Alternating Minimization, Scaling Algorithms, and the Null-Cone Problem from Invariant Theory*†

Peter Bürgisser[1], Ankit Garg[2], Rafael Oliveira[3], Michael Walter[4], and Avi Wigderson[5]

1 **Institut für Mathematik, Technische Universität Berlin, Germany**
pbuerg@math.tu-berlin.de
2 **Microsoft Research New England, Cambridge, USA**
garga@microsoft.com
3 **Department of Computer Science, University of Toronto, Canada and Department of Computer Science, Princeton University, USA**
rafael@cs.toronto.edu
4 **QuSoft, Korteweg-de Vries Institute for Mathematics, Institute of Physics, and Institute for Logic, Language and Computation, University of Amsterdam, The Netherlands and Stanford Institute for Theoretical Physics, Stanford University, USA**
m.walter@uva.nl
5 **Institute for Advanced Study, Princeton, USA**
avi@ias.edu

Abstract

Alternating minimization heuristics seek to solve a (difficult) global optimization task through iteratively solving a sequence of (much easier) local optimization tasks on different parts (or blocks) of the input parameters. While popular and widely applicable, very few examples of this heuristic are rigorously shown to converge to optimality, and even fewer to do so efficiently.

In this paper we present a general framework which is amenable to rigorous analysis, and expose its applicability. Its main feature is that the local optimization domains are each a group of invertible matrices, together naturally acting on tensors, and the optimization problem is minimizing the norm of an input tensor under this joint action. The solution of this optimization problem captures a basic problem in Invariant Theory, called the *null-cone problem*.

This algebraic framework turns out to encompass natural computational problems in combinatorial optimization, algebra, analysis, quantum information theory, and geometric complexity theory. It includes and extends to high dimensions the recent advances on (2-dimensional) *operator scaling* [14, 11, 22].

Our main result is a fully polynomial time approximation scheme for this general problem, which may be viewed as a multi-dimensional scaling algorithm. This directly leads to progress on some of the problems in the areas above, and a unified view of others. We explain how faster convergence of an algorithm for the same problem will allow resolving central open problems.

Our main techniques come from Invariant Theory, and include its rich *non-commutative duality theory*, and new bounds on the bitsizes of coefficients of *invariant polynomials*. They enrich the algorithmic toolbox of this very computational field of mathematics, and are directly related to some challenges in geometric complexity theory (GCT).

1998 ACM Subject Classification F.2.1 Numerical Algorithms and Problems, G.1.6 Optimization

* This work was partially supported by an NWO Veni grant no. 680-47-459 and a DFG grant BU 1371 2-2.
† A full version of the paper is available at https://arxiv.org/abs/1711.08039, [6].

9th Innovations in Theoretical Computer Science Conference (ITCS 2018).
Editor: Anna R. Karlin; Article No. 24; pp. 24:1–24:20

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

Keywords and phrases alternating minimization, tensors, scaling, quantum marginal problem, null cone, invariant theory, geometric complexity theory

Digital Object Identifier 10.4230/LIPIcs.ITCS.2018.24

1 Introduction and summary of results

Alternating minimization refers to a large class of heuristics commonly used in optimization, information theory, statistics and machine learning. It addresses optimization problems of the following general form. Given a function

$$f\colon \mathcal{X}_1 \times \cdots \times \mathcal{X}_d \to \mathbb{R} \tag{1}$$

the goal is to find a global optimum

$$\inf_{x_1 \in \mathcal{X}_1, \ldots, x_d \in \mathcal{X}_d} f(x_1, \ldots, x_d). \tag{2}$$

While both the function f and its domain may be extremely complicated, in particular non-convex, the decomposition of the domain to d *blocks* is such that the *local* optimization problems are all feasible. More precisely, for every $i \in [d]$, and for every choice of $\alpha_j \in \mathcal{X}_j$ with $j \neq i$, computing

$$\inf_{x_i \in \mathcal{X}_i} f(\alpha_1, \ldots, \alpha_{i-1}, x_i, \alpha_{i+1}, \ldots, \alpha_d)$$

is easy.

A natural heuristic in such cases is to repeatedly make local improvements to different coordinates. Namely, we start from some arbitrary vector $\alpha^{(0)}$, and generate a sequence $\alpha^{(0)}, \alpha^{(1)}, \ldots, \alpha^{(t)}, \ldots$, such that $\alpha^{(t+1)}$ is obtained by solving the above local optimization problem for some $i = i^{(t)}$, freeing up this variable while fixing all other coordinates according to $\alpha^{(t)}$. The argmin of the optimum replaces the ith coordinate in $\alpha^{(t)}$ to create $\alpha^{(t+1)}$.

There is a vast number of situations which are captured by this natural framework. For one example (we'll see more), consider the famous Lemke-Howson [28] algorithm for finding a Nash-equilibrium in a 2-player game. Here $d = 2$, and starting from an arbitrary pair of strategies, proceed by alternatingly finding a "best response" strategy for one player given the strategy of the other. This local optimization is simply a linear program that can be efficiently computed. As is well known, this algorithm always converges, but in some games it requires exponential time!

So, the basic questions which arise in some settings are under which conditions does this general heuristic converge at all, and even better, when does it converge efficiently. Seemingly the first paper to give *provable* sufficient conditions (the "5-point property") for convergence in a *general* setting was Csiszár and Tusnády [8]. An application is computing the distance (in their case, KL-divergence, but other metrics can be considered) between two convex sets in $\mathbb{R}^n$, as well as the two closest points achieving that distance. Again, the local problem (fixing one point and finding the closest to it in the other convex set) is a simple convex program that has a simple efficient algorithm. For two affine subspaces and the ℓ_2-distance, von Neumann's alternating projection method [36] is an important special case with numerous applications. As mentioned, in the past three decades numerous papers studied various variations and gave conditions for convergence, especially in cases where $d > 2$ (often called also "block-coordinate descent") – we cite only a few recent ones [46, 37, 44] which address a variety of problems and have many references. Much much fewer cases

are known in which convergence is fast, namely requires only a polynomial number of local iterations. Some examples include the recent works on matrix completion [23, 17]. Our aim is to develop techniques that expand *efficient* convergence results in the following algebraic setting, which as we shall see is surprisingly general and has many motivations.

We will consider the case of minimizing (1) with very specific domain and function, that we in turn explain and then motivate (so please bear with this necessary notation). First, each of the blocks $\mathcal{X}_i$ is a special linear group, $\mathrm{SL}_{n_i}(\mathbb{C})$ (which we will also abbreviate by $\mathrm{SL}(n_i)$), namely the invertible complex matrices of some size n_i and determinant one. Note that this domain is quite complex, as these groups are certainly not convex, and not even compact. To explain the function f to be minimized over this domain, consider the natural linear transformation (basis change) of the vector space $\mathbb{C}^{n_i}$ by matrices $A_i \in \mathrm{SL}(n_i)$. Thus, the product group $G := \mathrm{SL}(n_1) \times \mathrm{SL}(n_2) \times \cdots \times \mathrm{SL}(n_d)$ acts on tensors $X \in V := \mathrm{Ten}(n_0, n_1, n_2, \ldots, n_d) = \mathbb{C}^{n_0} \otimes \mathbb{C}^{n_1} \otimes \mathbb{C}^{n_2} \otimes \cdots \otimes \mathbb{C}^{n_d}$ of order $d + 1$[1], where (the basis change) A_i is applied to all vectors (slices, "fibers") of X along the ith dimension. Now the objective function $f = f_X$ depends on an input tensor $X \in V$; for any vector of matrices $A = (A_1, A_2, \ldots A_d) \in G$, $f_X(A)$ is defined simply as the ℓ_2-norm squared[2] of $A \cdot X$, the tensor resulting by the action of A on X. Summarizing this description, for input $X \in V$ we would like to compute (or approximate) the quantity we call *capacity* of X, denoted $\mathrm{cap}(X)$ defined by

$$\mathrm{cap}(X) := \inf_{A \in G} \|A \cdot X\|_2^2. \tag{3}$$

In particular, we would like to decide the *null-cone problem*: is the capacity of X zero or not?

While this formulation readily lends itself to an alternating minimization procedure (the local problems have an easily computed closed-form solution), the algorithms we will develop will be related to a dual optimization (scaling) problem that also has a similar form. *Our main result will be a fully polynomial time approximation scheme (FPTAS) for that dual problem!* This will also allow us to solve the null-cone problem. However we defer this discussion and turn to motivations.

Now where do such problems and this notion of capacity naturally arise? Let us list a few sources, revealing how fundamental this framework really is. Some of these connections go back over a century and some were discovered in the past couple of years. Also, some are less direct than others, and some overlap. Some of them, and others, will be elaborated on throughout the paper.

