On Strongest Algebraic Program Invariants

EHUD HRUSHOVSKI, Oxford University, UK
JOËL OUAKNINE, Max Planck Institute for Software Systems, Germany
AMAURY POULY, Université de Paris, IRIF, CNRS, France
JAMES WORRELL, Oxford University, UK

A polynomial program is one in which all assignments are given by polynomial expressions and in which all branching is nondeterministic (as opposed to conditional). Given such a program, an algebraic invariant is one that is defined by polynomial equations over the program variables at each program location. Müller-Olm and Seidl have posed the question of whether one can compute the strongest algebraic invariant of a given polynomial program. In this article, we show that, while strongest algebraic invariants are not computable in general, they can be computed in the special case of affine programs, that is, programs with exclusively linear assignments. For the latter result, our main tool is an algebraic result of independent interest: Given a finite set of rational square matrices of the same dimension, we show how to compute the Zariski closure of the semigroup that they generate.

CCS Concepts: • **Theory of computation** → *Verification by model checking*; **Abstraction**; *Quantitative automata*; **Invariants**; • **Mathematics of computing** → *Topology*; • **Computing methodologies** → **Algebraic algorithms;**

Additional Key Words and Phrases: Program verification, polynomial programs, algebraic invariants, matrix semigroups, Zariski closure

ACM Reference format:
Ehud Hrushovski, Joël Ouaknine, Amaury Pouly, and James Worrell. 2023. On Strongest Algebraic Program Invariants. *J. ACM* 70, 5, Article 29 (October 2023), 22 pages.
https://doi.org/10.1145/3614319

1 INTRODUCTION

Invariants are one of the most fundamental and useful notions in the quantitative sciences, appearing in a wide range of contexts, from gauge theory, dynamical systems, and control theory in physics, mathematics, and engineering, to program verification, static analysis, abstract interpretation, and programming language semantics (among others) in computer science. In spite of decades of scientific work and progress, automated invariant synthesis remains a topic of active

Joël Ouaknine is also affiliated with Keble College, Oxford as `emmy.network` Fellow; he was supported by ERC grant AVS-ISS (648701) and by DFG grant 389792660 as part of TRR 248 (see https://perspicuous-computing.science). James Worrell was supported by EPSRC Fellowship EP/N008197/1 and by UKRI Fellowship EP/X033813/1.
Authors' addresses: E. Hrushovski, Mathematical Institute, Oxford University, UK; email: ehud.hrushovski@maths.ox.ac.uk; J. Ouaknine, Max Planck Institute for Software Systems, Saarland Informatics Campus, Germany; email: joel@mpi-sws.org; A. Pouly, Université de Paris, IRIF, CNRS, F-75013 Paris, France; email: amaury.pouly@irif.fr; J. Worrell, Department of Computer Science, Oxford University, UK; email: jbw@cs.ox.ac.uk.

0004-5411/2023/10-ART29 $15.00
https://doi.org/10.1145/3614319

research, particularly in the fields of theorem proving and program analysis, and plays a central role in methods and tools seeking to establish correctness properties of computer programs; see, e.g., Reference [25], and particularly Section 8 therein.

Affine programs are a simple kind of nondeterministic imperative programs (which may contain arbitrarily nested loops) in which the only instructions are assignments whose right-hand sides are affine expressions, such as $x_3 := x_1 - 3x_2 + 7$. A conventional imperative program can be abstracted to an affine program by replacing conditionals with nondeterminism and conservatively over-approximating non-affine assignments; see, e.g., Reference [5]. In doing so, affine programs enable one to reason about more complex programs; a particularly striking example is the application of affine programs to several problems in inter-procedural program analysis [5, 18, 36, 37].

An affine invariant for an affine program with n variables assigns to each program location an affine subspace of $\mathbb{Q}^n$ such that the resulting family of subspaces is preserved under the transition relation of the program. Such an invariant is specified by giving a finite set of affine equations at each location. The strongest (i.e., smallest with respect to set inclusion) affine invariant is obtained by taking the affine hull of the set of reachable configurations (i.e., values of the program variables) at each program location. Equivalently, the strongest affine invariant is determined by giving, for each program location, the set of all affine equations holding at that location.

An algorithm due to Michael Karr in 1976 [24] computes the strongest affine invariant of an affine program. A more efficient reformulation of Karr's algorithm was given by Müller-Olm and Seidl [37], who moreover showed that if the class of affine programs is augmented with equality guards, then it becomes undecidable whether or not a given affine equation holds at a particular program location. A randomised algorithm for discovering affine equations was proposed by Gulwani and Necula [18].

A natural and more expressive generalisation of affine invariants are algebraic invariants. An algebraic invariant assigns to each program location an algebraic set (i.e., one defined by a conjunction of polynomial equations) such that the resulting family is preserved under the transition relation of the program. An algebraic invariant is specified by giving a set of polynomial equations that hold at each program location. The strongest algebraic invariant (i.e., smallest algebraic set with respect to set inclusion) is obtained by taking the Zariski closure of the set of reachable configurations in each location.

The problem of computing algebraic invariants for affine programs and related formalisms has been extensively studied over the past 15 years; see, e.g., References [6, 8, 12, 12, 20, 23, 25, 27, 29, 42–44]. However, in contrast to the case of affine invariants, as of yet no method is known to compute the *strongest* algebraic invariant, i.e., (a basis for) the set of all polynomial equations holding at each location of a given affine program. Existing methods are either heuristic in nature or only known to be complete relative to restricted classes of invariants or programs. For example, it is shown in Reference [37] (see also Reference [42]) that Karr's algorithm can be applied to compute the smallest algebraic invariant that is specified by polynomial equations of a fixed degree d. (The case of affine invariants corresponds to $d = 1$.) Reference [12] gives a method that finds all algebraic invariants for a highly restricted class of affine programs (in which all linear mappings have positive rational eigenvalues). The approach of References [20, 27] via so-called P-solvable loops does not encompass the whole class of affine programs either (although it does allow to handle certain classes of programs with polynomial assignments) [28].

In this article, we give a method to compute the set of *all* polynomial equations that hold at a given location of an affine program, or in other words, the strongest algebraic invariant. The output of the algorithm gives for each program location a finite basis of the ideal of all polynomial equations holding at that location.

Our main tool is an algebraic result of independent interest: We give an algorithm that, given a finite set of rational square matrices of the same dimension, computes the Zariski closure of the semigroup that they generate. Our algorithm generalises (and uses as a subroutine) an algorithm of Derksen, Jeandel, and Koiran [13] to compute the Zariski closure of a finitely generated group of invertible matrices.[1]

Our procedure for computing the Zariski closure of a matrix semigroup also generalises a result of Mandel and Simon [30] and, independently, of Jacob [21, 22], to the effect that it is decidable whether a finitely generated semigroup of rational matrices is finite. Note that for a field $\mathbb{K}$, an algebraic set that is given as the zero set of a polynomial ideal $I \subseteq \mathbb{K}[x_1, \ldots, x_n]$ is finite just in case the quotient $\mathbb{K}[x_1, \ldots, x_n]/I$ is finite-dimensional as a vector space over $\mathbb{K}$ [11, Chapter 5, Section 3]). The latter condition can be checked by computing a Gröbner basis for I.

As mentioned above, we make use of the result of Reference [13] that one can compute the Zariski closure of the group generated by a finite set of invertible rational matrices. That result itself relies on several non-trivial mathematical ingredients, including results of Masser [32] on computing multiplicative relations among given algebraic numbers and Schur's theorem that every finitely generated periodic subgroup of the general linear group $\mathrm{GL}_n(\mathbb{C})$ is finite.

Given a set A of rational square matrices of the same dimension, we leverage these group-theoretic results to compute the Zariski closure $\overline{\langle A \rangle}$ of the generated semigroup $\langle A \rangle$. To this end, we use multilinear algebra as well as structural properties of matrix semigroups to identify finitely many subsemigroups of $\overline{\langle A \rangle}$ that can be used to generate the entire semigroup. Pursuing this approach requires that we first generalise the result of Reference [13] to show that one can compute the Zariski closure of the group generated by a constructible (as opposed to finite) set of invertible matrices.

It is worth pointing out that whether a particular configuration is reachable at a certain program location of a given affine program is in general an undecidable problem—this follows quite straightforwardly from the undecidability of the membership problem for finitely generated matrix semigroups, discussed shortly. It is therefore somewhat remarkable that the Zariski closure (i.e., the smallest algebraic superset) of the set of reachable configurations at any particular location nevertheless turns out to be a computable object.

Finally, we consider a generalisation of the class of affine programs to the class of so-called *polynomial programs*, which allows polynomial assignments but still has only nondeterministic (as opposed to conditional) branching. The problem of computing all algebraic invariants of a given polynomial program was posed in Reference [36, Section 5] by Müller-Olm and Seidl. We show that this problem is undecidable in Section 7 by a reduction from the boundedness problem for reset Petri nets.

Related Work

Decision problems for matrix semigroups have also been studied for many decades, independently of program analysis. One of the most prominent such is the Membership Problem, i.e., whether a given matrix belongs to a finitely generated semigroup of integer matrices. An early and striking result on this topic is due to Markov, who showed undecidability of the Membership Problem in dimension 6 in 1947 [31]. Later, Paterson [40] improved this result to show undecidability in dimension 3, while decidability in dimension 2 remains open. A breakthrough was achieved in 2017 by Potapov and Semukhin, who showed decidability of membership for semigroups

[1] Related to this, Corollary 3.7 and Lemma 3.6a in Reference [19] reduce the question of computing the Zariski closure of a finitely generated group of invertible matrices to that of finding multiplicative relations among diagonal matrices. Note that if one begins with rational matrices, then such relations can be found simply using prime decomposition of the entries.

generated by *nonsingular* integer 2×2 matrices [41]. By contrast, the Membership Problem was shown to be polynomial-time decidable in any dimension by Babai et al. for *commuting* matrices over algebraic numbers [2]. As aptly noted by Stillwell, "noncommutative semigroups are hard to understand" [47]. Matrix semigroup theory also plays a central role in the analysis of weighted automata (such as probabilistic and quantum automata; see, e.g., References [4, 13]).

Algebraic invariants are stronger (i.e., more precise) than affine invariants. Various other types of domains have been considered in the setting of abstract interpretation, e.g., intervals, octagonal sets, and convex polyhedra (see, e.g., References [9, 10, 33] and references in Reference [5]). The precision of such domains in general is incomparable to that of algebraic invariants. Unlike with algebraic and affine invariants, there need not be a strongest convex polyhedral invariant for a given affine program. A natural decision problem in this setting is to ask for an inductive invariant that is disjoint from a given set of states (which one would like to show is not reachable). The version of this decision problem for convex invariants on affine programs was proposed by Monniaux [34] and remains open; if the convexity requirement is dropped, then the problem is shown to be undecidable in Reference [15].

The computation of semialgebraic invariants has also been considered in the context of discrete-time linear dynamical systems and linear loops (which can be viewed as highly restricted instances of affine programs); see, e.g., References [1, 16, 17].

2 TWO ILLUSTRATIVE EXAMPLES

We now present two simple examples to illustrate some of the ideas and concepts that are discussed in this article. Some of the notation and terminology that we use is only introduced in later sections; should this impede understanding, we recommend that the reader return to these examples after having read Sections 3 and 4.

As a first motivating example, consider the following linear loop:

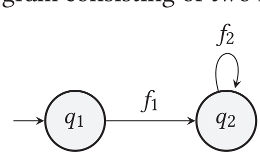
$x := 3;$

$y := 2;$

while $2y - x \geq -2$ **do**

$$\begin{pmatrix} x \\ y \end{pmatrix} := \begin{pmatrix} 10 & -8 \\ 6 & -4 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix};$$

This loop never halts, although this fact is perhaps not immediately obvious. Here, we show how the techniques developed in this article can help establish non-termination. To this end, we first turn our code into an affine program consisting of two locations, as follows:



Here, f_1 is the constant affine function assigning 3 to x and 2 to y, whereas f_2 is the linear transformation associated with the matrix appearing in our while loop. Note that we have discarded the loop guard.

