# Complexity lower bounds for approximation algebraic computation trees

Felipe Cucker*†
Department of Mathematics
City University of Hong Kong
83 Tat Chee Avenue, Kowloon
HONG KONG
e-mail: macucker@math.cityu.edu.hk

Dima Grigoriev*
IMR
Université de Rennes I
Campus de Beaulieu, Rennes 35042
FRANCE
e-mail: dima@maths.univ-rennes1.fr

**Abstract**

We prove lower bounds for approximate computations of piecewise polynomial functions which, in particular, apply for round-off computations of such functions.

The goal of this paper is to prove lower bounds for approximated computations. As it is customary for lower bounds, we consider some form of algebraic tree as our computational model (cf. [Bürgisser, Clausen, and Shokrollahi 1996] or [Blum, Cucker, Shub, and Smale 1998] for algebraic trees). But, unlike the usual proofs of lower bounds, which deal with decision problems, we will consider computations of real functions. That is, we consider trees computing functions $f : \mathbb{R}^n \to \mathbb{R}$ and, also unlike the usual results on lower bounds, we will allow for approximate computations. To understand the nature of our results let us look first at an example.

**Example 1** Given a strictly convex compact polygon $P \subset \mathbb{R}^2$ consider the function $f : \mathbb{R}^2 \to \mathbb{R}$ defined by

$$f(c) = \max_{x \in P} \langle c, x \rangle^2.$$

*This work was done while both authors were at MSRI. We would like to thank the institute for its support.

†Partially supported by CERG grants.

Obviously, there is a partition of $\mathbb{R}^2$ into a finite number of regions $V_i$ and for each such region there is a vertex $v_i$ of $P$ such that $f(c) = \langle c, v_i \rangle^2$ for all $c \in V_i$.

Let $T$ be an algebraic computation tree computing $f$ of Example 1. Then the number of leaves of $T$ is at least the number of 2-dimensional regions $V_i$ with pairwise different $v_i$. This follows from the fact that two different polynomials in $\mathbb{R}[x, y]$ can not coincide, as functions, on an open subset of $\mathbb{R}^2$. Therefore, since computation trees are binary, we have that the depth of $T$ is at least the $\log_2$ of this number. This argument is independent of the fact that the input space is $\mathbb{R}^2$ (any $\mathbb{R}^n$ could be considered instead; just replace polygon by polyhedra and $\mathbb{R}[x, y]$ by $\mathbb{R}[x_1, \ldots, x_n]$). We intend to replicate it for approximate computations.

Now consider a tree $T$ which computes a $\delta$-approximation of $f$ in the sense that the output $T(c)$ satisfies $|f(c) - T(c)| \leq \delta$ for all $c \in \mathbb{R}^2$.

If $\delta \neq 0$ a lower bound like the one above is no longer valid. To see why, consider a regular $n$-sided polygon inscribed in the unit circunference centered at the origin. For large $n$ the polygon becomes "close" to the circumference and for $n$ large enough $f(c)$ is $\delta$-approximated by $\|c\|^2 = c_1^2 + c_2^2$. And this function can be computed with only three operations. So the $\log_2 n$ bound above is far to apply.

Thus, in order to obtain meaningfull lower bounds one needs to impose some condition on the value of $\delta$. We devote the next section to define the main concepts of the paper and to state our main theorem, where this condition is made explicit. In Section 3 we extend our main result to round-off trees i.e., trees whose arithmetic operations are subject to some form of error. Finally, in Section 4, we briefly discuss extensions to other settings such as randomized or parallel trees.

# 1 Piecewise Polynomial Functions and Round-off Computation Trees

In this paper we will only deal with trees whose computation nodes perform additions, subtractions or multiplications.[1] It is immediate to prove that such a tree (with exact arithmetic) computes a very specific kind of functions, which we describe in the next definition.

**Definition 1** A function $f : \mathbb{R}^n \to \mathbb{R}$ is called *piecewise polynomial* if there exists a finite partition $\mathbb{R}^n = \cup_i V_i$ of $\mathbb{R}^n$ into semi-algebraic sets $V_i$ and for each $i$ a polynomial $f_i \in \mathbb{R}[x_1, \ldots, x_n]$ such that $f_{|V_i} = f_i$.

Without loss of generality we will assume that if $i \neq j$ then $f_i \neq f_j$.

