Decidability and Complexity of Decision Problems for Affine Continuous VASS

A. R. Balasubramanian

bayikudi@mpi-sws.org

Max Planck Institute for Software Systems

Kaiserslautern, Germany

Abstract

Vector addition system with states (VASS) is a popular model for the verification of concurrent systems. VASS consists of finitely many control states and a set of counters which can be incremented and decremented, but not tested for zero. VASS is a relatively well-studied model of computation and many results regarding the decidability of decision problems for VASS are well-known. Given that the complexity of solving almost all problems for VASS is very high, various tractable over-approximations of the reachability relation of VASS have been proposed in the literature. One such tractable over-approximation is the so-called *continuous VASS*, in which counters are allowed to have non-negative rational values and whenever an update is performed, the update is first scaled by an arbitrary non-zero fraction.

In this paper, we consider *affine* continuous VASS, which extend continuous VASS by allowing integer affine operations. Affine continuous VASS serve as an over-approximation to the model of affine VASS, in the same way that continuous VASS over-approximates the reachability relation of VASS. We investigate the tractability of affine continuous VASS with respect to the reachability, coverability and state-reachability problems for different classes of affine operations and we prove an almost-complete classification of the decidability of these problems. Namely, except for the coverability problem for a single family of classes of affine operations, we completely determine the decidability status of these problems for all classes. Furthermore, except for this single family, we also complement all of our decidability results with tight complexity-theoretic upper and lower bounds.

CCS Concepts

• **Theory of computation → Models of computation**.

Keywords

Vector addition systems, Reachability, Coverability, Decidability, Complexity

1 Introduction

Vector addition system with states (VASS) are one of the most popular and most studied infinite-state systems and have many applications in verifying and modeling concurrent and business processes [13, 14, 16, 22]. VASS consist of a finite set of states along with a set of counters with values ranging over $\mathbb{N}$. Hence, a configuration of a VASS is a pair consisting of the current state of the VASS along with the current values of its counters. Transitions of the VASS allow it to move from one state to another while incrementing or decrementing the values of the counters, provided that the new values of the counters do not drop below zero. Alternatively,

this can be summarized as having update instructions of the form $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{b}$ which are fireable as long as $\mathbf{x} + \mathbf{b} \geq \mathbf{0}$. One of the most central questions regarding VASS is the *reachability* problem: Given two configurations C and C' of a VASS, can C reach C'? After decades of being open, the complexity of the problem was recently shown to be ACKERMANN-complete [12, 19].

As mentioned before, the model of VASS has many applications. However, since basic verification questions such as reachability have extremely high complexity, much of the research has been concentrated on finding *over-approximations* of the reachability relation which trade expressiveness for tractability. Note that since reachability is usually used to check the existence of a path to an unsafe configuration, if we prove that the unsafe state is not reachable in an over-approximate model, then the same holds for the original model as well. Many heuristics and over-approximations for the reachability relation of a VASS are known [5] and they have been applied in practice quite successfully [5, 6]. Here we concentrate on one such over-approximation, namely the *continuous semantics*.

In the continuous semantics, transitions are allowed to be fired *fractionally*. More precisely, in order to fire a transition t, we are allowed to first pick a non-zero fraction $\alpha \in (0, 1]$ and then execute the updates of t by the fraction α, i.e., if t originally decremented (resp. incremented) a counter by c, we now decrement (resp. increment) that counter by $\alpha \cdot c$, as long as the new value does not drop below zero. As a result of this "fractional update", counters are now relaxed to have values over the non-negative rationals $\mathbb{Q}_+$. Note that the continuous semantics is an over-approximation of the usual semantics. It turns out that the continuous semantics is quite well-behaved in terms of complexity: Reachability in continuous VASS is NP-complete and is hence much lower than the complexity for the usual semantics. Besides being a tractable over-approximation, it is also useful in practice [6, 9, 10] and is also useful to derive sound and complete algorithms for various problems in parameterized verification [2, 3].

Coming back to VASS, as mentioned in [11], although they are useful for modeling many systems, the features of VASS might be too restricted in certain cases. For instance, VASS cannot model counter systems which are allowed to reset their counters to 0, or transfer or copy the contents of one counter to another. It turns out that such extensions are quite desirable to model certain applications [13, 18, 20]. Many of these extensions can be modeled as specific *classes of affine VASS*. Affine VASS are VASS extended with affine operations, i.e., transitions can have updates of the form $\mathbf{x} \leftarrow \mathbf{A}\mathbf{x} + \mathbf{b}$ for some matrix $\mathbf{A}$ and some vector $\mathbf{b}$ over the integers and such transitions are fireable as long as $\mathbf{A}\mathbf{x}+\mathbf{b} \geq 0$. However, this huge expressiveness comes at a cost; almost all interesting verification problems are either undecidable or have very high complexity for affine VASS [11].

As a result of this undecidability/high complexity barrier for affine VASS, it would then be natural to wonder if the tractability of over-approximations developed for VASS can also be transferred to affine VASS. In particular, a natural question might be to ask

> Does the tractability of the continuous semantics hold in the affine case as well?

Our contribution. In this paper, we consider the above question and introduce the model of *affine continuous VASS*. An affine continuous VASS is just like a VASS, except that transitions are now allowed to be fired fractionally, i.e., before firing a transition t (whose instruction is given by $\mathbf{x} \leftarrow \mathbf{Ax} + \mathbf{b}$ where $\mathbf{A}$ and $\mathbf{b}$ are a matrix and a vector over the integers respectively), we pick a non-zero fraction α and update the counters by the rule $\mathbf{x} \leftarrow \mathbf{Ax} + \alpha\mathbf{b}$ provided that $\mathbf{Ax} + \alpha\mathbf{b} \geq \mathbf{0}$. To put it in words, to get the new counter values, we multiply the old counter values with the matrix $\mathbf{A}$ and then add the vector $\alpha\mathbf{b}$, provided that the final result we get does not make the value of any counter drop below zero.

At this point, the reader might wonder why the semantics for the continuous case was not defined so that the update looks like $\mathbf{x} \leftarrow \alpha\mathbf{Ax} + \alpha\mathbf{b}$. First, defining the semantics this way does not allow us to model various important features such as resets or transfers. Second, we would ideally like the semantics that we define for affine continuous VASS to extend the semantics for continuous VASS, i.e., any meaningful extension of continuous VASS to the affine setting should allow for a continuous VASS to be interpreted as is in the extended semantics by thinking of each transition as being equipped with an identity matrix. If we define the semantics for the affine case by the rule $\mathbf{x} \leftarrow \alpha\mathbf{Ax} + \alpha\mathbf{b}$, then this feature is lost. For these two reasons, we choose the semantics of an affine continuous VASS so that the **matrix is not multiplied with the chosen fraction α.**

Having defined this model, we study three of the most fundamental verification problems: Reachability (can a given configuration reach another given configuration?), Coverability (given two configurations $(p, \mathbf{u})$ and $(q, \mathbf{v})$ can $(p, \mathbf{u})$ reach some $(q, \mathbf{v}')$ with $\mathbf{v}' \geq \mathbf{v}$?) and State-reachability (given a configuration $(p, \mathbf{u})$ and a state q, can $(p, \mathbf{u})$ reach some configuration whose state is q?). We present an almost-complete classification of the decidability status of these problems with respect to the various *classes of affine operations* that are allowed, i.e., with respect to the various classes of matrices that are allowed. We now briefly summarize our main contributions.

Reachability. As our first contribution, we completely resolve the decidability status of reachability. We show that reachability for a class of affine continuous VASS is decidable if the class contains only permutation matrices and is undecidable for *any other class*.

State-Reachability. As our second contribution, we completely resolve the decidability status of state-reachability. We show that state-reachability for a class of affine continuous VASS is decidable (and in PSPACE) if it contains only non-negative matrices and is undecidable for any other class. This proves, for instance, that state-reachability is in PSPACE for continuous VASS with resets, transfers, copies or doubling operations. To the best of our knowledge, the PSPACE upper bounds for these models were not known before.

Coverability. As our third contribution, except for a single family of classes of of affine matrices, we resolve the decidability status of coverability. We show that coverability is decidable for classes containing only permutation matrices and is undecidable for any class containing a matrix with a negative entry or a matrix with a row or a column that is entirely comprised of zeroes. Since continuous VASS with resets or transfers belong to the last category of matrices, it follows that *the coverability problem for continuous VASS with resets or transfers is undecidable.* This result is surprising, because the coverability problem for (usual) VASS with resets or transfers is decidable [15]. Hence, while the continuous semantics drastically reduces the complexity for VASS, for resets and transfers it actually strictly increases in complexity and power.

As we show in the next section, the only remaining classes of matrices for which we were not able to prove a decidability or undecidability result for coverability are non-negative classes without any zero-rows or columns which contain at least one matrix possessing a weighted edge or two overlapping edges. We will explain these notions formally in Section 2, but we describe them informally here. We can think of a matrix as a weighted adjacency matrix of some graph. A matrix is then said to have a weighted edge if there is a edge in its graph whose value is strictly larger than 1. Also, a matrix is said to have two overlapping edges if there are two edges in the graph whose target node is the same. Even though we were not able to resolve the decidability status for this family, we identify a non-trivial sub-family of matrices in this family for which coverability is decidable. This sub-family contains any class that contains only non-negative self-loop matrices, i.e., non-negative matrices such that each diagonal entry is non-zero. For any such class, we show that coverability is decidable and NP-complete, and hence no harder than coverability for continuous VASS.

Computational Complexity. We also complement all of our upper bounds with matching lower bounds, except for a single case: State-reachability for non-negative classes without any zero-rows or columns containing weighted/overlapping edges.

More precisely, we show that reachability is NEXP-complete for classes containing only permutation matrices with at least one non-trivial permutation matrix i.e., a permutation matrix that is not the identity matrix. We also show that coverability (resp. state-reachability) is NEXP-complete (resp. PSPACE-complete) for classes containing only permutation matrices with at least one non-trivial permutation matrix and is NP-complete for non-negative classes with only self-loop matrices. Finally, we show that state-reachability is PSPACE-complete for non-negative classes containing a matrix which has either a row or a column that is completely zero.

All of these results are formally defined and stated in the next section, with a summary in Table 2.2.

Related work. The closest work to ours is the paper [11]. In this paper, the authors consider reachability in affine VASS and affine integer VASS according to classes of affine operations and they completely classify the decidability of reachability for both these models across all classes. The latter model is an over-approximation of affine VASS in which the counters are allowed to go below zero. The authors also prove a complexity trichotomy for affine integer VASS. In particular, the authors prove that reachability for integer VASS with resets or transfers is decidable and mention that it is

undecidable for the continuous case. In this paper, we also use the formalization of classes used in [11].

The paper [8] also discusses affine integer VASS. The authors of that paper prove that when the set of matrices that are considered form a finite monoid, then reachability becomes decidable for affine integer VASS. They also prove some other results for infinite monoids.

Our techniques. Our results use a number of novel techniques that are different from the ones used for affine VASS and affine integer VASS. Below, we explain some of our techniques which illustrate why many of the results developed for affine VASS and affine integer VASS are not applicable to affine continuous VASS. This also exhibits many surprising phenomena in the continuous case.

Continuous VASS with resets. As a first point of departure from the usual and integer semantics, we show that coverability for continuous VASS with resets is undecidable. This is surprising, because in both the usual and the integer semantics, this problem is decidable. This illustrates the complex nature of the continuous semantics - while it reduces the complexity for VASS, for VASS with resets it actually makes it harder. To prove this undecidability result, we first show that the 1-bounded coverability problem for continuous VASS with zero-tests is undecidable. In this problem, we are given a continuous VASS which is allowed to perform zero-tests and two of its configurations. We are then asked to check if the first configuration can cover the second one by a run in which all the counter values always stay below 1. We show that this problem is undecidable by observing that runs in the continuous semantics can be shrunk by arbitrary fractions.

We then reduce 1-bounded coverability for continuous VASS with zero-tests to coverability in reset VASS. To do this, for every counter x of the given continuous VASS with zero-tests, we introduce a new *complementary counter* $\overline{x}$ and maintain the invariant that the sum of the values of x and $\overline{x}$ is 1 throughout. Then we replace every zero-test with a reset. The idea is that if the new machine resets some counter x when it has a non-zero value, then the invariant that $x + \overline{x} = 1$ is permanently destroyed and it instead becomes $x + \overline{x} < 1$. Since the sum of x and $\overline{x}$ in the final configuration will be 1, it would then follow that the new machine will reset a counter only when it achieves the value 0, thereby mimicking zero-tests.

Reachability for weighted/overlapping edges. While we do not resolve the status of coverability for classes with non-negative weighted/overlapping edges, we prove that reachability for such classes is undecidable. A similar result is true for affine VASS, which can be proved by showing that such VASS can implement the doubling operation [11]. The reduction in this case crucially hinges on being able to double the current value of a counter and also being able to add 1 to a counter. While the former can also be mimicked in the continuous case, the latter cannot be, as the additive updates are scaled with a non-deterministically chosen fraction $\alpha \in (0, 1]$. It might be tempting to overcome this by adding a fresh counter which always has the value 1 and then by using matrix operations, add this fresh counter's value whenever we want to add 1; however, this operation might not be allowed within the class. We overcome this problem by giving a reduction from the 1-bounded reachability

problem for continuous VASS with zero-tests. Given a continuous VASS with zero-tests, to every counter x, we attach a complementary counter $\overline{x}$ and a collection of other dummy counters. Further, we maintain the invariant that the sum of the values of x, $\overline{x}$ and the dummy counters is 1. We then identify a matrix from this class such that when it is multiplied with a counter valuation in which x and the dummy counters have value 0, then the invariant remains; otherwise the invariant is permanently broken. This then allows us to simulate zero-tests.

State-reachability for non-negative matrices. In affine VASS, state-reachability is inter-reducible with coverability, which is decidable for classes which contain only non-negative matrices, i.e., matrices in which all entries are non-negative. This is because affine VASS with non-negative matrices are well-structured [1, 15]. However, since non-negative rationals are not well-quasi ordered (under the usual ordering), we cannot apply the theory of well-structured transition systems in our setting of affine continuous VASS. Nevertheless, we show that the state-reachability problem is decidable and in PSPACE for affine continuous VASS for classes that contain only non-negative matrices. We do this by constructing a *support abstraction* graph from a given affine continuous VASS. The nodes of this graph only maintain the current control state and the current set of counters which have a non-zero value. Edges between the nodes of this graph are defined in a manner which satisfy some basic conditions: For instance, if a transition t increments some counter x and if $(p, S) \xrightarrow{t} (p', S')$ is an edge in this graph labelled by t then S' must contain x. This is because whenever t is fired in the affine continuous VASS it will necessarily lead to a configuration with a non-zero value in x. We then show that maintaining some such basic conditions is necessary and sufficient to decide state-reachability and that this can be done in PSPACE. As a particular example, this proves that state-reachability for continuous VASS with resets is decidable and proves that, unlike the case of affine VASS, coverability and state-reachability are two different problems for affine continuous VASS.

Coverability for self-loop matrices. Finally, to prove that coverability for non-negative self-loop matrices (i.e., non-negative matrices in which no diagonal entry is zero) is in NP, we use results developed for continuous VASS. By adapting some techniques developed for continuous VASS, we show that the coverability relation of an affine continuous VASS containing only non-negative self-loop matrices can be defined as a formula in the existential theory of rationals equipped with the addition and the order operation (ELRA). The techniques developed for continuous VASS cannot be applied in a black-box manner and require some careful restructuring of the arguments so that it can be used for the affine case, tailored specifically to the case of coverability. This is because for any class of matrices containing a self-loop matrix that is not the identity matrix, the reachability relation cannot be reduced to formulas in the ELRA theory, since our results for reachability imply undecidability in this case. This is in contrast to the case of continuous VASS for which the reachability relation can be embedded in ELRA.

To prove the decidability of coverability in self-loop matrices, we first identify counters which could be potentially *pumped*, i.e., these are counters which while taking some transition, either double

their old value or retain their old value and also get the value of another counter by means of the matrix of that transition. Since pumpable counters only depend on the matrix of a transition and not on its additive update, we can choose arbitrarily small fractions and still get big values for pumpable counters. Subject to some mild technical constraints, we show how to modify a given run that starts and ends at the same state and pumps some counters, into a run so that all pumped counters attain arbitrarily high values whilst not modifying the values of the non-pumped counters. This is done by roughly splitting a run into many copies of the same run with smaller fractions. In the end, this would mean that with respect to coverability, we can get rid of the pumpable counters and only focus on the non-pumpable ones. We then notice that the affine VASS with only the non-pumpable counters is a continuous VASS, for which we know how to decide coverability.

Organization of the paper. The rest of the paper is organized as follows. In Section 2 we introduce the notions of affine continuous VASS and classes of matrices and state our main results. In Section 3 we prove all of our undecidability results. In Section 4 we prove all of our decidability and upper bound results. In Section 5 we prove all of our lower bound results. Finally we conclude in Section 6.

All missing proofs can be found in the appendix of this paper.

2 Preliminaries

In this section, we define the notion of affine continuous VASS and the associated problems that we shall consider. Then, we define the notion of a *class of matrices*, akin to [11] and state our main results.

Throughout this paper we let $\mathbb{Q}_+$ denote the set of non-negative rational numbers and $\mathbb{Q}_{>0}$ denote the set of positive rational numbers. We usually write vectors $(u_1, \ldots, u_d) \in \mathbb{Q}^d$ in bold as $\mathbf{u}$ and we use $\mathbf{u}(i)$ to denote u_i. Given some $\alpha \in \mathbb{Q}$ and a vector $\mathbf{v}$, we let $\alpha \mathbf{v}$ be the vector given by $(\alpha \mathbf{v})(i) = \alpha \cdot \mathbf{v}(i)$. The support of a vector $\mathbf{v}$, denoted by $[\![\mathbf{v}]\!]$ is the set $\{i : \mathbf{v}(i) \neq 0\}$. We use $\mathbf{0}$ to denote the vector that is 0 everywhere and we let $\mathbf{e}_i$ denote the ith unit vector, i.e., $\mathbf{e}_i$ is 0 everywhere, except in the ith co-ordinate, where its value is 1. Given two vectors $\mathbf{u}, \mathbf{v}$ we say that $\mathbf{u} \leq \mathbf{v}$ if $\mathbf{u}(i) \leq \mathbf{v}(i)$ for every i.

Given a matrix $\mathbf{A}$ we use $\mathbf{A}(i, j)$ to denote its entry in the ith row and the jth column. Further, given a matrix $\mathbf{A}$ and a vector $\mathbf{u}$, we let $\mathbf{Au}$ denote the vector obtained by multiplying $\mathbf{A}$ with $\mathbf{u}$. Throughout this paper, we will only work with square matrices, i.e., matrices whose number of rows and columns are the same. A matrix $\mathbf{A}$ is called a *diagonal matrix* if all of its non-diagonal entries are 0, i.e., $\mathbf{A}_{i,j} = 0$ whenever $i \neq j$.

Affine continuous VASS. Let $d \in \mathbb{N}$. A d-affine continuous VASS (or simply affine continuous VASS) is a tuple $\mathcal{M} = (Q, T)$ where Q is a finite set of *control states* and $T \subseteq Q \times \mathbb{Z}^{d \times d} \times \mathbb{Z}^d \times Q$ is a finite set of *transitions*. Note that the matrices and the vectors appearing in the transitions range over the integers. Intuitively, $\mathcal{M}$ has access to d counters, each of which can store a non-negative rational value. The values of these counters can be manipulated by the transitions in T as follows.

Given a transition $t = (p, \mathbf{A}, \mathbf{b}, q)$, we will call $\mathbf{A}$ as the matrix of t and $\mathbf{b}$ as the additive update of t. A *configuration* of $\mathcal{M}$ is a pair $(p, \mathbf{u})$ (mostly written as $p(\mathbf{u})$), where $p \in Q$ is a state and $\mathbf{u} \in \mathbb{Q}_+^d$

is a non-negative vector representing the current values of the d counters. If $C = p(\mathbf{u})$ is a configuration, then we let $C(i) = \mathbf{u}(i)$ for any counter i. Given two configurations $p(\mathbf{u}), q(\mathbf{v})$, a non-zero fraction $\alpha \in (0, 1]$, and a transition t of the form $t = (p, \mathbf{A}, \mathbf{b}, q)$, we say that there is a step from $p(\mathbf{u})$ to $q(\mathbf{v})$ by means of the pair αt, denoted by $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ if $\mathbf{v} = \mathbf{Au} + \alpha \mathbf{b}$. Intuitively, from the configuration $p(\mathbf{u})$, the machine executes the transition t and goes to the state q and updates the counters by multiplying the current vector $\mathbf{u}$ with the matrix $\mathbf{A}$ and then adding the vector $\alpha \mathbf{b}$. Note that by the definition of a configuration, we implicitly require that $\mathbf{Au} + \alpha \mathbf{b} \geq \mathbf{0}$. We say that $p(\mathbf{u}) \to q(\mathbf{v})$, if there exists some non-zero fraction α and some transition t such that $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$.

A *firing sequence* σ is a sequence in $((0, 1] \times T)^*$. Given a firing sequence $\sigma = \alpha_1 t_1, \ldots, \alpha_k t_k$ and a non-zero fraction β, we let $\beta \sigma$ denote the firing sequence $\beta \sigma = \beta \alpha_1 t_1, \ldots, \beta \alpha_k t_k$.

Given two configurations C and C' and a firing sequence $\sigma = \alpha_1 t_1, \ldots, \alpha_k t_k$, we say that $C \xrightarrow{\sigma} C'$ if there exists configurations $C_1, \ldots, C_{k-1}$ such that $C \xrightarrow{\alpha_1 t_1} C_1 \xrightarrow{\alpha_2 t_2} C_2 \ldots C_{k-1} \xrightarrow{\alpha_k t_k} C'$. Finally, we write $C \xrightarrow{*} C'$ to mean that there is some firing sequence σ such that $C \xrightarrow{\sigma} C'$.

Note that if all the matrices appearing in an affine continuous VASS $\mathcal{M}$ are the identity matrices, then we get the usual notion of continuous VASS [7].

Classes of matrices. Our aim in this paper is to provide a classification of "classes of affine continuous VASS" with respect to decidability of the reachability, coverability and state-reachability problems. To this end, we need to formalize the notion of a "class of affine continuous VASS". For this purpose, we use the framework used in [11], which was used for classifying *affine VASS* and *affine integer VASS* with respect to the reachability problem.