The collecting semantics of this affine program assigns to location q_2 the set $S_{q_2} \subseteq \mathbb{Z}^2$ of all values taken by the pair of variables (x, y) in the unending execution of the program. As it turns out, the Zariski closure of S_{q_2}, regarded as a subset of real affine space $\mathbb{R}^2$, is the set

$$\{(x, y) \in \mathbb{R}^2 : x - 9x^2 - y + 24xy - 16y^2 = 0\}.$$

By construction, this algebraic invariant is stable under f_2 and over-approximates the set S_{q_2} of reachable (x, y)-configurations. Verifying that all tuples in this algebraic set moreover satisfy the guard $2y - x \geq -2$ is now a simple exercise in high-school algebra, from which one concludes that our original loop will indeed never terminate.

For our second example, consider the matrix semigroup $\langle S, T, E\rangle$ generated by the following matrices:

$$S := \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad T := \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad E := \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

We identify the set $M_2(\mathbb{R})$ of real 2×2 matrices with the real affine space $\mathbb{R}^4$ and define $G := \overline{\langle S, T, E\rangle}$ to be the Zariski closure of the above semigroup. We show that $G = \{M \in \mathbb{R}^{2\times 2} : \det(M) = 1 \text{ or } \det(M) = 0\}$ and in the process illustrate (in a very simple setting) the approach of computing the Zariski closure of a matrix semigroup by order of decreasing rank. This approach underlies the algorithm described in Section 6.

Consider first $G' := \{M \in G : \mathrm{rk}(M) = 2\}$. From the fact that the set of singular matrices in $M_2(\mathbb{R})$ is Zariski closed, one can show that $G' = \{M \in \overline{\langle S, T\rangle} : \mathrm{rk}(M) = 2\}$. Now, it is well known that S and T generate the semigroup $\mathrm{SL}_2(\mathbb{Z})$ of 2×2 integer matrices of determinant 1 and that the real Zariski closure of $\mathrm{SL}_2(\mathbb{Z})$ is the semigroup $\mathrm{SL}_2(\mathbb{R})$ of 2×2 *real* matrices of determinant 1[2]; hence, $G' = \mathrm{SL}_2(\mathbb{R})$. More generally, we can use the algorithm of Derksen, Jeandel, and Koiran [13] to compute the Zariski closure of any finitely generated semigroup of invertible matrices.

Now, we consider the sub-semigroup G'' of singular matrices in G. This is the real Zariski closure of the semigroup generated by the (constructible) set of matrices

$$\{MEM', ME, EM : M, M' \in \mathrm{SL}_2(\mathbb{R})\}.$$

It is straightforward to observe that this generating set already includes all rank-1 matrices in $M_2(\mathbb{R})$ and hence that the generated semigroup contains all singular matrices. We conclude that $G = G' \cup G''$ comprises all matrices in $M_2(\mathbb{R})$ of determinant 0 or 1.

3 MATHEMATICAL BACKGROUND

3.1 Linear Algebra

Matrices. Let $\mathbb{K}$ be a field. We denote by $M_n(\mathbb{K})$ the semigroup of square matrices of dimension n with entries in $\mathbb{K}$. We write $\mathrm{GL}_n(\mathbb{K})$ for the subgroup of $M_n(\mathbb{K})$ comprising all invertible matrices. Given a set of matrices $A \subseteq M_n(\mathbb{K})$, we denote by $\langle A\rangle$ the sub-semigroup of $M_n(\mathbb{K})$ generated by A. The rank of a matrix a is denoted by $\mathrm{rk}(a)$, its kernel by $\ker(a)$, and its image by $\mathrm{im}(a)$. We denote by $U \oplus V$ the direct sum of U and V.

Exterior Algebra and the Grassmannian. Given a vector space V over the field $\mathbb{K}$, its exterior algebra ΛV is a vector space that embeds V and is equipped with an associative, bilinear, and anti-symmetric map

$$\wedge : \Lambda V \times \Lambda V \to \Lambda V.$$

We can construct ΛV as a direct sum

$$\Lambda V = \Lambda^0 V \oplus \Lambda^1 V \oplus \Lambda^2 V \cdots,$$

where $\Lambda^r V$ denotes the rth-exterior power of V for $r \in \mathbb{N}$, that is, the subspace of ΛV generated by r-fold wedge products $v_1 \wedge \ldots \wedge v_r$ for $v_1, \ldots, v_r \in V$. If V is finite dimensional, with basis $e_1, \ldots, e_n$, then a basis of $\Lambda^r V$ is given by $e_{i_1} \wedge \cdots \wedge e_{i_r}$, $1 \leq i_1 < \ldots < i_r \leq n$. Thus, $\Lambda^r V$ has dimension $\binom{n}{r}$ (where $\binom{n}{r} = 0$ for $r > n$).

[2]The latter fact follows from the Borel density theorem [35, Sections 4.5 and 7.0], but can also be established directly by an elementary argument.

A basic property of the wedge product is that given vectors $u_1, \ldots, u_r \in V$, $u_1 \wedge \ldots \wedge u_r \neq 0$ if and only if $\{u_1, \ldots, u_r\}$ is a linearly independent set. Furthermore, given $w_1, \ldots, w_r \in V$, we have that $u_1 \wedge \ldots \wedge u_r$ and $w_1 \wedge \ldots \wedge w_r$ are scalar multiples of each other iff $\mathrm{span}(u_1, \ldots, u_r) = \mathrm{span}(v_1, \ldots, v_r)$.

The *Grassmannian* $\mathrm{Gr}(r, n)$ is the set of r-dimensional subspaces of $\mathbb{K}^n$. By the above-stated properties of the wedge product there is an injective function

$$\iota : \mathrm{Gr}(r, n) \to \Lambda^r(\mathbb{K}^n)$$

such that for any W, $\iota(W) = v_1 \wedge \cdots \wedge v_r$ where $v_1, \ldots, v_r$ is an arbitrarily chosen basis of W. Note that given two bases $v_1, \ldots, v_r$ and $u_1, \ldots, u_r$ of W, there exists $\alpha \in \mathbb{K}$ such that $v_1 \wedge \cdots \wedge v_r = \alpha(u_1 \wedge \cdots \wedge u_r)$. In other words, the particular choice of a basis for W only changes the value of $\iota(W)$ up to a constant. Given subspaces $W_1, W_2 \subseteq V$, we moreover have $W_1 \cap W_2 = 0$ iff $\iota(W_1) \wedge \iota(W_2) \neq 0$. We refer to Reference [39, Chapter 1.3] for more details about the Grassmannian.

3.2 Algebraic Geometry

In this section, we summarise some basic notions of algebraic geometry that will be used in the rest of the article.

Let $\mathbb{K}$ be a field. An *affine variety* or *algebraic set* $X \subseteq \mathbb{K}^n$ is the set of common zeros of a finite collection of polynomials, i.e., a set of the form

$$X = \{x \in \mathbb{K}^n : p_1(x) = p_2(x) = \cdots = p_\ell(x) = 0\} ,$$

where $p_1, \ldots, p_\ell \in \mathbb{K}[x_1, \ldots, x_n]$.[3] Given a polynomial ideal $I \subseteq \mathbb{K}[x_1, \ldots, x_n]$, by Hilbert's basis theorem the set

$$\mathbf{V}(I) = \{x \in \mathbb{K}^n : \forall p \in I,\ p(x) = 0\}$$

is a variety, called the *variety of* I. The two main varieties of interest to us are $X = M_n(\mathbb{K})$, which we identify with affine space $\mathbb{K}^{n^2}$ in the natural way, and $X = \mathrm{GL}_n(\mathbb{K})$, which we identify with the variety

$$\{(A, y) \in \mathbb{K}^{n^2+1} : \det(A) \cdot y = 1\} .$$

Given an affine variety $X \subseteq \mathbb{K}^n$, the *Zariski topology* on X has as closed sets the subvarieties of X, i.e., those sets $A \subseteq X$ that are themselves affine varieties in $\mathbb{K}^n$. For example, $\{a \in M_n(\mathbb{K}) : \mathrm{rk}(a) < r\}$ is a Zariski closed subset of $M_n(\mathbb{K})$, since for $a \in M_n(\mathbb{K})$ we have $\mathrm{rk}(a) < r$ iff all $r \times r$ minors of a vanish. Given an arbitrary set $S \subseteq X$, we write $\overline{S}$ for its closure in the Zariski topology on X.

Given $S \subseteq \mathbb{K}^n$, let $I \subseteq \mathbb{K}[x_1, \ldots, x_n]$ be the ideal of polynomials that vanish on S. Observe that if the elements of S lie in a subfield $\mathbb{F}$ of $\mathbb{K}$, then the ideal I has a basis of polynomials with coefficients in $\mathbb{F}$. Indeed, if we fix a monomial ordering, then, by linear algebra, for every polynomial $f \in \mathbb{K}[x_1, \ldots, x_n]$ that vanishes on S there is a polynomial $g \in \mathbb{F}[x_1, \ldots, x_n]$ that also vanishes on S such that f and g have the same leading monomial. It follows that I has a Gröbner basis of polynomials in $\mathbb{F}[x_1, \ldots, x_n]$ (cf. [11, Chapter 5.2, Corollary 6]).

A set $S \subseteq X$ is *irreducible* if for all closed subsets $A_1, A_2 \subseteq X$ such that $S \subseteq A_1 \cup A_2$, we have either $S \subseteq A_1$ or $S \subseteq A_2$. It is well known that the Zariski topology on a variety is Noetherian. In particular, any closed subset A of X can be written as a finite union of *irreducible components*, where an irreducible component of A is a maximal irreducible closed subset of A.

[3] We use the terms variety and algebraic set interchangeably. Many authors reserve the term variety for an *irreducible* algebraic set.

The *dimension* of a variety X is defined to be the maximum number $d \in \mathbb{N}$ such that there is a strictly increasing chain $S_0 \subset S_1 \subset \cdots \subset S_d$ of non-empty irreducible closed subsets of X. A variety $X \subseteq \mathbb{K}^n$ has dimension at most n.

The class of *constructible* subsets of a variety X is obtained by taking all Boolean combinations (including complementation) of Zariski closed subsets. Suppose that the underlying field $\mathbb{K}$ is algebraically closed. Since the first-order theory of algebraically closed fields admits quantifier elimination, the constructible subsets of X are exactly the subsets of X that are first-order definable over $\mathbb{K}$ in the language of rings, i.e., that are definable by first-order formulas with parameters from $\mathbb{K}$.

Suppose that $X \subseteq \mathbb{K}^m$ and $Y \subseteq \mathbb{K}^n$ are affine varieties. A function $\varphi : X \to Y$ is called a *regular map* if it arises as the restriction of a polynomial map $\mathbb{K}^m \to \mathbb{K}^n$. Chevalley's Theorem states that if $\mathbb{K}$ is algebraically closed and $\varphi : X \to Y$ is a regular map, then the image $\varphi(A)$ of a constructible set $A \subseteq X$ under φ is a constructible subset of Y. This result also follows from the fact that the theory of algebraically closed fields admits quantifier elimination.