The function $f$ of Example 1 is piecewise polynomial. Another example of this kind of function is provided by quantifier elimination in the theory of the reals. Such a procedure defines a piecewise polynomial function by associating, to each

[1] The extension of our results to the case of trees allowing divisions is an open problem.

tuple of coefficients of an input formula, a vector of coefficients of an equivalent quantifier-free formula.

Apparently, computation of piecewise polynomial (or more generally, rational) functions was considered for the first time over the complex numbers rather than over the reals, as in our case, by Strassen [1983] for the problem of computing GCDs of univariate polynomials.

Before defining what we mean by approximation we emphasize that we are considering computation trees rather than decision trees. In particular we recall that, associated to any leaf $\eta$ of a computation tree $T$, there is a polynomial $g_\eta \in \mathbb{R}[x_1, \ldots, x_n]$ such that, for any input $x \in \mathbb{R}^n$ which reaches $\eta$ in the course of the computation, the output $T(x)$ of $T$ coincides with $g_\eta(x)$ (cf. [Blum, Cucker, Shub, and Smale 1998] for details).

**Definition 2** Let $T$ be an algebraic computation tree with input space $\mathbb{R}^n$ and output space $\mathbb{R}$, and let $f : \mathbb{R}^n \to \mathbb{R}$ be a function.

We say that $T$ *approximates* $f$ *with absolute accuracy* $\delta$ if for every input $x \in \mathbb{R}^n$ the output $T(x)$ of $T$ satisfies $|T(x) - f(x)| \leq \delta$.

We say that $T$ *approximates* $f$ *with relative accuracy* $\delta$ if for every input $x \in \mathbb{R}^n$ the output $T(x)$ of $T$ satisfies $|T(x) - f(x)| \leq \delta |f(x)|$.

**Remark 1**

**1)** Approximate algorithms for a problem are a current practice to improve the efficiency over the known algorithms computing the exact solutions of that problem.
**2)** To the best of our knowledge very little is know on lower bounds for approximate (or round-off) computations. A worth noting exception is a paper by Renegar [Renegar 1987] which gives lower bounds for approximating zeros of univariate polynomials.

We now describe the condition we will impose on $\delta$ in order to obtain lower bounds for the depth of approximate computations. This condition takes the form of a bound $\delta \leq \Gamma$ where $\Gamma$ is a quantity depending only on the piecewise function $f$ (rather than on the tree). We actually provide a family of conditions parameterized by a positive parameter $\tau$ whose meaning will be discussed soon.

Let $\tau > 0$. If $f$ is piecewise polynomial we define

$$w(\tau) = \#\{i \mid V_i \text{ contains an } n\text{-dimensional cube of side } \tau\}.$$

For the rest of this paper we assume that $\tau$ satisfies $w(\tau) > 0$. Let

$$B_\tau = \inf\{b \in \mathbb{R} \mid \text{ there exist cubes as above which are contained in } [-b, b]^n\}.$$

Denote by $I_\tau$ the set of indices $i$ satisfiying the condition in the definition of $w(\tau)$ and let

$$d_\tau = \max_{i \in I_\tau} \text{degree}\,(f_i) \qquad \text{and} \qquad C_\tau = \min_{\substack{i \neq j \\ i,j \in I_\tau}} \|f_i - f_j\|_\infty$$

where polynomials are identified with their vectors of coefficients. Define

$$\Gamma_\tau = \frac{C_\tau}{2}\left(\frac{2^{\frac{(D_\tau-1)^2}{4}-1}}{(D_\tau+1)\mathbf{B}_\tau}\left(\frac{\tau}{N_\tau}\right)^{\frac{D_\tau(D_\tau+1)}{2}}\right)^n$$

where $D_\tau = \max\{d_\tau, w(\tau)\}$, $N_\tau = w(\tau)D_\tau n + 1$, and

$$\mathbf{B}_\tau = \begin{cases} B_\tau^{D_\tau^2} & \text{if } B_\tau \geq 1 \\ B_\tau^{D_\tau - 1} & \text{if } B_\tau < 1. \end{cases}$$

We can now state our main theorem.

