Search Problems in Algebraic Complexity, GCT, and Hardness of Generators for Invariant Rings

Ankit Garg
Microsoft Research, Bangalore, India
garga@microsoft.com

Christian Ikenmeyer
University of Liverpool, UK
christian.ikenmeyer@liverpool.ac.uk

Visu Makam
Institute for Advanced Study, Princeton, NJ, USA
visu@ias.edu

Rafael Oliveira
University of Waterloo, Canada
rafael@uwaterloo.ca

Michael Walter
Korteweg-de Vries Institute for Mathematics, Institute for Theoretical Physics,
Institute for Logic, Language & Computation,
University of Amsterdam, The Netherlands
m.walter@uva.nl

Avi Wigderson
Institute for Advanced Study, Princeton, NJ, US
avi@ias.edu

Abstract

We consider the problem of computing succinct encodings of lists of generators for invariant rings for group actions. Mulmuley conjectured that there are always polynomial sized such encodings for invariant rings of $\mathrm{SL}_n(\mathbb{C})$-representations. We provide simple examples that disprove this conjecture (under standard complexity assumptions).

We develop a general framework, denoted *algebraic circuit search problems*, that captures many important problems in algebraic complexity and computational invariant theory. This framework encompasses various proof systems in proof complexity and some of the central problems in invariant theory as exposed by the Geometric Complexity Theory (GCT) program, including the aforementioned problem of computing succinct encodings for generators for invariant rings.

2012 ACM Subject Classification Theory of computation → Algebraic complexity theory

Keywords and phrases generators for invariant rings, succinct encodings

Digital Object Identifier 10.4230/LIPIcs.CCC.2020.12

Related Version `https://arxiv.org/abs/1910.01251v1`

Funding *Christian Ikenmeyer*: DFG grant IK 116/2-1.
Visu Makam: NSF grant No. DMS -1638352 and NSF grant No. CCF-1412958.
Michael Walter: NWO Veni grant 680-47-459.
Avi Wigderson: NSF grant No. CCF-1412958 and NSF grant CCF-1900460.

1 Introduction

In complexity theory, one often encounters problems that ask for an efficiently computable collection of functions/polynomials satisfying a certain property. Once we are faced with such problems, two natural questions are: how do we represent the property? And how do

35th Computational Complexity Conference (CCC 2020).
Editor: Shubhangi Saraf; Article No. 12; pp. 12:1–12:17

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

we encode the required functions? The answer to these will depend on the context and use. We will first define the informal notion of an algebraic circuit search problem, and then give illustrative examples.

▶ **Definition 1** (Algebraic circuit search problems). *Given an input (of size n), construct an algebraic circuit in a complexity class $\mathcal{C}$ with $k(n)$-inputs and $m(n)$-outputs such that the polynomials they compute satisfy a desirable property $\mathcal{P}$.*

Let us illustrate this definition in the context of algebraic proof complexity: in Nullstellensatz-based proof systems, one is given a set of multivariate polynomials $g_1, \ldots, g_r$ over an algebraically closed field $\mathbb{F}$ and in variables $\mathbf{x} = (x_1, \ldots, x_n)$, and one wants to decide whether the system $g_1(\mathbf{x}) = g_2(\mathbf{x}) = \ldots = g_r(\mathbf{x}) = 0$ has a solution over $\mathbb{F}$. A fundamental result of Hilbert tells us that the system has no solution if and only if there is a set of polynomials $\{f_i\}_{i=1}^r$ such that $\sum_i f_i g_i = 1$. This brings us to the Ideal Proof System [22]:

- **Ideal Proof System (IPS)**: Given a collection of polynomials $g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_r(\mathbf{x})$, we ask to construct a polynomial sized circuit C with $n+r$ inputs. The desirable property $\mathcal{P}$ is that $C(x_1, \ldots, x_n, g_1(\mathbf{x}), \ldots, g_r(\mathbf{x})) = 1$ and that $C(x_1, \ldots, x_n, 0, \ldots, 0) = 0$. It is not so hard to see these conditions will give us a linear combination of the form $\sum_i f_i g_i = 1$ as required.

In [22] the authors show that super-polynomial lower bounds in this proof system imply algebraic circuit lower bounds (i.e., VP $\neq$ VNP), which remains a long standing open problem in complexity theory. Another important point to make is that an instance of 3-SAT, say ϕ, can be encoded as a collection of polynomials $\{g_i\}$ such that ϕ is satisfiable if and only if the $\{g_i\}$ have a common solution. In other words, ϕ is unsatisfiable if and only if $\exists$ polynomials f_i such that $\sum_i f_i g_i = 1$. This converts a co-NP complete problem (unsatisfiability of 3-SAT, called co-3-SAT) into an algebraic circuit search problem of the IPS form described above. The existence of a polynomial sized circuit as demanded by the IPS proof system would mean the existence of f_i with small circuits. But that would mean that co-3-SAT is in NP, thus proving NP = co-NP.

Other important examples of algebraic search problems in proof complexity (with different desirable properties) are the original Nullstellensatz proof system, Polynomial Calculus, and the Positivstellensatz[1] for sum of squares (SOS) proofs. For more on these systems we refer the reader to [29, Chapter 6].

▶ Remark 2. An analogous notion of a "boolean circuit search problem" can also be introduced in the boolean setting. Also here, important problems such as the construction of pseudorandom generators and the construction of extractors can be captured as boolean circuit search problems.

1.1 Geometric Complexity Theory

The GCT program was proposed by Mulmuley and Sohoni (see [32, 33]) as an approach (via representation theory and algebraic geometry) to the VP vs. VNP problem. While there have been some negative results[2] in recent years regarding the techniques one can use towards this program, these results do not disrupt the core framework of the GCT program. Instead, these results indicate the difficulty of the problem from the viewpoint of algebraic combinatorics,

[1] In this case our field is the real numbers, which is *not* algebraically closed.
[2] The results of Bürgisser, Ikenmeyer and Panova, which show that occurrence obstructions cannot give a super-polynomial lower bound on the determinantal complexity of the permanent polynomial (see [9]).

and have identified new directions of research in asymptotic algebraic combinatorics. In [31], Mulmuley views the VP vs. VNP problem through the lens of computational invariant theory, and identifies important and interesting problems in computational invariant theory that form a path towards resolving the VP vs. VNP problem. These include several conjectures, some of which fit into the framework of algebraic circuit search problems, and have important connections and consequences to problems in optimization, algebraic complexity, non-commutative computation, functional analysis and quantum information theory (see [19, 20, 6]). We therefore believe that a better understanding of algebraic circuit search problems will likely result in fundamental advances in the aforementioned areas. Some evidence for these conjectures has emerged over the past few years as they have been established for special cases (see, for example, [31, 19, 27, 17, 12, 13, 11]).

Let us briefly mention an important algebraic circuit search problem and one that will be central to this paper: given a group action, describe a set of generators for the invariant ring (we will elaborate on invariant theory in a subsequent section). Unfortunately, the number of generators for an invariant ring is usually exponential (in the input size of the description of the action). So, in order to get a computational handle on them, Mulmuley suggests in [31] that we should look for a *succinct encoding* (defined below in Definition 3) using some auxiliary variables. One amazing feature of such a succinct encoding is that it would immediately give efficient randomized algorithms for null cone membership and the orbit closure intersection problems which can then be derandomized in some cases (see, e.g., [17, 27, 13]). We will define these problems in a subsequent section, but here we are content to say that many important algorithmic problems such as graph isomorphism, bipartite matching, (non-commutative) rational identity testing, tensor scaling and a form of quantum entanglement distillation are all specific instances (or arise in the study) of null cone membership and orbit closure intersection problems.

Mulmuley conjectures ([31, Conjecture 5.3][3]) the existence of polynomial sized succinct encodings for generators of invariant rings. The main goal of this paper is to (conditionally) disprove this conjecture. More precisely we give an example of an invariant ring (for a torus action) where the existence of such a circuit would imply a polynomial time algorithm for the 3D-matching problem, which is well known to be NP-hard. We also give another example (where the group is $\mathrm{SL}_n(\mathbb{C})$) where the existence of such a circuit would imply VP $\neq$ VNP. Further, the nature of the latter example makes it clear that no simple modification of this conjecture can hold.

The rest of this section will proceed as follows. We first give a brief introduction to invariant theory. Then, we discuss the algebraic search problems of interest in computational invariant theory, followed by the precise statements of our main results. Finally, we discuss some open problems and future directions.

1.2 Invariant Theory

Invariant theory is the study of symmetries, captured by group actions on vector spaces (more generally, algebraic varieties), by focusing on the functions (usually, polynomials) that are left *invariant* under these actions. It is a rich mathematical field in which computational methods are sought and well developed (see [10, 39]). While significant advances have been made on computational problems involving invariant theory, most algorithms are based on Gröbner bases techniques, and hence still require exponential time (or longer).