A class of affine VASS extends the usual notion of a VASS by providing some extra operations that can be applied to the counters by means of the affine transformations that it provides, i.e., the set of allowed matrices in its transitions. Since affine VASS extend VASS, the identity matrix must always be allowed. Also, the authors of [11] noticed that the set of allowed matrices must be closed under multiplication. Finally, a class of affine VASS typically does not pose any restrictions on the number of counters that can be used, or on the subset of counters to which the operations can be applied. This means that a matrix can be extended to arbitrary dimensions and can be applied to any subset of counters. These observations are utilized in the definition of a class used in [11]. We now describe this formal defintion of a class and use it to classify affine continuous VASS as well.

For every $k \geq 1$, let $\mathbf{I}_k$ be the $k \times k$ identity matrix. We use $\mathcal{S}_k$ to denote the set of all permutations over $\{1, 2, \ldots, k\}$. For any permutation $\sigma \in \mathcal{S}_k$, let $\mathbf{P}_\sigma \in \{0, 1\}^{k \times k}$ be its corresponding matrix, i.e., for any $1 \leq i, j \leq k$, $\mathbf{P}_\sigma(i, j)$ is 1 if i is mapped to j and 0 otherwise. For any matrix $\mathbf{A} \in \mathbb{Z}^{k \times k}$ and any permutation $\sigma \in \mathcal{S}_k$ we let

$$\sigma(\mathbf{A}) := \mathbf{P}_\sigma \cdot \mathbf{A} \cdot \mathbf{P}_{\sigma^{-1}}$$

and for any $n \geq 1$, we let $\mathbf{A} \upharpoonright_n$ be the $(k+n) \times (k+n)$ matrix defined as

$$\mathbf{A} \upharpoonright_n := \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix}$$

Intuitively, if $\mathbf{A}$ is a $k \times k$ matrix that dictates the updates of some k counters of an affine continuous VASS, then $\sigma(\mathbf{A})$ essentially renames the k counters according to the permutation σ, applies $\mathbf{A}$ according to this renaming and then renames the counters back to the original names. The operation $\mathbf{A}\!\upharpoonright_n$ can be thought of as adding n fresh counters and then applying the matrix $\mathbf{A}$ only on the old k counters, while leaving the new n counters unchanged.

Now, a *class of matrices C* is any subset of matrices in $\cup_{k \geq 1} \mathbb{Z}^{k \times k}$ that satisfies the following four closure properties:

- Identity: For every n, $\mathbf{I}_n \in C$.
- Multiplication: If $\mathbf{A}, \mathbf{B} \in C$ are $k \times k$ matrices for some $k \geq 1$, then the matrix $\mathbf{A} \cdot \mathbf{B}$ is also in C.
- Extension: If $\mathbf{A} \in C$, then for every n, the matrix $\mathbf{A}\!\upharpoonright_n$ is also in C.
- Counter renaming: If $\mathbf{A} \in C$ is a $k \times k$ matrix, then $\sigma(\mathbf{A})$ is also in C for any permutation $\sigma \in \mathcal{S}_k$.

We also define a new operation called Application: If $\mathbf{A}$ is a $k \times k$ matrix and i, n are numbers such that $k \leq n$ and $i + k - 1 \leq n$, then $\mathrm{Apply}_n(\mathbf{A}, i)$ is the $n \times n$ matrix defined as

$$\mathrm{Apply}_n(\mathbf{A}, i) := \begin{pmatrix} \mathbf{I}_{i-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n-i-k+1} \end{pmatrix}$$

Intuitively, $\mathrm{Apply}_n(\mathbf{A}, i)$ applies the matrix A to dimensions $i, i + 1, \ldots, i + k - 1$ and leaves the other dimensions unchanged. Note that $\mathrm{Apply}_n(\mathbf{A}, i)$ can be obtained by extending $\mathbf{A}$ and renaming some counters of its extension: Indeed, $\mathrm{Apply}_n(\mathbf{A}, i) = \sigma(\mathbf{A}\!\upharpoonright_n)$ where σ is the permutation that maps every j to $i - 1 + j$ if $i - 1 + j \leq n$ and otherwise to $(i - 1 + j) \bmod n$. Hence, Application is simply a syntactic sugar and all classes are closed under the Application operation. As we shall see later on, the introduction of the Application operation will help us state us our constructions in an easier way.

We now show that the notion of a class of matrices can capture various sub-models of affine continuous VASS, each of which arise from continuous VASS by adding some new feature.

Example 2.1 (Classes of matrices). As mentioned before, a continuous VASS is an affine continuous VASS in which all the matrices are identity matrices. This is captured by the *identity class* of matrices, i.e., the class that contains only the set of all identity matrices. For a d-continuous VASS $\mathcal{M} = (Q, T)$ we will simply ignore the (identity) matrices in the transitions and denote T by $T \subseteq Q \times \mathbb{Z}^d \times Q$.

A continuous VASS with reset operations is a continuous VASS in which transitions are further allowed to reset the value of a counter to 0. This model is captured by affine continuous VASS with matrices from the class C_R which contains the set of all diagonal matrices with entries from $\{0, 1\}$. Intuitively, if $\mathbf{A} \in C_R$, and its ith diagonal entry is a 0, then this means that the ith counter is reset to 0 and if it is a 1, then the ith counter is left unchanged.

Another example is continuous VASS with transfer operations which extend continuous VASS with operations which allow the value of a counter i to be transferred to a counter j. This model is captured by affine continuous VASS with matrices from the class of all matrices with entries from $\{0, 1\}$ in which every column has at most one non-zero entry.

2.1 Contribution Part I - Decidability results

We now state the main contributions of the paper. Given a class C of matrices, we say that an affine continuous VASS $\mathcal{M}$ is a C-continuous VASS if all the matrices appearing in $\mathcal{M}$ are in C. We are now ready to define the problems of interest to us and state our main results.

The *reachability problem* for a class C of matrices is defined as the following:

> **Given:** A C-continuous VASS $\mathcal{M}$ and two of its configurations $p(\mathbf{u}), q(\mathbf{v})$.
> **Decide:** Whether $p(\mathbf{u})$ can reach $q(\mathbf{v})$.

Our first main result is a classification of the reachability problem, which completely characterizes the decidability/undecidability border. To state this result, we set up a notation: We say that a matrix $\mathbf{A}$ is a *permutation matrix* if $\mathbf{A} = \mathbf{P}_\sigma$ for some permutation σ. Our result for the reachability problem can then be phrased as the following.

THEOREM 2.2. *The reachability problem for a class C is decidable (and in NEXP) if C contains only permutation matrices. Otherwise, the reachability problem is undecidable.*

For our second result, we consider the *state-reachability problem* for a class C which is defined as follows:

> **Given:** A C-continuous VASS $\mathcal{M}$, a configuration $p(\mathbf{u})$ and a state q.
> **Decide:** Whether $p(\mathbf{u})$ can reach some configuration of the form $q(\mathbf{v}')$ for any non-negative $\mathbf{v}'$.

Our second main result completely classifies the decidability/undecidability border for the state-reachability problem. To state this result, we set up a notation: We say that a matrix $\mathbf{A}$ is a *non-negative matrix* if all of its entries are non-negative. Our result for the state-reachability problem can then be phrased as the following.

THEOREM 2.3. *The state-reachability problem for a class C is decidable (and in PSPACE) if C contains only non-negative matrices. Otherwise, the state-reachability problem is undecidable.*

Finally, we consider the *coverability problem* for a class C which is defined as follows:

> **Given:** A C-continuous VASS $\mathcal{M}$, and two of its configurations $p(\mathbf{u}), q(\mathbf{v})$.
> **Decide:** Whether $p(\mathbf{u})$ can *cover* $q(\mathbf{v})$, i.e., whether $p(\mathbf{u})$ can reach some configuration of the form $q(\mathbf{v}')$ where $\mathbf{v}' \geq \mathbf{v}$.

Our third main result classifies the decidability/undecidability border for the coverability problem, except for a single family of classes. To state our result for this problem we need a notation. A matrix is said to have a *zero-row* (resp. *zero-column*) if it contains a row that is completely zero (resp. a column that is completely zero). Our result for the coverability problem is then as follows.

THEOREM 2.4. *The coverability problem for a class C is undecidable if C contains a matrix with either a negative entry or a zero row or a zero column. It is decidable (and in NEXP) if C contains only permutation matrices.*

To see how this result classifies the coverability problem except for a single family, we need some definitions. A matrix $\mathbf{A}$ is said to

have a weighted edge if there exists indices i, j such that $\mathbf{A}(i, j) > 1$. Further, $\mathbf{A}$ is said to have two overlapping edges if there exists indices $i \neq j$ and k such that $\mathbf{A}(i, k) > 0$ and $\mathbf{A}(j, k) > 0$. The intuitive idea is to think of $\mathbf{A}$ as the weighted adjacency matrix of a graph. Having a weighted edge in $\mathbf{A}$ corresponds to having an edge in the graph whose value is strictly bigger than 1. Having two overlapping edges in $\mathbf{A}$ corresponds to having two edges in the graph whose target nodes are the same.

Now, suppose $\mathbf{A}$ is a non-negative matrix (over $\mathbb{Z}$) which does not contain a zero-row, a zero-column, a weighted edge or two overlapping edges. Hence, for every index ℓ, there is a unique index $\sigma(\ell)$ such that $\mathbf{A}(\sigma(\ell), \ell) = 1$ and $\mathbf{A}(\ell', \ell) = 0$ for every $\ell' \neq \sigma(\ell)$. We claim that $\sigma(i) \neq \sigma(j)$ whenever $i \neq j$. Suppose there exist $i \neq j$ such that $\sigma(i) = \sigma(j) = k$. Then, since $\mathbf{A}$ does not contain a zero-row, for every index ℓ, there is some index $\tau(\ell)$ such that $\mathbf{A}(\ell, \tau(\ell)) > 0$. Note that if $\ell' = \tau(\ell)$, then $\sigma(\ell') = \ell$.

Now, starting from i, using τ, we get a sequence of the form $\tau^0(i) := i$, $\tau^1(i) := \tau(i)$, $\tau^2(i) := \tau(\tau(i)) \ldots$ Hence, there must be some r such that $\tau^0(i), \cdots, \tau^r(i)$ are all distinct and $\tau^{r+1}(i) = \tau^e(i)$ for some $0 \leq e \leq r$. If $e \neq 0$, then $\mathbf{A}(\tau^{e-1}(i), \tau^e(i)) > 0$ and $\mathbf{A}(\tau^r(i), \tau^e(i)) > 0$, which is a contradiction because this leads to two overlapping edges. Hence, $e = 0$ and so $\tau^r(i) = \sigma(i) = k$. Similarly, when this argument is applied to j, we get a similar sequence of indices $j, \tau(j), \cdots, \tau^w(j) = k$ for some w.

Now, by definition of σ and τ we have that, $\sigma^r(k) = i$ and $\sigma^w(k) = j$. Since $i \neq j$, we have $w \neq r$. Without loss of generality, let $w < r$. This means that $\sigma^{r-w}(j) = i$ and so $\tau^{r-w}(i) = j$. Since $\tau^r(i) = k$ and since $\mathbf{A}(k, j) > 0$, we have $\mathbf{A}(\tau^{r-w-1}(i), j) > 0$ and $\mathbf{A}(\tau^r(i), j) > 0$, which is a contradiction because this leads to two overlapping edges. This means that $\sigma(i) \neq \sigma(j)$ whenever $i \neq j$. It can then be easily seen that $\mathbf{A}$ is a permutation matrix.

The discussion above then shows that our result for coverability covers all classes except those that contain matrices with weighted/overlapping edges, which we will call the weighted/overlapping family. However, even within this family, we identify a subfamily for which the coverability problem is decidable. We now describe this class of matrices.

A matrix $\mathbf{A}$ is said to be a *self-loop matrix* if no entry on the diagonal is zero. The intuition once again comes from graphs: If $\mathbf{A}$ is a self-loop matrix, then in the graph of $\mathbf{A}$, all nodes will have a self-loop. We show that

THEOREM 2.5. *The coverability problem for C is decidable (and in NP) if C contains only non-negative self-loop matrices.*

Note that NP is the best possible upper bound possible for a class, since the coverability problem for continuous VASS is NP-complete. Hence, we have strictly extended the model of continuous VASS whilst retaining the same complexity for the coverability problem.

2.2 Contribution Part II - Complexity results

In addition to the above results, we complement all of our upper bounds with lower bounds, except for a single family: The weighted/overlapping family. Our results are then as follows.

THEOREM 2.6. *The reachability and coverability problems for the class C are NEXP-complete if C contains only permutation matrices*

with at least one non-trivial permutation matrix. For the identity class both these problems are NP-complete.

Here a non-trivial permutation matrix is a permutation matrix that is not the identity matrix. Note that along with our decidability result for the reachability problem, this completely characterizes the complexity of reachability for every class. We now move on to the state-reachability problem.

THEOREM 2.7. *The state-reachability problem for C is PSPACE-complete, if C contains only permutation matrices with at least one non-trivial permutation matrix or if C contains a zero-row or a zero-column matrix. It is NP-complete if C is the identity class or C is any class containing only self-loop matrices.*

All of our results are summarized in Table 2.2. We note that the results for the identity class, i.e., continuous VASS, are already known [7] and are only included here and in the previous two theorems for the sake of completeness.

Table 1: All of our results. The leftmost column contains types of matrices and the uppermost row contains the problems. An undecidability/hardness result for a cell means that the problem in that column is undecidable/hard for any class containing *any* matrix in that row. A decidability/complexity upper bound result for a cell means that the problem in that column is decidable with the given upper bound for classes containing *no* matrices from any of the previous rows and containing *only* those matrices in that row.

	Reach	Cover	State-reach
neg. entry matrices	Undec.	Undec.	Undec.
zero-row/column	Undec.	Undec.	PSPACE-c
wt./overlap. edges	Undec.	?	PSPACE
		NP-c (self-loop)	NP-c (self-loop)
non-trivial perm.	NEXP-c	NEXP-c	PSPACE-c
identity matrices	NP-c	NP-c	NP-c

Generic reductions between reachability, coverability and state-reachability. It is easy to see that state-reachability is a special case of coverability where we want to cover the **0** vector. Further, coverability can be easily reduced to reachability: Given an instance $(\mathcal{M}, p(\mathbf{u}), q(\mathbf{v}))$ of the coverability problem, we modify $\mathcal{M}$ to first add a transition which moves from q to a new state q'. Then at q' we add loops to decrement every possible counter. It then follows that $p(\mathbf{u})$ can cover $q(\mathbf{v})$ in $\mathcal{M}$ if and only if $p(\mathbf{u})$ can reach $q'(\mathbf{v})$ in the new machine. Hence, we have

THEOREM 2.8. *For any class C, the state-reachability problem for C is poly.-time reducible to the coverability problem for C which in turn is poly.-time reducible to the reachability problem for C.*

We will use this theorem to simplify the presentation of many of our results.

3 Undecidability results

In this section, we prove all of our undecidability results. We begin with our undecidability result on classes containing a matrix with a negative entry.

3.1 State-reachability for classes with negative entries

Our main result of this section is that

THEOREM 3.1. *The reachability, coverability and state-reachability problems for C-continuous VASS is undecidable if C contains a matrix with some negative entry.*

Note that by Theorem 2.8, it suffices to prove the above theorem for only the state-reachability problem. To prove this result, we introduce the notion of a continuous VASS with zero tests. These are continuous VASS which can additionally test if the value of a counter is zero. Formally, a d-continuous VASS with zero tests is a tuple $M = (Q, T, T_{=0})$ where Q is a finite set of states, $T \subseteq Q \times \mathbb{Z}^d \times Q$ is a finite set of transitions and $T_{=0} \subseteq Q \times [d] \times Q$ is a finite set of zero-tests. Configurations of M and steps of M by transitions from T are defined in the same way as for a continuous VASS. In addition, we also have steps defined by zero-tests in the following manner. Given two configurations $p(\mathbf{u}), q(\mathbf{v})$ and a zero-test t of the form $t = (p, i, q)$, we say that there is a step from $p(\mathbf{u})$ to $q(\mathbf{v})$ by means of the zero-test t, denoted by $p(\mathbf{u}) \xrightarrow{t} q(\mathbf{v})$ if $\mathbf{u}(i) = 0$ and $\mathbf{v} = \mathbf{u}$. (Note that steps corresponding to a zero-test do not have any chosen fractions). A run is then defined as a sequence of steps (defined by transitions and zero-tests) from one configuration to another. Reachability, coverability and state reachability are then defined in the same way as for a continuous VASS. From [7, Theorem 4.17], we have the following result.

THEOREM 3.2. *The reachability, coverability and state-reachability problems for continuous VASS with zero-tests are undecidable.*

Using this result, we shall prove Theorem 3.1.

PROOF OF THEOREM 3.1. Let C be a class of matrices which has a matrix $\mathbf{A}$ (of size $k \times k$ for some k) such that $\mathbf{A}(i, j) < 0$ for some indices i and j. First, we note the following property: For any $\lambda \in \mathbb{Q}_+$, we have

$$\mathbf{A} \cdot \lambda \mathbf{e}_j = \begin{cases} \mathbf{0} & \text{if } \lambda = 0 \\ \lambda \cdot \mathbf{A}_j & \text{otherwise, where } \mathbf{A}_j \text{ is the } j^{th} \text{ column of } \mathbf{A} \end{cases}$$
$$(1)$$

This means that if we multiply $\mathbf{A}$ by a non-negative vector $\mathbf{v}$ which is zero everywhere, except possibly in the jth counter, then the only way we can have a non-negative vector again is if $\mathbf{v}$ is $\mathbf{0}$. Intuitively, this means that if we have a configuration which is zero everywhere, except possibly in the jth counter, then to test if the value of the jth counter is 0, we simply have to multiply the configuration with the matrix $\mathbf{A}$. In this way, we can simulate a zero-test on counter j by the matrix $\mathbf{A}$.

We now show that C-continuous VASS can simulate continuous VASS with zero tests, which by Theorem 3.2 will prove Theorem 3.1. The reduction is similar to the one given in [11, Proposition 4.3].

Let $M = (Q, T, T_{=0})$ be a d-continuous VASS with zero tests for some d. From M, we will create a $n = dk$ dimensional C-continuous VASS M'. For every $1 \leq i \leq d$, the $(j + (i-1)k)^{th}$ counter of M' (which we will denote by x_i) will be responsible for simulating the ith counter of M. These counters, $x_1, x_2, \ldots, x_d$, will be called the *primary counters* of M'. The remaining counters of M' will be called *dummy counters* which will always have the value 0.

Given a vector $\Delta \in \mathbb{Q}^d$, let $\text{ext}(\Delta) \in \mathbb{Q}^n$ be the vector which is 0 everywhere, except in counters $x_1, x_2, \ldots, x_d$ where the values are respectively $\Delta(1), \Delta(2), \ldots, \Delta(d)$. With this notation set up, we are ready to state the desired reduction.

The set of states of M' will be the same as Q. For each $t \in T \cup T_{=0}$ of M, M' will have a corresponding transition t'. Before we state this transition t', we will state two properties that we shall prove will be satisfied by t'.

> PROPERTY 1: If $(p, \text{ext}(\mathbf{u})) \xrightarrow{\alpha t'} q(\mathbf{v}')$ is a step in M', then $\mathbf{v}' = \text{ext}(\mathbf{v})$ for some vector $\mathbf{v} \in \mathbb{Q}_+^d$.
>
> PROPERTY 2A): If $t \in T$ is a transition of M, then $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in M if and only if $p(\text{ext}(\mathbf{u})) \xrightarrow{\alpha t'} q(\text{ext}(\mathbf{v}))$ is a step in M'.
>
> PROPERTY 2B): If $t \in T_{=0}$ is a zero-test of M, then $p(\mathbf{u}) \xrightarrow{t} q(\mathbf{v})$ is a step in M if and only if $p(\text{ext}(\mathbf{u})) \xrightarrow{\alpha t'} q(\text{ext}(\mathbf{v}))$ is a step in M' for all $\alpha \in (0, 1]$.

We now proceed to state the transitions of M'. If $t = (p, \Delta, q)$ is a transition in T, then corresponding to t, we will have the transition $t' = (p, \mathbf{I}_n, \text{ext}(\Delta), q)$ in M'. Note that t' updates the counters $x_1, x_2, \ldots, x_d$ of M' in the same way as t updates the counters $1, 2, \ldots, d$ of M. Further, t' does not update the dummy counters of M' at all. It can then be easily seen that t' satisfies both the properties.

If $t = (p, \ell, q)$ is a zero-test in $T_{=0}$, then corresponding to t, we will have the transition $t' = (p, \text{Apply}_n(\mathbf{A}, (\ell - 1)k + 1), \mathbf{0}, q)$ in M'. By definition of the Application operation, t' does not update any counters in the set $\{1, 2, \ldots, (\ell - 1)k\} \cup \{\ell k + 1, \ldots n\}$. Furthermore, by equation 1, if at some configuration, the values of the counters in the set $\{(\ell - 1)k + 1, \ldots, (\ell - 1)k + j - 1\} \cup \{(\ell - 1)k + j + 1, \ldots, \ell k\}$ are zero, then the only way t' can be fired from that configuration is if the value of the counter $(\ell - 1)k + j = x_\ell$ is also zero. Further, firing t' only results in all of the counters in the set $\{(\ell - 1)k + 1, (\ell - 1)k + 2, \ldots, \ell k\}$ taking the value 0. Hence, t' also satisfies both of these properties.

By using both the properties, it then follows that in M, a configuration $p(\mathbf{u})$ can reach a configuration with state q if and only if in M', the configuration $p(\text{ext}(\mathbf{u}))$ can reach a configuration with state q. Hence, the desired reduction is complete. □

3.2 Coverability for non-negative classes with zero-rows/columns

Because of the negative result from the previous subsection, it suffices to only consider classes which only contain non-negative matrices for the rest of the paper. Such a class will be called a non-negative class. We are now ready to state the main result of this subsection.

THEOREM 3.3. *The reachability and coverability problems for C-continuous VASS is undecidable if C is a non-negative class which contains a zero-row or a zero-column matrix.*

By Theorem 2.8, it suffices to prove the above theorem only for the coverability problem. We do this in two stages. In the first stage, we show that coverability for continuous VASS with resets, i.e., C_R-continuous VASS where C_R is the class of all diagonal matrices with

entries from $\{0, 1\}$ is undecidable. In the second stage, we reduce from coverability for continuous VASS with resets to coverability for C-continuous VASS where C contains a zero-row or a zero-column matrix.

