PROBLEMS FROM OPTIMIZATION AND COMPUTATIONAL ALGEBRA EQUIVALENT TO HILBERT'S NULLSTELLENSATZ

Markus Bläser*[1], Sagnik Dutta†[2], and Gorav Jindal‡[3]

[1]Saarland University, Saarland Informatics Campus, Saarbrücken, Germany
[2]Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany
[3]University of Warsaw, Warsaw, Poland

ABSTRACT

Solving polynomial systems is a powerful tool in optimization and computational algebra. The most efficient algorithms for many problems from optimization or algebra often stem from formulating these problems as a system of polynomial equations. This is due to the fact that there are astonishing algorithms for deciding feasibility of polynomial systems, like Koiran's AM-algorithm over the complex numbers (assuming GRH) or Renegar's PSPACE-algorithm over the reals.

To formalize this, Blum, Shub, and Smale [2] introduced Hilbert's Nullstellensatz Problem: Hilbert's Nullstellensatz Problem HN_R over some ring R asks whether a given set of multivariate polynomials over R has a common solution in R. $\mathsf{HN}_{\mathbb{F}_2}$ defines the class NP, since every Boolean formula can be arithmetized appropriately. Over the real numbers, $\mathsf{HN}_{\mathbb{R}}$ is equivalent to the existential theory of the reals $\exists\mathbb{R}$, which is astonishing, since in $\exists\mathbb{R}$, we can define semi-algebraic sets. We can view HN_R as a parameterized complexity class: for every ring R, we get a class by taking the downward closure of HN_R under polynomial-time many-one reductions. In this work, we show that for many important problems from optimization and algebra, formulating them as a system of polynomials equations is optimal, since we can reduce Hilbert's Nullstellensatz to them.

We first show that two well-known problems from algebra are equivalent to or at least as hard as Hilbert's Nullstellensatz over fields. The first one is the Affine Polynomial Projection Problem, which given two polynomials, asks whether one of them can be transformed into the other by an affine linear projection of the variables. Kayal [15] proves that this problem is NP-hard. Here we improve this lower bound by showing that it is harder than HN_F for any field F. As a corollary, we obtain that this problem is complete for the existential theory of the reals, which answers an open question asked in [30, A14, open question]. The second problem is the Sparse Shift Problem, which asks whether for a given polynomial, there is an affine shift that reduces the number of monomials. For integral domains R that are not fields, Chillara, Grichener, and Shpilka [7] showed that this problem is hard for HN_R. They left it as an open problem whether this is also true for fields. We here resolve this question in the positive. Over infinite fields F, where HN_F is complete for NP_F (in the BSS model), we show that the Sparse Shift Problem is even equivalent to HN_F.

Second, we turn to an important case of Hilbert's Nullstellensatz over the real numbers. Real-stable polynomials have been a successful tool in mathematics and computer science in the recent years. For instance, the solution to the Kadison-Singer problem by Marcus, Spielman, and Srivastava [21] is deeply connected to real stable polynomials, and so is the improvement of the approximation performance of metric TSP by Karlin, Klein, and Oveis Gharan [14], just to give two examples. We here prove that testing whether a given polynomial is real stable is equivalent to the complement of $\mathsf{HN}_{\mathbb{R}}$, or equivalently, to the universal theory of the reals $\forall\mathbb{R}$. We show that the same is true for testing convexity and testing hyperbolicity, as well as for testing whether a biquadratic form is nonnegative, completely settling the complexity of all of these problems.

*Email: mblaeser@cs.uni-saarland.de.
†Email: sadutta@mpi-inf.mpg.de.
‡Email: gjindal@mimuw.edu.pl.

1 Introduction

Solving systems of polynomial equations is a central technique in both optimization and computational algebra, serving as a unifying framework for a wide range of algorithmic problems. Many of the most efficient algorithms in these domains are built upon the insight that complex problems—ranging from feasibility in nonlinear programming to questions in algebraic geometry—can be reduced to the task of solving a polynomial system.

This approach is powerful because it takes advantage of the existence of surprisingly efficient algorithms for deciding the solvability of polynomial systems. Over the complex numbers, Koiran [17] demonstrated that the feasibility of polynomial systems lies in the class AM (Arthur-Merlin games), assuming the Generalized Riemann Hypothesis (GRH). This result (conditionally) places the problem in a probabilistic class that is believed to be much smaller than PSPACE. Canny [6] and Renegar [26] developed PSPACE-algorithms for deciding the feasibility over real numbers. These results highlight that, despite the apparent algebraic and geometric complexity of polynomial systems, their computational complexity can be surprisingly low. As a result, formulating problems as polynomial systems not only provides a unifying language, but also yields algorithms whose complexity is as low as possible. Note that in this context, even an upper bound like PSPACE is surprisingly low, since real numbers are infinite objects and until the work of Tarski [36], it was not even clear whether this problem was decidable at all. To formalize this, Blum, Shub, and Smale, see [2], introduced Hilbert's Nullstellensatz Problem: For an arbitrary ring R, the problem is defined as follows:

HN_R (Hilbert's Nullstellensatz over ring R) : Given a set of polynomial equations $f_1 = 0, \cdots, f_t = 0$ where each f_i is from $R[x_1, \cdots, x_n]$, decide if there exists a vector $(a_1, \cdots, a_n) \in R^n$ such that $f_i(a_1, \cdots, a_n) = 0$ for all $i \in [t]$.

This problem is also called the Polynomial Feasibility Problem, but in the present paper, we will use the term Hilbert's Nullstellensatz as coined by Blum, Shub and Smale. When R is an arbitrary ring, the coefficients of the input polynomials might not be efficiently representable on a Turing machine. Therefore, it makes more sense to use the Blum-Shub-Smale (BSS) model of computation where the registers can store arbitrary elements of the ring and each ring operation is assumed to cost unit time. However, often it's possible to consider the problem on the Turing machine model as well. If R is a finite field, then the coefficients of the input polynomials will have constant bit complexity. If $R = \mathbb{R}$ or $\mathbb{C}$, we can assume the coefficients of the inputs to be integers. For other rings, we can restrict the coefficients to be ring elements with small bit complexity. In such settings, $\mathsf{HN}_{\mathbb{F}}$ is known to be NP-hard for all fields $\mathbb{F}$.

The problem of HN over the real field $\mathbb{R}$ is particularly interesting because it is complete for the well-studied complexity class $\exists\mathbb{R}$. This class captures the complexity of deciding the truth of sentences in the existential theory of the reals (ETR). Formally, the ETR language consists of all true sentences of the form $\exists x_1, \ldots, x_n\ \phi(x_1, \ldots, x_n)$, interpreted over the reals, where ϕ is a quantifier-free formula written as a Boolean combination of atomic formulas involving the constants $0, 1$, variables, the operations $+, \cdot$, and the comparison operators $<, >, \leq, \geq, =$. The class $\exists\mathbb{R}$ is the set of all languages that are polynomial-time many-one reducible to the ETR language. Similarly, there is also the universal theory of reals, defined by its complete problem—the negation of $\mathsf{HN}_{\mathbb{R}}$, i.e., the problem of deciding whether a system of polynomials has no solutions over the reals. This theory corresponds to the class $\forall\mathbb{R}$, which captures the complexity of deciding the truth of universally quantified sentences over the reals. Over the past decade, these two classes have received significant attention in algorithms and complexity theory due to their ability to precisely characterize the complexity of numerous natural problems in fields such as algebra, computational geometry, game theory, and machine learning, among others. For a comprehensive account of $\exists\mathbb{R}$-complete and $\forall\mathbb{R}$-complete problems, see the survey [30].

For $\mathsf{HN}_{\mathbb{R}}$, the best known upper bound is due to Canny [6] and Renegar [26], who placed it in PSPACE. Similarly, the best known unconditional upper bound for $\mathsf{HN}_{\mathbb{C}}$ is also PSPACE. However, under the Generalized Riemann Hypothesis, Koiran [17] showed that $\mathsf{HN}_{\mathbb{C}}$ lies within the second level of the polynomial hierarchy (PH), and even AM. Matiyasevich [23] established that $\mathsf{HN}_{\mathbb{Z}}$ is undecidable, while the decidability of $\mathsf{HN}_{\mathbb{Q}}$ remains a longstanding open problem. These results illustrate that Hilbert's Nullstellensatz is generally much harder than NP. As a result, strengthening NP-hardness (or coNP-hardness) results by proving equivalence to Hilbert's Nullstellensatz (or its negation) significantly enhances our understanding of the complexity of the concerned problems and also shows that formulating them as systems of polynomial equations is optimal.

In this paper, we go beyond previously known NP-hardness and coNP-hardness results. We prove that many well-known open problems—previously only known to be NP-hard or coNP-hard—are in fact $\exists\mathbb{R}$-complete or $\forall\mathbb{R}$-complete. This establishes a significantly higher level of complexity for these problems. See Section 2 for the precise statements of our results.

1.1 Affine Polynomial Projection Problem

One of the key questions in Algebraic Complexity Theory is the problem of checking the equivalence of polynomials under affine transformations. For a field $\mathbb{F}$, the problem is defined as follows:

Problem 1.1 ($\mathsf{PolyProj}_{\mathbb{F}}$ (Affine polynomial projection))**.** *Given polynomials $f \in \mathbb{F}[y_1, \cdots, y_m], g \in \mathbb{F}[x_1, \cdots, x_n]$, decide if there exists an $m \times n$ matrix A and a vector $b \in \mathbb{F}^m$ such that $f(A\mathbf{x} + b) = g(\mathbf{x})$.*

The central problem of VP vs VNP in Algebraic Complexity Theory is an instance of PolyProj. We will have $\mathsf{VP} \neq \mathsf{VNP}$ if the permanent polynomial $\mathsf{Perm}_m = \sum_{\pi \in S_m} \prod_{i=1}^{m} x_{i\pi(i)}$ cannot be written as an affine projection of the determinant polynomial $\mathsf{Det}_n = \sum_{\pi \in S_n} \text{sign}(\pi) \prod_{i=1}^{n} x_{i\pi(i)}$ whenever $n = 2^{(\log m)^{O(1)}}$ [4, Chapter 2]. The classical problem of matrix multiplication is also an instance of PolyProj. Matrix multiplication will have an $\tilde{O}(n^2)$ algorithm if the matrix multiplication polynomial $\mathsf{Mat}_n = \sum_{1 \leq i,j,k \leq n} x_{ij} y_{jk} z_{ki}$ can be written as an affine projection of the sum of products polynomial $\mathsf{SP}_m = \sum_{i=1}^{m} x_{i1} x_{i2} x_{i3}$ for $m = \tilde{O}(n^2)$ [34, 5].

Kayal [15] showed that $\mathsf{PolyProj}_{\mathbb{F}}$ is NP-hard in general. In this paper, we improve his result by showing that it is $\mathsf{HN}_{\mathbb{F}}$-hard for all fields $\mathbb{F}$. When $\mathbb{F} = \mathbb{R}$, it implies that the affine projection problem is $\exists\mathbb{R}$-complete, which solves [30, A14, open question].

1.2 Finding Sparse Shift of Polynomials

Since the problem of polynomial equivalence under affine projections is hard in general, different restricted formulations of the problem have been considered. Kayal in [15] showed randomized polynomial-time algorithms for PolyProj when f is the permanent/determinant polynomial and A is required to be full rank. If A is required to be the identity matrix, then we obtain the following problem:

Problem 1.2 (Shift Equivalence Testing (SET))**.** *Given $f, g \in R[x_1, \cdots, x_n]$, decide if there exists a shift $b \in R^n$ such that $f(\mathbf{x} + b) = g(\mathbf{x})$.*

For polynomials of degree d and n variables, Grigoriev [11] gave three algorithms for SET: a deterministic algorithm for characteristic 0, a randomized algorithm for large enough characteristic $p > 0$ and a quantum algorithm for characteristic 2. All these algorithms run in time polynomial in $\binom{n+d}{d}$. Dvir, Oliviera and Shpilka [9] showed that given blackbox access to f and g, there is a randomized algorithm of running time $\text{poly}(n, d, s)$ where n is the number of variables, d is the degree bound and s is the size bound on the circuits for f and g. The only randomized part of their algorithm is where they solve instances of Polynomial Identity Testing (PIT) and their algorithm can be derandomized if and only if PIT can be derandomized.

A variant of SET asks if a given polynomial $f \in \mathbb{F}[x_1, \cdots, x_n]$ has a shift $b \in \mathbb{F}^n$ such that $f(\mathbf{x} + b)$ has at most t monomials for some parameter t. Such a shift is called a t-sparse shift. Lakshman and Saunders [18] considered this problem for univariate polynomials over fields of characteristic zero and produced $\text{poly}(t, d)$ time algorithms. Grigoriev and Lakshman extended this result to multivariate polynomials. Their algorithm computes t-sparse shifts for n-variate polynomials in deterministic running time $(nt)^{O(n^2)}$. After two decades of no improvement over this exponential time algorithm, Chillara, Grichener and Shpilka [7] were motivated to study the hardness of this problem and they considered the following slightly more general version of it:

Problem 1.3 ($\mathsf{SparseShift}_R$ (Sparsification of polynomial via shift))**.** *Given a polynomial $f \in R[x_1, \cdots, x_n]$, decide if there exists a vector $(a_1, \cdots, a_n) \in R^n$ such that $f(x_1 + a_1, \cdots, x_n + a_n)$ has strictly fewer monomials than $f(x_1, \cdots, x_n)$.*

They showed in [7] that $\mathsf{SparseShift}_R$ is HN_R-hard when R is an integral domain that is not a field. Their technique crucially relies on the non-invertibility of one element inside the integral domain. In this paper, we circumvent that problem and show that $\mathsf{SparseShift}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-hard for all infinite fields $\mathbb{F}$.

1.3 Convexity of Polynomials and Biquadratic Non-negativity

Convexity plays a key role in modern mathematical optimization, often serving as a criterion for an optimization problem's tractability. Many combinatorial optimization problems can be formulated as polynomial optimization problems, where the objective function is a polynomial and the constraints are also described by polynomials. However, determining whether a given polynomial optimization problem exhibits convexity is not always straightforward.