[3] In the conjecture, the group is specified to be $\mathrm{SL}_n(\mathbb{C})$, which was done for the purpose of accessibility and brevity, but it is natural to ask this problem for general connected reductive groups. We will discuss this further in a later section.

The basic setting is that of a continuous group[4] G acting (linearly) on a finite-dimensional vector space $V = \mathbb{C}^m$.

An *action* (also called a *representation*) of a group $G \subseteq \mathrm{GL}_n(\mathbb{C})$ on an m-dimensional complex vector space V is a group homomorphism $\pi\colon G \to \mathrm{GL}_m(\mathbb{C})$, that is, an association of an invertible $m \times m$ matrix $\pi(g)$ for every group element $g \in G$, satisfying $\pi(g_1 g_2) = \pi(g_1)\pi(g_2)$ for all $g_1, g_2 \in G$ (and $\pi(e) = \mathrm{I}_m$, where $e \in G$ is the identity element and I_m is the identity matrix). To be precise, a group element $g \in G$ acts on a vector $v \in V$ by the linear transformation $\pi(g)$, and in this paper we will be dealing with algebraic actions, that is, the entries of the matrix $\pi(g)$ will be rational functions in the entries of the matrix g. We will write $g \cdot v = \pi(g)v$. Invariant theory is nicest when the underlying field is $\mathbb{C}$ and the group G is either finite, the general linear group $\mathrm{GL}_n(\mathbb{C})$, the special linear group $\mathrm{SL}_n(\mathbb{C})$, or a direct product of these groups and their diagonal subgroups. We denote by $\mathbb{C}[V]$ the ring of polynomial functions on V.

Invariant Polynomials. Invariant polynomials are precisely those which cannot distinguish between a vector v and a translate of it by an element of the group, i.e., $g \cdot v$. In other words, a polynomial function $f \in \mathbb{C}[V]$ is called invariant if $f(g \cdot v) = f(v)$ for all $v \in V$ and $g \in G$. Equivalently, invariant polynomials are polynomial functions on V which are left invariant by the action of G. More precisely, the action of G on V gives an induced action of G on $\mathbb{C}[V]$, the space of polynomial functions on V. For a polynomial function p on V, the group element $g \in G$ sends it to the function $g \cdot p$ which is defined by the formula $(g \cdot p)(v) = p(g^{-1} \cdot v)$ for $v \in V$. Then, a polynomial function is invariant if and only if $g \cdot p = p$ for all $g \in G$. A set $\{f_i\}_{i \in I}$ of invariant polynomials is called a *generating set* if any other invariant polynomial can be written as a polynomial in the f_i's. Two simple and illustrative examples are

- The symmetric group $G = \mathcal{S}_n$ acts on $V = \mathbb{C}^n$ by permuting the coordinates. In this case, the invariant polynomials are *symmetric* polynomials, and the n elementary symmetric polynomials form a generating set (a result that dates back to Newton).
- The group $G = \mathrm{SL}_n(\mathbb{C}) \times \mathrm{SL}_n(\mathbb{C})$ acts on $V = \mathrm{M}_n(\mathbb{C})$ by a change of bases of the rows and columns, namely left-right multiplication: that is, the action of (A, B) sends X to AXB^T. Here, $\det(X)$ is an invariant polynomial and in fact every invariant polynomial must be a univariate polynomial in $\det(X)$. In other words, $\det(X)$ generates the invariant ring.

The above phenomenon that the ring of invariant of polynomials (denoted by $\mathbb{C}[V]^G$) is generated by a finite number of invariant polynomials is not a coincidence. The *finite generation theorem* due to Hilbert [24, 25] states that, for a large class of groups (including the groups mentioned above), the invariant ring must be finitely generated. These two papers of Hilbert are highly influential and laid the foundations of modern commutative algebra and algebraic geometry. In particular, "finite basis theorem" and "Nullstellansatz" were proved as "lemmas" on the way towards proving the finite generation theorem!

Orbits and Orbit Closures. The *orbit* of a vector $v \in V$, denoted by $\mathcal{O}_v$, is the set of all vectors obtained by the action of G on v. The *orbit closure* of v, denoted by $\overline{\mathcal{O}}_v$, is the closure of the orbit $\mathcal{O}_v$ in the Euclidean topology.[5] For actions of continuous groups, such

[4] In general, the theory works whenever the group is algebraic and reductive. However in this paper, we will deal with groups that are well understood such as a torus and the special linear group.

[5] It turns out mathematically more natural to look at closure under the Zariski topology. However, for the group actions we study, the Euclidean and Zariski closures match, a consequence of Chevalley's theorem on constructible sets.

as $\mathrm{GL}_n(\mathbb{C})$, it is more natural to look at orbit closures. The *null cone* for a group action is the set of all vectors which behave like the 0 vector i.e. the 0 vector lies in their orbit closure. Many fundamental problems in theoretical computer science (and many more across mathematics) can be phrased as questions about orbits and orbit closures. Here are some familiar examples:

- Graph isomorphism problem can be phrased as checking if the orbits of two graphs are the same or not, under the action of the symmetric group permuting the vertices.
- Geometric complexity theory (GCT) [32] formulates a variant of the VP vs. VNP question as checking if the (padded) permanent lies in the orbit closure of the determinant (of an appropriate size), under the action of the general linear group on polynomials induced by its natural linear action on the variables.
- Border rank (a variant of tensor rank) of a 3-tensor can be formulated as the minimum dimension such that the (padded) tensor lies in the orbit closure of the unit tensor, under the natural action of $\mathrm{GL}_r(\mathbb{C}) \times \mathrm{GL}_r(\mathbb{C}) \times \mathrm{GL}_r(\mathbb{C})$. In particular, this captures the complexity of matrix multiplication.

1.3 Computational invariant theory, Mulmuley's problems and conjectures

From its origins in the 19th century, the subject of classical invariant theory has been computational in nature – one of its central goals is explicit descriptions of generators of invariant rings, their relations, etc. With the more recent advent of the theory of computation, it is only natural to ask for the complexity of these descriptions. The influence of complexity theory has taken an important role in invariant theory as a consequence of the connections to fundamental problems such as VP vs. VNP that were uncovered as part of the GCT program by Mulmuley in [31]. In [31], Mulmuley considers the computational complexity of various invariant theoretic problems. Let G be a group acting on V.

1. (**Generators**) Output a list of polynomials that generate the invariant ring $\mathbb{C}[V]^G$.
2. (**NNL**) Output a list of polynomials $f_1, \ldots, f_r$, such that each f_i is a homogeneous polynomial and the invariant ring $\mathbb{C}[V]^G$ is integral over $\mathbb{C}[f_1, \ldots, f_r]$.[6]
3. (**Orbit closure intersection**) Given two elements of the vector space, do their orbit closures intersect?
4. (**Null cone membership**) Given an element of the vector space, does the 0 vector lie in its orbit closure?

Let us point out straight away that Generators and NNL (Noether Normalization Lemma) are both algebraic circuit search problems (we will define Generators as an algebraic circuit search problem more precisely below). Orbit closure intersection and Null cone membership are not algebraic circuit search problems, but are related to Generators and NNL in a way that will become clear in a later discussion. We will not get into the details of how the group is given and how the group action is described. It turns out that even for simple groups and group actions, these problems turn out to be interesting. They have been long studied and many algorithms have been developed in the invariant theory community [10, 39]. Mulmuley [31] introduced these problems to theoretical computer science with the hope of making progress on the polynomial identity testing (PIT) problem. Before describing the main conjectures in Mulmuley's paper, let us see what it even means to output a list of generating

[6] This is equivalent to the condition that the zero locus of $f_1, \ldots, f_r$ is precisely the null cone.

polynomials for an invariant ring. Typically the number of generating polynomials can be exponential in the dimension of the group and the vector space. To get around this issue, Mulmuley introduced the following notion of a *succinct encoding* of the generators of an invariant ring (which in fact applies to any collection of polynomials).

▶ **Definition 3** (Succinct encoding of generators). *Fix an action of a group G on a vector space $V = \mathbb{C}^m$. We say that an arithmetic circuit $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$* succinctly encodes *the generators of the invariant ring if the set of polynomials formed by evaluating the y-variables, $\{\mathcal{C}(x_1, \ldots, x_m, \alpha_1, \ldots, \alpha_r)\}_{\alpha_1,\ldots,\alpha_r \in \mathbb{C}}$, is a generating set for the invariant ring $\mathbb{C}[V]^G$.*

▶ Remark 4. The *size* of a succinct encoding as defined above is given by the size of the circuit $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$, which is measured by the bit complexity of the constants used in the computation of $\mathcal{C}$ as well as the number of gates of the computation graph of $\mathcal{C}$. In particular, this means that all constants used in the computation of $\mathcal{C}$ are rationals.