A regular map of interest to us is matrix multiplication $M_n(\mathbb{K}) \times M_n(\mathbb{K}) \to M_n(\mathbb{K})$. In particular, we have that for constructible sets of matrices $A, B \subseteq M_n(\mathbb{K})$ the set of products

$$A \cdot B := \{ab : a \in A, b \in B\}$$

is again constructible. Notice also that matrix inversion is a regular map $\mathrm{GL}_n(\mathbb{K}) \to \mathrm{GL}_n(\mathbb{K})$. Thus, if $A \subseteq \mathrm{GL}_n(\mathbb{K})$ is a constructible set, then so is $A^{-1} := \{a^{-1} : a \in A\}$. Finally, the projection $(A, y) \mapsto A$ yields an injective regular map $\mathrm{GL}_n(\mathbb{K}) \to M_n(\mathbb{K})$. Via this map, we can identify $\mathrm{GL}_n(K)$ with a constructible subset of $M_n(\mathbb{K})$.

On several occasions, we will use the facts that regular maps are continuous with respect to the Zariski topology and that the image of an irreducible set under a regular map is again irreducible. In particular, we have:

Lemma 1. *If $X, Y \subseteq \mathrm{GL}_n(\mathbb{K})$ are irreducible closed sets, then $\overline{X \cdot Y}$ is also irreducible.*

3.3 Algorithmic Manipulation of Constructible Sets

In this subsection, we briefly recall some algorithmic constructions on constructible subsets of a variety. We work over the field $\overline{\mathbb{Q}}$ of algebraic numbers. Not only is this field algebraically closed, but there are also symbolic representations of algebraic numbers with respect to which arithmetic is effective (see Reference [7, Section 4.2]), which allows us to use standard algebraic-geometry algorithms, such as procedures for computing Gröbner bases, and so on.

Representing Constructible Sets. Consider a variety $X \subseteq \overline{\mathbb{Q}}^n$ and let $I \subseteq \overline{\mathbb{Q}}[x_1, \ldots, x_n]$ be the ideal of polynomials that vanish on X. We represent Zariski closed subsets of X as zero sets of ideals in the coordinate ring $\overline{\mathbb{Q}}[X] = \overline{\mathbb{Q}}[x_1, \ldots, x_n]/I$ of X. The coordinate ring of $M_n(\overline{\mathbb{Q}})$ is just $\overline{\mathbb{Q}}[x_{1,1}, \ldots, x_{n,n}]$ while the coordinate ring of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ is

$$\overline{\mathbb{Q}}[x_{1,1}, \ldots, x_{n,n}, y]/(\det(x_{i,j})y - 1) .$$

Unions and intersections of Zariski closed subsets of X, respectively, correspond to products and sums of the corresponding ideals in $\overline{\mathbb{Q}}[X]$. We furthermore represent constructible subsets of X as Boolean expressions over Zariski closed subsets.

Irreducible Components. Let $A \subseteq X$ denote a Zariski closed set that is given as the variety of an ideal $I \subseteq \overline{\mathbb{Q}}[X]$. If $I = P_1 \cap \cdots \cap P_m$ is an irredundant decomposition of I into primary ideals, then $A = \mathbf{V}(P_1) \cup \ldots \cup \mathbf{V}(P_m)$ is a decomposition of A into irreducible components. One can compute the primary decomposition of an ideal using Gröbner basis techniques [3, Chapter 8].

Zariski Closure. At several points in our development, we will need to compute the Zariski closure of a constructible subset of a variety. Now, an arbitrary constructible subset of a variety X can be written as a union of differences of closed subsets of X. Thus, it suffices to be able to compute the closure of $A \setminus B$ for closed sets $A, B \subseteq X$. Furthermore, by first computing a decomposition of A as a union of irreducible closed sets, we may also assume that A is irreducible. But $A \subseteq \overline{A \setminus B} \cup (A \cap B)$; thus, by irreducibility of A, we have $\overline{A \setminus B} = \emptyset$ if $A \subseteq B$ and otherwise $\overline{A \setminus B} = A$. An algorithm (when using the representation above) for computing the Zariski closure of a constructible set, essentially following this recipe, is given in Reference [26, Theorem 1].

Images under Regular Maps. One can use an algorithm for quantifier elimination for the theory of algebraically closed fields to compute the image of a constructible set under a regular map. An explicit algorithm for this task, using Gröbner bases, is given in Reference [45, Section 4].

Finding an Element in a Constructible Set. The problem of finding an element in a given non-empty constructible set $A \subseteq \overline{\mathbb{Q}}^n$ is clearly computable in principle: Enumerate the elements of $\overline{\mathbb{Q}}^n$ and check each one for membership in A. A more efficient procedure is to proceed by induction on the dimension n. In dimension one, a constructible set A is a Boolean combination of finite algebraic sets, thus, one can find a point of A among the elements of these sets plus one additional fresh element. In dimension $n \geq 2$, one can project A on the first $n-1$ dimensions, find an algebraic point in the projection by induction, then substitute this point into the description A and reduce to the one-dimensional case.

4 ALGEBRAIC INVARIANTS FOR POLYNOMIAL PROGRAMS

In this section, we introduce the notions of polynomial programs and algebraic invariants. In discussing the latter, we work over the field $\overline{\mathbb{Q}}$ of algebraic numbers. However, as we note below, since polynomial programs are defined with rational data, the Zariski closure of the set of reachable configurations is the zero set of a collection of polynomials with rational coefficients, regardless of the field in which one takes the Zariski closure.[4] In this section, boldface symbols denote vectors.

A *polynomial program* of dimension n is a tuple $\mathcal{A} = (Q, E, q_{\text{init}})$, where Q is a finite set of *program locations*, $E \subseteq Q \times \mathbb{Q}[x_1, \ldots, x_n]^n \times Q$ is a finite set of *edges*, and $q_{\text{init}} \in Q$ is the *initial location*. We say that $\mathcal{A}$ is an *affine program* if for every edge $(q, f, q') \in E$, with $f = (f_1, \ldots, f_n)$, each polynomial f_i has degree at most one. We think of $x_1, \ldots, x_n$ as program variables that range over $\mathbb{Q}$ and a transition (p, f, q) as performing a simultaneous assignment $\boldsymbol{x} := f(\boldsymbol{x})$, where $\boldsymbol{x} = (x_1, \ldots, x_n)$.

A *configuration* of $\mathcal{A}$ is a pair $(q, \boldsymbol{a}) \in Q \times \mathbb{Q}^n$. Intuitively, an edge (q, f, q') induces a transition from configuration $(q, \boldsymbol{a})$ to configuration $(q', f(\boldsymbol{a}))$ (under the natural view of f as a function from $\mathbb{Q}^n$ to $\mathbb{Q}^n$). The *collecting semantics* of $\mathcal{A}$ assigns to each location q the set $S_q \subseteq \mathbb{Q}^n$ of all those $\boldsymbol{a} \in \mathbb{Q}^n$ such that the configuration $(q, \boldsymbol{a})$ is reachable from $(q_{\text{init}}, \boldsymbol{0})$. The family $\{S_q : q \in Q\}$ can be characterised as the least solution of the following system of inclusions (see Reference [37]):

$$\begin{array}{rcll} S_{q_{\text{init}}} & \supseteq & \{\boldsymbol{0}\} & \\ S_q & \supseteq & f(S_p) & \text{for all } (p, f, q) \in E\,. \end{array} \tag{1}$$

A family of sets $\mathcal{X} = \{X_q : q \in Q\}$, with $X_q \subseteq \overline{\mathbb{Q}}^n$, is said to be an *inductive invariant* of the program $\mathcal{A}$ if it satisfies the system of inclusions (1), i.e., $X_{q_{\text{init}}} \supseteq \{\boldsymbol{0}\}$ and $X_q \supseteq f(X_p)$ for all $(p, f, q) \in E$. Such a family is moreover said to be an *algebraic inductive invariant* if each X_q is an

[4]Note that our techniques allow us to compute the Zariski closure of affine programs with coefficients in $\overline{\mathbb{Q}}$ (not just $\mathbb{Q}$), in which case the Zariski closure would be defined by polynomials with coefficients in $\overline{\mathbb{Q}}$ also.

algebraic subset of $\overline{\mathbb{Q}}^n$. It is clear that the class of algebraic inductive invariants is closed under intersections (where the intersection of Q-indexed sets is defined pointwise) and hence there is a minimal algebraic inductive invariant.

The minimal inductive algebraic invariant can be characterised as the family of sets $\mathcal{X} = \{X_q : q \in Q\}$ such that $X_q := \overline{S_q}$ for all $q \in Q$, i.e., X_q is the Zariski closure of S_q in $\overline{\mathbb{Q}}^n$. Note that $\mathcal{X}$ is indeed an inductive invariant: For each edge $(p, f, q) \in E$, we have $f(X_p) = f(\overline{S_p}) \subseteq \overline{f(S_p)} \subseteq \overline{S_q} = X_q$, since the polynomial map f is Zariski continuous, and by Equation (1).

As we now explain, the minimal inductive algebraic invariant is determined by the collection of rational polynomial equations that hold at each program location. Given $P \in \mathbb{Q}[x_1, \dots, x_n]$, we say that the equation $P = 0$ holds at a program location $q \in Q$ if P vanishes on S_q. Define $I_q := \mathbf{I}(S_q) \subseteq \overline{\mathbb{Q}}[x_1, \dots, x_n]$ to be the ideal of all polynomials P that vanish on the set S_q. The variety corresponding to ideal I_q is $V_q := \mathbf{V}(I_q) = \overline{S_q}$, i.e., $\{V_q : q \in Q\}$ is the minimal inductive algebraic invariant. When we speak of computing the minimal inductive algebraic invariant, our goal is to compute a basis of the ideal I_q for all locations $q \in Q$. As noted in Section 3.2, the ideal I_q has a basis of polynomials with rational coefficients.

In the remainder of this section, we reduce the problem of computing the Zariski closure of the collecting semantics of an affine program to that of computing the Zariski closure of a related semigroup of matrices. The idea of this reduction is first to replace each affine assignment by a corresponding linear assignment by adding an extra dimension to the program. One then simulates a general affine program by a program with a single location.

Consider an affine program $\mathcal{A} = (Q, E, q_{\text{init}})$, where the set of locations is $Q = \{q_1, \dots, q_m\}$ and $q_{\text{init}} = q_1$. For each edge $e = (q_j, f, q_i)$, we define a square matrix $M^{(e)} \in M_{m(n+1)}(\mathbb{Q})$ comprising an $m \times m$ array of blocks, with each block a matrix in $M_{n+1}(\mathbb{Q})$. If the affine map f is given by $f(\boldsymbol{x}) = A\boldsymbol{x} + \boldsymbol{b}$, then the (i, j)-th block of $M^{(e)}$ is

$$\begin{pmatrix} A & \boldsymbol{b} \\ 0 & 1 \end{pmatrix},$$

while all other blocks are zero. Notice that for $\boldsymbol{x} \in \mathbb{Q}^n$, we have

$$\begin{pmatrix} A & \boldsymbol{b} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ 1 \end{pmatrix} = \begin{pmatrix} A\boldsymbol{x} + \boldsymbol{b} \\ 1 \end{pmatrix} = \begin{pmatrix} f(\boldsymbol{x}) \\ 1 \end{pmatrix}. \tag{2}$$

Given $i \in \{1, \dots, m\}$, define the projection $\Pi_i : \overline{\mathbb{Q}}^{m(n+1)} \to \overline{\mathbb{Q}}^{n+1}$ by $\Pi_i(\boldsymbol{x}_1, \dots, \boldsymbol{x}_m) = \boldsymbol{x}_i$ and define the injection $\text{in}_i : \overline{\mathbb{Q}}^n \to \overline{\mathbb{Q}}^{m(n+1)}$ by

$$\text{in}_i(\boldsymbol{x}) = (\boldsymbol{0}, \dots, (\boldsymbol{x}, 1), \dots, \boldsymbol{0}) \in \overline{\mathbb{Q}}^{m(n+1)},$$

where $(\boldsymbol{x}, 1)$ occurs in the ith block. We denote $\text{in}_1(\boldsymbol{0})$ by $\boldsymbol{v}_{\text{init}}$.

