Coin Flipping in Dynamic Programming Is Almost Useless

STASYS JUKNA*†, Goethe University Frankfurt, Germany

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), 30 pages. https://doi.org/S.Jukna

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 [36, 39]. 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.

*Research supported by the DFG grant JU 3105/1-1 (German Research Foundation).
†Also with Institute of Data Science and Digital Technologies, Faculty of Mathematics and Computer Science, Vilnius University, Lithuania.
[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, Institute of Computer Science, Goethe University Frankfurt, Robert-Mayer-Str. 11-15, Frankfurt am Main, 60325, Germany, stjukna@gmail.com.

1942-3454/0/0-ART0
https://doi.org/S.Jukna

A classical result of Adleman [1], extended to the case of two-sided error probability by Bennett and Gill [9], 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 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 efficiently 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. Actually, we will 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, 9] 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 5 in Section 4.3). 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 [45, 48]. 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 more details). All operations mentioned in the previous paragraph are semialgebraic of small description complexity; see Table 1 in Section 3 for more examples.

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, $\mathrm{Maj}(x_1, \ldots, x_m) = y$ if y occurs $> m/2$ times among the $x_1, \ldots, x_m$. 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 $b \le m$; see Table 1 in Section 3.

A *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 2-semialgebraic.

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 $m = O(n^2 s \log bs)$ 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 copies of the probabilistic circuit is only *logarithmic* is the description complexity b of individual gates.

Our next result extends Theorem 1 to the case when circuits are only required to *approximate* the values of a given function. 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 to the correct value $f(x)$ with probability at least $2/3$. A relation ϱ is *contiguous* if$x\, \varrho\, a$, $z\, \varrho\, a$ and $x \le y \le z$ imply $y\, \varrho\, a$. For example, if $x\, \varrho\, a$ holds precisely then x lies in an interval of a given fixed length around the number a, then ϱ is contiguous.

In the following theorem, $x\, \varrho\, y$ is any contiguous binary relation of description complexity t_ϱ, $f : \mathbb{R}^n \to \mathbb{R}$ is any semialgebraic function of description complexity t_f, and $\mathcal{B}$ is a basis consisting of b-semialgebraic gates. We assume that $\min(x, y)$ and $\max(x, y)$ operations belong to $\mathcal{B}$.

Theorem 2. *If f can be ϱ-approximated by a probabilistic circuit of size s over a basis $\mathcal{B}$, then f can also be ϱ-approximated by a deterministic circuit over $\mathcal{B}$ of size $O(ms + m \log m)$, where $m = n^2 s \log(sb + t_f + t_\varrho)$.*

Note that now, unlike in Theorem 1, the derandomized circuit is in a "pure" form, that is, it only uses gates from the basis $\mathcal{B}$ (no additional majority vote gates). This happens because we now require that min and max must be among the basis operations: the majority vote function of m variables can be computed using only $O(m \log m)$ min and max gates (see Claim 6 in Section 5.1).

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, 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* set 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* set of recursion equations valid for inputs of *all* dimensions. What we obtain is a *sequence* of deterministic DP algorithms, one for each dimension n. 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. In Section 2, we shortly recall previous work concerning the derandomization of circuits and decision trees working over infinite domains. In Section 3, we recall the notions of semialgebraic functions and probabilistic circuits. Theorem 1 is proved in Section 4.4, and Theorem 2 is proved in Section 5.2. Sections 6 and 7 are devoted to *tropical* circuits, that is, (min, +) and (max, +) circuits. Besides their own interest (tropical algebra and geometry are now actively studied topics in mathematics), these circuits are also important in the context of dynamic programming because many basic DP algorithms are just special (recursively constructed) tropical circuits. The paper is organized as follows.

(1) Derandomization of *exactly computing* semialgebraic circuits (Section 4).
(2) Derandomization of *approximating* semialgebraic circuits (Section 5).
(3) Derandomization of tropical circuits under the *one-sided error* scenario (Section 6).
(4) A Boolean *lower bound* for probabilistic tropical circuits (Section 7).

Results (1) and (2) are obtained by a proper combination of 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. Results (3) and (4) are obtained using direct and elementary arguments.

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 [9] used a simple "finite majority rule" (Lemma 5 in Section 4.3). 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.

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 [33] has found a surprisingly tight characterization of the minimum number of NOT gates required by *deterministic* $(\vee, \wedge, \neg)$ circuits to compute a given Boolean functions f in terms a natural combinatorial characteristic of f. A natural question therefore was: can randomness substantially reduce the number of NOT gates? Morizumi [38] has shown that Markov's result 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 have 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 7 below) following from the uniform convergence in probability results 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, the inclusion BPP $\subseteq$ P/poly was proved by Manber and Tompa [32], and Snir [49].

[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."

Meyer auf der Heide [34] proved the inclusion when arbitrary rational functions are allowed. He uses a result of Milnor [35] about the number of connected components of polynomial systems in $\mathbb{R}^n$ to upper-bound the minimum size of an "isolating" subset $X \subset \mathbb{R}^n$. Further explicit lower bounds on the depth of probabilistic decision trees were proved by Bürgisser, Karpinski and Lickteig [10], Grigoriev and Karpinski [20], Grigoriev et. al. [21], Grigoriev [19] and other authors. In [29], we have used Approach (A) to show that in the case of arithmetic $(+, -, \times, \div)$ circuits, randomization cannot spare even one single gate. In Section 6, we will also use approach (A) to derandomize probabilistic (min, +) and (max, +) circuits under the one-sided error probability scenario.

Approach (B) was used by Cucker et. al. [11] to prove the inclusion BPP $\subseteq$ P/poly for algebraic circuits over the basis $\{+, -, \times, \div, \mathrm{sgn}\}$. They combined the upper bound on the Vapnik–Chervonenkis dimension (VC dimension) of such circuits, obtained by Goldberg and Jerrum [17], with a uniform convergence in probability theorem of Haussler [22] 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 somewhat different, more direct way avoiding the detour through VC dimension (and Sauer's [44] lemma). Namely, we will directly combine the classical uniform convergence in probability theorem of Vapnik and Chervonenkis [51] with the upper bound of Warren [52] 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, 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 [24]: either BPP = P holds or *every* decision problem solvable by deterministic Turing machines in time $2^{O(n)}$ can be solved by Boolean circuits of sub-exponential size $2^{o(n)}$. Goldreich [18] 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

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. What such a circuit computes is a random function whose values depend on the values of the random input variables. Thus, a probabilistic circuit is specified by giving a deterministic circuit $F(x, y)$ of $n + k$ variables, together with some probability distribution $\Pr : \mathbb{R}^k \to [0, 1]$ of random variables. A probabilistic circuit $F(x, \boldsymbol{r})$ *computes* a given function $f : \mathbb{R}^n \to \mathbb{R}$ if $\Pr\left\{r \in \mathbb{R}^k : F(x, r) = f(x)\right\} \geq 2/3$ holds for each input $x \in \mathbb{R}^n$. That is, for every input $x \in \mathbb{R}^n$, the

circuit must output the correct value $f(x)$ with probability[4] at least $2/3$. We will sometimes call circuits *without* random inputs *deterministic*, just to distinguish them from probabilistic ones. In Section 5, we will relax the condition till "$F(x,r)$ approximates the value $f(x)$ with probability at least $2/3$."

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* $\operatorname{sgn} : \mathbb{R} \to \{-1,0,+1\}$ takes value $\operatorname{sgn} x = -1$ if $x < 0$, $\operatorname{sgn} 0 = 0$, and $\operatorname{sgn} x = +1$ 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

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

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 $m \le t$ 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, that is, if $x \in S$ and $x' \notin S$, then $\operatorname{sgn} P(x) \neq \operatorname{sgn} P(x')$.

A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is *t-semialgebraic* if its graph $S = \{(x,y) \colon y = f(x)\} \subseteq \mathbb{R}^{n+m}$ 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.

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 $>, \ge, =, \neq, \le, <$, 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 maximum degree.

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 $[p \le 0]$ by the formula $[p = 0] \vee [p < 0]$, each atomic predicate $[p \ge 0]$ 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. Note that, actually, any *two* of these three forms $[p < 0]$, $[p = 0]$ and $[p > 0]$ of atomic predicates suffice because each of these predicates is equivalent to the AND of the negations of the remaining two predicates.

An algebraic formula $\Phi(x)$ *defines* a set $S \subseteq \mathbb{R}^n$ if $S = \{x \in \mathbb{R}^n \colon \Phi(x) = 1\}$.

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.

[4]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.

Table 1. Examples of semialgebraic functions, where m is the number of distinct polynomials used in a formula, and d is their maximum degree, Here, $p(x)$ is an arbitrary real multivariate polynomial of degree d, and $\Psi(x)$ is a semialgebraic 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 formula Φ for the majority vote function, maj is the Boolean majority function.

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

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 $x' \notin S$. (In fact, then vectors $G(x)$ and $G(x')$ will differ in at least two positions.) 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 (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 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 $\diamond$.

