Coin Flipping in Dynamic Programming is Almost Useless

STASYS JUKNA*, Vilnius University, Lithuania

We consider probabilistic circuits working over the real numbers, and using arbitrary semialgebraic functions of bounded description complexity as gates. In particular, such circuits can use all arithmetic operations $+, -, \times, \div$, optimization operations min and max, conditional branching (if-then-else), and many more. We show that probabilistic circuits using any of these operations as gates can be simulated by deterministic circuits with only about a quadratical blowup in size. A not much larger blow up in circuit size is also shown when derandomizing approximating circuits. The algorithmic consequence, motivating the title, is that randomness cannot substantially speed up dynamic programming algorithms.

CCS Concepts: • **Theory of computation → Probabilistic computation**; **Circuit complexity**; Algebraic complexity theory.

Additional Key Words and Phrases: Derandomization, dynamic programming, semialgebraic functions, sign patterns of polynomials

ACM Reference Format:
Stasys Jukna. 0. Coin Flipping in Dynamic Programming is Almost Useless. *ACM Trans. Comput. Theory* 0, 0, Article 0 (0), 26 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Probabilistic algorithms can make random choices during their execution. Often, such algorithms are more efficient than *known* deterministic solutions; see, for example, the books [23, 25]. So, a natural questions arises: is randomness a really useful resource, can randomization indeed substantially speed up algorithms? In the computational complexity literature, this is the widely open[1] "BPP versus P" question. The nonuniform version of this question, known as the "BPP versus P/poly," question asks whether probabilistic *circuits* can be efficiently simulated by deterministic circuits.

A *probabilistic* circuit is a deterministic circuit that is allowed to use additional input variables, each being a *random variable* taking its values in the underlying domain. We allow arbitrary probability distributions of these random variables, so that our derandomization results will be distribution independent. Such a circuit *computes* a given function f if, on every input x, the circuit outputs the correct value $f(x)$ with probability at least[2] $2/3$. The *size* of a (deterministic or probabilistic) circuit is the number of used gates.

A classical result of Adleman [1], extended to the case of two-sided error probability by Bennett and Gill [7], has shown that randomness is useless in *Boolean* circuits: if a Boolean function f of n variables can be computed by a probabilistic Boolean circuit of size polynomial in n, then f can be

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

[1] BPP stands for "**b**ounded-error **p**robabilistic **p**olynomial time," and P for "deterministic **p**olynomial time."

[2] 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 in our bounds, all results will hold also then.

Author's address: Stasys Jukna, Faculty of Mathematics and Computer Science, Vilnius University, Lithuania.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2020 Association for Computing Machinery.
1942-3454/0/0-ART0 $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

also computed by a deterministic Boolean circuit of size polynomial in n. So, BPP $\subseteq$ P/poly holds for Boolean circuits.

In this paper, we are mainly interested in the BPP versus P/poly question for *dynamic programming* algorithms (DP algorithms):

- *Can randomization substantially speed up DP algorithms?*

We answer this question in the *negative*: randomized DP algorithms *can* be derandomized. That is, BPP $\subseteq$ P/poly holds also for DP algorithms. In fact, we prove a much stronger result: BPP $\subseteq$ P/poly holds for circuits over *any* basis consisting of semialgebraic operations $g : \mathbb{R}^l \to \mathbb{R}$ of bounded algebraic description complexity. We will also show that the inclusion BPP $\subseteq$ P/poly holds even when circuits are only required to *approximate* the values of given functions.

Proofs of BPP $\subseteq$ P/poly for *Boolean* circuits in [1, 7] crucially used the fact that the domain $\{0,1\}$ of such circuits is *finite*: the proof is then obtained by a simple application of the union and Chernoff's bounds (see Lemma 2 in Section 6). A trivial reason why such a simple argument cannot derandomize DP algorithms is that these algorithms work over *infinite* domains such as $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$ (inputs for optimization problems), so that already the union bound badly fails.

One also faces the "infinite domain" issue, say, in the polynomial identity testing problem over infinite fields; see, for example, surveys [28, 31]. But when derandomizing DP algorithms, we additionally face the "non-arithmetic basis" issue: besides arithmetic $+, -, \times, \div$ operations, such circuits can use additional non-arithmetic operations, like tropical min and max operations, sorting, conditional branching (if-then-else), argmin, argmax, and other complicated operations.

To nail all this (infinite domain and powerful gates), in this paper, we consider the derandomization of circuits that can use *any* semialgebraic functions of bounded description complexity as gates.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is *semialgebraic* if its graph can be obtained by finitely many unions and intersections of sets defined by a polynomial equality or strict inequality. The *description complexity* of f is the minimum number t for which such a representation of the graph of f is possible by using at most t distinct polynomials, each of degree at most t (see Section 3 for precise definitions). All operations mentioned above are semialgebraic of small description complexity; see Table 1 in Section 3 for more examples.

Derandomization of exactly computing circuits. The *majority vote* function is a partly defined function $\mathrm{Maj}(x_1, \ldots, x_m)$ which outputs the majority element of its input string, if there is one. That is,

$$\mathrm{Maj}(x_1, \ldots, x_m) = y \text{ if } y \text{ occurs} > m/2 \text{ times among the } x_1, \ldots, x_m. \tag{1}$$

For example, in the case of $m = 5$ variables, we have $\mathrm{Maj}(a, b, c, b, b) = b$, whereas the value of $\mathrm{Maj}(a, b, c, a, b)$ is undefined. The function $\mathrm{Maj}(x_1, \ldots, x_m)$ is b-semialgebraic for $b \leq m$; see Table 1 in Section 3.

A *deterministic copy* of a probabilistic circuit is a deterministic circuit obtained by fixing the values of its random input variables. A (deterministic or probabilistic) circuit is *b-semialgebraic* if each its basis operation (a gate) is b-semialgebraic. Note that b here is a *local* parameter: it bounds the description complexity of only *individual* gates, not of the entire function computed by the circuit. For example, circuits using any of the gates $+, -, \times, \div$, min, max, "if $x < y$ then u else v" are b-semialgebraic for $b \leq 3$, and the majority vote gate $\mathrm{Maj}(x_1, \ldots, x_m)$ is b-semialgebraic for $b \leq m$ (see Table 1).

Theorem 1. *If a function $f : \mathbb{R}^n \to \mathbb{R}$ can be computed by a probabilistic b-semialgebraic circuit of size s, then f can be also computed as a majority vote of $m = O(n^2 s \log bs)$ deterministic copies of this circuit.*

Note that, even though the majority vote functions are only *partially* defined, the derandomized circuit 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* (for every input x to the entire circuit) contain a majority element.

Note also that the upper bound on the number m of deterministic copies in the derandomized circuit only depends on the number n of deterministic input variables, on the number s of gates in the probabilistic circuit, and on the (logarithm of) the description complexity b of individual gates. But it depends neither on the fanin of gates, nor on the number of random input variables.

Derandomization of approximating circuits. Our next (and main) result derandomizes probabilistic circuits when they are only required to *approximate* the values of a given function (instead of computing the function exactly, as in Theorem 1).

Let $x \varrho y$ be any binary relation between real numbers $x, y \in \mathbb{R}$. One may interpret $x \varrho y$ (especially, in the context of approximating algorithms) as "x lies close to y." The description complexity of the relation ϱ is the description complexity of the set $S = \{(x, y) \in \mathbb{R}^2 : x \varrho y\}$.

Given a binary relation $x \varrho y$ between real numbers, we say that a probabilistic circuit $F(x, \boldsymbol{r})$ *ϱ-approximates* a given function $f(x)$ if, for every input $x \in \mathbb{R}^n$, $F(x, \boldsymbol{r}) \varrho f(x)$ holds with probability at least $2/3$. That is, on every input x, the circuit only has to output a value which is "close enough" to the correct value $f(x)$ with probability at least $2/3$.

Example 1. Some of the most basic relations are the following ones.

(1) Equality relation: $x \varrho y$ iff $x = y$.
(2) Sign relation: $x \varrho y$ iff $x = y = 0$ or $x \cdot y > 0$.
(3) Nullity relation: $x \varrho y$ iff $x = y = 0$ or $x \cdot y \neq 0$.
(4) Approximation relation: $x \varrho y$ iff $|x - y| \leq c$ for some fixed number $c \geq 0$.

In the case of approximating circuits, the first relation (1) corresponds to computing the values $f(x)$ exactly, as in Theorem 1. The second relation (2) corresponds to detecting signs of the values $f(x)$. In the case of relation (3), a circuit must recognize the roots of f, that is, must output 0 precisely when $f(x) = 0$. In the case of the last relation (4), the values computed by the circuit must lie not far away from the correct values $f(x)$.

A *majority ϱ-vote function* is a (partial) function $\mu : \mathbb{R}^m \to \mathbb{R}$ with the following property for any real numbers $a, x_1, \ldots, x_m$:

$$\text{if } x_i \varrho a \text{ holds for more than } m/2 \text{ positions } i, \text{ then } \mu(x_1, \ldots, x_m) \varrho a \text{ holds.} \tag{2}$$

That is, if more than half of the input numbers $x_1, \ldots, x_m$ lie close to the number a, then also the value of μ must lie close to a. For example, the majority vote function Maj is the unique majority ϱ-vote function for the equality relation (when $x \varrho y$ iff $x = y$). In general, however, there may be more than one majority ϱ-vote function. For example, for a function $\mu : \mathbb{R}^m \to \mathbb{R}$ to be a majority ϱ-vote function for the nullity relation ϱ it is enough that $\mu(x) = 0$ if more than half of input numbers are zeros, and $\mu(x) \neq 0$ otherwise.

In the following theorem, $x \varrho y$ is an arbitrary t_ϱ-semialgebraic relation, and $f : \mathbb{R}^n \to \mathbb{R}$ a t_f-semialgebraic function.

Theorem 2 (Main result). *If f can be ϱ-approximated by a probabilistic b-semialgebraic circuit of size s, then f can be also ϱ-approximated as a majority ϱ-vote of $m = O(n^2 s \log K)$ deterministic copies of this circuit, where $K = sb + t_f + t_\varrho$.*

Note that now (unlike in Theorem 1) the size of the derandomized circuit depends (albeit only logarithmically) on the description complexities t_f and t_ϱ of the function f approximated and of the approximation relation ϱ. Although t_f may be large, the description complexity of the

approximation relations ϱ is usually small; say, for all four relations mentioned in Example 1 we have $t_\varrho \le 2$.

The majority vote "issue". One issue still remains: just like in Theorem 1, the deterministic circuits given by Theorem 2 are not in a "pure" form: they require one additional majority ϱ-vote operation to output their values. To obtain a "pure" circuit, we have to compute this operation by a (possibly small) circuit using only basis operations.

In some weak bases, even the (standard) majority vote function (1) cannot be computed at all. For example, arithmetic $(+,-,\times)$ circuits, as well as tropical $(\min,+)$ and $(\max,+)$ circuits cannot compute majority vote functions (Claim 6 in Appendix A). In most bases, however, majority vote functions are easy to compute.

Call a relation $x\,\varrho\,y$ *contiguous* if $x \le y \le z$, $x\,\varrho\,a$ and $z\,\varrho\,a$ imply $y\,\varrho\,a$. That is, if the endpoints of an interval are close to a, then also all numbers in the interval are close to a. Note that the relations (1), (2) and (4) mentioned in Example 1 are contiguous. It can be easily shown (see Claim 7 in Appendix A) that:

- For every contiguous relation $x\,\varrho\,y$, a majority ϱ-vote function of m variables can be computed by a fanin-2 $(\min,\max)$ circuit of size $O(m\log m)$.

The nullity relation is *not* contiguous: take, for example, $x=-1, y=0$ and $z=a=1$. Then $x \le y \le z$, $x\,\varrho\,a$ and $z\,\varrho\,a$ hold but $y\,\varrho\,a$ does not hold: $y=0$ but $a\neq 0$. Still, majority ϱ-vote function of m variables for the nullity relation can be also computed by a $(\min,\max,\times)$ circuit using $O(m\log m)$ gates, and by a monotone arithmetic $(+,\times)$ circuit using $O(m^2)$ gates (Claim 8 in Appendix A).

Thus, if the approximation relation is contiguous, and if the operations $(+,\times)$ or the operations $(\min,\max,\times)$ are available as gates, then Theorem 2 gives a "pure" circuit (without a majority vote gate) of size $O(ms + m\log m) = O(n^2 s^2 \log K)$, that is:

- The blow up in the size of the derandomized circuit is only about quadratic.

Remark 1 (Relation to dynamic programming). 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 of small description complexity, like the selection or the "if-then–else" operations (see Table 1 in Section 3). So, Theorem 1 implies that randomization is (almost) *useless* in DP 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. In fact, the message of this paper is even stronger: Theorem 2 shows that randomization is almost useless also for *approximating* DP algorithms.

