Most Tensor Problems Are NP-Hard

CHRISTOPHER J. HILLAR, Mathematical Sciences Research Institute
LEK-HENG LIM, University of Chicago

We prove that multilinear (tensor) analogues of many efficiently computable problems in numerical linear algebra are NP-hard. Our list includes: determining the feasibility of a system of bilinear equations, deciding whether a 3-tensor possesses a given eigenvalue, singular value, or spectral norm; approximating an eigenvalue, eigenvector, singular vector, or the spectral norm; and determining the rank or best rank-1 approximation of a 3-tensor. Furthermore, we show that restricting these problems to symmetric tensors does not alleviate their NP-hardness. We also explain how deciding nonnegative definiteness of a symmetric 4-tensor is NP-hard and how computing the combinatorial hyperdeterminant is NP-, #P-, and VNP-hard.

Categories and Subject Descriptors: G.1.3 [**Numerical Analysis**]: Numerical Linear Algebra

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Numerical multilinear algebra, tensor rank, tensor eigenvalue, tensor singular value, tensor spectral norm, system of multilinear equations, hyperdeterminants, symmetric tensors, nonnegative definite tensors, bivariate matrix polynomials, NP-hardness, #P-hardness, VNP-hardness, undecidability, polynomial time approximation schemes

ACM Reference Format:
Hillar, C. J. and Lim, L.-H. 2013. Most tensor problems are NP-hard. *J. ACM* 60, 6, Article 45 (November 2013), 39 pages.
DOI:http://dx.doi.org/10.1145/2512329

1. INTRODUCTION

Frequently a problem in science or engineering can be reduced to solving a linear (matrix) system of equations and inequalities. Other times, solutions involve the extraction of certain quantities from matrices such as eigenvectors or singular values. In computer vision, for instance, segmentations of a digital picture along object boundaries can be found by computing the top eigenvectors of a certain matrix produced from the image [Shi and Malik 2000]. Another common problem formulation is to find low-rank matrix approximations that explain a given two-dimensional array of data, accomplished, as is now standard, by zeroing the smallest singular values in a singular value decomposition of the array [Golub and Kahan 1965; Golub and Reinsch 1970]. In

C. J. Hillar was partially supported by an NSA Young Investigators Grant and an NSF All-Institutes Postdoctoral Fellowship administered by the Mathematical Sciences Research Institute through its core grant DMS-0441170. L.-H. Lim was partially supported by an NSF CAREER Award DMS-1057064, and NSF Collaborative Research Grant DMS 1209136, and an AFOSR Young Investigator Award FA9550-13-1-0133.
Authors' addresses: C. J. Hillar, Mathematical Sciences Research Institute, Berkeley, CA 94720; email: chillar@msri.org; L.-H. Lim, Computational and Applied Mathematics Initiative, Department of Statistics, University of Chicago, Chicago, IL 60637; email: lekheng@galton.uchicago.edu.

DOI:http://dx.doi.org/10.1145/2512329

general, efficient and reliable routines computing answers to these and similar problems have been a workhorse for real-world applications of computation.

Recently, there has been a flurry of work on multilinear analogues to the basic problems of linear algebra. These "tensor methods" have found applications in many fields, including approximation algorithms [Brubaker and Vempala 2009; De La Vega et al. 2005], computational biology [Cartwright et al. 2009], computer graphics [Vasilescu and Terzopoulos 2004], computer vision [Shashua and Hazan 2005; Vasilescu and Terzopoulos 2002], data analysis [Coppi and Bolasco 1989], graph theory [Friedman 1991; Friedman and Wigderson 1995], neuroimaging [Schultz and Seidel 2008], pattern recognition [Vasilescu 2002], phylogenetics [Allman and Rhodes 2008], quantum computing [Miyake and Wadati 2002], scientific computing [Beylkin and Mohlenkamp 2002], signal processing [Comon 1994, 2004; Kofidis and Regalia 2001/02], spectroscopy [Smilde et al. 2004], and wireless communication [Sidiropoulos et al. 2000], among other areas. Thus, tensor generalizations to the standard algorithms of linear algebra have the potential to substantially enlarge the arsenal of core tools in numerical computation.

The main results of this article, however, support the view that tensor problems are almost invariably computationally hard. Indeed, we shall prove that many naturally occurring problems for 3-tensors are NP-hard; that is, solutions to the hardest problems in NP can be found by answering questions about 3-tensors. A full list of the problems we study can be found in Table I. Since we deal with mathematical questions over fields (such as the real numbers $\mathbb{R}$), algorithmic complexity is a somewhat subtle notion. Our perspective here will be the Turing model of computation [Turing 1936] and the Cook–Karp–Levin model of complexity involving NP-hard [Knuth 1974a, 1974b] and NP-complete problems [Cook 1971; Karp 1972; Levin 1973], as opposed to other computational models [Blum et al. 1989; Valiant 1979a; Weihrauch 2000]. We describe our framework in Section 1.3 along with a comparison to other models.

One way to interpret these findings is that 3-tensor problems form a boundary separating classes of tractable linear/convex problems from intractable nonlinear/nonconvex ones. More specifically, linear algebra is concerned with (inverting) vector-valued functions that are locally of the form $f(\mathbf{x}) = \mathbf{b} + A\mathbf{x}$; while convex analysis deals with (minimizing) scalar-valued functions that are locally approximated by $f(\mathbf{x}) = c + \mathbf{b}^\top\mathbf{x} + \mathbf{x}^\top A\mathbf{x}$ with A positive definite. These functions involve tensors of order 0, 1, and 2: $c \in \mathbb{R}$, $\mathbf{b} \in \mathbb{R}^n$, and $A \in \mathbb{R}^{n\times n}$. However, as soon as we move on to bilinear vector-valued or trilinear real-valued functions, we invariably come upon 3-tensors $\mathcal{A} \in \mathbb{R}^{n\times n\times n}$ and the NP-hardness associated with inferring properties of them.

The primary audience for this article are numerical analysts and computational algebraists, although we hope it will be of interest to users of tensor methods in various communities. Parts of our exposition contain standard material (e.g., complexity theory to computer scientists, hyperdeterminants to algebraic geometers, KKT conditions to optimization theorists, etc.), but to appeal to the widest possible audience at the intersection of computer science, linear and multilinear algebra, algebraic geometry, numerical analysis, and optimization, we have keep our discussion as self-contained as possible. A side contribution is a useful framework for incorporating features of computation over $\mathbb{R}$ and $\mathbb{C}$ with classical tools and models of algorithmic complexity involving Turing machines that we think is unlike any existing treatments [Blum et al. 1989, 1998; Hochbaum and Shanthikumar 1990; Vavasis 1991].

1.1. Tensors

We begin by first defining our basic mathematical objects. Fix a field $\mathbb{F}$, which for us will be either the rationals $\mathbb{Q}$, the reals $\mathbb{R}$, or the complex numbers $\mathbb{C}$. Also, let l, m,

Table I. Tractability of Tensor Problems

Problem	**Complexity**
Bivariate Matrix Functions over $\mathbb{R}$, $\mathbb{C}$	Undecidable (Proposition 12.2)
Bilinear System over $\mathbb{R}$, $\mathbb{C}$	NP-hard (Theorems 2.6, 3.7, 3.8)
Eigenvalue over $\mathbb{R}$	NP-hard (Theorem 1.3)
Approximating Eigenvector over $\mathbb{R}$	NP-hard (Theorem 1.5)
Symmetric Eigenvalue over $\mathbb{R}$	NP-hard (Theorem 9.3)
Approximating Symmetric Eigenvalue over $\mathbb{R}$	NP-hard (Theorem 9.6)
Singular Value over $\mathbb{R}$, $\mathbb{C}$	NP-hard (Theorem 1.7)
Symmetric Singular Value over $\mathbb{R}$	NP-hard (Theorem 10.2)
Approximating Singular Vector over $\mathbb{R}$, $\mathbb{C}$	NP-hard (Theorem 6.3)
Spectral Norm over $\mathbb{R}$	NP-hard (Theorem 1.10)
Symmetric Spectral Norm over $\mathbb{R}$	NP-hard (Theorem 10.2)
Approximating Spectral Norm over $\mathbb{R}$	NP-hard (Theorem 1.11)
Nonnegative Definiteness	NP-hard (Theorem 11.2)
Best Rank-1 Approximation over $\mathbb{R}$	NP-hard (Theorem 1.13)
Best Symmetric Rank-1 Approximation over $\mathbb{R}$	NP-hard (Theorem 10.2)
Rank over $\mathbb{R}$ or $\mathbb{C}$	NP-hard (Theorem 8.2)
Enumerating Eigenvectors over $\mathbb{R}$	#P-hard (Corollary 1.16)
Combinatorial Hyperdeterminant	NP-, #P-, VNP-hard (Theorems 4.1 , 4.2, Corollary 4.3)
Geometric Hyperdeterminant	Conjectures 1.9, 13.1
Symmetric Rank	Conjecture 13.2
Bilinear Programming	Conjecture 13.4
Bilinear Least Squares	Conjecture 13.5

Note: Except for positive definiteness and the combinatorial hyperdeterminant, which apply to 4-tensors, all problems refer to the 3-tensor case.

and n be positive integers. For the purposes of this article, a 3-*tensor* $\mathcal{A}$ over $\mathbb{F}$ is an $l \times m \times n$ array of elements of $\mathbb{F}$:

$$\mathcal{A} = [\![a_{ijk}]\!]_{i,j,k=1}^{l,m,n} \in \mathbb{F}^{l\times m\times n}. \tag{1}$$

These objects are natural multilinear generalizations of matrices in the following way.

For any positive integer d, let $\mathbf{e}_1, \dots, \mathbf{e}_d$ denote the standard basis[1] in the $\mathbb{F}$-vector space $\mathbb{F}^d$. A bilinear function $f : \mathbb{F}^m \times \mathbb{F}^n \to \mathbb{F}$ can be encoded by a matrix $A = [a_{ij}]_{i,j=1}^{m,n} \in \mathbb{F}^{m\times n}$, in which the entry a_{ij} records the value of $f(\mathbf{e}_i, \mathbf{e}_j) \in \mathbb{F}$. By linearity in each coordinate, specifying A determines the values of f on all of $\mathbb{F}^m \times \mathbb{F}^n$; in fact, we have $f(\mathbf{u}, \mathbf{v}) = \mathbf{u}^\top A\mathbf{v}$ for any vectors $\mathbf{u} \in \mathbb{F}^m$ and $\mathbf{v} \in \mathbb{F}^n$. Thus, matrices both encode 2-dimensional arrays of numbers and specify all bilinear functions. Notice also that if $m = n$ and $A = A^\top$ is symmetric, then

$$f(\mathbf{u}, \mathbf{v}) = \mathbf{u}^\top A\mathbf{v} = (\mathbf{u}^\top A\mathbf{v})^\top = \mathbf{v}^\top A^\top\mathbf{u} = \mathbf{v}^\top A\mathbf{u} = f(\mathbf{v}, \mathbf{u}).$$

Thus, symmetric matrices are bilinear maps invariant under coordinate exchange.

[1]Formally, $\mathbf{e}_i$ is the vector in $\mathbb{F}^d$ with a 1 in the ith coordinate and zeroes everywhere else. In this article, vectors in $\mathbb{F}^n$ will always be column-vectors.

These notions generalize: a 3-tensor is a trilinear function $f : \mathbb{F}^l \times \mathbb{F}^m \times \mathbb{F}^n \to \mathbb{F}$ which has a coordinate representation given by a *hypermatrix*[2] $\mathcal{A}$ as in (1). The subscripts and superscripts in (1) will be dropped whenever the range of i, j, k is obvious or unimportant. Also, a 3-tensor $[\![a_{ijk}]\!]_{i,j,k=1}^{n,n,n} \in \mathbb{F}^{n\times n\times n}$ is *symmetric* if

$$a_{ijk} = a_{ikj} = a_{jik} = a_{jki} = a_{kij} = a_{kji}. \tag{2}$$

These are coordinate representations of trilinear maps $f : \mathbb{F}^n \times \mathbb{F}^n \times \mathbb{F}^n \to \mathbb{F}$ with

$$f(\mathbf{u}, \mathbf{v}, \mathbf{w}) = f(\mathbf{u}, \mathbf{w}, \mathbf{v}) = f(\mathbf{v}, \mathbf{u}, \mathbf{w}) = f(\mathbf{v}, \mathbf{w}, \mathbf{u}) = f(\mathbf{w}, \mathbf{u}, \mathbf{v}) = f(\mathbf{w}, \mathbf{v}, \mathbf{u}).$$

We focus here on 3-tensors mainly for expositional purposes. One exception is the problem of deciding positive definiteness of a tensor, a notion nontrivial only in even orders.

When $\mathbb{F} = \mathbb{C}$, one may argue that a generalization of the notion of Hermitian or self-adjoint matrices would be more appropriate than that of symmetric matrices. For 3-tensors, such "self-adjointness" depends on a choice of a trilinear form, which might be natural in certain applications [Wei and Goldbart 2003]. Our complexity results for symmetric tensors apply as long as the chosen notion reduces to (2) for $\mathcal{A} \in \mathbb{R}^{n\times n\times n}$.

1.2. Tensor Eigenvalue

We now explain in detail the tensor eigenvalue problem since it is the simplest multilinear generalization. We shall also use the problem to illustrate many of the concepts that arise when studying other, more difficult, tensor problems. The basic notions for eigenvalues of tensors were introduced independently in Lim [2005] and Qi [2005], with more developments appearing in Ni et al. [2007] and Qi [2007]. Additional theory from the perspective of toric algebraic geometry and intersection theory was provided recently in Cartwright and Sturmfels [2013].

We describe the ideas more formally in Section 5, but for now it suffices to say that the usual eigenvalues and eigenvectors of a matrix $A \in \mathbb{R}^{n\times n}$ are the stationary values and points of its Rayleigh quotient, and this view generalizes to higher order tensors. This gives, for example, an *eigenvector* of a tensor $\mathcal{A} = [\![a_{ijk}]\!]_{i,j,k=1}^{n,n,n} \in \mathbb{F}^{n\times n\times n}$ as a nonzero vector $\mathbf{x} = [x_1, \dots, x_n]^\top \in \mathbb{F}^n$ satisfying:

$$\sum_{i,j=1}^{n} a_{ijk} x_i x_j = \lambda x_k, \quad k = 1, \dots, n, \tag{3}$$

for some $\lambda \in \mathbb{F}$, which is an *eigenvalue* of $\mathcal{A}$. We call the pairing $(\lambda, \mathbf{x})$ an *eigenpair* of $\mathcal{A}$. Notice that if $(\lambda, \mathbf{x})$ is an eigenpair, then so is $(t\lambda, t\mathbf{x})$ for any $t \neq 0$. Thus, we also say that two eigenpairs are *equivalent* if they are equal up to a nonzero scalar factor.

As in the matrix case, generic or "random" tensors over $\mathbb{F} = \mathbb{C}$ have a finite number of eigenpairs (up to this scaling equivalence), although their count is exponential in n. Still, it is possible for a tensor to have an infinite number of nonequivalent eigenvalues, but in that case they comprise a cofinite set of complex numbers. Another important fact is that over the reals ($\mathbb{F} = \mathbb{R}$), every 3-tensor has a real eigenpair. These results and more can be found in Cartwright and Sturmfels [2013].

The following problem is natural for applications.

Problem 1.1. Given $\mathcal{A} \in \mathbb{F}^{n\times n\times n}$, find $(\lambda, \mathbf{x}) \in \mathbb{F} \times \mathbb{F}^n$ with $\mathbf{x} \neq \mathbf{0}$ satisfying (3).

We first discuss the computability of this problem. When the entries of the tensor $\mathcal{A}$ are real numbers, there is an effective procedure that will output a finite

[2] We will not use the term hypermatrix but will simply regard a tensor as synonymous with its coordinate representation. See Lim [2013] for more details.

presentation of all real eigenpairs. A good reference for such methods in real algebraic geometry is Bochnak et al. [1998], and an overview of recent intersections between mathematical logic and algebraic geometry, more generally, can be found in Haskell et al. [2000]. Over $\mathbb{C}$, this problem can be tackled directly by computing a Gröbner basis with Buchberger's algorithm [Buchberger 1970] since an eigenpair is a solution to a system of polynomial equations over an algebraically closed field (e.g., Cartwright and Sturmfels [2013, Example 3.5]). Another approach is to work with Macaulay matrices of multivariate resultants [Ni et al. 2007]. References for such techniques suitable for numerical analysts are Cox et al. [2005, 2007].

Even though solutions to tensor problems are computable, all known methods quickly become impractical as the tensors become larger (i.e., as n grows). In principle, this occurs because simply listing the non-equivalent output to Problem 1.1 is already prohibitive. It is natural, therefore, to ask for faster methods checking whether a given λ is an eigenvalue or approximating a single eigenpair. We first analyze the following easier decision problem.

Problem 1.2 (*Tensor λ-Eigenvalue*). Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, and fix $\lambda \in \mathbb{Q}$. Decide if λ is an eigenvalue (with corresponding eigenvector in $\mathbb{F}^n$) of a tensor $\mathcal{A} \in \mathbb{Q}^{n\times n\times n}$.

Before explaining our results on Problem 1.2 and other tensor questions, we define the model of computational complexity that we shall utilize to study them.

1.3. Computability and Complexity

We hope this article will be useful to casual users of computational complexity—such as numerical analysts and optimization theorists—who nevertheless desire to understand the tractability of their problems in light of modern complexity theory. This section and the next provide a high-level overview for such an audience. In addition, we also carve out a perspective for real computation within the Turing machine framework that we feel is easier to work with than those proposed in Blum et al. [1989, 1998], Hochbaum and Shanthikumar [1990], and Vavasis [1991]. For readers who have no particular interest in tensor problems, the remainder of our article may then be viewed as a series of instructive examples showing how one may deduce the tractability of a numerical computing problem using the rich collection of NP-complete combinatorial problems.

Computational complexity is usually specified on the following three levels.

I. *Model of Computation*. What are inputs and outputs? What is a computation? For us, inputs will be rational numbers and outputs will be rational vectors or YES/NO responses, and computations are performed on a *Turing machine*. Alternatives for inputs include Turing computable numbers [Turing 1936; Weihrauch 2000] and real or complex numbers [Blum et al. 1989, 1998]. Alternatives for computation include the Pushdown Automaton [Sipser 2012], the Blum–Shub–Smale Machine [Blum et al. 1989, 1998], and the Quantum Turing Machine [Deutsch 1985].

II. *Model of Complexity*. What is the cost of a computation? In this article, we use *time complexity* measured in units of bit operations; that is, the number of READ, WRITE, MOVE, and other tape-level instructions on bits. This is the same for the ε-accuracy complexity model[3] [Hochbaum and Shanthikumar 1990]. In

[3]While the ε-accuracy complexity model is more realistic for numerical computations, it is not based on the IEEE floating-point standards [Kahan 1997; Overton 2001]. On the other hand, a model that combines both the flexibility of the ε-accuracy complexity model and the reality of floating-point arithmetic would inevitably be enormously complicated [Vavasis 1991, Section 2.4].

the Blum–Cucker–Shub–Smale (BCSS) model, it is time complexity measured in units of arithmetic and branching operations involving inputs of real or complex numbers. In quantum computing, it is time complexity measured in units of unitary operations on qubits. There are yet other models of complexity that measure other types of computational costs. For example, complexity in the Valiant model is based on arithmetic circuit size.

III. *Model of Reducibility*. Which problems do we consider equivalent in hardness? For us, it is the *Cook–Karp–Levin* (CKL) sense of reducibility [Cook 1971; Karp 1972; Levin 1973] and its corresponding problem classes: P, NP, NP-complete, NP-hard, etc. Reducibility in the BCSS model is essentially based on CKL. There is also reducibility in the Valiant sense, which applies to the aforementioned Valiant model and gives rise to the complexity classes VP and VNP [Bürgisser 2000; Valiant 1979a].

Computability is a question to be answered in Level I, whereas difficulty is to be answered in Levels II and III. In Level II, we have restricted ourselves to time complexity since this is the most basic measure and it already reveals that tensor problems are hard. In Level III, there is strictly speaking a subtle difference between the definition of reducibility by Cook [1971] and that by Karp and Levin [Karp 1972; Levin 1973]. We define precisely our notion of reducibility in Section 1.4.

Before describing our model more fully, we recall the well-known Blum–Cucker–Shub–Smale framework for studying complexity of real and complex computations [Blum et al. 1989, 1998]. In this model, an input is a list of n real or complex numbers, without regard to how they are represented. In this case, algorithmic computation (essentially) corresponds to arithmetic and branching on equality using a finite number of states, and a measure of computational complexity is the number of these basic operations[4] needed to solve a problem as a function of n. The central message of our article is that many problems in linear algebra that are efficiently solvable on a Turing machine become NP-hard in multilinear algebra. Under the BCSS model, however, this distinction is not yet possible. For example, while it is well-known that the feasibility of a linear program is in P under the traditional CKL notion of complexity [Khachiyan 1979], the same problem studied within BCSS is among the most daunting open problems in Mathematics (it is the 9th "Smale Problem" [Smale 2000]). The BCSS model has nonetheless produced significant contributions to computational mathematics, especially to the theory of polynomial equation solving (e.g., see Beltrán and Pardo [2009] and the references therein).