- **Combinatorial Optimization.** Computing matroid intersection (for linear matroids over $\mathbb{Q}$) easily reduces to checking zeroness of capacity for $d = 2$ for very "simple" inputs X. This was discovered in [14], who showed it remains true for an implicit version of this problem for which there was no known polynomial time algorithm. His alternating minimization algorithm (called today *operator scaling*) is efficient for such simple inputs. It also generalizes similar algorithms for such problems as perfect matchings in bipartite graphs and maximum flow in networks, where again zeroness of capacity captures the decision versions of the problems. The associated alternating minimization algorithm gives a very different and yet efficient way than is taught in algorithms texts for these classical problems (these go by the name of *matrix scaling*, originated in [38] and described explicitly in [30]).

[1] Equivalently, this means that we study n_0-tuples of tensors of order d.
[2] Namely, the sum of squares of the absolute values of the entries.

- **Non-commutative algebra.** The most basic computational problem in this field is the *word problem*, namely, given an expression over the free skew field (namely, a formula over $\mathbb{Q}$ in non-commutative variables), is it identically zero[3]. As shown in [11] this problem is reducible to testing if the capacity is 0, for $d = 2$. [11] significantly extends [14], proving that his algorithm is actually an FPTAS for *all* inputs, resolving the $d = 2$ of our general problem above, and providing the first polynomial time algorithm for this basic word problem (the best previous algorithm, deterministic or probabilistic, required exponential time)[4].
- **Analysis.** We move from efficiently testing *identities* to efficiently testing *inequalities*. The Brascamp-Lieb inequalities [4, 29] are an extremely broad class capturing many famous inequalities in geometry, probability, analysis and information theory (including Hölder, Loomis-Whitney, Shearer, Brunn-Minkowski, ...). A central theorem in this field [29] expresses feasibility, and the tightest possible constant if feasible, for every instantiation of such inequality in terms of capacity (defined somewhat differently, but directly related to ours). In [12] a direct alternating minimization algorithm[5] is given which efficiently approximates it on every input! Previous techniques to even bound capacity were completely ineffective, using compactness arguments.
- **Quantum Information Theory and Geometric Complexity Theory.** We finally move to arbitrary d. The *quantum marginal problem* (generalizing the classical *marginal problem* in probability theory) asks if a collection of marginals (namely, density matrices) on d subsystems of a given quantum system are *consistent*, namely is there a density matrix on the whole system whose partial traces on the given subsystems result in these marginals [27]. When the state of the whole system is pure and the marginals are proportional to identity matrices, each subsystem is maximally entangled with the others. It is an important task to distill entanglement, that is, given a tensor, to transform it into such a *locally maximally entangled* state by SLOCC[6] operations [2, 26]. It turns out that this amounts precisely to the capacity optimization problem, and the matrices which minimize it solve this distillation problem. One can thus view our FPTAS as achieving approximate entanglement distillation (see, e.g., [41, 43, 42]). We note that for $d = 3$, the non-zeroness of capacity captures a refinement of the *asymptotic positivity* of special *Kronecker coefficients*, a central notion in representation theory and geometric complexity theory (GCT) of [33, 5, 20]. We refer to [6] for more detail.
- **Invariant Theory.** We will be brief, and soon discuss this area at length, as it underlies most of the work in this paper. Invariant theory studies symmetries, namely group actions on sets, and their invariants. The action of our particular group G above on the particular linear space of tensors V is an example of the main object of study in invariant theory: actions of reductive[7] groups G on linear spaces V. A central notion in the (geometric) study of such actions is the *null cone*; it consists of all elements in $v \in V$ which G maps arbitrarily close to $0 \in V$. Here the connection to our problem is most direct: the null cone of the action in our framework is *precisely* the input tensors X whose capacity is 0! We stress that previous algorithms for the null cone (for general reductive actions) required

[3] This is an analog of the PIT problem in algebraic complexity, for non-commutative formulas with division.
[4] This problem was solved for finite characteristics by different methods in [22].
[5] Which can be viewed also as a reduction to the operator scaling algorithm in [11].
[6] Acronym for *Stochastic Local Operations and Classical Communication*; these are the natural actions when each of the subsystems is owned by a different party.
[7] A somewhat technical term which includes all classical linear groups.

doubly exponential time (e.g., [39, Algorithm 4.6.7]). In contrast, our algorithm (for the general framework above) decides if a given tensor is in the null cone in *singly* exponential time! We will also discuss special cases where our FPTAS achieves polynomial time. The same null-cone problem is a direct challenge among the algorithmic problems suggested by Mulmuley towards the GCT program in [32].

We now turn to discuss at more length the relevant features of invariant theory, explaining in particular the dual optimization problem that our algorithm actually solves. Then we will state our main results.

1.1 Invariant theory

Invariant theory is an area of mathematics in which computational methods and efficient algorithms are extremely developed (and sought). The reader is referred to the excellent texts [9, 39] for general background, focused on algorithms. Using invariant theory was key for the recent development of efficient algorithms for operator scaling mentioned above [11, 22], while these algorithms in turn solved basic problems in this field itself. This paper proceeds to expand the applications of invariant theory for algorithms (and computational complexity!) and to improve the efficiency of algorithms for basic problems in the field.

Invariant theory is an extremely rich theory, starting with seminal works of Cayley [7][8], to compute invariants of $\mathrm{SL}(n)$, of Hilbert [18, 19][9] and others in the 19th century. Again, we focus on linear actions of reductive groups G on vector spaces V. It turns out that in this linear setting the relevant invariants are polynomial functions of the variables (entries of vectors in V with respect to some fixed basis), and the central problem is to understand the *invariant ring*, namely all polynomials which are left invariant (compute the same value) under the action of G. Two familiar examples, which illustrate how good an understanding we can hope for, are the following:

- When $G = S_n$, the group of permutations of the coordinates of an n-vector in $V = \mathbb{F}^n$, the ring of invariants are (naturally) all *symmetric polynomials*, which is (finitely!) generated by the n elementary symmetric polynomials.
- When $G = \mathrm{SL}(n) \times \mathrm{SL}(n)$[10] acts on $n \times n$ matrices by changing basis of the rows and columns respectively, the ring of invariants is generated by one polynomial in the entries of this matrix: the *determinant*.

One of Hilbert's celebrated results in [18, 19] was that the invariant ring is *always* finitely generated. Further understanding calls for listing generators, finding their algebraic relations, making them as low degree as possible, as few as possible, and as easily computable as possible. Many new motivations for these arose from the GCT program mentioned above (and below).

The action of G naturally carves the space V into *orbits*, namely sets of the form $G \cdot v$ consisting of all images of $v \in V$ under the action of G. Understanding when are two vectors in the same orbit captures numerous natural problems, e.g., *graph isomorphism* (when is a graph in the orbit of another, under vertex renaming), *module isomorphism* (when is a vector of matrices in the orbit of another, under simultaneous conjugation), and others. Over the complex numbers $\mathbb{C}$ it is more natural to consider *orbit closures* and answer similar problems regarding membership of vectors in such orbit closures. For an important example,

[8] Which describes his ingenious algorithm, the Omega-process.
[9] In which he proves his celebrated Nullstellensatz and Finite Basis theorem.
[10] Here $\mathrm{SL}(n)$ is the group of invertible matrices of determinant 1.

the central approach of Mulmuley and Sohoni's GCT program [33] to Valiant's conjecture that VP≠VNP is to translate it to the question of whether the permanent is in the orbit closure of (a polynomially larger) determinant under a linear action on the matrix entries. Understanding orbit closures is the subject of *Geometric Invariant Theory*, starting with the seminal work of Mumford [34] in the middle of the last century. He proved that answering such questions is *equivalent* to computing invariants. That is, the orbit closures of two vectors $v, w \in V$ intersect if and only if $p(v) = p(w)$ for all invariant polynomials p.

An important special case is to understand the orbit closures that contain the zero vector $0 \in V$. Note that by the theorem above, these are all vectors $v \in V$ for which $p(v) = 0$ holds for all invariant polynomials p without constant term. This is an algebraic variety (nothing more than a zero set a system of polynomials), which is called the *null cone* of the action of G on V. Understanding the null cone is also central to geometric complexity theory from a topological viewpoint when *projectivizing* the space of orbits (making a point of each one, "modding out" by the group action).

A key to understanding the null cone is a duality theory for the group action, which may be viewed as a non-commutative analog of the duality theory for linear programming. We will elaborate on this duality theory, essential to our work, in Sections 3 and 4 and only sketch the essentials here. It is clear that a vector $v \in V$ is in the null cone if its capacity is equal to zero, and so can be "certified" by exhibiting a sequence of group elements g_t such that $\|g_t \cdot v\|$ approaches zero. How can we "certify" that a vector $v \in V$ is *not* in the null cone? The capacity formulation means that there must be some fixed $\delta > 0$ such that, for every element $w = g \cdot v$ in the orbit of v, $\|w\| \geq \delta$. Consider a point w attaining this minimum distance to 0 (assume it exists). There is a way to write down a non-commutative analog of the "Lagrange multiplier" conditions giving equations saying essentially that the derivative of the group action on w in any direction is zero. It turns out that the distance to satisfying these equations is *another, dual* optimization problem. In particular, to "certify" non-membership of v in the null cone, it suffices to exhibit a sequence of group elements g_t such that $g_t \cdot v$ approaches distance 0 from satisfying these equations. Again, we give plenty more detail on that in Sections 3 and 4.

