Electronic Colloquium on Computational Complexity, Report No. 51 (2018)

DERANDOMIZING DYNAMIC PROGRAMMING AND BEYOND

STASYS JUKNA*

Abstract. We consider *probabilistic* circuits working over the real numbers, and using arbitrary semialgebraic functions of bounded description complexity as gates. We show that such circuits can be simulated by *deterministic* circuits with an only polynomial blowup in size. An algorithmic consequence is that randomization cannot substantially speed up dynamic programming. In arithmetic circuits, randomization cannot spare even one single gate.

Key words: Derandomization; dynamic programming; tropical circuits; semialgebraic functions; VC dimension
AMS [2000] Primary 68W20 (randomized algorithms); Secondary 68Q17 (computational difficulty of problems)

1. Introduction

A classical result of Adleman [1], extended to the case of *two*-sided error probability by Bennett and Gill [7], shows that randomness is useless in *boolean* circuits: if a boolean function f of n variables can be computed by a *probabilistic* boolean circuit of polynomial in n size, then f can be also computed by a *deterministic* boolean circuit of polynomial in n size. In the computational complexity literature, this result is stated shortly as[1] $\mathsf{BPP} \subseteq \mathsf{P/poly}$. That is, probabilistic boolean circuits *can* be derandomized. But the proof of Adleman's theorem crucially used the fact that the domain $R = \{0, 1\}$ of boolean circuits is *finite*: the proof then follows from a simple *finite majority rule* (see Section 3), which itself is an easy application of Chernoff's bound.

In this paper, we are interested in the derandomization of circuits working over *infinite* (and even uncountable) domains R such as $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$, and using any semialgebraic functions of low description complexity as gates. A function $f : R^n \to R$ is *semialgebraic* if its graph $\{(x, y) : y = f(x)\}$ can be obtained by finitely many unions and intersections of sets defined by a polynomial equality or strict inequality. Simplest examples of such functions are all arithmetic operations $+, -, \times, \div$, tropical operations $\min, \max$, the signum, selection, if-then-else operations, and many more. Our primary interest in such circuits comes from them being able to simulate dynamic programming algorithms (DP algorithms).

A *probabilistic* circuit is a deterministic circuit which is allowed to use additional input variables, each being a *random variable* taking real values. The probability distributions of these random variables can be arbitrary, so that our derandomization results will be distribution-independent. Such a circuit *computes* a given function $f : R^n \to R$ if, for every input $x \in R^n$, the circuit outputs the correct value $f(x)$ with probability at least $2/3$. (There is nothing "magical" in the choice of this threshold value $2/3$: we do this only for definiteness. One can take any constant larger than $1/2$: since we ignore multiplicative constants, all results will hold also then.)

Our motivating question was: does $\mathsf{BPP} \subseteq \mathsf{P/poly}$ also holds for DP algorithms? That is, if an optimization problem $f : R^n \to R$ can be solved by a *probabilistic* DP algorithm using s operations,

*Research supported by the DFG grant JU 3105/1-1 (German Research Foundation).

*Institut für Informatik, Goethe Universität, Frankfurt am Main, Germany. Affiliated with the Institute of Data Science and Digital Technologies, Faculty of Mathematics and Informatics of Vilnius University, Lithuania. stjukna@gmail.com.

[1] BPP stands for "**b**ounded-error **p**robabilistic **p**olynomial time", and P/poly for "deterministic non-uniform polynomial time". Whether $\mathsf{BPP} \subseteq \mathsf{P}$ holds in the *uniform* setting (for Turing machines instead of circuits) remains a widely open problem.

ISSN 1433-8092

can then f be also solved by a *deterministic* DP algorithm using a polynomial in ns number of operations?

We answer this question in the *affirmative*.

2. Results

2.1. **En route to derandomization.** We will derandomize probabilistic circuits in three steps given by Theorems 2.1–2.3 below.

The *Vapnik–Chervonenkis dimension* (VC dimension), $\mathrm{VCdim}(\mathcal{F})$, of a family $\mathcal{F}$ of functions $f : X \to Y$ is the largest natural number $v \in \mathbb{N}$ for which there exist v functions $f_1, \dots, f_v$ in $\mathcal{F}$ that are *shattered* by points in the following sense: for every subset $S \subseteq \{1, \dots, v\}$ there is a point $(x, y) \in X \times Y$ such that $f_i(x) = y$ if and only if $i \in S$. That is, the graphs of exactly those functions f_i with $i \in S$ must contain this point. If this holds for an arbitrary large $v \in \mathbb{N}$, then the VC dimension is infinite.

The *majority vote* function of m variables is a partly defined function $\mathrm{maj}(x_1, \dots, x_m)$ which outputs the majority element of its input $x_1, \dots, x_m$, if there is one. That is,

$$\mathrm{maj}(x_1, \dots, x_m) = y \text{ if } y \text{ occurs} > m/2 \text{ times among the } x_1, \dots, x_m.$$

For example, in the case of $m = 5$ variables, we have $\mathrm{maj}(a, b, c, b, b) = b$ whereas $\mathrm{maj}(a, b, c, a, b)$ is undefined.

The following theorem reduces the derandomization of *probabilistic* circuits to upper-bounding the VC dimension of *deterministic* circuits. It holds for any class of circuits computing real-valued functions $f : \mathbb{R}^n \to \mathbb{R}$, and for any probability distributions of their random input variables.

Theorem 2.1 (Infinite majority rule)**.** *If a function f can be computed by a probabilistic circuit of size s, then f can be also computed as a majority vote of at most $O(v)$ deterministic circuits of size at most s, where v is the VC dimension of the family of functions computable by deterministic circuits of size at most s.*

The theorem is a relatively direct consequence of the "uniform convergence in probability" result of Haussler [24] (see Section 6). A slightly worse upper bound $O(v \log v)$ (which is also sufficient in the context of the BPP vs. P/poly problem) follows also from the original result of Vapnik and Chervonenkis [47].

By Theorem 2.1, the problem of derandomizing probabilistic circuits reduces to proving upper bounds on the VC dimension of deterministic circuits of up to a given size. In next two theorems we do this for circuits whose gates are semialgebraic functions.

Recall that a set $S \subseteq \mathbb{R}^n$ is *semialgebraic* if it can be obtained by finitely many unions and intersections of sets defined by a polynomial equality or strict inequality. If this can be done using at most r polynomials of degree at most r, then the set is *r-semialgebraic*. The smallest r for which this is possible is the *description complexity* of the set S. A function $f : \mathbb{R}^n \to \mathbb{R}$ is *r-semialgebraic* if its graph $S = \{(x, y) \colon y = f(x)\} \subseteq \mathbb{R}^{n+1}$ is such; see Section 4 for more formal definitions, and Table 1 for examples of semialgebraic function of small description complexity.

An important consequence of the Tarski–Seidenberg theorem [46, 43]–stating that every *quantified* algebraic formula has an equivalent quantifier-free formula—is that compositions of semialgebraic functions are also semialgebraic functions. In particular, this implies that functions computable by circuits over any base consisting of semialgebraic functions are also semialgebraic. We are interested in the *quantitative* aspect of this theorem:

- If the basis functions (gates) have description complexity at most b, how large can then the description complexity of functions computable by circuits of size up to s be?

The answer is given by the following theorem.

Theorem 2.2 (Description complexity of circuits)**.** *Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions. Then every function $f : \mathbb{R}^n \to \mathbb{R}$ computable by a circuit over $\mathcal{B}$ of size at most s is r-semialgebraic for r satisfying*

$$\log r = O(ns \log bs)\,.$$

Note that r does not depend on the fanin of gates. We prove this theorem in Section 7 by first encoding circuits by quantified algebraic formulas (Lemma 7.1), and then using a result of Renegar [39] to eliminate quantifiers.

By Theorem 2.2, we know that the functions computed by circuits using semialgebraic functions of bounded description complexity as gates are r-semialgebraic for appropriate description complexity r. So, the following theorem gives a desired (to apply Theorem 2.1) upper bound on the VC dimension of such circuits.

Theorem 2.3 (VC dimension from description complexity)**.** *If $\mathcal{F}$ is the family of all r-semialgebraic functions $f : \mathbb{R}^n \to \mathbb{R}$, then*

$$\log_2 \binom{n+r}{n} - 1 \leqslant \mathrm{VCdim}(\mathcal{F}) = O(n \log r)\,.$$

In applications to derandomization, important here is that the upper bound is only *logarithmic* in the description complexity r. As such, this fact is not new: a "logarithmic shrinkage" was already observed by several authors, including Goldberg [16], Goldberg and Jerrum [15], Ben-David and Lindenbaum [6]. The point, however, is that the proofs in [16, 15, 6] were based on (known at that time) upper bounds of Heintz [25], Milnor [36], Warren [48] on the number of sign-patterns of polynomials, which themselves were proved using rather heavy techniques from real algebraic geometry. In contrast, Rónyai, Babai and Ganapathy [40] discovered a surprisingly simple linear algebra argument leading to an even better upper bound on the number of zero-patterns of polynomials. Thank to this result, the entire proof of the upper bound of Theorem 2.3 in Section 8 is short, elementary and self-contained. The lower bound of Theorem 2.3 is proved (also in Section 8) using the *dual* VC dimension.

2.2. **Applications.** The following direct consequence of Theorems 2.2 and 2.3 gives a general upper bound of the VC dimension of semialgebraic circuits.

Corollary 2.4. *Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions, and let $\mathcal{F}$ be the family of functions on n variables computable by the circuits over $\mathcal{B}$ of size at most s. Then*

$$\mathrm{VCdim}(\mathcal{F}) = O(n^2 s \log bs)\,.$$

Together with Theorem 2.1, Corollary 2.4 yields

Corollary 2.5. *Let $\mathcal{B}$ be any basis consisting of b-semialgebraic functions. If a function $f : \mathbb{R}^n \to \mathbb{R}$ can be computed by a probabilistic circuit of size s over $\mathcal{B}$, then f can be also computed as a majority vote of $m = O(n^2 s \log bs)$ deterministic circuits, each of size at most s.*

The deterministic circuits in Corollary 2.5 are just *copies* (realizations) $F(x, r_1), \ldots, F(x, r_m)$ of one and the same probabilistic circuit $F(x, \boldsymbol{r})$, meaning that all these circuits have the same structure—only the random inputs are fixed to (apparently) different constants.

Note also that, even though the majority vote functions are only *partially* defined, the derandomized circuit in Corollary 2.5 ensures that, on every input $x \in \mathbb{R}^n$ to the circuit, the sequence of values given to the last majority vote gate will *always* contain a majority element. This is an important aspect guaranteed by the infinite majority rule (Theorem 2.1).

Derandomized circuits in the following corollary require no majority vote gates at all. For a relation ρ, we will denote by $[\rho]$ the predicate which outputs 1 if the relation ρ holds, and outputs

0 otherwise. Consider the following four bases of operations:

$$\begin{aligned}
\mathcal{B}_1 &= \{\min, \max\}\,, \\
\mathcal{B}_2 &= \{\min, +, -\}\,, \\
\mathcal{B}_3 &= \{+, \times, [x = y]\,, [x > y]\}\,, \\
\mathcal{B}_4 &= \{+, \times, -, \mathrm{sgn}\}\,,
\end{aligned} \tag{1}$$

where $\mathrm{sgn}(x) = 1$ if $x \geqslant 0$, and $\mathrm{sgn}(x) = 0$ otherwise.

Corollary 2.6. *Let $\mathcal{B}$ be any basis consisting of b-semialgebraic gates. If a function $f : \mathbb{R}^n \to \mathbb{R}$ can be computed by a probabilistic circuit of size s over $\mathcal{B}$, and if $\mathcal{B}$ contains at least one of the bases $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$, then f can be also computed by a deterministic circuit of size $S = O(n^4 s^2 \log^2 bs)$.*

Proof. By Corollary 2.5, it is enough to show that, over any of these four bases, the majority vote of m inputs can be computed by a circuit of size $O(m^2)$: then the resulting circuit without any majority vote gates will have size at most $O(m^2 + ms) = O(m^2)$, where now $m = O(n^2 s \log bs)$ is from Corollary 2.5.

Basis $\mathcal{B}_1$: In the first case, we can take the sorting $\{\min, \max\}$ network of Ajtai, Komlós and Szemerédi [3]. This network requires only $O(m \log m)$ gates, and computes the *sorting function* $\mathrm{sort} : \mathbb{R}^m \to \mathbb{R}^m$ which on input string $(x_1, \ldots, x_m) \in \mathbb{R}^m$ outputs its permutation $(y_1, \ldots, y_m)$ with $y_1 \leqslant y_2 \leqslant \ldots \leqslant y_m$. This circuit then also computes the majority vote function $\mathrm{maj}(x_1, \ldots, x_m)$: just take the i-th output gate y_i of sort for $i := \lfloor m/2 \rfloor$; it will always contain the majority element of $(x_1, \ldots, x_m)$, if there is one.

Basis $\mathcal{B}_2$: This case reduces to the previous one because $\max\{x, y\} = x + y - \min\{x, y\}$.

Basis $\mathcal{B}_3$: In order to find the majority element among $x_1, \ldots, x_m \in \mathbb{R}$ (if there is one), first compute all m sums $y_i = \sum_{j=1}^{m} [x_j = x_i]$; then $\sum_{i=1}^{m} x_i \cdot [y_i > m/2]$ gives the majority element.

Basis $\mathcal{B}_4$: It is enough to simulate the atomic predicates $[x = y]$ and $[x > y]$ using signum operation: $[x = y] = \mathrm{sgn}(x - y) \cdot \mathrm{sgn}(y - x)$ and $[x > y] = 1 - \mathrm{sgn}(y - x)$. □

Motivated by Corollary 2.6, call a basis $\mathcal{B}$ *majority vote capable*, if the majority vote function of m variables can be computed by a circuit over $\mathcal{B}$ of size polynomial in m. In particular, bases considered in Corollary 2.6 are clearly majority capable.

Corollary 2.7. $\mathsf{BPP} \subseteq \mathsf{P/poly}$ *holds for circuits over any majority vote capable basis $\mathcal{B}$ consisting of semialgebraic functions of description complexity $b \leqslant 2^{n^{O(1)}}$.*

Three important examples of bases that are *not* majority vote capable are: the basis $\{+, -, \times\}$ of arithmetic circuits, and the bases $\{\min, +\}$ and $\{\max, +\}$ of tropical circuits (see Appendix C).

Most (if not all) DP algorithms in discrete optimization use only several semialgebraic functions of small description complexity in their recursion equations: min, max, arithmetic operations, and apparently some additional, but still semialgebraic operations, like the selection operation, or the "if-then–else" constraint (see Table 1 in Section 4.2). So, Corollary 2.7 implies that randomization is (almost) *useless* in dynamic programming algorithms, at least as long as we are allowed to use different (deterministic) DP algorithms to solve optimization problems on inputs $x \in \mathbb{R}^n$ from different dimensions n.