Stage 1: Continuous VASS with resets. A continuous VASS with resets is simply an affine continuous VASS in which all the matrices that appear are diagonal matrices with entries from $\{0, 1\}$. For every d, i, let $\mathbf{R}^{(d,i)}$ be the $d \times d$ diagonal matrix which has a 1 in each entry of the diagonal, except at the position (i, i), where the value is 0. Intuitively, if $\mathbf{u}$ is some vector and $\mathbf{v} = \mathbf{R}^{(d,i)}(\mathbf{u})$, then $\mathbf{v}$ is the same as $\mathbf{u}$, except that $\mathbf{v}(i) = 0$. The idea is that if $\mathbf{u}$ represents the current contents of all the counters, then $\mathbf{R}^{(d,i)}$ resets the value of the counter i to 0 and does not alter the contents of the other counters. In the sequel, whenever we say that a transition resets counter i, we mean that the current counter values are multiplied by $\mathbf{R}^{(d,i)}$. We will now show that

Theorem 3.4. *The coverability problem for continuous VASS with resets is undecidable.*

This theorem is surprising, because coverability in the usual VASS model enriched with resets is decidable [15]. Hence, while the continuous semantics drastically reduces the expressive power in VASS, upon addition of resets, the continuous semantics becomes even more powerful than the usual semantics.

To prove Theorem 3.4, we first introduce the following notion. Suppose $\mathcal{M}$ is a continuous VASS *with zero-tests* and C and C' are configurations of $\mathcal{M}$. A 1-bounded run between C and C' is a run in which all the counter values along the run are at most 1; in particular this means that the counter values of C and C' are at most 1. The 1-bounded coverability problem for continuous VASS with zero-tests is the following: Given a tuple $(\mathcal{M}, C, C')$, decide if there is a 1-bounded run from C which can cover C'. In order to prove Theorem 3.4, we first prove the following lemma.

Lemma 3.5. *The 1-bounded coverability problem is undecidable for continuous VASS with zero-tests.*

The proof of this lemma can be found in Subsection A.1. The intuitive idea behind this lemma is that continuous VASS with zero-tests have a *scaling property*, which allows runs to be *shrunk* by arbitrary fractions. More precisely, it is easy to see that whenever $D \xrightarrow{\alpha t} D'$ is a step in a continuous VASS with zero-tests where t is a transition, then so is $\beta D \xrightarrow{\beta \alpha t} \beta D'$ for any $\beta > 0$ such that $\beta \cdot \alpha \leq 1$. (The converse holds as well). This means that runs and the configurations appearing in them can be shrunk down to very small values, hence enabling us to reduce the coverability problem for continuous VASS with zero-tests to its 1-bounded version.

Assuming that Lemma 3.5 is true, we shall now prove Theorem 3.4. Let $(\mathcal{M}, C, D)$ be an instance of the 1-bounded coverability problem for continuous VASS with zero-tests. To every counter x of $\mathcal{M}$, we add a *complementary counter* $\overline{x}$ and modify the transitions of $\mathcal{M}$ so that the sum of the values of x and $\overline{x}$ is 1 at any point. This can be done by modifying each transition t of $\mathcal{M}$ so that if t increments (resp. decrements) x by some value a, then t decrements (resp. increments) $\overline{x}$ by a and vice versa. Then, we replace each zero-test of $\mathcal{M}$ on the counter x, by a reset on the counter x. Call this new machine $\mathcal{M}'$. Note that $\mathcal{M}'$ is a continuous VASS with resets.

Let C' (resp. D') be the configuration of $\mathcal{M}'$ such that $C'(x) = C(x)$ and $C'(\overline{x}) = 1 - C(x)$ (resp. $D'(x) = D(x)$ and $D'(\overline{x}) = 1 - D(x)$) for every counter x of $\mathcal{M}$. Note that since $(\mathcal{M}, C, D)$ is an instance of the 1-bounded coverability problem, $C(x), D(x) \leq 1$ for every x and so $C'(\overline{x}), D'(\overline{x})$ are non-negative rationals.

If in any run of $\mathcal{M}'$, a counter x is reset when it already has the value 0, then this reset does not disturb the invariant that the sum of the values of x and $\overline{x}$ is 1. On the other hand, if a run of $\mathcal{M}'$ resets x when it has a positive value $\alpha > 0$, then this reset will make the sum of the values of x and $\overline{x}$ to be $1 - \alpha$. Since all the other transitions either reset a counter or maintain the sum of x and $\overline{x}$ to be a constant, it would follow that such a run can never cover the configuration D'. Hence, in a run which covers D' from C' in $\mathcal{M}'$, a counter x is reset only when it has the value 0. Since we have replaced zero-tests in $\mathcal{M}$ with resets in $\mathcal{M}'$, it then follows that any run that covers D from C in $\mathcal{M}$ can be uniquely mapped to a run that covers D' from C' and vice-versa. This then completes the proof of Theorem 3.4.

Stage 2: Reduction from resets to zero-row/zero-column matrices. We now show that affine classes that contain a matrix with a zero-row/column can simulate resets. To this end, let C be a non-negative class such that C contains a matrix $\mathbf{A}$ (of size $k \times k$ for some k) which either has a zero-row or a zero-column. Let j be the row or column of $\mathbf{A}$ which is completely zero. Note that

If j is a row then for any $\mathbf{u} \in \mathbb{Q}_+^k$, $\mathbf{A} \cdot \mathbf{u} = \mathbf{v}$

$$\text{where } \mathbf{v} \in \mathbb{Q}_+^k \text{ satisfies } \mathbf{v}(j) = 0 \quad (2)$$

and

$$\text{If } j \text{ is a column then for any } \lambda \in \mathbb{Q}_+, \mathbf{A} \cdot \lambda \mathbf{e}_j = \mathbf{0} \quad (3)$$

Equation 2 tells us that we can use the matrix to always reset the counter j. Equation 3 tells us that as long as all the counters apart from j on which we are applying $\mathbf{A}$ have the value 0, then we can reset the counter j. We can then exploit these equations to give a reduction from continuous VASS with resets to C-continuous VASS, similar in style to the proof of Theorem 3.1. For full details, we refer the reader to Section A.2 of the appendix.

3.3 Reachability for non-negative classes with weighted/overlapping edges

We now move on to our final undecidability result. Our main result of this subsection is that

Theorem 3.6. *The reachability problem for C-continuous VASS is undecidable if C is a non-negative class which contains a matrix with a weighted edge or two overlapping edges.*

The proof of this theorem will combine ideas from the previous two reductions. More specifically, we prove this theorem by giving a reduction from the 1-bounded reachability problem for continuous VASS with zero-tests. In this problem, we are given a continuous VASS with zero-tests $\mathcal{M}$ and two configurations C, D, and we have to check if C can reach D by a run in which all the counter values are at most 1 at any point. In Lemma 3.5, we showed that the 1-bounded *coverability* problem for continuous VASS with zero-tests is undecidable. By mimicking the transformation given

in Theorem 2.8, it then follows that the 1-bounded reachability problem for continuous VASS with zero-tests is also undecidable.

Proof of Theorem 3.6. Let C be a non-negative class. We can assume that C does not have any matrices with zero-rows/columns, as otherwise Theorem 3.3 applies. Let $\mathbf{A}$ be a matrix (of size $k \times k$ for some k) which either has a weighted edge or two overlapping edges. This implies that there is an index z such that $\sum_{1 \le i \le k} \mathbf{A}(i, z) > 1$. This then allows us to deduce the following equation.

$$\mathbf{A} \cdot \mathbf{u} = \begin{cases} \mathbf{0} & \text{if } \mathbf{u} = \mathbf{0} \\ \mathbf{v} & \text{with } \sum_{1 \le i \le k} \mathbf{v}(i) > \sum_{1 \le i \le k} \mathbf{u}(i) \text{ if } \mathbf{u}(z) > 0 \end{cases} \quad (4)$$

Intuitively, this equation gives us an ability to perform a zero-test on the zth counter. As in the reductions before, we can think of the zth counter as a primary counter and all the other $k - 1$ counters as dummy counters holding the value 0. In our reduction below, we will pair the zth counter with a complementary counter $\bar{z}$ such that any additive increase in the zth counter will be accompanied by an additive decrease in its complementary counter and vice-versa. This will ensure that the sum of the values of $z, \bar{z}$ and the $k - 1$ dummy counters will be 1 with respect to the additive updates. If at some point, the zth counter along with all the dummy counters have value 0, then multiplying by $\mathbf{A}$ has no effect. On the other hand, if the zth counter has a non-zero value, then multiplying by $\mathbf{A}$ has the effect of strictly increasing the sum of the values of these counters. Since, the overall sum of the values of these counters in the final configuration will also be 1, it follows that in any run between the initial and the final configuration, we can multiply our current valuation with the matrix $\mathbf{A}$ at some point only if the value of z at that point is 0, which enables us to do a zero-test. Hence, we can then give a reduction from the 1-bounded reachability problem for continuous VASS with zero-tests in a style similar to Theorem 3.1, which will prove the required undecidability result. For full details, we refer the reader to Section B of the appendix.

Note that this same reduction will *not* work for coverability, because in coverability it is permissible to have strictly higher values than the target configuration. □

Let us now look at the big picture for all of our undecidability results. We have shown all the undecidability results mentioned in Theorems 2.3 and 2.4. As for the reachability problem, the discussion given in Subsection 2.1 tells us that if C contains a matrix $\mathbf{A}$ that is not the permutation matrix, then $\mathbf{A}$ either has a negative entry, or has a zero-row/column, or has weighted/overlapping edges. It then follows that we have also proved all the undecidability results for the reachability problem mentioned in Theorem 2.2.

4 Decidability results and complexity upper bounds

We now prove all of our decidability results. We begin by showing decidability of state reachability for non-negative classes.

4.1 State-reachability for non-negative classes

Our main result of this subsection is that

Theorem 4.1. *The state-reachability problem for C-continuous VASS is in PSPACE if C contains only non-negative matrices.*

Let C be a class containing only non-negative matrices and let $\mathcal{M} = (Q, T)$ be a d-dimensional C-continuous VASS. For every counter x and every transition $t = (p, \mathbf{A}, \mathbf{b}, q)$ of $\mathcal{M}$, we say that

- x is additively incremented by t if $\mathbf{b}(x) > 0$.
- x is additively decremented by t if $\mathbf{b}(x) < 0$. In this case, we let $[\![t]\!]_x^-$ be the set $\{y : \mathbf{A}(x, y) > 0\}$.

The intuition behind additive increments/decrements is that if x is additively incremented/decremented by t, then the additive update of t increases/decreases the value of x. The intuition behind $[\![t]\!]_x^-$ is that, if x is additively decremented by t, then to fire t at some point, there must be some counter $y \in [\![t]\!]_x^-$ which has a non-zero value at that point. We will expand upon these intuitions in the upcoming paragraphs, but first, we state the main object of our algorithm, namely the *support abstraction*.

Corresponding to $\mathcal{M}$, we associate an edge-labelled graph $SU_{\mathcal{M}}$, called its support abstraction, as follows. The set of vertices of $SU_{\mathcal{M}}$ will be $Q \times 2^{\{1,\dots,d\}}$. Intuitively, each configuration $p(\mathbf{u})$ of $\mathcal{M}$ corresponds to the vertex $(p, [\![\mathbf{u}]\!])$ of $SU_{\mathcal{M}}$. The set of edges of $SU_{\mathcal{M}}$ (and its labels) are as follows. Suppose $t = (p, \mathbf{A}, \mathbf{b}, q)$ is a transition of $\mathcal{M}$. Then, there exists an edge between vertices (p, S) and (q, S') labelled by t if and only if for all counters x,

- If x is additively decremented by t, then $[\![t]\!]_x^- \cap S \ne \emptyset$
- If x is additively incremented by t, then $x \in S'$ and
- If $x \in S'$ then either x is additively incremented by t or $[\![t]\!]_x^- \cap S \ne \emptyset$.

The intuitive idea behind this edge relation is as follows: Suppose $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$ with $t = (p, \mathbf{A}, \mathbf{b}, q)$. Then $\mathbf{v} = \mathbf{A}\mathbf{u} + \alpha\mathbf{b}$. If x is additively decremented by t this means that $\alpha\mathbf{b}(x) < 0$, and so it must be the case that, $(\mathbf{A}\mathbf{u})(x) > 0$. Since $\mathbf{A}$ is a non-negative matrix, the latter condition is equivalent to $[\![t]\!]_x^- \cap [\![\mathbf{u}]\!] \ne \emptyset$. Furthermore, notice that if $\mathbf{b}(x) > 0$, then so is $\mathbf{v}(x)$ and if $\mathbf{v}(x) > 0$ then either $(\mathbf{A}\mathbf{u})(x) > 0$ or $\mathbf{b}(x) > 0$. This argument, along with the definition of edges in the support graph then allows us to conclude that

Soundness: Suppose $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$. Then $(p, [\![\mathbf{u}]\!]) \xrightarrow{t} (q, [\![\mathbf{v}]\!])$ is an edge in the support abstraction.

Furthermore by analysing the construction of the support graph, we can show that

Completeness: Suppose $(p, S) \xrightarrow{t} (q, S')$ is an edge in the support abstraction $SU_{\mathcal{M}}$ and suppose $p(\mathbf{u})$ is a configuration of $\mathcal{M}$ such that $S \subseteq [\![\mathbf{u}]\!]$. Then, there exists a configuration $q(\mathbf{v})$ and a fraction α such that $S' \subseteq [\![\mathbf{v}]\!]$ and $p(\mathbf{v}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$.

Combining these two properties along with a simple induction on the length of a run allows us to prove the following:

$p(\mathbf{u})$ can reach a configuration whose state is q in $\mathcal{M}$ if and only if $(p, [\![\mathbf{u}]\!])$ can reach some vertex of the form (q, S) in the graph $SU_{\mathcal{M}}$.

For more details behind the proof of correctness of these claims, we refer the reader to Section C.

Hence, to decide state reachability in $\mathcal{M}$, it suffices to explore the support abstraction graph. By standard techniques of constructing this graph on-the-fly in a non-deterministic manner, we get a non-deterministic polynomial space algorithm for deciding state reachability of $\mathcal{M}$. Applying Savitch's theorem then gives us a PSPACE algorithm for deciding state reachability.

4.2 Coverability for non-negative classes with only self-loop matrices

The main result of this subsection is that

THEOREM 4.2. *The coverability and state-reachability problems for C-continuous VASS is in NP if C is a non-negative class that only contains self-loop matrices.*

Note that by Theorem 2.8, we need only prove the above theorem for coverability. We will prove this by capturing the coverability relation of such a C-continuous VASS as a formula in existential linear rational arithmetic (ELRA). Here ELRA is the existential theory of the rationals equipped with the addition and order operations. Since satisfiability of this existential theory can be done in NP [21], Theorem 4.2 would then follow.

We capture the coverability relation in ELRA by adapting some techniques developed for continuous VASS. Let us fix a class C that contains only non-negative self-loop matrices and let us fix a d-dimensional C-continuous VASS $\mathcal{M}$ for the rest of this section. We first state a couple of basic facts regarding $\mathcal{M}$.

LEMMA 4.3. *The following statements are true for any firing sequence π and any fraction $\alpha \in (0, 1]$:*

(1) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, $p(\alpha\mathbf{u}) \xrightarrow{\alpha\pi} q(\alpha\mathbf{v})$.*
(2) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, for any $\mathbf{w} \in \mathbb{Q}_+^d$, $p(\mathbf{u} + \mathbf{w}) \xrightarrow{\pi} q(\mathbf{v}')$ where $\mathbf{v}' \geq \mathbf{v} + \mathbf{w}$.*
(3) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, $p(\mathbf{u}) \xrightarrow{\alpha\pi} q(\mathbf{v}')$ where $\mathbf{v}' \geq \alpha\mathbf{v} + (1 - \alpha)\mathbf{u}$.*

Indeed in all these cases, the proofs are immediate when π is a single step. The general case then follows by induction on the length of π. A formal proof can be found in Section D.1 of the appendix.

Now we move on to proving our main result, i.e., Theorem 4.2. To prove the result, first we introduce a notation. We say that an ELRA formula ϕ can be computed in NP, if there is a Turing machine running in non-deterministic polynomial time such that every accepting path of the machine computes a *disjunct* of the formula ϕ. Hence, if $\phi_1, \ldots, \phi_\ell$ are all of the produced formulas by all of the accepting paths, then $\phi = \vee_{i=1}^\ell \phi_i$. Notice that if we can compute ϕ in NP, then we can test for satisfiability of ϕ in non-deterministic polynomial time as well: Non-deterministically compute a disjunct ϕ_i of ϕ and then check for emptiness of ϕ_i, which can be done in NP [21].

Equipped with this notion, we now prove Theorem 4.2. For this, notice that it suffices to show the following lemma.

LEMMA 4.4. *Given a state p of $\mathcal{M}$, in NP, we can construct an ELRA formula ϕ_p such that $\phi_p(\mathbf{u}, \mathbf{v})$ is satisfiable if and only if $p(\mathbf{u})$ can cover $p(\mathbf{v})$.*

Indeed, by standard arguments on decomposing paths into steps and (non-simple) cycles [7], we can establish that a configuration $p(\mathbf{u})$ can cover a configuration $q(\mathbf{v})$ if and only if there is a *witness*, i.e., a sequence of configurations

$$p_1(\mathbf{u}_1), p_1(\mathbf{u}_1'), p_2(\mathbf{u}_2), p_2(\mathbf{u}_2'), \ldots, p_k(\mathbf{u}_k), p_k(\mathbf{u}_k')$$

such that each $p_i \neq p_j$ for $i \neq j$, $p_1(\mathbf{u}_1) = p(\mathbf{u})$, $p_k = q$, $\mathbf{u}_k' \geq \mathbf{v}$ and for each i, $p_i(\mathbf{u}_i)$ can cover $p_i(\mathbf{u}_i')$ and $p_i(\mathbf{u}_i') \rightarrow p_{i+1}(\mathbf{u}_{i+1})$. Since we can we can easily come up with a formula $\phi_{p,q}^s$ in ELRA such that $\phi_{p,q}^s(\mathbf{u}, \mathbf{v})$ is true if and only if $p(\mathbf{u}) \rightarrow q(\mathbf{v})$, i.e., $p(\mathbf{u})$ can reach $q(\mathbf{v})$ by a single step, Lemma 4.4 then allows us to solve coverability by reducing it to formulas in ELRA, which can be checked for satisfiabilty in NP [21]. More details can be found in Subsection D.2.

Proving Lemma 4.4. Hence, all that remains is to prove Lemma 4.4. To prove this, we introduce some definitions, then give a high-level overview of our proof strategy and then prove a number of preparatory results leading to the main result.

Recall that we have fixed a class C that contains only non-negative self-loop matrices and a d-dimensional C-continuous VASS $\mathcal{M}$ for the rest of this section. Given a transition $t = (p, \mathbf{A}, \mathbf{b}, q)$ of $\mathcal{M}$ and a counter x, we had already defined the notions of x being additively incremented by t, decremented by t and $[\![t]\!]_x^-$ in Subsection 4.1. Here, we introduce another notion called pumping. We say that a counter x can be pumped by transition t using y if either $y = x$ and $\mathbf{A}(x, y) > 1$ or $y \neq x$ and $\mathbf{A}(x, y) > 0$. We then let $[\![t]\!]_x^p = \{y : (y \neq x \wedge \mathbf{A}(x, y) > 0) \vee (y = x \wedge \mathbf{A}(x, y) > 1)\}$. If x can be pumped by t by using counter y, then since the matrix of t is a self-loop matrix, once we fire t, the new value of x will have the sum of the old values of x and y (and also the additive changes done by $\mathbf{b}$). Moreover, $[\![t]\!]_x^p$ contains all the counters with which t can pump x.

Given a run between $p(\mathbf{u})$ and $q(\mathbf{v})$ and a counter x, we say that counter x was *pumped* in this run if there exists a step $p'(\mathbf{u}') \xrightarrow{\alpha t} q'(\mathbf{v}')$ in this run such that x can be pumped by t using y and $\mathbf{u}'(\mathbf{y}) > 0$. As we shall see later on, counters that can be pumped in a run can be made to have arbitrarily high values subject to certain mild restrictions on the other counters.

With these notations set up, we now give a high-level overview behind the proof of Lemma 4.4. First, we prove a lemma called the Pumping Counters lemma which states that if a counter x is pumped in a run of the form $p(\mathbf{u}) \xrightarrow{*} p(\mathbf{v})$, then (subject to some technical constraints), we can get another run from $p(\mathbf{u})$ in which the final counter value of x can be made arbitrarily high and all the other counter values are at least as much as their respective values in $\mathbf{v}$. Hence, the Pumping Counters lemma allows us to *pump* the value of x whilst maintaining the values of the other counters as well. Hence, for the purposes of coverability, we can throw away all of these pumpable counters. As for the remaining counters, notice that the affine continuous VASS with respect to this set of counters behaves exactly like a continuous VASS, for which we know how to decide coverability. Hence, intuitively the overall idea is to then identify pumpable counters, remove them from $\mathcal{M}$ and only concentrate on the machine with the remaining counters which behaves like a continuous VASS and which we know how to solve. This reasoning is proved in another lemma which we call the

Coverability Characterization lemma. It then suffices to translate the conditions given in this lemma to a formula in ELRA, which we do in order to prove Lemma 4.4.

Having presented the high-level idea, we now begin the proof of Lemma 4.4 by first proving a number of preparatory results. Our first such result states that whenever there is a run from a state p back to itself, then by following the same path with different firing fractions, we can reach a configuration which is non-zero on every counter that was "affected" by the run. This proposition is similar to [7, Lemma 4.4].