We now explain our model of computation. All computations are assumed to be performed on a Turing machine [Turing 1936] with the standard notion of time complexity involving operations on bits. Inputs will be rational numbers[5] and specified by finite strings of bits. Outputs will consist of rational numbers or YES/NO responses. A decision problem is said to be *computable* (or *decidable*) if there is a Turing machine that will output the correct answer (YES/NO) for all allowable inputs in finitely many steps. It is said to be *uncomputable* (or *undecidable*) otherwise; that is, no Turing machine could always determine the correct answer in finitely many steps. Note that the

[4]To illustrate the difference between BCSS/CKL, consider the problem of deciding whether two integers r, s multiply to give an integer t. For BCSS, the time complexity is constant since one can compute $u = rs - t$ and check "$u = 0$?" in constant time. Under CKL, however, the problem has best-known time complexity of $N \log(N) 2^{O(\log^* N)}$, where N is the number of bits to specify r, s, and t [De et al. 2008; Fürer 2007].

[5]The only exception is when we prove NP-hardness of symmetric tensor eigenvalue (Section 9), where we allow input eigenvalues λ to be in the field $\mathbb{F} = \{a + b\sqrt{d} : a, b \in \mathbb{Q}\}$ for any fixed positive integer d. Note that such inputs may also be specified with a finite number of bits.

definition of a problem includes a specification of allowed inputs. We refer the reader to Sipser [2012] for a proper treatment and to Poonen [2012] for an extensive list of undecidable problems arising from many areas of modern mathematics.

We remark that although quantities such as eigenvalues, spectral norms, etc., of a tensor will in general not be rational, our reductions have been carefully constructed such that they are rational (or at least finite bit-length) in the cases we study.

The next section describes our notion of reducibility for tensor decision problems such as Problem 1.2 encountered in this article.

1.4. NP-Hardness

The following is the notion of NP-hardness that we shall use throughout this article. As described previously, inputs will be rational numbers, and input size is measured in the number of bits required to specify the input. Briefly, we say that a problem $\mathscr{D}_1$ is *polynomially reducible* to a problem $\mathscr{D}_2$ if the following holds: any input to $\mathscr{D}_1$ can be transformed in polynomially many steps (in the input size) into a set of polynomially larger inputs to $\mathscr{D}_2$ problems such that the corresponding answers can be used to correctly deduce (again, in a polynomial number of steps) the answer to the original $\mathscr{D}_1$ question. Informally, $\mathscr{D}_1$ polynomially reduces to $\mathscr{D}_2$ if there is a way to solve $\mathscr{D}_1$ by a deterministic polynomial-time algorithm when that algorithm is allowed to compute answers to instances of problem $\mathscr{D}_2$ in unit time. Note that the relation is not symmetric—$\mathscr{D}_1$ polynomially reduces to $\mathscr{D}_2$ does not imply that $\mathscr{D}_2$ also reduces to $\mathscr{D}_1$.

By the Cook–Levin Theorem, if one can polynomially reduce any particular NP-complete problem to a problem $\mathscr{D}$, then all NP-complete problems are so reducible to $\mathscr{D}$. We call a decision problem *NP-hard* if one can polynomially reduce any NP-complete decision problem (such as whether a graph is 3-colorable) to it [Knuth 1974a, 1974b]. Thus, an NP-hard problem is at least as hard as any NP-complete problem and quite possibly harder.[6]

Although tensor eigenvalue is computable for $\mathbb{F} = \mathbb{R}$, it is nonetheless NP-hard, as our next theorem explains. For two sample reductions, see Example 1.4.

THEOREM 1.3. *Graph* 3*-colorability is polynomially reducible to tensor* 0*-eigenvalue over* $\mathbb{R}$*. Thus, deciding tensor eigenvalue over* $\mathbb{R}$ *is NP-hard.*

A basic open question is whether deciding tensor eigenvalue is also NP-complete. In other words, if a nontrivial solution to (3) exists for a fixed λ, is there a polynomial-time verifiable certificate of this fact? A natural candidate for the certificate is the eigenvector itself, whose coordinates would be represented as certain zeroes of univariate polynomials with rational coefficients. The relationship between the size of these coefficients and the size of the input, however, is subtle and beyond our scope. In the case of linear equations, polynomial bounds on the number of digits necessary to represent a solution can already be found in Edmonds [1967] (see Vavasis [1991, Theorem 2.1]); and for the sharpest results to date on certificates for homogeneous linear systems, see Freitas et al. [2012]. In contrast, rudimentary bounds for the type of tensor problems considered in this article are, as far as we know, completely out of reach.

Example 1.4 (*Real Tensor* 0*-Eigenvalue Solves* 3*-Colorability*). Let $G = (V, E)$ be a simple, undirected graph with vertices $V = \{1, \dots, v\}$ and edges E. Recall that a *proper* (*vertex*) 3*-coloring* of G is an assignment of one of three colors to each of its vertices

[6] For those unfamiliar with these notions, we feel obliged to point out that the set of NP-hard problems is different than the set of NP-complete ones. First of all, an NP-hard problem may not be a decision problem, and secondly, even if we are given an NP-hard decision problem, it might not be in the class NP; that is, one might not be able to certify a YES decision in a polynomial number of steps. NP-complete problems, by contrast, are always decision problems and in the class NP.

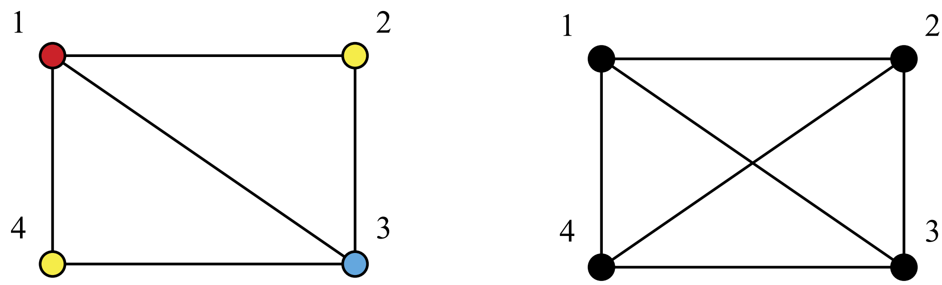

Fig. 1. Simple graphs with six proper 3-colorings (graph at the left) or none (graph at the right).

such that adjacent vertices receive different colors. We say that G is 3-*colorable* if it has a proper 3-coloring. See Figure 1 for an example of a 3-colorable graph on four vertices. Determining whether G is 3-colorable is a well-known NP-complete decision problem.

As we shall see in Section 2, proper 3-colorings of the left-hand side graph in Figure 1 can be encoded as the nonzero real solutions to the following square set of $n = 35$ quadratic polynomials in 35 real unknowns a_i, b_i, c_i, d_i $(i = 1, \ldots, 4)$, u, w_i $(i = 1, \ldots, 18)$:

$$\begin{aligned}
&a_1c_1 - b_1d_1 - u^2,\ b_1c_1 + a_1d_1,\ c_1u - a_1^2 + b_1^2,\ d_1u - 2a_1b_1,\ a_1u - c_1^2 + d_1^2,\ b_1u - 2d_1c_1,\\
&a_2c_2 - b_2d_2 - u^2,\ b_2c_2 + a_2d_2,\ c_2u - a_2^2 + b_2^2,\ d_2u - 2a_2b_2,\ a_2u - c_2^2 + d_2^2,\ b_2u - 2d_2c_2,\\
&a_3c_3 - b_3d_3 - u^2,\ b_3c_3 + a_3d_3,\ c_3u - a_3^2 + b_3^2,\ d_3u - 2a_3b_3,\ a_3u - c_3^2 + d_3^2,\ b_3u - 2d_3c_3,\\
&a_4c_4 - b_4d_4 - u^2,\ b_4c_4 + a_4d_4,\ c_4u - a_4^2 + b_4^2,\ d_4u - 2a_4b_4,\ a_4u - c_4^2 + d_4^2,\ b_4u - 2d_4c_4,\\
&a_1^2 - b_1^2 + a_1a_3 - b_1b_3 + a_3^2 - b_3^2,\ a_1^2 - b_1^2 + a_1a_4 - b_1b_4 + a_4^2 - b_4^2,\ a_1^2 - b_1^2 + a_1a_2 - b_1b_2 + a_2^2 - b_2^2,\\
&a_2^2 - b_2^2 + a_2a_3 - b_2b_3 + a_3^2 - b_3^2,\ a_3^2 - b_3^2 + a_3a_4 - b_3b_4 + a_4^2 - b_4^2,\ 2a_1b_1 + a_1b_2 + a_2b_1 + 2a_2b_2,\\
&2a_2b_2 + a_2b_3 + a_3b_2 + 2a_3b_3,\ 2a_1b_1 + a_1b_3 + a_2b_1 + 2a_3b_3,\ 2a_1b_1 + a_1b_4 + a_4b_1 + 2a_4b_4,\\
&2a_3b_3 + a_3b_4 + a_4b_3 + 2a_4b_4,\ w_1^2 + w_2^2 + \cdots + w_{17}^2 + w_{18}^2.
\end{aligned} \tag{4}$$

The coefficient matrices for the 35 quadratic forms appearing in (4) define a tensor $\mathcal{A} \in \mathbb{R}^{35\times 35\times 35}$ as in (3). In particular, deciding 0-eigenvalue over $\mathbb{F} = \mathbb{R}$ for this $\mathcal{A}$ is equivalent to solving an instance of an NP-hard problem.

Using symbolic algebra or numerical algebraic geometry software[7] (see the Appendix for a list), one can solve these equations to find six real solutions (without loss of generality, we may take $u = 1$ and all $w_j = 0$), which correspond to the proper 3-colorings of the graph G as follows. Fix one such solution and define $x_k = a_k + \mathrm{i}b_k \in \mathbb{C}$ for $k = 1, \ldots, 4$ (we set $\mathrm{i} = \sqrt{-1}$). By construction, these x_k are one of the three cube roots of unity $\{1, \alpha, \alpha^2\}$, where $\alpha = \exp(2\pi \mathrm{i}/3) = -\frac{1}{2} + \mathrm{i}\frac{\sqrt{3}}{2}$ (see also Figure 2).

To determine a 3-coloring from this solution, one "colors" each vertex i by the root of unity that equals x_i. It can be checked that no two adjacent vertices share the same color in a coloring; thus, they are proper 3-colorings. For example, one solution is:

$$x_1 = -\frac{1}{2} - \mathrm{i}\frac{\sqrt{3}}{2}, \quad x_2 = 1, \quad x_3 = -\frac{1}{2} + \mathrm{i}\frac{\sqrt{3}}{2}, \quad x_4 = 1.$$

[7]For code used in this article, see: `http://www.msri.org/people/members/chillar/code.html`.

Polynomials for the right-hand side graph in Figure 1 are the same as (4) except for two additional ones encoding a new restriction for colorings, the extra edge $\{2, 4\}$:

$$a_2^2 - b_2^2 + a_2a_4 - b_2b_4 + a_4^2 - b_4^2, \quad 2a_2b_2 + a_2b_4 + a_4b_2 + 2a_4b_4.$$

One can verify with the same software that these extra equations force the system to have no nonzero real solutions, and thus no proper 3-colorings.

Finally, note that if we could count real (nonequivalent) eigenvectors with eigenvalue $\lambda = 0$, we would solve the enumeration problem for proper 3-colorings of graphs (in particular, this proves Corollary 1.16).

1.5. Approximation Schemes

Although the tensor eigenvalue decision problem is NP-hard and the eigenvector enumeration problem #P-hard, it might be possible to approximate *some* eigenpair $(\lambda, \mathbf{x})$ efficiently, which is important for some applications (e.g., Schultz and Seidel [2008]). For those unfamiliar with these ideas, an *approximation scheme* for a tensor problem (such as finding a tensor eigenvector) is an algorithm producing a (rational) approximate solution to within ε of some solution. An approximation scheme is said to run in *polynomial time* (PTAS) if its running-time is polynomial in the input size for any fixed $\varepsilon > 0$, and *fully polynomial time* (FPTAS) if its running-time is polynomial in both the input size and $1/\varepsilon$. There are other notions of approximation [Hochbaum 1997; Vazirani 2003], but we limit our discussion to these.

Fix λ, which we assume is an eigenvalue of a tensor $\mathcal{A}$. Formally, we say it is *NP-hard to approximate an eigenvector* in an eigenpair $(\lambda, \mathbf{x})$ to within $\varepsilon > 0$ if (unless P = NP), there is no polynomial-time algorithm that always produces a nonzero $\hat{\mathbf{x}} = [\hat{x}_1, \dots, \hat{x}_n]^\top \in \mathbb{F}^n$ that approximates some solution $\mathbf{0} \neq \mathbf{x} \in \mathbb{F}^n$ to system (3) by satisfying for all i and j:

$$|\hat{x}_i/\hat{x}_j - x_i/x_j| < \varepsilon, \quad \text{whenever } x_j \neq 0. \tag{5}$$

This measure of approximation is natural in our context because of the scale equivalence of eigenpairs, and it is closely related to standard relative error $\|\hat{\mathbf{x}} - \mathbf{x}\|_\infty / \|\mathbf{x}\|_\infty$ in numerical analysis. We shall prove the following inapproximability result in Section 5.

THEOREM 1.5. *It is NP-hard to approximate tensor eigenvector over $\mathbb{R}$ to $\varepsilon = \frac{3}{4}$.*

COROLLARY 1.6. *No PTAS approximating tensor eigenvector exists unless P = NP.*

1.6. Tensor Singular Value, Hyperdeterminant, and Spectral Norm

Our next result involves the singular value problem. We postpone definitions until Section 6, but state the main result here.

THEOREM 1.7. *Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$, and fix $\sigma \in \mathbb{Q}$. Deciding whether σ is a singular value over $\mathbb{F}$ of a tensor is NP-hard.*

There is also a notion of *hyperdeterminant*, which we discuss in more depth in Section 3 (see also Section 4). Like the determinant, this is a homogeneous polynomial (with integer coefficients) in the entries of a tensor that vanishes if and only if the tensor has a zero singular value. The following problem is important for multilinear equation solving (e.g., Example 3.2).

Problem 1.8. Decide if the hyperdeterminant of a tensor is zero.

We were unable to determine the hardness of Problem 1.8, but conjecture that it is difficult.

CONJECTURE 1.9. *It is NP-hard to decide the vanishing of the hyperdeterminant.*

We are, however, able to evaluate the complexity of computing the spectral norm (see Definition 6.6), which is a special singular value of a tensor.

THEOREM 1.10. *Fix any nonzero $\sigma \in \mathbb{Q}$. Deciding whether σ is the spectral norm of a tensor is NP-hard.*

Determining the spectral norm of a tensor is an optimization (maximization) problem. Thus, while it is NP-hard to decide tensor spectral norm, there might be efficient ways to approximate it. A famous example of approximating solutions to problems whose decision formulations are NP-hard is the classical result of Goemans and Williamson [1995] which gives a polynomial-time algorithm to determine a cut size of a graph that is at least .878 times that of a maximum cut. In fact, it has been shown, assuming the Unique Games Conjecture, that Goemans-Williamson's approximation factor is best possible [Khot et al. 2007]. We refer the reader to De Klerk and Pasechnik [2002], Alon and Naor [2006], Bachoc and Vallentin [2008], Bachoc et al. [2009], Briët et al. [2010a, 2010b], and He et al. [2010] for some recent work in the field of approximation algorithms.

Formally, we say that it is *NP-hard to approximate the spectral norm* of a tensor to within $\varepsilon > 0$ if (unless P = NP) there is no polynomial-time algorithm giving a guaranteed lower bound for the spectral norm that is at least a $(1-\varepsilon)$-factor of its true value. Note that ε here might be a function of the input size. A proof of the following can be found in Section 6.

THEOREM 1.11. *It is NP-hard to approximate the spectral norm of a tensor $\mathcal{A}$ to within*

$$\varepsilon = 1 - \left(1 + \frac{1}{N(N-1)}\right)^{-1/2} = \frac{1}{2N(N-1)} + O(N^{-4}),$$

where N is the input size of $\mathcal{A}$.

COROLLARY 1.12. *No FPTAS to approximate spectral norm exists unless P = NP.*

PROOF. Suppose there is a FPTAS for the tensor spectral norm problem and take $\varepsilon = 1/(4N^2)$ as the approximation error desired for a tensor of input size N. Then, in time polynomial in $1/\varepsilon = 4N^2$ (and thus in N), it would be possible to approximate the spectral norm of a tensor with input size N to within $1 - \left(1 + \frac{1}{N(N-1)}\right)^{-1/2}$ for all large N. From Theorem 1.11, this is only possible if P = NP. □

1.7. Tensor Rank

The *outer product* $\mathcal{A} = \mathbf{x} \otimes \mathbf{y} \otimes \mathbf{z}$ of vectors $\mathbf{x} \in \mathbb{F}^l$, $\mathbf{y} \in \mathbb{F}^m$, and $\mathbf{z} \in \mathbb{F}^n$ is the tensor $\mathcal{A} = [\![a_{ijk}]\!]_{i,j,k=1}^{l,m,n}$ given by $a_{ijk} = x_i y_j z_k$. A nonzero tensor that can be expressed as an outer product of vectors is called *rank*-1. More generally, the *rank* of a tensor $\mathcal{A} = [\![a_{ijk}]\!] \in \mathbb{F}^{l\times m\times n}$, denoted $\operatorname{rank}(\mathcal{A})$, is the minimum r for which $\mathcal{A}$ is a sum of r rank-1 tensors [Hitchcock 1927a, 1927b] with $\lambda_i \in \mathbb{F}$, $\mathbf{x}_i \in \mathbb{F}^l$, $\mathbf{y}_i \in \mathbb{F}^m$, and $\mathbf{z}_i \in \mathbb{F}^n$:

$$\operatorname{rank}(\mathcal{A}) = \min\left\{r : \mathcal{A} = \sum\nolimits_{i=1}^{r} \lambda_i\, \mathbf{x}_i \otimes \mathbf{y}_i \otimes \mathbf{z}_i\right\}. \tag{6}$$

For a symmetric tensor $\mathcal{S} \in \mathbb{F}^{n\times n\times n}$, we shall require that the vectors in the outer product be the same:

$$\operatorname{srank}(\mathcal{S}) = \min\left\{r : \mathcal{S} = \sum\nolimits_{i=1}^{r} \lambda_i\, \mathbf{x}_i \otimes \mathbf{x}_i \otimes \mathbf{x}_i\right\}. \tag{7}$$

The number in (7) is called the *symmetric rank* of $\mathcal{S}$. It is still not known whether a symmetric tensor's symmetric rank is always its rank (this is the Comon Conjecture [Landsberg 2012]; see also Reznick [2013] and Comon et al. [2008]), although the best symmetric rank-1 approximation and the best rank-1 approximation coincide (see Section 10). Note that these definitions of rank agree with matrix rank when applied to a 2-tensor.

Our next result says that approximating a tensor with a single rank-1 element is already hard. A consequence is that data analytic models under the headings of PARAFAC, CANDECOMP, and TUCKER—originating from psychometrics but having newfound popularity in other areas of data analysis—are all NP-hard to fit in the simplest case.

THEOREM 1.13. *Rank-1 tensor approximation over $\mathbb{R}$ is NP-hard.*

As will become clear, tensor rank as defined in (6) implicitly depends on the choice of field. Suppose that $\mathbb{F} \subseteq \mathbb{E}$ is a subfield of a field $\mathbb{E}$. If $\mathcal{A} \in \mathbb{F}^{l\times m\times n}$ is as in (6), but we allow $\lambda_i \in \mathbb{E}$, $\mathbf{x}_i \in \mathbb{E}^l$, $\mathbf{y}_i \in \mathbb{E}^m$, and $\mathbf{z}_i \in \mathbb{E}^n$, then the number computed in (6) is called the *rank of $\mathcal{A}$ over* $\mathbb{E}$. We will write $\operatorname{rank}_{\mathbb{E}}(\mathcal{A})$ (a notation that we will use whenever the choice of field is important) for the rank of $\mathcal{A}$ over $\mathbb{E}$. In general, it is possible that

$$\operatorname{rank}_{\mathbb{E}}(\mathcal{A}) < \operatorname{rank}_{\mathbb{F}}(\mathcal{A}).$$

We discuss this in detail in Section 8, where we give a new result about the rank of tensors over changing fields (in contrast, the rank of a matrix does not change when the ground field is enlarged). The proof uses symbolic and computational algebra in a fundamental way.

THEOREM 1.14. *There is a rational tensor $\mathcal{A} \in \mathbb{Q}^{2\times 2\times 2}$ with* $\operatorname{rank}_{\mathbb{R}}(\mathcal{A}) < \operatorname{rank}_{\mathbb{Q}}(\mathcal{A})$.

