Completeness and Reduction in Algebraic Complexity Theory

Peter Bürgisser
Dedicated to
Brigitte and Ladina
Preface

One of the most important and successful theories in computational complexity is that of NP-completeness. This discrete theory is based on the Turing machine model and achieves a classification of discrete computational problems according to their algorithmic difficulty. Turing machines formalize algorithms which operate on finite strings of symbols over a finite alphabet. By contrast, in algebraic models of computation, the basic computational step is an arithmetic operation (or comparison) of elements of a fixed field, for instance of real numbers. Hereby one assumes exact arithmetic. In 1989, Blum, Shub, and Smale [12] combined existing algebraic models of computation with the concept of uniformity and developed a theory of NP-completeness over the reals (BSS-model). Their paper created a renewed interest in the field of algebraic complexity and initiated new research directions. The ultimate goal of the BSS-model (and its future extensions) is to unite classical discrete complexity theory with numerical analysis and thus to provide a deeper foundation of scientific computation (cf. [11, 101]).

Already ten years before the BSS-paper, Valiant [107, 110] had proposed an analogue of the theory of NP-completeness in an entirely algebraic framework, in connection with his famous hardness result for the permanent [108]. While the part of his theory based on the Turing approach (\#P-completeness) is now standard and well-known among the theoretical computer science community, his algebraic completeness result for the permanents received much less attention. The first account of Valiant’s algebraic theory with elaborated proofs was von zur Gathen’s survey [41]. A more recent treatment, with different proofs, can be found in the last chapter of the book [21] by Bürgisser et al.

In this research monograph, we further develop Valiant’s approach, and clarify its connections both to the discrete and to the BSS-model. We think this adds considerably to our understanding of the concept of completeness in algebraic models of computation.

This book is the author’s Habilitationsschrift in mathematics at the University of Zurich. It is organized as follows. The introduction overviews the three known theories of NP-completeness and explains our main results in an informal way. After a detailed treatment of Valiant’s model in Chap. 2, we proceed in Chap. 3 by showing that the generating functions of various NP (or \#P) complete graph properties are complete in Valiant’s sense. The proofs are mainly based on graph theoretical constructions.

In Chap. 4, we relate Valiant’s model to the classical discrete theory. Unexpectedly, parallel complexity classes enter. We rely on techniques from algebraic geometry and number theory, and our main result hinges on the generalized Riemann hypothesis.

The next chapter is devoted to investigations in the spirit of structural complexity. We prove that any countable poset can embedded in the poset of \( p \)-degrees, if Valiant’s hypothesis is true. A striking result here is the discovery
of a specific family of polynomials (cut enumerators), which is neither complete nor \( p \)-computable, provided the polynomial hierarchy does not collapse.

In Chap. 6 we deviate a little from our main line of investigation. We develop a fast algorithm to evaluate irreducible rational matrix representations of complex general linear groups with respect to a symmetry adapted basis (Gelfand-Tsetlin basis). The connection to our main topic then becomes clear in the next chapter.

Immanants are matrix functions defined in terms of characters of the symmetric group, which generalize the permanent and determinant. For a deeper understanding of the amazingly different complexity behaviour of determinants and permanents, they are a natural object to study. Moreover, they give also an indication of the complexity to evaluate single entries of invariant matrices of general linear groups. Our algorithm of Chap. 6 yields upper complexity bounds for immanants, which improve previous bounds due to Hartmann [47] and Barvinok [5]. The main efforts of Chap. 7 consist of proofs that the evaluation of immanants corresponding to certain hook diagrams or rectangular diagrams are complete in Valiant’s sense.

Finally, Chap. 8 contains some separation results and establishes a connection between Valiant’s and the Blum-Shub-Smale model. We discuss possible directions for a deeper understanding of this connection.

Seventeen conjectures and open problems, distributed throughout the text, suggest further research. Some of them have resisted serious attempts for solution by the author.

