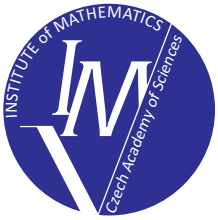

Teze disertace
k získání vědeckého titulu "doktor věd"
ve skupině věd fyzikálně-matematických

Complexity of Algebraic Computations

Komise pro obhajoby doktorských disertací v oboru
Matematické struktury

Jméno uchazeče: Pavel Hrubeš

Pracoviště uchazeče: Matematický ústav AVČR

Místo a datum: Praha, 20. srpna 2024

I would like to thank my scientific advisor Pavel Pudlák for introducing me to the world of science.

I will always be grateful for having the opportunity to learn from my colleagues and friends Avi Wigderson, Amir Yehudayoff, Amir Shpilka and Anup Rao.

Several results in this thesis were obtained in collaboration with A. Wigderson and A. Yehudayoff. Many of them were obtained at the Institute of Mathematics of the Czech Academy of Sciences which provided support and encouragement in my research.

Pavel Hrubeš

Contents

Abstract **3**

1 Introduction **5**

2 Complexity of arithmetic circuits **10**
2.1 Structural results 10
2.2 τ-conjectures 13
2.3 Non-commutative computations 16
2.4 The sum-of-squares problem 19

3 Future directions **22**

Publications covered by the dissertation **25**

References **27**

Abstract

Author's contribution to the mathematical area of *arithmetic circuit complexity* is described.

The area studies the complexity of computing polynomials over a field. Its objective is to find a dividing line between polynomials which can be computed efficiently as opposed to these which are hard to compute. The central open problem in the field is to present an explicit polynomial which is hard to compute.

The thesis consists of twelve selected publications of the author on this subject. They are divided into four categories. In Section 2.1, we present some structural results on arithmetic computations including a connection between monotone and general computations. Section 2.2 investigates τ-conjectures which connect hardness of computation with more tractable complexity measures. In Section 2.3, we present results on non-commutative compu-

tations. In Section 2.4, non-commutative computation is related to a classical problem of Hurwitz and some results on the latter problem are presented.

1. *Introduction*

Mathematics draws an abstract line between existence and non-existence. It shows which objects can in principle exist and what are their possible relations. There is a Platonic solid with eight faces but none with sixteen; every natural number is a sum of four squares whereas seven is not a sum of three squares. Mathematics is also a discipline invented by our finite and fallible minds. Hence it has always favored constructive or algorithmic solutions. The five Platonic solids can be built in physical space; a four-square representation of a number can be found using pen and paper.

The notion of an algorithm has received an exact mathematical formulation in the field of *computability theory*. Its generic goal is to find the line between problems which are algorithmically solvable and those that are not. This question has been further refined by the young field of *com-*

plexity theory which asks which problems can be solved by an algorithm *efficiently*, within reasonable time or memory constraints. This refinement has been largely motivated by an attempt to understand limitations of physical computers. There is also a philosophical undertone: since the human brain is limited, it can be viewed as an investigation of inherent limitations of the human mind. However, complexity theory is a mathematical discipline with exactly formulated theorems and problems. The field is especially rich in its famous *unsolved* problems – they capture our imagination, challenge us to discover new mathematical methods, and motivate development of methods in other areas of mathematics.

Computation of polynomials

In this thesis, we focus on complexity of computation of polynomials. Given a multivariate polynomial with coefficients from a field, it can always be computed from variables and constants by applying the operations addition and multiplication. The question is how many of these operations are needed.

The standard computational model is that of an *arithmetic circuit*: starting from variables or constants, the circuit computes new polynomials by means of addition

and multiplication operations. The model is not realistic in the sense that numbers can be added/multiplied at a unit cost. It nevertheless provides a clean mathematical paradigm for the study of algebraic computations. Several algorithms of practical interest can be phrased in this language. A prominent example is an efficient computation of the determinant of a matrix. This can in turn be used in many algorithms based on linear algebra. Other examples include fast matrix multiplication or fast Fourier transform (see, e.g., [BCS97]).