Convexity verification has important practical implications. In polynomial global minimization, verifying convexity ensures that every local minimum is global, simplifying optimization. We recall the formal definition of convexity now. A polynomial function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be *convex* if it satisfies the convexity condition:

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y), \quad \forall x, y \in \mathbb{R}^n, \forall \lambda \in [0,1] .$$

Equivalently, $f(x)$ is convex if and only if its Hessian matrix

$$H_f(x) = \nabla^2 f(x)$$

is *positive semidefinite* for all $x \in \mathbb{R}^n$. The convexity of odd-degree polynomials can be easily determined: linear polynomials are trivially convex, while polynomials of odd degree greater than one can never be convex. Moreover, for quadratic polynomials, checking the convexity means checking the positive semidefiniteness of its constant Hessian matrix, which can be done in polynomial time. Thus, quartic (degree 4) polynomials present the first non-trivial case where determining complexity of convexity testing remained open for a long time. This question was posed as one of seven open problems in complexity theory for numerical optimization in 1992 [24] and only in 2013, [1] made a major advance by showing this problem to be NP-hard under Turing reductions and coNP-hard under Karp reductions.

Problem 1.4 (Convexity Testing of Quartic Polynomials)**.** *Given an n-variate polynomial of degree 4, decide if it is convex.*

The hardness proof in [1] proceeds through a reduction from Problem 1.4 to the problem of deciding non-negativity of biquadratic forms. A biquadratic form $b(x, y)$ is a polynomial in the variables $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_m)$ that can be written as:

$$B(x,y) = \sum_{i \leq j, k \leq l} \alpha_{ijkl} x_i x_j y_k y_l.$$

for some $\alpha_{ijkl} \in \mathbb{R}$.

Problem 1.5 (Nonnegativity Testing of Biquadratic Forms)**.** *Given a biquadratic form $B(x_1, \ldots, x_n, y_1, \ldots, y_m)$, determine whether it is non-negative, i.e., decide whether*

$$B(x_1, \ldots, x_n, y_1, \ldots, y_m) \geq 0$$

for all $(x_1, \ldots, x_n, y_1, \ldots, y_m) \in \mathbb{R}^{n+m}$.

[19] showed Problem 1.5 to be coNP-hard under Karp reductions (and NP-hard under Turing reductions) and this implied the hardness result for convexity testing in [1]. [19] actually treated a more general version of Problem 1.5 where one asks for the minimum objective value of a biquadratic form $b(x, y)$ over the unit bi-sphere $\{(x, y) \in \mathbb{R}^{2n} : \|x\| = \|y\| = 1\}$. This biquadratic optimization problem is interesting in its own right. It arises from the strong ellipticity condition problem in solid mechanics [16, 27, 33, 37] and the fundamental quantum theoretic problem of determining if a quantum state is entangled [12].

For an upper bound, [6] showed that Problem 1.4 is in PSPACE. We make significant progress on both of these problems by showing their $\forall\mathbb{R}$-completeness, hence settling their complexity; see Section 2 for more details.

1.4 Checking Hyperbolicity and Real Stability of Polynomials

Real stable polynomials have proved to be a successful tool in some of the major mathematical advances. For instance, their theory was used to resolve the well-known Kadison-Singer problem [21] and to construct infinite families of Ramanujan graphs [22, 20]. These polynomials have also found application in improvement of approximation algorithms for metric TSP [14].

A multivariate real polynomial p is said to be *real stable* if it has no roots in the open upper half of the complex plane. Equivalently, p is *real stable* if and only if the univariate polynomial

$$p(te_1 + x_1, te_2 + x_2, \ldots, te_n + x_n)$$

is real-rooted for all choices of $e_1, \ldots, e_n > 0$ and $x_1, \ldots, x_n \in \mathbb{R}$. It can also be shown as an exercise that a real univariate polynomial is *real stable* if and only if it is real-rooted, meaning all its roots are real. This concept of *real stability* plays a fundamental role in probability theory, control theory, and various other areas of mathematics.

Hyperbolic polynomials generalize the concept of *real stable* polynomials. A homogeneous n-variate polynomial p is said to be *hyperbolic with respect to a vector* $e \in \mathbb{R}^n$ if $p(e) > 0$ and $\forall x \in \mathbb{R}^n$, the univariate polynomial $p(x + te) \in \mathbb{R}[t]$ is real-rooted. Hyperbolic polynomials are of great interest in optimization theory due to their generality. They define barrier functions for interior point methods in hyperbolic optimization, which is a powerful generalization of semidefinite programming. In this paper, we look at real stable and hyperbolic polynomials from a computational complexity perspective. Specifically, we aim to characterize the complexity of the following problems:

Problem 1.6 (Real Stability Testing)**.** *Given a real multivariate polynomial p, decide whether p is real stable.*

Problem 1.7 (Hyperbolicity Testing)**.** *Given a homogeneous multivariate polynomial p and a direction $e \in \mathbb{R}^n$, decide whether p is hyperbolic with respect to e.*

A polynomial-time algorithm for the univariate variant of Problem 1.6 follows immediately from [35]. The bivariate case of Problem 1.6 was shown to be decidable in polynomial time by Raghavendra, Ryder and Srivastava [25]. Then it is a natural question to ask about the complexity of Problem 1.6 and Problem 1.7 for multivariate polynomials. To this end, it was shown by Saunderson [28] that Problem 1.7 for cubic multivariate polynomials is coNP-hard.

For Problem 1.6, Chin [8] established the coNP-hardness of determining the real stability of homogeneous polynomials. Specifically, it is proven in [8] that deciding real stability for polynomials of degree $d \geq 3$ with rational coefficients is coNP-hard. For a more detailed discussion, see [25, 8] and the references therein. In this paper, we significantly improve these coNP-hardness results and prove that both of these problems are $\forall\mathbb{R}$-complete for quartic polynomials. Note that the prior works established coNP-hardness results for hyperbolicity and real stability of cubic polynomials whereas our reductions achieve $\forall\mathbb{R}$-hardness for quartic polynomials. We leave it open whether $\forall\mathbb{R}$-hardness of these problems can be shown for cubic polynomials as well.

2 Our Results

Before stating our results, we should discuss the issue of how the input polynomials are provided. We are mainly concerned with three kinds of white-box representations of the input polynomials: the *dense representation* where we are given a list of all the coefficients, the *sparse representation* where we are given a list of all the non-zero coefficients along with the corresponding monomials, and the *arithmetic circuit representation* where we are given a circuit computing the polynomial. For more details, see Section 3.

Our first result gives a reduction from Hilbert's Nullstellensatz to the Sparse Shift Problem under all of the above representations.

Theorem 2.1. *For all infinite fields $\mathbb{F}$, $\mathsf{SparseShift}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-hard in any of the white-box representations.*

This extends the result of [7] where they showed $\mathsf{SparseShift}_R$ is HN_R-hard for all integral domains R that are not fields. When R is a non-field, it contains an element without any inverse and this element was crucially used in the reduction of [7] to build the polynomial for the SparseShift instance. We produce a reduction that doesn't rely on the existence of such non-invertible elements.

Remark 2.1. *Our proof of Theorem 2.1 actually shows that for a field $\mathbb{F}$ of size at least $4n^4$, $\mathsf{HN}_{\mathbb{F}}$ on n variables can be reduced to $\mathsf{SparseShift}_{\mathbb{F}}$ in $\mathrm{poly}(n, s)$ time where s is the total size of the white-box representations of the input polynomials. Therefore, given an HN instance on n variables over a finite field $\mathbb{F}_q$, we can consider an extension field $K_{(n)}$ of $\mathbb{F}_q$ of size at least $4n^4$. If we can show that HN over $\mathbb{F}_q$ reduces to HN over $K_{(n)}$, then we obtain the interesting poly-time reduction of $\mathsf{HN}_{\mathbb{F}_q}$ on n variables to $\mathsf{SparseShift}_{K_{(n)}}$. This is indeed possible since a system $\{f_i(x_1, \cdots, x_n) = 0\}_{i=1}^t$ has a solution over $\mathbb{F}_q$ if and only if the system $\{f_i(x_1, \cdots, x_n) = 0\}_{i=1}^t \cup \{x_i^q - x_i = 0\}_{i=1}^n$ has a solution over $K_{(n)}$.*

Under the sparse or dense representation, $\mathsf{SparseShift}_R$ is in NP_R, the analogue of NP for the BSS model of computation. This is because given a vector $(a_1, \cdots, a_n)$ and the sparse representation of a polynomial $g(x_1, \cdots, x_n)$, we can efficiently verify if $f(x_1 + a_1, \cdots, x_n + a_n) = g(x_1, \cdots, x_n)$ using sparse polynomial interpolation and sparse polynomial identity testing. Thus, we have the following corollary of Theorem 2.1.

Corollary 2.2. *Let $\mathbb{F}$ be an infinite field such that $\mathsf{HN}_{\mathbb{F}}$ is $\mathsf{NP}_{\mathbb{F}}$-complete. Then, $\mathsf{SparseShift}_{\mathbb{F}}$ is $\mathsf{NP}_{\mathbb{F}}$-complete and equivalent to $\mathsf{HN}_{\mathbb{F}}$.*

[3] showed that $\mathsf{HN}_{\mathbb{F}}$ is $\mathsf{NP}_{\mathbb{F}}$-complete when $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Thus, SparseShift is complete for the existential theory of these fields. Our next result shows that the affine polynomial projection problem is harder than Hilbert's Nullstellensatz over all fields.

Theorem 2.2. *For all fields $\mathbb{F}$, the problem $\mathsf{PolyProj}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-hard under both sparse and arithmetic circuit representations.*

Kayal proved in [15] that this problem is NP-hard. Theorem 2.2 improves upon this lower bound. We also have a reduction from PolyProj to HN when the underlying field is $\mathbb{R}$ or $\mathbb{C}$. Given f and g, we need to guess the matrix A and the vector b such that $f(A\mathbf{x} + b) - g(\mathbf{x})$ is the zero polynomial and then we use the observation that polynomial identity testing reduces to the existential theory over the reals or the complex numbers. Therefore, we obtain as a

corollary of Theorem 2.2 that $\mathsf{PolyProj}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-complete when $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. This solves [30, A14, open question] which asked if PolyProj is complete for the existential theory of reals.

Ling, Nie, Qi and Ye [19] showed that it is coNP-hard to compute the minimum value of a biquadratic objective function over the bi-sphere. With a minor adaptation, their proof also implies coNP-hardness of deciding if a biquadratic form is non-negative. The following theorem shows that this problem is in fact complete for the universal theory of reals.

Theorem 2.3. *Deciding non-negativity of a biquadratic form is $\forall\mathbb{R}$-complete.*

In [31, Lemma 2.8], Schaefer and Stefankovic showed through an involved construction that the problem of deciding non-negativity of a degree 6 polynomial is $\forall\mathbb{R}$-hard. While they also described how to reduce the degree in the construction to 4 [31, Remark 2.9], the polynomial constructed is not a biquadratic form. Since biquadratic forms have a very restricted structure, we need an even more intricate construction for Theorem 2.3.

[1] reduced the problem of deciding non-negativity of biquadratic forms to deciding convexity of quartic polynomials, thereby obtaining coNP-hardness of the latter problem from that of the former. Using their reduction, we get $\forall\mathbb{R}$-hardness of convexity testing as a corollary of Theorem 2.3.

Theorem 2.4. *Deciding convexity of a degree 4 polynomial is $\forall\mathbb{R}$-complete.*

Note that the papers [19] and [1] claim to prove only NP-hardness of the problems of deciding non-negativity of biquadratic forms and deciding convexity of quartic polynomials, respectively. However, their proofs use Turing reductions instead of polynomial-time many-one reductions (Karp reductions), which we are more interested in. In fact, their arguments imply coNP-hardness of these problems under Karp reductions.

Saunderson [28] proved that the problem of deciding if a cubic homogeneous polynomial is hyperbolic with respect to a vector is coNP-hard. We show that if you allow quartic polynomials in the reduction, then this problem is $\forall\mathbb{R}$-hard as well.

Theorem 2.5. *Given a homogeneous quartic polynomial p and a vector e, the problem of deciding if p is hyperbolic with respect to e is $\forall\mathbb{R}$-complete.*

Chin [8] showed that testing hyperbolicity of the polynomial in Saunderson's construction reduces to testing real stability of a polynomial, thereby leveraging Saunderson's result to demonstrate coNP-hardness of testing real stability. Using Chin's approach, we get $\forall\mathbb{R}$-hardness of real stability testing as a corollary of Theorem 2.5.

Theorem 2.6. *Given a homogeneous quartic polynomial p, the problem of deciding if p is real stable is $\forall\mathbb{R}$-complete.*

Remark 2.3. *For Theorems 2.3 to 2.6, we deal with constant degree polynomials, hence all the white-box representations have the same complexity up to polynomial factors and it does not matter which one we use.*

2.1 Proof Idea for Theorem 2.1

Throughout this paper, the terms "monomial count" and "number of monomials" are used interchangeably to denote the number of nonzero terms in a polynomial.

Our goal is to establish a reduction from the HN problem to the SparseShift problem by constructing a polynomial Q_S such that Q_S has a sparse shift if and only if the given system S of polynomial equations has a solution.

In [7], they show such a reduction when the underlying ring is an integral domain, but not a field, and hence contains an element γ without any multiplicative inverse. For structured polynomial systems

$$S : \{g_1(x_1, \cdots, x_N) = 0, \cdots, g_t(x_1, \cdots, x_N) = 0\},$$

they show that the polynomial

$$Q_S(x_0, \cdots, x_N, w_1, \cdots, w_t) := w_1 \cdot g_1(\mathbf{x}) + \sum_{i=2}^{t} w_i \cdot (\gamma g_i(\mathbf{x}) + \sum_{k=0}^{N} x_k)$$

has a sparse shift if and only if S has a solution. The non-invertibility of γ helps in showing that after a shift, no cancellation of the monomials takes place if S is not satisfiable. A straightforward idea for extending this approach to fields could be to treat γ as a variable itself and try to make the same proof work. However, now γ itself can be shifted as well, which presents a serious obstacle to the proof. Therefore, we needed new ideas and more complex machinery to design Q_S, which we briefly sketch below.

