ACM MONOGRAPH SERIES

# A General Theory of Optimal Algorithms

J. F. Traub
H. Woźniakowski

# A General Theory of Optimal Algorithms

*J. F. TRAUB*
*Departments of Computer Science and Mathematics*
*Columbia University*
*New York, New York*

*H. WOŹNIAKOWSKI*
*Institute of Informatics*
*University of Warsaw*
*Warsaw, Poland*

1980

ACADEMIC PRESS
A Subsidiary of Harcourt Brace Jovanovich, Publishers
New York London Toronto Sydney San Francisco

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

*United Kingdom Edition published by*
ACADEMIC PRESS, INC. (LONDON) LTD.
24/28 Oval Road, London NW1 7DX

**Library of Congress Cataloging in Publication Data**

Traub, Joe Fred, Date
A general theory of optimal algorithms.

(ACM monograph series)
Includes bibliographies and index.
1. Approximation theory--Data processing.
2. Mathematical optimization--Data processing.
3. Computational complexity--Data processing.
I. Woźniakowski, H., joint author. II. Title.
III. Series: Association for Computing Machinery.
ACM monograph series.
QA297.T68 511'.4 79-8859
ISBN 0-12-697650-3

AMS (MOS) 1970 Subject Classifications: 68C25, 65J10, 65J15

PRINTED IN THE UNITED STATES OF AMERICA

80 81 82 83 9 8 7 6 5 4 3 2 1

# Preface

The purpose of this monograph is to create a general framework for the study of optimal algorithms for problems that are solved approximately. For generality the setting is abstract, but we present many applications to practical problems and provide examples to illustrate concepts and major theorems.

The work presented here is motivated by research in many fields. Influential have been questions, concepts, and results from complexity theory, algorithmic analysis, applied mathematics and numerical analysis, the mathematical theory of approximation (particularly the work on $n$-widths in the sense of Gelfand and Kolmogorov), applied approximation theory (particularly the theory of splines), as well as earlier work on optimal algorithms. But many of the questions we ask (see Overview) are new. We present a different view of algorithms and complexity and must request the reader's indulgence because new concepts and questions require some new vocabulary.

Because we believe our readers come from diverse backgrounds and have varied interests, we provide (see Recommended Reading) eight suggested tracks through the book.

The monograph marks the coming together of two streams of research. One stream, which involved work on optimal algorithms for problems such as the search for the maximum, integration, and approximation, had its inception with the work of Kiefer, Sard, and Nikolskij around 1950. These pioneer researchers, and those who followed them, generally worked, with only a few significant exceptions, on rather specific problems. Only rarely was the complexity of the optimal algorithm included. The second stream, which studied the solution of nonlinear equations, began with Traub in 1961. See Part C for a history and an extensive annotated bibliography.

In two long reports (General Theory of Optimal Error Algorithms and Analytic Complexity, Parts A and B) we showed that both streams could be united in one general framework. This monograph includes extended and improved material from these two reports.

Often a long monograph summarizes knowledge in a field. This monograph, however, may be viewed as a report on *work in progress.* We provide a foundation for a scientific field that is rapidly changing. Therefore we list numerous conjectures and open problems as well as alternative models which need to be explored. There are many more questions which we have chosen not to list.

We want to acknowledge many debts. B. Kacewicz and G. W. Wasilkowski coauthored some parts of this book as indicated in the table of contents. In addition, they carefully read the manuscript and suggested valuable improvements. H. T. Kung and K. Sikorski checked portions of the manuscript and A. G. Sukharev provided valuable references to the Soviet literature. A. Bojańczyk, A. Kiełbasiński, and A. Werschulz commented on an earlier version of this manuscript. D. Josephson, our superb technical typist, prepared the entire manuscript and was always patient about making just one more change. N. K. Brassfield was invaluable in helping us with proofreading and index preparation.

Much of the research was done in the splendid research environments of the Computer Science Department at Carnegie-Mellon University and of the Institute of Informatics at the University of Warsaw. We are indebted to the hospitality of the University of California at Berkeley, where we spent the academic year 1978–1979 while completing the research and writing of the manuscript. We are indebted to M. Blum and R. Karp for arranging this visit and for stimulating conversations. Finally, we are pleased to thank the National Science Foundation (Grant MCS-7823676) and the Office of Naval Research (Contract N00014-76-C-0370) for supporting the research reported here.

# Recommended Reading

We recommend different portions of the book to readers with various interests. To get some feel for the material, we suggest that all readers should be familiar with the Preface, Overview, Introductions to Parts A–C and the Introductions to each of the 10 chapters in Part A. For readers with particular interests, we make the following recommendations.

RESEARCHERS INTERESTED IN OPEN PROBLEMS Our theory suggests many new problems. Conjectures, questions, and open problems are scattered throughout this book. See, in particular, Part A: Chapter 6, Section 7; Chapter 8, Sections 3 and 7; Chapter 10, Sections 2 and 5; Part B: Section 10.

RESEARCHERS INTERESTED IN THE LITERATURE ON HISTORY Bibliographical references and historical material may be found in many places. Part C is devoted entirely to a history and an annotated bibliography. Bibliographies may be found at the end of Parts A and B. See also Part A: Chapter 1, Section 2; Chapter 4, Section 1; Chapter 6, Sections 3–6; Chapter 8, Section 1; Part B: Section 2.