PROPOSITION 4.5. *Suppose we have a run $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$ for some $p, \mathbf{u}, \mathbf{v}$ and π. Then, there is a firing sequence Rep-Half(π) such that*

- *If $\pi = \alpha_1 t_1, \alpha_2 t_2, \ldots, \alpha_\ell t_\ell$, then*

$$Rep\text{-}Half(\pi) = \alpha_1/2 t_1, \alpha_2/4 t_2, \ldots, \alpha_\ell/2^\ell t_\ell$$

- $p(\mathbf{u}) \xrightarrow{Rep\text{-}Half(\pi)} p(\mathbf{w})$ *for some $\mathbf{w}$ such that $[\![\mathbf{u}]\!] \subseteq [\![\mathbf{w}]\!]$, and*

- *If t is a transition in π and x is some counter, then*
  - *If x is additively incremented by t, then $\mathbf{w}(x) > 0$.*
  - *If x is additively decremented by t, then $[\![t]\!]_x^- \cap [\![\mathbf{w}]\!] \neq \emptyset$.*
  - *If x was pumped by t in the run $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$, then $[\![t]\!]_x^p \cap [\![\mathbf{w}]\!] \neq \emptyset$.*

The intuitive idea is that in the new path we repeatedly halve the firing fraction at each step. Rep-Half(π) now has the property that we leave a very small "residue" of every intermediate configuration that we visited by firing π from $p(\mathbf{u})$. More details can be found in Section D.3 of the appendix.

We now state a main technical lemma, namely the Pumping Counters Lemma. It states that whenever we have a run that pumps some counters, then, subject to some mild constraints, that run can be modified to arrive at a configuration in which all the pumped counters have an arbitrarily large value.

LEMMA 4.6 (PUMPING COUNTERS LEMMA). *Suppose $\rho : p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$ is a run. Let S be the set of counters such that $x \in S$ only if x is pumped by some transition t in ρ and $[\![t]\!]_x^p \cap [\![\mathbf{v}]\!] \neq \emptyset$. Then, for any $K > 0$, there exists a run π' such that*

- $\pi' = \pi/2 \cdot (\pi/2n)^n$ *for some n.*
- $p(\mathbf{u}) \xrightarrow{\pi'} p(\mathbf{w})$ *such that $\mathbf{w} \geq \mathbf{v}$ and*
- *For every $x \in S$, $\mathbf{w}(x) > K$.*

The intuitive idea is behind the proof is as follows: We first fire $\pi/2$ from $p(\mathbf{u})$. By Lemma 4.3 (3), this has the effect of reaching a configuration of the form $p(\mathbf{u}_1)$ with $\mathbf{u}_1 \geq \mathbf{v}/2 + \mathbf{u}/2$. By our assumption on S, this now means that for every counter $x \in S$ that was pumped by some t, there is some counter c_x such that $c_x \in [\![t]\!]_x^p \cap [\![\mathbf{u}_1]\!]$. Now, from $p(\mathbf{u}_1)$ we fire $\pi/2n$ to reach $p(\mathbf{u}_2)$. The effect of doing this is that now for every counter $x \in S$, the value of x in $\mathbf{u}_2$ is roughly at least $\mathbf{u}_1(x) + \mathbf{v}(c_x)/2$. Now, from $p(\mathbf{u}_2)$ we fire $\pi/2n$ again to reach $p(\mathbf{u}_3)$. Once again the effect of doing this is that for every counter $x \in S$, its new value is roughly at least $\mathbf{u}_2(x) + \mathbf{v}(c_x)/2$ which is roughly $\mathbf{u}_1(x) + 2\mathbf{v}(c_x)/2$. By iterating this procedure for n times, every counter $x \in S$ will roughly at least have the value $\mathbf{u}_1(x) + n\mathbf{v}(c_x)/2$. Choosing n to be large enough

will then make this value bigger than K. More details can be found in Subsection D.4 of the appendix.

We will now use the Pumping Counters lemma to prove a characterization of coverability in $\mathcal{M}$. To state this characterization we first need some notation. Given a firing sequence π and a configuration $p(\mathbf{u})$, we say that π is p-admissible from $p(\mathbf{u})$ if $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$ for some configuration $p(\mathbf{v})$. Given a set of counters X and a vector $\mathbf{v}$ we use $\mathbf{v}_X$ to denote the projection of $\mathbf{v}$ onto the counters X. Further, we let $\mathcal{M}_X$ be the $|X|$-dimensional *continuous VASS* obtained from $\mathcal{M}$, by converting each transition $t = (p, \mathbf{A}, \mathbf{b}, q)$ into a transition $t_X = (p, \mathbf{I}, \mathbf{b}_X, q)$ Given any firing sequence $\pi = \alpha_1 t_1, \ldots, \alpha_k t_k$ over $\mathcal{M}$, we will let π_X be the firing sequence $\pi_X = \alpha_1 t_{1_X}, \ldots, \alpha_k t_{k_X}$ in $\mathcal{M}_X$. Note that every firing sequence in $\mathcal{M}_X$ is of the form π_X for some firing sequence π in $\mathcal{M}$. With these notations set up, we can now state our characterization of coverability in $\mathcal{M}$.

LEMMA 4.7 (COVERABILITY CHARACTERIZATION LEMMA). $p(\mathbf{u})$ *can cover $p(\mathbf{v})$ in $\mathcal{M}$ if and only if there exists a set of counters X, a vector $\mathbf{w} \geq \mathbf{v}$ and firing sequences π', π_{fwd} over $\mathcal{M}$ such that*

- (1) $p(\mathbf{u}_X) \xrightarrow{\pi'_X} p(\mathbf{w}_X)$ *is a run in $\mathcal{M}_X$*
- (2) π_{fwd} *is p-admissible from $p(\mathbf{u})$ in $\mathcal{M}$*
- (3) *Every counter x that is **not in** X is pumped by the p-admissible run obtained from $p(\mathbf{u})$ by firing π_{fwd}*
- (4) $[\![\pi']\!] = [\![\pi_{fwd}]\!]$

If $p(\mathbf{u})$ can cover $p(\mathbf{v})$ in $\mathcal{M}$ by a run of the form $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{w})$ with $\mathbf{w} \geq \mathbf{v}$, then setting $\pi' = \pi_{\text{fwd}} = \pi$ and X to be the set of counters not pumped in π satisfies all the required conditions.

On the other hand, suppose we have $X, \mathbf{w}, \pi'$ and π_{fwd} satisfying all the conditions of the lemma. Intuitively, the counters that are in X are roughly the counters that are not pumped. For these counters, we can simply ignore the matrices and concentrate only on the underlying continuous VASS $\mathcal{M}_X$. By assumption, we have the run $p(\mathbf{u}_X) \xrightarrow{\pi'_X} p(\mathbf{w}_X)$ in $\mathcal{M}_X$. By using results about continuous VASS, we show that there is a small enough fraction λ and a configuration $p(\mathbf{z}')$ in $\mathcal{M}_X$ such that $p(\mathbf{u}_X) \xrightarrow{\lambda \text{Rep-Half}\left(\pi_{\text{fwd}_X}\right)} p(\mathbf{z}')$ and $p(\mathbf{z}')$ can reach $p(\mathbf{w}_X)$ by a firing sequence η_X in $\mathcal{M}_X$. This means that if in the original machine $\mathcal{M}$, we have a vector $\mathbf{s}$ such that $\mathbf{s}(x) \geq \mathbf{z}'(x)$ for every $x \in X$ and $\mathbf{s}(x)$ is very high for every $x \notin X$, then we can cover $p(\mathbf{w})$ from $p(\mathbf{s})$ in $\mathcal{M}$, by simply mimicking η_X in $\mathcal{M}$.

By then using the Pumping Counters lemma and the firing sequence π_{fwd}, we then show that such a vector $\mathbf{s}$ exists and also that $p(\mathbf{s})$ can be reached from $p(\mathbf{u})$ in $\mathcal{M}$. This means that to cover $p(\mathbf{v})$ from $p(\mathbf{u})$, we first move from $p(\mathbf{u})$ to $p(\mathbf{s})$ and then from there, cover $p(\mathbf{w})$ as mentioned in the previous paragraph. Since $\mathbf{w} \geq \mathbf{v}$, this would give the required run from $p(\mathbf{u})$. The formal details behind this proof can be found in Subsection D.5 in the appendix.

Hence, to embed the coverability relation as a formula $\phi_p(\mathbf{u}, \mathbf{v})$ of ELRA, it suffices to express the conditions given by the Coverability Characterization lemma. We will now give a non-deterministic polynomial-time algorithm which does that. Our algorithm will first guess a set of counters X and a set of transitions S. Then, it will write a formula of the form $\exists \mathbf{w} \phi_1(\mathbf{u}_X, \mathbf{w}_X) \wedge \phi_2(\mathbf{u})$. The formula $\phi_1(\mathbf{u}_X, \mathbf{w}_X)$ will express that there is a run from $\mathbf{u}_X$ to $\mathbf{w}_X$ in the continuous VASS $\mathcal{M}_X$ whose support is S. Thanks to results from

continuous VASS [7, Lemma 4.8], it is known how to construct $\phi_1(\mathbf{u}_X, \mathbf{w}_X)$ in polynomial-time. The formula $\phi_2(\mathbf{u})$ will express that there is a firing sequence π_{fwd} whose support is S, which is p-admissible from $p(\mathbf{u})$ in $\mathcal{M}$, and the run resulting from $p(\mathbf{u})$ and π_{fwd} pumps all the counters not in X, i.e., all the counters in the complement of X.

To construct ϕ_2, we first give a characterization of the existence of admissible runs which pump a given set of counters Y. To state this characterization, we need some notations. The graph of $\mathcal{M}$, denoted by $\mathcal{G}_{\mathcal{M}}$ is the graph whose nodes are the states of $\mathcal{M}$ and there is an edge between p and q if and only if there is a transition t such that $t = (p, \mathbf{A}, \mathbf{b}, q)$. In this case, we also let $in(t)$ be p, $out(t)$ be q and $\llbracket t \rrbracket^+ = \{x : \mathbf{b}(x) > 0\}$, i.e., $\llbracket t \rrbracket^+$ is the set of counters additively incremented by t. We then have the following lemma.

Lemma 4.8. *There exists a p-admissible run from $p(\mathbf{u})$ which uses exactly the transitions from S and which pumps all the counters in a set Y if and only if there exists an injection $f : S \to \mathbb{Q}_{>0}$ such that for every $t, t' \in S$,*

- *If $f(t)$ has the minimum value among all transitions in S, then $in(t) = p$.*
- *If $f(t) < f(t')$, then there exists a path in $\mathcal{G}_{\mathcal{M}}$ from $out(t)$ to $in(t')$ which uses only the transitions from the set $\{s : s \in S, f(s) < f(t')\}$.*
- *If $f(t)$ has the maximum value among all transitions in S, then there exists a path in $\mathcal{G}_{\mathcal{M}}$ from $out(t)$ to p which uses only the transitions from S.*
- *For every counter y additively decremented by t,*

$$\llbracket t \rrbracket_y^- \bigcap \left(\llbracket \mathbf{u} \rrbracket \cup \bigcup_{s \in S, f(s) < f(t)} \llbracket s \rrbracket^+ \right) \neq \emptyset$$

- *For every $y \in Y$, there exists a transition $r \in S$ such that*

$$\llbracket r \rrbracket_y^p \bigcap \left(\llbracket \mathbf{u} \rrbracket \cup \bigcup_{s \in S} \llbracket s \rrbracket^+ \right) \neq \emptyset$$

The intuition behind this lemma is that f characterizes the *first order appearance* of the transitions that appear in a p-admissible run, i.e., if $f(t) < f(t')$, then t appears before t' in the p-admissible run. The proof of this lemma is very similar to the one from [7, Lemma 4.7]. and can be found in Subsection D.6 of the appendix.

Hence to construct ϕ_2 (given p, S and the complement of X), it is sufficient to express these five conditions in ELRA. In [7, Lemma 4.8], it is shown how to express the first four conditions in ELRA. It is also easy to see that the fifth condition can also be expressed in ELRA . This then completes the proof of Lemma 4.4 and hence also Theorem 4.2.

4.3 Reachability for non-negative classes with only permutation matrices

We now show our final decidability result.

Theorem 4.9. *The reachability and coverability problems for C-continuous VASS are in NEXP if C contains only permutation matrices.*

By Theorem 2.8, it suffices to prove the above theorem for only the reachability problem. We prove this by giving a reduction to the reachability problem for continuous VASS. This reduction is exactly

the same as the one given in [11, Proposition 4.1], which gives a similar reduction in the case of affine VASS. The intuitive idea is that since C contains only permutation matrices, essentially each affine operation of $\mathcal{M}$ is only a renaming of the counters. Hence, at each point, we can keep track of the current renaming in the control states, which allows us to reduce the reachability problem for C-continuous VASS to reachability in (an exponentially) bigger continuous VASS, which is in NP. More formal details can be found in Section E of the appendix.

Let us now review all of our decidability results and see how they prove the decidability and complexity upper bound claims made in Section 2. Note that the NP upper bounds for the identity class are already known [7]. With this in mind, it is then easy to check that the three decidability (and upper bound) results proved in this section prove all of the decidability (and upper bound) claims in Theorems 2.2, 2.3, 2.4, 2.5, 2.6 and 2.7.

5 Complexity lower bounds

We now prove complexity lower bounds for almost all of the upper bounds that we proved in the previous section. First, note that since any class containing self-loop matrices generalizes the identity class, the coverability and state-reachability problems are NP-hard for self-loop classes. We now move on to the non-trivial lower bounds.

5.1 State-reachability for non-negative classes with zero-rows/columns

In this subsection, we show that

Theorem 5.1. *The state-reachability problem for C-continuous VASS is PSPACE-hard if C contains a non-negative matrix with a zero-row or a zero-column.*

We show that this theorem is true for continuous VASS with resets. The general case then follows because the reduction given for coverability in Stage 2 in Subsection 3.2 is also valid for state-reachability.

To prove that state-reachability for continuous VASS with resets is PSPACE-hard, we give a reduction from state-reachability for Boolean programs, for which state-reachability is PSPACE-hard [17]. Intuitively, a Boolean program is a finite-state automaton which has access to d Boolean variables (for some d), which can either be set to 0 or 1 and which can be tested for its current value. Given a Boolean program with d variables, we will construct a $2d$-continuous VASS with resets that will simulate the Boolean program. The intuitive idea is that each Boolean variable will be simulated by two counters. Exactly one of these two counters will have a non-zero value at some point. If the first counter (resp. second) has a non-zero value, then this corresponds to the Boolean variable being true (resp. false). Setting a Boolean variable to 0 or 1 can be done by resetting one of the counters and incrementing the other. Testing the current value of a Boolean variable is accomplished by trying to decrement the appropriate counter. For a formal proof, we refer the reader to Section F of the appendix.

5.2 Non-negative classes with only permutation matrices

In this subsection, we show that

THEOREM 5.2. *If C contains a non-trivial permutation matrix, then state-reachability is PSPACE-hard and reachability and coverability are NEXP-hard.*

To this end, let C contain a non-trivial permutation matrix $\mathbf{P}_\sigma$. Since $\mathbf{P}_\sigma$ is not the identity matrix, there exists a sequence of distinct indices $i_1, i_2, \ldots, i_\ell$ with $\ell \geq 2$ such that σ maps i_1 to i_2, i_2 to i_3 and so on and finally maps i_ℓ to i_1. Let $z = i_1$. Further, let $n \geq 2$ be the least number such that $\mathbf{P}_\sigma^n = \mathbf{I}$. Note that since $\mathbf{P}_\sigma$ is a permutation matrix, such a number must exist.

Now, suppose $\mathbf{u}$ is a vector such that $\mathbf{u}(j) = 0$ for every $j \neq z$. Then, $\mathbf{P}_\sigma \cdot \mathbf{u} = \mathbf{v}$ where $\mathbf{v}(j) = 0$ for every $j \neq \sigma(z)$. Similarly, suppose $\mathbf{u}$ is a vector such that $\mathbf{u}(j) = 0$ for every $j \neq \sigma(z)$. Then, $\mathbf{P}_\sigma^{n-1} \cdot \mathbf{u} = \mathbf{v}$ where $\mathbf{v}(j) = 0$ for every $j \neq z$. Intuitively, $\mathbf{P}_\sigma$ transfers the value of z to $\sigma(z)$ and does nothing to the other counters. Similarly, $\mathbf{P}_\sigma^{n-1}$ transfers the value of $\sigma(z)$ to z and does nothing to the other counters. Hence, just like in the previous subsection, we can use z and $\sigma(z)$ to encode a Boolean variable. This lets us reduce state-reachability for Boolean programs to state-reachability for C-continuous VASS, Further, the same reduction allows us to reduce *continuous Boolean programs* to C-continuous VASS, for which the reachability and coverability problems are NEXP-hard [4]. The formal proofs can be found in Section G of the appendix.

Finally, we note that the results proved in this subsection cover all the lower bounds mentioned in Theorems 2.6 and 2.7.

6 Conclusion

We have introduced the model of affine continuous VASS and studied the reachability, state-reachability and coverability problems for different classes of integer affine operations. We have shown a classification for decidability of reachability and state-reachability and an almost-complete classification for coverability. We have also complemented most of our decidability results with tight complexity bounds.

As part of future work, we would like to completely classify the decidability status of the coverability problem by solving it for the weighted/overlapping family. Extending the current approach for self-loop classes to this general case is challenging. This is because for the self-loop classes, we crucially exploited the fact that at each step, the matrix in the underlying transition does not allow the counter values to drop. This property is however lost in the general case, which indicates that different methods might be necessary to solve the general case.

Finally, it might be worth studying affine operations in which the matrices are allowed to have entries from the rationals. Many of our undecidability and decidability results already apply in this case as well; however, a complete classification of problems in this case is a possible direction for future work.

References

[1] Parosh Aziz Abdulla, Karlis Cerans, Bengt Jonsson, and Yih-Kuen Tsay. General decidability theorems for infinite-state systems. In *Proceedings, 11th Annual IEEE Symposium on Logic in Computer Science,*, pages 313–321, 1996. doi:10.1109/LICS.1996.561359.

[2] 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, December 14-18, 2020, BITS Pilani, K K Birla Goa Campus, Goa, India (Virtual Conference)*, volume 182 of *LIPIcs*, pages 37:1–37:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. URL: https://doi.org/10.4230/LIPIcs.FSTTCS.2020.37, doi:10.4230/LIPICS.FSTTCS.2020.37.

[3] 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. URL: https://doi.org/10.46298/lmcs-19(4:2)2023, doi:10.46298/LMCS-19(4:2)2023.

[4] A. R. Balasubramanian, Rupak Majumdar, Ramanathan S. Thinniyam, and Georg Zetzsche. Reachability in continuous pushdown VASS. *Proc. ACM Program. Lang.*, 8(POPL):90–114, 2024. doi:10.1145/3633279.

[5] Michael Blondin. The ABCs of Petri net reachability relaxations. *ACM SIGLOG News*, 7(3):29–43, 2020. doi:10.1145/3436980.3436984.

[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 *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12, Reykjavik, Iceland, June 2017. IEEE. doi:10.1109/LICS.2017.8005068.

[8] Michael Blondin, Christoph Haase, Filip Mazowiecki, and Mikhail A. Raskin. Affine extensions of integer vector addition systems with states. *Log. Methods Comput. Sci.*, 17(3), 2021. URL: https://doi.org/10.46298/lmcs-17(3:1)2021, doi:10.46298/LMCS-17(3:1)2021.

[9] Michael Blondin, Christoph Haase, and Philip Offtermatt. Directed reachability for infinite-state systems. In Jan Friso Groote and Kim Guldstrand Larsen, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 27th International Conference, TACAS 2021, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2021, Luxembourg City, Luxembourg, March 27 - April 1, 2021, Proceedings, Part II*, volume 12652 of *Lecture Notes in Computer Science*, pages 3–23. Springer, 2021. doi:10.1007/978-3-030-72013-1_1.

[10] Michael Blondin, Filip Mazowiecki, and Philip Offtermatt. Verifying generalised and structural soundness of workflow nets via relaxations. In Sharon Shoham and Yakir Vizel, editors, *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part II*, volume 13372 of *Lecture Notes in Computer Science*, pages 468–489. Springer, 2022. doi:10.1007/978-3-031-13188-2_23.

[11] Michael Blondin and Mikhail A. Raskin. The complexity of reachability in affine vector addition systems with states. *Log. Methods Comput. Sci.*, 17(3), 2021. URL: https://doi.org/10.46298/lmcs-17(3:3)2021, doi:10.46298/LMCS-17(3:3)2021.

[12] Wojciech Czerwiński and Łukasz Orlikowski. Reachability in Vector Addition Systems is Ackermann-complete. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1229–1240, February 2022. doi:10.1109/FOCS52979.2021.00120.

[13] Giorgio Delzanno, Jean-François Raskin, and Laurent Van Begin. Towards the automated verification of multithreaded Java programs. In Joost-Pieter Katoen and Perdita Stevens, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 8th International Conference, TACAS 2002, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2002, Grenoble, France, April 8-12, 2002, Proceedings*, volume 2280 of *Lecture Notes in Computer Science*, pages 173–187. Springer, 2002. doi:10.1007/3-540-46002-0_13.

[14] Javier Esparza, Pierre Ganty, Jérôme Leroux, and Rupak Majumdar. Verification of population protocols. *Acta Informatica*, 54(2):191–215, 2017. URL: https://doi.org/10.1007/s00236-016-0272-3, doi:10.1007/S00236-016-0272-3.

[15] Alain Finkel and Philippe Schnoebelen. Well-structured transition systems everywhere! *Theor. Comput. Sci.*, 256(1-2):63–92, 2001. doi:10.1016/S0304-3975(00)00102-X.

[16] 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.

[17] Patrice Godefroid and Mihalis Yannakakis. Analysis of Boolean programs. In Nir Piterman and Scott A. Smolka, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings*, volume 7795 of *Lecture Notes in Computer Science*, pages 214–229. Springer, 2013. doi:10.1007/978-3-642-36742-7_16.

[18] Alexander Kaiser, Daniel Kroening, and Thomas Wahl. A widening approach to multithreaded program verification. *ACM Trans. Program. Lang. Syst.*, 36(4):14:1–14:29, 2014. doi:10.1145/2629608.

[19] 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.

[20] Jake Silverman and Zachary Kincaid. Loop summarization with rational vector addition systems. In Isil Dillig and Serdar Tasiran, editors, *Computer Aided Verification - 31st International Conference, CAV 2019, New York City, NY, USA, July 15-18, 2019, Proceedings, Part II*, volume 11562 of *Lecture Notes in Computer Science*, pages 97–115. Springer, 2019. doi:10.1007/978-3-030-25543-5_7.

[21] Eduardo D. Sontag. Real addition and the polynomial hierarchy. *Information Processing Letters*, 20(3):115–120, April 1985. doi:10.1016/0020-0190(85)90076-6.

[22] 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.