First, we utilize the *Structure Lemmas* from [7] to transform S into an equivalent system where all polynomials have degree at most 2, and at most one equation has a nonzero constant term. This structured form simplifies the analysis

and polynomial construction. Next, we introduce a new set of variables $\boldsymbol{\alpha}$ corresponding to the monomials of degree at most 2. The original system is then rewritten in terms of these new variables, leading to an equivalent system $S_1 \cup S_2 \cup S_3$, where:

- S_1 captures the original equations reformulated in terms of the new variables.
- S_2 encodes quadratic consistency constraints.
- S_3 ensures a specific sum condition on the variables.

It is then shown that the original polynomial system S has a solution if and only if the system $S_1 \cup S_2 \cup S_3$ has one. We then introduce two additional sets of variables, $\boldsymbol{\beta}$ and Y.

Next, we construct a polynomial P_S over the expanded set of variables $(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$ such that a sparsifying shift (satisfying certain conditions) exists for P_S if and only if S has a solution. This is achieved by analyzing how the monomial count of P_S changes under shifts in its variables. By carefully tracking these changes, we establish conditions under which a shift reduces monomial count.

Finally, we construct Q_S such that the existence of a sparsifying shift for P_S (under a particular linear constraint) directly corresponds to the existence of a sparsifying shift for Q_S in the unrestricted setting, thereby completing the reduction.

One of the key ideas in our reduction is *amplification of the monomial count increase* in different parts of the polynomial under shift. This allows us to enforce the satisfiability of an equation in the following way: if the equation is not satisfied by a shift, then the monomial count of the corresponding part increases and we can amplify this monomial count increase heavily so that there is an incentive against not satisfying the equation. This amplification is achieved as follows: let P be a polynomial in the variables $x_1, \ldots, x_m$. If we consider the polynomial

$$Q := \left(\sum_{i \in [k]} y_i \right) P(x_1, \ldots, x_m),$$

then, any shift that decreases the monomial count of Q can clearly be assumed to leave the y variables unchanged. Moreover, if a shift in the x variables increases the monomial count of P by ℓ, then the same shift increases the monomial count of Q by ℓk.

We apply this idea to various parts of the polynomial constructed in our reduction to penalize the number of monomials that can be added by a shift. The full proof, which is significantly more intricate and involves substantial technical challenges, is provided in Section 4. It requires carefully constructing the polynomial P_S and thoroughly analyzing how shifts affect its monomial count. For a complete understanding, we refer the reader to the detailed exposition in Section 4.

2.2 Proof Idea for Theorem 2.2

We are given a system $S = \{f_1(X) = 0, f_2(X) = 0, \cdots, f_t(X) = 0\}$ of polynomial equations defined over the variable set $X = \{x_1, \cdots, x_n\}$. Using Lemma 4.1, we can assume that each f_i is of degree at most 2. We introduce new variables $x_0, w_1, \cdots, w_t, z_1, \cdots, z_t$. Our polynomials f, g for the PolyProj instance will be defined over the variable vector $Y = (x_0, \cdots, x_n, w_1, \cdots, w_t, z_1, \cdots, z_t)$. For two increasing sequences of integers $d_1 < \cdots < d_t$ and $D_1 < \cdots < D_n$ satisfying $2d_t < D_1$, we define f, g as below:

$$f(Y) = \sum_{i=1}^{t} w_i^{d_i}(f_i(X) + z_i^{d_i}) + x_0 + \sum_{i=1}^{n} x_i^{D_i} \qquad \text{and}$$

$$g(Y) = \sum_{i=1}^{t} w_i^{d_i} z_i^{d_i} + x_0.$$

We show that S has a solution if and only if there exist a matrix $A \in \mathbb{F}^{|Y| \times |Y|}$ and a vector $b \in \mathbb{F}^{|Y|}$ such that $f(AY + b) = g(Y)$. Note that $AY + b$ is simply a vector of affine linear polynomials over Y.

If S has a solution $(a_1, \cdots, a_n)$, then the vector $Y' = (x_0 - \sum_{i=1}^{n} a_i^{D_i}, a_1, \cdots, a_n, w_1 \cdots, w_t, z_1, \cdots, z_t)$ satisfies

$$f(Y') = \sum_{i=1}^{t} w_i^{d_i}(0 + z_i^{d_i}) + x_0 - \sum_{i=1}^{n} a_i^{D_i} + \sum_{i=1}^{n} a_i^{D_i} = g(Y).$$

Thus, we can find a matrix A and a vector b such that $f(AY + b) = f(Y') = g(Y)$.

Conversely, suppose that $Y' = (L_0, \cdots, L_n, P_1, \cdots, P_t, Q_1, \cdots, Q_t)$ is a vector of affine linear polynomials satisfying $f(Y') = g(Y)$. The main goal now is to show that L_i must be constant for all i.

The proof proceeds by showing that each L_i must be a constant to maintain the degree constraints imposed by $g(Y)$. The crucial observation is that the highest degree terms in $f(Y')$ must match those in $g(Y)$, and this forces each L_i to be constant. More specifically, for $i = n$, if L_n were not constant, then $f(Y')$ would have terms of degree D_n, contradicting with the fact that $g(Y)$ has degree much smaller than D_n. Similarly, for smaller values of i, we use the fact that the highest degree term must be controlled at each step to show that L_i is constant.

Since L_i's are all constants, we now analyze P_iQ_i. We establish that $P_iQ_i = w_iz_i$ using the fact that any additional terms in the product would lead to monomials that are not present in $g(Y)$. By induction, this property propagates across all indices. From $P_iQ_i = w_iz_i$, we deduce that $f_i(L_1, \ldots, L_n) = 0$ for all i, meaning $(L_1, \ldots, L_n)$ is a valid solution of S.

2.3 Proof Idea for Theorem 2.3

To prove Theorem 2.3, we follow and generalize the ideas used in the proof of Lemma 2.8 in [31]. Specifically, we reduce an existing $\exists\mathbb{R}$-complete problem—determining whether a set of quadratic polynomials has a common root in the unit ball—to the problem of deciding whether a biquadratic form ever attains a negative value, we call this problem to be BiquadraticNegativity.

Recall that a biquadratic form over a variable set $X \sqcup Y$ is a degree 4 polynomial such that each monomial of it contains exactly two X-variables and exactly two Y-variables. Let us call a polynomial *semi-biquadratic* if every monomial of it contains at most two X-variables and at most two Y-variables. Our construction of the final biquadratic form from the system of quadratic polynomials goes through several stages. At each stage, we maintain a semi-biquadratic polynomial whose negativity is to be checked. Only at the final step, we appropriately homogenize the semi-biquadratic polynomial to make it a biquadratic form.

First, we establish a generalization of Lemma 2.7 in [31], stated as Lemma 6.1. This lemma provides a *rapid decay bound* for the sequence y_m in terms of its initial value y_0. It states that if the sequence $\{y_k, z_k\}$ satisfies a given inequality, then y_m decreases *doubly exponentially* as a function of m.

Now, suppose $f_1, f_2, \ldots, f_t$ are quadratic polynomials over a variable set x, and we want to check whether they have a common root in the unit ball. We first construct a semi-biquadratic polynomial $g(x, w)$ whose solutions correspond to common solutions of these quadratic polynomials, where w is a new set of variables. This polynomial is constructed such that if the polynomials f_i have a common solution inside the unit ball, then g has a root $(x, w) \in \mathbb{R}^{2n+2}$ satisfying $\sum_{i=1}^{n} x_i^2 \leq 1$. Conversely, if the polynomials f_i do not have a common solution inside the unit ball, we show that the minimum value of $g(x, w)$ on the semi-algebraic set $S := \{(x, w) \in \mathbb{R}^{2n+2} : \sum_{i=1}^{n}(x_i^2 + w_i^2) \leq 3\}$ can be lower-bounded by a doubly exponentially small number.

Following the proof strategy of Lemma 2.8 in [31], we now construct a new semi-biquadratic polynomial $h(x, w, y, z)$, which consists of $g(x, w)$ along with a modified version of the inequality used in Lemma 6.1.

Next, we prove that $h(x, w, y, z)$ attains negative values if and only if the polynomials f_i have a common solution inside the unit ball. The forward direction—showing that a common solution of the f_i's implies that $h(x, w, y, z)$ takes a negative value for some (x, w, y, z)—is straightforward. Conversely, if h takes negative values, then using the Lemma 6.1, we establish that $y_m^2 < 2^{-2^m}$, which forces $g(x, w)$ to be *extremely small* on the set S. Combining this with the previously established lower bound on the minimum value of $g(x, w)$ over S, we conclude that the polynomials f_i have a common solution inside the unit ball.

Finally, we homogenize the polynomial $h(x, w, y, z)$ using two variables s, t to obtain a biquadratic form $Q(x, w, y, z, s, t)$, completing the reduction and proving the $\exists\mathbb{R}$-hardness of BiquadraticNegativity.

2.4 Proof Idea for Theorem 2.5

The proof that hyperbolicity is in $\forall\mathbb{R}$ follows directly from the definition of hyperbolicity. Specifically, the condition that $p(x + te)$ is real-rooted for all x can be decided in $\forall\mathbb{R}$. To see this, observe that one can efficiently construct the real and imaginary parts of the evaluation of a real polynomial at a complex point. This holds even when p is given as an arithmetic circuit, ensuring that the problem remains in $\forall\mathbb{R}$.

To establish the $\forall\mathbb{R}$-hardness of hyperbolicity, we reduce the problem of deciding the non-negativity of a biquadratic polynomial to the problem of deciding hyperbolicity. By Theorem 2.3, the problem of deciding the non-negativity of a biquadratic polynomial is $\forall\mathbb{R}$-hard.

Given a biquadratic form Q, we construct a homogeneous polynomial of degree 4 and a vector e such that p is hyperbolic with respect to e if and only if Q is non-negative. In particular, we define the polynomial $p(x_0, \ldots, x_n)$ as:

$$p(x_0, \ldots, x_n) = x_0^4 - \beta x_0^2 \|x\|^2 + Q(x_1, \ldots, x_n)$$

for a suitably large constant β, and we take e to be the vector $(1, 0, \ldots, 0)$.

By Theorem 3.2, we know that p is hyperbolic with respect to e if and only if $B_{p,e}(x)$ is positive semidefinite for all x with $x_0 = 0$, where $B_{p,e}(x)$ is the parametrized Bézoutian matrix defined in Section 3.3. Now, we compute $B_{p,e}(x)$ and use the Schur complement to derive the condition for hyperbolicity. A simple calculation shows that p is hyperbolic with respect to e if and only if

$$\forall x_1, \ldots, x_n \in \mathbb{R}, \quad 0 \leq Q(x_1, \ldots, x_n) \leq \frac{\beta^2}{4}\|x\|^4.$$

The proof is completed by choosing β to be a sufficiently large constant so that the upper bound condition $Q(x_1, \ldots, x_n) \leq \frac{\beta^2}{4}\|x\|^4$ is always satisfied.

3 Preliminaries

3.1 Polynomials

A *polynomial* $f(x_1, \cdots, x_n)$ over a ring R is a finite R-linear combination of monomials:

$$f(x_1, \cdots, x_n) = \sum_{(e_1, \cdots, e_n) \in I} \alpha_{e_1, \cdots, e_n} x_1^{e_1} \cdots x_n^{e_n}$$

where the index set I is a finite subset of $\mathbb{Z}^n_{\geq 0}$ and the coefficients $\alpha_{e_1, \cdots, e_n}$ are elements of the ring R.

The degree of a monomial $x_1^{e_1} \cdots x_n^{e_n}$ is the sum $e_1 + \cdots + e_n$. A polynomial is called *homogeneous* if all of its monomials have the same degree.

A homogeneous degree-4 polynomial $Q(x_1, \cdots, x_n)$ is called a *biquadratic form* if the index set $[n]$ can be partitioned into two subsets S and T such that Q can be written as

$$Q(x_1, \cdots, x_n) = \sum_{i,j \in S} \sum_{k,l \in T} \alpha_{ijkl} x_i x_j x_k x_l \ .$$

From the computational perspective, there are a number of models to represent polynomials. An algorithm may access the input polynomial as black-box or it may have access to one of the following white-box representations of the polynomial:

- In the *dense representation*, an n-variate degree d polynomial is given as a list of the coefficients of its $\binom{n+d}{d}$ monomials.
- The *sparse representation* is more compact. Here, the polynomial is given as a list of pairs of monomials and coefficients. Hence, the list has length equal to the monomial count of the polynomial.
- In the *arithmetic circuit representation*, we are given an arithmetic circuit computing the polynomial. The complexity of the algorithm is considered in terms of the bit length of the circuit's description.

3.2 PSD-ness and Convexity

We have the notion of positive-semidefiniteness for both polynomials as well as matrices. A polynomial $p \in \mathbb{R}[x_1, \cdots, x_n]$ is said to be *positive semidefinite (PSD)* or *non-negative* if $p(x) \geq 0$ for all $x \in \mathbb{R}^n$. On the other hand, a real symmetric matrix $A_{n \times n}$ is called PSD if $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$. We denote this by $A \succeq 0$.

The *Hessian* of a polynomial $f \in \mathbb{R}[x_1, \cdots, x_n]$ is the $n \times n$ polynomial matrix $H_f(x)$ defined by $[H_f(x)]_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j}$. We call a polynomial f *convex* if and only if $H_f(x) \succeq 0$ for all $x \in \mathbb{R}^n$.

3.3 Hyperbolic and Real Stable Polynomials

Let $\mathbb{R}[x_1, \cdots, x_n]_d$ denote the set of real homogeneous polynomials of degree d. A polynomial $p \in \mathbb{R}[x_1, \cdots, x_n]_d$ is said to be *hyperbolic with respect to* $e \in \mathbb{R}^n$ if $p(e) > 0$ and $\forall x \in \mathbb{R}^n$, the univariate polynomial $p(x + te) \in \mathbb{R}[t]$ has only real roots.