The major open question in the field is to prove a super-polynomial lower bound against arithmetic circuits. Namely, we want to find an explicit n-variate polynomial of degree $d \leq n^{O(1)}$ which requires an arithmetic circuit of a super-polynomial size in n. This is known as the VP vs VNP problem and is the algebraic analogue of the famed P vs. NP question. Despite decades of work, the best lower bound for circuits computing an explicit n-variate polynomial of degree d is $\Omega(n \log d)$, due to Baur and Strassen [Str73b, BS83]. Better lower bounds are known for a variety of more restricted computational models, such as monotone circuits, multilinear or bounded depth circuits (see, e.g., [JS82, Raz04]).

In algebraic complexity theory, two polynomials are of central interest: the determinant and the permanent of a

square matrix[1]

$$\det_n(X) = \sum_\sigma \operatorname{sgn}(\sigma) \prod_{i=1}^{n} x_{i,\sigma(i)}\,,\ \operatorname{perm}_n(X) = \sum_\sigma \prod_{i=1}^{n} x_{i,\sigma(i)}\,.$$

Despite the similarity of these expressions, the two polynomials stand on opposite sides of the complexity landscape. In [Val79], Valiant defined algebraic analogues of complexity classes P and NP, which we now call VP and VNP, and showed that the permanent polynomial is complete for the class VNP (if the underlying field is of characteristic different from two). This means that proving VP $\neq$ VNP is equivalent to proving super-polynomial circuit lower bounds for the permanent polynomial. In contrast, the determinant is efficiently computable by a polynomial size arithmetic circuit, and hence lies in the class VP. This allows to rephrase the VP vs. VNP question as a clean mathematical problem which does not refer to computations at all: how large an m is needed so that the permanent polynomial perm_n can be expressed as the determinant of an $m \times m$ matrix with affine functions as entries? This formulation seems open to techniques from algebraic geometry and representation theory and has lead to developement of *geometric complexity theory* [Lan13].

[1] Here, σ ranges over permutations of $\{1, \ldots, n\}$ and $\operatorname{sgn}(\sigma) \in \{1, -1\}$ is the sign of σ.

An arithmetic circuit is an algebraic counterpart of a *Boolean circuit*. A Boolean circuit computes a Boolean function by means of elementary operation $\wedge, \vee, \neg$. In complexity theory, the latter plays a more fundamental role: a super-polynomial lower bound on Boolean circuit size for a function in NP would imply P $\neq$ NP. This explains one motivation for the study of arithmetic lower bounds. An arithmetic circuit can be viewed as a restricted version of a Boolean circuit. Hence, hardness results in the arithmetic setting are believed to be easier to obtain than in the Boolean setting, and VP vs. VNP can be seen as a toy version of P vs. NP.

Arithmetic computations display a rich structure in their own right. For example, [VSBR83] shows that arithmetic computations are efficiently parallelizable and [BS83] shows that there is no quantitative difference between computing a single polynomial and a set of polynomials. These results have no known counterpart in the Boolean setting. The essence of algebraic computations naturally invites tools from very different mathematical disciplines such as algebraic geometry, analysis or topology. In turn, questions in complexity theory motivate development of these fields, and this interplay creates a rich soil in which answers will blossom one day.

2. *Complexity of arithmetic circuits*

2.1 Structural results

Arithmetic circuit lower bounds can be obtained if other restrictions on their computational power are imposed. Most notably, a *monotone* circuit is an arithmetic circuit over the reals involving only non-negative constants. Exponential lower bounds for monotone circuits computing the permanent (and other monotone polynomials) have been obtained in [JS82, Val80]. There it was also shown that monotone computations are exponentially weaker than general ones. These results are similar to monotone lower bounds in Boolean complexity [Raz85, AB87]. Let us remark that the Boolean results are significantly harder to prove.

The relative ease with which monotone arithmetic lower bounds can be proved creates an illusion that monotone circuits are understood completely. A result of the author shows that this is by far not the case.