Acknowledgments

I owe special thanks to Michael Claeysen for his encouragement and feedback throughout this project. I am indebted to Amin Shokrollahi for pointing out to me the paper [5] by Barvinok, and I would like to thank Luis Miguel Pardo for indicating that Koiran’s method [67] can be improved using the results of [68, 69]. I thank Lenore Blum, Felipe Cucker, Maurice Rojas, and Steve Smale for their interest and numerous discussions about my work.

I am grateful to the Institute for Applied Mathematics of the University of Zürich for making this research possible. In particular, I would like to thank Andrew Barbour and Peter Gabriel for their support. I am greatly indebted to Steve Smale for inviting me to the Liu Bie Ju Centre for Mathematical Sciences at the City University of Hong Kong, where this work was completed.

I would like to thank the staff at Springer Verlag Heidelberg for editorial advise and help. I am grateful to Fred and Lilly Hämmerli for their help in the final stage of this project. Finally, I wish to thank my wife Brigitte for her support, patience, and understanding, and our daughter Ladina for her cheerful presence.

Zürich, August 1998

Peter Bürgisser
Contents

Index

1 Introduction 1
  1.1 Classical Model .............................................. 1
  1.2 Blum-Shub-Smale Model ................................. 3
  1.3 Valiant’s Model .............................................. 4
  1.4 Overview of Main Results ............................... 6

2 Valiant’s Algebraic Model of NP-Completeness 11
  2.1 The Complexity Classes VP and VNP ...................... 11
  2.2 Completeness of the Permanent Family ................. 16
    2.2.1 p-Definability and Formula Size .................... 16
    2.2.2 Universality of the Permanent ...................... 19
    2.2.3 Completeness of the Permanent ...................... 21
  2.3 Closure Properties ......................................... 25
  2.4 Parallel Complexity ......................................... 30
  2.5 Completeness of the Determinant Family .............. 34

3 Some Complete Families of Polynomials 37
  3.1 Generating Functions of Graph Properties ............ 37
  3.2 p-Computable Families ...................................... 39
  3.3 VNP-Complete Families ...................................... 41
    3.3.1 Matchings ............................................. 42
    3.3.2 Cliques .................................................. 43
    3.3.3 Cycle Format Polynomials ............................. 44
    3.3.4 Graph Factors ......................................... 48
  3.3.5 Hamilton Cycles of Planar Graphs .................... 50
  3.3.6 Self Avoiding Walks .................................... 53
  3.3.7 Connectivity ............................................ 55
4 Cook’s versus Valiant’s Hypothesis ............................................................... 61
4.1 Dependence on the Field ................................................................. 61
4.2 Statement of Main Results .............................................................. 64
4.3 Review of Discrete Complexity Classes .......................................... 66
4.4 Relating NP to Counting Classes ..................................................... 69
4.5 A Bound on the Heights ................................................................. 71
4.6 Roots of Univariate Polynomials Modulo a Prime .............................. 75
4.7 Proof of Thm. 4.5 ............................................................................ 76

5 The Structure of Valiant’s Complexity Classes ......................................... 81
5.1 Outline and Comparison with Previous Work ..................................... 81
5.2 An Abstract Diagonalization Theorem .............................................. 82
5.3 An Abstract Embedding Theorem .................................................... 85
5.4 Structure of Valiant’s Complexity Classes ...................................... 89
5.5 A Specific Family neither Complete nor p-Computable ..................... 92
5.6 Relativized Complexity Classes ....................................................... 96

6 Fast Evaluation of Representations of General Linear Groups ............... 105
6.1 Description of the Problem ............................................................. 105
6.2 Preliminaries on Representations of GL_m ....................................... 107
6.3 Auxiliary Fast Linear Algebra Algorithms ...................................... 109
6.4 An Algorithm for Evaluating Representations .................................. 112
6.5 A Lower Bound ............................................................................. 114
6.6 Fast Evaluation of Legendre Functions .......................................... 115