Håstad has famously shown that tensor rank over $\mathbb{Q}$ is NP-hard [Håstad 1990]. Since tensor rank over $\mathbb{Q}$ differs in general from tensor rank over $\mathbb{R}$, it might be possible that tensor rank is not NP-hard over $\mathbb{R}$ and $\mathbb{C}$. In Section 8, we explain how the argument in Håstad [1990] also proves the NP-hardness of tensor rank over any extension of $\mathbb{Q}$.

THEOREM 1.15. [HÅSTAD 1990]. *Tensor rank is NP-hard over $\mathbb{R}$ and $\mathbb{C}$.*

1.8. Symmetric Tensors

One may wonder if NP-hard problems for general nonsymmetric tensors might perhaps become tractable for symmetric ones. We show that restricting these problems to the class of symmetric tensors does not remove NP-hardness. As with their nonsymmetric counterparts, eigenvalue, singular value, spectral norm, and best rank-1 approximation problems for symmetric tensors all remain NP-hard. In particular, the NP-hardness of symmetric spectral norm in Theorem 10.2 answers an open problem in Brubaker and Vempala [2009].

1.9. #P-Hardness and VNP-Hardness

As is evident from the title of our article and the list in Table I, we have used NP-hardness as our primary measure of computational intractability. Valiant's notions of *#P-completeness* [Valiant 1979b] and *VNP-completeness* [Valiant 1979a] are nonetheless relevant to tensor problems. The following result about tensor eigenvalue is directly implied by our work (see the discussion near the end of Example 1.4).

COROLLARY 1.16. *It is #P-hard to count tensor eigenvectors over $\mathbb{R}$.*

Because of space constraints, we will not elaborate on these notions except to say that #P-completeness applies to enumeration problems associated with NP-complete decision problems while VNP-completeness applies to polynomial evaluation problems. For example, deciding whether a graph is 3-colorable is an NP-complete decision problem, but counting the number of proper 3-colorings is a #P-complete enumeration problem.

The VNP complexity classes involve questions about the minimum number of arithmetic operations required to evaluate multivariate polynomials. An illuminating example [Landsberg 2012, Example 13.3.1.2] is given by

$$p_n(x,y) = x^n + nx^{n-1}y + \binom{n}{2}x^{n-2}y^2 + \cdots + \binom{n}{2}x^2y^{n-2} + nxy^{n-1} + y^n,$$

which at first glance requires $n(n+1)$ multiplications and n additions to evaluate. However, from the binomial expansion, we have $p_n(x,y) = (x+y)^n$, and so the operation count can be reduced to $n-1$ multiplications and 1 addition. In fact, this last count is not minimal, and a more careful study of such questions involves arithmetic circuit complexity. We refer the reader to Bürgisser [2000] for a detailed exposition.

As in Section 1.4, one may also analogously define notions of #P-hardness and VNP-hardness: A problem is said to be *#P-hard* (respectively, *VNP-hard*) if every #P-complete enumeration problem (respectively, VNP-complete polynomial evaluation problem) may be polynomially reduced to it. Evidently, a #P-hard (respectively, VNP-hard) problem is at least as hard and quite possibly harder than a #P-complete (respectively, VNP-complete) problem.

1.10. Intractable Matrix Problems

Not all matrix problems are tractable. For example, matrix (p,q)-norms when $1 \le q \le p \le \infty$ [Steinberg 2005], nonnegative rank of nonnegative matrices [Arora et al. 2012; Vavasis 2009], sparsest null vector [Coleman and Pothen 1986], and rank minimization with linear constraints [Natarajan 1995] are all known to be NP-hard; the matrix p-norm when $p \neq 1, 2, \infty$ is NP-hard to approximate [Hendrickx and Olshevsky 2010]; and evaluating the permanent of a $\{0,1\}$-valued matrix is a well-known #P-complete problem [Valiant 1979b]. Our intention is to highlight the sharp distinction between the computational intractability of certain tensor problems and the tractability of their matrix specializations. As such, we do not investigate tensor problems that are known to be hard for matrices.

1.11. Quantum Computers

Another question sometimes posed to the authors is whether quantum computers might help with these problems. This is believed unlikely because of the seminal works [Bernstein and Vazirani 1997; Fortnow and Rogers 1999] (see also the survey [Fortnow 2009]). These authors have demonstrated that the complexity class of bounded error quantum polynomial time (BQP) is not expected to overlap with the complexity class of NP-hard problems. Since BQP encompasses the decision problems solvable by a quantum computer in polynomial time, the NP-hardness results in this article show that quantum computers are unlikely to be effective for tensor problems.

1.12. Finite Fields

We have restricted our discussion in this article to extension fields of $\mathbb{Q}$ as these are most relevant for the numerical computation arising in science and engineering. Corresponding results over finite fields are nonetheless also of interest in computer science; for instance, quadratic feasibility arises in cryptography [Courtois et al. 2002] and tensor rank arises in boolean satisfiability problems [Håstad 1990].

2. QUADRATIC FEASIBILITY IS NP-HARD

Since it will be a basic tool for us in proving results about tensors (e.g., Theorem 1.3 for tensor eigenvalue), we examine the complexity of solving quadratic equations.

Problem 2.1. Let $\mathbb{F} = \mathbb{Q}, \mathbb{R}$, or $\mathbb{C}$. For $i = 1, \ldots, m$, let $A_i \in \mathbb{Q}^{n\times n}$, $\mathbf{b}_i \in \mathbb{Q}^n$, and $c_i \in \mathbb{Q}$. Also, let $\mathbf{x} = [x_1, \ldots, x_n]^\top$ be a vector of unknowns, and set $G_i(\mathbf{x}) = \mathbf{x}^\top A_i \mathbf{x} + \mathbf{b}_i^\top \mathbf{x} + c_i$. Decide if the system $\{G_i(\mathbf{x}) = 0\}_{i=1}^m$ has a solution $\mathbf{x} \in \mathbb{F}^n$.

Another quadratic problem that is more natural in our setting is the following.

Problem 2.2 (*Quadratic Feasibility*). Let $\mathbb{F} = \mathbb{Q}, \mathbb{R}$, or $\mathbb{C}$. For $i = 1, \ldots, m$, let $A_i \in \mathbb{Q}^{n\times n}$ and set $G_i(\mathbf{x}) = \mathbf{x}^\top A_i \mathbf{x}$. Decide if the system of equations $\{G_i(\mathbf{x}) = 0\}_{i=1}^m$ has a nonzero solution $\mathbf{0} \neq \mathbf{x} \in \mathbb{F}^n$.

Remark 2.3. It is elementary that the (polynomial) complexity of Problem 2.1 is the same as that of Problem 2.2 when $\mathbb{F} = \mathbb{Q}$ or $\mathbb{R}$. To see this, homogenize each equation $G_i = 0$ in Problem 2.1 by introducing z as a new unknown: $\mathbf{x}^\top A_i \mathbf{x} + \mathbf{b}_i^\top \mathbf{x} z + c_i z^2 = 0$. Next, introduce the quadratic equation $x_1^2 + \cdots + x_n^2 - z^2 = 0$. This new set of equations is easily seen to have a nonzero solution if and only if the original system has any solution at all. The main trick used here is that $\mathbb{R}$ is a *formally real field*: that is, we always have $\sum x_i^2 = 0 \;\Rightarrow\; x_i = 0$ for all i.

Problem 2.2 for $\mathbb{F} = \mathbb{R}$ was studied in Barvinok [1993]. There, it is shown that for fixed n, one can decide the real feasibility of $m \gg n$ such quadratic equations $\{G_i(\mathbf{x}) = 0\}_{i=1}^m$ in n unknowns in a number of arithmetic operations that is polynomial in m. In contrast, we shall show that quadratic feasibility is NP-hard over $\mathbb{R}$ and $\mathbb{C}$.

To give the reader a sense of the generality of nonlinear equation solving, we first explain, in very simplified form, the connection of quadratic systems to the Halting Problem established in the seminal works [Davis et al. 1961; Matijasevič 1970]. Collectively, these papers resolve (in the negative) Hilbert's 10th Problem [Hilbert 1902]: Is there a finite procedure to decide the solvability of general polynomial equations over the integers (the so-called *Diophantine Problem* over $\mathbb{Z}$). For an exposition of the ideas involved, see Davis et al. [1976], Jones and Matijasevič [1984], or the introductory book [Matijasevič 1993].

The following fact in theoretical computer science is a basic consequence of these papers. Fix a universal Turing machine. There is a listing of all Turing machines $\mathscr{T}_x$ $(x = 1, 2, \ldots.)$ and a finite set $S = S(x)$ of integral quadratic equations in the parameter x and other unknowns with the following property: For each particular positive integer $x = 1, 2, \ldots$, the system $S(x)$ has a solution in positive integers if and only if Turing machine $\mathscr{T}_x$ halts with no input. In particular, polynomial equation solving over the positive integers is undecidable.

THEOREM 2.4 (DAVIS–PUTNAM–ROBINSON, MATIJASEVIC). *Problem 2.1 is undecidable over* $\mathbb{Z}$.

PROOF. Consider this system S of polynomials. For each of the unknowns y in S, we add an equation in four new variables a, b, c, d encoding (by Lagrange's Four-Square Theorem) that y should be a positive integer:

$$y = a^2 + b^2 + c^2 + d^2 + 1.$$

Let S' denote this new system. Polynomials $S'(x)$ have a common zero in integers if and only if $S(x)$ has a solution in positive integers. Thus, if we could decide the solvability of quadratic equations over the integers, we would solve the Halting problem. □

Remark 2.5. Using Jones [1982, pp. 552], one can construct an explicit set of quadratics whose solvability over $\mathbb{Z}$ encodes whether a given Turing machine halts.

We note that the decidability of Problem 2.1 over $\mathbb{Q}$ is unknown, as is the decidability of the general Diophantine problem over $\mathbb{Q}$. See Poonen [2003] for progress on this difficult problem.

While a system of quadratic equations exhibited in Problem 2.1 determined by coefficients $\mathcal{A} \in \mathbb{Z}^{m\times n\times n}$ is undecidable, we may decide whether a system of *linear* equations $A\mathbf{x} = \mathbf{b}$ with $A \in \mathbb{Z}^{m\times n}$, $\mathbf{b} \in \mathbb{Z}^m$ has an integral solution $\mathbf{x}$ by computing the Smith normal form of the coefficient matrix A (e.g., see Yap [2000]). We view this as another instance in which the transition from matrices A to tensors $\mathcal{A}$ has a drastic effect on computability. Another example of a matrix problem that becomes undecidable when one states its analogue for 3-tensors is given in Section 12.

We next study quadratic feasibility when $\mathbb{F} = \mathbb{R}$ and $\mathbb{C}$ and show that it is NP-hard. Variations of this basic result appear in Bayer [1982], Lovász [1994], and Grenet et al. [2010].

THEOREM 2.6. *Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Graph 3-colorability is polynomially reducible to quadratic feasibility over $\mathbb{F}$. Thus, Problem 2.2 over $\mathbb{F}$ is NP-hard.*

The idea of turning colorability problems into questions about polynomials appears to originate with Bayer's thesis although it has arisen in several other places, including Lovász [1994], De Loera [1995], and De Loera et al. [2008]. For a recent application of polynomial algebra to deciding unique 3-colorability, see Hillar and Windfeldt [2008].

To prove Theorem 2.6, we shall reduce graph 3-colorability to quadratic feasibility over $\mathbb{C}$. The result for $\mathbb{F} = \mathbb{R}$ then follows from the following fact.

LEMMA 2.7. *Let $\mathbb{F} = \mathbb{R}$ and A_i, G_i be as in Problem 2.2. Consider a new system $H_j(\mathbf{x}) = \mathbf{x}^\top B_j \mathbf{x}$ of $2m$ equations in $2n$ unknowns given by:*

$$B_i = \begin{bmatrix} A_i & 0 \\ 0 & -A_i \end{bmatrix}, \quad B_{m+i} = \begin{bmatrix} 0 & A_i \\ A_i & 0 \end{bmatrix}, \quad i = 1, \ldots, m.$$

The equations $\{H_j(\mathbf{x}) = 0\}_{j=1}^{2m}$ have a nonzero real solution $\mathbf{x} \in \mathbb{R}^{2n}$ if and only if the equations $\{G_i(\mathbf{z}) = 0\}_{i=1}^{m}$ have a nonzero complex solution $\mathbf{z} \in \mathbb{C}^n$.

PROOF. By construction, a nonzero solution $\mathbf{z} = \mathbf{u} + i\mathbf{v} \in \mathbb{C}^n$ with $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$ to equations $\{G_i(\mathbf{z}) = 0\}_{i=1}^{m}$ corresponds to a nonzero real solution $\mathbf{x} = [\mathbf{u}^\top, \mathbf{v}^\top]^\top$ to $\{H_j(\mathbf{x}) = 0\}_{j=1}^{2m}$. □

The following trivial observation gives flexibility in specifying quadratic feasibility problems over $\mathbb{R}$. This is useful since the system defining an eigenpair is a square system.

LEMMA 2.8. *Let $G_i(\mathbf{x}) = \mathbf{x}^\top A_i \mathbf{x}$ for $i = 1, \ldots, m$ with $A_i \in \mathbb{R}^{n\times n}$. Consider a new system $H_i(\mathbf{x}) = \mathbf{x}^\top B_i \mathbf{x}$ of $r \geq m+1$ equations in $s \geq n$ unknowns given by $s \times s$ matrices:*

$$B_i = \begin{bmatrix} A_i & 0 \\ 0 & 0 \end{bmatrix}, \; i = 1, \ldots, m; \quad B_j = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \; j = m+1, \ldots, r-1; \quad B_r = \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix};$$

in which I is the $(s-n) \times (s-n)$ identity matrix. Equations $\{H_i(\mathbf{x}) = 0\}_{i=1}^{r}$ have a nonzero solution $\mathbf{x} \in \mathbb{R}^s$ if and only if $\{G_i(\mathbf{x}) = 0\}_{i=1}^{m}$ have a nonzero solution $\mathbf{x} \in \mathbb{R}^n$.

The following set of polynomials C_G allows us to relate feasibility of a polynomial system to 3-colorability of a graph G. An instance of this encoding (after applying Lemmas 2.7 and 2.8 appropriately) can be found in Example 1.4.

Definition 2.9. The *color encoding* of a graph $G = (V, E)$ with v vertices is the set of $4v$ quadratic polynomials in $2v + 1$ unknowns $x_1, \ldots, x_v, y_1, \ldots, y_v, z$:

$$C_G = \begin{cases} x_i y_i - z^2, \quad y_i z - x_i^2, \quad x_i z - y_i^2, & i = 1, \ldots, v, \\ \sum_{j:\{i,j\} \in E} (x_i^2 + x_i x_j + x_j^2), & i = 1, \ldots, v. \end{cases} \tag{8}$$

LEMMA 2.10. *C_G has a nonzero complex solution if and only if G is 3-colorable.*

PROOF. Suppose that G is 3-colorable and let $[x_1, \ldots, x_v]^\top \in \mathbb{C}^n$ be a proper 3-coloring of G, encoded using cube roots of unity as in Example 1.4. Set $z = 1$ and $y_i = 1/x_i$ for $i = 1, \ldots, v$; we claim that these numbers are a common zero of C_G. It is clear that the first $3v$ polynomials in (8) evaluate to zero. Next consider any expression of the form $p_i = \sum_{j:\{i,j\} \in E} x_i^2 + x_i x_j + x_j^2$. Since we have a 3-coloring, $x_i \neq x_j$ for $\{i, j\} \in E$; thus,

$$0 = x_i^3 - x_j^3 = \frac{x_i^3 - x_j^3}{x_i - x_j} = x_i^2 + x_i x_j + x_j^2.$$

In particular, each p_i evaluates to zero as desired.

Conversely, suppose that the polynomials C_G have a common nontrivial solution,

$$\mathbf{0} \neq [x_1, \ldots, x_v, y_1, \ldots, y_v, z]^\top \in \mathbb{C}^{2v+1}.$$

If $z = 0$, then all of the x_i and y_i must be zero as well. Thus $z \neq 0$, and since the equations are homogenous, we may assume that our solution has $z = 1$. It follows that $x_i^3 = 1$ for all i so that $[x_1, \ldots, x_v]^\top$ is a 3-coloring of G. We are left with verifying that it is proper. If $\{i, j\} \in E$ and $x_i = x_j$, then $x_i^2 + x_i x_j + x_j^2 = 3x_i^2$; otherwise, if $\{i, j\} \in E$ and $x_i \neq x_j$, then $x_i^2 + x_i x_j + x_j^2 = 0$. Thus, $3r_i x_i^2 = p_i$ $(= 0)$, where r_i is the number of vertices j adjacent to i that have $x_i = x_j$. It follows that $r_i = 0$ so that $x_i \neq x_j$ for all $\{i, j\} \in E$, and therefore G has a proper 3-coloring. □

We close with a proof that quadratic feasibility over $\mathbb{C}$ (and therefore, $\mathbb{R}$) is NP-hard.

PROOF OF THEOREM 2.6. Given a graph G, construct the color encoding polynomials C_G. From Lemma 2.10, the homogeneous quadratics C_G have a nonzero complex solution if and only if G is 3-colorable. Thus, solving Problem 2.2 over $\mathbb{C}$ in polynomial time would allow us to do the same for graph 3-colorability. □

3. BILINEAR SYSTEM IS NP-HARD

We next consider some natural bilinear extensions to the quadratic feasibility problems encountered earlier, generalizing Example 3.2. The main result is Theorem 3.7, which shows that the following feasibility problem over $\mathbb{R}$ or $\mathbb{C}$ is NP-hard. In Section 6, we use this to show that certain singular value problems are also NP-hard.

Problem 3.1 (*Tensor Bilinear Feasibility*). Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Let $\mathcal{A} = [\![a_{ijk}]\!] \in \mathbb{Q}^{l \times m \times n}$, and set $A_i(j,k) = a_{ijk}$, $B_j(i,k) = a_{ijk}$, and $C_k(i,j) = a_{ijk}$ to be all the slices of $\mathcal{A}$. Decide if the following set of equations:

$$\begin{cases} \mathbf{v}^\top A_i \mathbf{w} = 0, & i = 1, \ldots, l; \\ \mathbf{u}^\top B_j \mathbf{w} = 0, & j = 1, \ldots, m; \\ \mathbf{u}^\top C_k \mathbf{v} = 0, & k = 1, \ldots, n; \end{cases} \tag{9}$$

has a solution $\mathbf{u} \in \mathbb{F}^l$, $\mathbf{v} \in \mathbb{F}^m$, $\mathbf{w} \in \mathbb{F}^n$, with all $\mathbf{u}, \mathbf{v}, \mathbf{w}$ nonzero.

Multilinear systems of equations have been studied since the early 19th century. For instance, the following result was known more than 150 years ago [Cayley 1845].

Example 3.2 (2 × 2 × 2 *hyperdeterminant*). For $\mathcal{A} = [\![a_{ijk}]\!] \in \mathbb{C}^{2\times 2\times 2}$, define

$$\mathrm{Det}_{2,2,2}(\mathcal{A}) = \frac{1}{4}\left[\det\left(\begin{bmatrix} a_{000} & a_{010} \\ a_{001} & a_{011}\end{bmatrix} + \begin{bmatrix} a_{100} & a_{110} \\ a_{101} & a_{111}\end{bmatrix}\right) - \det\left(\begin{bmatrix} a_{000} & a_{010} \\ a_{001} & a_{011}\end{bmatrix} - \begin{bmatrix} a_{100} & a_{110} \\ a_{101} & a_{111}\end{bmatrix}\right)\right]^2 - 4\det\begin{bmatrix} a_{000} & a_{010} \\ a_{001} & a_{011}\end{bmatrix}\det\begin{bmatrix} a_{100} & a_{110} \\ a_{101} & a_{111}\end{bmatrix}.$$

Given a matrix $A \in \mathbb{C}^{n\times n}$, the pair of linear equations $\mathbf{x}^\top A = \mathbf{0}, A\mathbf{y} = \mathbf{0}$ has a nontrivial solution ($\mathbf{x}, \mathbf{y}$ both nonzero) if and only if $\det(A) = 0$. Cayley proved a multilinear version that parallels the matrix case. The following system of bilinear equations:

$$\begin{aligned} a_{000}x_0y_0 + a_{010}x_0y_1 + a_{100}x_1y_0 + a_{110}x_1y_1 = 0, &\quad a_{001}x_0y_0 + a_{011}x_0y_1 + a_{101}x_1y_0 + a_{111}x_1y_1 = 0, \\ a_{000}x_0z_0 + a_{001}x_0z_1 + a_{100}x_1z_0 + a_{101}x_1z_1 = 0, &\quad a_{010}x_0z_0 + a_{011}x_0z_1 + a_{110}x_1z_0 + a_{111}x_1z_1 = 0, \\ a_{000}y_0z_0 + a_{001}y_0z_1 + a_{010}y_1z_0 + a_{011}y_1z_1 = 0, &\quad a_{100}y_0z_0 + a_{101}y_0z_1 + a_{110}y_1z_0 + a_{111}y_1z_1 = 0, \end{aligned}$$

has a nontrivial solution ($\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{C}^2$ all nonzero) if and only if $\mathrm{Det}_{2,2,2}(\mathcal{A}) = 0$. □

A remarkable result established in Gelfand et al. [1992, 1994] is that hyperdeterminants generalize to tensors of arbitrary orders, provided that certain dimension restrictions (10) are satisfied. We do not formally define the hyperdeterminant[8] here; however, it suffices to know that $\mathrm{Det}_{l,m,n}$ is a homogeneous polynomial with integer coefficients in the variables x_{ijk} where $i = 1,\dots,l$, $j = 1,\dots,m$, and $k = 1,\dots,n$. Such a polynomial defines a function $\mathrm{Det}_{l,m,n} : \mathbb{C}^{l\times m\times n} \to \mathbb{C}$ by evaluation at $\mathcal{A} = [\![a_{ijk}]\!] \in \mathbb{C}^{l\times m\times n}$, that is, setting $x_{ijk} = a_{ijk}$. The following generalizes Example 3.2.