Example 1 (Sorting operation). The *sorting operation* sort : $\mathbb{R}^n \to \mathbb{R}^n$ takes a sequence $x_1, \ldots, x_n$ of real numbers, and outputs its ordered permutation $y_1 \le y_2 \le \ldots \le y_n$. The graph of this operation can be defined by the following algebraic formula of $2n$ variables:

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

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 sort : $\mathbb{R}^n \to \mathbb{R}^n$ is t-semialgebraic for $t = n^2 + n - 1$.

Number of zero 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 of n variables can have? A trivial upper bound is $|\{-1, 0, +1\}^m| = 3^m$.

A fundamental result of Warren [52, 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 3 (Warren [52]). *No sequence of $m \ge n$ polynomials in $\mathbb{R}[x_1, \ldots, x_n]$ of degree at most $d \ge 1$ 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 by several authors, including Alon and Scheinerman [4], Pudlák and Rödl [41], Goldberg and Jerrum [17], 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 $|X| \le 3^m$ vectors witnessing all distinct sign patterns of this sequence. Take

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

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.

Remark 3. The condition $m \ge n$ in Warren's upper bound on the number W of all $(-1, +1)$ sign patterns is not crucial: it comes just from trying to simplify the form of this bound. His general upper bound on W holds for *any* parameters $n, m, d \ge 1$, and is of the form

$$W \le 2(2d)^n \sum_{k=0}^{n} 2^k \binom{m}{k},$$

where $\binom{m}{k} = 0$ when $k > m$. He then just shows that for $m \ge n$, Stirling's formula yields a more handy upper bound $W \le (4emd/n)^n$. On the other hand, for $1 \le m < n$, the binomial theorem yields $W \le 2(2d)^n 3^m \le (6d)^n$.

What function 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 must vanish 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 $S_i = \{x \in \mathbb{R}^n : p_i(x) = 0, q_{i,1}(x) > 0, \ldots, q_{i,k_i}(x) > 0\}$, 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 2. 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 sucht that $n \leq x$. The interior of the graph $S = \{(x, y) \in \mathbb{R} \times \mathbb{Z} : \lfloor x \rfloor = y\}$ 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.

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

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

  over all choices of m columns, 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 $m \geq 1$.

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.

- The *graph matrix* of $F(x, \boldsymbol{r})$ is the Boolean matrix $M_F : \mathbb{R}^{n+1} \times \mathbb{R}^k \to \{0, 1\}$ 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 a 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 $\geq 2/3$. So, the correctness matrix M is then 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 $f(x)$.

So, 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 semialgebraic circuit with n deterministic input variables, and with s gates, each of description complexity at most b.

Step 1 The description complexity of the graph matrix M_F of F is $t \leq (bs)^{Cns}$ for a constant C. (Lemma 3). Here we will use a result of Basu, Pollack and Roy [6] on the quantifier elimination.

Step 2 The growth function of the graph matrix M_F satisfies $\Pi_{M_F}(m) \leq (8emt^2/n)^n$ (Lemma 4). Here we will use Warren's theorem (Theorem 3) on sign patterns of polynomials.

Step 3 Every probabilistically dense submatrix of M_F has the m-majority property for any $m \geq 2/c$ satisfying $\Pi_{M_F}(m) \leq \mathrm{e}^{cm}$, where $c > 0$ is an absolute constant (Lemma 6). Here we will use the uniform convergence in probability theorem of Vapnik and Chervonenkis [51].

Now, if the circuit $F(x, \boldsymbol{r})$ computes a given function $f : \mathbb{R}^n \to \mathbb{R}$, then the correctness matrix M with respect to this function f, *is* a probabilistically dense submatrix of M_F *per se*. By Steps 1–3, the matrix M has the m-majority property for any m satisfying the inequality $\left(8emt^2/n\right)^n \leq \mathrm{e}^{cm}$ for $\log t = O(ns \log bs)$, from which the upper bound $m = O(n^2 s \log bs)$ given in Theorem 1 follows.

4.1 Step 1: Description complexity of circuits

An important consequence of the Tarski–Seidenberg theorem [47, 50]–stating that every *quantified* algebraic formula has an equivalent quantifier-free formula—is that compositions of semialgebraic functions are also semialgebraic functions. In particular, this implies that functions computable by circuits over any basis consisting of semialgebraic functions are also semialgebraic. We, however, are interested in the *quantitative* aspect of this theorem:

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

The answer is given in Lemma 3 bellow. To prove the lemma, we first turn a semialgebraic circuit into a quantified algebraic formula (Lemma 1), and then use a known result on quantifier elimination over the reals (Lemma 2).

An *existential* algebraic formula with q quantifiers and n free variables is a formula of a form

$$(\exists z_1 \in \mathbb{R})\ (\exists z_2 \in \mathbb{R})\ \ldots\ (\exists z_q \in \mathbb{R})\ \ \Phi(x_1, \ldots, x_n, z_1, \ldots, z_q)\,,$$

where Φ is a (quantifier-free) algebraic formula.

Lemma 1. *If a function $f : \mathbb{R}^n \to \mathbb{R}$ is computable by a deterministic b-semialgebraic circuit of size at most s, then the graph of f can be defined by an existential algebraic formula using at most sb polynomials of degree at most b, and at most $s - 1$ quantifiers.*

Proof. See Appendix A. □

The following lemma is a special case of a more general result of Basu, Pollack and Roy [6, Theorem 1.3.1] for quantified algebraic formulas where a bounded number of blocks of alternations of existential and universal quantifiers is allowed. For our purposes, existential quantifiers alone will suffice.

Lemma 2 (Basu, Pollack and Roy [6]). *If an existential algebraic formula has n free variables, q quantified variables and uses ℓ polynomials of degree at most d, then there is an equivalent quantifier-free algebraic formula which uses at most $(\ell d)^{Cnq}$ polynomials of degree at most d^{Cq}, where C is an absolute constant.*

Remark 4. A similar result with a worse bound $(\ell d)^{Cq}$ on the degree of the quantifier-free formula was earlier proved by Renegar [43]. In fact, both results [6, 43] are more general, and hold also for quantified formulas using both quantifiers $\exists$ and $\forall$: if there are ω blocks of alternating quantifiers with q_i variables in the i-th block, then the same upper bound holds with q replaced by the product $2^\omega q_1 \cdots q_\omega$ [43], and even by only $q_1 \cdots q_\omega$ [6].

The following direct consequence of Lemmas 1 and 2 answers the question asked at the beginning of this section.

Lemma 3. *If a function $f : \mathbb{R}^n \to \mathbb{R}$ is computable by a deterministic b-semialgebraic circuit of size at most s, then f is t-semialgebraic for t satisfying $\log t = O(ns \log bs)$.*

Proof. By Lemma 1, the graph of the function $f : \mathbb{R}^n \to \mathbb{R}$ can be defined by an existential algebraic formula of size $\ell \le sb$, degree b and with $q \le s-1$ quantifiers. Lemma 2 yields a quantifier-free algebraic formula which also defines the graph of f, has size and degree $t \le (\ell b)^{Cnq} \le (sb^2)^{Cns}$, as desired. □

4.2 Step 2: Growth functions from description complexity

A Boolean matrix $M : \mathbb{R}^n \times \mathbb{R}^k \to \{0,1\}$ is *semialgebraic* if the set $S = \{(x,y) \in \mathbb{R}^{n+k} : M[x,y] = 1\}$ of its 1-entries is such. The *description complexity* of a column $r \in \mathbb{R}^k$ is the description complexity of the set $S_r = \{x \in \mathbb{R}^n : M[x,r] = 1\}$ of its 1-entries.

Note that the description complexity of individual columns does not exceed the description complexity of the entire matrix, but may be smaller, in general. Moreover, the description complexity of columns may be bounded even if the matrix itself is not semialgebraic. Consider, for example, the matrix $M : \mathbb{R} \times \mathbb{R} \to \{0,1\}$ whose entries are defined by: $M[x,y] = 1$ if and only if $x = \lfloor y \rfloor$. The matrix is not semialgebraic (see Example 2), 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) number. Hence, the description complexity of each individual column is 1.

Lemma 4. *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 $x \in A$. So, $\Pi_M(m)$ is at most the number of distinct 0-1 patterns $\big(\Phi_1(x), \ldots, \Phi_m(x)\big)$ when x ranges over the entire set $\mathbb{R}^n$ of row labels.

Let $p_1, \ldots, p_s$ be all polynomials $\mathbb{R}[x_1, \ldots, x_n]$ used in at least one of the formulas $\Phi_1, \ldots, \Phi_m$. So, we have a sequence of $n \le s \le tm$ n-variate polynomials of degree at most t. By Claim 1 (in

Section 3), we can assume that the formulas Φ_i only use atomic predicates of the form $[p_i < 0]$, $[p_i = 0]$ and $[p_i > 0]$. This means that the values of formulas $\Phi_1, \ldots, \Phi_m$ can only depend on the sign-patterns of the polynomials $p_1, \ldots, p_s$. Consequently, the number of distinct 0-1 patterns $\big(\Phi_1(x), \ldots, \Phi_m(x)\big)$ cannot exceed the number of distinct sign patterns of the polynomials $p_1, \ldots, p_s$. Since the number s of polynomials satisfies $n \le s \le tm$, Warren's theorem (Theorem 3) implies that the later number cannot exceed $(8est/n)^n \le (8emt^2/n)^n$, as desired. □

4.3 Step 3: Majority property from growth 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. The following "folklore" observation shows that, if the number of rows is *finite*, then the m-majority property holds already for m about the logarithm of this number.

