Linear Equations with Ordered Data

Piotr Hofman[1]
University of Warsaw, ul. Banacha 2, 02-097 Warsaw, Poland
piotrek.hofman@gmail.com
https://orcid.org/0000-0001-9866-3723

Sławomir Lasota[2]
University of Warsaw, ul. Banacha 2, 02-097 Warsaw, Poland
sl@mimuw.edu.pl
https://orcid.org/0000-0001-8674-4470

Abstract

Following a recently considered generalization of linear equations to *unordered data* vectors, we perform a further generalization to *ordered data* vectors. These generalized equations naturally appear in the analysis of vector addition systems (or Petri nets) extended with ordered data. We show that nonnegative-integer solvability of linear equations is computationally equivalent (up to an exponential blowup) to the reachability problem for (plain) vector addition systems. This high complexity is surprising, and contrasts with NP-completeness for unordered data vectors. This also contrasts with our second result, namely polynomial time complexity of the solvability problem when the nonnegative-integer restriction on solutions is relaxed.

2012 ACM Subject Classification Theory of computation → Parallel computing models, Theory of computation → Timed and hybrid models, Theory of computation → Automata over infinite objects

Keywords and phrases Linear equations, Petri nets, Petri nets with data, vector addition systems, sets with atoms, orbit-finite sets

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2018.24

Related Version All the missing proofs are to be found in the full version of this paper, at [10], `https://arxiv.org/abs/1802.06660`.

Acknowledgements We thank anonymous reviewers for their detailed comments.

1 Introduction

Systems of linear equations are useful for approximate analysis of vector addition systems (VAS), or Petri nets. For instance, the relaxation of semantics of Petri nets, where the configurations along a run are not required to be nonnegative, yields the so called *state equation*, a system of linear equations with nonnegative-integer restriction on solutions. This is equivalent to *integer linear programming*, a well-known NP-complete problem [13]. When the nonnegative-integer restriction is further relaxed to nonnegative-rational one (or nonnegative-real one), we get a weaker but more tractable approximation, equivalent to *linear*

[1] Partially supported by Polish NCN grant 2016/21/D/ST6/01368.
[2] Partially supported by Polish NCN grant 2016/21/B/ST6/01505.

29th International Conference on Concurrency Theory (CONCUR 2018).
Editors: Sven Schewe and Lijun Zhang; Article No. 24; pp. 24:1–24:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

programming and solvable in PTime. We refer to [24] for an exhaustive overview of linear-algebraic and integer-linear-programming techniques in analysis of Petri nets; usefulness of these techniques is confirmed by multiple applications including, for instance, recently proposed efficient tools for the coverability problem of Petri nets [9, 1].

Motivations. Our starting point is an extension of Petri nets, or VAS, with data [17, 11]. The extension significantly enhances expressibility of the model but also increases the complexity of analysis. In case of *unordered* data (a countable set of data values that can be tested for equality only), the coverability problem is decidable (in non-elementary complexity) [17] but the decidability status of the reachability problem remains open. In case of *ordered* data (a countable dense total order), the coverability problem is still decidable [17] while reachability is not [23]. (Petri nets with ordered data are equivalent to a timed extension of Petri nets, as shown in [5].) One can also consider other data domains, and the coverability problem remains decidable as long as the data domain is homogeneous [16] (not to be confused with *homogeneous* systems of linear equations), but always in non-elementary complexity. In view of these high complexities, a natural need arises for efficient over-approximations.

A configuration of a Petri net with data domain $\mathbb{D}$ is a nonnegative integer *data vector*, i.e., a function $\mathbb{D} \to \mathbb{N}^d$ that maps only finitely many data values to a non-zero vector in $\mathbb{N}^d$. In a search for efficient over-approximations of Petri nets with data, a natural question appears: May linear algebra techniques be generalised so that the role of vectors is played by data vectors? In case of unordered data, this question was addressed in [12], where first promising results have been shown: the nonnegative-integer solvability of linear equations over unordered data domain is NP-complete. Thus, for unordered data, the problem remains within the same complexity class as its plain (data-less) counterpart. The same question for the second most natural data domain, i.e. ordered data, seems to be even more important; ordered data enables modelling features like fresh names creation [23] or time dependencies [5].

Contributions. In this paper we do a further step and investigate linear equations with ordered data, for which we fully characterise the complexity of the solvability problem. Firstly, we show that nonnegative-integer solvability of linear equations is computationally equivalent (up to an exponential blowup in one direction) with the reachability problem for plain Petri nets (or VAS). In consequence, decidability and ExpSpace-hardness of our problem follows. This high complexity is surprising, and contrasts NP-completeness for unordered data vectors.

Secondly, we prove that the complexity of the solvability problem drops to PTime, when the nonnegative-integer restriction on solutions is relaxed to rational, nonnegative-rational, or integer. The two latter problems may be thus used as two tractable but incomparable over-approximations of the reachability relation for VAS-es with ordered data. Thirdly, as a conceptual contribution we notice that systems of linear equations with (un)ordered data are a special case of systems of linear equations that are infinite but finite up to an automorphism of data domain. This can be formalized in the framework of *sets with atoms* [3, 4, 14], where finiteness is relaxed to *orbit-finiteness*.

Outline. In Section 2 we introduce the setting we work in, and formulate our results. Then the rest of the paper is devoted to proofs. First, in Section 3 we provide a lower bound for the nonnegative-integer solvability problem, by a reduction from the VAS reachability problem. Then, in Section 4 we suitably reformulate our problem in terms *multihistograms*, which are matrices satisfying certain combinatorial property. This reformulation is used in the next Section 5 to provide a reduction from the nonnegative-integer solvability problem to the VAS reachability problem, thus proving decidability of our problem. In Section 6 we investigate the relaxations of the nonnegative-integer restriction on solutions and work out a PTime decision procedure in each case. In the concluding Section 7 we sketch upon a generalised setting of orbit-finite systems of linear equations.

2 Vector addition systems and linear equations

In this section we introduce the setting of linear equations with data, and formulate our results. For a gentle introduction of the setting, we start by recalling classical linear equations.

Let $\mathbb{Q}$ denote the set of rationals, and $\mathbb{Q}_+, \mathbb{Z}$, and $\mathbb{N}$ denote the subsets of nonnegative rationals, integers, and nonnegative integers. Classical linear equations are of the form

$$a_1 x_1 + \ldots + a_m x_m = a,$$

where $x_1 \ldots x_m$ are variables (unknowns), and $a_1 \ldots a_m \in \mathbb{Z}$ are integer coefficients (equivalently, rational coefficients could be allowed). For a finite system $\mathcal{U}$ of such equations over the same variables $x_1, \ldots, x_m$, a solution of $\mathcal{U}$ is a vector $(n_1, \ldots, n_m) \in \mathbb{Q}^m$ such that the valuation $x_1 \mapsto n_1, \ldots, x_m \mapsto n_m$ satisfies all equations in $\mathcal{U}$. In the sequel we are most often interested in nonnegative integer solutions $(n_1, \ldots, n_m) \in \mathbb{N}^m$, but one may consider also other solution domains than $\mathbb{N}$. It is well known that the *nonnegative-integer solvability problem* ($\mathbb{N}$-solvability problem) of linear equations, i.e. the question whether $\mathcal{U}$ has a nonnegative-integer solution, is NP-complete (for hardness see [13]; NP-membership is a consequence of [21]). The complexity remains the same for other natural variants of this problem, for instance for inequalities instead of equations (a.k.a. integer linear programming). On the other hand, for any $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}, \mathbb{Q}_+\}$, the $\mathbb{X}$-*solvability problem*, i.e., the question whether $\mathcal{U}$ has a solution $(n_1, \ldots, n_m) \in \mathbb{X}^m$, is decidable in PTime.

The $\mathbb{X}$-solvability problem is equivalently formulated as follows: for a given finite set of coefficient vectors $A = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\} \subseteq_{\text{fin}} \mathbb{Z}^d$ and a target vector $\mathbf{a} \in \mathbb{Z}^d$ (we use bold fonts to distinguish vectors from other elements), check whether $\mathbf{a}$ is an $\mathbb{X}$-*sum* of A, i.e., a sum of the following form $\mathbf{a} = n_1 \cdot \mathbf{a}_1 + \ldots + n_m \cdot \mathbf{a}_m$ for some $n_1, \ldots, n_m \in \mathbb{X}$. The dimension d corresponds to the number of equations in $\mathcal{U}$.