7 The Complexity of Immanants ................................................................. 117
7.1 Motivation and Outline of Chapter ............................................... 117
7.2 Fast Evaluation of Immanants ......................................................... 118
7.3 Completeness Results for Immanants ............................................. 120
7.4 Character Formulas for the Symmetric Group .................................. 121
7.5 p-Definability of Immanants .......................................................... 126
7.6 Completeness Proofs .................................................................... 128

8 Separation Results and Future Directions ............................................ 137
8.1 Specific Families which are not p-Definable ..................................... 137
8.2 Separations for the Complexity Class VQP ..................................... 141
8.3 Possible Connections to Univariate Polynomials .............................. 143
8.4 Connections to the BSS-Model ....................................................... 145

References ............................................................................................. 151

List of Notation ....................................................................................... 161

Index ........................................................................................................ 164
1

Introduction

We start with a brief description of the classical discrete theory of NP-completeness, give an overview of its generalization to the Blum-Shub-Smale model, and present the main features of Valiant’s algebraic model. Then we outline the organization of the book and present some of its highlights in an informal way.

1.1 Classical Model

Computational complexity theory provides a framework for understanding the cost required to solve algorithmic problems. Nowadays, the dominant approach to this theory is based on the computational model of the Turing machine. This provides a formalization of algorithms that operate on finite strings from a finite alphabet. One of the most important and successful concepts developed in computational complexity is that of NP-completeness, originating in the work of Cook [25], Karp [60], and Levin [74]. We are going to describe briefly the main features of this theory. (For details see Garey and Johnson [38] or Papadimitriou [87].)

A language $A \subseteq \{0, 1\}^*$ is a set of finite strings over the alphabet $\{0, 1\}$. The length of a string $x$ is denoted by $|x|$. The focus is on decision problems, which are formalized by languages. The complexity class P of polynomial time decidable languages consists of all languages $A$ such that membership of $x \in \{0, 1\}^n$ to $A$ can be decided by a deterministic Turing machine in a number of steps, which is bounded by a polynomial in $n$ (p-bounded in $n$). This class serves as a rough idealization for the problems that are computationally tractable. Another, more subtle class NP is considered, that contains all problems for which a solution can be verified in a polynomial number of steps. Formally, this class can be defined as follows. Let $R \subseteq \{0, 1\}^* \times \{0, 1\}^*$ be a relation between strings, which is decidable in polynomial time, and balanced in the following sense: there is a p-bounded function $t: \mathbb{N} \to \mathbb{N}$ satisfying $|y| \leq t(|x|)$ for all $(x, y) \in R$. The language $A = \{x \in \{0, 1\}^* \mid \exists y (x, y) \in R\}$ is in the class NP, and all languages in NP are obtained this way. Hereby, the string $y$ is interpreted as a (short) witness for the membership of $x$ to $A$.

A fundamental conjecture, Cook’s hypothesis, claims that $P \neq NP$. This is undoubtedly the most famous open problem of theoretical computer science. Although there is a lot of practical evidence towards it, a proof is far out of sight.
Languages $A$ and $B$ are compared by means of reductions. $A$ is said to be polynomial time (many one) reducible to $B$ iff there exists a string function $\rho: \{0,1\}^* \to \{0,1\}^*$, which is computable in polynomial time on a Turing machine, such that $A = \rho^{-1}(B)$. A language in NP is called NP-complete iff every problem in NP can be polynomial time reduced to it. Usually, a proof that a decision problem is NP-complete, is taken as evidence of intractability. Indeed, such a problem is not in $P$ iff Cook’s hypothesis is true.

The importance of the P-NP-theory lies in the fact that countless problems of great practical significance in logic, number theory, and combinatorial optimization have been identified as NP-complete ones. Basic NP-complete problems are for instance the satisfiability problem for Boolean formulas, or the problem to decide whether a given graph has a Hamilton cycle. (Of course, these problems have to be encoded as languages.)