A   Proofs of Subsection 3.2

A.1   Proof of Lemma 3.5

LEMMA 3.5. *The 1-bounded coverability problem is undecidable for continuous VASS with zero-tests.*

PROOF. We prove this lemma by giving a series of reductions that starts at the coverability problem for continuous VASS with zero-tests and ends at the 1-bounded coverability problem for continuous VASS with zero-tests.

Let $\mathcal{M}$ be a continuous VASS with zero-tests. By definition of steps in $\mathcal{M}$, note that the following is true:

> SCALING PROPERTY: Let D and D' be configurations of $\mathcal{M}$. For any transition t, any $\alpha \in (0, 1]$ and any $\beta > 0$ such that $\beta \cdot \alpha \leq 1$, we have $D \xrightarrow{\alpha t} D'$ if and only if $\beta D \xrightarrow{\beta \cdot \alpha t} \beta D'$. Further, for any zero-test t, we have $D \xrightarrow{t} D'$ if and only if for any $\beta > 0$, we have $\beta D \xrightarrow{t} \beta D'$.

For some β, we say that all the appearing fractions in a run are at most β if all the fractions appearing in the firing sequence of that run are at most β. Note that repeated applications of the Scaling property imply that if we have a run from D to D' in $\mathcal{M}$ of the form $D_0 = D \to D_1 \to \ldots \to D' = D_k$, then by choosing β to be so small that $\beta \cdot D_i(j) \leq 1$ for every $0 \leq i \leq k$ and every counter j, we can ensure that there is a 1-bounded run from βD to $\beta D'$ in which all the appearing fractions are at most β. Similarly, the converse can also be shown by repeatedly applying the Scaling property. This means that it is undecidable to check given a triple $(\mathcal{M}, C, C')$, whether there exists some $\beta \in (0, 1]$ such that βC can reach (resp. cover) $\beta C'$ by a 1-bounded run in which all the appearing fractions are at most β.

Now, to $\mathcal{M}$, add two more counters z and $\tilde{z}$. Replace each transition $t = (p, \Delta, q)$ of $\mathcal{M}$ by two transitions - the first transition goes from p to a fresh state s_t, increases z by 1, decreases $\tilde{z}$ by 1 and does exactly what t does to the remaining counters; the second transition goes from s_t to q, decreases z by 1 and increases $\tilde{z}$ by 1. Intuitively, the value in z at any point controls the fraction with which t can be fired; if the value in z is β at some point, then t can only be fired with fraction at most β. Call this new machine $\overline{\mathcal{M}}$. For any configuration D of $\mathcal{M}$ and any α, let $\overline{D}_\alpha$ be a configuration of $\overline{\mathcal{M}}$ such that $\overline{D}_\alpha(x) = D(x)$ for every counter x of $\mathcal{M}$, $\overline{D}_\alpha(z) = \alpha$ and $\overline{D}_\alpha(\overline{z}) = 0$. The following claim is easy to prove by induction:

> There exists $\beta \in (0, 1]$ such that βC can reach (resp. cover) $\beta C'$ by an 1-bounded run in which all appearing fractions are at most β in $\mathcal{M}$ iff there exists $\beta \in (0, 1]$ such that $\overline{\beta C}_\beta$ can reach (resp. cover) $\overline{\beta C'}_\beta$ by an 1-bounded run in in $\overline{\mathcal{M}}$.

Now, let the states of the configurations C, C' be p and q respectively. To $\overline{\mathcal{M}}$, we add two new states p_i and q_f and a fresh counter st. From p_i we add a transition to p which increments each counter x of $\mathcal{M}$ by $C(x)$, counter z by 1 and decrements counter st by 1. Similarly, from q we add a transition to q_f which decrements each counter x of $\mathcal{M}$ by $C'(x)$, counter z by 1 and increments counter

st by 1. Call this new machine $\mathcal{M}_1$ and let C_1 and C_1' be the configurations of $\mathcal{M}_1$ whose counter values are all 0, except in the counter st where they have the value 1 and whose states are q_i and q_f respectively. The idea is that the transition from p_i to p non-deterministically guesses a fraction β, sets every counter x of $\mathcal{M}$ to $\beta C(x)$, sets counter z to β and sets st to $1 - \beta$. From there it simply mimics $\overline{\mathcal{M}}$ until it reaches q. Finally from q, it can fire the transition to q_f, which decrements every counter x of $\mathcal{M}$ by $\beta C'(x)$, decrements z by β and increments st by β. If this final transition chooses to be fired with some fraction other than β, then either the value of st will be less than 1 or the value of z will becomes less than 0. With this in mind, the following is then easy to observe:

> There exists $\beta \in (0, 1]$ such that $\overline{\beta C}_\beta$ can reach
> (resp. cover) $\overline{\beta C'}_\beta$ by an 1-bounded run in $\overline{\mathcal{M}}$ if
> and only if C_1 can reach (resp. cover) C_1' by an 1-
> bounded run in $\mathcal{M}_1$.

This completes the reduction and proves the lemma. □

A.2 Proof of Stage 2: Reduction from resets to zero-row/zero-column matrices.

We now show that affine classes that contain a matrix with a zero-row/column can simulate resets. To this end, let C be a non-negative class such that C contains a matrix $\mathbf{A}$ (of size $k \times k$ for some k) which either has a zero-row or a zero-column. Let j be the row or column of $\mathbf{A}$ which is completely zero. Note that

If j is a row then for any $\mathbf{u} \in \mathbb{Q}_+^k$, $\mathbf{A} \cdot \mathbf{u} = \mathbf{v}$
$$\text{where } \mathbf{v} \in \mathbb{Q}_+^k \text{ satisfies } \mathbf{v}(j) = 0 \quad (5)$$

and

$$\text{If } j \text{ is a column then for any } \lambda \in \mathbb{Q}_+, \mathbf{A} \cdot \lambda \mathbf{e}_j = \mathbf{0} \quad (6)$$

Equation 5 tells us that we can use the matrix to always reset the counter j. Equation 6 tells us that as long as all the counters apart from j on which we are applying $\mathbf{A}$ have the value 0, then we can reset the counter j. We will exploit these equations now to give a reduction from continuous VASS with resets to C-continuous VASS.

Let $\mathcal{M} = (Q, T)$ be a d-continuous VASS with resets for some d. For the purposes of the reduction, we can assume that in each transition, $\mathcal{M}$ resets at most one counter and if a transition does indeed reset some counter, then the additive update of that transition is $\mathbf{0}$. We can assume this since the undecidability proof given in the previous stage already holds for continuous VASS with resets with this restriction. Hence, each transition $t \in T$ is either of the form $t = (p, \mathbf{I}_d, \mathbf{b}, q)$ or $t = (p, \mathbf{R}^{(d,i)}, \mathbf{0}, q)$.

From $\mathcal{M}$, we will create a $n = dk$ dimensional C-continuous VASS $\mathcal{M}'$. As in the reduction of Theorem 3.1, the counters $x_1 = j, x_2 = j + k, \ldots, x_d = j + (d-1)k$ of $\mathcal{M}'$ will be called the primary counters and will simulate the counters $1, 2, \ldots, d$ of $\mathcal{M}$ respectively. The remaining counters of $\mathcal{M}'$ will be called dummy counters and they will either always have the value 0 (if $\mathbf{A}$ is a zero-column matrix) or their values will in no way affect the values of the primary counters (if $\mathbf{A}$ is a zero-row matrix).

Given a vector $\mathbf{u} \in \mathbb{Q}^d$, we define $\text{ext}(\mathbf{u}) \in \mathbb{Q}^n$ as the vector which is 0 everywhere, except in counters $x_1, x_2, \ldots, x_d$ where the values are respectively $\mathbf{u}(1), \mathbf{u}(2), \ldots, \mathbf{u}(d)$. Furthermore,

- If the jth row of $\mathbf{A}$ is zero, then we set $S(\mathbf{u}) = \{\mathbf{v} \in \mathbb{Q}_+^n : \mathbf{v}(x_i) = \mathbf{u}(i) \text{ for all primary counters } x_i\}$.
- If the jth column of $\mathbf{A}$ is zero, then we set $S(\mathbf{u}) = \{\text{ext}(\mathbf{u})\}$

Intuitively, if the jth row is zero, then $S(\mathbf{u})$ contains any vector in $\mathbb{Q}_+^n$ whose primary counter values coincide with the counter values of $\mathbf{u}$. On the other hand, if the jth column is zero, then $S(\mathbf{u})$ contains the unique vector in $\mathbb{Q}_+^n$ whose primary counter values coincide with the values of $\mathbf{u}$ and whose dummy values are 0.

The set of states of $\mathcal{M}'$ will be the same as Q. Each transition $t \in T$ of $\mathcal{M}$ will have a corresponding transition t' in $\mathcal{M}'$. Before we state this transition t', we will state two properties that will be satisfied by t'.

> PROPERTY 1: If $p(\mathbf{u}') \xrightarrow{\alpha t'} q(\mathbf{v}')$ is a step in $\mathcal{M}'$,
> where $\mathbf{u}' \in S(\mathbf{u})$ for some $\mathbf{u}$ then there exists $\mathbf{v}$ such
> that $\mathbf{v}' \in S(\mathbf{v})$.

> PROPERTY 2: For any configurations $p(\mathbf{u}), q(\mathbf{v})$, we
> have that $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$ if and only
> if for every $\mathbf{u}' \in S(\mathbf{u})$, there exists $\mathbf{v}' \in S(\mathbf{v})$ such
> that $p(\mathbf{u}') \xrightarrow{\alpha t} q(\mathbf{v}')$.

We now proceed to define the transition t' in $\mathcal{M}'$ that uniquely corresponds to the transition t in $\mathcal{M}$. Suppose $t = (p, \mathbf{I}_d, \mathbf{b}, q)$ for some states p, q and some vector $\mathbf{b}$. Then we set $t' = (p, \mathbf{I}_n, \text{ext}(\mathbf{b}), q)$. It is easily seen that t' satisfies both the properties.

Suppose $t = (p, \mathbf{R}^{(d,i)}, \mathbf{0}, q)$ for some states p, q and some counter i. Then we set $t' = (p, \text{Apply}_n(\mathbf{A}, (i-1)k+1)), \mathbf{0}, q)$. Intuitively we are applying $\mathbf{A}$ to the counters $(i-1)k+1, \ldots, (i-1)k + j - 1, (i-1)k + j = x_i, (i-1)k + j + 1, \ldots, ik$. We now show that t' satisfies Properties 1 and 2. First note that if $\mathbf{A}$ is a zero-row matrix, then any vector $\mathbf{w}' \in \mathbb{Q}_+^n$ is in $S(\mathbf{w})$ where $\mathbf{w}$ is obtained from $\mathbf{w}'$ by projecting it onto its primary counters. Hence Property 1 is trivially true for t' when $\mathbf{A}$ is a zero-row matrix. On the other hand, suppose $\mathbf{A}$ is a zero-column matrix. In this case, $S(\mathbf{w}) = \{\text{ext}(\mathbf{w})\}$. Hence, to prove Property 1, it simply suffices to show that if the value of the dummy counters was 0 before applying t', then they will remain 0 even after applying t'. By definition of the Application operation, $\text{Apply}_n(\mathbf{A}, (i-1)k + 1)$ cannot change the value of any counters in the range $\{1, 2, \ldots, (i-1)k\} \cup \{ik + 1, \cdots, dk\}$. Furthermore, by equation 6 it follows that if all the dummy counters in the range $\{(i-1)k+1, \ldots, ik\}$ have value 0 before multiplying by $\mathbf{A}$, then they will also have the value 0 afterwards. Hence, this proves Property 1 for t'.

Let us now prove Property 2, first in the case when $\mathbf{A}$ is a zero-row matrix. Suppose $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$. Then $\mathbf{v}$ is the same as $\mathbf{u}$, except that its value in the counter i is 0. Let $\mathbf{u}' \in S(\mathbf{u})$. The definition of the Application operation along with equation 5 then means that from $p(\mathbf{u}')$ we can fire the transition t' with fraction α to reach a configuration $q(\mathbf{v}')$ where $\mathbf{v}' \in \mathbb{Q}_+^n$ is the same as $\mathbf{u}'$ in all the counters in the range $\{1, 2, \ldots, (i-1)k\} \cup \{ik + 1, \ldots, dk\}$ and furthermore satisfies $\mathbf{v}'(x_i) = 0$. Hence, $\mathbf{v}' \in S(\mathbf{v})$. Similarly, suppose for every $\mathbf{u}' \in S(\mathbf{u})$, there exists $\mathbf{v}' \in S(\mathbf{v})$ such that $p(\mathbf{u}') \xrightarrow{\alpha t'} q(\mathbf{v}')$. Take $\mathbf{u}' = \text{ext}(\mathbf{u})$ and let $p(\mathbf{u}') \xrightarrow{\alpha t'} q(\mathbf{v}')$. By

definition of the Application operation along with equation 5, this means that the primary counters of $\mathbf{v}'$ have the same values as that of the primary counters of $\mathbf{u}'$, except that $\mathbf{v}'(x_i) = 0$. This means that the counter values of $\mathbf{u}$ and $\mathbf{v}$ are the same, except that there is a possible change at counter i, where $\mathbf{v}(i) = 0$. Hence, we have that $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$. This proves Property 2 in the case when $\mathbf{A}$ is a zero-row matrix. The case when $\mathbf{A}$ is a zero-column matrix is very similar and uses Equation 6 instead of 5.

By using induction along with Properties 1 and 2, it is easy to see that a configuration $p(\mathbf{u})$ can cover a configuration $q(\mathbf{v})$ in $\mathcal{M}$ if and only if the configuration $p(\text{ext}(\mathbf{u}))$ can cover $q(\text{ext}(\mathbf{v}))$ in $\mathcal{M}'$. This completes the proof of correctness of the reduction and proves Theorem 3.3.

B Proof of Theorem 3.6

THEOREM 3.6. *The reachability problem for C-continuous VASS is undecidable if C is a non-negative class which contains a matrix with a weighted edge or two overlapping edges.*

Let C be a non-negative class. We can assume that C does not have any matrices with zero-rows/columns, as otherwise Theorem 3.3 applies. Let $\mathbf{A}$ be a matrix (of size $k \times k$ for some k) which either has a weighted edge or two overlapping edges. This implies that there is an index z such that $\sum_{1 \le i \le k} \mathbf{A}(i, z) > 1$. This then allows us to deduce the following equation.

$$\mathbf{A} \cdot \mathbf{u} = \begin{cases} \mathbf{0} & \text{if } \mathbf{u} = \mathbf{0} \\ \mathbf{v} & \text{with } \sum_{1 \le i \le k} \mathbf{v}(i) > \sum_{1 \le i \le k} \mathbf{u}(i) \text{ if } \mathbf{u}(z) > 0 \end{cases} \tag{7}$$

Indeed, the first part of the equation is clear. For the second part, first note that since $\mathbf{A}$ is not a zero-column matrix, for each index i, there is some index $\sigma(i)$ such that $\mathbf{A}(\sigma(i), i) \ge 1$. Hence, if $\mathbf{v} = \mathbf{A}\mathbf{u}$ then

$$\begin{aligned} \sum_{1 \le i \le k} \mathbf{v}(i) &= \sum_{1 \le i \le k} \sum_{1 \le j \le k} \mathbf{A}(i, j)\mathbf{u}(j) \\ &= \sum_{1 \le j \le k} \sum_{1 \le i \le k} \mathbf{A}(i, j)\mathbf{u}(j) \\ &\ge \sum_{1 \le j \le k, j \ne z} \mathbf{A}(\sigma(j), j)\mathbf{u}(j) + \sum_{1 \le i \le k} \mathbf{A}(i, z)\mathbf{u}(z) \\ &> \sum_{1 \le j \le k} \mathbf{u}(j) \quad \text{if } \mathbf{u}(z) > 0 \end{aligned}$$

Also note that the same argument tells us

$$\mathbf{A} \cdot \mathbf{u} = \mathbf{v} \quad \text{with } \sum_{1 \le i \le k} \mathbf{v}(i) \ge \sum_{1 \le i \le k} \mathbf{u}(i) \tag{8}$$

Let us now prove the undecidability result by giving a reduction from the 1-bounded reachability problem for continuous VASS with zero-tests. To this end, let $(\mathcal{M}, C, C')$ be an instance of the 1-bounded reachability problem for continuous VASS with zero-tests where $\mathcal{M}$ is d-dimensional for some d. From $\mathcal{M}$, we will create a $n = d(k + 1)$-C-continuous VASS $\mathcal{M}'$. Of the $d(k + 1)$ counters of $\mathcal{M}'$, the counters $x_1 = z, x_2 = z + (k + 1), \ldots, x_d = z + (d-1)(k+1)$ will be called the primary counters of $\mathcal{M}'$ and they simulate the counters $1, 2, \ldots, d$ of $\mathcal{M}$ respectively. The counters $\overline{x_1} = k + 1, \overline{x_2} = 2(k + 1), \ldots, \overline{x_d} = d(k + 1)$ will be called the

complementary counters of $x_1, x_2, \ldots, x_d$ respectively. Intuitively, each complementary counter $\overline{x_i}$ will hold 1 minus the value of the counter x_i at any point in a run from the initial configuration to the final configuration. Finally, the remaining counters of $\mathcal{M}'$ will be called *dummy counters* which are always supposed to have the value 0 in a run from the initial configuration to the final configuration.

Given a vector $\delta \in \mathbb{Q}^d$, let $\text{ext}(\delta) \in \mathbb{Q}^n$ be the vector which is 0 everywhere, except in co-ordinates $x_1, x_2, \ldots, x_d$ where the values are $\delta(1), \delta(2), \ldots, \delta(d)$ respectively and in the co-ordinates $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_d}$ where the values are $-\delta(1), -\delta(2), \ldots, -\delta(d)$ respectively. Further, given a vector $\mathbf{u} \in \mathbb{Q}^d$, let $S(\mathbf{u}) \in \mathbb{Q}^n$ be the vector which is 0 everywhere except in co-ordinates $x_1, x_2, \ldots, x_d$ where it has the values $\mathbf{u}(1), \ldots, \mathbf{u}(d)$ and in co-ordinates $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_d}$ where it has the values $1 - \mathbf{u}(1), 1 - \mathbf{u}(2), \ldots, 1 - \mathbf{u}(d)$. With this notation set up, we are ready to state the desired reduction.

The set of states of $\mathcal{M}'$ will be the same as Q. We say that a configuration $p(\mathbf{u})$ of $\mathcal{M}'$ is good if the values of the dummy counters of $\mathbf{u}$ are all zero and for each primary counter x_i, $\mathbf{u}(x_i) + \mathbf{u}(\overline{x_i}) = 1$. Note that $p(\mathbf{u})$ is good if and only if $\mathbf{u} = S(\mathbf{v})$ for some $\mathbf{v} \in \mathbb{Q}_+^d$. On the other hand, $p(\mathbf{u})$ will be called bad if for some i, the sum of the values of the counters in the range $\{(i-1)(k+1)+1, \ldots, i(k+1)\}$ is strictly bigger than 1. Finally an undesirable configuration is one which is neither good nor bad. Our transitions will ensure that starting from a good configuration, no undesirable configuration can be reached. Further, they will also ensure that starting from a good configuration we can reach another good configuration by a run only if the run correctly simulated the machine $\mathcal{M}$. This will then allow us to reduce the 1-bounded reachability problem for $\mathcal{M}$ to the reachability problem for $\mathcal{M}'$.

We will now state the transitions of $\mathcal{M}'$. For each $t \in T \cup T_{=0}$ of $\mathcal{M}$, $\mathcal{M}'$ will have a corresponding transition t'. Suppose $t = (p, \Delta, q) \in T$ is a transition of $\mathcal{M}$. Corresponding to t, we have the transition $t' = (p, \mathbf{I}_n, \text{ext}(\Delta), q)$ in $\mathcal{M}'$. Intuitively, t' updates the primary counters exactly in the way t updates its counters, and it updates the complementary counters so that the sum of each primary counter and its complementary counter remain the same and does not update the dummy counters at all. Note that t' satisfies the following properties, whose proofs follow immediately from the definition of t' and good and bad configurations.

PROPERTY 1A): Suppose $C \xrightarrow{\alpha t'} D$ is a step in $\mathcal{M}'$. Then D is good (resp. bad) iff C is good (resp. bad).

PROPERTY 1B): Suppose $p(\mathbf{u})$ and $q(\mathbf{v})$ are configurations of $\mathcal{M}$ such that $0 \le \mathbf{u}(i), \mathbf{v}(i) \le 1$ for every counter i. Then $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$ if and only if $p(S(\mathbf{u})) \xrightarrow{\alpha t'} q(S(\mathbf{v}))$ is a step in $\mathcal{M}'$.

Now, suppose $t = (p, i, q) \in \delta_{=0}$ is a zero-test of $\mathcal{M}$. Corresponding to t, we have the transition $t' = (p, \text{Apply}_n(\mathbf{A}, (i - 1)(k + 1) + 1), \mathbf{0}, q)$ in $\mathcal{M}'$. Intuitively, t' does not update any counter apart from the ones in the range $\{(i-1)(k+1)+1, \ldots, i(k+1)-1\}$. Within this range, it updates the counters by multiplying them with the matrix $\mathbf{A}$. Note that by equations 7 and 8, t' satisfies the following invariants.

PROPERTY 2A : Suppose $p(\mathbf{u}) \xrightarrow{\alpha t'} q(\mathbf{v})$ is a step in $\mathcal{M}'$. If $p(\mathbf{u})$ is good and $\mathbf{u}(x_i) = 0$ then $q(\mathbf{v})$ is

good. If $p(\mathbf{u})$ is good and $\mathbf{u}(x_i) > 0$ or $p(\mathbf{u})$ is bad, then $q(\mathbf{v})$ is bad.

PROPERTY 2B: Suppose $p(\mathbf{u})$ and $q(\mathbf{v})$ are configurations of $\mathcal{M}$ such that $0 \leq \mathbf{u}(i), \mathbf{v}(i) \leq 1$ for every co-ordinate i. Then $p(\mathbf{u}) \xrightarrow{t} q(\mathbf{v})$ is a step in $\mathcal{M}$ if and only if $p(S(\mathbf{u})) \xrightarrow{\alpha t'} q(S(\mathbf{v}))$ is a step in $\mathcal{M}'$ for any α'.

