An Introduction to Algorithmic Fewnomial Theory Over $\mathbb{R}$

J. Maurice Rojas

1.1. **Which equations are simplest?** In grade school we learn how to solve simple equations, such as linear systems or quadratic equations, with an eye toward calculus. Usually the solutions are rational numbers, to avoid floating point arithmetic, and one ultimately works over the real numbers. However there is another family of equations, arguably as simple, that can naturally introduce us to convex geometry and numerical algebraic geometry: those defined by n-variate $(n+k)$-nomials, with $k \leq 2$.

We will give more precise definitions shortly, but it is fair to ask now: *What does it means to solve a polynomial equation in n variables over the real numbers, particularly when $n \geq 2$?* Our answer is to consider an arbitrary system of equations solved over $\mathbb{R}$ when the following questions have been answered:

S1. *Are there any real roots?*

S2. *How many real roots are there?*

S3. *If the number of real roots is infinite then how many connected components does the set of real roots have?*

S4. *Can one somehow classify the shape of the real zero set, e.g., compute its homology groups or isotopy type in some explicit way?*

S5. *Can one efficiently describe an approximation of the real zero set?*[1]

We will approach these questions assuming just some background in linear algebra and point set topology.

1.2. Why Should We Care? Numerous problems in physics, optimization, engineering, robotics, computational biology, learning theory, chemistry, and complexity theory — to name just a few areas — reduce to approximating the real solutions of a system of polynomial equations.

Chemical reaction networks are one concrete example where the underlying monomial term structure plays an important role. Recall that their theory attempts to predict the concentrations of new chemical compounds as a chemical reaction evolves (see, e.g., [16]): Certain reactions involving n chemical compounds (called *species*), with the ith species having concentration x_i, can be modelled via a dynamical system of the following form:

$$\frac{dx}{d\tau} = \Gamma\,[x^{a_1}, \ldots, x^{a_t}]^\top,$$

where $(\cdot)^\top$ denotes matrix transpose, $\tau \in \mathbb{R}$ denotes time, $x := (x_1, \ldots, x_n)^\top$, $\Gamma \in \mathbb{R}^{n \times t}$ is a matrix obtained from the underlying chemical reactions (including rate constants), $a_1, \ldots, a_t \in \mathbb{Z}^n$, $a_i := (a_{1,i}, \ldots, a_{n,i})$, and $x^{a_i} := x_1^{a_{1,i}} \cdots x_n^{a_{n,i}}$. The set of exponent vectors $\mathcal{A} := \{a_1, \ldots, a_t\}$ depends on a graph defined by the underlying collection of chemical reactions and can, in theory, be any finite subset of $\mathbb{Z}^n$. Let $\mathbb{R}_+$ denote the positive real numbers.

Chemists are interested in the *steady states* of such dynamical systems, i.e., the values of $x \in (\mathbb{R}_+ \cup \{0\})^n$ making the right-hand side the zero vector. Finding steady states is thus equivalent to solving a system of n polynomial equations in n variables, each involving at most t monomial terms, over the nonnegative orthant. Phosphorylation reactions [19] involving certain proteins can in fact lead to n and t respectively as large as 453 and 600, or even larger.

Biochemists are particularly interested in modelling switch-like or clock-like behavior in living organisms, meaning that the number of states is of great importance. In other words, the number of nonnegative solutions of polynomial systems with fixed monomial term structure is of great importance to biochemists.

This example partially motivates why monomial term structure plays a large role in this brief note. But an even bigger motivation comes from recent advances in complexity theory: Sufficiently sharp estimates on the number of real roots of certain structured univariate polynomials, as a function of their input complexity, are now known to imply new separations of complexity classes near the **P** vs. **NP** Problem and derandomization [21, 15]. In particular, univariate polynomials defined by *arithmetic circuits* (as in [21, 15]) can be naturally encoded, and thus

J. Maurice Rojas is executive associate head of teaching operations and full professor in the mathematics department at Texas A&M University. His email address is rojas@tamu.edu
This work was partially supported by NSF grant CCF-1900881.
Communicated by Notices *Associate Editor Han-Bom Moon.*

DOI: https://doi.org/10.1090/noti3185

[1] For instance, if there are only $N < \infty$ roots in $\mathbb{R}^n$, and one is given an $\varepsilon > 0$, can one quickly produce a set S of N points in $\mathbb{Q}^n$ with each root within $< \varepsilon$ of some point of S?

studied, via systems of n-variate $(n+k)$-nomials.

1.3. *n-variate $(n+1)$-nomials.* While most of us know how to plot real quadratic curves, it is less well-known that certain high degree curves can also be plotted easily. More generally, let $\mathbb{R}\left[x_1^{\pm 1},\ldots,x_n^{\pm 1}\right]$ denote the set of n-variate Laurent polynomials with real coefficients, $f \in \mathbb{R}\left[x_1^{\pm 1},\ldots,x_n^{\pm 1}\right]$, and let $Z_+(f)$ denote the zero set of f in the positive orthant $\mathbb{R}^n_+$. Also let $x := (x_1,\ldots,x_n)$ and $\operatorname{Log} x := (\operatorname{Log} x_1,\ldots,\operatorname{Log} x_n)$ (base e). It is easily checked that $Z_+(f)$ and $\operatorname{Log} Z_+(f)$ are *ambiently isotopic (as subsets of $\mathbb{R}^n$)*, i.e., there is a continuous map $H : \mathbb{R}^n \times [0,1] \longrightarrow \mathbb{R}^n$ such that $H(\cdot,0)$ is the identity map on $\mathbb{R}^n$ (so $H(Z_+(f),0) = Z_+(f)$ in particular), $H(\cdot,t)$ is a homeomorphism for all $t \in [0,1]$, and $H(Z_+(f),1) = \operatorname{Log} Z_+(f)$. (This is a stronger condition than $Z_+(f)$ and $\operatorname{Log} Z_+(f)$ being homeomorphic.[2]) But taking coordinate-wise logarithms reveals more: a natural connection between zero sets and piece-wise linear geometry.

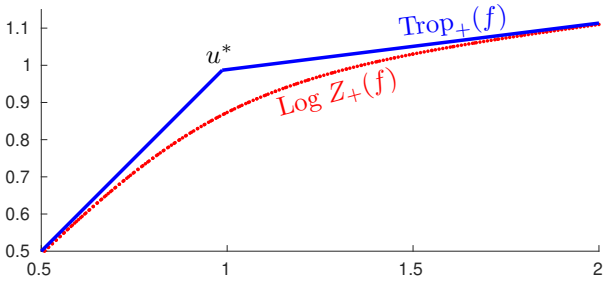
Example 1.1. *For any $(a,b)\in\mathbb{Z}^2\setminus\{(0,0)\}$ and $c\in\mathbb{R}\setminus\{0\}$, we have that $\operatorname{Log} Z_+(x_1^a x_2^b - c)$ is either empty or a line perpendicular to (a,b), according as $c\leq 0$ or $c>0$.* $\diamond$

Zero sets of binomials are particularly simple but can form the building blocks for much more complicated zero sets in $\mathbb{R}^n_+$. Let $\mathbb{R}^* := \mathbb{R}\setminus\{0\}$.

Definition 1.2. *For any $v=(v_1,\ldots,v_n)$ and $w=(w_1,\ldots,w_n)$ in $\mathbb{R}^n$ let $v\cdot w := v_1w_1+\cdots+v_nw_n$. Suppose $c_i \in \mathbb{R}^*$ for all $i\in\{1,\ldots,t\}$, and $\mathcal{A} := \{a_1,\ldots,a_t\}\subset\mathbb{Z}^n$ has cardinality t. We call $f(x)=\sum_{i=1}^t c_i x^{a_i}$ a* (real) *n-variate t-nomial, $\mathcal{A}$ the* support *of f, and define the* positive tropical variety *of $f(x)=\sum_{i=1}^t c_i x^{a_i}$, denoted $\operatorname{Trop}_+(f)$, to be the set of $u\in\mathbb{R}^n$ such that $\max_i |c_i e^{a_i\cdot u}|$ is attained at some pair of indices j and j' with $c_jc_{j'}<0$.* $\diamond$

Tropical geometry connects piece-wise linear geometry and the metric structure of non-Archimedean fields (like the p-adic complex numbers and Puiseux series fields) to give us new tools to approach subtle algebraic geometry problems (see, e.g., [4]). Here, we present an analogue geared toward approximating polynomial zero sets in $\mathbb{R}^n_+$.