Remark 1 (On "uniformity"). Usually, a DP algorithm is described by giving *one* set of recursion equations which can be applied to inputs of *any* dimension n. In this respect, DP algorithms are "uniform" (like Turing machines). Probabilistic DP algorithms may use some random parameters in their recursion equations. However, when derandomizing such algorithms, we do not obtain also *one* set of recursion equations valid for inputs of *all* dimensions. What we obtain is a *sequence* of deterministic DP algorithms, one for each dimension n. In the "uniform" setting (with P instead of P/poly), the inclusion BPP $\subseteq$ P is *not* known to hold even for DP algorithms, and even for "pure" DP algorithms using only $(\min,+)$ or $(\max,+)$ operations in their recursion equations.

2.3. **Recognizing roots.** In Theorem 2.1, we require that, on every input $x \in R^n$, the probabilistic circuit $F(x,\boldsymbol{r})$ must output the correct *value* $f(x)$ with probability at least $2/3$. One can, however, relax this and only require that the values $F(x,\boldsymbol{r})$ must have some *properties* with this probability. If we are given a target function $f : R^n \to R$, then such properties may, for example, be:

- output correct values: $F(x,\boldsymbol{r}) = f(x)$ (the property we considered so far);
- have the same roots: $F(x,\boldsymbol{r}) = 0$ if and only if $f(x) = 0$;
- have the same sign: $F(x,\boldsymbol{r}) > 0$ if and only if $f(x) > 0$;
- have the same integrality: $F(x,\boldsymbol{r}) \in \mathbb{Z}$ if and only if $f(x) \in \mathbb{Z}$,

and many more. Our arguments allow also to derandomize circuits only recognizing some properties of given functions. To illustrate this, say that a probabilistic circuit $F(x,\boldsymbol{r})$ *recognizes the roots* of a given function $h : R^n \to R$ if, for every input $x \in R^n$,

$$\Pr\left\{F(x,\boldsymbol{r})^2 + h(x)^2 = 0 \text{ or } F(x,\boldsymbol{r}) \cdot h(x) \neq 0\right\} \geqslant 2/3\,.$$

That is, if $h(x) = 0$ then $F(x,\boldsymbol{r}) = 0$ with probability $\geqslant 2/3$, and if $h(x) \neq 0$ then $F(x,\boldsymbol{r}) \neq 0$ with probability $\geqslant 2/3$.

Theorem 2.8. *Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions, and containing the basis $\{+,\times\}$ or any of the bases listed in Eq. (1). If the roots of a p-semialgebraic function $h : \mathbb{R}^n \to \mathbb{R}$ can be recognized by a probabilistic circuit over $\mathcal{B}$ of size s, then the roots of h can be also recognized by a deterministic circuit over $\mathcal{B}$ of size $O(n^4 s^2 \log^2 \max\{sb, p\})$.*

The proof of Theorem 2.8 is given in Section 9.

2.4. **Randomization in arithmetic circuits is hopeless.** Our procedure for derandomizing semialgebraic circuits consisted of the following three steps:

1. upper-bound the size of derandomized circuits in terms of VC dimension (Theorem 2.1).
2. upper-bound the description complexity of semialgebraic circuits (Theorem 2.2).
3. upper-bound the VC dimension of semialgebraic functions in terms of their description complexity (Theorem 2.3).

Thank to Rónyai, Babai and Ganapathy [40], the proof for step (3) (in Section 8) is elementary and self-contained. However, the proofs of steps (1) and (2) are based on a rather involved proofs of Haussler [24] (on uniform convergence in probability) and Renegar [39] (on quantifier elimination). Still, in some cases, one can derandomize circuits in more direct ways. In Section 11, we demonstrate this in the case of tropical $(\max,+)$ and $(\min,+)$ circuits: here step (2) is not necessary.

In Section 10, we use an elementary argument, avoiding the need of all three steps (1)–(3), to prove the following *optimal* derandomization result for arithmetic circuits.

Theorem 2.9. *If a rational function f can be computed by a probabilistic circuit of size s over the basis $\{+,-,\times,\div\}$ with an arbitrarily small success probability $\epsilon > 0$, then f can be also computed by a deterministic circuit over $\{+,-,\times,\div\}$ of the same size s*

That is, randomization in arithmetic circuits cannot spare even one single gate.

Moreover, if $F(x,\boldsymbol{r})$ is a probabilistic circuit computing f, then the obtained deterministic circuit is a realization $F(x,r)$ of this circuit obtained by fixing the random inputs to constants.

Note, however, that such a tight derandomization, as in Theorem 2.9, is no more possible if we relax (as in Theorem 2.8) the requirement on the probabilistic circuits $F(x,\boldsymbol{r})$ from "compute the correct *values*" of a given function $f(x)$ to something like "have the same roots" as f (with positive success probabilities). For example, the roots of the polynomial

$$f(X,Y,Z)=\sum_{i=1}^{n}\sum_{j=1}^{n}\Big(z_{i,j}-\sum_{k=1}^{n}x_{i,k}y_{k,j}\Big)^2$$

are triples A,B,C of $n\times n$ matrices such that $AB=C$. A naive arithmetic circuit for this polynomial requires about n^3 gates. More sophisticated known matrix multiplication algorithms still result in circuits of size about n^{ω}, where $\omega=2.3729$.

But already 40 years ago, Freivalds [14] gave a simple *probabilistic* circuit of size only about n^2 detecting the roots of f with success probability $>1/2$: take $S=\{0,1,2,3,4\}$, let $\boldsymbol{r}$ be a uniformly distributed in S^n random vector, compute $u=(XY-Z)\boldsymbol{r}$ as the difference of *matrix-vector* products $u=X(Y\boldsymbol{r})-Z\boldsymbol{r}$, and output $F=u_1^2+\cdots+u_n^2$. Note that F outputs 0 if and only if u is the all-0 vector. After a triple of input matrices A,B,C is fixed, $F=F(\boldsymbol{r})$ turns into a polynomial of degree $d=2$ in random variables. If $AB=C$ (the triple *is* a root of f), then this is the zero polynomial and, hence, $F(\boldsymbol{r})=0$ holds with probability 1. If $AB\neq C$, then $F(\boldsymbol{r})$ is a *nonzero* polynomial, and Lemma 10.1 yields $\Pr\{\boldsymbol{r}\in S^n\colon F(\boldsymbol{r})=0\}\leqslant d/|S|=2/5$.

So, Theorem 2.9 does not exclude that randomization can still help, if we do not insist on computing the actual *values* of polynomials but are rather satisfied with, say, recognizing whether they are zero or not. On the other hand, Theorem 2.8 shows that also then the help cannot be super-polynomial. (Of course, in the special case of Freivald's polynomial, the upper bound given by Theorem 2.8 is trivial.)

Organization of the paper. In the next section (Section 3), we recall previous derandomization results for probabilistic circuits and decision trees. In Section 4, we specify our main concepts: probabilistic circuits and semialgebraic functions. Our basic derandomization tools are described in Section 5. The tools are "circuit independent:" they establish some combinatorial properties of *infinite* boolean matrices, and could be applied in other contexts. Theorems 2.1–2.3 are proved in subsequent sections 6-8. Theorem 2.9 (a tight derandomization of arithmetic circuits) is proved in Section 10.

Our *technical* contribution is actually minor: modulo deep results of Haussler [24] and Renegar [39], our proofs are fairly simple, and the paper is self-contained. Our contribution we rather see in a proper application of tools from different fields—combinatorial algebraic geometry (zero-patterns of polynomials), statistical learning theory (uniform convergence in probability), and quantifier elimination theory over the reals—to derandomize large classes of algorithms working over infinite domains, including all dynamic programming algorithms, the latter being our main motivation for this paper. Our main message is that randomization *cannot* substantially speed-up dynamic programming algorithms.

3. Previous work

As we mentioned at the beginning, our starting point is the result of Adleman [1] that[2] BPP $\subseteq$ P/poly holds for *boolean* circuits. In fact, Adleman proved this only when *one-sided* error is allowed. To prove the two-sided error version, Bennett and Gill [7] used the simple "finite majority rule", a direct consequence of Chernoff and union bounds (in a spirit of Claim 10.2 in Section 10):

Lemma 3.1 (Finite majority rule). *If a probabilistic circuit computes a given function $f: D\to R$ on a finite domain D with success probability $\epsilon\geqslant 1/2+c$ for a constant $c>0$, then the majority*

[2] Actually, the result is stronger, and should be stated as "'BPP/poly = P/poly": even probabilistic *circuits*, not only probabilistic Turing machines (*uniform* sequences of circuits) can be derandomized. We, however, prefer to use the less precise but more familiar shortcut "BPP $\subseteq$ P/poly."

vote of some $m = O(\log |D|)$ realizations of the circuit (deterministic circuits) also computes f on the entire domain D.

In the *boolean* case, the domain is $D = \{0,1\}^n$, and the majority vote functions turns into boolean majority functions: output 1 if and only if more than half of the input bits are 1s. Since the majority function has small boolean circuits, even monotone ones, the resulting deterministic circuits will then be not much larger than the probabilistic circuits.

There are models of boolean circuits—like constant depth circuits—where the majority function cannot be computed in polynomial size. Still, using different arguments, Ajtai and Ben-Or [2], were able to show that $\mathsf{BPP} \subseteq \mathsf{P/poly}$ holds also for constant-depth circuits.

Morizumi [37] considered another (than size) measure of boolean $(\vee, \wedge, \neg)$ circuits—the number of used NOT gates. A natural question is: can randomness substantially reduce the number of NOT gates? Markov [34] has found a surprisingly tight characterization of the minimum number of NOT gates required by deterministic circuits to compute a given boolean functions f in terms a natural combinatorial characteristic of f. Morizumi [37] observed that this result already gives a negative answer: random circuits can save at most a *constant* number of NOT gates; the constant depends only on the success probability.

The derandomization of circuits working over *infinite* domains D is a more delicate task: we have to somehow "cope" with the infinity of the domain: Chernoff's bounds alone do not help then. Two general approaches have emerged along this line of research:

(A) Find (or just prove a mere existence of) a *finite* set $X \subset D$ of inputs which is "isolating" in the following sense: if a (deterministic) circuit computes a given function f correctly on all inputs $x \in X$, then it must compute f correctly on *all* inputs $x \in D$. Then use the finite majority rule on inputs from X.

(B) Use the "infinite majority rule" following from the "uniform convergence in probability" results in the statistical learning theory: this allows to replace $\log |D|$ in the finite majority rule by the Vapnik–Chervonenkis dimension of the (deterministic) circuits of up to some given size (see Theorem 2.1).

Approach (A) was used by many authors to show $\mathsf{BPP} \subseteq \mathsf{P/poly}$ for various types of decision trees. The complexity measure here is the depth of a tree. These trees work over $\mathbb{R}$, and branch according to the sign of values of rational functions. In the case when only linear functions are allowed, the inclusion $\mathsf{BPP} \subseteq \mathsf{P/poly}$ was proved by Manber and Tompa [33], and Snir [45]. Meyer auf der Heide [35] proved the inclusion when arbitrary rational functions are allowed. He uses a result of Milnor [36] about the number of connected components of polynomial systems in $\mathbb{R}^n$ to upper-bound the minimum size of an "isolating" subset $X \subset \mathbb{R}^n$. Further explicit lower bounds on the depth of probabilistic decision trees were proved by Bürgisser, Karpinski and Lickteig [8], Grigoriev and Karpinski [20], Grigoriev et.al. [21], Grigoriev [19] and other authors.

Approach (B) was used by Cucker et. al. [11] to prove $\mathsf{BPP} \subseteq \mathsf{P/poly}$ for arithmetic circuits over the basis $\{+, -, \times, \div, \mathrm{sgn}\}$: if an n-variate polynomial f can be computed by a probabilistic circuit of size at most s, then f can be also computed as a majority vote of $O(ns)$ deterministic circuits, each of size s. Our Theorem 2.9 shows that for arithmetic circuits over the basis $\{+, -, \times, \div\}$ (without signum gates), randomization cannot spare even one single gate!

The BPP vs. P problem in the *uniform* setting, that is, in terms of Turing machines, is even more delicate task. Here neither finite nor infinite majority rule can help. The reason for this failure is that Turing machines are allowed to behave *arbitrarily* on any *finite* number of inputs they receive. So, the Vapnik–Chervonenkis dimension of Turing machines working in even linear time is infinite.

Still, a strong indication that $\mathsf{BPP} = \mathsf{P}$ should hold also in the uniform setting was given by Impagliazzo and Wigderson [29]: either $\mathsf{BPP} = \mathsf{P}$ holds or *every* decision problem solvable by deterministic Turing machines in time $2^{O(n)}$ can be solved by boolean circuits of sub-exponential size $2^{o(n)}$. Goldreich [17] related the BPP vs. P problem with the existence of pseudorandom generators: BPP = P if and only if there exists suitable pseudorandom generators; the "if" direction was known for decades—the novelty is in the converse.

4. Preliminaries

4.1. **Probabilistic circuits.** Let $R \subseteq \mathbb{R}$ be some set (a *domain*), and $\mathcal{B}$ some fixed family of functions $g : R^m \to R$ (a *basis*). A *circuit* over a given basis $\mathcal{B}$ is just a sequence $F = (f_1, \ldots, f_s)$ of functions $f_i : R^n \to R$, called *gates*, where each f_i is obtained by applying one of the basis operations to functions in $R \cup \{x_1, \ldots, x_n, f_1, \ldots, f_{i-1}\}$; scalars $a \in R$ are also (constant) functions $a : R^n \to R$ with $a(x) = a$ for all $x \in R^n$, and each variable x_i is the projection function of vectors $x \in R^n$ to the i-th coordinate. The *size* of a circuit is the number s of functions in the sequence, and the function $F : R^n \to R$ *computed* by the circuit is the function computed by the last gate f_s in the sequence.

One usually views a circuit as a directed acyclic graph; parallel edges joining the same pair of nodes are allowed. Each indegree-zero node holds either one of the variables $x_1, \ldots, x_n$ or an element of R. Every other node, a *gate*, performs one of the operations $g \in \mathcal{B}$ on the results computed at its input gates.

A *probabilistic circuit* over a basis $\mathcal{B}$ is a deterministic circuit which, besides the actual variables $x_1, \ldots, x_n$, is allowed to use additional variables $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_k$, each being a random variable taking its values in R. So, what such circuits compute are random functions $F : R^{n+k} \to R$ whose values depend on the values of the random input variables. Such a circuit *computes* a given function $f : R^n \to R$ with a *success probability* ϵ if, for every input $x \in R^n$, the circuit outputs the correct value $f(x)$ with probability at least ϵ. We will sometimes call a circuit without random inputs a *deterministic* circuit, just to distinguish it from a probabilistic one.