The above notion of a succinct encoding motivates us to define the following algebraic search problem.

▶ **Problem 5** (Generators). *Let G be a group of dimension n and that acts algebraically on an m-dimensional vector space V by linear transformations. Output a* $\mathrm{poly}(n, m)$ *sized circuit $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$ such that the polynomials $\{\mathcal{C}(x_1, \ldots, x_m, \alpha_1, \ldots, \alpha_r)\}_{\alpha_1,\ldots,\alpha_r \in \mathbb{C}}$ form a generating set for the invariant ring $\mathbb{C}[V]^G$. In other words, the problem asks to output a* $\mathrm{poly}(n, m)$ *sized succinct encoding for the generators of $\mathbb{C}[V]^G$.*

▶ **Conjecture 6** (Mulmuley). *In the case that G is a connected reductive algebraic group*[7]*, Problem 5 has a positive answer. That is, there exists a* $\mathrm{poly}(n, m)$ *sized circuit which succinctly encodes the generators of $\mathbb{C}[V]^G$.*

Mulmuley requires the circuit family (that succinctly encodes the generators) to be uniformly computable by a polynomial time algorithm, but we will see that even this weaker conjecture is false (under standard complexity assumptions).

In [31, Conjecture 5.3], Mulmuley states the above conjecture for actions of the group $\mathrm{SL}_n(\mathbb{C})$. However, it is evident that there is nothing special about $\mathrm{SL}_n(\mathbb{C})$ with regard to the GCT program and it is natural to state the conjecture in the generality of connected reductive groups. Let us also note that it was already evident to Mulmuley that one cannot drop the "connected" assumption on the group, because the permanent appears as an invariant polynomial for a non-connected reductive group that would disprove the conjecture immediately using a similar line of reasoning to the one we use in the next section (see, e.g., [4]).

To understand Mulmuley's motivation for the conjecture, let us see what it means for the problems of orbit closure intersection and null cone membership. By definition, invariant polynomials are constant on the orbits (and thus on orbit closures as well). Thus, if $\overline{\mathcal{O}}_{v_1} \cap \overline{\mathcal{O}}_{v_2} \neq \emptyset$, then $p(v_1) = p(v_2)$ for all invariant polynomials $p \in \mathbb{C}[V]^G$. A remarkable theorem due to Mumford says that the converse is also true for the large class of reductive groups:

▶ **Theorem 7** ([34]). *Fix an action of a reductive group G on a vector space V. Given two vectors $v_1, v_2 \in V$, we have $\overline{\mathcal{O}}_{v_1} \cap \overline{\mathcal{O}}_{v_2} \neq \emptyset$ if and only if $p(v_1) = p(v_2)$ for all $p \in \mathbb{C}[V]^G$.*

[7] We have not defined what a connected reductive algebraic group is. One should think of simple groups like the general linear group $\mathrm{GL}_n(\mathbb{C})$, the special linear group $\mathrm{SL}_n(\mathbb{C})$, or a direct product of these groups and their diagonal subgroups.

Now suppose one had a succinct encoding $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$ for action of a group G on $V = \mathbb{C}^m$. Then because of Mumford's theorem, for two vectors v_1 and v_2, their orbit closures intersect iff the two polynomials $\mathcal{C}(v_1(1), \ldots, v_1(m), y_1, \ldots, y_r)$, $\mathcal{C}(v_2(1), \ldots, v_2(m), y_1, \ldots, y_r)$ are identically the same. These are instances of polynomial identity testing (PIT)! Thus if Conjecture 6 were true (and additionally the succinct encoding circuits were polynomial time computable), it immediately gives randomized polynomial time algorithms for the orbit closure intersection and null cone membership problems. This also gives us a nice family of PIT problems to play with. Perhaps one might hope that solving these PIT instances will result in development of new techniques which might shed a light on the general PIT problem. In fact, for the first few group actions that were studied in this line of work, *simultaneous conjugation* [31, 17] and *left-right action* [19, 27, 12], for which there are polynomial sized succinct encodings of generators, the null cone membership problems correspond to PIT problems for restricted models of computation: *read-once algebraic branching programs* and *non-commutative formulas with division*[8], both of which have been successfully derandomized, see [17, 19, 27].

1.4 Our results

While the truth of Conjecture 6 would have great implications, we prove that it is false under plausible complexity hypotheses. We first state our counterexamples (they are very simple, and probably many others exist), and then discuss how a related conjecture may be true and almost as powerful as the original.

For our first counterexample, we analyze a simple (torus) action on 3-tensors. Here, $\mathrm{ST}_n(\mathbb{C})$ denotes the group of $n \times n$ diagonal matrices with determinant 1.

▶ **Theorem 8.** *Consider the natural action of* $G = \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C})$ *on* $V = \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$. *Then any set of generators for the invariant ring cannot have a polynomial sized (in* n*) succinct encoding, unless* $\mathrm{NP} \subseteq \mathrm{P/poly}$.

▶ **Corollary 9.** *Conjecture 6 is false, unless* $\mathrm{NP} \subseteq \mathrm{P/poly}$.

▶ Remark 10. As mentioned previously, a primary motivation for succinct encodings of generators is that they imply (randomized) polynomial time algorithms for null cone membership problem. For the action in Theorem 8, it is important to note that even though we do not have a succinct encoding for generators, we still have a polynomial time algorithm for null cone membership since once can reduce it to an instance of linear programming. For a general connection between null cone membership and optimization, see [5].

For the above counterexample for the torus action, the notion of a succinct encoding is quite crucial to our argument, and it is natural to wonder if tweaking the notion would get rid of the issue. We give another counterexample where it becomes apparent that the precise form of encoding of the generators is not quite as crucial, as we identify an invariant that is hard to compute and is *essential* to any generating set in a sense that we will make precise in Section 4. Moreover, it is an $\mathrm{SL}_n(\mathbb{C})$-action, which provides a counterexample to the exact formulation of the conjecture in [31].

▶ **Theorem 11.** *Let* $k \geq 2$ *be even. Consider the action of* $G = \mathrm{SL}_{2kn}$ *on* $V = \bigotimes^{2k} \mathbb{C}^{2kn}$. *Then any set of generators for the invariant ring cannot have a polynomial sized (in* n*) succinct encoding, unless* $\mathrm{VP} = \mathrm{VNP}$.

▶ **Corollary 12.** *Conjecture 6 is false, unless* $\mathrm{VP} = \mathrm{VNP}$.

[8] Actually a stronger model concerning inverses of matrices.

1.5 Conclusion, open problems and future directions

We have disproved a conjecture of Mulmuley about the existence of polynomial sized succinct encodings of generators for invariant rings. We want to emphasize that this only serves a first guiding light for Mulmuley's program of understanding the orbit closure intersection problems (and null cone membership problems) and connections to PIT. To solve the orbit closure intersection problems, one does not necessarily need a generating set of generators. This motivates the following definition.

▶ **Definition 13** (Separating set of invariants). *For a group G acting algebraically on a vector space V by linear transformations, a subset $S \subseteq \mathbb{C}[V]^G$ is called a* separating set of invariants *if for all $u, v \in V$ such that $\overline{\mathcal{O}}_u \cap \overline{\mathcal{O}}_v \neq \emptyset$, there exists $f \in S$ such that $f(u) \neq f(v)$.*

This leads to a natural algebraic search problem that corresponds to the algorithmic problem of orbit closure intersection. Mulmuley already suggested that a positive answer to the following search problem would suffice for the purposes of GCT.

▶ **Problem 14** (Separators). *Let G be a group of dimension n and suppose it acts algebraically on an m-dimensional vector space V by linear transformations. Output a $\mathrm{poly}(n, m)$ sized circuit $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$, if one exists, such that the set of polynomials $S = \{\mathcal{C}(x_1, \ldots, x_m, \alpha_1, \ldots, \alpha_r)\}_{\alpha_1,\ldots,\alpha_r \in \mathbb{C}}$ is a separating set of invariants.*

Similarly, we can define a search problem that corresponds to the algorithmic problem of null cone membership.