Lemma 5 (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 $m = O(\log |A|)$.*

Proof. Since the matrix M is probabilistically dense, there is a probability distribution $\Pr : B \to [0,1]$ such that $\Pr\{b \in B\colon M[a,b] = 1\} \ge 2/3$ holds for every row $a \in A$. Let $b_1, \ldots, b_m$ be m independent copies of b. The expected value μ of the sum $\xi = M[a,b_1] + \cdots + M[a,b_m]$ is at least $2m/3$. Thus, the event $\xi \le m/2$ implies the event $\xi \le \mu - m/3$. By the Chernoff–Hoeffding bound (see, for example, [12, Theorem 1.1]), the probability of the latter event is at most $e^{-2(m/3)^2/m} < e^{-m/5}$. By taking $m = \lceil 5 \log |A| \rceil$, this probability is strictly smaller than 1. Since we only have $|A|$ rows, the union bound implies that the matrix M has the m-majority property for this value of m. □

Lemma 5 allows us to efficiently 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 circuits simulating dynamic programming algorithms. These circuits work over *infinite* (or even uncountable) domains like $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$; elements of the domain are possible weights of items in optimization problems. So, in this case, the finite majority rule is of no use at all.

Fortunately, results from the statistical learning theory come to rescue. The classical *uniform convergence in probability* theorem of Vapnik and Chervonenkis [51] ensures the majority property also for matrices M with an infinite number of rows, as long as its growth function $\Pi_M(m)$ grows not too fast (Lemma 7 below).

4.3.1 Uniform convergence in probability. 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 $A = \{x \in X\colon h(x) = 1\}$. 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 are 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.

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,

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

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 function $h \in H$ on these points. Note that we always have $1 \leq \Pi_H(m) \leq 2^m$.

The *uniform convergence theorem* of Vapnik and Chervonenkis [51] states that if the class H is "simple" in that $\Pi_H(m)$ grows not too fast, and if we draw samples independently (with replacement) from X according to any distribution, then with high probability, the empirical frequency $\mathrm{ave}_h(\boldsymbol{x})$ of *every* function $h \in H$ will be close to the theoretical probability p_h of h.

Remark 5. In this theorem, a mild measurability condition on the class H of functions is necessary (to avoid "pathological" situations). 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 $H = \{f(t,\cdot) \colon t \in \mathbb{R}^n\}$ 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 4 (Vapnik and Chervonenkis [51]). *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*

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

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

$$m \geq 2/c \quad \text{and} \quad \Pi_H(m) \leq \mathrm{e}^{cm}\,, \tag{3}$$

then there exists a sequence $x = (x_1, \ldots, x_m)$ of (not necessarily distinct) points in X such that $\mathrm{ave}_h(x) \geq p_h - \epsilon$, that is,

$$h(x_1) + \cdots + h(x_m) \geq (p_h - \epsilon)m \tag{4}$$

holds for *all* functions $h \in H$.

Remark 6. As the constant c in Eq. (3) we can take, for example, $c := \epsilon^2/24$. Then $\Pi_H(m) \leq \mathrm{e}^{cm}$ implies $4 \cdot \Pi_H(2m) \leq 4 \cdot \mathrm{e}^{2cm} = \mathrm{e}^{\epsilon^2 m/12 + \ln 4} < \mathrm{e}^{\epsilon^2 m/12 + 2}$. For every $m \geq 2/c = 48/\epsilon^2$, we have $\epsilon^2 m/12 + 2 \leq \epsilon^2 m/8$ because then $\epsilon^2 m/8 - \epsilon^2 m/12 = \epsilon^2 m/24 \geq 2$. Thus, for $c = \epsilon^2/24$, Eq. (3) ensures that the probability in Eq. (2) is strictly smaller than 1.

4.3.2 Infinite majority rule. 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 $H = \{h_t \colon t \in T\}$ 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 6. *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. (3) holds with this choice of ϵ (by Remark 6, taking $c = \epsilon^2/24 = 1/1176$ is enough). By Eq. (4), there exists a sequence $x_1, \ldots, x_m$ of (not necessarily distinct) columns of M such that

$$h(x_1) + \cdots + h(x_m) \geq (p_h - \epsilon)\, m = \left(p_h - \tfrac{1}{7}\right) m$$

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 this submatrix. Thus, for every row $h \in H'$, we have

$$h(x_1) + \cdots + h(x_m) \geq \left(p_h - \tfrac{1}{7}\right) m \geq \left(\tfrac{2}{3} - \tfrac{1}{7}\right) m = \tfrac{11}{21} m > \tfrac{1}{2} m\,,$$

meaning that the matrix M' has the m-majority property, as desired. □

Lemma 7 (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 any 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 4 gives us an upper bound

$$\Pi_{M'}(m) \leq \Pi_M(m) \leq \left(\frac{8\mathrm{e}mt^2}{n}\right)^n . \tag{5}$$

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 6, 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. (5), in order to ensure the m-majority property for the submatrix M', it is enough that m satisfies the inequality

$$\left(\frac{8\mathrm{e}mt^2}{n}\right)^n \leq \mathrm{e}^{cm} . \tag{6}$$

By taking logarithms and setting $w = m/n$, Eq. (6) turns into the inequality $\ln w + \ln(8\mathrm{e}t^2) \leq cw$. If $w \leq 8\mathrm{e}t^2$, then it is enough that $2\ln(8\mathrm{e}t^2) \leq cw$ holds, which happens if $w = C \log t$ for a large enough constant C. If $w \geq 8\mathrm{e}t^2$, then it is enough that $2 \ln w \leq cw$ holds, which happens if $w = C$ itself is a large enough constant. In both cases, we have that $w \leq C \log t$ and, hence, $m \leq Cn \log t$ for a large enough constant C satisfies the inequality Eq. (6). □

4.4 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 7). 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 [47, 50] implies that 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 3 implies that the function 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 7) 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. □

5 DERANDOMIZATION OF APPROXIMATING CIRCUITS

In Theorem 1, the probabilistic circuit is required to compute *exact* values $f(x)$ of a given function f (with probability at least $2/3$). We will now prove a much more general result (Theorem 5) showing that even probabilistic *approximating* circuits can be efficiently derandomized.

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 $S = \{(x,y) \in \mathbb{R}^2\colon x \varrho y\}$.

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 must output a value which is close to the correct value $f(x)$ with probability at least $2/3$.

Example 3. 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 terms of 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 $\varphi : \mathbb{R}^m \to \mathbb{R}$ with the following property for any real numbers $a, x_1, \ldots, x_m$:

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

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 φ 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, there may be more than one majority ϱ-vote function.

The following theorem derandomizes approximating semialgebraic probabilistic circuits.

Theorem 5. *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 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, in the case of the equality relation ϱ, Theorem 1 gives an upper bound $m = O(n^2 s \log K)$ with $K = sb$. In particular, the description complexity t_f of the function f itself plays no role then.

Proof. Suppose that a probabilistic b-semialgebraic circuit $F(x, \boldsymbol{r})$ of size s with k random input variables ϱ-approximates a function $f : \mathbb{R}^n \to \mathbb{R}$. 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 of every its column is at most $\tau = K^{cns}$ for an absolute constant c.*

Proof. The probabilistic circuit $F(x, \boldsymbol{r})$ computes some function $F : \mathbb{R}^n \times \mathbb{R}^k \to \mathbb{R}$. Since the circuit is b-semialgebraic, Lemma 1 gives us an existential algebraic formula $\Psi_F(x, y, r)$ with $n+k+1$ free variables, at most $s-1$ quantifiers which uses at most sb polynomials of degree at most b, and defines the graph of the function F. That is, $\Psi_F(x, y, r) = 1$ if and only if $y = F(x, r)$.

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

$$\Psi(x, r) = \exists y_1\, \exists y_2\, \overbrace{\Psi_F(x, y_1, r)}^{sb} \wedge \overbrace{\Phi_f(x, y_2)}^{t_f} \wedge \overbrace{\Phi_\varrho(y_1, y_2)}^{t_\varrho}.$$

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, the formula $\Psi(x, r)$ has an equivalent quantifier-free algebraic formula. This show 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 uses $\ell \le sb + t_f + t_\varrho$ polynomials of degree is at most $d \le \max\{b, t_f, t_\varrho\}$. The formula has n free variables (x-variables). The formulas Φ_f and Φ_ϱ have no quantifiers, and Ψ_F has at most $s-1$ existential quantifiers. So, the entire existential formula Ψ has only $q \le s+1$ quantifiers. Lemma 2 gives us an equivalent quantifier-free algebraic formula using $\tau \le (\ell d)^{O(nq)} \le (\ell d)^{O(ns)} \le (sb + t_f + t_\varrho)^{O(ns)} = K^{O(ns)}$ polynomials of degree at most τ, and defining the entries of the rth column of the matrix M. Thus, the description complexity of each column of M is at most τ, as desired. □

Since the circuit $F(x, \boldsymbol{r})$ ϱ-approximates f, the correctness matrix M is probabilistically dense. So, together with Claim 5, the infinite majority rule (Lemma 7) implies that the matrix M has the m-majority property for $m = O(n \log \tau) = O(n^2 s \log K)$.