Example 1.3. *If $(n,t)=(2,3)$, the points $a_1,a_2,a_3\in\mathbb{Z}^2$ are non-collinear, $c_1,c_3>0$, and $c_2<0$, then it is easily checked that $\operatorname{Trop}_+(f)$ is the union of two rays intersecting only at the unique point $u^*\in\mathbb{R}^2$ satisfying $\operatorname{Log}|c_1|+a_1\cdot u^* = \operatorname{Log}|c_2|+a_2\cdot u^* = \operatorname{Log}|c_3|+a_3\cdot u^*$. The two rays are respectively perpendicular to the vectors a_2-a_1 and a_3-a_2. For instance, when $f(x) = x_2^4 - x_1^4 + x_1^3x_2^8/1000$, we have $u^* = \frac{\log 1000}{7}(1,1) \approx 0.9868(1,1)$, and $\operatorname{Trop}_+(f)$ and $\operatorname{Log} Z_+(f)$ are drawn below. Note that for $(\operatorname{Log}(x_1)-1)^2+(\operatorname{Log}(x_2)-1)^2>1$, the curves appear to approach each other exponentially fast.* $\diamond$



The maximum in Definition 1.2 merely compares the monomials of f evaluated at $x=(e^{u_1},\ldots,e^{u_n})=:e^u$. So $\operatorname{Trop}_+(f)$ approximates $\operatorname{Log} Z_+(f)$ by using $c_je^{a_j\cdot u}+c_{j'}e^{a_{j'}\cdot u}$ as a proxy for $f(e^u)$, in the subregions of $u\in\mathbb{R}^n$ where $c_jc_{j'}<0$, $|c_je^{a_j\cdot u}|=|c_{j'}e^{a_{j'}\cdot u}|$, and this value is the *maximal* absolute value of any monomial of f. In particular, we see that both curves from our last example are ambiently isotopic to lines, and $\operatorname{Trop}_+(f)$ and $\operatorname{Log} Z_+(f)$ turn out to be close *metrically* as well as topologically. We now generalize beyond curves.

Definition 1.4. *For any $\mathcal{A}=\{a_1,\ldots,a_t\}\subset\mathbb{R}^n$ we define the $(n+1)\times t$ matrix $\widehat{\mathcal{A}} := \begin{bmatrix} 1 & \cdots & 1 \\ a_1 & \cdots & a_t \end{bmatrix}$ and, when f has support $\mathcal{A}$, we set $\delta_f := \operatorname{Rank}\widehat{\mathcal{A}} - 1$.* $\diamond$

Remark 1.5. *We always have $\delta_f+1\leq\min\{n+1,t\}$ thanks to basic linear algebra. In particular, δ_f is merely the dimension of the smallest affine subspace containing the support of f. So $t\geq\delta_f+1$ for any n-variate t-nomial f and, in Example 1.3, we had $\mathcal{A}$ the vertices of a triangle congruent to*  *and $(\delta_f,t)=(2,3)$.* $\diamond$

One should think of δ_f as a more honest measure of the number of variables on which f depends, e.g., $f(x_1,x_2,x_3) = 1 - 3x_1x_2x_3 + 2x_1^7x_2^7x_3^7 + x_1^{25}x_2^{25}x_3^{25}$ has $\delta_f=1$ because in fact $f(x)=g(x_1x_2x_3)$ where $g(u)=1-3u+2u^7+u^{25}$.

We denote the empty set by $\emptyset$. $\operatorname{Trop}_+(f)$ then approximates $\operatorname{Log} Z_+(f)$ as follows:

Theorem 1.6. *If f is a n-variate t-nomial with $t=\delta_f+1$ and $t\geq 2$ then:*

0. *The set $\{x\in\mathbb{R}^n_+ \mid f(x)>0\}$ is either empty or contractible.*
1. *The following are equivalent: (a) $Z_+(f)=\emptyset$, (b) $\operatorname{Trop}_+(f)=\emptyset$, and (c) the coefficients of f all have the same sign.*
2. *$Z_+(f)$ (resp. $\operatorname{Trop}_+(f)$) is ambiently isotopic in $\mathbb{R}^n$ to an affine hyperplane when it is non-empty. Furthermore, if $t=2$ and either of $\operatorname{Log} Z_+(f)$ or*

[2] For instance, the circle ◯ and the trefoil knot  , as subsets of $\mathbb{R}^3$, are homeomorphic but *not* ambiently isotopic.

$\mathrm{Trop}_+(f)$ *is non-empty, then both are equal to the* same *affine hyperplane.*

3. *Every point of* $\mathrm{Log}\ Z_+(f)$ *is within distance* $\log \delta_f$ *of some point of* $\mathrm{Trop}_+(f)$. *Also,* $\mathrm{Trop}_+(f)$ *is a union of at most* $\frac{1}{4}(\delta_f+1)^2$ *many* $(n-1)$*-dimensional polyhedral cones meeting only along common faces.*

Theorem 1.6 refines more general results [2, Lem. 3.2 & Thm. 3.4] on complex roots.[3] Note that the cases $t \in \{0,1\}$ are trivial, since $Z_+(0)=\mathbb{R}^n_+$ and $Z_+(cx^a)=\emptyset$ for any $c \in \mathbb{R}^*$ and $a \in \mathbb{Z}^n$. So we have started with the simplest non-vacuous case $t=\delta_f+1$, also known as the *simplex case*.

Theorem 1.6 thus approximately solves a single n-variate $(n+1)$-nomial equation $f(x)=0$ over $\mathbb{R}^n_+$. Before discussing *systems* of equations let us focus on Question (S5) for the simplest univariate polynomial equations of arbitrary degree.

Remark 1.7. $\mathrm{Trop}_+(f)$ *is a straightforward refinement of the* Archimedean *tropical variety from [2], which gives rough metric information on* $\mathrm{Log}\,|Z_{\mathbb{C}}(f)|$ *where* $Z_{\mathbb{C}}(f)$ *denotes the set of complex roots of* f *(and the absolute value is applied coordinate-wise).* $\mathrm{Trop}_+(f)$ *can also be thought of as an instance of a* chart *in the sense of* Viro's patchworking *[25], but augmented with an additional metric guarantee not present in earlier work. Tropical varieties over* $\mathbb{C}$ *have precursors, in one variable, dating back to early 20th century work of Ostrowski and late 19th century work of Hadamard: See, e.g., the references in [2].* $\diamond$

1.4. Approximating Radicals. Let $\mathbb{N}$ denote the positive integers. An often overlooked family of equations is the humble univariate binomial equation $x^d=c$, say with $c,d \in \mathbb{N}$. It will be instructive to look at both the *arithmetic* complexity [9] and *bit* complexity [1] of computing dth roots, formalized as follows:

Problem 1.8. *Given* $b,c,d \in \mathbb{N}$, *find a positive rational* $\tilde{x}$ *with* $|\tilde{x}-c^{1/d}|<\frac{1}{2^b}$.

Note that just counting field operations to measure complexity misses an important subtlety: Can we efficiently decide whether an approximate solution $\tilde{x}$ is bigger, equal, or smaller than $c^{1/d}$? One could equivalently compare $\tilde{x}^d$ and c (since we don't know the value of $c^{1/d}$ a priori), but then $\tilde{x}^d$ can have d times as many digits as $\tilde{x}$. So, particularly if one is working by hand or with limited resources, one should also care about the *size* of the intermediate calculations, in addition to counting field operations and inequality decisions. Luckily, Problem 1.8 admits an algorithm with low arithmetic complexity that naturally leads to an implementable algorithm with low bit complexity as well.

Definition 1.9. *Recall the usual computer science big-O and* Ω *notation: When we compare functions* $r,s:\mathbb{R}^k \longrightarrow \mathbb{R}$ *by saying* $r=O(s)$ *(resp.* $r=\Omega(s)$*), this will simply mean that there exist constants* $C,M>0$ *such that* $r(x_1,\ldots,x_k) \leq Ms(x_1,\ldots,x_k)$ *(resp.* $r(x_1,\ldots,x_k) \geq Ms(x_1,\ldots,x_k)$*) for all* $x_1,\ldots,x_k \geq C$. $\diamond$

For Problem 1.8 we can naturally interpret the total size of our input and output as $O(\log(c)+\log(d)+b)$, i.e., a quantity that reasonably bounds from above (up to a constant multiple) the number of bits of needed to write c, d, and $\tilde{x}$. A *polynomial-time algorithm* for Problem 1.8 should thus have bit complexity $(\log(d)+\log(c)+b)^{O(1)}$.