THEOREM 3.3 (GELFAND–KAPRANOV–ZELEVINSKY). *Given a tensor $\mathcal{A} \in \mathbb{C}^{l\times m\times n}$, the hyperdeterminant $\mathrm{Det}_{l,m,n}$ is defined if and only if l, m, n satisfy:*

$$l \le m + n - 1, \quad m \le l + n - 1, \quad n \le l + m - 1. \tag{10}$$

In particular, hyperdeterminants exist when $l = m = n$. Given any $\mathcal{A} = [\![a_{ijk}]\!] \in \mathbb{C}^{l\times m\times n}$ with (10) *satisfied, the system*

$$\begin{aligned} \textstyle\sum_{j,k=1}^{m,n} a_{ijk}v_jw_k = 0 &\quad i = 1,\dots,l; \\ \textstyle\sum_{i,k=1}^{l,n} a_{ijk}u_iw_k = 0, &\quad j = 1,\dots,m; \\ \textstyle\sum_{i,j=1}^{l,m} a_{ijk}u_iv_j = 0, &\quad k = 1,\dots,n; \end{aligned} \tag{11}$$

has a nontrivial complex solution if and only if $\mathrm{Det}_{l,m,n}(\mathcal{A}) = 0$.

Remark 3.4. Condition (10) is the 3-tensor equivalent of "$m \le n$ and $n \le m$" for the existence of determinants of matrices.

[8]Roughly speaking, the hyperdeterminant is a polynomial defining the set of all tangent hyperplanes to the set of rank-1 tensors in $\mathbb{C}^{l\times m\times n}$. Gelfand et al. [1992, 1994] showed that this set is a hypersurface (i.e., defined by the vanishing of a single polynomial) if and only if the condition (10) is satisfied. Also, to be mathematically precise, these sets lie in projective space.

We shall also examine the following closely related problem. Such systems of bilinear equations have appeared in other contexts [Cohen and Tomasi 1997].

Problem 3.5 (*Triple Bilinear Feasibility*). Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Let $A_k, B_k, C_k \in \mathbb{Q}^{n\times n}$ for $k = 1,\dots,n$. Decide if the following set of equations:

$$\begin{cases} \mathbf{v}^\top A_i \mathbf{w} = 0, & i = 1,\dots,n; \\ \mathbf{u}^\top B_j \mathbf{w} = 0, & j = 1,\dots,n; \\ \mathbf{u}^\top C_k \mathbf{v} = 0, & k = 1,\dots,n; \end{cases} \tag{12}$$

has a solution $\mathbf{u},\mathbf{v},\mathbf{w} \in \mathbb{F}^n$, with all $\mathbf{u},\mathbf{v},\mathbf{w}$ nonzero.

One difference between Problem 3.5 and Problem 3.1 is that coefficient matrices A_i, B_j, C_k in (12) are allowed to be arbitrary rather than slices of a tensor $\mathcal{A}$ as in (9). Furthermore, we always assume $l = m = n$ in Problem 3.5 whereas Problem 3.1 has no such requirement.

If one could show that Problem 3.5 is NP-hard for A_i, B_j, C_k arising from $\mathcal{A} \in \mathbb{C}^{n\times n\times n}$ or that Problem 3.1 is NP-hard on the subset of problems where $\mathcal{A} \in \mathbb{C}^{l\times m\times n}$ has l, m, n satisfying (10), then deciding whether the bilinear system (11) has a nonzero solution would be NP-hard. It would follow that deciding whether the hyperdeterminant is zero is also NP-hard. Unfortunately, our proofs do not achieve either of these. The hardness of the hyperdeterminant is therefore still open (Conjecture 1.9).

Before proving Theorem 3.7, we first verify that, as in the case of quadratic feasibility, it is enough to show NP-hardness of the problem over $\mathbb{C}$.

LEMMA 3.6. *Let $\mathcal{A} \in \mathbb{R}^{l\times m\times n}$. There is a tensor $\mathcal{B} \in \mathbb{R}^{2l\times 2m\times 2n}$ such that tensor bilinear feasibility over $\mathbb{R}$ for $\mathcal{B}$ is the same as tensor bilinear feasibility over $\mathbb{C}$ for $\mathcal{A}$.*

PROOF. Let $A_i = [a_{ijk}]_{j,k=1}^{m,n}$ for $i = 1,\dots,l$. Consider the tensor $\mathcal{B} = [\![b_{ijk}]\!] \in \mathbb{R}^{2l\times 2m\times 2n}$ given by setting its slices $B_i(j,k) = b_{ijk}$ as follows:

$$B_i = \begin{bmatrix} A_i & 0 \\ 0 & -A_i \end{bmatrix},\ B_{l+i} = \begin{bmatrix} 0 & A_i \\ A_i & 0 \end{bmatrix}, \quad i = 1,\dots,l.$$

It is straightforward to check that nonzero real solutions to (9) for the tensor $\mathcal{B}$ correspond in a one-to-one manner with nonzero complex solutions to (9) for $\mathcal{A}$. □

We now come to the proof of the main theorem of this section. For the argument, we shall need the following elementary fact of linear algebra (easily proved by induction on the number of equations): a system of m homogeneous linear equations in n unknowns with $m < n$ has at least one nonzero solution.

THEOREM 3.7. *Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Graph 3-colorability is polynomially reducible to Problem 3.1 (tensor bilinear feasibility). Thus, Problem 3.1 is NP-hard.*

PROOF OF THEOREM 3.7. Given a graph $G = (V,E)$ with $v = |V|$, we shall form a tensor $\mathcal{A} = \mathcal{A}_G \in \mathbb{Z}^{l\times m\times n}$ with $l = v(2v+5)$ and $m = n = (2v+1)$ having the property that system (9) has a nonzero complex solution if and only if G has a proper 3-coloring.

Consider the following vectors of unknowns:

$$\mathbf{v} = [x_1,\dots,x_v,y_1,\dots,y_v,z]^\top \quad \text{and} \quad \mathbf{w} = [\hat{x}_1,\dots,\hat{x}_v,\hat{y}_1,\dots,\hat{y}_v,\hat{z}]^\top.$$

The 2×2 minors of the matrix formed by placing $\mathbf{v}$ and $\mathbf{w}$ side-by-side are $v(2v+1)$ quadratics, $\mathbf{v}^\top A_i \mathbf{w}$, $i = 1,\dots,v(2v+1)$, for matrices $A_i \in \mathbb{Z}^{(2v+1)\times(2v+1)}$ with entries in $\{-1,0,1\}$. By construction, these polynomials have a common nontrivial zero $\mathbf{v},\mathbf{w}$ if and only if there is $c \in \mathbb{C}$ such that $\mathbf{v} = c\mathbf{w}$. Next, we write down the $3v$ polynomials $\mathbf{v}^\top A_i \mathbf{w}$ for $i = v(2v+1)+1, \ldots, v(2v+1)+3v$ whose vanishing (along with the equations above) implies that the x_i are cube roots of unity; see (8). We also encode v equations $\mathbf{v}^\top A_i \mathbf{w}$ for $i = v(2v+4)+1, \ldots, v(2v+4)+v$ whose vanishing implies that x_i and x_j are different if $\{i, j\} \in E$. Finally, $\mathcal{A}_G = [\![a_{ijk}]\!] \in \mathbb{Z}^{l \times m \times n}$ is defined by $a_{ijk} = A_i(j, k)$.

We verify that $\mathcal{A}$ has the claimed property. Suppose that there are three nonzero complex vectors $\mathbf{u}, \mathbf{v}, \mathbf{w}$ which satisfy tensor bilinear feasibility. Then, from construction, $\mathbf{v} = c\mathbf{w}$ for some $c \neq 0$, and also $\mathbf{v}$ encodes a proper 3-coloring of the graph G by Lemma 2.10. Conversely, suppose that G is 3-colorable with a coloring represented using a vector $[x_1, \ldots, x_v]^\top \in \mathbb{C}^v$ of cube roots of unity. Then, the vectors $\mathbf{v} = \mathbf{w} = [x_1, \ldots, x_v, x_1^{-1}, \ldots, x_v^{-1}, 1]^\top$ satisfy the first set of equations in (9). The other sets of equations define a homogeneous linear system for the vector $\mathbf{u}$ consisting of $4v + 2$ equations in $l = v(2v + 5) > 4v + 2$ unknowns. In particular, there is always a nonzero $\mathbf{u}$ solving them, proving that tensor bilinear feasibility is true for $\mathcal{A}$. □

Note that the l, m, n in this construction do not satisfy (10); thus, NP-hardness of the hyperdeterminant does not follow from Theorem 3.7. We now prove the following.

THEOREM 3.8. *Problem 3.5, triple bilinear feasibility, is NP-hard over* $\mathbb{R}$.

PROOF. Since the encoding in Theorem 2.6 has more equations than unknowns, we may use Lemma 2.8 to further transform this system into an equivalent one that is square (see Example 1.4). Thus, if we could solve square quadratic feasibility ($m = n$ in Problem 2.2) over $\mathbb{R}$ in polynomial time, then we could do the same for graph 3-colorability. Using this observation, it is enough to prove that a given square, quadratic feasibility problem can be polynomially reduced to Problem 3.5.

Therefore, suppose that A_i are given $n \times n$ matrices for which we would like to determine if $\mathbf{x}^\top A_i \mathbf{x} = 0$ $(i = 1, \ldots, n)$ has a solution $0 \neq \mathbf{x} \in \mathbb{R}^n$. Let E_{ij} denote the matrix with a 1 in the (i, j) entry and 0's elsewhere. Consider a system S as in (12) in which

$$B_1 = C_1 = E_{11} \quad \text{and} \quad B_i = C_i = E_{1i} - E_{i1}, \quad \text{for } i = 2, \ldots, n.$$

Consider also changing system S by replacing B_1 and C_1 with matrices consisting of all zeroes, and call this system S'.

We shall construct a decision tree based on answers to feasibility questions involving systems having form S or S'. This will give us an algorithm to determine whether the original quadratic problem is feasible. We make two claims about solutions to S, S'.

CLAIM 1. *If S has a solution, then $u_1 = v_1 = w_1 = 0$.*

First note that $u_1 v_1 = 0$ since S has a solution. Suppose that $u_1 = 0$ and $v_1 \neq 0$. The form of the matrices C_i forces $u_2 = \cdots = u_n = 0$. But then $\mathbf{u} = 0$, which contradicts S having a solution. A similar examination with $u_1 \neq 0$ and $v_1 = 0$ proves that $u_1 = v_1 = 0$. It is now easy to see that we must also have $w_1 = 0$.

CLAIM 2. *Suppose that S has no solution. If S' has a solution, then $\mathbf{v} = c\mathbf{u}$ and $\mathbf{w} = d\mathbf{u}$ for some $0 \neq c, d \in \mathbb{R}$. Moreover, if S' has no solution, then the original quadratic problem has no solution.*

To verify Claim 2, suppose first that S' has a solution $\mathbf{u}$, $\mathbf{v}$, and $\mathbf{w}$, but S does not. In that case, we must have $u_1 \neq 0$. Also, $v_1 \neq 0$ since otherwise the third set of equations $\{u_1 v_i - u_i v_1 = 0\}_{i=2}^n$ would force $\mathbf{v} = 0$. But then $\mathbf{v} = c\mathbf{u}$ for $c = \frac{v_1}{u_1}$ and $\mathbf{w} = d\mathbf{u}$ for $d = \frac{w_1}{u_1}$ as desired. On the other hand, suppose that both S and S' have no solution. We claim that $\mathbf{x}^\top A_i \mathbf{x} = 0$ $(i = 1, \ldots, n)$ has no solution $\mathbf{x} \neq 0$ either. Indeed, if it did, then setting $\mathbf{u} = \mathbf{v} = \mathbf{w} = \mathbf{x}$, we would get a solution to S', a contradiction.

We are now prepared to give our method for solving quadratic feasibility using at most $n+2$ queries to the restricted version ($B_i = C_i$ for all i) of Problem 3.5.

First check if S has a solution. If it does not, then ask if S' has a solution. If it does not, then output "NO". This answer is correct by Claim 2. If S has no solution but S' does, then there is a solution with $\mathbf{v} = c\mathbf{u}$ and $\mathbf{w} = d\mathbf{u}$, both c and d nonzero. But then $\mathbf{x}^\top A_i \mathbf{x} = 0$ for $\mathbf{x} = \mathbf{u}$ and each i. Thus, we output "YES".

If instead, S has a solution, then the solution necessarily has $(u_1, v_1, w_1) = (0, 0, 0)$. Consider now the $n-1$-dimensional system T in which A_i becomes the lower-right $(n-1) \times (n-1)$ block of A_i, and C_i and D_i are again of the same form as the previous ones. This is a smaller system with one less unknown. We now repeat the previous examination inductively with start system T replacing S.

If we make it to the final stage of this process without outputting an answer, then the original system S has a solution with

$$u_1 = \cdots = u_{n-1} = v_1 = \cdots = v_{n-1} = w_1 = \cdots = w_{n-1} = 0 \text{ and } u_n, v_n, w_n \text{ are all nonzero.}$$

It follows that the (n, n) entry of each A_i $(i = 1 \ldots, n)$ is zero, and thus there is a nonzero solution $\mathbf{x}$ to the the original quadratic feasibility problem; so we output "YES".

We have therefore verified the algorithm terminates with the correct answer and it does so in polynomial time using an oracle that can solve Problem 3.5. □

Although in this proof, we have $l = m = n$ and thus (10) is satisfied, our choice of the coefficient matrices A_k, B_k, C_k do not arise from slices of a single tensor $\mathcal{A}$. So again, NP-hardness of deciding the vanishing of the hyperdeterminant does not follow.

4. COMBINATORIAL HYPERDETERMINANT IS NP-, #P-, AND VNP-HARD

There is another notion of hyperdetermnant, which we shall call the *combinatorial hyperdeterminant*[9] and denote by the all lowercase $\det_n$. This quantity (only nonzero for even values of d) is defined for tensors $\mathcal{A} = [a_{i_1 i_2 \cdots i_d}] \in \mathbb{C}^{n \times n \times \cdots \times n}$ by the formula:

$$\det_n(\mathcal{A}) = \sum_{\pi_2, \ldots, \pi_d \in \mathfrak{S}_n} \operatorname{sgn}(\pi_2 \cdots \pi_d) \prod_{i=1}^{n} a_{i \pi_2(i) \cdots \pi_d(i)}.$$

For $d = 2$, this definition reduces to the usual expression for the determinant of an $n \times n$ matrix. For such hyperdeterminants, we have the following hardness results from Barvinok [1995, Corollary 5.5.2] and Gurvits [2005].

THEOREM 4.1. [BARVINOK 1995]. *Let* $\mathcal{A} \in \mathbb{Z}^{n \times n \times n \times n}$. *Deciding if* $\det_n(\mathcal{A}) = 0$ *is NP-hard.*

Barvinok proved Theorem 4.1 by showing that any directed graph G may be encoded as a 4-tensor $\mathcal{A}_G$ with integer entries in such a way that the number of Hamiltonian paths between two vertices is $\det_n(\mathcal{A}_G)$. The #P-hardness follows immediately since enumerating Hamiltonian paths is a well-known #P-complete problem [Valiant 1979b].

THEOREM 4.2. [GURVITS 2005]. *Let* $\mathcal{A} \in \{0,1\}^{n \times n \times n \times n}$. *Computing* $\det_n(\mathcal{A})$ *is #P-hard.*

Theorem 4.2 is proved by showing that one may express the permament in terms of the combinatorial hyperdeterminant; the required #P-hardness then follows from the #P-completeness of the permanent [Valiant 1979a]. Even though VNP-hardness was

[9]The hyperdeterminants discussed earlier in Section 3 are then called *geometric hyperdeterminants* for distinction [Lim 2013]; these are denoted Det. The combinatorial determinants are also called *Pascal determinants* by some authors.

not discussed in Gurvits [2005], one may deduce the following result from the same argument and the VNP-completeness of Valiant [1979b].

COROLLARY 4.3. *The homogeneous polynomial* $\det_n$ *is VNP-hard to compute.*

5. TENSOR EIGENVALUE IS NP-HARD

The eigenvalues and eigenvectors of a symmetric matrix $A \in \mathbb{R}^{n\times n}$ are the stationary values and points of its Rayleigh quotient $\mathbf{x}^\top A\mathbf{x}/\mathbf{x}^\top\mathbf{x}$. Equivalently, one may consider the problem of maximizing the quadratic form $\mathbf{x}^\top A\mathbf{x}$ constrained to the unit ℓ^2-sphere:

$$\|\mathbf{x}\|_2^2 = x_1^2 + x_2^2 + \cdots + x_n^2 = 1, \tag{13}$$

which has the associated Lagrangian, $L(\mathbf{x},\lambda) = \mathbf{x}^\top A\mathbf{x} - \lambda(\|\mathbf{x}\|_2^2 - 1)$. The first-order condition, also known as the Karush–Kuhn–Tucker (KKT) condition, at a stationary point $(\lambda, \mathbf{x})$ yields the familiar eigenvalue equation $A\mathbf{x} = \lambda\mathbf{x}$, which is then used to define eigenvalue/eigenvector pairs for any (not necessarily symmetric) square matrices.

This discussion extends to give a notion of eigenvalues and eigenvectors for 3-tensors. They are suitably constrained stationary values and points of the cubic form:

$$\mathcal{A}(\mathbf{x},\mathbf{x},\mathbf{x}) = \sum_{i,j,k=1}^{n} a_{ijk}x_ix_jx_k, \tag{14}$$

associated with a tensor $\mathcal{A} \in \mathbb{R}^{n\times n\times n}$. However, one now has several natural generalizations of the constraint. One may retain (13). Alternatively, one may choose

$$\|\mathbf{x}\|_3^3 = |x_1|^3 + |x_2|^3 + \cdots + |x_n|^3 = 1 \tag{15}$$

or a unit sum-of-cubes,

$$x_1^3 + x_2^3 + \cdots + x_n^3 = 1. \tag{16}$$

Each choice has an advantage: condition (15) defines a compact set while condition (16) defines an algebraic set, and both result in scale-invariant eigenvectors. Condition (13) defines a set that is both compact and algebraic, but produces eigenvectors that are not scale-invariant. These were proposed independently in Lim [2005] and Qi [2005].

By considering the stationarity conditions of the Lagrangian, $L(\mathbf{x},\lambda) = \mathcal{A}(\mathbf{x},\mathbf{x},\mathbf{x}) - \lambda c(\mathbf{x})$, for $c(\mathbf{x})$ defined by the conditions in (13), (15), or (16), we obtain the following.

Definition 5.1. Fix $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. The number $\lambda \in \mathbb{F}$ is called an ℓ^2-*eigenvalue* of the tensor $\mathcal{A} \in \mathbb{F}^{n\times n\times n}$ and $\mathbf{0} \neq \mathbf{x} \in \mathbb{F}^n$ its corresponding ℓ^2-*eigenvector* if (3) holds. Similarly, $\lambda \in \mathbb{F}$ is an ℓ^3-*eigenvalue* and $\mathbf{0} \neq \mathbf{x} \in \mathbb{F}^n$ its ℓ^3-*eigenvector* if

$$\sum_{i,j=1}^{n} a_{ijk}x_ix_j = \lambda x_k^2, \quad k = 1,\ldots,n. \tag{17}$$

Using the tools we have developed, we prove that real tensor eigenvalue is NP-hard.

PROOF OF THEOREM 1.3. The case $\lambda = 0$ of tensor λ-eigenvalue becomes square quadratic feasibility ($m = n$ in Problem 2.2) as discussed in the proof of Theorem 3.8. Thus, deciding if $\lambda = 0$ is an eigenvalue of a tensor is NP-hard over $\mathbb{R}$ by Theorem 1.3. A similar situation holds when we use (17) to define ℓ^3-eigenpairs. □

We will see in Section 9 that the eigenvalue problem for *symmetric* 3-tensors is also NP-hard. We close this section with a proof that it is even NP-hard to approximate an eigenvector of a tensor.