This finishes the construction of $\mathcal{M}'$. By Properties 1b and 2b it follows that there is a 1-bounded run between C and C' in $\mathcal{M}$ if and only if there is a run between $S(C)$ and $S(C')$ in which all the configurations are good. By Properties 1a and 1b, it follows that no run from $S(C)$ to $S(C')$ can have a configuration which is bad or undesirable. This then proves that C can reach C' by a 1-bounded run in $\mathcal{M}$ if and only if $S(C)$ can reach $S(C')$ in $\mathcal{M}'$, which completes the proof of the theorem.

C   Proofs of Subsection 4.1

First we show the following Soundness property.

> SOUNDNESS: Suppose $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$. Then $(p, [\![\mathbf{u}]\!]) \xrightarrow{t} (q, [\![\mathbf{v}]\!])$ is an edge in the support abstraction.

Indeed, suppose $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$ with $t = (p, \mathbf{A}, \mathbf{b}, q)$. Then $\mathbf{v} = \mathbf{A}\mathbf{u} + \alpha\mathbf{b}$. If x is additively decremented by t this means that $\alpha\mathbf{b}(x) < 0$, and so it must be the case that, $(\mathbf{A}\mathbf{u})(x) > 0$. Since $\mathbf{A}$ is a non-negative matrix, the latter condition is equivalent to $[\![t]\!]_x^- \cap [\![\mathbf{u}]\!] \neq \emptyset$. Also, if x is additively incremented by t this means that $x \in [\![\mathbf{v}]\!]$. Finally, if $\mathbf{v}(x) > 0$ then either $(\mathbf{A}\mathbf{u})(x) > 0$ or $\mathbf{b}(x) > 0$, which is equivalent to saying that if $x \in [\![\mathbf{v}]\!]$ then either $[\![t]\!]_x^- \cap S \neq \emptyset$ or x is additively incremented by t. By the construction of the support abstraction graph, it follows that the Soundness property is true.

Now we prove the following Completeness property.

> COMPLETENESS: Suppose $(p, S) \xrightarrow{t} (q, S')$ is an edge in the support abstraction $SU_{\mathcal{M}}$ and suppose $p(\mathbf{u})$ is a configuration of $\mathcal{M}$ such that $S \subseteq [\![\mathbf{u}]\!]$. Then, there exists a configuration $q(\mathbf{v})$ and a fraction α such that $S' \subseteq [\![\mathbf{v}]\!]$ and $p(\mathbf{v}) \xrightarrow{\alpha t} q(\mathbf{v})$ is a step in $\mathcal{M}$.

To this end, let $t = (p, \mathbf{A}, \mathbf{b}, q)$ and let α be a fraction in $(0, 1]$ such that $\alpha < (\mathbf{A}\mathbf{u})(x)/|\mathbf{b}(x)|$ for every counter x that is additively decremented by t, i.e., whenever $\mathbf{b}(x) < 0$.

Note that whenever $\mathbf{b}(x) < 0$, by definition of edges in $SU_{\mathcal{M}}$, there must be y such that $y \in S \subseteq [\![\mathbf{u}]\!]$ and $\mathbf{A}(x, y) > 0$. Since, $\mathbf{A}$ is a non-negative matrix, this means that $(\mathbf{A}\mathbf{u})(x) > 0$ and so it is always possible to choose such a fraction α.

Let $\mathbf{v} = \mathbf{A}\mathbf{u} + \alpha\mathbf{b}$. We claim that $\mathbf{v}$ is indeed a non-negative vector. For the sake of contradiction, suppose $\mathbf{v}(x) < 0$ for some x. This must mean that $(\mathbf{A}\mathbf{u})(x) < -\alpha\mathbf{b}(x)$. Since $\mathbf{A}$ is a non-negative matrix, $(\mathbf{A}\mathbf{u})(x) \geq 0$ and so $\mathbf{b}(x)$ must be negative. Hence $-\alpha \cdot \mathbf{b}(x) = \alpha \cdot |\mathbf{b}(x)|$ and we initially chose α so that it satisfies $\alpha \cdot |\mathbf{b}(x)| < (\mathbf{A}\mathbf{u})(x)$, which leads to a contradiction.

Now, we claim that $S' \subseteq [\![\mathbf{v}]\!]$. Indeed, by the construction of $SU_{\mathcal{M}}$, if $x \in S'$ then either x is additively incremented by t or $[\![t]\!]_x^- \cap S \neq \emptyset$. If x is additively incremented by t, then $\alpha\mathbf{b}(x) > 0$ and since $\mathbf{A}$

is a non-negative matrix, it follows that $\mathbf{v}(x) > 0$ as well. On the other hand if $[\![t]\!]_x^- \cap S \neq \emptyset$, then there must be $y \in S \subseteq [\![\mathbf{u}]\!]$ such that $\mathbf{A}(x, y) > 0$ and so $(\mathbf{A}\mathbf{u})(x) > 0$. If x is additively decremented by t, then by the choice of α, it follows that $\mathbf{v}(x) > 0$. If x is not additively decremented by t, then $\mathbf{v}(x) \geq (\mathbf{A}\mathbf{u})(x) > 0$. Hence, in either case, $x \in [\![\mathbf{v}]\!]$ and so $S' \subseteq [\![\mathbf{v}]\!]$.

It is then easy to see that $p(\mathbf{u}) \xrightarrow{\alpha t} q(\mathbf{v})$ is indeed a step in $\mathcal{M}$ which proves the Completeness property.

Finally, by using the Soundness and Completeness properties and by induction on the length of the run, it is then easy to show the required claim.

> CLAIM: $p(\mathbf{u})$ can reach a configuration whose state is q in $\mathcal{M}$ if and only if $(p, [\![\mathbf{u}]\!])$ can reach some vertex of the form (q, S) in the graph $SU_{\mathcal{M}}$.

This concludes the required proof.

D   Proofs of Subsection 4.2

D.1   Proof of Lemma 4.3

LEMMA 4.3. *The following statements are true for any firing sequence π and any fraction $\alpha \in (0, 1]$:*

(1) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, $p(\alpha\mathbf{u}) \xrightarrow{\alpha\pi} q(\alpha\mathbf{v})$.*

(2) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, for any $\mathbf{w} \in \mathbb{Q}_+^d$, $p(\mathbf{u} + \mathbf{w}) \xrightarrow{\pi} q(\mathbf{v}')$ where $\mathbf{v}' \geq \mathbf{v} + \mathbf{w}$.*

(3) *Suppose $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$. Then, $p(\mathbf{u}) \xrightarrow{\alpha\pi} q(\mathbf{v}')$ where $\mathbf{v}' \geq \alpha\mathbf{v} + (1 - \alpha)\mathbf{u}$.*

PROOF. All of these claims are clear for the case when π is the empty sequence. We will prove all of these claims when π is a single step; the general case follows by a straightforward induction on the number of steps of π.

Throughout we fix two configurations $p(\mathbf{u}), q(\mathbf{v})$ and a pair γt where $\gamma \in (0, 1]$ and $t = (p, \mathbf{A}, \mathbf{b}, q)$ is a transition such that $p(\mathbf{u}) \xrightarrow{\gamma t} q(\mathbf{v})$. Hence, this means that $\mathbf{v} = \mathbf{A}\mathbf{u} + \gamma\mathbf{v}$.

Proof of 1: Notice that $\alpha\mathbf{v} = \alpha\mathbf{A}\mathbf{u} + \alpha\gamma\mathbf{b} = \mathbf{A}(\alpha\mathbf{u}) + (\alpha\gamma)\mathbf{b}$ and hence $p(\alpha\mathbf{u}) \xrightarrow{\alpha\gamma t} q(\alpha\mathbf{v})$.

Proof of 2: Notice that $\mathbf{A}(\mathbf{u} + \mathbf{w}) + \gamma\mathbf{b} = \mathbf{A}\mathbf{u} + \gamma\mathbf{b} + \mathbf{A}\mathbf{w} = \mathbf{v} + \mathbf{A}\mathbf{w}$. Now, since $\mathbf{A}$ is a non-negative self-loop matrix, $\mathbf{A}(x, x) \geq 1$ for every counter x and so $\mathbf{A}\mathbf{w}(x) \geq \mathbf{w}(x)$. It then follows that if we let $\mathbf{v}' = \mathbf{A}(\mathbf{u} + \mathbf{w}) + \gamma\mathbf{b}$, then $\mathbf{v}' \geq \mathbf{v} + \mathbf{w}$. It is then easy to see that $p(\mathbf{u} + \mathbf{w}) \xrightarrow{\gamma t} q(\mathbf{v}')$.

Proof of 3: Notice that by (1), we have $p(\alpha\mathbf{u}) \xrightarrow{\alpha\gamma t} q(\alpha\mathbf{v})$. By (2), we have $p(\alpha\mathbf{u} + (1 - \alpha)\mathbf{u}) = p(\mathbf{u}) \xrightarrow{\alpha\gamma t} q(\mathbf{v}')$ where $\mathbf{v}' \geq \alpha\mathbf{v} + (1 - \alpha)\mathbf{u}$. □

D.2   Lemma 4.4 implies Theorem 4.2

In this subsection, we will prove that Lemma 4.4 implies Theorem 4.2.

We claim that a configuration $p(\mathbf{u})$ can cover a configuration $q(\mathbf{v})$ if and only if there is a *witness*, i.e., a sequence of configurations

$$p_1(\mathbf{u}_1), p_1(\mathbf{u}_1'), p_2(\mathbf{u}_2), p_2(\mathbf{u}_2'), \ldots, p_k(\mathbf{u}_k), p_k(\mathbf{u}_k')$$

such that each $p_i \neq p_j$ for $i \neq j$, $p_1(\mathbf{u}_1) = p(\mathbf{u})$, $p_k = q$, $\mathbf{u}'_k \geq \mathbf{v}$ and for each i, $p_i(\mathbf{u}_i)$ can cover $p_i(\mathbf{u}'_i)$ and $p_i(\mathbf{u}'_i) \rightarrow p_{i+1}(\mathbf{u}_{i+1})$. Indeed, if such a sequence exists, then by applying Lemma 4.3 (2), each $p_i(\mathbf{u}_i)$ can cover $p_{i+1}(\mathbf{u}_{i+1})$. Repeatedly applying Lemma 4.3 (2) then implies that $p(\mathbf{u}) = p_1(\mathbf{u}_1)$ can cover $p_k(\mathbf{u}'_k)$. Since $\mathbf{u}'_k \geq \mathbf{v}$, this proves that $p(\mathbf{u})$ can cover $q(\mathbf{v})$.

On the other hand, suppose we have a run ρ from $p(\mathbf{u})$ which covers $q(\mathbf{v})$. We proceed by an induction on the length of the run ρ, where the induction hypothesis asserts the existence of a witness for $p(\mathbf{u})$ covering $q(\mathbf{v})$ such that all the states of the witness appear in the run ρ. The base case of a single step is clear. For the induction step, let i be the last position in ρ in which a configuration with state p appears. Let $p(\mathbf{w})$ and $p'(\mathbf{w}')$ be the configuration in positions i and $i + 1$ of ρ. We apply the induction hypothesis to the run from $p'(\mathbf{w}')$ to obtain a witness $p_1(\mathbf{u}_1), \ldots, p_k(\mathbf{u}_k)$ for $p'(\mathbf{w}')$ covering $q(\mathbf{v})$. It then follows that $p(\mathbf{u}), p(\mathbf{w}), p_1(\mathbf{u}_1), \ldots, p_k(\mathbf{u}_k)$ is a witness for $p(\mathbf{u})$ covering $q(\mathbf{v})$.

Note that from the definition of the step relation, given two states p, q, we can easily come up with a formula $\phi^s_{p,q}$ in ELRA such that $\phi^s_{p,q}(\mathbf{u}, \mathbf{v})$ is true if and only if $p(\mathbf{u}) \rightarrow q(\mathbf{v})$, i.e., $p(\mathbf{u})$ can reach $q(\mathbf{v})$ by a single step. Hence, to decide coverability, we only have to find a witness. Note that if $p_1(\mathbf{u}_1), \ldots, p_k(\mathbf{u}'_k)$ is a witness then the condition that $p_i \neq p_j$ for $i \neq j$ in a witness implies that $k \leq |Q|$. Hence, to find a witness, we just have to guess a sequence of states $p_1, \ldots, p_k$ of the desired kind (which can be done in polynomial time, thanks to the fact that $k \leq |Q|$) and then check for satisfiability of the formula

$$\exists \mathbf{u}_1, \mathbf{u}'_1, \ldots, \mathbf{u}_k, \mathbf{u}'_k \bigwedge_{1 \leq i \leq k} \phi_{p_i}(\mathbf{u}_i, \mathbf{u}'_i) \wedge \bigwedge_{1 \leq i \leq k-1} \phi^s_{p_i, p_{i+1}}(\mathbf{u}'_i, \mathbf{u}_{i+1})$$
$$\wedge \, \mathbf{u}_1 = \mathbf{u} \wedge \mathbf{u}'_k \geq \mathbf{v}$$

Since satisfiability of formulas in ELRA can be done in NP, this then proves Theorem 4.2.

D.3 Proof of Proposition 4.5

PROPOSITION 4.5. *Suppose we have a run* $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$ *for some* $p, \mathbf{u}, \mathbf{v}$ *and* π. *Then, there is a firing sequence Rep-Half(π) such that*

- *If* $\pi = \alpha_1 t_1, \alpha_2 t_2, \ldots, \alpha_\ell t_\ell$, *then*

  $$Rep\text{-}Half(\pi) = \alpha_1/2t_1, \alpha_2/4t_2, \ldots, \alpha_\ell/2^\ell t_\ell$$

- $p(\mathbf{u}) \xrightarrow{Rep\text{-}Half(\pi)} p(\mathbf{w})$ *for some* $\mathbf{w}$ *such that* $[\![\mathbf{u}]\!] \subseteq [\![\mathbf{w}]\!]$, *and*

- *If* t *is a transition in* π *and* x *is some counter, then*
  - *If* x *is additively incremented by* t, *then* $\mathbf{w}(x) > 0$.
  - *If* x *is additively decremented by* t, *then* $[\![t]\!]^-_x \cap [\![\mathbf{w}]\!] \neq \emptyset$.
  - *If* x *was pumped by* t *in the run* $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$, *then* $[\![t]\!]^p_x \cap [\![\mathbf{w}]\!] \neq \emptyset$.

PROOF. Suppose $\pi = \alpha_1 t_1, \alpha_2 t_2, \ldots, \alpha_\ell t_\ell$. Let $p_0(\mathbf{u}_0) = p(\mathbf{u}) \xrightarrow{\alpha_1 t_1}$ $p_1(\mathbf{u}_1) \xrightarrow{\alpha_2 t_2} p_2(\mathbf{u}_2) \ldots \xrightarrow{\alpha_\ell t_\ell} p_\ell(\mathbf{u}_\ell) = p(\mathbf{v})$.

We let Rep-Half$(\pi) = \alpha_1/2t_1, \alpha_2/4t_2, \ldots, \alpha_\ell/2^\ell t_\ell$ and we claim that this satisfies all the claims of the proposition. We now show

by induction on $k \in [0, \ell]$, that

$$p_0(\mathbf{u}_0) \xrightarrow{\text{Rep-Half}(\pi)[1 \cdots k]} p_k(\mathbf{w}_k) \text{ where } \mathbf{w}_k \geq \frac{1}{2^k} \mathbf{u}_k + \sum_{i=0}^{k-1} \frac{1}{2^{i+1}} \mathbf{u}_i$$

Indeed, the base case of $k = 0$ is immediate. For the induction step, note that since $p_{k-1}(\mathbf{w}_{k-1}) \xrightarrow{\alpha_k t_k} p_k(\mathbf{w}_k)$, by Lemma 4.3 (1), it follows that $p_{k-1}\left(\frac{1}{2^k}\mathbf{u}_{k-1}\right) \xrightarrow{\frac{\alpha_k}{2^k} t_k} p_k\left(\frac{1}{2^k}\mathbf{u}_k\right)$. Then, if we add $\sum_{i=0}^{k-1} \frac{1}{2^{i+1}} \mathbf{u}_i$ to both sides, by Lemma 4.3 (2), we get

$$p_{k-1}\left(\frac{1}{2^{k-1}}\mathbf{u}_{k-1} + \sum_{i=0}^{k-2} \frac{1}{2^{i+1}}\mathbf{u}_i\right) \xrightarrow{\frac{\alpha_k}{2^k} t_k} p_k(\mathbf{w}_k)$$

$$\text{where } \mathbf{w}_k \geq \frac{1}{2^k} \mathbf{u}_k + \sum_{i=0}^{k-1} \frac{1}{2^{i+1}} \mathbf{u}_i \quad (9)$$

By induction hypothesis, we have that

$$p_0(\mathbf{u}_0) \xrightarrow{\text{Rep-Half}(\pi)[1 \cdots (k-1)]} p_{k-1}(\mathbf{w}_{k-1})$$

where $\mathbf{w}_{k-1} \geq \frac{1}{2^{k-1}}\mathbf{u}_{k-1} + \sum_{i=0}^{k-2}\frac{1}{2^{i+1}}\mathbf{u}_i$. Hence, by Lemma 4.3 (2) and Equation 9 we can conclude the proof of the induction.

Let $\mathbf{w} = \mathbf{w}_\ell$. Suppose a counter x was additively incremented (resp. decremented) by some transition t in Rep-Half(π). Since $[\![\text{Rep-Half}(\pi)]\!] = [\![\pi]\!]$, it follows that t is also a transition in π. Hence, there must be some k such that $\mathbf{u}_k(x) > 0$ or $[\![t]\!]^-_x \cap [\![\mathbf{u}_k]\!] \neq \emptyset$ respectively. Since $\mathbf{w} \geq 1/2^{k+1}\mathbf{u}_k$, it would then follow that a similar property is satisfied by $\mathbf{w}$ as well.

Suppose a counter x was pumped by t in the run $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$. By definition of pumping, there is some $\mathbf{u}_k$ such that $[\![t]\!]^p_x \cap [\![\mathbf{u}_k]\!] \neq \emptyset$. Since $\mathbf{w} \geq 1/2^{k+1}\mathbf{u}_k$, we have $[\![t]\!]^p_x \cap [\![\mathbf{w}]\!] \neq \emptyset$ and this finishes the proof. □

D.4 Proof of the Pumping Counters Lemma (Lemma 4.6)

To prove this lemma, we first prove a bunch of small results.

PROPOSITION D.1 (MARKING EQUATION). *Let* $p(\mathbf{u}) \xrightarrow{\pi} q(\mathbf{v})$ *be a run such that* $\pi = \alpha_1 t_1, \alpha_2 t_2, \ldots, \alpha_\ell t_\ell$ *and each* $t_i = (p_{i-1}, \mathbf{A}_i, \mathbf{b}_i, p_i)$. *Then,*

$$\mathbf{v} = \left(\prod_{j=\ell}^{1} \mathbf{A}_j\right)\mathbf{u} + \sum_{j=1}^{\ell}\left(\prod_{k=\ell}^{j+1} \mathbf{A}_k\right)\alpha_j \mathbf{b}_j$$

where $\prod_{j=\ell}^{1} \mathbf{A}_j$ *denotes the matrix product* $\mathbf{A}_\ell \cdot \mathbf{A}_{\ell-1} \cdots \mathbf{A}_1$; *similarly* $\prod_{k=\ell}^{j+1} \mathbf{A}_k$ *denotes the product* $\mathbf{A}_\ell \cdot \mathbf{A}_{\ell-1} \cdots \mathbf{A}_{j+1}$.

PROOF. Follows by induction on the length of π and the definition of a step. □

PROPOSITION D.2 (FUNDAMENTAL PROPERTY OF SELF-LOOP MATRICES). *Suppose* $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_\ell$ *are non-negative self-loop matrices. Then,* $\mathbf{B} = \prod_{k=1}^{\ell} \mathbf{A}_k$ *is also a non-negative self-loop matrix such that* $\mathbf{B}(i, j) \geq \mathbf{A}_k(i, j)$ *for every* i, j *and* k

PROOF. Follows easily by induction on ℓ. □

LEMMA 4.6 (PUMPING COUNTERS LEMMA). *Suppose* $\rho : p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{v})$ *is a run. Let S be the set of counters such that $x \in S$ only if x is pumped by some transition t in ρ and $[\![t]\!]^p_x \cap [\![\mathbf{v}]\!] \neq \emptyset$. Then, for any $K > 0$, there exists a run π' such that*

- $\pi' = \pi/2 \cdot (\pi/2n)^n$ *for some n.*
- $p(\mathbf{u}) \xrightarrow{\pi'} p(\mathbf{w})$ *such that $\mathbf{w} \geq \mathbf{v}$ and*
- *For every $x \in S$, $\mathbf{w}(x) > K$.*

PROOF. We prove the lemma in two steps.

Step 1: Fire $\pi/2$ from $p(\mathbf{u})$. Let $\pi = \alpha_1 t_1, \alpha_2 t_2, \ldots, \alpha_\ell t_\ell$. Let each $t_i = (p_{i-1}, \mathbf{A}_i, \mathbf{b}_i, p_i)$. By the marking equation we have that

$$\mathbf{v} = \left(\prod_{j=\ell}^{1} \mathbf{A}_j \right) \mathbf{u} + \sum_{j=1}^{\ell} \left(\prod_{k=\ell}^{j+1} \mathbf{A}_k \right) \alpha_j \mathbf{b}_j$$

Let

$$\mathbf{z} = \sum_{j=1}^{\ell} \left(\prod_{k=\ell}^{j+1} \mathbf{A}_k \right) \alpha_j \mathbf{b}_j$$

By Lemma 4.3 (2), it follows that $p(\mathbf{u}) \xrightarrow{\pi/2} p(\mathbf{u}')$ where $\mathbf{u}' \geq \mathbf{v}/2 + \mathbf{u}/2$. This completes Step 1.