▶ **Problem 15** (Null cone definers). *Let G be a group of dimension n and suppose G acts algebraically on an m-dimensional vector space V by linear transformations. Output a $\mathrm{poly}(n, m)$ sized circuit $\mathcal{C}(x_1, \ldots, x_m, y_1, \ldots, y_r)$ with the property that the set $S = \{\mathcal{C}(x_1, \ldots, x_m, \alpha_1, \ldots, \alpha_r)\}_{\alpha_1,\ldots,\alpha_r \in \mathbb{C}}$ consists of invariant polynomials whose zero locus is precisely the null cone $\mathcal{N}_G(V) = \{v \in V \mid 0 \in \overline{\mathcal{O}}_v\}$.*

We conclude the introduction with some open open problems:

1. Are there polynomial sized succinct encodings for separating invariants or, even simpler, invariants defining the null cone? In other words, do we have positive answers to Problems 14 and 15 for connected reductive groups? Perhaps the first non-trivial example is the natural action of $G = \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C})$ on $V = \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$. Here a tensor T is in the null cone iff there exists vectors $x, y, z \in \mathbb{R}^n$ s.t. $x_i + y_j + z_k > 0$ for all $(i, j, k) \in \mathrm{supp}(T)$[9] and $\sum_i x_i = \sum_j y_j = \sum_k z_k = 0$ (by the Hilbert-Mumford criterion). Is there a polynomial sized circuit $\mathbb{C}((z_{i,j,k}), y_1, \ldots, y_r)$ s.t. $\mathbb{C}(T, y_1, \ldots, y_r)$ is identically zero (as a polynomial in the y-variables) iff T is in the null cone?
2. For the natural action of $\mathrm{SL}_n(\mathbb{C}) \times \mathrm{SL}_n(\mathbb{C}) \times \mathrm{SL}_n(\mathbb{C})$ on $V = \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$, it is not even clear if there exists one invariant which has a polynomial sized circuit. Either produce such an invariant or prove that all invariants are hard to compute.[10]
3. Are there polynomial time algorithms for the orbit closure intersection and null cone membership problems? The analytic approach pursued in the papers [19, 7, 2, 5] seems the most promising approach towards getting such algorithms.

[9] $\mathrm{supp}(T) = \{(i, j, k) \in [n] \times [n] \times [n] : T_{i,j,k} \neq 0\}$.

[10] This problem is known to experts in the field, but has not been written down explicitly anywhere to the best of our knowledge.

4. More broadly, invariant theory is begging for its own complexity theory and connecting it with ours. This includes finding reductions and completeness results, and characterizations/dichotomies about hard/easy actions. An example of a completeness reduction is the reduction from all quiver actions to the simple left-right action [15, 16, 35, 12, 20]. Also the papers [31, 19, 27, 17, 12, 13, 11, 23, 30, 5], as well as the current paper, are trying to identify easy and hard problems in invariant theory.

2 Preliminaries

In this section we establish notation and we formally state basic facts and definitions which we will need in later sections.

▶ **Definition 16** (3-dimensional matching [28]). *The* 3-dimensional matching *problem is defined as follows:*

- ***Input:*** *a set $U \subseteq [n] \times [n] \times [n]$, representing the edges of a tripartite, 3-uniform hypergraph.*
- ***Output:*** *YES, if there is a set of hyperedges $W \subseteq U$ such that $|W| = n$ and no two elements of W agree in any coordinate (that is, they form a matching in this hypergraph). NO, if there is no such set.*

▶ **Theorem 17** (NP-completeness of 3-dimensional matching [28]). *The 3-dimensional matching problem is NP-complete.*

2.1 Basic facts from algebraic complexity

We now give basic facts that from algebraic complexity which we will use in the next sections.

The next proposition shows that homogeneous components of low degree of an arithmetic circuit can be efficiently computed, with a small blow-up in circuit size and without the use of any extra constants. This proposition was originally proved by Strassen in [38] and its proof can be found in [37, Theorem 2.2]. In the following proposition, given a polynomial $p(\mathbf{x})$, we denote its degree-d homogeneous component by $H_d[p(\mathbf{x})]$.

▶ **Proposition 18** (Efficient computation of homogeneous components). *Given a circuit $\mathcal{C}(\boldsymbol{x})$ of size s, then for every $r \in \mathbb{N}$ there is a homogeneous circuit $\Psi(\boldsymbol{x})$ of size $O(r^2 s)$ computing $H_0[\mathcal{C}(\boldsymbol{x})], H_1[\mathcal{C}(\boldsymbol{x})], \ldots, H_r[\mathcal{C}(\boldsymbol{x})]$. Moreover, the constants used in the computation of the components $H_i[\mathcal{C}(\boldsymbol{x})]$ are a subset of the coefficients used in the computation of $\mathcal{C}(\boldsymbol{x})$.*

The next theorem, proved by [1, Theorem 4.10] gives us a randomized polynomial time algorithm to test whether an algebraic circuit of polynomial size, with rational coefficients, is identically zero. Another randomized algorithm easily follows from [36, Lemma 2], when adapted for polynomials with rational coefficients.

▶ **Theorem 19** (PIT for poly-sized circuits [1]). *Let $P(\boldsymbol{x}) \in \mathbb{Q}[\boldsymbol{x}]$ be a polynomial in the variables $\boldsymbol{x} = (x_1, \ldots, x_n)$, with each variable x_i having degree bounded by d_i, and whose coefficients are rationals with bit complexity bounded by B. If $P(\boldsymbol{x})$ is given as an arithmetic circuit of size s, then there exists a randomized algorithm running in time $\mathrm{poly}(n, s, \log(B), 1/\epsilon)$ and using $O\left(\sum_{i=1}^{n} \log(d_i) + \log(B)\right)$ random bits which tests whether $P(\boldsymbol{x})$ is identically zero. If $P(\boldsymbol{x})$ is the identically zero polynomial then the algorithm always succeeds. Otherwise, it errs with probability at most ϵ.*

3 Hardness of Generators for torus actions

Let $\mathbb{C}^*$ denote the multiplicative group consisting of all non-zero complex numbers. A direct product $\mathrm{T}_n = (\mathbb{C}^*)^n$ is called a torus, and is clearly an abelian group. Tori are important examples of reductive groups – any abelian connected reductive group is a torus! It is often the case that it is easier to understand tori in comparison with more general (non-abelian) reductive groups. This is no different for invariant theory, see for example [10, 40]. We also point to [14, Proposition 3.3] for an elementary linear algebraic description of the invariant ring for torus actions. Conjecture 6 already fails in this well behaved setting. This is the content of our Theorem 8, which we will prove in this section. Recall that $\mathrm{ST}_n(\mathbb{C}) \cong \{\mathbf{z} \in \mathrm{T}_n : z_1 \cdots z_n = 1\}$, which is itself a torus.

▶ **Theorem 20** (Theorem 8, restated). *Consider the natural action of* $G = \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C})$ *on* $V = \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$*, where an element* $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \in G$ *acts on a tensor* $u \in V$ *as follows:* $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}) \cdot u := v$*, such that* $v_{ijk} = a_i b_j c_k u_{ijk}$*. Any set of generators for the invariant ring of this action cannot have a polynomial sized (in* n*) succinct encoding, unless* $\mathrm{NP} \subseteq \mathrm{P/poly}$.

Proof. Suppose that the natural action above has a set of generators with a polynomial sized succinct encoding. Thus, there is an arithmetic circuit $\mathcal{C}(\mathbf{x}, \mathbf{y})$ of size $s = \mathrm{poly}(n)$, where $\mathbf{x} = (x_{ijk})_{i,j,k=1}^n$ is the set of variables corresponding to V and $\mathbf{y} = (y_1, \ldots, y_r)$ is the set of auxiliary variables, with $r = \mathrm{poly}(n)$. Moreover, from the definition of the size of a succinct encoding we also have that the constants used in the computation of $\mathcal{C}(\mathbf{x}, \mathbf{y})$ are rational numbers with bit complexity bounded by $b = \mathrm{poly}(n)$. In particular, $\mathcal{C}(\mathbf{x}, \mathbf{y}) \in \mathbb{Q}[\mathbf{x}, \mathbf{y}]$.

Let us consider the circuit $\mathcal{C}(\mathbf{x}, \mathbf{y})$ as a circuit whose constants are in $\mathbb{Q}[\mathbf{y}]$ and whose variables are only the $\mathbf{x}$ variables, that is, a circuit in $\mathbb{Q}[\mathbf{y}][\mathbf{x}]$. Then, Proposition 18 tells us that there exists a homogeneous circuit $\mathcal{C}_n(\mathbf{x}, \mathbf{y})$, in the $\mathbf{x}$ variables, of degree n and size $O(n^2 s)$ that computes the homogeneous component of $\mathcal{C}(\mathbf{x}, \mathbf{y})$ of degree n as a function of $\mathbf{x}$. Moreover, the constants of this circuit are a subset of the constants used in the circuit $\mathcal{C}(\mathbf{x}, \mathbf{y})$. Since we consider the latter as a circuit in only the $\mathbf{x}$ variables, the constants in this case are given by the elements of $\mathbb{Q}$ used in the computation of $\mathcal{C}$ as well as the auxiliary variables $\mathbf{y}$. In particular, $\mathcal{C}_n(\mathbf{x}, \mathbf{y})$ can be written in the following way:

$$\mathcal{C}_n(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{m} \in \mathcal{N}_n(\mathbf{x})} f_{\mathbf{m}}(\mathbf{y}) \cdot \mathbf{m}, \tag{1}$$

where $\mathcal{N}_n(\mathbf{x})$ is the set of all monomials of degree n in the variables $\mathbf{x}$ and $f_{\mathbf{m}}(\mathbf{y})$ are polynomials in the variables $\mathbf{y}$ of degree at most 2^s, as the circuit $\mathcal{C}$ has size at most s.