- In [Hru20a], it has been shown that a *sufficiently strong* lower bound on monotone arithmetic circuits implies lower bounds in the unrestricted setting. More precisely, if a polynomial $f \in \mathbb{R}[x_1, \ldots, x_n]$ of degree d has an arithmetic circuit of size s then $(x_1 + \cdots + x_n + 1)^d + \epsilon f$ has a monotone arithmetic circuit of size $O(sd^2 + n \log n)$, for some $\epsilon > 0$.

Hence, a strengthening of current monotone lower bound techniques can in principle resolve the VP vs VNP problem. [Hru20a] also contains results pertaining to Boolean circuit complexity: the task of proving Boolean circuit lower bounds is related to bounding the *non-negative rank* of a certain explicit matrix.

A classical result [VSBR83] shows that arithmetic computations are efficiently parallelizable: an arithmetic circuit of size s computing a polynomial of degree d can be converted to a circuit of depth $O(\log d(\log s + \log d))$. This is one of the key aspects of arithmetic computations that have no Boolean counterpart. It also implies that arithmetic circuits computing low-degree polynomials can by quasi-polynomially simulated by arithmetic *formulas* –

a model capturing parallel computations. Whether this quasi-polynomial simulation can be improved to a polynomial one is an open problem. It is unresolved because it is not known how to prove lower bounds on arithmetic formula size.

In the world of monotone computations, tightness of [VSBR83] and similar transformations can often be proved be exhibiting separations between corresponding models [SS79].

- [HY16] (with A. Yehudayoff) investigates *algebraic branching programs* – a model capturing complexity of computations based on linear algebra. It is shown that the transformation of [VSBR83] is tight in this setting by giving a super-polynomial separation between monotone circuits and branching programs.

A combinatorial contribution of the paper consists in a characterisation of expansion properties of finite binary trees.

An arithmetic circuit computes a polynomial over a fixed underlying field $\mathbb{F}$. How does the underlying field affect the computational power? For example, the VP vs. VNP problem is not a single problem but rather a multitude of questions depending on the choice of $\mathbb{F}$. It is in principle possible, though believed unlikely, that the question would have a different answer over different fields.

A field has two main properties: size and characteristic. The influence of field size on arithmetic computations was investigated in [HY11]. The characteristic of $\mathbb{F}$ is a more influential parameter. Many classical polynomial identities which are used in complexity theory hold in characteristic zero only – Newton's identities are one such example. Among positive characteristics, characteristic 2 seems to behave differently than the others – for example, the proof of VNP-completeness of permanent requires characteristic different from 2.

- In [Hru16b] new VNP-complete families in characteristic 2 were designed. They correspond to polynomial-time decidable problems, a phenomenon previously encountered only in characteristic $\neq 2$.

This indicates that characteristic 2 may not be that special after all.

2.2 τ-conjectures

A specific approach towards arithmetic lower bounds comes from *τ-conjecture* of Blum and Shub [SS95]. The conjecture asserts that any univariate polynomial which is easy to compute can have only a small number of integer roots. The conjecture was originally designed to imply that P is

different from NP in Blum-Shub-Smale model of computation [BSS89]. This is a model of Turing-style computations over $\mathbb{R}$ where elementary arithmetic operations can be performed at a unit cost.

It was later shown in [Bür09] that τ-conjecture implies exponential arithmetic circuit lower bounds and hence[1] VP$\neq$ VNP. One drawback of the conjecture is that, by referring to *integer* roots, it leads one to the area of number theory which is notorious for its hard problems.

However, the conjecture has several modifications, relating computational complexity with the number of *real* roots, or with a geometric structure of the Newton polytope of a polynomial [Koi11, KPTT15]. They take an inspiration from Descartes' rule of signs which implies that a real univariate polynomial f with k non-zero coefficients can have at most $(k-1)$ positive roots. This holds regardless of the degree of f and suggests that in the real setting, sparsity of a polynomial has a role similar to the notion of degree in the complex setting. Indeed, this analogy was developed by Khovanskii in his general theory of fewnomials [Kho91]. The τ-conjecture of Koiran [Koi11] can be seen as an extension of Descartes' rule to a class of polynomials with a simple arithmetic structure.

