On Implications between P-NP-Hypotheses: Decision versus Computation in Algebraic Complexity *

Peter Bürgisser

Dept. of Mathematics and Computer Science, University of Paderborn, D-33095 Paderborn, Germany, Email: buergisser@upb.de

Abstract. Several models of NP-completeness in an algebraic framework of computation have been proposed in the past, each of them hinging on a fundamental hypothesis of type P≠NP. We first survey some known implications between such hypotheses and then describe attempts to establish further connections. This leads us to the problem of relating the complexity of computational and decisional tasks and naturally raises the question about the connection of the complexity of a polynomial with those of its factors. After reviewing what is known with this respect, we discuss a new result involving a concept of approximative complexity.

1 Introduction

Algebraic complexity theory is the study of the intrinsic difficulty of computational problems that can be posed in an algebraic or numerical framework. Instead of basing this study on the model of the Turing machine, it uses "algebraic models" of computation. Besides the fact that these models form a natural theoretical framework for the algorithms commonly known to solve the problems under consideration, the main motivation for this choice of model is the idea that various methods from pure mathematics like algebraic geometry or topology can be employed to establish lower bounds in this more structured world. This project has been successful for problems of polynomially bounded complexity, as illustrated by the large body of work presented in the textbook [14].

The theory of NP-completeness is one of the main cornerstones of computational complexity, although still resting on the unproven hypothesis that $P \neq NP$. In seminal works by Valiant [45, 47] and Blum, Shub and Smale [9] models of NP-completeness in an algebraic framework of computation have been proposed, motivated by the hope to prove the elusive separation of P and NP in these frameworks. So far, this hope has not been fulfilled except in rather trivial cases. However, there have been successful attempts in relating these different models to each other and in establishing implications between the various P-NP-hypotheses presently studied. The known "transfer theorems" either relate such

* To appear in Proc. MFCS 2001, Springer LNCS,

hypotheses of the same type refering to different fields, or provide implications from the separation of P and NP in the classical bit model to a corresponding separation in an algebraic model of computation. As an exception to this rule, Fournier and Koiran [20] (see also [31]) recently proved an implication in the reverse direction for a restricted algebraic model. This has challenged the hope that a P-NP-separation in the algebraic model might be easier to prove than in the bit model. We review some of the known transfer results in Section 3.

In Section 4 we will discuss an attempt [12] to relate the P-NP hypothesis in Valiant's framework to the one in the Blum, Shub, Smale (BSS) framework, as well as an attempt [42] to establish a connection to the complexity of certain univariate polynomials. This leads us to the problem of relating the complexity of computational and decisional tasks and naturally raises the question about the connection of the complexity of a polynomial with those of its factors. This relationship is not well understood and turns out to be the bottleneck in both attempts.

In Section 5 we review what is known about the complexity of factors. We mention a result by Kaltofen [27], which seems to be widely unknown in the community (and has been independently discovered by the author). It states that the complexity of an irreducible factor g of a polynomial f can be polynomially bounded in the complexity of f, the degree of g, and the multiplicity of g. We believe that the dependence on the multiplicity is not necessary, but we have been unable to prove this. Instead, we present a new result [10], which states that the dependence on the multiplicity can be avoided when replacing complexity by the related notion of "approximative complexity", which is introduced in Section 6.

As a major application, we obtain the following relative hardness result about decision complexity over the reals: Checking the values of polynomials forming complete families in Valiant's sense cannot be done with a polynomial number of arithmetic operations and tests, unless the permanent has a p-bounded approximative complexity, which seems unlikely. This hardness result extends to randomized algorithms with two-sided error, formalized by randomized algebraic computation trees.

2 Algebraic Models of NP-Completeness

2.1 Blum-Shub-Smale model

Blum, Shub, and Smale [9] have extended the classical theory of NP-completeness to a theory of computation over arbitrary rings, the three most interesting cases being the field of real or complex numbers, and the finite field $\mathbb{F}_2$. In the latter case, the classical theory of computation is recovered. As usual in algebraic complexity theory, a basic computational step is an arithmetic operation, an equality test, or a $\leq$-test if the field is ordered, and we make the idealizing assumption that this can be done with infinite precision. Moreover a uniformity condition is assumed to be satisfied. For a recent account see [8]. Poizat [40] describes an elegant approach to a P-NP-framework over general structures.

Many ideas of discrete structural complexity can be extended to such a framework; in particular the complexity classes P, NP and the notion of NP-completeness. In this framework, a natural NP-complete problem turns out to be the feasibility problem to decide for a given system of polynomials whether they have a common root. In all three settings $(\mathbb{F}_2, \mathbb{R}, \mathbb{C})$ it is a fundamental open problem whether P $\neq$ NP is true. It has been shown [7] that this question has the same answer over all algebraically closed fields of characteristic zero. Over the reals, no corresponding transfer theorem is known.

2.2 Valiant's algebraic model

In [45, 47] Valiant proposed an analogue of the theory of #P-completeness in a framework of algebraic complexity, in connection with his famous hardness result for the permanent [46]. This theory features algebraic complexity classes VP and VNP as well as VNP-completeness results for many families of generating functions of graph properties, the most prominent being the family of permanents. For a comprehensive presentation of this theory, we refer to [21, 14] and to the recent account [12].