Linear equations may serve as an over-approximation of the reachability set of a Petri net, or equivalently, of a *vector addition system* – we prefer to work with the latter model. *A vector addition system (VAS)* $\mathcal{A} = (A, \mathbf{i}, \mathbf{f})$ is defined by a finite set of integer vectors $A \subseteq_{\text{fin}} \mathbb{Z}^d$ together with two nonnegative integer vectors $\mathbf{i}, \mathbf{f} \in \mathbb{N}^d$, the initial one and the final one. The set A determines a transition relation $\longrightarrow$ between configurations, that are nonnegative integer vectors $\mathbf{c} \in \mathbb{N}^d$: there is a transition $\mathbf{c} \longrightarrow \mathbf{c}'$ if $\mathbf{c}' = \mathbf{c} + \mathbf{a}$ for some $\mathbf{a} \in A$. The VAS reachability problems asks whether the final configuration is reachable from the initial one by a sequence of transitions (called a *run*), i.e., $\mathbf{i} \longrightarrow^* \mathbf{f}$. We stress that intermediate configurations are required to be nonnegative. The problem is ExpSpace-hard [19] and decidable [20, 15], but nothing is known about its complexity except for the cubic Ackermann upper bound of [18]. For a given VAS, a necessary condition for $\mathbf{i} \longrightarrow^* \mathbf{f}$ is $\mathbb{N}$-solvability of the system of linear equations defined by the set A and the target vector $\mathbf{a} = \mathbf{f} - \mathbf{i}$, called (in case of Petri nets) the *state equation*. For further details we refer the reader to an exhaustive overview of linear-algebraic approximations for Petri nets [24], where both $\mathbb{N}$- and $\mathbb{Q}_+$-solvability problems are considered.

2.1 Vector addition systems and linear equations, with ordered data

The model of VAS, and linear equations, can be naturally extended with data. In this paper we assume that the data domain $\mathbb{D}$ is a countable set, ordered by a dense total order $\leq$ with no minimal nor maximal element. Thus, up to isomorphism, $(\mathbb{D}, \leq)$ is the set of rational numbers with the natural ordering. We call elements of $\mathbb{D}$ *data values*. In the following we use order preserving permutations (called *data permutations* in short) of $\mathbb{D}$, i.e. bijections $\rho : \mathbb{D} \to \mathbb{D}$ such that $x \leq y$ implies $\rho(x) \leq \rho(y)$.

A *data vector* is a function $\mathbf{v} : \mathbb{D} \to \mathbb{Q}^d$ such that the *support*, i.e. the set $supp(\mathbf{v}) \stackrel{\text{def}}{=} \{\alpha \in \mathbb{D} \mid \mathbf{v}(\alpha) \neq \mathbf{0}\}$, is finite (again, we use bold fonts to distinguish data vectors from other elements). The vector addition is lifted to data vectors pointwise: $(\mathbf{v}+\mathbf{w})(\alpha) \stackrel{\text{def}}{=} \mathbf{v}(\alpha)+\mathbf{w}(\alpha)$. A data vector $\mathbf{v}$ is *nonnegative* if $\mathbf{v} : \mathbb{D} \to (\mathbb{Q}_+)^d$, and $\mathbf{v}$ is *integer* if $\mathbf{v} : \mathbb{D} \to \mathbb{Z}^d$. Writing $\circ$ for function composition, we see that $\mathbf{v} \circ \rho$ is a data vector for any data vector $\mathbf{v}$ and any order preserving data permutation $\rho : \mathbb{D} \to \mathbb{D}$. For a set V of data vectors we define

$$\textsc{Orbit}(V) = \{\mathbf{v} \circ \rho \mid \mathbf{v} \in V, \rho \text{ a data permutation}\}.$$

A data vector $\mathbf{x}$ is said to be a *permutation sum* of a finite set of data vectors V if, for some $\mathbf{v_1}, \ldots, \mathbf{v_m} \in \textsc{Orbit}(V)$, not necessarily pairwise different, $\mathbf{x} = \sum_{i=1}^{m} \mathbf{v}_i$. We investigate the following decision problem:

Permutation sum problem.
Input: a finite set V of integer data vectors and an integer data vector $\mathbf{x}$.
Output: is $\mathbf{x}$ a permutation sum of V?

In the special case when the supports of $\mathbf{x}$ and all vectors in V are all singletons, the Permutation sum problem is just $\mathbb{N}$-solvability of linear equations and thus it is trivially NP-hard.

▶ **Proviso 1.** *For complexity estimations we assume binary encoding of numbers appearing in the input to all decision problems discussed in this paper.*

As the first main result, we prove the following inter-reducibility:

▶ **Theorem 1.** *There is a polynomial-time reduction from the VAS reachability problem to the* Permutation sum problem*, and an exponential-time reduction in the opposite direction.*

As a direct consequence, the Permutation sum problem is decidable and ExpSpace-hard. Our setting generalises the setting of *unordered* data, where the data domain $\mathbb{D}$ is *not* ordered, and hence data permutations are all bijections $\mathbb{D} \to \mathbb{D}$. In the case of unordered data the Permutation sum problem is NP-complete, as shown in [12]. The increase of complexity caused by the order in data is thus remarkable.

Similarly as the state equation in the data-less setting, Permutation sum problem may be used as an overapproximation of the reachability in vector addition systems with ordered data, which are defined exactly as ordinary VAS but in terms of data vectors instead of ordinary vectors. A *VAS with ordered data* $\mathcal{V} = (V, \mathbf{i}, \mathbf{f})$ consists of $V \subseteq_{\text{fin}} \mathbb{D} \to \mathbb{Z}^d$ a finite set of integer data vectors, and the initial and final nonnegative integer data vectors $\mathbf{i}, \mathbf{f} \in \mathbb{D} \to \mathbb{N}^d$. The configurations are nonnegative integer data vectors, and the set V induces a transition relation between configurations as follows: $\mathbf{c} \longrightarrow \mathbf{c}'$ if $\mathbf{c}' = \mathbf{c} + \mathbf{v}$ for some $\mathbf{v} \in \textsc{Orbit}(V)$. Similarly as for plain VAS, the reachability problem asks whether the final configuration is reachable from the initial one by a sequence of transitions (called a *run*), i.e., $\mathbf{i} \longrightarrow^* \mathbf{f}$; but contrarily to plain VAS, the problem is undecidable for VAS with ordered data [17]. (The decidability status of the problem for VAS with *unordered* data is unknown.) As long as reachability is concerned, VAS with (un)ordered data are equivalent to Petri nets with (un)ordered data [11].

The PERMUTATION SUM PROBLEM is easily generalised to other domains $\mathbb{X} \subseteq \mathbb{Q}$ of solutions. To this end we introduce scalar multiplication: for $c \in \mathbb{Q}$ and a data vector $\mathbf{v}$ we put $(c \cdot \mathbf{v})(\alpha) \stackrel{\text{def}}{=} c\mathbf{v}(\alpha)$. A data vector $\mathbf{x}$ is said to be a $\mathbb{X}$-*permutation sum* of a finite set of data vectors V if for some $\mathbf{v_1}, \ldots, \mathbf{v_m} \in \text{ORBIT}(V)$ and coefficients $n_1, \ldots, n_m \in \mathbb{X}$,

$$\mathbf{x} = n_1 \cdot \mathbf{v_1} + \ldots + n_m \cdot \mathbf{v_m}.$$

This leads to the following version of the problem, parametrized by the choice of solution domain $\mathbb{X}$ (the PERMUTATION SUM PROBLEM is a particular case, for $\mathbb{X} = \mathbb{N}$):

$\mathbb{X}$-PERMUTATION SUM PROBLEM.
Input: a finite set V of integer data vectors and an integer data vector $\mathbf{x}$.
Output: is $\mathbf{x}$ an $\mathbb{X}$-permutation sum of V?

Our second main result is the following:

▶ **Theorem 2.** *For any* $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}, \mathbb{Q}_+\}$*, the* $\mathbb{X}$-PERMUTATION SUM PROBLEM *is in* PTIME.

For $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}\}$, the above theorem is a direct consequence of a more general fact, where $\mathbb{Q}$ or $\mathbb{Z}$ is replaced by any commutative ring $\mathbb{R}$, under a proviso that data vectors are defined in a more general way, as finitely supported functions $\mathbb{D} \to \mathbb{R}^d$. With this more general notion, we prove that the $\mathbb{R}$-PERMUTATION SUM PROBLEM reduces polynomially to the $\mathbb{R}$-solvability of linear equations with coefficients from $\mathbb{R}$ (cf. Theorem 17 in Section 6.2).

The case $\mathbb{X} = \mathbb{Q}_+$ in Theorem 2 is more involved but of particular interest, as it recalls continuous Petri nets [22, 8] where fractional firings of transitions are allowed. Moreover, faced with the high complexity of Theorem 1, it is expected that Theorem 2 may become a cornerstone of linear-algebraic techniques for VAS with ordered data.

3 Lower bound for the Permutation Sum Problem

In this section we assume all data vectors to be integer data vectors. We are going to show a polynomial-time reduction from the VAS reachability problem to the PERMUTATION SUM PROBLEM. Fix a VAS $\mathcal{A} = (A, \mathbf{i}, \mathbf{f})$. We are going to define a set of data vectors V and a target data vector $\mathbf{x}$ such that the following conditions are equivalent:

C1: $\mathbf{f}$ is reachable from $\mathbf{i}$ in $\mathcal{A}$;
C2: $\mathbf{x}$ is a permutation sum of V.

The set V, to be defined below, will contain only data vectors $\mathbf{v}$ satisfying the following conditions (such data vectors we call *increasing*):

- $\mathbf{v}$ is supported by two data values: $supp(\mathbf{v}) = \{\alpha, \beta\}$ for some data values $\alpha < \beta$;
- $\mathbf{v}(\alpha) \in (-\mathbb{N})^d$ is nonpositive;
- $\mathbf{v}(\beta) \in \mathbb{N}^d$ is nonnegative.