Remark 2 (The "uniformity" issue). Usually, a DP algorithm is described by giving *one* collection of recursion equations that can be applied to inputs of *any* dimension n. In this respect, DP algorithms are "uniform" (like Turing machines). Probabilistic DP algorithms may use random input weights in their recursion equations. However, when derandomizing such algorithms, we do not obtain also *one* collection of recursion equations valid for inputs of *all* dimensions. What we obtain is a *sequence* of deterministic DP algorithms, one for each dimension n. To our best knowledge, in the "uniform" setting (with P instead of P/poly), the inclusion BPP $\subseteq$ P remains *not* known to hold for DP algorithms, and even for "pure" DP algorithms using only $(\min,+)$ or $(\max,+)$ operations in their recursion equations.

Organization. Section 2 shortly summarizes previous work towards derandomization of probabilistic circuits working over infinite domains. In Section 3, we recall the notions of semialgebraic functions

and probabilistic circuits. Section 4 describes the three steps in which we will come from probabilistic to deterministic circuits. The next three sections (Sections 5 and 7) contain technical results used to implement these three steps. After these technical preparations, Theorem 1 is proved in Section 8, and Theorem 2 is proved in Section 9. In the last section (Section 10), we show that probabilistic arithmetic and tropical circuits can be also derandomized using elementary arguments by using so-called "isolating sets" for arithmetic and tropical polynomials.

2 RELATED WORK

As we mentioned at the beginning, our starting point is the result of Adleman [1] that[3] 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 a simple "finite majority rule" (Lemma 2 in Section 6). This rule follows directly from the Chernoff and union bounds, and allows us to simulate any probabilistic circuit of size s on n input variables taking their values in a *finite* domain D as a majority vote of $O(n \log |D|)$ deterministic circuits, each of size at most s.

In the *Boolean* case, the domain $D = \{0, 1\}$ is clearly finite, and the majority vote functions turn into Boolean majority functions: output 1 if and only if more than half of the input bits are 1s. Since majority functions have small Boolean circuits, even monotone ones, the resulting deterministic circuits are then not much larger than the probabilistic ones, is only $O(ns)$.

Using entirely different arguments (not relying on the finite majority rule), Ajtai and Ben-Or [2] have shown that BPP $\subseteq$ P/poly holds also for Boolean constant-depth circuits, known also as AC^0 circuits. Note that this extension is far from being trivial, because the majority function itself requires AC^0 circuits of exponential size.

Markov [20] has found a surprisingly tight combinatorial characterization of the minimum number of NOT gates required by *deterministic* $(\vee, \wedge, \neg)$ circuits to compute a given Boolean functions f. A natural question therefore was: can randomness substantially reduce the number of NOT gates? Morizumi [24] has shown that Markov's result itself already gives a negative answer: in probabilistic circuits, the decrease of the number of NOT gates is at most by an *additive* constant, where the constant depends only on the success probability.

The derandomization of circuits working over *infinite* domains D, such as $\mathbb{N}$, $\mathbb{Z}$ or $\mathbb{R}$, is a more delicate task. Here we have to somehow "cope" with the infinity of the domain: Chernoff's and union bounds alone do not help then. Two general approaches emerged along this line of research.

(A) Find (or just prove a mere existence of) a *finite* set $X \subset D^n$ of input vectors that 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 $x \in D^n$. Then use the finite majority rule on inputs from X.

(B) Use the "infinite majority rule" (Lemma 4 below) following from the uniform convergence in probability results, proved by researchers in the statistical learning theory.

Approach (A) was used by many authors to show the inclusion BPP $\subseteq$ 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 for branching, the inclusion BPP $\subseteq$ P/poly was proved by Manber and Tompa [19], and Snir [32]. Meyer auf der Heide [21] proved the inclusion BPP $\subseteq$ P/poly for the decision tree depth when arbitrary rational functions are allowed. He uses a result of Milnor [22] about the number of connected components of polynomial systems in $\mathbb{R}^n$ to upper-bound the minimum

[3] 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."

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 [15], Grigoriev et. al. [16], Grigoriev [14] and other authors.

Approach (B) was used by Cucker et. al. [9] to prove the inclusion BPP $\subseteq$ P/poly for circuits over the basis $(+, -, \times, \div, \mathrm{sgn})$, that is, for arithmetic $(+, -, \times, \div)$ circuits with signum gates. They combined the upper bound on the Vapnik–Chervonenkis dimension (VC dimension) of such circuits, obtained by Goldberg and Jerrum [12], with a uniform convergence in probability theorem of Haussler [17] for classes of functions with bounded VC dimension. In the proofs of Theorems 1 and 2 we will also use Approach (B), but in a more *direct* way avoiding the detour through VC dimension and Sauer's lemma: we will directly combine the classical uniform convergence in probability theorem of Vapnik and Chervonenkis [34] with the upper bound of Warren [35] on the number of sign patterns of real polynomials.

The BPP vs. P problem in the *uniform* setting, that is, in terms of Turing machines (instead of circuits), is an even more delicate task. Still, a strong indication that BPP = P "should" hold also in the uniform setting was given by Impagliazzo and Wigderson [18]: either BPP = P holds or *every* decision problem solvable by deterministic Turing machines in time $2^{O(n)}$ can be solved by a Boolean circuit of sub-exponential size $2^{o(n)}$. Goldreich [13] 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 direction.

3 PRELIMINARIES

In this section, we define more precisely the concepts used in the paper (probabilistic circuits, algebraic formulas, description complexity of sets and functions), and recall two known results used in the proof of our main results (Theorems 1 and 2).

3.1 Semialgebraic sets and functions

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?

The *sign function* $\mathrm{sgn} : \mathbb{R} \to \{-1, 0, +1\}$ takes value $\mathrm{sgn}\, x = -1$ if $x < 0$, $\mathrm{sgn}\, 0 = 0$, and $\mathrm{sgn}\, x = +1$ if $x > 0$. Let $P = (p_1, \ldots, p_m)$ be a sequence of polynomials in $\mathbb{R}[x_1, \ldots, x_n]$. The *sign pattern* of this sequence at a point $x \in \mathbb{R}^n$ is the vector

$$\mathrm{sgn}\, P(x) = \big(\mathrm{sgn}\, p_1(x), \ldots, \mathrm{sgn}\, p_m(x)\big) \in \{-1, 0, +1\}^n$$

of signs taken by these polynomials at the point x.

A set $S \subseteq \mathbb{R}^n$ is *t-semialgebraic* if there is a sequence $P = (p_1, \ldots, p_m)$ of $m \leq t$ polynomials of degree at most t such that the membership of points $x \in \mathbb{R}^n$ in the set S can be determined from sign patterns of these polynomials at these points, that is, if $x \in S$ and $x' \notin S$, then $\mathrm{sgn}\, P(x) \neq \mathrm{sgn}\, P(x')$.

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *t-semialgebraic* if its graph $S = \{(x, y) : y = f(x)\} \subseteq \mathbb{R}^{n+m}$ is such. The *description complexity* of a semialgebraic set (or function) is the smallest number t for which this set (or function) is t-semialgebraic.

3.2 Algebraic formulas

The description complexity of sets and functions can be defined more explicitly using the language of "algebraic formulas." An *algebraic formula* is an arbitrary Boolean combination of atomic predicates, each being of the form $[p(x) \diamond 0]$ for some polynomial p in $\mathbb{R}[x_1, \ldots, x_n]$, where $\diamond$ is one of the standard relations $>, \geq, =, \neq, \leq, <$, 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 $[p(x) \diamond q(x)]$ is equivalent to $[p(x) - q(x) \diamond 0]$, so that we can also make comparisons between polynomials. The *description complexity* of an algebraic formula is $\max\{m, d\}$, where m is the number of distinct polynomials used in the formula, and d is their maximal degree. An algebraic formula $\Phi(x)$ *defines* a set $S \subseteq \mathbb{R}^n$ if $S = \{x \in \mathbb{R}^n : \Phi(x) = 1\}$.

Claim 1. *For every algebraic formula there is a algebraic formula of the same description complexity which only uses atomic predicates of the form* $[p < 0]$, $[p = 0]$ *and* $[p > 0]$.

The claim is trivial: just replace each atomic predicate $[p \leq 0]$ by the formula $[p = 0] \vee [p < 0]$, each atomic predicate $[p \geq 0]$ by the formula $[p = 0] \vee [p > 0]$, and each atomic predicate $[p \neq 0]$ by the formula $[p < 0] \vee [p > 0]$. Neither the number of distinct polynomials used, nor their degree increases during these transformations.

Claim 2. *The description complexity of a semialgebraic set is the minimum description complexity of an algebraic formula defining this set.*

In the literature, this fact is often used as the *definition* of the description complexity of sets.

Proof. Let $S \subseteq \mathbb{R}^n$ be a set of vectors. Our goal is to show that the description complexity of S is at most t if and only if the set S can be defined by an algebraic formula Φ of description complexity at most t.

($\Leftarrow$) By Claim 1, we can assume that only atomic predicates of the form $[p < 0]$, $[p = 0]$ and $[p > 0]$ are used in the formula Φ. Hence, the values of the formula Φ only depend on the sign patterns of the sequence $P = (p_1, \ldots, p_m)$ of all $m \leq t$ polynomials of degree at most t used in the formula Φ.

($\Rightarrow$) Let $P = (p_1, \ldots, p_m)$ be a sequence of $m \leq t$ polynomials of degree at most t such that the membership of points $x \in \mathbb{R}^n$ in the set S can be determined from sign patterns of these polynomials on these points. Consider the $s = 3m$ functions $g_i : \mathbb{R}^n \to \{0, 1\}$ defined by: $g_i = [p_i < 0]$ for $1 \leq i \leq m$, $g_i = [p_i = 0]$ for $m + 1 \leq i \leq 2m$, and $g_i = [p_i > 0]$ for $2m + 1 \leq i \leq 3m$.

We know that for every two points $x \in S$ and $x' \notin S$, $\operatorname{sgn} P(x) \neq \operatorname{sgn} P(x')$ must hold. In particular, this means that the operator $G = (g_1, \ldots, g_s) : \mathbb{R}^n \to \{0, 1\}^s$ cannot take the same value (output the same vector) on any pair of points $x \in S$ and $x' \notin S$. Thus, there is a Boolean function $f : \{0, 1\}^s \to \{0, 1\}$ such that, for every $x \in \mathbb{R}^n$, $f(G(x)) = 1$ holds precisely when $x \in S$. It remains to take *any* Boolean formula $F(y_1, \ldots, y_s)$ computing the function f, replace its inputs y_i by the corresponding atomic predicates $g_i(x)$, and the resulting algebraic formula Φ then defines the set S. The number of *distinct* polynomials used by the formula Φ is $m \leq t$ (note that in atomic predicates $[p_i < 0]$, $[p_i = 0]$ and $[p_i > 0]$, the *same* polynomial p_i is used), and their degree is at most t. The actual size of the Boolean formula F itself (number of gates in it) is irrelevant: important only is that the algebraic formula Φ uses at most t distinct polynomials of degree at most t. □

By Claim 2, a function $f : \mathbb{R}^n \to \mathbb{R}$ is t-semialgebraic if there is an algebraic formula $\Phi(x, y)$ of description complexity at most t such that for every $x \in \mathbb{R}^n$ and $y \in \mathbb{R}$, $\Phi(x, y) = 1$ holds precisely when $y = f(x)$. Table 1 gives a sample of some basic semialgebraic functions of small description complexity.

Let us stress that, in algebraic formulas, we only count the number of *distinct* polynomials used, *not* the number of their *occurrences* in the formula: one and the same polynomial can appear many times, and under different relations $\diamond$.

Namely, a set $S \subseteq \mathbb{R}^n$ is t-semialgebraic if there is a Boolean function $\varphi(y_1, \ldots, y_{3t})$ of $3t$ Boolean variables, and there are $m \leq t$ polynomials $p_1, \ldots, p_m$, each of degree $d \leq t$, such that for every

Table 1. Examples of semialgebraic functions, where m is the number of distinct polynomials used in a formula, and d is their degree, that is, the largest sum of degrees of variables appearing in any monomial of these polynomials. Here, $p(x)$ is an arbitrary real multivariate polynomial of degree d, and $\Psi(x)$ is a algebraic formula using s polynomials of maximum degree $d \geq 1$; $\mathrm{Sel}(x_1,\ldots,x_n\,|\,y)$ is a partly defined function that outputs x_i if $y=i$. In the algebraic formulas for the majority vote functions, maj is the Boolean majority function.