Step 2: Choosing n and firing $1/2n$-scaled down versions of π. Let $m = \min\{\mathbf{v}(x) : \mathbf{v}(x) \neq 0\}$. Choose n so that $nm/2 > K$. By assumption, for every counter $x \in S$ there exists some counter c_x such that $c_x \in [\![t]\!]^p_x \cap [\![\mathbf{v}]\!]$. We now construct a sequence of configurations $p(\mathbf{u}_1), p(\mathbf{u}_2), \ldots, p(\mathbf{u}_{n+1})$ such that each $p(\mathbf{u}_i)$ satisfies the following property: For every counter x,

- Property 1: If $x \notin S$, then $\mathbf{u}_i(x) \geq \frac{\mathbf{v}(x)}{2} + \frac{(n-i+1)\mathbf{u}(x)}{2n} + \frac{(i-1)\mathbf{v}(x)}{2n}$
- Property 2: If $x \in S$, $\mathbf{u}_i(x) \geq \frac{\mathbf{v}(x)}{2} + \frac{(n-i+1)\mathbf{u}(x)}{2n} + \frac{(i-1)\mathbf{v}(x)}{2n} + \frac{(i-1)\mathbf{v}(c_x)}{2}$

Notice that if we let $p(\mathbf{u}_1) = p(\mathbf{u}')$, then $p(\mathbf{u}_1)$ satisfies the above property. Now, assume that $p(\mathbf{u}_i)$ already satisfies this property. Hence, $\mathbf{u}_i \geq \mathbf{u}/2n$. By Lemma 4.3 (1 and 2) it then follows that we can fire $\pi/2n$ from $\mathbf{u}_i$. Let $p(\mathbf{u}_i) \xrightarrow{\pi/2n} p(\mathbf{u}_{i+1})$. By the marking equation, we have that

$$\mathbf{u}_{i+1} = \left(\prod_{j=\ell}^{1} \mathbf{A}_j \right) \mathbf{u}_i + \frac{1}{2n} \sum_{j=1}^{\ell} \left(\prod_{k=\ell}^{j+1} \mathbf{A}_k \right) \alpha_j \mathbf{b}_j = \left(\prod_{j=\ell}^{1} \mathbf{A}_j \right) \mathbf{u}_i + \frac{\mathbf{z}}{2n} \quad (10)$$

Let $\mathbf{B} = \left(\prod_{j=\ell}^{1} \mathbf{A}_j \right)$. By the induction hypothesis, we have that $\mathbf{u}_i = \frac{\mathbf{v}}{2} + \frac{(n-i+1)}{2n}\mathbf{u} + \frac{(i-1)\mathbf{v}}{2n} + \mathbf{u}'_i$ for some non-negative vector $\mathbf{u}'_i$ such that $\mathbf{u}'_i(x) \geq \frac{(i-1)\mathbf{v}(c_x)}{2}$ for every $x \in S$. Hence,

$$\mathbf{B}\mathbf{u}_i = \mathbf{B} \left(\frac{\mathbf{v}}{2} + \frac{(n-i+1)\mathbf{u}}{2n} + \frac{(i-1)\mathbf{v}}{2n} + \mathbf{u}'_i \right)$$

By the Fundamental property of non-negative self-loop matrices $\mathbf{B}$ is also a non-negative self-loop matrix and hence all of its diagonal entries are at least 1. Hence,

$$\mathbf{B} \left(\frac{\mathbf{v}}{2} + \frac{(n-i+1)\mathbf{u}}{2n} + \frac{(i-1)\mathbf{v}}{2n} + \mathbf{u}'_i \right)$$
$$\geq \mathbf{B}\frac{\mathbf{v}}{2} + \mathbf{B}\frac{\mathbf{u}}{2n} + \frac{(n-i)\mathbf{u}}{2n} + \frac{(i-1)\mathbf{v}}{2n} + \mathbf{u}'_i$$

Combining the last three equations gives us,

$$\mathbf{u}_{i+1} \geq \mathbf{B}\frac{\mathbf{v}}{2} + \mathbf{B}\frac{\mathbf{u}}{2n} + \frac{(n-i)\mathbf{u}}{2n} + \frac{(i-1)\mathbf{v}}{2n} + \mathbf{u}'_i + \frac{\mathbf{z}}{2n}$$

By equation 10, $\mathbf{B}\frac{\mathbf{u}}{2n} + \frac{\mathbf{z}}{2n} = \frac{\mathbf{v}}{2n}$ and so

$$\mathbf{u}_{i+1} \geq \mathbf{B}\frac{\mathbf{v}}{2} + \frac{(n-i)\mathbf{u}}{2n} + \frac{i\mathbf{v}}{2n} + \mathbf{u}'_i$$

If $x \notin S$, then since $\mathbf{B}$ is a self-loop matrix, we have that $\left(\mathbf{B}\frac{\mathbf{v}}{2} \right)(x) \geq \frac{\mathbf{v}(x)}{2}$ and so Property 1 for $\mathbf{u}_{i+1}$ is satisfied. On the other hand, suppose $x \in S$. Then x is pumped in the run ρ and so this means that for some i, $\mathbf{A}_i(x, c_x) > 0$ if $c_x \neq x$ and $\mathbf{A}_i(x, c_x) > 1$ if $c_x = x$. By the Fundamental property of self-loop matrices, it follows that $\mathbf{B}(x, c_x) \geq \mathbf{A}_i(x, c_x)$ and so it follows that $\left(\mathbf{B}\frac{\mathbf{v}}{2} \right)(x) \geq \frac{\mathbf{v}(x)}{2} + \frac{\mathbf{v}(c_x)}{2}$. Since $\mathbf{u}'_i(x) \geq \frac{(i-1)\mathbf{v}(c_x)}{2}$ for every $x \in S$, it follows that Property 2 is also satisfied for $\mathbf{u}_{i+1}$.

Now, by assumption on $\mathbf{v}$, for every $x \in S$, $\mathbf{v}(c_x) > 0$. By definition of m this means that $\mathbf{v}(c_x) \geq m$. Hence, when Property 1 and 2 are applied to $p(\mathbf{u}_{n+1})$, we have

- If $x \notin S$, then $\mathbf{u}_{n+1}(x) \geq \mathbf{v}(x)$
- If $x \in S$, then $\mathbf{u}_{n+1}(x) \geq \mathbf{v}(x) + \frac{n\mathbf{v}(c_x)}{2} \geq nm/2 > K$

Hence, if we set $\mathbf{w} = \mathbf{u}_{n+1}$, then the lemma becomes true. □

D.5 Proof of the Coverability Characterization lemma (Lemma 4.7)

LEMMA 4.7 (COVERABILITY CHARACTERIZATION LEMMA). *$p(\mathbf{u})$ can cover $p(\mathbf{v})$ in $\mathcal{M}$ if and only if there exists a set of counters X, a vector $\mathbf{w} \geq \mathbf{v}$ and firing sequences π', π_{fwd} over $\mathcal{M}$ such that*

(1) *$p(\mathbf{u}_X) \xrightarrow{\pi'_X} p(\mathbf{w}_X)$ is a run in $\mathcal{M}_X$*
(2) *π_{fwd} is p-admissible from $p(\mathbf{u})$ in $\mathcal{M}$*
(3) *Every counter x that is **not in** X is pumped by the p-admissible run obtained from $p(\mathbf{u})$ by firing π_{fwd}*
(4) *$[\![\pi']\!] = [\![\pi_{fwd}]\!]$*

PROOF. Suppose $p(\mathbf{u})$ can cover $p(\mathbf{v})$ in $\mathcal{M}$. Hence, $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{w})$ for some $\mathbf{w} \geq \mathbf{v}$. Let X be the set of counters not pumped by this run. Clearly we can set $\pi_{fwd} = \pi$, $\pi' = \pi$ and conditions 2, 3 and 4 are all satisfied. To prove condition 1, let the run $p(\mathbf{u}) \xrightarrow{\pi} p(\mathbf{w})$. be $p_0(\mathbf{u}_0) = p(\mathbf{u}) \xrightarrow{\alpha_1 t_1} p_1(\mathbf{u}_1) \ldots \xrightarrow{\alpha_\ell t_\ell} p_\ell(\mathbf{u}_\ell) = p(\mathbf{w})$ where each $t_i = (p_{i-1}, \mathbf{A}_i, \mathbf{b}_i, p_i)$. If $x \in X$, then x is not pumped in this run and so for every i, $\mathbf{u}_i(x) = \mathbf{u}_{i-1}(x) + \alpha_i \mathbf{b}_i$. It then follows that $p(\mathbf{u}_X) \xrightarrow{\pi_X} p(\mathbf{w}_X)$ and so condition 1 is satisfied.

Now, suppose there exists a set of counters X and firing sequences π', π_{fwd} over $\mathcal{M}$ which satisfies all the four conditions. Let ρ be the p-admissible run obtained from $p(\mathbf{u})$ by firing π_{fwd}. Condition 3 tells us that if $x \notin X$, then x is pumped by ρ. Without loss of generality, we can assume that if $x \in X$, then x is not pumped by ρ. Indeed, if there is a set of counters $S \subseteq X$, that is pumped by ρ, then we set $X' = X \setminus S$. Since condition 1 guarantees us that $p(\mathbf{u}_X) \xrightarrow{\pi'_X} p(\mathbf{w}_X)$ is a run in $\mathcal{M}_X$, we have a run $p(\mathbf{u}_{X'}) \xrightarrow{\pi'_{X'}} p(\mathbf{w}_{X'})$ in $\mathcal{M}_{X'}$. This new X' along with π' and π_{fwd} still satisfies conditions 1 - 4. Hence, for the rest of the proof, we will assume that a counter is pumped by ρ iff it is not in X.

As mentioned before, let ρ be the run obtained from $p(\mathbf{u})$ by firing π_{fwd}. We will now prove the required claim in multiple steps.

Step 1: Constructing an alternate run in $\mathcal{M}_X$. In this step, we will concentrate on $\mathcal{M}_X$. We will modify the run $p(\mathbf{u}_X) \xrightarrow{\pi'_X} p(\mathbf{w}_X)$ in $\mathcal{M}_X$ into a different run with the same end-points which we will then use in the next step.

Let us look at the firing sequences $\pi_{\text{fwd}_X} = \alpha_1 t_{1_X}, \dots, \alpha_\ell t_{\ell_X}$ and Rep-Half$(\pi_{\text{fwd}_X}) = \alpha_1/2 t_{1_X}, \dots, \alpha_\ell/2^\ell t_{\ell_X}$ in $\mathcal{M}_X$. Because ρ did not pump any counters in X, it follows that we can fire π_{fwd_X} from $p(\mathbf{u}_X)$ in $\mathcal{M}_X$. Proposition 4.5 then tells us that we can fire Rep-Half(π_{fwd_X}) from $p(\mathbf{u}_X)$ to reach a configuration $p(\mathbf{z}'')$ such that for every transition $t_X \in [\![\pi_{\text{fwd}_X}]\!] = [\![\text{Rep-Half}(\pi_{\text{fwd}_X})]\!]$ and every counter x, if x is additively incremented or decremented by t_X, then $\mathbf{z}''(x) > 0$.

The proof of [7, Proposition 4.5] shows how to use this vector $\mathbf{z}''$ along with condition 1 to show that we can pick a small enough fraction λ to get a run of the following form in the continuous VASS $\mathcal{M}_X$.

$$p(\mathbf{u}_X) \xrightarrow{\lambda\text{Rep-Half}\left(\pi_{\text{fwd}_X}\right)} p(\mathbf{z}') \xrightarrow{\eta_X} p(\mathbf{w}_X) \tag{11}$$

for some firing sequence $\eta_X = \beta_1 r_{1_X}, \dots, \beta_k r_{k_X}$ in $\mathcal{M}_X$ such that $[\![\eta_X]\!] = [\![\pi_X]\!] = [\![\pi_{\text{fwd}_X}]\!]$.

Note that this induces a firing sequence $\eta = \beta_1 r_1, \dots, \beta_k r_k$ in $\mathcal{M}$ where each $r_i = (p_{i-1}, \mathbf{A}_i, \mathbf{b}_i, p_i)$ and $[\![\eta]\!] = [\![\pi]\!] = [\![\pi_{\text{fwd}}]\!]$. Define

$$\epsilon = \max \left\{ \left(\sum_{j=1}^{i} \left(\prod_{e=i}^{j+1} \mathbf{A}_e \right) \alpha_j |\mathbf{b}_j| \right)(x) : 1 \le i \le \ell', 1 \le x \le d \right\} \tag{12}$$

Intuitively ϵ is the maximum amount that any step of η can decrement on any counter. This completes Step 1. In the next steps, we will show how to use η to construct what we want.

Step 2: Preparing to fire η in $\mathcal{M}$. In this step, we will concentrate on $\mathcal{M}$. We will figure out a way to reach a configuration from $p(\mathbf{u})$ in $\mathcal{M}$, from which we can fire η.

Condition 3 tells us that $\pi_{\text{fwd}} = \alpha_1 t_1, \dots, \alpha_\ell t_\ell$ can be fired from $p(\mathbf{u})$. By Proposition 4.5, it follows that there is a firing sequence Rep-Half$(\pi_{\text{fwd}}) = \alpha_1/2 t_1, \dots, \alpha_\ell/2^\ell t_\ell$ that can be fired from $p(\mathbf{u})$ to reach some $p(\mathbf{u}'')$ such that if a counter x is pumped by some transition t in ρ, then $[\![t]\!]_x^p \cap [\![\mathbf{u}'']\!] \ne \emptyset$. By Lemma 4.3 (3), we then have that $p(\mathbf{u}) \xrightarrow{\lambda\text{Rep-Half}(\pi_{\text{fwd}})} p(\mathbf{u}')$ where $\mathbf{u}' \ge \lambda\mathbf{u}'' + (1-\lambda)\mathbf{u}$ and so $[\![\mathbf{u}'']\!] \subseteq [\![\mathbf{u}']\!]$. Hence, if a counter x is pumped by some transition t in ρ, then $[\![t]\!]_x^p \cap [\![\mathbf{u}']\!] \ne \emptyset$.

We now claim that $\mathbf{u}'_X \ge \mathbf{z}'$. Indeed, by definitions of $\mathbf{u}'$ and $\mathbf{z}'$ we have

$$p(\mathbf{u}) \xrightarrow{\lambda\text{Rep-Half}(\pi_{\text{fwd}})} p(\mathbf{u}') \text{ is a run in } \mathcal{M}$$

and

$$p(\mathbf{u}_X) \xrightarrow{\lambda\text{Rep-Half}\left(\pi_{\text{fwd}_X}\right)} p(\mathbf{z}') \text{ is a run in } \mathcal{M}_X$$

Now, the first run is of the form $p(\mathbf{u}) = q_0(\mathbf{u}_0) \xrightarrow{\lambda\alpha_1/2 t_1} q_1(\mathbf{u}_1) \dots$ $\xrightarrow{\lambda\alpha_\ell/2^\ell t_\ell} q_\ell(\mathbf{u}_\ell) = p(\mathbf{u}')$ and the second run is of the form $p(\mathbf{u}_X) =$

$q_0(\mathbf{z}'_0) \xrightarrow{\lambda\alpha_1/2 t_{1_X}} q_1(\mathbf{z}'_1) \dots \xrightarrow{\lambda\alpha_\ell/2^\ell t_{\ell_X}} q_\ell(\mathbf{z}'_\ell) = p(\mathbf{z}')$. By induction on i, it is easy to verify that $\mathbf{u}_{i_X} \ge \mathbf{z}'_i$. Hence, $\mathbf{u}'_X \ge \mathbf{z}'$. This completes Step 2.

Step 3: Firing η to cover $p(\mathbf{w})$. In the third step, we focus on firing η. This will ultimately allow us to cover $p(\mathbf{w})$ and since $\mathbf{w} \ge \mathbf{v}$, this will cover $p(\mathbf{v})$.

From the previous step, we know that $p(\mathbf{u}) \xrightarrow{\lambda\text{Rep-Half}(\pi_{\text{fwd}})} p(\mathbf{u}')$ such that whenever a counter $x \notin X$ then there is a transition $t \in [\![\pi]\!] = [\![\pi_{\text{fwd}}]\!] = [\![\lambda\text{Rep-Half}(\pi_{\text{fwd}})]\!]$ such that $[\![t]\!]_x^p \cap [\![\mathbf{u}']\!] \ne \emptyset$. Applying the Pumping Counters lemma, we now get a run $p(\mathbf{u}) \xrightarrow{*} p(\mathbf{s})$ such that $\mathbf{s} \ge \mathbf{u}'$ and for every counter $x \notin X, \mathbf{s}(x) > \ell'\epsilon + \mathbf{w}(x)$.

Now, let $p(\mathbf{z}') = p_0(\mathbf{z}_0) \xrightarrow{\beta_1 r_{1_X}} p_1(\mathbf{z}_1) \xrightarrow{\beta_2 r_{2_X}} \dots \xrightarrow{\beta_k r_{k_X}}$ $p(\mathbf{z}_k) = p(\mathbf{w}_X)$ be the run obtained from firing η_X in $\mathcal{M}_X$ from Equation 11. By induction on i, we will now show that $p(\mathbf{s}) = p_0(\mathbf{s}_0) \xrightarrow{\beta_1 r_1} p_1(\mathbf{s}_1) \dots \xrightarrow{\beta_i r_i} p_i(\mathbf{s}_i)$ is a run in $\mathcal{M}$ where $\mathbf{s}_i(x) > (\ell' - i)\epsilon + \mathbf{w}(x)$ if $x \notin X$ and $\mathbf{s}_i(x) \ge \mathbf{z}_i(x)$ if $x \in X$. Indeed, the base case of $i = 0$ follows from the definition of $\mathbf{s}$ and the fact that $\mathbf{s}_X \ge \mathbf{u}'_X \ge \mathbf{z}'$. For the induction step, notice that by definition of ϵ, each r_i can decrement any counter by at most ϵ and so $\mathbf{s}_{i+1}(x) \ge \mathbf{s}_i(x) - \epsilon > (\ell' - (i+1))\epsilon + \mathbf{w}(x)$ if $x \notin X$. Further, it is easy to see that $\mathbf{s}_{i+1}(x) \ge \mathbf{z}_{i+1}(x)$. Hence, it follows that from $p(\mathbf{s})$ we can reach a configuration $p(\mathbf{s}_k)$ such that for every $x \notin X$, we have $\mathbf{s}_k(x) > \mathbf{w}(x)$ and for every $x \in X$ $\mathbf{s}_k(x) \ge \mathbf{z}_k(x) = \mathbf{w}(x)$. Hence, we have a run from $p(\mathbf{z}')$ covering $p(\mathbf{w})$ which in turn covers $p(\mathbf{v})$. Hence, $p(\mathbf{z}')$ can cover $p(\mathbf{v})$ and since $p(\mathbf{u})$ can reach $p(\mathbf{z}')$, $p(\mathbf{u})$ can also cover $p(\mathbf{v})$ which finishes the proof of the lemma. $\qquad\square$

D.6 Proof of Lemma 4.8

Before we prove Lemma 4.8, we first prove another lemma.

LEMMA D.3. *Let $p(\mathbf{u})$ be a configuration, p be some state, Y be some set of counters and let $\rho = t_1, t_2, \dots, t_\ell$ be some sequence of transitions. If ρ is a valid path in $\mathcal{G}_\mathcal{M}$ from p to p such that*

- *For every counter y and every transition t_i, if $[\![t_i]\!]_y^-$ is non-empty, then*

$$[\![t_i]\!]_y^- \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{1 \le j < i} [\![t_j]\!]^+ \right) \ne \emptyset$$

- *For every counter $y \in Y$, there exists some transition t_i such that*

$$[\![t]\!]_y^p \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{1 \le j \le \ell} [\![t_j]\!]^+ \right) \ne \emptyset$$

then there exists a p-admissible sequence π from $p(\mathbf{u})$ such that the resulting run pumps all counters in Y.

PROOF. Since ρ is a valid path in $\mathcal{G}_\mathcal{M}$, it follows that we can let each $t_i = (p_{i-1}, \mathbf{A}_i, \mathbf{b}_i, p_i)$ where $p_0 = p_\ell = p$. By induction on i, we will now show that there is a run of the form

$$p_0(\mathbf{u}_0) = p(\mathbf{u}) \xrightarrow{\alpha_1 t_1, \dots, \alpha_i t_i} p_i(\mathbf{u}_i)$$

for some fractions $\alpha_1, \dots, \alpha_i$ and some non-negative vector $\mathbf{u}_i$ such that $[\![\mathbf{u}]\!] \subseteq [\![\mathbf{u}_i]\!]$ and for every counter x and every transition t_j with $j \le i$,

- If x is additively incremented by t_j, then $\mathbf{u}_i(x) > 0$.
- If x is additively decremented by t_j, then $[\![t_j]\!]_x^- \cap [\![\mathbf{u}_i]\!] \neq \emptyset$.

The base case of $i = 0$ is clear.

For the induction step, assume that we have a run $p(\mathbf{u}) \xrightarrow{\alpha_1 t_1, \ldots, \alpha_i t_i} p_i(\mathbf{u}_i')$ of the desired form. Now for each counter x that is additively decremented by t_{i+1}, by assumption, there is some counter $c_x \in [\![t_{i+1}]\!]_x^-$ such that either $c_x \in [\![\mathbf{u}]\!]$ or c_x is additively incremented by some t_j with $j \leq i + 1$. In either case this implies that $\mathbf{u}_i(c_x) > 0$. Choose β_i to be so small that $0 < \beta_i < -\mathbf{u}_i(c_x)/\mathbf{b}_{i+1}(x)$ for any x that is additively decremented by t_{i+1}. It is then easy to see that from $\mathbf{u}_i$ we can fire $\beta_i t_i$ to arrive at a configuration of the desired form $p_{i+1}(\mathbf{u}_{i+1})$, which then completes the induction.

Now, we have shown that there is a p-admissible path from $p(\mathbf{u})$ to some $p(\mathbf{w})$ such that $[\![\mathbf{u}]\!] \subseteq [\![\mathbf{w}]\!]$ and for every counter x and every transition t_j with $j \leq \ell$,

- If x is additively incremented by t_j, then $\mathbf{w}(x) > 0$.
- If x is additively decremented by t_j, then $[\![t_j]\!]_x^- \cap [\![\mathbf{w}]\!] \neq \emptyset$.

Let ϵ be a very small fraction so that $\epsilon < -\mathbf{w}(x)/(\mathbf{b}_i(y) \cdot \ell)$ for any x such that $\mathbf{w}(x) > 0$ and any y, i such that $\mathbf{b}_i(y) < 0$. Let $m = \max\{-\mathbf{b}_i(y) : \mathbf{b}_i(y) < 0\}$. We now show that from $p(\mathbf{w})$ we can fire $\epsilon t_1, \epsilon t_2, \ldots, \epsilon t_\ell$ such that the resulting run pumps all the counters in Y.