**Theorem 1** *If $T$ approximates a piecewise polynomial function $f$ with absolute accuracy $\delta$ and*

$$\delta \leq \Gamma_\tau$$

*then the depth $k$ of $T$ satisfies*

$$k \geq \log_2 w(\tau).$$

**Remark 2** Before proving Theorem 1 it may be helpful to say a few words on the meaning of $\tau$. Let $I$ be the set of indices such that $\dim V_i = n$. Then, we can define

$$\tau^* = \inf\{r \mid V_i \text{ contains an } n\text{-dimensional cube of side } r \text{ for all } i \in I\}.$$

For $\tau \geq \tau^*$ the inclusion $I_\tau \subseteq I$ may be strict and therefore $w(\tau)$ may be smaller than $w(\tau^*)$. But, in exchange, we have $D_\tau \geq D_{\tau^*}$ and $C_\tau \geq C\tau^*$. Therefore, $\Gamma_\tau$ may be greater than $\Gamma_{\tau^*}$. We conclude that by increasing $\tau$ beyond $\tau^*$ the lower bound may be decreased but the accuracy requirement may be relaxed. The exact form of this trade-off will depend on the function $f$ at hand and when applying Theorem 1 we will choose a $\tau$ which best fits our interests.

In proving Theorem 1 the following lemma is essential.

**Lemma 1** *Let $f \in \mathbb{R}[x_1, \ldots, x_n]$ with $\operatorname{degree}_{x_i}(f) \leq d$ and $M = \|f\|_\infty$. Let $b_1, \ldots, b_n \in \mathbb{R}$, $|b_i| \leq B$, $N \in \mathbb{N}$, $N > d$, and consider the uniform grid $\overline{S}$ with mesh $\tau/N$ in the cube*

$$\prod_{i=1}^n [b_i - \tau, b_i].$$

*Let $S \subseteq \overline{S}$ with $|S| = s$. If*

$$s > s_n = N^n\left(1 - \left(1 - \frac{d}{N}\right)^n\right) = N^n - (N-d)^n$$

*then there exists $x \in S$ such that*

$$|f(x)| > \alpha = M \left( \frac{2^{\frac{(d-1)^2}{4}-1}}{(d+1)\mathbf{B}} \left(\frac{\tau}{N}\right)^{\frac{d(d+1)}{2}} \right)^n$$

*where*

$$\mathbf{B} = \begin{cases} B^{d^2} & \text{if } B \geq 1 \\ B^{d-1} & \text{if } B < 1. \end{cases}$$

PROOF. By induction on $n$.

BASE CASE, $n = 1$. In this case, $s_1 = d$, so assume there is a subset $S_0$ of $\overline{S}$ having $d+1$ points $w_0, \ldots, w_d$ in $\overline{S}$ such that $|f(w_i)| \leq \alpha$ for $i = 0, \ldots, d$. Then, interpolating $f$ at these points we express each coefficient of $f$ as a fraction

$$\frac{\sum_{x \in S_0} a_x f(x)}{\Delta}$$

where

$$\Delta = \prod_{\substack{w_i, w_j \in S_0 \\ 0 \leq j < i \leq d}} (w_i - w_j)$$

is the determinant of the Vandermonde matrix

$$V = \begin{pmatrix} 1 & w_0 & w_0^2 & \ldots & w_0^d \\ 1 & w_1 & w_1^2 & \ldots & w_1^d \\ \vdots & & & & \vdots \\ 1 & w_d & w_d^2 & \ldots & w_d^d \end{pmatrix}$$

and $a_x$ are the determinants of suitable minors of $V$. The smallest possible value of $|\Delta|$ occurs when $w_0, \ldots, w_d$ are consecutive in $S$ (i.e. $w_i - w_{i-1} = \tau/N$) and in this case we have

$$|\Delta| \geq \prod_{i=1}^{d} \left(\frac{i\tau}{N}\right)^{d-i+1} = \left(\frac{\tau}{N}\right)^{\frac{d(d+1)}{2}} \prod_{i=1}^{d} i!.$$