[1] This applies in the so-called *constant-free* setting which we will not discuss here.

Any one of these conjectures, if true, is known to imply exponential arithmetic lower bounds. Their main appeal is their mathematical clarity. Futhermore, they relate the mysterious notion of complexity with more tractable measures.

- In [Hru13], it was shown that instead of studying real roots of a polynomial, it is enough to study the distribution of arguments of its complex roots on the complex plane.

This is important because complex zeros are easier to determine than real zeros (which are in turn easier to handle than integer zeros). Furthermore, [Hru13] relates the problem with other topics such as Hurwitz stable polynomials or equidistribution of sequences or roots of Erdös and Turán [ET50].

- [HY23] (with A. Yehudayoff) investigates connections between arithmetic complexity of a polynomial and geometric complexity of its Newton polytope. It is shown that some versions of τ-conjecture hold in the monotone setting while some generalizations do not.

The paper also relates the topic with a specific estimate on the number of vertices of a 2-dimensional shadow of the Birkhoff polytope (cf. Chapter 3, Problem 2).

- The paper [Hru20b] relates the distribution of roots of a univariate polynomial with the number of vertices of the Newton polytope of a related bivariate polynomial. This implies a connection between different versions of the τ-conjecture.

[Hru20b] is mainly a self-contained result in pure mathematics which gives quantitative estimates on distribution of zeros of univariate polynomials.

It is possible that these results will be ultimately extended to a *counterexample* to some versions of τ-conjecture. This will, at the same time, allow us to focus on the correct notion of geometric complexity which may eventually lead to circuit lower bounds.

2.3 Non-commutative computations

A restricted class of arithmetic circuits are *non-commutative* circuits, where multiplication does not commute. Starting with seminal works of Hyafil [Hya79] and Nisan [Nis91], non-commutative circuits are a well-studied object. The lack of commutativity is a severe limitation on its computational power which makes the task of proving circuit lower bounds apparently easier. In [Nis91], an exponential lower bound on non-commutative arithmetic *formulas*

was presented. Nevertheless, proving lower bounds in the setting of non-commutative circuits is still open.

- [HY13] (with A. Yehudayoff) gave a strong separation between formulas and monotone circuits in the non-commutative setting, answering an open problem from [Nis91].

The arithmetic circuit model can be easily modified to encompass computation of *rational functions* instead of polynomials. This can be achieved by allowing a division or an inverse gate as an extra operation. In the commutative setting, this extension is merely cosmetic. A computation of a rational function f can be viewed as a computation of a pair of polynomials g, h with $f = gh^{-1}$. Moreover, a classical result of Strassen [Str73a] shows that inverse gates do not help to cumpute a polynomial: if a polynomial has a small circuit with inverse gates, it also has a small circuit without them.

Non-commutative rational functions display a richer structure giving rise to new phenomena. While every commutative rational function can be expressed using one inverse gate only, non-commutatively an arbitrary number of inverses may be necessary. Moreover, inverse operations may be *nested* as in $(x + ux^{-1}v)^{-1}$. Non-commutative rational functions form a free skew field. A beautiful theory of these skew fields has been developed by P. M. Cohn

[Coh95] and others. They are equipped with invariants not present in the standard commutative field of fractions.

- In [HW15], the author (with A. Wigderson) initiated the study of non-commutative arithmetic computations with inversions. A basic theory of these computations was developed. It was shown that the inverse of a matrix consisting of n^2 non-commuting variables can be computed by polynomial size circuit with inversions, whereas it requires an exponential size formula. It was also shown that matrix inverse displays similar completeness properties as the commutative determinant.

