# Fixed Points, Nash Equilibria, and the Existential Theory of the Reals

Marcus Schaefer
School of Computing
DePaul University
243 South Wabash
Chicago, Illinois 60604, USA
mschaefer@cdm.depaul.edu

Daniel Štefankovič
Computer Science Department
University of Rochester
Rochester, NY 14627-0226
stefanko@cs.rochester.edu

**Abstract**

We introduce the complexity class $\exists\mathbb{R}$ based on the existential theory of the reals. We show that the definition of $\exists\mathbb{R}$ is robust in the sense that even the fragment of the theory expressing solvability of systems of strict polynomial inequalities leads to the same complexity class. Several natural and well-known problems turn out to be complete for $\exists\mathbb{R}$; here we show that the complexity of decision variants of fixed-point problems, including Nash equilibria, are complete for this class, complementing work by Etessami and Yannakakis [13].

**Keywords:** Fixed point problems, Brouwer, existential theory of the real numbers, Nash equilibrium, computational complexity

## 1 Introduction

Many computational problems in geometry, graph drawing and other areas can be shown decidable using the (existential) theory of the real numbers, including the rectilinear crossing number, the Steinitz problem, and finding a Nash equilibrium; what is less often realized, though there are some exceptions, is that the existential theory of the reals captures the computational complexity of many of these problems precisely. In previous papers, the first author investigated some geometric problems related to graph drawing [30, 31]. In the current paper, we present tools to deal with semialgebraic and algebraic sets, such as effective lower bounds on the distance between two semialgebraic sets. These tools are useful in solving computational complexity problems related to the existential theory of the reals.

We illustrate this by applying them to a variety of fixed point-problems and Nash equilibria, complementing work of Etessami and Yannakakis [13].

From an algebraic point of view, there are two ways to define the existential theory of the reals depending on whether we allow equality or not; for example, take the rectilinear crossing number, which is the smallest number of crossings in a straight-line drawing of a graph. The rectilinear crossing number problem can be expressed as a system of strict inequalities, and, as a consequence, a drawing realizing the rectilinear crossing number of a graph can be assumed to have vertices with rational coordinates (even if some of them may require exponential precision, see [4]); similarly, intersection graph problems can typically be captured by strict inequalities (for example, the problems in [30], including segment intersection graphs). On the other hand, fixed-point problems need equality to be modeled in the existential theory of the reals, and so their solution sets do not necessarily contain rational points: the fixed point of $f(x) = 2/x$ is $\sqrt{2}$. In Section 4 we prove the rather unexpected result that from a computational point of view, these two variants of the existential theory of the reals are the same, justifying the introduction of a single complexity class $\exists\mathbb{R}$. Section 2 reviews the logical and computational side of the existential theory of the reals, and Section 3 presents some tools based on algebraic geometry which turn out to be useful in handling problems in the class $\exists\mathbb{R}$. In Section 5 we then show that several fixed-point problems are complete for this class.

Since the class $\exists\mathbb{R}$ was first introduced (in an earlier version of this paper, as well as [30, 31]), there have been several new $\exists\mathbb{R}$-completeness results, including:

- straight-line realizability of abstract topological graphs (even the complete graph) [22],
- recognizing unit disk graphs and dot-product graphs [18],
- simultaneous geometric planarity [10],
- a data exchange problem for arithmetic schema mapping [35],
- stretchability of pseudocircles [19].

Together with the results from earlier papers this already gives us a sizable collection of complete problems for $\exists\mathbb{R}$ from many different areas (see [24, 32, 4, 20, 34, 28, 27, 12, 8], for example; a survey on the topic is in preparation [29]; there is a wikipedia page on $\exists\mathbb{R}$ [39]).

We assume that the reader is familiar with basic notions of computational complexity, including polynomial time, polynomial-time many-one reducibilities and complexity classes such as **NP**, and **PSPACE** [25, 33].

## 2 The Existential Theory of the Reals

The existential theory of the reals, ETR, is the set of true sentences of the form

$$(\exists x_1, \ldots, x_n)\ \varphi(x_1, \ldots, x_n),$$

where $\varphi$ is a quantifier-free $(\vee, \wedge, \neg)$-Boolean formula over the signature $(0, 1, +, *, <, \leq, =)$ and the sentence is interpreted over the universe of real numbers.[1]

In a 1948 paper entitled "A Decision Method for Elementary Algebra and Geometry", Alfred Tarski proved a quantifier elimination result for the existential theory of the reals, which implied that the *theory of the reals*, with arbitrary quantifiers, is decidable. In his 1988 dissertation, Canny showed that ETR can be decided in **PSPACE**, to date the best theoretical upper bound on ETR. For a recent survey, see [23], for experimental comparisons of running times, see [15].

We will find it useful to distinguish two special cases of ETR. Let INEQ be the subset of ETR, in which we do not allow $\vee, \neg$ and $=$, that is $\varphi$ is a conjunction of atoms of the form $s < t$ and $s \leq t$ ($s = t$ can be expressed as $s \leq t \wedge t \leq s$ so not allowing equality is not a real restriction). Furthermore, let STRICT INEQ be the subset of INEQ, in which we do not allow $\leq$, that is, $\varphi$ is a conjunction of strict inequalities $s < t$.

Following our first impulse as complexity theorists we use STRICT INEQ and INEQ to define complexity classes $\exists_<\mathbb{R}$ and $\exists_=\mathbb{R}$ as the downward closures of these problems under polynomial-time many-one reductions; with this definition $\exists_<\mathbb{R} \subseteq \exists_=\mathbb{R}$ and there seems to be evidence that these two classes are different: solutions to an INEQ-type problem can require irrational numbers, e.g. $x^2 = 2$, while solutions to STRICT INEQ can always be perturbed slightly to make them rational. These differences are of an algebraic nature and, in a slightly surprising twist of events, do not affect the computational complexity of these problems. It turns out that $\exists_<\mathbb{R} = \exists_=\mathbb{R}$ as we will see in Section 4. In other words, INEQ polynomial-time many-one reduces to STRICT INEQ.

[1] When writing formulas in the existential theory of the reals, we will freely use integers and rationals, since these can easily be eliminated without affecting the length of the formula substantially.

Note that $\mathbf{NP} \subseteq \exists_< \mathbb{R}$ (a result first explicitly stated by Shor [32]), since we can express satisfiability of a Boolean formula in $\exists_< \mathbb{R}$. For example, $\varphi = (x \vee \overline{y} \vee z) \wedge (\overline{x} \vee y \vee z) \wedge (\overline{x} \vee \overline{y} \vee \overline{z})$ is equivalent to

$$\begin{aligned}(\exists x, y, z, \varepsilon)[\ &(-\varepsilon < x < 1 + \varepsilon) \wedge (-\varepsilon < y < 1 + \varepsilon) \wedge (-\varepsilon < z < 1 + \varepsilon) \\ &\wedge (x(1-y)z) + ((1-x)yz) + ((1-x)(1-y)(1-z)) < \varepsilon, \\ &\wedge \varepsilon > 0 \wedge\ \varepsilon < 1/104].\end{aligned}$$

If the formula is satisfiable, then we assign a variable the value 0 if it is true and 1 otherwise, so that the sum becomes $0 < \varepsilon$; in the example: $x = y = 0$ and $z = 1$ will do. For the reverse direction, assume $x, y, z$, and $\varepsilon$ satisfy the formula. Note that $0 < \varepsilon < 1/104 = 1/8(1 + 4m)$, where $m = 3$ is the number of clauses. Each term of the sum is at least $-\varepsilon \cdot (1 + \varepsilon)^2 \geq -4\varepsilon$; so the whole sum is at least $-4m\varepsilon \geq -12/104$. For the sum to be less than $1/104$, every term must be less than $1/104 + 12/104 = 1/8$. Each term is the product of three factors, so at least one factor must be less than $(1/8)^{1/3} = 1/2$. Let the corresponding variable be true if the factor is of the form $x$ and false if it is of the form $1 - x$. This yields a satisfying assignment of the original Boolean formula $\varphi$.

So, with respect to classical complexity classes, we can summarize our present knowledge of the existential theory of the reals by

$$\mathbf{NP} \subseteq \exists_< \mathbb{R} \subseteq \exists_= \mathbb{R} \subseteq \mathbf{PSPACE},$$

where the last implication is due to Canny's result [9].

**Remark 2.1** (Models of Real Computation)**.** We are not treating the existential theory of the real numbers as a model of real computation in two senses: there are no "real" real numbers in $\mathsf{ETR}$ (other than rationals), and we do not present a machine model for the classes $\exists_< \mathbb{R}$ and $\exists_= \mathbb{R}$. It turns out that it is possible to construct a machine model for $\exists_= \mathbb{R}$; this was essentially done by Blum, Shub, and Smale [6] whose results imply that the languages in $\{0,1\}^*$ decided by a real non-deterministic polynomial-time Turing machine that has no registers for real numbers are precisely the languages in $\exists_= \mathbb{R}$.[2] This connection between $\mathsf{ETR}$ and the BSS-model does not

[2]The class of Boolean languages decided by real non-deterministic Turing machines without real constants was introduced under the name BP($\mathbf{NP}^0_{\mathbb{R}}$) by Bürgisser and Cucker [7, Corollary 8.2] who observed that the feasibility problem $\mathsf{FEAS}$ (which we will define in Section 4) is complete for that class, based on work by Blum, Shub, and Smale [6]. Since $\mathsf{FEAS}$ is also complete for $\exists_= \mathbb{R}$, as we will show in Theorem 4.1, the two classes coincide.

give any insights on the problems dealt with in the current paper, which is why we do not discuss this (or other models of real computation) any further. However, the BSS-model could be a first step toward more structural results such as oracle separations. ◁

# 3 Semi-Algebraic Sets of Bounded Complexity

Our goal in this section is to collect a couple of tools for dealing with semi-algebraic sets of bounded complexity. In particular, we want to show that such sets always contain a point not too far from the origin (Corollary 3.4), that we can find a ball that contains a bounded semi-algebraic set (Corollary 3.7), and that there is a lower bound on the distance between two semi-algebraic sets that have positive distance (Corollary 3.8).