While the complexity classes in the BSS-modell capture decision problems, the Valiant classes deal with the computational problem to evaluate multivariate polynomials. Only straight-line computations are considered and uniformity is not taken into account. The basic object studied is a p-family over a fixed field k, which is a sequence (f_n) of multivariate polynomials such that the number of variables as well as the degree of f_n are polynomially bounded (p-bounded) functions of n. The complexity class VP over k consists of the p-families (f_n) which are p-computable, which means that the straight-line complexity of f_n is p-bounded in n. Note that although X^{2^n} can be computed with only n multiplications, the corresponding sequence is not considered to be p-computable, as the degrees grow exponentially. A p-family (f_n) is called p-definable iff there exists $(g_n) \in$ VP such that for all n

$$f_n(X_1, \ldots, X_{v(n)}) = \sum_{e \in \{0,1\}^{u(n)-v(n)}} g_n(X_1, \ldots, X_{v(n)}, e_{v(n)+1}, \ldots, e_{u(n)}).$$

The set of p-definable families form the complexity class VNP. The class VP is obviously contained in VNP, and Valiant's hypothesis claims that this inclusion is strict. It has been shown in [12, § 4.1] that over algebraically closed fields, this hypothesis depends at most on the characteristic of the field.

3 Known Implications between P-NP-Hypotheses

In each of the algebraic models discussed before we have raised the fundamental question whether P $\neq$ NP. Recently, a considerable amount of research has been directed towards establishing "transfer theorems" that provide implications from the separation of P and NP in the classical bit model to a corresponding

separation in an algebraic model of computation. These results rely on various techniques to eliminate the real or complex constants that may be used by a computation in the algebraic model on a Boolean input. One of the first results of this kind was established by Koiran [29] for the additive BSS-model over the reals, which allows additions and subtractions as the only arithmetic operations. Koiran showed that

$$\text{P} \neq \text{NP nonuniformly} \Longrightarrow \text{P} \neq \text{NP over } \mathbb{R} \text{ as an ordered group.} \qquad (1)$$

Here, the elimination of constants is based on polyhedral geometry, in particular on the fact that a nonempty polyhedron defined by a system of inequalities with small coefficients has a small rational point. Quite astonishingly, the converse of the above implication is also true, as recently established by Fournier and Koiran [20]. The proof of the converse of (1) relies on Meyer auf der Heide's [37, 39] construction of small depth linear decision trees for locating points in arrangements of hyperplanes.

For the unrestricted BSS-model over the reals (with order) no implication in either direction is known. Over the complex numbers, we know the following about the unrestricted BSS-model

$$\text{P} \neq \text{NP nonuniformly} \Longrightarrow \text{P} \neq \text{NP over } \mathbb{C}, \qquad (2)$$

as noticed independently by several researchers. The essential point here is that using modular arithmetic, it is possible to test in random polynomial time whether an integer given by a straight-line program equals zero. Actually, the left-hand side can be replaced by the weaker statement $\text{NP} \not\subseteq \text{BPP}$. (A proof can be found in Cucker et al. [18], although somewhat hidden.) We do not know how to efficiently test for positivity of an integer given by a straight-line program, and this seems to be the main obstacle for establishing an implication analogous to (2) over the reals.

In [13] we have shown the following implication for Valiant's model over $\mathbb{F}_2$:

$$\text{NC}^2 \neq \oplus\text{P nonuniformly} \Longrightarrow \text{VP} \neq \text{VNP over } \mathbb{F}_2\,. \qquad (3)$$

More specifically, by interpreting families of polynomials over $\mathbb{F}_2$ as Boolean functions, we can assign to an algebraic class $\mathcal{C}$ its Boolean part $\text{BP}(\mathcal{C})$. It turns out that over $\mathbb{F}_2$ we have

$$\text{NC}^1/\text{poly} \subseteq \text{BP(VP)} \subseteq \text{NC}^2/\text{poly}, \quad \text{BP(VNP)} = \oplus\text{P/poly},$$

which immediately implies (3). Note that the class VNP over $\mathbb{F}_2$ is by definition closely related to $\oplus$P. On the other hand, the Boolean function corresponding to a p-computable family lies in NC^2/poly, due to the fact [6] that an algebraic straight-line program of size $n^{O(1)}$ computing a polynomial of degree $n^{O(1)}$ can be efficiently parallelized, i.e., simulated by a straight-line program of size $n^{O(1)}$ and depth $O(\log^2 n)$. A challenging open problem is to find out to what extent implication (3) might be reversed: does $\text{BP(VP)} = \text{BP(VNP)}$ imply that VP = VNP over $\mathbb{F}_2$? As a first step in this direction, it seems rewarding to locate

BP(VP) exactly in the hierarchy of known complexity classes between $\mathrm{NC}^1/\mathrm{poly}$ and $\mathrm{NC}^2/\mathrm{poly}$.

For Valiant's model in characteristic zero, we have proved in [13] the following implication similar to (3):

$$\mathrm{FNC}^3 \neq \#\mathrm{P} \text{ nonuniformly} \Longrightarrow \mathrm{VP} \neq \mathrm{VNP} \text{ in characteristic zero,}$$

conditional on the generalized Riemann hypothesis. Besides the ideas mentioned for the field $\mathbb{F}_2$, the proof is based on some elimination of constants, which is achieved by a general result about the frequency of primes p with the property that a system of integer polynomial equations solvable over $\mathbb{C}$ has a solution modulo p.

In Section 4.2 we will address the question whether VP $\neq$ VNP implies P $\neq$ NP over $\mathbb{C}$. The difficulty here is not to eliminate constants, but to link the complexity of decisional problems to the complexity of computational problems.