In [HW15], the following question emerged: given a matrix M with affine functions as entries, can we give an efficient algorithm for testing whether M is invertible over the skew field of fractions? This question lead to beautiful subsequent results [GGdOW20, IQS18] which give a polynomial-time algorithm for testing invertibility of M. (Over the *commutative* field of fractions, this is still an open problem).

2.4 The sum-of-squares problem

In [HWY11], the problem of non-commutative lower bounds was related to a classical problem of Hurwitz on sum-of-squares composition formulas.

The problem of Hurwitz [Hur98] asks for which integers n, m, k does there exist a sum-of-squares identity

$$(x_1^2 + \cdots + x_n^2)(y_1^2 + \cdots + y_m^2) = f_1^2 + \cdots + f_k^2 \,,$$

where $f_1, \ldots, f_k$ are bilinear forms in x and y with complex coefficients. Focusing on the case $m = n$, the question is how large a $k = k(n)$ is needed in terms of n. A seminal theorem of Hurwitz asserts that k must be strictly larger than n except for the special cases $n \in \{1, 2, 4, 8\}$. Note that for $n = 2$, $k = 2$ is achieved by the identity

$$(x_1^2 + x_2^2)(y_1^2 + y_2^2) = (x_1 y_1 - x_2 y_2)^2 + (x_1 y_2 + x_2 y_1)^2$$

which can be interpreted as multiplicativity of the norm on complex numbers. Similarly, the case of $n = 4$ is solved by Euler's identity which is related to multiplicativity of the norm on quaternions.

The asymptotic dependence of k on n in Hurwitz's problem is not known. An elementary argument gives $n \leq k(n) \leq n^2$. Owing to Hurwitz's theorem, the first inequality is strict with the exception of $n \in \{1, 2, 4, 8\}$. The

upper bound can be slightly improved to $O(n^2/\log n)$ using another classical result of Hurwitz and Radon [Hur23]. An intriguing question is whether k grows as a super-linear function of n.

- [HWY11] (with Wigderson and Yehudayoff) asserts that if k grows sufficiently fast in terms of n in Hurwitz's problem (namely, as $\Omega(n^{1+\epsilon})$ for some $\epsilon > 0$) then we obtain exponential circuit lower bounds in the non-commutative setting.

In other words, a solution to the classical mathematical problem of Hurwitz can provide computational hardness results.

In subsequent works, the author made several attempts to resolve Hurwitz's problem itself.

- In [HWY13] (with Wigderson and Yehudayoff) such a solution was provided under the assumption that f_i have integer coefficients. A lower bound of $\Omega(n^{6/5})$ is proved under this assumption.

- In [Hru16a], the problem was reduced to a problem about rank of matrices in a family of pairwise anti-commuting matrices.

- In [Hru24], a new construction of sum-of-squares composition formulas was presented. It gives $k \leq O(n^{1.62})$

which is the first asymptotically truly sub-quadratic construction to have been obtained.

Classical constructions of sum-of-squares identities involve integer coefficients and it is unclear whether using real or complex coefficients can give any advantage. This gives hope that the lower bound from [HWY13] can be extended to the case of general coefficients as well. On the other hand, [Hru24] goes in the opposite direction. Over complex numbers, it gives a more efficient construction than previously believed possible.

3. *Future directions*

Major open problems in arithmetic circuit complexity remain open. In this thesis, we identified some viable approaches towards their solution which should be further pursued.

The first one is to analyze geometrical properties of the Newton polytope of multivariate polynomials, or the structure of real and complex roots of univariate polynomials. The goal is to obtain geometric measures which guarantee hardness of computation. Results from Section 2.2 focused on pre-existing forms of τ-conjecture of Koiran et al. Other variants can be imagined – for example, to relate computational complexity of a polynomial with the magnitude of its derivative. There is no convincing reason why any of these conjectures should be true. It is however quite possible that their study will eventually lead to a discovery of the correct complexity measure which guarantees

hardness of computation.