Graph of f	(m,d)	Algebraic formula Φ
$z=p(x)$	$(1,d)$	$[z=p(x)]$
$z=\lvert x\rvert$	$(3,1)$	$([x\geq 0]\wedge[z=x])\vee([x<0]\wedge[z=-x])$
$z=x^{1/k}$	$(2,k)$	$\left[x=z^k\right]$ (odd k), $[x\geq 0]\wedge\left[x=z^k\right]$ (even k)
$z=\lVert x-y\rVert$	$(2,2)$	$[z\geq 0]\wedge\left[z^2=(x_1-y_1)^2+\cdots+(x_n-y_n)^2\right]$
$z=x/y$	$(2,2)$	$[y\neq 0]\wedge[y\cdot z=x]$
$z=\min(x,y)$	$(2,1)$	$[z\leq x]\wedge[z\leq y]\wedge([z=x]\vee[z=y])$
$z=\max(x,y)$	$(2,1)$	$[z\geq x]\wedge[z\geq y]\wedge([z=x]\vee[z=y])$
$z=\mathrm{Maj}(x_1,\ldots,x_n)$	$(n,1)$	$\mathrm{maj}\left([z=x_1],\ldots,[z=x_n]\right)$
$z=\mathrm{Sel}(x_1,\ldots,x_n\lvert y)$	$(2n,1)$	$\bigvee_{i=1}^{n}[y=i]\wedge[z=x_i]$
$z=$ "if $\Psi(x)=1$ then u else v"	$(s+2,d)$	$(\Psi(x)\wedge[z=u])\vee(\neg\Psi(x)\wedge[z=v])$

point $x\in\mathbb{R}^n$, we have $x\in S$ precisely when

$$\varphi\big([p_1(x)<0],[p_1(x)=0],[p_1(x)>0],\ldots,[p_m(x)<0],[p_m(x)=0],[p_m(x)>0]\big)=1\,.$$

The actual Boolean circuit (or formula) *complexity* of φ is irrelevant here: only the number m of distinct polynomials p_i used and their maximum degree d do count. We use algebraic formulas just to *explicitly* define the corresponding Boolean functions φ.

Example 2 (Sorting operation). The *sorting operation* $\mathrm{sort}:\mathbb{R}^n\to\mathbb{R}^n$ takes a sequence $x_1,\ldots,x_n$ of real numbers, and outputs its ordered permutation $y_1\leq y_2\leq\ldots\leq y_n$. The graph of this operation can be defined by the following algebraic formula of $2n$ variables:

$$\Phi(x,y)=\bigwedge_{i=1}^{n-1}[y_i\leq y_{i+1}]\wedge\Big(\bigvee_{\sigma\in S_n}\bigwedge_{i=1}^{n}\big[y_i=x_{\sigma(i)}\big]\Big),$$

where S_n is the set of all permutations of $\{1,\ldots,n\}$. The total number of occurrences of atomic predicates in this formula (the "size" of the formula) is huge (is even larger than $n!$), but the formula only uses $m=n^2+n-1$ distinct polynomials $y_{i+1}-y_i$ for $i=1,\ldots,n-1$, and y_i-x_j for $i,j=1,\ldots,n$ of degree $d=1$. Thus, the sorting operation $\mathrm{sort}:\mathbb{R}^n\to\mathbb{R}^n$ is t-semialgebraic for $t=n^2+n-1$.

3.3 Quantifier elimination

A *quantified algebraic formula* $\Psi(x)$ with n free variables $x=(x_1,\ldots,x_n)$ is of a form

$$(Q_1\vec{z}_1\in\mathbb{R}^{k_1})\ \ldots\ (Q_\omega\vec{z}_\omega\in\mathbb{R}^{k_\omega})\ \Phi(x,\vec{z}_1,\ldots,\vec{z}_\omega)\,, \tag{3}$$

where $Q_i\in\{\exists,\forall\}$, $Q_i\neq Q_{i+1}$, and Ψ is an algebraic formula using m polynomials of degree at most d. That is, we have ω alternating blocks of quantifiers with k_i quantified variables in the ith block. Let $K=\prod_{i=1}^{\omega}k_i$.

By the classical Tarski–Seidenberg theorem [30, 33], every *quantified* algebraic formula has an equivalent quantifier-free formula. Thus, a set is semialgebraic if it can be defined by a quantified

algebraic formula. But the description complexity of the resulting quantifier formula resulting from [30, 33] was huge, was not even bounded by any function which is a tower of exponents (in the input parameters n, m, d and K) of a fixed height. In a series of works by many authors, the blow up in description complexity was vastly decreased. In particular, the following upper bound was proved in [5].

Theorem 3 (Basu, Pollack and Roy [5]). *For every quantified formula Eq. (3) there is an equivalent quantifier-free algebraic formula $\Phi(x_1, \ldots, x_n)$ which uses at most $(sd)^{O(nK)}$ polynomials of degree at most $d^{O(K)}$.*

We will only use a very special consequence of this result for *existential* algebraic formulas, that is, for formulas Eq. (3) with $\omega = 1$, $Q_1 = \exists$ and $k_1 = q$.

Corollary 1. *If $S = \{x \in \mathbb{R}^n : (\exists z \in \mathbb{R}^q)\ \Phi(x, z) = 1\}$ for some quantifier-free algebraic formula Φ of description complexity κ, then S is t-semialgebraic for $\log t = O(nq \log \kappa)$.*

3.4 Number of sign patterns

By the definition, a set $S \subseteq \mathbb{R}^n$ is t-semialgebraic if the membership of points $x \in \mathbb{R}^n$ in S can be determined from seeing the sign patterns of some fixed sequence of t polynomials of degree at most t on these points x. So, a natural question arises: how many distinct sign patterns a given sequence of m polynomials on n variables can have? A trivial upper bound is $|\{-1, 0, +1\}^m| = 3^m$.

A fundamental result of Warren [35, Theorem 3] shows that, when we have more than n polynomials of bounded degree, then the critical parameter is not their number m but rather the number n of variables.

Theorem 4 (Warren [35]). *No sequence of $m \geq n$ polynomials in $\mathbb{R}[x_1, \ldots, x_n]$ of degree at most $d \geq 1$ can have more than $(8emd/n)^n$ distinct sign patterns.*

What Warren actually proved is the upper bound $(4emd/n)^n$ on the number of sign patterns lying in the set $\{-1, +1\}^n$. But as observed in [4, 12, 27], by "doubling" each polynomial, this bound can be easily extended to the upper bound $(8emd/n)^n$ on the number of *all* sign patterns. To see this, let $p_1, \ldots, p_m$ be a sequence of polynomials in $\mathbb{R}[x_1, \ldots, x_n]$ of degree at most d. The sequence can clearly have at most 3^m distinct sign patterns. So, there is a *finite* set $X \subset \mathbb{R}^n$ of $|X| \leq 3^m$ vectors witnessing all distinct sign patterns of this sequence. Take

$$\epsilon = \tfrac{1}{2} \cdot \min\{p_i(x) : x \in X \text{ and } p_i(x) \neq 0\},$$

and consider the sequence $p_1 - \epsilon, p_1 + \epsilon, \ldots, p_m - \epsilon, p_m + \epsilon$ of $2m$ polynomials. By the choice of ϵ, each two distinct $(-1, 0, +1)$ patterns of the original sequence lead to also distinct $(-1, +1)$ patterns of the new sequence.

We will use the following direct consequence of Warren's theorem.

Corollary 2. *Let $\Phi_1(x), \ldots, \Phi_m(x)$ be a sequence algebraic formulas on the same n variables. If each of these formulas have description complexity at most t, then*

$$\left|\big(\Phi_1(x), \ldots, \Phi_m(x)\big) : x \in \mathbb{R}^n\right| \leq \left(\frac{8emt^2}{n}\right)^n.$$

This follows from Theorem 4 because the values of any such sequence of algebraic formulas only depend on the sign patterns of the sequence of $\leq mt$ polynomials of degree $\leq t$ used in these formulas.

3.5 What functions are not semialgebraic?

To show what kind of operations we do *not* allow to be used as gates, let us recall the following well known *necessary* condition for a set to be semialgebraic.

Claim 3. *If a set $S \subseteq \mathbb{R}^n$ is semialgebraic, then either the interior of S is nonempty, or some nonzero polynomial vanishes on all points of S.*

Proof. By observing that a system of equations $p_1(x) = 0, \ldots, p_m(x) = 0$ 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 it is a finite union $S = S_1 \cup S_2 \cup \cdots \cup S_m$ of (nonempty) sets 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. □

Example 3. Claim 3 can be used to show that some functions are *not* semialgebraic. Consider, for example, the rounding function $f(x) = \lfloor x \rfloor$. That is, for a real number $x \in \mathbb{R}$, $f(x)$ is the largest integer n such that $n \leq x$. The interior of the graph $S = \{(x, y) \in \mathbb{R} \times \mathbb{Z} \colon \lfloor x \rfloor = y\}$ 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, the polynomial $p(x, n)$ has an infinite (and, hence, larger than d) number of roots $x \in [n, n+1)$, for every integer n; so, $p_i(n) = 0$ for all i. Since this holds for infinitely many numbers n, all polynomials $p_0, p_1, \ldots, p_d$ must be zero polynomials. So, the rounding function is not semialgebraic.

3.6 Probabilistic circuits

A circuit *basis* is any family $\mathcal{B}$ of multivariate real-valued functions. A *circuit* over a basis $\mathcal{B}$ is a sequence $F = (f_1, \ldots, f_s)$ of real-valued functions, where each f_i is obtained by applying one of the basis operations to the functions in $\mathbb{R} \cup \{x_1, \ldots, x_n, f_1, \ldots, f_{i-1}\}$; scalars $a \in \mathbb{R}$ can be also viewed as (constant) functions. The *size* of a circuit is the number s of functions in the sequence, and the function $f : \mathbb{R}^n \to \mathbb{R}$ *computed* by the circuit is the last function $f = f_s$ in the sequence. Every circuit can be also viewed 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 a scalar $a \in \mathbb{R}$. Every other node, a *gate*, performs one of the operations $g \in \mathcal{B}$ on the results computed at its input gates. A circuit is *b-semialgebraic* if each its basis operation (a gate) is b-semialgebraic.

A *probabilistic circuit* is a deterministic circuit which, besides the actual (deterministic) variables $x_1, \ldots, x_n$, is allowed to use some number k of additional variables $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_k$, each being a *random* variable taking its values in $\mathbb{R}$. As we already mentioned in the introduction, the probability distribution of these random variables can be arbitrary: our derandomization results will hold for *any* distribution.

4 THE ROUTE TO DERANDOMIZATION

In our derandomization of probabilistic circuits, the following parameters of (finite or infinite) Boolean matrices $M : A \times B \to \{0, 1\}$ will be crucial.

- The matrix M has the *m-majority property* if there is a sequence $b_1, \ldots, b_m \in B$ of not necessarily distinct columns of M such that

$$M[a, b_1] + \cdots + M[a, b_m] > m/2$$

holds for every row $a \in A$.

- The matrix M is *probabilistically dense* if there exists a probability distribution $\Pr : B \to [0,1]$ on the set of columns such that

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

holds for every row $a \in A$. Note that the mere *existence* of at least one probability distribution with this property is sufficient. Thus, density is a property of matrices, not of probability distributions on their columns.
- The *growth function* of M is the function $\Pi_M : \mathbb{N} \to \mathbb{N}$ whose value $\Pi_M(m)$ for each integer $m \geq 1$ is the maximum

$$\Pi_M(m) = \max_{b_1,\ldots,b_m} \left|\left\{\big(M[a,b_1]\ldots,M[a,b_m]\big)\colon a \in A\right\}\right|$$

over all choices of m columns of M, of the number of distinct 0-1 patterns from $\{0,1\}^m$ appearing as rows of M in these columns. Note that $1 \leq \Pi_M(m) \leq 2^m$ for every $m \geq 1$. The maximum number m (if there is one) for which $\Pi(m) = 2^m$ holds is know as the *Vapnik–Chervonenkis dimension* or just *VC-dimension* of the matrix M.
- A Boolean matrix $M : \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ is *semialgebraic* if the set

$$S = \{(x,y) \in \mathbb{R}^{n+k}\colon M[x,y] = 1\}$$

of its 1-entries is such. The *description complexity* of a column $r \in \mathbb{R}^k$ is the description complexity of the set $S_r = \{x \in \mathbb{R}^n\colon M[x,r] = 1\}$ of its 1-entries.

Given a probabilistic circuit $F(x,\boldsymbol{r})$ computing a given function $f : \mathbb{R}^n \to \mathbb{R}$, the following two Boolean matrices naturally arise, where k is the number of random input variables $\boldsymbol{r} = (\boldsymbol{r}_1,\ldots,\boldsymbol{r}_k)$.

- The *graph matrix* of $F(x,\boldsymbol{r})$ is the Boolean matrix $M_F\colon \mathbb{R}^{n+1} \times \mathbb{R}^k \to \{0,1\}$ with entries defined by:

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

The graph matrix M_F gives us a full information about all functions computed by the circuits $F(x,r)$ obtained from $F(x,\boldsymbol{r})$ by setting the random inputs $\boldsymbol{r}$ of F to all possible values $r \in \mathbb{R}^k$.
- The *correctness matrix* of $F(x,\boldsymbol{r})$ with respect to the given function $f : \mathbb{R}^n \to \mathbb{R}$ is the Boolean matrix $M : \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ with entries defined by:

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

Note that M is a *submatrix* of the graph matrix M_F: just remove all rows of M_F labeled by pairs (x,y) such that $y \neq f(x)$, and replace the label (x,y) of each remaining row by x.