We will use the following notations throughout this paper: $[n]$ as an abbreviation for the set $\{1, \ldots, n\}$, $\|x\|^2 := \sum_{i \in [n]} x_i^2$ is the square of the distance of $x$ from the origin, the *distance* $d(A, B)$ between two sets $A, B \subseteq \mathbb{R}^n$ is defined as $d(A, B) = \inf\{d(a, b) : a \in A, b \in B\}$, where $\|a - b\|$ is the Euclidean distance between two points. The *bitlength of an integer* $n$ is the smallest number $b(n)$ of bits to write down the number in binary digits. So $b(n) = \lfloor(\log_2(n)\rfloor + 1$ for $n \geq 1$ and $b(0) = 1$. In particular, $n < 2^{b(n)}$ for all $n$.

## 3.1 Definitions and Basic Results

In algebraic geometry *semi-algebraic sets* are solution sets to systems of polynomial equalities and inequalities; taking a more logical approach, we say a set $S \subseteq \mathbb{R}^n$ is *semi-algebraic* if there is a $(\vee, \wedge, \neg)$-Boolean quantifier-free formula over the signature $(0, 1, +, *, <, \leq, =)$ and with (free) variables $x = (x_1, \ldots, x_n)$ so that $S = \{x \in \mathbb{R}^n : \varphi(x)\}$. If $\varphi$ does not contain $\vee$ or $\neg$, we call the set a *basic semi-algebraic set*. If, moreover, $\varphi$ does not contain $<$ and $\leq$, the set is *algebraic*. We use $|\varphi|$ to denote the *length* of $\varphi$, that is, the number of bits necessary to write down $\varphi$. The *(bit)-complexity of a semi-algebraic set* is the shortest length of any formula defining the set.

In algebraic geometry, semi-algebraic sets are defined as finite unions of basic semi-algebraic sets; since we gave a definition via defining formulas, we have to prove that result. We also show that the complexity of the basic semi-algebraic sets need be no larger than the complexity of the original set.

**Lemma 3.1.** *Every semi-algebraic set of complexity at most $L$ is the finite union of basic semi-algebraic sets each of complexity at most $L$. We can*

*assume that the defining formula of each of the basic semi-algebraic sets does not contain the comparison operator* $\leq$.

*Proof.* Let $\varphi$ be a formula of bitlength at most $L$ defining the semi-algebraic set $S = \{x : \mathbb{R}^n : \varphi(x)\}$. If $\varphi$ contains any negations, we push them to the lowest level of the formula, and absorb them in the atomic formulas: $\overline{s < t}$ becomes $t \leq s$, $\overline{s \leq t}$ becomes $t < s$, and $\overline{s = t}$ turns into $s < t \vee t < s$. Replace all inequalities of type $s \leq t$ by $s < t \vee s = t$ and convert the resulting formula $\psi$ into disjunctive normal form: $\psi = \bigvee_{i \in I} \psi_i$ for some $I \subseteq \mathbb{N}$. Then each $\psi_i$ defines a basic semi-algebraic set (not using $\leq$), and $S$ is the union of these sets. Each $\psi_i$ uses at most one clause of each disjunction we introduced when rewriting $s \leq t$ and $\overline{s = t}$, so each of its clauses stems from a different clause in the original $\varphi$ and so $|\psi_i| \leq |\varphi|$. □

The following lemma gives us a way to replace the defining formula of a semi-algebraic set with a single multivariate polynomial. Multivariate polynomials are sums (or differences) of monomials, the complexity of a polynomial is the number of bits needed to write it down in this form; this means, that while we may write $(x + 1)(y + 1)$ to simplify notation, the polynomial has to be written out as $xy + x + y + 1$ explicitly. Other measures of complexity for polynomials include bounds on the bitlength of the coefficients (as integers) of monomials, typically written as $\tau$, and the *(total) degree* of the polynomial $f$ which is defined as the maximum over the sum of the variable exponents in each monomial term occurring in $f$. E.g. $f(x, y, x) = 5x^7y^2 - 2x^3yz^6$ has total degree $3 + 1 + 6 = 10$.

**Lemma 3.2.** *If $S$ is a semi-algebraic set in $\mathbb{R}^n$ given by a formula $\varphi$ of complexity at most $L \geq 3$, then we can efficiently (that is in time polynomial in $L$ and $n$) construct*

(*i*) *a family of quadratic polynomials $f_j : \mathbb{R}^{n+m} \to \mathbb{R}$, $j \in [k]$, so that $S = \{x \in \mathbb{R}^n : (\exists y \in \mathbb{R}^m) \bigwedge_{j \in [k]} f_j(x, y) = 0\}$, for some $m, k \leq 3L$,*

(*ii*) *a non-negative polynomial $g : \mathbb{R}^{n+m} \to \mathbb{R}$ of degree at most 4 so that $S = \{x \in \mathbb{R}^n : (\exists y \in \mathbb{R}^m)\, g(x, y) = 0\}$, for some $m \leq 3L$.*

*The coefficients of the polynomials $f_j$ have bitlength at most $L$ and coefficients in $g$ have bitlength at most $2L$.*

This lemma is an efficient version of the well-known fact that every semi-algebraic set is the projection of an algebraic set (the set of zeros of a polynomial). Similar to the situation in the Blum-Shub-Smale model of

real computation, it is unlikely that the degree of $g$ can be reduced below 4, since it can be decided in polynomial time whether a polynomial of degree at most 3 has a zero [36].

*Proof of Lemma 3.2.* Clearly, $(i)$ implies $(ii)$: with the family $f$ from $(i)$, define $g(x,y) = \sum_{1\le j\le k}(f_j(x,y))^2$. Then $f_j(x,y) = 0$ for all $j \in [k]$ if and only if $g(x,y) = 0$, and the bitlength of coefficients at most doubles. Note that $g$ is non-negative. Hence, it is sufficient to prove $(i)$. Let $S = \{x \in \mathbb{R}^n : \varphi(x)\}$, where $\varphi$ has complexity at most $L$. As in Lemma 3.1, push all negations to the atomic level, and replace $\overline{s < t}$ with $t \le s$, $\overline{s \le t}$ with $t < s$, and $\overline{s = t}$ with $s < t \vee t < s$. This at most doubles the length of $\varphi$. We now use a trick based on an idea due to Tseitin [37, 21], to build quadratic polynomials $f_j(x,y)$ with new variables $y \in \mathbb{R}^m$, for some $m$, so that $\varphi(x) \equiv (\exists y) \bigwedge_{j\in[k]} f_j(x,y) = 0$.

For any subformula $\gamma$ of $\varphi$ create a new real variable $y_\gamma$ and, as needed, $y'_\gamma$ and $y''_\gamma$. For any subterm $s$ of $\varphi$ create a new real variable $y_s$. We will ensure that for any $x$, $y$ with the property that $f_i(x,y) = 0$ for every $i \in [k]$, we have that $y_s = s$ and that if $y_\gamma = 0$, then $\gamma(x)$ is true. The variables $y'_\gamma$ and $y''_\gamma$ are needed for intermediate calculations only.

To simplify notation, we index the family of polynomials $f$ using subformulas and subterms of $\varphi$. Let $s$ be any subterm of $\varphi$. If $s = t \circ u$, we define $f_s(x,y) = s - (t \circ u)$ (where $\circ \in \{+,-,*\}$); if $s = x_i$, we define $f_s(x,y) = s - x_i$

Let $\gamma$ be any subformula of $\varphi$. If $\gamma = \alpha \vee \beta$, we define $f_\gamma(x,y) = y_\gamma - y_\alpha y_\beta$; if $\gamma = \alpha \wedge \beta$, we define $f_\gamma(x,y) = y_\gamma - (y_\alpha^2 + y_\beta^2)$. If $\gamma = (s = t)$, we define $f_\gamma(x,y) = y_\gamma - (y_s - y_t)$. If $\gamma = (s < t)$, we need to define two polynomials, let us call them $f_{\gamma,0}$, and $f_{\gamma,1}$; we define $f_{\gamma,0}(x,y) = y'_\gamma - (y''_\gamma)^2$, and $f_{\gamma,1}(x,y) = (y_t - y_s)y'_\gamma - (1 - y_\gamma)$. Note that if $f_{\gamma,0}(x,y) = f_{\gamma_1}(x,y) = 0$ and $y_\gamma = 0$, then $y'_\gamma \ge 0$ and $1 - y_\gamma = 1$, so $y_t > y_s$; the reverse need not be true.

Now, if $\varphi(x)$ is true, then by construction, we can choose values for $y_s$, $y_\gamma$, $y'_\gamma$ and $y''_\gamma$ so that for all $j$ we have $f_j(x,y) = 0$. If, on the other hand, for all $j$ we have $f_j(x,y) = 0$, then all the terms $f_\gamma$ and $f_s$ are zero. In particular, $y_\varphi = 0$, so, $\gamma$ is true (since this implication holds for each step of the recursive construction of $y_\varphi$).

The degree of the polynomials in $f$ are at most 2, and the bitlengths of coefficients was not increased by the construction. Since a formula $\varphi$ of length $L$ can have at most $L$ subformulas and subterms, we conclude that $k \le 2L$ (at most two polynomials per subformula), and $m \le 3L$ (at most three variables per subformula). □

### 3.2 Main Tools

The three main corollaries in this section (3.4, 3.7, 3.8) are based on corresponding results from algebraic geometry on systems of polynomial equalities and inequalities. For example, Vorobjov [38] and Grigor'ev and Vorobjov [14] were the first to show that every semi-algebraic set of complexity at most $L$ contains a point of distance at most $2^{2^{O(L)}}$ from the origin.[3] We use a more recent result due to Basu and Roy [3] which gives explicit constants (which may be of interest in applications).[4]

**Theorem 3.3** (Basu, Roy [3, Theorem 4])**.** *Let $(f_i)_{i\in[s]}$ be a family of polynomials of type $\mathbb{R}^n \to \mathbb{R}$ all of degree at most $d$ and with coefficients of bitlength at most $\tau$. Define*

$$R = ((2DN(2N-1)+1)2^{(2N-1)(\tau'+b(2N-1)+b(2DN+1))})^{1/2},$$

*with $d' = \max(2(d+1),6)$, $D = n(d'-2)+2$, $N = d'(d'-1)^{n-1}$, $\tau' = N(\tau_2 + b(N) + 2b(2D+1)+1)$, $\tau_2 = \tau_1 + 2(n-1)b(N) + (2n-1)b(n)$, $\tau_1 = D(\tau_0+4b(2D+1)+b(N))-2b(2D+1)-b(N)$, $\tau_0 = 2\tau+nb(d+1)+b(2d')+b(s)$. Then a ball of radius $R$ around the origin contains a point of every semi-algebraic set $S = \{x \in \mathbb{R}^n : f_i(x) \triangle_i 0\}$ that can be defined by choosing $\triangle_i \in \{>, <, =\}$.*

These estimates are much finer than what is needed for our purposes, since we are only bounding the overall complexity of the formula. Deriving the following bound is tedious, but straightforward, the details can be found in Appendix A.