Remark 1.10. *It is important to clarify input size because many earlier papers in computational algebra use a quantity closer to* $d+\log(c)+b$ *as the input size for Problem 1.8. So our version of polynomial-time implies significantly faster algorithms.* $\diamond$

Lemma 1.11. *There is an algorithm that, for any input* $b,c,d \in \mathbb{N}$, *finds an* $\tilde{x} \in \mathbb{Q}$ *satisfying* $|\tilde{x}-c^{1/d}|<\frac{1}{2^b}$ *using just* $O(\log(d)\log(b+\log c))$ *field operations and inequality decisions over* $\mathbb{Q}$. ■

The preceding arithmetic complexity upper bound is in fact asymptotically optimal, up to the underlying O-constant, when both c and d are fixed.

Theorem 1.12. *[11] Computing* $\sqrt{2}$ *to accuracy* $\frac{1}{2^b}$ *requires* $\Omega(\log b)$ *arithmetic operations.* ■

Recall that a complexity *lower* bound (such as Theorem 1.12) is in essence a speed limit for *all* algorithms solving a particular computational problem, within a particular computational model. The underlying computational model for Theorem 1.12 can be taken to be the *RAM model with operations* $\{+,-,\times,/,\lfloor\cdot\rfloor,>\}$ *on real numbers, and operations* `AND`, `OR`, `NOT`, *and* `XOR` *on bits* [11].

If we look more closely at the algorithm underlying Lemma 1.11 (first derived by Yinyu Ye around 1994), and apply some Diophantine approximation (e.g., in the sense of [3]), then we can also solve Problem 1.8 within polynomial *bit* complexity. In particular, the two main phases of Ye's algorithm are:

1. $O(\log\log c)$ steps of *bisection* to find the unique $j \in \{1,\ldots,\lceil \frac{\log c}{d\log((4d-4)/(4d-5))}\rceil\}$ with the interval $((\frac{4d-4}{4d-5})^{(j-1)d},(\frac{4d-4}{4d-5})^{jd}]$ containing c.
2. $\ell=O(\log(b+\log c))$ steps of *Newton iteration*, starting from $z^{(0)}:=(\frac{4d-4}{4d-5})^j$, to obtain $\tilde{x}:=z^{(\ell)}$.

[3] See Appendix, downloadable from Item #89 of `https://people.tamu.edu/~rojas/list2.html` .

Bisection simply refers to the classical technique of halving the size of an interval $I := (r, s)$, with $g(r)g(s) < 0$, containing a root of a continuous $g : I \longrightarrow \mathbb{R}$: Compute the sign of $g(\frac{r+s}{2})$ and replace I with its left (resp. right) half, according as $g(r)g(\frac{r+s}{2}) < 0$ or not.[4] *Newton iteration* is the classical technique of starting with a sufficiently good guess, $z^{(0)}$, for a root ζ of an analytic $f : \mathbb{C} \longrightarrow \mathbb{C}$, and then forming a sequence of approximations $\left(z^{(\ell)}\right)$ with $z^{(\ell)} \longrightarrow \zeta$ via the recurrence $z^{(\ell+1)} := z^{(\ell)} - \frac{f(z^{(\ell)})}{(f'(z^{(\ell)}))^{-1}}$. The size of the interval from Phase 1 above happens to be small enough so that Newton iteration converges quickly enough for the complexity bound from Lemma 1.11 to be attained (see, e.g., [9, Ch. 8]).

Evaluating the binomial $x^d - c$ at a rational number other than 1 could result in $\Omega(d)$ bits to keep track of. So one might think that the bit complexity of computing dth roots is doomed to be $\Omega(d)$ instead of our desired $(\log(c) + \log(d) + b)^{O(1)}$. However, Phase 1 of Ye's algorithm from Lemma 1.11 only needs the *sign* of $x^d - c$ at a rational x. Determining this sign is the $m = 2$ case of the following fundamental problem, which can be solved efficiently enough for our purposes:

Rational Binomial Sign Problem (RBSP). *Given $\alpha_1, \ldots, \alpha_m \in \mathbb{Q} \cap \mathbb{R}_+$ with (integer) numerators and denominators no greater than $\mathscr{A}$, and integers $b_1, \ldots, b_m$ with absolute value at most B, what is the sign of the binomial $(\prod_{i=1}^m \alpha_i^{b_i}) - 1$?*

Identifying (deterministic) *time* with (deterministic) bit complexity, our central complexity bound is:

Theorem 1.13. *[23] The RBSP can be solved in time $O\big(31^m \log^m(\mathscr{A}) \log(B) \log^2(\log(\mathscr{A}) \log B)\big)$.* ■

The key trick is determining $\operatorname{sign}(\sum_{i=1}^m b_i \log \alpha_i)$ instead of comparing the underlying monomial to 1. The sign of the sum can be computed efficiently by approximating the underlying logarithms (see, e.g., [23, Thm. 3.5]): The number of bits of accuracy needed to correctly solve RBSP this way can be derived via Baker's famous theorem on *Linear Forms in Logarithms* (see, e.g., [3]).

Recall the usual computer science little-o notation: When $r, s : \mathbb{R} \longrightarrow \mathbb{R}$ the equality $r = o(s)$ simply means that $\lim_{x \longrightarrow \infty} \frac{r(x)}{s(x)} = 0$. We can then solve Problem 1.8 by combining Lemma 1.11 and Theorem 1.13:

Corollary 1.14. *For any $b, c, d \in \mathbb{N}$ we can find a positive rational $\tilde{x}$ with $|\tilde{x} - c^{1/d}| < \frac{1}{2^b}$ within time $\log(d)[\log^{2+o(1)}(cd) + (b + \log c)^{1+o(1)}]$, i.e., in time quasi-cubic in the input size.* ■

[4] So in Lemma 1.11 one takes $g(x) = ((\frac{4d-4}{4d-5})^d)^x - c$ and $I := [0, 2^k]$ where $2^{k-1} < \frac{\log c}{d \log((4d-4)/(4d-5))} \leq 2^k$.

While there are more general (and fully implemented) algorithms for solving arbitrary univariate polynomial equations over $\mathbb{R}$ and $\mathbb{C}$, these more general algorithms have worst-case complexity super-linear in d, which is *not* polynomial-time in our setting.

We now extend our observations to the simplest (*non*-linear) *square* systems, i.e., systems of multivariate equations with equally many equations and variables.

1.5. $(n+1)$-nomial $n \times n$ Systems. Let us now consider polynomial systems of the form

$$F := (f_1, \ldots, f_n) = \begin{cases} c_{1,1}x^{a_1} + \cdots + c_{1,t}x^{a_t} \\ \vdots \\ c_{n,1}x^{a_1} + \cdots + c_{n,t}x^{a_t} \end{cases} \quad (1)$$

where the $c_{i,j}$ are real and $\{a_1, \ldots, a_t\} \subset \mathbb{Z}^n$ has cardinality t. We call such an F a *t-nomial $n \times n$ system (over $\mathbb{R}$) supported in $\{a_1, \ldots, a_t\}$*. The simplest non-trivial case is $t = \delta_f + 1$ (see Remark 1.5). Indeed, if $t \leq n$ and $F(\zeta) = 0$ for some $\zeta \in (\mathbb{R}^*)^n$, then the vector of monomials $[\zeta^{a_1}, \ldots, \zeta^{a_t}]^\top$ must be a right null-vector for $[c_{i,j}]$. Since $[c_{i,j}]$ generically[5] has 0-dimensional right null-space when $t \leq n$, we see that such an F generically has no roots in $(\mathbb{R}^*)^n$. So, for this subsection, let us assume $\delta_f = n$ and $t = n + 1$.