On the other hand, by bounding each of the $d!$ terms in the definition of determinant we have $|a_x| \leq B^{d^2} d!$ if $B \geq 1$ and $|a_x| \leq B^{d-1} d!$ if $B < 1$. That is, $|a_x| \leq \mathbf{B} d!$. Therefore, the absolute value of each coefficient of $f$ is less than

$$\frac{(d+1)\mathbf{B} d! \alpha}{\left(\frac{\tau}{N}\right)^{\frac{d(d+1)}{2}} \prod_{i=1}^{d} i!} \leq \frac{(d+1)\mathbf{B}\alpha}{\left(\frac{\tau}{N}\right)^{\frac{d(d+1)}{2}} 2^{\frac{(d-1)^2}{4}-1}}$$

the last inequality since $\prod_{i=1}^{d} i! \geq 2^{\frac{d^2}{4}-1}$ for all $d \geq 1$. But then

$$M < \frac{(d+1)\mathbf{B}\alpha}{\left(\frac{\tau}{N}\right)^{\frac{d(d+1)}{2}} 2^{\frac{(d-1)^2}{4}-1}}$$

which is in contradiction with the definition of $\alpha$.

INDUCTION STEP, $n \geq 2$. Write $f = \sum_I f_I X^I$ where $f_I \in \mathbb{R}[x_n]$ and $X^I$ is a monomial in $x_1, \ldots, x_{n-1}$. Now, take $f_{I_0}$ such that $\|f_{I_0}\|_\infty = M$. By the base of the induction, for all but at most $d$ points $x$ in the set

$$L = \left\{ b_n - \tau, b_n - \tau + \frac{\tau}{N}, b_n - \tau + \frac{2\tau}{N}, \ldots, b_n - \tau + \frac{N\tau}{N} = b_n \right\}$$

we have

$$|f_{I_0}(x)| > \frac{M \left( 2^{\frac{(d-1)^2}{4} - 1} \frac{\tau}{N} \right)^{\frac{d(d-1)}{2}}}{(d+1)\mathbf{B}}. \tag{1}$$

Therefore, there are more than $s_n - dN^{n-1}$ points in $S$ whose last coordinate satisfies (1). We conclude that there exists one such point $x^* \in L$ such that, moreover,

$$|S \cap \{x_n = x^*\}| \geq \frac{s_n - dN^{n-1}}{N - d} = s_{n-1}.$$

Now apply the inductive hypothesis to the polynomial $f_{|x_n = x^*} \in \mathbb{R}[x_1, \ldots, x_{n-1}]$ using that

$$\|f_{|x_n = x^*}\|_\infty > \frac{M \left( 2^{\frac{(d-1)^2}{4} - 1} \frac{\tau}{N} \right)^{\frac{d(d-1)}{2}}}{(d+1)\mathbf{B}}$$

and the conclusion follows. □

## 2 Proof of Theorem 1

Recall that $I_\tau$ is the set of indices $i$ satisfiying the condition of the definition of $w(\tau)$, $D_\tau = \max\{w(\tau), d_\tau\}$, and $N_\tau = w(\tau) D_\tau n + 1$. Let $i \in I_\tau$ and consider the grid $\overline{S} \subset V_i$ as in Lemma 1. We say that a leaf $\eta$ of $T$ is *attached* to $V_i$ if $\eta$ is reached by at least $N_\tau^n / w(\tau)$ points of $\overline{S}$.

We claim that one leaf of $T$ can not be attached to two different sets $V_i$. From this claim it follows that $k \geq \log_2 w(\tau)$. Indeed, if $k < \log_2 w(\tau)$ then $|\text{Leaves}(T)| < w(\tau)$ and, by the pigeonhole principle, there is a leaf of $T$ attached to $V_i$. So, every $V_i$ has a leaf attached to it. And, by hypothesis, each leaf of $T$ is attached to at most one $V_i$. But then $|\text{Leaves}(T)| \geq w(\tau)$ and therefore, $k \geq \log_2 w(\tau)$.

To prove the claim, assume that there exist sets $V_i$ and $V_j$, $i \neq j \in I_\tau$, such that a leaf $\eta$ is attached to both of them. Let $g_\eta$ be the polynomial computed along the branch leading to $\eta$ and $C' = \|f_i - f_j\|_\infty$. Then either $\|f_i - g_\eta\|_\infty \geq C'/2$ or $\|f_j - g_\eta\|_\infty \geq C'/2$. We can assume, w.l.o.g., that the first inequality holds.