We will say that BPP $\subseteq$ P/poly holds for circuits over a given basis when there are constants a and b for which the following holds: if a function f of n variables can be computed by a probabilistic circuit of size s, then there is a deterministic circuit computing f whose size is at most $a(ns)^b$, that is if the size of derandomized circuits is polynomial in n and s.

4.2. **Semialgebraic Functions.** We will consider circuits using semialgebraic functions $f : \mathbb{R}^n \to \mathbb{R}$ as gates. Recall that a set $S \subseteq \mathbb{R}^n$ is *semialgebraic* if it can be obtained by finitely many unions and intersections of sets defined by a polynomial equality or strict inequality. For us, important will be not the mere fact that a set S *is* semialgebraic, but rather "how much semialgebraic" it actually is: how many distinct polynomials and of what degree do we need to define this set? To capture this quantitative aspect, we use "algebraic formulas".

An *algebraic formula* is an arbitrary boolean combination of atomic predicates, each being of the form $[p(x) \lozenge 0]$ for some polynomial p in $\mathbb{R}[x_1, \ldots, x_n]$, where $\lozenge$ is one of the standard relations $>$, $\geqslant$, $=$, $\neq$, $\leqslant$, $<$, and the predicate $[\rho]$ for a relation ρ outputs 1 if the relation ρ holds, and outputs 0 otherwise. So, for example, $[p(x) = 0] = 1$ if and only if $p(x) = 0$. Note that $[p(x) \lozenge q(x)]$ is equivalent to $[p(x) - q(x) \lozenge 0]$, so that we can also make comparisons between polynomials. The *size* of a formula is the number of used polynomials, while the *degree* of the formula is the maximum degree of these polynomials. It is important to note that the size only counts distinct polynomials: one and the same polynomial may appear many times, and under different relations $\lozenge$.

Note that a set $S \subseteq \mathbb{R}^n$ is semialgebraic if and only if it can be *recognized* by some algebraic formula $\Phi : \mathbb{R}^n \to \{0, 1\}$ in that $S = \{x \in \mathbb{R}^n \colon \Phi(x) = 1\}$ holds. A function $f : \mathbb{R}^n \to \mathbb{R}$ is *semialgebraic* if its graph $S = \{(x, y) \colon y = f(x)\} \subseteq \mathbb{R}^{n+1}$ is such, that is, if there is an algebraic formula $\Phi(x, y)$ such that, for every $x \in \mathbb{R}^n$ and $y \in \mathbb{R}$,

$$\Phi(x, y) = 1 \text{ holds precisely when } y = f(x).$$

TABLE 1. Some basic semialgebraic functions f of small description complexity $r(f)$. Here, $p(x)$ is an arbitrary real multivariate polynomial of degree d; $\Psi(x)$ is a d-semialgebraic formula viewed as a function $\Psi : \mathbb{R}^n \to \{0,1\}$; $\mathrm{sgn} : \mathbb{R} \to \{0,1\}$ takes value $\mathrm{sgn}(x) = 1$ if $x \geqslant 0$, and $\mathrm{sgn}(x) = 0$ otherwise; $\mathrm{maj}(x_1, \dots, x_m)$ is the *majority vote function* which outputs the number appearing more than $m/2$ times in the string $x_1, \dots, x_m$, if there is one; $\mathrm{sel}(x_1, \dots, x_m|y)$ is also a partly defined function which outputs x_i if $y = i$.

Graph of function f	$r(f)$	Algebraic formula Φ
$y = p(x)$	d	$[y = p(x)]$
$y = \Psi(x)$	$d+2$	$[y = 1] \wedge \Psi(x) \vee [y = 0] \wedge \neg\Psi(x)$
$y = \mathrm{sgn}(x)$	2	$[x \geqslant 0] \wedge [y = 1] \vee [x < 0] \wedge [y = 0]$
$y = \|x\|$	2	$[y = x] \vee [y = -x]$
$y = x^{1/k}$	k	$\big[x = y^k\big]$ (odd k) $[x \geqslant 0] \wedge \big[x = y^k\big]$ (even k)
$z = \\|x - y\\|$	2	$[z \geqslant 0] \wedge \big[z^2 = (x_1 - y_1)^2 + \cdots + (x_n - y_n)^2\big]$
$z = \frac{x}{y}$	2	$[y \neq 0] \wedge [yz = x]$
$z = \min\{x_1, \dots, x_m\}$	m	$\bigwedge_{i=1}^m [z \leqslant x_i] \wedge \big(\bigvee_{i=1}^m [z = x_i]\big)$
$z = \max\{x_1, \dots, x_m\}$	m	$\bigwedge_{i=1}^m [z \geqslant x_i] \wedge \big(\bigvee_{i=1}^m [z = x_i]\big)$
$y = \mathrm{maj}(x_1, \dots, x_m)$	m	$\mathrm{maj}\,\big([y = x_1], \dots, [y = x_m]\big)$
$z = \mathrm{sel}(x_1, \dots, x_m\|y)$	$2m$	$\bigvee_{i=1}^m [y = i] \wedge [z = x_i]$
$z =$ "if $\Psi(x) = 1$ then u else v"	$d+2$	$\Psi(x) \wedge [z = u] \vee \neg\Psi(x) \wedge [z = v]$

Note that this definition makes sense also for *partly* defined functions: if the value $f(x)$ is undefined, then it is enough that $\Phi(x, y) = 0$ holds for all $y \in \mathbb{R}$.

The *description complexity* of a semialgebraic function f is the smallest number r such that the graph of f can be recognized by an algebraic formula of size and degree at most r. An r-*semialgebraic* function is a semialgebraic function of description complexity $\leqslant r$. Table 1 gives a sample of some basic semialgebraic functions of small description complexity. Recall that we only count the number of distinct polynomials used (and their degree); say, in the case of the min operation, we only use linear polynomials $z - x_i$.

Remark 2. Note also that every boolean function $\phi : \{0,1\}^n \to \{0,1\}$ of n variables is $(2n)$-semialgebraic. Indeed, take an arbitrary boolean formula $F(z_1, \dots, z_m)$ (say, a DNF) computing f, where each input literal is either a variable or its negation. If a literal is x_i (resp., $\neg x_i$), then replace it by the atomic predicate $[x_i - 1 = 0]$ (resp., by $[x_i = 0]$). The resulting algebraic formula has size $2n$ and degree 1: only $2n$ distinct linear polynomials are used.

As we already mentioned above, one of the most basic facts about semialgebraic sets is the famous Tarski–Seidenberg theorem [46, 43], implying that projections of semialgebraic sets are semialgebraic sets. In particular, the sets recognizable by *quantified* algebraic formulas are also semialgebraic sets. This gives us a powerful tool to show that many other important (say, in the context of dynamic programming) operations are r-semialgebraic for relatively small values of r. We illustrate this (on an example of $\arg\max$ operation) in Appendix B.

5. Majority property of infinite boolean matrices

We will consider boolean (not necessarily finite) matrices $M : A \times B \to \{0,1\}$. We assume that their rows and columns can be *indexed* using a finite number of real parameters, that is, there are integers $n, k \geqslant 1$ such that $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^k$. The smallest number n for which such an indexing is possible is the *row-dimension* of the matrix.

Definition 1. A boolean matrix has the *m-majority property* if there are m columns such that every row has more than $m/2$ ones in these columns.

Our goal is to ensure this property for m possibly small. The connection of this property with the derandomization of probabilistic circuits is the following. Let $F(x, \boldsymbol{r})$ be probabilistic circuit with $n+k$ input variables (n deterministic and k random variables) computing a given function $f : \mathbb{R}^n \to \mathbb{R}$. We can associate with F a boolean matrix M whose rows correspond to vectors $x \in \mathbb{R}^n$, columns to vectors $r \in \mathbb{R}^k$, and the entries are defined by: $M[x,r] = 1$ if and only if $F(x,r) = f(x)$. Now, if this matrix has an m-majority property, then there exist m assignments $r_1, \ldots, r_m \in \mathbb{R}^k$ to the random variables such that $\mathrm{maj}(F(x,r_1), \ldots, F(x,r_m)) = f(x)$ holds for all $x \in \mathbb{R}^n$. That is, the majority vote of m *deterministic* circuits computes our function f correctly on all inputs.

The matrices arising from probabilistic circuits have an additional property formulated in the next definition.

Definition 2. A boolean matrix $M : A \times B \to \{0,1\}$ *probabilistically dense* if there exists a probability distribution $\mathrm{Pr} : B \to [0,1]$ on the set of columns such that[3]

$$\mathrm{Pr}\,\{b \in B \colon M[a,b] = 1\} \geqslant 2/3$$

holds for every row $a \in A$.

Note that the mere *existence* of at least one probability distribution with this property is sufficient; so, density is a property of matrices, not of probability distributions on their columns.

The (finite) majority rule (see Section 3) implies that every probabilistically dense boolean matrix with a finite number $|A|$ of rows has the m-majority property already for $m = O(\log |A|)$. This upper bound is, however, useless for matrices with an *infinite* number of rows. Still, also then one can upper bound m in terms of the "algebraic description complexity" of their entries.

Definition 3. A boolean matrix $M : A \times B \to \{0,1\}$ is *r-semialgebraic* if for every column $b \in B$ there is an algebraic formula $\Phi_b(x)$ of size and degree at most r such that $M[a,b] = \Phi_b(a)$ holds for every row $a \in A$.

Theorem 5.1. *Every probabilistically dense r-semialgebraic boolean matrix of row-dimension n has the m-majority property for $m \leqslant cn \log r$, where c is an absolute constant.*

This theorem is a direct consequence of Lemmas 5.2 and 5.4 below. The first of these lemmas (Lemma 5.2) upper-bounds m in terms of the Vapnik–Chervonenkis dimension of matrices, whereas the second lemma (Lemma 5.4) upper-bounds this dimension in terms of the "algebraic description complexity" of matrices.

5.1. Majority property and the VC dimension.

Recall that the *Vapnik–Chervonenkis dimension* (or *VC dimension*) of a boolean matrix M is the maximum number v of its columns such that the rows of M take all 2^v possible 0-1 patterns from $\{0,1\}^v$ in these columns (possibly, with repetitions). If this happens for arbitrarily large vs, then the VC dimension of M is infinite. Note

[3] Again, there is nothing "magical" in the choice of this threshold value $\epsilon = 2/3$: one can take any constant larger than $1/2$. Only the constant c in Lemma 5.2 depends on the constant ϵ.

that, if the number $|A|$ of rows is finite, then we always have $v \leqslant \log |A|$: all possible 2^v patterns must be present.

A result of Haussler [24] in the statistical learning theory implies that any probabilistically dense boolean matrix with a finite VC dimension v has the majority property already when $m = O(v)$. To avoid pathological situations when the number of rows is *uncountable*, a mild measurability condition, called *permissibility* of matrices, is required. As shown by Haussler [24, Appendix 9.2], a boolean matrix $M : A \times B \to \{0,1\}$ is permissible, as long as its rows and columns can be *indexed* using a finite number of real parameters. So, since we only consider such boolean matrices, all they will be automatically permissible.

Lemma 5.2. *There is a constant c such that every permissible probabilistically dense boolean matrix of a finite VC dimension v has the m-majority property for $m \leqslant c \cdot v$.*

A simple derivation of this lemma form Haussler's result [24, Corollary 2] is given in Appendix A.

5.2. Zero patterns of polynomials.

To upper-bound the VC dimension, we will use the following upper bound of Rónyai, Babai and Ganapathy [40] on the number of zero patterns of polynomials.

A *zero pattern* of a sequence $\vec{p} = (p_1, \ldots, p_m)$ of polynomials $p_i \in \mathbb{F}[x_1, \ldots, x_n]$ over a field $\mathbb{F}$ is a subset $S \subseteq \{1, \ldots, m\}$ for which there exists an input $a \in \mathbb{F}^n$, a *witness* for S, such that $p_i(a) \neq 0$ holds if and only if $i \in S$. Let $Z(\vec{p})$ denote the number of distinct zero patterns of the sequence $\vec{p}$; hence, $1 \leqslant Z(\vec{p}) \leqslant 2^m$.

Lemma 5.3 (Rónyai, Babai and Ganapathy [40])**.** *Let $\vec{p} = (p_1, \ldots, p_m)$ be a sequence of n-variate polynomials of degree at most d over a field $\mathbb{F}$. Then*

$$Z(\vec{p}) \leqslant \binom{md}{n} \leqslant \left(\frac{\mathrm{e}md}{n}\right)^n .$$

For completeness, we include their amazingly simple linear algebra proof of a slightly worse bound $Z(\vec{p}) \leqslant \binom{n+md}{n} \leqslant (2\mathrm{e}md/n)^n$; since we do not specify the constat c in Theorem 5.1, this bound also suffices for our purposes. Previous proofs of similar upper bounds by Heintz [25], Milnor [36], and Warren [48] used heavy techniques from real algebraic geometry.

Proof. Let $a_1, \ldots, a_t \in \mathbb{F}^n$ be witnesses to all $t = Z(\vec{p})$ zero-patterns. Consider the polynomials

$$f_i = \prod_{s \in S_i} p_s ,$$

where $S_i = \{s : p_s(a_i) \neq 0\}$ is the zero-pattern witnessed by the i-th vector a_i, for $i = 1, \ldots, t$. We claim that the polynomials $f_1, \ldots, f_t$ must be linearly independent. For this, assume contrariwise that a nontrivial linear relation

$$\lambda_1 \cdot f_1 + \cdots + \lambda_t \cdot f_t = 0$$

with not all zero coefficients $\lambda_i \in \mathbb{F}$ exists. Let j be a subscript such that $|S_j|$ is minimal among the S_i with $\lambda_i \neq 0$, and substitute vector a_j in the relation. Since

$$f_i(a_j) \neq 0 \text{ holds if and only if } S_i \subseteq S_j ,$$

we have $\lambda_j \cdot f_j(a_j) \neq 0$ and $\lambda_i \cdot f_i(a_j) = 0$ for all $i \neq j$, a contradiction. So, the polynomials $f_1, \ldots, f_t$ are linearly independent. Since each f_i has degree at most $D = md$, and since the dimension of the space of n-variate polynomials of degree at most D is $\binom{n+D}{D} = \binom{n+D}{n} \leqslant (2\mathrm{e}D/n)^n$, the desired upper bound on the number t of zero patterns follows. □

5.3. **VC dimension of semialgebraic matrices.** The proof of the following lemma is based on a simple observation (made already by many authors): if $\Phi : \mathbb{R}^n \to \{0,1\}$ is an algebraic formula, then its values $\Phi(x)$ on points $x \in \mathbb{R}^n$ depend not on the actual *values* taken by the involved in Φ polynomials, but merely on the *signum patterns* of these polynomials at these points x. We will additionally observe that actually only *zero patterns* of these polynomials do matter.