**Corollary 3.4.** *Every semi-algebraic set in $\mathbb{R}^n$ of complexity at most $L \geq 4$ contains a point of distance at most $2^{L^{8n}}$ from the origin.*

The remaining two corollaries we base on a result by Jeronimo and Perrucci who showed that a positive polynomial (all values are greater than 0) defined over a simplex can be bounded away from 0.[5] Let $\Delta_n = \{x \in \mathbb{R}^n_{\geq 0}$ with $\sum_{i\in[n]} x_i \leq 1\}$ be the *standard simplex* in $\mathbb{R}^n$.

[3] The theorem can also be found in [2, Theorem 13.15] though the statement contains a typo in the radius of the ball.

[4] As far as complexity theory is concerned, the original result by Grigor'ev and Vorobjov would be sufficient, however.

[5] Using the simplex results to get estimates with explicit constants, was suggested to us by Jiří Matoušek.

**Theorem 3.5** (Jeronimo, Perruci [16]). *If $f : \mathbb{R}^n \to \mathbb{R}$ is a polynomial of degree $d$ so that $f(x) > 0$ for all $x \in \Delta_n$, and all coefficients of $f$ have bitlength at most $\tau$, then*

$$f(x) > 2^{-(\tau+1)d^{n+1}} d^{-(n+1)d^{n+1}}$$

*for all $x \in \Delta_n$.*

The obvious generalization from $\Delta_n$ to $\mathbb{R}^n$ fails; for example, $f(x, y) = x^2 + (1 - xy)^2$ is positive for all $x, y \in \mathbb{R}$ but cannot be bounded away from 0. Instead, we require that we know that $f$ is bounded away from 0.

**Corollary 3.6.** *If $f : \mathbb{R}^n \to \mathbb{R}$ is a polynomial of degree $d$ so that $f(x) \geq \delta > 0$ for all $x \in \mathbb{R}^n$ and some fixed $\delta$, and all coefficients of $f$ have bitlength at most $\tau$, then*

$$f(x) > 2^{-(\tau n^d+1)d^{n+1}} d^{-(n+1)d^{n+1}}$$

*for all $x \in \mathbb{R}^n$.*

*Proof.* Let $\Delta'_n = \{y \in \mathbb{R}^n_{\geq 0} : \sum_{i\in[n]} y_i < 1\}$ and define $r(y) = y/(1 - \sum_{i\in[n]} y_i)$ for $y \in \Delta'_n$. Then $r$ is a homeomorphism between $\Delta'_n$ and $\mathbb{R}^n_{\geq 0}$. Let $f(x) = \sum_{j\in J} a_j x^j$, where $J \subseteq \mathbb{N}^n$ and $x^j := x_1^{j_1} \cdots x_n^{j_n}$. The function $f \circ r$ is a rational function and can be written as $f(r(y)) = g(y)/h(y)$, where $g(y) = \sum_{j\in J} a_j y^j (1 - \sum_{i\in[n]} y_i)^{d-(\sum_{i\in[n]} j_i)}$ and $h(y) = (1 - \sum_{i\in[n]} y_i)^d$. We see that both $g$ and $h$ are polynomials of degree at most $d$, and the bitlength of their coefficients is bounded by $\tau' = n^d\tau$. Now for $y \in \Delta'_n$ we have $h(y) \leq 1$ and, by Theorem 3.5, $g(y) > 2^{-(\tau'+1)d^{n+1}} d^{-(n+1)d^{n+1}} = 2^{-(\tau n^d+1)d^{n+1}} d^{-(n+1)d^{n+1}}$. Therefore, $f(r(y)) = g(y)/h(y) \geq g(y) \geq 2^{-(\tau n^d+1)d^{n+1}} d^{-(n+1)d^{n+1}}$ for all $y \in \Delta'_n$. Hence $f(x) \geq 2^{-(\tau n^d+1)d^{n+1}} d^{-(n+1)d^{n+1}}$ for all $x \in \mathbb{R}^n_{\geq 0}$. By modifying the definition of $r$ we can establish this lower bound on $f$ for each of the hyperoctants of $\mathbb{R}^n$ proving the result. □

We derive two further consequences from Corollary 3.6: an upper bound (in terms of distance) on all points in a bounded semi-algebraic set, and a lower bound on the distance between two semi-algebraic sets that have a positive distance.

**Corollary 3.7.** *If a bounded semi-algebraic set in $\mathbb{R}^n$ has complexity at most $L \geq 5n$, then all its points have distance at most $2^{2^{L+5}}$ from the origin.*

There is a finer bound in terms of $n$, $d$, and $\tau$ due to Basu and Roy [3].

*Proof of Corollary 3.7.* Let $S = \{x \in \mathbb{R}^n : \varphi(x)\}$, where $\varphi$ has complexity at most $L$, be a bounded semi-algebraic set. Then $R = \sup_{x \in S} \|x\|^2 < \infty$. By Lemma 3.2, there is a polynomial $g$ of degree at most 4 with coefficients of bitlength at most $2L$, so that $S = \{x \in \mathbb{R}^n : (\exists y \in \mathbb{R}^m)\ g(x, y) = 0\}$. Let $f : \mathbb{R}^{n+m+1} \to \mathbb{R}$ be the polynomial defined by $f(x, y, u) = u^2 + (u\|x\|^2 - 1)^2 + g(x, y)$. Then $f$ has degree at most 4 and coefficients of bitlength at most $2L$. Moreover, $\inf_{x,y,u} f(x, y, u) \leq 1/R^2$: Let $x^{(j)}$ be a sequence of points in $S$ such that $\|x^{(j)}\|^2$ converges to $R$. Then for each $x^{(j)}$ there is a $y^{(j)}$ so that $g(x^{(j)}, y^{(j)}) = 0$. Then $f(x^{(j)}, y^{(j)}, 1/R) = 1/R^2 + (\|x^{(j)}\|^2/R - 1)^2$ which tends to $1/R^2$ as $j \to \infty$. By Corollary 3.6, $f$ can be bounded below by

$$2^{-(\tau n^d + 1)d^{n+1}} d^{-(n+1)d^{n+1}} \geq 2^{-(\tau n^4 + 2n + 3)4^{n+1}} \geq 2^{-2^{L+6}},$$

where we used $d \leq 4$ in the first inequality, and $\tau \leq 2L$ and $n \leq L/5$ in the second. Since (1) is a lower bound on $1/R^2$, we get $R^2 \leq 2^{2^{L+6}}$ and hence $R \leq 2^{2^{L+5}}$. □

The second consequence is a lower bound on the distance between two semi-algebraic sets.

**Corollary 3.8.** *If two semi-algebraic sets in $\mathbb{R}^n$ each of complexity at most $L \geq 5n$ have positive distance (for example, if they are disjoint and compact), then that distance is at least $2^{-2^{L+5}}$.*

Jeronimo, Perrucci, Tsigaridas [17, Theorem 2] give bounds on the minimum distance between two semi-algebraic sets in terms of $n$, $\tau$ and $d$, which is more than we need. Their result makes the stronger assumption that one of the two sets be compact.

*Proof of Corollary 3.8.* Let $S = \{x \in \mathbb{R}^n : \varphi(x)\}$ and $T = \{x \in \mathbb{R}^n : \psi(x)\}$ so that $\varphi$ and $\psi$ have complexity at most $L$. We can assume that both $S$ and $T$ are non-empty. By Lemma 3.1 both $S$ and $T$ are the finite union of basic semi-algebraic sets of complexity at most $L$, so we can choose basic semi-algebraic subsets of $S$ and $T$ that realize the minimum distance. So let us assume that $S$ and $T$ are basic. Lemma 3.2 gives us polynomials $g$ and $h$ each of degree at most 4 and with coefficients of bitlength at most $2L$ so that $S = \{x \in \mathbb{R}^n : (\exists y \in \mathbb{R}^m)\ g(x, y) = 0$ and $T = \{x : \mathbb{R}^n : (\exists y \in \mathbb{R}^m)\ h(x, y) = 0\}$ (we can pad $y$ if necessary so $m$ is the same and $g$ and $h$ have the same number of variables). Consider the polynomial $f(x, y, x', y') = g(x, y) + h(x', y') + \|x - x'\|^2$. Then $\inf_{x,x',y,y'} f(x, y, x', y')$ is a lower bound

on the square of the distance between $S$ and $T$ (pick $x$ in the closure of $S$ and $x'$ in the closure of $T$ so that $d(x,x') = d(S,T)$, then choose sequences of elements in $S$ and $T$ converging to $x$ and $x'$, with corresponding choices of $y$ and $y'$). Since $f$ has degree $d \leq 4 \leq 2L$ and its coefficients have bitlength $\tau \leq 2L$ as well, Corollary 3.6 implies that $f$ has a lower bound of $2^{-2^{L+6}}$ (just as in Equation (1) above, using the same estimates). Since this is a lower bound on the square of the distance which is less than 1, $2^{-2^{L+5}}$ is a lower bound on the distance. □

# 4 The Complexity Class $\exists\mathbb{R}$

We have defined three variants of the existential theory of the reals, ETR, INEQ, and STRICT INEQ; in this section we will show that they are all computationally equivalent. In particular, it will follow that $\exists_<\mathbb{R} = \exists_=\mathbb{R}$, which is a bit of a surprise, since algebraically these two classes differ. For the proof, we will use an intermediate problem FEAS which restricts ETR to formulas not containing $\neq$, $\vee$, $\wedge$, $<$ and $\leq$; in other words, FEAS asks whether a multivariate polynomial is *feasible*, that is, has a root over the reals.

