Preface

About "quantitative perspective." The subtitle of the book seems to be redundant and requires an explanation. The main purpose of computational complexity is to measure the amount of time, or of space, or of some other resource, that is necessary to solve a computational problem. Thus, by its very nature, computational complexity is a quantitative theory. However, a look at some of the best-known results in complexity (e.g., results asserting the absolute or the conditional separation of complexity classes, or the hardness of certain computational tasks) reveals that the quantitative component is, in many aspects, quite weak. For instance, from the deterministic time hierarchy theorem, we know that there exists a problem that is solvable in exponential time but not in polynomial time. This result, important as it is, raises several questions of a quantitative nature. We would like to know (a) something about the abundance of such problems, (b) if the hardness of the problem manifests itself for just a few rare and accidental inputs, or, on the contrary, for most inputs, and (c) if there is perhaps some approximation of the problem, in some natural sense, that is solvable in polynomial time.

This book analyzes such quantitative aspects of some of the most important results in computational complexity.

From a certain point of view, most theorems in computational complexity can be divided into two types. Type one consists of those theorems that state a complexity-related attribute of one individual function. Type two consists of those theorems that involve an entire class, or several classes, of functions. This taxonomy is important because it indicates the technical tools that we can use for the quantitative analysis. For type one results, the quantitative attributes of interest have a concrete numerical formulation expressed as a function of the input length. For example, if a function is hard (in some sense), we can ask on how many inputs of length n it is hard. The quantitative analysis of type two results is not so straightforward. A generic formulation of many theorems in this category is that there exists a function f in some class $\mathcal{C}$ that has property Q (for example: "There is a function in EXP that is not polynomial-time computable," or, "There exists a computable function that is speedable"). The obvious quantitative question, "How many functions f in $\mathcal{C}$ have property Q," is usually not too meaningful. Indeed, for most classes $\mathcal{C}$ and properties Q, it holds that if a function f in $\mathcal{C}$ has property Q, then almost every finite variation of f is in $\mathcal{C}$ and has property Q as well, and,

therefore, the answer is trivially "An infinity." Fortunately, mathematicians have already developed tools and theories to handle this type of situations. The solution is to rephrase the question as "How large is the subset of function in $\mathcal{C}$ that have property Q," and to seek answers such as "small," or "large," or several nuances inbetween by using concepts from topology and measure theory. The theoretical foundations of this approach are presented in Chapter 1, Section 1.2.

A result enhanced with relevant quantitative attributes is more informative and more convincing and, therefore, clearly has theoretical merit. Besides that, a quantitative result can have, especially for type one results, practical value as well. At some point, there has been a common perception that computational complexity is a theory of "bad news," because common results, such as showing that a problem is NP-complete, assert that real-world and innocent-looking tasks are not feasible (in general, if we assume some reasonable hypothesis). In fact, "bad news" is a relative term, and, indeed, in some situations, we want an adversary to not be able to perform a task, e.g., to not be able to break our cryptography protocol. However, a "bad news" result does not automatically become useful in such a scenario. For this to happen, its hardness features have to be quantitatively evaluated and shown to manifest extensively.

Audience. My intention has been to write the book so that it appeals to a large audience. Experts in computational complexity may be interested in the special "quantitative" angle from which most results are presented. However, my primary target audience is elsewhere. In my intention, the book should benefit the most a reader who knows already the basic tenets of complexity, enjoys a rigorous mathematical treatment of a subject, wants to find out more about complexity than what is covered in a standard course, and, in particular, is interested in the novel major developments in complexity. The book is self-contained and can serve as a textbook for a course in advanced computational complexity. In general, the book should appeal to graduate computer science students and postdocs, and to researchers who have an interest in theory and need a good understanding of computational complexity, e.g., researchers in algorithms, AI, logic, and other disciplines.

Topics. The intended audience has influenced the choice of topics. Most of them are relevant outside the immediate scope of computational complexity. Also, most of them go beyond the material that is covered in a standard first course in complexity theory. One chapter is dedicated to abstract complexity theory, an older field which, however, deserves attention because it lays out the foundations of complexity. The other chapters, on the other hand, focus on recent and important developments in complexity. The book presents in a fairly detailed manner concepts that have been at the center of the main research lines in complexity in the last decade or so, such as: average-complexity, quantum computation, hardness amplification, resource-bounded measure, the relation between one-way functions and pseudo-random generators, the relation between hard predicates and pseudo-random generators, extractors, derandomization of bounded-error probabilistic algorithms, probabilistically checkable proofs, non-approximability of optimization

problems, and others. In some cases, it has not been possible, given the book's scope, to present the ultimate results regarding some of these concepts. However, I have included a presentation of the proof techniques that are required to obtain such results.

Chapter 1 presents basic facts from the theory of computation, computational complexity, topology, and measure theory that are used throughout the book. Depending on the reader's familiarity with these matters, this chapter should be read first, or just browsed and used as a reference.

The other chapters are independent and can be read in any order.

Chapter 2 presents the most important results in abstract complexity theory. These are classical results, which are displayed here from a novel angle that emphasizes some important quantitative facets.

Chapter 3 explores quantitative issues regarding the most important complexity classes, namely P, NP, E, and EXP. It includes a section on average-case complexity.

Chapter 4 is dedicated to quantum computation. The discussion concentrates on the potential of quantum computation to vastly outperform classical computation.

Chapter 5 focuses on some of the basic primitive objects that are used in cryptography: One-way functions, pseudo-random generators, and hard functions and predicates. The presentation emphasizes the quantitative attributes of these primitives, an aspect that is essential for their utilization in cryptography.

Chapter 6 is dedicated to NP optimization problems. The chapter concentrates on the issue of whether individual problems from this category admit polynomial-time good approximation algorithms.

I will maintain a website for this book (accessible from my web page at `http://triton.towson.edu/~mzimand`). It will contain a list of comments and updates for the topics presented in the book, and a list of errata. Please send me your comments and any error that you find in the book.

Acknowledgments. I would like to thank Saul Lubkin who gave me the idea to write this book and encouraged me over the years to complete this undertaking. I am grateful to Richard Chang, William Gasarch, Sanjay Gupta, Omer Horvitz, and Jon Squire who read parts of the book and gave me useful suggestions. I thank Mirko Janc for his expert advice on typesetting issues and for the many hours he spent struggling with my poorly Latex-edited manuscript.

Special thanks to my wife Iliana and my son Paul. This book would not exist without their patience, support, and love.

Baltimore, February 18, 2004. *Marius Zimand*