Proposition 2. *Let $\mathcal{M}$ be the semigroup generated by the set of matrices $\{M^{(e)} : e \in E\}$. Then, for $i = 1, \dots, m$, we have*

$$S_{q_i} = \{\boldsymbol{x} \in \mathbb{Q}^n : \text{in}_i(\boldsymbol{x}) \in \{M\boldsymbol{v}_{\text{init}} : M \in \mathcal{M}\}\}.$$

Proof. For an edge $e = (q_i, f, q_j)$ of the affine program $\mathcal{A}$, we have

$$M^{(e)}\text{in}_i(\boldsymbol{x}) = \text{in}_j(f(\boldsymbol{x}))$$

and

$$M^{(e)}\text{in}_k(\boldsymbol{x}) = \boldsymbol{0}$$

for $k \neq i$. Now, consider a sequence of edges

$$e_1 = (q_{i_1}, f_1, q_{j_1}), e_2 = (q_{i_2}, f_2, q_{j_2}), \ldots, e_\ell = (q_{i_\ell}, f_\ell, q_{j_\ell}) .$$

If this sequence is a legitimate execution of $\mathcal{A}$, i.e., $i_1 = 1$ and $j_k = i_{k+1}$ for $k = 1, \ldots, \ell - 1$, then we have

$$M^{(e_\ell)} \cdots M^{(e_1)} \boldsymbol{v}_{\text{init}} = \text{in}_{j_\ell}(f_\ell(\ldots f_1(\mathbf{0}) \ldots)) .$$

If the sequence is not a legitimate execution of $\mathcal{A}$, then we have

$$M^{(e_\ell)} \cdots M^{(e_1)} \boldsymbol{v}_{\text{init}} = \mathbf{0} .$$

From the above it follows that for all $i \in \{1, \ldots, m\}$,

$$S_{q_i} = \{\boldsymbol{x} \in \mathbb{Q}^n : \text{in}_i(\boldsymbol{x}) \in \{M\boldsymbol{v}_{\text{init}} : M \in \mathcal{M}\}\} .$$ □

Theorem 3. *Given an affine program $\mathcal{A}$, we can compute $\{V_q : q \in Q\}$—the Zariski closure of the collecting semantics. This is the smallest algebraic inductive invariant of $\mathcal{A}$.*

Proof. Let $\mathcal{M}$ be the semigroup generated by the set of matrices $\{M^{(e)} : e \in E\}$. From Proposition 2, we have

$$\begin{aligned} S_{q_i} &= \{\boldsymbol{x} \in \mathbb{Q}^n : \text{in}_i(\boldsymbol{x}) \in \{M\boldsymbol{v}_{\text{init}} : M \in \mathcal{M}\}\} \\ &= \{\boldsymbol{x} \in \mathbb{Q}^n : (\boldsymbol{x}, 1) \in \Pi_i (\{M\boldsymbol{v}_{\text{init}} : M \in \mathcal{M}\})\} . \end{aligned}$$

By Theorem 16, we can compute the Zariski closure $\overline{\mathcal{M}}$ of the matrix semigroup $\mathcal{M}$. Since the projection Π_i and the map $M \mapsto M\boldsymbol{v}_{\text{init}}$ are both Zariski continuous, we have that

$$\begin{aligned} S_{q_i} &\subseteq \left\{\boldsymbol{x} \in \overline{\mathbb{Q}}^n : (\boldsymbol{x}, 1) \in \Pi_i \left(\{M\boldsymbol{v}_{\text{init}} : M \in \overline{\mathcal{M}}\}\right)\right\} \\ &\subseteq \overline{S_{q_i}} . \end{aligned}$$

Thus, we can compute $\overline{S_{q_i}}$ as the Zariski closure of

$$\left\{\boldsymbol{x} \in \overline{\mathbb{Q}}^n : (\boldsymbol{x}, 1) \in \Pi_i \left(\{M\boldsymbol{v}_{\text{init}} : M \in \overline{\mathcal{M}}\}\right)\right\} ,$$

since the latter is a constructible set. It is clear that any algebraic invariant must contain the Zariski closure of the collecting semantics. Furthermore, we have already explained at the beginning of this section that this invariant is inductive by the Zariski-continuity of the multiplication map. □

5 ZARISKI CLOSURE OF A SUBGROUP OF $\mathrm{GL}_n(\overline{\mathbb{Q}})$

In this section, we show how to compute the Zariski closure of the subgroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ generated by a given constructible subset of $\mathrm{GL}_n(\overline{\mathbb{Q}})$. We show this by a reduction to the problem of computing the Zariski closure of a finitely generated subgroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$. An algorithm for the latter problem was given by Derksen, Jeandel, and Koiran [13]. Recall that for $X \subseteq \mathrm{GL}_n(\overline{\mathbb{Q}})$, we use $\langle X \rangle$ to denote the *sub-semigroup* of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ generated by X. But, we have:

Lemma 4 ([13]). *A closed subsemigroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ is a subgroup.*

This lemma is useful in conjunction with the following fact: If $S \subseteq \mathrm{GL}_n(\overline{\mathbb{Q}})$ is a subsemigroup, then $\overline{S}$ is a subsemigroup. This is a consequence of the Zariski-continuity of the multiplication map of matrices. In particular, if $X \subseteq \mathrm{GL}_n(\overline{\mathbb{Q}})$, then $\overline{\langle X \rangle}$ is a subgroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$. Our aim is to generalise the following result.

Theorem 5 ([13]). *Given matrices $a_1, \ldots, a_k \in \mathrm{GL}_n(\overline{\mathbb{Q}})$, we can compute the closed subgroup $\overline{\langle a_1, \ldots, a_k \rangle}$.*

The first generalisation is as follows:

Corollary 6. *Let $a_1, \ldots, a_k \in \mathrm{GL}_n(\overline{\mathbb{Q}})$ and let $Y \subseteq \mathrm{GL}_n(\overline{\mathbb{Q}})$ be an irreducible variety containing the identity I_n. Then $\overline{\langle a_1, \ldots, a_k, Y \rangle}$ is computable from Y and the a_i.*

Proof. Let $G = \overline{\langle a_1, \ldots, a_k \rangle}$ and let H be the smallest Zariski closed subgroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ that contains Y and is closed under conjugation by $a_1, \ldots, a_k$ (i.e., such that $a_i H a_i^{-1} \subseteq H$ for $i = 1, \ldots, k$). We claim that $\overline{\langle a_1, \ldots, a_k, Y \rangle} = \overline{G \cdot H}$.

To prove the claim, note that, since H is closed under conjugation by $a_1, \ldots, a_k$, H is also closed under conjugation by any $g \in \langle a_1, \ldots, a_k \rangle$. Moreover, since the map $g \mapsto ghg^{-1}$ is Zariski continuous for each fixed $h \in H$, we have that H is closed under conjugation by any $g \in G = \overline{\langle a_1, \ldots, a_k \rangle}$. It follows that $G \cdot H$ is a sub-semigroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ and so $\overline{G \cdot H}$ is a group by Lemma 4. But

$$\{a_1, \ldots, a_k\} \cup Y \subseteq G \cdot H \subseteq \overline{\langle a_1, \ldots, a_k, Y \rangle}$$

and hence $\overline{G \cdot H} = \overline{\langle a_1, \ldots, a_k, Y \rangle}$.

It remains to show that we can compute $\overline{G \cdot H}$. Now, we can compute G by Theorem 5. To compute H, we use the following algorithm:

Procedure FinPlusIrredClosure($a_1, \ldots, a_k, Y$)

```
input : Irreducible variety Y ⊆ GL_n(Q̄) containing I_n
input : a_1, ..., a_k ∈ GL_n(Q̄)
H := Y
S = {a_1, ..., a_k, I_n}
repeat
    H_old := H
    for y ∈ S do
        H := closure(H · yHy^{-1})
until H_old = H
return H
```

We show that Algorithm FinPlusIrredClosure computes the smallest subgroup H of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ that is Zariski closed, contains Y, and is closed under conjugation by $a_1, \ldots, a_k$. To this end, notice that, since Y contains the identity, the successive values taken by H in the algorithm form an increasing chain of sub-varieties of $\mathrm{GL}_n(\overline{\mathbb{Q}})$. Moreover, by Lemma 1, this chain is in fact an increasing chain of *irreducible* sub-varieties. But such a chain has bounded length, since $\mathrm{GL}_n(\overline{\mathbb{Q}})$ has finite dimension and hence the algorithm must terminate.

We know that $Y \subseteq H$ on termination. Moreover, from the loop termination condition, it is clear that on termination H must be closed under conjugation by $a_1, \ldots, a_k$, and be a Zariski closed sub-semigroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ (and hence a sub-group of $\mathrm{GL}_n(\overline{\mathbb{Q}})$ by Lemma 4). Finally, by construction, H is the smallest such subgroup of $\mathrm{GL}_n(\overline{\mathbb{Q}})$. This concludes the proof. □

We can now prove the main result of this section.

Theorem 7. *Given a constructible subset A of $\mathrm{GL}_n(\overline{\mathbb{Q}})$, we can compute $\overline{\langle A \rangle}$.*

Proof. Let $X_1, \ldots, X_k$ be the irreducible components of $\overline{A}$, which are computable from A. For each i, compute a point $a_i \in X_i$ (see Section 3.3). Form $Y_i = a_i^{-1} X_i$, which is an irreducible variety containing the identity and let $Y = \overline{Y_1 \cdot Y_2 \cdots Y_k}$, which by Lemma 1 is also an irreducible variety containing the identity. Note that Y is computable as the closure of the image of a variety under a polynomial map (the map $(y_1,\ldots,y_k) \mapsto y_1\cdots y_k)$). We then have that $\overline{\langle A\rangle} = \overline{\langle a_1,\ldots,a_k,Y\rangle}$. Indeed,

$$\langle A\rangle \subseteq \langle a_1,\ldots,a_k,Y_1\cdot Y_2\cdots Y_k\rangle \subseteq \overline{\langle A\rangle},$$

where the last inclusion holds, since $\overline{\langle A\rangle}$ is a group by Lemma 4, $a_i, a_i^{-1} \in \overline{\langle A\rangle}$ and hence $Y_i \subseteq \overline{\langle A\rangle}$. It follows that

$$\begin{aligned}\overline{\langle A\rangle} &= \overline{\langle a_1,\ldots,a_k,Y_1\cdot Y_2\cdots Y_k\rangle}\\ &= \overline{\left\langle a_1,\ldots,a_k,\overline{Y_1\cdot Y_2\cdots Y_k}\right\rangle}.\end{aligned}$$

We can compute the closure of $\langle a_1,\ldots,a_k,Y\rangle$, thanks to Corollary 6. □

6 ZARISKI CLOSURE OF A FINITELY GENERATED MATRIX SEMIGROUP

In this section, we give a procedure to compute the Zariski closure of a finitely generated matrix semigroup. We proceed by induction on the rank of the generators. To this end, it is useful to generalise from finite sets of generators to constructible sets of generators. In particular, we will use Theorem 7 on the computability of the Zariski closure of the group generated by a constructible set of invertible matrices.

We first introduce a graph structure on the set of generators that allows us to reason about all products of generators that have a given rank.

6.1 A Generating Graph

Given integers n and r, let $A \subseteq M_n(\overline{\mathbb{Q}})$ be a set of matrices of rank r. We define a labelled directed graph $\mathcal{K}(A)$ as follows:

- There is a vertex (U,V) for each pair of subspaces $U,V \subseteq \overline{\mathbb{Q}}^n$ such that $\dim(V) = r$, $\dim(U) = n-r$, and $U \cap V = 0$.
- There is a labelled edge $(U,V) \xrightarrow{a} (U',V')$ for each pair of vertices (U,V) and (U',V'), and each matrix $a \in A$ such that $\ker(a) = U$ and $\operatorname{im}(a) = V'$.