The relation of the majority property of matrices to the derandomization of probabilistic circuits is quite natural. Suppose that a probabilistic circuit $F(x,\boldsymbol{r})$ computes the correct values $f(x)$ of a given function f with probability $\geq 2/3$. Then the correctness matrix M is probabilistically dense *per se*. On the other hand, if the matrix M has the m-majority property, then there are m (not necessarily distinct) assignments $r_1,\ldots,r_m \in \mathbb{R}^k$ to the random input variables such that, for every input $x \in \mathbb{R}^n$, the *deterministic* circuit $F(x) = \mathrm{Maj}(F(x,r_1),\ldots,F(x,r_m))$ outputs the correct value $f(x)$.

Thus, the derandomization of probabilistic circuits boils down to showing that their correctness matrices have the m-majority property for possibly small values of m. We will show this in the following three steps, where $F(x,\boldsymbol{r})$ is a probabilistic circuit with n deterministic input variables, k random input variables, and s gates.

Let t be the minimal number t such that for every $r \in \mathbb{R}^k$, the function $F_r : \mathbb{R}^n \to \mathbb{R}$ computed by the deterministic circuit $F_r(x) = F(x,r)$ is t-semialgebraic.

Step 1 (Lemma 1 in Section 5) The growth function of the graph matrix M_F of F satisfies

$$\ln \Pi_{M_F}(m) \le n \ln(8emt^2/n) = 2n \ln t + n \ln(8em/n)\,.$$

Step 2 (Lemma 3 in Section 6) There is an absolute constant $c > 0$ such that every probabilistically dense submatrix of M_F has the m-majority property for any $m \ge 2/c$ satisfying

$$\ln \Pi_{M_F}(m) \le cm\,.$$

Step 3 (Lemma 5 in Section 7) If the description complexity of each single gate of F does not exceed b, then

$$\ln t = O(ns \ln bs)\,.$$

Now, if the probabilistic circuit $F(x, \boldsymbol{r})$ computes a given function $f : \mathbb{R}^n \to \mathbb{R}$, then the correctness matrix M of this circuit with respect to the function f, *is* a probabilistically dense *per se*. Also, as we have shown right after its definition, the correctness matrix M is a *submatrix* of the graph matrix M_F of the circuit F. Thus, by Steps 1–3, the matrix M has the m-majority property for any m satisfying the inequality $2n \ln t + n \ln(8em/n) \le cm$, where $2n \ln t = O(n^2 s \log bs)$. This inequality is satisfied by taking $m = Cn^2 s \log bs$ for a sufficiently large (but absolute) constant C.

The case of *approximating* (not necessarily exactly computing) probabilistic circuits requires an additional idea. The reason is that then the "approximate correctness" matrix M of the circuit F approximating the function f is *not* necessarily a submatrix of the graph matrix M_F of the circuit F. For example, if $F(x, r) = z$ for some z such that $z \ne f(x)$ but $z \varrho f(x)$, then $M_F[(x, f(x)), r] = 0$ but the corresponding entry (x, r) in the "approximate correctness" matrix M will then be $M[x, r] = 1$. This is why in Theorem 2, unlike in Theorem 1, also the description complexities t_ϱ and t_f of the approximation relation ϱ and of the approximated function f come to play.

We now turn to detailed proofs.

5 STEP 1: GROWTH FUNCTIONS FROM DESCRIPTION COMPLEXITY

The following lemma is an easy consequence of Warren's theorem (Theorem 4).

Lemma 1. *Let $M : \mathbb{R}^n \times \mathbb{R}^k \to \{0, 1\}$ be a Boolean matrix. If the description complexity of every column of M does not exceed t, then for all $m \ge n$, the growth function $\Pi_M(m)$ of M satisfies*

$$\Pi_M(m) \le \left(\frac{8emt^2}{n}\right)^n\,.$$

Proof. Take arbitrary m columns $r_1, \ldots, r_m \in \mathbb{R}^k$ of M. Since every column of M is t-semialgebraic, for every $i = 1, \ldots, m$ there is an algebraic formula $\Phi_i(x)$ which uses at most t distinct polynomials of degree at most t, and satisfies $M[x, r_i] = \Phi_i(x)$ for all $x \in \mathbb{R}^n$. So, $\Pi_M(m)$ is at most the number of distinct 0-1 strings $(\Phi_1(x), \ldots, \Phi_m(x))$ when x ranges over the entire set $\mathbb{R}^n$ of row labels of the matrix M. By Corollary 2, the number of such strings is at most $\left(8emt^2/n\right)^n$. □

Remark 3. In the case when the description complexity of the *entire* matrix M is bounded by t, similar upper bounds on the growth function were already derived from the Warren's theorem by Goldberg and Jerrum [12], and by Ben-David and Lindenbaum [6]. Our observation is that the same upper bound actually holds when only the description complexities of *individual* columns are bounded by t. In the context of derandomization, this will alow us to make the blowup in size of derandomized circuits *independent* on the number k of random input variables (note that k may be as large as the size of the probabilistic circuits).

This observation also extends the bound to a properly larger class of matrices. The point is that the description complexity of individual columns may be much smaller than that of the entire matrix M. Even more: the former can even then be small, when the entire matrix is *not* semialgebraic at all (its

description complexity is unbounded). As a trivial example, consider the matrix $M : \mathbb{R} \times \mathbb{R} \to \{0,1\}$ whose entries are defined by: $M[x,r] = 1$ if and only if $x = \lfloor r \rfloor$. The matrix is not semialgebraic (see Example 3), but for every *fixed* column $r \in \mathbb{R}$, the set of 1-entries of the rth column is defined by a semialgebraic formula $[x - c = 0]$, where $c = \lfloor r \rfloor$ is a (fixed) integer. Hence, the description complexity of each individual column is 1.

6 STEP 2: MAJORITY PROPERTY FROM GROWTHS FUNCTIONS

As we mentioned at the beginning of Section 4, the derandomization of probabilistic circuits boils down to showing that their correctness matrices have the m-majority property for possibly small values of m. In this section, we target this "ensuring majority property" task.

6.1 Finite majority rule

Let H be a class of 0-1 functions $h : X \to \{0,1\}$ on a set X, and $\Pr : X \to [0,1]$ a probability distribution on the set X. Draw independently (with repetitions) a sequence $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ of samples $\boldsymbol{x}_i \in X$ according to this probability distribution. The *empirical frequency* of $h \in H$ on $\boldsymbol{x}$ is the average value

$$\mathrm{ave}_h(\boldsymbol{x}) := \frac{h(\boldsymbol{x}_1) + \cdots + h(\boldsymbol{x}_m)}{m},$$

while the *theoretical probability* of the function h itself is its expected value

$$p_h := \Pr\{x \in X \colon h(x) = 1\}\ .$$

Every function $h : X \to \{0,1\}$ defines the event $\{x \in X \colon h(x) = 1\}$. The law of large numbers says that, for each single event, its empirical frequency in a sequence of independent trials converges (with high probability) to its theoretical probability. We, however, are now interested not in a single event but in a whole family of events. We would like to know whether the empirical frequency of every event in the family converges to its theoretical probability *simultaneously*. This is the content of so-called "uniform convergence in probability" results in statistics.

Let $\epsilon > 0$ be a constant, and draw independently (with repetitions) a sequence $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ of samples $\boldsymbol{x}_i \in X$ according to a given probability distribution. We are interested in upper-bounding the probability

$$\Pr\{\exists h \in H \colon\ |\mathrm{ave}_h(\boldsymbol{x}) - p_h| > \epsilon\}\ . \tag{4}$$

When the class H of functions is *finite*, then the this can be easily done. By Hoeffding bound (see, for example, [10, Theorem 1.1]), we have $\Pr\{|\mathrm{ave}_h(\boldsymbol{x}) - p_h| > \epsilon\} \le 2\mathrm{e}^{-2m\epsilon^2}$ for each function $h \in H$. By the union bound, this implies that the probability (4) is at most $2|H|\mathrm{e}^{-2m\epsilon^2}$, which is strictly smaller than 1 as long as the sample has $m = \Omega(\ln 2|H|)$ elements. Thus, for such an m, there exists a sequence $x = (x_1, \ldots, x_m)$ of (not necessarily distinct) points in X such that $\mathrm{ave}_h(x) \ge p_h - \epsilon$, that is,

$$h(x_1) + \cdots + h(x_m) \ge (p_h - \epsilon)m \tag{5}$$

holds for *all* functions $h \in H$. This gives the well-known fact that, if the number of rows is *finite*, then the m-majority property holds already for m about the logarithm of this number.

Lemma 2 (Finite majority rule). *Every probabilistically dense Boolean matrix $M : A \times B \to \{0,1\}$ with a finite number $|A|$ of rows has the m-majority property for $m = O(\log |A|)$. In particular, at least one column of M has more than $|A|/2$ ones.*

Proof. View each row $a \in A$ as a function $h : B \to \{0,1\}$ given by $h(b) := M[a,b]$. Since the matrix M is probabilistically dense, we have $p_h \ge 2/3$ for each $h \in H$. So, when applied with, say $\epsilon = 1/7$, Eq. (5) gives us at most $m = O(\log |H|) = O(\log |A|)$ columns $b_1, \ldots, b_m \in B$ such that $h(b_1) + \cdots + h(b_m) \ge (p_h - \epsilon)m = (p_h - 1/7)m \ge (2/3 - 1/7)m > m/2$ holds for every $h \in H$.

Thus, the matrix M has the m-majority property for $m = O(\log |A|)$. This shows the first claim. The second claim follows by double-counting: the number of ones in the corresponding m columns of M is $> (m/2)|A|$. So, at least one of these columns must have $> |A|/2$ ones. □

Lemma 2 allows us to derandomize probabilistic circuits working over any *finite* domain (including Boolean circuits): if the probabilistic circuit has size s, then the obtained deterministic circuit (with one additional majority vote operation as the output gate) will have size $O(ns)$. We are, however, interested in derandomizing circuits working over *infinite* domains, like $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$. Sometimes even then the finite majority rule *can* be applied. In particular, this happens for arithmetic $(+, \times, -, /)$ circuits working over entire domain $\mathbb{R}$ (see Theorem 6 in Section 10.1). But this is just a rare exception: in most cases (including circuits simulating dynamic programs) the finite majority rule is of no use at all.

6.2 Uniform convergence in probability

Fortunately, in infinite domains, results from the statistical learning theory come to rescue. The classical *uniform convergence in probability* theorem of Vapnik and Chervonenkis [34] ensures the majority property also for matrices M with an infinite number of rows, as long as their growth functions $\Pi_M(m)$ grow not too fast (Lemma 3 below).

Let H be a class of 0-1 functions $h : X \to \{0,1\}$ on a set X, and $\Pr : X \to [0,1]$ a probability distribution on the set X. The *growth function* of the family H is the function $\Pi_H : \mathbb{N} \to \mathbb{N}$ whose value $\Pi_H(m)$ for each integer $m \geq 1$ is the maximum,

$$\Pi_H(m) = \max_{x_1,\ldots,x_m} \left| \left\{ \big(h(x_1), \ldots, h(x_m)\big) : h \in H \right\} \right|$$

over all sequences $x_1, \ldots, x_m$ of (not necessarily distinct) points in X, of the number of distinct 0-1 patterns from $\{0,1\}^m$ produced by the functions $h \in H$ on these points. Note that we always have $1 \leq \Pi_H(m) \leq 2^m$.

Note that even if our family H of functions $h : X \to \{0,1\}$ is *infinite* or even uncountable, there is only a *finite* number $\Pi_H(m)$ of their classes such that the functions lying within the same class take the *same* values on *all* m sampled points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m \in X$. By combining this simple observation with insightful ideas, Vapnik and Chervonenkis [34] proved the following result.

Remark 4. In this result, a mild measurability condition on the class H of functions is necessary (to avoid "pathological" situations when H is uncountable). A class H is *permissible* if the individual functions $h \in H$ as well as the supremum function $\pi(x) = \sup_{h \in H} |\mathrm{ave}_h(\boldsymbol{x}) - \mu_h|$ are measurable. That is, we need that for a random sample $\boldsymbol{x} \in X^m$, $\pi(\boldsymbol{x})$ is a random variable. In our applications, the classes H will correspond to the rows of graph matrices of semialgebraic circuits. So, each class H will consist of 0-1 valued *semialgebraic* functions $h : X \to \{0,1\}$, where $X = \mathbb{R}^k$ for some finite $k \geq 1$, and will be of the form $H = \{f(t, \cdot) : t \in \mathbb{R}^n\}$ for a finite $n \geq 1$, where the indexing function $f : \mathbb{R}^n \times X \to \{0,1\}$ (the matrix itself) is also semialgebraic. Such classes H are permissible; see Appendix B for more details.