In Proposition 21 below, we will show that the invariants of minimum degree of our action are in degree n, and these are spanned by the (maximum) 3-dimensional matching monomials. Thus, if a monomial of degree n is invariant under our action, it must be the case that this monomial corresponds to a 3-dimensional matching. Moreover, the action maps any monomial (invariant or not) to a constant times itself. As $\mathcal{C}_n(\mathbf{x}, \mathbf{y})$ must only compute invariant polynomials, this implies that equation (1) is actually of the following form:

$$\mathcal{C}_n(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{m} \in \mathcal{M}_n(\mathbf{x})} f_{\mathbf{m}}(\mathbf{y}) \cdot \mathbf{m}, \tag{2}$$

where $\mathcal{M}_n(\mathbf{x})$ is the set of all 3-dimensional matching monomials in the variables $\mathbf{x}$. Moverover, since $\mathcal{C}(\mathbf{x}, \mathbf{y})$ succinctly encodes of a set of generators, the span of $\{\mathcal{C}_n(\mathbf{x}, \alpha)\}_{\alpha \in \mathbb{C}^r}$ must necessarily be the same as the span of the 3-dimensional matching polynomials.

We will now show that the existence of the circuit $\mathcal{C}_n(\mathbf{x}, \mathbf{y})$ implies that NP $\subseteq$ P/poly. For that purpose, we will show that given $\mathcal{C}_n(\mathbf{x}, \mathbf{y})$ one can solve the 3-dimensional matching problem in P/poly. Let H be a tripartite 3-uniform hypergraph, whose edges are given by a subset $E \subseteq [n] \times [n] \times [n]$. We can associate to this graph the tensor $v \in V$ where $v_{ijk} = 1$ if hyperedge $(i, j, k) \in E$ and $v_{ijk} = 0$ otherwise. Note that H has a 3-dimensional matching of size n if and only if at least one of the 3-dimensional matching monomials *does not vanish* on our tensor v. This last condition is equivalent to the fact that the circuit $\mathcal{C}_n(v, \mathbf{y})$ does not compute the zero polynomial (as we know that the span of the set $\{\mathcal{C}_n(\mathbf{x}, \alpha)\}_{\alpha \in \mathbb{C}^r}$ is the same as the span of all 3-dimensional matching monomials). Thus, to solve the 3-dimensional matching problem in P/poly it is enough to give a randomized polynomial time algorithm for testing whether $\mathcal{C}_n(v, \mathbf{y})$ is the zero polynomial or not.[11]

Since $\mathcal{C}_n(v, \mathbf{y})$ is a circuit of size poly(n) with rational constants of bit complexity poly(n), it computes a polynomial $P(\mathbf{y})$ with rational coefficients having bit complexity at most exp(poly(n)) and degree at most exp(poly(n)). This is the setting in which Theorem 19 applies, giving us the desired randomized polynomial time algorithm. This concludes our proof modulo Proposition 21, which we will now turn our attention to. ◀

In the following proposition, we denote by $\mathcal{S}_n$ the symmetric group on n letters.

▶ **Proposition 21.** *The maximum 3-dimensional matching monomials $\prod_{i=1}^n x_{i\sigma(i)\tau(i)}$, where $\sigma, \tau \in \mathcal{S}_n$, span the invariants of degree n of the natural action of $G = \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C}) \times \mathrm{ST}_n(\mathbb{C})$ on $V = \mathbb{C}^n \otimes \mathbb{C}^n \otimes \mathbb{C}^n$. Moreover, there are no nonconstant invariants of degree less than n for this action.*

Proof. Since the action maps any monomial to a constant times itself, it is easy to see that the invariant polynomials are generated by invariant monomials. To prove the proposition, it is therefore enough to show that the matching monomials are invariant, that there are no other invariant monomials of degree n, and that there are no invariant monomials of smaller degree.

We first prove that the matching monomials are invariant. Note that the natural action of G on V induces the following action on the variables x_{ijk}: $(\mathbf{a}, \mathbf{b}, \mathbf{c}) \cdot x_{ijk} = (a_i b_j c_k)^{-1} \cdot x_{ijk}$.[12] Additionally, note that $\prod_{\ell=1}^n a_\ell = \prod_{\ell=1}^n b_\ell = \prod_{\ell=1}^n c_\ell = 1$. Given a matching monomial $\prod_{i=1}^n x_{i\sigma(i)\tau(i)}$, we therefore have that

$$\begin{aligned}(\mathbf{a}, \mathbf{b}, \mathbf{c}) \cdot \prod_{i=1}^n x_{i\sigma(i)\tau(i)} &= \prod_{i=1}^n \left((a_i b_{\sigma(i)} c_{\tau(i)})^{-1} \cdot x_{i\sigma(i)\tau(i)}\right) \\ &= \prod_{i=1}^n (a_i b_{\sigma(i)} c_{\tau(i)})^{-1} \cdot \prod_{i=1}^n x_{i\sigma(i)\tau(i)} \\ &= \prod_{i=1}^n x_{i\sigma(i)\tau(i)}\end{aligned}$$

where in the last equality we note that for any permutation $\sigma \in \mathcal{S}_n$ (or τ) we have $1 = \prod_{\ell=1}^n a_\ell = \prod_{\ell=1}^n a_{\sigma(\ell)}$ (and similarly for $\mathbf{b}$ and $\mathbf{c}$). This proves that the matching monomials are invariant monomials of the natural G-action on V.

Now, let us prove that no other monomial of degree n is an invariant for this action. Let $\prod_{m=1}^n x_{i_m j_m k_m}$ be a monomial, where $(i_m, j_m, k_m) \in [n]^3$, that is not a matching monomial. Then there exists some coordinate, w.l.o.g. the first coordinate, for which the set $\{i_m\}_{m=1}^n$ is a

[11] It is enough to give a randomized polynomial time algorithm because we know that BPP/poly = P/poly.
[12] The inverse comes from the general formula $(g \cdot p)(v) := p(g^{-1} \cdot v)$.

strict subset of $[n]$. Equivalently, there is an element $\ell \in [n]$ such that $\ell \notin \{i_m\}_{m=1}^n$. W.l.o.g., we can assume that $\ell = 1$. Thus, the action of $\mathbf{a} = (\alpha^{n-1}, \alpha^{-1}, \ldots, \alpha^{-1}), \mathbf{b} = \mathbf{c} = (1, \ldots, 1)$ on our monomial $\prod_{m=1}^n x_{i_m j_m k_m}$ is as follows:

$$(\mathbf{a}, \mathbf{b}, \mathbf{c}) \cdot \prod_{m=1}^{n} x_{i_m j_m k_m} = \alpha^n \cdot \prod_{m=1}^{n} x_{i_m j_m k_m}$$

which proves that this monomial is not an invariant. This completes the proof that the matching monomials span the invariants of degree n.

Now we are left with proving that there are no nonconstant monomials of degree less than n that are invariant. Note that if we have a monomial with degree less than n, we can represent it as $\prod_{m=1}^{d} x_{i_m j_m k_m}$, where $d < n$ and by the pigeonhole principle, we know that there exists $\ell \in [n]$ such that ℓ does not appear as a first coordinate entry in the set of tuples $\{(i_m, j_m, k_m)\}$. If $d > 0$ then, analogously to the previous paragraph, we know that such monomials cannot be invariants of the natural action of G over V, therefore showing that no nonconstant monomial of degree $< n$ can be an invariant. This completes the proof. ◀

4 Invariant Theory for $\mathrm{SL}_n(\mathbb{C})$ and Mulmuley's conjecture

In this section, we will give another example of a group action on tensors for which any set of generating invariants is hard to compute, i.e., we will prove Theorem 11. Even though the previous section already gives a counterexample, this example illustrates something more. The feature of this group action is that invariants of minimial degree span a 1-dimensional space. In other words, up to scaling, we have a unique invariant of minimal degree. This unique invariant in the minimal degree is called the *hyperpfaffian polynomial* (introduced by Barvinok in 1995 as a natural generalization of the well-known Pfaffian polynomial to higher order tensors). We then study the hyperpfaffian's computational complexity and prove that it is VNP-complete. The importance of this example is that such a unique invariant in the minimal degree is *essential* in any generating set.[13] So, it is not even possible to give a generating set consisting of invariant polynomials that are easy to compute, even if we remove all restrictions on the size of the generating set.[14] Moreover, the group action is by $\mathrm{SL}_n(\mathbb{C})$ rather than a torus. Therefore our counterexample disproves Mulmuley's original conjecture in a strong sense.