Since dividing each equation by $x^{a_{n+1}}$ leaves the roots in $(\mathbb{R}^*)^n$ unchanged, we may assume $a_{n+1} = \mathbf{O}$. Assuming in addition that the left $n \times n$ submatrix of the coefficient matrix $[c_{i,j}]$ is invertible (also true generically), we can then simply apply row-reduction to the System (1) to obtain the equivalent system:

$$(x^{a_1} - \gamma_1, \ldots, x^{a_n} - \gamma_n) \quad (2)$$

with $a_1, \ldots, a_n$ linearly independent (since we assumed $\delta_f = n$). Since System (2) has no roots in $\mathbb{R}^n_+$ if any γ_i is zero, and since the γ_i are all nonzero for generic $[c_{ij}]$, let us assume further that $\gamma_1 \cdots \gamma_n \neq 0$.

One can then solve System (2) via linear algebra over the integers. More precisely, we use a classic factorization of integer matrices summarized below.

Definition 1.15. *[23, Sec. 2.1] We call a matrix $U \in \mathbb{Z}^{n \times n}$ with determinant ± 1* unimodular*. Given any matrix $A \in \mathbb{Z}^{n \times t}$, we then call any identity of the form $UAV = S$, with $U \in \mathbb{Z}^{n \times n}$ and $V \in \mathbb{Z}^{t \times t}$ both unimodular, and $S \in (\mathbb{N} \cup \{0\})^{n \times t}$ diagonal with (i, i) diagonal entry s_i such that $s_1 | s_2, \ldots, s_{n-1} | s_n$, a* Smith factorization *of A. We also call S* the Smith normal form *of A.* ⋄

While Smith factorizations are never unique (e.g., $(\pm I)I(\pm I) = I$), Smith normal *forms* are always unique and efficiently computable. The connection

[5] A condition involving parameters $w_1, \ldots, w_N \in \mathbb{C}^N$ holds *generically* if and only if there is a polynomial $Q \in \mathbb{C}[x_1, \ldots, x_N] \setminus \{0\}$ whose non-vanishing at $(w_1, \ldots, w_N)$ implies the condition holds. For the case at hand, letting Q be any $t \times t$ sub-determinant of $[c_{i,j}]$ suffices.

between solving binomial systems and Smith factorization comes from the following basic fact:

Proposition 1.16. *Suppose $A, B \in \mathbb{Z}^{n\times n}$, $x=(x_1,\ldots,x_n)$ is a vector of indeterminates, and x^A is the vector of monomials $\left(x_1^{a_{1,1}}\cdots x_n^{a_{n,1}},\ldots,x_1^{a_{1,n}}\cdots x_n^{a_{n,n}}\right)$, where $A=[a_{i,j}]$. Then $(x^A)^B = x^{AB}$ and, if A is unimodular, the function defined by $x \mapsto x^A$ defines an analytic group automorphism of $(\mathbb{C}^*)^n$ that restricts to an analytic group automorphism of $\mathbb{R}^n_+$.* ■

This proposition is classic in the toric geometry/Lie group literature. In particular, we see that for any equality of the form $x^A = c$, applying a *monomial* changes of variables $x = y^U$ is equivalent to left-multiplying A by U. Similarly, multiplying and dividing equations in $x^A = c$ is nothing more than right-multiplying A by V (and computing c^V) for some suitable matrix V.

Example 1.17. *Consider the binomial system* $\left(x_1^{-3}x_2^{10}x_3^4,\ \ x_1^5x_2^{-2}x_3^{20},\ \ x_1^{-11}x_2^{18}x_3^{16}\right) = (2,3,5)$, *i.e.,* $x^A = (2,3,5)$ *with* $A = \begin{bmatrix} -3 & 5 & -11 \\ 10 & -2 & 18 \\ 4 & 20 & 16 \end{bmatrix}$. *Consider* $U = \begin{bmatrix} 3 & 1 & 0 \\ -18 & -7 & 4 \\ 1694 & 659 & -377 \end{bmatrix}$ *and* $V = \begin{bmatrix} 1 & 444 & 457 \\ 0 & -33 & -34 \\ 0 & 1 & 1 \end{bmatrix}$ *(which each happen to be unimodular). Then* $UAV = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 388 \end{bmatrix}$. *Letting* $S := UAV$, $x := (y_1,y_2,y_3)^U$, *and computing* $(2,3,5)^V$ *then yields* $y^{UAV} = y^S = \left(y_1^1, y_2^4, y_3^{388}\right) = \left(2^1, 2^{444}\cdot 3^{-33}\cdot 5^1, 2^{457}\cdot 3^{-34}\cdot 5^1\right)$, *i.e., we can solve our binomial system by computing some explicit monomials in (positive) radicals. Note also that while the largest entry of U,V,S is about* 83 *times bigger than the largest entry of A, the number of* digits *in the largest entry of U,V,S is merely* twice *the number of digits of the largest entry of A.* ⋄

Letting ω denote the matrix multiplication exponent (currently known to be no larger than 2.371339, thanks to recent work of Alman, Duan, Vassilevska-Williams, Xu, Xu, and Zhou), we have the following:

Theorem 1.18. *[23, Sec. 2.1] Suppose $A \in \mathbb{Z}^{n\times t}$ has rank r and all entries of absolute value at most d. Then a Smith factorization for A can be found in time $ntr^{\omega-1}\log^{1+o(1)}(rd)\log(nt)$, with all the nonzero entries of U,V,S having* log*-absolute value $O(r\log(rd))$.* ■

The bound above assumes that we use an $O(b\log b)$-time algorithm for b-bit integer multiplication, e.g., [20].

1.6. Binomial Systems and a Curse of Dimensionality. For convenience, we will call roots in $\mathbb{R}^n_+$ of a polynomial system *positive*. Theorem 1.18 and Proposition 1.16 enable us to efficiently count the positive roots of binomial systems as follows:

Corollary 1.19. *[23, Prop. 2.7] Suppose $c = (c_1,\ldots,c_n) \in (\mathbb{R}^*)^n$, $A = [a_1,\ldots,a_n] \in \mathbb{Z}^{n\times n}$, $UAV = S$ is a Smith factorization of A, and $c' := (c'_1,\ldots,c'_n) := c^V$. Then $G := (x^{a_1}-c_1,\ldots,x^{a_n}-c_n)$ and $(y_1^{s_1}-c'_1,\ldots,y_n^{s_n}-c'_n)$ have the same number of roots in $\mathbb{R}^n_+$ (resp. $(\mathbb{R}^*)^n$, $(\mathbb{C}^*)^n$). In particular, G has exactly 0, 1, or infinitely many roots in $\mathbb{R}^n_+$ under the following respective conditions:*

- $\boxed{0}$: *Some c_i is negative or* $[\mathrm{Rank}(A) = j < n$ *and* $c'_i \neq 1$ *for some* $i \in \{j+1,\ldots,n\}]$.
- $\boxed{1}$: $c \in \mathbb{R}^n_+$ *and* $\det A \neq 0$.
- $\boxed{\infty}$: $c \in \mathbb{R}^n_+$, $\mathrm{Rank}(A) = j < n$, *and* $c'_{j+1} = \cdots = c'_n = 1$. ■

So we now have all the key ingredients to efficiently answer our earlier Questions (S1)–(S4) for solving $n \times n$ binomial systems, i.e., we can detect positive roots, decide whether there are finitely many, and count them exactly. Also, in the event of infinitely many solutions (which can happen for non-generic coefficients), it is not hard to show that the set of positive solutions of such a system is ambiently isotopic to a proper linear subspace of $\mathbb{R}^n$.[6]

To solve Question (S5) for binomial systems — and to fully understand numerical accuracy — we will need to review the classical notion of *height*:

Definition 1.20. *For any nonzero rational number $\frac{p}{q}$ with $p,q \in \mathbb{Z}$ and $\gcd(p,q) = 1$, its* (absolute logarithmic) height *is $h(p/q) := \max\{\log|p|, \log|q|\}$.*

(We set $h(0) := 0$.) Note in particular that the number of bits needed to write the numerator and denominator of an $x \in \mathbb{Q}$ (in lowest terms) is $O(h(x))$.

Corollary 1.19 gives us a clear reduction from solving $n \times n$ binomial systems to solving n instances of the univariate binomial case. However, a satisfying answer to Question (S5), on *numerically* approximating coordinates of solutions, is still not clear. The key difficulty is that the number of bits needed to write the solutions can blow up exponentially in the ambient dimension n. A simple example is the binomial system $F_n := (x_1 - 2, \frac{x_2}{x_1^2} - 1, \ldots, \frac{x_n}{x_{n-1}^2} - 1)$, which, in the positive orthant, has unique root $(2, 2^{2^1}, \ldots, 2^{2^{n-1}})$ *with last coordinate of height* $\Omega(2^n)$.