What is remarkable is that in our setting, when G is a product of d groups as above, the new optimization has the exact same form as the original one we started with – only with a different function to minimize! Moreover, the set of equations decouples to d subsets, and optimizing each *local* one is efficiently computable. In short, this makes our new optimization problem again amenable to alternating minimization. Further, the conditions that we need to satisfy may be viewed as "scaling conditions", which for $d = 2$ can be shown to be equivalent to the matrix scaling conditions in the commutative case, and to the operator scaling conditions in the non-commutative case. Thus, these scaling algorithms in combinatorial optimization to "doubly stochastic" position naturally arise from general considerations of duality in geometric invariant theory. We will continue to borrow this name and call a tensor that satisfies the minimization conditions *"d-stochastic"*, and we quantify how well a tensor Y satisfies d-stochasticity by a distance measure denoted $\mathrm{ds}(Y)$. For an input tensor X, we then seek to compute the minimal distance to d-stochasticity, denoted $\mathrm{dds}(X)$, which is the infimum of $\mathrm{ds}(Y)$ over all Y in the orbit of X (and formally defined in Theorem 6). Thus, summarizing, with G and V as before Eq. (3), our new, dual optimization problem is

$$\mathrm{dds}(X) := \inf_{A \in G} \mathrm{ds}(A \cdot X). \tag{4}$$

In our framework, $\mathrm{ds}(Y)$ has a very simple form. Taking the quantum information theory view of the the problem (described above in the motivation), the tensor Y captures a quantum

system of d local parts; then $\mathrm{ds}(Y)$ is simply the total ℓ_2-distance squared of the d subsystems to being maximally mixed (that is, proportional to normalized identity density matrices).

We now proceed to describe our main results: the problems we address, our algorithms for them, and their complexities. More formal statements of all will appear in the technical sections.

1.2 Our results

We first describe our technical results and then the conceptual contribution of our paper.

Technical results

We fix G and V as above. While we work over the complex numbers, we will think of an input tensor $X \in V$ as an array of integers (one can consider rational numbers, but this is no gain in generality) represented in binary. The input size parameters will be the number of tensor entries $n = n_0 \times n_1 \times \cdots \times n_d$, and the maximum binary length of each entry, which will be denoted by b. So the total input length may be thought of as nb.

Recall again that we have considered two dual optimization problems[11] above for which the input is X:

$$\mathrm{cap}(X) = \inf_{A \in G} \|A \cdot X\|_2^2, \qquad \mathrm{dds}(X) = \inf_{A \in G} \mathrm{ds}(A \cdot X). \tag{3,4}$$

These lead to both exact and approximate computational problems for which we give new algorithms.

The exact *null-cone problem* we will consider is to test if the input tensor X is in the null cone of the group action G. As we discussed, X is in the null cone iff $\mathrm{cap}(X) = 0$ and iff $\mathrm{dds}(X) > 0$ (!).

We will give two different *exponential* time algorithms for this problem. We note that the previous best algorithms (which work in the greater generality of all reductive group actions) required *doubly exponential* time! These algorithms may be very useful for the study of invariants for the actions of "small", specific groups. We will discuss these algorithms soon.

The approximate problem we will consider will be to approximate $\mathrm{dds}(X)$[12]. Our approximation of dds runs in polynomial time in the input length nb and the (additive) approximation ϵ – thus a FPTAS is our main result[13], and we state it first.

▶ **Theorem 1** (Main theorem). *There is a* $\mathrm{poly}(n, b, \frac{1}{\varepsilon})$ *time deterministic algorithm (Algorithm 1) that, given a tensor* $X \in V$ *with integer coordinates of bit size bounded by* b*, either identifies that* X *is in the null cone or outputs a "scaling"* $Y \in G \cdot X$ *such that* $\mathrm{ds}(Y) < \varepsilon$.

[11] While we do not know of any precise relationship between the value of these optimization problems, the vanishing behaviour of these two problems, as explained above, is dual to each other.

[12] One could also consider the same for $\mathrm{cap}(X)$, but unlike $\mathrm{dds}(X)$, we have no faster algorithm for approximating $\mathrm{cap}(X)$ than computing it exactly.

[13] Some readers might object to our use of the term FPTAS for our algorithm since we have an additive approximation guarantee whereas the term is commonly used for multiplicative guarantees. However, note that a multiplicative guarantee for either $\mathrm{cap}(X)$ or $\mathrm{dds}(X)$ will need to determine their vanishing behaviour. This captures the null-cone problem, a polynomial time algorithm for which is the main open problem left open by this work. Also note that we don't have any guarantee on approximating $\mathrm{dds}(X)$ if $\mathrm{dds}(X) > 0$, which is also left as an interesting open problem. However, approximating $\mathrm{dds}(X)$ when it is 0 is more fundamental because of the connection to the null cone.

We present the algorithm right below in Section 3.1 and analyze it in Section 3.2. It is a realization of the alternating minimization scheme discussed before, and may be viewed as a *scaling algorithm*. Indeed, it generalizes the FPTAS given in [11] for $d = 2$, which is the operator scaling case. While the analysis is similar in spirit, and borrows many ideas from [11], we note that for higher dimensions $d > 2$, the understanding of invariant polynomials was much poorer. Thus, new ideas are needed, and in particular we give explicit[14] generators (via classical Invariant Theory techniques) of the invariant ring (for all degrees), all of which have small coefficients, and we bound the running time in a way which is *independent* of degree bounds.

Moving to the algorithms for the exact null-cone problem, as discussed, we have two exponential time ones, improving the known doubly exponential ones.

The first algorithm is a natural instantiation of the FPTAS above, with the right choice of ε. Specifically, we prove that when if X is not in the null cone, $\mathrm{dds}(X) \geq \exp(-O(n \log n))$. So picking $\varepsilon = \exp(-O(n \log n))$ suffices. This algorithm is described in Theorem 8.

Our second algorithm is much more general, and applies to the action of a product of general linear groups on *any* vector space V, not just the the set of tensors. Here, instead of giving explicit generators, we simply prove their existence, via Cayley's famous *Omega-process*. Namely, we prove that in this very general case there are always generating invariants which have both exponential degree and whose coefficients have exponential bit length. These bounds turns out to be sufficient to carry out our time analysis. The bound on the size of coefficients is (to the best of our knowledge) a new structural result in Invariant Theory, and what our analysis demonstrates is its usefulness for solving computational problems.

The corollaries of these main results to specific areas, which we described in some of the motivation bullets earlier in the introduction, are presented in the different technical sections.

Conceptual contributions

We believe that the general framework we present, namely an *algebraic framework of alternating minimization*, establishes an important connection between the fields of optimization and invariant theory. As we saw, this framework is very general and encompasses many natural problems in several fields. It exposes the importance of *symmetry* in many of these problems, and so invites the use of tools from invariant theory which studies symmetry to the analysis of algorithms. This is akin to the GCT program, which connects algebraic complexity theory with invariant theory and representation theory, exposing the importance of using symmetry to proving lower bounds.

At the same time we expose basic computational problems of invariant theory, which are in great need of better algorithms and algorithmic techniques from computer science and optimization to solve them. The advent of operating scaling already pointed out to this power in their utility of alternate minimization and scaling *numerical* algorithms to a field in which most work was *symbolic*, and we expand on this theme in this paper. But there are many other optimization methods which can be useful there, and we note only one other example, that may in particular be useful for the very null-cone problem we study. While the problem of capacity optimization is not convex under the usual Euclidean metric, it is actually *geodesically convex* [24, 35, 45] if one moves to the natural (hyperbolic) metric on the group. This opens up this problem (and related ones) to avenues of attack from techniques of classical optimization (gradient and other descent methods, interior point methods, etc.)

[14] The generators are explicitly defined but may not be efficiently computable by algebraic circuits.

when generalized and adapted to such metrics. We believe that the area of geodesic convex optimization, which itself in its nascent stages (from an algorithmic standpoint), is likely to have a significant impact on algorithmic invariant theory. See more on algorithmic techniques for geodesic convexity in the work of [47] and the references therein.