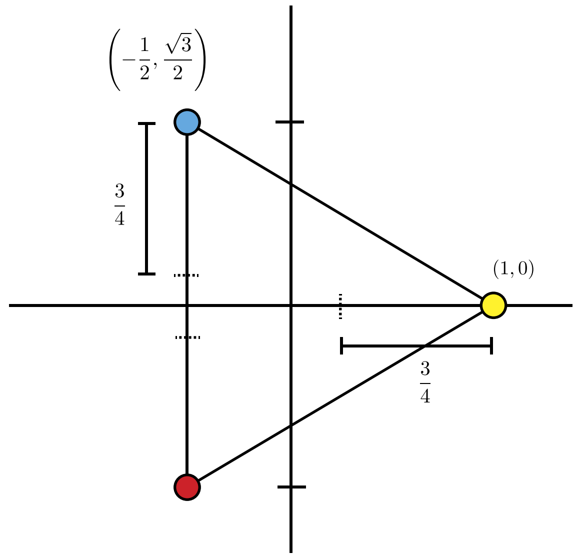

Fig. 2. It is NP-hard to approximate a real eigenvector. Each colored circle above in the complex plane represents a pair of real numbers which are coordinates of a cube root of unity. If one could approximate an eigenvector of a rational tensor to within $\varepsilon = \frac{3}{4}$ in each real coordinate, then one would be able to properly color the vertices of a 3-colorable graph G (see Example 1.4).

PROOF OF THEOREM 1.5. Suppose that one could approximate in polynomial time a tensor eigenvector with eigenvalue $\lambda = 0$ to within $\varepsilon = \frac{3}{4}$ as in (5). By the discussion in Section 2, given a graph G, we can form a square set of polynomial equations (see Example 1.4), having eigenvectors of a rational tensor $\mathcal{A}_G$ as solutions, encoding proper 3-colorings of G. Since such vectors represent cube roots of unity separated by a distance of at least $\frac{3}{2}$ in each real or imaginary part (see Figure 2), finding an approximate real eigenvector to within $\varepsilon = \frac{3}{4}$ of an actual one in polynomial time would allow one to also decide graph 3-colorability in polynomial time. □

6. TENSOR SINGULAR VALUE AND SPECTRAL NORM ARE NP-HARD

It is easy to verify that the singular values and singular vectors of a matrix $A \in \mathbb{R}^{m \times n}$ are the stationary values and stationary points of the quotient $\mathbf{x}^\top A\mathbf{y}/\|\mathbf{x}\|_2\|\mathbf{y}\|_2$. Indeed, the associated Lagrangian is

$$L(\mathbf{x}, \mathbf{y}, \sigma) = \mathbf{x}^\top A\mathbf{y} - \sigma(\|\mathbf{x}\|_2\|\mathbf{y}\|_2 - 1), \tag{18}$$

and the first-order condition yields, at a stationary point $(\mathbf{x}, \mathbf{y})$, the familiar singular value equations:

$$A\mathbf{v} = \sigma\mathbf{u}, \quad A^\top\mathbf{u} = \sigma\mathbf{v},$$

where $\mathbf{u} = \mathbf{x}/\|\mathbf{x}\|_2$ and $\mathbf{v} = \mathbf{y}/\|\mathbf{y}\|_2$.

This derivation has been extended to define singular values and singular vectors for higher-order tensors [Lim 2005]. For $\mathcal{A} \in \mathbb{R}^{l \times m \times n}$, we have the trilinear form[10]:

$$\mathcal{A}(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \sum_{i,j,k=1}^{n} a_{ijk} x_i y_j z_k, \tag{19}$$

[10]When $l = m = n$ and $\mathbf{x} = \mathbf{y} = \mathbf{z}$, the trilinear form in (19) becomes the cubic form in (14).

and consideration of its stationary values on a product of unit ℓ^p-spheres leads to the Lagrangian,

$$L(\mathbf{x}, \mathbf{y}, \mathbf{z}, \sigma) = \mathcal{A}(\mathbf{x}, \mathbf{y}, \mathbf{z}) - \sigma(\|\mathbf{x}\|_p \|\mathbf{y}\|_p \|\mathbf{z}\|_p - 1).$$

The only ambiguity is choice of p. As for eigenvalues, natural choices are $p = 2$ or 3.

Definition 6.1. Fix $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Let $\sigma \in \mathbb{F}$, and suppose that $\mathbf{u} \in \mathbb{F}^l$, $\mathbf{v} \in \mathbb{F}^m$, and $\mathbf{w} \in \mathbb{F}^n$ are all nonzero. The number $\sigma \in \mathbb{F}$ is called an ℓ^2-*singular value* and the nonzero $\mathbf{u}, \mathbf{v}, \mathbf{w}$ are called ℓ^2-*singular vectors* of $\mathcal{A}$ if

$$\begin{aligned} \sum\nolimits_{j,k=1}^{m,n} a_{ijk} v_j w_k &= \sigma u_i, \quad i = 1, \ldots, l; \\ \sum\nolimits_{i,k=1}^{l,n} a_{ijk} u_i w_k &= \sigma v_j, \quad j = 1, \ldots, m; \\ \sum\nolimits_{i,j=1}^{l,m} a_{ijk} u_i v_j &= \sigma w_k, \quad k = 1, \ldots, n. \end{aligned} \tag{20}$$

Similarly, σ is called an ℓ^3-*singular value* and nonzero $\mathbf{u}, \mathbf{v}, \mathbf{w}$ ℓ^3-*singular vectors* if

$$\begin{aligned} \sum\nolimits_{j,k=1}^{m,n} a_{ijk} v_j w_k &= \sigma u_i^2, \quad i = 1, \ldots, l; \\ \sum\nolimits_{i,k=1}^{l,n} a_{ijk} u_i w_k &= \sigma v_j^2, \quad j = 1, \ldots, m; \\ \sum\nolimits_{i,j=1}^{l,m} a_{ijk} u_i v_j &= \sigma w_k^2, \quad k = 1, \ldots, n. \end{aligned} \tag{21}$$

When $\sigma = 0$, definitions (20) and (21) agree and reduce to tensor bilinear feasibility (Problem 3.1). In particular, if condition (10) holds, then $\operatorname{Det}_{l,m,n}(\mathcal{A}) = 0$ iff 0 is an ℓ^2-singular value of $\mathcal{A}$ iff 0 is an ℓ^3-singular value of $\mathcal{A}$ [Lim 2005].

The following is immediate from Theorem 3.7, which was proved by a reduction from 3-colorability.

THEOREM 6.2. *Let $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Deciding whether $\sigma = 0$ is an (ℓ^2 or ℓ^3) singular value over $\mathbb{F}$ of a tensor is NP-hard.*

The tools we developed in Section 3 also directly apply to give an analogue of Theorem 1.5 for approximating singular vectors corresponding to singular value $\sigma = 0$.

THEOREM 6.3. *It is NP-hard to approximate a triple of tensor singular vectors over $\mathbb{R}$ to within $\varepsilon = \frac{3}{4}$ and over $\mathbb{C}$ to within $\varepsilon = \frac{\sqrt{3}}{2}$.*

COROLLARY 6.4. *Unless P = NP, there is no PTAS for approximating tensor singular vectors.*

Note that verifying whether $0 \neq \sigma \in \mathbb{Q}$ is a singular value of $\mathcal{A}$ is the same as checking whether 1 is a singular value of $\mathcal{A}/\sigma$. In this section, we reduce computing the max-clique number of a graph to a singular value problem for $\sigma = 1$, extending some ideas of Nesterov [2003] and He et al. [2010]. In particular, we shall prove the following.

THEOREM 6.5. *Fix $0 \neq \sigma \in \mathbb{Q}$. Deciding whether σ is an ℓ^2-singular value over $\mathbb{R}$ of a tensor is NP-hard.*

We next define the closely related concept of spectral norm of a tensor.

Definition 6.6. The *spectral norm* of a tensor $\mathcal{A}$ is

$$\|\mathcal{A}\|_{2,2,2} = \sup_{\mathbf{x},\mathbf{y},\mathbf{z}\neq\mathbf{0}} \frac{|\mathcal{A}(\mathbf{x},\mathbf{y},\mathbf{z})|}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2\|\mathbf{z}\|_2}.$$

The spectral norm is either the maximum or minimum value of $\mathcal{A}(\mathbf{x},\mathbf{y},\mathbf{z})$ constrained to the set $\{(\mathbf{x},\mathbf{y},\mathbf{z}) : \|\mathbf{x}\|_2 = \|\mathbf{y}\|_2 = \|\mathbf{z}\|_2 = 1\}$, and thus is an ℓ^2-singular value of $\mathcal{A}$. At the end of this section, we will show that the corresponding spectral norm questions are NP-hard (Theorems 1.10 and 1.11).

We now explain our setup for the proof of Theorem 6.5. Let $G = (V, E)$ be a simple graph on vertices $V = \{1, \dots, v\}$ with e edges E, and let $\omega(G)$ be the *clique number* of G (that is, the number of vertices in a largest clique). Given a graph G and $l \in \mathbb{N}$, deciding whether $\omega(G) \geq l$ is one of the first decision problems known to be NP-complete [Karp 1972]. An important result linking an optimization problem to $\omega(G)$ is the following classical theorem [Motzkin and Straus 1965]. It can be used to give an elegant proof of Turán's Graph Theorem, which bounds the number of edges in a graph in terms of its clique number (e.g., see Aigner [1995]).

THEOREM 6.7 (MOTZKIN–STRAUS). *Let $\Delta_v = \{(x_1, \dots, x_v) \in \mathbb{R}^v_{\geq 0} : \sum_{i=1}^v x_i = 1\}$ and let $G = (V, E)$ be a graph on v vertices with clique number $\omega(G)$. Then,*

$$1 - \frac{1}{\omega(G)} = 2 \cdot \max_{\mathbf{x}\in\Delta_v} \sum\nolimits_{\{i,j\}\in E} x_i x_j.$$

Let A_G be the adjacency matrix of the graph G and set $\omega = \omega(G)$. For each positive integer l, define $Q_l = A_G + \frac{1}{l}J$, in which J is the all-ones matrix. Also, let

$$M_l = \max_{\mathbf{x}\in\Delta_v} \mathbf{x}^\top Q_l \mathbf{x} = 1 + \frac{\omega - l}{l\omega},$$

where the second equality follows from Theorem 6.7. We have $M_\omega = 1$ and also

$$M_l > 1 \text{ if } l < \omega; \quad M_l < 1 \text{ if } l > \omega. \tag{22}$$

For $k = 1, \dots, e$, let $E_k = \frac{1}{2}E_{i_k j_k} + \frac{1}{2}E_{j_k i_k}$ in which $\{i_k, j_k\}$ is the kth edge of G. Here, the $v \times v$ matrix E_{ij} has a 1 in the (i,j)-th spot and zeroes elsewhere. For each positive integer l, consider the following optimization problem (having rational input):

$$N_l = \max_{\|\mathbf{u}\|_2=1} \left\{ \sum_{i=1}^{l} \left(\mathbf{u}^\top \frac{1}{l} I \mathbf{u}\right)^2 + 2\sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{u})^2 \right\}.$$

LEMMA 6.8. *For any graph G, we have $M_l = N_l$.*

PROOF. By construction, $N_l = \frac{1}{l} + 2 \cdot \max_{\|\mathbf{u}\|_2=1} \sum_{\{i,j\}\in E} u_i^2 u_j^2$, which is easily seen to equal M_l. □

We next state a beautiful result of Banach [Banach 1938; Pappas et al. 2007] that will be very useful for us here as well as in Section 10. The result essentially says that the spectral norm of a symmetric tensor may be expressed symmetrically.

THEOREM 6.9 (BANACH). *Let $\mathcal{S} \in \mathbb{R}^{n\times n\times n}$ be a symmetric 3-tensor. Then*

$$\|\mathcal{S}\|_{2,2,2} = \sup_{\mathbf{x},\mathbf{y},\mathbf{z}\neq\mathbf{0}} \frac{|\mathcal{S}(\mathbf{x},\mathbf{y},\mathbf{z})|}{\|\mathbf{x}\|_2\|\mathbf{y}\|_2\|\mathbf{z}\|_2} = \sup_{\mathbf{x}\neq\mathbf{0}} \frac{|\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x})|}{\|\mathbf{x}\|_2^3}. \tag{23}$$

Let $\mathcal{S} \in \mathbb{R}^{n\times n\times n\times n}$ be a symmetric 4-tensor. Then

$$\|\mathcal{S}\|_{2,2,2,2} = \sup_{\mathbf{w},\mathbf{x},\mathbf{y},\mathbf{z}\neq\mathbf{0}} \frac{|\mathcal{S}(\mathbf{w},\mathbf{x},\mathbf{y},\mathbf{z})|}{\|\mathbf{w}\|_2\|\mathbf{x}\|_2\|\mathbf{y}\|_2\|\mathbf{z}\|_2} = \sup_{\mathbf{x}\neq\mathbf{0}} \frac{|\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})|}{\|\mathbf{x}\|_2^4}. \tag{24}$$

While we have restricted ourselves to orders 3 and 4 for simplicity, Banach's result holds for arbitrary order. Furthermore, $\mathbb{C}$ may replace $\mathbb{R}$ without affecting its validity.

The following interesting fact, which is embedded in the proof of He et al. [2010, Proposition 2], may be easily deduced from Theorem 6.9.

PROPOSITION 6.10 (HE–LI–ZHANG). *Let $A_1,\ldots,A_m \in \mathbb{R}^{n\times n}$ be symmetric. Then,*

$$\max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=1} \sum_{k=1}^{m} (\mathbf{u}^\top A_k \mathbf{v})^2 = \max_{\|\mathbf{v}\|_2=1} \sum_{k=1}^{m} (\mathbf{v}^\top A_k \mathbf{v})^2. \tag{25}$$

PROOF. Define

$$f(\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{x}) = \sum_{k=1}^{m} (\mathbf{u}^\top A_k \mathbf{v})(\mathbf{w}^\top A_k \mathbf{x}).$$

Clearly, we must have

$$\max_{\|\mathbf{v}\|_2=1} f(\mathbf{v},\mathbf{v},\mathbf{v},\mathbf{v}) \le \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=1} f(\mathbf{u},\mathbf{v},\mathbf{u},\mathbf{v}) \le \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=\|\mathbf{x}\|_2=1} f(\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{x}).$$

Note that we may write $f(\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{x}) = \mathcal{S}(\mathbf{u},\mathbf{v},\mathbf{w},\mathbf{x})$ for some symmetric 4-tensor $\mathcal{S} \in \mathbb{R}^{n\times n\times n\times n}$. Since the first and last terms in this inequality are equal by (24) in Banach's theorem, we obtain (25). □

LEMMA 6.11. *The maximization problem*

$$T_l = \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1} \left\{ \sum_{i=1}^{l} \left(\mathbf{u}^\top \frac{1}{l} I \mathbf{v}\right) w_i + \sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{v}) w_{l+k} + \sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{v}) w_{m+l+k} \right\} \tag{26}$$

has optimum value $T_l = M_l^{1/2}$. Thus,

$$T_l = 1 \quad \textit{iff} \quad l = \omega; \quad T_l > 1 \quad \textit{iff} \quad l < \omega; \quad \textit{and} \quad T_l < 1 \quad \textit{iff} \quad l > \omega,$$

PROOF. Fixing $\mathbf{a} = [a_1,\ldots,a_s]^\top \in \mathbb{R}^s$, the Cauchy-Schwarz inequality implies that a sum $\sum_{i=1}^{s} a_i w_i$ with $\|\mathbf{w}\|_2 = 1$ achieves a maximum value of $\|\mathbf{a}\|_2$ (with $w_i = a_i/\|\mathbf{a}\|_2$ if $\|\mathbf{a}\|_2 \neq 0$). Thus,

$$\begin{aligned} T_l &= \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1} \sum_{i=1}^{l} \left(\mathbf{u}^\top \frac{1}{l} I \mathbf{v}\right) w_i + \sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{v}) w_{l+k} + \sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{v}) w_{e+l+k} \\ &= \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=1} \sqrt{\sum_{i=1}^{l} \left(\mathbf{u}^\top \frac{1}{l} I \mathbf{v}\right)^2 + 2\sum_{k=1}^{e} (\mathbf{u}^\top E_k \mathbf{v})^2} \\ &= M_l^{1/2}, \end{aligned}$$

where the last equality follows from Lemma 6.8 and Proposition 6.10. □

We can now prove Theorem 6.5, and Theorems 1.10 and 1.11 from the introduction.

PROOF OF THEOREM 6.5. We cast (26) in the form of a tensor singular value problem. Set $\mathcal{A}_l$ to be the three dimensional tensor with a_{ijk} equal to the coefficient of the term $u_i v_j w_k$ in the multilinear form (26). Then T_l is just the maximum ℓ^2-singular value of $\mathcal{A}_l$. We now show that if we could decide whether $\sigma = 1$ is an ℓ^2-singular value of $\mathcal{A}_l$, then we would solve the max-clique problem.

Given a graph G, construct the tensor $\mathcal{A}_l$ for each integer $l \in \{1, \ldots, v\}$. The largest value of l for which 1 is a singular value of $\mathcal{A}_l$ is $\omega = \omega(G)$. To see this, notice that if l is larger than ω, the maximum singular value of $\mathcal{A}_l$ is smaller than 1 by Lemma 6.11. Therefore, $\sigma = 1$ can not be a singular value of $\mathcal{A}_l$ in these cases. However, $\sigma = 1$ is a singular value of the tensor $\mathcal{A}_\omega$. □

PROOF OF THEOREM 1.10. In this reduction, used to prove Theorem 6.5, it suffices to decide which tensor $\mathcal{A}_l$ has spectral norm equal to 1. □

PROOF OF THEOREM 1.11. Suppose that we could approximate spectral norm to within a factor of $1 - \varepsilon = (1 + 1/N(N-1))^{-1/2}$, where N is tensor input size. Consider the tensors $\mathcal{A}_l$ as in the proof of Theorems 6.5 and 1.10, which have input size N that is at least the number of vertices v of the graph G. For each l, we are guaranteed an approximation for the spectral norm of $\mathcal{A}_l$ of at least

$$(1-\varepsilon) \cdot M_l^{1/2} > \left(1 + \frac{1}{v(v-1)}\right)^{-1/2} \left(1 + \frac{\omega - l}{l\omega}\right)^{1/2}. \tag{27}$$

It is easy to verify that (27) implies that any spectral norm approximation of $\mathcal{A}_l$ is greater than 1 whenever $l \leq \omega - 1$. In particular, as $\mathcal{A}_\omega$ has spectral norm exactly 1, we can determine ω by finding the largest $l = 1, \ldots, v$ for which a spectral norm approximation of $\mathcal{A}_l$ is 1 or less. □

7. BEST RANK-1 TENSOR APPROXIMATION IS NP-HARD

We shall need to define the *Frobenius norm* and *inner product* for this and later sections. Let $\mathcal{A} = [\![a_{ijk}]\!]_{i,j,k=1}^{l,m,n}$ and $\mathcal{B} = [\![b_{ijk}]\!]_{i,j,k=1}^{l,m,n} \in \mathbb{R}^{n\times n\times n}$. Then we define:

$$\|\mathcal{A}\|_F^2 = \sum_{i,j,k=1}^{l,m,n} |a_{ijk}|^2, \qquad \langle \mathcal{A}, \mathcal{B}\rangle = \sum_{i,j,k=1}^{l,m,n} a_{ijk}b_{ijk}.$$

Clearly $\|\mathcal{A}\|_F^2 = \langle \mathcal{A}, \mathcal{A}\rangle$ and $\langle \mathcal{A}, \mathbf{x}\otimes\mathbf{y}\otimes\mathbf{z}\rangle = \mathcal{A}(\mathbf{x},\mathbf{y},\mathbf{z})$, where $\mathcal{A}(\mathbf{x},\mathbf{y},\mathbf{z})$ is as in (19).

As we explain next, the best rank-r approximation problem for a tensor is well-defined only when $r = 1$. The general problem can be expressed as solving:

$$\min_{\mathbf{x}_i,\mathbf{y}_i,\mathbf{z}_i} \|\mathcal{A} - \lambda_1\mathbf{x}_1\otimes\mathbf{y}_1\otimes\mathbf{z}_1 - \cdots - \lambda_r\mathbf{x}_r\otimes\mathbf{y}_r\otimes\mathbf{z}_r\|_F.$$

Unfortunately, a solution to this optimization problem does not necessarily exist; in fact, the set $\{\mathcal{A} \in \mathbb{R}^{l\times m\times n} : \operatorname{rank}_{\mathbb{R}}(\mathcal{A}) \leq r\}$ is not closed, in general, when $r > 1$. The following simple example is based on an exercise in Knuth [1998].