Theorem 5 (Vapnik and Chervonenkis [34]). *Let H be a permissible class of 0-1 functions $h : X \to \{0,1\}$ on a set X, and $\Pr : X \to [0,1]$ a probability distribution on the set X. Let $\epsilon > 0$, and draw independently (with repetitions) a sequence $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m)$ of $m \geq 2/\epsilon^2$ samples $\boldsymbol{x}_i \in X$ according to this probability distribution. Then*

$$\Pr\{\exists h \in H : \ |\mathrm{ave}_h(\boldsymbol{x}) - p_h| > \epsilon\} \leq 4 \cdot \Pi_H(2m) \cdot \mathrm{e}^{-\epsilon^2 m/8}. \tag{6}$$

In particular, for every constant $0 < \epsilon \leq 1$ there is a constant $c > 0$ with the following property: if the sample size m satisfies

$$m \geq 2/c \quad \text{and} \quad \Pi_H(m) \leq \mathrm{e}^{cm}, \tag{7}$$

then there exists a sequence $x = (x_1, \ldots, x_m)$ of (not necessarily distinct) points in X such that $\text{ave}_h(x) \geq p_h - \epsilon$, that is, the inequality Eq. (5) holds for *all* functions $h \in H$.

We now turn back to the language of matrices. Let $M : T \times X \to \{0,1\}$ be a Boolean matrix. Each row $t \in T$ of M gives us a 0-1 valued function $h_t : X \to \{0,1\}$ whose values are $h_t(x) = M[t,x]$. We say that the matrix M is *permissible* if the class $H = \{h_t : t \in T\}$ of functions corresponding to its rows is permissible.

Recall that the *growth function* $\Pi_M(m)$ of the matrix M is the maximum, over all choices of up to m columns, of the number of distinct 0-1 patterns from $\{0,1\}^m$ appearing as rows in these columns. Note that $\Pi_M(m)$ coincides with the growth function $\Pi_H(m)$ of the class of functions H defined by the rows of M. In what follows, under a *submatrix* of a matrix M we will understand a submatrix obtained by removing some rows of M; that is, we do not remove columns.

Lemma 3. *There is an absolute constant $c > 0$ for which the following holds. If a Boolean matrix M is permissible, then every probabilistically dense submatrix of M has the m-majority property for any integer $m \geq 2/c$ satisfying $\Pi_M(m) \leq \mathrm{e}^{cm}$.*

Proof. Let $M : T \times X \to \{0,1\}$ be a permissible matrix, and let $H = \{h_t : t \in T\}$ be the class of functions $h_t(x) = M[t,x]$ defined by the rows $t \in T$ of M. Let M' be any probabilistically dense submatrix of M, and $H' \subseteq H$ be the class of functions corresponding to the rows of M'. Hence, there is a probability distribution $\Pr : X \to [0,1]$ on the set X of columns such that the probability $p_h = \Pr\{x \in X : h(x) = 1\}$ is at least $2/3$ for every row $h \in H'$ of the submatrix M'.

Fix $\epsilon := 1/7$, and let $c > 0$ be a constant for which Eq. (7) (and, hence, also Eq. (5)) holds with this choice of ϵ. By Eq. (5) there exists a sequence $x_1, \ldots, x_m$ of (not necessarily distinct) columns of M such that $h(x_1) + \cdots + h(x_m) \geq (p_h - \epsilon)\, m = \left(p_h - \frac{1}{7}\right) m$ holds for every row $h \in H$ of M. For some rows $h \in H$ of M (those with $p_h \leq \epsilon$), this lower bound is trivial. But since the submatrix M' is probabilistically dense, we know that $p_h \geq 2/3$ holds for all rows $h \in H'$ of the submatrix M'. Thus, for every row $h \in H'$, we have $h(x_1) + \cdots + h(x_m) \geq \left(p_h - \frac{1}{7}\right) m \geq \left(\frac{2}{3} - \frac{1}{7}\right) m = \frac{11}{21} m > \frac{1}{2} m$, meaning that the submatrix M' has the m-majority property, as desired. □

6.3 Infinite majority rule

Recall that every probabilistically dense Boolean matrix $M : A \times B \to \{0,1\}$ with a *finite* number of rows has the m-majority property for $m = O(\log |A|)$. When combined with Warren's theorem (Theorem 4), the theorem of Vapnik and Chervonenkis (Theorem 5) yields the following result for *infinite* matrices.

Lemma 4 (Infinite majority rule). *Let $M : \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ be a semialgebraic Boolean matrix. If the description complexity of every column of M does not exceed t, then every probabilistically dense submatrix of M has the m-majority property for $n \leq m = O(n \log t)$.*

Proof. Let M' be a submatrix of M, and assume that the matrix M' is probabilistically dense. Since M' is a submatrix of M, its growth function satisfies $\Pi_{M'}(m) \leq \Pi_M(m)$ for all $m \geq 1$. Hence, Lemma 1 gives us an upper bound

$$\Pi_{M'}(m) \leq \Pi_M(m) \leq \left(\frac{8\mathrm{e}mt^2}{n}\right)^n . \tag{8}$$

on the growth function of the matrix M', for all $m \geq n$. On the other hand, since the matrix M is semialgebraic, it is permissible (see Appendix B). So, by Lemma 3, the submatrix M' of M has the m-majority property for any $m \geq 2/c$ satisfying $\Pi_{M'}(m) \leq \mathrm{e}^{cm}$, where $c > 0$ is an absolute constant. Thus, by Eq. (8), in order to ensure the m-majority property for the submatrix M', it is

enough that m satisfies the inequality

$$\left(\frac{8emt^2}{n}\right)^n \le \mathrm{e}^{cm}. \tag{9}$$

By taking logarithms and setting $w := m/n$, Eq. (9) turns into the inequality $\ln w + \ln(8et^2) \le cw$. If $w \le 8et^2$, then it is enough that $2\ln(8et^2) \le cw$ holds, which happens if $w = C\log t$ for a large enough constant C. If $w \ge 8et^2$, then it is enough that $2\ln w \le cw$ holds, which happens if $w = C$ itself is a large enough constant. In both cases, we have that $w \le C\log t$ and, hence, any integer $m \le Cn\log t$ for a large enough constant C satisfies the inequality Eq. (9). □

Remark 5. Note that in order to apply Lemma 4 for a boolean matrix M, an upper bound on the description complexity t of its individual columns does not suffice: to ensure the permissibility of the entire matrix, we have also to ensure that the matrix itself is semialgebraic, that is, has an arbitrary large but *finite* description complexity. This is because even when the description complexity of individual columns is bounded, the entire matrix M may be *not* semialgebraic (see Remark 3). Fortunately, already the classical Tarski–Seidenberg theorem [30, 33] (superpositions of semialgebraic functions are semialgebraic) ensures that graph matrices of probabilistic circuits consisting of semialgebraic gates *are* semialgebraic.

7 STEP 3: DESCRIPTION COMPLEXITY OF CIRCUITS

An important consequence of the Tarski–Seidenberg theorem [30, 33] is that compositions of semialgebraic functions are also semialgebraic functions. This, in particular, implies that functions computable by circuits over any basis consisting of semialgebraic functions are also semialgebraic. But what about the description complexity of such functions?

(∗) If each basis function (gate) has 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 in the following lemma.

Lemma 5. *Every function $f : \mathbb{R}^n \to \mathbb{R}$ computable by a deterministic b-semialgebraic circuit of size at most s has the following properties.*

(i) *The graph $\{(x, y) : f(x) = y\}$ of f can be defined by an existential algebraic formula of description complexity at most sb, and with at most $s-1$ (existential) quantifiers.*
(ii) *The function f is t-semialgebraic for t satisfying $\log t = O(ns\log bs)$.*

Proof. The second property (ii) follows directly from (i) and Corollary 1. So, it is enough to prove the first property (i).

Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions. Let F be a circuit of size s over $\mathcal{B}$ computing the function $f : \mathbb{R}^n \to \mathbb{R}$.

The circuit F is a sequence $F = (f_1, \ldots, f_s)$ of functions $f_i : \mathbb{R}^n \to \mathbb{R}$, where $f_s = f$ and each f_i is obtained by applying one of the basis operations (a gate) to $\mathbb{R} \cup \{x_1, \ldots, x_n, f_1, \ldots, f_{i-1}\}$. Since every basis operation $g_i : \mathbb{R}^k \to \mathbb{R}$ is b-semialgebraic, there must be an algebraic formula $\Phi_i(x, y)$ using at most b polynomials of degree at most b 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 $f_i = g_i(f_{i_1}, \ldots, f_{i_k})$ with $g_i \in \mathcal{B}$ and each f_{i_j} in $\mathbb{R} \cup \{x_1, \ldots, x_n, f_1, \ldots, f_{i-1}\}$ gives us an equation $z_i = g_i(w_i)$, where w_i is a vector in $(\mathbb{R} \cup \{x_1, \ldots, x_n, z_1, \ldots, z_{i-1}\})^k$. So, $\Phi_i(w_i, z_i) = 1$ if and only if $z_i = g_i(w_i)$. The value of the first variable z_1 in the sequence $z_1, \ldots, z_s$ is determined by the actual inputs $\mathbb{R} \cup \{x_1, \ldots, x_n\}$ to the circuit (is obtained by applying the basis operation g_1 to these inputs), whereas the value of each subsequent variable z_i ($i \ge 2$) is obtained by applying the ith base operation g_i to these inputs

and some of the previous values $z_1, \ldots, z_{i-1}$. So, the existential formula

$$\begin{aligned}\Psi(x,y) &= \exists z_1 \ldots \exists z_{s-1}\, [z_1 = g_1(w_1)] \wedge \cdots \wedge [z_{s-1} = g_{s-1}(w_{s-1})] \wedge [y = g_s(w_s)] \\ &= \exists z_1 \ldots \exists z_{s-1} \Phi_1(w_1, z_1) \wedge \ldots \wedge \Phi_{s-1}(w_{s-1}, z_{s-1}) \wedge \Phi_s(w_s, y)\end{aligned}$$

defines the graph $\{(x,y)\colon y = f(x)\}$ of the function $f = f_s$ computed by our circuit F. Existential quantifiers just guess the possible values taken at intermediate gates, and the equalities ensure their correctness. Since each algebraic formula Φ_i uses at most b polynomials of degree at most b, the formula Ψ uses at most sb polynomials of degree at most b, and contains only $s-1$ quantifiers. □

8 PROOF OF THEOREM 1

Suppose that a probabilistic b-semialgebraic circuit $F(x, \boldsymbol{r})$ of size s with k random input variables computes a function $f : \mathbb{R}^n \to \mathbb{R}$. Our goal is to show then there are $m = O(n^2 s \log bs)$ deterministic copies $F_1(x, r_1), \ldots, F_m(x, r_m)$ of $F(x, \boldsymbol{r})$ such that, for every input $x \in \mathbb{R}^n$, more than the half of these circuits will output the correct value $f(x)$.

Let $M\colon \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ be the correctness matrix of the circuit F (with respect to the given function f). Hence, the entries of M are defined by: $M[x,r] = 1$ if and only if $F(x,r) = f(x)$.

Claim 4. *The matrix M has the m-majority property for $m = O(n^2 s \log bs)$.*

Proof. We are going to apply the infinite majority rule (Lemma 4). Recall that the graph matrix of the circuit $F(x, \boldsymbol{r})$ is the Boolean matrix $M_F\colon \mathbb{R}^{n+1} \times \mathbb{R}^k \to \{0,1\}$ with entries defined by: $M_F[(x,y), r] = 1$ if and only if $y = F(x,r)$.

Since the circuit F only uses semialgebraic functions as gates, Tarski–Seidenberg theorem [30, 33] implies that the function $F : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$ computed by the circuit F and, hence, also the graph matrix M_F of F is also semialgebraic. Furthermore, for every assignment $r \in \mathbb{R}^n$ of the values to the random input variables, $F(x,r)$ is a deterministic b-semialgebraic circuit of size s computing some function $F_r : \mathbb{R}^n \to \mathbb{R}$. Lemma 5 implies that each of the functions F_r is t-semialgebraic for t satisfying $\log t = O(ns \log bs)$. Thus, the description complexity of every column of M_F does not exceed t.