2 Null-cone problem as an optimization problem

In the introduction, we defined the *null cone* as the set of tensors X such that the zero vector $0 \in V$ lies in the orbit closure $\overline{G \cdot X}$, i.e., there exists a sequence of group elements $A^{(1)}, A^{(2)}, \ldots$ sending the vector to zero, $\lim_{j\to\infty} A^{(j)} \cdot X = 0$. Thus the *null-cone problem* amounts to the optimization problem (3): A tensor $X \in V = \mathrm{Ten}(n_0, n_1, \ldots, n_d)$ is in the null cone if and only if the *capacity* is equal to zero, namely:

$$\mathrm{cap}(X) = \inf_{A\in G} \|A \cdot X\|_2^2 = \min_{Y\in\overline{G\cdot X}} \|Y\|_2^2 = 0.$$

▶ Remark. We are denoting the above optimization problem by $\mathrm{cap}(X)$, short for *capacity*, to remain consistent with similar notions defined in previous papers like *[16, 14, 15, 11]*. In most of these cases, the capacity notion that they consider is also looking at the optimization problem: "minimum ℓ_2-norm in an orbit closure" for specific group actions. As phrased in these papers, they might look different (e.g., they involve determinants in place of ℓ_2 norms) but there is a tight connection between the two via the AM-GM inequality (i.e., the inequality of arithmetic and geometric means) in one direction and a single alternating minimization step in the other direction.

A fundamental theorem of Hilbert [19] and Mumford [34] gives an alternative characterization of the null cone. It states that the null cone is precisely the set of tensors X on which all invariant polynomials P (without constant term) vanish. This is the starting point to the field of Geometric Invariant Theory.

It is instructive to see why $P(X) = 0$ must hold for any tensor X in the null cone: Since P is G-invariant, $P(Y) = P(X)$ for all Y in the G-orbit of X, and, by continuity, also for all Y in the closure of the orbit. If the tensor X is in the null cone, then $Y = 0 \in V$ is in its closure. But then $P(X) = P(0) = 0$, since the polynomial P has no constant term. In [6] we derive a more subtle, *quantitative* version of this observation. It will be a fundamental ingredient to the analysis of our algorithm.

Since the group $G = \mathrm{SL}(n_1) \times \cdots \times \mathrm{SL}(n_d)$ consists of tuples $A = (A_1, \ldots, A_d)$, an alternating minimization algorithm suggests itself for solving the optimization problem $\mathrm{cap}(X)$: Fix an index $i \in [d]$ and optimize only over a single A_i, leaving the $(A_j)_{j\neq i}$ unchanged. This gives rise to an optimization problem of the following form:

$$\inf_{A_S\in\mathrm{SL}(n_S)} \|(A_S \otimes I_{n_T}) \cdot Y\|_2^2$$

Here $n_S = n_i$, $n_T = \prod_{j\neq i} n_j$ and $Y = A \cdot X$. This optimization problem has a closed form solution in terms of the following quantum mechanic analog of the marginals of a probability distribution:

▶ **Definition 2** (Quantum marginal)**.** Given a matrix ρ acting on $\mathcal{H}_S \otimes \mathcal{H}_T$, where $\mathcal{H}_S$ and $\mathcal{H}_T$ are Hilbert spaces of dimensions n_S and n_T, respectively (e.g., $\mathcal{H}_S = \mathbb{C}^{n_S}$ and $\mathcal{H}_T = \mathbb{C}^{n_T}$). Then there is a unique matrix ρ_S acting on $\mathcal{H}_S$, called the *quantum marginal* (or reduced density matrix) of S, such that

$$\mathrm{tr}\left[(M_S \otimes I_T)\,\rho\right] = \mathrm{tr}\left[M_S \rho_S\right] \tag{5}$$

for every M_S acting on $\mathcal{H}_S$.

This point of view gives rise to an important interpretation of our result [6].

It can be easily seen that if ρ is positive semidefinite (PSD) then ρ_S is PSD, and that it has the same trace as ρ. It is given by the following explicit formula

$$\rho_S = \sum_{k=1}^{n_T} (I_S \otimes e_k)^\dagger \, \rho \, (I_S \otimes e_k) \, .$$

Here the e_k's are the elementary column vectors of dimension n_T.

Now observe that

$$\|(A_S \otimes I_T) \cdot Y\|_2^2 = \mathrm{tr}\left[\left(A_S^\dagger A_S \otimes I_T\right) YY^\dagger\right] .$$

Let ρ_i be the partial trace of $\rho = YY^\dagger$ obtained by tracing out all the Hilbert spaces except the i^{th} one, and rename $A_i = A_S$. Then, using Eq. (5),

$$\mathrm{tr}\left[\left(A_S^\dagger A_S \otimes I\right) YY^\dagger\right] = \mathrm{tr}\left[A_i^\dagger A_i \rho_i\right] .$$

Hence we are left with the optimization problem:

$$\inf_{A_i \in \mathrm{SL}(n_i)} \mathrm{tr}\left[A_i A_i^\dagger \rho_i\right] . \tag{6}$$

We can see by the AM-GM inequality that the optimum of the program (6) is $n_i \det(\rho_i)^{1/n_i}$, which is achieved for $A_i = \det(\rho_i)^{1/2n_i} \rho_i^{-1/2}$ (if ρ_i is not invertible then we define the inverse on the support of ρ_i; the infimum will at any rate be zero).

We thus obtain the core of an alternating minimization algorithm for the capacity $\mathrm{cap}(X)$. At each step t, select an index $i \in [d]$, compute the quantum marginal ρ_i of the current tensor $X^{(t)}$, and update by performing the following *scaling step*:

$$X^{(t+1)} \leftarrow \det(\rho_i)^{1/2n_i} \rho_i^{-1/2} \cdot X^{(t)}. \tag{7}$$

Here and in the following we use the abbreviation

$$A_i \cdot Y := (I_{n_0} \otimes I_{n_1} \otimes \ldots \otimes I_{n_{i-1}} \otimes A_i \otimes I_{n_{i+1}} \otimes \ldots \otimes I_{n_d}) \cdot Y,$$

where we act nontrivially on the i-th tensor factor only.

We will analyze essentially this algorithm, augmented by an appropriate method for selecting the index i at each step and a stopping criterion (see Algorithm 1 in the next section).

3 Noncommutative duality and the tensor scaling algorithm

To obtain a scaling algorithm with rigorous guarantees, we will now derive a dual formulation of the optimization problem $\mathrm{cap}(X)$. This will achieved by a theorem from Geometric Invariant Theory, due to Kempf and Ness, which can be understood as part of a noncommutative duality theory.

We briefly return to the setting of a general action of a group G on a vector space V to explain this result. Fix a vector $v \in V$ and consider the function $f_v(g) := \|g \cdot v\|_2^2$. The *moment map* at v, $\mu(v)$, measures how much $f_v(g)$ changes when we perturb g around the identity. So $\mu(v)$ is just the derivative of the function $f_v(g)$ at the identity element (in a precise sense which we do not need to define here). Now suppose v is not in the null cone. Then there exists a *nonzero* vector $w \in \overline{G \cdot v}$ of minimal norm in the orbit closure. Since

the norm is minimal, perturbing w by the action of group elements g close to the identity element does not change the norm to first order. Hence $\mu(w) = 0$. To summarize, if v is not in the null cone, then there exists $0 \neq w \in \overline{G \cdot v}$ such that $\mu(w) = 0$. This condition can be understood as a vast generalization of the notion of "doubly stochastic", which one might be tempted to call *"G-stochastic"*.

A remarkable theorem due to Kempf and Ness [24] asserts that this is not only a necessary but in fact also sufficient condition, i.e., v is not in the null cone if and only if there exists $0 \neq w \in \overline{G \cdot v}$ such that $\mu(w) = 0$. This is a *duality theorem* and (along with the Hilbert-Mumford criterion discussed below) should be thought of as a non-commutative version of linear programming duality, which arises when the group G is commutative (see Section 3 below, [40], and [6]). For a detailed discussion of moment maps, we refer the reader to [45, 42].

We now return to our setting where $G = \mathrm{SL}(n_1) \times \cdots \times \mathrm{SL}(n_d)$ acts on tensors in $V = \mathrm{Ten}(n_0, n_1, \ldots, n_d)$. We first define the notion of a scaling and then instantiate the Kempf-Ness theorem in the situation at hand:

▶ **Definition 3** (Scaling). A tensor $Y \in \mathrm{Ten}(n_0, n_1, \ldots, n_d)$ is called a (*tensor*) *scaling* of $X \in \mathrm{Ten}(n_0, n_1 \ldots, n_d)$ if $Y \in G \cdot X$.

This notion of scaling generalizes the notions of operator scaling [14, 11] and matrix scaling [38]. It is immediate from the definition of the capacity that a tensor X is in the null cone if and only if any of its scalings Y lies in the null cone.

We now state the Kempf-Ness theorem for tensors:

▶ **Theorem 4** (Duality, special case of [24]). *A tensor $X \in \mathrm{Ten}(n_0, \ldots, n_d)$ is* not *in the null cone if and only if there exists $Y \in \overline{G \cdot X}$ such that the quantum marginals $\rho_1, \ldots, \rho_d$ of $\rho = YY^\dagger / Y^\dagger Y$ of the last d tensor factors are given by $\rho_i = I_{n_i}/n_i$.*

We note that the trace of $\rho = YY^\dagger / Y^\dagger Y$ is one. This normalization is very natural from the quantum viewpoint. In quantum theory, PSD matrices of unit trace describe quantum states of multi-partite systems and the quantum marginals describe the state of subsystems. The condition that ρ_i is proportional to the identity means that the state of the k^{th} subsystem is maximally mixed or, for ρ as above, that the k^{th} system is maximally entangled with the rest. We discuss applications of our result to quantum information theory in [6].