The choice of data values α, β is irrelevant, as we only need to define V up to data permutation. Up to data permutation, the vectors $\mathbf{a} = \mathbf{v}(\alpha)$ and $\mathbf{b} = \mathbf{v}(\beta)$ determine the increasing vector as above uniquely. We thus write $[\mathbf{a}, \mathbf{b}]$ to denote the increasing data vector determined by $\mathbf{a}$ and $\mathbf{b}$ (and some arbitrary but fixed data values $\alpha < \beta$).

Every integer vector $\mathbf{a} \in \mathbb{Z}^d$ is uniquely presented as a sum $\mathbf{a} = \mathbf{a}^- + \mathbf{a}^+$ of a nonnegative vector $\mathbf{a}^+ \in \mathbb{N}^d$ and a nonpositive one $\mathbf{a}^- \in (-\mathbb{N})^d$, defined as follows:

$$\mathbf{a}^+(i) = \begin{cases} \mathbf{a}(i), & \text{if } \mathbf{a}(i) \geq 0 \\ 0, & \text{if } \mathbf{a}(i) < 0 \end{cases} \qquad \mathbf{a}^-(i) = \begin{cases} \mathbf{a}(i), & \text{if } \mathbf{a}(i) \leq 0 \\ 0, & \text{if } \mathbf{a}|(i) > 0. \end{cases}$$

The idea of the reduction is to simulate every vector $\mathbf{a} = \mathbf{a}^- + \mathbf{a}^+ \in A$ by the increasing data vector $[\mathbf{a}^-, \mathbf{a}^+]$, which we call *data realization* of $\mathbf{a}$. In addition, we will need the increasing data vectors of the form $[-\mathbf{1}_i, \mathbf{1}_i]$, where $\mathbf{1}_i \in \mathbb{N}^d$ has 1 on coordinate i and 0 on all other coordinates. We call data vectors $[-\mathbf{1}_i, \mathbf{1}_i]$ *unit increases.* We thus define V as:

$$V \;=\; \{[\mathbf{a}^-, \mathbf{a}^+] \mid \mathbf{a} \in A\} \quad \cup \quad \{[-\mathbf{1}_i, \mathbf{1}_i] \mid i = 1, \ldots, d\}.$$

As the target data vector we take $\mathbf{x} = [-\mathbf{i}, \mathbf{f}]$. It remains to show the equivalence of conditions C1 and C2.

For the proof it will be useful to consider a *VAS with ordered data* $\mathcal{V} = (V, \bar{\mathbf{i}}, \bar{\mathbf{f}})$ (recall the definition in Section 2.1) with the same set of data vectors V, the initial configuration $\bar{\mathbf{i}}$ a data vector supported by one data value, which maps this data value to $\mathbf{i}$, and similarly the final configuration $\bar{\mathbf{f}}$, with the proviso that the singleton support of $\bar{\mathbf{f}}$ is greater than the singleton support of $\bar{\mathbf{i}}$. Clearly, the permutation sum problem overapproximates reachability in $\mathcal{V}$: existence of a run $\bar{\mathbf{i}} \longrightarrow^* \bar{\mathbf{f}}$ in $\mathcal{V}$ implies that $\mathbf{x} = [-\mathbf{i}, \mathbf{f}] = \bar{\mathbf{f}} - \bar{\mathbf{i}}$ is a permutation sum of V. Furthermore, $\mathcal{A}$ also overapproximates $\mathcal{V}$: every run in $\mathcal{V}$ can be transformed into a run in $\mathcal{A}$, by simply getting rid of data in data realizations and dropping all the unit increases.

Condition C1 implies condition C2. Indeed, every run $\mathbf{i} \longrightarrow^* \mathbf{f}$ in $\mathcal{A}$ can be transformed into a run $\bar{\mathbf{i}} \longrightarrow^* \bar{\mathbf{f}}$ in $\mathcal{V}$: replace every vector $\mathbf{a} \in A$ appearing in the former run with its data realization $[\mathbf{a}^-, \mathbf{a}^+] \circ \theta$ (for a suitably chosen data permutation θ), preceded, if necessary, by a number of unit increases of the form $[-\mathbf{1}_i, \mathbf{1}_i] \circ \theta$ (again, for suitably chosen θ), in order to gather, intuitively speaking, the whole vector $\mathbf{a}^-$ at the same data value. Then C2 follows by the overapproximation of reachability in $\mathcal{V}$ by the permutation sum problem.

For the converse implication suppose C2 holds, i.e., $\mathbf{x} = \sum_{i=1}^{n} \mathbf{w}_i$, where $\mathbf{w}_i = \mathbf{v}_i \circ \theta_i$ and $\mathbf{v}_i \in V$. By construction of V, for every $i \leq n$ the data vector $\mathbf{v}_i$ is either a data realization of some $\mathbf{a} \in A$, or a unit increase. Let $\mathbf{v}_{i_1}, \ldots, \mathbf{v}_{i_l}$ denote the subsequence of $\mathbf{v}_1, \ldots, \mathbf{v}_n$ containing the former ones. We claim that the corresponding vectors $\mathbf{a}_1 \ldots \mathbf{a}_l$, of which $\mathbf{v}_{i_1}, \ldots, \mathbf{v}_{i_l}$ are data realizations, can be arranged into a sequence being a correct run of the VAS $\mathcal{A}$ from $\mathbf{i}$ to $\mathbf{f}$. For this purpose we define a binary relation of *succession* on data vectors $\mathbf{w}_i$: we say that $\mathbf{w}_j$ succeeds $\mathbf{w}_i$ if $\max(\mathit{supp}(\mathbf{w}_i)) \leq \mathit{min}(\mathit{supp}(\mathbf{w}_j))$. We observe that the succession relation is a partial order – indeed, antisymmetry follows due to the fact that all data vectors $\mathbf{w}_i$ are increasing. Let $\prec$ denote an arbitrary extension of the partial order to a total order, and assume w.l.o.g. that $\mathbf{w}_1 \prec \mathbf{w}_2 \prec \ldots \prec \mathbf{w}_n$. We argue that the corresponding sequence $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_l$ of vectors from A is a correct run of the VAS $\mathcal{A}$ from $\mathbf{i}$ to $\mathbf{f}$. As $\mathcal{A}$ overapproximates $\mathcal{V}$, it is enough to demonstrate that the sequence $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_n$ is a correct run in $\mathcal{V}$ from $\bar{\mathbf{i}}$ to $\bar{\mathbf{f}}$. We thus need to prove that the data vector $\mathbf{u}_i = \bar{\mathbf{i}} + \sum_{i=1}^{j} \mathbf{w}_i$ is nonnegative for every $j \in \{0, \ldots, n\}$. To this aim fix $\alpha \in \mathbb{D}$ and $l \in \{1, \ldots, d\}$, and consider the sequence of numbers

$$\mathbf{u}_0(\alpha, l), \quad \mathbf{u}_1(\alpha, l), \quad \ldots \quad \mathbf{u}_n(\alpha, l) \tag{1}$$

appearing as the value of the consecutive data vectors $\mathbf{u}_0$, $\mathbf{u}_1$, …, $\mathbf{u}_n$ at data value α and coordinate l. We know that the first element of the sequence $\mathbf{u}_0(\alpha, l) = \bar{\mathbf{i}}(\alpha, l) \geq 0$ and the last element of the sequence $\mathbf{u}_n(\alpha, l) = \bar{\mathbf{f}}(\alpha, l) \geq 0$. Furthermore, by the definition of the ordering $\preceq$ we know that the sequence (1) is first non-decreasing, and then non-increasing. These conditions imply nonnegativeness of all numbers in the sequence.

4 Histograms

The purpose of this section is to transform the Permutation sum problem to a more manageable form. As the first step, we eliminate data by rephrasing the problem in terms of matrices (in Lemma 4). Then, we distinguish matrices with certain combinatorial property, called *histograms*. Finally, in Lemma 12 we provide a final characterisation of the problem, using *multihistograms*. The characterisation will be crucial for effectively solving the Permutation sum problem in Section 5.

▶ **Proviso 2.** *In this section, all matrices are integer ones, and all data vectors are integer ones.*

Eliminating data. Matrices with r rows and c columns we call $r\times c$-matrices, and r (resp. c) we call row (resp. column) dimension of an $r\times c$-matrix. We are going to represent a data vector $\mathbf{v} \in \mathbb{D} \to \mathbb{Z}^d$ as a $d\times|supp(\mathbf{v})|$-matrix $M_{\mathbf{v}}$ as follows: if $supp(\mathbf{v}) = \{\alpha_1 < \alpha_2 < \ldots < \alpha_n\}$, we put $M_{\mathbf{v}}(i,j) \stackrel{\text{def}}{=} \mathbf{v}(i)(\alpha_j)$. A *0-extension* of an $r\times c$-matrix M is any $r\times c'$-matrix M', $c' \geq c$, obtained from M by inserting into M arbitrarily $c' - c$ additional zero columns $\mathbf{0} \in \mathbb{Z}^r$. Thus row dimension is preserved by 0-extension, and column dimension may grow arbitrarily. We denote by 0-ext(M) the (infinite) set of all 0-extensions of a matrix M. In particular, $M \in$ 0-ext(M). For a set $\mathcal{M}$ of matrices we denote by 0-ext$(\mathcal{M})$ the set of all 0-extensions of all matrices in $\mathcal{M}$.