Note that the correctness matrix M is a *submatrix* of the matrix M_F obtained by removing all rows of M_F labeled by pairs (x,y) such that $y \neq f(x)$, and replacing the label (x,y) of each remaining row by x. Moreover, since the (probabilistic) circuit $F(x, \boldsymbol{r})$ computes f, the correctness matrix M is probabilistically dense. (The graph matrix M_F itself does not need to be such.) So, the infinite majority rule (Lemma 4) implies that the correctness matrix M has the m-majority property for $m = O(n \log t) = O(n^2 s \log bs)$. □

Claim 4 implies that there must be some m (not necessarily distinct) columns $r_1, \ldots, r_m$ of M such that, for every input $x \in \mathbb{R}^n$, the inequality $|\{i\colon M[x, r_i] = 1\}| > m/2$ and, hence, also the inequality $|\{i\colon F(x, r_i) = f(x)\}| > m/2$ holds. Thus, on every input $x \in \mathbb{R}^n$, more than the half of the values computed by deterministic copies $F_1(x, r_1), \ldots, F_m(x, r_m)$ of the circuit $F(x, \boldsymbol{r})$ compute the correct value $f(x)$, as desired. □

Remark 6. We could apply Lemma 5 to the function $F : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$ computed by the entire circuit $F(x, \boldsymbol{r})$, but this would result in quadratic increase of the size of the derandomized circuit. Namely, Lemma 5 would then imply that this function is t-semialgebraic with $\log t$ at most about $(n+k)s \log bs$. Since the number k of random input variables may be as large as the total number s of gates, this is then about $s^2 \log bs$, and we would obtain by a multiplicative factor s worse upper bound $m = O(n^2 s^2 \log bs)$ in Claim 4.

9 PROOF OF THEOREM 2

Let $x \,\varrho\, y$ be a t_ϱ-semialgebraic relation, and $f : \mathbb{R}^n \to \mathbb{R}$ a t_f-semialgebraic function. Suppose that f can be ϱ-approximated by a probabilistic b-semialgebraic circuit $F(x, \boldsymbol{r})$ of size s. Our goal is to show that then f can be also ϱ-approximated as a majority ϱ-vote of $m = O(n^2 s \log K)$ deterministic copies of this circuit, where $K = sb + t_f + t_\varrho$.

Consider the *correctness matrix* $M\colon \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ with entries defined by:

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

Since the circuit F ϱ-approximates the function, the matrix M is probabilistically dense.

Claim 5. *The correctness matrix M is semialgebraic, and the description complexity t of every its column satisfies* $\log t = O(ns \log K)$.

Proof. The probabilistic circuit $F(x, \boldsymbol{r})$ computes some function $F : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$ of $n + k$ variables. Let $\Phi_F(x, y, r)$ an existential algebraic formula ensured by Lemma 5(i). Hence, the formula Φ_F has at most $s - 1$ quantifiers, has description complexity $\kappa \le sb$, and defines the graph of, that is, $\Phi_F(x, y, r) = 1$ if and only if $y = F(x, r)$.

Similarly, since the function f is t_f-semialgebraic, there is an algebraic formula $\Phi_f(x, y)$ of size and degree at most t_f such that $\Phi_f(x, y) = 1$ if and only if $y = f(x)$. Finally, since the relation ϱ is t_ϱ-semialgebraic, there is an algebraic formula $\Phi_\varrho(x, y)$ of size and degree at most t_ϱ such that $\Phi_\varrho(x, y) = 1$ if and only if $x \,\varrho\, y$. Consider the existential algebraic formula

$$\Psi(x, r) = \exists y_1\ \exists y_2\ \Phi_F(x, y_1, r)\ \wedge\ \Phi_f(x, y_2)\ \wedge\ \Phi_\varrho(y_1, y_2)\,.$$

It is easy to see that for every row $x \in \mathbb{R}^n$ and every column $r \in \mathbb{R}^k$ of M, we have $M[x, r] = 1$ if and only if $\Psi(x, r) = 1$. Indeed, since both $F(x, r)$ and $f(x)$ are everywhere defined functions, on every point (x, r) they output some unique values $F(x, r) = y_1$ and $f(x) = y_2$. So, the first part $\exists y_1\ \exists y_2\ \Phi_F(x, y_1, r) \wedge \Phi_f(x, y_2)$ of the formula Ψ is a tautology, that is, outputs 1 on all inputs. But the last formula $\Phi_\varrho(y_1, y_2)$ outputs 1 precisely when $y_1 \,\varrho\, y_2$ holds, which happens precisely when $F(x, r) \,\varrho\, f(x)$ holds.

Thus, the existential formula $\Psi(x, r)$ defines the correctness matrix M. By the Tarski–Seidenberg theorem [30, 33], the formula $\Psi(x, r)$ has an equivalent quantifier-free algebraic formula. This shows that the correctness matrix M is semialgebraic, and it remains to upper bound the description complexity of its columns.

So, fix a column $r \in \mathbb{R}^k$ of M, and consider the existential formula $\Psi_r(x) := \Psi(x, r)$ obtained from the formula $\Psi(x, r)$ by fixing the r-variables to the corresponding values. This formula defines the rth column of M, and its description complexity is is at most the sum $\kappa + t_f + t_\varrho \le sb + t_f + t_\varrho \le K$ of the description complexities of formulas Ψ_F, Φ_f and Φ_ϱ. The formula has n free variables (x-variables). The formulas Φ_f and Φ_ϱ have no quantifiers, and Ψ_F has at most s existential quantifiers. So, the entire existential formula Ψ has only $q \le s + 2$ quantifiers, and its description complexity is at most K. Corollary 1 gives us an equivalent quantifier-free algebraic formula of description complexity t satisfying $\log t = O(nq \log K) = O(ns \log K)$. Thus, the description complexity t of each single column of M satisfies $\log t = O(ns \log K)$, as desired. □

Since the circuit $F(x, \boldsymbol{r})$ ϱ-approximates our function f, the correctness matrix M is probabilistically dense. By Claim 5, the description complexity t of every its column satisfies $\log t = O(ns \log K)$. So, by the infinite majority rule (Lemma 4), the matrix M has the m-majority property for $m = O(n \log t) = O(n^2 s \log K)$. This means that there must be some m (not necessarily distinct) columns $r_1, \ldots, r_m$ of M such that, for every input $x \in \mathbb{R}^n$, the inequality $|\{i\colon M[x, r_i] = 1\}| > m/2$ and, hence, also the inequality $|\{i\colon F(x, r_i) \,\varrho\, f(x)\}| > m/2$ holds. Thus, if $\mu : \mathbb{R}^m \to \mathbb{R}$ is a majority

ϱ-vote function, then $\mu(F_1(x,r_1),\dots,F_m(x,r_m))\ \varrho\ f(x)$ holds for every input $x\in\mathbb{R}^n$. That is, the obtained deterministic circuit (with one majority ϱ-vote output gate) ϱ-approximates the values $f(x)$ of our function f, as desired. □

9.1 Circuits approximating optimization problems

Since one of the motivations in this paper is to derandomize probabilistic dynamic programming algorithms, let us demonstrate Theorem 2 on semialgebraic circuits solving optimization problems. The minimization problem $f:\mathbb{R}^n\to\mathbb{R}$ on a finite set $A\subset\mathbb{N}^n$ of feasible solutions is to compute the values $f(x)=\min\{a_1x_1+\cdots a_nx_n\colon a\in A\}$ on all input weighings $x\in\mathbb{R}^n$.

A probabilistic circuit $F(x,\boldsymbol{r})$ approximates the problem f within a given factor $c\ge 0$ if for every input weighting $x\in\mathbb{R}^n$, $|F(x,\boldsymbol{r})-f(x)|\le c$ holds with probability at least $2/3$.

The relation ϱ is this case is: $x\,\varrho\,y$ if and only if $|x-y|\le c$ (the fourth relation in Example 1). This relation can be defined by a trivial algebraic formula $[x\ge y-c]\wedge[x\le y+c]$. The formula uses only two polynomials $x-y-c$ and $x-y+c$ of degree 1; so, the description complexity is $t_\varrho\le 2$. The relation is clearly contiguous: if $x\le y\le z$, $|x-a|\le c$ and $|z-a|\le c$, then also $|y-a|\le c$.

Let $\mathcal{B}$ be any basis containing the optimization operations $\min(x,y)$, $\max(x,y)$ and any other operations of a constant description complexity $b=O(1)$. For example, besides min and max, the basis may contain any of the arithmetic operations $+,-,\times,\div$, any branching operations "if $x\diamond y$ then u else v" with $\diamond\in\{>,\ge,=,\le,<\}$, and other operations.

Corollary 3. *If a minimization problem $f(x)=\min\{a_1x_1+\cdots a_nx_n\colon a\in A\}$ can be approximated within some fixed factor by a probabilistic circuit of size s over the basis $\mathcal{B}$, then f can be also approximated within the same factor by a deterministic circuit over $\mathcal{B}$ of size at most a constant times $n^2s^2\log(s+|A|)$.*

Proof. The graph $\{(x,y)\colon y=f(x)\}$ of the function f can be defined by an algebraic formula

$$\bigwedge_{a\in A}[a_1x_1+\cdots a_nx_n-y\ge 0]\wedge\Big(\bigvee_{a\in A}[a_1x_1+\cdots a_nx_n-y=0]\Big)$$

using $|A|$ polynomials of degree 1. So, the description complexity of f is $t_f\le|A|$. Since the approximation relation ϱ in our case has a constant description complexity $t_\varrho\le 2$, and since the description complexity b of every gate is also constant, Theorem 2 implies that the minimization problem f can be approximated as a majority ϱ-vote function of $m=O(n^2s\log K)$ deterministic copies of the probabilistic circuit, where $K=sb+t_f+t_\varrho=O(s+|A|)$.

Since the relation ϱ is contiguous, and since both min and max operations are available, a majority ϱ-vote function of m variables can be computed by a circuit over $\mathcal{B}$ of size $O(m\log m)$ (see Claim 7 in Appendix A). Thus, the size of the derandomized circuit is at most a constant times $m\cdot s+m\log m$, which is at most a constant times $n^2s^2\log(s+|A|)$, as desired. □

Remark 7. Note that the upper bound on the size S of the derandomized circuit, given by Theorem 2, is only *logarithmic* in the number $|A|$ of feasible solutions of the minimization problem f. In most optimization problems, the set A of feasible solutions is the set $A\subseteq\{0,1\}^n$ of characteristic 0-1 vectors of objects of interest: spanning trees, perfect matchings, etc. In these cases, $\log|A|$ is at most the number n of variables. Thus, for such problems f, the size of the derandomized circuit is at most a constant times $n^3s^2\log s$.

10 DERANDOMIZATION VIA ISOLATING SETS

Theorems 1 and 2 derandomize very general classes of probabilistic circuits, but their proofs rely on deep tools from three different fields: combinatorial algebraic geometry (sign patterns of polynomials), probability theory (uniform convergence in probability), and quantifier elimination

theory over the reals. When directly applied, elementary tools like the finite majority rule (Lemma 2) fail for such circuits already because the domain is infinite.

In some cases, however, it is still possible to apply even such elementary tools also for circuits working over infinite domains. In particular, this happens if the functions computed by a given class of circuits have finite "isolating sets." In this section, we will demonstrate this approach on arithmetic and tropical circuits.

Given a family $\mathcal{H}$ of functions $h : D \to R$ and a function $f : D \to R$, a set $X \subseteq D$ *isolates* the function f *within* $\mathcal{F}$ if for every function $h \in \mathcal{H}$,

$$h(x) = f(x) \text{ for all } x \in X \text{ implies that } h(x) = f(x) \text{ holds for all } x \in D.$$

That is, if $h(x) \neq f(x)$ for some point $x \in D$ of the entire domain D, then $h(x) \neq f(x)$ for at least one point $x \in X$.

10.1 Arithmetic circuits

In the case of (arithmetic) polynomials, we have the following strong isolation property.

Lemma 6. *Let $f(x)$ be a nonzero n-variate polynomial of degree d over $\mathbb{R}$, and $S \subset \mathbb{R}$ a finite subset of $|S| \geq d+1$ elements. Then every subset $X \subseteq S^n$ of size $|X| > d|S|^{n-1}$ isolates f within all polynomials of degree at most d.*

Proof. Let $S \subset \mathbb{R}$ a finite subset of $|S| \geq d+1$ elements. Take an arbitrary n-variate polynomial $g(x)$ of degree at most d, and suppose that $g(a) \neq f(a)$ holds for at least one point $a \in \mathbb{R}^n$. Then $p(x) := f(x) - g(x)$ is a nonzero polynomial of degree at most d. By the Schwartz–Zippel lemma [29, 36], we then have $|\{a \in S^n : p(a) = 0\}| \leq d|S|^{n-1}$. So, since $|X| > d|S|^{n-1}$, $p(a) \neq 0$ must hold for at least one point $a \in X$, as desired. □

Theorem 6. *If a rational function $f : \mathbb{R}^n \to \mathbb{R}$ can be computed by a probabilistic arithmetic $(+, \times, -, /)$ circuit, then f can be also computed by a deterministic arithmetic $(+, \times, -, /)$ circuit of the same size.*

Proof. The function f is of the form $f(x) = p(x)/q(x)$ for some polynomials p and q. Suppose that f can be computed by a probabilistic arithmetic $(+, \times, -, /)$ circuit $F(x, \boldsymbol{r})$ of size s. Set $d := r+2^s$, where r is the maximum degree of p and q. Take an arbitrary subset $S \subseteq \mathbb{R}$ of size $|S| \geq 2d$. By the finite majority rule (Lemma 2), there is an assignment $r \in \mathbb{R}^k$ to the random input variables, and a subset $X \subset S^n$ of size $|X| > \frac{1}{2}|S|^n$ such that the deterministic copy $F_r(x) = F(x, r)$ of the probabilistic circuit F computes f correctly on all inputs from X. The circuit F_r computes some rational function $F_r(x) = P(x)/Q(x)$. Since the gates have fanin two, the polynomials P and Q have degrees at most 2^s. Consider the polynomial $g(x) := p(x) \cdot Q(x) - q(x) \cdot P(x)$. By the choice of d, the degree of the polynomial g is at most d. We have only to show that g is a null polynomial, i.e., that $g(x) = 0$ holds for all $x \in \mathbb{R}^n$.