Lemma 5.4. *There is a constant c such that the VC dimension of every r-semialgebraic matrix of row-dimension n is at most $4n \log r + 5n$.*

Proof. Let $M : A \times B \to \{0,1\}$ be a boolean matrix with $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^k$, and suppose that M is r-semialgebraic. Our goal is to show that then the VC dimension v of M cannot exceed $2n \log(2\mathrm{e}r^2) \leqslant 4n \log r + 5n$. For this, recall that, by the definition of the VC dimension of matrices, there must be v columns $b_1, \ldots, b_v \in B$ of matrix M such that

$$\big\{\big(M[x,b_1], \ldots, M[x,b_v]\big) \colon x \in A\big\} = \{0,1\}^v \,. \tag{2}$$

Since the matrix M is r-semialgebraic, for every $i = 1, \ldots, v$ there must be an algebraic formula $\Phi_i'(x)$ which uses at most r distinct polynomials, each of degree at most r, and satisfies $M[x,b_i] = \Phi_i'(x)$ for all $x \in A$.

We can assume that each formula Φ_i' is in a *reduced* form, meaning that each atomic predicate is of the form $[p < 0]$, $[p = 0]$ or $[p > 0]$. For this, just replace each atomic predicate $[p \leqslant 0]$ by the formula $[p = 0] \vee [p < 0]$, each $[p \geqslant 0]$ by the formula $[p = 0] \vee [p > 0]$, and each $[p \neq 0]$ by the formula $[p < 0] \vee [p > 0]$. Note that the size of the formula (number of distinct polynomials used) and its degree remain unchanged when doing these transformations.

Now, replace each predicate $[p < 0]$ by $[-p > 0]$. The size of the formula can only double when doing this (if the polynomials $-p$ were not already used). What we achieve by these simple tricks is that now the value of a predicate defined by a polynomial p only depends on whether $p = 0$ or $p \neq 0$. The size of the resulting formula Φ_i (the number of used polynomials) is at most $2r$. By (2), the formulas $\Phi_1, \ldots, \Phi_v$ must satisfy the equality

$$\big\{\big(\Phi_1(x), \ldots, \Phi_v(x)\big) \colon x \in A\big\} = \{0,1\}^v \,. \tag{3}$$

Let $p_1, \ldots, p_m$ be the polynomials used in at least one of the formulas $\Phi_1, \ldots, \Phi_v$. So, we have a sequence $\vec{p} = (p_1, \ldots, p_m)$ of $m \leqslant 2rv$ n-variate polynomials of degree at most r.

Since atomic predicates $[p(x) = 0]$ and $[p(x) > 0]$ in any of the formulas Φ_i only depend on whether $p(x) = 0$ or $p(x) \neq 0$, the sequence $\vec{p} = (p_1, \ldots, p_m)$ of polynomials used in any of these formula must have

$$Z(\vec{p}) \geqslant 2^v \tag{4}$$

distinct zero patterns to ensure the equality (3). But, by Lemma 5.3, this sequence can only have

$$Z(\vec{p}) \leqslant \left(\frac{\mathrm{e}mr}{n}\right)^n \tag{5}$$

distinct zero patterns. By comparing the lower bound (4) with upper bound (5), we have that the VC dimension v must satisfy the inequality

$$2^v \leqslant \left(\frac{\mathrm{e}mr}{n}\right)^n \leqslant \left(\frac{2\mathrm{e}vr^2}{n}\right)^n$$

or equivalently $2^{v/n} \leqslant 2\mathrm{e}r^2(v/n)$. By taking logarithms, we obtain $v/n \leqslant \log(2\mathrm{e}r^2) + \log(v/n)$. If $2\mathrm{e}r^2 \geqslant v/n$, then $v/n \leqslant 2\log(2\mathrm{e}r^2)$ and, hence, $v \leqslant 2n\log(2\mathrm{e}r^2)$. If $2\mathrm{e}r^2 < v/n$, then $v/n \leqslant 2\log(v/n)$ implying that $v/n \leqslant 4$ and, hence, $v \leqslant 4n$. So, in both cases, we have $v \leqslant 2n\log(2\mathrm{e}r^2)$, as desired. □

6. Proof of Theorem 2.1: Infinite Majority Rule

Let $R \subseteq \mathbb{R}$ be some real domain, and fix some class $\mathcal{C}$ of circuits computing functions from R^n to R. Take any such function $f : R^n \to R$, and a probabilistic circuit $F(x, \boldsymbol{r})$ of size s in $n+k$ variables computing f. Let $\mathcal{C}_s$ be the family of deterministic circuits in this class of size at most s, and v be the VC dimension of the family of functions computable by circuits in $\mathcal{C}_s$. Our goal is to show that then f can be also computed as a majority vote of at most $O(v)$ deterministic circuits of size at most s.

For this, associate with the circuit F the boolean matrix M_F whose rows correspond to points $(x, y) \in R^{n+1}$, columns correspond to points $r \in R^k$, and the values are defined by:

$$M_F[(x, y), r] = 1 \text{ if and only if } F(x, r) = y.$$

Since for every assignment $r \in R^k$ of values to the random inputs $\boldsymbol{r}$, the circuit $F(x, r)$ belongs to $\mathcal{C}_s$, the VC dimension of the matrix M_F is also at most v.

Consider the boolean matrix $M \colon R^n \times R^k \to \{0, 1\}$ with entries $M[x, r] = 1$ if and only if $F(x, r) = f(x)$. This matrix is a submatrix of M_F obtained by removing all rows (x, y) with $y \neq f(x)$, and replacing the label $(x, f(x))$ of each remaining row by x. Since removal of rows (or columns) can only decrease the VC dimension of matrices, the VC dimension of M is also at most v. Since the probabilistic circuit F computes our function f, the matrix M is probabilistically dense. So, by Lemma 5.2, the matrix M has an m-majority property for some $m \leqslant c \cdot v$. That is, there must be some m columns $r_1, \ldots, r_m$ of M such that $\text{maj}(F(x, r_1), \ldots, F(x, r_m)) = f(x)$ holds for all $x \in R^n$. □

7. Proof of Theorem 2.2: Description complexity of semialgebraic circuits

We will prove this theorem by first encoding circuits by quantified algebraic formulas (Lemma 7.1), and then eliminating the quantifiers. The quantifier-free algebraic formulas resulting form the Tarski–Seidenberg theorem [46, 43] may be of extremely large size and degree: towers of exponentials in the size and degree of the original quantified formula.

Fortunately, the so-called *critical point method* (a method for finding at least one point in every semi-algebraically connected component of an algebraic set) has led to much smaller blowup factors. This method was pioneered by Grigoriev and Vorobjov [22, 23], Renegar [39], and later improved in various ways by several researchers including Canny [9], Heintz, Roy and Solernó [27], Basu, Pollack and Roy [5] amongst others. For our purposes, the result of Renegar [39] will fit best.

Under an *existential* algebraic formula with l quantifiers we will understand a formula $\Psi(x)$ of a form

$$(\exists z_1 \in \mathbb{R})\ (\exists z_2 \in \mathbb{R})\ \ldots\ (\exists z_l \in \mathbb{R})\ \Phi(x, z_1, \ldots, z_l)\,, \tag{6}$$

where Φ is a (quantifier-free) algebraic formula. The size and degree of Ψ is the size and degree of Φ. Recall that a (quantified or non-quantified) algebraic formula $\Psi(x, y)$ recognizes the graph of a function $f : \mathbb{R}^n \to \mathbb{R}$ if for every point $(x, y) \in \mathbb{R}^{n+1}$, $\Psi(x, y) = 1$ holds precisely when $y = f(x)$.

Lemma 7.1 (Circuits as quantified formulas)**.** *Let $\mathcal{B}$ be basis consisting of b-semialgebraic functions $g : \mathbb{R}^k \to \mathbb{R}$. If a function $f : \mathbb{R}^n \to \mathbb{R}$ can be computed by a circuit over $\mathcal{B}$ of size s, then the graph of f can be recognized by an existential algebraic formula of size at most $s \cdot b$, degree at most b and with $s - 1$ quantifiers.*

Proof. Recall that a circuit over $\mathcal{B}$ is just a sequence $F = (f_1, \ldots, f_s)$ of functions $f_i : \mathbb{R}^n \to \mathbb{R}$, called *gates*, where each f_i is obtained by applying one of the basis operations $g_i \in \mathcal{B}$ to functions in $\mathbb{R} \cup \{x_1, \ldots, x_n, f_1, \ldots, f_{i-1}\}$. Since every basis function $g_i : \mathbb{R}^k \to \mathbb{R}$ is b-semialgebraic, there must be an algebraic formula $\Phi_i(x, y)$ of size and degree at most d such that $\Phi_i(x, y) = 1$ if and only if $y = g_i(x)$.

Replace now each gate f_i in F by a new variable z_i. Then every gate $f_i = g_i(f'_{i_1}, \dots, f'_{i_k})$ with each f'_{i_j} in $\mathbb{R} \cup \{x_1, \dots, x_n, f_1, \dots, f_{i-1}\}$ turns into equation $z_i = g_i(\vec{w}_i)$, where $\vec{w}_i$ is a vector in $(\mathbb{R} \cup \{x_1, \dots, x_n, z_1, \dots, z_{i-1}\})^k$. So, $\Phi_i(\vec{w}_i, z_i) = 1$ if and only if $z_i = g_i(\vec{w}_i)$, implying that the existential formula

$$\begin{aligned}\Psi(x,y) &= \exists z_1 \dots \exists z_{s-1}\ \Phi_1(\vec{w}_1, z_1) \wedge \dots \wedge \Phi_{s-1}(\vec{w}_{s-1}, z_{s-1}) \wedge \Phi_s(\vec{w}_s, y) \\ &= \exists z_1 \dots \exists z_{s-1}\ [z_1 = g_1(\vec{w}_1)] \wedge \cdots \wedge [z_{s-1} = g_{s-1}(\vec{w}_{s-1})] \wedge [y = g_s(\vec{w}_s)]\end{aligned}$$

recognizes the graph $\{(x,y) \colon y = f_s(x)\}$ of the function $f = f_s$ computed by our circuit F. Since each algebraic formula Φ_i has size and degree at most d, the formula Ψ has size at most $s \cdot b$, degree at most b, and contains only $s-1$ quantifiers. □

To apply Theorem 2.3, we have to eliminate quantifiers from the formulas given by Lemma 7.1. We will use the following general result of Renegar [39, Theorem 1.2]. This result deals with quantified formulas $\Psi(x)$ of the form

$$(Q_1\, \vec{z}_1 \in \mathbb{R}^{n_1})\ \dots (Q_\omega\, \vec{z}_\omega \in \mathbb{R}^{n_\omega})\ \Phi(x, \vec{z}_1, \dots, \vec{z_\omega}), \tag{7}$$

where $Q_i \in \{\exists, \forall\}$ are alternating quantifiers, each $\vec{z}_i$ is a sequence of n_i real variables, and Φ is an algebraic formula of size m and degree d. The important parameters of the formula (7) are:

n_0 = number of free variables (x-variables);
n_i = number of z-variables in the i-th block of quantifiers;
m = number of polynomials used in the formula Φ;
d = maximal degree of these polynomials;
$\omega - 1$ = number of alternations between quantifiers $\exists$ and $\forall$.

Theorem 7.2 (Renegar [39])**.** *There exists a constant c such that every quantified formula $\Psi(x)$ of the form (7) can be written as a quantifier free algebraic formula of the form*

$$\bigvee_{i=1}^{L} \bigwedge_{j=1}^{D} [p_{ij}(x) \Diamond_{ij}\ 0] \quad \textit{with} \quad L \leqslant (md)^{2^{c \cdot \omega} N} \quad \textit{and} \quad D \leqslant (md)^{2^{c \cdot \omega} N / n_0},$$

where $N = \prod_{i=0}^{\omega} n_i$, the p_{ij}s are polynomials of degree at most d, and each $\Diamond_{i,j}$ is one of the relations $>, \geqslant, =, \neq, \leqslant, <$.

Note that L and D are doubly exponential only in the number $\omega - 1$ of *alternations* of quantifiers. Results of Weispfenning [49], and Davenport and Heintz [12] show that the double exponential dependence on ω in the upper bound on D cannot be improved in the worst case.

Remark 3. After a very large bound resulting from the Tarski–Seidenberg theorem, the first reasonable upper bound $L \leqslant (md)^E$ with the exponent $E = 2^{O(n_0 + \cdots + n_\omega)}$ was proved by Collins [10]. This bound is still double exponential in the number of variables. The next major complexity breakthrough was made by Grigoriev in [18], where he achieved $E = O(n_0 + \cdots + n_\omega)^{4\omega}$; the bound on L is then double exponential only in the alternations of quantifiers. Both Collins' and Grigoriev's results require integer coefficients. For real polynomials, Heintz, Roy and Solernó [26] achieved $E = (n_0 + \cdots + n_\omega)^{O(\omega)}$. Renagar's theorem (Theorem 7.2) achieves $E = 2^{O(\omega)} n_0 n_1 \cdots n_\omega$.

Now, in the existential formulas arising in Lemma 7.1 from circuits of size at most s , we have $\omega = 1$ (no quantifier alternations), d is the description complexity of basis operations, $m \leqslant s \cdot b$, $n_0 = n$ (the number of input variables in circuits), and $n_1 = s - 1$. So, in our context, Renegar's result gives the smallest exponent $E = O(n_0 n_1) = O(ns)$ in terms of s: the circuit size s is a critical parameter in derandomization, and we want the size of derandomized circuits be at most s^c for as small constant c as possible. Note, however, that in order to show a mere inclusion $\mathsf{BPP} \subseteq \mathsf{P/poly}$, bounds of [18] and [26] would also suffice.

Proof of Theorem 2.2. Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions, and let $f : \mathbb{R}^n \to \mathbb{R}$ be a function computable by a circuit over $\mathcal{B}$ of size at most s. Our goal is to show than then f must be r-semialgebraic for r satisfying $\log r = O(ns \log bs)$.

Lemma 7.1 implies that the graph of f can be recognized by an existential algebraic formula $\Psi(x, y)$ of size $m \leqslant s \cdot b$, degree at most b, with $n_1 \leqslant s - 1$ quantifiers, and with $n_0 = n$ free variables. Theorem 7.2 yields a quantifier-free algebraic formula $\Phi(x, y)$ which does the same, has size $L \leqslant (md)^{cn_0 n_1} \leqslant (sb^2)^{cns}$ and degree $D \leqslant (md)^{cn_1} \leqslant (sb^2)^{cs}$. So, the function f is r-semialgebraic for $r = \max\{L, D\} = L$, as desired. □

8. Proof of Theorem 2.3: VC Dimension of Semialgebraic Functions

Let $\mathcal{F}$ be the family of all r-semialgebraic functions $f : \mathbb{R}^n \to \mathbb{R}$. Our goal is to show that then the VC dimension $\mathrm{VCdim}(\mathcal{F})$ of $\mathcal{F}$ satisfies

$$\log_2 \binom{n+r}{n} - 1 \leqslant \mathrm{VCdim}(\mathcal{F}) = O(n \log r)\,. \tag{8}$$

The upper bound follows directly from Lemma 5.4. Namely, consider the boolean matrix $M : \mathbb{R}^{n+1} \times \mathcal{F} \to \{0, 1\}$ whose columns correspond to functions $f \in \mathcal{F}$, rows correspond to points $(x, y) \in \mathbb{R}^n \times \mathbb{R}$, and the entries are defined by: $M[(x, y), f] = 1$ if and only if $y = f(x)$. The VC dimension of this matrix is exactly the VC dimension of the family $\mathcal{F}$. Moreover, since the functions in $\mathcal{F}$ are r-semialgebraic, the matrix M is also r-semialgebraic. So, Lemma 5.4 yields the desired upper bound $O(n \log r)$ on the VC dimension of M and, hence, also on $\mathrm{VCdim}(\mathcal{F})$.