If p is hyperbolic with respect to e, then we can define the *hyperbolicity cone*

$$\Lambda_+(p, e) = \{x \in \mathbb{R}^n \mid \text{all roots of } p(te - x) \text{ are positive}\} .$$

The following is a well-known result on hyperbolicity:

Theorem 3.1 ([13]). *If p is hyperbolic with respect to e, then it is also hyperbolic with respect to all $x \in \Lambda_+(p, e)$. Moreover, $\Lambda_+(p, e)$ is the connected component of $\{x \in \mathbb{R}^n \mid p(x) \neq 0\}$ containing e.*

For $p \in \mathbb{R}[x_1, \cdots, x_n]_d$ and $e \in \mathbb{R}^n$, the directional derivative of p in the direction of e is $D_e p(x) = \frac{d}{du} p(x+ue)|_{u=0}$. The *parametrized Bezoutian* $B_{p,e}(x)$ is defined to be the $d \times d$ symmetric matrix with polynomial entries given by the following identity:

$$\frac{p(x + te)D_e p(x + se) - p(x + se)D_e p(x + te)}{t - s} = \sum_{i,j=1}^{d} [B_{p,e}(x)]_{ij} t^{i-1} s^{j-1}$$

where s, t are variables.

The following theorem relates the hyperbolicity of a polynomial to the PSD-ness of the parametrized Bezoutian.

Theorem 3.2 ([28], Corollary 3.9). *Let $p \in \mathbb{R}[x_1, \cdots, x_n]_d$, $e \in \mathbb{R}^n$ and let $W \subseteq \mathbb{R}^n$ be a subspace of codimension 1 such that $e \notin W$. Then, p is hyperbolic with respect to e if and only if $B_{p,e}(x) \succeq 0$ for all $x \in W$.*

A polynomial $p \in \mathbb{C}[x_1, \cdots, x_n]$ is called *stable* if $p(z_1, \cdots, z_n) \neq 0$ whenever each z_i is a complex number with a positive imaginary part. Additionally, we say p is *real stable* if all the coefficients of p are real.

There is an equivalent definition of real stability which does not use the complex field. A polynomial $p \in \mathbb{R}[x_1, \cdots, x_n]$ is real stable if and only if for every point $a \in \mathbb{R}^n_{>0}$ and $b \in \mathbb{R}^n$, the univariate polynomial $q_{a,b}(t) = p(at + b)$ is not identically zero and has only real roots. If p is homogeneous (of positive degree), then the above condition is equivalent to p being hyperbolic with respect to $\mathbb{R}^n_{>0}$.

4 Reduction from HN to SparseShift

In this section, we prove Theorem 2.1. Given a system S of n-variate polynomial equations over a field $\mathbb{F}$ of size more than $4n^4$, we will show how to construct a polynomial Q_S such that Q_S admits a sparse shift if and only if S has a solution.

As mentioned in Section 2, [7] proved Theorem 2.1 over integral domains that are not fields. Crucially, their proof relies on the existence of a nonzero element γ that is not invertible. A natural idea to extend their argument to fields would be to treat γ as an additional variable. However, this introduces the complication that γ itself can be shifted, making it difficult to preserve the desired hardness property. Consequently, the hardness proof in [7] does not generalize to fields in a straightforward way.

In this section, we resolve this difficulty through a novel and technically involved construction. We will use the following lemma from [7], which lets us assume additional structure on the polynomial system S.

Lemma 4.1 ([7], Lemmas 7 and 8). *Let $S = \{f_i(X) = 0\}_{i=1}^r$ be a system of polynomial equations over an integral domain R where each f_i is given either as a list of coefficients or as an arithmetic circuit. In both cases, we can obtain in polynomial time a system T of polynomial equations $\{g_j(Y) = 0\}_{j=1}^t$ such that*

- *S has a solution in $R^{|X|}$ if and only if T has a solution in $R^{|Y|}$.*
- *Each $g_j(Y)$ is of degree at most 2.*
- *There is at most one $j \in [t]$ such that $g_j(Y)$ has a non-zero constant term.*

Remark 4.2. *Lemmas 7 and 8 of [7] describe how to obtain a system T satisfying only the first two conditions above. However, Section 3.2 of [7] describes a simple trick to achieve the third condition as well: if T contains the equations $g_1(Y) = 0, \cdots, g_t(Y) = 0$ with constant terms $c_1 \neq 0, c_2, \cdots, c_t$ respectively, then obtain T' from T by simply updating each polynomial $g_j(Y)$ as follows: $g_j(Y) \leftarrow c_1 g_j(Y) - c_j g_1(Y)$. Then, T' has a solution if and only if T does (since R is an integral domain) and T' contains only one polynomial with a non-zero constant term.*

Therefore, using Lemma 4.1, we can assume that we are given a system

$$S = \{f_1(X) = 0, f_2(X) = 0, \cdots, f_t(X) = 0\}$$

of polynomial equations over the variable set $X = \{x_1, \cdots, x_n\}$ where each f_i is of degree at most 2 and f_1 is the only polynomial with a non-zero constant term. Let

$$M_X \coloneqq \{x_j \mid 1 \le j \le n\} \cup \{x_j x_k \mid 1 \le j \le k \le n\}.$$

be the set of all *non-constant* monomials in the variables X of degree at most 2.

We can write $f_i = \sum_{m \in M_X} c_{i,m} m$ for $2 \le i \le t$ and $f_1 = c + \sum_{m \in M_X} c_{1,m} m$. First, note that we can assume $c \neq 0$; otherwise, the polynomial system S has the trivial solution $(0, \ldots, 0)$. Hence, we have $c \in \mathbb{F}^*$ and $c_{i,m} \in \mathbb{F}$.

Lemma 4.3. *Define a new variable set $\boldsymbol{\alpha}$ and new system of polynomial equations $S_1 \cup S_2 \cup S_3$:*

$$\begin{aligned}
\boldsymbol{\alpha} &\coloneqq \{\alpha_0\} \cup \{\alpha_m \mid m \in M_X\} \\
S_1 &\coloneqq \left\{ c + \sum_{m \in M_X} c_{1,m}\alpha_m = 0, \quad \sum_{m \in M_X} c_{2,m}\alpha_m = 0, \quad \cdots, \quad \sum_{m \in M_X} c_{t,m}\alpha_m = 0 \right\}, \\
S_2 &\coloneqq \left\{ \alpha_{x_i x_j} - \alpha_{x_i}\alpha_{x_j} = 0 \mid 1 \le i, j \le n \right\}, \\
S_3 &\coloneqq \left\{ \alpha_0 + \sum_{j=1}^{n} \alpha_{x_j} = 0 \right\}.
\end{aligned}$$

Then the original system S has a solution if and only if the system $S_1 \cup S_2 \cup S_3$ has a solution.

Proof. Suppose $(u_1, \ldots, u_n)$ is a solution to S. Set $\alpha_{x_i} \coloneqq u_i$, $\alpha_{x_i x_j} \coloneqq u_i u_j$, and $\alpha_0 \coloneqq -\sum_j u_j$. Then each equation in S_1 is satisfied since the polynomials f_i vanish at $(u_1, \ldots, u_n)$. The constraints in S_2 and S_3 hold by construction.

Conversely, suppose $(\alpha_0, (\alpha_m)_{m \in M_X})$ satisfies $S_1 \cup S_2 \cup S_3$, and define $u_i \coloneqq \alpha_{x_i}$. Then S_2 implies $\alpha_{x_i x_j} = u_i u_j$, and S_1 ensures that $f_i(u_1, \ldots, u_n) = 0$ for all i. Thus, $(u_1, \ldots, u_n)$ satisfies S. □

The construction of our polynomial for the SparseShift instance proceeds in two stages. First, we build a polynomial P_S with the following property: $S_1 \cup S_2 \cup S_3$ has a solution if and only if P_S can be sparsified through shift by a vector satisfying a certain system $\mathcal{L}$ of linear equations. In the second stage, we construct a polynomial Q_S such that P_S can be sparsified through shift by a vector satisfying $\mathcal{L}$ if and only if Q_S has a sparsifying shift. We begin by introducing some new objects and notation required for the construction of the polynomial P_S:

Define $N \coloneqq n^2 + 2n + 1$. Define a new set of variables

$$\boldsymbol{\beta} = \left\{ \beta_{k, x_i x_j} \mid 1 \le k \le N,\ 1 \le i, j \le n \right\}.$$

Next, jointly enumerate the following two families of quadratic polynomials:

- the polynomials from S_2, namely $\{\alpha_{x_i x_j} - \alpha_{x_i}\alpha_{x_j} \mid 1 \le i, j \le n\}$, and
- the auxiliary polynomials $\{\beta_{k, x_i x_j} - \alpha_{x_i}\alpha_{x_j} \mid 1 \le k \le N,\ 1 \le i, j \le n\}$,

as a single sequence $g_1, \ldots, g_r$. Define g_0 to be the first polynomial in the system S_1, written as

$$g_0 \coloneqq c + \sum_{m \in M_X} c_{1,m}\alpha_m.$$

Introduce three disjoint sets of new variables:

- $W = \{w_1, \ldots, w_r\}$, one weight variable per polynomial g_i,
- Y_1, a set of $n^2 + n + 1$ variables, and
- Y_2, a set of n variables.

Define $Y \coloneqq W \sqcup Y_1 \sqcup Y_2$ and supoose $\gamma_1, \ldots, \gamma_r$ are distinct nonzero elements of the field $\mathbb{F}$. This is possible since $r \leq (N+1)n^2 \leq 4n^4$, and we assume $|\mathbb{F}| > 4n^4$. For any variable set Z, we denote by $\mathsf{LIN}(Z) \coloneqq \sum_{z \in Z} z$ the sum of all variables in Z. The polynomial P_S is now defined over the variable set $\boldsymbol{\alpha} \sqcup \boldsymbol{\beta} \sqcup Y$ as below:

$$P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y) \coloneqq \underbrace{\mathsf{LIN}(Y_1) \cdot g_0(\boldsymbol{\alpha})}_{\text{Part 1}} + \underbrace{\sum_{i=1}^{r} w_i \left[g_i(\boldsymbol{\alpha}, \boldsymbol{\beta}) + \gamma_i \left(\alpha_0 + \sum_{j=1}^{n} \alpha_{x_j} \right) \right]}_{\text{Part 2}} + \underbrace{\mathsf{LIN}(Y_2) \cdot \sum_{i=1}^{n} \alpha_{x_i}^2}_{\text{Part 3}}.$$

Lemma 4.4. *Let $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$ be the polynomial defined above, and consider a shift $(a, b, y) \in \mathbb{F}^{|\boldsymbol{\alpha}|} \times \mathbb{F}^{|\boldsymbol{\beta}|} \times \mathbb{F}^{|Y|}$ of variables. Then, the following are true about the monomial count of $P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y)$:*

1. *If $g_0(a) = 0$, then the monomial count of Part 1 decreases by $|Y_1| = n^2 + n + 1$; otherwise, it remains unchanged.*
2. *If $a_{x_1} = \cdots = a_{x_n} = 0$, then the monomial count of Part 3 remains unchanged; otherwise, it increases by some $v \in [2n, (n+1)n]$.*
3. *If $a_0 + \sum_{j=1}^{n} a_{x_j} = 0$, then the monomial count of Part 2 changes by at least $v_1 - v_2$, where*
$$v_1 \coloneqq |\{i \in [r] : g_i(a, b) \neq 0\}|, \quad v_2 \coloneqq |\{j \in [n] : a_{x_j} = \gamma_i \text{ for some } i \in [r]\}|.$$
4. *For any $(a, b, y) \in \mathbb{F}^{|\boldsymbol{\alpha}|} \times \mathbb{F}^{|\boldsymbol{\beta}|} \times \mathbb{F}^{|Y|}$, the monomial count of $P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y)$ is **at least** the monomial count of $P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y)$.*

Proof. We proceed by analyzing how the shift affects the monomial count of each corresponding part of P_S.

Monomial count change in Part 1: Part 1 is

$$\mathsf{LIN}(Y_1) \cdot g_0(\boldsymbol{\alpha}) = \mathsf{LIN}(Y_1) \cdot \left(c + \sum_{m \in M_X} c_{1,m} \alpha_m \right)$$

After the shift $\boldsymbol{\alpha} \mapsto \boldsymbol{\alpha} + a$, it becomes

$$\mathsf{LIN}(Y_1) \cdot \left(g_0(a) + \sum_{m \in M_X} c_{1,m} \alpha_m \right)$$

If $g_0(a) = 0$, then the constant term vanishes and the only remaining terms are those with $\boldsymbol{\alpha}$, so the $|Y_1|$ monomials corresponding to cy_i disappear, causing the monomial count of Part 1 to decrease by exactly $|Y_1|$. Otherwise, these monomials remain, so the monomial count is unchanged. This proves (1).

Monomial count change in Part 3: Part 3 is

$$\mathsf{LIN}(Y_2) \cdot \sum_{i=1}^{n} \alpha_{x_i}^2.$$

Under the shift $\boldsymbol{\alpha} \mapsto \boldsymbol{\alpha} + a$, this becomes

$$\mathsf{LIN}(Y_2) \cdot \sum_{i=1}^{n} (\alpha_{x_i} + a_{x_i})^2 = \mathsf{LIN}(Y_2) \cdot \left(\sum_{i=1}^{n} (\alpha_{x_i}^2 + 2a_{x_i} \alpha_{x_i} + a_{x_i}^2) \right).$$

If $a_{x_i} = 0$ for all i, then no new terms are introduced, so monomial count remains the same. Otherwise, the terms $2a_{x_i}\alpha_{x_i}$ and the constants $a_{x_i}^2$ create new monomials, increasing monomial count by at least $2n$ and at most $(n+1)n$, depending on how many a_{x_i} are nonzero. Recall that $|Y_2| = n$. This proves (2).