4.1 Invariant Rings and Symmetric Tensors

The special linear group $\mathrm{SL}_n(\mathbb{C})$ consists of complex $n \times n$-matrices with unit determinant and acts canonically on $\mathbb{C}^n$ by matrix-vector multiplication. This action extends to any m-th tensor power $\bigotimes^m \mathbb{C}^n$ by

$$g \cdot (v_1 \otimes \cdots \otimes v_m) := (g \cdot v_1) \otimes \cdots \otimes (g \cdot v_m) \tag{3}$$

and linear continuation. We will always use the standard bilinear form on $\bigotimes^m \mathbb{C}^n$ that satisfies $\langle g^T \cdot v, w \rangle = \langle v, g \cdot w \rangle$ for all $v, w \in \bigotimes^m \mathbb{C}^n$, $g \in \mathrm{SL}_n(\mathbb{C})$.

[13] Unique invariants in minimal degree have been studied and used in the context of GCT before in [8], although the problems pursued there are quite different from the one we are considering in this paper.

[14] A very similar argument also works in the action of $\mathrm{SL}_n^{\times 4}$ on $(\mathbb{C}^n)^{\otimes 4}$, in which case, the unique minimal invariant is called Pascal determinant and also known to be VNP–hard. This appeared in an earlier version of this paper, see [21]. However, our current example using the hyperpfaffian has the added advantage of disproving the exact formulation of Mulmuley's conjecture.

Let V be an arbitrary finite-dimensional $\mathrm{SL}_n(\mathbb{C})$-representation (such as $V = \bigotimes^m \mathbb{C}^n$). Then $\mathrm{SL}_n(\mathbb{C})$ also acts on $\mathbb{C}[V]_d$, the vector space of degree-d homogeneous polynomials on V, via the formula

$$(g \cdot p)(v) := p(g^T \cdot v),$$

where $p \in \mathbb{C}[V]_d, g \in G$ and $v \in V$. The formula above is the dual representation of the action on the ring of all polynomial functions $\mathbb{C}[V] = \bigoplus_{d=0}^{\infty} \mathbb{C}[V]_d$ that we explained in in Section 1.2. Using the dual here is only for presentation purposes, as it gives a clearer connection to multilinear algebra as follows. Note that a polynomial $p \in \mathbb{C}[V]$ is invariant if and only if $\forall g \in \mathrm{SL}_n(\mathbb{C})$ we have $g \cdot p = p$.

It is convenient to identify polynomial functions with symmetric tensors. Note that $\mathrm{SL}_n(\mathbb{C})$ acts canonically on any d-th tensor power $\bigotimes^d V$ of V. This action restricts to the d-th symmetric tensor power $\mathrm{Sym}^d V$, i.e., the $\mathcal{S}_d$-invariant subspace of $\bigotimes^d V$. Recall that $\mathcal{S}_d$ is the symmetric group on d letters; it acts on $V^{\otimes d}$ by permuting tensor factors. For any $t \in \mathrm{Sym}^d V$, we can define a homogeneous degree-d polynomial $p \in \mathbb{C}[V]_d$ by $p(v) := \langle t, v^{\otimes d} \rangle$. Here we use the quadratic form on $\mathrm{Sym}^d V$ induced by a non-degenerate bilinear form on V that satisfies $\langle g^t \cdot v, w \rangle = \langle v, gw \rangle$ for all $v, w \in V$, $g \in \mathrm{SL}_n(\mathbb{C})$ as above. Then, p is invariant if and only if the symmetric tensor t is invariant, i.e., if $\forall g \in \mathrm{SL}_n(\mathbb{C})$ we have $g \cdot t = t$. We will tacitly go back and forth between symmetric tensors in $\mathrm{Sym}^d \bigotimes^m \mathbb{C}^n$ and homogeneous polynomials in $\mathbb{C}[\bigotimes^m \mathbb{C}^n]_d$.

Now, we turn to studying hyperpfaffians.

4.2 Hyperpfaffians

The *Pfaffian* is the unique (up to scale) homogeneous $\mathrm{SL}_{2n}(\mathbb{C})$-invariant of degree n on $\mathbb{C}^{2n} \otimes \mathbb{C}^{2n}$. There are no $\mathrm{SL}_{2n}(\mathbb{C})$-invariants in lower degrees. If we identify $\mathbb{C}^{2n} \otimes \mathbb{C}^{2n}$ with the space of complex $2n \times 2n$ matrices A, then the Pfaffian is invariant under the action of $\mathrm{SL}_{2n}(\mathbb{C})$ given by $g \cdot A := gAg^T$. The defining property of the Pfaffian generalizes to tensors of even order as follows (the classical Pfaffian is the special case of $k = 1$):

▶ **Proposition 22.** *For any k and n, there is a unique (up to scale) homogeneous $\mathrm{SL}_{2kn}(\mathbb{C})$-invariant polynomial $\mathrm{Pf}_{k,n}$ of degree n on $\bigotimes^{2k} \mathbb{C}^{2kn}$. $\mathrm{Pf}_{k,n}$ identifies with the symmetric tensor $e_1 \wedge \cdots \wedge e_{2kn} \in \mathrm{Sym}^n \bigotimes^{2k} \mathbb{C}^{2kn}$. There are no nonconstant $\mathrm{SL}_{2kn}(\mathbb{C})$-invariants of lower degree.*

Before proving Proposition 22 we recall some representation theory. The material is well-known, and we refer to standard texts (e.g., [18]) for details. A *partition* λ is a nonincreasing sequence of natural numbers with finite support. We write $\lambda \vdash_n m$ to say that $|\lambda| := \sum_i \lambda_i = m$ and $\lambda_{n+1} = 0$. If $\lambda_{n+1} = 0$, then we say that λ is an n-partition. The irreducible polynomial $\mathrm{GL}_n(\mathbb{C})$-representations are indexed by n-partitions. For a partition λ, let $\{\lambda\}$ denote the irreducible $\mathrm{GL}_n(\mathbb{C})$-representation corresponding to λ. Restricted to $\mathrm{SL}_n(\mathbb{C})$, the representation $\{\lambda\}$ is trivial if and only if $\lambda_1 = \ldots = \lambda_n$; note that this implies that $n \mid m$. The irreducible representations of $\mathcal{S}_m$ are indexed by partitions λ with $|\lambda| = m$. Let $[\lambda]$ denote the irreducible $\mathcal{S}_m$-representation corresponding to λ.

Consider $\bigotimes^m \mathbb{C}^n$. This space has an action of $\mathrm{SL}_n(\mathbb{C})$ by (3), but also an action of $\mathcal{S}_m$ that permutes the tensor factors. Both actions commute, so we have an action of the product group $\mathrm{SL}_n(\mathbb{C}) \times \mathcal{S}_m$. The following well-known result will be crucial for our purposes.

▶ **Theorem 23** (Schur–Weyl duality). *As an* $\mathrm{SL}_n(\mathbb{C}) \times \mathcal{S}_m$*-representation, we have the decomposition:*

$$\bigotimes^m \mathbb{C}^n = \bigoplus_{\lambda \vdash_n m} \{\lambda\} \otimes [\lambda].$$

Using Schur–Weyl duality, one sees immediately that $\bigotimes^m \mathbb{C}^n$ contains nonzero $\mathrm{SL}_n(\mathbb{C})$-invariant vectors if and only if $n \mid m$. This is because a vector is invariant if and only if it spans a trivial irreducible representation – but $\{\lambda\}$ is trivial if and only if $\lambda_1 = \ldots = \lambda_n$, as mentioned above. For $m = n$, there is a unique (up to scale) $\mathrm{SL}_n(\mathbb{C})$-invariant vector. This is because the invariants in $\bigotimes^n \mathbb{C}^n$ correspond to the component $\{1^n\} \otimes [1^n]$, where we write 1^n for the partition $\lambda_1 = \ldots = \lambda_n = 1$. Here, $\{1^n\}$ is the trivial representation of $\mathrm{SL}_n(\mathbb{C})$ and $[1^n]$ is one-dimensional, as it is the sign representation of $\mathcal{S}_n$. Thus the space of invariants is one-dimensional. This unique vector (up to scale) is given by the wedge product $e_1 \wedge e_2 \wedge \cdots \wedge e_n$, where $a \wedge b := \frac{1}{2}(a \otimes b - b \otimes a)$, and higher order wedge products are defined analogously.