**Theorem 4.1.** *The following problems are polynomial-time equivalent:* ETR*,* FEAS*,* STRICT INEQ*.*

*Proof.* In slightly different language, we already saw that ETR reduces to FEAS, that was what we proved in Lemma 3.2. Since STRICT INEQ is a special case of ETR, we are left with the proof that FEAS reduces to STRICT INEQ.

So suppose we are given a multivariate polynomial $g$ and ask whether there is an $x \in \mathbb{R}^n$ so that $g(x) = 0$. Let $L$ be the complexity of $g$ (recall that this is the number of bits required to write down $g$ as a sum of monomials). By Corollary 3.4 we know that if $S = \{x \in \mathbb{R}^n : g(x) = 0\}$ is not empty, it contains a point of distance at most $R = 2^{L^{8n}}$ from the origin. Consider the two semi-algebraic sets $\{(z,x) \in \mathbb{R}^{n+1} : g(x) = z, \|x\|^2 \leq R^2\}$ and $\{(z,x) \in \mathbb{R}^{n+1} : z = 0, \|x\|^2 \leq R^2\}$. If these two sets do not intersect, they have positive distance (both being compact), and, by Corollary 3.8, that distance is at least $2^{-2^{L+5}}$. Hence, $g = 0$ is equivalent to the system

$$-\delta < g(x), g(x) < \delta, \delta < 2^{-2^{L+5}}, \|x\|^2 \leq R^2$$

of strict inequalities being solvable. The inequality $\delta < 2^{-2^{L+5}}$ can be replaced by a sequence of at most $L+5$ inequalities using repeated squaring,

so we have shown that FEAS can be reduced to STRICT INEQ. Note that this reduction does not (and cannot) maintain the realization space of the system (the set of solutions). □

As a corollary of Theorem 4.1 we obtain:

**Corollary 4.2.** $\exists_< \mathbb{R} = \exists_= \mathbb{R}$.

The corollary allows us to simplify our notation and call our new complexity class simply $\exists\mathbb{R}$. Our computational world now looks as follows:

$$\mathbf{NP} \subseteq \exists\mathbb{R} \subseteq \mathbf{PSPACE}.$$

**Remark 4.3.** Following standard usage, we will say a problem is $\exists\mathbb{R}$-*hard*, if every problem in $\exists\mathbb{R}$ polynomial-time many-one reduces to it; it is $\exists\mathbb{R}$-complete, if it is $\exists\mathbb{R}$-hard and belongs to $\exists\mathbb{R}$. The complements of problems in $\exists\mathbb{R}$ can be said to belong to $\forall\mathbb{R}$, or *co*$\exists\mathbb{R}$ (used in [35]). ◁

Let QUAD be the computational problem asking whether a family of quadratic polynomials $f_i : \mathbb{R}^n \to \mathbb{R}$, $i \in [k]$, has a common zero, and let 4-FEAS be the special case of FEAS in which the polynomial has degree at most 4. The algebraic versions of these problems are known to be hard for $\mathbf{NP}_\mathbb{R}$, the analogue of **NP** in the Blum-Shub-Smale model [5, Section 5.4].

**Corollary 4.4.** QUAD *and* 4-FEAS *are* $\exists\mathbb{R}$*-complete.*

In [31, Lemma 3.9] it is shown (assuming Corollary 4.4) that QUAD remains $\exists\mathbb{R}$-complete if we ask for a common zero in the unit ball $B_n(0, 1)$.

*Proof.* Lemma 3.2 shows that ETR reduces to QUAD, which in turn reduces to 4-FEAS. Obviously, both problem belong to $\exists\mathbb{R}$. □

Ten Cate, Kolaitis and Othman [35] recently showed that $\exists\mathbb{R}$ is downward closed under **NP**-reductions, that is, if a problem **NP**-reduces to a problem in $\exists\mathbb{R}$, then it also belongs to $\exists\mathbb{R}$, where an **NP**-reduction is a many-one reduction computed by a non-deterministic polynomial time Turing machine. This allows the authors to show that the data exchange problem they are interested in lies in $\exists\mathbb{R}$. They show $\exists\mathbb{R}$-completeness by reducing the rectilinear crossing number problem to the data exchange problem.

Let us mention one more tool that may be useful in showing some problem lies in $\exists\mathbb{R}$; the first author used this result in [31] (without proof) to show that non-rigidity of linkages lies in $\exists\mathbb{R}$.

**Lemma 4.5.** *Let*

$$\Phi(\varepsilon, y) = (\exists x)\ \varphi(\varepsilon, x, y),$$

*with* $\varepsilon > 0$, $x \in \mathbb{R}^k$, $y \in \mathbb{R}^\ell$, *be such that* $\Phi(\varepsilon, y)$ *implies* $\Phi(\varepsilon', y)$ *for all* $\varepsilon' > \varepsilon$. *Then we can find a quantifier-free formula* $\psi(\varepsilon, x, y, z)$, *with* $z \in \mathbb{R}^m$, *of length at most* $|\varphi| + dm$, *where* $m = |\varphi| + 5$ *so that*

$$(\forall \varepsilon > 0)(\exists x)\ \varphi(\varepsilon, x, y)$$

*is equivalent to*

$$(\exists \varepsilon > 0, x, z)\ \psi(\varepsilon, x, y, z).$$

*Proof.* We assume $y \in \mathbb{R}^\ell$ is fixed and will drop it from the formulas (that is, we really prove the case $\ell = 0$).

Define two sets $A := \{(\varepsilon, x) \in \mathbb{R}^{k+1} : \varphi(\varepsilon, x), \varepsilon > 0\}$ and $B := \{0\} \times \mathbb{R}^k$. If $d(A, B) = 0$, then for every $\delta > 0$ there must be an $\varepsilon$ so that $\delta > \varepsilon > 0$ and $(\varepsilon, x) \in A$ for some $x$ which, by monotonicity, implies that $(\varepsilon', x) \in A$ for all $\varepsilon' > \varepsilon$. Since $\delta$ can be chosen arbitrarily small, this means that $(\forall \varepsilon > 0)(\exists x)\ \varphi(\varepsilon, x)$ is true.

Otherwise, $d(A, B) > 0$ and, by Corollary 3.8, $d(A, B) > 2^{-2^{L+5}}$. By construction, $d(A, B)$ is a lower bound on the infimum over all $\varepsilon > 0$ for which there is an $x$ so that $\varphi(x)$ is true; hence, $\varphi(\varepsilon, x)$ is false for all $x$ and all $\varepsilon < 2^{-2^{|\varphi|+5}}$, so $(\forall \varepsilon > 0)(\exists x)\ \varphi(\varepsilon, x)$ is false.

In other words, the truth of $(\forall \varepsilon > 0)(\exists x)\ \varphi(\varepsilon, x)$ is equivalent to $(\exists \varepsilon > 0)(\exists x)\ [\varphi(\varepsilon, x) \wedge \varepsilon < 2^{-2^{L+5}}]$. Using repeated squaring, $\varepsilon < 2^{-2^{|\varphi|+5}}$ can be expressed using a formula with at most $m = |\varphi| + 6$ variables $z$. Combining this formula with $\varphi(\varepsilon, x)$ we obtain $\psi(\varepsilon, x, z)$ so that the conclusion of the lemma holds. □

To see how this lemma can be useful, we give two examples, the first is from [31]. Let $\mathsf{ISO}$ be be the problem of deciding whether a point $x \in \mathbb{R}^n$ is an isolated zero of a family $(f_i)_{i\in[s]}$ of multivariate polynomials. Then $((f_i)_{i\in[s]}, x)$ is *not* an instance of $\mathsf{ISO}$, if $x$ is not a zero of $(f_i)_{i\in[s]}$ or $(\forall \varepsilon > 0)(\exists y)\ [\sum_{i\in[n]}(x_i - y_i)^2 < \varepsilon \wedge \bigwedge_{i\in[s]} f_i(y) = 0]$. By Lemma 4.5 the monotone all-quantification over $\varepsilon$ can be replaced with an existential quantifier, and we conclude that $\overline{\mathsf{ISO}}$ belongs to $\exists\mathbb{R}$; in other words $\mathsf{ISO}$ belongs to $\forall\mathbb{R}$.

Our second example is new:

**Lemma 4.6.** *Deciding whether two semi-algebraic sets have distance zero is* $\exists\mathbb{R}$*-complete.*

It is tempting to conjecture that the condition is equivalent to the closure of the two algebraic sets having a non-empty intersection, but that is not correct, as the earlier example $f(x,y) = x^2 + (1-xy)^2$ and $g(x,y) = 0$ shows.

*Proof.* $\exists\mathbb{R}$-hardness is an obvious reduction from FEAS: a multivariate polynomial $f : \mathbb{R}^n \to \mathbb{R}$ has a zero if and only if $\{(x,y) \in \mathbb{R}^{n+1} : y = f(x), \|x\| \leq R\}$ and $\{(x,0) \in \mathbb{R}^{n+1} : \|x\| \leq R\}$ have distance 0, where $R$ is an upper bound on a zero of $f$ (use Corollary 3.4 applied to $\{x \in \mathbb{R}^n : f(x) = 0\}$).