Were g a nonzero polynomial, then Lemma 6 would require the set X to have cardinality $|X| \leq d|S|^{n-1}$. But then we would have $\frac{1}{2}|S|^n < |X| \leq d|S|^{n-1}$ and, hence, also $|S| < 2d$, which contradicts our choice of S. □

10.2 Tropical circuits

We now consider circuits over the tropical semiring $(\mathbb{R}_+, \max, +)$. Since the basis operations $\min(x, y)$ and $x + y$ of such circuits have very small (constant) description complexities, Theorem 1 implies that 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 also solved as a majority vote of about $n^2 s \log s$ deterministic copies of this circuits.

But tropical circuits *cannot* compute the majority vote function Maj at all (see Claim 6 in Appendix A). The reason here is that, in such circuits, only one of the operations min or max is allowed. So, the resulting deterministic circuit is *not* a tropical circuit.

On the other hand, tropical circuits are interesting in optimization, because they simulate so-called *pure* dynamic programming algorithms (pure DP algorithms). This raises the question: can probabilistic pure DP algorithms be (efficiently) derandomized at least in the one-sided error probability scenario? In this section, we will give an *affirmative* answer: under the *one-sided* error probability scenario, the resulting deterministic circuits are also tropical circuits (do not use majority vote gates), and the derandomization itself is then elementary.

What circuits over the arithmetic semiring $(\mathbb{R}_+, +, \times)$ compute are polynomials

$$p(x) = \sum_{a \in A} c_a \prod_{i=1}^{n} x_i^{a_i}, \tag{10}$$

where $A \subset \mathbb{N}^n$ is some finite set of nonnegative integer exponent vectors, and $c_a \in \mathbb{R}_+$ are positive coefficients. In the tropical semiring $(\mathbb{R}_+, \max, +)$, "addition" $x + y$ turns into taking the maximum $\max(x, y)$, and "multiplication $x \times y$ turns into addition $x + y$. So, what a tropical $(\max, +)$ circuit computes is a tropical polynomial

$$f(x) = \max_{a \in A} \langle a, x \rangle + c_a, \tag{11}$$

where $\langle a, x \rangle = a_1 x_1 + \cdots + a_n x_n$ stands for the scalar product of vectors a and x. That is, $(\max, +)$ circuits solve maximization problems with linear objective functions; the set A is then the set of feasible solutions. An (arithmetic) polynomial Eq. (10) is *monic* if $c_a = 1$ for all $a \in A$, and *multilinear* if the degree of every variable is at most 1, that is, if $A \subseteq \{0,1\}^n$. By analogy with arithmetic polynomials, we call a tropical polynomial Eq. (11) *monic* if $c_a = 0$ holds for all $a \in A$, and is *multilinear* if $A \subseteq \{0,1\}^n$. (Note that, in the tropical semiring, the multiplicative unity element "1" is 0, because $x + 0 = 0 + x = x$.)

Under a *probabilistic* $(\max, +)$ *circuit* of size s we will now understand an *arbitrary* random variable $\boldsymbol{F}$ taking its values in the set of all deterministic $(\max, +)$ circuits of size at most s. That is, we now do not insist that the randomness into the circuits can be only introduced via random input variables. Such a circuit *solves* a given maximization problem $f : R^n \to R$ with *one-sided* success probability $0 \le p \le 1$ if for every input weighting $x \in \mathbb{R}_+^n$, we have

$$\Pr\{\boldsymbol{F}(x) > f(x)\} = 0 \text{ and } \Pr\{\boldsymbol{F}(x) < f(x)\} \le 1 - p.$$

That is, the circuit is not allowed to output any better than "optimum" value $f(x)$, but is allowed to output worse values with probability at most $1 - p$. In particular, $p = 1$ means that the circuit must correctly compute f, while $p = 0$ means that the circuit can do "almost everything," it only must never output better than optimal values.

As in Section 10.1, we will use the approach of isolating sets. Let f be an n-variate $(\max, +)$ polynomial. A set $X \subseteq \mathbb{R}_+^n$ of input weights is *isolating* for f if for every n-variate $(\max, +)$ polynomial $h \in \mathcal{H}$,

$$h(x) = f(x) \text{ for all } x \in X \text{ implies that } h(x) = f(x) \text{ holds for all } x \in \mathbb{R}_+^n.$$

In the case of tropical polynomials, we do not have such a strong isolation fact as Lemma 6. Still, also in the tropical case, some specific sets of input weighings are isolating. In the case of $(\max, +)$ polynomials, such is the set of all 0-1 weighings.

Lemma 7. *Let f be a $(\max, +)$ polynomial of n variables. If f is multilinear and monic, then the set $X = \{0,1\}^n$ is isolating for f.*

Proof. Let $f(x) = \max_{a\in A}\langle a, x\rangle + c_a$ be a $(\max, +)$ polynomial. Since f is multilinear, we have $A \subseteq \{0,1\}^n$, and since f is monic, we also have $c_a = 0$ for all $a \in A$. Now take an arbitrary $(\max, +)$ polynomial $h(x) = \max_{b\in B}\langle b, x\rangle + c_b$, and suppose that

$$h(x) = f(x) \text{ holds for all input weightings } x \in \{0,1\}^n. \tag{12}$$

Our goal is to show that then $h(x) = f(x)$ also holds for all nonnegative real weighings $x \in \mathbb{R}^n_+$.

Since the polynomial f is monic, $f(\vec{0}) = 0$ holds for the all-0 input weighting $\vec{0}$. Together with Eq. (12), this yields $h(\vec{0}) = 0$. Since the "coefficients" $c_b \in \mathbb{R}_+$ of the polynomial must be nonnegative, and since the polynomial h takes the *maximum* of the values $\langle b, x\rangle + c_b$, the equality $h(\vec{0}) = 0$ implies $c_b = 0$ for all $b \in B$. So, both polynomials f and h are monic.

Furthermore, since $g(x) = f(x)$ must hold for each of n input weighings $x \in \{0,1\}^n$ with exactly one 1, all vectors in B must also be 0-1 vectors. So, both polynomials f and h are monic and multilinear; in particular, $B \subseteq \{0,1\}^n$ holds. The vectors a in A and B can be therefore identified with their supports $S_a = \{i\colon a_i = 1\}$. We claim that:

(i) the support of every vector of B lies in the support of at least one vector of A, and
(ii) the support of every vector of A lies in the support of at least one vector of B.

Now, for every input weighting $x \in \mathbb{R}^n_+$, property (i) gives the inequality $h(x) \le f(x)$, while (ii) gives the converse inequality. It thus remains to prove the properties (i) and (ii). To show (i), suppose contrariwise that there is a vector $b \in B$ such that $S_b \setminus S_a \neq \emptyset$ holds for all $a \in A$. Then on the 0-1 input $x = b \in \{0,1\}^n$, we have $g(x) \ge \langle b, x\rangle = \langle b, b\rangle = |S_b|$. But since every vector $a \in A$ has a zero in some position $i \in S_b$, we have $\langle a, x\rangle = \langle a, b\rangle \le |S_b| - 1$ and, hence, also $f(x) \le |S_b| - 1$, a contradiction with Eq. (12). The argument for the property (ii) is the same with the roles of A and B interchanged. □

Theorem 7. *If a multilinear and monic* $(\max, +)$ *polynomial* f *can be computed by a probabilistic* $(\max, +)$ *circuit of size* s *with one-sided success probability* $p > 0$ *then* f *can be also computed by a deterministic* $(\max, +)$ *circuit of size at most* $(s+1)\lceil n/p\rceil$.

Note that the size of the obtained deterministic circuits remains proportional to ns even if the success probability $p > 0$ is an arbitrarily small constant. This is in sharp contrast with the two-sided error scenario, where we required the success probability to be $p \ge 1/2 + c$ for a constant $c > 0$ (for definiteness, we have used $p = 2/3$).

Proof. By Lemma 7, we know that the set $X = \{0,1\}^n$ isolates f within all $(\max, +)$ polynomials. Let $\boldsymbol{F}$ be a probabilistic $(\max, +)$ circuit of size s computing f with a one-sided success probability $p > 0$. Take $m = \lceil (1/p)\log|X|\rceil = \lceil n/p\rceil$ independent copies $\boldsymbol{F}_1, \ldots, \boldsymbol{F}_m$ of the circuit $\boldsymbol{F}$, and consider the probabilistic $(\max, +)$ circuit $\boldsymbol{H}(x) = \max\{\boldsymbol{F}_1(x), \ldots, \boldsymbol{F}_m(x)\}$.

Fix a vector $x \in X$. Since only *one-sided* error $\epsilon = 1 - p$ is allowed, we know that $\boldsymbol{F}_i(x) \le f(x)$ must hold for all i. Hence, $\boldsymbol{H}(x) \neq f(x)$ can only happen when *all* the values $\boldsymbol{F}_1(x), \cdots, \boldsymbol{F}_m(x)$ are strictly smaller than the optimal value $f(x)$, and this can only happen with probability at most $\epsilon^m = (1-p)^m \le \mathrm{e}^{-pm}$. So, by the union bound, the probability that $\boldsymbol{H}(x) \neq f(x)$ holds for at least one of the inputs $x \in X$ does not exceed $|X|\epsilon^m \le |X|\mathrm{e}^{-pm}$, which is smaller than 1, because $m \ge (1/p)\log|X|$ (and $\log \mathrm{e} > 1$).

There must therefore be a realization $H(x) = \max\{F_1(x), \ldots, F_m(x)\}$ of the probabilistic circuit $\boldsymbol{H}$ such that the polynomial $h(x)$ computed by $H(x)$ satisfies $h(x) = f(x)$ for all $x \in X$. The size of the obtained deterministic circuit $H(x)$ is at most $ms + m - 1 \le (s+1)\lceil n/p\rceil$. Since the set X is isolating for f, the fact that $h(x) = f(x)$ holds for all $x \in X$ implies this implies $h(x) = f(x)$ holds for all $x \in \mathbb{R}^n_+$, that is, the obtained deterministic circuit H correctly computes f on all possible inputs. □

ACKNOWLEDGMENTS

I am thankful to Joshua Grochow, Pascal Koiran, Igor Sergeev and Hans Ulrich Simon for inspiring discussions at the initial stages of this investigation. I am also thankful to the referees for very helpful comments and suggestions. This work is supported by the German Research Foundation (DFG) under Grant JU 3105/1-1.