We note in passing that $\mathcal{K}(A)$ can be seen as an edge-induced subgraph of the *Karoubi envelope* [46] of the semigroup $M_n(\overline{\mathbb{Q}})$.

A *path* in $\mathcal{K}(A)$ is a non-empty sequence of consecutive edges

$$(U_0,V_0) \xrightarrow{a_1} (U_1,V_1) \xrightarrow{a_2} (U_2,V_2) \xrightarrow{a_3} \ldots \xrightarrow{a_m} (U_m,V_m).$$

The length of such a path is m and its *label* is the product $a := a_m\cdots a_1$. Matrix a has rank r, since $\ker(a_{i+1})\cap\operatorname{im}(a_i) = 0$ for $i = 1,\ldots,m-1$. It is moreover clear that $\{a \in \langle A\rangle : \operatorname{rk}(a) = r\}$ is precisely the set of labels over all paths in $\mathcal{K}(A)$. We will denote that there is a path from (U,V) to (U',V') with label a by writing $(U,V) \overset{a}{\Rightarrow} (U',V')$.

The following sequence of propositions concerns the structure of the **strongly connected components (SCCs)** in $\mathcal{K}(A)$. The respective proofs make repeated use of the fact that for each vertex (U,V) of $\mathcal{K}(A)$, we have $\iota(U) \wedge \iota(V) \neq 0$ and that $\dim \Lambda^r(\overline{\mathbb{Q}}^n) = \binom{n}{r}$ (cf. Section 3). We say that an SCC of $\mathcal{K}(A)$ is *non-trivial* if it contains a vertex (U,V) such that there is a path from (U,V) back to itself. Figure 1 summarises the structural results on $\mathcal{K}(A)$.

Proposition 8. *$\mathcal{K}(A)$ has at most $\binom{n}{r}$ non-trivial SCCs.*

Proof. Let $(U_1,V_1),\ldots,(U_m,V_m)$ be an arbitrary finite set of vertices drawn from distinct non-trivial SCCs of $\mathcal{K}(A)$. To prove the proposition it suffices to show that $m \le \binom{n}{r}$.

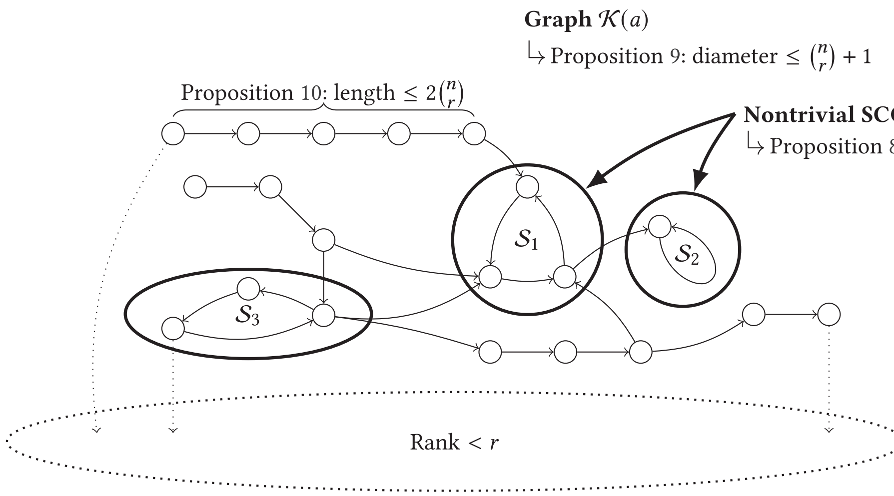


Fig. 1. Graphical representation of $\mathcal{K}(A)$, vertex and edge labels omitted for clarity. Note that the graph can have infinitely many vertices. Proposition 8 shows that there are only finitely many nontrivial SCCs. Proposition 9 shows that the graph has finite diameter. Proposition 10 shows that paths avoiding nontrivial SCCs must be short. All paths in $\mathcal{K}(A)$ are labelled by rank r matrices. Dotted arrows represent products in the semigroup where the rank becomes less than r: those products do not correspond to labels in $\mathcal{K}(A)$ and need to be handled separately.

Assume that the vertices $(U_1, V_1), \ldots, (U_m, V_m)$ are given according to a topological ordering of SCCs—so there is no path from (U_j, V_j) back to (U_i, V_i) for $i < j$. By assumption, for $i = 1, \ldots, m$ there exists a path $(U_i, V_i) \overset{a_i}{\Longrightarrow} (U_i, V_i)$.

On the one hand, for all $1 \leq i < j \leq m$, we have $\iota(U_i) \wedge \iota(V_j) = 0$ (equivalently, $U_i \cap V_j \neq 0$)—for otherwise there would be a path

$$(U_j, V_j) \overset{a_j}{\Longrightarrow} (U_i, V_j) \overset{a_i}{\Longrightarrow} (U_i, V_i),$$

contrary to the topological ordering. On the other hand, we have that $\iota(U_j) \wedge \iota(V_j) \neq 0$ (equivalently, $U_j \cap V_j = 0$) for all $j \in \{1, \ldots, m\}$ by definition of $\mathcal{K}(A)$. It follows that for all $j \in \{1, \ldots, m\}$,

$$\iota(U_j) \notin \operatorname{span}\{\iota(U_i) : i = 1, \ldots, j-1\},$$

since any element U in this span satisfies $\iota(U) \wedge \iota(V_j) = 0$ by bilinearity of the wedge product. We conclude that

$$\dim \operatorname{span}\{\iota(U_i) \in \Lambda^r(\overline{\mathbb{Q}}^n) : i = 1, \ldots, j\} = j$$

for all $1 \leq j \leq m$ and hence $m \leq \dim \Lambda^r(\overline{\mathbb{Q}}^n) = \binom{n}{r}$, as we wished to prove. □

Proposition 9. *If there is a path from (U, V) to (U', V') in $\mathcal{K}(A)$, then there is a path from (U, V) to (U', V') of length at most $\binom{n}{r} + 1$.*

Proof. Let

$$(U_0, V_0) \xrightarrow{a_1} (U_1, V_1) \xrightarrow{a_2} \ldots \xrightarrow{a_m} (U_m, V_m) \tag{3}$$

be a shortest path from $(U_0, V_0) = (U, V)$ to $(U_m, V_m) = (U', V')$. By construction, we have that $U_i \cap V_i = 0$ for $i = 0, \ldots, m$. Furthermore, we have $U_j \cap V_i \neq 0$ for all $0 < i < j < m$, for otherwise, we would have a shortcut

$$(U_{i-1}, V_{i-1}) \xrightarrow{a_i} (U_j, V_i) \xrightarrow{a_{j+1}} (U_{j+1}, V_{j+1}),$$

contradicting the minimality of Equation (3). But then $\iota(V_j) \notin \operatorname{span}\{\iota(V_i) : 1 \le i < j\}$ for $j = 1, \ldots, m-1$: indeed, any element V in this span satisfies $\iota(U_j) \wedge \iota(V) = 0$ by bilinearity of the wedge product, but we know that $\iota(U_j) \wedge \iota(V_j) \neq 0$. We conclude that

$$\dim \operatorname{span}\{\iota(V_i) \in \Lambda^r(\overline{\mathbb{Q}}^n) : i \in \{1, \ldots, j\}\} = j$$

for all $j = 1, \ldots, m-1$. It follows that $m - 1 \le \binom{n}{r}$. □

Proposition 10. *Given any path $(U_0, V_0) \xrightarrow{a_1} (U_1, V_1) \xrightarrow{a_2} \ldots \xrightarrow{a_m} (U_m, V_m)$ in $\mathcal{K}(A)$, where $m = 2\binom{n}{r}$, some vertex (U_i, V_i) lies in a non-trivial SCC.*

Proof. The set of $\binom{n}{r} + 1$ vectors $\{\iota(U_0), \iota(U_2), \iota(U_4), \ldots, \iota(U_m)\}$ is linearly dependent, since $\dim \Lambda^r(\overline{\mathbb{Q}}^n) = \binom{n}{r}$. Thus, there must exist $i \in \{0, \ldots, m\}$ such that $\iota(U_i) \in \operatorname{span}\left\{\iota(U_j) : j \le i-2\right\}$. Now, by definition of $\mathcal{K}(A)$, we have $U_i \cap V_i = 0$ and hence $\iota(U_i) \wedge \iota(V_i) \neq 0$. Thus, by bilinearity of the wedge product there must exist $j \le i-2$ such that $\iota(U_j) \wedge \iota(V_i) \neq 0$, that is, $U_j \cap V_i = 0$. But then we have a path

$$(U_{i-1}, V_{i-1}) \xrightarrow{a_i} (U_j, V_i) \xrightarrow{a_{j+1}} (U_{j+1}, V_{j+1}),$$

showing that (U_{i-1}, V_{i-1}) and (U_{j+1}, V_{j+1}) lie in the same (necessarily non-trivial) SCC. Indeed, recall that $j \le i-2$, so either $(U_{j+1}, V_{j+1}) \Rightarrow (U_{i-1}, V_{i-1})$ or $(U_{j+1}, V_{j+1}) = (U_{i-1}, V_{i-1})$ in the original path. □

6.2 Adding Pseudo-inverses

We now focus on individual SCCs within $\mathcal{K}(A)$. Let $\mathcal{S}$ be such a non-trivial SCC. For each edge $(U, V) \xrightarrow{a} (U', V')$ in $\mathcal{S}$, define its *pseudo-inverse* to be a directed edge $(U', V') \xrightarrow{a^+} (U, V)$, where $a^+ \in M_n(\overline{\mathbb{Q}})$ is the unique matrix such that $\ker(a^+) = U'$, $\operatorname{im}(a^+) = V$, $a^+av = v$ for all $v \in V$, and $aa^+v = v$ for all $v \in V'$. We write $\mathcal{S}^+$ for the graph obtained from $\mathcal{S}$ by adding pseudo-inverses of every edge in $\mathcal{S}$.

The graph $\mathcal{S}^+$ can be seen as the generator of a groupoid in which the above-defined pseudo-inverse matrices are genuine inverses. We do not develop this idea, except to observe that not only edges but also paths in $\mathcal{S}$ have pseudo-inverses in $\mathcal{S}^+$. Specifically, given a path $(U, V) \overset{a}{\Rightarrow} (U', V')$ in $\mathcal{S}$, one obtains a path $(U', V') \overset{a^+}{\Longrightarrow} (U, V)$ in $\mathcal{S}^+$ by taking the pseudo-inverse of each constituent edge. In the remainder of this section, we show that the pseudo-inverses of all paths in $\mathcal{S}$ are already present in the Zariski closure $\overline{\langle A \rangle}$.