The interesting part is showing that the problem lies in $\exists\mathbb{R}$. Using Lemma 4.5 this is easy now, because we can express that two semi-algebraic sets defined by polynomials $f(x,y)$ and $g(x,y)$ (use Lemma 3.2) have distance 0 as $(\forall\varepsilon)(\exists x, y, x', y')[f(x,y) = 0 \wedge g(x',y') = 0 \wedge \|x - x'\| \leq \varepsilon]$. □

# 5 Fixed Points and the Nash Equilibrium

How hard is it to find a fixed point of a function? Consider a simple version of that problem called FIXED in which we ask whether a family $f$ of polynomials $f_i : \mathbb{R}^n \to \mathbb{R}$, $i \in [n]$, has a *fixed point*, that is an $x \in \mathbb{R}^n$ so that $f(x) = (f_1(x), \ldots, f_n(x)) = x$. FIXED is $\exists\mathbb{R}$-complete: Obviously, it is a special case of INEQ, and we can reduce FEAS to it, since $g : \mathbb{R}^n \to \mathbb{R}$ has a zero, if and only if the family of polynomials defined by $f_i(x) = g(x) + x_i$ has a fixed point.

In this section we consider continuous functions from a convex, compact set to itself. Such functions always have a fixed point by the Brouwer Fixed-Point Theorem, trivializing the question we asked for FIXED, but also giving a first hint that encoding is going to be harder with these functions. There are several computational questions that can be asked for this problem (see the detailed discussion by Etessami and Yannakakis [13]). We start with the decision version of the problem and discuss variants and the Nash Equilibrium problem in Section 5.2.

## 5.1 The Brouwer Fixed-Point Problem

Brouwer's fixed point theorem implies that a continuous function from a convex compact set to itself has a fixed point. We are interested in the computational complexity of deciding whether there is a fixed point of the function in a given neighborhood. To slightly simplify the argument, we work over the domain $B_n(x,r)$, the closed ball around $x \in \mathbb{R}^n$ of radius

$r$ in the $\ell^\infty$-metric; in other words, $B_n(x,r)$ is an $n$-dimensional box; for example, $B_n(0,1/2)$ is the unit cube centered at the origin.

There are several choices for what continuous functions we allow. Typically, functions defined using straight-line programs (a very compact representation of polynomials) or even extended straight-line programs (a compact representation of a class of functions that includes all rational functions) are allowed in this context (see [13], for example). Since we want to show that the problem is hard, we obtain a stronger result, the more limited our set of continuous functions, so we settle on the set of polynomials, represented explicitly, that is, in the form $f(x) = \sum_{i \in I} c_i x^i$, where $x \in \mathbb{R}^n$, $I \subseteq \mathbb{N}^n$, $c_i \in \mathbb{Q}$, and $x^i = (x_1^{i_1}, \ldots, x_n^{i_n})$. We even restrict $c_i$ to the set of values $\{-1, -1/2, 0, 1/2, 1\}$. We can now define the computational version of the Brouwer fixed-point problem:

**BROUWER**

**Given:** A polynomial family $f : B_n(0,1) \to B_n(0,1)$, represented explicitly, $x \in \mathbb{Q}^n$, $r \in \mathbb{Q}$.

**Question:** Does $f$ have a fixed point in $B_n(x,r)$?

Our goal is to show that BROUWER is $\exists\mathbb{R}$-hard. The strategy for the proof is simple: reduce the fixed-point problem FIXED to BROUWER. To encode FIXED, we need to scale the computations, since $f$ has to take values in $B_n(0,1)$. This is rather hard to achieve with explicitly represented polynomials, but becomes much easier if we use the (extended) straight-line representation. Consequently, the proof is in two parts: we show that (1) FIXED reduces to a fixed-point problem for extended straight-line programs (Theorem 5.3), and (2) explicitly represented polynomials have roughly the same power as extended straight-line programs when it comes to sets of fixed point (Lemma 5.1).

We start with the second part, for which we need a formal definition of the two variants of straight-line programs we mentioned. A *straight-line program (SLP)* is a sequence of assignments of the form $S_i := c, c \in \{-1, -1/2, 0, 1/2, 1\}$, $S_i := x_j, i \in [\ell], j \in [n]$ or $S_i := S_j \circ S_k$, where $1 \leq j, k < i \leq \ell$ and $\circ \in \{+, -, *\}$; $\ell$ is the *length* of the program.[6] We can think of the straight-line program as a succinct way of describing a multivariate

[6] We could allow arbitrary assignments $S_i := c$, where $c \in \mathbb{Q}$ or $c \in [-1,1] \cap \mathbb{Q}$, the following results would still be true if we redefine length in this case to include the number of bits needed to write down any rational constants used. We will see presently that this would not significantly change the model as far as fixed point computations are concerned: allowing division does not yield any additional computational power.

polynomial in variables $x_j, j \in [n]$, and we will sometimes write $S_i(x)$ if we want to emphasize the dependence of $S_i$ on the input variables $x$. A *straight-line program for a function* $f = (f_i)_{i\in[m]} : U \to V$, where $U \subseteq \mathbb{R}^n$, $V \subseteq \mathbb{R}^m$, is a straight-line program in which the first $n$ assignments are $S_i := x_i$, and the last $m$ assignments calculate $f_i(x_1, \ldots, x_n)$ so that $S_{\ell-m+i}(x_1, \ldots, x_n) = f_i(x_1, \ldots, x_n)$, for $i \in [m]$. In an *extended straight-line program (ESLP)* we also allow operations /, max, min, and $\sqrt[k]{\cdot}$. (The definition in this case implies that for inputs in $U$ no division by zero or even roots of negative numbers occur.)

The following lemma shows that ESLPs have no edge on explicitly represented polynomials with respect to capturing sets of fixed points: for each ESLP there is such a polynomial family that has essentially the same set of fixed points. We write $F_f$ for the set of fixed points of a function $f$ in its domain, and use **1** and **0** for the vector consisting of all ones or zeros (of appropriate dimension).

**Lemma 5.1.** *If $f : B_n(0,1) \to B_n(0,1)$ is a function given by an ESLP, then we can construct in polynomial time a polynomial family $g : B_{n'}(0,1) \to B_{n'}(0,1)$, $n' \geq n$, so that $F_f \cup \{\mathbf{1}\} = \pi_n(F_g)$, where $\pi_n : \mathbb{R}^{n'} \to \mathbb{R}^n$ projects a vector on its first $n$ coordinates. Moreover, we can ensure that $F_f \cap B_n(0,1/2) = \pi_n(F_g \cap B_{n'}(0,1/2))$.*

**Remark 5.2.** Two comments about the lemma: (*i*) There is nothing special about adding **1** as a fixed point when going from $f$ to $g$, the construction we will use could be adapted to add any point in $[-1,1]^{n'}$ describable by a polynomial. So one way to eliminate that point is by making this added fixed point a fixed point of $f$, however that would require the ability to find some (any) fixed point of $f$ and that we will probably not be able to do in polynomial time: Papadimitriou showed that this problem is **PPAD**-complete [26]. It seems possible that there is a different construction which obviates the need to add an extra point as fixed point. One way to approach the problem would be to start with the stronger assumption that there is an SLP computing $f$. (*ii*) As it is, Lemma 5.1 tells us that finding an arbitrary fixed point of a function $f : B_n(0,1) \to B_n(0,1)$ specified by an ESLP can be reduced to finding at least two (arbitrary) fixed points of a polynomial family $g : B_{n'}(0,1) \to B_{n'}(0,1)$. ◁

*Proof of Lemma 5.1.* Let $(S_i)_{i\in[\ell]}$ be the ESLP computing $f$. The first $n$ instructions have the form $S_i := x_i$, $i \in [n]$, and the last $n$ variables, $S_{\ell-n+1}, \ldots, S_\ell$ contain the outputs. We can assume that divisions are of the form $S_i := 1/S_k$; moreover, since $\max\{x,y\} = (x + y + |x - y|)/2$,

$\min\{x,y\} = x + y - \max\{x,y\}$ and $|x| = \sqrt{x^2}$, we can assume that the ESLP does not contain max or min (Etessami and Yannakakis use the same trick in [13]). Finally, we replace any instruction $S_i := \sqrt[k]{S_j}$ for even $k$ with two instructions: $A := \sqrt[2k]{S_j}$, $S_i := A * A$; here $A$ is a new variable of the ESLP we insert just before $S_i$ (and which is only used to calculate $S_i$). This modified ESLP will calculate $S_i$ (and thus the rest of the program) correctly as the positive $k$th root of $S_j$, independently of whether $A = \sqrt[2k]{S_j}$ evaluates to the positive or negative $2k$-th root of $S_j$.

We can consider each $S_i$ as a function $S_i(x_1, \ldots, x_n)$ from $\mathbb{R}^n$ to $\mathbb{R}$; we know that every $S_i$ is well-defined (no divisions by zero or even roots of negative arguments), but while the $S_i$ calculating the output are restricted to values in $[-1, 1]$, the intermediate values can be large; however, since the input set is compact, there is an $M$ so that $S_i(x_1, \ldots, x_n) \leq M/2$ for all $i \in [\ell]$ and $(x_1, \ldots, x_n) \in [-1, 1]^n$. Consider $\mathcal{S} := \{(S_1(x), \ldots, S_\ell(x)) : x \in [-1, 1]^n\}$. Then $\mathcal{S}$ is a semi-algebraic set (all instructions can be rewritten as polynomial (in)equalities, e.g. $S_i := \sqrt[2k]{S_j}$ becomes $S_i \geq 0 \wedge S_i^{2k} - S_j = 0$ and $S_i := \sqrt[2k+1]{S_j}$ becomes $S_i^{2k+1} - S_j = 0$); hence, by Corollary 3.7, we can choose $M = 2^{2^{\log(c\ell)\rceil+5}}$ (for some $c > 0$ which bounds the number of symbols needed to express a straight-line instruction).

The polynomial family $g$ will have $n' = 3n + 2\ell + 1 + m$ scalar variables grouped as $x = (x_1, \ldots, x_n)$, $y = (y_0, y_1, \ldots, y_\ell)$, $y' = (y'_1, \ldots, y'_\ell)$, $z = (z_1, \ldots, z_n)$, $z' = (z'_1, \ldots, z'_n)$, and $u' = (u_1, \ldots, u_m)$, where $m = \lceil \log(c\ell) \rceil + 5$. For a fixed point $g(x, y, y', z, z', u) = (x, y, y', z, z', u)$ we will ensure that

- $u_i = 2^{-2^i}$ for $i \in [m]$,
- $y'_i = 0$, and $y_i = S_i/M$ for $i \in [\ell]$,
- $z'_i = 0$, and $z_i = S_{\ell-n+i}$ for $i \in [n]$,

unless $y_0 = 1$, in which case we guarantee that $x = \mathbf{1}$. This means that (as long as $y_0 \neq 1$), $u_m = 1/M$, the $y_i$ simulate the calculations of the straight-line program scaled by the factor $1/M$, and the $z_i$ contain the actual output $f(x)$. To simplify the presentation, we label the components of $g$ by their variables, so we write $g_{x_i}$ or $g_{u_j}$ rather than using a uniform integer labeling $g_i$.