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, is $\varphi : \mathbb{R}^m \to \mathbb{R}$ is a majority ϱ-vote function, then

$$\varphi(F_1(x, r_1), \ldots, F_m(x, r_m))\, \varrho\, f(x)$$

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. □

Remark 7. Note that, unlike in the proof of Theorem 1 (corresponding to the equality relation ϱ), in the case of other relations ϱ, the correctness matrix M does not need to be 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 correctness matrix M will then be $M[x, r] = 1$. This is why, unlike in Theorem 1, now also the description complexity t_f of the approximated function f comes to play.

Remark 8. One could *directly* apply Lemma 2 to eliminate quantifiers from the entire formula $\Psi(x, r)$. This would yield a quantifier-free algebraic formula which defines *all* entries of the correctness matrix M. Note, however, that the formula $\Psi(x, r)$ has $n + k$ free variables, instead of only n in formulas $\Psi_r(x)$. So, the upper bound on the number of polynomials used in the quantifier formula would then be of the form $(\ell d)^{O(nq+kq)} \le (\ell d)^{O(ns+ks)}$. The number k of random variables may be as larger as the circuit size s. This would increase the size of the resulting deterministic circuit by an additional multiplicative factor of s. On the other hand, when dealing with individual columns of M *separately*, we make the resulting upper bound on the size of the derandomized circuit *independent* on the number k of random variables in the probabilistic circuit.

5.1 Majority vote for contiguous relations

One small issue still remains: just like in Theorem 1, the deterministic circuits given by Theorem 5 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 most situations, this can be easily done. In particular, we have the following simple fact.

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 3 are contiguous.

Claim 6. *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 $x_{i_1} \le \ldots \le x_{i_m}$. 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 show 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. □

Remark 9. The nullity relation ϱ (the third relation in Example 3) 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 $a \ne 0$. Still, a majority ϱ-vote function for this relation can also computed by a small monotone arithmetic $(+, \times)$ circuit (see Claim 8 in Appendix C).

5.2 Proof of Theorem 2

Let $x \,\varrho\, y$ be any contiguous relation of description complexity t_ϱ, and $f : \mathbb{R}^n \to \mathbb{R}$ any function of the description complexity t_f. Let $\mathcal{B}$ be a basis containing min, max and any other b-semialgebraic operations. Suppose that the function f can be ϱ-approximated by a probabilistic circuit of size s over $\mathcal{B}$. Our goal is to show that then f can be also ϱ-approximated by a deterministic circuit over $\mathcal{B}$ of size $O(ms + m \log m)$, where $m = n^2 s \log(sb + t_f + t_\varrho)$.

This is a direct consequence of Theorem 5 and Claim 6. Namely, by Theorem 5, f can be approximated as a majority ϱ-vote function of $m = O(n^2 s \log K)$ deterministic copies of this circuit, where $K = sb + t_f + t_\varrho$. Since our basis $\mathcal{B}$ contains both min and max operations, Claim 6 implies that the majority ϱ-vote function of m variables can be computed by a circuit over $\mathcal{B}$ of size $S = O(m \log m)$. Thus, the entire derandomized circuit has at most a constant times $m \cdot s + S$ gates. □

5.3 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 $f(x) = \min\{a_1 x_1 + \cdots a_n x_n : a \in A\}$ on all input weightings $x \in \mathbb{R}^n$.

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 $2/3$ holds.

The relation ϱ is this case is simple: $x \,\varrho\, y$ if and only if $|x - y| \le c$ (the fourth relation in Example 3). 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 1. *If a minimization problem $f(x) = \min\{a_1 x_1 + \cdots a_n x_n : a \in A\}$ can be approximated within some fixed factor c by a probabilistic circuit of size s over the basis $\mathcal{B}$, then f can be also approximated within the same factor c by a deterministic circuit over $\mathcal{B}$ of size at most a constant times $n^2 s^2 \log(s + |A|)$.*

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

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

using $|A|$ polynomials of degree 1. So, the description complexity of f is $t_f \leq |A|$. Since the approximation relation ϱ in our case has a constant description complexity $t_\varrho \leq 2$, and since the description complexity b of every gate is also constant, Theorem 5 implies that the minimization problem f can be approximated as a majority ϱ-vote function of $m = O(n^2 s \log K)$ 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, Claim 6 implies that a majority ϱ-vote function of m variables can be computed by a circuit over $\mathcal{B}$ of size $O(m \log m)$. 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 $n^2 s^2 \log(s + |A|)$, as desired. □

Remark 10. Note that the upper bound on the size S of the derandomized circuit, given by Theorem 5, 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 case, $\log |A|$ 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 $n^3 s^2 \log s$.

6 PROBABILISTIC TROPICAL CIRCUITS WITH ONE-SIDED ERROR

Many dynamic programming (DP) algorithms for discrete optimization problems are *pure* in that their recursion equations only use (min, +) or (max, +) as operations, and the recursion equations do not depend on input weights. Every such DP algorithm is just a special (recursively constructed) tropical circuit.

Notable examples of pure DP algorithms are the well-known Bellman–Ford–Moore DP algorithm for the shortest s-t path problem [7, 16, 37], the Floyd–Warshall DP algorithm for the all-pairs shortest paths problem [15, 53], the Held–Karp DP algorithm for the traveling salesman problem [23], the Dreyfus–Levin–Wagner DP algorithm for the weighted Steiner tree problem [8, 31].

Since the basis operations $\min(x, y)$, $\max(x, y)$ and $x + y$ of tropical circuits are b-semialgebraic for very small b (namely, for $b \leq 2$, see Table 1), 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.

However, this result has two drawbacks. The first (not a real) drawback is that our proof of Theorem 1 relies on three deep results: Pollack and Roy [6] (quantifier elimination, Lemma 2), Warren [52] (upper bound on sign patterns, Theorem 3), Vapnik and Chervonenkis [51] (uniform convergence in probability, Theorem 4). The second, more important, drawback is that the majority vote function Maj itself cannot be computed by a tropical (min, +) and (max, +) circuit at all (see Claim 7 in Appendix C). Recall that Maj *can* be computed is *both* min and max operations are allowed to be used as gates (Claim 6), but tropical circuits can only use one of these two operations.

It turns out that, under the *one-sided* error probability scenario, these two drawbacks can be completely eliminated. The resulting deterministic circuits are then also tropical circuits (do not use majority vote gates), and derandomization itself is then much simpler.

In order not to treat (min, +) and (max, +) circuits separately, we will consider circuits over semirings $(R, \oplus, \otimes)$ which are commutative and idempotent, that is, where $x \oplus y = y \oplus x$, $x \otimes y = y \otimes x$ and $x \oplus x = x$ hold. A *circuit* over the semiring is a circuit using the (fanin-2) semiring operations $\oplus$ and $\otimes$ as gates. Each input holds either one of the variables $x_1, \ldots, x_n$ or some "constant" $c \in R$,

a semiring element. What such a circuit computes is a polynomial function of the form

$$f_A(x) = \sum_{a\in A} c_a \prod_{i=1}^{n} x_i^{a_i}\,, \tag{7}$$

where $A \subset \mathbb{N}^n$ is a finite set of exponent vectors, and the coefficients $c_a \in R$ are semiring elements.

In order to treat the one-sided error scenario in general semirings, we use the *intrinsic* (or "better-than") ordering $\leq_R$ in semirings $(R, \oplus, \otimes)$ defined by $a \leq_R b$ iff $a \oplus c = b$ for some $c \in R$. For example, if R is the Boolean $(\vee, \wedge)$ or tropical (max, +) semiring, then $a \leq_R b$ iff $a \leq b$ (larger is better). In the tropical (min, +) semiring, we have $a \leq_R b$ iff $a \geq b$ (smaller is better).

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

$$\text{if } a_1 \leq_R b, \ldots, a_m \leq_R b \text{ and } b \in \{a_1, \ldots, a_m\}, \text{ then } a_1 \oplus \cdots \oplus a_m = b. \tag{8}$$

Under a *probabilistic circuit* of size s over a semiring R we will now understand an *arbitrary* random variable $\boldsymbol{F}$ taking its values in the set of all deterministic circuits over R 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 *computes* a given function $f : R^n \to R$ with *one-sided* error probability $0 \leq \epsilon \leq 1$ if $\Pr\{\boldsymbol{F}(x) \neq f(x)\} \leq \epsilon$ and $\Pr\{\boldsymbol{F}(x) \leq_R f(x)\} = 1$ hold for every input $x \in R^n$. 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 ϵ. In particular, $\epsilon = 0$ means that the circuit must correctly compute f, while $\epsilon = 1$ means that the circuit can do "almost everything," it only must never output better than optimal values.

