguish these problems by *unary* and *binary* encodings, depending on how the input is encoded. Thus, for each fixed d, we obtain eight variants, determined by reachability vs. coverability, $\mathbb{Q}$- vs. $\mathbb{Q}_+$-semantics, and unary vs. binary encoding.

The main result of this paper is a complete complexity classification of these eight problems. This requires the (logspace-uniform) complexity class AC^1, defined as the set of decision problems solvable by polynomial-sized unbounded fan-in Boolean circuits with depth $O(\log n)$. It is known that $\mathsf{NL} \subseteq \mathsf{AC}^1 \subseteq \mathsf{P}$ [27, Fig. 4.6]. Also, iterated integer multiplication, i.e., the task of multiplying a given set of n integers, each encoded in binary using at most n bits, is also known to be in AC^1 [19, Corollary 4.1].

We are now ready to state the two main results of the paper, which are as follows. (See the top part of Table 1 for a succinct description of these results).

▶ **Theorem 1.** *1-*Reach$_{\mathbb{Q}}$*, 1-*Cover$_{\mathbb{Q}}$*, 1-*Reach$_{\mathbb{Q}_+}$ *and 1-*Cover$_{\mathbb{Q}_+}$ *are all in* AC^1 *for both unary and binary encoding.*

▶ **Theorem 2.** *For any $d \geq 2$, d-*Reach$_{\mathbb{Q}}$*, d-*Cover$_{\mathbb{Q}}$*, d-*Reach$_{\mathbb{Q}_+}$ *and d-*Cover$_{\mathbb{Q}_+}$ *are all* NP*-complete for both unary and binary encoding.*

In addition to this main result, we also consider a restricted version of CVASS where we only allow the transition vectors to range over the integers. Formally, we say that a d-CVASS $\mathcal{M} = (Q, T)$ is a CVASS *with integer updates* if $T \subseteq Q \times \mathbb{Z}^d \times Q$, i.e., all the vectors appearing in the transitions are over $\mathbb{Z}^d$. Our results in this regard are the following: (See the bottom part of Table 1 for a succinct description of these results).

▶ **Theorem 3.** *For CVASS with integer updates, 1-*Reach$_{\mathbb{Q}}$*, 1-*Cover$_{\mathbb{Q}}$*, 1-*Reach$_{\mathbb{Q}_+}$ *and 1-*Cover$_{\mathbb{Q}_+}$ *are* NL*-complete over the unary encoding and in* AC^1 *over the binary encoding.*

▶ **Theorem 4.** *For CVASS with integer updates, 2-*Reach$_{\mathbb{Q}}$*, 2-*Cover$_{\mathbb{Q}}$*, 2-*Reach$_{\mathbb{Q}_+}$ *and 2-*Cover$_{\mathbb{Q}_+}$ *are* NP*-complete over the binary encoding.*

▶ **Theorem 5.** *For CVASS with integer updates, for any $d \geq 3$, d-*Reach$_{\mathbb{Q}_+}$ *and d-*Cover$_{\mathbb{Q}_+}$ *are* NP*-complete for both unary and binary encoding.*

We note that all of the NP-hardness results hold even when the CVASS is acyclic.

All of the NP membership results in the above theorems follow from the fact that reachability and coverability for d-CVASS (over both types of encoding and both semantics) is in NP for any d [7, Section IV.B]. Hence, the rest of this paper is dedicated to proving all of the other results of the theorems (or equivalently all of the other results in Table 1). Since there are a lot of problems that we need to consider, we make a remark that roughly reduces the number of problems by half.

Table 1 Complexity of reachability and coverability in fixed-dimensional CVASS; A row depicts all possible combinations of problems that can be obtained by the entries in that row; for instance the first row depicts all of the eight possible problems for dimension 1, and for all of them, the upper bound is AC^1.

Problem	Semantics	Encoding	Dimension	Transitions	Complexity	Theorem
Coverability/Reachability	$\mathbb{Q}_+$ / $\mathbb{Q}$	Unary / Binary	1	Rational Updates	in AC^1	Theorem 1
Coverability/Reachability	$\mathbb{Q}_+$ / $\mathbb{Q}$	Unary / Binary	≥ 2	Rational Updates	NP-complete	Theorem 2
Coverability/Reachability	$\mathbb{Q}_+$ / $\mathbb{Q}$	Unary	1	Integer Updates	NL-complete	Theorem 3
Coverability/Reachability	$\mathbb{Q}_+$ / $\mathbb{Q}$	Binary	1	Integer Updates	in AC^1	Theorem 3
Coverability/Reachability	$\mathbb{Q}_+$ / $\mathbb{Q}$	Binary	≥ 2	Integer Updates	NP-complete	Theorem 4
Coverability/Reachability	$\mathbb{Q}_+$	Unary / Binary	≥ 3	Integer Updates	NP-complete	Theorem 5

▶ Remark 6. We can easily reduce coverability to reachability in logarithmic space: Given a d-CVASS $\mathcal{M} = (Q, T)$ with configurations $p(\mathbf{u})$ and $q(\mathbf{v})$, we modify $\mathcal{M}$ by adding a new transition which moves from q to a new state q'. At q', we add d self-loops, each decrementing one of the counters by 1. It is easy to see that (over any of the semantics) $p(\mathbf{u})$ can cover $q(\mathbf{v})$ in $\mathcal{M}$ iff (over the corresponding semantics) $p(\mathbf{u})$ can reach $q'(\mathbf{v})$.

The rest of this paper is structured as follows: In Section 3, we prove Theorems 1 and 3. In Section 4, we prove Theorems 2 and 4. Finally, in Section 5, we prove Theorem 5.

3 Upper Bounds for 1-CVASS

In this section, we prove Theorems 1 and 3. To this end, we prove the following main result.

▶ **Theorem 7.** *1-*Reach$_{\mathbb{Q}}$ *and 1-*Reach$_{\mathbb{Q}_+}$ *are both*
- NL*-complete for 1-CVASS with integer updates over the unary encoding*
- *and in* AC^1 *for 1-CVASS over both unary and binary encoding.*

By Theorem 7 and Remark 6, Theorems 1 and 3 immediately follow. (The NL-hardness mentioned in Theorem 3 is inherited from graph reachability). So, the rest of this section is dedicated to proving Theorem 7. For this, we first make a remark.

▶ Remark 8. Since we will be dealing with a single counter for the rest of this section, we will use variables like x, y instead of their boldface counterparts $\mathbf{x}, \mathbf{y}$ to denote counter values.

Now, to prove the upper bounds for 1-CVASS, it suffices to give the algorithms in the case when the input instance is of the form $(\mathcal{M}, p(x), q(y))$ with $x \leq y$. Indeed, if we are given an instance where $x > y$, we can consider the *reverse* CVASS $\mathcal{M}^\dagger$ of $\mathcal{M}$ which has the same states as $\mathcal{M}$ with the transitions reversed: (p', w, q') is a transition of $\mathcal{M}$ iff $(q', -w, p')$ is a transition of $\mathcal{M}^\dagger$. It is immediately seen that there is a $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $p(x)$ to $q(y)$ in $\mathcal{M}$ iff there is a $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $q(y)$ to $p(x)$ in $\mathcal{M}^\dagger$.

We now move on to proving Theorem 7. We do this in three stages: First, we show that reachability for 1-CVASS over $\mathbb{Q}_+$-semantics (and indeed over a slight generalization of the $\mathbb{Q}_+$-semantics) can be reduced to $\mathbb{Q}$-semantics. In the second stage, we give a characterization of reachability in 1-CVASS over $\mathbb{Q}$-semantics. Then, in the final stage, we use this characterization to give the required upper bounds for 1-CVASS. We note that some of the arguments in the first two stages are similar to some of the arguments in [8], but our presentation makes it simpler to arrive at the upper bound. Furthermore, in the third stage, we improve upon the NC^2 bound in [8] to an AC^1 bound.

Stage 1: Reducing $\mathbb{Q}_+$-semantics to $\mathbb{Q}$-semantics

In the first stage, we shall show that reachability for 1-CVASS over the $\mathbb{Q}_+$-semantics reduces to reachability over the $\mathbb{Q}$-semantics. To this end, we consider a slight generalization of the $\mathbb{Q}_+$-semantics so that it also covers an extension of 1-CVASS as considered by [8].