▶ **Example 3.** For a data vector $\mathbf{v}$ with $supp(\mathbf{v}) = \{\alpha_1 < \alpha_2\}$, $\mathbf{v}(\alpha_1) = (1,3,0) \in \mathbb{Z}^3$ and $\mathbf{v}(\alpha_2) = (2,0,2) \in \mathbb{Z}^3$, here is the corresponding matrix and two of possible 0-extension:

$$M_{\mathbf{v}} = \begin{bmatrix} 1 & 2 \\ 3 & 0 \\ 0 & 2 \end{bmatrix} \qquad \begin{bmatrix} 0 & 1 & 2 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 2 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \in \text{0-ext}(M_{\mathbf{v}}).$$

Below, whenever we add matrices we silently assume that they have the same row and column dimensions. For a finite set $\mathcal{M}$ of matrices, we say that a matrix N is a sum of 0-extensions of $\mathcal{M}$ if

$$N = M_1 + \ldots + M_m \tag{2}$$

for some matrices $M_1, \ldots, M_m \in$ 0-ext$(\mathcal{M})$, necessarily all of the same row and column dimension. We claim that the Permutation sum problem is equivalent to the question whether some 0-extension of a given matrix X is a sum of 0-extensions of $\mathcal{M}$.

Up to 0-extension sum problem.
Input: a finite set $\mathcal{M}$ of matrices, and a matrix X, all of the same row dimension d.
Output: is some 0-extension of X a sum of 0-extensions of $\mathcal{M}$?

▶ **Lemma 4.** *The* Permutation sum problem *is polynomially time equivalent to the* Up to 0-extension sum problem.

Histograms. From now on we concentrate on solving the Up to 0-extension sum problem. For a matrix H, we write $\sum H(i, 1\ldots j)$ instead of $\sum_{1\leq l\leq j} H(i,l)$. In particular, $\sum H(i, 1\ldots 0) = 0$ by convention. We call an integer matrix nonnegative if it only contains nonnegative integers. Histograms, to be defined now, are an adaptation (a strengthening) of histograms of [12] to ordered data.

▶ **Definition 5.** A nonnegative integer $r{\times}c$-matrix H we call a *histogram* if the following conditions are satisfied:

- for some $s \geq 0$, called the *degree* of H, for every $1 \leq i \leq r$ we have $\sum H(i, 1 \ldots c) = s$;
- for every $1 \leq i < r$ and $0 \leq j < c$, we have $\sum H(i, 1 \ldots j) \geq \sum H(i+1, 1 \ldots j+1)$.

Note that the zero matrix is a histogram, for $s = 0$. If $s > 0$, the definition enforces $r \leq c$. Histograms of degree 1 are called *simple*. The following combinatorial property of histograms will be crucial in the sequel:

▶ **Lemma 6.** *H is a histogram of degree $s > 0$ if and only if H is a sum of s simple histograms.*

▶ **Example 7.** A histogram of degree 2 may be decomposed as a sum of two simple histograms:

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Below, whenever we multiply matrices we assume that the column dimension of the first one is the same as the row dimension of the second one. Simple histograms are useful for characterising 0-extensions:

▶ **Lemma 8.** *For matrices N and M, $N \in$ 0-ext(M) if and only if $N = M \cdot S$, for a simple histogram S.*

▶ **Example 9.** Recall the matrix $M = M_{\mathbf{v}}$ from Example 3. One of the matrices from 0-ext(M) is presented as the multiplication of M and a simple histogram as follows:

$$\begin{bmatrix} 1 & 0 & 2 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 2 \\ 3 & 0 \\ 0 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

We use Lemmas 6 and 8 to characterise the Up to 0-extension sum problem:

▶ **Lemma 10.** *For a matrix N and a finite set of matrices $\mathcal{M}$, the following conditions are equivalent:*

1. *N is a sum of 0-extensions of $\mathcal{M}$;*
2. *$N = \sum_{M \in \mathcal{M}} M \cdot H_M$, for some histograms $\{H_M \mid M \in \mathcal{M}\}$.*

Multihistograms. Using Lemma 10 we are now going to work out our final characterisation of the Up to 0-extension sum problem, as formulated in Lemma 12 below. The characterisation will use the notion of *multihistogram*, which is an indexed family $\mathcal{H} = \{H_1, \ldots, H_k\}$ of histograms satisfying Definition 11 below.

We write $H(i, \_)$ and $H(\_, j)$ for the i-th row and the j-th column of a matrix H, respectively. For an indexed family $\{H_1, \ldots, H_k\}$ of matrices, its j-th column is defined as the indexed family of j-th columns of respective matrices $\{H_1(\_, j), \ldots, H_k(\_, j)\}$.

Fix an input of the Up to 0-extension sum problem: a matrix X and a finite set $\mathcal{M} = \{M_1, \ldots, M_k\}$ of matrices, all of the same row dimension d. Let c_l stand for the column dimension of M_l. Relying on Lemma 10, suppose that some $N \in$ 0-ext(X) and some indexed family $\mathcal{H} = \{H_1, \ldots, H_k\}$ of histograms satisfies

$$N = M_1 \cdot H_1 + \ldots + M_k \cdot H_k.$$

(The row dimension of every H_l is necessarily c_l.) Boiling down the equation to entries of a single column $N(\_,j) \in \mathbb{Z}^d$ of N we get the system of d linear equations:

$$N(\_,j) \;=\; M_1 \cdot H_1(\_,j) \;+\; \ldots \;+\; M_k \cdot H_k(\_,j) \;=\; \begin{bmatrix} M_1 \mid \ldots \mid M_k \end{bmatrix} \cdot \begin{bmatrix} H_1(\_,j) \\ \ldots \\ H_k(\_,j) \end{bmatrix}.$$

Therefore, the j-th column of $\mathcal{H}$, treated as a single column vector of length $s = c_1 + \ldots + c_k$, is a nonnegative-integer solution of a system of d linear equations $\mathcal{U}_{\mathcal{M},N(\_,j)}$ with s unknowns $x_1 \ldots x_s$ of the form:

$$N(\_,j) \quad = \quad \begin{bmatrix} M_1 \mid \ldots \mid M_k \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \ldots \\ x_s \end{bmatrix}.$$

Observe that the system $\mathcal{U}_{\mathcal{M},N(\_,j)}$ depends on $\mathcal{M}$ and $N(\_,j)$ but not on j. For succinctness, for $\mathbf{a} \in \mathbb{Z}^d$ we put $\mathcal{C}_\mathbf{a} := \mathbb{N}\text{-sol}(\mathcal{U}_{\mathcal{M},\mathbf{a}})$ to denote the set of all nonnegative integer solutions of $\mathcal{U}_{\mathcal{M},\mathbf{a}}$. Thus every j-th column of $\mathcal{H}$ belongs to $\mathcal{C}_{N(\_,j)}$.

Now recall that $N \in 0\text{-ext}(X)$. Treating $\mathcal{H}$ as a sequence of its column vectors in $\mathbb{N}^s$ we arrive at the following condition:

▶ **Definition 11.** Let *the word* of an indexed family $\mathcal{H} = \{H_1, \ldots, H_k\}$ of histograms be the sequence of its consecutive column vectors. We say that $\mathcal{H}$ is an $(X, \mathcal{M})$-*multihistogram* if its word belongs to the following language (where n is the column dimension of X):

$$(\mathcal{C}_\mathbf{0})^* \; \mathcal{C}_{X(\_,1)} \; (\mathcal{C}_\mathbf{0})^* \; \mathcal{C}_{X(\_,2)} \; \ldots \; (\mathcal{C}_\mathbf{0})^* \; \mathcal{C}_{X(\_,n)} \; (\mathcal{C}_\mathbf{0})^*. \tag{3}$$

We have just shown existence of an $(X, \mathcal{M})$-*multihistogram* whenever some 0-extension N of X is a sum of 0-extensions of $\mathcal{M}$. As the reasoning above is reversible, we obtain:

▶ **Lemma 12.** *The* UP TO 0-EXTENSION SUM PROBLEM *is equivalent to the following one:*

MULTIHISTOGRAM PROBLEM.
Input: *a finite set $\mathcal{M}$ of matrices and a matrix X, all of the same row dimension d.*
Output: *does there exist an $(X, \mathcal{M})$-multihistogram?*

5 Upper bound for the Permutation Sum Problem

We reduce in this section the MULTIHISTOGRAM PROBLEM (and hence also the PERMUTATION SUM PROBLEM, due to Lemmas 4 and 12) to the VAS reachability problem (with single exponential blowup), thus obtaining decidability. Fix in this section an input to the MULTIHISTOGRAM PROBLEM: an integer matrix X (of column dimension n) and a finite set $\mathcal{M} = \{M_1, \ldots, M_k\}$ of integer matrices, all of the same row dimension d. We perform the reduction in two steps: we start by proving an effective exponential bound on vectors appearing as columns of $(X, \mathcal{M})$-multihistograms; then we construct a VAS whose runs correspond to the words of exponentially bounded $(X, \mathcal{M})$-multihistograms.