The counting complexity class $\#P$ introduced by Valiant [108] is relevant for our purposes. Any problem in NP is defined by a balanced, polynomial time decidable string relation $R$ as above. The associated counting problem is the following: given $x$, how many $y$ are there such that $(x, y) \in R$? So we ask to compute the function $\phi: \{0,1\}^* \to \mathbb{N}$ given by

$$\phi(x) := \#R_x := \# \{ y \mid (x, y) \in R \} .$$

The class $\#P$ is defined as the set of all such functions $\phi$. A reduction from the counting problem of $R$ to the counting problem of a string relation $S$ is given by two polynomial time computable functions $\rho: \{0,1\}^* \to \{0,1\}^*$ and $\sigma: \mathbb{N} \to \mathbb{N}$ (natural numbers encoded in binary) such that $\#R_x = \sigma(\#S_{\rho(x)})$ for all $x$. This reduction is called parsimonious iff $\sigma$ is the identity. A counting function in $\#P$ is called $\#P$-complete iff every counting function in $\#P$ can be reduced to it. For instance, the problem to count the number of Hamilton cycles in a given graph can be proven to be $\#P$-complete. It is an open problem whether the counting problem associated with an NP-complete problem is always $\#P$-complete.

Valiant [108] was able to prove that the problem to count all perfect matchings of a given bipartite graph is $\#P$-complete. This is particularly interesting, since the corresponding decision problem, also known as the marriage problem, is solvable in polynomial time. The number of perfect matchings of a bipartite graph $G$ can be expressed as the permanent

$$\text{per}(A) := \sum_{\pi \in S_n} A_{1,\pi(1)} \cdot \cdots \cdot A_{n,\pi(n)}$$

of the adjacency matrix $A$ of $G$. So we have the following important result.

**Theorem 1.1 (Valiant)** The problem to evaluate the permanent of a 0,1-matrix is $\#P$-complete.

If $\phi: \{0,1\}^* \to \mathbb{N}$ is a counting function as above, we may just ask for the parity of $\phi(x)$. The complexity class parity polynomial time $\oplus P$, introduced by
Papadimitriou and Zachos [80], comprises all such decision problems. One can prove that the problem to find the parity of the number of Hamilton cycles in a given graph is @P-complete with respect to polynomial time reductions. Remarkably, one can compute the parity of the number of perfect matchings of a bipartite graph in polynomial time, since this amounts to evaluating a determinant modulo 2. (Permanent and determinant coincide in characteristic two.) This also implies that there is no parsimonious reduction from the problem to count Hamilton cycles to the problem to count the number of perfect matchings.

1.2 Blum-Shub-Smale Model

Algebraic complexity theory is the study of the intrinsic algorithmic difficulty of algebraic and numeric problems. It is based on models of computation that are adapted to the problems under investigation: straight-line programs and computation trees. The basic computational step is an arithmetic operation (or comparison) of elements of an algebraic structure, for instance of a finite field, or the field of real or complex numbers. In the latter cases, an important idealization is the assumption of exact arithmetic. The BSS-model, introduced by Blum, Shub, and Smale [12], adds a new feature to this, by combining the algebraic models of computation with the concept of uniformity, a condition sine qua non in the classical theory of computation. (Before, uniformity was not studied systematically in algebraic complexity, mainly because it is unknown how to exploit the uniformity condition for lower bound proofs.) Instead of considering for each dimension $n$ a separate algorithm solving problems of that dimension (for instance, evaluating the permanent of an $n$ by $n$ matrix), we clearly want to have one uniform algorithm capable of solving problems of any dimension. This aspect is incorporated in the definition of the BSS-machine over a field $k$ with the introduction of shift nodes that are used to access the contents of registers. We do not attempt to give a formal definition of such machines here; instead, the reader is referred to the recent textbook [11] for more details. Such a machine is a finite object except that it comes with a finite number of constants in $k$. (For instance in the case $k = \mathbb{R}$, these constants might not have a finite description.) It is important that BSS-machines over the field $\mathbb{F}_2$ with two elements are equivalent to Turing machines. This implies that the classical theory of complexity and computation is a special case of the more general theory to be developed.