Example 7.1. Let $\mathbf{x}_i, \mathbf{y}_i \in \mathbb{R}^m$, $i = 1, 2, 3$. Let

$$\mathcal{A} = \mathbf{x}_1\otimes\mathbf{x}_2\otimes\mathbf{y}_3 + \mathbf{x}_1\otimes\mathbf{y}_2\otimes\mathbf{x}_3 + \mathbf{y}_1\otimes\mathbf{x}_2\otimes\mathbf{x}_3,$$

and for $n \in \mathbb{N}$, let

$$\mathcal{A}_n = \mathbf{x}_1\otimes\mathbf{x}_2\otimes(\mathbf{y}_3 - n\mathbf{x}_3) + \left(\mathbf{x}_1 + \frac{1}{n}\mathbf{y}_1\right)\otimes\left(\mathbf{x}_2 + \frac{1}{n}\mathbf{y}_2\right)\otimes n\mathbf{x}_3.$$

One can show that $\operatorname{rank}_{\mathbb{R}}(\mathcal{A}) = 3$ if and only if the pair $\mathbf{x}_i, \mathbf{y}_i$ are linearly independent, $i = 1, 2, 3$. Since $\operatorname{rank}_{\mathbb{F}}(\mathcal{A}_n) \leq 2$ and

$$\lim_{n\to\infty} \mathcal{A}_n = \mathcal{A},$$

the rank-3 tensor $\mathcal{A}$ has no best rank-2 approximation. □

The phenomenon of a tensor failing to have a best rank-r approximation is widespread, occurring over a range of dimensions, orders, and ranks, regardless of the norm (or Brègman divergence) used. These counterexamples occur with positive probability and sometimes with certainty (in $\mathbb{R}^{2\times 2\times 2}$, no tensor of rank-3 has a best rank-2 approximation). We refer the reader to De Silva and Lim [2008] for further details.

On the other hand, the set of rank-1 tensors (together with zero) is closed. In fact, it is the Segre variety in classical algebraic geometry [Landsberg 2012]. Consider the problem of finding the best rank-1 approximation to a tensor $\mathcal{A}$:

$$\min_{\mathbf{x},\mathbf{y},\mathbf{z}} \|\mathcal{A} - \mathbf{x}\otimes\mathbf{y}\otimes\mathbf{z}\|_F. \tag{28}$$

By introducing an additional parameter $\sigma \geq 0$, we may rewrite the rank-1 term in the form $\mathbf{x}\otimes\mathbf{y}\otimes\mathbf{z} = \sigma\mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}$ where $\|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = \|\mathbf{w}\|_2 = 1$. Then,

$$\begin{aligned}\|\mathcal{A} - \sigma\mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\|_F^2 &= \|\mathcal{A}\|_F^2 - 2\sigma\langle\mathcal{A}, \mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\rangle + \sigma^2\|\mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\|_F^2 \\ &= \|\mathcal{A}\|_F^2 - 2\sigma\langle\mathcal{A}, \mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\rangle + \sigma^2.\end{aligned}$$

This expression is minimized when

$$\sigma = \max_{\|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1} \langle\mathcal{A}, \mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\rangle = \|\mathcal{A}\|_{2,2,2}$$

since $\langle\mathcal{A}, \mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\rangle = \mathcal{A}(\mathbf{u},\mathbf{v},\mathbf{w})$. If $(\mathbf{x},\mathbf{y},\mathbf{z})$ is a solution to the optimization problem (28), then σ may be computed as

$$\sigma = \|\sigma\mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\|_F = \|\mathbf{x}\otimes\mathbf{y}\otimes\mathbf{z}\|_F = \|\mathbf{x}\|_2\|\mathbf{y}\|_2\|\mathbf{z}\|_2.$$

We conclude that determining the best rank-1 approximation is also NP-hard, which is Theorem 1.13 from the introduction. We will see in Section 10 that restricting to symmetric 3-tensors does not make the best rank-1 approximation problem easier.

8. TENSOR RANK IS NP-HARD

It was shown in Håstad [1990] that any 3SAT Boolean formula[11] can be encoded as a 3-tensor $\mathcal{A}$ over a finite field or $\mathbb{Q}$ and that the satisfiability of the formula is equivalent to checking whether $\operatorname{rank}(\mathcal{A}) \leq r$ for some r that depends on the number of variables and clauses (the tensor rank being taken over the respective field). In particular, tensor rank is NP-hard over $\mathbb{Q}$ and NP-complete over finite fields.

Since the majority of recent applications of tensor methods are over $\mathbb{R}$ and $\mathbb{C}$, a natural question is whether tensor rank is also NP-hard over these fields. In other words, is it NP-hard to decide whether $\operatorname{rank}_{\mathbb{R}}(\mathcal{A}) \leq r$ or if $\operatorname{rank}_{\mathbb{C}}(\mathcal{A}) \leq r$ for a given tensor $\mathcal{A}$ with rational entries and a given $r \in \mathbb{N}$?

One difficulty with the notion of tensor rank is that it depends on the base field. For instance, there are real tensors with rank over $\mathbb{C}$ strictly less than their rank over $\mathbb{R}$ [De Silva and Lim 2008]. We will show here that the same can happen for tensors with rational entries. In particular, Håstad's result for tensor rank over $\mathbb{Q}$ does not directly apply to $\mathbb{R}$ and $\mathbb{C}$. Nevertheless, Håstad's proof shows, as we explain in Theorem 8.2, that tensor rank remains NP-hard over both $\mathbb{R}$ and $\mathbb{C}$.

[11] Recall that this a Boolean formula in n variables and m clauses where each clause contains exactly three variables, for example, $(x_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (x_1 \vee x_2 \vee x_4)$.

PROOF OF THEOREM 1.14. We explicitly construct a rational tensor $\mathcal{A}$ with $\operatorname{rank}_{\mathbb{R}}(\mathcal{A}) < \operatorname{rank}_{\mathbb{Q}}(\mathcal{A})$. Let $\mathbf{x} = [1,0]^\top$ and $\mathbf{y} = [0,1]^\top$. First observe that

$$\overline{\mathbf{z}} \otimes \mathbf{z} \otimes \overline{\mathbf{z}} + \mathbf{z} \otimes \overline{\mathbf{z}} \otimes \mathbf{z} = 2\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x} - 4\mathbf{y} \otimes \mathbf{y} \otimes \mathbf{x} + 4\mathbf{y} \otimes \mathbf{x} \otimes \mathbf{y} - 4\mathbf{x} \otimes \mathbf{y} \otimes \mathbf{y} \in \mathbb{Q}^{2\times 2\times 2},$$

where $\mathbf{z} = \mathbf{x} + \sqrt{2}\mathbf{y}$ and $\overline{\mathbf{z}} = \mathbf{x} - \sqrt{2}\mathbf{y}$. Let $\mathcal{A}$ be this tensor; thus, $\operatorname{rank}_{\mathbb{R}}(\mathcal{A}) \le 2$. We claim that $\operatorname{rank}_{\mathbb{Q}}(\mathcal{A}) > 2$. Suppose not and that there exist $\mathbf{u}_i = [a_i, b_i]^\top$, $\mathbf{v}_i = [c_i, d_i]^\top \in \mathbb{Q}^2$, $i = 1, 2, 3$, with

$$\mathcal{A} = \mathbf{u}_1 \otimes \mathbf{u}_2 \otimes \mathbf{u}_3 + \mathbf{v}_1 \otimes \mathbf{v}_2 \otimes \mathbf{v}_3. \tag{29}$$

Identity (29) gives eight equations found in (30). Thus, by Lemma 8.1, $\operatorname{rank}_{\mathbb{Q}}(\mathcal{A}) > 2$. □

LEMMA 8.1. *The system of* 8 *equations in* 12 *unknowns:*

$$\begin{aligned} &a_1a_2a_3 + c_1c_2c_3 = 2,\ a_1a_3b_2 + c_1c_3d_2 = 0,\ a_2a_3b_1 + c_2c_3d_1 = 0,\\ &a_3b_1b_2 + c_3d_1d_2 = -4,\ a_1a_2b_3 + c_1c_2d_3 = 0,\ a_1b_2b_3 + c_1d_2d_3 = -4,\\ &a_2b_1b_3 + c_2d_3d_1 = 4,\ b_1b_2b_3 + d_1d_2d_3 = 0 \end{aligned} \tag{30}$$

has no solution in rational numbers a_1, a_2, a_3, b_1, b_2, b_3, c_1, c_2, c_3, and d_1, d_2, d_3.

PROOF. One may verify in exact symbolic arithmetic (see the Appendix) that the following two equations are polynomial consequences of (30):

$$2c_2^2 - d_2^2 = 0 \quad \text{and} \quad c_1d_2d_3 - 2 = 0.$$

Since no rational number when squared equals 2, the first equation implies that any rational solution to (30) must have $c_2 = d_2 = 0$, an impossibility by the second. Thus, no rational solutions to (30) exist. □

We now provide an addendum to Håstad's result.

THEOREM 8.2. *Tensor rank is NP-hard over fields $\mathbb{F} \supseteq \mathbb{Q}$; in particular, over $\mathbb{R}$, $\mathbb{C}$.*

PROOF. Håstad [1990] contains a recipe for encoding any given 3SAT Boolean formula in n variables and m clauses as a tensor $\mathcal{A} \in \mathbb{Q}^{(n+2m+2)\times 3n\times(3n+m)}$ with the property that the 3SAT formula is satisfiable if and only if $\operatorname{rank}_{\mathbb{F}}(\mathcal{A}) \le 4n + 2m$. The recipe defines $(n + 2m + 2) \times 3n$ matrices for $i = 1, \dots, n, j = 1, \dots, m$:

— V_i: 1 in $(1, 2i - 1)$ and $(2, 2i)$, 0 elsewhere;
— S_i: 1 in $(1, 2n + i)$, 0 elsewhere;
— M_i: 1 in $(1, 2i - 1)$, $(2 + i, 2i)$, and $(2 + i, 2n + i)$, 0 elsewhere;
— C_j: depends on jth clause (more involved) and has entries $0, \pm 1$;

and the 3-tensor

$$\mathcal{A} = [V_1, \dots, V_n, S_1, \dots, S_n, M_1, \dots, M_n, C_1, \dots, C_m] \in \mathbb{Q}^{(n+2m+2)\times 3n\times(3n+m)}.$$

Observe that the matrices V_i, S_i, M_i, C_j are defined with $-1, 0, 1$ and that the argument in Håstad [1990] uses only the field axioms. In particular, it holds for any $\mathbb{F} \supseteq \mathbb{Q}$. □

9. SYMMETRIC TENSOR EIGENVALUE IS NP-HARD

It is natural to ask if the eigenvalue problem remains NP-hard if the general tensor in (3) or (17) is replaced by a symmetric one.

Problem 9.1 (*Symmetric eigenvalue*). Given a symmetric tensor $\mathcal{S} \in \mathbb{Q}^{n\times n\times n}$ and $d \in \mathbb{Q}$, decide if $\lambda \in \mathbb{Q}(\sqrt{d})$ is an eigenvalue with (3) or (17) for some $\mathbf{0} \ne \mathbf{x} \in \mathbb{R}^n$.

As will become clear later, inputs λ in this problem may take values in $\mathbb{Q}(\sqrt{d}) = \{a + b\sqrt{d} : a, b \in \mathbb{Q}\}$ for any particular $d \in \mathbb{Q}$. This is not a problem since such numbers can be represented by rationals (a, b, d) and arithmetic in $\mathbb{Q}(\sqrt{d})$ is rational arithmetic.

Let $G = (V, E)$ be a simple graph with vertices $V = \{1, \dots, v\}$ and edges E. A subset of vertices $S \subseteq V$ is said to be *stable* (or *independent*) if $\{i, j\} \notin E$ for all $i, j \in S$, and the *stability number* $\alpha(G)$ is defined to be the size of a largest stable set. This quantity is closely related to the clique number that we encountered in Section 6; namely, $\alpha(G) = \omega(\overline{G})$, where $\overline{G}$ is the dual graph of G. Nesterov has used the Motzkin–Straus Theorem to prove an analogue for the stability number [De Klerk 2008; Nesterov 2003].[12]

THEOREM 9.2 (NESTEROV). *Let $G = (V, E)$ on v vertices have stability number $\alpha(G)$. Let $n = v + \frac{v(v-1)}{2}$ and $\mathbb{S}^{n-1} = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^v \times \mathbb{R}^{v(v-1)/2} : \|\mathbf{x}\|_2^2 + \|\mathbf{y}\|_2^2 = 1\}$. Then,*

$$\sqrt{1 - \frac{1}{\alpha(G)}} = 3\sqrt{\frac{3}{2}} \cdot \max_{(\mathbf{x},\mathbf{y}) \in \mathbb{S}^{n-1}} \sum\nolimits_{i<j,\, \{i,j\} \notin E} x_i x_j y_{ij}. \tag{31}$$

We will deduce the NP-hardness of symmetric tensor eigenvalue from the observation that every homogeneous cubic polynomial corresponds to a symmetric 3-tensor whose maximum eigenvalue is the maximum on the right-hand side of (31).

For any $1 \le i < j < k \le v$, let

$$s_{ijk} = \begin{cases} 1 & 1 \le i < j \le v,\ k = v + \varphi(i, j),\ \{i, j\} \notin E, \\ 0 & \text{otherwise}, \end{cases}$$

where $\varphi(i, j) = (i-1)v - i(i-1)/2 + j - i$ is a lexicographical enumeration of the $v(v-1)/2$ pairs $i < j$. For the other cases, $i < k < j, \dots, k < j < i$, we set

$$s_{ijk} = s_{ikj} = s_{jik} = s_{jki} = s_{kij} = s_{kji}.$$

Also, whenever two or more indices are equal, we put $s_{ijk} = 0$. This defines a symmetric tensor $\mathcal{S} = [\![s_{ijk}]\!] \in \mathbb{R}^{n \times n \times n}$ with the property that

$$\mathcal{S}(\mathbf{z}, \mathbf{z}, \mathbf{z}) = 6 \sum\nolimits_{i<j,\, \{i,j\} \notin E} x_i x_j y_{ij},$$

where $\mathbf{z} = (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^v \times \mathbb{R}^{v(v-1)/2} = \mathbb{R}^n$.

Since $\lambda = \max_{\|\mathbf{z}\|_2 = 1} \mathcal{S}(\mathbf{z}, \mathbf{z}, \mathbf{z})$ is necessarily a stationary value of $\mathcal{S}(\mathbf{z}, \mathbf{z}, \mathbf{z})$ constrained to $\|\mathbf{z}\|_2 = 1$, it is an ℓ^2-eigenvalue of $\mathcal{S}$. Moreover, Nesterov's Theorem implies

$$\lambda = 2\sqrt{\frac{2}{3}\left(1 - \frac{1}{\alpha(G)}\right)}.$$

Given a graph G and $l \in \mathbb{N}$, deciding whether $\alpha(G) = l$ is equivalent to deciding whether $\omega(\overline{G}) = l$. Hence, the former is an NP-complete problem given that the latter is an NP-complete problem [Karp 1972], and we are led to the following:

THEOREM 9.3. *Symmetric tensor eigenvalue over $\mathbb{R}$ is NP-hard.*

PROOF. For $l = v, \dots, 1$, we check whether $\lambda_l = 2\sqrt{\frac{2}{3}\left(1 - \frac{1}{l}\right)}$ is an ℓ^2-eigenvalue of $\mathcal{S}$. Since $\alpha(G) \in \{1, \dots, v\}$, at most v answers to Problem 9.1 with inputs $\lambda_v, \dots, \lambda_1$

[12]We caution the reader that the equivalent of (31) in Nesterov [2003] is missing a factor of $1/\sqrt{2}$; the mistake was reproduced in De Klerk [2008].

(taken in decreasing order of magnitude so that the first eigenvalue identified would be the maximum) would reveal its value. Hence, Problem 9.1 is NP-hard. □

Remark 9.4. Here we have implicitly used the assumption that inputs to the symmetric tensor eigenvalue decision problem are allowed to be quadratic irrationalities of the form $2\sqrt{\frac{2}{3}\left(1-\frac{1}{l}\right)}$ for each integer $l \in \{1, \ldots, v\}$.

It is also known that $\alpha(G)$ is NP-hard to approximate[13] [Håstad 1999; Zuckerman 2006] (see also the survey [De Klerk 2008]).

THEOREM 9.5 (HÅSTAD, ZUCKERMAN). *It is impossible to approximate $\alpha(G)$ in polynomial time to within a factor of $v^{1-\varepsilon}$ for any $\varepsilon > 0$, unless P = NP.*

Theorem 9.5 implies the following inapproximability result for symmetric tensors.

COROLLARY 9.6. *Unless P = NP, there is no FPTAS for approximating the largest ℓ^2-eigenvalue of a real symmetric tensor.*

10. SYMMETRIC SINGULAR VALUE, SPECTRAL NORM, AND RANK-1 APPROXIMATION ARE NP-HARD

We will deduce from Theorem 9.3 and Corollary 9.6 a series of hardness results for symmetric tensors parallel to earlier ones for the nonsymmetric case.

We first state Theorem 6.9 in an alternative form; namely, that the best rank-1 approximation of a symmetric tensor and its best symmetric-rank-1 approximation may be chosen to be the same. Again, while we restrict ourselves to symmetric 3-tensors, this result holds for symmetric tensors of arbitrary order.

THEOREM 10.1 (BANACH). *Let $\mathcal{S} \in \mathbb{R}^{n\times n\times n}$ be a symmetric 3-tensor. Then,*

$$\min_{\sigma\geq 0,\ \|\mathbf{u}\|_2=\|\mathbf{v}\|_2=\|\mathbf{w}\|_2=1} \|\mathcal{S} - \sigma \mathbf{u}\otimes\mathbf{v}\otimes\mathbf{w}\|_F = \min_{\lambda\geq 0,\ \|\mathbf{v}\|_2=1} \|\mathcal{S} - \lambda \mathbf{v}\otimes\mathbf{v}\otimes\mathbf{v}\|_F. \tag{32}$$

Furthermore, the optimal σ and λ may be chosen to be equal.

PROOF. This result follows from carrying over our discussion relating spectral norm, largest singular value, and best rank-1 approximation (for nonsymmetric tensors) in Section 7 to the case of symmetric tensors. This gives

$$\lambda = \max_{\|\mathbf{v}\|_2=1} \langle \mathcal{S}, \mathbf{v}\otimes\mathbf{v}\otimes\mathbf{v}\rangle = \|\mathcal{S}\|_{2,2,2},$$

where λ is the optimal solution for the right-hand side of (32) and the last equality holds by Theorem 6.9. □

Theorems 6.9 and 10.1, together with Theorem 9.3 and Corollary 9.6, prove:

THEOREM 10.2. *The following problems are all NP-hard over $\mathbb{F} = \mathbb{R}$:*

(i) *Deciding the largest ℓ^2-singular value or eigenvalue of a symmetric 3-tensor.*
(ii) *Deciding the spectral norm of a symmetric 3-tensor.*
(iii) *Determining the best symmetric rank-1 approximation of a symmetric 3-tensor.*

Furthermore, unless P = NP, there are no FPTAS for these problems.

[13] Håstad's original result required NP $\neq$ ZPP, but Zuckerman weakened this to P $\neq$ NP.

PROOF. Let $\mathcal{S} \in \mathbb{Q}^{n\times n\times n}$ be a symmetric 3-tensor. By Theorem 10.1, the optimal σ in the best rank-1 approximation of $\mathcal{S}$ (left-hand side of (32)) equals the optimal λ in the best symmetric rank-1 approximation of $\mathcal{S}$ (right-hand side of (32)). Since the optimal σ is also the largest ℓ^2-singular value of $\mathcal{S}$ and the optimal λ is the largest ℓ^2-eigenvalue of $\mathcal{S}$, these also coincide. Note that the optimal σ is also equal to the spectral norm $\|\mathcal{S}\|_{2,2,2}$. The NP-hardness and nonexistence of FPTAS of problems (i)–(iii) now follow from Theorem 9.3 and Corollary 9.6. □

Theorem 10.2 answers a question in Brubaker and Vempala [2009] about the computational complexity of spectral norm for symmetric tensors.