We start by defining $g_{x_i}$, $g_{z_i}$, $g_{z'_i}$, and $g_{u_i}$. Let $p(x) := 1 - (x - (1/4))^2 * (1/4)$ and $q(x) := x^2 * (1/16)$, where $(1/4)$ is short for $1/2 * 1/2$ and similarly for $(1/16)$.

- $g_{x_i} = z_i$ for $i \in [n]$,

- $g_{z_i} = p(y_0) * z_i + 1 - p(y_0)$ for $i \in [n]$,
- $g_{z'_i} = q(y_{\ell-n+i} - z_i * u_m)$ for $i \in [n]$,
- $g_{u_1} = 1/2$, and $g_{u_{i+1}} = u_i^2$ for $i \in [m-1]$.

Based on the instructions in the ESLP for $f$ we construct the polynomials $g_{y'_i}$ for $i \in [\ell]$:

$$\begin{aligned}
S_i := c &\rightarrow g_{y'_i} := q(y_i - c * u_m) \\
S_i := x_j &\rightarrow g_{y'_i} := q(y_i - x_j * u_m) \\
S_i := S_j + S_k &\rightarrow g_{y'_i} := q(y_i - (y_j + y_k)) \\
S_i := S_j * S_k &\rightarrow g_{y'_i} := q(y_i * u_m - (y_j * y_k)) \\
S_i := 1/S_j &\rightarrow g_{y'_i} := q(y_i * y_j - 1 * u_m^2) \\
S_i := \sqrt[k]{S_j} &\rightarrow g_{y'_i} := q(y_i^k - y_j * u_m^{k-1}).
\end{aligned}$$

Finally, let

- $g_{y_0} = 1 - (1 - y_0) * \left(1 - \sum_{i\in[\ell]}((y'_i)^2 + (z'_i)^2) * (1/2)^{\lceil \log 2\ell \rceil}\right)$, and
- $g_{y_i} = y_i$, for $i \in [\ell]$.

This completes the definition of $g$ which clearly is a polynomial family represented explicitly (the terms in $g_{y_0}$ can be multiplied out easily). We need to show that $g$ also is a function from $[-1,1]^{3n+\ell+1+m}$ to $[-1,1]^{3n+\ell+1+m}$. This is obvious for $g_{x_i}$, $g_{y_i}$, and $g_{u_i}$. Note that $g_{y'_i}$ and $g_{z'_i}$ take on values in $[0,1]$ by choice of $q$: the terms to which $q$ is applied all lie in the range $[-3,3]$, since $y_i$, $x_j$, $z_k$, and $u_m$ all lie in $[-1,1]$, so by applying $q$ we obtain numbers in $[0,1]$. Finally, $g_{z_i} \in [-1,1]$, since $g_{z_i} = p(y_0) * z_i + 1 - p(y_0) \leq p(y_0) + 1 - p(y_0) = 1$ and $g_{z_i} \geq 1 - 2p(y_0) \geq -1$, and $g_{y_0} \in [-1,1]$, since $0 \leq \sum_{i\in[\ell]}(y'_i)^2 + (z'_i)^2 \leq 2\ell \leq 2^{\lceil \log 2\ell \rceil}$.

We next show that $\pi_n(F_g) \subseteq F_f \cup \{\mathbf{1}\}$. To that end, let $(x, y, y', z, z', u)$ be an arbitrary fixed point of $g$, that is, $g(x, y, y', z, z', u) = (x, y, y', z, z', u)$. From the definition of $g_{u_i}$, for $i \in [m]$, we conclude that $u_m = 1/M$.

If $y_0 = 1$, then $p(y_0) < 1$. Since $z_i = g_{z_i} = p(y_0) * z_i + 1 - p(y_0)$, we have $z_i(1 - p(y_0)) = 1 - p(y_0)$ and thus $z_i = 1$ for $i \in [n]$; but then $x_i = g_{x_i} = z_i = 1$ for $i \in [n]$, so $x = \mathbf{1}$.

If, on the other hand, $y_0 \neq 1$, then because of $y_0 = g_{y_0} = 1 - (1 - y_0) * (1 - (\sum_{i\in[\ell]}(y'_i)^2 + (z'_i)^2) * (1/2)^{\lceil \log 2\ell \rceil})$ we have $1 - y_0 = (1 - y_0) * (1 - (\sum_{i\in[\ell]}(y'_i)^2 + (z'_i)^2) * (1/2)^{\lceil \log 2\ell \rceil})$ and thus $\sum_{i\in[\ell]}(y'_i)^2 + (z'_i)^2 = 0$ which

implies $y'_i = 0$ for all $i \in [\ell]$ and $z'_i = 0$ for all $i \in [n]$. To argue that $y_i = S_i/M$, we distinguish cases based on the instruction defining $S_i$; we use that $q(x) = 0$ implies that $x = 0$.

- $S_i := c$: since $y'_i = 0$ we get that $q(y_i - c * u_m) = 0$ and thus $y_i = c * u_m = c/M$,
- $S_i := x_j$: since $y'_i = 0$ we get that $q(y_i - x_j * u_m) = 0$ and thus $y_i = x_j * u_m = x_j/M$, or, in other words, $M * y_i = x_j$,
- $S_i := S_j + S_k$; we get $y_i = y_j + y_k$,
- $S_i := S_j * S_k$ we get $y_i/M = (y_j * y_k)$, so $M * y_i = (M * y_j) * (M * y_k)$,
- $S_i := 1/S_j$; we get $y_i * y_j = 1/M^2$, so $M * y_i = 1/(M * y_j)$,
- $S_i := \sqrt[k]{S_j}$; we get $y_i^k = y_j/M^{k-1}$, so $(M * y_i)^k = M * y_j$ (recall that if $k$ is even we have ensured that $S_j \geq 0$).

This is enough to show inductively that $y_i = S_i/M$ in the first five cases and $|y_i| = |S_i|/M$ in the last case (which is sufficient as we saw earlier). A similar argument about the $g_{z'_i}$ with $z'_i = 0$ shows that $y_{\ell-n+i} = z_i/M$, so $z_i = M * y_{\ell-n+i} = S_{\ell-n+i}$; now, since $x_i = g_{x_i} = z_i$, this shows that the fixed point of $g$, if projected on its first $n$ coordinates $(x_1, \ldots, x_n)$, is a fixed point of $f$. This completes the proof that $\pi_n(F_g) \subseteq F_f \cup \{\mathbf{1}\}$ which, in turn, implies that $\pi_n(F_g \cap B_{n'}(0, 1/2)) \subseteq \pi_n(F_g) \cap B_n(0, 1/2) \subseteq F_f \cap B_n(0, 1/2)$, where $n' = 3n + 2\ell + 1 + m$.

To see that $F_f \cup \{\mathbf{1}\} \subseteq \pi_n(F_g)$, let $x$ be a fixed point of $f$, that is, $f(x) = x$. We set $u_i = 2^{-2^i}$ satisfying $g_{u_i} = u_i$. Let $y_0 = 1/4$, so that $p(y_0) = 1$, and thus $g_{z_i} = p(y_0) * z_i + 1 - p(y_0) = z_i$ for any choice of $z_i \in [-1, 1]$. Then letting $y'_i = 0$, and $y_i = S_i(x)/M$ for $i \in [\ell]$, and $z'_i = 0$, and $z_i = S_{\ell-n+i}(x)$ for $i \in [n]$, satisfies the remaining clauses of $g$, showing that $(x, y, y', z, z', u)$ is a fixed point of $g$ and $F_f \subseteq \pi_n(F_g)$. Moreover, for the same values of $y$, $y'$, $z'$, and $u$, we see that $(\mathbf{1}, y, y', \mathbf{1}, z', u)$ is a fixed point of $g$ as well (recall that there is no restriction on the $z_i$, since $g(z_i) = z_i$ for the particular value of $y_0$ we chose, and $x = z$ is all that is required to satisfy the $g_{x_i}$), showing that $\mathbf{1} \in \pi_n(F_g)$.

We note that something slightly stronger than $F_f \subseteq \pi_n(F_g)$ is true, since we can bound by $1/2$ all the intermediate variables for the fixed point $(x, y, y', z, z', u)$ of $g$ corresponding to the fixed point $x$ of $f$: We have $y_0 = 1/4$, $|y_i| = |S_i(x)/M| \leq 1/2$ (by choice of $M$), $|u_i| \leq 1/2$, $y'_i = 0$, and $z'_i = 0$. So $(x, y, y', z, z', u) \in F_f \times B_{2\ell+1}(0, 1/2) \times F_f \times B_{n+m}(0, 1/2)$. In particular, $F_f \cap B_n(0, 1/2) \subseteq \pi_n(F_g \cap B_{n'}(0, 1/2)$, where $n' = 3n + 2\ell + 1 + m$.

In summary, $\pi_n(F_g) = F_f \cup \{\mathbf{1}\}$, and $F_f \cap B_n(0, 1/2) = \pi_n(F_g \cap B_{n'}(0, 1/2))$, concluding the proof of the lemma. □

With Lemma 5.1 it is now easy to show that BROUWER is $\exists\mathbb{R}$-hard.

**Theorem 5.3.** *Deciding* BROUWER *is $\exists\mathbb{R}$-complete, even for $x = 0$ and $r = 1/2$.*

The theorem remains true for any other appropriate choice of $x$ and $r$. For fixed dimension, e.g. $n = 1$ or $n = 2$, BROUWER can be decided in $\mathbf{P}$ using quantifier elimination for the fixed number of quantifiers.