Let $k^\infty := \bigcup_{n \geq 0} k^n$ be the set of finite sequences over $k$. A BSS-machine $M$ over $k$ takes as inputs elements $x$ of $k^\infty$ and computes an output $y \in k^\infty$ if the computation stops. The number of steps spent during this computation measures the number of arithmetic operations and comparisons of elements of $k$, as well as the number of shift operations used to address registers. We mention that if we restrict ourselves to inputs $x \in k^n$ of a fixed length $n$, then the BSS-machines become equivalent to the model of computation trees.
Over a fixed field $k$, one can now define the complexity classes $P_k$ and $NP_k$ in complete analogy with the classical situation. For instance, the class $P_k$ consists of all decision problems $A$, defined as a subset of $k^{\infty}$, such that membership of $x \in k^{\infty}$ to $A$ can be decided by a BSS-machine over $k$ in a $p$-bounded number of steps in the length $n$ of an input $x \in k^n$. We have a notion of polynomial time reduction between decision problems, by which we define NP-completeness over $k$.

Are there natural NP-complete problems in this setting? Consider the following decision version of the fundamental problem to find zeros of systems of polynomial equations, called Hilbert Nullstellensatz (HN):

Given a system of polynomials $f_1, \ldots, f_r$ in $n$ variables over $\mathbb{C}$, decide whether the $f_i$ have a common complex zero.

The length of an input to this problem is the number of coefficients of the polynomials $f_i$. Note that (HN) is clearly in NP$_C$, since a solution can be easily verified by plugging in into the polynomials.

Theorem 1.2 (Blum, Shub, Smale) The Hilbert Nullstellensatz problem is NP-complete over the complex numbers.

We remark that there is an analogous completeness result over the reals. Over the field $\mathbb{F}_2$, a corresponding result easily implies the completeness of the satisfiability problem for Boolean formulas.

In this context, there is the fundamental conjecture that $P_k \neq NP_k$, which we will call the BSS-hypothesis. Note that it might depend on the underlying field $k$. The BSS-hypothesis over $\mathbb{C}$ expresses that there is no polynomial time algebraic algorithm (formalized by BSS-machines), that solves the Hilbert Nullstellensatz problem. Smale [102] considers the question of whether $P \neq NP$ (over $\mathbb{F}_2$ or $\mathbb{C}$) as one of the most important open problems of mathematics, at position three right after Riemann’s hypothesis and the Poincaré conjecture!

The ultimate goal of the effort started with the introduction of the BSS-model is to combine ideas developed in theoretical computer science with numerical analysis in order to create a deeper foundation of the latter. The assumption of infinite precision arithmetic in this model is of course very idealized. We remark that there are ongoing efforts towards a more realistic model of computation over the reals which takes into account the conditioning of inputs as well as round-off errors (see the survey [101]).

1.3 Valiant’s Model

Already ten years before the BSS-paper, Valiant [107, 110] had proposed an analogue of the theory of NP-completeness in an entirely algebraic framework, in connection with his famous hardness result for the permanent [108] (see
also [111]). The goal of this research monograph is to further develop Valiant’s proposal, and to clarify its connections both to the classical and to the BSS-model.

Let us first outline the main features of Valiant’s model. (A detailed presentation can be found in Chap. 2.) In contrast to the theories explained before, who deal with decision problems, we study here very basic computational problems: the evaluation of multivariate polynomials.

A \( p \)-family over a fixed field \( k \) is a sequence \( f = (f_n) \) of multivariate polynomials such that the number of variables as well as the degree of \( f_n \) are \( p \)-bounded functions of \( n \). The algorithmic problem to study is the evaluation of these polynomials. An interesting example of a \( p \)-family is the permanent family \( \text{PER} = (\text{PER}_n) \), where \( \text{PER}_n \) is the permanent of an \( n \times n \) matrix with independent indeterminate entries.