Exponentially bounded multihistograms. First, we need to recall a characterisation of nonnegative-integer solution sets of systems of linear equations as exponentially bounded hybrid-linear sets, i.e., of the form $B + P^\oplus$, for $B, P \subseteq \mathbb{N}^k$, where k is the number of variables and $P^\oplus$ stands for the set of all finite sums of vectors from P (see e.g. [6, 7, 21]). We denote system of linear equations determined by a matrix M and a column vector $\mathbf{a}$ by $\mathcal{U}_{M,\mathbf{a}}$ and the corresponding *homogeneous* systems of linear equations by $\mathcal{U}_{M,\mathbf{0}}$. Again, for the size $|\mathcal{U}_{M,\mathbf{a}}|$ of $\mathcal{U}_{M,\mathbf{a}}$ we assume that numbers in M and $\mathbf{a}$ are encoded in binary.

▶ **Lemma 13** ([6] Prop. 2). *$\mathbb{N}$-sol$(\mathcal{U}_{M,\mathbf{a}}) = B + P^{\oplus}$, where $B, P \subseteq \mathbb{N}^k$ such that all vectors in $B \cup P$ are bounded exponentially w.r.t. $|\mathcal{U}_{M,\mathbf{a}}|$ and $P \subseteq \mathbb{N}$-sol$(\mathcal{U}_{M,\mathbf{0}})$.*

We will use Lemma 13 together with the following operation on multihistograms. A *j-smear* of a histogram H is any nonnegative matrix H' obtained by replacing j-th column $H(\_, j)$ of H by two columns that sum up to $H(\_, j)$. Here is an example ($j = 5$):

$$\begin{bmatrix} 3 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 2 \end{bmatrix} \to \begin{bmatrix} 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 2 \end{bmatrix}.$$

Formally, a j-smear of H is any nonnegative matrix H' satisfying:

$$H'(\_, l) = H(\_, l) \quad \text{for } l < j \qquad H'(\_, j) + H'(\_, j+1) = H(\_, j)$$
$$H'(\_, l+1) = H(\_, l) \quad \text{for } l > j.$$

One easily verifies that a smear preserves the defining condition of the histogram:

▶ **Claim 5.1.** *A smear of a histogram is a histogram.*

Finally, a j-smear of a family of matrices $\{H_1, \ldots, H_k\}$ is any indexed family of matrices $\{H_1', \ldots, H_k'\}$ obtained by applying a j-smear simultaneously to all matrices H_l. We omit the index j when it is irrelevant.

So prepared, we claim that every $(X, \mathcal{M})$-multihistogram $\mathcal{H} = \{H_1, \ldots, H_k\}$ can be transformed by a number of smears into an $(X, \mathcal{M})$-multihistogram containing only numbers exponentially bounded with respect to the sizes of X, $\mathcal{M}$. Indeed, recall (3) and suppose that $N = M_1 \cdot H_1 + \ldots + M_k \cdot H_k \in$ 0-ext(X). Take an arbitrary (say j-th) column $\mathbf{w} \in \mathcal{C}_{\mathbf{a}} = \mathbb{N}$-sol$(\mathcal{U}_{\mathcal{M},\mathbf{a}})$ of $\mathcal{H}$, where $\mathbf{a} = N(\_, j)$, treated as a single column vector $\mathbf{w} \in \mathbb{N}^s$ (for s the sum of row dimensions of $H_1, \ldots, H_k$), and present it (using Lemma 13) as a sum $\mathbf{w} = \mathbf{b} + \mathbf{p}_1 + \ldots + \mathbf{p}_m$, for some exponentially bounded $\mathbf{b} \in \mathcal{C}_{\mathbf{a}}$ and $\mathbf{p}_1, \ldots, \mathbf{p}_m \in \mathcal{C}_{\mathbf{0}}$. Apply smear m times, replacing the j-th column by $m + 1$ columns $\mathbf{b}, \mathbf{p}_1, \ldots, \mathbf{p}_m$. As $\mathbf{b}$ is a solution of the system $\mathcal{U}_{\mathcal{M},\mathbf{a}}$ and every $\mathbf{p}_l$ is a solution of the homogeneous system $\mathcal{U}_{\mathcal{M},\mathbf{0}}$,

$$\Big[M_1 \mid \ldots \mid M_k\Big] \cdot \mathbf{b} = \Big[M_1 \mid \ldots \mid M_k\Big] \cdot \mathbf{w} \qquad \Big[M_1 \mid \ldots \mid M_k\Big] \cdot \mathbf{p}_l = \mathbf{0},$$

the family $\mathcal{H}' = \{H_1', \ldots, H_k'\}$ obtained in the same way still satisfies the condition $M_1 \cdot H_1' + \ldots + M_k \cdot H_k' \in$ 0-ext(X). Using Claim 5.1 we deduce that $\mathcal{H}'$ is an $(X, \mathcal{M})$-multihistogram. Repeating the same operation for every column of $\mathcal{H}$ yields the required exponential bound.

Construction of a VAS. Given X and $\mathcal{M}$ we now construct a VAS whose runs correspond to the words of exponentially bounded $(X, \mathcal{M})$-multihistograms. Think of the VAS as reading (or nondeterministically guessing) consecutive column vectors (i.e., the word) of a potential $(X, \mathcal{M})$-multihistogram $\mathcal{H} = \{H_1, \ldots, H_k\}$. The VAS has to check two conditions:

(A) the word of $\mathcal{H}$ belongs to the language (3);

(B) the matrices $H_1, \ldots, H_k$ satisfy the histogram condition (cf. Definition 5).

The first condition, under the exponential bound proved above, amounts to the membership in a regular language and can be imposed by a VAS in a standard way. The second condition is a conjunction of k histogram conditions, and again the conjunction can be realised in a standard way. We thus focus, from now on, only on showing that a VAS can check that its input is a histogram.

To this aim it will be profitable to have the following characterisation of histograms. For an arbitrary $r \times c$-matrix H, define the $(r-1)\times c$-matrix Δ_H:

$$\Delta_H(i, j+1) \quad \stackrel{\text{def}}{=} \quad \sum H(i, 1 \ldots j) - \sum H(i+1, 1 \ldots j+1).$$

Intuitively, Δ_H represents the excess in the second condition in Definition 5. Moreover, consider the $(r-1)\times c$-matrix $(\overline{H} + \Delta_H)$, where $\overline{H}$ is H with the last row truncated.

▶ **Lemma 14.** *A nonnegative $r \times c$-matrix H is a histogram if and only if Δ_H is nonnegative and $(\overline{H} + \Delta_H)(\_, c) = \mathbf{0}$.*

Proof. Indeed, nonnegativeness of Δ_H is equivalent to saying that

$$\sum H(i, 1 \ldots j) \ \geq \ \sum H(i+1, 1 \ldots j+1)$$

for every $1 \leq i < r$ and $0 \leq j < c$; moreover, $(\overline{H} + \Delta_H)(\_, c) = \mathbf{0}$ is equivalent to saying that $\sum H(i, 1 \ldots c)$ is the same for every $i = 1, \ldots, r$. ◀

For the construction of a VAS it is important to note that every two consecutive entries $(\overline{H} + \Delta_H)(i, j-1)$ and $(\overline{H} + \Delta_H)(i, j)$ are related by the following formula:

$$(\overline{H} + \Delta_H)(i, j) \ = \ (\overline{H} + \Delta_H)(i, j-1) - H(i+1, j) + H(i, j). \tag{4}$$

Let r be the row dimension of the input matrix, and let $\mathcal{C} \subseteq \mathbb{N}^r$ denote the exponential set of all column vectors that can appear in a histogram, as derived above. We define a VAS of dimension $2(r-1)$ that reads consecutive columns of an exponentially bounded matrix H and accepts if and only if the matrix is a histogram. The VAS transitions will obey the following invariant: after j steps,

$$\text{counter}_i + \text{counter}_{r-1+i} \ = \ (\overline{H} + \Delta_H)(i, j), \qquad \text{for } i = 1, \ldots, r-1. \tag{5}$$

The counters are initially set to 0. Informally, in its j-th step, the VAS will subtract $H(i+1, j)$ from the counter $(r-1)+i$ and simultaneously add $H(i, j)$ to the counter i, for $i = 1 \ldots r-1$, in accordance with (4); due to the duplication of counters, by sole nonnegativeness of every counter $(r-1)+i$ the VAS will thereby check that $\Delta_H(i, j+1) \geq 0$. Formally, for every vector $\mathbf{w} = (w_1, \ldots, w_r) \in \mathcal{C}$, the VAS has a 'reading' transition that adds $(w_1, \ldots, w_{r-1}) \in \mathbb{N}^{r-1}$ to its counters $1, \ldots, r-1$, and subtracts $(w_2, \ldots, w_r) \in \mathbb{N}^{r-1}$ from its counters $r, \ldots, 2(r-1)$ (think of $\mathbf{w}(i) = H(i, j)$ in the equation (4)). Furthermore, for every $i = 1, \ldots, r-1$ the VAS has a 'moving' transition that subtracts 1 from counter i and adds 1 to counter $r-1+i$. Observe that these transitions preserve the invariant (5).