This complexity blow-up can be mitigated by approximating more carefully and applying some earlier work of Smale on the convergence of Newton's method (extended to analytic functions $F : \mathbb{C}^n \longrightarrow \mathbb{C}^n$ [9, Rem. 7, pg. 166]). Smale's approach is more intrinsic than approximating within ε or $\frac{1}{2^b}$, and avoids such extraneous parameters: Output a rational start point, of *low* height, close enough to a true root so that Newton iteration converges quickly. This also makes storing solutions dramatically more efficient.

We use $|z|$ to denote the usual L^2-norm of any $z \in \mathbb{R}^n$.

[6] See Appendix, downloadable from Item #89 of `https://people.tamu.edu/~rojas/list2.html` .

Definition 1.21. *[9] Given any analytic function $F : \mathbb{C}^n \longrightarrow \mathbb{C}^n$, we call $\zeta \in \mathbb{C}^n$ a* non-degenerate *root of F if and only if the Jacobian $F'(\zeta)$ is invertible. We then call $z^{(0)}$* an approximate root of F (in the sense of Smale, with associated true root ζ) *if and only if F has a non-degenerate root $\zeta \in \mathbb{C}^n$ such that the Newton iterates $(z^{(n)})_{n\in\mathbb{N}\cup\{0\}}$ satisfy $|z^{(n)} - \zeta| \leq (\frac{1}{2})^{2^{n-1}}|z^{(0)} - \zeta|$ for all $n \geq 1$.* $\diamond$

Theorem 1.12 shows us that the above convergence rate is asymptotically the best one can hope for, up to replacing $\frac{1}{2}$ by a smaller positive constant.

Example 1.22. *Recalling the system F_n, $z^{(0)} := (1, \ldots, 1)$ is in fact an approximate root in the sense of Smale for F_n! In particular, one can prove via induction (and an elementary calculation) that n Newton iterations yield $z^{(n)}$* exactly equal *to the unique root of F_n in $\mathbb{R}^n_+$. More to the point, $z^{(0)}$ requires only n bits to write down, saving us from storing $(2^0 + 1) + (2^1 + 1) + \cdots + (2^{n-1} + 1) = \Omega(2^n)$ bits to record the unique positive root of F_n exactly.* $\diamond$

Approximate roots are useful only if they exist and can be found and written down quickly. For binomial systems, we can find them by reducing to n instances of the univariate case and then carefully applying bisection to each univariate binomial. Applying [9, Ch. 8] to our framework then enables a solution to our numerical solving question (S5).

Theorem 1.23. *Following the notation of Corollary 1.19, assume further that $a_1, \ldots, a_n$ are linearly independent, $c_i \in \mathbb{Q} \cap \mathbb{R}_+$, and $h(c_i) \leq h$ for all i. Then in time $\left[(n \log d)^2(n + h)\right]^{2+o(1)}$ we can find $w_{i,j} \in \mathbb{Q} \cap \mathbb{R}_+$, having height $O(h \log(d) + n \log(dn))$ for all i, j such that, if $y_i := (w_{i,1}, \ldots, w_{i,n})^{v_i}$ and $z^{(0)} = (y_1, \ldots, y_n)^U$, then $z^{(0)}$ is an approximate root of G.* $\blacksquare$

While $z^{(0)}$ can have coordinates of height exponential in n, we need only store $[w_{i,j}]$, U, and V (occupying bit-size polynomial in the input size $nh + n^2 \log d$) until we need the bits comprising the unique root of G.

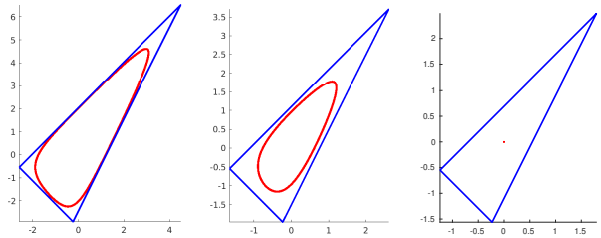
2 n-variate $(n + 2)$-nomials

Examples 1.1 and 1.3 gave our first hint that positive zero sets of polynomials are related to piecewise linear geometry. Denote the line segment connecting points $x, y \in \mathbb{R}^n$ by $\overline{xy} := \{\lambda x + (1 - \lambda)y \mid 0 \leq \lambda \leq 1\}$. Recall that a set $C \subseteq \mathbb{R}^n$ is *convex* if and only if $\overline{xy} \subseteq C$ for all $x, y \in C$. The *convex hull* of any subset $S \subset \mathbb{R}^n$, denoted $\mathrm{Conv}S$, can be defined as the intersection of all convex sets containing S.

A *polytope* is the convex hull of any finite subset, and the very simplest (non-empty) polytopes turn out to be *n-simplices*: These are the convex hulls of sets $\{a_1, \ldots, a_{n+1}\} \subset \mathbb{R}^d$ that do *not* lie in any affine n-plane. For instance, 0-simplices are points, 1-simplices are line segments of positive length, 2-simplices are non-degenerate triangles, and 3-simplices are pyramids with triangular bases. Clearly, $\mathbb{R}^n$ contains no d-simplex with $d > n$.

Vertex sets of simplices (e.g., the supports from our last section) are the simplest non-empty point sets we can study, so the next step up is a *(combinatorial) circuit*: A set of points $\{a_1, \ldots, a_{n+2}\} \subset \mathbb{R}^d$ of cardinality $n + 2$ such that the $(d + 1) \times (n + 2)$ matrix $\widehat{\mathcal{A}} := \begin{bmatrix} 1 & \cdots & 1 \\ a_1 & \cdots & a_{n+2} \end{bmatrix}$ has rank $n + 1$. The terminology comes from matroid theory, not to be confused with *arithmetic* circuits from complexity theory [1].

2.1. Circuits Deform to Quadrics. The following 3 elliptic curve images illustrate a subtlety as we go from $(n + 1)$-nomials to $(n + 2)$-nomials:



$c = -23$ $\quad$ $c = -9$ $\quad$ $c = -6$

Example 2.1. *From left to right, the images in each plot above are the union of $\mathrm{Trop}_+(f)$ and $\mathrm{Log}\, Z_+(f)$ where f is the 2-variate 4-nomial $3x_2^2 + 2x_1^3 + 1 + cx_1x_2$, and c is respectively -23, -9, or -6. The subtlety is that $\mathrm{Trop}_+(f)$ fails to capture the correct isotopy type of $Z_+(f)$ in the right-most case above. In particular, $Z_+(f)$ is isotopic to a circle, a single point, or $\emptyset$, according as c is less than, equal to, or greater than -6. On the other hand, $\mathrm{Trop}_+(f)$ is isotopic to a circle, a point, or $\emptyset$, according as c is less than, equal to, or greater than $-2^{1/3} \cdot 3^{1/2} = -2.182247273...$.* $\diamond$

We can in fact *always* correctly recover the isotopy type of $Z_+(f)$, and *often* still get a metric guarantee of nearness for $\mathrm{Trop}_+(f)$, provided one considers the sign of a particular binomial in the coefficients of f.

Definition 2.2. *For any circuit $\mathcal{A} = \{a_1, \ldots, a_{n+2}\}$, we define its* vector of affine relations *(or b-vector) to be any generator $b = (b_1, \ldots, b_{n+2})^\top \in \mathbb{R}^{(n+2)\times 1}$ for the right null space of $\widehat{\mathcal{A}}$. We also call $\mathcal{A}$ a* degenerate *circuit if and only if some coordinate of b is 0. Finally, for any non-degenerate circuit $\mathcal{A}$, we define $\mathcal{A}_\pm := \{a_i \mid \mathrm{sign}(b_i) = \pm\}$ and call $\{\mathcal{A}_+, \mathcal{A}_-\}$ a* Radon partition *of $\mathcal{A}$.* $\diamond$

For example, a possible b-vector for the circuit $\{(0, 2), (3, 0), (0, 0), (1, 1)\}$ from Example 2.1 is

$(3,2,1,-6)^\top$. Also, it is easy to see that the thickened points in  and  each define circuits, but the thickened points in  define a degenerate circuit. Note that while the b-vector for a circuit is *not* unique, it is determined by a 1-dimensional subspace that *is* uniquely determined by the circuit $\mathcal{A}$. Similarly, the *un*ordered pair $\{\mathcal{A}_+, \mathcal{A}_-\}$ is uniquely determined by $\mathcal{A}$. For example, the convex hulls of the Radon partition for  are the line segments  (drawn separately, since they in fact intersect in a unique point). Similarly, the Radon partition for  can be visualized in color as  . Our next central construction is a variant of [18, Prop. 1.8, pg. 274].