Let $S \subset \overline{S}$ be the set of points reaching the leaf $\eta$. Then,

$$|S| \geq \frac{N_\tau^n}{w(\tau)}.$$

Since $N_\tau > w(\tau) D_\tau n$, we have

$$|S| > N_\tau^n \frac{D_\tau n}{N_\tau} = N_\tau^{n-1} D_\tau n \geq N_\tau^n - (N_\tau - D_\tau)^n.$$

Thus we can apply Lemma 1 to the polynomial $f = f_i - g_\eta$ with $M = C'/2$, $d = D_\tau$ and $B = B_\tau$ and we deduce that there is a point $x \in S$ such that

$$|f_i(x) - g_\eta(x)| > \frac{C'}{2} \left( \frac{2^{\frac{(D_\tau - 1)^2}{4} - 1}}{(D_\tau + 1)\mathbf{B}_\tau} \left( \frac{\tau}{N_\tau} \right)^{\frac{D_\tau(D_\tau+1)}{2}} \right)^n \geq \Gamma_\tau$$

since $C' \geq C_\tau$. But this, together with the hypothesis on $\Gamma_\tau$, contradicts the fact that $|T(x) - f_i(x)| \leq \delta$. □

A lower bound for relative approximations easily follows from the proof of Theorem 1. Let

$$H_\tau = \max_{x \in [-B_\tau, B_\tau]^n} |f(x)|.$$

**Corollary 1** *If $T$ approximates a piecewise polynomial function $f$ with relative accuracy $\delta$ and*

$$\delta \leq \frac{\Gamma_\tau}{H_\tau}$$

*then the depth $k$ of $T$ satisfies*

$$k \geq \log_2 w(\tau).$$

□

**Remark 3** In the sequel we will state our results only for approximations with absolute accuracy $\delta$. Results for those with relative accuracy $\delta$, such as Corollary 1, follow immediately from the former.

**Remark 4** The lower bound in Theorem 1 (or that in Corollary 1) is on the depth of $T$. A more involved issue is the consideration of the *topological complexity* of $f$ (cf. [Smale 1987] for this concept, see also [Vassiliev 1992]), i.e. the number of leaves of $T$. This number is essentially the amount of branching necessary for solving the problem. In our discussion of Example 1 we saw that the topological complexity of $f$ is at least the number of 2-dimensional regions $V_i$ with pairwise different $v_i$ which is at least $w(\tau)$ for each $\tau > 0$.

For the problem MAX, consisting of finding the largest coordinate of an input $x \in \mathbb{R}^n$ and for which the number of pieces is $n$, the question of the topological complexity is open (see [Grigoriev, Karpinski, and Yao 1998] for the discussion and the exponential lower bound for *ternary* rathen than the usual binary computation trees).

Implicit in the proof of Theorem 1 is the fact that, if $k = \log_2 w(\tau)$, then the topological complexity of $T$, $\mathrm{TC}(T)$, satisfies $\mathrm{TC}(T) \geq w(\tau)$. It is unclear to us whether one can trade topological for arithmetical complexity, that is, whether one can reduce the topological complexity of an approximated computation at the expense of increasing the degree of the computed polynomials. We can prove, however, a trade-off between these complexities (and the approximation accuracy $\delta$). Let $T$ be an algebraic computation tree and $g_\eta$ the polynomial computed at leaf $\eta$. Define

$$d_T = \max_{\eta \text{ a leaf of } T} \text{degree}\,(g_\eta).$$

Note that $d_T \leq 2^k$ where $k$ is the depth of $T$. Now define $D_{(\tau,T)} = \max\{d_\tau, d_T\}$ and

$$\Gamma_{(\tau,T)} = \frac{C_\tau}{2}\left(\frac{2^{\frac{(D_{(\tau,T)}-1)^2}{4}-1}}{(D_{(\tau,T)}+1)\mathbf{B}_{(\tau,T)}}\left(\frac{\tau}{N\,\tau}\right)^{\frac{D_{(\tau,T)}(D_{(\tau,T)}+1)}{2}}\right)^n$$

with $\mathbf{B}_{(\tau,T)}$ as in Section 1. The arguments of Theorem 1 yield the following.

**Theorem 2** *If $T$ approximates $f$ with absolute accuracy $\delta$ and*

$$\delta \leq \Gamma_{(\tau,T)}$$

*then the topological complexity* $\mathrm{TC}(T)$ *of $T$ satisfies*

$$\mathrm{TC}(T) \geq w(\tau).$$

□

# 3 Round-off trees

A *round-off* tree is an algebraic computation $T$ whose arithmetic operations are subject to some form of error. Typical examples arise when considering computations in floating-point or fixed-point arithmetic.