8.1. Proof of the lower bound in (8).

When proving *lower* bounds on the VC dimension $\mathrm{VCdim}(\mathcal{F})$ of families $\mathcal{F}$ of functions, it is often easier to prove a lower bound on the "dual" VC dimension, and then translate it to a lower bound on the (primal) dimension $\mathrm{VCdim}(\mathcal{F})$.

The *dual VC dimension*, $\mathrm{VCdim}^*(\mathcal{F})$, of a family $\mathcal{F}$ of functions $f : X \to Y$ is the largest number v for which there exist v points $(x_1, y_1), \dots, (x_v, y_v)$ in $X \times Y$ that are *shattered* by $\mathcal{F}$ in the following sense: for every subset $S \subseteq \{1, \dots, v\}$ there is a function $f \in \mathcal{F}$ such that

$$f(x_i) = y_i \text{ if and only if } i \in S.$$

That is, in the definition of the (primal) VC dimension $\mathrm{VCdim}(\mathcal{F})$, we shatter functions by points, whereas in the definition of the dual dimension $\mathrm{VCdim}^*(\mathcal{F})$, we shatter points by functions, which is often an easier task.

Recall that the (primal) VC dimension, $\mathrm{VCdim}(\mathcal{F})$, of $\mathcal{F}$ is the VC dimension of the *graph matrix* $M = M_{\mathcal{F}}$ of functions in $\mathcal{F}$. Rows of M correspond to points $(x, y) \in X \times Y$, columns correspond to functions $f \in \mathcal{F}$, and the entries are defined by:

$$M[(x, y), f] = 1 \text{ if and only if } y = f(x).$$

So, the dual VC dimension $\mathrm{VCdim}^*(\mathcal{F})$ of $\mathcal{F}$ is just the VC dimension of the *transpose* M^t of the graph-matrix $M = M_{\mathcal{F}}$ of $\mathcal{F}$: the rows of M^t correspond to functions $f \in \mathcal{F}$, columns correspond o points $(x, y) \in X \times Y$, and the entries of M^t are defined by:

$$M^t[f, (x, y)] = 1 \text{ if and only if } y = f(x).$$

Together with this observation, the following simple lemma immediately yields a useful inequality:

$$\mathrm{VCdim}(\mathcal{F}) \geqslant \log_2 \mathrm{VCdim}^*(\mathcal{F}) - 1\,. \tag{9}$$

Lemma 8.1 (Assouad [4]). *If the transpose of a boolean matrix M has a finite VC dimension v, then the VC dimension of M is at least $\log_2 v - 1$.*

Proof. Let $\ell := \lfloor \log_2 v \rfloor \geqslant \log_2 v - 1$, and consider the a boolean $v \times \ell$ matrix B whose *rows* are binary representations of numbers $0, 1, \ldots, 2^\ell - 1$. The matrix B has v rows of length ℓ. Since the VC dimension of the transpose of M is v, there must be v rows of M such that every binary pattern from $\{0,1\}^v$ appears as a column in these rows. In particular, this means that every column of B also appears among these rows. So, B must be a submatrix of M. Since all 2^ℓ rows of B are distinct, this implies that the VC dimension of M is at least $\ell \geqslant \log_2 v - 1$, as desired. □

The following proposition can be derived from the standard fact that every m-dimensional vector space can be written as a direct sum of its m one-dimensional subspaces. For completeness, we include a simple and direct proof suggested by Igor Sergeev (personal communication).

Lemma 8.2. *If V is an m-dimensional vector space of functions $f : X \to \mathbb{R}$, then there are m points $a_1, \ldots, a_m$ in X and m functions $f_1, \ldots, f_m$ in V such that $f_i(a_i) = 1$ and $f_i(a_j) = 0$ for all $j \neq i$.*

Since V forms a vector space, Lemma 8.2 yields: $\{(f(a_1), \ldots, f(a_m)) \colon f \in V\} = \mathbb{R}^m$.

Proof. We argue by induction on m. The basis case $m = 1$ is obvious, because then V must contain at least one nonzero function. For the induction step, take a point a_1 and a function $f_1 \in V$ such that $f_1(a_1) = 1$ (we can do this since V is a vector space of nonzero dimension). Let U be the one-dimensional vector space spanned by f_1. Associate with every function $f \in V$ the scalar $\lambda_f := f(a_1)$, and consider the family of functions $W = \{f - \lambda_f \cdot f_1 \colon f \in V\}$. This family forms a subspace of V. Since $h(a_1) = 0$ holds for all $h \in W$, $V = U \oplus W$ is a direct sum of these spaces: every function $f \in V$ can be written as a sum $\lambda_f \cdot f_1 + h$, where the function $h = f - \lambda_f \cdot f_1$ belongs to W. So, $\dim W = \dim V - \dim U = m - 1$, and the induction hypothesis gives us $m - 1$ points $a_2, \ldots, a_m$ in X and $m - 1$ functions $f_2, \ldots, f_m$ in $W \subseteq V$ such that $f_i(a_i) = 1$ and $f_i(a_j) = 0$ for all $j \neq i$. Moreover, $f_i(a_1) = 0$ for all $i = 2, \ldots, m$, since all these functions f_i belong to W. □

Proof of the lower bound in Theorem 2.3. Let $\mathcal{F}$ be the family of all r-semialgebraic functions $f : \mathbb{R}^n \to \mathbb{R}$, and let $\mathcal{P}$ be the family of all polynomials of degree at most r in $\mathbb{R}[x_1, \ldots, x_n]$. Then $\mathcal{P} \subset \mathcal{F}$, and it is enough to show the desired lower bound $\mathrm{VCdim}(\mathcal{P}) \geqslant \log_2 \binom{n+r}{n} - 1$ on the VC dimension of $\mathcal{P}$.

The family $\mathcal{P}$ of polynomials forms an m-dimensional vector space of functions $p : \mathbb{R}^n \to \mathbb{R}$ for $m = \binom{n+r}{n} \geqslant (1 + r/n)^n$. By Lemma 8.2, there are m points $a_1, \ldots, a_m$ in $\mathbb{R}^n$ and m polynomials $p_1, \ldots, p_m$ in $\mathcal{P}$ such that $p_i(a_i) = 1$ and $p_i(a_j) = 0$ for all $j \neq i$. Consider the points $(a_1, 1), \ldots, (a_m, 1) \in \mathbb{R}^{n+1}$, and take an arbitrary subset $S \subseteq \{1, \ldots, m\}$. Take the polynomial $p_S(x) = \sum_{i \in S} p_i(x)$; since the degree does not increase, this polynomial belongs to $\mathcal{P}$. For $i \in S$, we have $p_S(a_i) = p_i(a_i) = 1$, whereas for $j \notin S$, we have $p_i(a_j) = 0$ for all $i \in S$ and, hence, $p_S(a_j) = 0$. So, for every $i \in \{1, \ldots, m\}$, we have

$$p_S(a_i) = 1 \text{ if and only if } i \in S.$$

Since this holds for all subsets S, the points $(a_1, 1), \ldots, (a_m, 1)$ *are* shattered by polynomials in $\mathcal{P}$. Hence, the dual VC dimension of $\mathcal{P}$ is $\mathrm{VCdim}^*(\mathcal{P}) \geqslant m$. Inequality (9) gives the desired lower bound $\mathrm{VCdim}(\mathcal{P}) \geqslant \log m - 1$ on the primal VC dimension of $\mathcal{P}$. □

9. Proof of Theorem 2.8: recognizing roots

Recall that a probabilistic circuit $F(x, \boldsymbol{r})$ *recognizes the roots* of a given function $h : R^n \to R$ if, for every input $x \in R^n$,

$$\Pr\left\{F(x, \boldsymbol{r})^2 + h(x)^2 = 0 \text{ or } F(x, \boldsymbol{r}) \cdot h(x) \neq 0\right\} \geqslant 2/3\,.$$

That is, if $h(x) = 0$ then $F(x, \boldsymbol{r}) = 0$ with probability $\geqslant 2/3$, and if $h(x) \neq 0$ then $F(x, \boldsymbol{r}) \neq 0$ with probability $\geqslant 2/3$.

Now let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions, and containing the basis $\{+,\times\}$ or any of the four bases listed in Eq. (1) of Section 2.2. Let $h : \mathbb{R}^n \to \mathbb{R}$ be a p-semialgebraic function, and suppose that the roots of h can be recognized by a probabilistic circuit $F(x, \boldsymbol{r})$ over $\mathcal{B}$ of size s. Our goal is to show that then the roots of h can be also recognized by a deterministic circuit over $\mathcal{B}$ of size $O(n^4 s^2 \log^2 \max\{sb, p\})$.

Let k be the number of random inputs in $\boldsymbol{r}$, and consider the boolean matrix $M \colon \mathbb{R}^n \times \mathbb{R}^k \to \{0, 1\}$ with entries $M[x, r] = 1$ if and only if $F(x, r)^2 + h(x)^2 = 0$ or $F(x, r) \cdot h(x) \neq 0$. Since F recognizes the roots of h, we know that $\Pr\left\{r \in \mathbb{R}^k \colon M[x, r] = 1\right\} \geqslant 2/3$ holds for every $x \in \mathbb{R}^n$. So, the matrix M is probabilistically dense.

Recall (see Definition 3) that a boolean matrix $N : A \times B \to \{0, 1\}$ with $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^k$ is *t-semialgebraic* if for every column $b \in B$ there is an algebraic formula $\Phi_b(x)$ of size and degree at most t such that $N[a, b] = \Phi_b(a)$ holds for every row $a \in A$.

Claim 9.1. *The matrix M is t-semialgebraic for* $\log t = O(ns \log \max\{sb, p\})$.

Proof. Fix an arbitrary column of M, and let $r \in \mathbb{R}^k$ be the vector indexing this column. The circuit $F(x, r)$ is a deterministic circuit over $\mathcal{B}$ of size at most s. By Lemma 7.1, the graph of the function $f : \mathbb{R}^n \to \mathbb{R}$ computed by $F(x, r)$ can be recognized by an existential algebraic formula $\Phi_r(x, y)$ of size at most sb, degree at most b and with $s-1$ existential quantifiers. Since the function h is p-semialgebraic, there must also be an algebraic formula $\Phi_h(x, z)$ (a quantifier-free formula) of size and degree at most p recognizing the graph of h, that is, $\Phi_h(x, z) = 1$ if and only if $z = h(x)$. Consider the (quantified) formula

$$\Psi(x) := \Phi_r(x, 0) \wedge \Phi_h(x, 0) \vee \exists y \exists z \ \Phi_r(x, y) \wedge \Phi_h(x, z) \wedge [y \neq 0] \wedge [z \neq 0] \ .$$

Note that, for every row $x \in \mathbb{R}^n$ of M, $\Psi(x) = 1$ if and only if $F(x, r)^2 + h(x)^2 = 0$ or $F(x, r) \cdot h(x) \neq 0$ which, by the definition of the matrix M, happens precisely when $M[x, r] = 1$. So, it remains to show that the formula Ψ can be written as a *quantifier-free* formula Ψ' of size and the degree at most t.

The formula Ψ has size $\ell \leqslant 2sb + 2p$, degree $d \leqslant \max\{b, p\}$, and $n_1 \leqslant (s-1) + 2 = s + 1$ (existential) quantifiers. By Renegar's theorem (Theorem 7.2), the formula Ψ can be written as a quantifier-free formula Ψ' of size and degree at most $t = (\ell d)^{O(ns)}$. So, $\log t$ is at most a constant times $ns \log \max\{sb, p\}$, as desired. □

Together with Claim 9.1, Lemma 5.4 implies that the VC dimension of the matrix M is at most $v = O(n \log t) = O(n^2 s \log \max\{sb, p\})$. Since the matrix M is probabilistically dense, Lemma 5.2 implies that it must have the m-majority property for $m = O(v)$. That is, there must be some m columns $r_1, \ldots, r_m$ of M such that the (deterministic) circuit $\mathrm{maj}(F(x, r_1), \ldots, F(x, r_m))$ also recognizes the roots of our function h. So, it remains to prove the following claim: the roots of a majority vote maj function of m variables can be recognized by a circuit over $\mathcal{B}$ of size $O(m^2)$.

If $\mathcal{B}$ contains any of the four bases listed in Eq. (1), then (as shown in the proof of Corollary 2.6) even the *values* of the majority vote function can be computed using only $O(m^2)$ gates. So, we only have to prove the claim in the case when $\mathcal{B}$ only has addition and multiplication operations, that is, when we have monotone arithmetic circuits.

We will show a slightly more general fact: for every $k = 1, \ldots, m$, there is a monotone arithmetic circuit $F_{m,k}(x)$ of size $O(km)$ with the property that, for every input $x \in \mathbb{R}^m$, $F_{m,k}(x) = 0$ precisely when k or more positions in vector x are zeros. Such a circuit can be easily constructed using dynamic programming. The basis case $k = 1$ (at least one zero) is easy: just take $F_{m,1}(x_1, \ldots, x_m) = x_1 \cdot x_2 \cdots x_m$. For $k \geqslant 2$, we can use the following recursive construction:

$$F_{m,k}(x_1, \ldots, x_m) = F_{m-1,k}(x_1, \ldots, x_{m-1}) \cdot \left[F_{m-1,k-1}(x_1, \ldots, x_{m-1})^2 + x_m^2\right] \ .$$

The first term is 0 iff there are at least k zeros already among the first $m-1$ positions, whereas the second is 0 iff there are at least $k-1$ zeros among the first $m-1$ positions, and the last position is also zero; we take squares here just to avoid possible cancellations. □

Remark 4. The fact that the *roots* of the majority vote functions can be recognized by small arithmetic $\{+,\times\}$ circuits is interesting: it can be easily shown (see Appendix C) that the *values* of these functions cannot be computed by (even non-monotone) arithmetic $\{+,-,\times\}$ circuits at all.