Monomial count change in Part 2: Part 2 is

$$\sum_{i=1}^{r} w_i \left[g_i(\boldsymbol{\alpha}, \boldsymbol{\beta}) + \gamma_i (\alpha_0 + \sum_{j=1}^{n} \alpha_{x_j}) \right].$$

Each g_i is of the form $\delta - \alpha_{x_p}\alpha_{x_q}$, where δ is either $\alpha_{x_p x_q}$ or some $\beta_{k,x_p x_q}$. Under $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \mapsto (\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b)$, the i-th summand becomes

$$w_i\Big[\delta - \alpha_{x_p}\alpha_{x_q} + g_i(a,b) - a_{x_p}\alpha_{x_q} - a_{x_q}\alpha_{x_p} + \gamma_i\Big(\alpha_0 + \sum_{j=1}^{n}\alpha_{x_j} + a_0 + \sum_{j=1}^{n} a_{x_j}\Big)\Big].$$

Using $a_0 + \sum_j a_{x_j} = 0$, this simplifies to

$$w_i\Big[\delta - \alpha_{x_p}\alpha_{x_q} + g_i(a,b) + (\gamma_i - a_{x_p})\alpha_{x_q} + (\gamma_i - a_{x_q})\alpha_{x_p} + \gamma_i\Big(\alpha_0 + \sum_{j\in[n]\setminus\{p,q\}}\alpha_{x_j}\Big)\Big]$$

Now, we observe that if $g_i(a,b) \neq 0$, a new monomial $g_i(a,b)w_i$ appears, increasing the monomial count by 1; these contribute to v_1. If $a_{x_p} = \gamma_i$ or $a_{x_q} = \gamma_i$, then the terms $w_i\alpha_{x_q}$ or $w_i\alpha_{x_p}$ vanish, decreasing monomial count; these contribute to v_2. Hence, the net monomial count change in Part 2 is at least $v_1 - v_2$. This proves (3).

Proof of (4): Since P_S is linear in the Y variables, shifting $Y \mapsto Y + y$ can only add new monomials (by expanding linear terms with constants y), but cannot remove any existing monomial. Hence,

$$\text{Monomial count of } (P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y)) \geq \text{Monomial count of } (P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y)).$$

This completes the proof. □

Lemma 4.5 below completes the first stage of the proof by showing that the system $S_1 \cup S_2 \cup S_3$ has a solution if and only if the polynomial P_S can be sparsified using shifts that satisfy certain linear constraints.

Lemma 4.5. *The system $S_1 \cup S_2 \cup S_3$ has a solution if and only if there exists a vector $(a, b, y) \in \mathbb{F}^{|\boldsymbol{\alpha}|} \times \mathbb{F}^{|\boldsymbol{\beta}|} \times \mathbb{F}^{|Y|}$ satisfying the following linear equations:*

$$\sum_{m\in M_X} c_{2,m}\alpha_m = 0, \cdots, \sum_{m\in M_X} c_{t,m}\alpha_m = 0, \tag{1}$$

$$\alpha_{x_i x_j} - \beta_{k,x_i x_j} = 0 \quad \textit{for all } 1 \leq k \leq N,\ 1 \leq i, j \leq n, \tag{2}$$

$$\alpha_0 + \sum_{j=1}^{n}\alpha_{x_j} = 0. \tag{3}$$

such that the shifted polynomial $P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y)$ has strictly fewer monomials than $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$.

Proof. Suppose first that $S_1 \cup S_2 \cup S_3$ has a solution $a \in \mathbb{F}^{|\boldsymbol{\alpha}|}$. Define $b \in \mathbb{F}^{|\boldsymbol{\beta}|}$ by setting $b_{k,x_i x_j} := a_{x_i x_j}$ for all $1 \leq k \leq N$ and $1 \leq i, j \leq n$. Then the shift $(a, b, 0)$ reduces the number of monomials in $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$ because:

- the monomial count of *Part 1* decreases by $n^2 + n + 1$, which follows from (1) in Lemma 4.4,
- the monomial count of *Part 2* does not increase, because v_1 is zero in (3) of Lemma 4.4,
- the monomial count of *Part 3* increases by at most $(n + 1)n$, which follows from (2) in Lemma 4.4.

Also, the vector (a, b) satisfies:

- Equation (1), since a satisfies S_1,
- Equation (2), by the construction of b, and
- Equation (3), since a satisfies S_3.

Conversely, suppose there exists a vector (a, b, y) such that $P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y)$ has fewer monomials than $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$ and (a, b) satisfies Equations (1) to (3). Using (4) of Lemma 4.4, we may assume without loss of generality that $y = 0$.

We claim that a satisfies $S_1 \cup S_2 \cup S_3$. Suppose for contradiction that it does not. Since a satisfies the last $(t - 1)$ equations of S_1 (by Equation (1)) and S_3 (by Equation (3)), we must have one of the following cases:

Case 1 (When $a_{x_p x_q} \neq a_{x_p}a_{x_q}$ for some $p, q \in [n]$):

Then, by Equation (2), for all $k \in [N]$, we have $b_{k,x_p x_q} = a_{x_p x_q}$, so $b_{k,x_p x_q} - a_{x_p}a_{x_q} \neq 0$. Hence, (3) of Lemma 4.4 implies that the the quantity v_1 is at least N, and the monomial count of *Part 2* increases by at least $v_1 - v_2 \geq N - n$.

By again using (1) and (2) from Lemma 4.4, the monomial count of *Part 1* decreases by at most $n^2 + n + 1$, and that of *Part 3* does not decrease. Thus, the total monomial count increases by at least $N - n - (n^2 + n + 1) \geq 0$, a contradiction!

Case 2 (When $g_0(a) \neq 0$ and $a_{x_j} \neq 0$ for some $j \in [n]$): In this case, we have:

- Monomial count of *Part 1* remains unchanged, which follows from (1) of Lemma 4.4.
- Monomial count of *Part 2* decreases by at most $v_2 \leq n$, because v_1 is zero in (3) of Lemma 4.4.
- Monomial count of *Part 3* increases by at least $2n$, by using (2) of Lemma 4.4.

Thus, overall monomial count increases by at least $2n - n > 0$, a contradiction!

Case 3 (When $g_0(a) \neq 0$ and $a_{x_j} = 0$ for all $j \in [n]$): In this case, we have:

- Monomial count of *Part 1* and *Part 3* remain unchanged, which follows from (1) and (2) of Lemma 4.4.
- Monomial count of *Part 2* increases by v_1, because v_2 is zero in (3) of Lemma 4.4, as all the γ_i's are chosen to be non-zero.

In all the above cases, overall monomial count does not decrease, a contradiction! Therefore, a must satisfy $S_1 \cup S_2 \cup S_3$. □

We now complete the second stage of the proof. We enumerate the linear constraints from Equations (1) to (3) as a single list:

$$L_1(\boldsymbol{\alpha}, \boldsymbol{\beta}) = 0, \quad \ldots, \quad L_s(\boldsymbol{\alpha}, \boldsymbol{\beta}) = 0.$$

By Lemma 4.4, any shift can reduce the monomial count of P_S by at most

$$(n^2 + n + 1) + v_2 \leq n^2 + 2n + 1.$$

Let $M := n^2 + 2n + 1$ denote this upper bound. We now introduce s disjoint sets of new variables $Z_1, \ldots, Z_s$, each of size M, and let $Z := Z_1 \sqcup \cdots \sqcup Z_s$. We now define a new polynomial Q_S for our instance of SparseShift as follows:

$$Q_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y, Z) := P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y) + \sum_{i=1}^{s} \mathsf{LIN}(Z_i) \cdot L_i(\boldsymbol{\alpha}, \boldsymbol{\beta}).$$

Now we are ready to prove Theorem 2.1, which we restate below for convenience.

Theorem 2.1. *For all infinite fields $\mathbb{F}$, $\mathsf{SparseShift}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-hard in any of the white-box representations.*

Proof of Theorem 2.1. We know that the system S has a solution if and only if $S_1 \cup S_2 \cup S_3$ has a solution. By Lemma 4.5, it suffices to prove the following claim:

There exists $(a', b', y', z) \in \mathbb{F}^{|\boldsymbol{\alpha}|} \times \mathbb{F}^{|\boldsymbol{\beta}|} \times \mathbb{F}^{|Y|} \times \mathbb{F}^{|Z|}$ such that $Q_S(\boldsymbol{\alpha} + a', \boldsymbol{\beta} + b', Y + y', Z + z)$ has fewer monomials than $Q_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y, Z)$ if and only if there exists $(a, b, y) \in \mathbb{F}^{|\boldsymbol{\alpha}|} \times \mathbb{F}^{|\boldsymbol{\beta}|} \times \mathbb{F}^{|Y|}$ such that:

$$P_S(\boldsymbol{\alpha} + a, \boldsymbol{\beta} + b, Y + y) \text{ has fewer monomials than } P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y),$$
$$L_1(a, b) = \cdots = L_s(a, b) = 0.$$

Forward direction: Suppose there exists (a', b', y', z) such that $Q_S(\boldsymbol{\alpha} + a', \boldsymbol{\beta} + b', Y + y', Z + z)$ has fewer monomials than $Q_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y, Z)$. Each L_i is a constant-free linear polynomial. Therefore:

- If $L_i(a', b') = 0$, the monomial count of $\mathsf{LIN}(Z_i) \cdot L_i(\boldsymbol{\alpha}, \boldsymbol{\beta})$ remains unchanged after the shift.
- If $L_i(a', b') \neq 0$, then $L_i(\boldsymbol{\alpha} + a', \boldsymbol{\beta} + b')$ gains a constant term, so its monomial count increases by at least 1. Consequently, the monomial count of $\mathsf{LIN}(Z_i) \cdot L_i(\boldsymbol{\alpha}, \boldsymbol{\beta})$ increases by at least $|Z_i| = M$.

Since the monomial count of P_S can decrease by at most M, it follows that to achieve an overall monomial count decrease in Q_S, each L_i must vanish under the shift: $L_i(a', b') = 0$ for all $i \in [s]$. Thus, $P_S(\boldsymbol{\alpha} + a', \boldsymbol{\beta} + b', Y + y')$ must have fewer monomials than $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$, and the linear constraints are satisfied.

Reverse direction: Suppose (a, b, y) satisfies the conditions:

- $P_S(\boldsymbol{\alpha}+a, \boldsymbol{\beta}+b, Y+y)$ has fewer monomials than $P_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y)$,
- $L_1(a,b) = \cdots = L_s(a,b) = 0$.

Then, for each i, $L_i(\boldsymbol{\alpha}+a, \boldsymbol{\beta}+b)$ has the same monomial count as $L_i(\boldsymbol{\alpha}, \boldsymbol{\beta})$. Therefore, $\mathsf{LIN}(Z_i) \cdot L_i(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is unaffected by the shift. Set $(a', b', y', z) := (a, b, y, 0)$. It follows that $Q_S(\boldsymbol{\alpha}+a', \boldsymbol{\beta}+b', Y+y', Z+z)$ has fewer monomials than $Q_S(\boldsymbol{\alpha}, \boldsymbol{\beta}, Y, Z)$.

Finally, observe that Q_S has at most $\mathrm{poly}(n,t)$ monomials and can be constructed in polynomial time from the input. □

5 PolyProj reduces to HN

Theorem 2.2. *For all fields $\mathbb{F}$, the problem $\mathsf{PolyProj}_{\mathbb{F}}$ is $\mathsf{HN}_{\mathbb{F}}$-hard under both sparse and arithmetic circuit representations.*

Proof of Theorem 2.2. We show that PolyProj reduces to HN.

We are given a system $S = \{f_1(X) = 0, f_2(X) = 0, \cdots, f_t(X) = 0\}$ of polynomial equations defined over the variable set $X = \{x_1, \cdots, x_n\}$. Using Lemma 4.1, we can assume that each f_i is of degree at most 2.

Introduce new variables $x_0, w_1, \cdots, w_t, z_1, \cdots, z_t$. Our polynomials f, g for the PolyProj instance will be defined over the variable vector $Y = (x_0, \cdots, x_n, w_1, \cdots, w_t, z_1, \cdots, z_t)$. For $1 \le i \le t$, let $d_i \coloneqq i+1$ and for $1 \le i \le n$, let $D_i \coloneqq i + 2(t+1)$. Now, define

$$f(Y) = \sum_{i=1}^{t} w_i^{d_i}(f_i(X) + z_i^{d_i}) + x_0 + \sum_{i=1}^{n} x_i^{D_i} \qquad \text{and}$$

$$g(Y) = \sum_{i=1}^{t} w_i^{d_i} z_i^{d_i} + x_0.$$

We will show that S has a solution if and only if there exist a matrix $A \in \mathbb{F}^{|Y|\times|Y|}$ and a vector $b \in \mathbb{F}^{|Y|}$ such that $f(AY+b) = g(Y)$. Note that $AY+b$ is simply a vector of affine linear polynomials over Y.

If S has a solution $(a_1, \cdots, a_n)$, then the vector
$Y' = (x_0 - \sum_{i=1}^{n} a_i^{D_i}, a_1, \cdots, a_n, w_1 \cdots, w_t, z_1, \cdots, z_t)$ satisfies

$$f(Y') = \sum_{i=1}^{t} w_i^{d_i}(0 + z_i^{d_i}) + x_0 - \sum_{i=1}^{n} a_i^{D_i} + \sum_{i=1}^{n} a_i^{D_i} = g(Y)\,.$$

Thus, we can find a matrix A and a vector b such that $f(AY+b) = f(Y') = g(Y)$. Note that the resulting matrix A is not invertible as some of the variables are mapped to constants.

Conversely, let $Y' = (L_0, \cdots, L_n, P_1, \cdots, P_t, Q_1, \cdots, Q_t)$ be a vector of affine linear polynomials over Y satisfying $f(Y') = g(Y)$, i.e.,

$$\sum_{i=1}^{t} P_i^{d_i}(f_i(L_1, \cdots, L_n) + Q_i^{d_i}) + L_0 + \sum_{i=1}^{n} L_i^{D_i} = \sum_{i=1}^{t} w_i^{d_i} z_i^{d_i} + x_0\,.$$

Below we claim that such an equality is possible only when each L_i is a constant. The proof is postponed to the end of this section.