Let \( L(f_n) \) denote the total complexity of \( f_n \), that is, the minimum number of arithmetic operations \( +,-,\,* \) sufficient to compute \( f_n \) from the variables \( X_i \) and constants in \( k \) by a straight-line program \( \Gamma_n \) (for a formal definition see Sect. 2.1). A \( p \)-family \( (f_n) \) is called \( p \)-computable iff \( L(f_n) \) is a \( p \)-bounded function of \( n \). The \( p \)-computable families constitute the complexity class \( \text{VP} \). (\( V \) is an acronym for Valiant.)

We remark that these definitions do not take uniformity into account: \( f_n \) is not required to be “uniformly describable” in dependence on \( n \), nor do the straight-line programs \( \Gamma_n \) need to have anything in common. (But in all interesting situations, this will be the case.) We could incorporate a uniformity assumption into this model, for instance by requiring that \( f_n \) or \( \Gamma_n \) be polynomial time computable by a BSS-machine over \( k \). In our opinion, this would just complicate the theory, but not make it more meaningful. At this point, we would also like to mention the paper [100] by Skyum and Valiant, which develops a nonuniform Boolean complexity theory along the same lines as it is done here.

We remark that our model is very simple, as only straight-line computations are considered. So we do not have branchings if \( \text{then else} \) according to the outcome of comparison tests. In fact, it is rather astonishing that completeness results can be obtained in such a restricted framework!

One more comment: the restriction to \( p \)-bounded degrees is a severe one; 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.

We define now an analogue of the class \( \text{NP} \). A \( p \)-family \( f = (f_n) \) is called \( p \)-definable iff there exists a \( p \)-computable family \( g = (g_n) \) such that for all \( n \)

\[
    f_n(X_1, \ldots, X_{v(n)}) = \sum_{e \in \{0,1\}^{s(n)} \cdot s(n)} g_n(X_1, \ldots, X_{v(n)}, e_{v(n)+1}, \ldots, e_{u(n)})
\]

The set of \( p \)-definable families form the complexity class \( \text{VNP} \). The class \( \text{VP} \) is obviously contained in \( \text{VNP} \), and Valiant’s hypothesis claims that this inclusion is strict. Note that as in the BSS-setting, this hypothesis a priori depends on the underlying field \( k \). This possible dependence will be investigated in Sect. 4.1.
We shall now define a reduction notion called \( p \)-projection, which will serve to compare \( p \)-families. A polynomial \( f_n \) is said to be a projection of a polynomial \( g_m \in k[X_1, \ldots, X_d] \), for short \( f_n \leq g_m \), if \( f_n(X_1, \ldots, X_{v(n)}) = g_m(a_1, \ldots, a_u) \) for some \( a_i \in k \cup \{ X_1, \ldots, X_{v(n)} \} \). That is, \( f_n \) can be derived from \( g_m \) through substitution by indeterminates and constants. Let us call a function \( t : \mathbb{N} \rightarrow \mathbb{N} \) \( p \)-bounded from above and below iff there exists some \( c > 0 \) such that \( n^{1/c} - c \leq t(n) \leq n^c + c \) for all \( n \). We call a \( p \)-family \( f = (f_n) \) a \( p \)-projection of \( g = (g_m) \), in symbols \( f \preceq_p g \), iff there exists a function \( t : \mathbb{N} \rightarrow \mathbb{N} \) which is \( p \)-bounded from above and below such that

\[
\exists n_0 \ \forall n \geq n_0 : f_n \leq g_{t(n)}.
\]

(This definition of \( \leq_p \) differs slightly from the one given in [107].) Finally, a \( p \)-family \( g \in \text{VNP} \) is called \text{VNP-complete} iff any \( f \in \text{VNP} \) is a \( p \)-projection of \( g \).

The main result of the theory is the following algebraic analogue of Thm. 1.1.