*Proof.* The problem is easily seen to lie in $\exists\mathbb{R}$ (this remains true even if $f$ is specified by an SLP or an ESLP). We saw earlier that deciding whether a family of multivariate polynomials $f = (f_i)_{i\in[n]} : \mathbb{R}^n \to \mathbb{R}^n$ has a fixed point is hard for $\exists\mathbb{R}$; since these polynomials are given explicitly, it is easy to construct an SLP $S$ computing $f$. By Corollary 3.4 if $f$ has a fixed point, there has to be a fixed point at distance less than $R/2 = 2^{(c\ell)^{8n}}$ from 0, where $\ell$ is the length of $S$, and $c$ is a fixed constant. Now $f$ maps $B_n(0, R)$ to $B_n(0, R')$, where $R' = \lceil R^{2^\ell} \rceil \leq \lceil 2^{2^{c'\ell n}} \rceil$ (each coordinate can be at most squared in each of the at most $\ell$ steps of the computation; $c'$ only depends on $c$, so it is a fixed constant). Let $g$ be the continuous map that is the identity on $B_n(0, R/2)$ and bijectively maps $B_n(0, R') - B_n(0, R/2)$ to $B_n(0, R) - B_n(0, R/2)$ defined component-wise by:

$$g_i(x) = \begin{cases} x_i & \text{if } x_i \in B_1(0, R/2) \\ \operatorname{sgn}(x_i)\frac{R}{2}\left(\frac{|x_i|-(R/2)}{R'-R/2} + 1\right) & \text{if } x_i \in B_1(0, R') - B_1(0, R/2) \end{cases}$$

for $i \in [n]$, where $\operatorname{sgn}(x)$ is the sign function. Then $g \circ f$ maps $B_n(0, R)$ to $B_n(0, R)$; moreover, any fixed point of $f$ in $B_n(0, R/2)$ is still a fixed point of $g \circ f$ in $B_n(0, R/2)$ and vice versa. Finally, let $h$ be a scaling by $R$, that is $h$ is a continuous bijection between $B_n(0, R)$ and $B_n(0, 1)$. Thus $h \circ g \circ f \circ h^{-1} : B_n(0, 1) \to B_n(0, 1)$ has a fixed point in $B_n(0, 1/2)$ if and only if $f$ has a fixed point (in $\mathbb{R}^n$).

Now, there will not, in general, be an SLP computing $h \circ g \circ f \circ h^{-1}$ since such an SLP would require division and case distinction; however, it is easy to see that there is an ESLP: this is clear for $h$ and most of $g$; the only interesting question is how to perform the case distinction, but that can be

done using max and min:

$$g_i(x) = \max(0, \min(x_i, R/2) + \frac{R}{2}\max(0, \frac{x_i - R/2}{R' - R/2})) \\ - \max(0, \min(-x_i, R/2) + \frac{R}{2}\max(0, \frac{-x_i - R/2}{R' - R/2})).$$

Finally, ESLPs are closed under composition of functions, so we can conclude that there is an ESLP for $h \circ g \circ f \circ h^{-1}$ and thus, by Lemma 5.1 an explicitly represented polynomial family $f' : B_{n'}(0,1) \to B_{n'}(0,1)$ so that $F_{h\circ g\circ f\circ h^{-1}} \cup \{\mathbf{1}\} = \pi_n(F_{f'})$. Moreover, the lemma allows us to conclude that

$F_{h\circ g\circ f\circ h^{-1}} \cap B_n(0,1/2) = \pi_n(F_{f'} \cap B_{n'}(0,1/2))$.

Now $f$ has a fixed point if and only if $h \circ g \circ f \circ h^{-1}$ has a fixed point in $B_n(0,1/2)$ if and only if $f'$ has a fixed point in $B_{n'}(0,1/2)$, which is the BROUWER problem for the explicitly represented polynomial family $f'$. □

## 5.2 The Nash Equilibrium

Etessami and Yannakakis [13] studied in depth the search versions of fixed-point problems and Nash equilibriums[7]: Suppose we are given a function $f : B_n(0,1) \to B_n(0,1)$ via an ESLP. How hard is it to find some (any) fixed point of $f$? This, of course, is a problem over real numbers, but one can turn it into a discrete problem as follows. For any input $r \in \mathbb{Q}^n$ we are allowed to ask questions of the type $x\Theta r$, where $\Theta$ is one of $\{\leq, \geq, <, >\}^n$, a vector of comparison operators. If $x\Theta r$ for all fixed points $x$ of $f$ the answer has to be "yes", if $x\Theta r$ is false for all fixed points of $f$ the answer has to be "no"; otherwise, the answer can be either "yes" or "no". Etessami and Yannakakis call this the *decision problem* (in their terminology, BROUWER is an existence problem, not a decision problem). The class of all such fixed-point decision problems they call $\mathbf{FIXP_d}$. $\mathbf{FIXP_d}$ is rather robust, for example it is not affected by changing the domain of the function (to a cube, or a sphere, say). Etessami and Yannakakis [13, Theorem 4.7] also show that $\mathbf{FIXP_d}$ remains the same if ESLPs are restricted to $\{+, *, \max\}$.[8]

Moreover, $\mathbf{FIXP_d}$ has natural complete problems, including, among several others, the decision (in Etessami and Yannakakis's terminology) versions of BROUWER and the Nash equilibrium problem for 3 players. Clearly,

[7] We refer the reader to their paper—specifically their Section 2.2—for all terminology and definitions related to equilibria used in this section.

[8] The proof uses Nash equilibria, compare Lemma 5.1 which gets rid of max as well, but adds a fixed point.

$\mathbf{FIXP_d} \subseteq \exists\mathbb{R}$, and Etessami and Yannakakis show that PosSLP reduces to any $\mathbf{FIXP_d}$-complete problem, where PosSLP is the problem of deciding whether a given SLP computes a positive number; currently this is the best known lower bound on $\mathbf{FIXP_d}$ (in turn, the best-known upper bound on PosSLP is the counting hierarchy, due to a result by Allender, Bürgisser, Kjeldgaard-Pedersen, and Miltersen [1]). Etessami and Yannakakis point out that it is unlikely that any $\mathbf{FIXP_d}$-problem is **NP**-hard, since in that case it is also **coNP**-hard, which would imply $\mathbf{coNP} \subseteq \exists\mathbb{R}$, which is not impossible, but seems counterintuitive. Similarly, $\mathbf{FIXP_d} = \exists\mathbb{R}$ would imply that $\exists\mathbb{R}$ is closed under complement, which again appears unlikely.

Here we consider decision versions of the Nash equilibrium problem in which we ask whether there is a Nash equilibrium in a given ball $B_n(x, r)$. Etessami and Yannakakis's core result works in this setting as well; we summarize it in the following lemma:

**Lemma 5.4** (Etessami, Yannakakis [13, Claim 2 in Theorem 4.3])**.** *Given a function $f : B_n(0,1) \to B_n(0,1)$ specified by an SLP of length $\ell$, one can construct in polynomial time a 3-player game $G$ and compute an integer $N = O(\ell)$ so that*

- *if $x^*$ is a fixed point of $f$ then there is a Nash equilibrium $z = (z_1, z_2, z_3)$ of $\Gamma$ so that $z_1[1:n] = x^*/N$,*
- *if $z = (z_1, z_2, z_3)$ is a Nash equilibrium of $\Gamma$, then $x^* = Nz_1[1:n]$ is a fixed point of $f$.*

By Theorem 5.3, BROUWER is $\exists\mathbb{R}$-hard, and Lemma 5.4 shows that there is a reduction from BROUWER to the Nash equilibrium problem, so the following corollary is immediate now.

**Corollary 5.5.** *Deciding whether a 3-player game $\Gamma$ has a Nash equilibrium in $B_n(x, r)$ for $x \in \mathbb{Q}^n$, $r \in \mathbb{Q}$ is $\exists\mathbb{R}$-complete even for $x = 0$.*

In the corollary, we can take $r = 1/(2N)$, where $N$ is as in Lemma 5.4. More precisely, given an instance of BROUWER with $r = 1/2$ we use Lemma 5.4 to construct a 3-player game $G$ and $N$ and then use $G$ and $r = 1/(2N)$ as an instance of the "Nash equilibrium in a ball" problem.

**Remark 5.6.** Datta showed the universality of 3-player totally mixed Nash equilibria [11]; algebraically this is a stronger result, since it shows that arbitrary semi-algebraic sets can be encoded as Nash equilibria; however, the reduction is not polynomial time, since some players in her game use

$\Omega(d^n)$ pure strategies, where $d$ is the highest power of any variable in the polynomial equations encoding the semi-algebraic sets, see [11, Theorem 2]. It may be an interesting open problem whether Datta's universality theorem can be improved to an efficient, that is, polynomial-time, reduction. ◁

Etessami and Yannakakis have also given a reduction from BROUWER (with max) to the exchange equilibrium problem (see [13, Proposition 4.4] for details); starting with our more restrictive version of BROUWER, we can conclude that the problem remains hard if the ESLP is restricted further.