The condition that each ρ_i is proportional to the identity matrix generalizes the notion of "doubly stochastic", which arises in the study of the left-right action/operator scaling [14, 11]. We will refer to it as *"d-stochastic"*, which we define formally below.

▶ **Definition 5** (d-stochastic). A tensor $Y \in \mathrm{Ten}(n_0, \ldots, n_d)$ is called *d-stochastic* if the quantum marginals of $\rho = YY^\dagger / Y^\dagger Y$ of the last d tensor factors are given by $\rho_i = I_{n_i}/n_i$ for $i \in [d]$.

More explicitly, we can state the condition that $\rho_i = I_{n_i}/n_i$ as follows: We want that the slices of the tensor Y in the k^{th} direction are pairwise orthogonal and that their norm squares are equal to $1/n_i$. That is, for each $i \in [d]$ consider the order d tensors $Y^{(1)}, \ldots, Y^{(n_i)} \in \mathrm{Ten}(n_0, \ldots, n_{i-1}, n_{i+1}, \ldots, n_d)$ defined as

$$Y^{(j)}(j_0, \ldots, j_{i-1}, j_{i+1}, \ldots, j_d) := Y(j_0, \ldots, j_{i-1}, j, j_{i+1}, \ldots, j_d);$$

then the following should hold:

$$\langle Y^{(j)}, Y^{(j')} \rangle = \frac{1}{n_i} \delta_{j,j'}.$$

Here $\delta_{i,j}$ is the Kronecker delta function and the inner product is the Euclidean inner product of tensors.

Theorem 4 implies that X is *not* in the null cone if and only if we can find scalings Y whose last d quantum marginals are arbitrarily close to I_{n_i}/n_i. We can therefore rephrase the null-cone problem as *another, dual* optimization problem, where we seek to minimize the distance of the marginals to being maximally mixed. This is captured by the following definitions:

▶ **Definition 6** (Distance to d-stochasticity)**.** Let $Y \in \mathrm{Ten}(n_0, n_1, \ldots, n_d)$ be a tensor and $\rho = YY^\dagger / Y^\dagger Y$, with quantum marginals $\rho_1, \ldots, \rho_d$ on the last d systems. Then we define the *distance to d-stochasticity* $\mathrm{ds}(Y)$ as the (squared) distance between the marginals and d-stochastic marginals,

$$\mathrm{ds}(Y) := \sum_{i=1}^{d} \left\| \rho_i - \frac{I_{n_i}}{n_i} \right\|_F^2,$$

where $\|M\|_F := (\operatorname{tr} M^\dagger M)^{1/2}$ is the Frobenius norm. Following Eq. (4), we further define the *minimal distance to d-stochasticity* as

$$\mathrm{dds}(X) := \inf_{A \in G} \mathrm{ds}(A \cdot X).$$

Using this language, Theorem 4 states that X is in the null cone if and only if the minimal distance to d-stochasticity is nonzero. We summarize the duality between the two optimization problems:

$$X \text{ is in the null cone} \quad \Longleftrightarrow \quad \mathrm{cap}(X) = 0 \quad \Longleftrightarrow \quad \mathrm{dds}(X) > 0 \tag{8}$$

▶ Remark. According to Eq. (8), for any tensor X *exactly one* of the following two statements is true: X is in the null cone ($\mathrm{cap}(X) = 0$) or its orbit closure contains a d-stochastic tensor ($\mathrm{dds}(X) = 0$). Such dichotomies are well-known from the duality theory of linear programming (Farkas' lemma, Gordan's lemma, the duality between membership in a convex hull and the existence of separating hyperplanes, etc.). In the case of the (commutative) group of diagonal matrices, one recovers precisely these linear programming dualities from the general noncommutative framework.

3.1 The tensor scaling algorithm

If X is *not* in the null cone then there exist scalings Y such that $\mathrm{ds}(Y)$ is arbitrarily small. The main technical result of this paper is an algorithm that finds such scalings for any fixed $\varepsilon > 0$. It is a generalization of the results obtained for matrix and operator scaling in [30, 16, 14, 11]. Recall that we use n to denote the total number of coordinates of the tensor, $n = n_0 \cdots n_d$.

▶ **Theorem 7.** *Let $X \in \mathrm{Ten}(n_0, \ldots, n_d)$ be a (nonzero) tensor whose entries are integers of bitsize no more than b. Let $\ell = \min_{i \geq 1} n_i$ and suppose that $\varepsilon \leq d/(\max_{i \geq 1} n_i^2)$. Then Algorithm 1 with $T \geq \frac{18 \ln 2}{\ell \varepsilon} d\,(b + \log n)$ iterations either identifies that X is in the null cone or outputs a scaling $Y \in G \cdot X$ such that $\mathrm{ds}(Y) \leq \varepsilon$.*

To motivate Algorithm 1, note that, for every $i \in [d]$, we would like the quantum marginal of the i^{th} system to be proportional to the identity matrix. Suppose the quantum marginal on the i^{th} system is ρ_i. Then by acting on this system by any $A_i \in \mathrm{SL}(n_i)$ proportional to $\rho_i^{-1/2}$

Algorithm 1 Scaling algorithm for the null-cone problem.

Input: A tensor X in $\mathrm{Ten}(n_0, n_1, \ldots, n_d)$ with integer entries (specified as a list of entries, each encoded in binary, with bit size $\leq b$).

Output: Either the algorithm correctly identifies that X is in the null cone, or it outputs a scaling Y of X such that $\mathrm{ds}(Y) \leq \varepsilon$.

Algorithm:

1. If any of the quantum marginals of X is singular, then output that the tensor X is in the null cone and return. Otherwise set $Y^{(1)} = X$ and proceed to step 2.
2. For $t = 1, \ldots, T = \mathrm{poly}(n, b, 1/\varepsilon)$, repeat the following:
   - Compute the quantum marginals ρ_i of $Y^{(t)}Y^{(t),\dagger}/Y^{(t),\dagger}Y^{(t)}$ and find the index $i \in [d]$ for which $\left\|\rho_i - \frac{I_{n_i}}{n_i}\right\|_F^2$ is largest. If $\left\|\rho_i - \frac{I_{n_i}}{n_i}\right\|_F^2 < \varepsilon/d$, output $Y^{(t)}$ and return.
   - Otherwise, set $Y^{(t+1)} \leftarrow \det(\rho_i)^{1/2n_i}\rho_i^{-1/2} \cdot Y^{(t)}$.
3. Output that the tensor X is in the null cone.

we can precisely arrange for the quantum marginal on i^{th} system to be proportional to the identity matrix[15]. (But this might disturb the quantum marginals on the other systems!) An appropriate choice of A_i that ensures that it has determinant one is $A_i = \det(\rho_i)^{1/2n_i}\rho_i^{-1/2}$. Thus we find that the alternating minimization heuristics in Eq. (7) that we previously derived for $\mathrm{cap}(X)$ is equally natural from the perspective of the dual problem of minimizing $\mathrm{dds}(X)$! Remarkably, iterating this operation in an appropriate order of subsystems i does ensure that *all* the quantum marginals get arbitrarily close to the normalized identity matrices – provided that X is not in the null cone. (If X is in the null cone, then it will lead to a sequence of scalings of norm arbitrarily close to zero, certifying that X is in the null cone.)

▶ Remark (Rational Entries). Notice that we required our input tensor X to have integer entries. In general, an input tensor X could have rational entries. If this is the case, we can simply multiply X by the denominators to obtain an integral tensor X', on which we can then apply our algorithm above. Since multiplying a tensor by a nonzero number does not change the property of being in the null cone, our algorithm will still be correct. The following remarks discuss the changes in bit complexity of our algorithm. In this case, the size of X' is at most $b \cdot n$, and and therefore the bound for the number of iterations T is still $\mathrm{poly}(n, b, 1/\varepsilon)$, as we wanted.

When analyzing bit complexity, Algorithm 1 is not directly applicable, since it computes inverse square roots of matrices. Even if the entries of the square roots were always rational, the algorithm would also iteratively multiply rationals with very high bit complexity. We can handle these numerical issues by truncating the entries of the scalings $Y^{(t)}$ of X, as well as the quantum marginals ρ_k, to $\mathrm{poly}(n, b, \frac{1}{\varepsilon})$ many bits after the decimal point. Then, in a similar way as done in [14, 11], we can prove that there is essentially no change in the convergence required in the number of iterations T. Since each arithmetic operation will be done with numbers of $\mathrm{poly}(n, b, \frac{1}{\varepsilon})$ many bits, this truncation ensures that we run in polynomial time. As a consequence, we obtain our main theorem, which we had already announced in the introduction:

▶ **Theorem 1** (Main theorem). *There is a* $\mathrm{poly}(n, b, \frac{1}{\varepsilon})$ *time deterministic algorithm (Algorithm 1) that, given a tensor* $X \in V$ *with integer coordinates of bit size bounded by* b*, either identifies that* X *is in the null cone or outputs a "scaling"* $Y \in G \cdot X$ *such that* $\mathrm{ds}(Y) < \varepsilon$.