Definition 2.3. *If $f \in \mathbb{R}\left[x_1^{\pm 1}, \ldots, x_n^{\pm 1}\right]$ can be written as $f(x) = \sum_{i=1}^{n+2} c_i x^{a_i}$ with all the c_i nonzero and $\mathcal{A} = \{a_1, \ldots, a_{n+2}\}$ a non-degenerate circuit with b-vector b, then we call f* sign-compatible *if and only if $\operatorname{sign}(b_1 c_1) = \cdots = \operatorname{sign}(b_{n+2} c_{n+2})$. We then define the* circuit discriminant function for $\mathcal{A}$ *to be*

$$\Xi_{\mathcal{A}}(f) := \Xi_{\mathcal{A}}(c_1, \ldots, c_n) := \prod_{i=1}^{n+2} \left|\frac{c_i}{b_i}\right|^{b_i} - 1. \qquad \diamond$$

For instance, returning to Example 2.1, $3x_2^2 + 2x_1^3 + 1 - 9x_1x_2$ and $-3x_2^2 - 2x_1^3 - 1 + 9x_1x_2$ are sign-compatible with their underlying circuit, while $x_2^2 + x_1^3 + 1 + x_1x_2$ and $x_2^2 - x_1^3 + 1 + x_1x_2$ are not. Note that while $\Xi_{\mathcal{A}}$ depends on the b-vector of $\mathcal{A}$, the *zero set* of $\Xi_{\mathcal{A}}$ determines a partition of the sign-compatible polynomials supported on $\mathcal{A}$ that depends *only* on $\mathcal{A}$. For instance, in Example 2.1, we could take $b = (3,2,1,-6)^\top$, which then yields $\Xi_{\mathcal{A}}(c_1,c_2,c_3,c_4) = \left|\frac{c_1^3 c_2^2 c_3^1 \cdot 6^6}{3^3 \cdot 2^2 \cdot 1^1 c_4^6}\right| - 1$. But regardless of which b-vector we picked, $\operatorname{sign}(\Xi_{\mathcal{A}})$ is completely determined by the sign of the transcendental expression

$$3\log|c_1| + 2\log|c_2| + \log|c_3| - 6\log|c_4| - 3\log(3) - 2\log(2) + 6\log(6).$$

Remark 2.4. *Theorem 1.18 easily implies that, for any circuit $\mathcal{A}$, we can efficiently find a suitable $b \in \mathbb{Z}^{n+2}$. Our earlier Theorem 1.13 then implies that we can efficiently determine $\operatorname{sign}(\Xi_{\mathcal{A}}(f))$ for any fixed n, provided f has rational coefficients. However, our results on positive zero sets in fact generalize to real zero sets of exponential sums of the form $\sum_{i=1}^{t} c_i e^{a_i \cdot y_i}$ with the $a_i \in \mathbb{R}^n$ [17, 6].* $\diamond$

Radon partitions and $\operatorname{sign}(\Xi_{\mathcal{A}})$ are exactly the computable quantities that determine the isotopy type of $Z_+(f)$ in the sign-compatible case. Let $Z_{\mathbb{R}}(f)$ denote the real zero set of a polynomial f, and call any connected component of a subset of $S \subseteq \mathbb{R}^n$ a *piece*.

Theorem 2.5. *[6] Suppose $f \in \mathbb{R}\left[x_1^{\pm 1}, \ldots, x_n^{\pm 1}\right]$ is sign-compatible with a non-degenerate circuit $\mathcal{A}$ of cardinality $n+2$, let $\varepsilon := \operatorname{sign}(\Xi_{\mathcal{A}}(f)) \in \{-1, 0, 1\}$, let $\{\mathcal{A}_+, \mathcal{A}_-\}$ be a Radon partition for $\mathcal{A}$, and let $p+1$ be the cardinality of $\mathcal{A}_+$. Then $Z_+(f)$ is isotopic to $Z_{\mathbb{R}}\left(\left(\sum_{i=1}^{p} y_i^2\right) + \varepsilon - \left(\sum_{i=p+1}^{n} y_i^2\right)\right)$. In particular, f has a degenerate root in $\mathbb{R}^n_+$ if and only if $\Xi_{\mathcal{A}}(f) = 0$, in which case, the degenerate root is unique. Furthermore, $Z_+(f)$ always has at most 2 pieces, and $Z_+(f)$ has 2 pieces implies that $\mathcal{A}_+$ or $\mathcal{A}_-$ has cardinality 2.* ■

For instance, one can check that in Example 2.1, with $b = (3,2,1,-6)^\top$, we have $\Xi_{\mathcal{A}}(3,2,1,c) < 0 \iff c < -6$. Also, $\mathcal{A}_+ = \{(0,2),(3,0),(0,0)\}$ and $\mathcal{A}_- = \{(1,1)\}$ define a Radon partition for the underlying support and, for our choice of b, we see that $\Xi_{\mathcal{A}}(3,2,1,-9) < 0$ and, indeed, $Z_+(3x_2^2 + 2x_1^3 + 1 - 9x_1x_2)$ is isotopic to the circle defined by $Z_{\mathbb{R}}(y_1^2 + y_2^2 - 1)$.

The sign-*in*compatible case turns out to be much easier than the sign-compatible case.

Lemma 2.6. *[14] Following the notation of Theorem 2.5, if $\mathcal{A}$ is a non-degenerate circuit, but f is sign-incompatible, then $Z_+(f) = \emptyset$ if and only if all the coefficients of f have the same sign. In particular, if $Z_+(f) \neq \emptyset$ then $Z_+(f)$ is isotopic to $\mathbb{R}^{n-1}$.* ■

Remark 2.7. *For brevity, we note that the degenerate circuit case is very similar to the non-degenerate circuit case, except that the resulting quadric can depend on $< n$ variables [6].* $\diamond$

Beyond the correct determination of isotopy type, one can also use $\operatorname{Trop}_+(f)$ in the circuit case to efficiently approximately locate $\log Z_+(f)$, at least for *most* choices of the coefficients:

Theorem 2.8. *[2, 6] Following the notation of Definition 2.3 and Theorem 2.5, let $\mathcal{B} := \sum_{i=1}^{n+2} b_i \log|b_i|$. Then, under the conditions below, $Z_+(f)$ and $\operatorname{Trop}_+(f)$ are ambiently isotopic in $\mathbb{R}^n$, and every point of $\operatorname{Log} Z_+(f)$ is within distance $\log(n+1)$ of some point of $\operatorname{Trop}_+(f)$: (1) f is sign incompatible, or (2) f is sign compatible and $\operatorname{sign}(\mathcal{B}) \sum_{i=1}^{n+2} b_i \log|c_i|$ is either > 0 or $< -|\mathcal{B}|$.* ■

So in the circuit case, in all but two orthants of $\mathbb{R}^{n+2}$, $\operatorname{Trop}_+(f)$ is guaranteed to be *both* topologically and metrically close to $\operatorname{Log} Z_+(f)$, just as in the simplex case. In the remaining two orthants, we still have the same guarantee, save for a neighborhood of width $|\mathcal{B}|$ about a hyperplane in log-coefficient space. More generally, as k increases, the positive zero sets of real n-variate $(n+k)$-nomials become much harder to predict, due to the increasing intricacy of the $\mathcal{A}$-discriminant complement in coefficient space.