Claim 5.1. *For $1 \le i \le n$, L_i is a constant. As a result, $\sum_{i=1}^{n} L_i^{D_i} = c$ and for all $i \in [t]$, $f_i(L_1, \cdots, L_n) = c_i$ for some constants $c, c_1, \cdots, c_t$.*

By Claim 5.1, we have $f(Y') = \sum_{i=1}^{t} P_i^{d_i}(c_i + Q_i^{d_i}) + L_0 + c$ for the constants $c, c_1, \cdots, c_t$ defined in the claim.

For $1 \leq k \leq t$, define $f^{(k)}(Y') := \sum_{i=1}^{k} P_i^{d_i}(c_i + Q_i^{d_i}) + L_0 + c$ and $g^{(k)}(Y) := \sum_{i=1}^{k} w_i^{d_i} z_i^{d_i} + x_0$. Note that $f^{(t)}(Y') = f(Y')$ and $g^{(t)}(Y) = g(Y)$. We make another claim, whose proof is presented at the end of this section.

Claim 5.2. *For every k, the equality $f^{(k)}(Y') = g^{(k)}(Y)$ implies that $P_k Q_k = w_k z_k$.*

Since we have $f^{(t)}(Y') = f(Y') = g(Y) = g^{(t)}(Y)$, plugging $k = t$ into Claim 5.2 gives us $P_t Q_t = w_t z_t$. We show that more is true: in fact, $P_i Q_i = w_i z_i$ for all $i \in [t]$. This is proven using reverse induction on i.

The base case $i = t$ is true, as already mentioned. For the induction step, assume that $P_i Q_i = w_i z_i$ for all $i \geq k+1$. In that case, substituting $w_i = z_i = 0 \; \forall i \geq k+1$ into $f(Y') = g(Y)$ yields $f^{(k)}(Y') = g^{(k)}(Y)$. By our previous claim, it follows that $P_k Q_k = w_k z_k$.

Therefore, $P_i Q_i = w_i z_i$ for all $i \in [t]$. In that case, for any $i \in [t]$, $f(Y') = \sum_{i=1}^{t} P_i^{d_i}(c_i + Q_i^{d_i}) + L_0 + c$ would contain a monomial of degree d_i unless $c_i = 0$. Thus, $f_i(L_1, \cdots, L_n) = c_i = 0$ for all $i \in [t]$ and the constant vector $(L_1, \cdots, L_n)$ is our desired solution of S. □

Now we prove the claims made in the above proof.

Proof of Claim 5.1. For sake of contradiction, let $k \in [n]$ be the highest index such that L_k is not a constant.

Since L_i is a constant for all $i \geq k+1$, $f(Y') - L_k^{D_k}$ has degree at most $D_{k-1} < D_k$. Therefore, if L_k is not a constant, then $f(Y')$ must have degree D_k. This is a contradiction as $g(Y)$ has degree $2d_t < D_k$. Hence, our claim is true. □

Proof of Claim 5.2. Suppose $f^{(k)}(Y') = g^{(k)}(Y)$. Notice that $f^{(k)}(Y') - P_k^{d_k} Q_k^{d_k}$ has degree at most $2d_{k-1} < 2d_k$. Therefore, the highest degree term $w_k^{d_k} z_k^{d_k}$ of $g^{(k)}(Y)$ must come from $P_k^{d_k} Q_k^{d_k}$ i.e. $P_k Q_k$ should contain the monomial $w_k z_k$. However, $P_k Q_k$ can not contain any other monomial of degree 2 because then $f^{(k)}(Y')$ will contain an uncancelled monomial of degree $2d_k$ other than $w_k^{d_k} z_k^{d_k}$, which contradicts with the fact that $w_k^{d_k} z_k^{d_k}$ is the only monomial of degree $2d_k$ in $g^{(k)}(Y)$. Thus, $P_k Q_k = (w_k + a)(z_k + b)$ for some constants a, b. We want to show that $a = b = 0$. If $a \neq 0$, then $P_k^{d_k} Q_k^{d_k} = (w_k + a)^{d_k}(z_k + b)^{d_k}$ contains the monomial $w_k^{d_k - 1} z_k^{d_k}$. This monomial will remain uncancelled in $f^{(k)}(Y')$ as $f^{(k)}(Y') - P_k^{d_k} Q_k^{d_k}$ has degree at most $2d_{k-1} < 2d_k - 1$. This leads to a contradiction as $g^{(k)}(Y)$ doesn't contain the monomial $w_k^{d_k - 1} z_k^{d_k}$. Thus, $a = 0$. Similarly, $b = 0$. It follows that $P_k Q_k = w_k z_k$, which proves our claim. □

6 $\forall\mathbb{R}$-completeness of Deciding Convexity and Biquadratic Non-negativity

In this section, we prove Theorem 2.3 and Theorem 2.4. Our proof of Theorem 2.3 is inspired by [31, Lemma 2.8] where they show $\forall\mathbb{R}$-hardness of deciding if a degree 6 polynomial is non-negative, but our construction is significantly more intricate as we want the polynomial produced by the reduction to have the special structure of biquadratic forms. First, we prove a technical lemma in the spirit of [31, Lemma 2.7].

Lemma 6.1. *Let $y_0, \cdots, y_m, z_0, \cdots, z_m \in \mathbb{R}$ satisfy the inequality*

$$400 \left(\sum_{k=1}^{m} (y_k - y_{k-1} z_{k-1})^2 + \sum_{k=0}^{m} (y_k - z_k)^2 \right) < y_m^2 \,. \tag{4}$$

If $0 < y_0 < 1/2$ and $|z_m| < 1$, then $y_m^2 \leq y_0^{(2^{m+2}+2)/3}$.

Proof. By considering only the second term in the summation on the left-hand side of Inequality 4, we have

$$|y_k - z_k| < \frac{|y_m|}{20} \quad \text{for} \quad 0 \leq k \leq m,$$

and hence,

$$y_k - \frac{|y_m|}{20} < z_k < y_k + \frac{|y_m|}{20} \,. \tag{5}$$

In particular, $|y_m|/20 > |y_m - z_m| \geq |y_m| - |z_m|$ implying that $|y_m| < (20/19)|z_m| < 20/19$.

For $1 \leq k \leq m$, we have

$$\begin{aligned} & y_{k-1}z_{k-1} - \frac{|y_m|}{20} < y_k < y_{k-1}z_{k-1} + \frac{|y_m|}{20} \\ \implies & y_{k-1}^2 - y_{k-1}\frac{|y_m|}{20} - \frac{|y_m|}{20} < y_k < y_{k-1}^2 + y_{k-1}\frac{|y_m|}{20} + \frac{|y_m|}{20} \end{aligned} \tag{6}$$

where the implication follows from Inequality 5.

Suppose there exists $k \in [m]$ such that $|y_{k-1}| < |y_m|/2$. Then, Inequality 6 implies that

$$y_k < \frac{|y_m|^2}{4} + \frac{|y_m|^2}{40} + \frac{|y_m|}{20} = \frac{|y_m|}{2}\left(\frac{11}{20}|y_m| + \frac{1}{10}\right) < \frac{|y_m|}{2}$$

where the last inequality follows from $|y_m| < 20/19$. Similarly,

$$-y_k < \frac{|y_m|}{2}\left(\frac{1}{10} - \frac{9}{10}|y_m|\right) < \frac{|y_m|}{2}\,.$$

Thus, $|y_k| < |y_m|/2$ and by induction, we can conclude that $|y_m| < |y_m|/2$, which is a contradiction. Therefore, $|y_k| \geq |y_m|/2$ for all $0 \leq k \leq m$.

Now, we will show that $0 < y_i \leq y_0^{(2^{i+1}+1)/3}$ for $0 \leq i \leq m$, which would imply the desired bound on y_m^2. The proof is by induction on i.

The base case $i = 0$ is trivial.

Assume the statement is true for $i = k - 1$. Then, by the first inequality of Inequality 6,

$$y_{k-1}^2 < y_k + y_{k-1}\frac{|y_m|}{20} + \frac{|y_m|}{20} < y_k + \frac{|y_k|}{5}$$

where the last inequality follows from $y_{k-1} \leq y_0^{(2^k+1)/3} < 1$ and $|y_m| \leq 2|y_k|$. The above implies that $y_k > 0$. In a similar way, by the second inequality of Inequality 6,

$$\begin{aligned} & y_k < y_{k-1}^2 + y_{k-1}\frac{|y_m|}{20} + \frac{|y_m|}{20} < y_{k-1}^2 + \frac{|y_k|}{5} = y_{k-1}^2 + \frac{y_k}{5} \\ \implies & y_k < \frac{5}{4}y_{k-1}^2 \\ \implies & y_k < y_0^{-1/3} y_0^{2\cdot(2^k+1)/3} = y_0^{(2^{k+1}+1)/3} \end{aligned}$$

where the last inequality follows from $y_0^{-1/3} > 2^{1/3} > 5/4$. □

The following result will be used in the proof of Theorem 2.3 as it actually helps us lower bound the minimum value of a polynomial that only takes positive values on a certain semi-algebraic set.

Theorem 6.1 ([32], Corollary 3.4)**.** *If two semi-algebraic sets $S = \{x \in \mathbb{R}^n |\ \phi(x)\}$ and $T = \{x \in \mathbb{R}^n |\ \psi(x)\}$ have positive distance and each of the formulas ϕ and ψ have bitlength at most $L \geq 5n$, then the distance between the two sets is at least $2^{-2^{L+5}}$.*

Theorem 2.3. *Deciding non-negativity of a biquadratic form is $\forall\mathbb{R}$-complete.*

Proof of Theorem 2.3. It suffices to show that the following problem is $\exists\mathbb{R}$-complete: given a biquadratic form $Q \in \mathbb{R}[x_1, \cdots, x_n]$, check whether there exists $(x_1, \cdots, x_n) \in \mathbb{R}^n$ such that $Q(x_1, \cdots, x_n) < 0$. Let's call this problem BiquadraticNegativity.

BiquadraticNegativity is trivially in $\exists\mathbb{R}$. To show its $\exists\mathbb{R}$-hardness, we reduce the following $\exists\mathbb{R}$-complete problem to BiquadraticNegativity: given a set of quadratic polynomials $\{f_1(x), \cdots, f_t(x)\}$, check if they have a common root x inside the unit ball $B^n(0, 1)$ centered at 0. This problem was shown to be $\exists\mathbb{R}$-complete in [29, Lemma 3.9]. Further, their construction of these polynomials f_i satisfies an additional property: if these polynomials don't have a common root inside $B^n(0, 1)$, then they don't have a common root inside $B^n(0, \sqrt{3/2})$ either.

For each $i \in [t]$, let $f_i(x) = \sum_{1 \le j \le k \le n} a_{ijk} x_j x_k + \sum_{1 \le j \le n} b_{ij} x_j + c_i$ for constants a_{ijk}, b_{ij}, c_i. Introducing the new variables x_0 and $w_0, w_1, \cdots, w_n$, define for each $i \in [t]$ the bilinear form $g_i(x, w) := \sum_{1 \le j \le k \le n} a_{ijk} x_j w_k + \sum_{1 \le j \le n} b_{ij} x_j w_0 + c_i x_0 w_0$. Then, there is a natural bijective correspondence between the common roots of $\{f_1(x), \cdots, f_t(x)\}$ and the roots of $g(x, w) := \sum_{i=1}^{t} g_i^2(x, w) + \sum_{i=1}^{n} (x_i - w_i)^2 + (x_0 - 1)^2 + (w_0 - 1)^2$. A common root $(x_1, \cdots, x_n)$ of the polynomials f_i corresponds to a root $(x_0 = 1, x_1, \cdots, x_n, w_0 = 1, w_1 = x_1, \cdots, w_n = x_n)$ of g.

Therefore, if the polynomials f_i have a common root inside $B^n(0, 1)$, then g has a root $(x, w) \in \mathbb{R}^{2n+2}$ satisfying $\sum_{i=1}^{n} x_i^2 \le 1$.

On the other hand, if they don't have a common root inside $B^n(0, 1)$ and hence inside $B^n(0, \sqrt{3/2})$, then g doesn't have a root $(x, w) \in \mathbb{R}^{2n+2}$ satisfying $\sum_{i=1}^{n} (x_i^2 + w_i^2) \le 3$. Consequently, the compact sets

$$S = \{(x, w, g(x, w)) \mid \sum_{i=1}^{n} (x_i^2 + w_i^2) \le 3\} \text{ and } T = \{(x, w, 0) \mid \sum_{i=1}^{n} (x_i^2 + w_i^2) \le 3\}$$

will have positive distance and by Theorem 6.1, this distance is at least 2^{-2^m} for some $m = O(L + n)$, where L is the bit complexity of g. Thus, when the polynomials f_i don't have a common root inside $B^n(0, 1)$, we have

$$\min_{\substack{(x,w) \in \mathbb{R}^{2n+2}: \\ \sum_{i=1}^{n} (x_i^2 + w_i^2) \le 3}} g(x, w) \ge 2^{-2^m} . \tag{7}$$

Now, consider the inequality

$$\begin{aligned} & g(x, w) + \left(\sum_{i=1}^{n+1} x_i w_i - 1 \right)^2 + (x_{n+1} - w_{n+1})^2 \\ & + 400 \left((y_1 - 1/16)^2 + \sum_{k=2}^{m} (y_k - y_{k-1} z_{k-1})^2 + \sum_{k=1}^{m} (y_k - z_k)^2 \right) \qquad < 2 y_m z_m - z_m^2 - y_m^2 z_m^2 . \end{aligned} \tag{8}$$

If the polynomials f_i have a common root in $B^n(0, 1)$ and hence g has a root $(x_0, \cdots, x_n, w_0, \cdots, w_n)$ satisfying $\sum_{i=1}^{n} x_i^2 \le 1$, then this $(x_0, \cdots, x_n, w_0, \cdots, w_n)$ along with $x_{n+1} = w_{n+1} = \sqrt{1 - \sum_{i=1}^{n} x_i^2}$ and $y_k = z_k = 4^{-2^k}$ for $1 \le k \le m$ will satisfy Inequality 8, as the LHS will be zero and the RHS will be positive.