[15] It is not hard to see that if X is not in the null cone then ρ_i is invertible [6]

We remark that for $d = 2$, our algorithm in essence reduces to the algorithm of [16, 14, 15, 11]; for $n_0 = 1$, it refines the algorithm proposed previously in [41] without a complexity analysis. Indeed, Theorem 1 is a generalization of the main theorem of [11] on operator scaling (or the left-right action). There it was enough to take $\varepsilon = 1/n$ to be sure that the starting tensor is not in the null cone.

In our much more general setting it is still true that there exists some $\varepsilon = \varepsilon(n) > 0$ such that $\mathrm{ds}(Y) > \varepsilon$ for any Y in the null cone. Unfortunately, in our general setting, ε will be exponentially small (see Theorem 16 in the next section). Therefore, Theorem 1 with this choice of ε gives an exponential time algorithm for deciding the null-cone problem:

▶ **Theorem 8** (Null-cone problem). *There is a* $\mathrm{poly}(b)\exp(O(n\log n))$ *time deterministic algorithm (Algorithm 1 with* $\varepsilon = \exp(-O(n\log n))$*) that, given a tensor* $X \in \mathrm{Ten}(n_0, n_1, \ldots, n_d)$ *with integer coordinates of bit size at most* b*, decides whether* X *is in the null cone.*

Thus our algorithm does not solve the null-cone problem in polynomial time for general tensor actions, and this remains an excellent open question!

Nevertheless, Theorem 1 as such, with its promise that $\mathrm{ds}(Y) < \varepsilon$, is of interest beyond merely solving the null-cone problem. It has applications to quantitative notions of instability in geometric invariant theory, to a form of multipartite entanglement distillation in quantum information theory, and to questions about the slice rank of tensors, which underpinned recent breakthroughs in additive combinatorics. We discuss these in Section 4 and [6].

3.2 Analysis sketch

To analyze our algorithm and prove Theorem 7, we follow a three-step argument similar to the analysis of the algorithms for matrix scaling and operator scaling in [30, 14, 11] once we have identified the appropriate potential function and distance measure. In the following we sketch the main ideas and we refer to [6] for all technical details.

As potential function, we will use the *norm squared* of the tensor, denoted $f(Y) = \|Y\|_2^2$, and the distance measure is $\mathrm{ds}(Y)$ defined above in Theorem 6. Note that these the two functions are exactly dual to each other in our noncommutative optimization framework (see Eqs. (3), (4) and (8))!

The following two properties of the capacity will be crucial for the analysis:

A. $\mathrm{cap}(X) \geq 1/(n_1 \cdots n_d)^2$ if $X \in \mathrm{Ten}(n_0, \ldots, n_d)$ is a tensor with integral entries that is not in the null cone ([6]).

B. $\mathrm{cap}(X)$ is invariant under the action of the group $\mathrm{SL}(n_1) \times \cdots \times \mathrm{SL}(n_d)$.

Using the above properties, a three-step analysis follows the following outline:

1. An upper bound on $f(Y^{(1)}) = f(X)$ from the input size.

2. As long as $\mathrm{ds}(Y) \geq \varepsilon$, the norm squared decreases by a factor $2^{-\frac{\ell\varepsilon}{6d\ln 2}}$ in each iteration: $f(Y^{(t+1)}) \leq 2^{-\frac{\ell\varepsilon}{6d\ln 2}} f(Y^{(t)})$, where we recall that $\ell = \min_{i\geq 1} n_i$.

3. A lower bound on $f(Y^{(t)})$ for all t. This follows from properties A and B above.

The above three steps imply that we must achieve $\mathrm{ds}(Y^{(t)}) < \varepsilon$ for some $t \in [T]$.

Step 1 is immediate and step 2 follows from the G-invariance of the capacity (property B) and a quantative form the AM-GM inequality.

Step 3, or rather the lower bound on $\mathrm{cap}(X)$ (property A) is the technically most challenging part of the proof. It is achieved by quantifying the basic argument given on p. 9. The main tool required to carry out this analysis is the existence of a set of invariant polynomials with "small" integer coefficients that spans the space of invariant polynomials. We do so by giving an "explicit" construction of such a set via Schur-Weyl

duality [6]. Remarkably, we do not need to use the known exponential bounds on the degree of generating invariants in [10].

We give an alternative lower bound on $\mathrm{cap}(X)$ using Cayley's Omega-process in [6]. This proof is more general (it applies to arbitrary actions of product of SL's) but gives a weaker bound (although sufficient to yield a polynomial time analysis for the algorithm). Here we do need to rely on recent exponential bounds on the degree of generating invariants by Derksen.

The size of the integer coefficients of a spanning set of invariant polynomials is an interesting invariant theoretic quantity that appears to not have been studied in the invariant theory literature. It is crucial for our analysis, and we believe it could be of independent interest in computational invariant theory.

4 Hilbert-Mumford criterion and quantifying instability

In this section, we explain a special case of the Hilbert-Mumford criterion, which allows us to characterize tensors that lie in the null cone in terms of simple, one-dimensional subgroups of our group $\mathrm{SL}(n_1) \times \cdots \times \mathrm{SL}(n_d)$. Moreover, we define the instability (Theorem 11), which quantifies how fast a tensor in the null cone can be sent to the zero vector $0 \in V$ by the group action. The instability is an important quantity in invariant theory and we present an algorithm for the $(0, \varepsilon)$-gap problem (Theorem 15).

We again consider briefly the general setting of a group G acting on a vector space V. We know from Section 2 that a vector is in the null cone if and only if its orbit closure contains the zero vector, $0 \in \overline{G \cdot v}$. The *Hilbert-Mumford criterion* [19, 34] shows that it suffices to consider certain one-dimensional (and hence in particular commutative) subgroups of G. More precisely, recall that a *one-parameter subgroup* (*1-PSG*) of G is an algebraic group homomorphism $\lambda : \mathbb{C}^* \to G$, i.e., an algebraic map that satisfies $\lambda(zw) = \lambda(z)\lambda(w)$. (For example, any 1-PSG of $\mathbb{C}^*$ is of the form $z \mapsto z^k$ for some integer k.) Then the Hilbert-Mumford criterion states that a vector v is in the null cone if and only if there exists a 1-PSG λ of G such that $\lim_{z \to 0} \lambda(z) \cdot v = 0$.

A familiar example appears when the group $G = \mathrm{SL}(n)$ acts by left multiplication on a single matrix X. In this case, the null cone consists of the singular matrices and the Hilbert-Mumford criterion tells us that if X is singular, then there is a 1-PSG given by $\lambda(z) = B^{-1} \operatorname{diag}(z^{a_1}, \ldots, z^{a_n}) B$ which drives X to zero. It is easy to see that in this case such one-parameter subgroup can be found by taking B to be a matrix in $\mathrm{SL}(n)$ which makes X upper triangular (through row eliminations and permutations) with its last row being all zeros, and that the exponents a_i can be taken such that $a_1 = \cdots = a_{n-1} = 1$ and $a_n = 1 - n$.

We now return to our setup, where $G = \mathrm{SL}(n_1) \times \cdots \times \mathrm{SL}(n_d)$. Here any 1-PSG λ is of the following form:

$$\lambda(z) = \left(B_1^{-1} \operatorname{diag}(z^{a_{1,1}}, \ldots, z^{a_{1,n_1}}) B_1, \ldots, B_d^{-1} \operatorname{diag}(z^{a_{d,1}}, \ldots, z^{a_{d,n_d}}) B_d \right), \tag{9}$$

where $B_1, \ldots, B_d$ are invertible matrices (B_i of dimension $n_i \times n_i$) and $(a_{i,j})$ is a tuple of integers in

$$\Gamma := \left\{ (a_{i,j})_{i \in [d], j \in [n_i]} \;\middle|\; a_{i,j} \in \mathbb{Z}, \sum_{j=1}^{n_i} a_{i,j} = 0 \text{ for } i \in [d] \right\}. \tag{10}$$

Intuitively, the matrices B_i are a change of basis after which the action of G becomes reduced to an action of diagonal groups (similar to actions arising in the study of the classical matrix or tensor scalings).