Proposition 11. *Let (U, V) be a vertex of $\mathcal{S}$ and let $B \subseteq M_n(\overline{\mathbb{Q}})$ be a constructible set of matrices such that there is a path $(U, V) \overset{b}{\Rightarrow} (U, V)$ in $\mathcal{S}$ for all $b \in B$. Then, $\overline{\langle B \rangle}$ is computable from B and for every $b \in \langle B \rangle$ the pseudo-inverse $(U, V) \overset{b^+}{\Longrightarrow} (U, V)$ is such that $b^+ \in \overline{\langle B \rangle}$.*

Proof. By construction, all elements of B have kernel U and image V, where $U \oplus V = \overline{\mathbb{Q}}^n$. Thus, there is an invertible matrix $y \in \mathrm{GL}_n(\overline{\mathbb{Q}})$ such that for every $b \in B$ there exists $c \in \mathrm{GL}_r(\overline{\mathbb{Q}})$ with

$$y^{-1}by = \begin{bmatrix} c & 0 \\ 0 & 0 \end{bmatrix}. \tag{4}$$

Such a matrix can be computed from U and V as follows: Let $u_1, \ldots, u_{n-r}$ be a basis of U and $v_1, \ldots, v_r$ a basis of V, and define y to be the matrix $y = \begin{bmatrix} v_1 & \cdots & v_r & u_1 & \cdots & u_{n-r} \end{bmatrix}$. This matrix is invertible, because $U \oplus V = \overline{\mathbb{Q}}^n$ and, given $b \in B$, one easily checks that $y^{-1}by$ has the form shown in Equation (4). Let

$$C := \left\{ c \in \mathrm{GL}_r(\overline{\mathbb{Q}}) : \exists b \in B \,. y^{-1}by = \begin{bmatrix} c & 0 \\ 0 & 0 \end{bmatrix} \right\},$$

which is constructible. We can compute $\overline{\langle C \rangle}$ (the Zariski closure of $\langle C \rangle$ in the variety $\mathrm{GL}_r(\overline{\mathbb{Q}})$) using Theorem 7. But then

$$\left\{ y \begin{bmatrix} c & 0 \\ 0 & 0 \end{bmatrix} y^{-1} : c \in \overline{\langle C \rangle} \right\}$$

is a constructible subset of $M_n(\overline{\mathbb{Q}})$ whose closure equals $\overline{\langle B \rangle}$. Note that we are using the fact that $\overline{\langle C \rangle}$ is a subvariety of $\mathrm{GL}_n(\overline{\mathbb{Q}})$, thus, it is constructible in $M_n(\overline{\mathbb{Q}})$. Finally, if $b = y \left[\begin{smallmatrix} c & 0 \\ 0 & 0 \end{smallmatrix}\right] y^{-1} \in \langle B \rangle$, then $b^+ = y \begin{bmatrix} c^{-1} & 0 \\ 0 & 0 \end{bmatrix} y^{-1} \in \overline{\langle B \rangle}$, since $c^{-1} \in \overline{\langle C \rangle}$ (which is a group by Lemma 4). □

Corollary 12. *Suppose that $(U, V) \overset{a}{\Rightarrow} (U', V')$ is a path in $\mathcal{S}$ with pseudo-inverse $(U', V') \overset{a^+}{\Longrightarrow} (U, V)$. Then, $a^+ \in \overline{\langle A \rangle}$.*

Proof. Since $\mathcal{S}$ is strongly connected, there is a path $(U', V') \overset{b}{\Rightarrow} (U, V)$. Consider the path $(U, V) \overset{ba}{\Longrightarrow} (U, V)$ and its pseudo-inverse $(U, V) \overset{(ba)^+}{\Longrightarrow} (U, V)$. By Proposition 11, we have $(ba)^+ \in \overline{\langle A \rangle}$. We moreover have $a^+ = a^+b^+b = (ba)^+b$ and hence $a^+ \in \overline{\langle A \rangle}$, since $\overline{\langle A \rangle}$ is a semigroup. □

6.3 Maximum-rank Matrices in the Closure

Let $\mathcal{S}$ be a non-trivial SCC in $\mathcal{K}(A)$. Write $B \subseteq M_n(\overline{\mathbb{Q}})$ for the set of labels of all paths in $\mathcal{S}^+$ of length at most $\binom{n}{r} + 2$. Moreover, fix a vertex (U_*, V_*) in $\mathcal{S}^+$ and write B_* for the set of labels of all paths in $\mathcal{S}^+$ of length at most $2\binom{n}{r} + 3$ that are self-loops on (U_*, V_*).

Proposition 13. *Let $\langle \mathcal{S} \rangle$ denote the set of labels of all paths in $\mathcal{S}$. Then,*

$$\langle \mathcal{S} \rangle \subseteq B\overline{\langle B_* \rangle}B \subseteq \overline{\langle A \rangle}.$$

Proof. By Corollary 12, we have that $B, B_* \subseteq \overline{\langle A \rangle}$. Thus, the right-hand inclusion follows from the fact that $\overline{\langle A \rangle}$ is a semigroup.

To establish the left-hand inclusion, consider a path

$$(U_0, V_0) \xrightarrow{a_1} (U_1, V_1) \xrightarrow{a_2} (U_2, V_2) \xrightarrow{a_3} \ldots \xrightarrow{a_n} (U_n, V_n)$$

within $\mathcal{S}$. Proposition 9 ensures that for each vertex (U_i, V_i) there is a path $(U_*, V_*) \overset{f_i}{\Rightarrow} (U_i, V_i)$ in $\mathcal{S}$ of length at most $\binom{n}{r} + 1$. Such a path has a pseudo-inverse $(U_i, V_i) \overset{f_i^+}{\Longrightarrow} (U_*, V_*)$ in $\mathcal{S}^+$. Now, by the definition of a pseudo-inverse, we have $a_i f_{i-1} f_{i-1}^+ = a_i$ for all $i \in \{1, \ldots, n\}$. Thus, we have (cf. Figure 2):

$$\begin{aligned} a_n \ldots a_2 a_1 &= a_n f_{n-1} f_{n-1}^+ a_{n-1} f_{n-2} f_{n-2}^+ \cdots f_2 f_2^+ a_2 f_1 f_1^+ a_1 \\ &= a_n f_{n-1} (f_{n-1}^+ a_{n-1} f_{n-2}) \cdots (f_2^+ a_2 f_1) f_1^+ a_1 \,. \end{aligned}$$

The result follows from the observation that $a_n f_{n-1}$ and $f_1^+ a_1$ are both elements of B and that $f_i^+ a_i f_{i-1} \in B_*$ for $i = 2, \ldots, n-1$. □

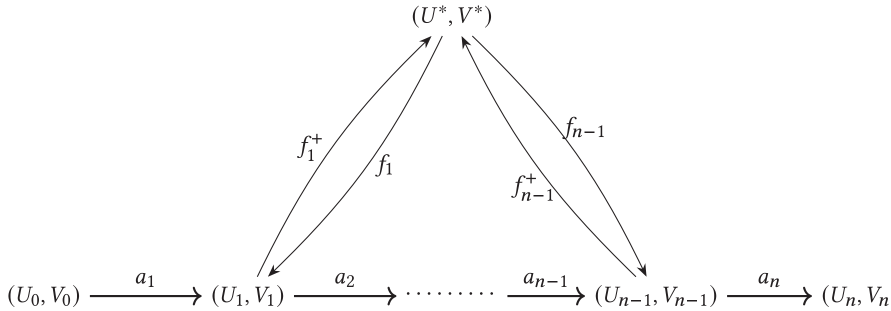


Fig. 2. Expressing a path in an SCC $\mathcal{S}$ in terms of "short" cycles on the distinguished vertex (U_*, V_*).

Recall from Proposition 8 that the graph $\mathcal{K}(A)$ has at most $\binom{n}{r}$ non-trivial SCCs. Let $\mathcal{S}_1, \ldots, \mathcal{S}_\ell$ be a list of the non-trivial SCCs in $\mathcal{K}(A)$ and write

$$\mathsf{P} := A \cup \langle \mathcal{S}_1 \rangle \cup \cdots \cup \langle \mathcal{S}_\ell \rangle \ . \tag{5}$$

Lemma 14. *Given $a \in \langle A \rangle$ with $\mathrm{rk}(a) = r$, we have $a \in \mathsf{P} \cup \mathsf{P}^2 \cup \cdots \cup \mathsf{P}^\kappa$, where $\kappa = 2\binom{n}{r}^2 + 3\binom{n}{r}$.*

Proof. Suppose that a is the label of a path

$$(U_0, V_0) \xrightarrow{a_1} (U_1, V_1) \xrightarrow{a_2} (U_2, V_2) \xrightarrow{a_3} \ldots \xrightarrow{a_m} (U_m, V_m) \tag{6}$$

in $\mathcal{K}(A)$. The vertices along this path can be partitioned into maximal blocks of contiguous vertices all lying in the same SCC of $\mathcal{K}(A)$. By Proposition 8 there are at most $\binom{n}{r}$ such blocks corresponding to non-trivial SCCs. The remaining blocks, corresponding to trivial SCCs, are singletons. By Proposition 10 there can be at most $2\binom{n}{r}$ consecutive such blocks anywhere along the path. We conclude that there at most $\kappa = 2\binom{n}{r}^2 + 3\binom{n}{r}$ blocks in total.

Now, we can factor the path into single edges that connect vertices in different blocks and subpaths all of whose vertices lie in the same block. There are at most κ such factors (the same as the number of blocks) and the label of each factor lies in the set P defined in Equation (5). This completes the proof. ◻

Let $R_r = \{x \in M_n(\overline{\mathbb{Q}}) : \mathrm{rk}(x) = r\}$, which is a constructible set, and $R_{<r} = \{x \in M_n(\overline{\mathbb{Q}}) : \mathrm{rk}(x) < r\}$, which is closed.

Proposition 15. *Let $A \subseteq M_n(\overline{\mathbb{Q}})$ be a constructible set of matrices, all of rank r. Then, we can compute $\overline{\langle A \rangle} \cap R_r$ from A.*

Proof. By Proposition 13 (see also Section 6.5 for remarks on effectiveness), for $i = 1, \ldots, \ell$, we can compute a constructible set $E_i \subseteq M_n(\overline{\mathbb{Q}})$ such that $\langle \mathcal{S}_i \rangle \subseteq E_i \subseteq \overline{\langle A \rangle}$. Writing $E := A \cup E_1 \cup \ldots \cup E_\ell$, we have $\mathsf{P} \subseteq E \subseteq \overline{\langle A \rangle}$.

By Lemma 14, we have $\langle A \rangle \cap R_r \subseteq X$, where $X := E \cup E^2 \cup \ldots \cup E^{2\binom{n}{r}^2 + 3\binom{n}{r}}$. Now,

$$\begin{aligned} \langle A \rangle \cap R_r &\subseteq X \subseteq \overline{\langle A \rangle} \\ \overline{\langle A \rangle \cap R_r} &\subseteq \overline{X} \subseteq \overline{\langle A \rangle} \\ \overline{\langle A \rangle \cap R_r} \cap R_r &\subseteq \overline{X} \cap R_r \subseteq \overline{\langle A \rangle} \cap R_r . \end{aligned}$$

We claim that

$$\overline{\langle A \rangle \cap R_r} \cap R_r = \overline{\langle A \rangle} \cap R_r, \tag{7}$$

which shows that

$$\overline{\langle A\rangle} \cap R_r = \overline{X} \cap R_r$$

is constructible and computable. It remains to see why Equation (7) holds. Since all matrices in A have rank r, all matrices in $\langle A\rangle$ have rank r or less, thus,

$$\begin{aligned}
\langle A\rangle &= \Big(\langle A\rangle \cap R_r\Big) \cup \Big(\langle A\rangle \cap R_{<r}\Big)\\
\overline{\langle A\rangle} &= \overline{\langle A\rangle \cap R_r} \cup \overline{\langle A\rangle \cap R_{<r}}\\
\overline{\langle A\rangle} \cap R_r &= \Big(\overline{\langle A\rangle \cap R_r} \cap R_r\Big) \cup \underbrace{\Big(\overline{\langle A\rangle \cap R_{<r}} \cap R_r\Big)}_{=\varnothing}.
\end{aligned}$$

Indeed, $\langle A\rangle \cap R_{<r} \subseteq R_{<r}$, thus, $\overline{\langle A\rangle \cap R_{<r}} \subseteq R_{<r}$, because $R_{<r}$ is closed, and $R_{<r} \cap R_r = \varnothing$.

□

6.4 Computing the Closure

We now present the main result of the article.

Theorem 16. *Given a constructible set of matrices $A \subseteq M_n(\overline{\mathbb{Q}})$, one can compute $\overline{\langle A\rangle}$—the Zariski closure of the semigroup generated by A.*

Proof. The proof is by induction on the maximum rank r of the matrices in A. The base case $r = 0$ is trivial. For the induction step, write $A_r := \{a \in A : \mathrm{rk}(a) = r\}$ for the subset of matrices in A of maximum rank and $B := \{a \in \overline{\langle A_r\rangle} : \mathrm{rk}(a) = r\}$. Now, B is computable by Proposition 15.