Conversely, assume that Inequality 8 holds for some $(x, w, y, z) \in \mathbb{R}^{n+2} \times \mathbb{R}^{n+2} \times \mathbb{R}^{m+1} \times \mathbb{R}^{m+1}$. Since the LHS is a sum of squares, we have

$$\begin{aligned} & 2 y_m z_m - z_m^2 - y_m^2 z_m^2 > 0 \\ \implies & z_m \left(z_m - \frac{2 y_m}{1 + y_m^2} \right) < 0 \\ \implies & |z_m| < 1 \qquad \text{as } 2 y_m / (1 + y_m^2) \in [-1, 1] \text{ for all } y_m \in \mathbb{R}. \end{aligned}$$

Since RHS $= y_m^2 - (y_m - z_m)^2 - y_m^2 z_m^2 \le y_m^2$, we can now use Lemma 6.1 with $y_0 = z_0 = 1/4$ to conclude that $y_m^2 \le (1/4)^{(2^{m+2}+2)/3} = 2^{-4(2^{m+1}+1)/3} < 2^{-2^m}$.

Therefore, we have $g(x, w) + \left(\sum_{i=1}^{n+1} x_i w_i - 1 \right)^2 + (x_{n+1} - w_{n+1})^2 <$ RHS $\le y_m^2 < 2^{-2^m}$. We can bound each of the three summands here by 2^{-2^m}, which yields the following:

$$\begin{aligned} \sum_{i=1}^{n+1} (x_i^2 + w_i^2) &= \sum_{i=1}^{n} (x_i - w_i)^2 + (x_{n+1} - w_{n+1})^2 + 2 \sum_{i=1}^{n+1} x_i w_i \\ &\le g(x, w) + (x_{n+1} - w_{n+1})^2 + 2 \left(\sum_{i=1}^{n+1} x_i w_i - 1 \right) + 2 \\ &< 2^{-2^m} + 2^{-2^m} + 2 \cdot 2^{-2^{m-1}} + 2 < 3 . \end{aligned}$$

Thus, there exists (x, w) such that $g(x, w) < 2^{-2^m}$ and $\sum_{i=1}^{n}(x_i^2 + w_i^2) < 3$. Using Inequality 7, we conclude that the polynomials f_i have a common root in $B^n(0, 1)$.

We have shown that the polynomials f_i have a common root in $B^n(0, 1)$ if and only if $\exists(x, w, y, z) \in \mathbb{R}^{n+2} \times \mathbb{R}^{n+2} \times \mathbb{R}^{m+1} \times \mathbb{R}^{m+1}$ such that

$$h(x, w, y, z) := g(x, w) + \left(\sum_{i=1}^{n+1} x_i w_i - 1\right)^2 + (x_{n+1} - w_{n+1})^2$$
$$+ 400\left((y_1 - 1/16)^2 + \sum_{k=2}^{m}(y_k - y_{k-1}z_{k-1})^2 + \sum_{k=1}^{m}(y_k - z_k)^2\right) - (2y_m z_m - z_m^2 - y_m^2 z_m^2) < 0\,.$$

For the final step, it remains to turn h into a biquadratic form Q. Define

$$Q(x, w, y, z, \alpha, \beta) = \alpha^2\beta^2 \cdot h(x/\alpha, w/\beta, y/\alpha, z/\beta)$$

be the homogenization of the polynomial $h(x, w, y, z)$ using two new variables α and β. Note that Q is indeed a biquadratic form with respect to the partition of the variable set into the two groups (x, y, α) and (w, z, β).

If $\exists(x, w, y, z)$ such that $h(x, w, y, z) < 0$, then we have $Q(x, w, y, z, 1, 1) < 0$.

Conversely, if $\exists(x, w, y, z, \alpha, \beta)$ such that $Q(x, w, y, z, \alpha, \beta) < 0$, then α and β are nonzero as $Q(x, w, y, z, \alpha, \beta)$ is a sum of squares when $\alpha = 0$ or $\beta = 0$. Therefore, we have $h(x/\alpha, w/\beta, y/\alpha, z/\beta) = \frac{1}{\alpha^2\beta^2}Q(x, w, y, z, \alpha, \beta) < 0$. □

Remark 6.2. *The result of Theorem 2.3 easily generalizes to show that deciding non-negativity of triquadratic forms is also $\forall\mathbb{R}$-complete. Indeed, one can multiply the biquadratic form $Q(x, w, y, z, \alpha, \beta)$ constructed above by the square of a variable from a third group to obtain a triquadratic form T. More generally, the same construction extends to show that testing non-negativity of k-quadratic forms is $\forall\mathbb{R}$-complete for any $k \geq 2$.*

Now, we prove Theorem 2.4. We use the following reduction from BiquadraticNegativity to deciding convexity of quartic polynomials shown in [1].

Theorem 6.2 ([1], Theorem 2.3)**.** *Let $b(X, Y)$ be a biquadratic form with respect to the partition $X \sqcup Y$ of the variable set and let $|X| = |Y| = n$. Define the $n \times n$ polynomial matrix $C(X, Y)$ by setting*

$$[C(X, Y)]_{ij} = \frac{\partial b(X, Y)}{\partial X_i Y_j}$$

and let γ be the largest coefficient, in absolute value, of any monomial present in some entry of the matrix $C(X, Y)$. Let f be the form given by

$$f(X, Y) := b(X, Y) + \frac{n^2\gamma}{2}\left(\sum_{i=1}^{n}(X_i^4 + Y_i^4) + \sum_{1 \leq i < j \leq n}(X_i^2 X_j^2 + Y_i^2 Y_j^2)\right)\,.$$

Then, $b(X, Y)$ is non-negative if and only if $f(X, Y)$ is convex.

Using this theorem along with Theorem 2.3, we easily get that deciding if a quartic polynomial is convex is $\forall\mathbb{R}$-hard. It only remains to show that this problem is also in $\forall\mathbb{R}$.

A polynomial $f \in \mathbb{R}[x_1, \cdots, x_n]$ is convex
$\iff \forall x \in \mathbb{R}^n\ H_f(x) \succeq 0$
$\iff \forall x, z \in \mathbb{R}^n\ z^T H_f(x) z \geq 0.$

Since we can compute the polynomial entries of $H_f(x)$ in polynomial time, we can decide in $\forall\mathbb{R}$ if f is convex.

7 $\forall\mathbb{R}$-completeness of Hyperbolicity and Real Stability

First, we prove Theorem 2.5 which states that deciding if a homogeneous quartic polynomial p is hyperbolic with respect to a vector e is $\forall\mathbb{R}$-complete.

Deciding hyperbolicity is in $\forall\mathbb{R}$: It is enough to show that the following problem is in $\exists\mathbb{R}$: given a quartic homogeneous polynomial $p(x_1, \cdots, x_n)$ and a vector $e \in \mathbb{R}^n$, decide if p is **not** hyperbolic with respect to e.

From the definition of hyperbolicity, we have the following.
p is not hyperbolic with respect to e

$$\iff \Big(p(e) \le 0\Big) \vee \Big(\exists x \in \mathbb{R}^n\ \exists t \in \mathbb{C} \setminus \mathbb{R}\ \ p(x+te) = 0\Big)$$

$$\iff \Big(p(e) \le 0\Big) \vee \Big(\exists x \in \mathbb{R}^n\ \exists (y,z) \in \mathbb{R}^2\ \ z \neq 0\ \wedge\ p(x+ye+ize) = 0\Big)$$

$$\iff \exists x \in \mathbb{R}^n\ \exists (y,z) \in \mathbb{R}^2\ \ \Big(p(e) \le 0\Big) \vee \Big(z \neq 0\ \wedge\ \Re\big(p(x+ye+ize)\big) = 0\ \wedge\ \Im\big(p(x+ye+ize)\big) = 0\Big)$$

where $\Re(f)$ and $\Im(f)$ denote the real and imaginary parts of the polynomial f respectively. The real and imaginary parts of $p(x+ye+ize)$ can be computable in polynomial time since p is of degree only 4.

Thus, we can decide in $\exists\mathbb{R}$ if p is not hyperbolic with respect to e.

Deciding hyperbolicity is $\forall\mathbb{R}$-hard: By Theorem 2.3, it suffices to reduce the problem of deciding non-negativity of a biquadratic polynomial to the problem of deciding hyperbolicity of a homogeneous quartic polynomial with respect to a vector.

Given a biquadratic form $Q(x_1, \cdots, x_n)$, we will construct a homogeneous polynomial p of degree 4 and a vector e such that p is hyperbolic with respect to e iff $\forall x_1, \cdots, x_n \in \mathbb{R}$, $Q(x_1, \cdots, x_n) \ge 0$.

Let $\|x\| = \sqrt{x_1^2 + \cdots + x_n^2}$. Choose β large enough so that

$$\forall x_1, \cdots, x_n \in \mathbb{R},\ Q(x_1, \cdots, x_n) \le \frac{\beta^2}{4}\|x\|^4 .$$

It's easy to see that $\beta = 2n^2C$ satisfies this constraint where C is the largest absolute value of a coefficient of Q.

Now, we define our polynomial

$$p(x_0, \cdots, x_n) = x_0^4 - \beta x_0^2\|x\|^2 + Q(x_1, \cdots, x_n)$$

and vector $e = (1, 0, \cdots, 0) \in \mathbb{R}^{n+1}$. Note that $p(e) > 0$, which is one of the conditions for hyperbolicity.

Let W be the hyperplane in $\mathbb{R}^{n+1}$ given by the equation $x_0 = 0$. Note that $e \notin W$. Hence, by Theorem 3.2, p is hyperbolic with respect to e if and only if $B_{p,e}(x) \succeq 0$ for all $x \in W$. We compute this $B_{p,e}(x)$ for $x \in W$:

When $x \in W$, $p(x+te) = p(t, x_1, \cdots, x_n) = t^4 - \beta t^2\|x\|^2 + Q(x_1, \cdots, x_n)$

and $D_e p(x+te) = \frac{d}{du}p(x+te+ue)|_{u,x_0=0} = 4t^3 - 2\beta t\|x\|^2$. As a result, we have

$$\begin{aligned}&\frac{p(x+te)D_e p(x+se) - p(x+se)D_e p(x+te)}{t-s}\\ &= 2\beta\|x\|^2 Q + (2\beta^2\|x\|^4 - 4Q)st - 4Q(s^2+t^2) + 2\beta\|x\|^2(s^2t^2 - s^3t - st^3) + 4s^3t^3 .\end{aligned}$$

Hence,

$$B_{p,e}(x) = \begin{bmatrix} 2\beta\|x\|^2 Q & & -4Q & \\ & 2\beta^2\|x\|^4 - 4Q & & -2\beta\|x\|^2 \\ -4Q & & 2\beta\|x\|^2 & \\ & -2\beta\|x\|^2 & & 4 \end{bmatrix} .$$

Computing the Schur complement (with respect to the bottom-right 2×2 block) of $B_{p,e}(x)$ and dividing by 4, we obtain

$$\begin{aligned}B_{p,e}(x) \succeq 0 &\iff \begin{bmatrix} \frac{2Q}{\beta\|x\|^2}(\frac{\beta^2}{4}\|x\|^4 - Q) & \\ & \frac{\beta^2}{4}\|x\|^4 - Q \end{bmatrix} \succeq 0\\ &\iff Q \ge 0 \text{ and } \frac{\beta^2}{4}\|x\|^4 - Q \ge 0 .\end{aligned}$$

Here we have used the fact that $B_{p,e}(x) \succeq 0$ if and only if its Schur complement is $\succeq 0$. Thus, p is hyperbolic with respect to e

$\iff \forall x_1, \cdots, x_n \in \mathbb{R}, 0 \le Q(x_1, \cdots, x_n) \le \frac{\beta^2}{4}\|x\|^4$
$\iff \forall x_1, \cdots, x_n \in \mathbb{R}, Q(x_1, \cdots, x_n) \ge 0$
as the upper bound is always satisfied by our choice of β.

Now, we prove Theorem 2.6 which states that deciding if a homogeneous quartic polynomial p is real stable is $\forall\mathbb{R}$-complete.

Deciding real stability is in $\forall\mathbb{R}$:

p is real stable
$\iff \forall z \in \mathbb{C}^n \ (\bigwedge_{k=1}^n \Im(z_k) > 0) \implies p(z) \neq 0$
$\iff \forall x, y \in \mathbb{R}^n \ (\bigwedge_{k=1}^n y_k > 0) \implies (\Re(p(x+iy)) \neq 0 \vee \Im(p(x+iy)) \neq 0)$
where the real and imaginary parts of $p(x+iy)$ are computable in polynomial time since p is of degree only 4.

Thus, we can decide in $\forall\mathbb{R}$ if p is real stable.

Deciding real stability is $\forall\mathbb{R}$-hard: Saunderson [28] showed coNP-hardness of deciding if a cubic homogeneous polynomial p is hyperbolic with respect to a vector e. Chin [8] demonstrated a way to leverage this result to prove coNP-hardness of deciding real stability of a polynomial. For the polynomial p and the vector e in the construction of [28], they showed how to build a cubic homogeneous polynomial $\tilde{p}$ such that p is hyperbolic with respect to e if and only if $\tilde{p}$ is real stable.