We want to understand what it means for a 1-PSG λ and a tensor X that $\lim_{z\to 0} \lambda(z)\cdot X = 0$. For this, we write the tensor in the basis corresponding to $B = (B_1, \ldots, B_d)$, i.e., $Y = B\cdot X$. We define the *support* of Y as

$$\operatorname{supp}(Y) := \left\{(j_1, \ldots, j_d) \in [n_1] \times \cdots \times [n_d] \;\middle|\; \exists j_0 \text{ s.t. } Y_{j_0, j_1, \ldots, j_d} \neq 0\right\}.$$

Thus $(j_1, \ldots, j_d)$ is in the support of Y iff at least one of the slices $Y^{(i)}$ in the 0^{th} direction of the tensor Y has a nonzero entry at this position. Now note that

$$(B \cdot \lambda(z) \cdot X)_{j_0, j_1 \ldots, j_d} = z^{\sum_{i=1}^{d} a_{i,j_i}} Y_{j_0, j_1, \ldots, j_d}. \tag{11}$$

It follows that $\lim_{z\to 0} \lambda(z)\cdot X = 0$ is equivalent to the support $\operatorname{supp}(Y)$ being deficient in the following sense:

▶ **Definition 9** (Deficiency)**.** We call a subset $S \subseteq [n_1] \times \cdots \times [n_d]$ *deficient* if there exists $(a_{i,j}) \in \Gamma$ such that $\forall\, (j_1, \ldots, j_d) \in S \quad \sum_{i=1}^{d} a_{i,j_i} > 0$.

We note that in the case $d = 2$, a subset $S \subseteq [n] \times [n]$ is deficient if and only if the bipartite graph corresponding to S does not admit a perfect matching, as can be proved via Hall's matching theorem. For $d > 2$, we do not know of such a clean combinatorial characterization, although the characterization above is by a linear program, and therefore is efficiently testable.

In the Hilbert-Mumford criterion, one can do slightly better, namely restrict attention to the one-parameter subgroups compatible with a maximally compact subgroup of the group G. What this means in our case is that we will be able to take the matrices $B_1, \ldots, B_d$ to be unitary matrices (see Theorem 12)! We can thus summarize the statement of the Hilbert-Mumford criterion as follows:

▶ **Proposition 10** (Special case of [19, 34])**.** *A tensor $X \in \operatorname{Ten}(n_0, \ldots, n_d)$ is in the null cone of the natural action of the group $G = \operatorname{SL}(n_1) \times \cdots \times \operatorname{SL}(n_d)$ iff there exist unitary $n_i \times n_i$ matrices U_i, $i \in [d]$, such that the support $\operatorname{supp}(Y)$ of the tensor $Y = (U_1, \ldots, U_d)\cdot X$ is deficient.*

For completeness, we provide a proof of the criterion in [6].

▶ Remark. Deficiency can also be understood in terms of the null cone of the action of the subgroup $T \subseteq \operatorname{SL}(n_1) \times \cdots \times \operatorname{SL}(n_d)$ formed by tuples of diagonal matrices. Indeed, when we fixed B but varied the $a_{i,j}$, we were precisely varying over the 1-PSGs of T. Thus, a tensor $Y \in \operatorname{Ten}(n_1, \ldots, n_d)$ is in the null cone for the action of the diagonal subgroup if and only if $\operatorname{supp}(Y)$ is deficient. (This follows from the Hilbert-Mumford criterion for the commutative group T, or directly from Farkas' lemma.) Thus Proposition 10 is a special case of a general reduction principle in geometric invariant theory, from actions of non-commutative groups to commutative groups up to a basis change.

In geometric invariant theory [34], vectors in the null cone are also referred to as *unstable*. The Hilbert-Mumford criterion suggests a natural way of quantifying the instability of a vector: instead of merely asking whether there exists a 1-PSG that sends the vector to zero, we may measure the (suitably normalized) rate at which the vector is sent to zero in Eq. (11). This rate, also known as *Mumford's numerical function*, takes the following form for the tensor action that we are studying:

▶ **Definition 11** (Deficiency, instability)**.** Given a set $S \subseteq [n_1] \times \cdots \times [n_d]$, we define its *deficiency* as

$$\operatorname{def}(S) := \max_{(a_{i,j}) \in \Gamma} \frac{\min_{(j_1, \ldots, j_d) \in S} \left(\sum_{i=1}^{d} a_{i,j_i}\right)}{\sqrt{\sum_{i=1}^{d} \sum_{j=1}^{n_i} a_{i,j}^2}}.$$

where we recall that the set Γ was defined in Eq. (10). We then define the *instability* of a tensor $X \in \mathrm{Ten}(n_0, \dots, n_d)$ by

$$\mathrm{instability}(X) := \max_{\substack{U=(U_1,\dots,U_d) \text{ tuple} \\ \text{of unitary matrices}}} \mathrm{def}(\mathrm{supp}(U \cdot X)),$$

The instability can also be defined using general invertible matrices:

▶ **Lemma 12.** *For any $X \in \mathrm{Ten}(n_0, \dots, n_d)$, we have that*

$$\mathrm{instability}(X) = \max_{\substack{B=(B_1,\dots,B_d) \text{ tuple} \\ \text{of invertible matrices}}} \mathrm{def}(\mathrm{supp}(B \cdot X)).$$

In particular, $X \mapsto \mathrm{instability}(X)$ is a G-invariant function.

In our case, this follows from the QR decomposition, and we give a succinct proof in [6]. Clearly, a subset S is deficient if and only if its deficiency is positive, $\mathrm{def}(S) > 0$. And by the Hilbert-Mumford criterion, a tensor X is in the null cone if and only if its instability is positive, $\mathrm{instability}(X) > 0$. This suggests a natural relaxation of the null-cone problem:

▶ **Problem 13.** *For $\varepsilon > 0$, the problem* ε-instability *is defined as follows: Given a tensor $X \in \mathrm{Ten}(n_0, \dots, n_d)$ with integer coordinates such that either*
1. *X is not in the null cone (i.e., $\mathrm{instability}(X) \leq 0$), or*
2. *$\mathrm{instability}(X) \geq \varepsilon$,*

decide which is the case.

We will now show that the ε-instability problem can be solved as a consequence of our main Theorem 1. Importantly, the instability of a tensor is tightly connected to the distance from d-stochasticity, as defined in Theorem 6.

▶ **Lemma 14** (Special case of [35, Lemma 3.1, (iv)])**.** *For all tensors $X \in \mathrm{Ten}(n_0, \dots, n_d)$, $\mathrm{instability}(X) \leq \sqrt{\mathrm{dds}(X)}$.*

In fact, the inequality in Theorem 14 is tight for tensors in the null cone (e.g., [13, Corollary 11.2]), but we will not need this here (if X is not in the null cone then $\mathrm{instability}(X) \leq 0$ while $\mathrm{ds}(X) = 0$). We obtain the following corollary of Theorem 1.

▶ **Theorem 15.** *There is $\mathrm{poly}\left(n, b, \frac{1}{\varepsilon}\right)$ time deterministic algorithm (Algorithm 1) for the ε-instability problem (Problem 13) for tensors with entries of bit size bounded by b.*

The algorithm is obtained by running Algorithm 1 and outputting "X is not in the null cone" if Algorithm 1 produces a scaling Y with $\mathrm{ds}(Y) < \varepsilon$ and otherwise outputting "$\mathrm{instability}(X) \geq \varepsilon$".

Thus ε-instability can be decided efficiently if $\varepsilon = \Omega(1/\mathrm{poly}(n))$. Unfortunately, there may well exist tensors X which are in the null cone and whose instability is exponentially small. However, the instability cannot be worse than exponentially small, as we prove in [6]:

▶ **Lemma 16.** *Suppose a tensor $X \in \mathrm{Ten}(n_0, \dots, n_d)$ is in the null cone. Then $\sqrt{\mathrm{dds}(X)} \geq \mathrm{instability}(X) = \exp(-O(n \log n))$, where we recall that $n = n_0 \cdots n_d$.*

This begs the question: are there natural examples where we can apply the above algorithm in Theorem 15? One set of examples comes from the notion of slice rank of tensors – discovered and widely used in the recent breakthroughs in additive combinatorics regarding cap sets and other combinatorial objects (see, e.g., [3]). We discuss this in [6].

5 Conclusion and open problems

This paper continues the recent line of works studying the computational complexity of problems in invariant theory [31, 11, 21, 22, 20, 5, 32, 1]. There are many beautiful algorithms known in invariant theory [9, 39], but most of them come without a complexity analysis or will have at least exponential runtime. Designing efficient algorithms for problems in invariant theory is important for the GCT program [31, 32], and we believe that viewing invariant theory from the computational lens will provide significant new structural insights. Several open problems arise from this work. We mention the most interesting ones:

1. Is there a polynomial time algorithm for deciding the null-cone problem for the tensor actions that we study? It would suffice to give an analog of the algorithm in Theorem 1 with running time $\mathrm{poly}(n, b, \log(1/\varepsilon))$. One might wonder if known optimization techniques are sufficient to yield this runtime guarantee. While the optimization problem in Eq. (3) is not convex, it is nevertheless known to be *geodesically* convex [24, 35, 45] (roughly speaking one needs to move from the usual Euclidean geometry to the geometry of the group to witness convexity). The theory of geodesically convex optimization is just starting to emerge (see, e.g., [47] and the references therein) and it is an open problem whether there are computationally efficient analogs of the ellipsoid algorithm and interior point methods (algorithms that in Euclidean convex optimization guarantee $\mathrm{poly}(\log(1/\varepsilon))$ runtime).
2. Is there a polynomial time algorithm for the null-cone problem for more general group actions? As mentioned before, there is a natural notion of *"G-stochasticity"* using moment maps provided by the noncommutative duality theory. A first important goal is to design an algorithm that achieves the analog of our Theorem 1 (i.e., getting to a point in the orbit where the "distance" to satisfying G-stochasticity is at most ε). While it is not clear how far the alternating minimization approach can be taken, Kirwan's gradient flow [25] seems to be a promising alternative first proposed in [43, 42].
3. Is there a polynomial time algorithm for the one-body quantum marginal problems described in [6]? There is a natural scaling algorithm generalizing Algorithm 1 but it has so far evaded analysis. Even obtaining a polynomial time algorithm for a promise problem along the lines of Theorem 1 would be rather interesting.