Indeed, by induction on i, we will show that there is a run of the form
$$p_0(\mathbf{w}_0) = p(\mathbf{w}) \xrightarrow{\epsilon t_1, \ldots, \epsilon t_i} p_i(\mathbf{w}_i)$$
for some non-negative vector $\mathbf{w}_i$ such that $[\![\mathbf{w}]\!] \subseteq [\![\mathbf{w}_i]\!]$ and $\mathbf{w}_i(x) > (\ell - i)\epsilon m$ for every $x \in [\![\mathbf{w}]\!]$ and for every $y \in S$, if $[\![t_i]\!]_y^p$ is non-empty, then y was pumped by t_i. The base case of $i = 0$ is clear.

For the induction step, assume that we have already constructed such a run till $p_i(\mathbf{w}_i)$. Consider t_{i+1}. If t_{i+1} additively decrements some counter x, then by assumption on $\mathbf{w}$, it follows that $[\![t_{i+1}]\!]_x^- \cap [\![\mathbf{w}]\!] \neq \emptyset$. Hence, there is some counter c_x such that $c_x \in [\![t_{i+1}]\!]_x^- \cap [\![\mathbf{w}]\!]$. Since $[\![\mathbf{w}]\!] \subseteq [\![\mathbf{w}_i]\!]$, it follows that $\mathbf{w}_i(c_x) > (\ell - i)\epsilon m$. Notice that t_{i+1} can additively decrement some counter by at most m and so we can now fire ϵt_{i+1} and all counters that were not additively decremented would be at least as large as their value in $\mathbf{w}_i$ and all counters that were additively decremented would be strictly larger than $(\ell - i - 1)\epsilon m$. It then follows that $[\![\mathbf{w}]\!] \subseteq [\![\mathbf{w}_i]\!] \subseteq [\![\mathbf{w}_{i+1}]\!]$ and $\mathbf{w}_{i+1}(x) > (\ell - i - 1)\epsilon m$ for every $x \in [\![\mathbf{w}]\!]$. The only thing that we have to show is that if $y \in S$ is such that $[\![t_{i+1}]\!]_y^p$ is non-empty, then y was pumped by t_{i+1}.

By assumption, if $y \in S$ and $x \in [\![t_{i+1}]\!]_y^p$ then x is either in $[\![\mathbf{u}]\!]$ or x is additively incremented by some t_j. By definition of $\mathbf{w}$, it follows that x is in $[\![\mathbf{w}]\!] \subseteq [\![\mathbf{w}_i]\!]$. Hence, when firing ϵt_{i+1} from $\mathbf{w}_i$, we must have pumped counter y. Hence, the induction step is complete and the proof is done. □

LEMMA 4.8. *There exists a p-admissible run from $p(\mathbf{u})$ which uses exactly the transitions from S and which pumps all the counters in a set Y if and only if there exists an injection $f : S \to \mathbb{Q}_{>0}$ such that for every $t, t' \in S$,*

- *If $f(t)$ has the minimum value among all transitions in S, then $in(t) = p$.*
- *If $f(t) < f(t')$, then there exists a path in $\mathcal{G}_\mathcal{M}$ from $out(t)$ to $in(t')$ which uses only the transitions from the set $\{s : s \in S, f(s) < f(t')\}$.*

- *If $f(t)$ has the maximum value among all transitions in S, then there exists a path in $\mathcal{G}_\mathcal{M}$ from $out(t)$ to p which uses only the transitions from S.*
- *For every counter y additively decremented by t,*

$$[\![t]\!]_y^- \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{s \in S, f(s) < f(t)} [\![s]\!]^+ \right) \neq \emptyset$$

- *For every $y \in Y$, there exists a transition $r \in S$ such that*

$$[\![r]\!]_y^p \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{s \in S} [\![s]\!]^+ \right) \neq \emptyset$$

PROOF. Suppose there is a firing sequence π such that $[\![\pi]\!] = S$ and firing π from $p(\mathbf{u})$ results in a p-admissible run which pumps all the counters in Y. Let $t_1, t_2, \ldots, t_{|S|}$ be such that t_1 is the first distinct transition that appears in π, t_2 is the second distinct transition that appears in π and so on. Let $f(t_i) = i$. It is easy to see that f satisfies the first three conditions. Furthermore, note that if some counter x is additively decremented by some t_j, then there must be $y \in [\![t_j]\!]_x^-$ that either had a non-zero value at $\mathbf{u}$ or that must have been additively incremented by some transition in the set $\{t_1, \ldots, t_{j-1}\}$. Indeed, if this is not the case, the value of all counters in $[\![t_j]\!]_x^-$ will be zero when t_j is fired, which will result in x having a negative value. Hence condition 4 is also satisfied. Similarly we can show that condition 5 is also satisfied.

Suppose there exists an injection f that satisfies all the conditions mentioned above. Let t_1 be the transition which has the least f-value among all transitions in S, i.e., $f(t_1) \leq f(t)$ for every $t \in S$ and let t_2 be the transition which has the second least f-value among all transitions in S and so on. From the conditions, it follows that there is a path $r_1, r_2, \ldots, r_\ell$ in $\mathcal{G}_\mathcal{M}$ from p to p using only the transitions from S such that

- For every counter y and every transition r_i, if $[\![r_i]\!]_y^-$ is non-empty, then

$$[\![r_i]\!]_y^- \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{1 \leq j < i} [\![r_j]\!]^+ \right) \neq \emptyset$$

- For every counter $y \in Y$, there exists some transition r_i such that

$$[\![r_i]\!]_y^p \bigcap \left([\![\mathbf{u}]\!] \cup \bigcup_{1 \leq j \leq \ell} [\![r_j]\!]^+ \right) \neq \emptyset$$

Applying Lemma D.3 then finishes the proof. □

E Proofs of Subsection 4.3

We show that the reachability problem for C-continuous VASS is in NEXP if C contains only permutation matrices. We do this by giving an exponential-time reduction to the reachability problem for continuous VASS which is in NP [7, Corollary 4.10].

To this end, let C be a non-negative class which contains only permutation matrices. Let $\mathcal{M} = (Q, T)$ be a d-dimensional C-continuous VASS for some d. Since all matrices in C are non-negative, without loss of generality, we can assume that if $(p, \mathbf{A}, \mathbf{b}, q) \in T$ then either $\mathbf{b} = \mathbf{0}$ or $\mathbf{A} = \mathbf{I}$. Indeed, if there is some transition $(p, \mathbf{A}, \mathbf{b}, q)$ which does not satisfy this assumption, then we add

a new state q' and replace this transition with $(p, \mathbf{A}, \mathbf{0}, q')$ and $(q', \mathbf{I}, \mathbf{b}, q)$. It can be easily verified that this new state preserves the reachability, coverability and state-reachability relation of the given affine VASS.

With this assumption, we now construct a d-dimensional continuous VASS $\mathcal{M}' = (Q', T')$ that simulates $\mathcal{M}$. Intuitively, the idea is that since each affine operation of $\mathcal{M}$ is only a renaming of the counters, we can store this renaming in the states of the continuous VASS. With this intuition, we now define $\mathcal{M}'$.

The set of states Q' of $\mathcal{M}'$ will be $\{q_\sigma : q \in Q, \sigma \in \mathcal{S}_d\}$. Intuitively, each state of $\mathcal{M}'$ stores a state of $\mathcal{M}$ and remembers the current renaming that needs to be done to the counters. For each transition $t \in T$ of $\mathcal{M}$, and each permutation π, there will be a transition $t_\pi \in T'$ of $\mathcal{M}'$ defined as follows:

- If $t = (p, \mathbf{P}_\sigma, \mathbf{0}, q)$, then t_π is defined as $t_\pi = (p_\pi, \mathbf{0}, q_{\sigma \circ \pi})$. Intuitively, if we are remembering that the current renaming of the counters is π, then applying $\mathbf{P}_\sigma$ amounts to remembering that the new renaming of the counters is $\sigma \circ \pi$.
- If $t = (p, \mathbf{I}, \mathbf{b}, q)$, then t_π is defined as $t_\pi = (p_\pi, \mathbf{P}_\pi \mathbf{b}, q_\pi)$. Intuitively, if we are remembering that the current renaming of the counters is π, then adding $\mathbf{b}$ in $\mathcal{M}$ amounts to adding $\mathbf{P}_\pi \mathbf{b}$ in $\mathcal{M}'$.

A very easy induction then shows that

$p(\mathbf{u})$ can reach $q(\mathbf{v})$ in $\mathcal{M}$ if and only if for some permutation σ, $p_{id}(\mathbf{u})$ can reach $q_\sigma(\mathbf{P}_\sigma \mathbf{v})$ in $\mathcal{M}'$ where id is the identity permutation.

Our algorithm for solving reachability in $\mathcal{M}$ is then as follows: Construct $\mathcal{M}'$, guess a permutation and check whether $p_{id}(\mathbf{u})$ can reach $q_\sigma(\mathbf{P}_\sigma \mathbf{v})$ in $\mathcal{M}'$. Since $\mathcal{M}'$ can be constructed in exponential time and since reachability in continuous VASS is in NP, it follows that reachability of $\mathcal{M}'$ can be decided in NEXP.

F   Proofs of Subsection 5.1

Here we will prove that state-reachability for continuous VASS with resets is PSPACE-hard. To prove this, we give a reduction from state-reachability for Boolean programs. Intuitively, a Boolean program is a finite-state automaton which has access to d Boolean variables (for some d), which it can either set to 0 or 1 and which can be tested for its current value. Formally, a d-Boolean program is a tuple $\mathcal{M} = (Q, T)$ where Q is a finite set of control states and $T \subseteq Q \times \{\text{test}_j(i), \text{set}_j(i) : 1 \leq i \leq d, j \in \{0, 1\}\} \times Q$ is a finite set of transitions. A configuration of $\mathcal{M}$ is a tuple $p(\mathbf{u})$ where $\mathbf{u} \in \{0, 1\}^d$. Given a transition $t = (p, a, q)$ and two configurations $p(\mathbf{u})$ and $q(\mathbf{v})$, we say that there is a step from $p(\mathbf{u})$ to $q(\mathbf{v})$ by means of the transition t, denoted by $p(\mathbf{u}) \xrightarrow{t} q(\mathbf{v})$ if the following holds:

- If $a = \text{test}_j(i)$, then $\mathbf{u}(i) = j$ and $\mathbf{v} = \mathbf{u}$
- If $a = \text{set}_j(i)$, then $\mathbf{v}(i) = j$ and $\mathbf{u}(k) = \mathbf{v}(k)$ for every $k \neq i$.

The notion of runs, reachability and state-reachability can then be easily defined for Boolean programs. It is known that the state-reachability problem for Boolean programs is PSPACE-hard [17]. We now give a reduction from the state-reachability problem for Boolean programs to the state-reachability problem for continuous VASS with resets.

Suppose $\mathcal{M} = (Q, T)$ is a d-Boolean program. We will construct a $2d$-continuous VASS with resets $\mathcal{M}' = (Q', T')$ that will simulate $\mathcal{M}$. Intuitively, for each Boolean variable i of $\mathcal{M}$, we will have two counters i and $d + i$ in $\mathcal{M}'$. Exactly one of these two counters will be non-zero at any given point. If counter i is non-zero, then this corresponds to the Boolean variable i being true in $\mathcal{M}$ and if counter $d + i$ is non-zero, then this corresponds to i being false in $\mathcal{M}$. Tests and sets on Boolean variables can be accomplished by appropriate increments/decrements and resets on counters in $\mathcal{M}'$.

Recall that for any $1 \leq i \leq 2d$, the matrix $\mathbf{R}^{(2d,i)}$ is the $2d \times 2d$ diagonal matrix over $\{0, 1\}$ which is 1 everywhere on its diagonal, except in the ith position, where it is a 0. Intuitively, multiplying by $\mathbf{R}^{(2d,i)}$ corresponds to resetting the value of the ith counter of $\mathcal{M}'$ and not changing the values of the other counters. With this in mind, we now describe the states and transitions of $\mathcal{M}'$.

Formally, the set of states Q' of $\mathcal{M}'$ is $Q \cup \{q_t : t \in T\}$. Corresponding to each transition t of $\mathcal{M}$, we will have two transitions t^s, t^e in $\mathcal{M}'$ in the following way:

- If $t = (p, \text{test}_j(i), p')$ for some $1 \leq i \leq d$ and some $j \in \{0, 1\}$, then $t^s = (p, \mathbf{I}, -\mathbf{e}_{dj+i}, q_t)$ and $t^e = (q_t, \mathbf{I}, \mathbf{e}_{dj+i}, p')$. Intuitively, if we want to test that the value of the ith Boolean variable is j, then we first decrement the counter $dj+i$ by some value and then increment it again. Recall that the Boolean variable i having value j in $\mathcal{M}$ corresponds to the counter $dj + i$ having a non-zero value in $\mathcal{M}'$ and these transitions check exactly that criterion.
- If $t = (p, \text{set}_j(i), p')$ for some $1 \leq i \leq d$ and some $j \in \{0, 1\}$, then $t^s = (p, \mathbf{R}^{(2d, d(1-j)+i)}, \mathbf{e}_{dj+i}, q_t)$ and $t^e = (q_t, \mathbf{I}, \mathbf{0}, p')$. Intuitively, setting the value of the Boolean variable i to j corresponds to first resetting the value of the counter $d(1 - j) + i$ to zero and then ensuring that the value of the counter $dj + i$ is non-zero.

This completes the description of $\mathcal{M}'$. A configuration $p(\mathbf{u})$ of $\mathcal{M}'$ is called good if $p \in Q$ and for every $1 \leq i \leq d$, exactly one of the counters i and $d + i$ have a non-zero value. Each good configuration C of $\mathcal{M}'$ is easily seen to correspond to a unique configuration $\mathbb{B}(C)$ of $\mathcal{M}$. It can then be easily seen that if $C \xrightarrow{\alpha t^s, \beta t^e} C'$ is a run in $\mathcal{M}'$ from some good configuration C then C' is also a good configuration and moreover this run can happen if and only if $\mathbb{B}(C) \xrightarrow{t} \mathbb{B}(C')$ is a step in $\mathcal{M}$. This then allows for a straightforward reduction from the state-reachability problem for $\mathcal{M}$ to the state-reachability problem for $\mathcal{M}'$.

G   Proofs of Subsection 5.2

Let C contain a non-trivial permutation matrix $\mathbf{P}_\sigma$ (of dimension $k \times k$ for some k). Since $\mathbf{P}_\sigma$ is not the identity matrix, there exists a sequence of distinct indices $i_1, i_2, \ldots, i_\ell$ with $\ell \geq 2$ such that σ maps i_1 to i_2, i_2 to i_3 and so on and finally maps i_ℓ to i_1. Let $z = i_1$. Further, let $n \geq 2$ be the least number such that $\mathbf{P}_\sigma^n = \mathbf{I}$. Note that since $\mathbf{P}_\sigma$ is a permutation matrix, such a number must exist.

Now, suppose $\mathbf{u}$ is a vector in $\mathbb{Q}_+^k$ such that $\mathbf{u}(j) = 0$ for every $j \neq z$. Then,

$$\mathbf{P}_\sigma \cdot \mathbf{u} = \mathbf{v} \text{ where } \mathbf{v}(j) = 0 \text{ for every } j \neq \sigma(z) \qquad (13)$$

Similarly, suppose $\mathbf{u}$ is a vector in $\mathbb{Q}_+^k$ such that $\mathbf{u}(j) = 0$ for every $j \neq \sigma(z)$. Then,

$$\mathbf{P}_\sigma^{n-1} \cdot \mathbf{u} = \mathbf{v} \text{ where } \mathbf{v}(j) = 0 \text{ for every } j \neq z \qquad (14)$$

Intuitively, $\mathbf{P}_\sigma$ transfers the value of z to $\sigma(z)$ and does nothing to the other counters, provided the other counters all have the value 0. Similarly, $\mathbf{P}_\sigma^{n-1}$ transfers the value of $\sigma(z)$ to z and does nothing to the other counters, provided the other counters all have the value 0. We will use these equations to prove our desired hardness results.

G.1 PSPACE lower bound for state-reachability

First, we will show that the state-reachability problem for C-continuous VASS is PSPACE-hard by giving a reduction from the state-reachability problem for Boolean programs. To this end, let $\mathcal{M} = (Q, T)$ be a d-Boolean program. We will construct a dk-dimensional C-continuous VASS which will simulate $\mathcal{M}$. The counters $x_1 = z, x_2 = z + k, \ldots, x_d = z + (d-1)k$ will be called the positive primary counters of $\mathcal{M}'$, the counters $\overline{x_1} = \sigma(z), \overline{x_2} = \sigma(z)+k, \ldots, \overline{x_d} = \sigma(z)+(d-1)k$ will be called the negative secondary counters of $\mathcal{M}'$ and the remaining counters will be called dummy counters, which will always have the value zero. The intuition here is that the Boolean variable i of $\mathcal{M}$ will be simulated by the positive primary counter x_i and the negative primary counter $\overline{x_i}$ in $\mathcal{M}'$. At any point, we will have the following invariant in $\mathcal{M}'$: Exactly one of the counters x_i and $\overline{x_i}$ will have a non-zero value. If x_i (resp. $\overline{x_i}$) has a non-zero value, then this corresponds to the Boolean variable i being true (resp. false).

Note that with this invariant, tests on the Boolean variable i in $\mathcal{M}$ can be simulated by appropriate increments/decrements on the primary counters x_i and $\overline{x_i}$. On the other hand, to simulate set transitions on the Boolean variable i in $\mathcal{M}'$, we will rely on equations 13 and 14. The idea is that if we want to set the Boolean variable i to 1 in $\mathcal{M}$, then we have to make the value of x_i to be non-zero and the value of $\overline{x_i}$ to be zero. We first check which of these current values is non-zero, by non-deterministically decrementing either x_i or $\overline{x_i}$. Note that exactly one of these choices can succeed, thanks to our invariant. If we successfully decrement x_i, then it turns out that the original value of x_i was already non-zero and so we simply have to increment x_i, in order to fully simulate the set transition on the Boolean variable i. On the other hand, suppose we decrement $\overline{x_i}$. Then this means that originally the value of $\overline{x_i}$ was non-zero and the value of x_i was zero. We then increment $\overline{x_i}$ and then transfer the value of $\overline{x_i}$ to x_i by multiplying with $\text{Apply}_{dk}(\mathbf{P}_\sigma^{n-1}, (i-1)k+1)$. By the definition of the Apply operation and equation 14, it follows that the value of x_i is now non-zero, the value of $\overline{x_i}$ is zero and the value of every dummy counter is 0 and so we have successfully simulated a set transition to the value 1. Similarly by using equation 13, we can simulate a set transition to the value 0.

Formally, corresponding to each transition t of $\mathcal{M}$, we will construct transitions in $\mathcal{M}'$ as follows:

- If $t = (p, \text{test}_j(i), p')$, then we construct two transitions $t^s = (p, \mathbf{I}_{dk}, -\mathbf{e}_x, q_t)$, $t^e = (q_t, \mathbf{I}_{dk}, \mathbf{e}_x, p')$ where $x = x_i$ if $j = 1$ and $x = \overline{x_i}$ if $j = 0$. Note that this construction is very similar to the one given for continuous VASS with resets. We test that the value of the Boolean variable i
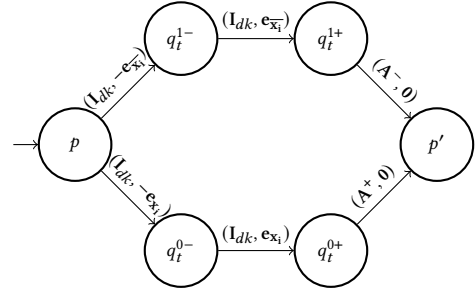

Figure 1: Gadget corresponding to the transition $t = (p, \text{set}_j(i), p')$. $\mathbf{A}^+ = \mathbf{I}$ and $\mathbf{A}^- = \text{Apply}_{dk}(\mathbf{P}_\sigma^{n-1}, (i-1)k+1)$ if $j = 1$ and $\mathbf{A}^- = \mathbf{I}$ and $\mathbf{A}^+ = \text{Apply}_{dk}(\mathbf{P}_\sigma, (i-1)k+1)$ if $j = 0$.

is j by decrementing and incrementing the value of the appropriate primary counter.
- If $t = (p, \text{set}_j(i), p')$, then we construct the gadget in Figure 1. Note that in the figure $\mathbf{A}^+ = \mathbf{I}_{dk}$ and $\mathbf{A}^- = \text{Apply}_{dk}(\mathbf{P}_\sigma^{n-1}, (i-1)k+1)$ if $j = 1$ and $\mathbf{A}^- = \mathbf{I}_{dk}$ and $\mathbf{A}^+ = \text{Apply}_{dk}(\mathbf{P}_\sigma, (i-1)k+1)$ if $j = 0$. The intuition behind this gadget was already discussed in the previous paragraphs.

A configuration $p(\mathbf{u})$ of $\mathcal{M}'$ is called good if p is a state of $\mathcal{M}$ and for each i, exactly one of the counters x_i and $\overline{x_i}$ has a non-zero value and all the dummy counters have value 0. It is easy to see that every good configuration C of $\mathcal{M}'$ uniquely corresponds to a configuration $\mathbb{B}(C)$ of $\mathcal{M}$. By our construction it can be verified that if C is a good configuration of $\mathcal{M}'$ and $C \xrightarrow{\alpha t^s, \beta t^e} C'$ is a run in $\mathcal{M}'$ where t is a test transition of $\mathcal{M}$, then C' is a good configuration and moreover this run can happen if and only if $\mathbb{B}(C) \xrightarrow{t} \mathbb{B}(C')$ is a step in $\mathcal{M}$. Further, a very similar claim can also be verified when t is a set transition, which sets the value of some Boolean variable i to either 0 or 1. This then allows us to conclude that state-reachability for $\mathcal{M}$ can be reduced to state-reachability for $\mathcal{M}'$, thereby deriving the desired PSPACE lower bound.

G.2 NEXPTIME lower bound for coverability and reachability

Now, we move on to proving NEXP lower bounds for the coverability and reachability problems for C-continuous VASS. In the previous paragraphs, we have shown that C-continuous VASS can simulate Boolean programs. We can enrich the model of Boolean programs by allowing continuous counters to get the model of continuous Boolean programs where, in addition to a set of Boolean variables we are also given access to a set of continuous counters. From [4], it is known that the coverability and reachability problems for continuous Boolean programs are NEXP-hard. The reduction given in the previous subsection then allows us to derive a NEXP lower bound for the coverability and reachability problems for C-continuous VASS in a straightforward manner.