6.1 Isolating sets and derandomization

Let $\mathcal{F}$ be some family of polynomials over a semiring R. A set $X \subseteq R^n$ is *isolates* a polynomial $f \in \mathcal{F}$ *within* $\mathcal{F}$ if for every polynomial $g \in \mathcal{F}$,

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

That is, if $g(x) \neq f(x)$ holds for some $x \in R^n$, then we also have $g(x) \neq f(x)$ for some $x \in X$. A set X is *isolating* for f if this holds for the family $\mathcal{F}$ of all polynomials over R.

Lemma 8. *Let R be an idempotent semiring, and f a polynomial over R. Suppose that f can be computed by a probabilistic circuit over R of size s with one-sided error probability $\epsilon < 1$. If f has a finite isolating set $X \subseteq R^n$, then f can be also computed by a deterministic circuit over R of size at most $\frac{s+1}{1-\epsilon} \cdot \log |X|$.*

Proof. Let $\boldsymbol{F}$ be a probabilistic circuit over R of size s computing f with a one-sided error $\epsilon < 1$. Set $p := 1 - \epsilon$, take $m = \lceil (1/p) \log |X| \rceil$ independent copies $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$ of the vector $\boldsymbol{r}$ of random input variables, and consider the probabilistic circuit $\boldsymbol{H}(x) = F(x, \boldsymbol{r}_1) \oplus \cdots \oplus F(x, \boldsymbol{r}_m)$ over the same semiring R.

Fix a vector $x \in X$. Since only one-sided error is allowed, we know that $F(x, \boldsymbol{r}_i) \leq_R f(x)$ must hold for all i. Hence, by property (8), $\boldsymbol{H}(x) \neq f(x)$ can only happen when *all* the values $F(x, \boldsymbol{r}_1), \ldots, F(x, \boldsymbol{r}_m)$ are strictly worse than the optimal value $f(x)$, and this can only happen with probability at most $\epsilon^m = (1-p)^m \leq \mathrm{e}^{-pm}$. 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 $|X|\epsilon^m \leq |X|\mathrm{e}^{-pm}$, which is smaller than 1, because $m \geq (1/p) \log |X|$ (and $\log \mathrm{e} > 1$).

There must therefore be a realization $H(x) = F(x, r_1) \oplus \cdots \oplus F(x, r_m)$ 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 $ms + m - 1 \leq [(s+1)/p] \log |X|$. The circuit H computes some polynomial function $h : R^n \to R$ over R. 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 $x \in R^n$, that is, the obtained deterministic circuit H correctly computes f on all possible inputs. □

6.2 Derandomizing tropical circuits

Lemma 8 reduces the derandomization of probabilistic tropical (max, +) and (min, +) circuits under the one-sided error scenario to proving the existence of small isolating sets $X \subset \mathbb{R}_+^n$ for tropical polynomials.

Existence of small isolating sets for polynomials over *fields* is long known: an easy consequence of the Schwartz–Zippel lemma [46, 54] is that *any* set X of the form $X = S^n$ for *any* set $S \subset \mathbb{R}$ of size $|S| \geq d+1$ isolates every n-variate real polynomial f of degree at most d within all polynomials of degree at most d.

Over tropical (max, +) or (min, +) semirings, that is, for tropical polynomials, we do not have such a strong isolation fact. Still, also then we can show that the (particular) set $X = \{0, 1, n+1\}^n$ is isolating for every monic and multilinear n-variate polynomial (Lemma 9 below). We will only require tropical circuits to work correctly on nonnegative input weights $x \in \mathbb{R}_+^n$; see Remark 12 below for the justification of only considering nonnegative weights.

In the tropical (max, +) semiring $(R, \oplus, \otimes)$, we have $R = \mathbb{R}_+$, $x \oplus y = \max(x, y)$ and $x \otimes y = x + y$. So, over this semiring, every polynomial of the form Eq. (7) turns into a tropical (max, +) polynomial (maximization problem) of the form $f(x) = \max_{a \in A} \langle a, x \rangle + c_a$, where $A \subset \mathbb{N}^n$ is some finite set of nonnegative integer vectors ("exponent" vectors), and $c_a \in \mathbb{R}_+$ are nonnegative "coefficients." Tropical (min, +) polynomials are of the same form with min instead of max.

Just as in the case of ordinary (arithmetic) polynomials, we call a tropical (max, +) polynomial $f(x) = \max_{a \in A} \langle a, x \rangle + c_a$ *monic* if $c_a = 0$ holds for all $a \in A$, and *multilinear* if $A \subseteq \{0, 1\}^n$. A *sub-polynomial* of f is any polynomial $h(x) = \max_{a \in A'} \langle a, x \rangle + c_a$ with $A' \subseteq A$. Two n-variate polynomials f and g are *equivalent* if $f(x) = g(x)$ holds for all input weightings $x \in \mathbb{R}_+^n$.

We will stick on optimization problems given by tropical polynomials which are both *monic* and *multilinear*. Most of the problems in combinatorial optimization are defined by such polynomials. Namely, we have some family $\mathcal{F}$ of feasible solutions (spanning trees, perfect matchings, etc), and the goal is, given an assignment x of nonnegative weights to the items (e.g., edges), to compute the minimum or maximum weight $x(S) = \sum_{i \in S} x_i$ of a feasible solution $S \in \mathcal{F}$.

Lemma 9. *Let $f(x_1, \ldots, x_n)$ be a tropical polynomial. If f is monic and multilinear, then the set $X = \{0, 1, n+1\}^n$ is isolating for f.*

Proof. That the set $X = \{0, 1, n+1\}^n$ isolates f within all *monic* tropical polynomials was (implicitly) proved in [27, Lemma 7] for (max, +) polynomials, and in [28, Appendix A] for (min, +) polynomials. So, it is enough to extend this result to the isolation within *all* (no necessarily monic) polynomials. The complete proof of Lemma 9 is given in Appendix D. □

Remark 11. The well known zero-one principle for comparator networks states (see, e.g. Knuth [30]): if a comparator network sorts every 0-1 sequence, then the network also sorts *any* sequence of real numbers. Lemma 9 is of similar type: if a tropical circuit solves a monic and multilinear optimization problem f on all inputs from $X = \{0, 1, n+1\}^n$, then the circuit also solves f on all inputs from $\mathbb{R}_+^n$. In fact, in the case of (max, +) circuits, this holds already for $X = \{0, 1\}^n$ (Lemma 10 in Appendix D).

Together with Lemma 9, Lemma 8 directly yields the following derandomization of probabilistic tropical circuits under the one-sided error scenario.

Theorem 6. *Let $f(x_1,\ldots,x_n)$ be a tropical* $(\max,+)$ *or* $(\min,+)$ *polynomial which is monic and multilinear. If f can be computed by a probabilistic tropical circuit of size s with a one-sided error $\epsilon<1$, then f can be also computed by a deterministic tropical circuit of size at most $2n(s+1)/(1-\epsilon)$.*

Note that even if the allowed error probability is a very close to 1 constant, say $\epsilon=0.999$, the size of the obtained deterministic circuits is still proportional to ns. This is in sharp contrast with the two-sided probability scenario, where we required that the error probability is $\epsilon\le 1/2-c$ for a constant $c>0$ (for definiteness, we have used $\epsilon=1/3$).

Remark 12 (Why only nonnegative weights?). When dealing with tropical $(\min,+)$ and $(\max,+)$ circuits, we have restricted us to circuits working correctly only over the domain $\mathbb{R}_+$ of *nonnegative* real numbers (nonnegative input weights). The reason for doing this is that only then tropical circuits can show their power, only then they can be more powerful than monotone arithmetic $(+,\times)$ circuits. Namely, Jerrum and Snir [25, Theorem 2.5] proved that if tropical circuits must correctly work over the entire domain $\mathbb{R}$, then the tropical circuit complexity of tropical polynomials is essentially the same as the monotone arithmetic $(+,\times)$ circuit complexity of the arithmetical versions of these polynomials.

In contrast, if tropical circuits only need to correctly work over $\mathbb{R}_+$, then their size can be even exponentially smaller than that of arithmetic $(+,\times)$ circuits. For example, then the shortest s-t path problem on n-vertex graphs can be solved by a $(\min,+)$ circuit of size only $O(n^3)$ via simulating the Bellman–Ford DP algorithm. But, as shown by Jerrum and Snir [25], the corresponding to this problem arithmetical polynomial requires arithmetic $(+,\times)$ circuits of size $2^{\Omega(n)}$.

7 A LOWER BOUND FOR PROBABILISTIC TROPICAL CIRCUITS

After the derandomization of a probabilistic tropical $(\min,+)$ or $(\max,+)$ circuits via Theorem 1, we are forced to use a majority vote function as an output gate. But majority vote functions cannot be computed by tropical circuits at all (see Claim 7 in Appendix C). So, the resulting deterministic circuits are *not* tropical circuits, and known lower bounds for deterministic tropical circuits (see, for example, [25, 28]) do *not* imply lower bounds for probabilistic tropical circuits.

Still, it is possible to derive lower bounds for probabilistic tropical circuits from lower bounds on the size of deterministic monotone *Boolean* circuits solving the "decision versions" of the corresponding optimization problems.