Let $\mathcal{I}$ be an interval over $\mathbb{Q}$, i.e., $\mathcal{I}$ is of the form $(a,b),(a,b],[a,b)$ or $[a,b]$ for some $a \leq b$ with $a,b \in \{-\infty,\infty\} \cup \mathbb{Q}$, with each of them interpreted the usual way. Given a 1-CVASS $\mathcal{M} = (Q,T)$, the $\mathcal{I}$-semantics of $\mathcal{M}$ is the restriction of the $\mathbb{Q}$-semantics to only configurations of the form $p(x)$ with $p \in Q$ and $x \in \mathcal{I}$, i.e., we say that $p(x) \to_{\mathcal{I}} q(y)$ if $p(x) \to_{\mathbb{Q}} q(y)$ and $x, y \in \mathcal{I}$. Similar to $\mathbb{Q}$-semantics, we can now define runs over the $\mathcal{I}$-semantics and the reachability problem 1-Reach$_{\mathcal{I}}$ for it. Note that the $\mathbb{Q}_+$-semantics is simply the $\mathcal{I}$-semantics where $\mathcal{I} = [0,\infty)$. As our main result of this stage, we show that

▶ **Theorem 9.** *For any interval $\mathcal{I}$, 1-Reach$_{\mathcal{I}}$ can be reduced in logarithmic space to 1-Reach$_{\mathbb{Q}}$. Furthermore the reduction preserves the type of encoding (unary or binary) and the domain of the update vectors (rationals or integers).*

Proof Sketch. Let $\mathcal{M} = (Q,T)$ be an instance of reachability in 1-CVASS with two configurations $p(x)$ and $q(y)$ and let $\mathcal{I}$ be an interval. The same argument mentioned in Remark 8 allows us to assume that $x \leq y$. We shall now construct another 1-CVASS $\mathcal{M}'$ with two configurations $p'(x')$ and $q'(y')$ such that $p(x) \xrightarrow{*}_{\mathcal{I}} q(y)$ in $\mathcal{M}$ iff $p'(x') \xrightarrow{*}_{\mathbb{Q}} q'(y')$ in $\mathcal{M}'$.

First note that if at least one of $x, y \notin \mathcal{I}$, then there can be no $\mathcal{I}$-run from $p(x)$ to $q(y)$ in $\mathcal{M}$. In that case, we can simply take $(\mathcal{M}', p'(x'), q'(y'))$ to be some trivial no-instance of 1-Reach$_{\mathbb{Q}}$. Hence, we can assume that $x, y \in \mathcal{I}$. The required construction of $\mathcal{M}'$ is now obtained by means of a case analysis on x, y. To this end, we say that x (resp. y) is an extreme point of $\mathcal{I}$ if $\mathcal{I} = [x,b)$ or $\mathcal{I} = [x,b]$ for some b (resp. if $\mathcal{I} = (a,y]$ or $\mathcal{I} = [a,y]$ for some a). Our case analysis considers all four possibilities.

Case 1: x, y are not extreme points of $\mathcal{I}$. Therefore, there exists an $\epsilon > 0$ such that $[x-\epsilon, y+\epsilon] \subseteq \mathcal{I}$. In this case, we show that $p(x) \xrightarrow{*}_{\mathcal{I}} q(y)$ is possible in $\mathcal{M}$ iff $p(x) \xrightarrow{*}_{\mathbb{Q}} q(y)$ is possible in $\mathcal{M}$, which then trivially gives the desired reduction. The left-to-right implication is immediate. For the other side, suppose $p(x) := p_0(x_0) \xrightarrow{\alpha_1 t_1}_{\mathbb{Q}} p_1(x_1) \xrightarrow{\alpha_2 t_2}_{\mathbb{Q}} p_2(x_2) \dots \xrightarrow{\alpha_n t_n}_{\mathbb{Q}} p_n(x_n) := q(y)$ is a run over the $\mathbb{Q}$-semantics. We say that the transition $t_i := (p_{i-1}, w_i, p_i)$ is positive, negative or neutral, if $w_i > 0, w_i < 0$ or $w_i = 0$, respectively.

Now, if all the intermediate counter values are already in $[x - \epsilon/2, y + \epsilon/2] \subseteq \mathcal{I}$, then we are done. Otherwise, (since $x_0 = x_0$ and $x_n = y$), let i be the first position and j be the last position such that $x_{i+1}, x_{j-1} \notin [x - \epsilon/2, y + \epsilon/2]$. Hence, all the counter values up till x_i and all the counter values from x_j onwards belong to $[x - \epsilon/2, y + \epsilon/2]$. We now give another run between $p_i(x_i)$ and $p_j(x_j)$ so that all the intermediate counter values remain in $[x - \epsilon, y + \epsilon]$. Since $[x - \epsilon, y + \epsilon] \subseteq \mathcal{I}$, plugging this new run between $p_i(x_i)$ and $p_j(x_j)$ into the old run will then give a $\mathcal{I}$-run from $p(x)$ to $q(y)$, thereby completing the proof of this case.

We accomplish this by further considering three different subcases: The first subcase is when $x_i = x_j$. Here, it suffices to simply fire each positive and negative transition in $\{t_{i+1}, \dots, t_j\}$ with a sufficiently small fraction so that each of their combined effects is equal and at most $\epsilon/2$, which will ensure that we always stay in the range $[x_i - \epsilon/2, x_j + \epsilon/2] \subseteq [x - \epsilon, y + \epsilon]$. The second subcase is when $x_i < x_j$. Hence, intuitively, negative transitions can be suppressed as much as needed. So, we fire each negative transition in $\{t_{i+1}, \dots, t_j\}$ with a sufficiently small fraction so that their combined effect (say E_-) is at most $\epsilon/2$. Then we scale each positive transition with a suitable fraction so that its combined effect equals *exactly* $x_j - x_i + E_-$. This will ensure that at any point in the run, the counter values stay in the range $[x_i - \epsilon/2, x_j + \epsilon/2] \subseteq [x - \epsilon, y + \epsilon]$. Furthermore, by construction, the combined

effect of the positive and negative transitions is exactly $x_j - x_i$ and so we reach counter value x_j at the end. Finally, the case of $x_i > x_j$ is dual to the previous case, with the roles of positive and negative transitions swapped.

Case 2: x is an extreme point, y is not. Therefore, $\mathcal{I}$ is of the form $[x, b)$ or $[x, b]$ with $y < b$. This means that in any $\mathcal{I}$-run from $p(x)$ and $q(y)$ in $\mathcal{M}$, the first non-neutral transition must be a positive transition, as otherwise the counter value would have to leave the interval $\mathcal{I}$. Hence, any $\mathcal{I}$-run from $p(x)$ to $q(y)$ in $\mathcal{M}$ must necessarily be of the form $p(x) \xrightarrow{*}_{\mathbb{Q}} d(x) \to_{\mathbb{Q}} e(z) \xrightarrow{*}_{\mathcal{I}} q(y)$ where $z > x$ and the run between $p(x)$ and $d(x)$ consists only of neutral transitions. Furthermore, since $z > x$, it follows that there is an $\epsilon > 0$ such that $[z - \epsilon, y + \epsilon] \subseteq \mathcal{I}$. Applying the previous case to the run $e(z) \xrightarrow{*}_{\mathcal{I}} q(y)$, we can conclude that this can happen iff $e(z) \xrightarrow{*}_{\mathbb{Q}} q(y)$. Hence, $p(x) \xrightarrow{*}_{\mathcal{I}} q(y)$ is possible iff there is a run of the form $p(x) \xrightarrow{*}_{\mathbb{Q}} d(x) \to_{\mathbb{Q}} e(z) \xrightarrow{*}_{\mathbb{Q}} q(y)$ where $z > x$ and the run between $p(x)$ and $d(x)$ consists only of neutral transitions. We can now easily construct another 1-CVASS $\mathcal{M}'$ that explicitly tracks $\mathbb{Q}$-runs of this form, by ensuring that the first non-neutral transition that is fired is a positive transition. This can be done by allowing for only neutral transitions until a positive transition is fired, after which any transition is allowed.

Case 3: x is not an extreme point, y is. This case is a dual to Case 2, where the last non-neutral transition fired must be a positive transition. Hence, we now have to track the last non-neutral transition that is fired, which can be done in a manner dual to Case 2.

Case 4: x, y are both extreme points. Therefore, $\mathcal{I} = [x, y]$. If $x = y$, we modify $\mathcal{M}$ so that it has only neutral transitions. Otherwise, we have to track both the first and last fired non-neutral transitions. So we combine the constructions from the above two cases. ◀

We now move on to the second stage.