**Corollary 5.7.** *The exchange equilibrium problem with an excess demand function given by an ESLP is $\exists\mathbb{R}$-complete. Indeed, this remains true if the ESLP is restricted to division (that is, no roots,* max *or* min *operations are allowed).*

# A Proof of Corollary 3.4

Let $S = \{x \in \mathbb{R}^n : \varphi(x)\}$ be the semi-algebraic set defined by $\varphi$ of complexity at most $L$. If $S = \emptyset$, there is nothing to show, so we can assume that $S \neq \emptyset$. By Lemma 3.1 there is a conjunction $\psi = \bigwedge_{i=1}^{\ell} s_i \triangle_i t_i$ with $\triangle_i \in \{<, =\}$ so that $S' = \{x \in \mathbb{R}^n : \psi(x)\}$ is a non-empty subset of $S$ and $|\psi| \leq |\varphi| \leq L$. In particular, $\ell < L$. Now $S' = \{x \in \mathbb{R}^n : s_i - t_i \triangle_i 0\}$. Let $f_i := s_i - t_i$, and $s = \ell$. Then $s < L$, each $f_i$ has degree $d$ at most $L - 2$ (we need two symbols for $\triangle_i$ and 0), and the bitlength of each coefficient of $f_i$ is bounded by $L - 1$, so $\tau < L$ (these are wildly generous bounds). Hence, we can apply Theorem 3.3 to conclude that $S'$, and therefore $S$ contains a point at distance at most $R$ from the origin if it is non-empty. We are left with the estimate of $R$. Let us first simplify the expression for $R$:

$$\begin{aligned} R &\leq ((4DN^2)2^{2N(\tau'+b(2N)+b(2DN+1))})^{1/2} \\ &\leq 2DN2^{N(\tau'+b(2N)+b(2DN+1))} \\ &\leq 2^{b(N)+b(2D)+N(\tau'+b(2N)+b(2DN+1))}. \end{aligned}$$

We know that $d' < 2L$ (using $L \geq 4$); then $D \leq nd' < 2nL$, and $2D + 1 \leq 4nL$. With this $b(2D+1) \leq b(4nL) \leq 3 + \log(nL) \leq nL + 3$ (we're using $b(x) \leq \log(x) + 1$). Now $N \leq (d')^n \leq (2L)^n$, so $b(N) = b((2L)^n) \leq 1 + n\log(2L) \leq nL + 1$ (for $L \geq 4$), and $b(2N) \leq nL + 2$. We can now evaluate the $\tau$-values: $\tau_0 \leq 2L + (n+1)b(L) + b(4L) \leq 2L + (n+2)\log L + (n+4) \leq 5nL$ (for $L \geq 4$); with that, $\tau_1 \leq D(\tau_0 + 4b(2D+1) + b(N)) \leq$

$2nL(5nL + 4(nL + 3) + (nL + 1)) \leq 27n^2L^2$, and $\tau_2 \leq \tau_1 + 2n(b(N) + b(n)) \leq \tau_1 + 2n^2L^2 \leq 29n^2L^2$, $\tau' \leq N(\tau_2 + (nL + 1) + 2(nL + 3) + 1) \leq (2L)^n(31n^2L^2 + 8) \leq (2L)^n 32n^2L^2 \leq 32n^2L^{3n}$. This allows us to evaluate the expression $\tau' + b(2N) + b(2DN + 1) \leq \tau' + 1 + 2b(N) + b(2D + 1) \leq 32n^2L^{3n} + 3nL + 5 \leq 35n^2L^{3n}$. Finally,

$$\begin{aligned} R &\leq 2^{b(N)+b(2D)+N(\tau'+b(2N)+b(2DN+1))} \\ &\leq 2^{(nL+1)+(nL+3)+N(35n^2L^{3n})} \\ &\leq 2^{(2nL+4)+(2L)^n(35n^2L^{3n})} \\ &\leq 2^{35n^2L^{5n}} \\ &\leq 2^{L^{8n}}, \end{aligned}$$

which is what we had to show.


### Acknowledgments

We'd like to thank Dejan Jovanović, Nicolai Vorobjov, Leonardo De Moura, and Jiří Matoušek for useful comments and suggestions on an earlier version of this paper. Finally, we are grateful for detailed comments and improvements received from several referees.


## References


[1] Eric Allender, Peter Bürgisser, Johan Kjeldgaard-Pedersen, and Peter Bro Miltersen. On the complexity of numerical analysis. *SIAM J. Comput.*, 38(5):1987–2006, 2008.

[2] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in real algebraic geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, second edition, 2006.

[3] Saugata Basu and Marie-Françoise Roy. Bounding the radii of balls meeting every connected component of semi-algebraic sets. *J. Symbolic Comput.*, 45(12):1270–1279, 2010.

[4] Daniel Bienstock. Some provably hard crossing number problems. *Discrete Comput. Geom.*, 6(5):443–459, 1991.

[5] Lenore Blum, Felipe Cucker, Michael Shub, and Steve Smale. *Complexity and real computation*. Springer-Verlag, New York, 1998. With a foreword by Richard M. Karp.

[6] 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. *Bull. Amer. Math. Soc. (N.S.)*, 21(1):1–46, 1989.

[7] Peter Bürgisser and Felipe Cucker. Counting complexity classes for numeric computations. II. Algebraic and semialgebraic sets. *J. Complexity*, 22(2):147–191, 2006.

[8] Jonathan F. Buss, Gudmund S. Frandsen, and Jeffrey O. Shallit. The computational complexity of some problems of linear algebra. *J. Comput. System Sci.*, 58(3):572–596, 1999.

[9] John Canny. Some algebraic and geometric computations in pspace. In *STOC '88: Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 460–469, New York, NY, USA, 1988. ACM.

[10] Jean Cardinal and Vincent Kusters. The complexity of simultaneous geometric graph embedding. *CoRR*, abs/1302.7127, 2013.

[11] Ruchira S. Datta. Universality of Nash equilibria. *Math. Oper. Res.*, 28(3):424–432, 2003.

[12] Ernest Davis, Nicholas Mark Gotts, and Anthony G. Cohn. Constraint networks of topological relations and convexity. *Constraints*, 4(3):241–280, 1999.

[13] Kousha Etessami and Mihalis Yannakakis. On the complexity of Nash equilibria and other fixed points. *SIAM J. Comput.*, 39(6):2531–2597, 2010.

[14] D. Yu. Grigor'ev and N. N. Vorobjov. Solving systems of polynomial inequalities in subexponential time. *J. Symb. Comput.*, 5(1-2):37–64, 1988.

[15] Hoon Hong. Comparison of several decision algorithms for the existential theory of the reals. Technical Report 91-41, RISC-Linz, Johannes Kepler University, Linz, Austria, 1991.

[16] Gabriela Jeronimo and Daniel Perrucci. On the minimum of a positive polynomial over the standard simplex. *J. Symbolic Comput.*, 45(4):434–442, 2010.

[17] Gabriela Jeronimo, Daniel Perrucci, and Elias Tsigaridas. On the Minimum of a Polynomial Function on a Basic Closed Semialgebraic Set and Applications. *SIAM J. Optim.*, 23(1):241–255, 2013.

[18] Ross J. Kang and Tobias Müller. Sphere and dot product representations of graphs. *Discrete Comput. Geom.*, 47(3):548–568, 2012.

[19] Ross J. Kang and Tobias Müller. Arrangements of pseudocircles and circles. Unpublished manuscript, 2013.

[20] Jan Kratochvíl and Jiří Matoušek. Intersection graphs of segments. *J. Combin. Theory Ser. B*, 62(2):289–315, 1994.

[21] Daniel Kroening and Ofer Strichman. *Decision procedures*. Texts in Theoretical Computer Science. An EATCS Series. Springer-Verlag, Berlin, 2008. An algorithmic point of view, With a foreword by Randal E. Bryant.

[22] Jan Kynčl. Simple realizability of complete abstract topological graphs in P. *Discrete Comput. Geom.*, 45(3):383–399, 2011.

[23] Bhubaneswar Mishra. Computational real algebraic geometry. In *Handbook of discrete and computational geometry*, CRC Press Ser. Discrete Math. Appl., pages 537–556. CRC, Boca Raton, FL, 1997.

[24] N. E. Mnëv. The universality theorems on the classification problem of configuration varieties and convex polytopes varieties. In *Topology and geometry—Rohlin Seminar*, volume 1346 of *Lecture Notes in Math.*, pages 527–543. Springer, Berlin, 1988.

[25] Christos H. Papadimitriou. *Computational complexity*. Addison-Wesley Publishing Company, Reading, MA, 1994.

[26] Christos H. Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. System Sci.*, 48(3):498–532, 1994. 31st Annual Symposium on Foundations of Computer Science (FOCS) (St. Louis, MO, 1990).

[27] Jürgen Richter-Gebert. *Realization spaces of polytopes*, volume 1643 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1996.

[28] Jürgen Richter-Gebert and Günter M. Ziegler. Realization spaces of 4-polytopes are universal. *Bull. Amer. Math. Soc. (N.S.)*, 32(4):403–412, 1995.

[29] Marcus Schaefer. The real logic of drawing graphs. Unpublished Manuscript.

[30] Marcus Schaefer. Complexity of some geometric and topological problems. In David Eppstein and Emden R. Gansner, editors, *Graph Drawing*, volume 5849 of *Lecture Notes in Computer Science*, pages 334–344. Springer, 2009.

[31] Marcus Schaefer. Realizability of graphs and linkages. In János Pach, editor, *Thirty Essays on Geometric Graph Theory*, pages 461–482. Springer, 2012.

[32] Peter W. Shor. Stretchability of pseudolines is NP-hard. In *Applied geometry and discrete mathematics*, volume 4 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 531–554. Amer. Math. Soc., Providence, RI, 1991.

[33] Michael Sipser. *Introduction to the Theory of Computation*. Course Technology, 2nd edition, 2005.

[34] Paul J. Tanenbaum, Michael T. Goodrich, and Edward R. Scheinerman. Characterization and recognition of point-halfspace and related orders (preliminary version). In *Graph drawing (Princeton, NJ, 1994)*, volume 894 of *Lecture Notes in Comput. Sci.*, pages 234–245. Springer, Berlin, 1995.

[35] Balder ten Cate, Phokion G. Kolaitis, and Walied Othman. Data exchange with arithmetic operations. In Giovanna Guerrini and Norman W. Paton, editors, *EDBT*, pages 537–548. ACM, 2013.

[36] E. Triesch. A note on a theorem of Blum, Shub, and Smale. *J. Complexity*, 6(2):166–169, 1990.

[37] G. S. Tseitin. On the complexity of derivation in propositional logic. In Graham Wrightson Jörg Siekmann, editor, *Automation of Reasoning: Classical Papers on Computational Logic 1967–1970*, volume 2, pages 466–483. Springer, 2009.

[38] N. N. Vorob′ev. Estimates of real roots of a system of algebraic equations. *Zap. Nauchn. Sem. Leningrad. Otdel. Mat. Inst. Steklov. (LOMI)*, 137:7–19, 1984. Theory of the complexity of computations, II.

[39] Wikipedia. Existential theory of the reals, 2012. (Online; accessed 12-September-2015).