4 Attempts to Establish Further Connections

4.1 Linking Decisional to Computational Complexity

Are there functions, whose value can be checked in polynomial time but which cannot be computed in polynomial time? In fact, this is a basic assumption in cryptography, since it turns out to be equivalent to the existence of one-way functions [24, 41]. We ask now this question for a real or complex polynomial function g. More specifically, we ask whether deciding $g(x) = 0$ can be done considerably faster than just by computing g at input x?

In order to make this formal, we introduce the computational complexity $L(g)$ and the decisional complexity $C(g)$. The first one refers to the straight-line model of computation and counts all arithmetic operations (divisions are not allowed for simplicity). The decisional complexity refers to algebraic computation trees and counts besides arithmetic operations also branchings according to equality tests (and $\leq$-tests over the reals). Clearly, $C(g) \leq L(g) + 1$. In both cases, we allow that any real or complex numbers may be used as constants.

Assume that g is the product of m real linear polynomials: $g = h_1 \cdots h_m$. In the case $n = 1$ we have obviously $C(g) = O(\log m)$ using binary search. The result by Meyer auf der Heide [37, 39] and Meiser [36] extends this to higher dimensions and states that it is possible to locate a given point $x \in \mathbb{R}^n$ in the hyperplane arrangement given by $h_1, \ldots, h_m$ by an algebraic computation tree with depth $(n \log m)^{O(1)}$. (In fact, linear decision trees are sufficient for this, see [19].) We therefore have $C(g) = (n \log m)^{O(1)}$. On the other hand, one can show that the complexity for evaluating g equals $\Theta(mn)$ if the h_i are in general position [14, Chap. 5]. This example shows that computational and decisional complexity may differ dramatically. In Section 7.1 we will provide strong evidence that this phenomenon does neither occur over the complex numbers nor over the reals if g is irreducible and has a degree polynomially bounded in its complexity.

The following well-known lemma [16, 8] provides a link from decisional to computational complexity. It naturally leads to the question of relating the complexity of a polynomial to those of its multiples, which will be investigated systematically in Section 5.

Lemma 4.1. *There exists a nonzero multiple f of g such that $L(f) \leq C(g)$. Over $\mathbb{C}$ this is true without additional assumption, over $\mathbb{R}$ we have to require that g is irreducible.*

4.2 Does Valiant's hypothesis imply the BSS hypothesis over $\mathbb{C}$?

In [12, §8.4] we have conjectured that Valiant's hypothesis implies the BSS hypothesis over $\mathbb{C}$. Loosely speaking, this means that if the permanent is intractable, then solving systems of polynomial equations over $\mathbb{C}$ is intractable as well. The following reasonings from [12], similar as in Heintz and Morgenstern [25], have lead us to this conjecture.

The real, weighted cycle cover problem is the following decision problem: given a real $n \times n$ matrix $[w_{i,j}]$ of weights and a real number s, one has to decide whether there exists some permutation $\pi \in S_n$ such that $\sum_{i=1}^{n} w_{i,\pi(i)} = s$. (Note that a permutation can be visualized as a cycle cover.) By the above mentioned result [37, 39], this decision problem can be solved by algebraic computation trees over the reals with depth $n^{O(1)}$. We reformulate this problem now as follows: let $x_{i,j} = 2^{w_{i,j}}$ and $y = 2^{-s}$. Then the above question amounts to test whether the (reducible) polynomial

$$g_n := \prod_{\pi \in S_n} (1 - Y X_{1,\pi(1)} \cdots X_{n,\pi(n)})$$

vanishes for a given (positive) real matrix $x = [x_{i,j}]$ and $y > 0$.

If we expand g_n according to powers of Y, we see that the permanent of the matrix $[X_{i,j}]$ is the coefficient of Y: in fact, $g_n = 1 - Y \operatorname{PER}_n(X) + O(Y^2)$. A variant of a well-known result on the computation of homogeneous parts implies $L(\operatorname{PER}_n) \leq 4L(g_n)$ (compare [14, § 7.1]). Therefore, g_n is hard to compute if Valiant's hypothesis is true.

It is easy to see that the problem to check whether $g_n(x, y) = 0$ gives rise to a problem in the BSS complexity class NP over the reals. Indeed, for a given matrix x and $y \in \mathbb{R}$ it suffices to guess the permutation π and to check that $y x_{1,\pi(1)} \cdots x_{n,\pi(n)} = 1$.

We present now an attempt to deduce P $\neq$ NP over $\mathbb{C}$ from Valiant's hypothesis, based on the decision problem $g_n(x, y) = 0$ over the complex numbers. Assume that P = NP over $\mathbb{C}$. Then the above decision problem would lie in P and therefore could be solved by algebraic computation trees of depth polynomially bounded in n, hence $C(g_n) = n^{O(1)}$. By Lemma 4.1 there is a nonzero multiple f_n of g_n for each n such that $L(f_n) \leq C(g_n)$, hence $L(f_n) = n^{O(1)}$. If we could derive from this that the factor g_n has a complexity polynomially bounded in n, then we could conclude $L(\operatorname{PER}_n) = n^{O(1)}$, which contradicts Valiant's hypothesis, as the the permanent family is VNP-complete.