Stage 2: Characterizing Reachability over $\mathbb{Q}$-semantics

In this stage, we give a precise characterization of reachability over the $\mathbb{Q}$-semantics. To state our characterization, we need to set up some notation.

Let $\mathcal{M} = (Q, T)$ be a 1-CVASS. First, notice that $\mathcal{M}$ naturally defines a finite labelled graph $\mathcal{G}_{\mathcal{M}}$ whose vertices are Q and whose edges are given by the transition relation T, i.e., any transition (p, w, q) corresponds to a labelled edge between p and q with label w. Given a path P in this graph $p_0 \xrightarrow{w_1} p_1 \xrightarrow{w_2} \dots p_{n-1} \xrightarrow{w_n} p_n$ we define the sets $T_+(P), T_0(P)$ and $T_-(P)$ as $\{i : w_i > 0\}, \{i : w_i = 0\}$ and $\{i : w_i < 0\}$, respectively.

Now, let $p(x)$ and $q(y)$ be two configurations of $\mathcal{M}$. By Remark 8, we can assume that $x \leq y$. We now show how to decide if $p(x)$ can reach $q(y)$, first in the case when $x = y$ and then in the case when $x < y$.

Case 1: $x = y$. In this case, the characterization for reachability is simple and is as follows.

▶ **Theorem 10.** *There is a $\mathbb{Q}$-run from $p(x)$ to $q(x)$ in $\mathcal{M}$ iff there is a path P in $\mathcal{G}_{\mathcal{M}}$ from p to q such that either $T_+(P), T_-(P) \neq \emptyset$ or $T_+(P) = T_-(P) = \emptyset$.*

With this theorem in hand, we now move on to the next case.

Case 2: $x < y$. To state the characterization for this case, we need some notation and a couple of preparatory intermediate results. For a path P in $\mathcal{G}_{\mathcal{M}}$ given by $p_0 \xrightarrow{w_1} p_1 \xrightarrow{w_2} \cdots p_{n-1} \xrightarrow{w_n} p_n$, we let $W(P) = \sum_{1 \leq i \leq n} w_i$. Having introduced this notation, we prove the following proposition, which gives a simple sufficient condition for reachability.

▶ **Proposition 11.** *Suppose P is a path from d to e in $\mathcal{G}_\mathcal{M}$ such that $W(P) \neq 0$, $W(P) \geq v-u$ for some u, v and $W(P)$ has the same sign as $v-u$, i.e., $W(P) < 0 \iff v-u < 0$. Then there is a $\mathbb{Q}$-run from $d(u)$ to $e(v)$ in $\mathcal{M}$.*

Proof. Let P be a path with $W(P) \geq v-u$ given by $d := p_0 \xrightarrow{w_1} p_1 \xrightarrow{w_2} \dots p_{n-1} \xrightarrow{w_n} p_n := e$. Let $t_i := (p_{i-1}, w_i, p_i)$ be the transition in $\mathcal{M}$ corresponding to the i^{th} edge of P. Then, firing each t_i with fraction $\frac{v-u}{W(P)}$ is easily seen to be a $\mathbb{Q}$-run from $d(u)$ to $e(v)$ in $\mathcal{M}$. ◀

The next preparatory result that we will need is regarding cycles in $\mathcal{G}_\mathcal{M}$. To this end, a cycle C of $\mathcal{G}_\mathcal{M}$ is said to be *pumpable* if at least one of the labels in C is strictly positive, i.e., $T_+(C) \neq \emptyset$. As the name suggests, we can use pumpable cycles to increase the counter value.

▶ **Proposition 12.** *Suppose C is a pumpable cycle from r to r, for some state r. Then, for any u, v with $u < v$, we have a $\mathbb{Q}$-run from $r(u)$ to $r(v)$.*

Proof Sketch. Let C be a pumpable cycle given by $r := r_0 \xrightarrow{w_1} r_1 \xrightarrow{w_2} \dots r_{n-1} \xrightarrow{w_n} r_n := r$. Let $t_i := (r_{i-1}, w_i, r_i)$ be the transitions in $\mathcal{M}$ corresponding to this path. We will fire each negative transition corresponding to $T_-(C)$ with a very tiny fraction so that their combined effect is of the form $-\epsilon/m$ for a small ϵ and large m. We then fire each positive transition corresponding to $T_+(C)$ with an appropriately chosen fraction so that their combined effect is exactly equal to $(v-u+\epsilon)/m$. This enables us to construct a run from $r(u)$ to $r(u+\Delta)$ where $\Delta = \frac{y-x}{m}$. Repeating this same run over and over, we can reach $r(u+2\Delta), r(u+3\Delta)$ and so on till $r(u+m\Delta) := r(v)$. ◀

Now, with the above two propositions in hand, we are almost ready to state our characterization of reachability over $\mathbb{Q}$-semantics. The final definition we need is the following: We say that a triple (I, C, F) is a *knot* of $\mathcal{G}_\mathcal{M}$ from i to f, if there is a state c such that I is a path from i to c, C is a pumpable cycle from c to c and F is a path from c to f. We are now ready to state our characterization of reachability.

▶ **Theorem 13.** *For any $x < y$, there is a $\mathbb{Q}$-run from $p(x)$ to $q(y)$ in $\mathcal{M}$ iff either there is a knot K from p to q in $\mathcal{G}_\mathcal{M}$ or there is a path P of length at most $|Q|$ from p to q in $\mathcal{G}_\mathcal{M}$ such that $W(P) \geq y-x$.*

For the left-to-right direction, given any $\mathbb{Q}$-run from $p(x)$ to $q(y)$, either it contains a cycle with a positive transition, and so we can extract a knot from it, or it only contains cycles with no positive transitions, each of which can be safely deleted to get a path P which must satisfy $W(P) \geq y-x$. For the other direction, we use Propositions 11 and 12.

Finally, by Remark 8, we do not have to consider separately the case when $x > y$. Hence, this allows us to complete the second stage and move on to the third stage.

Stage 3: Upper Bounds for Reachability in 1-CVASS

In this final stage, we use the characterization from the previous stage to give upper bounds for $\mathbb{Q}$-reachability in 1-CVASS. By the characterizations given in Theorem 10 and Theorem 13, the existence of a $\mathbb{Q}$-run between $p(x)$ and $q(y)$ in a 1-CVASS $\mathcal{M}$ with $x \leq y$ is governed by one of the following three conditions:

- Condition 1: $x = y$ and there is a path P in $\mathcal{G}_\mathcal{M}$ from p to q such that either $T_+(P), T_-(P) \neq \emptyset$ or $T_+(P) = T_-(P) = \emptyset$.
- Condition 2: $x < y$ and there is a knot K in $\mathcal{G}_\mathcal{M}$ from p to q.
- Condition 3: $x < y$ and there is a path P of length at most $|Q|$ in $\mathcal{G}_\mathcal{M}$ from p to q such that $W(P) \geq y-x$.

We prove that

▶ **Lemma 14.** *Conditions 1 and 2 can be decided in* NL*. Condition 3 can be decided in* NL *if $\mathcal{M}$ is a 1-CVASS with integer updates over unary encoding and in* AC^1 *otherwise.*

Indeed, the first two conditions can be easily seen to be decidable in NL. Indeed, for the first condition, if there is such a path, there has to be one of length at most $3|Q|$. Similarly, we can show that for the second condition, if there is such a knot, there has to be one of length at most $4|Q|$. These two characterizations immediately place the problem in NL, by allowing for a guess-and-check algorithm on the graph $\mathcal{G}_\mathcal{M}$. Similarly, the third condition can be decided in NL if $\mathcal{M}$ is a 1-CVASS with integer updates and unary encoding.

If $\mathcal{M}$ is a 1-CVASS with integer updates and binary encoding, we note that the problem of checking for a path P in $\mathcal{G}_\mathcal{M}$ of length at most $|Q|$ with $W(P) \geq y - x$ is equivalent to an instance of *coverability in 1-dimensional integer VASS*, i.e., 1-CVASS over the $\mathbb{Q}$-semantics where all the fractions have to be 1. This latter problem is known to be in AC^1 [24, Theorem III.2]. Finally, if $\mathcal{M}$ has rational updates, we can first multiply the labels of $\mathcal{G}_\mathcal{M}$ with the product B of all of its denominators, as well as the denominators of x, y. In this new graph (which only has integer labels) we have to find a path P with $W(P) \geq B(y - x)$. Hence, once again the problem is equivalent to an instance of coverability in 1-dimensional integer VASS. Since the product of a given collection of numbers can be computed in AC^1 [19, Corollary 4.1], this proves that the third condition can be tested in AC^1 when $\mathcal{M}$ has rational updates as well.