In order not to treat $(\min,+)$ and $(\max,+)$ circuits separately, we will (again) consider circuits over arbitrary commutative semirings $(R,\oplus,\otimes,\mathbb{0},\mathbb{1})$. But unlike in the previous section, now we insist that the semiring must have additive and multiplicative unity elements $\mathbb{0}$ and $\mathbb{1}$ satisfying $\mathbb{0}\oplus x=x$, $\mathbb{0}\otimes x=\mathbb{0}$ and $\mathbb{1}\otimes x=x$. As in the previous section, a *circuit* over a semiring $(R,\oplus,\otimes,\mathbb{0},\mathbb{1})$ is a circuit using the (fanin-2) semiring operations $\oplus$ and $\otimes$ as gates. Each input holds either one of the variables $x_1,\ldots,x_n$ or some "constant" $c\in R$, a semiring element.

A circuit is *constant-free* if it has no constant inputs $c\notin\{\mathbb{0},\mathbb{1}\}$, that is, when $\mathbb{0}$ and $\mathbb{1}$ are the only allowed constant inputs. Every constant-free circuit over R computes some polynomial

$$f(x)=\sum_{a\in A}\prod_{i=1}^{n}x_i^{a_i}\,, \tag{9}$$

where $A\subset\mathbb{N}^n$ is a finite multi-set of vectors (exponent vectors); A being a multi-set means that one and the same monomial may appear several times in the polynomial.

As in Section 6, under a *probabilistic circuit* of size s over a semiring R we will understand an *arbitrary* random variable $\boldsymbol{F}$ taking its values in the set of all deterministic circuits over R of size at most s. Such a circuit *computes* a given function $f:R^n\to R$ if $\Pr\{\boldsymbol{F}(x)=f(x)\}\ge 2/3$ holds for every input $x\in R^n$. That is, unlike in Section 6, we now allow two-sided error.

The *decision version* of a polynomial of the form (9) is the monotone Boolean function

$$\hat{f}(x) = \bigvee_{a \in A} \bigwedge_{i\colon a_i \neq 0} x_i \,.$$

A semiring $(R, \oplus, \otimes, \mathbb{0}, \mathbb{1})$ is of *zero characteristic*, if $\mathbb{1} \oplus \mathbb{1} \oplus \cdots \oplus \mathbb{1} \neq \mathbb{0}$ holds for any finite sum of the multiplicative unity $\mathbb{1}$. Note that polynomials (9) over such semirings have the following property: on every input $x \in \{\mathbb{0}, \mathbb{1}\}^n$, we have $f(x) \neq \mathbb{0}$ if and only if there is a vector $a \in A$ such that x has the multiplicative unity element $\mathbb{1}$ in all positions i where $a_i \neq 0$. This happens precisely when the decision version $\hat{f}$ accepts the corresponding Boolean version of x (with element $\mathbb{0}$ replaced by 0, and element $\mathbb{1}$ replaced by 1). That is, the decision version $\hat{f}$ of a polynomial f captures the behavior of f when restricted to only inputs in $\{\mathbb{0}, \mathbb{1}\}^n$. This observation gives an idea behind the following lower bound.

For a polynomial f over some semiring, let $\mathrm{B}(f)$ denote the minimum size of a monotone Boolean $\{\vee, \wedge\}$ circuit computing the decision version $\hat{f}$ of f.

Theorem 7. *Let f be a monic n-variate polynomial over a semiring R of zero characteristic. The minimum size of every probabilistic constant-free circuit computing f over R is at least a constant times* $\mathrm{B}(f)/n - O(\log n)$.

Proof. Take a probabilistic constant-free circuit $\boldsymbol{F}(x)$ over R of size s computing the polynomial $f : R^n \to R$. We will concentrate us on inputs from the set $X = \{\mathbb{0}, \mathbb{1}\}^n \subseteq R^n$. Since the circuit $\boldsymbol{F}(x)$ correctly computes $f(x)$ on all inputs $x \in X$, the finite majority rule (Lemma 5) gives us a sequence $F_1, \ldots, F_m$ of $m = O(\log |X|) = O(n)$ deterministic circuits of size at most s over the semiring R such that also the (deterministic) circuit $F(x) = \mathrm{Maj}(F_1(x), \ldots, F_m(x))$ correctly computes the polynomial $f(x)$ on all inputs $x \in X$. We are now going to turn the circuit F into a monotone Boolean circuit computing the decision version $\hat{f}$ of f.

Consider the set $S = \{\overline{n}\colon n \in \mathbb{N}\} \subseteq R$ of semiring elements, where $\overline{n} := \mathbb{0}$ (the additive unity) for $n = 0$, and $\overline{n} := \mathbb{1} \oplus \cdots \oplus \mathbb{1}$ is the n-fold sum of the multiplicative unity $\mathbb{1}$ for all $n \geq 1$. Since $\overline{n} \oplus \overline{m} = \overline{n+m}$ and $\overline{n} \otimes \overline{m} = \overline{n \cdot m}$, the set S is closed under both semiring operations. So, $(S, \oplus, \otimes, \mathbb{0}, \mathbb{1})$ is a subsemiring of R. Since the circuit F is constant-free (can only use $\mathbb{0}$ and $\mathbb{1}$ as constant inputs), this implies that, when we restrict the circuit to take inputs only from $X = \{\mathbb{0}, \mathbb{1}\}^n$, all intermediate results computed at the gates of $F(x)$ all belong to S.

Let H be a Boolean circuit obtained from F as follows: replace each $\oplus$ gate by an OR gate, and each $\otimes$ gate by an AND gate; the majority vote gate remains as it is. Consider the mapping $h : S \to \{0, 1\}$ given by $h(\mathbb{0}) := 0$ and $h(\overline{n}) := 1$ for all $n \geq 1$. Since R has zero-characteristic, $\overline{n} = \mathbb{0}$ holds if and only if $n = 0$. Thus, the mapping h is a homomorphism from the semiring $(S, \oplus, \otimes, \mathbb{0}, \mathbb{1})$ to the Boolean semiring $(\{0, 1\}, \vee, \wedge, 0, 1)$: $h(x \oplus y) = h(x) \vee h(y)$ and $h(x \otimes y) = h(x) \wedge h(y)$ holds for all $x, y \in S$. So, since the circuit F computes the polynomial f correctly on all inputs in $\{\mathbb{0}, \mathbb{1}\}^n$, the boolean circuit H correctly computes the decision version $\hat{f}$ of f on all (Boolean) inputs in $\{0, 1\}^n$.

In the Boolean version H of the circuit F, the output Maj gate only receives Boolean 0-1 inputs and, hence, is the Boolean majority function (which outputs 1 exactly when the input 0-1 string has more than half 1s). The sorting network of Ajtai, Komlós and Szemerédi [3] gives a monotone Boolean circuit of size $O(m \log m)$ computing all threshold functions of m variables and, hence, also the majority function. So, since in our case $m = O(n)$, we obtain a monotone Boolean circuit of size at most $t = ms + O(m \log m) = O(ns + n \log n)$ computing the decision version $\hat{f}$ of our polynomial f. Since $t \geq \mathrm{B}(f)$, the desired lower bound on the size s of the probabilistic circuit follows. □

Strong, even super-polynomial lower bounds on the size of monotone Boolean circuits are long known; see, for example, [5, 26] and the references therein. Together with Theorem 7, these bounds immediately yield lower bounds on the size of probabilistic tropical circuits solving the corresponding optimization problems. We restrict ourselves with just one application.

The identity elements in the $(\min, +)$ semiring are $\mathbb{0} = +\infty$ and $\mathbb{1} = 0$, whereas in the $(\max, +)$ semiring they are $\mathbb{0} = -\infty$ and $\mathbb{1} = 0$. These semirings are clearly of zero characteristic: here $x \oplus y$ is either $\min(x, y)$ or $\max(x, y)$. Functions computed by tropical polynomials are optimization problems. Consider, for example, the permanent polynomial

$$\mathrm{Per}_n(x) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i,\sigma(i)}\,,$$

where S_n is the set of all permutations of $1, 2, \ldots, n$. In the tropical $(\min, +)$ semiring, this polynomial turns into the well-known *assignment problem*: given a bipartite $n \times n$ graph with nonnegative weights on its edges, compute the minimum weight of a perfect matching in this graph (non-edges hold weight $\mathbb{0} = +\infty$).