Proof of Proposition 22. It suffices to show that $\mathrm{Sym}^d \bigotimes^{2k} \mathbb{C}^{2kn}$ contains no $\mathrm{SL}_{2kn}(\mathbb{C})$-invariant vector if $0 < d < n$ and that it contains a unique such vector if $d = n$. Note that $\mathrm{Sym}^d \bigotimes^{2k} \mathbb{C}^{2kn}$ is a subspace of $\bigotimes^d \bigotimes^{2k} \mathbb{C}^{2kn} \simeq \bigotimes^{2kd} \mathbb{C}^{2kn}$. Thus the first claim holds since $\bigotimes^{2kd} \mathbb{C}^{2kn}$ contains $\mathrm{SL}_{2kn}(\mathbb{C})$-invariant vectors only if $2kn \mid 2kd$. Thus if $0 < d < n$, there are no invariants. For $d = n$, $\bigotimes^d \bigotimes^{2k} \mathbb{C}^{2kn} \simeq \bigotimes^{2kn} \mathbb{C}^{2kn}$ contains the unique $\mathrm{SL}_{2kn}(\mathbb{C})$-invariant vector $v = (e_1 \wedge \cdots \wedge e_{2k}) \wedge \cdots \wedge (e_{2k(n-1)+1} \wedge \cdots \wedge e_{2kn})$. It remains to show that v is symmetric, i.e., an element of $\mathrm{Sym}^d \bigotimes^{2k} \mathbb{C}^{2kn}$, which is a subspace of $\bigotimes^d \bigotimes^{2k} \mathbb{C}^{2kn}$. But this is easy to see since each of the d blocks has even size $2k$ and the wedge product is skew-commutative. This proves the second claim. ◀

The polynomial $\mathrm{Pf}_{k,n}$ was introduced in [3, Def. 3.4] in its monomial presentation, where it is called the *hyperpfaffian*. Note that, for fixed k, $\mathrm{Pf}_k := (\mathrm{Pf}_{k,1}, \mathrm{Pf}_{k,2}, \ldots)$ is a p-family (i.e., both the degree and the number of variables are polynomially bounded), since $\mathrm{Pf}_{k,n}$ has degree n and $(2kn)^{2k}$ variables. The monomial presentation in [3] immediately yields that $\mathrm{Pf}_k \in \mathrm{VNP}$.

▶ **Theorem 24.** *For even* k*,* Pf_k *is* VNP*-complete.*

Proof. We present a projection of $\mathrm{Pf}_{k,d}$ to the $d \times d$ permanent. The same projection yields the determinant if k is odd, which explains why the proof does not work for the classical Pfaffian ($k = 1$). The case $k = 2$ is enough to disprove Mulmuley's conjecture.

By Proposition 22, the Pfaffian $\mathrm{Pf}_{k,d}$ identifies with the symmetric tensor

$$v := e_1 \wedge \cdots \wedge e_{2kd} \in \mathrm{Sym}^d \bigotimes^{2k} \mathbb{C}^{2kd}.$$

Thus, the evaluation $\mathrm{Pf}_{k,d}(p)$ at a tensor $p \in \bigotimes^{2k} \mathbb{C}^{2kd}$ is given by $\langle v, p^{\otimes d} \rangle$ (cf. [26, Sec. 4.2(A)]). We choose

$$p = \sum_{i,j=0}^{d-1} x_{i+1,j+1}(e_{1+2ki} \otimes e_{2+2ki} \otimes \cdots \otimes e_{k+2ki} \otimes e_{k+1+2kj} \otimes e_{k+2+2kj} \otimes \cdots \otimes e_{2k+2kj}),$$

where the $x_{i,j}$ $(1 \le i, j \le d)$ are formal variables.

The point p is parametrized linearly by the $x_{i,j}$, so the evaluation of $\mathrm{Pf}_{k,d}$ at p is a projection of $\mathrm{Pf}_{k,d}$. We verify that the evaluation of $\mathrm{Pf}_{k,n}$ at p gives the $d \times d$ permanent (up to a constant nonzero scalar) as follows.

$$p^{\otimes d} = \sum_{i_1,j_1,\ldots,i_d,j_d=0}^{d-1} x_{i_1+1,j_1+1}\cdots x_{i_d+1,j_d+1} \\ (e_{1+2ki_1}\otimes e_{2+2ki_1}\otimes\cdots\otimes e_{k+2ki_1}\otimes e_{k+1+2kj_1}\otimes e_{k+2+2kj_1}\otimes\cdots\otimes e_{2k+2kj_1}) \\ \otimes\cdots\otimes(e_{1+2ki_d}\otimes e_{2+2ki_d}\otimes\cdots\otimes e_{k+2ki_d}\otimes e_{k+1+2kj_d}\otimes e_{k+2+2kj_d}\otimes\cdots\otimes e_{2k+2kj_d})$$

and by linearity

$$\langle v, p^{\otimes d}\rangle = \sum_{i_1,j_1,\ldots,i_d,j_d=0}^{d-1} x_{i_1+1,j_1+1}\cdots x_{i_d+1,j_d+1} \\ \langle v, (e_{1+2ki_1}\otimes e_{2+2ki_1}\otimes\cdots\otimes e_{k+2ki_1}\otimes e_{k+1+2kj_1}\otimes e_{k+2+2kj_1}\otimes\cdots\otimes e_{2k+2kj_1}) \\ \otimes\cdots\otimes(e_{1+2ki_d}\otimes e_{2+2ki_d}\otimes\cdots\otimes e_{k+2ki_d}\otimes e_{k+1+2kj_d}\otimes e_{k+2+2kj_d}\otimes\cdots\otimes e_{2k+2kj_d})\rangle$$

A crucial property of v is that $\langle v, e_{\pi(1)}\otimes e_{\pi(2)}\otimes\cdots\otimes e_{\pi(n)}\rangle \neq 0$ iff π is a permutation, in which case it is equal to the sign of the permutation. It follows that the nonzero summands in $\langle v, p^{\otimes d}\rangle$ are precisely those for which $i = (i_1,\ldots,i_d)$ and $j = (j_1,\ldots,j_d)$ are permutations of $\{0,\ldots,d-1\}$. For a single summand with i and j permutations we see:

$$x_{i_1+1,j_1+1}\cdots x_{i_d+1,j_d+1} \\ \langle v, (e_{1+2ki_1}\otimes e_{2+2ki_1}\otimes\cdots\otimes e_{k+2ki_1}\otimes e_{k+1+2kj_1}\otimes e_{k+2+2kj_1}\otimes\cdots\otimes e_{2k+2kj_1}) \\ \otimes\cdots\otimes(e_{1+2ki_d}\otimes e_{2+2ki_d}\otimes\cdots\otimes e_{k+2ki_d}\otimes e_{k+1+2kj_d}\otimes e_{k+2+2kj_d}\otimes\cdots\otimes e_{2k+2kj_d})\rangle \\ = \operatorname{sgn}(i)^k\operatorname{sgn}(j)^k x_{i_1+1,j_1+1}\cdots x_{i_d+1,j_d+1}.$$

Hence, for even k we obtain $\langle v, p^{\otimes d}\rangle = d!\operatorname{Per}_d$. ◀

Finally, we put together the preceding results to prove Theorem 11.

▶ **Theorem 25** (Theorem 11, restated). *Let $k \geq 2$ be even. Consider the action of $G = \mathrm{SL}_{2kn}(\mathbb{C})$ on $V = \bigotimes^{2k}\mathbb{C}^{2kn}$. Then any set of generators for the invariant ring cannot have a polynomial sized (in n) succinct encoding, unless* VP = VNP.

Proof. We summarize the results so far. Let $k \geq 2$. Consider the action of $G = \mathrm{SL}_{2kn}(\mathbb{C})$ on $V = \bigotimes^{2k}\mathbb{C}^{2kn}$. Then:

1. There are no homogeneous invariant polynomials of degree $< n$.
2. The space of homogeneous invariant polynomials of degree n is 1-dimensional, and spanned by the hyperpfaffian polynomial $\mathrm{Pf}_{k,n}$.
3. The hyperpfaffian polynomial $\mathrm{Pf}_{k,n}$ is VNP-complete.

The rest of the proof proceeds along the same lines as in the proof of Theorem 8 in Section 3. If we had a poly-sized succinct encoding for the generators of this invariant ring, then one would be able to extract the lowest degree part, which would yield a poly-sized circuit computing $\mathrm{Pf}_{k,n}$. This is not possible unless VP = VNP, since $\mathrm{Pf}_{k,n}$ is VNP-complete. ◀

References

1 Manindra Agrawal and Somenath Biswas. Primality and identity testing via chinese remaindering. *Journal of the ACM (JACM)*, 50(4):429–443, 2003.