Acknowledgements. We would like to thank Henry Cohn, Eric Naslund, Ion Nechita, Will Sawin, Suvrit Sra, Josué Tonelli, and John Wright for helpful discussions.

References

1 Velleda Baldoni, Michèle Vergne, and Michael Walter. Computation of dilated Kronecker coefficients. *Journal of Symbolic Computation*, 84:113–146, 2018.

2 Charles H Bennett, Sandu Popescu, Daniel Rohrlich, John A Smolin, and Ashish V Thapliyal. Exact and asymptotic measures of multipartite pure-state entanglement. *Physical Review A*, 63(1):012307, 2000.

3 Jonah Blasiak, Thomas Church, Henry Cohn, Joshua A. Grochow, Eric Naslund, William F Sawin, and Chris Umans. On cap sets and the group-theoretic approach to matrix multiplication. *Discrete Analysis*, 2017. doi:10.19086/da.1245.

4 Herm Brascamp and Elliot Lieb. Best constants in Young's inequality, its converse and its generalization to more than three functions. *Advances in Mathematics*, 20:151–172, 1976.

5 Peter Bürgisser, Matthias Christandl, Ketan D Mulmuley, and Michael Walter. Membership in moment polytopes is in NP and coNP. *SIAM Journal on Computing*, 46(3):972–991, 2017.

6 Peter Bürgisser, Ankit Garg, Rafael Mendes de Oliveira, Michael Walter, and Avi Wigderson. Alternating minimization, scaling algorithms, and the null-cone problem from invariant theory. *CoRR*, abs/1711.08039, 2017. arXiv:1711.08039.

7 Arthur Cayley. On linear transformations. *Cambridge and Dublin Mathematical Journal*, 1:104–122, 1846.

8 Imre Csiszár and Gábor Tusnády. Information geometry and alternating minimization procedures. *Stat. Decis. Suppl.*, 1:205–237, 1984.

9 H. Derksen and G. Kemper. *Computational Invariant Theory*, volume 130. Springer-Verlag, Berlin, 2002.

10 Harm Derksen. Polynomial bounds for rings of invariants. *Proceedings of the American Mathematical Society*, 129(4):955–963, 2001.

11 Ankit Garg, Leonid Gurvits, Rafael Mendes de Oliveira, and Avi Wigderson. A deterministic polynomial time algorithm for non-commutative rational identity testing. *CoRR*, abs/1511.03730, 2015. arXiv:1511.03730.

12 Ankit Garg, Leonid Gurvits, Rafael Mendes de Oliveira, and Avi Wigderson. Algorithmic aspects of brascamp-lieb inequalities. *CoRR*, abs/1607.06711, 2016. arXiv:1607.06711.

13 Valentina Georgoulas, Joel W Robbin, and Dietmar A Salamon. The moment-weight inequality and the Hilbert-Mumford criterion. *math*, abs/1311.0410, 2013. arXiv:1311.0410.

14 Leonid Gurvits. Classical complexity and quantum entanglement. *Journal of Computer and System Sciences*, 69(3):448–484, 2004.

15 Leonid Gurvits. Hyperbolic polynomials approach to van der waerden/schrijver-valiant like conjectures: sharper bounds, simpler proofs and algorithmic applications. *STOC*, pages 417–426, 2006.

16 Leonid Gurvits and Peter N. Yianilos. The deflation-inflation method for certain semidefinite programming and maximum determinant completion problems. *Technical Report, NECI*, 1998.

17 Moritz Hardt. Understanding alternating minimization for matrix completion. *FOCS*, pages 651–660, 2014.

18 David Hilbert. Ueber die Theorie der algebraischen Formen. *Mathematische Annalen*, 36(4):473–534, 1890.

19 David Hilbert. Uber die vollen Invariantensysteme. *Math. Ann.*, 42:313–370, 1893.

20 Christian Ikenmeyer, Ketan D. Mulmuley, and Michael Walter. On vanishing of Kronecker coefficients. *Computational Complexity*, 2017.

21 Gabor Ivanyos, Youming Qiao, and K. V. Subrahmanyam. Non-commutative Edmonds' problem and matrix semi-invariants. *Computational Complexity*, 2016.

22 Gábor Ivanyos, Youming Qiao, and K. V. Subrahmanyam. Constructive noncommutative rank computation in deterministic polynomial time over fields of arbitrary characteristics. *ITCS*, 2017.

23 Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. *STOC*, pages 665–674, 2013.

24 George Kempf and Linda Ness. The length of vectors in representation spaces. In *Algebraic Geometry*, volume 732 of *Lecture Notes in Math.*, pages 233–243. Springer, 1979. doi:10.1007/BFb0066647.

25 Frances Kirwan. *Cohomology of quotients in symplectic and algebraic geometry*, volume 31 of *Mathematical Notes*. Princeton University Press, 1984.

26 Alexander Klyachko. Coherent states, entanglement, and geometric invariant theory. *Quantum Physics*, abs/quant-ph/0206012, 2002. arXiv:quant-ph/0206012.

27 Alexander A. Klyachko. Quantum marginal problem and n-representability. *Journal of Physics: Conference Series*, 36(1):72, 2006. doi:10.1088/1742-6596/36/1/014.

28 Carlton Lemke and J. T. Howson. Equilibrium points of bimatrix games. *SIAM Journal on Applied Mathematics*, 12(2):413–423, 1964.

29 Elliot Lieb. Gaussian kernels have only Gaussian maximizers. *Inventiones Mathematicae*, 102:179–208, 1990.

30 Nati Linial, Alex Samorodnitsky, and Avi Wigderson. A deterministic strongly polynomial algorithm for matrix scaling and approximate permanents. *STOC*, pages 644–652, 1998.

31 Ketan Mulmuley. Geometric Complexity Theory V: Equivalence between Blackbox Derandomization of Polynomial Identity Testing and Derandomization of Noether's Normalization Lemma. *FOCS*, pages 629–638, 2012.

32 Ketan D. Mulmuley. Geometric Complexity Theory V: Efficient algorithms for Noether Normalization. *J. Amer. Math. Soc.*, 30(1):225–309, 2017. doi:10.1090/jams/864.

33 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.

34 David Mumford. *Geometric invariant theory*. Springer-Verlag, Berlin-New York, 1965.

35 Linda Ness and David Mumford. A stratification of the null cone via the moment map. *American Journal of Mathematics*, 106(6):1281–1329, 1984. URL: http://www.jstor.org/stable/2374395.

36 John von Neumann. *Functional Operators, Vol. II: The Geometry of Orthogonal Spaces*, volume 22 of *Annals of Math. Studies*. Princeton University Press, 1950.

37 Meisam Razaviyayn, Mingyi Hong, and Zhi-Quan Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, 23(2):1126–1153, 2013.

38 R. Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The Annals of Mathematical Statistics*, 35:876–879, 1964.

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

40 B. Sury. An elementary proof of the Hilbert-Mumford criterion. *Electron. J. Linear Algebra*, 7:174–177, 2000. doi:10.13001/1081-3810.1053.

41 Frank Verstraete, Jeroen Dehaene, and De Moor Bart. Normal forms and entanglement measures for multipartite quantum states. *Physical Review A*, 68(1), 2003.

42 Michael Walter. *Multipartite Quantum States and their Marginals*. PhD thesis, ETH Zurich, 2014.

43 Michael Walter, Brent Doran, David Gross, and Matthias Christandl. Entanglement polytopes: multiparticle entanglement from single-particle information. *Science*, 340(6137):1205–1208, 2013.

44 Yu Wang, Wotao Yin, and Jinshan Zeng. Global convergence of ADMM in nonconvex nonsmooth optimization. *math*, abs/1511.06324, 2015. arXiv:1511.06324.

45 Chris Woodward. Moment maps and geometric invariant theory. *math*, abs/0912.1132, 2011. arXiv:0912.1132.

46 Yangyang Xu and Wotao Yin. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on Imaging Sciences*, 6(3):1758–1789, 2013.

47 Hongyi Zhang and Suvrit Sra. First-order methods for geodesically convex optimization. *math*, abs/1602.06053, 2016. arXiv:1602.06053.