**Theorem 1.3 (Valiant)** The \( p \)-family \( \text{PER} \) is \text{VNP-complete} if \( \text{char} k \neq 2 \).

### 1.4 Overview of Main Results

We outline the organization of the book and present some of our highlights in an informal way.

Chap. 2 is devoted to a detailed introduction into Valiant’s model. In particular, we give a complete and detailed proof of the VNP-completeness of the permanent family (Thm. 1.3).

In Chap. 3 we prove that the families of generating functions corresponding to various NP (or \#P) complete graph properties are VNP-complete. We just mention here one particularly nice result. Let \( F \) and \( G \) be graphs and assume that \( F \) is connected. We define an \( F \)-factor of \( G \) as a spanning subgraph of \( G \), all of whose connected components are isomorphic to \( F \). For instance, if \( F \) is the connected graph \( K_2 \) with two nodes, then an \( F \)-factor is just a perfect matching of \( G \). We assume now that \( G \) is the complete graph with \( n \) nodes and that all its edges \( e \) have a weight \( X_e \), where the \( X_e \) are independent indeterminates. We then define the weight of an \( F \)-factor as the product of the weights of all its edges. The sum of the weights of all \( F \)-factors of \( G \) is called the \( F \)-factor polynomial \( \text{Fact}_n(F) \). Note the following: if we would work with the complete bipartite graph \( G \) and take \( F = K_2 \), then we would get the permanent polynomial \( \text{PER}_n \).

In Thm. 3.16 we will show that the sequence of \( F \)-factor polynomials is VNP-complete if \( F \) has at least two nodes and \( \text{char} k \neq 2 \). The point is to prove that the permanent family is a \( p \)-projection of the family of \( F \)-factor polynomials. This can be shown with a similar technique as in Kirkpatrick and Hell [65].

We have discussed three theories of NP-completeness in this introduction. Each of them hinges on a fundamental hypothesis of type \( P \neq \text{NP} \), which seems
to be a very hard mathematical problem. Can we at least say something about the interrelations between the different P-NP-hypotheses? More specifically, what are the connections of Cook’s hypothesis with the BSS-hypothesis and Valiant’s hypothesis over the complex numbers? Our knowledge about this is depicted in Fig. 1.1. It has been shown independently by several people, including ourselves, that the nonunifom version of Cook’s hypothesis $P/\text{poly} \neq \text{NP/poly}$ implies $P \neq \text{NP}$ over $\mathbb{C}$. (For a proof see Cucker et al. [31]. A definition of these nonuniform complexity classes can be found in Sect. 4.3.)

Whether the corresponding implication is also true over the reals (where machines can branch according to $\leq$-test) is a major open problem in the BSS-theory.

![Figure 1.1: Interrelations between different P-NP-hypotheses.](image)

In Chap. 4, we will prove that the nonuniform version of Cook’s hypothesis also implies Valiant’s hypothesis over $\mathbb{C}$, under the generalized Riemann hypothesis. In fact, a connection to parallel complexity shows up here: Valiant’s hypothesis over $\mathbb{C}$ is also implied by the separation $\text{NC/poly} \neq \text{P/poly}$. The main difficulty here is to eliminate the complex constants. For doing this, we have developed a general result about the frequency of primes $p$ such that a system of integer polynomial equations solvable over $\mathbb{C}$ has a solution modulo $p$. We think this result (Thm. 4.4) is interesting in its own right.

We conjecture that Valiant’s hypothesis implies the BSS hypothesis over the complex numbers. Loosely speaking, this means that if the permanent is intractable, then solving systems of polynomial equations is intractable as well. In Chap. 8 we will prove some weaker implications and clarify the problems that
have to be overcome in order to settle this conjecture.