Finally, since $\mathsf{NL} \subseteq \mathsf{AC}^1$, by Remark 8 and by Theorem 9, this then proves Theorem 7, which completes the main result of this section.

▶ Remark 15. Our upper bound can also be extended to work for 1-CVASS *with zero-tests*. In this extension, a subset of states Z of the 1-CVASS is designated as the set of zero-test states. A run is then considered valid iff every configuration of the form $q(x)$ appearing in it with $q \in Z$ satisfies $x = 0$. In [8, Section 3.5], it is shown how to reduce the reachability problem for this extension to reachability over 1-CVASS. That reduction, along with our improved upper bound for 1-CVASS, allows us to conclude that reachability for 1-CVASS with zero-tests is also in AC^1.

4 Lower Bounds for 2-CVASS

In this section, we prove Theorems 2 and 4. To this end, we prove the following main result.

▶ **Theorem 16.** *2-*Cover$_\mathbb{Q}$ *and 2-*Cover$_{\mathbb{Q}_+}$ *are both* NP*-hard for acyclic 2-CVASS*
- *with integer updates encoded in binary;*
- *with rational updates encoded in unary.*

Let us first see how this result implies Theorems 2 and 4. First, by Remark 6, all of the NP-hardness results in the statements of these two theorems follow from Theorem 16. Finally, all of the NP membership results in the statements of these two theorems follow from the fact that reachability for d-CVASS (over both types of encoding and both semantics) is in NP for any d [7, Section IV.B]. So, the rest of this section is dedicated to proving Theorem 16. We begin by showing that the first part of Theorem 16 follows from the second part.

Let $\mathcal{M} = (Q, T)$ be an acyclic unary 2-CVASS and $p(\mathbf{u})$, $q(\mathbf{v})$ be some configurations of $\mathcal{M}$. Let B be the product of all the denominators of the numbers appearing in T as well as in $\mathbf{u}, \mathbf{v}$. Let $\mathcal{M}'$ be the acyclic *binary* 2-CVASS that we obtain by multiplying all of the transition vectors of $\mathcal{M}$ by B. It is immediately clear that $p(\mathbf{u})$ can cover $q(\mathbf{v})$ in $\mathcal{M}$ (over

either of the semantics) iff $p(B\mathbf{u})$ can cover $q(B\mathbf{v})$ in $\mathcal{M}'$ (over the corresponding semantics). Since it is easily seen that B can be computed in binary in polynomial time, this proves that the first part of Theorem 16 follows from the second part.

Hence it suffices to prove the second part of Theorem 16, which we do in two stages. In the first stage, we consider a restricted version of the reachability problem and show that this restricted reachability problem is NP-hard already for acyclic unary 1-CVASS (over both the semantics). In the second stage, we then reduce restricted reachability to coverability by the addition of an extra counter.

Stage 1: Hardness of Restricted Reachability of Acyclic Unary 1-CVASS

In this stage, we show NP-hardness of a restricted version of the reachability problem (over both the semantics). To define this problem formally, we set up some notation. A restricted $\mathbb{Q}$-run of a CVASS $\mathcal{M}$ is a $\mathbb{Q}$-run in which all of the fractions appearing in each step are exactly 1. More precisely, a restricted $\mathbb{Q}$-run is any run of the form $q_0(\mathbf{u}_0) \xrightarrow{\alpha_1 t_1}_{\mathbb{Q}} q_1(\mathbf{u}_1) \xrightarrow{\alpha_2 t_2}_{\mathbb{Q}} q_2(\mathbf{u}_2) \cdots \xrightarrow{\alpha_n t_n}_{\mathbb{Q}} q_n(\mathbf{u}_n)$ where each $\alpha_i = 1$. In the restricted $\mathbb{Q}$-reachability problem, we are given a CVASS $\mathcal{M}$ and two configurations $p(\mathbf{x})$ and $q(\mathbf{y})$ and we have to decide if $p(\mathbf{x})$ can reach $q(\mathbf{y})$ by a restricted $\mathbb{Q}$-run. Similarly, we can define restricted $\mathbb{Q}_+$-runs and restricted $\mathbb{Q}_+$-reachability. The main result of this stage is to show that

▶ **Theorem 17.** *Restricted $\mathbb{Q}$-reachability and restricted $\mathbb{Q}_+$-reachability is* NP*-hard for acyclic unary* 1*-CVASS.*

Theorem 17 is surprising because, by restricting the fractions to always be 1, the model behaves similar to a 1-dimensional unary (integer) VASS, for which reachability is in NL [15, 5]. However, unlike those machines, the difference here is that transition updates in a 1-CVASS are allowed to have rational values, which as we will see allows for much more power.

To prove the NP-hardness, we give a reduction from a constrained version of 3-SAT. More specifically, we consider as input 3-CNF formulas where each variable appears at most 4 times, and we want to decide if such a formula is satisfiable. This problem is NP-complete [25].

Hence, let φ be a given 3-CNF formula over n variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$ where each $C_i = \ell_i^1 \vee \ell_i^2 \vee \ell_i^3$ with $\ell_i^1, \ell_i^2, \ell_i^3 \in \{x_1, \ldots, x_n\} \cup \{\overline{x}_1, \ldots, \overline{x}_n\}$. Furthermore each literal $x_i, \overline{x}_i$ appears in at most 4 clauses. For each $1 \leq i \leq n$, let $\#(x_i), \#(\overline{x}_i)$ be the number of times the literals $x_i, \overline{x}_i$ appear in the formula φ, respectively. Additionally, to each literal $\ell \in \{x_1, \ldots, x_n\} \cup \{\overline{x}_1, \ldots, \overline{x}_n\}$ we assign a distinct prime number $P(\ell) \geq 5$. By the prime number theorem the n^{th} prime is of magnitude $O(n \log n)$. Hence all these primes can be written down in unary in polynomial time in the size of n, by running a sieve algorithm such as the Sieve of Eratosthenes.

We now encode an assignment to the variables $x_1, \ldots, x_n$ by means of a *sum of reciprocal of the prime numbers*. That is, we map an assignment A to the number $f(A) := \sum_{i=1}^{n} \frac{f_i}{r_i}$ where $(f_i, r_i) = (\#(x_i), P(x_i))$ if $A(x_i)$ is true and $(f_i, r_i) = (\#(\overline{x}_i), P(\overline{x}_i))$ if $A(x_i)$ is false. We call this the *Egyptian prime fraction* encoding, named after the Egyptian fractions technique of using sums of fractions of the form $1/n$ to encode non-negative rational numbers. As the following proposition shows, this way of using sums of reciprocals of primes to encode assignments is unique.

▶ **Proposition 18.** *Suppose $\sum_{i=1}^{k} \frac{f_i}{r_i} = \sum_{j=1}^{\ell} \frac{g_j}{s_j}$ where $\{r_1, \ldots, r_k\}$ (resp. $\{s_1, \ldots, s_\ell\}$) is a set of pairwise distinct prime numbers greater or equal to 5 and $f_i, g_j \leq 4$ for all $i \in [k], j \in [\ell]$. Then, $\{r_1, \ldots, r_k\} = \{s_1, \ldots, s_\ell\}$.*

Proof. Suppose $\sum_{i=1}^{k} \frac{f_i}{r_i} = \sum_{j=1}^{\ell} \frac{g_j}{s_j}$ and suppose $r_d \notin \{s_1, \ldots, s_\ell\}$ for some d. Hence, we get that $\frac{f_d}{r_d} = \sum_{j=1}^{\ell} \frac{g_j}{s_j} - \sum_{i=1, i\neq d}^{k} \frac{f_i}{r_i}$. Multiplying both sides by $P = \prod_{i=1}^{k} r_i \cdot \prod_{j=1}^{\ell} s_j$ we get $\frac{Pf_d}{r_d} = \sum_{j=1}^{\ell} \frac{Pg_j}{s_j} - \sum_{i=1, i\neq d}^{k} \frac{Pf_i}{r_i}$. Notice that each summand on the RHS is a natural number divisible by r_d, but the LHS is not divisible by r_d. Hence $\{r_1, \ldots, r_k\} \subseteq \{s_1, \ldots, s_\ell\}$. By a symmetric proof, we also have $\{s_1, \ldots, s_\ell\} \subseteq \{r_1, \ldots, r_k\}$ and so they are the same. ◀

This proposition immediately implies that $f(A) = f(A')$ iff $A = A'$ and therefore, our encoding of the assignments is unique. With this encoding in mind, we now construct an acyclic unary 1-CVASS $\mathcal{M}$ which consists of two gadgets, an *assignment gadget* $\mathcal{M}_A$ which encodes an assignment using the above encoding, and a *clause gadget* $\mathcal{M}_C$ which checks if this assignment satisfies all the clauses (see also Fig. 1).