In what follows, we will prove lower bounds for round-off trees. We will not rely on any special kind of error. These errors can be produced by rounding or by chopping, and can satisfy bounds either for their absolute or relative magnitude. Actually, the only hypothesis for our lower bounds to hold will be the usual bound on the outcome's accuracy and an additional hypothesis requiring that the sequence of arithmetic operation performed by the tree produces an equally accurate result. Let's describe this more precisely.

If $\eta$ is a leave of $T$, denote by $g_\eta$ the polynomial computed with exact arithmetic along the path ending in $\eta$ and by $\widetilde{g_\eta}$ the function computed along this path when errors are allowed.

**Definition 3** Let $T$ be a round-off tree with input space $\mathbb{R}^n$ and output space $\mathbb{R}$, and let $f : \mathbb{R}^n \to \mathbb{R}$ be a function.

We say that $T$ *approximates* $f$ *with absolute accuracy* $\delta$ if for every input $x \in \mathbb{R}^n$ the output $T(x)$ of $T$ satisfies

1) $|T(x) - f(x)| \leq \delta$, and

2) If the round-off computation of $T$ with input $x$ leads to the leave $\eta$ then $|\widetilde{g_\eta}(x) - g_\eta(x)| \leq \delta$.

Similarly, we say that $T$ *approximates* $f$ *with relative accuracy* $\delta$ upon replacing $\delta$ by $\delta|f(x)|$ in the two conditions above.

**Remark 5** Notice that the adjectives "absolute" and "relative" can apply to both the errors occuring along the computation (round-off errors) and the accuracy of its outcome. However, there is no need to bound in the same way the accuracy and the round-off errors and one finds instances of algorithms with combinations of different kinds. For instance, algorithms in numerical linear algebra, say for linear equation solving, usualy consider both relative round-off errors and relative accuracy (see [Demmel 1997]); relative round-off errors are actually common in numerical analysis since they correspond to floating-point arithmetic. The main result of [Cucker and Smale 1997] considers absolute round-off errors but infinite accuracy in the answer (the problem considered there, being decisional, does not allow for approximate answers). Also, for some results on integration (cf. [Koiran 1995]), absolute accuracy is considered for exact algorithms. The list of combinations may continue but we will stop here.

A version of Theorem 1 for round-off trees follows.

**Theorem 3** *Let $T$ be a round-off tree with depth $k$. If $T$ approximates a piecewise polynomial function $f$ with absolute accuracy $\delta$ and*

$$\delta \leq \frac{\Gamma_\tau}{2}$$

*then $k \geq \log_2 w(\tau)$.*

PROOF. One proceeds as in the proof of Theorem 1 to show that if $k < \log_2(w(\tau)$ then there is a point $x \in \mathbb{R}^n$ whose computation ends in a leave $\eta$ of $T$ satisfying

$$|f(x) - g_\eta(x)| > \Gamma_\tau.$$

But since $T$ $\delta$-approximates $f$ we have

$$|f(x) - T(x)| \leq \delta \leq \frac{\Gamma_\tau}{2} \qquad \text{and} \qquad |T(x) - g_\eta(x)| \leq \delta \leq \frac{\Gamma_\tau}{2}$$

the latter since $T(x) = \widetilde{g_\eta}(x)$. Therefore $|f(x) - g_\eta(x)| \leq \Gamma_\tau$ which is a contradiction. □

# 4 Extensions

Theorem 1 can be extended to some contexts where trees are endowed with additional capabilities. In this section we briefly discuss how this is carried out for two such capabilities: randomization and parallelism. We will state our results only for exact approximation trees. The result for round-off trees holds as well in the case of randomized trees but we do not know how to prove it for parallel trees.

## 4.1 Randomized Trees

One can define randomized versions of approximation trees by allowing "coin tossing" and requiring the output to be a $\delta$-approximation with high probability. More precisely, we consider trees with input space $\mathbb{R}^n \times \{0,1\}^m$ (for the arguments which follow the exact value of $m$ is not important) and we fix a confidence degree $\sigma$ satisfying $0 < \sigma \leq 1$. Then, such a tree approximates $f$ with absolute accuracy $\delta$ when, for each $x \in \mathbb{R}^n$ and for at least $\sigma 2^m$ points $b$ in $\{0,1\}^m$, we have $|T(x,b) - f(x)| \leq \delta$.