10. Proof of Theorem 2.9: arithmetic circuits

We will obtain Theorem 2.9 as a simple consequence of Chernoff's bound and the following well know extension of the "fundamental theorem of algebra" to multivariate polynomials. This extension is usually called the "Schwartz–Zippel lemma," although its various version were earlier proved by other authors, starting by Ore [38].

Lemma 10.1 (Schwartz [42])**.** *Let f be a nonzero n-variate polynomial of degree at most $d \leqslant |\mathbb{F}|$ over a field $\mathbb{F}$, and $S \subseteq \mathbb{F}$ a finite subset of $|S| \geqslant d$ field elements. Then $|\{x \in S^n \colon f(x) = 0\}| \leqslant d|S|^{n-1}$.*

Now let $f : \mathbb{R}^n \to \mathbb{R}$ be a rational function, and suppose that f can be computed by a randomized arithmetic $\{+,-,\times,\div\}$ circuit $F(x,\boldsymbol{r})$ of size s with a positive success probability $\epsilon > 0$. Our goal is to show that some realization $F(x,r)$ of F, a deterministic $\{+,-,\times,\div\}$ circuit, must also compute f.

Since f is rational, there are two real polynomials p and q such that $f(x) = p(x)/q(x)$. Set

$$d := \max\{\deg(p), \deg(q)\} + 2^s\,,$$

and take an arbitrary subset $S \subseteq \mathbb{R}$ of size $|S| \geqslant 2d/\epsilon$.

Claim 10.2. *There is a deterministic circuit $F(x)$ of size at most s such that $F(a) = f(a)$ holds for more than $\epsilon|S|^n/2$ inputs $a \in S^n$.*

Proof. Let $A = S^n$, and take $m := \lceil 4\epsilon^{-2} \ln|A| \rceil$ independent copies of our probabilistic circuit $F(x,\boldsymbol{r})$. For a fixed input $a \in A$, let $X_{a,i}$ be the Bernoulli 0/1-random variable with $X_{a,i} = 1$ if and only if the i-th copy outputs the correct value $f(a)$ on input a. Since $\Pr\{X_{a,i} = 1\} \geqslant \epsilon$ holds for every i, the expected value μ of the sum $X_a = X_{a,1} + \cdots + X_{a,m}$ is $\mu \geqslant \epsilon m$. So, for $\alpha := \epsilon/2$, we have $\Pr\{X_a \leqslant \alpha m\} \leqslant \Pr\{X_a \leqslant \mu - \alpha m\}$. By the Chernoff bound (see, for example, [13, Theorem 1.1]), the latter probability is at most $p = \mathrm{e}^{-2\alpha^2 m} = \mathrm{e}^{-\epsilon^2 m/2} \leqslant |A|^{-2}$. By the union bound, the probability that $X_a \leqslant \alpha m$ will hold for *at least one* input $a \in A$ is at most $p \cdot |A|$, which is strictly smaller than 1. Thus, the probability that, for *every* input $a \in A$, more than αm of the m copies of our probabilistic circuit will output the correct value $f(a)$ is nonzero.

There must therefore exist m assignments $r_1, \ldots, r_m$ of constants to random inputs of $F(x,\boldsymbol{r})$ such that, on *every* input $a \in A$, more that αm of the (deterministic) circuits $F(x,r_1), \ldots, F(x,r_m)$ will output the correct value $f(a)$. By double counting, at least one of these m deterministic circuits must then output correct values $f(a)$ on more than $\alpha|A| = \frac{\epsilon}{2}|A|$ inputs $a \in A$. □

By Claim 10.2, there must be a subset $X \subseteq S^n$ of $|X| > \epsilon|S|^n/2$ input vectors and a *deterministic* arithmetic circuit F of size at most s such that $F(a) = f(a)$ holds for all $a \in X$. The circuit computes some rational function $F(x) = P(x)/Q(x)$ where P and Q are real polynomials. Since the circuit has only s gates, the degrees of these two polynomial cannot exceed 2^s. So, the degree of the polynomial

$$g(x) := p(x) \cdot Q(x) - q(x) \cdot P(x)$$

does not exceed d. We claim that g must be the zero polynomials, which implies that $F(a) = f(a)$ must hold for all inputs $a \in \mathbb{R}^n$, as desired.

So, suppose contrariwise that g is a nonzero polynomial. Then Lemma 10.1 implies that $g(a) = 0$ can hold for at most $d|S|^{n-1}$ inputs $a \in S^n$. But we know that $F(a) = f(a)$ and, hence, $g(a) = 0$ must hold for all inputs a in the set X of $|X| > \epsilon|S|^n/2$ inputs. So, $\epsilon|S|^n/2 < d|S|^{n-1}$ and, hence, also $|S| < 2d/\epsilon$ must hold, which contradicts our choice of S. □

11. BPP ⊆ P/poly for Tropical Circuits

In this section, we will show that probabilistic tropical $(\max, +)$ and $(\min, +)$ circuits can be derandomized *without* using Theorem 2.2, the proof of which is based on a rather involved quantifier elimination result of Renegar (Theorem 7.2).

In *tropical* $(\max, +)$ circuits, the domain can be any subset $R \subseteq \mathbb{R}$ closed under the sum operation (like $\mathbb{N}, \mathbb{Z}, \mathbb{R}_+$ or $\mathbb{R}$), and the basis consists of two operations $x + y$ and $\max\{x, y\}$. Since addition distributes over max operation ($x + \max\{y, z\} = \max\{x + y, x + z\}$), such a circuit solves some maximization problem with linear objective function:

$$f_A(x) = \max\{p_a(x) \colon a \in A\} \quad \text{with} \quad p_a(x) = a_1x_1 + \cdots + a_nx_n + a_{n+1}\,, \tag{10}$$

where $A \subset \mathbb{N}^n \times R$. Here $A \subset \mathbb{N}^n$ is the set of feasible solutions of this maximization problem, and inputs $x \in \mathbb{R}^n$ are assignments of weights to the n items $1, \ldots, n$. In the $(\min, +)$ circuits, we use $\min\{x, y\}$ instead of the maximum. So, such circuits solve minimization problems.

The following theorem yields $\mathsf{BPP} \subseteq \mathsf{P/poly}$ for tropical circuits, if we allow deterministic circuits to use one majority vote gate to output their values. Eliminating the need of this additional gate is an interesting open problem (see Question 1 in Section 12).

Theorem 11.1. *There is a constant c for which the following holds. If an optimization problem $f : \mathbb{R}^n \to \mathbb{R}$ can be solved by a probabilistic tropical circuit of size s, then f can be solved by a majority vote of cn^2s deterministic tropical circuits of size at most s.*

The infinite majority rule (Theorem 2.1) implies that f can be computed as a majority vote of at most $O(v)$ copies of the probabilistic circuit, where v is the VC dimension of all (deterministic) tropical circuits of size at most s. So, Theorem 11.1 directly follows from the following lemma.

Lemma 11.2. *The VC dimension of tropical circuits of size at most s in n variables is at most a constant times $n^2 \log(n + 2^s)$.*

Proof. We first consider the case of $(\max, +)$ circuits; the case of $(\min, +)$ circuits is the same: just take $\geqslant$ instead of $\leqslant$ in the formula (11) below.

Since circuits (over any semiring) of size at most s can only compute polynomials of degree at most $d = 2^s$, it is enough to show that the VC dimension of the family $\mathcal{F}$ of all max-plus polynomials (10) of degree at most d is at most a constant times $n^2 \log(n + d)$. By Theorem 2.3, it is enough to show that every tropical n-variate polynomial f of degree at most d is r-semialgebraic for $r \leqslant (n + d)^n$.

So, take such a polynomial f. It has the form (10) for some finite set $A \subset \mathbb{N}^n \times R$. Since the degree of the polynomial f does not exceed d, we have that $a_1 + \cdots + a_n \leqslant d$ must hold for all $a \in A$. We can clearly assume that every two vectors in A differ in at least one of the first n positions: if some two coincide in all these positions, then remove the polynomial p_a with the smaller "free coefficient" a_{n+1}; since we have a maximization problem, the function computed by the resulting polynomial will remain the same. So, the number $|A|$ of vectors in A cannot exceed the number $m := \binom{n+d}{n}$ of nonnegative integer solutions of $z_1 + \cdots + z_n \leqslant d$.

Consider the following algebraic formula over the real field $\mathbb{R}$, where $p_a(x)$ are linear polynomials given in (10):

$$\Phi(x,y) = \bigwedge_{a\in A} [p_a(x) \leqslant y] \wedge \bigvee_{a\in A} [p_a(x) = y] \ . \tag{11}$$

It is clear that this formula recognizes the graph of f: the first And ensures that the maximum does not exceed y, whereas the Or ensures that the value y is achieved. Since only $|A|$ distinct linear polynomials $p_a(x) - y$ are used, the size of Φ is $m = |A|$ and the degree is one. This means that our polynomial f is r-semialgebraic for $r \leqslant \max\{1, m\} = m = \binom{n+d}{n} \leqslant (n+d)^n$. □

12. Conclusion and open problems

In this paper, we dealt with the BPP versus P/poly questions for circuits working over *infinite* domains, including $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}_+$ and $\mathbb{R}$. The (unfortunate) message we deliver is that coin flipping *cannot* help much, as long as we live in a "non-uniform" world, where it is allowed to use different algorithms for inputs $x \in \mathbb{R}^n$ from differen dimensions n. This, in particularly, implies that, in this "non-uniform world", randomness is of no big use in dynamic programming.

Besides one 'big" question—to prove $\mathsf{BPP} \subseteq \mathsf{P}$ also in the *uniform* setting (cf. Remark 1), some more specific but still interesting questions also remain open.

12.1. **The rolle of majority vote gates.** Strongly speaking, our proof of $\mathsf{BPP} \subseteq \mathsf{P/poly}$ for tropical circuits (Theorem 11.1) is not fully satisfying: we allow that the derandomized (deterministic) circuit can use a majority vote function $\mathrm{maj}_m(x_1, \ldots, x_m)$ to output its values. This function is neither convex nor concave (see Appendix C). So, it *cannot* be computed by a tropical circuit at all, and it remains unclear whether such an additional gate can substantially increase the power of tropical circuits. Note, however, that maj_m can be simulated by a *sorting* gate: just sort the input, and output the $\lceil m/2\rceil$-th entry of the sorted string. Hence, the following interesting question about the power of sorting; an expected answer is *negative*, but the point is to *prove* this.

Question 1. Can one sorting gate substantially decrease the size of tropical circuits?

Let us note that, in this question, the only "dangerous" optimization problems are those, which require large tropical circuits to solve them, but whose boolean (decision) versions are easy to solve by monotone boolean circuits. To be more specific, let us consider $(\min,+)$ circuits, and assume that they are *constant-free*: inputs are only variables $x_1, \ldots, x_n$ (no constant inputs). Each such circuit solves some minimization problem $f(x) = \min_{a\in A} \sum_{i=1}^n a_i x_i$ with $A \subset \mathbb{N}^n$. The *boolean version* of this problem is the monotone boolean function $\hat{f}(x) = \bigvee_{a\in A} \bigwedge_{i:\, a_i \neq 0} x_i$. Let $\mathrm{T}(f)$ be the minimal size of a $(\min,+)$ circuit solving the problem f, and let $\mathrm{T}_{\mathrm{maj}}(f)$ denote the minimal size of a $(\min,+)$ circuit also solving f, but which can use a majority vote gate to output its values. Finally, let $\mathrm{B}(f)$ be the minimal size of monotone boolean circuit computing the boolean function $\hat{f}$. Then we have $\mathrm{B}(f) = O(t \log t)$ with $t = \mathrm{T}_{\mathrm{maj}}(f)$.

Proof. Take a $(\min,+)$ circuit F of size $t = \mathrm{T}_{\mathrm{maj}}(f)$ which solves a given minimization problem f by taking a majority vote of $(\min,+)$ circuits. Replace each min gate by an OR gate, and each + gate by an AND gate. Let $S = \mathbb{N} \cup \{+\infty\}$, and consider the mapping $h : S \to \{0,1\}$ given by $h(+\infty) := 0$ and $h(x) := 1$ for all $x \in \mathbb{R}$. This mapping h is a homomorphism from the tropical semiring $(S, \min, +)$ to the boolean $(\{0,1\}, \vee, \wedge)$ semiring: $h(\min\{x,y\}) = h(x) \vee h(y)$ and $h(x+y) = h(x) \wedge h(y)$ holds for all $x, y \in S$. So, the boolean version $F'(y)$ of $F(x)$ with inputs replaced by $y_1 = h(x_1), \ldots, y_n = h(x_n)$ computes the boolean function $\hat{f}(y)$. Indeed, by letting

$S_a=\{i\colon a_i\neq 0\}$, we have

$$h(f(x))=h\Big(\min_{a\in A}\sum_{i\in S_a}a_ix_i\Big)=\bigvee_{a\in A}h\Big(\sum_{i\in S_a}a_ix_i\Big)=\bigvee_{a\in A}\bigwedge_{i\in S_a}h(x_i)=\hat{f}(y)\,,$$

where $h(a_ix_i)=h(x_i)$ holds because $a_ix_i=x_i+x_i+\cdots+x_i$ (a_i times). The circuit F' is a monotone boolean $(\vee,\wedge)$ circuit of size t, whose output gate is a (boolean) majority function of at most t variables. Using the sorting network of Ajtai, Komlós and Szemerédi [3], the majority function of t input variables can be computed by a monotone boolean circuit of size $O(t\log t)$. □

So, the only "dangerous" in the context of Question 1 are optimization problems f with very large gaps $\mathrm{T}(f)/\,\mathrm{B}(f)$. Such is, for example, the minimum spanning tree problem f for n-vertex graphs: the Floyd–Warshall DP algorithm for graph connectivity yields $\mathrm{B}(f)=O(n^3)$, but we have shown in [32] that $\mathrm{T}(f)=2^{\Omega(n)}$. Hence, a specific question (with an expected *affirmative* answer) arises.

Question 2. Is also $\mathrm{T}_{\mathrm{maj}}(f)$ exponential for the minimum weight spanning tree problem f?

12.2. **Derandomizing tropical circuits *without* using VC dimension?** In Section 11, we have shown that probabilistic tropical $(\max,+)$ and $(\min,+)$ circuits can be derandomized *without* using deep results in quantifier elimination theory. Still, this proof uses another an (also deep) result of Haussler [24] from the statistical learning theory reducing the task of derandomization to upper-bounding the VC dimension (the "infinite majority rule," Theorem 2.1). Since tropical circuits can simulate many classical DP algorithms, it would be desirable to have a more direct derandomization arguments for these circuits.