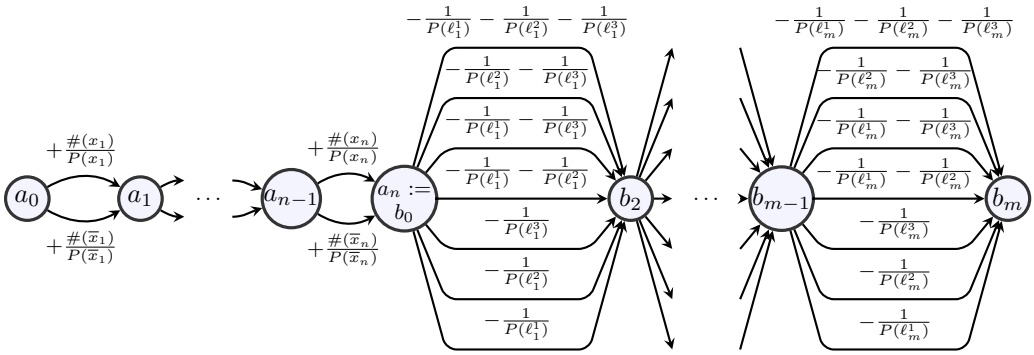

Figure 1 The CVASS $\mathcal{M}$. The first half (from a_0 to a_n) depicts the assignment gadget $\mathcal{M}_A$. The second half (from a_n to b_m) depicts the clause gadget.

The assignment gadget. $\mathcal{M}_A$ has states $a_0, \ldots, a_n$, and for each $0 \leq i < n$ there are two transitions $(a_i, \#(x_i)/P(x_i), a_{i+1})$ and $(a_i, \#(\overline{x}_i)/P(\overline{x}_i), a_{i+1})$. See the first half of Fig. 1 for a representation of $\mathcal{M}_A$. It is then immediately clear that there is a restricted $\mathbb{Q}$-run from $a_0(0)$ to $a_n(k)$ iff $k = f(A)$ for some assignment A.

The clause gadget. The clause gadget $\mathcal{M}_C$ checks whether the assignment A generated by the assignment gadget $\mathcal{M}_A$ satisfies the formula φ. It does this, by guessing for each clause C_i, the subset of literals $\ell_i^1, \ell_i^2, \ell_i^3$ that are satisfied by A and then decrementing the counter by the appropriate inverse of the primes. For example, if it guesses that the assignment satisfies only ℓ_i^2, ℓ_i^3, it will decrement the counter by $1/P(\ell_i^2) + 1/P(\ell_i^3)$.

More precisely $\mathcal{M}_C$ has the states $b_0, b_1, \ldots, b_m$ and for each $0 \leq i < m$ and for each of the 7 non-empty subsets $S_i \subseteq \{\ell_i^1, \ell_i^2, \ell_i^3\}$, $\mathcal{M}_C$ has a transition $(b_i, -\sum_{\ell \in S_i} 1/P(\ell), b_{i+1})$. See the second half of Fig. 1 for a representation of $\mathcal{M}_C$. We have the following proposition which asserts the main property of the clause gadget that we want.

▶ **Proposition 19.** *There is a restricted $\mathbb{Q}$-run from $b_0(f(A))$ to $b_m(0)$ iff A satisfies the formula φ.*

The $\Rightarrow$ direction holds from the uniqueness of encodings given by Proposition 18. For the $\Leftarrow$ direction it suffices to always pick the transition decrementing the counter by $\sum_{\ell \in S_i} \frac{1}{P(\ell)}$ for the non-empty set of literals of C_i that are true per A.

Let $\mathcal{M}$ be the acyclic 1-CVASS we obtain by combining $\mathcal{M}_A$ and $\mathcal{M}_C$ by fusing together the states a_n and b_0 (See Figure 1). Notice that any $\mathbb{Q}$-run in $\mathcal{M}$ from $a_0(0)$ to $b_m(0)$ must also be a $\mathbb{Q}_+$-run: Indeed, in any such run, the counter is only incremented in the stretch between a_0 to a_n and only decremented in the stretch between a_n and b_m. By the above properties of $\mathcal{M}_A$ and $\mathcal{M}_C$ we hence get that

▶ **Theorem 20.** *There is a restricted $\mathbb{Q}$-run (resp. restricted $\mathbb{Q}_+$-run) from $a_0(0)$ to $b_m(0)$ in $\mathcal{M}$ iff φ is satisfiable.*

Stage 2: Reducing Restricted Runs to Coverability

We now show how to get rid of the restricted run assumption in the 1-CVASS $\mathcal{M}$ from the previous subsection. First, by construction, notice that any run in $\mathcal{M}$ from $a_0(0)$ to $b_m(0)$ takes exactly $K := n + m$ steps. Furthermore, any single transition of $\mathcal{M}$ can update the counter by a number whose absolute value is at most 3. Indeed, the largest possible absolute value appearing in the transitions is either of the form $\frac{1}{p} + \frac{1}{q} + \frac{1}{r}$ or of the form $\frac{d}{p}$ for some $d \leq 4$. Hence, it follows that the maximum absolute value attainable in the counter in any run is at most $3K$.

We exploit this observation and modify $\mathcal{M}$ into an acyclic 2-CVASS $\mathcal{M}'$ by adding one extra counter to $\mathcal{M}$, and changing each transition (p, w, q) of $\mathcal{M}$ into $(p, (w, 1-w), q)$ in $\mathcal{M}'$. We can now prove that

▶ **Lemma 21.** *There is a restricted $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $a_0(0)$ to $b_m(0)$ in $\mathcal{M}$ iff there is a $\mathbb{Q}$-run (resp. $\mathbb{Q}_+$-run) from $a_0(0, 3K)$ covering $b_m(0, 4K)$ in $\mathcal{M}'$.*

Intuitively, the reason behind this lemma is that each transition of $\mathcal{M}'$ increases the sum of the two counters by 1. Hence, by requiring that the sum of the two counters at the very end has increased by at least K, we force the run in $\mathcal{M}'$ satisfying that requirement to fire each step with the fraction 1, thereby inducing a restricted run in $\mathcal{M}$.

Combined with Theorem 20, this then allows us to conclude that 2-Cover$_{\mathbb{Q}}$ and 2-Cover$_{\mathbb{Q}_+}$ are both NP-hard over unary encoding. This proves the second part of Theorem 16, which completes the main result of this section.

5 Lower Bounds for 3-CVASS (with Integer Updates)

In this section, we prove Theorem 5. To this end, we prove the following main result.

▶ **Theorem 22.** *3-Cover$_{\mathbb{Q}_+}$ for acyclic unary 3-CVASS with integer updates is NP-hard.*

By Remark 6, it is easily seen that this implies all of the NP lower bounds mentioned in Theorem 5. As before, all of the NP upper bounds in Theorem 5 follow from [7, Section IV.B]. So the rest of this section is dedicated to proving Theorem 22, which we do in two stages: First, we show that coverability for acyclic unary 2-CVASS *with zero-tests* over the $\mathbb{Q}_+$-semantics is NP-hard. In the next stage, we show how to reduce this problem to coverability for acyclic unary 3-CVASS.

Throughout this section, we shall exclusively consider the $\mathbb{Q}_+$-semantics.