REFERENCES

[1] L.M. Adleman. 1978. Two theorems on random polynomial time. In *Proc. of 19th Ann. IEEE Symp. on Foundations of Computer Sci. (FOCS)*. 78–83.
[2] M. Ajtai and M. Ben-Or. 1984. A theorem on probabilistic constant depth computations. In *Proc. of 16th Ann. ACM Symp. on Theory of Computing (STOC)*. 471–474.
[3] M. Ajtai, J. Komlós, and E. Szemerédi. 1983. Sorting in $c \log n$ parallel steps. *Combinatorica* 3, 1 (1983), 1–19.
[4] N. Alon and E. R. Scheinerman. 1988. Degrees of freedom versus dimension for containment orders,. *Order* 5 (1988), 11–16.
[5] S. Basu, R. Pollack, and M.-F. Roy. 1996. On the combinatorial and algebraic complexity of quantifer elimination. *J. ACM* 43, 6 (1996), 1002–1045.
[6] S. Ben-David and M. Lindenbaum. 1998. Localization vs. identification of semi-algebraic sets. *Machine Learning* 32 (1998), 207–224.
[7] C.H. Bennett and J. Gill. 1981. Relative to a random oracle A, $P^A \neq NP^A \neq \text{co-}NP^A$ with probability 1. *SIAM J. Comput.* 10, 1 (1981), 96–113.
[8] P. Bürgisser, M. Karpinski, and T. Lickteig. 1993. On randomized semi-algebraic test complexity. *J. Complexity* 9, 2 (1993), 231–251.
[9] F. Cucker, M. Karpinski, P. Koiran, T. Lickteig, and K. Werther. 1995. On real Turing machines that toss coins. In *Proc. of 27th Ann. ACM Symp. on Theory of Computing (STOC)*. 335–342.
[10] D. Dubhashi and A. Panconesi. 2009. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press.
[11] R.M. Dudley. 1984. *A course on empirical processes*. Lecture Notes in Mathematics, Vol. 1097. Springer.
[12] P. Goldberg and M. Jerrum. 1995. Bounding the Vapnik-Chervonenkis dimension of concept classes parametrized by real numbers. *Machine Learning* 18 (1995), 131–148.
[13] O. Goldreich. 2011. In a world of P = BPP. In *Studies in Complexity and Cryptography*. Lect. Notes in Comput. Sci., Vol. 6650. Springer, 191–232.
[14] D. Grigoriev. 1999. Complexity lower bounds for randomized computation trees over zero characteristic fields. *Computational Complexity* 8, 4 (1999), 316–329.
[15] D. Grigoriev and M. Karpinski. 1997. Randomized $\Omega(n^2)$ lower bound for knapsack. In *Proc. of 29-th Ann. ACM Symp. on Theory of Computing (STOC)*. 76–85.
[16] D. Grigoriev, M. Karpinski, F. Meyer auf der Heide, and R. Smolensky. 1997. A lower bound for randomized algebraic decision trees. *Computational Complexity* 6, 4 (1997), 357–375.
[17] D. Haussler. 1992. Decision theoretic generalizations of the PAC model for neural nets and other learning applications. *Inf. Comput.* 100 (1992), 78–150.
[18] R. Impagliazzo and A. Wigderson. 1997. P = BPP unless E has subexponential circuits: derandomizing the XOR lemma. In *Proc. of 29th ACM Symp. on Theory of Computing (STOC)*. 220–229.
[19] U. Manber and M. Tompa. 1985. The complexity of problems on probabilistic, nondeterministic, and alternating decision trees. *J. ACM* 32, 3 (1985), 720–732.
[20] A.A. Markov. 1958. On the inversion complexity of systems of Boolean functions. *J. ACM* 5, 4 (1958), 331–334.
[21] F. Meyer auf der Heide. 1985. Simulating probabilistic by deterministic algebraic computation trees. *Theor. Comput. Sci.* 41 (1985), 325–330.
[22] J. Milnor. 1964. On the Betti numbers of real varieties. *Proc. Amer. Math. Soc.* 15 (1964), 275–280.
[23] M. Mitzenmacher and E. Upfal. 2005. *Probability and computing: randomized algorithms and probabilistic analysis*. Cambridge University Press.
[24] H. Morizumi. 2012. *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.
[25] R. Motwani and P. Raghavan. 1995. *Randomized Algorithms*. Cambridge University Press.
[26] D. Pollard. 1984. *Convergence of Stochastic Processes*. Springer-Verlag.
[27] P. Pudlák and V. Rödl. 1992. A combinatorial approach to complexity. *Combinatorica* 12, 2 (1992), 221–226.
[28] N. Saxena. 2009. Progress on polynomial identity testing. *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS* 99 (2009), 49–79.

[29] J.T. Schwartz. 1980. Fast probabilisitic algorithms for verification of polynomial identities. *J. ACM* 27, 4 (1980), 701–717.
[30] A. Seidenberg. 1954. A new decision method for elementary algebra. *Ann. of Math.* 60 (1954), 365–374.
[31] A. Shpilka and A. Yehudayoff. 2010. Arithmetic Circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science* 5, 3-4 (2010), 207–388.
[32] M. Snir. 1985. Lower bounds on probabilistic linear decision trees. *Theor. Comput. Sci.* 38 (1985), 69–82.
[33] A. Tarski. 1951. *A decision method for elementary algebm and geometry* (2nd ed.). University of California Press, Berkeley and Los Angeles, Calif.
[34] V.N. Vapnik and A.Ya. Chervonenkis. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.* 16 (1971), 264–280.
[35] H.E. Warren. 1968. Lower bounds for approximation by non-linear manifolds. *Trans. Amer. Math. Soc.* 133 (1968), 167–178.
[36] R. Zippel. 1979. Probabilistic algorithms for sparse polynomials. In *Lect. Notes in Comput. Sci.*, Vol. 72. Springer, 216–226.

A CIRCUITS FOR MAJORITY VOTE

Recall that the *majority vote* function of m variables is a partly defined function $\mathrm{Maj}_n(x_1,\ldots,x_n)$ that outputs the majority element of its input string $x_1,\ldots,x_n$, if there is one.

Claim 6. *Arithmetic* $(+,-,\times)$ *circuits, as well as tropical* $(\min,+)$ *and* $(\max,+)$ *circuits cannot compute majority vote functions.*

Proof. Functions computed by circuits over the arithmetic basis $\{+,-,\times\}$ are polynomial functions. So, suppose contrariwise that we can express $\mathrm{Maj}(x,y,z)$ as a polynomial $f(x,y,z) = ax+by+cz+h(x,y,z)$, where the polynomial 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.

Let us now show that also tropical circuits cannot compute majority vote functions. Every tropical $(\min,+)$ circuit computes some tropical $(\min,+)$ polynomial. The functions $f:\mathbb{R}^n\to\mathbb{R}$ computed by tropical $(\min,+)$ polynomials are piecewise linear *concave* functions. In particular, $f(\frac{1}{2}x+\frac{1}{2}y)\geq\frac{1}{2}f(x)+\frac{1}{2}f(y)$ must hold for all $x,y\in\mathbb{R}^n$:

$$\min_{v\in V}\langle v,x+y\rangle \geq \min_{v\in V}\langle v,x\rangle + \min_{v\in V}\langle v,y\rangle\,.$$

But already the majority vote function $\mathrm{Maj}:\mathbb{R}^3\to\mathbb{R}$ of three variables is not concave. To see this, take two input vectors $x=(a,a,c)$ and $y=(a,b,b)$ with $a<b$ and $c=2a-b$. Then $\mathrm{Maj}(\frac{1}{2}x+\frac{1}{2}y)=\mathrm{Maj}(a,(a+b)/2,a)=a$ but $\frac{1}{2}\mathrm{Maj}(x)+\frac{1}{2}\mathrm{Maj}(y)=\frac{1}{2}a+\frac{1}{2}b>a$ since $b>a$. So, Maj is not concave. Similar argument shows that Maj is not *convex* and, hence, cannot be computed by tropical $(\max,+)$ circuits. □

Recall that a binary relation $\varrho\subseteq\mathbb{R}\times\mathbb{R}$ *contiguous* if $x\leq y\leq z$, $x\,\varrho\,a$ and $z\,\varrho\,a$ imply $y\,\varrho\,a$. That is, if the endpoints of an interval are close to a, then also all numbers in the interval are close to a.

Claim 7. *For every contiguous relation* $x\,\varrho\,y$*, a majority* ϱ*-vote function of* m *variables can be computed by a fanin-2* $(\min,\max)$ *circuit of size* $O(m\log m)$.

Proof. Given a sequence $x_1,\ldots,x_m$ of real numbers, the *median function* outputs the middle number $x_{i_{\lceil m/2\rceil}}$ of the sorted sequence $x_{i_1}\leq\ldots\leq x_{i_m}$. So, the sorting network of Ajtai, Komlós and Szemerédi [3] computes the median function using only $O(m\log m)$ min and max operations. On the other hand, it is easy to see that the median function is a majority ϱ-vote function for every contiguous relation $x\,\varrho\,y$.

Indeed, let $x_1 \le \ldots \le x_m$ be a sorted sequence of real numbers, and a a real number. Call a position i *good*, if $x_i \varrho\, a$ holds. Suppose that more than half of the positions i are good. Since the relation ϱ is contiguous, good positions constitute a contiguous *interval* of length $> m/2$. So, the median of $x_1, \ldots, x_m$ must be the number x_i in a good position i. □

Recall that the *nullity relation* $x \varrho\, y$ holds precisely when either both $x = 0$ and $y = 0$, or both $x \neq 0$ and $y \neq 0$ hold. A *zero vote function* of n variables is any function $f : \mathbb{R}^n \to \mathbb{R}$ such that $f(x_1, \ldots, x_n) = 0$ precisely when more than $n/2$ of the numbers x_i are zeros. Note that every zero-vote function is a majority ϱ-vote function for the nullity relation ϱ : either more than half of all numbers $x_1, \ldots, x_n$ are zeros, or more than half of them are nonzero.

Claim 8. *A zero-vote function of n variables can be computed a* $(\min, \max, \times)$ *circuit of size* $O(n \log n)$*, as well as by a monotone fanin-2 arithmetic* $(+, \times)$ *circuit of size* $O(n^2)$.

Proof. First, suppose that we have $(\min, \max, \times)$ among the basis operations. Then we can just sort the sequence $x_1^2, \ldots, x_n^2$ of squares using $O(n \log n)$ $(\min, \max)$ gates, and output the median of the sorted sequence $y_1 \le \ldots \le y_n$. Since the squared sequence has only nonnegative numbers, zeros (if any) will lie at the beginning of the sorted sequence.

In the case of arithmetic $(+, \times)$ circuits, we can use the standard dynamic programming. We have only to show how to efficiently compute polynomials $P_{m,k}$ such that $P_{m,k}(x_1, \ldots, x_m) = 0$ precisely when at least k of the numbers $x_1, \ldots, x_m$ are zeros. For the base cases, we can take $P_{m,k}(x_1, \ldots, x_m) = x_1^2 \cdots x_m^2$ for $k = 1$, $P_{m,k}(x_1, \ldots, x_m) = x_1^2 + \cdots + x_m^2$ for $k = m$, and $P_{m,k}(x_1, \ldots, x_m) = 1$ $(\neq 0)$ for $k > m$. (We take squares just to avoid possible cancelations.) Then we can use the recursion

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

The first polynomial $P_{m-1,k}$ in this product is 0 iff there are at least k zeros already among the first $m-1$ positions, whereas the second term is 0 iff there are at least $k-1$ zeros among the first $m-1$ positions, and the last position is also zero. For $m = n$ and $k = \lfloor n/2 \rfloor + 1$, the obtained arithmetic $(+, \times)$ circuit has size $O(kn) = O(n^2)$, and computes the zero vote function. □

B NOTES ON MEASURABILITY

In the uniform convergence result of Vapnik and Chervonenkis given in Theorem 5, the class of functions H is required to be permissible (see Remark 4). While every *countable* class H is permissible, uncountable classes need not automatically be such.

Haussler in [17, Appendix 9.2] gives a sufficient condition for a class H of (not necessarily 0-1 valued) functions $h : X \to \mathbb{R}$ to be permissible. He calls a class H *indexed* by a set T if there is a real valued function f on $T \times X$ such that $H = \{f(t, \cdot) : t \in T\}$, where $f(t, \cdot)$ denotes the real-valued function on X obtained from f by fixing the first parameter to t. Haussler shows that the following conditions are already sufficient for the class H to be permissible: (1) every function $h \in H$ is measurable, (2) the class H can be indexed by a set $T = \mathbb{R}^n$ for a finite $n \ge 1$, and (3) the indexing function $f : T \times X \to \mathbb{R}$ itself is measurable.

In the case of Boolean semialgebraic matrices $M : T \times X \to \{0, 1\}$, we have a class H of 0-1 functions $h_t : X \to \{0, 1\}$, where $X = \mathbb{R}^k$ and $h_t(x) = M[t, x]$. The class H is indexed by the set T of the form $T = \mathbb{R}^n$, and the indexing function $f = M$ is the matrix M itself. Since the matrix M is semialgebraic, the functions $h_t \in H$ as well as the indexing function f are semialgebraic. Since the functions h_t and the indexing function f are 0-1 valued functions, this implies that all these functions are measurable.

Indeed, every semialgebraic set $S \subseteq \mathbb{R}^n$ is a *finite* union of *finite* intersections of sets of the form $\{x \in \mathbb{R}^n : p(x) = 0\}$ and $\{x \in \mathbb{R}^n : p(x) > 0\}$, where p is a polynomial. So, semialgebraic sets are

measurable. Recall that a function $h : X \to \mathbb{R}$ is measurable if the set X itself is a measurable set, and for each real number r, the set $S_r = \{x \in X : h(x) > r\}$ is measurable. In our case, functions $h : X \to \{0, 1\}$ are 0-1 valued functions. Each such function is the characteristic function of the set $S = \{x \in X : h(x) = 1\}$. Then each set S_r is either $\emptyset$, S or X. Hence, a 0-1 valued function h is measurable if and only if the set $S = h^{-1}(1)$ it represents is measurable. Since semialgebraic sets are measurable, we have that every semialgebraic 0-1 valued function is measurable.

The books of Dudley [11, Chapter 10] and Pollard [26, Appendix C] discuss more general sufficient conditions for classes of not necessarily 0-1 valued functions $h : X \to \mathbb{R}$ to be permissible.

Received 20.08.2018; revised 24.04.2019; accepted 0