One possible approach would be to try to only use the much simpler "finite majority rule" (Lemma 3.1). The point is that, even though tropical circuits work over *infinite* domain $\mathbb{R}^n_+$, there are *finite* sets $X\subset\mathbb{R}^n+$ which are *isolating* in the following sense: if a (deterministic) circuit computes a given function f correctly on all inputs $x\in X$, then it must compute f correctly on *all* inputs $x\in\mathbb{R}^n_+$. As shown in [30, Lemma 7], if the function is of the form $f(x)=\max\{a_1x_1+\cdots+a_nx_n\colon a\in A\}$ with $A\subseteq\{0,1\}^n$ (a 0-1 maximization problem), then already the set $X=\{0,1\}^n$ is isolating for f. In the case of minimization problems f, the (also finite) set $X=\{0,1,n+1\}^n$ (see [31, Appendix A]). The proofs of these two facts are direct and elementary. Using these facts, it *is* already possible to *directly* derandomize tropical circuits under the *one-sided error* scenario (see Appendix D).

The case of *two-sided* error is, however, more complicated. If we have a probabilistic tropical circuit solving a given optimization problem $f:\mathbb{R}^n\to\mathbb{R}$, then the finite majority rule gives us $m=O(\log|X|)=O(n)$ deterministic tropical circuits $F_1,\ldots,F_m$ (realizations of the probabilistic circuit) such that the circuit $F=\mathrm{maj}(F_1,\ldots,F_m)$ solves the problem f correctly on all input weightings $x\in X$. The problem, however, is that the function maj is neither convex nor concave (see Appendix C), and so, it cannot be computed by a tropical circuit at all. It remains therefore not clear whether the same set X remains isolating also for the obtained circuit F, that is, whether the fact that $F(x)=f(x)$ holds for all $x\in X$ (which we know) indeed implies $F(x)=f(x)$ for all $x\in\mathbb{R}^n$.

Question 3. Can probabilistic tropical circuits be derandomized using only the *finite* majority rule?

12.3. **The rolle of rounding gates.** Another interesting problem is to show that $\mathsf{BPP}\subseteq\mathsf{P/poly}$ holds for DP algorithms also when some *non-semialgebraic* operations are allowed to be used in their recursion equations. Of special interest is the *rounding* operation $\lfloor x\rfloor$ because it turned out to be useful in designing efficient *approximating* DP algorithms, say, for the knapsack problem [28]. This operations is *not* semialgebraic because its graph does not fulfill the following necessary condition for a set to be semialgebraic:

> If a set $S \subseteq \mathbb{R}^n$ is semialgebraic, then either the interior of S is nonempty, or some nonzero polynomial must vanish on all points of S.

Indeed, by observing that a system of equations $p_1(x) = \ldots = p_m(x) = 0$ is equivalent to one equation $p_1(x)^2 + \cdots + p_m(x)^2 = 0$, and that $p(x) < 0$ is the same as $-p(x) > 0$, we have that a set $S \subseteq \mathbb{R}^n$ is semialgebraic if and only if its is a finite union $S = S_1 \cup S_2 \cup \cdots \cup S_m$ of *basic semialgebraic sets*, each being of the form

$$S_i = \{x \in \mathbb{R}^n \colon p_i(x) = 0, q_{i,1}(x) > 0, \ldots, q_{i,k_i}(x) > 0\},$$

where p_i and $q_{i,j}$ are real polynomials. So, if some p_i is the zero polynomial, then S has a nonempty interior. Otherwise, $p_1 \cdot p_2 \cdots p_m$ is a nonzero polynomial vanishing on all points of S.

Now, the interior of the graph $S = \{(x, y) \in \mathbb{R} \times \mathbb{Z} \colon y = \lfloor x \rfloor\}$ of $\lfloor x \rfloor$ is clearly empty, because y can only take integer values. But the only polynomial $p(x, y) = \sum_{i=0}^{d} p_i(y) \cdot x^i$ vanishing on all points of S must be the zero polynomial. Indeed, since p vanishes on S, for every integer m, the polynomial $p(x, m)$ has an infinite number of roots $x \in [m, m+1)$; so, $p_i(m) = 0$ for all i. Since this holds for infinitely many numbers m, all polynomials $p_0, p_1, \ldots, p_d$ must be zero polynomials. Similar argument shows that some other functions, like e^x and $\sin x$, are also not semialgebraic. Hence, a specific question (with an expected *affirmative* answer).

Question 4. Does $\mathsf{BPP} \subseteq \mathsf{P/poly}$ holds for arithmetic or tropical circuits augmented with rounding gates $\lfloor x \rfloor$ and $\lceil x \rceil$?

The question is also interesting when *any* non-semialgebraic operations are allowed to be used as gates.

Appendix A. Proof of Lemma 5.2: infinite majority rule

Our goal is to show that any probabilistically dense boolean matrix with finite VC dimension v has the majority property for $m = O(v)$. We will obtain this as a direct consequence of one result of Haussler [24] (Theorem A.1 below).

For this result to hold also for matrices whose sets of rows are *uncountable*, we need a mild measurability condition, called "permissibility". Matrices with *countable* sets of rows are permissible. As shown by Haussler [24, Appendix 9.2], an arbitrary (even uncountable) family $\mathcal{H}$ of functions from some set Z to $\mathbb{R}$ is permissible if $\mathcal{H}$ can be *indexed* in the following sense: there is an integer $m \geqslant 1$ and a function (an *indexing function*) $\phi : \mathbb{R}^m \times Z \to \mathbb{R}$ such that $\mathcal{H} = \{\phi(t, \cdot) \colon t \in \mathbb{R}^m\}$; that is, the functions in $\mathcal{H}$ can be indexed using a finite number of real parameters.

Recall that a matrix $M : A \times B \to \{0, 1\}$ is *probabilistically dense* if there is a probability distribution $\Pr : B \to [0, 1]$ on columns under which

$$\mu_a := \Pr\{b \in B \colon M[a, b] = 1\} \geqslant 2/3$$

holds for every row $a \in A$. We want to randomly sample (with replacement, according to the distribution Pr) a possibly small number m of columns $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m$ so that, with large probability, the relative frequencies

$$\hat{\mu}_a := \frac{M[a, \boldsymbol{b}_1] + \cdots + M[a, \boldsymbol{b}_m]}{m}$$

of *all* rows $a \in A$ do not deviate much from their densities μ_a.

By viewing boolean matrices $M : A \times B \to \{0, 1\}$ as families $\mathcal{F} = \{f_a \colon a \in A\}$ of functions $f_a : B \to \{0, 1\}$ defined by their rows as $f_a(b) = M[a, b]$, a special case of Corollary 2 in [24] translates to the following result. (Haussler's result is more general, and applies also to non-boolean matrices.) The matrix M is *permissible* if the corresponding (to its rows) family of functions is such.

Theorem A.1 (Haussler [24])**.** *Let M be a permissible boolean matrix of finite VC dimension v, and $\epsilon, \delta > 0$. It is enough to sample*

$$m \geqslant \frac{64}{\epsilon^2}\left(2v \ln \frac{16\mathrm{e}}{\epsilon} + \ln \frac{8}{\delta}\right) \tag{12}$$

columns of M to ensure that $\Pr\{\forall a \in A\colon\ |\hat{\mu}_a - \mu_a| \leqslant \epsilon\} \geqslant \delta$.

> This kind of results are known as "uniform convergence of relative frequencies of events to their probabilities". The following version of the above theorem was proved already by Vapnik and Chervonenkis [47]: for $m \geqslant 2/\epsilon^2$, the probability of the "bad" event E that $|\hat{\mu}_a - \mu_a| > \epsilon$ holds for some row $a \in A$ does not exceed $4 \cdot \Pi_M(2m) \cdot \mathrm{e}^{-\epsilon^2 m/8}$, where $\Pi_M(m)$ is the maximum, over all choices of m columns, of the number of distinct 0-1 patterns from $\{0,1\}^m$ appearing as rows in these columns. Hence, the VC dimension of M is the maximum m for which $\Pi_M(m) = 2^m$ holds.
>
> At a very high level, the intuition behind this upper bound is that, even if the number of rows in M is *infinite*, there is only a *finite* number $\Pi_M(m)$ of their classes such that the rows in each of class have the *same* values in the sampled columns $b_1, \ldots, b_m$. The upper bound on $\Pr\{E\}$ is then obtained by reducing the problem to a finite case, and applying Hoeffding's inequality.
>
> We clearly have $\Pi_M(m) \leqslant 2^m$ for all $m \geqslant 1$, and the maximum m for which the equality holds is the VC dimension of M. This trivial upper bound is, however, exponential in m, and the above upper bound on $\Pr\{E\}$ is then trivial. Fortunately, if the number m of sampled columns is only slightly larger than the VC dimension v of M, then we have a much smaller upper bound
>
> $$\Pi_M(m) \leqslant \sum_{i=0}^{v} \binom{m}{i} \leqslant m^v\,.$$
>
> This important result is usually attributed to Sauer [41], but was discovered independently and almost simultaneously by several authors, including Shelah [44], and Vapnik and Chervonenkis [47]. Thus, if we take $m \geqslant cv \log v$, then $\Pi_M(2m) \cdot \mathrm{e}^{-\epsilon^2 m/8} \leqslant \mathrm{e}^{v \log 2m - \epsilon^2 m/8}$ goes down rapidly. Haussler's theorem eliminates the logarithmic factor $\log v$ in the condition $m \geqslant cv \log v$ and, more importantly, holds also for non-boolean matrices.

Proof of Lemma 5.2. We want to show that there is a constant c such that in every permissible and probabilistically dense boolean matrix $M : A \times B \to \{0,1\}$ of a finite VC dimension v, there are $m \leqslant c \cdot v$ columns with the property that every row has more than $m/2$ ones in these columns.

We are going to apply Haussler's theorem with $\epsilon = 1/7$ and $\delta = 1/2$. For this, take a constant c for which the right-hand of the inequality (12) for this choice of ϵ and δ is at most cv. Now pick independently $m = cv$ columns $b_1, \ldots, b_m$. According to Theorem A.1, we then have that with probability at least $1/2$, the inequality

$$\left|\frac{1}{m}\sum_{i=1}^{m} M[a, b_i] - 2/3\right| \leqslant 1/7$$

holds for all rows $a \in A$. Thus, there must be at most m columns $b_1, \ldots, b_m$ such that every row $a \in A$ has $\sum_{i=1}^{m} M[a, b_i] \geqslant (2/3 - 1/7)m > m/2$ ones in these columns. □

Appendix B. Showing that an operation is semialgebraic

When trying to show that a given operation is semialgebraic, one of the most basic facts about semialgebraic sets—the famous Tarski–Seidenberg theorem [46, 43] that projections of semialgebraic sets are semialgebraic sets—is often of great help.

When translated to the language of algebraic formulas, this theorem states that every *quantified* algebraic formula is equivalent over $\mathbb{R}$ to some *non-quantified* algebraic formula. That is, in order to show that a given function is semialgebraic, it is enough to show that its graph can be recognized by a *quantified* algebraic formula. We sketch the idea on the $\arg\max$ operation; the reader can easily find more examples, and so extend our results to more and more powerful DP algorithms.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a semialgebraic function, and $\Phi(x, u)$ some algebraic formula. An *arg max* operation is a (not uniquely defined but with a unique graph) operation $\mathsf{arg\,max}_f : \mathbb{R}^m \to \mathbb{R}$ with the property that $\mathsf{arg\,max}_f(x) = v$ if and only if v is a point in the maximization domain $\{u\colon \Phi(x, u) = 1\}$ (defined by the input x) on which f achieves its maximum. That is,

$$v = \mathsf{arg\,max}_f(x) \text{ if and only if } f(v) = \max\{f(u)\colon \Phi(x, u) = 1\}.$$

Since the function f is semialgebraic, there is an algebraic formula $F(u, y)$ which recognizes the graph of f, that is, $F(u, y) = 1$ holds precisely then when $y = f(u)$. Then the following quantified algebraic formula recognizes the graph $\{(x, v)\colon v = \mathsf{arg\,max}_f(x)\}$ of $\arg\max$ operation:

$$\Psi(x, v) = \exists y \forall u \forall w\ F(v, y) \wedge \Phi(x, u) \wedge \big(F(u, w) \wedge \Phi(x, u) \Rightarrow [w \leqslant y]\big)\,.$$

The formula $F(v, y) \wedge \Phi(x, v)$ ensures that $y = f(v)$ and v lies in the maximization domain $\{u\colon \Phi(x, u) = 1\}$ (defined by the input x), while the implication ensures that $f(v)$ is the maximum value in this domain. So, the $\arg\max$ operation is semialgebraic. Clearly, the same holds also for the arg-min operation.

Note that the quantifier-elimination result of Renegar [39] (Theorem 7.2 above) yields the following: if the graph of a function $f : \mathbb{R}^n \to \mathbb{R}$ can be recognized by a quantified algebraic formula of degree and size at most p using a constant number of quantifiers, then f is r-semialgebraic for $r = p^{O(n)}$. Recall that our general $\mathsf{BPP} \subseteq \mathsf{P/poly}$ result (Corollary 2.7) holds also when r-semialgebraic functions with $\log r = n^{O(1)}$ are allowed to be used as gates. So, the inclusion holds also when $\arg\max$ and $\arg\min$ gates are allowed.

Appendix C. Arithmetic and tropical circuits are not majority capable

Recall that a class of circuits is *majority vote capable*, if the majority vote function maj of m variables can be computed by a circuit of size polynomial in m. Our goal is to show that arithmetic and tropical circuits cannot compute the maj_m at all, not even in exponential size.

Arithmetic circuits. Consider arithmetic circuits over the basis $\mathcal{B} = \{+, -, \times\}$. Suppose we can express $\mathrm{maj}(x, y, z)$ as a polynomial $f(x, y, z) = ax + by + cz + h(x, y, z)$, where h is either a zero polynomial or has degree > 1. Then $f(x, x, z) = x$ implies $c = 0$, $f(x, y, x) = x$ implies $b = 0$, and $f(x, y, y) = y$ implies $a = 0$. This holds because, over fields of zero characteristic, equality of polynomial-functions means equality of coefficients. We have thus shown that $h = \mathrm{maj}$. So, the polynomial h cannot be the zero polynomial. But then h has degree > 1, so $h(x, x, x) = x$ for all $x \in \mathbb{R}$ is impossible. This simple argument is due to Sergey Gashkov (personal communication). □

Tropical circuits. Tropical basis $\mathcal{B} = \{\min, +\}$ (as well as $\mathcal{B} = \{\max, +\}$) is also *not* majority vote capable. Indeed, every function $f : \mathbb{R}^n \to \mathbb{R}$ computable by a circuit over $\{\min, +\}$ (a minimization problem) must be *concave*. This holds because $\min_{a\in A}\langle a, x + y\rangle \geqslant \min_{a\in A}\langle a, x\rangle + \min_{a\in A}\langle a, y\rangle$. In particular, $f(\frac{1}{2}a + \frac{1}{2}b) \geqslant \frac{1}{2}f(a) + \frac{1}{2}f(b)$ must hold for all $a, b \in \mathbb{R}^n$. But the majority vote function $f(x, y, z) := \mathrm{maj}(x, y, z)$ is not concave. To see this, take two input vectors $a = (x, x, z)$ and $b = (x, y, y)$ with $x < y$ and $z = 2x - y$. Then $f(\frac{1}{2}a + \frac{1}{2}b) = f(x, (x + y)/2, x) = x$ but $\frac{1}{2}f(a) + \frac{1}{2}f(b) = \frac{1}{2}(x + y) > x$ since $y > x$. So, f is not concave. □

Appendix D. One-sided error

Let $(R,\oplus,\otimes)$ be a commutative semiring. A circuit over R is a circuit using the semiring operations $\oplus$ and $\otimes$ as gates. Every such circuit computes some polynomial

$$f_A(x)=\sum_{a\in A}c_a\prod_{i=1}^n x_i^{a_i} \tag{13}$$

in a natural way, where $A\subset\mathbb{N}^n$ is a finite set of exponent vectors, and $c_a\in R$.