4.3 Univariate polynomials and P $\neq$ NP over $\mathbb{C}$

For given $n \in \mathbb{N}$ consider the problem ("Twenty Questions") to decide for a given complex number x whether $x \in \{1, 2, \ldots, n\}$. The complexity of this problem in the model of computation trees over $\mathbb{C}$ thus equals $C(p_n)$, where

$$p_n(X) := (X-1)(X-2)\cdots(X-n)$$

are the Pochhammer-Wilkinson polynomials. Shub and Smale [42] made the following reasoning similar to the one in Section 4.2. By considering the parameter n also as an input, the above decision problem is easily seen to lie in the class NP over $\mathbb{C}$, when we consider the pair (n, x) as an instance of size $\ell = \lceil \log n \rceil$. The point is that one can check whether x is an integer in the range $0, 1, \ldots, 2^\ell - 1$ by guessing complex numbers $w_0, \ldots, w_{\ell-1}$ and checking that $x = \sum_{i=0}^{\ell-1} w_i 2^i$ and $w_i(w_i - 1) = 0$ for all i.

Assume that P = NP over $\mathbb{C}$. Then the above decision problem would lie in P and we would have $C(p_n) = n^{O(1)}$. By Lemma 4.1 there would exist a nonzero multiple f_n of p_n with complexity $L(f_n) \leq C(p_n) = n^{O(1)}$. This argument can be refined: by eliminating the finite set of complex constants used by a BSS-machine, it is possible to achieve that the multiple f_n of p_n is an integer polynomial computed by a straight-line program of length $n^{O(1)}$ using 1 as the only constant. We call the minimal length of a straight-line program satisfying this additional condition the τ-complexity $\tau(f)$. Clearly, $L(f) \leq \tau(f)$.

Shub and Smale [42] set up the so-called τ-conjecture, which claims the following connection between the number $z(f)$ of distinct integer roots of a univariate integer polynomial f and its τ-complexity:

$$z(f) \leq (1 + \tau(f))^c \ ,$$

where $c > 0$ is a universal constant. The τ-conjecture thus implies P $\neq$ NP over the field of complex numbers.

In order to illustrate that the τ-conjecture is of a number theoretic quality, let us ask a more general question. Let k be a field, f be a polynomial in n variables over k, and $d \in \mathbb{N}$. We write $N_d(f)$ for the number of irreducible factors of f over k having degree at most d, not counting multiplicity. For a fixed field k, we raise the following question:

$$\exists c > 0 \ \forall n, d \ \forall f \in k[X_1, \ldots, X_n] : N_d(f) \leq (L(f) + d)^c. \tag{4}$$

It is clear that this statement over $\mathbb{Q}$ implies the τ-conjecture: the difference being that we take $n = d = 1$, count all rational roots of f, and measure complexity with L instead of τ. Refering to question (4), we observe the following:

(i) If (4) is true over some field k, then it must be true over the rationals.
(ii) Question (4) is false over finite fields, real or algebraically closed fields, and p-adic fields.
(iii) Over number fields one may equivalently take $n = 1$ in (4).

The proof is simple: first note that property (4) is inherited by subfields of k. A counterexample over $k = \mathbb{F}_q$ is provided by the factorization of $f = X^{q^d} - X$ into the product of all monic irreducible polynomials over $\mathbb{F}_q$ whose degree is a divisor of d. This example can be lifted to the p-adics. Over $\mathbb{R}$ and $\mathbb{C}$ one may use $f = X^n - 1$ or the Chebychev polynomials. The proof of (iii) is an immediate consequence of the Hilbert irreducibility theorem [32]. (We note that a corresponding conclusion with τ instead of L is not clear.)

It is interesting to note that the Pochhammer-Wilkinson polynomial p_n can be evaluated with $O(\sqrt{n}\log^2 n)$ arithmetic operations. Indeed, assume $n = m^2$ and write for $x \in \mathbb{R}$

$$p_{m^2}(x) = \prod_{\substack{0 \le q < m \\ 0 \le r < m}} (x - qm - r) = \prod_{q=0}^{m-1} h_m(q),$$

where $h_m(Y) := \prod_{r=0}^{m-1}(x - mY - r) = \sum_{i=0}^{m} a_i(x)Y^i$. Using FFT-based fast arithmetic (cf. [14, Chap. 2]) one can compute on input x all coefficients $a_i(x)$ and then use multiple evaluation to obtain $h_m(q)$ for all $q < m$ using only $O(m\log^2 m)$ arithmetic operations. A more detailed reasoning yields $\tau(p_{m^2}) = O(m\log^2 m \log\log m)$. We remark that a similar idea was first formulated for the efficient computation of factorials [44].

We do not know whether the sequence of Pochhammer-Wilkinson polynomials (p_n) is hard to compute in the sense that $L(p_n) \ge n^\epsilon$ for some $\epsilon > 0$. However, we can make the following interesting observation:

$$p_n(X^2) = q_n \cdot \overline{q}_n := \prod_{j=1}^{n}(X - \sqrt{j}) \cdot \prod_{j=1}^{n}(X + \sqrt{j}).$$

Using techniques of algebraic complexity theory, Heintz and Morgenstern [25] were able to prove that each of the sequences (q_n) and $(\overline{q}_n)$ is hard to compute (see also Baur [4]).

If we were able to extend the hardness proof for q_n to all of its nonzero multiples, then we had proved that all nonzero multiples of p_n are hard and thus P $\neq$ NP over $\mathbb{C}$. The problem to relate the complexity of a polynomial with its nonzero multiples appears here closely related to the P-NP problem over the complex numbers.

We remark that Aldaz at al. [1] showed that $\prod_{i=1}^{n}(X - 2^{2^i})$ is hard to compute and Baur and Halupczok [5] proved a corresponding lower bound for all the nonzero multiples of these polynomials.