2.2. Finding Isotopy Type Faster. Via Remarks 2.4 and 2.7, we easily obtain the following algorithmic complexity addendum to Theorem 2.5:

Corollary 2.9. *Following the notation of Theorem 2.5, assume additionally that the coordinates of $\mathcal{A}$ have absolute value at most d, and the coefficients of f are integers of height at most h. Also, let us allow $\mathcal{A}$ to possibly be a degenerate circuit, and f to possibly be sign-incompatible. Then in time $(h + n\log(dn))^{n+2+o(1)}$ we can count the number of pieces of $Z_+(f)$, determine whether $Z_+(f)$ is a smooth hypersurface, and find an explicit quadric hypersurface isotopic to $Z_+(f)$.* ■

The best previous complexity bounds for topologically classifying real zero sets (see, e.g., [5]), specialized to our setting, yield a complexity upper bound of $(n^{n\log n}d^n)^{O(\log^2 n)}$ arithmetic operations. So our speed-ups are significant.

However, our complexity bounds for determining the isotopy type of positive zero sets in the circuit case are unfortunately still exponential in n. The key bottleneck turns out to be determining $\mathrm{sign}(\Xi_A(f))$, which is clearly equivalent to our earlier problem RBSP. Since we solved RBSP in Theorem 1.13 by using the theory of linear forms in logarithms, we can try to use newer Diophantine bounds of this type to speed up our algorithms here.

Such an improvement follows immediately [3] from the truth of the following *refined* version of the famous (1985) Masser-Oesterlé *abc*-Conjecture:

Baker's Refined *abc*-Conjecture (1998). *For any $n\in\mathbb{N}$ let $s(n)$ (resp. $\mathrm{Rad}(n)$) denote the number of primes (resp.* product *of all primes) dividing n. Let $a,b,c\in\mathbb{N}$ satisfy $a+b=c$ and $\gcd(a,b,c)=1$. Then for all such a,b,c we must have*

$$c = O\Big(\frac{\log^{s(abc)}\mathrm{Rad}(abc)}{s(abc)!}\mathrm{Rad}(abc)\Big).$$

Theorem 2.10. *[6] If Baker's abc-Conjecture is true then the complexity bound from Corollary 2.9 can be improved to $n^8(h\log d)^{4+o(1)}$.* ■

The (weaker) original *abc*-Conjecture was known to already imply numerous arithmetic geometry results outside our current reach, e.g., work of Elkies gives a short proof of an *effective* version of Falting's Theorem on rational points on curves of genus ≥ 2 if Masser-Oesterlé's version is true (see, e.g., [10, Ch. 12] and the references therein). There is an even older conjecture — the *Lang-Waldschmidt Conjecture* (from 1978, on linear forms in logarithms) — that also implies the speed-up in Theorem 2.10 [6]. Unfortunately, the latter conjecture, as well as Baker's *abc*-Conjecture, appear far out of reach as of early 2025.

However, a tantalizing aspect of Theorem 2.10 is its contrapositive: If we can somehow prove that counting pieces for the positive zero set of a polynomial supported on a circuit is *not* doable in time polynomial in $nh\log d$, then we can prove the *falsity* of Baker's refined *abc*-Conjecture *and* the Lang-Waldschmidt Conjecture. This hints at why proving certain complexity lower bounds is so hard.

A more pragmatic strategy to speeding up Corollary 2.9 is to instead prove that isotopy type can be computed much faster for "most" input circuit polynomials. This can be done if the underlying bottleneck problem, RBSP, can be solved quickly for "most" inputs. The latter was proved recently in [13, Cor. 1.16], so here is an immediate consequence:

Corollary 2.11. *Following the notation of Corollary 2.9, fix $\mathcal{A}$. Then for a fraction of $1-O\big(\frac{1}{2^{n+h}}\big)$ of such f, we can improve the complexity bound from Corollary 2.9 to $O(n^{3.371}h^3\log^3(nd))$.* ■

2.3. Systems Supported on Circuits. The first complication with solving circuit systems is that the number of positive roots of a generic $n\times n$ circuit system can be any integer in $\{0,\ldots,n+1\}$, unlike the simplex case where we merely had to distinguish between 0 and 1. Fortunately, counting can still be done efficiently, at least for *fixed* n.

Theorem 2.12. *[23] Suppose $F := (f_1,\ldots,f_n)$ is supported on a circuit $\mathcal{A}\subset\mathbb{Z}^n$, with all coordinates in $\{-d,\ldots,d\}$, and the coefficient matrix $[c_{i,j}]$ of F has integer entries of height $\leq h$ and is generic. Then, in time $(n^2h\log d)^{(2+o(1))n}$, we can determine the number of roots of F in $\mathbb{R}^n$, $(\mathbb{R}^*)^n$, and $\mathbb{R}^n_+$.* ■

Similar to Corollary 2.11, one can seek provable speed-ups for most inputs, and this is currently in progress (see, e.g., [13, Cor. 1.23]).

A more subtle problem with solving circuit systems is that distinct roots can be so close that we need exponentially many bits to distinguish them.

Example 2.13. *Consider, for $n\geq 2$, the following $n\times n$ circuit system, G_n, found by Weixun Deng: $(x_1-\frac{1}{2},\frac{x_2}{x_1^d}-1,\ldots,\frac{x_{n-1}}{x_{n-2}^d}-1,(\frac{x_n}{x_{n-1}^d}-2)(\frac{x_n}{x_{n-1}^d}-1))$. G_n has degenerate underlying circuit, bit size $O(n\log d)$, and exactly two roots in $\mathbb{R}^n_+$:*

$$\rho := (\tfrac{1}{2^{d^0}},\ldots,\tfrac{1}{2^{d^{n-2}}},\tfrac{1}{2^{d^{n-1}-1}})$$
$$\text{and } \zeta := (\tfrac{1}{2^{d^0}},\ldots,\tfrac{1}{2^{d^{n-2}}},\tfrac{1}{2^{d^{n-1}}}),$$

both with last coordinate of height $\Omega(d^{n-1})$, i.e., exponential in the input size. Moreover, these roots satisfy $\log|\rho-\zeta| = \log(\rho_n-\zeta_n) = -\Omega(d^{n-1})$, i.e., their coordinates match in their first $\Omega(d^{n-1})$ most significant bits. ⋄

Curiously, Koiran proved that, for $n=1$, the complex roots of $n\times n$ circuit systems are *well*-separated, i.e., in the special case of *univariate trinomials*, we only need polynomially many bits to distinguish roots (see, e.g., the discussion around [24, Thm. 1.6]). More to the point, although one might sometimes

need dramatically more accuracy to distinguish roots of $n \times n$ circuit systems with $n \geq 2$, certain *monomials* in the roots will always be much easier to distinguish.

For instance, for our G_n above, observe that $\rho_n/\rho_{n-1}^d = 2$ while $\zeta_n/\zeta_{n-1}^d = 1$, and thus these monomials in the roots are at distance 1. Note also that the underlying vector power $(0, \ldots, -d, 1)$ lies in the support of G_n. This approach to finding well-separated monomials in the roots of circuit systems extends to *all* circuit systems, and is pursued further in the Texas A&M Ph.D. thesis of Weixun Deng. Furthermore, recent *condition number estimates* for polynomial systems with random coefficients (see, e.g., [22] and more recent work by Ergür and Tonelli-Cueto) give evidence that the roots of circuit systems should be well-spaced on average, i.e., examples like G_n might be provably rare.

3 $(n+k)$-nomials with $k \geq 3$, and $\mathbb{Q}_p$

n-variate t-nomials with $t \leq \delta_f + 2$ are precisely the cases where we can completely classify isotopy type for positive zero sets *and* (in many cases) practically solve such systems numerically. For $t \geq \delta_f + 3$, we quickly run into interesting obstructions:

O1. *The number of bits needed to distinguish positive roots can be exponential in the input size already for univariate tetranomials (see, e.g., [24, Thm. 1.5]).*

O2. *There is currently no complete classification of the possible isotopy types for $Z_+(f)$, even when f is a n-variate $(n+3)$-nomial with $\delta_f = n$. However, we at least now know the maximal number of pieces: 3 [14], optimally improving a recent $O(n)$ upper bound [7].*

O3. *For any $\varepsilon > 0$, merely deciding $Z_+(f) \stackrel{?}{=} \emptyset$ is already* **NP***-hard when $t = n + n^\varepsilon$ and $\delta_f = n$ [23, Intro.]. Whether this* **NP***-hardness persists for $t = n + O(1)$ (e.g., $t = n+3$ or even $t = n+2$) is unknown.*

O4. *The best current upper bounds for the number of pieces of $Z_+(f)$ for arbitrary real n-variate $(n+k)$-nomials (assuming $Z_+(f)$ is smooth), and the number of non-degenerate positive roots of an arbitrary $(n+k)$-nomial $n \times n$ system, appear to be far from tight: They are respectively $8(n+1)^{k-2}2^{(k-2)(k-3)/2}$ (when $k \geq 2$) [17, 7] and $\frac{e^2+3}{4}n^{k-1}2^{(k-1)(k-2)/2}$ [8].*

Nevertheless, just as we saw that randomizing the input can yield better algorithmic and geometric complexity bounds for $t \leq \delta_f + 2$, the same appears to hold for $t \geq \delta_f + 3$ as well. Perhaps the most beautiful example of this is a recent *average-case* fewnomial bound of Bürgisser, Ergür, and Tonelli-Cueto [12]: For any choice of independent real Gaussians $c_{i,j}$ with mean 0, and variance depending only on j, the average number of positive roots of the resulting random F is $\leq \frac{1}{2^{n-1}}\binom{n+k}{n}$, i.e., polynomial in n (resp. k) if k (resp. n) is fixed!