Using the exact same method as in [8], we can leverage the $\forall\mathbb{R}$-hardness result in Theorem 2.5 to obtain $\forall\mathbb{R}$-hardness of deciding if a quartic homogeneous polynomial is real stable. Basically, we can show the following:

Theorem 7.1. *Let Q be a quartic homogeneous polynomial and C be the largest absolute value of a coefficient in Q. Further, define the vector $e_0 = (1, 0, \cdots, 0) \in \mathbb{R}^{n+1}$ and the polynomial $p(x_0, x) = x_0^4 - \beta x_0^2 \|x\|^2 + Q(x_1, \cdots, x_n)$ for $\beta = 2n^2C$. Then, we can construct a quartic homogeneous polynomial $\tilde{p}$ such that $\tilde{p}$ is real stable if and only if p is hyperbolic with respect to e_0. Moreover, $\tilde{p}$ can be constructed in polynomial time in the total bit complexity of p.*

We can prove this theorem by replicating [8, Section 3] with minute changes and the proof is given in Section A. The polynomial p and the vector e_0 in this theorem describe our construction in the $\forall\mathbb{R}$-hardness result of Theorem 2.5. Thus, we can conclude that it is $\forall\mathbb{R}$-hard to decide if a quartic homogeneous polynomial is stable.

References

[1] Amir Ahmadi, Alex Olshevsky, Pablo Parrilo, and John Tsitsiklis. NP-hardness of deciding convexity of quartic polynomials and related problems. *Mathematical Programming*, 137, 12 2010.

[2] Lenore Blum, Felipe Cucker, Michael Shub, and Stephen Smale. *Complexity and Real Computation*. Springer-Verlag, New York, 1998.

[3] Lenore Blum, Mike Shub, and Steve Smale. On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines. *Bulletin of the American Mathematical Society*, 21(1):1–46, 1989.

[4] Peter Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and computation in mathematics*. Springer, 2000.

[5] Peter Bürgisser and Christian Ikenmeyer. Geometric complexity theory and tensor rank. In *Proceedings of the 43rd ACM Symposium on Theory of Computing (STOC)*, STOC '11, page 509–518, New York, NY, USA, 2011. Association for Computing Machinery.

[6] John Canny. Some algebraic and geometric computations in pspace. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, page 460–467, New York, NY, USA, 1988. Association for Computing Machinery.

[7] Suryajith Chillara, Coral Grichener, and Amir Shpilka. On Hardness of Testing Equivalence to Sparse Polynomials Under Shifts. In Petra Berenbrink, Patricia Bouyer, Anuj Dawar, and Mamadou Moustapha Kanté, editors, *40th International Symposium on Theoretical Aspects of Computer Science (STACS 2023)*, volume 254 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 22:1–22:20, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[8] Tracy Chin. Real stability and log concavity are conp-hard, 2024.

[9] Zeev Dvir, Rafael Mendes de Oliveira, and Amir Shpilka. Testing equivalence of polynomials under shifts. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming*, pages 417–428, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg.

[10] Jeff Erickson, Ivor van der Hoog, and Tillmann Miltzow. Smoothing the gap between NP and ER. *SIAM J. Comput.*, 53(6):S20–102, 2024.

[11] D. Grigoriev. Testing shift-equivalence of polynomials by deterministic, probabilistic and quantum machines. *Theoretical Computer Science*, 180(1):217–228, 1997.

[12] Leonid Gurvits. Classical deterministic complexity of edmonds' problem and quantum entanglement. *Conference Proceedings of the Annual ACM Symposium on Theory of Computing*, 10, 04 2003.

[13] LARS GÅRDING. An inequality for hyperbolic polynomials. *Journal of Mathematics and Mechanics*, 8(6):957–965, 1959.

[14] Anna R. Karlin, Nathan Klein, and Shayan Oveis Gharan. A (slightly) improved approximation algorithm for metric TSP. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 32–45. ACM, 2021.

[15] Neeraj Kayal. Affine projections of polynomials: extended abstract. In *Proceedings of the Forty-Fourth Annual ACM Symposium on Theory of Computing*, STOC '12, page 643–662, New York, NY, USA, 2012. Association for Computing Machinery.

[16] J. K. Knowles and Eli Sternberg. On the ellipticity of the equations of nonlinear elastostatics for a special material. *Journal of Elasticity*, 5(3):341–361, Nov 1975.

[17] Pascal Koiran. Hilbert's Nullstellensatz is in the polynomial hierarchy. *Journal of Complexity*, 12(4):273–286, 1996.

[18] Y. N. Lakshman and B. David Saunders. On computing sparse shifts for univariate polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, ISSAC '94, page 108–113, New York, NY, USA, 1994. Association for Computing Machinery.

[19] Chen Ling, Jiawang Nie, Liqun Qi, and Yinyu Ye. Biquadratic optimization over unit spheres and semidefinite programming relaxations. *SIAM Journal on Optimization*, 20(3):1286–1310, 2010.

[20] Adam Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families i: Bipartite ramanujan graphs of all degrees. In *Proceedings of the 2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, FOCS '13, page 529–537, USA, 2013. IEEE Computer Society.

[21] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families. II: Mixed characteristic polynomials and the Kadison-Singer problem. *Ann. Math. (2)*, 182(1):327–350, 2015.

[22] Adam W. Marcus, Daniel A. Spielman, and Nikhil Srivastava. Interlacing families iv: Bipartite ramanujan graphs of all sizes. *SIAM Journal on Computing*, 47(6):2488–2509, 2018.

[23] Yu. V. Matiyasevich. The Diophantineness of enumerable sets. *Dokl. Akad. Nauk SSSR*, 191:279–282, 1970.

[24] Panos Pardalos and Stephen Vavasis. Open questions in complexity theory for numerical optimization. *Math. Program.*, 57:337–339, 05 1992.

[25] Prasad Raghavendra, Nick Ryder, and Nikhil Srivastava. Real Stability Testing. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 5:1–5:15, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[26] James Renegar. On the computational complexity and geometry of the first-order theory of the reals, part I: introduction. preliminaries. the geometry of semi-algebraic sets. the decision problem for the existential theory of the reals. *J. Symb. Comput.*, 13(3):255–300, 1992.

[27] Phoebus Rosakis. Ellipticity and deformations with discontinuous gradients in finite elastostatics. *Archive for Rational Mechanics and Analysis*, 109:1–37, 03 1990.

[28] James Saunderson. Certifying polynomial nonnegativity via hyperbolic optimization. *SIAM Journal on Applied Algebra and Geometry*, 3(4):661–690, 2019.

[29] Marcus Schaefer. Realizability of graphs and linkages. *Thirty Essays on Geometric Graph Theory*, 07 2013.

[30] Marcus Schaefer, Jean Cardinal, and Tillmann Miltzow. The existential theory of the reals as a complexity class: A compendium. *CoRR*, abs/2407.18006, 2024.

[31] Marcus Schaefer and Daniel Stefankovic. Beyond the existential theory of the reals. *Theory Comput. Syst.*, 68(2):195–226, 2024.

[32] Marcus Schaefer and Daniel Stefankovič. Fixed points, nash equilibria, and the existential theory of the reals. *Theory of Computing Systems*, 60, 02 2017.

[33] H. C. Simpson and S. J. Spector. On copositive matrices and strong ellipticity for isotropic elastic materials. *Archive for Rational Mechanics and Analysis*, 84(1):55–68, Mar 1983.

[34] V. Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13:354–356, 1969.

[35] C.F. Sturm. *Mémoire sur la résolution des équations numériques*. .s.n., 1835.

[36] Alfred Tarski. A decision method for elementary algebra and geometry. Technical report, The Rand Corporation, Santa Monica, CA, 1948.

[37] Y. Wang and M. Aron. A reformulation of the strong ellipticity conditions for unconstrained hyperelastic media. *Journal of Elasticity*, 44(1):89–96, Jul 1996.

A Deciding Real Stability Reduces to Deciding Hyperbolicity

We show how to prove Theorem 7.1 by using the strategy of [8, Section 3]. As described in Theorem 7.1, let Q be a quartic homogeneous polynomial and C be the largest absolute value of a coefficient in Q. Define the polynomial $p(x_0, x) = x_0^4 - \beta x_0^2 \|x\|^2 + Q(x_1, \cdots, x_n)$ for $\beta = 2n^2 C$. For $i \in [n+1]$, let e_i denote the vector in $\mathbb{R}^{n+1}$ with one in the i-th coordinate and zeros in the rest.

Lemma A.1. *Let the polynomials p, Q and the constants C, β be as defined above. If $0 < \varepsilon < \min\{\frac{1}{\sqrt{2\beta}}, \frac{1}{2n}\}$, then $p(\|x\|, \varepsilon x) > 0$ for all $x \in \mathbb{R}^n \setminus \{0\}$.*

Proof. Fix $x \in \mathbb{R}^n \setminus \{0\}$. We have

$$p(\|x\|, \varepsilon x) = \|x\|^4 - \beta\varepsilon^2 \|x\|^4 + Q(\varepsilon x) = \|x\|^4 (1 - \beta\varepsilon^2 + \varepsilon^4 Q(x/\|x\|)) \, .$$

Since Q is a quartic homogeneous polynomial, it has at most n^4 monomials and every monomial of Q has absolute value ≤ 1 at the point $x/\|x\|$. Therefore, $Q(x/\|x\|) \geq -Cn^4$.

Thus, when $0 < \varepsilon < \min\{1/\sqrt{2\beta}, 1/(2n)\}$,

$$\begin{aligned} p(\|x\|, \varepsilon x) &\geq \|x\|^4 (1 - \beta\varepsilon^2 - \varepsilon^4 C n^4) \\ &\geq \|x\|^4 \left(1 - \beta \cdot \frac{1}{2\beta} - Cn^4 \cdot \min\left\{\frac{1}{4.4n^4C^2}, \frac{1}{16n^4}\right\}\right) > 0 \, . \end{aligned}$$

□

Lemma A.2. *Let p be as defined above. For $0 < \varepsilon < \min\{\frac{1}{\sqrt{2\beta}}, \frac{1}{2n}\}$, define $K_\varepsilon = \text{cone}\{e_0 \pm \varepsilon e_i \mid i \in [n]\}$. Then, p is hyperbolic with respect to e_0 if and only if it is hyperbolic with respect to K_ε.*

Proof. One direction is trivial as $e_0 \in K_\varepsilon$.

For the other direction, assume that p is hyperbolic with respect to e_0. Then, p is hyperbolic with respect to all the vectors in the hyperbolicity cone $\Lambda_+(p, e_0)$. We want to show that the cone K_ε is contained in the cone $\Lambda_+(p, e_0)$. Since $p(e_0) > 0$ and by Theorem 3.1, $\Lambda_+(p, e_0)$ is exactly the connected component of $\{(x_0, x) \in \mathbb{R}^{n+1} : p(x_0, x) \neq 0\}$ containing e_0, it suffices to show that $p(x_0, x) > 0$ on K_ε.

For all $(x_0, x) \in K_\varepsilon$, we have $\|x\| = \delta x_0$ for some $\delta \leq \varepsilon$. Hence, by Lemma A.1, $p(x_0, x) > 0$ for all $(x_0, x) \in K_\varepsilon$. □

Theorem A.1. *Let the polynomial p and the constant β be as defined above. For $0 < \varepsilon < \min\{\frac{1}{\sqrt{2\beta}}, \frac{1}{2n}\}$, let M be the $(n+1) \times 2n$ matrix whose columns are $e_0 \pm \varepsilon e_i$ for $i \in [n]$. Define*

$$\tilde{p}(x_1, \cdots, x_{2n}) = p(Mx) \, .$$

Then, p is hyperbolic with respect to e_0 if and only if $\tilde{p}$ is real stable.

Proof. Let $\tilde{p}$ be real stable. We want to show that for every $z \in \mathbb{R}^{n+1}$, $p(te_0 + z)$ has only real roots. Since M has full row rank, there exists $x \in \mathbb{R}^{2n}$ such that $z = Mx$. Then,

$$p(te_0 + z) = p\left(\frac{t}{2n} M\mathbb{1} + Mx\right) = \tilde{p}\left(\frac{t}{2n}\mathbb{1} + x\right)$$

where $\mathbb{1}$ is the vector of all ones. Since $\frac{1}{2n}\mathbb{1} \in \mathbb{R}^{2n}_{>0}$ and $\tilde{p}$ is real stable, $p(te_0 + z)$ has only real roots and hence p is hyperbolic with respect to e_0.

Conversely, let p be hyperbolic with respect to e_0. By Lemma A.2, p is hyperbolic with respect to K_ε. Then, for any $v \in \mathbb{R}^{2n}_{>0}$ and $x \in \mathbb{R}^{2n}$,

$$\tilde{p}(tv + x) = p(t(Mv) + Mx)$$

which has only real roots since $Mv \in K_\varepsilon$. Therefore, $\tilde{p}$ is real stable. □

B Machine Models for the Existential Theory of the Reals

[10] expands the concept of word RAMs to encompass real computations. The real RAMs introduced in [10] facilitate both integer and real computations simultaneously, unlike the BSS model of real computation [3]. The real RAMs in [10] are more convenient to program, as they include features such as indirect memory access to real registers, which are crucial for implementing algorithms over real numbers in an efficient manner.

We give a brief outline how the real RAMs work: A real RAM gets two input vectors: one vector of integers and one of real numbers. It has two types of registers: word registers and real registers. Word registers store integers with w bits, where w is the word size. The total number of registers of each type is 2^w, since we can address at most that many registers with w bits. Arithmetic operations and bitwise Boolean operations are performed on word registers, which interpret words as integers between 0 and $2^w - 1$. Real registers, however, only support arithmetic operations. Word registers can indirectly address both types of registers, and control flow is managed through conditional jumps based on comparisons between word registers or between a real register and the constant 0. For more details, refer to the original paper [10].

Real RAMs characterize the existential theory of the reals. The main result of [10] is that a problem belongs to $\exists\mathbb{R}$ if and only if there exists a polynomial-time real verification algorithm for it, given as a real RAM. Thus, real RAMs allows us to prove that a problem is in $\exists\mathbb{R}$ by giving an algorithm. Besides the input x, which is a sequence of words, the real verification algorithm receives a certificate c comprising a sequence of real numbers and another sequence of words. The input x is contained in the language recognized by the verifying algorithm if there exists a certificate c that makes the real verification algorithm accept. Conversely, x is not in the language if the real verification algorithm rejects all certificates. One of the main results in [10] is the following:

Theorem B.1. *$L \in \exists\mathbb{R}$ if and only if there is a polynomial time real verification algorithm for L.*