5 Complexity of Factors

We proceed now with a systematic investigation of the relationship in complexity of a polynomial f with those of its factors. The first question to ask is whether the complexity of a factor g can always be polynomially bounded in the complexity of f. Our developments in Section 4.3 already indicate that the answer to this question is negative, since a positive answer would provide a proof of P $\neq$ NP over $\mathbb{C}$. The answer is indeed negative, as first discovered by Lipton

and Stockmeyer [35]. The simplest known example illustrating this is as follows: consider $f_n = X^{2^n} - 1 = \prod_{j<2^n}(X - \zeta^j)$, where $\zeta = \exp(2\pi i/2^n)$. By repeated squaring we get $L(f_n) \leq n+1$. On the other hand, one can prove that for almost all $M \subseteq \{0, 1, \ldots, 2^n - 1\}$ the random factor $\prod_{j\in M}(X - \zeta^j)$ has a complexity which is exponential in n, cf. [14, Exercise 9.8]. A similar reasoning can be made over the rationals based on the factorization into the cyclotomic polynomials. This idea yields reducible factors of high complexity.

Problem 5.1. Construct a sequence of *irreducible* polynomials g_n with complexity exponential in n, which have a nonzero multiple f_n with complexity polynomially bounded in n.

In the above example, the degree of the factor g is exponential in the complexity of f. We restrict now our attention to factors having a degree polynomially bounded in the complexity of f. A well-known result by Kaltofen [28] describes a randomized polynomial time algorithm for factoring a multivariate polynomial f given by a straight-line program. Hereby, the upper bound is polynomial in the straight-line complexity of f and the degree of f. In a widely unknown paper, Kaltofen [27] also proved that the complexity of any irreducible factor g is polynomially bounded in the complexity of f, the degree of g and the multiplicity of g. This result, stated explicitly below, was independently found by the author, compare [12, Thm. 8.14]. Hereby, the notation $M(d)$ stands for an upper bound on the complexity for the multiplication of two univariate polynomials of degree d over k, e.g. $M(d) = O(d \log d)$ if the field k "supports" fast Fourier transforms.

Theorem 5.2. *Assume $f = g^e h$ with polynomials $g, h \in k[X_1, \ldots, X_n]$ which are coprime. Let $d \geq 1$ be the degree of g and suppose that k is a field of characteristic zero. Then we have*

$$L(g) = O\big(M(d^3 e)(L(f) + d \log e)\big).$$

In [12, Conj. 8.3] we have conjectured that the dependence on the multiplicity e can be omitted, that is, we think that the complexity of the factor g is polynomially bounded in the complexity of f and the degree d of g.

In Section 7.1 we will present a new result [10] stating that the dependence on the multiplicity in Theorem 5.2 can indeed be omitted when replacing complexity by the related notion of "approximative complexity", to be defined next.

The fact that a computation with a polynomial number of steps may produce intermediate results of exponential degree is well-known to cause considerable complications. Actually, the so-called weak BSS model of computation [30] was defined in order to cope with this phenomenon, simply by forbidding such an exponential growth of degree. The fact that the P-NP separation in the weak model is trivial to obtain clearly shows that this model is a large oversimplification. We remark that the Valiant model also excludes an exponential growth of degrees during a computation. However, one can show [43] that this is no restriction for the study of polynomials of p-bounded degree, like permanents.

Note that resultants of systems of polynomial equations have a huge degree and are not captured by Valiant's framework.

6 Approximative Complexity

The concept of approximative complexity has been systematically studied in the framework of bilinear complexity (border rank) and there it has turned out to be one of the main keys to the currently best known fast matrix multiplication algorithms [17]. For computations of polynomials or rational functions, approximative complexity has been investigated in less detail. Griesser [22] generalized most of the known lower bounds for multiplicative complexity to approximative complexity. Lickteig [33] as well as Grigoriev and Karpinski [23] employ the notion of approximative complexity for proving lower bounds. We refer to [14, Chap. 15] and the references there for further information.

6.1 Algebraic definition and topological characterization

Let $f = f_q(X_1, \ldots, X_n)Y^q + f_{q+1}(X_1, \ldots, X_n)Y^{q+1} + \ldots$ be the expansion of a polynomial f with respect to the variable Y. We do not know whether the complexity of the leading coefficient f_q can be polynomially bounded in the the complexity of f. However, we can make the following observation. For the moment assume that k is the field of real or complex numbers. We have $\lim_{y \to 0} y^{-q} f(X, y) = f_q(X)$ and $L(f(X, y)) \leq L(f)$ for all $y \in k$. Thus we can approximate f_q with arbitrary precision by polynomials having complexity at most $L(f)$. We could say that f_q has "approximate complexity" at most $L(f)$.

We will formalize this in an algebraic way; a topological interpretation will be given later. In what follows, $K := k(\epsilon)$ is a rational function field in the indeterminate ϵ over the field k and R denotes the local subring of K consisting of the rational functions defined at $\epsilon = 0$. We write $F_{\epsilon=0}$ for the image of $F \in R[X]$ under the morphism $R[X] \to k[X]$ induced by $\epsilon \mapsto 0$.

Definition 6.1. Let $f \in k[X_1, \ldots, X_n]$. The *approximative complexity* $\underline{L}(f)$ of the polynomial f is the smallest natural number r such that there exists F in $R[X_1, \ldots, X_n]$ satisfying $F_{\epsilon=0} = f$ and $L(F) \leq r$. Here the complexity L is to be interpreted with respect to the larger field of constants K.