In order to investigate the one-sided error scenario, we will use the *intrinsic* (or "better-than") ordering $\leqslant_R$ in semirings defined by $a\leqslant_R b$ iff $a\oplus c=b$ for some $c\in R$. For example, if R is the boolean or the tropical $(\max,+)$ semiring, then $a\leqslant_R b$ iff $a\leqslant b$ (larger is better). In the tropical $(\min,+)$ semiring, we have $a\leqslant_R b$ iff $a\geqslant b$ (smaller is better). In the semiring R of integer-division, we have $R=\mathbb{N}$, $a\oplus b:=\mathrm{lcm}(a,b)$ (least common multiple), and $a\otimes b:=\gcd(a,b)$ (greatest common divisor). So, then $a\leqslant_R b$ iff a divides b.

Note that in idempotent semirings (where $x\oplus x=x$ holds), we have that $a\leqslant_R b$ iff $a\oplus b=b$. Indeed, if $a\oplus c=b$ for some $c\in R$, then $a\oplus b=a\oplus a\oplus c=a\oplus c=b$; the other direction is trivial. Thus, we have the following useful property of the intrinsic order in idempotent semirings:

$$\text{if } a_1\leqslant_R b,\dots,a_m\leqslant_R b \text{ and } b\in\{a_1,\dots,a_m\}, \text{ then } a_1\oplus\cdots\oplus a_m=b. \tag{14}$$

A probabilistic circuit $F(x,\boldsymbol{r})$ over R *computes* a polynomial $f\in R[x_1,\dots,x_n]$ with *one-sided success probability* ϵ (or with *one-sided error probability* $1-\epsilon$) if for every input $x\in R^n$,

$$\Pr\{F(x,\boldsymbol{r})\leqslant_R f(x)\}=1 \text{ and } \Pr\{F(x,\boldsymbol{r})=f(x)\}\geqslant\epsilon.$$

That is, the circuit is not allowed to output any better than "optimum" value, but is allowed to output "worse" values with probability $1-\epsilon$.

A set $X\subseteq R^n$ is *isolating* for a polynomial $f\in R[x_1,\dots,x_n]$, if for every polynomial g in $R[x_1,\dots,x_n]$, $g(x)=f(x)$ for all $x\in X$ implies that $g(x)=f(x)$ for all $x\in R^n$. In particular, the set $X=R^n$ is isolating for *every* polynomial over R of n variables.

Lemma D.1. *Let $f\in R[x_1,\dots,x_n]$ be a polynomial over an idempotent semiring R, and $X\subseteq R^n$ be isolating for f. If f can be computed on X by a probabilistic circuit over R of size s with a one-sided success probability $\epsilon=\epsilon(n)>0$, then f can be also computed by a deterministic circuit over R of size $O(\epsilon^{-1}s\log|X|)$.*

Proof. Let $F(x,\boldsymbol{r})$ be a probabilistic constant-free circuit over R of size s computing f on X with one-sided success probability ϵ. So, for every input $x\in R^n$, the value $F(x,\boldsymbol{r})$ can only be "smaller" than $f(x)$, but must coincide with $f(x)$ with probability at least ϵ. Take $m=\lceil\epsilon^{-1}\log|X|\rceil$ independent copies $\boldsymbol{r}_1,\dots,\boldsymbol{r}_m$ of $\boldsymbol{r}$. Then for every fixed input $x\in X$, the probability that $F(x,\boldsymbol{r}_i)\neq f(x)$ will hold for all $i=1,\dots,m$ is at most $(1-\epsilon)^m\leqslant \mathrm{e}^{-\epsilon m}$, which is strictly smaller than $1/|X|$. The union bound then implies that there must be a realizations $r_1,\dots,r_m$ of $\boldsymbol{r}_1,\dots,\boldsymbol{r}_m$ such that for every $x\in X$, $F(x,r_i)\leqslant f(x)$ holds for all i, and $F(x,r_i)=f(x)$ holds for at least one i. Fix such realizations, and consider the deterministic circuit $H(x)=F(x,r_1)\oplus\cdots\oplus F(x,r_m)$ over the same semiring R. This circuit has size $O(ms)=O(\epsilon^{-1}s\log|X|)$ and, by (14), $H(x)=f(x)$ holds for all $x\in X$. But since the set X is isolating for f, this implies $H(x)=f(x)$ for all $x\in R^n$, as desired. □

A 0-1 *maximization problem* is a function $f:\mathbb{R}_+^n\to\mathbb{R}_+$ of the form $f(x)=\max_{a\in A}\langle a,x\rangle$ for some $A\subseteq\{0,1\}^n$.

Lemma D.2. *Let $f:\mathbb{R}_+^n\to\mathbb{R}_+$ be a* 0-1 *maximization problem. If f can be solved by a probabilistic* $(\max,+)$ *circuit of size s with one-sided success probability $\epsilon>0$, then f can be also solved by a deterministic* $(\max,+)$ *circuit of size $O(ns/\epsilon)$.*

Proof. By Lemma D.1, it is enough to show that the set $X = \{0,1\}^n$ of boolean input weightings is isolating for f. So, take an arbitrary polynomial $g \in R[x_1, \ldots, x_n]$. It has the form $g(x) = \max_{b \in B} \langle b, x\rangle + c_b$, where $B \subset \mathbb{N}^n$ and $c_b \in \mathbb{R}_+$. Assume that $g(x) = f(x)$ holds for all $x \in X$. Since then $g(\vec{0}) = f(\vec{0})$ must also hold for the all-0 input vector $\vec{0} \in X$, and since $f(\vec{0}) = 0$ (f is a 0-1 maximization problem), $c_b = 0$ must hold for all $b \in B$, that is, g must be a "monic" tropical polynomial. In this case, [30, Lemma 7] implies that $g(x) = f(x)$ must hold for all $x \in \mathbb{R}^n_+$. So, the set $X = \{0,1\}^n$ is isolating for f, and the desired result follows directly from Lemma D.1. $\square$

Acknowledgments. I am thankful to Sergey Gashkov, Joshua Grochow, Pascal Koiran, Igor Sergeev and Hans Ulrich Simon for inspiring discussions. The work was supported by the German Research Foundation (DFG) under the grant JU 3105/1-1.

References

[1] L.M. Adleman. Two theorems on random polynomial time. In *19th Ann. IEEE Symp. on Foundations of Computer Sci. (FOCS)*, pages 78–83, 1978.

[2] M. Ajtai and M. Ben-Or. A theorem on probabilistic constant depth computations. In *Proc. of 16th Ann. ACM Symp. on Theory of Computing (STOC)*, pages 471–474, 1984.

[3] M. Ajtai, J. Komlós, and E. Szemerédi. Sorting in $c \log n$ parallel steps. *Combinatorica*, 3(1):1–19, 1983.

[4] P. Assouad. Densité et dimension. *Annales de l'institut Fourier*, 33:233–282, 1983.

[5] S. Basu, R. Pollack, and M.-F. Roy. On the combinatorial and algebraic complexity of quantifer elimination. *J. ACM*, 43(6):1002–1045, 1996.

[6] S. Ben-David and M. Lindenbaum. Localization vs. identification of semi-algebraic sets. *Machine Learning*, 32:207–224, 1998.

[7] C.H. Bennett and J. Gill. Relative to a random oracle A, $P^A \neq NP^A \neq$ co-NP^A with probability 1. *SIAM J. Comput.*, 10(1):96–113, 1981.

[8] P. Bürgisser, M. Karpinski, and T. Lickteig. On randomized semi-algebraic test complexity. *J. Complexity*, 9(2):231–251, 1993.

[9] J. Canny. Computing road maps in general semi-algebraic sets. *The Computer Journal*, 36:504–514, 1993.

[10] G.E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages, 2nd GI Conference, Kaiserslautern, May 20-23, 1975*, volume 33 of *Lecture Notes in Computer Science*, pages 134–183. Springer, 1975.

[11] F. Cucker, M. Karpinski, P. Koiran, T. Lickteig, and K. Werther. On real Turing machines that toss coins. In *Proc. of 27th Ann. ACM Symp. on Theory of Computing (STOC)*, pages 335–342, 1995.

[12] J. Davenport and J. Heintz. Real quantifier elimination is doubly exponential. *J. Symbolic Comput.*, 5:29–35, 1988.

[13] D. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

[14] R. Freivalds. Probabilistic machines can use less running time. In *IFIP Congress*, pages 839–842, 1977.

[15] P. Goldberg and M. Jerrum. Bounding the Vapnik-Chervonenkis dimension of concept classes parametrized by real numbers. *Machine Learning*, 18:131–148, 1995.

[16] P.W. Goldberg. *PAC-learning geometrical figures*. PhD thesis, Department of Computer Science, University of Edinburgh, UK, 1992.

[17] O. Goldreich. In a world of P = BPP. In *Studies in Complexity and Cryptography*, volume 6650 of *Lect. Notes in Comput. Sci.*, pages 191–232. Springer, 2011.

[18] D. Grigoriev. The complexity of deciding tarski algebra. *J. Symbolic Comput*, 5:65–108, 1988.

[19] D. Grigoriev. Complexity lower bounds for randomized computation trees over zero characteristic fields. *Computational Complexity*, 8(4):316–329, 1999.

[20] D. Grigoriev and M. Karpinski. Randomized $\Omega(n^2)$ lower bound for knapsack. In *29-th Ann. ACM Symp. on Theory of Computing (STOC)*, pages 76–85, 1997.

[21] D. Grigoriev, M. Karpinski, F. Meyer auf der Heide, and R. Smolensky. A lower bound for randomized algebraic decision trees. *Computational Complexity*, 6(4):357–375, 1997.

[22] D. Grigoriev and N. Vorobjov. Solving systems of polynomial inequalities in subexponential time. *J. Symbolic Comput.*, 5(1–2):37–64, 1988.

[23] D. Grigoriev and N. Vorobjov. Counting connected components of a semi-algebraic set in subexponential time. *Comput. Complexity*, 2(2):133–186, 1992.

[24] D. Haussler. Decision theoretic generalizations of the PAC model for neural nets and other learning applications. *Inf. Comput.*, 100:78–150, 1992.

[25] J. Heintz. Definability and fast quantifier elimination in algebraically closed fields. *Theor. Comput. Sci.*, 24:239–278, 1983.

[26] J. Heintz, M.-F. Roy, and P. Solernó. Sur la complexite du principe de Tarski–Seidenberg. *Bull. Soc. Math. France*, 118:101–126, 1990.

[27] J. Heintz, M.-F. Roy, and P. Solernó. Description of the connected components of a semialgebraic set in single exponential time. *Discrete and Comput. Geometry*, 11:121–140, 1994.

[28] O.H. Ibarra and J.C. Kim. Fast approximation algprithms for the knapsack and sum of subset problems. *J. ACM*, 22:463–468, 1975.

[29] R. Impagliazzo and A. Wigderson. P = BPP unless E has subexponential circuits: derandomizing the XOR lemma. In *Proc. of 29th ACM Symp. on Theory of Computing (STOC)*, pages 220–229, 1997.

[30] S. Jukna. Lower bounds for tropical circuits and dynamic programs. *Theory of Comput. Syst.*, 57(1):160–194, 2015.

[31] S. Jukna. Tropical complexity, Sidon sets and dynamic programming. *SIAM J. on Discrete Math.*, 30(4):2064–2085, 2016.

[32] S. Jukna and H. Seiwert. Greedy can also beat pure dynamic programming. Technical Report Reprt Nr. 49, Electronic Colloq. Comput. Complexity (ECCC), 2018.

[33] U. Manber and M. Tompa. The complexity of problems on probabilistic, nondeterministic, and alternating decision trees. *J. ACM*, 32(3):720–732, 1985.

[34] A.A. Markov. On the inversion complexity of systems of boolean functions. *J. ACM*, 5(4):331–334, 1958.

[35] F. Meyer auf der Heide. Simulating probabilistic by deterministic algebraic computation trees. *Theor. Comput. Sci.*, 41:325–330, 1985.

[36] J. Milnor. On the Betti numbers of real varieties. *Proc. Amer. Math. Soc.*, 15:275–280, 1964.

[37] H. Morizumi. Limiting negations in probabilistic circuits. New Trends in Algorithms and Theory of Computation, Departmental Bulletin Paper 1799, pages 81–83, Kyoto University Research Information Repository, Juni 2012.

[38] Ö. Ore. über höhere Kongruenzen. *Norsk Mat. Forenings Skrifter Ser. I*, (7), 1922. 15 pages.

[39] J. Renegar. On the computational complexity and geometry of the first-order theory of the reals. *J. Symbolic Computation*, 13:255–299, 1992.

[40] L. Rónyai, L. Babai, and M.K. Ganapathy. On the number of zero-patterns of a sequence of polynomials. *J. Amer. Math. Soc.*, 14(3):717–735, 2001.

[41] N. Sauer. On the density of families of sets. *J. Comb. Theory, Ser. A*, 13:145–147, 1972.

[42] J.T. Schwartz. Fast probabilisitic algorithms for verification of polynomila identities. *J. ACM*, 27(4):701–717, 1980.

[43] A. Seidenberg. A new decision method for elementary algebra. *Ann. of Math.*, 60:365–374, 1954.

[44] S. Shelah. A combinatorial problem; stability and order for models and theories in infinite languages. *Pacific J. Math.*, 41(1):271–276, 1972.

[45] M. Snir. Lower bounds on probabilistic linear decision trees. *Theor. Comput. Sci.*, 38:69–82, 1985.

[46] A. Tarski. *A decision method for elementary algebm and geometry.* University of California Press, Berkeley and Los Angeles, Calif., 2nd edition, 1951.

[47] V.N. Vapnik and A.Ya. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16:264–280, 1971.

[48] H.E. Warren. Lower bounds for approximation by non-linear manifolds. *Trans. Amer. Math. Soc.*, 133:167–178, 1968.

[49] V. Weispfenning. The complexity of linear problems in fields. *J. Symbolic Comput.*, 5:3–27, 1988.

Institut für Informatik, Goethe Universität, Frankfurt am Main, Germany