Stage 1: Hardness for Acyclic Unary 2-CVASS with Zero-Tests

In this stage, we show NP-hardness of coverability for an extended model of CVASS, namely CVASS with zero-tests. A d-CVASS with zero-tests is a triple $\mathcal{M} = (Q, T, Z)$ where (Q, T) is a d-CVASS and $Z : Q \to 2^{[d]}$ is a mapping which mandates that in a configuration $p(\mathbf{x})$,

every counter in $Z(p)$ must be zero. Formally, a configuration of $\mathcal{M}$ is a pair $p(\mathbf{x})$ where $p \in Q, \mathbf{x} \in \mathbb{Q}_+^d$ and $\mathbf{x}[i] = 0$ for every $i \in Z(p)$. Given two configurations $p(\mathbf{x})$ and $q(\mathbf{y})$, a transition $(p, \mathbf{w}, q)$ and a non-zero fraction $\alpha \in (0, 1]$, we say that $p(\mathbf{x}) \xrightarrow{\alpha \mathbf{w}} q(\mathbf{y})$ if $\mathbf{x}[i] = 0$ for all $i \in Z(p)$, $\mathbf{y} = \mathbf{x} + \alpha\mathbf{w}$ and $\mathbf{y}[i] = 0$ for all $i \in Z(q)$. Similarly to CVASS, we define the notions of steps, $\mathbb{Q}_+$-runs, and $\mathbb{Q}_+$-reachability for CVASS with zero-tests. Our main result in this stage is to show that

▶ **Theorem 23.** *Reachability for acyclic unary 2-CVASS with integer updates and zero-tests over the $\mathbb{Q}_+$-semantics is* NP*-hard.*

Similar to the previous section, we will prove this theorem by giving a reduction from a constrained version of 3-SAT, in which each variable appears at most 4 times in a formula. To this end, let φ be a 3-CNF formula over n variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$ where each $C_i = \ell_i^1 \vee \ell_i^2 \vee \ell_i^3$ with $\ell_i^1, \ell_i^2, \ell_i^3 \in \{x_1, \ldots, x_n\} \cup \{\overline{x}_1, \ldots, \overline{x}_n\}$. Furthermore, each literal $x_i, \overline{x}_i$ appears at most 4 times in φ. For each $1 \leq i \leq n$, let $\#(x_i), \#(\overline{x}_i)$ be the number of times the literals $x_i, \overline{x}_i$ appear in the formula φ, respectively. Additionally, to each literal $\ell \in \{x_1, \ldots, x_n\} \cup \{\overline{x}_1, \ldots, \overline{x}_n\}$ we assign a distinct prime number $P(\ell) \geq 5$, such that $P(\ell) = P(\ell')$. We can now uniquely encode an assignment A to the variables as the fraction $f(A) := \frac{1}{r_1^{f_1} \cdot r_2^{f_2} \cdot \ldots \cdot r_n^{f_n}}$ where $(r_i, f_i) = (P(x_i), \#(x_i))$ if $A(x_i)$ is true and $(r_i, f_i) = (P(\overline{x}_i), \#(\overline{x}_i))$ if $A(x_i)$ is false.

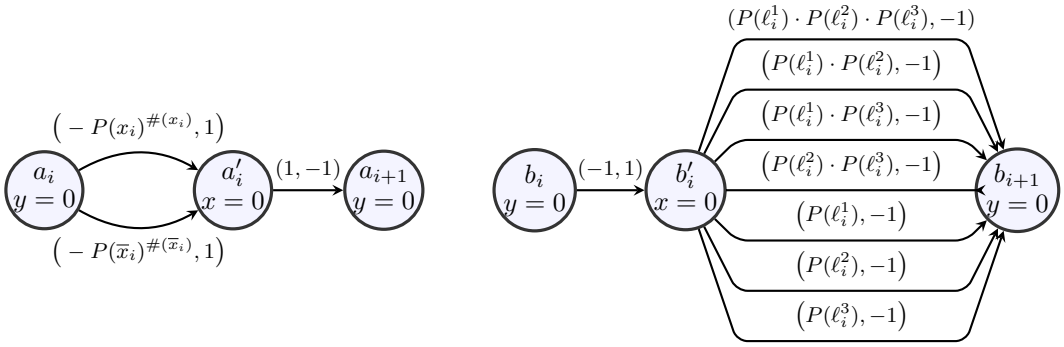

Figure 2 A section of the assignment gadget (left) and the clause gadget (right). Here we use $x = 0$ (resp. $y = 0$) inside a state to denote that that state zero-tests the first (resp. second) counter.

We now construct an acyclic unary 2-CVASS with integer updates and zero-tests $\mathcal{M}$ which consists of two gadgets, the *assignment gadget* $\mathcal{M}_A$ which encodes an assignment, and a *clause gadget* $\mathcal{M}_C$ which checks if this assignment satisfies all the clauses

The assignment gadget. $\mathcal{M}_A$ has states $a_0, \ldots, a_n, a'_0, \ldots, a'_{n-1}$ with corresponding zero-tests given by $Z(a_0) = \ldots = Z(a_n) = \{2\}$ and $Z(a'_0) = \ldots = Z(a'_{n-1}) = \{1\}$. Furthermore, for each $0 \leq i < n$, there are three transitions $(a_i, (-P(x_i)^{\#(x_i)}, 1), a'_i)$, $(a_i, (-P(\overline{x}_i)^{\#(\overline{x}_i)}, 1), a'_i)$ and $(a'_i, (1, -1), a_{i+1})$. (See Fig. 2).

Intuitively, the first two transitions divide the current value of the first counter by either $P(x_i)^{\#(x_i)}$ or $P(\overline{x}_i)^{\#(\overline{x}_i)}$ while transferring this divided value to the second counter. The third transition then transfers this divided value back to the first counter. This effect of these transitions is formalized by the following proposition, whose proof follows by a case analysis of the possible fractions that can be used, under the influence of the zero-tests on the states.

▶ **Proposition 24.** *For any $0 \leq i < n$, there is a $\mathbb{Q}_+$-run from $a_i(g_i, h_i)$ to $a_{i+1}(g_{i+1}, h_{i+1})$ iff $h_i = h_{i+1} = 0$ and either $g_{i+1} = g_i / P(x_i)^{\#(x_i)}$ or $g_{i+1} = g_i / P(\overline{x}_i)^{\#(\overline{x}_i)}$.*

Repeated application of this proposition implies that there is a $\mathbb{Q}_+$-run from $a_0(1,0)$ to $a_n(g,h)$ for any g,h iff $g = f(A)$ for some assignment A and $h = 0$. Further, since each $P(\ell) \in O(n \log n), \#(\ell) \leq 4$, it follows that all of the numbers in the transitions can be encoded in unary in polynomial time.

The clause gadget. The clause gadget $\mathcal{M}_C$ checks whether the assignment A generated by the assignment gadget $\mathcal{M}_A$ satisfies the formula φ. It does this, by guessing for each clause C_i, the subset of literals $\ell_i^1, \ell_i^2, \ell_i^3$ that are satisfied by A and then executing a gadget that *multiplies* the counter by the corresponding primes. More precisely $\mathcal{M}_C$ has the states $b_0, \ldots, b_m, b'_0, \ldots, b'_{m-1}$ with corresponding zero-tests given by $Z(b_0) = \cdots = Z(b_m) = \{2\}$ and $Z(b'_0) = \cdots = Z(b'_{m-1}) = \{1\}$. Furthermore, for each $0 \leq i < m$, there is one transition $(b_i, (-1,1), b'_i)$ and seven other transitions given by $(b'_i, (\prod_{\ell \in S_i} P(\ell), -1), b_{i+1})$ for every non-empty subset $S_i \subseteq \{\ell_i^1, \ell_i^2, \ell_i^3\}$ (See Fig. 2). We now have the following proposition, whose proof is similar to Proposition 24.

▶ **Proposition 25.** *For any $0 \leq i < m$, there is a $\mathbb{Q}_+$-run from $b_i(g_i, h_i)$ to $b_{i+1}(g_{i+1}, h_{i+1})$ iff $h_i = h_{i+1} = 0, g_i \leq 1$ and $g_{i+1} = g_i \cdot p$ where p is the product of some non-empty subset of $\{P(\ell_i^1), P(\ell_i^2), P(\ell_i^3)\}$.*

Hence, repeated applications of this proposition means that any run from b_0 to b_m essentially multiplies the first counter by a subset of the primes $\{P(\ell_i^1), P(\ell_i^2), P(\ell_i^3)\}$ for each i. This then implies that,

▶ **Lemma 26.** *There is a $\mathbb{Q}_+$-run from $b_0(f(A), 0)$ to $b_m(1,0)$ iff A satisfies φ.*

However, since our goal is to show hardness for coverability, we have to modify the condition in the above lemma so that it becomes a coverability condition. To that end, we add a state b'_m with zero test $Z(b'_m) = \{1\}$ and a transition $(b_m, (-1,1), b'_m)$ to $\mathcal{M}_C$. It immediately follows that

▶ **Lemma 27.** *There is a $\mathbb{Q}_+$-run from $b_0(f(A), 0)$ covering $b'_m(0,1)$ iff A satisfies φ.*

It is now clear that if we let $\mathcal{M}$ be the acyclic 2-CVASS with zero-tests obtained by combining $\mathcal{M}_A$ and $\mathcal{M}_C$ by fusing together a_n and b_0, we get

▶ **Theorem 28.** *There is a $\mathbb{Q}_+$-run from $a_0(1,0)$ covering $b'_m(0,1)$ iff φ is satisfiable.*

This completes Stage 1 of the reduction. We now move on to Stage 2.