Relying on Lemma 14 we claim that the VAS defined in this way reaches nontrivially (i.e., along a nonempty run) the zero configuration (all counters equal 0) iff its input H is a histogram with all entries belonging to $\mathcal{C}$. In one direction, the nonnegativeness of counters $r \ldots 2(r-1)$ (as discussed above) assures that Δ_H is nonnegative; and the invariant (5) together with the final zero configuration assures that $(\overline{H} + \Delta_H)(\_, c) = \mathbf{0}$. In the opposite direction, if the VAS inputs a histogram, it has a run ending in the zero configuration. The VAS is computable in exponential time (as the set $\mathcal{C}$ above can be computed in exponential time).

Thus, given X and $\mathcal{M}$ one can effectively (in exponential time) build a VAS that admits reachability if and only if there exists an $(X, \mathcal{M})$-multihistogram.

6 PTime decision procedures

In this section we prove Theorem 2, namely we provide polynomial-time decision procedures for the $\mathbb{X}$-Permutation sum problem, where $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}, \mathbb{Q}_+\}$. The most interesting case $\mathbb{X} = \mathbb{Q}_+$ is treated in Section 6.1. The remaining ones are in fact special cases of a more general result, shown in Section 6.2, that applies to an arbitrary commutative ring.

6.1 $\mathbb{X} = \mathbb{Q}_+$

We start by noticing that the whole development of (multi-)histograms in Section 4 is not at all specific for $\mathbb{X} = \mathbb{N}$ and works equally well for $\mathbb{X} = \mathbb{Q}_+$. First, one adapts the Up to 0-extension sum problem and considers a sum of 0-extensions of $\mathcal{M}$ *multiplied by nonnegative rationals*. Accordingly, one relaxes the definition of histogram: instead of a nonnegative integer matrix, let histogram be now a *nonnegative rational* matrix satisfying exactly the same conditions as in Definition 5 in Section 4. In particular, the degree of a histogram is now a nonnegative rational, and simple histograms are these with exactly one nonzero entry in every row. The same relaxation as for histograms we apply to multihistograms, and in the definition of the latter (cf. the language (3) at the end of Section 4) we consider nonnegative-rational solutions of linear equations instead of nonnegative-integer ones. With these adaptations, the $\mathbb{Q}_+$-Permutation sum problem is equivalent to the following decision problem (whenever a risk of confusion arises, we specify explicitly which matrices are integer ones, and which rational ones):

$\mathbb{Q}_+$-Multihistogram problem.
Input: a finite set $\mathcal{M}$ of integer matrices, and an integer matrix X, all of the same row dimension d.
Output: does there exist a rational $(X, \mathcal{M})$-multihistogram?

From now on we concentrate on the polynomial-time decision procedure for this problem. We proceed in two steps. First, we define *homogeneous linear Petri nets*, a variant of Petri nets generalising continuous PN [22], and show how to solve its reachability problem using $\mathbb{Q}_+$-solvability of a slight generalisation of linear equations (linear equations with implications), following the approach of [8]. Next, using a similar construction as in Section 5, combined with the above characterisation of reachability, we encode the $\mathbb{Q}_+$-Multihistogram problem by systems of linear equations with implications.

Homogeneous linear Petri nets. A *homogeneous linear Petri net* (homogeneous linear PN) of dimension d is a finite set of homogeneous[3] systems of linear equations $\mathcal{V} = \{\mathcal{U}_1, \ldots, \mathcal{U}_m\}$, called *transition rules*, all over the same $2d$ variables $x_1, \ldots, x_{2d}$. The transition rules determine a transition relation $\longrightarrow$ between configurations, which are nonnegative rational vectors $\mathbf{c} \in (\mathbb{Q}_+)^d$, as follows: there is a transition $\mathbf{c} \longrightarrow \mathbf{c}'$ if, for some $i \in \{1, \ldots, m\}$ and $\mathbf{v} \in \mathbb{Q}_+\text{-sol}(\mathcal{U}_i)$, the vector $\mathbf{c} - \pi_{1\ldots d}(\mathbf{v})$ is still nonnegative, and

$$\mathbf{c}' = \mathbf{c} - \pi_{1\ldots d}(\mathbf{v}) + \pi_{d+1\ldots 2d}(\mathbf{v}).$$

(The vectors $\pi_{1\ldots d}(\mathbf{v})$ and $\pi_{d+1\ldots 2d}(\mathbf{v})$ are projections of $\mathbf{v}$ on respective coordinates.) The binary reachability relation $\mathbf{c} \longrightarrow^* \mathbf{c}'$ holds, if there is a sequence of transitions from $\mathbf{c}$ to $\mathbf{c}'$.

[3] If non-homogeneous systems were allowed, the model would subsume (ordinary) Petri nets.

A class of *continuous PN* [22] can be seen as a subclass of homogeneous linear PN, where every system $\mathcal{U}_i$ has 1-dimensional solution set of the form $\{c\mathbf{v} \mid c \in \mathbb{Q}_+\}$, for some fixed $\mathbf{v} \in \mathbb{N}^{2d}$.

Linear equations with implications. A $\Rightarrow$*-system* is a finite set of linear equations, all over the same variables, plus a finite set of implications of the form $x > 0 \implies y > 0$, where x, y are variables appearing in the linear equations. The solutions of a $\Rightarrow$-system are defined as usually, but additionally they must satisfy all implications. The $\mathbb{Q}_+$-solvability problem asks if there is a nonnegative-rational solution. In [8] (Algorithm 2) and also in [2] (where a PTime fragment of existential FO($\mathbb{Q}$, + ,<) has been identified that captures $\Rightarrow$-system), it has been shown (within a different notation) how to solve the problem in PTime:

▶ **Lemma 15** ([8, 2]). *The $\mathbb{Q}_+$-solvability problem for $\Rightarrow$-systems is decidable in* PTime*.*

Due to [8], the reachability problem for continuous PNs reduces to the $\mathbb{Q}_+$-solvability of $\Rightarrow$-systems. We generalise this result and prove the reachability relation of a homogeneous linear PN to be effectively described by a $\Rightarrow$-system:

▶ **Lemma 16.** *Given a homogeneous linear PN $\mathcal{V}$ of dimension d one can compute in* PTime *a $\Rightarrow$-system whose $\mathbb{Q}_+$-solution set, projected onto a subset of $2d$ variables, describes the binary reachability relation of $\mathcal{V}$.*

Polynomial-time decision procedure. Now, we are ready to sketch out a decision procedure for the $\mathbb{Q}_+$-Multihistogram problem, by a polynomial-time reduction to the $\mathbb{Q}_+$-solvability problem of $\Rightarrow$-systems.

Fix an input to the $\mathbb{Q}_+$-Multihistogram problem, i.e., X and $\mathcal{M} = \{M_1, \ldots, M_k\}$. As in Section 4, for $\mathbf{a} \in \mathbb{Z}^d$ we denote the solution set of a system $\mathcal{U}_{\mathcal{M},\mathbf{a}}$ of linear equations determined by the matrix $\left[M_1 \mid \ldots \mid M_k\right]$ and the column vector $\mathbf{a}$ by $\mathcal{C}_\mathbf{a}$; but this time we care about *nonnegative-rational* solutions. We thus put $\mathcal{C}_\mathbf{a} := \mathbb{Q}_+\text{-sol}(\mathcal{U}_{\mathcal{M},\mathbf{a}}) \subseteq (\mathbb{Q}_+)^r$. Recall the language (3). Our aim is to check existence of a rational $(X, \mathcal{M})$-multihistogram, i.e., of a family $\mathcal{H} = \{H_1, \ldots, H_k\}$ of nonnegative rational matrices, such that the following conditions are satisfied:

(A) the word of $\mathcal{H}$ belongs to the language (3) (interpreted in nonnegative rationals);
(B) the matrices $H_1, \ldots, H_k$ satisfy the histogram condition.

We construct in polynomial time a $\Rightarrow$-system $\mathcal{S}$ that is solvable if and only if conditions (A) and (B) are met. The solvability of $\mathcal{S}$ itself is decidable in PTime according to Lemma 15. The idea is to characterise conditions (A)–(B) by a sequence of runs in a homogeneous linear PN interleaved by single steps described by non-homogeneous systems of linear equations (where n is the column dimension of X):

$$\mathbf{0} \xrightarrow{\mathcal{C}_\mathbf{0}^*} \mathbf{c}_1 \xrightarrow{\mathcal{C}_{X(\_,1)}} \mathbf{c}_2 \xrightarrow{\mathcal{C}_\mathbf{0}^*} \mathbf{c}_3 \xrightarrow{\mathcal{C}_{X(\_,2)}} \ldots \xrightarrow{\mathcal{C}_\mathbf{0}^*} \mathbf{c}_{2n-2} \xrightarrow{\mathcal{C}_\mathbf{0}^*} \mathbf{c}_{2n-1} \xrightarrow{\mathcal{C}_{X(\_,n)}} \mathbf{c}_{2n} \xrightarrow{\mathcal{C}_\mathbf{0}^*} \mathbf{0}.$$

Conceptually, the construction follows the construction of a VAS in Section 5. We define a homogeneous linear PN $\mathcal{V}_\mathbf{0}$, recognizing the language $(\mathcal{C}_\mathbf{0})^*$ and, using Lemma 16, we compute in PTime a $\Rightarrow$-system $\mathcal{S}_\mathbf{0}$ such that the projection $P_\mathbf{0}$ of $\mathbb{Q}_+$-sol($\mathcal{S}_\mathbf{0}$) to some of its variables describes the reachability relation of $\mathcal{V}_\mathbf{0}$. Ignoring some technical details, the final $\Rightarrow$-system $\mathcal{S}$ imposes the following constraints (for all j):

1. there is a run from $\mathbf{c}_{2j}$ to $\mathbf{c}_{2j+1}$ in $\mathcal{V}_\mathbf{0}$, i.e., $(\mathbf{c}_{2j}, \mathbf{c}_{2j+1}) \in P_\mathbf{0}$;
2. $\mathbf{c}_{2j} - \mathbf{c}_{2j-1} \in \mathcal{C}_{X(\_,j)} = \mathbb{Q}_+\text{-sol}(\mathcal{U}_{\mathcal{M},X(\_,j)})$.