Even though L refers to division-free straight-line programs, divisions will occur implicitly since our model allows the free use of any elements of K as constants. In fact, the point is that even though F is defined with respect to the morphism $\epsilon \mapsto 0$, the intermediate results of the computation may not be so. Note that $\underline{L}(f) \leq L(f)$.

We remark that the assumption that any elements of K are free constants is made for achieving conceptual simplicity. We could as well require to build up the needed elements of K from ϵ, ϵ^{-1} and elements of k. One can show that this would not significantly change our main conclusions.

The topological characterization of approximative complexity, to be presented next, shows that this is a very natural notion from a mathematical point of view. Assume k to be an algebraically closed field. There is a natural way to put a Zariski topology on the polynomial ring $A_n := k[X_1, \ldots, X_n]$ as a limit of the Zariski topologies on the finite dimensional subspaces $\{f \in A_n \mid \deg f \leq d\}$ for $d \in \mathbb{N}$. If k is the field of complex numbers, we may define the Euclidean topology on A_n in a similar way. If $f \in A_n$ satisfies $\underline{L}(f) \leq r$, then it easy to see that f lies in the closure (Zariski or Euclidean) of the set $\{f \in A_n \mid L(f) \leq r\}$. Alder [2] has shown that the converse is true and obtained the following topological characterization of the approximative complexity.

Theorem 6.2. *The set $\{f \in A_n \mid \underline{L}(f) \leq r\}$ is the closure of the set $\{f \in A_n \mid L(f) \leq r\}$ for the Zariski topology. If $k = \mathbb{C}$, this is also true for the Euclidean topology.*

6.2 Computation of p-adic coefficients

Let $f, p \in k[X_1, \ldots, X_m][Y]$ and assume p to be monic of degree $d \geq 1$ in Y. Let $f = \sum_i f_i p^i$ be the p-adic representation of f, that is, $f_i \in k[X, Y]$ and $\deg_Y f_i < d$. Using the idea in [14, § 7.1] it is not hard to see that the complexity of the coefficient polynomial f_i of Y^i can be polynomially bounded in d, i, and $L(f)$. The following observation shows that the dependence on the degree i cannot be avoided in general.

Proposition 6.3. *The complexity of coefficient polynomials in a p-adic representation of a polynomial f is not polynomially bounded in $L(f)$ and $d = \deg p$, unless Valiant's hypothesis is false.*

Consider the Y-adic representation of the following polynomial f_n of complexity $L(f_n) = O(n^2)$

$$f_n := \prod_{i=1}^{n} \left(\sum_{j=1}^{n} X_{ij} Y^{2^{j-1}} \right) = \sum_i f_{n,i}(X) Y^i$$

and observe that the coefficient $f_{n,2^n-1}(X)$ equals the the permanent of the matrix $[X_{ij}]$. This already provides the proof of Proposition 6.3.

Assume now that the p-adic representation $f = f_\ell p^\ell + f_{\ell+1} p^{\ell+1} + \ldots$ starts at order ℓ, $f_\ell \neq 0$. We can consider f_ℓ as the leading coefficient of f with respect to the basis p. By contrast with Proposition 6.3, we can say the following about the approximative complexity of the leading coefficient in relation to the complexity of f (cf. [10]).

Proposition 6.4. *The approximative complexity of the leading coefficient f_ℓ is polynomially bounded in d and $\underline{L}(f)$: we have*

$$\underline{L}(f_\ell) = O(M(d)\underline{L}(f)).$$

7 Decision versus Computation

7.1 Approximative complexity of factors

Here is the result from [10], which eliminates the dependence on the multiplicity in Theorem 5.2 by switching to approximative complexity. The number $2 \leq \omega \leq 2.38$ denotes the exponent of matrix multiplication (cf. [14, Chap. 15]).

Theorem 7.1. *Assume that g is an irreducible factor of degree d of a polynomial $f \in \mathbb{R}[X_1, \ldots, X_n]$. We assume that the zeroset of g is a hypersurface in $\mathbb{R}^n$. Then we have for any $\epsilon > 0$ that*

$$\underline{L}(g) = O\big(M(d^4)\underline{L}(f) + d^{2\omega+\epsilon}M(d)\big).$$

A corresponding result holds over $\mathbb{C}$. Moreover, we remark that in the special situation, where g is the generator of the graph of a polynomial function, we obtain considerably better bounds, valid over any infinite field of characteristic zero.

The idea of the proof of Theorem 7.1 is as follows: After a suitable coordinate transformation, one can interpret the zeroset of the factor g locally around the origin as the graph of some analytic function φ. In order to cope with a possibly large multiplicity of g, we apply a small perturbation to the polynomial f without affecting its complexity too much. This results in a small perturbation of φ. We compute now the homogeneous parts of the perturbed φ by a Newton iteration up to a certain order. Using efficient polynomial arithmetic, this gives us an upper bound on the approximative complexity of the homogeneous parts of φ up to a predefined order. In the special case, where the factor g is the generator of the graph of a function, we are already done. Otherwise, we view the factor g as the minimal polynomial of φ in the variable $Y := X_n$ over the field $k(X_1, \ldots, X_{n-1})$. We show that the Taylor approximations up to order $2d^2$ uniquely determine the factor g and compute the bihomogeneous components of g with respect to the degrees in the X-variables and Y by fast linear algebra.