Stage 2: Removing Zero-Tests by One Extra Counter

We now show how to get rid of the zero-tests in $\mathcal{M}$ by adding one extra counter. To this end, we adapt the idea of the *controlling counter* technique from VASS [10]. The idea is as follows: Consider the CVASS $\overline{\mathcal{M}}$ that we get from $\mathcal{M}$ by simply removing its zero-tests. As with $\mathcal{M}$, any $\mathbb{Q}_+$-run in $\overline{\mathcal{M}}$ from $a_0(1,0)$ covering $b'_m(0,1)$ must be of the form

$$a_0(1,0) := a_0(g_0,h_0) \xrightarrow{\alpha_0 v_0}_{\mathbb{Q}_+} a'_0(g'_0,h'_0) \xrightarrow{\alpha'_0 v'_0}_{\mathbb{Q}_+} a_1(g_1,h_1) \cdots \xrightarrow{\alpha'_{n-1} v'_{n-1}}_{\mathbb{Q}_+} a_n(g_n,h_n) :=$$
$$b_0(k_0,r_0) \xrightarrow{\beta_0 w_0}_{\mathbb{Q}_+} b'_0(k'_0,r'_0) \xrightarrow{\beta'_0 w'_0}_{\mathbb{Q}_+} b_1(k_1,r_1) \cdots b_m(k_m,r_m) \xrightarrow{\beta'_m w'_m}_{\mathbb{Q}_+} b'_m(k'_m,r'_m) \quad (1)$$

with $k'_m \geq 0, r'_m \geq 1$. This $\mathbb{Q}_+$-run in $\overline{\mathcal{M}}$ is also a $\mathbb{Q}_+$-run in $\mathcal{M}$ iff each $h_i, r_i = 0$ and each $g'_i, k'_i = 0$. Let us call such $\mathbb{Q}_+$-runs as *good $\mathbb{Q}_+$-runs* of $\overline{\mathcal{M}}$. Hence a good $\mathbb{Q}_+$-run of $\overline{\mathcal{M}}$ exists iff there is a $\mathbb{Q}_+$-run of $\mathcal{M}$ from $a_0(1,0)$ covering $b'_m(0,1)$. Since we consider the $\mathbb{Q}_+$-semantics it holds that $h_i, r_i, g'_i, k'_i \geq 0$ for all i. Hence all of them are 0 iff their combined sum is 0, i.e., a $\mathbb{Q}_+$-run of the form Eq. 1 is a good run iff

$$\sum_{i=0}^{n-1} g'_i + \sum_{i=1}^{n} h_i + \sum_{i=0}^{m} k'_i + \sum_{i=1}^{m} r_i = 0 \tag{2}$$

Now, we claim that the above equation can be described differently by expressing the LHS of Equation 2 in a different way. For instance, we know that each g'_i is essentially the value of the first counter in the sum $(1,0) + \sum_{j=0}^{i} \alpha_i v_i + \sum_{j=0}^{i-1} \alpha'_i v'_i$. Similarly, h_i, k'_i, r_i can also be written as such sums. Combining all of these sums into one, we can rewrite the above identity as the one given by the following proposition.

▶ **Proposition 29.** *A $\mathbb{Q}_+$-run of the form Eq. 1 is a good $\mathbb{Q}_+$-run iff*

$$n+(m+1)+\left(\sum_{i=0}^{n-1} \alpha_i\big((n-i+m)(v_i[1]+v_i[2])+v_i[1]\big)\right)+\left(\sum_{i=0}^{n-1} \alpha'_i(n-i+m)(v'_i[1]+v'_i[2])\right)+ \left(\sum_{i=0}^{m} \beta_i\big((m-i)(w_i[1]+w_i[2])+w_i[1]\big)\right)+\left(\sum_{i=0}^{m-1} \beta'_i(m-i)(w'_i[1]+w'_i[2])\right)=0 \tag{3}$$

We call the number on the LHS of Eq. 3 as the *tester* of the run. (We stress that this tester is the same as the value $\sum_{i=0}^{n-1} g'_i + \sum_{i=1}^{n} h_i + \sum_{i=0}^{m} k'_i + \sum_{i=1}^{m} r_i$ and is hence always non-negative). By Proposition 29 it follows that

▶ **Lemma 30.** *There is a $\mathbb{Q}_+$-run from $a_0(1,0)$ covering $b'_m(1,0)$ in $\mathcal{M}$ iff there is a $\mathbb{Q}_+$-run from $a_0(1,0)$ covering $b'_m(1,0)$ in $\overline{\mathcal{M}}$ whose tester equals 0.*

In order to remove the tester condition, we now introduce another counter to keep track of the current value of the tester encountered so far. More precisely, for any prefix P of a run of the form Eq. 1, we assign a number $C(P)$ as follows: For the empty prefix, $C(P)$ is simply $n+m+1$. Now, for any prefix $P = P's$, $C(P) = C(P') + C(s)$ where

- $C(s) = \alpha_i\big((n-i+m)(v_i[1]+v_i[2])+v_i[1]\big)$ if $s \equiv a_i(g_i,h_i) \xrightarrow{\alpha_i v_i} a'_i(g'_i,h'_i)$.
- $C(s) = \alpha'_i(n-i+m)(v'_i[1]+v'_i[2])$ if $s \equiv a'_i(g'_i,h'_i) \xrightarrow{\alpha'_i v'_i} a_{i+1}(g_{i+1},h_{i+1})$.
- $C(s) = \beta_i\big((m-i)(w_i[1]+w_i[2])+w_i[1]\big)$ if $s \equiv b_i(k_i,r_i) \xrightarrow{\beta_i w_i} b'_i(k'_i,r'_i)$.
- $C(s) = \beta'_i(m-i)(w'_i[1]+w'_i[2])$ if $s \equiv b'_i(k'_i,r'_i) \xrightarrow{\beta'_i w'_i} b_{i+1}(k_{i+1},r_{i+1})$.

By construction, it is easily seen that $C(R)$ for a $\mathbb{Q}_+$-run R is precisely its tester. Hence, in order to compute $C(R)$, we add a third counter to $\overline{\mathcal{M}}$ with initial value $n+m+1$, and at each step s, we simply add $C(s)$ to the third counter. We then finally require this third counter to have value 0. Whilst correct, this only gives a reachability condition on the third counter. Furthermore, while adding $C(s)$ at each step, the third counter might go below 0.

We now show how to avoid both these problems in one shot. To this end, let K be the maximum *absolute value* of all of the numbers in the transitions of $\overline{\mathcal{M}}$ and let L be $n+(m+1)+(n\cdot(n+m)\cdot 3K)+(n\cdot(n+m)\cdot 2K)+((m+1)\cdot m\cdot 3K)+(m\cdot m\cdot 2K)$. Note that L can be written down in unary in polynomial time. By construction of L, it immediately follows that $C(P)$ for any prefix has to lie in the range $[-L, L]$. Hence, if we now begin the third counter with the initial value $L-(n+m+1)$ and *subtract* $C(s)$ at each step s, we are guaranteed that it will never go below zero. Furthermore, by construction, at the end of a run R, the value of the third counter will be $L - C(R)$. As mentioned before, $C(R)$ is always non-negative and so the only way the third counter can reach a value $\geq L$ at the end is if $C(R) = 0$. This gives a coverability condition on the third counter that is equivalent to checking if the tester of the run is zero. If we call this new machine (with three counters) as $\mathcal{M}'$ it then follows that

▶ **Lemma 31.** *There is a $\mathbb{Q}_+$-run from $a_0(1,0)$ covering $b'_m(0,1)$ in $\overline{\mathcal{M}}$ whose tester equals 0 iff there is a $\mathbb{Q}_+$-run from $a_0\big(1,0,L-(n+m+1)\big)$ covering $b'_m(0,1,L)$ in $\mathcal{M}'$.*

By combining Theorem 28 and Lemmas 30 and 31 we get the following theorem.