Now, $\mathcal{S}$ is solvable iff some rational $(X, \mathcal{M})$-multihistogram exists.

6.2 $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}\}$

In this, and only in this section we generalise slightly our setting and consider a fixed commutative ring $\mathbb{R}$, instead of just the ring of integers $\mathbb{Z}$ or rationals $\mathbb{Q}$. Accordingly, by a data vector we mean in this section a function $\mathbb{D} \to \mathbb{R}^d$ from data values to d-tuples of elements of $\mathbb{R}$ that maps almost all data values (i.e. all except for a finite number of data values) to the zero vector $\mathbf{0} \in \mathbb{R}^d$. With this more general notion of data vectors, we define $\mathbb{R}$-permutation sums and the $\mathbb{R}$-PERMUTATION SUM PROBLEM analogously as in Section 2.1. Furthermore, we define analogously $\mathbb{R}$-sums and consider linear equations with coefficients from $\mathbb{R}$ and their $\mathbb{R}$-solvability problem.

▶ **Theorem 17.** *For any commutative ring $\mathbb{R}$, the $\mathbb{R}$-PERMUTATION SUM PROBLEM reduces polynomially to the $\mathbb{R}$-solvability problem of linear equations.*

Clearly, Theorem 17 implies the remaining cases of Theorem 2, namely $\mathbb{X} \in \{\mathbb{Z}, \mathbb{Q}\}$, as in these cases the $\mathbb{X}$-solvability of linear equations is in PTIME. Theorem 17 follows immediately by Lemma 18 stated below. For a data vector $\mathbf{v}$, we define the vector $\textsc{sum}(\mathbf{v}) \in \mathbb{R}^d$ and a finite set of vectors $\textsc{vectors}(\mathbf{v}) \subseteq_{\text{fin}} \mathbb{R}^d$:

$$\textsc{sum}(\mathbf{v}) \stackrel{\text{def}}{=} \sum_{\alpha \in supp(\mathbf{v})} \mathbf{v}(\alpha) \qquad \textsc{vectors}(\mathbf{v}) \stackrel{\text{def}}{=} \{\mathbf{v}(\alpha) \mid \alpha \in supp(\mathbf{v})\}.$$

Clearly both operations commute with data permutations: $\textsc{sum}(\mathbf{v}) = \textsc{sum}(\mathbf{v} \circ \theta)$ and $\textsc{vectors}(\mathbf{v}) = \textsc{vectors}(\mathbf{v} \circ \theta)$, and can be lifted naturally to finite sets of data vectors:

$$\textsc{sum}(V) \stackrel{\text{def}}{=} \{\textsc{sum}(\mathbf{v}) \mid \mathbf{v} \in V\} \qquad \textsc{vectors}(V) \stackrel{\text{def}}{=} \bigcup_{\mathbf{v} \in V} \textsc{vectors}(\mathbf{v}).$$

▶ **Lemma 18.** *Let $\mathbf{x}$ be a data vector and V be a finite set of data vectors V. Then $\mathbf{x}$ is an $\mathbb{R}$-permutation sum of V if and only if*
1. $\textsc{sum}(\mathbf{x})$ *is an $\mathbb{R}$-sum of* $\textsc{sum}(V)$*, and*
2. *every $\mathbf{a} \in \textsc{vectors}(\mathbf{x})$ is an $\mathbb{R}$-sum of* $\textsc{vectors}(V)$.

Proof. The proof is inspired by Theorem 15 in [12]. The only if direction is immediate: if $\mathbf{x} = z_1 \cdot \mathbf{w}_1 + \ldots + z_n \cdot \mathbf{w}_n$ for $z_1, \ldots, z_n \in \mathbb{R}$ and $\mathbf{w}_1, \ldots, \mathbf{w}_n \in \textsc{Orbit}(V)$, then clearly $\textsc{sum}(\mathbf{x}) = z_1 \cdot \textsc{sum}(\mathbf{w}_1) + \ldots + z_n \cdot \textsc{sum}(\mathbf{w}_n)$ and hence $\textsc{sum}(\mathbf{x})$ is a $\mathbb{R}$-sum of $\textsc{sum}(V)$ (using the fact that $\textsc{sum}(\_)$ commutes with data permutations). Also $\mathbf{x}(\alpha)$ is necessarily an $\mathbb{R}$-sum of $\textsc{vectors}(V)$ for every $\alpha \in supp(\mathbf{x})$.

Now we focus on the if direction. For a vector $\mathbf{a} \in \mathbb{R}^d$, we define an $\mathbf{a}$-*move* as an arbitrary data vector that maps some data value to $\mathbf{a}$, some other data value to $-\mathbf{a}$, and all other data values to $\mathbf{0}$.

▶ **Claim 6.1.** *Every $\mathbf{a}$-move, for $\mathbf{a} \in \textsc{vectors}(\mathbf{v})$, is an $\mathbb{R}$-permutation sum of $\{\mathbf{v}\}$.*

Indeed, for $\mathbf{a} = \mathbf{v}(\alpha)$, consider a data permutation θ that preserves all elements of $supp(\mathbf{v})$ except that it maps α to a data value α' related in the same way as α by the order $\leq$ to all other data values in $supp(\mathbf{v})$. Then $\mathbf{a}$-moves are exactly data vectors $(\mathbf{v} - \mathbf{v} \circ \theta) \circ \rho = \mathbf{v} \circ \rho - \mathbf{v} \circ (\theta \circ \rho)$.

For the if direction, suppose point 1. holds: $\textsc{sum}(\mathbf{x})$ is an $\mathbb{R}$-sum of $\textsc{sum}(V)$. Treat the vector $\textsc{sum}(\mathbf{x})$ and the vectors in $\textsc{sum}(V)$ as data vectors with the same singleton support. Observe that $\textsc{sum}(\mathbf{v})$ for any $\mathbf{v} \in V$ is an $\mathbb{R}$-permutation sum of $\{\mathbf{v}\}$; indeed, by Claim 6.1 we can use $\mathbf{a}$-moves to transfer all nonzero vectors for data in $supp(\mathbf{v})$ into one datum. With this view in mind we have:

- $\textsc{sum}(\mathbf{x})$ is an $\mathbb{R}$-permutation sum of V.

Furthermore, suppose point 2. holds: every $\mathbf{a} \in \textsc{vectors}(\mathbf{x})$ is an $\mathbb{R}$-sum of $\textsc{vectors}(V)$. Thus every $\mathbf{a}$-move, for $\mathbf{a} \in \textsc{vectors}(\mathbf{x})$, is an $\mathbb{R}$-sum of $\{\mathbf{b}\text{-move} \mid \mathbf{b} \in \textsc{vectors}(V)\}$. By Claim 6.1 we know that every element of the latter set is an $\mathbb{R}$-permutation sum of V. Thus we entail:

- every $\mathbf{a}$-move, for $\mathbf{a} \in \textsc{vectors}(\mathbf{x})$, is an $\mathbb{R}$-permutation sum of V.

We have shown that $\textsc{sum}(\mathbf{x})$, as well as all $\mathbf{a}$-moves (for all $\mathbf{a} \in \textsc{vectors}(\mathbf{x})$), are $\mathbb{R}$-permutation sums of V. We use the $\mathbf{a}$-moves to transform $\textsc{sum}(\mathbf{x})$ into $\mathbf{x}$. This proves that $\mathbf{x}$ is an $\mathbb{R}$-permutation sum of V as required. ◀

7 Concluding remarks

The main result of this paper is determining the computational complexity of solving linear equations with integer (or rational) coefficients, in the setting of ordered data. We observed the huge gap: while the $\mathbb{N}$-solvability problem is equivalent (up to an exponential blowup) to the VAS reachability problem, the $\mathbb{Z}$-, $\mathbb{Q}$-, and $\mathbb{Q}_+$-solvability problems are all in PTime. This has a consequence for possible linear-algebraic overapproximations of the reachability in VAS with ordered data: instead of $\mathbb{N}$-solvability, one should apply $\mathbb{Z}$- or $\mathbb{Q}_+$-solvability, or even the combination of both.