We claim that $\overline{\langle A\rangle} = \overline{B} \cup \overline{\langle C\rangle}$, where

$$C = \{a \in A \cup BA \cup AB \cup BAB : \mathrm{rk}(a) < r\}\ .$$

The theorem follows from the claim, since $\overline{\langle C\rangle}$ is computable by the induction hypothesis.

It remains to prove the claim. For the right-to-left inclusion, notice that, since $A, B \subseteq \overline{\langle A\rangle}$ and $\overline{\langle A\rangle}$ is a Zariski-closed semigroup, $\overline{\langle A\rangle}$ contains both $\overline{B}$ and $\overline{\langle C\rangle}$.

For the left-to-right inclusion, it suffices to show that $\langle A\rangle \subseteq \overline{B} \cup \overline{\langle C\rangle}$. To this end, consider a non-empty product $a := a_1 a_2 \cdots a_m$, where $a_1, \ldots, a_m \in A$. Suppose first that $\mathrm{rk}(a) = r$. Then, of course, $a_1, \ldots, a_m \in A_r$ and hence $a \in B$. Suppose now that $\mathrm{rk}(a) < r$. We show that $a \in \langle C\rangle$ by induction on m. Let $a_1 \cdots a_\ell$ be a prefix of minimum length that has rank less than r. Clearly, such a prefix lies in $A \cup BA$. Moreover, the corresponding suffix $a_{\ell+1} \cdots a_m$ is either empty, has rank r (and hence is in B), or has rank $< r$ and hence is in $\langle C\rangle$ by induction. In all cases, we have that $a \in \langle C\rangle$. □

6.5 Effectively Representing the Generating Graph

We conclude by filling in some details about how the generating graph $\mathcal{K}(A)$ can be effectively represented and thereby how one enumerates the (finitely many) non-trivial SCCs and, given a representative of each SCC, computes the sets B and B^* described in Proposition 13. Throughout this section, by *definable*, we mean first-order definable by a formula in the language of rings with parameters from $\overline{\mathbb{Q}}$.

Let $A \subseteq M_n(\overline{\mathbb{Q}})$ be a constructible set of matrices. Representing vector spaces by bases, the set of vertices of the generating graph $\mathcal{K}(A)$ is a definable set. More precisely, since the same vector space has many different bases, the set of vertices is the quotient of a definable set by a definable equivalence relation. Then, for every fixed $m \in \mathbb{N}$, the binary relation of two vertices being connected in $\mathcal{K}(A)$ by a path of length m is effectively definable if one already has a formula defining the set of matrices A: Indeed, there is a length-m path from (U, V) to (U', V') if there exist

$a_1, \ldots, a_m \in A$ with $\ker(a_m \cdots a_1) = U$ and $\Im(a_m \cdots a_1) = V'$. Thanks to Propositions 9 and 10, it follows in turn that the binary reachability relation on $\mathcal{K}(A)$ is also effectively definable and hence also the binary relation of two nodes being in the same SCC of $\mathcal{K}(A)$. Recall that Proposition 8 gives an upper bound on the number of SCCs of $\mathcal{K}(A)$. One can now enumerate a finite list of nodes of $\mathcal{K}(A)$ that contains precisely one representative of each non-trivial SCC: Such a list can be constructed by an iterative process that at each stage maintains a formula defining all nodes lying in a non-trivial SCC other than the SCCs of the representatives found so far, and then uses the procedure described in Section 3.3 to pick an algebraic point in this set, which thus becomes a representative of a new SCC. Finally, given a representative node (U_*, V_*) of an SCC $\mathcal{S}$, the sets B and B_* in Proposition 13 are also effectively definable, since they are labels of paths of bounded length. Note here that it is easy to include pseudo-inverses in the set B_*, since the property of being a pseudo-inverse is first-order definable.

7 UNDECIDABILITY

In this section, we show that there is no algorithm that computes the minimal algebraic invariant of a polynomial program. In fact, we show that one cannot decide whether the minimal algebraic invariant has dimension at most one. We prove this by reduction from the problem of deciding boundedness of ***reset vector addition systems with states*** **(reset VASS)**—which is undecidable [14]. We refer to Section 3.2 for the definition of the dimension of an algebraic set. Below, we will also give an algebraic characterisation of dimension in terms of polynomial equations.

Syntactically, we can consider a reset VASS as a special kind of affine program. However, VASS have a different semantics to affine programs, since the program variables in a VASS are only allowed to assume nonnegative-integer values. Formally, we define a reset VASS to be an affine program $\mathcal{A} = (Q, E, q_{\text{init}})$ such that for each edge $(q, (f_1, \ldots, f_n), q') \in E$, for all $i \in \{1, \ldots, n\}$ the polynomial $f_i \in \mathbb{Q}[x_1, \ldots, x_n]$ lies in the set $\{0, x_i, x_i + 1, x_i - 1\}$. Intuitively a reset VASS corresponds to a program in which variables can only be incremented, decremented, and reset to zero and moreover in which every transition that attempts to decrement a zero variable is blocked. The nonnegativity requirement on the program variables of a VASS is formalised by modifying the definition of the collecting semantics (cf. Equation (1)). For the given reset VASS $\mathcal{A}$, we define the collection of reachable counter values $S_q \subseteq \mathbb{Z}^n_{\geq 0}$ in location q to be the least solution of the following system of inclusions:

$$\begin{array}{rcl} S_{q_{\text{init}}} & \supseteq & \{\mathbf{0}\} \\ S_q & \supseteq & f(S_p) \cap \mathbb{Z}^n_{\geq 0} \quad \text{for all } (p, f, q) \in E\,. \end{array} \tag{8}$$

In the *Boundedness Problem for Reset VASS* the input is a reset VASS $\mathcal{A} = (Q, E, q_{\text{init}})$ and a distinguished location $q \in Q$, and the question is whether the set S_q of reachable counter values in location q is finite. This problem is undecidable [14].

In the remainder of the section, we reduce the Boundedness Problem for Reset VASS to the problem of computing the minimal algebraic invariant of a polynomial program. Let $\mathcal{A} = (Q, E, q_{\text{init}})$ be a reset VASS in dimension n. The idea is to define a polynomial program $\mathcal{A}'$ in dimension $n+1$ whose computations simulate those of $\mathcal{A}$. We think of a configuration $(q, \boldsymbol{a})$ of $\mathcal{A}$ as being represented by any configuration $(q, \boldsymbol{b})$ of $\mathcal{A}'$ such that $b_{n+1} \neq 0$ and $a_i = b_i / b_{n+1}$ for $i = 1, \ldots, n$. We simulate updates in $\mathcal{A}$ by homogeneous updates in $\mathcal{A}'$, e.g., an increment operation $x_i := x_i + 1$ in $\mathcal{A}$ is simulated in $\mathcal{A}'$ by the instruction $x_i := x_i + x_{n+1}$. Likewise a reset operation $x_i := 0$ in $\mathcal{A}$ is simulated in $\mathcal{A}'$ by the syntactically identical operation $x_i := 0$. The value of x_{n+1} is initialised to 1 by the first transition of $\mathcal{A}'$.

Note that we can "rescale" a configuration of $\mathcal{A}'$ by multiplying all components by a nonzero scalar $\lambda \in \mathbb{Z}$ without changing the encoded configuration of $\mathcal{A}$. We use this fact to simulate in $\mathcal{A}'$

the semantic requirement that the variables of $\mathcal{A}$ remain nonnegative. For example, after executing an assignment $x_i := x_i - x_{n+1}$ in $\mathcal{A}'$ (representing a decrement in $\mathcal{A}$), we immediately perform a simultaneous update $x_j := x_j(x_i + x_{n+1})$, $j = 1, \dots, n+1$. Applying such an assignment to a vector $\boldsymbol{b} \in \mathbb{N}^{n+1}$, the resulting vector has the form $\lambda\boldsymbol{b}$, where the scaling factor λ is equal to zero if and only if $b_i/b_{n+1} = -1$. Hence, any run of $\mathcal{A}$ that leads to a negative counter value is simulated in $\mathcal{A}'$ by a run that leads to (and forever remains in) the zero configuration.

Proceeding more formally, given an update polynomial $f(x_1, \dots, x_n) = cx_i + d$ occurring in $\mathcal{A}$, where $(c, d) \in \{(0, 0), (1, -1), (1, 0), (1, 1)\}$, we define a corresponding homogeneous map $f^*(x_1, \dots, x_{n+1}) := cx_i + dx_{n+1}$. Using this notation, we define the polynomial automaton $\mathcal{A}' = (Q', E', q'_{\text{init}})$ as follows:

(1) The set of locations is $Q' := Q \cup \{q'_{\text{init}}\}$, where $q'_{\text{init}} \notin Q$ is the initial location.
(2) For each edge $(q, (f_1, \dots, f_n), q') \in E$ there is an edge $(q, (g_1, \dots, g_{n+1}), q') \in E'$ such that

$$g_i(\boldsymbol{x}) := h(\boldsymbol{x}) \cdot f_i^*(\boldsymbol{x}) \text{ for all } i \in \{1, \dots, n\},$$
$$g_{n+1}(\boldsymbol{x}) := h(\boldsymbol{x}) \cdot x_{n+1},$$

where $h(\boldsymbol{x}) = 2(f_1^*(\boldsymbol{x}) + x_{n+1}) \cdots (f_n^*(\boldsymbol{x}) + x_{n+1})$.
(3) There is an edge $(q'_{\text{init}}, (f_1, \dots, f_{n+1}), q_{\text{init}}) \in E'$, where $f_i(\boldsymbol{x}) := 0$ for $i \in \{1, \dots, n\}$ and $f_{n+1}(\boldsymbol{x}) := 1$.

In Item 2, the term $h(\boldsymbol{x})$ can be thought of as a scaling factor that becomes zero when the state of the polynomial program encodes a VASS configuration with negative counter values.

Denote the collecting semantics of $\mathcal{A}$ by the indexed family of sets $\{S_q : q \in Q\}$ and similarly denote the collecting semantics of $\mathcal{A}'$ by $\{S'_q : q \in Q'\}$.

Proposition 17. *For all $q \in Q$, S_q is finite if and only if $\overline{S'_q}$ has dimension at most one.*

Proof. As described above, the construction of $\mathcal{A}'$ is such that for all $\boldsymbol{a} \in S_q$ there exists $\boldsymbol{b} \in S'_q$ such that $b_{n+1} \neq 0$ and $a_i = b_i/b_{n+1}$ for $i = 1, \dots, n$ and, conversely, for all $\boldsymbol{b} \in S'_q$ such that $b_{n+1} \neq 0$ the vector $\boldsymbol{a} \in \mathbb{Z}^n$ defined by $a_i = b_i/b_{n+1}$, $i = 1, \dots, n$, lies in S_q. Moreover, the only $\boldsymbol{b} \in S'_q$ such that $b_{n+1} = 0$ is $\boldsymbol{b} = \boldsymbol{0}$.

Let $q \in Q$ and suppose that S_q is finite. For each configuration $(q, \boldsymbol{a}) \in S_q$ the corresponding configurations $(q, \boldsymbol{b})$ in S'_q, with $a_i = b_i/b_{n+1}$ for $i = 1, \dots, n$, all lie on a common line through the origin in $\mathbb{Q}^{n+1}$. Thus, $\overline{S'_q}$ is contained in a finite union of lines and thereby has dimension at most one.