2 Zeyuan Allen-Zhu, Ankit Garg, Yuanzhi Li, Rafael Oliveira, and Avi Wigderson. Operator scaling via geodesically convex optimization, invariant theory and polynomial identity testing. *STOC*, 2018.

3 Alexander I. Barvinok. New algorithms for linear k-matroid intersection and matroid k-parity problems. *Mathematical Programming*, 69(1):449–470, 1995.

4 Markus Bläser, Christian Ikenmeyer, Gorav Jindal, and Vladimir Lysikov. Generalized matrix completion and algebraic natural proofs. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2018, pages 1193–1206, New York, NY, USA, 2018. ACM. doi:10.1145/3188745.3188832.

5 Peter Bürgisser, Cole Franks, Ankit Garg, Rafael Mendes de Oliveira, Michael Walter, and Avi Wigderson. Towards a theory of non-commutative optimization: Geodesic 1st and 2nd order methods for moment maps and polytopes. In David Zuckerman, editor, *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019, Baltimore, Maryland, USA, November 9-12, 2019*, pages 845–861. IEEE Computer Society, 2019. doi:10.1109/FOCS.2019.00055.

6 Peter Bürgisser, Cole Franks, Ankit Garg, Rafael Oliveira, Michael Walter, and Avi Wigderson. Efficient algorithms for tensor scaling, quantum marginals, and moment polytopes. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 883–897. IEEE, 2018.

7 Peter Bürgisser, Ankit Garg, Rafael Oliveira, Michael Walter, and Avi Wigderson. Alternating minimization, scaling algorithms, and the null-cone problem from invariant theory. *Proceedings of Innovations in Theoretical Computer Science (ITCS 2018)*, 2017. arXiv:1711.08039.

8 Peter Bürgisser and Christian Ikenmeyer. Fundamental invariants of orbit closures. *J. Algebra*, 477:390–434, 2017. doi:10.1016/j.jalgebra.2016.12.035.

9 Peter Bürgisser, Christian Ikenmeyer, and Greta Panova. No occurrence obstructions in geometric complexity theory. *J. Amer. Math. Soc.*, 32(1):163–193, 2019. doi:10.1090/jams/908.

10 Harm Derksen and Gregor Kemper. *Computational invariant theory*, volume 130 of *Encyclopaedia of Mathematical Sciences*. Springer, Heidelberg, enlarged edition, 2015. With two appendices by Vladimir L. Popov, and an addendum by Norbert A'Campo and Popov, Invariant Theory and Algebraic Transformation Groups, VIII. doi:10.1007/978-3-662-48422-7.

11 Harm Derksen and Visu Makam. Generating invariant rings of quivers in arbitrary characteristic. *J. Algebra*, 489:435–445, 2017. doi:10.1016/j.jalgebra.2017.06.035.

12 Harm Derksen and Visu Makam. Polynomial degree bounds for matrix semi-invariants. *Adv. Math.*, 310:44–63, 2017. doi:10.1016/j.aim.2017.01.018.

13 Harm Derksen and Visu Makam. Algorithms for orbit closure separation for invariants and semi-invariants of matrices. *arXiv e-prints*, January 2018. arXiv:1801.02043.

14 Harm Derksen and Visu Makam. An exponential lower bound for the degrees of invariants of cubic forms and tensor actions. *Advances in Mathematics*, 368:107136, 2020. doi:10.1016/j.aim.2020.107136.

15 Harm Derksen and Jerzy Weyman. Semi-invariants of quivers and saturation for littlewood-richardson coefficients. *Journal of the American Mathematical Society*, 13(3):467–479, 2000.

16 Mátyás Domokos and Alexander N Zubkov. Semi-invariants of quivers as determinants. *Transformation groups*, 6(1):9–24, 2001.

17 Michael A. Forbes and Amir Shpilka. Explicit Noether normalization for simultaneous conjugation via polynomial identity testing. In *Approximation, randomization, and combinatorial optimization*, volume 8096 of *Lecture Notes in Comput. Sci.*, pages 527–542. Springer, Heidelberg, 2013. doi:10.1007/978-3-642-40328-6_37.

18 William Fulton and Joe Harris. *Representation theory*, volume 129 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1991. A first course, Readings in Mathematics. doi:10.1007/978-1-4612-0979-9.

19 Ankit Garg, Leonid Gurvits, Rafael Oliveira, and Avi Wigderson. A deterministic polynomial time algorithm for non-commutative rational identity testing. In *57th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2016*, pages 109–117. IEEE Computer Soc., Los Alamitos, CA, 2016.

20 Ankit Garg, Leonid Gurvits, Rafael Oliveira, and Avi Wigderson. Algorithmic and optimization aspects of brascamp-lieb inequalities, via operator scaling. *Geometric and Functional Analysis*, 28(1):100–145, 2018.

21 Ankit Garg, Visu Makam, Rafael Oliveira, and Avi Wigderson. Search problems in algebraic complexity, GCT, and hardness of generator for invariant rings. *arXiv e-prints*, October 2019. arXiv:1910.01251v1.

22 Joshua A Grochow and Toniann Pitassi. Circuit complexity, proof complexity, and polynomial identity testing. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 110–119. IEEE, 2014.

23 Joshua A. Grochow and Youming Qiao. Isomorphism problems for tensors, groups, and cubic forms: completeness and reductions. *CoRR*, abs/1907.00309, 2019. arXiv:1907.00309.

24 David Hilbert. Ueber die Theorie der algebraischen Formen. *Math. Ann.*, 36(4):473–534, 1890. doi:10.1007/BF01208503.

25 David Hilbert. Ueber die vollen Invariantensysteme. *Math. Ann.*, 42(3):313–373, 1893. doi:10.1007/BF01444162.

26 Christian Ikenmeyer. *Geometric Complexity Theory, Tensor Rank, and Littlewood-Richardson Coefficients*. PhD thesis, Institute of Mathematics, University of Paderborn, 2012. Online available at http://digital.ub.uni-paderborn.de/ubpb/urn/urn:nbn:de:hbz:466:2-10472.

27 Gábor Ivanyos, Youming Qiao, and K. V. Subrahmanyam. Constructive non-commutative rank computation is in deterministic polynomial time. *Comput. Complexity*, 27(4):561–593, 2018. doi:10.1007/s00037-018-0165-7.

28 Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.

29 Jan Krajíček. *Proof complexity*, volume 170. Cambridge University Press, 2019.

30 Visu Makam and Avi Wigderson. Singular tuples of matrices is not a null cone (and, the symmetries of algebraic varieties). *arXiv e-prints*, September 2019. arXiv:1909.00857.

31 Ketan D. Mulmuley. Geometric complexity theory V: Efficient algorithms for Noether normalization. *Journal of the American Mathematical Society*, 30(1):225–309, 2017. doi:10.1090/jams/864.

32 Ketan D. Mulmuley and Milind Sohoni. Geometric complexity theory. I. An approach to the P vs. NP and related problems. *SIAM J. Comput.*, 31(2):496–526, 2001. doi:10.1137/S009753970038715X.

33 Ketan D. Mulmuley and Milind Sohoni. Geometric complexity theory. II. Towards explicit obstructions for embeddings among class varieties. *SIAM J. Comput.*, 38(3):1175–1206, 2008. doi:10.1137/080718115.

34 David Mumford. *Geometric invariant theory*. Ergebnisse der Mathematik und ihrer Grenzgebiete, Neue Folge, Band 34. Springer-Verlag, Berlin-New York, 1965.

35 Aidan Schofield and Michel Van den Bergh. Semi-invariants of quivers for arbitrary dimension vectors. *Indagationes Mathematicae*, 12(1):125–138, 2001.

36 Jacob T Schwartz. Probabilistic algorithms for verification of polynomial identities. In *International Symposium on Symbolic and Algebraic Manipulation*, pages 200–215. Springer, 1979.

37 Amir Shpilka, Amir Yehudayoff, et al. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends® in Theoretical Computer Science*, 5(3–4):207–388, 2010.

38 Volker Strassen. Vermeidung von divisionen. *Journal für die reine und angewandte Mathematik*, 264:184–202, 1973.

39 Bernd Sturmfels. *Algorithms in Invariant Theory*. Texts & Monographs in Symbolic Computation. Springer, 2nd edition, 2008.

40 David L. Wehlau. Constructive invariant theory for tori. *Ann. Inst. Fourier (Grenoble)*, 43(4):1055–1066, 1993. URL: http://www.numdam.org/item?id=AIF_1993__43_4_1055_0.