The second one is the sum-of-squares problem of Hurwitz from Section 2.4. This is a clean mathematical problem with a long history whose solution may eventually present itself. The recent result [Hru24] indicates that the problem is more complicated than originally hoped. On the hand, as pointed out in [HWY11], the problem of Hurwitz can be modified in several ways to make it easier, while at the same time retaining its applications to complexity theory.

As is common, many new open problems were encountered in this research. We include some specific open questions.

Specific open problems

1. [HY23] The Birkhoff polytope $\mathrm{DS}_n \subseteq \mathbb{R}^{n\times n}$ is the set of $n \times n$ doubly-stochastic matrices. Let $m(n)$ be the largest m such that there exists a linear map $L : \mathbb{R}^{n\times n} \to \mathbb{R}^2$ such that the polytope $L(\mathrm{D}S_n)$ has m vertices. Does $m(n)$ grow exponentially with n?

2. [HY23] Find an explicit monotone polynomial f (with polynomially many variables and of a polynomial degree) such that g requires a super-polynomial monotone arithmetic circuit whenever $g \neq 0$ and f divides

g.

3. [Hru24] Prove a truly sub-quadratic *upper* bound on Hurwitz's problem over the real numbers $\mathbb{R}$.

4. [HWY13] Prove a super-linear *lower* bound on Hurwitz's problem over Gaussian integers $\mathbb{Z}[i]$.

5. [Hru20a] Show that $(\prod_{j=1}^{n} \sum_{i=1}^{n} x_{i,j} - \mathrm{perm}_n)$ requires a monotone arithmetic circuit of super-polynomial size. How about $(\prod_{j=1}^{n} \sum_{i=1}^{n} x_{i,j} + \mathrm{perm}_n)$?

6. [HW15] Assume that a non-commutative polynomial f can be computed by a non-commutative circuit with inverse gates of size s. Give a non-trivial upper bound on the size of a circuit *without* inverses computing f.

Publications covered by the dissertation

[Hru13] P. Hrubeš. On the real τ-conjecture and the distribution of complex roots. *Theory of Computing*, 9(10):403–411, 2013.

[Hru16a] P. Hrubeš. On families of anticommuting matrices. *Linear Algebra and Applications*, 493:494–507, 2016.

[Hru16b] P. Hrubeš. On hardness of multilinearization and VNP-completeness in characteristics two. *ACM Transactions on Computation Theory*, 9(1), 2016.

[Hru20a] P. Hrubeš. On ϵ-sensitive monotone computations. *Computational Complexity*, 29(2), 2020.

[Hru20b] P. Hrubeš. On the distribution of runners on a circle. *European Journal of Combinatorics*, 89, 2020.

[Hru24] P. Hrubeš. A subquadratic upper bound on sum-of-squares composition formulas. In *Computational Complexity Conference*, 2024.

[HW15] P. Hrubeš and A. Wigderson. Non-commutative arithmetic circuits with division. *Theory of Computing*, 11:357–393, 2015.

[HWY11] P. Hrubeš, A. Wigderson, and A. Yehudayoff. Non-commutative circuits and the sum of squares problem. *J. Amer. Math. Soc.*, 24:871–898, 2011.

[HWY13] P. Hrubeš, A. Wigderson, and A. Yehudayoff. An asymptotic bound on the composition number of integer sums of squares formulas. *Canadian Mathematical Bulletin*, 56:70–79, 2013.

[HY13] P. Hrubeš and A. Yehudayoff. Formulas are exponentially stronger than monotone circuits in non-commutative setting. In *Conference on Computational Complexity*, 2013.

[HY16] P. Hrubeš and A. Yehudayoff. On isoperimetric profiles and computational complexity. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP*, 2016.

[HY23] P. Hrubeš and A. Yehudayoff. Shadows of Newton polytopes. *Israel Journal of Mathematics*, 256:311–343, 2023.

References

[AB87] N. Alon and R. B. Boppana. The monotone circuit complexity of Boolean functions. *Combinatorica*, 7(1):1–22, 1987.

[BCS97] P. Bürgisser, M. Clausen, and M. A. Shokrollahi. *Algebraic complexity theory*, volume 315 of *A series of comprehensive studies in mathematics*. Springer, 1997.