7.2 Applications to decision complexity

By combining Theorem 7.1 with Lemma 4.1 we immediately get the following result stating that the approximative complexity of a polynomial g can be bounded polynomially in the decision complexity and the degree of g.

Corollary 7.2. *Let g be the generator of an irreducible hypersurface in $\mathbb{R}^n$ or in $\mathbb{C}^n$, $d = \deg g$. Then we have for any $\epsilon > 0$ that*

$$\underline{L}(g) = O\big(M(d^4)C(g) + d^{2\omega+\epsilon}M(d)\big).$$

It is quite natural to incorporate the concept of approximative complexity into Valiant's algebraic framework of NP-completeness.

Definition 7.3. An *approximatively p-computable family* is a p-family (f_n) such that $\underline{L}(f_n)$ is a p-bounded function of n. The complexity class $\underline{\text{VP}}$ comprises all such families over a fixed field k.

It is obvious that $\text{VP} \subseteq \underline{\text{VP}}$. If the polynomial f is a projection of a polynomial g, then we clearly have $\underline{L}(f) \leq \underline{L}(g)$. Therefore, the complexity class $\underline{\text{VP}}$ is closed under p-projections. We remark that $\underline{\text{VP}}$ is also closed under the polynomial oracle reductions introduced in [11].

At the moment, we know very little about the relations between the complexity classes VP, $\underline{\text{VP}}$, and VNP.

Problem 7.4. 1. Is the class VP strictly contained $\underline{\text{VP}}$?
2. Is the class $\underline{\text{VP}}$ contained in VNP?

Since the class VP is closed under p-projections, the following strengthening of Valiant's hypothesis is equivalent to saying that VNP-complete families are not approximately p-computable.

Conjecture 7.5. The class VNP is not contained in the class $\underline{\text{VP}}$.

This conjecture should be compared with the known work on polynomial time deterministic or randomized approximation algorithms for the permanent of non-negative matrices [34, 3, 26]. Based on the Markov chain approach, Jerrum, Sinclair and Vigoda [26] have recently established a fully-polynomial randomized approximation scheme for computing the permanent of an arbitrary real matrix with non-negative entries. We note that this result does not contradict Conjecture 7.5 since the above mentioned algorithm works only for matrices with *non-negative* entries while approximative straight-line programs work on all real inputs.

Under the hypothesis $\text{VNP} \not\subseteq \underline{\text{VP}}$, we can can conclude that checking the values of polynomials forming VNP-complete families is hard, even when we allow randomized algorithms with two-sided error, formalized by randomized algebraic computation trees. This follows for deterministic computations easily from Corollary 7.2. The extension to randomized trees is straight-forward using [38, 15, 18].

Corollary 7.6. *Assume* $\text{VNP} \not\subseteq \underline{\text{VP}}$ *over* $\mathbb{R}$. *Then for any* VNP*-complete family* (g_n)*, checking the value* $y = g_n(x)$ *over the reals cannot be done by deterministic or randomized algebraic computation trees with a polynomial number of arithmetic operations and tests in* n.

By applying Corollary 7.2 to the permanent polynomial, we see that Conjecture 7.5 implies the following separation of complexity classes in the BSS-model of computation (cf. [8]).

Corollary 7.7. *If* $\text{VNP} \not\subseteq \underline{\text{VP}}$ *is true, then we have* $\text{P} \neq \text{PAR}$ *in the BSS-model over the reals.*

References

1. M. Aldaz, J. Heintz, G. Matera, J.L. Montaña, and L.M. Pardo. Time-space tradeoffs in algebraic complexity theory. *J. Compl.*, 16:2–49, 2000.
2. A. Alder. *Grenzrang und Grenzkomplexität aus algebraischer und topologischer Sicht.* PhD thesis, Zürich University, 1984.
3. A.I. Barvinok. Polynomial time algorithms to approximate permanents and mixed discriminants within a simply exponential factor. *Random Structures and Algorithms*, 14:29–61, 1999.
4. W. Baur. Simplified lower bounds for polynomials with algebraic coefficients. *J. Compl.*, 13:38–41, 1997.
5. W. Baur and K. Halupczok. On lower bounds for the complexity of polynomials and their multiples. *Comp. Compl.*, 8:309–315, 1999.
6. S. Berkowitz, C. Rackoff, S. Skyum, and L. Valiant. Fast parallel computation of polynomials using few processors. *SIAM J. Comp.*, 12:641–644, 1983.
7. L. Blum, F. Cucker, M. Shub, and S. Smale. Algebraic Settings for the Problem "$P \neq NP$?". In *The mathematics of numerical analysis*, number 32 in Lectures in Applied Mathematics, pages 125–144. Amer. Math. Soc., 1996.
8. L. Blum, F. Cucker, M. Shub, and S. Smale. *Complexity and Real Computation.* Springer, 1998.
9. L. Blum, M. Shub, and S. Smale. On a theory of computation and complexity over the real numbers. *Bull. Amer. Math. Soc.*, 21:1–46, 1989.
10. P. Bürgisser. The complexity of factors of multivariate polynomials. Preprint, University of Paderborn, 2001, submitted.
11. P. Bürgisser. On the structure of Valiant's complexity classes. *Discr. Math. Theoret. Comp. Sci.*, 3:73–94, 1999.
12. P. Bürgisser. *Completeness and Reduction in Algebraic Complexity Theory*, volume 7 of *Algorithms and Computation in Mathematics.* Springer Verlag, 2000.
13. P. Bürgisser. Cook's versus Valiant's hypothesis. *Theoret. Comp. Sci.*, 235:71–88, 2000.
14. P. Bürgisser, M. Clausen, and M.A. Shokrollahi. *Algebraic Complexity Theory*, volume 315 of *Grundlehren der mathematischen Wissenschaften.* Springer Verlag, 1997.
15. P. Bürgisser, M. Karpinski, and T. Lickteig. On randomized semialgebraic decision complexity. *J. Compl.*, 9:231–251, 1993.
16. P. Bürgisser, T. Lickteig, and M. Shub. Test complexity of generic polynomials. *J. Compl.*, 8:203–215, 1992.
17. D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *J. Symb. Comp.*, 9:251–280, 1990.
18. F. Cucker, M. Karpinski, P. Koiran, T. Lickteig, and K. Werther. On real Turing machines that toss coins. In *Proc. 27th ACM STOC, Las Vegas*, pages 335–342, 1995.
19. H. Fournier and P. Koiran. Are lower bounds easier over the reals? In *Proc. 30th ACM STOC*, pages 507–513, 1998.
20. H. Fournier and P. Koiran. Lower bounds are not easier over the reals: Inside PH. In *Proc. ICALP 2000*, LNCS 1853, pages 832–843, 2000.
21. J. von zur Gathen. Feasible arithmetic computations: Valiant's hypothesis. *J. Symb. Comp.*, 4:137–172, 1987.
22. B. Griesser. Lower bounds for the approximative complexity. *Theoret. Comp. Sci.*, 46:329–338, 1986.