11. TENSOR NONNEGATIVE DEFINITENESS IS NP-HARD

There are two senses in which a symmetric 4-tensor $\mathcal{S} \in \mathbb{R}^{n\times n\times n\times n}$ can be nonnegative definite. We shall reserve the term *nonnegative definite* to describe $\mathcal{S}$ for which $\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$ is a nonnegative polynomial; that is,

$$\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) = \sum\nolimits_{i,j,k,l=1}^{n} s_{ijkl}x_ix_jx_kx_l \geq 0, \quad \text{for all } \mathbf{x} \in \mathbb{R}^n. \tag{33}$$

On the other hand, we say that $\mathcal{S}$ is *Gramian* if it can be decomposed as a positive combination of rank-1 terms:

$$\mathcal{S} = \sum_{i=1}^{r} \lambda_i \mathbf{v}_i \otimes \mathbf{v}_i \otimes \mathbf{v}_i \otimes \mathbf{v}_i, \quad \lambda_i > 0,\ \|\mathbf{v}_i\|_2 = 1; \tag{34}$$

or equivalently, if $\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$ can be written as a sum of fourth powers of linear forms:

$$\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) = \sum_{i=1}^{r} (\mathbf{w}_i^\top \mathbf{x})^4. \tag{35}$$

The correspondence between (34) and (35) is to set $\mathbf{w}_i = \lambda_i^{1/4}\mathbf{v}_i$. Note that for a symmetric matrix $S \in \mathbb{R}^{n\times n}$, the condition $\mathbf{x}^\top S\mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$ and the condition $S = B^\top B$ for some matrix B (i.e., S is a Gram matrix) are equivalent characterizations of the positive semidefiniteness of S. For tensors of even order $d > 2$, condition (34) is strictly stronger than (33), but both are valid generalizations of the notion of nonnegative definiteness. In fact, the cone of nonnegative definite tensors as defined by (33) and the cone of Gramian tensors as defined by (34) are dual [Reznick 1992].

We consider tensors of order 4 because any symmetric 3-tensor $\mathcal{S} \in \mathbb{R}^{n\times n\times n}$ is *indefinite* since $\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x})$ can take both positive and negative values (as $\mathcal{S}(-\mathbf{x},-\mathbf{x},-\mathbf{x}) = -\mathcal{S}(\mathbf{x},\mathbf{x},\mathbf{x})$). Order-3 symmetric tensors are also trivially Gramian since $-\lambda\mathbf{v}\otimes\mathbf{v}\otimes\mathbf{v} = \lambda(-\mathbf{v})\otimes(-\mathbf{v})\otimes(-\mathbf{v})$, and thus λ may always be chosen to be positive.

We deduce the NP-hardness of both notions of nonnegative definiteness from Murty and Kabadi [1987] and Dickinson and Gijben [2012]. Let $A \in \mathbb{R}^{n\times n}$ be symmetric. The matrix A is said to be *copositive* if $A(\mathbf{y},\mathbf{y}) = \mathbf{y}^\top A\mathbf{y} \geq 0$ for all $\mathbf{y} \geq \mathbf{0}$, and A is said to be *completely positive* if $A = BB^\top$ for some $B \in \mathbb{R}^{n\times r}$ with $B \geq 0$ (i.e., all entries nonnegative). The set of copositive matrices in $\mathbb{R}^{n\times n}$ is easily seen to be a cone and it is dual to the set of completely positive matrices, which is also a cone. Duality here means that $\mathrm{tr}(A_1A_2) \geq 0$ for any copositive A_1 and completely positive A_2.

THEOREM 11.1 (MURTY–KABADI, DICKINSON–GIJBEN). *Deciding copositivity and complete positivity are both NP-hard.*

Let $A = [a_{ij}] \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Consider the symmetric 4-tensor $\mathcal{S} = [\![s_{ijkl}]\!] \in \mathbb{R}^{n\times n\times n\times n}$ defined by

$$s_{ijkl} = \begin{cases} a_{ij} & \text{if } i = k \text{ and } j = l, \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\mathcal{S}$ is symmetric since $a_{ij} = a_{ji}$. Now $\mathcal{S}$ is nonnegative definite if and only if

$$\mathcal{S}(\mathbf{x}, \mathbf{x}, \mathbf{x}, \mathbf{x}) = \sum_{i,j,k,l=1}^{n} s_{ijkl} x_i x_j x_k x_l = \sum_{i,j=1}^{n} a_{ij} x_i^2 x_j^2 \geq 0,$$

for all $\mathbf{x} \in \mathbb{R}^n$, which is in turn true if and only if $\mathbf{y}^\top A\mathbf{y} \geq 0$ for all $\mathbf{y} \geq \mathbf{0}$ ($y_i = x_i^2$), that is, A is copositive. On the other hand, the tensor $\mathcal{S}$ is Grammian if and only if

$$s_{ijkl} = \sum_{p=1}^{r} w_{ip} w_{jp} w_{kp} w_{lp}, \quad i, j, k, l = 1, \ldots, n,$$

for some $r \in \mathbb{N}$, which is to say that

$$a_{ij} = \sum_{p=1}^{r} w_{ip}^2 w_{jp}^2, \quad i, j = 1, \ldots, n,$$

or $A = BB^\top$ where $B = [w_{ip}^2] \in \mathbb{R}^{n\times r}$ has nonnegative entries; that is, A is completely positive. Hence, we have deduced the following.

THEOREM 11.2. *Deciding whether a symmetric 4-tensor is nonnegative definite is NP-hard. Deciding whether a symmetric 4-tensor is Grammian is also NP-hard.*

The first statement in Theorem 11.2 has appeared before in various contexts, most notably as the problem of deciding the nonnegativity of a quartic; see, for example, Ahmadi et al. [2013]. It follows from the second statement and (35) that deciding whether a quartic polynomial is a sum of fourth powers of linear forms is NP-hard.

The reader may wonder about a third common characterization of nonnegative definiteness: A symmetric matrix is nonnegative definite if and only if all its eigenvalues are nonnegative. It turns out that for tensors this does not yield a different characterization of nonnegative definiteness. The exact same equivalence is true for symmetric tensors with our definition of eigenvalues in Section 5 [Qi 2005, Theorem 5]:

THEOREM 11.3 (QI). *The following are equivalent for a symmetric $\mathcal{S} \in \mathbb{R}^{n\times n\times n\times n}$:*

(*i*) *$\mathcal{S}$ is nonnegative definite.*
(*ii*) *All ℓ^2-eigenvalues of $\mathcal{S}$ are nonnegative.*
(*iii*) *All ℓ^4-eigenvalues of $\mathcal{S}$ are nonnegative.*

This result is deduced from the fact that ℓ^2- and ℓ^4-eigenvalues are Lagrange multipliers. With this observation, the following is an immediate corollary of Theorem 11.2.

COROLLARY 11.4. *Determining the signature, that is, the signs of the real eigenvalues, of symmetric 4-tensors is NP-hard.*

12. BIVARIATE MATRIX FUNCTIONS ARE UNDECIDABLE

While we have focused almost exclusively on *complexity* in this article, we would like to add a word more about *computability* in this antepenultimate section.

There has been much interest in computing various functions of a matrix [Higham 2008]. The best-known example is probably the matrix exponential $\exp : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$, which is important in many applications. Recenty, there have been attempts to generalize such studies to functions of two matrices, notably [Kressner 2013], whose approach we shall adopt. For bivariate polynomials, $f(x,y) = \sum_{i,j=0}^{d} a_{ij}x^i y^j \in \mathbb{C}[x,y]$, and a pair of *commuting* matrices $A_1, A_2 \in \mathbb{C}^{n\times n}$, we define $f(A_1, A_2)$ as the matrix function $f(A_1,A_2) : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ given by:

$$f(A_1,A_2)(X) = \sum_{i,j=0}^{d} a_{ij}A_1^i X A_2^j, \quad X \in \mathbb{C}^{n\times n}. \tag{36}$$

Note that a pair of matrices may be regarded as a 3-tensor $\mathcal{A} = [A_1,A_2] \in \mathbb{C}^{n\times n\times 2}$, where A_1, A_2 are the two "slices" of $\mathcal{A}$.

If, however, we do not assume that A_1, A_2 be a commuting pair (these are rare, the set of commuting pairs has measure zero in $\mathbb{C}^{n\times n\times 2}$), then (36) is inadequate and we need to include all possible noncommutative monomials. Consider the simplest case in which $f(A_1,A_2)$ consists of a single monic monomial and $X = I$, the identity matrix, but we no longer assume that A_1, A_2 necessarily commute. Then,

$$f(A_1,A_2)(I) = A_1^{m_1}A_2^{n_1}A_1^{m_2}A_2^{n_2}\cdots A_1^{m_r}A_2^{n_r}, \tag{37}$$

where $m_1,\ldots,m_r$, $n_1,\ldots,n_r$, and r are nonnegative integers. If A_1 and A_2 commute, and if we write $m = m_1 + \cdots + m_r$ and $n = n_1 + \cdots + n_r$, then (37) reduces to (36) with $a_{mn} = 1$ and all other $a_{ij} = 0$; that is, $f(A_1,A_2)(I) = A_1^m A_2^n$. Consider the following seemingly innocuous problem concerning (37).

Problem 12.1 (*Bivariate Matrix Monomials*). Given $\mathcal{A} = [A_1,A_2] \in \mathbb{C}^{n\times n\times 2}$, is there a bivariate monic monomial function f such that $f(A_1,A_2)(I) = 0$?

This is in fact the matrix mortality problem for two matrices [Halava et al. 2007], and as a consequence, we have the following.

PROPOSITION 12.2 (*Halava–Harju–Hirvensalo*). *Problem 12.1 is undecidable when* $n > 20$.

This fact is to be contrasted with its univariate equivalent: Given $A \in \mathbb{C}^{n\times n}$, a monic monomial f exists with $f(A) = 0$ if and only if A is nilpotent.

13. OPEN PROBLEMS

We have tried to be thorough in our list of tensor problems, but there are some that we have not studied. We state a few of them here as open problems. The first involve the hyperdeterminant. Let $\mathbb{Q}[\mathrm{i}] = \{a + b\mathrm{i} \in \mathbb{C} : a, b \in \mathbb{Q}\}$ be the field of *Gaussian rationals*.

CONJECTURE 13.1. *Let* $l, m, n \in \mathbb{N}$ *satisfy GKZ condition* (10)*:*

$$l \le m + n - 1, \quad m \le l + n - 1, \quad n \le l + m - 1,$$

and let $\mathrm{Det}_{l,m,n}$ *be the* $l \times m \times n$ *hyperdeterminant.*

(*i*) *Deciding* $\mathrm{Det}_{l,m,n}(\mathcal{A}) = 0$ *is an NP-hard decision problem for* $\mathcal{A} \in \mathbb{Q}[\mathrm{i}]^{l\times m\times n}$.

(*ii*) *It is NP-hard to decide or approximate for inputs* $\mathcal{A} \in \mathbb{Q}[\mathrm{i}]^{l\times m\times n}$ *the value of:*

$$\min_{\mathrm{Det}_{l,m,n}(\mathcal{X})=0} \|\mathcal{A} - \mathcal{X}\|_{2,2,2}. \tag{38}$$

(*iii*) *Evaluating the magnitude of* $\mathrm{Det}_{l,m,n}(\mathcal{A})$ *is #P-hard for inputs* $\mathcal{A} \in \{0,1\}^{l\times m\times n}$.
(*iv*) *The homogeneous polynomial* $\mathrm{Det}_{l,m,n}$ *is VNP-hard to compute.*
(*v*) *All statements above remain true in the special case* $l = m = n$.

We remark that resolutions to these conjectures are likely to have implications for applications. For instance, in quantum computing, the magnitude of the hyperdeterminant in (13.1) is the *concurrence*, a measure of the amount of entanglement in a quantum system [Hill and Wootters 1997], and the hyperdeterminant in question would be one satisfying (13.1). The decision problem in (13.1) is also key to deciding whether a system of multilinear equations has a nontrivial solution, as we have seen from Section 3.

The optimization problem (38) in (13.1) defines a notion of *condition number* for 3-tensors. Note that for a nonsingular matrix $A \in \mathbb{C}^{n\times n}$, the corresponding problem for (13.1) has solution given by its inverse $X = A^{-1}$ [Higham 2002, Theorem 6.5]:

$$\min_{\det(X)=0} \|A - X\|_{2,2} = \|A^{-1}\|_{2,2}^{-1}.$$

In this case, the optimum value normalized by the spectral norm of the input gives the reciprocal of the condition number:

$$\frac{\|A^{-1}\|_{2,2}^{-1}}{\|A\|_{2,2}} = \kappa(A)^{-1}.$$

Thus, for a nonzero $\mathcal{A} \in \mathbb{C}^{l\times m\times n}$, we expect the spectral norm $\|\mathcal{A}\|_{2,2,2}$ divided by the optimum value of (13.1) to yield an analogue of condition number for the tensor. Conjecture 13.1 then says that the condition number of a tensor is NP-hard to compute.

One reason for our belief in the intractability of problems involving the hyperdeterminant is that checking whether the general multivariate resultant vanishes for a system of n polynomials in n variables is known to be NP-hard over any field [Grenet et al. 2010]. Theorem 3.7 strengthens this result by saying that these polynomials may be chosen to be bilinear forms. Conjecture 13.1(i) further specializes by stating that these forms (11) can be associated with a 3-tensor satisfying GKZ condition (10).

Another motivation for our conjectures is that the hyperdeterminant is a complex object; for instance, the $2 \times 2 \times 2 \times 2$-hyperdeterminant has 2.9 million monomials [Huggins et al. 2008]. Of course, this does not force the intractability of the problems above. For instance, the determinant and permanent of an $n \times n$ matrix have $n!$ terms, but one is efficiently computable while the other is #P-complete [Valiant 1979b].

In Section 8, we explained that tensor rank is NP-hard over any extension field $\mathbb{F}$ of $\mathbb{Q}$, but we did not investigate the corresponding questions for the symmetric rank of a symmetric tensor (Definition 7). We conjecture the following.

CONJECTURE 13.2. *Let* $\mathbb{F}$ *be an extension field of* $\mathbb{Q}$. *Let* $\mathcal{S} \in \mathbb{Q}^{n\times n\times n}$ *be a symmetric 3-tensor and* $r \in \mathbb{N}$. *Deciding if* $\mathrm{srank}_{\mathbb{F}}(\mathcal{S}) \le r$ *is NP-hard.*

While tensor rank is NP-hard over $\mathbb{Q}$, we suspect that it is also undecidable.

CONJECTURE 13.3. *Tensor and symmetric tensor rank over* $\mathbb{Q}$ *are undecidable.*

We have shown that deciding the existence of an exact solution to a system of bilinear equations (12) is NP-hard. There are two closely related problems: (i) when the equalities in (12) are replaced by inequalities and (ii) when we seek an approximate least-squares solution to (12). These lead to multilinear variants of linear programming and linear least squares. We state them formally here.

CONJECTURE 13.4 (BILINEAR PROGRAMMING FEASIBILITY). *Let $A_k, B_k, C_k \in \mathbb{Q}^{n\times n}$ and $\alpha_k, \beta_k, \gamma_k \in \mathbb{Q}$ for each $k = 1, \dots, n$. It is NP-hard to decide if inequalities:*

$$\begin{cases} \mathbf{y}^\top A_i \mathbf{z} \le \alpha_k, & i = 1, \dots, n; \\ \mathbf{x}^\top B_j \mathbf{z} \le \beta_k, & j = 1, \dots, n; \\ \mathbf{x}^\top C_k \mathbf{y} \le \gamma_k, & k = 1, \dots, n; \end{cases} \tag{39}$$

define a nonempty subset of $\mathbb{R}^n$.

CONJECTURE 13.5 (BILINEAR LEAST SQUARES). *Given $3n$ coefficient matrices $A_k, B_k, C_k \in \mathbb{Q}^{n\times n}$ and $\alpha_k, \beta_k, \gamma_k \in \mathbb{Q}$, $k = 1, \dots, n$, the bilinear least squares problem:*

$$\min_{\mathbf{x},\mathbf{y},\mathbf{z}\in\mathbb{R}^n} \sum_{k=1}^n (\mathbf{x}^\top A_k \mathbf{y} - \alpha_k)^2 + (\mathbf{y}^\top B_k \mathbf{z} - \beta_k)^2 + (\mathbf{z}^\top C_k \mathbf{x} - \gamma_k)^2 \tag{40}$$

is NP-hard to approximate.

Unlike the situation of (12), where $\mathbf{x} = \mathbf{y} = \mathbf{z} = \mathbf{0}$ is considered a trivial solution, we can no longer disregard an all-zero solution in (39) or (40). Consequently, the problem of deciding whether a homogeneous system of bilinear Equations (12) has a *nonzero* solution is not a special case of (39) or (40).

It is also natural to consider the skew-symmetric or alternating tensor equivalents of the problems addressed in this work. We leave these as further open problems for our readers.

14. CONCLUSION

Although this article argues that most tensor problems are NP-hard, we should not be discouraged in our search for solutions to them. For instance, while computations with Gröbner bases are doubly exponential in the worst case [Yap 2000, pp. 400], they nonetheless proved useful for Theorem 1.14. It is also important to note that NP-hardness is an asymptotic property; for example, it applies to scenarios where tensor size n goes to infinity. Nonetheless, in many applications, n is usually fixed and often small; for example, $n = 2 : |0\rangle, |1\rangle$ (qubits, [Miyake and Wadati 2002]), $n = 3 : x, y, z$ (spatial coordinates, [Schultz and Seidel 2008]), $n = 4 : A, C, G, T$ (DNA nucleobases, [Allman and Rhodes 2008]), etc. For example, while Theorem 11.2 gives an NP-hardness result for general n, the case $n = 3$ has a tractable convex formulation [Lim and Schultz 2013].

Bernd Sturmfels once made the remark to us that “All interesting problems are NP-hard.” In light of this, we would like to view our article as evidence that most tensor problems are interesting.

APPENDIX

We give here the complete details for the proof of Lemma 8.1, which was key to proving Theorem 1.14. We used the symbolic computing software[14] SINGULAR, and in

[14]One can use commercially available Maple, `http://www.maplesoft.com/products/maple`, or Mathematica, `http://www.wolfram.com/mathematica`; free SINGULAR, `http://www.singular.uni-kl.de`, Macaulay 2, `http://www.math.uiuc.edu/Macaulay2`, or Sage, `http://www.sagemath.org`. For numerical packages, see Bertini, `http://www.nd.edu/~sommese/bertini` and PHCpack, `http://homepages.math.uic.edu/~jan/download.html`.

particular the function **lift** to find the polynomials $H_1, \ldots, H_8$ and $G_1, \ldots, G_8$. Define three sets of polynomials:

$$F_1 = a_1a_2a_3 + c_1c_2c_3 - 2,\ F_2 = a_1a_3b_2 + c_1c_3d_2,\ F_3 = a_2a_3b_1 + c_2c_3d_1,$$
$$F_4 = a_3b_1b_2 + c_3d_1d_2 + 4,\ F_5 = a_1a_2b_3 + c_1c_2d_3,\ F_6 = a_1b_2b_3 + c_1d_2d_3 + 4,$$
$$F_7 = a_2b_1b_3 + c_2d_1d_3 - 4,\ F_8 = b_1b_2b_3 + d_1d_2d_3.$$

$G_1 = -\frac{1}{8}b_1b_2b_3c_2d_2 + \frac{1}{8}a_2b_1b_3d_2^2,\ G_2 = \frac{1}{8}b_1b_2b_3c_2^2 - \frac{1}{8}a_2b_1b_3c_2d_2,\ G_3 = -\dfrac{1}{2}c_2d_2,\ G_4 = \dfrac{1}{2}c_2^2,$

$G_5 = \dfrac{1}{8}a_3b_1b_2c_2d_2 - \dfrac{1}{8}a_2a_3b_1d_2^2,\ G_6 = -\dfrac{1}{8}a_3b_1b_2c_2^2 + \dfrac{1}{8}a_2a_3b_1c_2d_2,\ G_7 = -\frac{1}{2}c_1^2,$

$G_8 = \frac{1}{8}a_2a_3b_1c_1^2 - \frac{1}{8}a_1a_2a_3c_1d_1.$

$H_1 = 0,\ H_2 = -\frac{1}{32}b_1b_2b_3c_2d_1d_3 + \frac{1}{32}a_2b_1b_3d_1d_2d_3, H_3 = \frac{1}{32}b_1b_2b_3c_1d_2d_3 - \frac{1}{32}a_1b_2b_3d_1d_2d_3,$

$H_4 = \frac{1}{32}a_1b_2b_3c_2d_1d_3 - \frac{1}{32}a_2b_1b_3c_1d_2d_3,\ H_5 = -\frac{1}{8}b_1b_2b_3,\ H_6 = \frac{1}{2}, H_7 = \frac{1}{8}a_1b_2b_3 - \frac{1}{8}c_1d_2d_3,$

$H_8 = \frac{1}{8}c_1c_2d_3.$

Both $g = 2c_2^2 - d_2^2$ and $h = c_1d_2d_3 - 2$ are polynomial combinations of $F_1, \ldots, F_8$:

$$g = \sum_{k=1}^{8} F_kG_k \quad \text{and} \quad h = \sum_{k=1}^{8} F_kH_k. \tag{41}$$

Thus, if a rational point makes $F_1, \ldots, F_8$ all zero, then both g and h must also vanish on it. We remark that expressions such as (41) are far from unique.

ACKNOWLEDGMENTS

We thank Jan Draisma, Dorit Hochbaum, Jiawang Nie, Pablo Parrilo, Sasha Razborov, Steve Smale, Bernd Sturmfels, Steve Vavasis, and Shuzhong Zhang for helpful comments that enhanced the quality of this work. We also would like to thank the anonymous referees for comments that improved the article.