Assume that this happens and let $X$ be the union of the grids $\overline{S}$ associated to the sets $V_i$ with $i \in I_\tau$. Then there exists a point $b^* \in \{0,1\}^m$ such that for at least $\sigma|X|$ points in $X$ we have $|T(x,b) - f(x)| \leq \delta$. Fix the coin tossing $b^*$ and call these points *good* (with respect to $b^*$).

**Lemma 2** *At least* $\frac{\sigma}{2-\sigma} w(\tau)$ *sets* $V_i$ *contain more than* $\frac{\sigma}{2} N_\tau^n$ *good points.*

Proof. Let $\alpha$ be the number of sets $V_i$ containing more than $\frac{\sigma}{2} N_\tau^n$ good points. Then

$$|\text{good points}| \leq \alpha N_\tau^n + (w(\tau) - \alpha) N_\tau^n \frac{\sigma}{2}$$

and since the number of good points is at least $\sigma N_\tau^n w(\tau)$ the result follows. □

To replicate the proof of Theorem 1 we now consider the deterministic tree resulting from replacing the coin tossing by the fixed point $b^*$ and we modify the quantities appearing in the definition of $\Gamma_\tau$ to allow for the confidence $\sigma$. Thus, we define $I_{(\tau,\sigma)}$ to be the subset of $I_\tau$ with those indices $i$ such that $V_i$ satisfies Lemma 2. Then, one defines $d_{(\tau,\sigma)}, C_{(\tau,\sigma)}, D_{(\tau,\sigma)}, N_{(\tau,\sigma)}$ and $\Gamma_{(\tau,\sigma)}$ as in Section 1.

Notice that $d_{(\tau,\sigma)} \leq d_\tau$, $C_{(\tau,\sigma)} \geq C_\tau$, etc. and so $\Gamma_{(\tau,\sigma)} \geq \Gamma_\tau$.

**Theorem 4** *If* $T$ *is a randomized tree which approximates* $f$ *with absolute accuracy* $\delta$ *and confidence* $\sigma$, *and*

$$\delta \leq \Gamma_{(\tau,\sigma)}$$

*then the depth* $k$ *of* $T$ *satisfies*

$$k \geq \log_2 \left( \frac{\sigma}{2} w(\tau) \right).$$

Sketch of proof. We say that a leaf $\eta$ is attached to $V_i$ if $\eta$ is reached by at least $N^n_{(\tau,\sigma)}/w(\tau)$ *good* points in $\overline{S}$.

Again, we claim that a leaf can not be attached to two different sets $V_i$ and from this claim it follows the theorem. Indeed, if

$$k < \log_2\left(\frac{\sigma}{2}w(\tau)\right)$$

then $|\text{Leaves}(T)| < \sigma w(\tau)/2$ and, by the pigeonhole principle, there is a leaf of $T$ attached to $V_i$. So, every $V_i$ has a leaf attached to it. And, by hypothesis, each leaf of $T$ is attached to at most one $V_i$. But then

$$|\text{Leaves}(T)| \geq \frac{\sigma}{2-\sigma}w(\tau) \geq \frac{\sigma}{2}w(\tau)$$

and therefore, $k \geq \log_2(\frac{\sigma}{2}w(\tau))$.

The claim is proved as in Theorem 1. □

**Remark 6** When dealing with decision problems, the confidence degree $\sigma$ is assumed to be greater than 1/2 (or in other words, the probability error $\varepsilon = 1 - \sigma$ is assumed to be smaller than 1/2). This is due to the fact that an algorithm consisting of tossing a coin and answering `Yes` or `No` according to the outcome of that coin tossing (and independently of the input) is already a probabilistic algorithm of confidence 1/2. Theorem 4 shows that such a simple algorithm is not going to work in the non-decisional case.

We also mention that a complexity lower bound for a probabilistic tree deciding an arrangement of hyperplanes or a polyhedron was obtained in [Grigoriev 1998]. This bound is logarithmic in the number of faces.