23. D.Yu. Grigoriev and M. Karpinski. Randomized quadratic lower bound for knapsack. In *Proc. 29th ACM STOC*, pages 76–85, 1997.
24. J. Grollmann and A.L. Selman. Complexity measures for public-key cryptosystems. *SIAM J. Comp.*, 17(2):309–335, 1988.
25. J. Heintz and J. Morgenstern. On the intrinsic complexity of elimination theory. *Journal of Complexity*, 9:471–498, 1993.
26. M.R. Jerrum, A. Sinclair, and E. Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries. *Electronic Colloquium on Computational Complexity*, 2000. Report No. 79.
27. E. Kaltofen. Single-factor Hensel lifting and its application to the straight-line complexity of certain polynomials. In *Proc. 19th ACM STOC*, pages 443–452, 1986.
28. E. Kaltofen. Factorization of polynomials given by straight-line programs. In S. Micali, editor, *Randomness and Computation*, pages 375–412. JAI Press, Greenwich CT, 1989.
29. P. Koiran. Computing over the reals with addition and order. *Theoret. Comp. Sci.*, 133:35–47, 1994.
30. P. Koiran. A weak version of the Blum, Shub & Smale model. *J. Comp. Syst. Sci.*, 54:177–189, 1997.
31. P. Koiran. Circuits versus trees in algebraic complexity. In *Proc. STACS 2000*, number 1770 in LNCS, pages 35–52. Springer Verlag, 2000.
32. S. Lang. *Fundamentals of Diophantine Geometry*. Springer Verlag, 1983.
33. T. Lickteig. On semialgebraic decision complexity. Technical Report TR-90-052, Int. Comp. Sc. Inst., Berkeley, 1990. Habilitationsschrift, Universität Tübingen.
34. N. Linial, A. Samorodnitsky, and A. Wigderson. A deterministic polynomial algorithm for matrix scaling and approximate permanents. In *Proc. 30th ACM STOC*, pages 644–652, 1998.
35. R.J. Lipton and L.J. Stockmeyer. Evaluation of polynomials with superpreconditioning. *J. Comp. Syst. Sci.*, 16:124–139, 1978.
36. S. Meiser. Point location in arrangements of hyperplanes. *Information and Computation*, 106:286–303, 1993.
37. F. Meyer auf der Heide. A polynomial linear search algorithm for the n-dimensional knapsack problem. *J. ACM*, 31:668–676, 1984.
38. F. Meyer auf der Heide. Simulating probabilistic by deterministic algebraic computation trees. *Theoret. Comp. Sci.*, 41:325–330, 1985.
39. F. Meyer auf der Heide. Fast algorithms for n-dimensional restrictions of hard problems. *J. ACM*, 35:740–747, 1988.
40. B. Poizat. *Les Petits Cailloux*. Number 3 in Nur Al-Mantiq War-Ma'rifah. Aléas, Lyon, 1995.
41. A.L. Selman. A survey of one-way functions in complexity theory. *Math. Systems Theory*, 25:203–221, 1992.
42. M. Shub and S. Smale. On the intractability of Hilbert's Nullstellensatz and an algebraic version of "NP $\neq$ P?". *Duke Math. J.*, 81:47–54, 1995.
43. V. Strassen. Vermeidung von Divisionen. *Crelles J. Reine Angew. Math.*, 264:184–202, 1973.
44. V. Strassen. Einige Resultate über Berechnungskomplexität. *Jahr. Deutsch. Math. Ver.*, 78:1–8, 1976.
45. L.G. Valiant. Completeness classes in algebra. In *Proc. 11th ACM STOC*, pages 249–261, 1979.
46. L.G. Valiant. The complexity of computing the permanent. *Theoret. Comp. Sci.*, 8:189–201, 1979.

47. L.G. Valiant. Reducibility by algebraic projections. In *Logic and Algorithmic: an International Symposium held in honor of Ernst Specker*, volume 30, pages 365–380. Monogr. No. 30 de l'Enseign. Math., 1982.