As for approximating real roots of random polynomial systems, there is important recent work on estimating the condition numbers for such systems. Summarizing this work adequately would require far more space than we have here. So for brevity we merely point the reader to Lairez's recent positive solution of *Smale's 17th Problem* [22]: This problem concerned the complexity of finding a single *complex* approximate root of a random polynomial system, in time polynomial in the number of monomial terms, in a setting where all monomials up to degree d are assumed to appear and the coefficients are complex.

Smale himself suggested that the analogous problem over the *real* numbers is harder. For example, when the supports are fixed and the coefficients are, say, real Gaussians, the number of roots is *not* a fixed constant with probability 1 (unlike the number of complex roots, which here would be constant with probability 1). So counting real roots should be addressed separately from approximating real roots via the following problem:

Problem 3.1. *Consider any support $\mathcal{A} \subset \mathbb{Z}^n$ of cardinality $n+k$, with points having coordinates of absolute value at most d. Find an algorithm that, for a random real $n \times n$ system supported in $\mathcal{A}$, computes the number of positive roots of F exactly, using $[\binom{n+k}{n} \log d]^{O(1)}$ many arithmetic operations on average.*

Just as in Smale's 17th Problem, we have left the underlying probability distribution unspecified. Note, however, that for $k \ll d$ our desired complexity bound is much smaller asymptotically than the complexity bound ultimately derived in the solution to the original Smale's 17th Problem: $\binom{n+d}{n}^{O(1)}$.

In closing, we point out that most of our main results also have analogues (subtly different, some in progress) over the p-adic rationals $\mathbb{Q}_p$: See the final section of the Appendix[7], as well as [24].

Acknowledgements. I thank Weixun Deng, Alperen Ergür, Alicia Dickenstein, Erin Lipman, Gregorio Malajovich, Grigoris Paouris, Bruce Reznick, and Máté Telek for important conversations around the theme of this paper. I also thank the referees for their suggestions that significantly improved this paper.

References

[1] S. Arora and B. Barak, *Computational complexity. A modern approach.* Cambridge U. Press, 2009.

[7] Downloadable from Item #89 of https://people.tamu.edu/~rojas/list2.html .

[2] M. Avendaño; R. Kogan; M. Nisse; and J. M. Rojas, *"Metric Estimates and Membership Complexity for Archimedean Amoebae and Tropical Hypersurfaces,"* J. Complexity, Vol. 46, June 2018, pp. 45–65.

[3] A. Baker, *"Logarithmic forms and the abc-conjecture,"* Number theory (Eger, 1996), pp. 37–44, de Gruyter, Berlin, 1998.

[4] M. Baker and S. Payne, *Nonarchimedean and Tropical Geometry,* Simons Symposia, Springer-Verlag, 2016.

[5] S. Basu and M.-F. Roy, *"Divide and conquer roadmap for algebraic sets,"* Discrete Comput. Geom., 52:278–343, 2014.

[6] F. Bihan; E. Croy; W. Deng; K. Phillipson; R. J. Rennie; and J. M. Rojas, *"Quickly Computing Isotopy Type for Exponential Sums Over Circuits,"* ACM Communications in Computer Algebra, Volume 57, Issue 3, pp. 152–155 (2023). https://doi.org/10.1145/3637529.3637538.

[7] F. Bihan, T. Humbert, S. Tavenas, *"New bounds for the number of connected components of fewnomial hypersurfaces,"* Math ArxiV preprint `arXiv:2208.4590`

[8] F. Bihan and F. Sottile, *"New fewnomial upper bounds from Gale dual polynomial systems,"* Moscow Mathematics Journal, 7 (2007), No. 3, pp. 387–407.

[9] L. Blum; F. Cucker; M. Shub; and S. Smale, *Complexity and Real Computation,* Springer-Verlag, 1998.

[10] E. Bombieri and W. Gubler, *Heights in Diophantine Geometry,* new mathematical monographs: 4, Cambridge U. Press, 2006.

[11] N. H. Bshouty; Y. Mansour; B. Schieber; and P. Tiwari, *"A tight bound for approximating the square root,"* Inform. Process. Lett. 63 (1997), no. 4, pp. 211–213.

[12] P. Bürgisser, A. A. Ergür, and J. Tonelli-Cueto, *"On the Number of Real Zeros of Random Fewnomials,"* SIAM J. Appl. Algebra Geom., 3(4), pp. 721–732, 2019.

[13] W. Deng; A. Ergür; G. Paouris; and J. M. Rojas, *"Feasibility of Circuit Polynomials without Purple Swans,"* in Proceedings of ISSAC 2024, ACM Press, pp. 429–436.

[14] W. Deng, J. M. Rojas, and C. Russell, *"Optimal Bounds for the Number of Pieces of Near-Circuit Hypersurfaces,"* submitted, 2025.[8]

[15] P. Dutta, N. Saxena, and T. Thierauf, *"Weighted Sum-of-Squares Lower Bounds for Univariate Polynomials Imply* $\mathbf{VP} \neq \mathbf{VNP}$*,"* comput. complex. (2024) 33:3.

[16] M. Feinberg, *Foundations of Chemical Reaction Network Theory,* Applied Mathematical Sciences, Vol. 202, Springer-Verlag, 2019.

[17] J. Forsgård, M. Nisse, J. M. Rojas, *"New Subexponential Fewnomial Hypersurface Bounds,"* Math ArXiV preprint `arXiv:1710.00481`

[18] I. M. Gel'fand; M. M. Kapranov; and A. V. Zelevinsky, *Discriminants, Resultants and Multidimensional Determinants,* Birkhäuser, 1994.

[19] F. Gnad, S. Ren, J. Cox, J. V. Olsen, B. Macek, M. Oroshi, and M. Mann, *"PHOSIDA (phosphorylation site database): management, structural and evolutionary investigation, and prediction of phosphosites,"* Genome Biology **8**, R250 (2007).

[20] D. Harvey and J. van der Hoeven, *"Integer multiplication in time* $O(n \log n)$*,"* Ann. Math. 193(2) (March 2021), pp. 563–617.

[21] P. Koiran, *"Shallow circuits with high-powered inputs,"* in Proc. ICS 2011 (2nd Symposium on Innovations in Computer Science), Tsinghua University Press, Beijing.

[22] P. Lairez, *"A deterministic algorithm to compute approximate roots of polynomial systems in polynomial average time,"* Found. Comput. Math. 17:5 (2017), pp. 1265–1292.

[23] J. M. Rojas, *"Counting Real Roots in Polynomial-Time via Diophantine Approximation,"* Found. Comput. Math. (2024), `https://doi.org/10.1007/s10208-022-09599-z`

[24] J. M. Rojas and Y. Zhu, *"Root Repulsion and Faster Solving for Very Sparse Polynomials Over p-adic Fields,"* J. Number Theory, Vol. 241, Dec. 2022, pp. 655–699.

[25] O. Y. Viro, *"Patchworking Real Algebraic Varieties,"* Math ArXiV preprint `arXiv:math0611382`

J. Maurice Rojas

Credits

Figure 1 and Figure 2 are courtesy of J. Maurice Rojas.
Photo of J. Maurice Rojas is courtesy of Texas A&M Arts & Sciences

[8] Downloadable as Item #90 from `people.tamu.edu/~rojas/list2.html` .