### 4.2 Parallel Trees

Parallel computations can be modelled by a particular kind of trees. If $p$ denotes the number of processors, at each computational node, the tree performs an arithmetic operation and stores its result in at most $p$ coordinates of the state space. Also, at each branching node, the sign of at most $p$ such coordinates is tested, giving thus rise to $2^p$ possible outcomes. An elementary computation yields an upper bound of $2^{pk}$ leaves for such a tree with depth $k$. Since in most parallel models the number of processors is bounded by $2^k$ this upper bound becomes $2^{k2^k}$.

If the computations are performed exactly (without errors) it turns out that most of these leaves are irrelevant in the sense that there are no points in $\mathbb{R}^n$ reaching them. More precisely, Yao [1981] (see also [Montaña and Pardo 1993]) shows that in this case, the number of leaves which are reached by points in $\mathbb{R}^n$ is bounded by

$$2^{\mathcal{O}(k^2 n)}.$$

Notice that from this it follows the inequality

$$k \geq \Omega\left(\sqrt{\frac{\log |\text{Leaves}(T)|}{n}}\right).$$

We remark that an upper bound close to the latter lower one (for small dimensions) for the parallel complexity of deciding an arrangement of hyperplanes or a polyhedron (as in Remark 6) was given in [Grigoriev 1997].

An almost verbatim repetition of the proof of Theorem 1 yields the following which, we recall, we can only prove for exact trees.

**Theorem 5** *If $T$ is a parallel tree which approximates $f$ with absolute accuracy $\delta$ and*

$$\delta \leq \Gamma_\tau$$

*then the depth $k$ of $T$ satisfies*

$$k \geq \Omega\left(\sqrt{\frac{\log_2 w(\tau)}{n}}\right).$$

□

**Remark 7** The requirement of exact arithmetic for $T$ in Theorem 5 seems unavoidable if we want to use Yao's bound on the number of relevant leaves. To see why, consider a set of $s$ lines in $\mathbb{R}^2$ given by linear polynomials $\ell_1, \ldots, \ell_s$ and assume that these lines pass through a common point $\xi$. Now consider a branch node which tests the signs of $\ell_1, \ldots, \ell_s$ at a point $x$. If $x = \xi$ and round-off errors are allowed when computing $\ell_i(\xi)$, $i = 1, \ldots, s$, we may get up to $2^s$ possible outcomes.

# References


Blum, L., F. Cucker, M. Shub, and S. Smale (1998). *Complexity and Real Computation.* Springer-Verlag.

Bürgisser, P., M. Clausen, and A. Shokrollahi (1996). *Algebraic Complexity Theory.* Springer-Verlag.

Cucker, F. and S. Smale (1997). Complexity estimates depending on condition and round-off error. To appear in *Journal of the ACM.*

Demmel, J. (1997). *Applied Numerical Linear Algebra.* SIAM.

Grigoriev, D. (1997). Nearly sharp complexity bounds for multiprocessor algebraic computations. *Journal of Complexity 13*, 50–64.

Grigoriev, D. (1998). Randomized complexity lower bounds. In *30th annual ACM Symp. on the Theory of Computing*, pp. 219–223.

GRIGORIEV, D., M. KARPINSKI, and A. YAO (1998). An exponential lower bound on the size of algebraic decision trees for MAX problem. To appear in *Computational Complexity.*

KOIRAN, P. (1995). Approximating the volume of definable sets. In *36th annual IEEE Symp. on Foundations of Computer Science*, pp. 134–141.

MONTAÑA, J. and L. PARDO (1993). Lower bounds for arithmetic networks. *Applicable Algebra in Engineering, Communication and Computing 4*, 1–24.

RENEGAR, J. (1987). On the worst case arithmetic complexity of approximating zeros of polynomials. *Journal of Complexity 3*, 90–113.

SMALE, S. (1987). On the topology of algorithms I. *Journal of Complexity 3*, 81–89.

STRASSEN, V. (1983). The computational complexity of continued fractions. *SIAM Journal on Computing 12*, 1–27.

VASSILIEV, V. (1992). *Complements of Discriminants of Smooth Maps: Topology and Applications*, Volume 98 of *Translations of Mathemathical Monographs*. American Mathematical Society.

YAO, A. (1981). On the parallel computation of the knapsack problem. In *13th annual ACM Symp. on the Theory of Computing*, pp. 123–127.