Corollary 2. *Every probabilistic* $(\min, +)$ *circuit solving the assignment problem must have* $n^{\Omega(\log n)}$ *gates.*

Proof. Razborov [42] has proved that, over the Boolean semiring, deterministic circuits computing Per_n require super-polynomial size $n^{\Omega(\log n)}$. Together with Theorem 7, this implies that probabilistic $(\min, +)$ circuits solving the assignment problem also require this number of gates. □

ACKNOWLEDGMENTS

I am thankful to Sergey Gashkov, Joshua Grochow, Pascal Koiran, Igor Sergeev and Hans Ulrich Simon for inspiring discussions at the initial stage 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] Alon and Scheinerman. 1988. Degrees of freedom versus dimension for containment orders,. *Order* 5 (1988), 11–16.
[5] N. Alon and R. Boppana. 1987. The monotone circuit complexity of boolean functions. *Combinatorica* 7, 1 (1987), 1–22.
[6] 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.
[7] R. Bellman. 1958. On a routing problem. *Quarterly of Appl. Math.* 16 (1958), 87–90.
[8] R.E. Bellman and S.E. Dreyfus. 1962. *Applied Dynamic Programming*. Princeton University Press.
[9] C.H. Bennett and J. Gill. 1981. Relative to a random oracle A, $P^A \neq NP^A \neq$ co-NP^A with probability 1. *SIAM J. Comput.* 10, 1 (1981), 96–113.
[10] P. Bürgisser, M. Karpinski, and T. Lickteig. 1993. On randomized semi-algebraic test complexity. *J. Complexity* 9, 2 (1993), 231–251.
[11] 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.
[12] D. Dubhashi and A. Panconesi. 2009. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press.
[13] R.M. Dudley. 1984. *A course on empirical processes*. Lecture Notes in Mathematics, Vol. 1097. Springer.
[14] K. Fan. 1956. On systems of linear inequalities. In *Linear Inequalities and Relatex Systems*, H.W. Kuhn and A.W. Tucker (Eds.). Vol. 38. Princeton University Press, 99–156.

[15] R.W. Floyd. 1962. Algorithm 97, shortest path. *Comm. ACM* 5 (1962), 345.

[16] L.R. Ford. 1956. *Network flow theory.* Technical Report P-923. The Rand Corp.

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

[18] 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.

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

[20] 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.

[21] 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.

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

[23] M. Held and R.M. Karp. 1962. A dynamic programming approach to sequencing problems. *SIAM J. on Appl. Math.* 10 (1962), 196–210.

[24] 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.

[25] M. Jerrum and M. Snir. 1982. Some exact complexity results for straight-line computations over semirings. *J. ACM* 29, 3 (1982), 874–897.

[26] S. Jukna. 2012. *Boolean Function Complexity: Advances and Frontiers.* Springer-Verlag.

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

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

[29] S. Jukna. 2019. Coin flipping cannot shorten arithmetic computations. *Amer. Math. Monthly* 126, 4 (2019), 364–366.

[30] D.E. Knuth. 1998. *The Art of Computer Programming.* Vol. 3. Addison–Wesley.

[31] A.Y. Levin. 1971. Algorithm for the shortest connection of a group of graph vertices. *Sov. Math. Dokl.* 12 (1971), 1477–1481.

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

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

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

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

[36] M. Mitzenmacher and E. Upfal. 2005. *Probability and computing: randomized algorithms and probabilistic analysis.* Cambridge University Press.

[37] E.F. Moore. 1959. The shortest path through a maze. In *Proc. of Int.. Symp. on Switching Theory*, Vol. II. 285–292.

[38] 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.

[39] R. Motwani and P. Raghavan. 1995. *Randomized Algorithms.* Cambridge University Press.

[40] D. Pollard. 1984. *Convergence of Stochastic Processes.* Springer-Verlag.

[41] P. Pudlák and V. Rödl. 1992. A combinatorial approach to complexity. *Combinatorica* 12, 2 (1992), 221–226.

[42] A.A. Razborov. 1985. Lower bounds on monotone complexity of the logical permanent. *Math. Notes of the Acad. of Sci. of the USSR* 37, 6 (1985), 485–493.

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

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

[45] N. Saxena. 2009. Progress on polynomial identity testing. *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS* 99 (2009), 49–79.

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

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

[48] 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.

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

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

[51] 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.

[52] H.E. Warren. 1968. Lower bounds for approximation by non-linear manifolds. *Trans. Amer. Math. Soc.* 133 (1968), 167–178.
[53] S. Warshall. 1962. A theorem on boolean matrices. *J. ACM* 9 (1962), 11–12.
[54] R. Zippel. 1979. Probabilistic algorithms for sparse polynomials. In *Lect. Notes in Comput. Sci.*, Vol. 72. Springer, 216–226.

A PROOF OF LEMMA 1: FROM CIRCUITS TO QUANTIFIED FORMULAS

Let $\mathcal{B}$ be a basis consisting of b-semialgebraic functions. Let F be a circuit of size s over $\mathcal{B}$ computing some function $f : \mathbb{R}^n \to \mathbb{R}$. Our goal is to show that then the graph $\{(x, y)\colon f(x) = y\}$ of f can be defined by an existential algebraic formula of size at most sb, degree at most b and with $s-1$ (existential) quantifiers.

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 $\Phi_i(x, y)$ of size and 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 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 ($i \geq 2$) is obtained by applying the i-th 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 has size and degree at most b, the formula Ψ has size at most sb, degree at most b, and contains only $s-1$ quantifiers. □

B MEASURABILITY

In order to obtain the uniform convergence result of Vapnik and Chervonenkis given in Theorem 4 (as well as its subsequent extensions for not necessarily 0-1 valued functions), certain measurability assumptions have to be made concerning the class of functions H when this class is *uncountable*.

Haussler in [22, 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 $H = \{f(t, \cdot)\colon t \in T\}$, 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 $n \geq 1$, 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 [13, Chapter 10] and Pollard [40, Appendix C] discuss more general sufficient conditions for classes of not necessarily 0-1 valued functions $h : X \to \mathbb{R}$ to be permissible.

C 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 7. *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 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.

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

Proof. For nonnegative integers $m \leq n$ and $0 \leq k \leq m - 1$, consider the following polynomials

$$f_{m,k}(x_1, \ldots, x_n) = \sum_{\substack{S \subseteq [m] \\ |S| = m-k+1}} \prod_{i \in S} x_i^2 .$$

Note that $f_{m,k}(x_1,\ldots,x_n) = 0$ precisely when at least k numbers among the first m numbers $x_1,\ldots,x_m$ are zeros. (We have taken squares just to avoid possible cancellations.) Hence, $f = f_{n,k}$ for $k = \lfloor n/2 \rfloor + 1$ computes the zero vote function of n variables. The polynomials $f_{m,k}$ can be computed by first taking $f_{m,1}(x_1,\ldots,x_n) = x_1^2 \cdots x_m^2$, $f_{m,m}(x_1,\ldots,x_n) = x_1^2 + \cdots + x_m^2$ for $m = 1,\ldots,n$, and $f_{m,k}(x_1,\ldots,x_n) = 1$ $(\neq 0)$ for $k > m$, and then using the recursion $f_{m,k} = f_{m-1,k} + f_{m-1,k-1} \cdot x_m^2$. The resulting arithmetic $(+,\times)$ circuit uses only $O(n^2)$ gates. We take squares only to avoid possible cancellations of terms. □

D PROOF OF LEMMA 9: ISOLATING SETS FOR TROPICAL POLYNOMIALS

Recall that tropical (max, +) polynomials are of the form $f(x) = \max_{a\in A}\langle a, x\rangle + c_a$, where $A \subset \mathbb{N}^n$ is some finite set of nonnegative integer vectors ("exponent" vectors), and $c_a \in \mathbb{R}_+$ are nonnegative "coefficients." Such a polynomial is *monic* if $c_a = 0$ holds for all $a \in A$, and is *multilinear* if $A \subseteq \{0,1\}^n$. Tropical (min, +) polynomials are of the form $f(x) = \min_{a\in A}\langle a, x\rangle + c_a$, with min instead of max.

In the following proofs we will use the following notation for two sets $A, B \subseteq \mathbb{R}^n$ of vectors.

- *B lies below A*, denoted $B \leq A$, if $(\forall b \in B)\ (\exists a \in A)\ \ b \leq a$.
- *B lies above A*, denoted $B \geq A$. if $(\forall b \in B)\ (\exists a \in A)\ \ b \geq a$.

Here, as customary, $b \leq a$ means $b_i \leq a_i$ for all positions i. Note that $B \leq A$ does not imply $A \geq B$, and vice versa. For a vector x, let $S_x = \{i\colon x_i \neq 0\}$ denote its support.

Lemma 10 (Maximization). *Let $f(x_1,\ldots,x_n)$ be a monic and multilinear* (max, +) *polynomial. Then the set $X = \{0,1\}^n$ isolates f within all* (max, +) *polynomials.*

Proof. The polynomial f is of the form $f(x) = \max_{a\in A}\ \langle a, x\rangle + c_a$, where $A \subseteq \{0,1\}^n$ (f is multilinear), and $c_a = 0$ for all $a \in A$ (f is monic). Take an arbitrary (max, +) polynomial $g(x) = \max_{b\in B}\ \langle b, x\rangle + c_b$ with $B \subset \mathbb{N}^n$ and $c_b \in \mathbb{R}_+$. Suppose that $g(x) = f(x)$ holds for all input weightings $x \in \{0,1\}^n$. Our goal is to show that then $g(x) = f(x)$ must also hold for all input weightings $x \in \mathbb{R}_+^n$.