Now, suppose that S_q is infinite. Without loss of generality, say that $\{a_1 : \boldsymbol{a} \in S_q\}$ is infinite. We will show that $\overline{S'_q}$ has dimension at least two. For this, it will suffice to show that no non-zero polynomial that mentions only the variables x_1 and x_{n+1} vanishes on S'_q. Here, we use the fact that the dimension of an affine variety $X \subseteq \overline{\mathbb{Q}}^{n+1}$ is equal to the largest number d for which there exist d variables $x_{i_1}, \dots, x_{i_d}$ such that no non-zero polynomial mentioning only variables $x_{i_1}, \dots, x_{i_d}$ vanishes on X (see, e.g., Reference [11, Chapter 9, Section 5]).

By assumption, $\{b_1/b_{n+1} : \boldsymbol{b} \in S'_q, b_{n+1} \neq 0\}$ is infinite. Since each transition of $\mathcal{A}'$ multiplies the value of the variable x_{n+1} by at least two and increases the value of the quotient x_1/x_{n+1} by at most one, we deduce that for all $\ell \in \mathbb{N}$ there exists $\boldsymbol{b} \in S'_q$ such that $b_1/b_{n+1} = \ell$ and $b_{n+1} \geq 2^\ell$. It is now straightforward that the only polynomial mentioning only the variables x_1 and x_{n+1} that vanishes on S'_q is the zero polynomial. Indeed, consider such a polynomial F and denote by $G(y, x_{n+1})$ the polynomial that is obtained from F by substituting $x_{n+1}y$ for x_1. Since this substitution maps distinct monomials of F to distinct monomials of G, it suffices to show that G is the zero polynomial. But, by construction, for all $\ell \in \mathbb{N}$ there exists $m \geq 2^\ell$ such that

$G(\ell, m) = 0$. By a simple argument on dominating terms, this entails that G is identically zero. This concludes the argument that $\overline{S'_q}$ has dimension at least two and the proof of the proposition is complete. □

Theorem 18. *There is no algorithm that computes the Zariski closure of the collecting semantics of a given polynomial program.*

Proof. Given a representation of an algebraic set as the zero set of a polynomial ideal, we can compute its dimension (see, e.g., Reference [11, Chapter 9, Section 3]). Hence, if we can compute the Zariski closure of the collecting semantics $\{S'_q : q \in Q\}$ of the polynomial automaton $\mathcal{A}'$, then we can compute the dimension of sets $\overline{S'_q}$, for each $q \in Q$, and hence determine boundedness of the reset VASS $\mathcal{A}$ (which, recall, is an undecidable problem). □

8 CONCLUSION

The main technical contribution of this article is a procedure to compute the Zariski closure of the semigroup generated by a given finite set of rational square matrices of the same dimension. We have not attempted to analyse the complexity of this procedure. However a recent paper [38] gives explicit complexity bounds for the problem of computing the Zariski closure of a finitely generated group of invertible matrices, which is an important component of our algorithm for semigroups. It may be that the techniques developed in this article can be used to obtain explicit bounds on the degree of the generators of an ideal representing the Zariski closure of a given finitely generated matrix semigroup. If this were the case, then one could compute a set of generators essentially using only linear algebra (in the spirit of the algorithm of Reference [37] for computing algebraic invariants of a given maximum degree for a given affine program).

REFERENCES

[1] S. Almagor, D. Chistikov, J. Ouaknine, and J. Worrell. 2018. O-minimal invariants for linear loops. In *Proceedings of the 45th International Colloquium on Automata, Languages and Programming*. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

[2] L. Babai, R. Beals, J.-Y. Cai, G. Ivanyos, and E. M. Luks. 1996. Multiplicative equations over commuting matrices. In *Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms.* 498–507.

[3] T. Becker and V. Weispfenning. 1993. *Gröbner Bases: A Computational Approach to Commutative Algebra (Graduate Texts in Mathematics,* Vol. 141). Springer-Verlag, New York.

[4] V. D. Blondel, E. Jeandel, P. Koiran, and N. Portier. 2005. Decidable and undecidable problems about quantum automata. *SIAM J. Comput.* 34, 6 (2005), 1464–1473.

[5] A. R. Bradley and Z. Manna. 2007. *The Calculus of Computation—Decision Procedures with Applications to Verification.* Springer.

[6] D. Cachera, T. P. Jensen, A. Jobin, and F. Kirchner. 2014. Inference of polynomial invariants for imperative programs: A farewell to Gröbner bases. *Sci. Comput. Program.* 93 (2014), 89–109.

[7] Henri Cohen. 1993. *A Course in Computational Algebraic Number Theory.* Springer-Verlag.

[8] M. Colón. 2007. Polynomial approximations of the relational semantics of imperative programs. *Sci. Comput. Program.* 64, 1 (2007), 76–96.

[9] P. Cousot and R. Cousot. 1977. Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *Proceedings of the 4th ACM Symposium on Principles of Programming Languages.* 238–252.

[10] P. Cousot and N. Halbwachs. 1978. Automatic discovery of linear restraints among variables of a program. In *Proceedings of the 5th Annual ACM Symposium on Principles of Programming Languages.* 84–96.

[11] D. A. Cox, J. B. Little, and D. O'Shea. 1997. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra* (2nd ed.). Springer-Verlag.

[12] S. de Oliveira, S. Bensalem, and V. Prevosto. 2016. Polynomial invariants by linear algebra. In *Proceedings of the 14th International Symposium on Automated Technology for Verification and Analysis*, Vol. 9938. 479–494.

[13] H. Derksen, E. Jeandel, and P. Koiran. 2005. Quantum automata and algebraic groups. *J. Symb. Comput.* 39, 3-4 (2005), 357–371.

[14] C. Dufourd, A. Finkel, and Ph. Schnoebelen. 1998. Reset nets between decidability and undecidability. In *Proceedings of the 25th International Colloquium on Automata, Languages and Programming (Lecture Notes in Computer Science*, Vol. 1443). Springer, 103–115.

[15] Nathanaël Fijalkow, Engel Lefaucheux, Pierre Ohlmann, Joël Ouaknine, Amaury Pouly, and James Worrell. 2019. On the Monniaux problem in abstract interpretation. In *Proceedings of the 26th International Symposium on Static Analysis*.

[16] N. Fijalkow, P. Ohlmann, J. Ouaknine, A. Pouly, and J. Worrell. 2017. Semialgebraic invariant synthesis for the Kannan-Lipton Orbit problem. In *Proceedings of the 34th Symposium on Theoretical Aspects of Computer Science.* 29:1–29:13.

[17] Nathanaël Fijalkow, Pierre Ohlmann, Joël Ouaknine, Amaury Pouly, and James Worrell. 2019. Complete semialgebraic invariant synthesis for the Kannan-Lipton orbit problem. *Theor. Comput. Syst.* 63, 5 (2019), 1027–1048.

[18] S. Gulwani and G. C. Necula. 2003. Discovering affine equalities using random interpretation. In *Proceedings of the 30th SIGPLAN-SIGACT Symposium on Principles of Programming Languages.* ACM, 74–84.

[19] E. Hrushovski. 2002. Computing the Galois group of a linear differential equation. In *Banach Center Publications (Differential Galois Theory)*, Vol. 58. Institute of Mathematics, Polish Academy of Sciences.

[20] A. Humenberger, M. Jaroschek, and L. Kovács. 2018. Invariant generation for multi-path loops with polynomial assignments. In *Proceedings of the 19th International Conference on Verification, Model Checking, and Abstract Interpretation (Lecture Notes in Computer Science*, Vol. 10747). Springer, 226–246.

[21] G. Jacob. 1977. Un algorithme calculant le cardinal, fini ou infini, des demi-groupes de matrices. *Theor. Comput. Sci.* 5, 2 (1977), 183–204.

[22] G. Jacob. 1978. La finitude des représentations linéaires des semi-groupes est décidable. *J. Algeb.* 52, 2 (1978), 437–459.

[23] D. Kapur. 2013. Elimination techniques for program analysis. In *Programming Logics - Essays in Memory of Harald Ganzinger (Lecture Notes in Computer Science*, Vol. 7797). 194–215.

[24] M. Karr. 1976. Affine relationships among variables of a program. *Acta Inf.* 6 (1976), 133–151.

[25] Z. Kincaid, J. Cyphert, J. Breck, and T. W. Reps. 2018. Non-linear reasoning for invariant synthesis. *PACMPL* 2, POPL (2018), 54:1–54:33.

[26] P. Koiran. 2000. The complexity of local dimensions for constructible sets. *J. Complex.* 16, 1 (2000), 311–323.

[27] L. Kovács. 2008. Reasoning algebraically about P-Solvable loops. In *Proceedings of the 14th International Conference on Tools and Algorithms for the Construction and Analysis of Systems, Held as Part of the Joint European Conferences on Theory and Practice of Software (Lecture Notes in Computer Science*, Vol. 4963). Springer, 249–264.

[28] L. Kovacs. 2018. personal communication.

[29] L. Ildikó Kovács and T. Jebelean. 2005. An algorithm for automated generation of invariants for loops with conditionals. In *Proceedings of the 7th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'05).* IEEE Computer Society, 245–249.

[30] A. Mandel and I. Simon. 1977. On finite semigroups of matrices. *Theor. Comput. Sci.* 5, 2 (1977), 101–111.

[31] A. Markov. 1947. On certain insoluble problems concerning matrices. *Doklady Akad. Nauk SSSR* 57, 6 (1947), 539–542.

[32] D. W. Masser. 1988. Linear relations on algebraic groups. In *New Advances in Transcendence Theory.* Cambridge University Press.

[33] A. Miné. 2001. The octagon abstract domain. In *Proceedings of the 8th Working Conference on Reverse Engineering*.

[34] David Monniaux. 2019. On the decidability of the existence of polyhedral invariants in transition systems. *Acta Inf.* 56, 4 (2019), 385–389.

[35] D. W. Morris. 2001. Introduction to Arithmetic Groups. arXiv:math/0106063.

[36] Markus Müller-Olm and Helmut Seidl. 2004. Computing polynomial program invariants. *Inf. Process. Lett.* 91, 5 (2004), 233–244.

[37] M. Müller-Olm and H. Seidl. 2004. A note on Karr's algorithm. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (Lecture Notes in Computer Science*, Vol. 3142). Springer, 1016–1028.

[38] Klara Nosan, Amaury Pouly, Sylvain Schmitz, Mahsa Shirmohammadi, and James Worrell. 2022. On the computation of the Zariski closure of finitely generated groups of matrices. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation.* ACM, 129–138.

[39] Jan Okniński. 1998. *Semigroups of Matrices.* World Scientific.

[40] M. Paterson. 1970. Unsolvability in 3 × 3 matrices. *Stud. Appl. Math.* 49, 1 (1970), 105–107.

[41] I. Potapov and P. Semukhin. 2017. Decidability of the membership problem for 2 × 2 integer matrices. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms.* 170–186.

[42] E. Rodríguez-Carbonell and D. Kapur. 2007. Automatic generation of polynomial invariants of bounded degree using abstract interpretation. *Sci. Comput. Program.* 64, 1 (2007), 54–75.

[43] E. Rodríguez-Carbonell and D. Kapur. 2007. Generating all polynomial invariants in simple loops. *J. Symb. Comput.* 42, 4 (2007), 443–476.

[44] S. Sankaranarayanan, H. Sipma, and Z. Manna. 2004. Non-linear loop invariant generation using Gröbner bases. In *Proceedings of the 31st ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*. ACM, 318–329.

[45] P. Schauenburg. 2007. A Gröbner-based treatment of elimination theory for affine varieties. *J. Symb. Comput.* 9 (2007), 859–870.

[46] B. Steinberg. 2016. *Representation Theory of Finite Monoids*. Springer.

[47] J. Stillwell. 2016. *Elements of Mathematics: From Euclid to Gödel*. Princeton University Press.

Received 8 April 2023; revised 11 July 2023; accepted 17 July 2023