REFERENCES

A. A. Ahmadi, A. Olshevsky, P. A. Parrilo, and J. N. Tsitsiklis. 2013. NP-hardness of deciding convexity of quartic polynomials and related problems. *Math. Program. Ser. A 137*, 1–2, 453–476.

M. Aigner. 1995. Turán's graph theorem. *Amer. Math. Monthly 102*, 9, 808–816.

E. S. Allman and J. A. Rhodes. 2008. Phylogenetic ideals and varieties for the general Markov model. *Adv. Appl. Math. 40*, 2, 127–148.

N. Alon and A. Naor. 2006. Approximating the cut-norm via Grothendieck's inequality. *SIAM J. Comput. 35*, 4, 787–803.

S. Arora, R. Ge, R. Kannan, and A. Moitra. 2012. Computing a nonnegative matrix factorization — provably. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 145–162.

C. Bachoc and F. Vallentin, 2008. New upper bounds for kissing numbers from semidefinite programming. *J. Amer. Math. Soc. 21*, 3, 909–924.

C. Bachoc, G. Nebe, F. M. de Oliveira Filho, and F. Vallentin. 2009. Lower bounds for measurable chromatic numbers. *Geom. Funct. Anal. 19*, 3, 645–661.

S. Banach. 1938. Über homogene polynome in (L^2). *Studia Math. 7*, 1, 36–44.

A. I. Barvinok, 1993. Feasibility testing for systems of real quadratic equations. *Discrete Comput. Geom. 10*, 1, 1–13.

A. I. Barvinok. 1995. New algorithms for linear k-matroid intersection and matroid k-parity problems. *Math. Prog. 69*, 3, 449–470.

D. Bayer. 1982. The division algorithm and the Hilbert scheme. Ph.D. thesis, Department of Mathematics, Harvard University.

C. Beltrán and L. M. Pardo. 2009. Efficient polynomial system-solving by numerical methods. *J. Fixed Point Theory Appl. 6*, 1, 63–85.

E. Bernstein and U. Vazirani. 1997. Quantum complexity theory. *SIAM J. Comput. 26*, 5, 1411–1473.

G. Beylkin and M. J. Mohlenkamp. 2002. Numerical operator calculus in higher dimensions. *Proc. Natl. Acad. Sci. 99*, 16, 10246–10251.

L. Blum, M. Shub, and S. Smale. 1989. On a theory of computation and complexity over the real numbers. *Bull. Amer. Math. Soc. 21*, 1, 1–46.

L. Blum, F. Cucker, M. Shub, and S. Smale. 1998. *Complexity and Real Computation*. Springer-Verlag, New York.

J. Bochnak, M. Coste, and M. F. Roy. 1998. *Real Algebraic Geometry* 2nd Ed. Ergebnisse der Mathematik und ihrer Grenzgebiete, 36, Springer-Verlag, Berlin.

J. Briët, F. M. de Oliveira Filho, and F. Vallentin. 2010a. The positive semidefinite Grothendieck problem with rank constraint. In *Automata, Languages and Programming*, Lecture Notes in Computer Science, vol. 6198, S. Abramsky et al. Eds., Springer-Verlag, Berlin, 31–42.

J. Briët, F. M. de Oliveira Filho, and F. Vallentin. 2010b. Grothendieck inequalities for semidefinite programs with rank constraint. Preprint, `http://arxiv.org/abs/1011.1754`.

S. C. Brubaker and S. S. Vempala. 2009. Random tensors and planted cliques. In *Proceedings of APPROX and RANDOM'09*, Lecture Notes in Computer Science, vol. 5687, Dinur et al. Eds., Springer-Verlag, Berlin.

B. Buchberger. 1970. Ein algorithmisches kriterium für die Lösbarkeit eines algebraischen Gleichungssystems. *Aequationes Math. 4*, 3, 374–383.

P. Bürgisser. 2000. Completeness and reduction in algebraic complexity theory. In *Algorithms and Computations in Mathematics*, vol. 7, Springer-Verlag, Berlin.

P. Bürgisser, M. Clausen, and M. A. Shokrollahi. 1996. *Algebraic Complexity Theory*. Grundlehren der mathematischen Wissenschaften, vol. 315, Springer-Verlag, Berlin.

D. A. Cartwright and B. Sturmfels. 2013. The number of eigenvalues of a tensor. *Linear Algebra Appl. 438*, 2, 942–952.

D. A. Cartwright, S. M. Brady, D. A. Orlando, B. Sturmfels, and P. N. Benfey. 2009. Reconstructing spatiotemporal gene expression data from partial observations. *Bioinform. 25*, 19, 2581–2587.

A. Cayley. 1845. On the theory of linear transformation. *Cambridge Math. J. 4*, 193–209.

S. Cohen and C. Tomasi. 1997. Systems of bilinear equations. Tech. rep. 97-1588, Department of Computer Science, Stanford University.

T. F. Coleman and A. Pothen. 1986. The null space problem I: Complexity. *SIAM J. Alg. Disc. Meth. 7*, 4, 527–537.

P. Comon. 1994. Independent component analysis: A new concept? *Signal Process. 36*, 3, 287–314.

P. Comon. 2004. Blind identification and source separation in 2×3 under-determined mixtures. *IEEE Trans. Signal Process. 52*, 1, 11–22.

P. Comon, G. H. Golub, L.-H. Lim, and B. Mourrain. 2008. Symmetric tensors and symmetric tensor rank. *SIAM J. Matrix Anal. Appl. 30*, 3, 1254–1279.

S. Cook. 1971. The complexity of theorem proving procedures. In *Proceedings of the ACM Symposium on the Theory of Computing (STOC)*. 151–158.

R. Coppi and S. Bolasco, Eds. 1989. *Multiway Data Analysis*. Elsevier Science, Amsterdam, The Netherlands.

N. Courtois, L. Goubin, W. Meier, and J.-D. Tacier. 2002. Solving underdefined systems of multivariate quadratic equations. In *Proceedings of the Symposium on Public Key Cryptography (PKC'02)*, Lecture Notes in Computer Science, vol. 2274, D. Naccache and P. Paillier Eds., Springer-Verlag, Berlin.

D. A. Cox, J. B. Little, and D. O'Shea. 2005. *Using Algebraic Geometry* 2nd Ed. Springer-Verlag, New York.

D. A. Cox, J. B. Little, and D. O'Shea. 2007. *Ideals, Varieties, and Algorithms* 3rd Ed. Springer-Verlag, New York, NY.

M. Davis, H. Putnam, and J. Robinson. 1961. The decision problem for exponential diophantine equations. *Ann. Math. 74*, 3, 425–436.

M. Davis, Y. Matijasevich, and J. Robinson. 1976. Diophantine equations: Positive aspects of a negative solution. In *Mathematical Developments Arising from Hilbert Problems Proceedings of Symposia in Pure Mathematics*, vol. XXVIII, AMS, Providence.

A. De, P. P. Kurur, C. Saha, and R. Saptharishi. 2008. Fast integer multiplication using modular arithmetic. In *Proceedings of the ACM Symposium on the Theory of Computing (STOC)*. 499–506.

E. De Klerk. 2008. The complexity of optimizing over a simplex, hypercube or sphere: A short survey. *Cent. Eur. J. Oper. Res. 16*, 2, 111–125.

E. De Klerk and D. V. Pasechnik. 2002. Approximation of the stability number of a graph via copositive programming. *SIAM J. Optim. 12*, 4, 875–892.

W. F. De La Vega, R. Kannan, M. Karpinski, and S. Vempala. 2005. Tensor decomposition and approximation schemes for constraint satisfaction problems. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 747–754.

J. A. De Loera. 1995. Gröbner bases and graph colorings. *Beiträge Alg. Geom. 36*, 1, 89–96.

J. A. De Loera, J. Lee, P. N. Malkin, and S. Margulies. 2008. Hilbert's nullstellensatz and an algorithm for proving combinatorial infeasibility. In *Proceedings of the International Symposium on Symbolic Algebra Computing (ISSAC)*. 197–206.

V. De Silva and L.-H. Lim. 2008. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM J. Matrix Anal. Appl. 30*, 3, 1084–1127.

D. Deutsch. 1985. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proc. Roy. Soc. London Ser. A 400*, 1818, 97–117.

P. J. C. Dickinson and L. Gijben. 2012. On the computational complexity of membership problems for the completely positive cone and its dual. Preprint
http://www.optimization-online.org/DB_HTML/2011/05/3041.html.

J. Edmonds. 1967. Systems of distinct representatives and linear algebra. *J. Res. Nat. Bur. Standards 71B*, 4, 241–245.

L. Fortnow. 2009. The status of the P versus NP problem. *Commun. ACM 52*, 9, 78–86.

L. Fortnow and J. Rogers. 1999. Complexity limitations on quantum computation. *J. Comput. System Sci. 59*, 2, 240–252.

P. J. Freitas, S. Friedland, and G. Porta. 2012. Upper bounds on the magnitude of solutions of certain linear systems with integer coefficients. *Electron. J. Linear Alg. 24*, 113–125.

J. Friedman. 1991. The spectra of infinite hypertrees. *SIAM J. Comput. 20*, 5, 951–961.

J. Friedman and A. Wigderson. 1995. On the second eigenvalue of hypergraphs. *Combinatorica 15*, 1, 43–65.

M. Fürer. 2007. Faster integer multiplication. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 57–66.

I. M. Gelfand, M. M. Kapranov, and A. V. Zelevinsky. 1992. Hyperdeterminants. *Adv. Math. 96*, 2, 226–263.

I. M. Gelfand, M. M. Kapranov, and A. V. Zelevinsky. 1994. *Discriminants, Resultants, and Multidimensional Determinants*. Birkhäuser Publishing, Boston, MA.

M. X. Goemans and D. P. Williamson. 1995. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM 42*, 6, 1115–1145.

G. H. Golub and W. Kahan. 1965. Calculating the singular values and pseudo-inverse of a matrix. *J. Soc. Indust. Appl. Math. Ser. B2*, 205–224.

G. H. Golub and C. Reinsch. 1970. Singular value decomposition and least squares solutions. *Numer. Math. 14*, 5, 403–420.

B. Grenet, P. Koiran, and N. Portier. 2010. The multivariate resultant is NP-hard in any characteristic. In *Mathematical Foundations of Computer Science (MFCS)*, Lecture Notes in Computer Science, vol. 6281, P. Hliněný and A. Kučera Eds., Springer-Verlag, Berlin.

L. Gurvits. 2005. On the complexity of mixed discriminants and related problems. In *Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, vol. 3618, Springer, Berlin, 447–458.

V. Halava, T. Harju, and M. Hirvensalo. 2007. Undecidability bounds for integer matrices using Claus instances. *Internat. J. Found. Comput. Sci. 18*, 5, 931–948.

D. Haskell, A. Pillay, and C. Steinhorn, Eds. 2000. *Model Theory, Algebra, and Geometry*. Mathematical Sciences Research Institute Publications, vol. 39, Cambridge University Press, Cambridge.

J. Håstad. 1990. Tensor rank is NP-complete. *J. Algor. 11*, 4, 644–654.

J. Håstad. 1999. Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Math. 182*, 1, 105–142.

S. He, Z. Li, and S. Zhang. 2010. Approximation algorithms for homogeneous polynomial optimization with quadratic constraints. *Math. Prog. Ser. B 125*, 2, 353–383.

J. M. Hendrickx and A. Olshevsky. 2010. Matrix p-norms are NP-hard to approximate if $p \neq 1, 2, \infty$. *SIAM J. Matrix Anal. Appl. 31*, 5, 2802–2812.

N. J. Higham. 2002. *Accuracy and Stability of Numerical Algorithms* 2nd Ed. SIAM, Philadelphia, PA.

N. J. Higham. 2008. *Functions of Matrices: Theory and Computation*. SIAM, Philadelphia, PA.

D. Hilbert. 1902. Mathematical problems. *Bull. Amer. Math. Soc. 8*, 10, 437–479.

S. Hill and W. K. Wootters. 1997. Entanglement of a pair of quantum bits. *Phys. Rev. Lett. 78*, 26, 5022–5025.

C. Hillar and T. Windfeldt. 2008. Algebraic characterization of uniquely vertex colorable graphs. *J. Combin. Theory Ser. B 98*, 2, 400–414.

F. L. Hitchcock. 1927a. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys. 6*, 1, 164–189.

F. L. Hitchcock. 1927b. Multiple invariants and generalized rank of a p-way matrix or tensor. *J. Math. Phys. 7*, 1, 39–79.

D. S. Hochbaum, Ed. 1997. *Approximation Algorithms for NP-Hard Problems*. PWS Publishing, Boston, MA.

D. S. Hochbaum and J. G. Shanthikumar. 1990. Convex separable optimization is not much harder than linear optimization. *J. ACM 37*, 4, 843–862.

P. Huggins, B. Sturmfels, J. Yu, and D. S. Yuster. 2008. The hyperdeterminant and triangulations of the 4-cube. *Math. Comp. 77*, 263, 1653–1679.

J. P. Jones. 1982. Universal Diophantine equation. *J. Symb. Logic 47*, 3, 549–571.

J. P. Jones and Y. V. Matijasevič. 1984. Register machine proof of the theorem on exponential diophantine representation of enumerable sets. *J. Symb. Logic 49*, 3, 818–829.

W. Kahan. 1997. *IEEE Standard 754 for Binary Floating-Point Arithmetic*. `http://www.cs.berkeley.edu/~wkahan/ieee754status/ieee754.ps`, Berkeley, CA.

R. M. Karp. 1972. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, R. E. Miller and J. W. Thatcher Eds., Plenum, New York.

L. G. Khachiyan. 1979. A polynomial algorithm in linear programming. *Dokl. Akad. Nauk SSSR 244*, 5, 1093–1096. (English translation in *Soviet Math. Dokl. 20*, 1, 191–194).

S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. 2007. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM J. Comput. 37*, 1, 319–357.

D. Kressner. 2013. Bivariate matrix functions. *Oper. Matrices*, to appear.

D. E. Knuth. 1974a. A terminology proposal. *SIGACT News 6*, 1, 12–18.

D. E. Knuth. 1974b. Postscript about NP-hard problems. *SIGACT News 6*, 2, 15–16.

D. E. Knuth. 1998. *The Art of Computer Programming*, 2: *Seminumerical Algorithms* 3rd Ed. Addision-Wesley, Reading, MA.

E. Kofidis and P. A. Regalia. 2001/2002. On the best rank-1 approximation of higher-order supersymmetric tensors. *SIAM J. Matrix Anal. Appl. 23*, 3, 863–884.

J. M. Landsberg. 2012. *Tensors: Geometry and Applications*. AMS, Providence, RI.

L. A. Levin. 1973. Universal sequential search problems. *Probl. Inf. Transm. 9*, 3, 265–266.

L.-H. Lim. 2005. Singular values and eigenvalues of tensors: A variational approach. In *Proceedings of the IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Process (CAMSAP)*. 129–132.

L.-H. Lim. 2013. Tensors and hypermatrices. In *Handbook of Linear Algebra* 2nd Ed. L. Hogben Ed., CRC Press, Boca Raton, FL.

L.-H. Lim and T. Schultz. 2013. Moment tensors and high angular resolution diffusion imaging. Preprint.

L. Lovász. 1994. Stable sets and polynomials. *Disc. Math. 124*, 1–3, 137–153.

Y. V. Matijasevič. 1970. The Diophantineness of enumerable sets. *Dokl. Akad. Nauk SSSR 191*, 2, 279–282, (English translation in *Soviet Math. Dokl. 11*, 2, 354–358.)

Y. V. Matijasevič. 1993. *Hilbert's Tenth Problem*. MIT Press, Cambridge, MA.

A. Miyake and M. Wadati. 2002. Multipartite entanglement and hyperdeterminants. *Quant. Inf. Comput. 2*, 540–555.

T. Motzkin and E. G. Straus. 1965. Maxima for graphs and a new proof of a theorem of Turán. *Canad. J. Math. 17*, 533–540.

K. G. Murty and S. N. Kabadi. 1987. Some NP-complete problems in quadratic and nonlinear programming. *Math. Prog. 39*, 2, 117–129.

B. K. Natarajan. 1995. Sparse approximate solutions to linear systems. *SIAM J. Comput. 24*, 2, 227–234.

Yu. Nesterov. 2003. Random walk in a simplex and quadratic optimization over convex polytopes. Preprint. `http://edoc.bib.ucl.ac.be:83/archive/00000238/01/dp2003-71.pdf`.

G. Ni, L. Qi, F. Wang, and Y. Wang. 2007. The degree of the E-characteristic polynomial of an even order tensor. *J. Math. Anal. Appl. 329*, 2, 1218–1229.

M. L. Overton. 2001. *Numerical Computing with IEEE Floating Point Arithmetic*. SIAM, Philadelphia, PA.

A. Pappas, Y. Sarantopoulos, and A. Tonge. 2007. Norm attaining polynomials. *Bull. Lond. Math. Soc. 39*, 2, 255–264.

B. Poonen. 2002. Undecidable problems: A sampler. `http://arxiv.org/abs/1204.0299`.

B. Poonen. 2003. Hilbert's tenth problem and Mazur's conjecture for large subrings of $\mathbb{Q}$. *J. AMS 16*, 4, 981–990.

L. Qi. 2007. Eigenvalues and invariants of tensors. *J. Math. Anal. Appl. 325*, 2, 1363–1377.

L. Qi. 2005. Eigenvalues of a real supersymmetric tensor. *J. Symb. Comput. 40*, 6, 1302–1324.

B. Reznick. 1992. Sums of even powers of real linear forms. *Mem. Amer. Math. Soc. 96*, 463.

B. Reznick. 2013. On the length of binary forms. In *Quadratic and Higher Degree Forms*, Developments in Mathematics, K. Alladi, M. Bhargava, D. Savitt, and P. Tiep Eds., Springer, New York. http://arxiv.org/abs/1007.5485.

T. Schultz and H.-P. Seidel. 2008. Estimating crossing fibers: A tensor decomposition approach. *IEEE Trans. Vis. Comput. Graphics 14*, 6, 1635–1642.

A. Shashua and T. Hazan. 2005. Non-negative tensor factorization with applications to statistics and computer vision. In *Proceedings of the International Conference on Machine Learning (ICML)*. 792–799.

J. Shi and J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell. 22*, 8, 888–905.

N. D. Sidiropoulos, R. Bro, and G. B. Giannakis. 2000. Parallel factor analysis in sensor array processing. *IEEE Trans. Signal Process. 48*, 8, 2377–2388.

M. Sipser. 2012. *Introduction to the Theory of Computation* 3rd Ed., Cengage Learning, Boston, MA.

S. Smale. 2000. Mathematical problems for the next century. In *Mathematics: Frontiers and Perspectives*, V. Arnold, M. Atiyah, P. Lax, and B. Mazur Eds., AMS, Providence, RI.

A. Smilde, R. Bro, and P. Geladi. 2004. *Multi-Way Analysis: Applications in the Chemical Sciences*. John Wiley, West Sussex.

D. Steinberg. 2005. *Computation of Matrix Norms with Applications to Robust Optimization*, M.Sc. dissertation, Technion, Haifa, Israel. http://www2.isye.gatech.edu/~nemirovs/Daureen.pdf.

A. M. Turing. 1937. On computable numbers, with an application to the Entscheidungsproblem. *Proc. Lond. Math. Soc. 2nd Series 42*, 1, 230–265. (see also correction in *43*, 6, 544–546.)

L. G. Valiant. 1979a. Completeness classes in algebra. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 249–261.

L. G. Valiant. 1979b. The complexity of computing the permanent. *Theoret. Comput. Sci. 8*, 2, 189–201.

M. A. O. Vasilescu. 2002. Human motion signatures: Analysis, synthesis, recognition. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 456–460.

M. A. O. Vasilescu and D. Terzopoulos. 2002. Multilinear image analysis for facial recognition. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, 511–514.

M. A. O. Vasilescu and D. Terzopoulos. 2004. TensorTextures: Multilinear image-based rendering. In *Proceedings of ACM SIGGRAPH*. 336–342.

S. A. Vavasis. 1991. *Nonlinear Optimization: Complexity Issues*. International Series of Monographs on Computer Science, Oxford University Press.

S. A. Vavasis. 2009. On the complexity of nonnegative matrix factorization. *SIAM J. Optim. 20*, 3, 1364–1377.

V. Vazirani. 2003. *Approximation Algorithms*. Springer-Verlag, Berlin.

T.-C. Wei and P. M. Goldbart. 2003. Geometric measure of entanglement and applications to bipartite and multipartite quantum states. *Phys. Rev. A 68*, 4, 042307-1–12.

K. Weihrauch. 2000. *Computable Analysis: An Introduction*. Springer-Verlag, Berlin.

C. K. Yap. 2000. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press.

D. Zuckerman. 2006. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proceedings of the ACM Symposium on Theory of Computing (STOC)*. 681–690.

Received April 2012; revised February 2013, June 2013; accepted July 2013