[BS83] W. Baur and V. Strassen. The complexity of partial derivatives. *Theoretical Computer Science*, 22:317–330, 1983.

[BSS89] 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.

[Bür09] P. Bürgisser. On defining integers and proving arithmetic circuit lower bounds. *Computational Complexity*, 18(1):81–103, 2009.

[Coh95] P. M. Cohn. *Skew Fields*, volume 57 of *Encyclopedia of Mathematics*. Cambridge University Press, 1995.

[ET50] P. Erdös and P. Turán. On the distribution of roots of polynomials. *Annals of Mathematics*, 51:105–119, 1950.

[GGdOW20] A. Garga, L. Gurvits, R. de Oliveira, and A. Wigderson. Operator scaling: theory and applications. *Fund. Comput. Math*, 20:223–290, 2020.

[Hur98] A. Hurwitz. Über die Komposition der quadratischen Formen von beliebigvielen Variabeln. *Nach. Ges. der Wiss. Göttingen*, pages 309–316, 1898.

[Hur23] A. Hurwitz. Über die Komposition der quadratischen Formen. *Math. Ann.*, 88:1–25, 1923.

[HY11] P. Hrubeš and A. Yehudayoff. Arithmetic complexity in ring extensions. *Theory of Computing*, 7:119–129, 2011.

[Hya79] L. Hyafil. On the parallel evaluation of multivariate polynomials. *SIAM J. Comput.*, 8(2):120–123, 1979.

[IQS18] G. Ivanyos, Y. Qiao, and K. V. Subramanyam. Constructive non-commutative

rank computation is in deterministic polynomial time. *Computational Complexity*, 27:223–290, 2018.

[JS82] M. Jerrum and M. Snir. Some exact complexity results for straight-line computations over semirings. *Journal of the ACM*, 1982.

[Kho91] A. G. Khovanskii. *Fewnomials*, volume 88 of *Translations of Mathematical Monographs*. American Mathematical Society, 1991.

[Koi11] P. Koiran. Shallow circuits with high-powered inputs. In *Symposium on Innovations in Computer Science*. Tsingua University Press, Beijing, 2011.

[KPTT15] P. Koiran, N. Portier, S. Tavenas, and S. Thomassé. A τ-conjecture for Newton polygons. *Foundations of computational mathematics*, 15(1):187–197, 2015.

[Lan13] J. M. Landsberg. Geometric complexity theory: an introduction for geometers, 2013.

[Nis91] N. Nisan. Lower bounds for non-commutative computation. In *Proceeding of the 23th STOC*, pages 410–418, 1991.

[Raz85] A.A. Razborov. Lower bounds on the monotone complexity of some boolean functions. *Soviet Mathematics Doklady*, 31:354–357, 1985.

[Raz04] R. Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. In *Proceeding of the 36th STOC*, pages 633–641, 2004.

[SS79] E. Shamir and M. Snir. On the depth complexity of formulas. *Journal Theory of Computing Systems*, 13(1):301–322, 1979.

[SS95] M. Schub and S. Smale. On the intractability of Hilbert's nullstellensatz and an algebraic version of P=NP. *Duke Mathematical Journal*, 81(1):47–54, 1995.

[Str73a] V. Strassen. Vermeidung von divisionen. *J. of Reine Angew. Math.*, 264:182–202, 1973.

[Str73b] Volker Strassen. Die Berechnungskomplexität von elementarsymmetrischen Funktionen und von Interpolationskoeffizienten. *Numerische Mathematik*, 20(3):238–251, 1973.

[Val79] L. G. Valiant. Completeness classes in algebra. In *STOC*, pages 249–261, 1979.

[Val80] L. G. Valiant. Negation can be exponentially powerful. *Theoretical Computer Science*, 12:303–314, 1980.

[VSBR83] L. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast parallel computation of polynomials using few processors. *Siam J. Comp.*, 12:641–644, 1983.