▶ **Theorem 32.** *There is a $\mathbb{Q}_+$-run from $a_0\big(1,0,L-(n+m+1)\big)$ covering $b'_m(1,0,L)$ in $\mathcal{M}'$ iff φ is satisfiable.*

This completes the proof of Theorem 22, which is the main result of this section.

6 Conclusion

In this paper, we studied the complexity of reachability and coverability for fixed-dimensional CVASS over both unary/binary encoding as well as over both $\mathbb{Q}$-/$\mathbb{Q}_+$- semantics. As our main result, we have completely characterized the complexity of all of these problems for all dimensions. Furthermore, we have also given an almost-complete classification in the case when the update vectors are restricted to be over the integers. One possible direction for future work is to study these problems for unary 2-CVASS with integer updates over $\mathbb{Q}_+$ semantics as well as for unary d-CVASS with integer updates over $\mathbb{Q}$-semantics for any $d \geq 2$.

References

1. A. R. Balasubramanian. Parameterized complexity of safety of threshold automata. In Nitin Saxena and Sunil Simon, editors, *40th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2020, BITS Pilani, K K Birla Goa Campus, Goa, India (Virtual Conference), December 14-18, 2020*, LIPIcs, pages 37:1–37:15. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.FSTTCS.2020.37.
2. A. R. Balasubramanian, Javier Esparza, and Mikhail A. Raskin. Finding cut-offs in leaderless rendez-vous protocols is easy. *Log. Methods Comput. Sci.*, 19(4), 2023. doi:10.46298/LMCS-19(4:2)2023.
3. Pascal Baumann, Flavio D'Alessandro, Moses Ganardi, Oscar H. Ibarra, Ian McQuillan, Lia Schütze, and Georg Zetzsche. Unboundedness problems for machines with reversal-bounded counters. In Orna Kupferman and Pawel Sobocinski, editors, *Foundations of Software Science and Computation Structures - 26th International Conference, FoSSaCS 2023, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2023, Paris, France, April 22-27, 2023, Proceedings*, Lecture Notes in Computer Science, pages 240–264. Springer, 2023. doi:10.1007/978-3-031-30829-1_12.
4. Michael Blondin. The ABCs of Petri net reachability relaxations. *ACM SIGLOG News*, 7(3):29–43, 2020. doi:10.1145/3436980.3436984.
5. Michael Blondin, Matthias Englert, Alain Finkel, Stefan Göller, Christoph Haase, Ranko Lazic, Pierre McKenzie, and Patrick Totzke. The reachability problem for two-dimensional vector addition systems with states. *J. ACM*, 68(5):34:1–34:43, 2021. doi:10.1145/3464794.
6. Michael Blondin, Alain Finkel, Christoph Haase, and Serge Haddad. The logical view on continuous Petri nets. *ACM Trans. Comput. Log.*, 18(3):24:1–24:28, 2017. doi:10.1145/3105908.
7. Michael Blondin and Christoph Haase. Logics for continuous reachability in Petri nets and vector addition systems with states. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005068.

8 Michael Blondin, Tim Leys, Filip Mazowiecki, Philip Offtermatt, and Guillermo A. Pérez. Continuous one-counter automata. *ACM Trans. Comput. Log.*, 24(1):3:1–3:31, 2023. doi:10.1145/3558549.

9 Wojciech Czerwinski, Ismaël Jecker, Slawomir Lasota, and Lukasz Orlikowski. Reachability in 3-vass is elementary. In Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis, editors, *52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, Aarhus, Denmark, July 8-11, 2025*, LIPIcs, pages 153:1–153:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2025. doi:10.4230/LIPIcs.ICALP.2025.153.

10 Wojciech Czerwinski and Lukasz Orlikowski. 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*, pages 1229–1240. IEEE, 2021. doi:10.1109/FOCS52979.2021.00120.

11 René David and Hassane Alla. *Discrete, continuous, and hybrid Petri nets*. Springer, 2005. doi:10.1007/978-3-642-10669-9.

12 Javier Esparza, Pierre Ganty, Jérôme Leroux, and Rupak Majumdar. Verification of population protocols. *Acta Informatica*, 54(2):191–215, 2017. doi:10.1007/S00236-016-0272-3.

13 Javier Esparza, Ruslán Ledesma-Garza, Rupak Majumdar, Philipp J. Meyer, and Filip Niksic. An SMT-based approach to coverability analysis. In Armin Biere and Roderick Bloem, editors, *Computer Aided Verification - 26th International Conference, CAV 2014, Held as Part of the Vienna Summer of Logic, VSL 2014, Vienna, Austria, July 18-22, 2014. Proceedings*, Lecture Notes in Computer Science, pages 603–619. Springer, 2014. doi:10.1007/978-3-319-08867-9_40.

14 Steven M. German and A. Prasad Sistla. Reasoning about systems with many processes. *J. ACM*, 39(3):675–735, 1992. doi:10.1145/146637.146681.

15 Eitan M. Gurari and Oscar H. Ibarra. The complexity of the equivalence problem for simple programs. *J. ACM*, 28(3):535–560, 1981. doi:10.1145/322261.322270.

16 Stefan Haar and Juraj Kolcák. Continuous petri nets faithfully fluidify most permissive boolean networks. In François Fages and Sabine Pérès, editors, *Computational Methods in Systems Biology - 23rd International Conference, CMSB 2025, Lyon, France, September 10-12, 2025, Proceedings*, Lecture Notes in Computer Science, pages 89–105. Springer, 2025. doi:10.1007/978-3-032-01436-8_6.

17 Monika Heiner, David R. Gilbert, and Robin Donaldson. Petri nets for systems and synthetic biology. In Marco Bernardo, Pierpaolo Degano, and Gianluigi Zavattaro, editors, *Formal Methods for Computational Systems Biology, 8th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2008, Bertinoro, Italy, June 2-7, 2008, Advanced Lectures*, Lecture Notes in Computer Science, pages 215–264. Springer, 2008. doi:10.1007/978-3-540-68894-5_7.

18 Mostafa Herajy and Monika Heiner. Adaptive and bio-semantics of continuous petri nets: Choosing the appropriate interpretation. *Fundam. Informaticae*, 160(1-2):53–80, 2018. doi:10.3233/FI-2018-1674.

19 William Hesse, Eric Allender, and David A. Mix Barrington. Uniform constant-depth threshold circuits for division and iterated multiplication. *J. Comput. Syst. Sci.*, 65(4):695–716, 2002. doi:10.1016/S0022-0000(02)00025-9.

20 Jérôme Leroux. 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*, pages 1241–1252. IEEE, 2021. doi:10.1109/FOCS52979.2021.00121.

21 C. Mahulea, L. Recalde, and M. Silva. On performance monotonicity and basic servers semantics of continuous Petri nets. In *2006 8th International Workshop on Discrete Event Systems*, pages 345–351, 2006. doi:10.1109/WODES.2006.382530.

22 Charles Rackoff. The covering and boundedness problems for vector addition systems. *Theor. Comput. Sci.*, 6:223–231, 1978. doi:10.1016/0304-3975(78)90036-1.

23 Sylvain Schmitz. The complexity of reachability in vector addition systems. *ACM SIGLOG News*, 3(1):4–21, 2016. doi:10.1145/2893582.2893585.

24 Yousef Shakiba, Henry Sinclair-Banks, and Georg Zetzsche. A complexity dichotomy for semilinear target sets in automata with one counter. In *40th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2025, Singapore, June 23-26, 2025*, pages 594–608. IEEE, 2025. doi:10.1109/LICS65433.2025.00051.

25 Craig A. Tovey. A simplified np-complete satisfiability problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984. doi:10.1016/0166-218X(84)90081-7.

26 Wil M. P. van der Aalst. The application of Petri nets to workflow management. *J. Circuits Syst. Comput.*, 8(1):21–66, 1998. doi:10.1142/S0218126698000043.

27 Heribert Vollmer. *Introduction to Circuit Complexity - A Uniform Approach*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 1999. doi:10.1007/978-3-662-03927-4.