Except for the last Section 6.2, the coefficients and solutions are assumed to belong to the ring $\mathbb{Q}$ of rationals, but clearly one can consider other commutative rings as well. There is another possible axis of generalisation, namely orbit-finite systems of linear equations over an orbit-finite set of variables, which can be introduced as follows. Fix an arbitrary commutative ring $\mathbb{R}$ and an arbitrary data domain $\mathbb{D}$. Consider *orbit-finite* sets (see, e.g., [3, 4]), i.e., sets that are finite up to the natural action of data automorphisms of $\mathbb{D}$. For instance, in case of the ordered data domain $\mathbb{D}$, the natural action of a monotonic bijection $\theta : \mathbb{D} \to \mathbb{D}$ maps a pair $(d, i) \in \mathbb{D} \times \{1, \ldots, d\}$ to $(\theta(d), i)$; and maps a data vector $\mathbf{v}$ to $\mathbf{v} \circ \theta^{-1}$. Therefore $\mathbb{D} \times \{1, \ldots, d\}$ is orbit-finite (the number of orbits is d) and $\textsc{Orbit}(V)$ is orbit-finite whenever V is finite (the number of orbits is at most the cardinality of V). For an orbit-finite set $\mathcal{Y}$, by an *$\mathcal{Y}$-vector* we mean (think of $\mathcal{Y} = \mathbb{D} \times \{1, \ldots, d\}$) any function $\mathcal{Y} \to \mathbb{R}$ that maps almost all elements of $\mathcal{Y}$ to $0 \in \mathbb{R}$; let $\mathbb{R}^{\mathcal{Y}}$ be the set of all $\mathcal{Y}$-vectors. A $\mathcal{Y}$-matrix is an orbit-finite family of (column) $\mathcal{Y}$-vectors, $\mathcal{M} \subseteq_{\text{orbit-finite}} \mathbb{R}^{\mathcal{Y}}$. Such a $\mathcal{Y}$-matrix $\mathcal{M}$, together with a (column) $\mathcal{Y}$-vector $\mathbf{a}$, determines a system of linear equations $\mathcal{U}_{\mathcal{M},\mathbf{a}}$, whose solutions are those $\mathcal{M}$-vectors that, treated as coefficients of a linear combination of vectors $m \in \mathcal{M}$, yield $\mathbf{a} \in \mathcal{Y}$:

$$\text{sol}(\mathcal{U}_{M,\mathbf{a}}) \quad = \quad \{\mathbf{v} \in \mathbb{R}^{\mathcal{M}} \mid \sum_{m \in \mathcal{M}} \mathbf{v}(m) \cdot m = \mathbf{a}\}.$$

Note that the sum is well defined as $\mathbf{v}(m) \neq 0$ for only finitely many elements $m \in \mathcal{M}$. The setting of this paper is nothing but a special case, where $\mathbb{R} = \mathbb{Q}$ and $\mathcal{Y} = \mathbb{D} \times \{1, \ldots, d\}$ and $\mathcal{M} = \textsc{Orbit}(V)$ for a finite set V of data vectors. Similarly, another special case has been investigated in [12], where finiteness up to the natural action of automorphisms of the data domain $(\mathbb{D}, =)$ played a similar role. As another example, in [14] the solvability problem has been investigated (in the framework of CSP) for the same data domain $(\mathbb{D}, =)$, in the case where $\mathbb{R}$ is a finite field.

It is an exciting research challenge to fully understand the complexity landscape of orbit-finite systems of linear equations, as a function of the choice of data domain. The results of this paper are a step towards this goal, and indicate that development of the uniform theory will be hard: the case of ordered data, compared to the case of unordered data investigated in [12], requires significantly new techniques and the complexity of the nonnegative integer solvability differs significantly too. Even more broadly, investigation of orbit-finite dimensional linear algebra, together with its possible applications in the analysis of data-enriched systems, seems to be a tempting continuation of this work.

References

1 Michael Blondin, Alain Finkel, Christoph Haase, and Serge Haddad. Approaching the coverability problem continuously. In *Proc. TACAS 2016*, pages 480–496, 2016. doi:10.1007/978-3-662-49674-9_28.

2 Michael Blondin and Christoph Haase. Logics for continuous reachability in Petri nets and vector addition systems with states. In *Proc. LICS 2017*, pages 1–12, 2017. doi:10.1109/LICS.2017.8005068.

3 Mikołaj Bojańczyk. Slightly infinite sets. A draft of a book available at https://www.mimuw.edu.pl/~bojan/paper/atom-book. URL: https://www.mimuw.edu.pl/~bojan/paper/atom-book.

4 Mikołaj Bojańczyk, Bartek Klin, Slawomir Lasota, and Szymon Toruńczyk. Turing machines with atoms. In *Proc. LICS 2013*, pages 183–192, 2013. doi:10.1109/LICS.2013.24.

5 Rémi Bonnet, Alain Finkel, Serge Haddad, and Fernando Rosa-Velardo. Comparing Petri data nets and timed Petri nets. Technical Report LSV-10-23, Laboratoire Spécification et Vérification, ENS Cachan, France, 2010.

6 Dmitry Chistikov and Christoph Haase. The taming of the semi-linear set. In *Proc. ICALP'16*, pages 128:1–128:13, 2016.

7 Eric Domenjoud. Solving systems of linear Diophantine equations: An algebraic approach. In *Proc. MFCS 1991*, pages 141–150, 1991.

8 Estíbaliz Fraca and Serge Haddad. Complexity analysis of continuous Petri nets. *Fundam. Inform.*, 137(1):1–28, 2015. doi:10.3233/FI-2015-1168.

9 Thomas Geffroy, Jérôme Leroux, and Grégoire Sutre. Occam's razor applied to the Petri net coverability problem. In *Proc. Reachability Problems 2016*, pages 77–89, 2016. doi:10.1007/978-3-319-45994-3_6.

10 Piotr Hofman and Sławomir Lasota. Linear equations with ordered data. *CoRR*, arXiv:1802.06660, 2018. URL: http://arxiv.org/abs/1802.06660.

11 Piotr Hofman, Slawomir Lasota, Ranko Lazić, Jérôme Leroux, Sylvain Schmitz, and Patrick Totzke. Coverability Trees for Petri Nets with Unordered Data. In *Proc. FoSSaCS*, Eindhoven, Netherlands, 2016. URL: https://hal.inria.fr/hal-01252674.

12 Piotr Hofman, Jérôme Leroux, and Patrick Totzke. Linear combinations of unordered data vectors. In *Proc. LICS 2017*, pages 1–11, 2017.

13 Richard M. Karp. Reducibility among combinatorial problems. In *Proc. Complexity of Computer Computations*, pages 85–103, 1972. URL: http://www.cs.berkeley.edu/~luca/cs172/karp.pdf.

14 Bartek Klin, Eryk Kopczyński, Joanna Ochremiak, and Szymon Toruńczyk. Locally finite constraint satisfaction problems. In *Proc. LICS 2015*, pages 475–486, 2015. doi:10.1109/LICS.2015.51.

15 S. Rao Kosaraju. Decidability of reachability in vector addition systems. In *STOC'82*, pages 267–281. ACM, 1982. doi:10.1145/800070.802201.

16 Slawomir Lasota. Decidability border for Petri nets with data: WQO dichotomy conjecture. In *Proc. PETRI NETS 2016*, pages 20–36, 2016. doi:10.1007/978-3-319-39086-4_3.

17 Ranko Lazic, Thomas Christopher Newcomb, Joël Ouaknine, A. W. Roscoe, and James Worrell. Nets with tokens which carry data. *Fundam. Inform.*, 88(3):251–274, 2008.

18 Jérôme Leroux and Sylvain Schmitz. Demystifying reachability in vector addition systems. In *Proc. LICS 2015*, pages 56–67, 2015. doi:10.1109/LICS.2015.16.

19 Richard Lipton. The reachability problem requires exponential space. Technical Report 62, Yale University, 1976.

20 Ernst W. Mayr. An algorithm for the general Petri net reachability problem. In *Proc. STOC 1981*, pages 238–246. ACM, 1981. doi:10.1145/800076.802477.

21 Loic Pottier. Minimal solutions of linear Diophantine systems: Bounds and algorithms. In *Proc. RTA 1991*, pages 162–173, 1991. URL: http://dl.acm.org/citation.cfm?id=647192.720494.

22 Laura Recalde, Serge Haddad, and Manuel Silva Suárez. Continuous Petri nets: Expressive power and decidability issues. *Int. J. Found. Comput. Sci.*, 21(2):235–256, 2010. doi:10.1142/S0129054110007222.

23 Fernando Rosa-Velardo and David de Frutos-Escrig. Decidability and complexity of Petri nets with unordered data. *Theor. Comput. Sci.*, 412(34):4439–4451, 2011. doi:10.1016/j.tcs.2011.05.007.

24 Manuel Silva Suárez, Enrique Teruel, and José Manuel Colom. Linear algebraic and linear programming techniques for the analysis of place or transition net systems. In *Lectures on Petri Nets I: Basic Models, Advances in Petri Nets*, pages 309–373, 1996. doi:10.1007/3-540-65306-6_19.