Since the coefficients c_b must be nonnegative, and since the polynomial g takes the maximum of the values $\langle b, x\rangle + c_b$, the equality $g(\vec{0}) = f(\vec{0}) = 0$ implies $c_b = 0$ for all $b \in B$. So, the polynomial g is monic. Furthermore, since $g(x) = f(x)$ must hold for each of n input weightings $x \in \{0,1\}^n$ with exactly one 1, all vectors in B must also be 0-1 vectors, that is, $B \subseteq \{0,1\}^n$ holds

We claim that $B \leq A$. Suppose contrariwise that there is a vector $b \in B$ such that $b \not\leq a$ for all vectors $a \in A$. Since b is a 0-1 vector, we have $S_b \setminus S_a \neq \emptyset$ for all $a \in A$. But then on the 0-1 input $x = b \in \{0,1\}^n$, we have $g(x) \geq \langle b, x\rangle = \langle b, b\rangle = |S_b|$, whereas $f(x) \leq |S_b| - 1$, a contradiction with $g(x) = f(x)$.

We claim that $A \subseteq B$. Suppose contrariwise that there is a vector $a \in A$ such that $b \neq a$ for all vectors $b \in B$. Since all vectors are 0-1 vectors, this means that $|S_b \cap S_a| \leq |S_a| - 1$ holds for every vector $b \in B$. But then on input $x = a \in \{0,1\}^n$, we have $\langle b, x\rangle = \langle b, a\rangle \leq |S_a| - 1$ for all $b \in B$ and, hence, also $g(x) \leq |S_a| - 1 < |S_a| = \langle a, x\rangle \leq f(x)$, a contradiction with $g(x) = f(x)$.

Now, for every input weighting $x \in \mathbb{R}_+^n$, $B \leq A$ yields $g(x) \leq f(x)$, while the inclusion $A \subseteq B$ yields $f(x) \leq g(x)$. Thus, $g(x) = f(x)$ for all $x \in \mathbb{R}_+^n$, as desired. □

The case of (min, +) polynomial is more involved. The difficulty is enforced by our restriction: we only consider *nonnegative* weights and, consequently, cannot use the equality $\min(x, y) = -\max(-x, -y)$ to reduce (min, +) polynomials to (max, +) polynomials; see Remark 12 for a justification of this restriction (to nonnegative weights).

A *sub-polynomial* of a (min, +) polynomial $f(x) = \max_{b\in B}\langle b, x\rangle + c_b$ is any polynomial $h(x) = \min_{b\in B'}\langle b, x\rangle + c_b$ with $B' \subseteq B$. Two n-variate (min, +) polynomials f and g are *equivalent* if $f(x) = g(x)$ holds for all input weightings $x \in \mathbb{R}_+^n$.

If two (max, +) polynomials f and g are equivalent, and if f is monic, then g must also be monic just because, as observed in the proof of Lemma 10, $g(\vec{0}) = f(\vec{0}) = 0$ implies that g cannot have any nonzero "coefficients." In the case of minimization, such a trivial argument does not work. Still, also then the following lemma shows that the polynomial g must be "essentially" monic.

Lemma 11. *Let f and g be equivalent* (min, +) *polynomials. If f is monic, then f is equivalent to some monic sub-polynomial of g.*

Proof. We will use the following corollary from Farkas' lemma shown by Jerrum and Snir [25]. For a vector $x \in \mathbb{R}^n$, let $\tilde{x}$ denote the extended vector $(x, 1)$ in $\mathbb{R}^{n+1}$. Jerrum and Snir [25, Corollary A3] have shown that for any two finite sets $U, V \subset \mathbb{R}^{n+1}$ of vectors, the following two assertions are equivalent:

(1) $\min_{v \in V} \langle v, \tilde{x} \rangle \geq \min_{u \in U} \langle u, \tilde{x} \rangle$ holds for all $x \in \mathbb{R}^n_+$;
(2) V lies above the convex hull of U.

The direction (2) $\Rightarrow$ (1) is obvious, and the (1) $\Rightarrow$ (2) direction is derived in [25] from a version of Farkas' lemma proved by Fan [14, Theorem 4].

Now let $f_A(x) = \min_{a \in A} \langle a, x \rangle + c_a$ and $f_B(x) = \min_{b \in B} \langle b, x \rangle + c_b$ be two equivalent polynomials over the (min, +) semiring. Suppose that f_A is monic, that is, $c_a = 0$ holds for all $a \in A$. Consider the sets $V = \{(a, 0) \colon a \in A\}$ and $U = \{(b, c_b) \colon b \in B\}$ of vectors in $\mathbb{N}^n \times \mathbb{R}_+ \subseteq \mathbb{R}^{n+1}_+$. Since the polynomials f_A and f_B are equivalent, $\min_{v \in V} \langle v, \tilde{x} \rangle = \min_{u \in U} \langle u, \tilde{x} \rangle$ holds for all $x \in \mathbb{R}^n_+$. By the implication (1) $\Rightarrow$ (2), the set V must lie above the convex hull of U. But since all vectors of V have zeros in the last position, only vectors of U with zeros in the last position can participate in the corresponding convex combinations. Thus, the set A must lie even above the convex hull of the subset $B' = \{b \in B \colon c_b = 0\}$ of B. (This subset is nonempty because $f_B(\vec{0}) = f_A(\vec{0}) = 0$). By the implication (2) $\Rightarrow$ (1), $f_A(x) \geq f_{B'}(x)$ must hold for all $x \in \mathbb{R}^n_+$. Since B' is a subset of B, we also have $f_A(x) = f_B(x) \leq f_{B'}(x)$ for all $x \in \mathbb{R}^n_+$. So, $f_{B'}$ is a desired monic sub-polynomial of f_B equivalent to f_A. □

Lemma 12 (Minimization). *Let $f(x_1, \ldots, x_n)$ be a monic and multilinear* (min, +) *polynomial. Then the set $X = \{0, 1, n+1\}^n$ isolates f within all* (min, +) *polynomials.*

Proof. The polynomial f is of the form $f(x) = \min_{a \in A} \langle a, x \rangle + c_a$, where $A \subseteq \{0, 1\}^n$ (f is multilinear), and $c_a = 0$ for all $a \in A$ (f is monic). Take an arbitrary (min, +) polynomial $g'(x) = \max_{b \in B'} \langle b, x \rangle + c_b$ with $B \subset \mathbb{N}^n$ and $c_b \in \mathbb{R}_+$. Suppose that $g'(x) = f(x)$ holds for all input weightings $x \in \{0, 1, n+1\}^n$. Our goal is to show that then $g'(x) = f(x)$ must also hold for all input weightings $x \in \mathbb{R}^n_+$.

Since the polynomial f is monic, Lemma 11 gives us a *monic* sub-polynomial $g(x) = \max_{b \in B} \langle b, x \rangle$ of g' with $B \subseteq B'$ such that $g(x) = f(x)$ holds for all $x \in \mathbb{R}^n_+$. In particular, $g(x) = f(x)$ holds for all $x \in \{0, 1, n+1\}^n$. We only have to show that then $g(x) = f(x)$ also holds for all $x \in \mathbb{R}^n_+$. Since we only consider nonnegative weights $x \in \mathbb{R}^n_+$, we can assume that A is an antichain, that is, no two its vectors are comparable under $\leq$: if A contains two vectors $a \neq a'$ with $a \leq a'$, then remove vector a'; the function computed remains the same.

We claim that $B \geq A$. Suppose contrariwise that there is a vector $b \in B$ such that $b \not\geq a$ for all vectors $a \in A$. Since vectors in A are 0-1 vectors, this means that $S_a \setminus S_b \neq \emptyset$ holds for all vectors $a \in A$. So, if we take an input $x \in \{0, 1\}^n$ with $x_i = 0$ for $i \in S_b$, and $x_i = 1$ for $i \notin S_b$, then $\langle a, x \rangle \geq 1$ holds for all $a \in A$ and, hence, also $f(x) \geq 1$. But, on this input x, we have $g(x) \leq \langle b, x \rangle = 0$, a contradiction.

We claim that $A \subseteq B$. Suppose contrariwise that there is a vector $a \in A$ such that $b \neq a$ for all vectors $b \in B$. Then, for every vector $b \in B$, there are only three possibilities: (1) $S_b \subset S_a$ (a proper inclusion), (2) $S_b = S_a$ but $b_i \geq a_i + 1$ for some position $i \in S_a$, and (3) $S_b \not\subseteq S_a$, that is, $S_b \setminus S_a \neq \emptyset$.

Since A is an antichain, and since $a' \leq b$ must hold for some vector $a' \in A$, the case (1), a proper inclusion $S_b \subset S_a$, is impossible, for otherwise we would have a proper inclusion $S_{a'} \subset S_a$. So, if we take the input $x \in \{1, n+1\}^n$ with $x_i = 1$ for $i \in S_a$, and $x_i = n+1$ for $i \notin S_a$, then we have $\langle b, x\rangle \geq \langle a, a\rangle + 1 > |S_a|$ in the case (2), and $\langle b, x\rangle \geq n+1 > |S_a|$ in the case (3). Thus, on this weighting x, the value $g(x)$ is strictly larger than $|S_a|$, whereas the value $f(x)$ is at most $\langle a, x\rangle = |S_a|$, a contradiction with $g(x) = f(x)$. □

Received 20.08.2018; revised 0; accepted 0