Chap. 5 is devoted to investigations in the spirit of structural complexity, similar as in Ladner [70] and Schöning [96] for the classical model. We call two \( p \)-definable families \( f,g \) \( p \)-equivalent iff we have \( f \leq_p g \) and \( g \leq_p f \). The equivalence classes form the poset of \( p \)-definable \( p \)-degrees. Assuming Valiant’s hypothesis, we will prove that any countable poset can embedded in the poset of \( p \)-degrees. This proof will imply the existence of \( p \)-definable families of intermediate complexity: neither \( \text{VNP} \)-complete nor \( p \)-computable. Rather surprisingly, we found an explicit example for such a family over \( \mathbb{F}_2 \); the family of cut enumerators (Cut\(^2\)), which is defined as follows:

\[
\text{Cut}_n^2 := \sum_S \prod_{i \in A, j \in B} X_{ij} .
\]

Here, the sum is over all cuts \( S = \{A, B\} \) of the complete graph \( K_n \) on the set of nodes \( \{1, 2, \ldots, n\} \). (The \( X_{ij} \) are independent indeterminates and we assume \( X_{ji} = X_{ij} \). A cut of a graph is a partition of its set of nodes into two nonempty subsets.) We remark that in the classical and the BSS-model, one can also show the existence of problems of intermediate complexity, but no specific examples are known.

Chap. 6 and 7 are a continuation of work by Hartmann [47] and Barvinok [5] on the complexity of immanants.

From the viewpoint of computational complexity, determinant and permanent have, in spite of the similarity in their definition, very little in common. While the determinant can be evaluated with a polynomial number of arithmetic operations (Gaussian elimination), Valiant’s completeness result indicates that this is not possible for the permanent. (The fastest known algorithm for the permanent due to Ryser [93] needs \( O(m2^m) \) operations for an \( m \) by \( m \) matrix.)

Immanants are matrix functions which generalize the permanent and determinant. Let \( A \) be an \( m \) by \( m \) matrix and \( \lambda \) be a partition of \( m \). The immanant of \( A \) corresponding to \( \lambda \) is defined as

\[
\text{im}_\lambda(A) = \sum_{\pi \in S_m} \chi_\lambda(\pi) \prod_{i=1}^m A_{i,\pi(i)} ,
\]

where \( \chi_\lambda \) is the irreducible character of the symmetric group \( S_m \) belonging to \( \lambda \) (cf. [13, 51]). For \( \lambda = (1,\ldots,1) \) this specializes to the determinant (\( \chi_\lambda = \text{sgn} \)), and for \( \lambda = (m) \) we obtain the permanent (\( \chi_\lambda = 1 \)). For a deeper understanding of the amazingly different complexity behaviour of determinants and permanents, immanants are a natural object to study.

We have developed an efficient algorithm which evaluates the immanant \( \text{im}_\lambda \) with \( O(m^2(m + s_\lambda d_\lambda)) \) nonscalar operations (the total complexity is slightly bigger, see Thm. 7.3). Here, \( s_\lambda \) and \( d_\lambda \) denote the number of standard, and semistandard tableaus on the Young diagram of \( \lambda \), respectively. So the invari-
ant $d_\lambda$ dominates the upper complexity bound. If this invariant is huge, the evaluation problem seems to become intractable. For instance, we have $d_\lambda = \binom{2m-1}{m}$ for permanents.

Our search for upper complexity bounds for immanants has lead us to the discovery of a fast algorithm to evaluate irreducible rational matrix representations of complex general linear groups with respect to a symmetry adapted basis (Gelfand-Tsetlin basis). This result (Thm. 6.6) is of considerable interest in its own right and may well have practical applications, given the importance of such representations in quantum mechanics. We confine ourselves with a simple application to the fast evaluation of associated Legendre functions.

In Chap. 7 we complement our algorithmic results with completeness proofs in Valiant’s model for immanants corresponding to certain hook or rectangular diagrams. Even though these results clarify the picture and solve an open problem posed by Strassen [106], the complexity of immanants is still far from being completely understood.

In the next chapter, we shall now proceed with a thorough treatment of Valiant’s model.