THEORETICAL COMPUTER SCIENTISTS [those interested in algorithms and complexity but not specifically in analytic complexity] Part A: Chapter 1, Sections 2, 3; Chapters 5, 9, 10; Part B: Sections 2, 8, 11.

MATHEMATICIANS [material of greatest mathematical interest] Part A: Chapters 2 and 7; Part B: Sections 4–7.

MATHEMATICAL THEORY OF APPROXIMATION Part A: Chapter 2, Section 6; Chapter 3, Section 5; Chapter 7, Section 4.

APPLIED APPROXIMATION THEORY Part A: Chapters 4 and 6; Part C.

NUMERICAL ANALYSTS AND APPLIED MATHEMATICIANS Part A: Chapters 3, 4, 6, 8–10; Part C.

SCIENTISTS AND ENGINEERS Appropriate sections of the applications chapters (Part A: Chapters 6 and 8), depending on individual interests.

# Contents*



*Chapter 4 of Part A was written jointly with G. W. Wasilkowski, University of Warsaw. Sections 4–8 of Part B were written jointly with B. Kacewicz, University of Warsaw.

## PART C BRIEF HISTORY AND ANNOTATED BIBLIOGRAPHY

*Although this may seem a paradox, all exact science is dominated by the idea of approximation.*

B. Russell

# Overview

We provide the mathematical foundations of a general theory of optimal algorithms for problems which are solved approximately. This theory is called analytic computational complexity. See Part A, Chapter 1, Section 4 for a discussion of the nature of this field.

In this Overview we have to use words such as problem, information, and optimal algorithm without definition. They are defined rigorously and in great generality later.

We pose new kinds of questions and provide at least partial answers to many of them. Some of these questions are listed later in this Overview. Because of the richness of our domain, we must leave many open problems which we hope will be settled later.

We cover topics ranging from those of concern to theoretical computer scientists and mathematicians to those also of practical interest. We give two examples.

Problem complexity is a measure of the intrinsic difficulty of obtaining the solution to a problem no matter how that solution is obtained. We shall define it (see Part A, Chapter 1, Sections 3 and 4) as the complexity of the "optimal complexity algorithm" for solving the problem. (Problem complexity can only be specified with respect to a model of computation and a class of "permissible" information operators; we ignore such specification here.) The determination of problem complexity is difficult and deep; it has been completely solved for very few problems. When the complexity of a number of problems is at least approximately determined, we can hierarchically arrange problems according to their difficulty. Several partial problem hierarchies are summarized in Part A, Chapter 9. We believe that the determination of problem complexities and problem hierarchies should be one of the central problems of theoretical computer science and mathematics.

A practical application of our work is the rationalization of the synthesis of algorithms. Traditionally, algorithms are derived by ad hoc criteria. We shall find that algorithms obtained by commonly used criteria may have no relation to optimal algorithms (see Part A, Chapter 6, Remark 4.1) and that algorithms using commonly used information may pay an arbitrarily high penalty compared with the optimal algorithm using optimal information (see Part A, Chapter 6, Section 6).

A central issue in computer science is the selection of the best algorithm for solving a problem. Selection of the best algorithm is a multivariate optimization problem with dimensions including time complexity, space complexity, simplicity of program implementation, robustness, and stability. In this monograph we deal only with time complexity although conclusions concerning space complexity could also be easily obtained by changing the set of primitives in the model of computation. In later work we shall investigate tradeoffs in various dimensions for important problems. First we must understand how to optimize in the important time complexity dimension.

The analysis needed to characterize and construct an optimal algorithm (optimal in any of the senses of this book) for a particular problem can be a difficult mathematical problem and will, in any case, require substantial analysis. This may, however, be viewed as a precomputing cost.

We have not stated the domain or scope of this monograph. We must defer a more precise specification until Part A, Chapter 1, Section 4 after certain ideas have been introduced. Here we limit ourselves to a vague description.

We study problems which either cannot be solved exactly with finite complexity or problems which we choose to solve only approximately for reasons of efficiency. To the first class belong "most" problems of mathematics, science, and engineering. See Part A, Chapters 6 and 8 for many applications. Prominent exceptions are combinatorial and certain algebraic problems. Examples of the second class are the approximate solution of large sparse linear systems and of certain hard (for example, NP-complete) combinatorial optimization problems. Indeed, our model includes algebraic complexity as a special case. See Part A, Chapter 1, Examples 3.2 and 3.3, and Section 4.

The generality and power of the theory stems from the central role of information. Adversary arguments based on the information used by an algorithm lead to lower bound theorems. While the notion of information leads to generality it also permits remarkable simplicity. Numerous earlier papers obtain an optimal algorithm under various technical assumptions on the class of algorithms and the class of problem elements. These assumptions are often not verifiable. Our results depend only on the information used and broad verifiable properties of a class of problem elements. They are independent of the structure of an algorithm; they depend only on the information the algorithm uses.

Our theory uses two mathematical models; for convenience we refer to them here as models $\alpha$ and $\beta$. In what follows we limit our description to Part A; a similar dichotomy exists for Part B (see Introduction to Part B).

In model $\alpha$ the basic optimality concepts of "optimal error algorithm" and "optimal information" are defined independently of a model of computation. Model $\beta$ consists of model $\alpha$ as well as concepts related to computation; the basic optimality concept here is "optimal complexity algorithm" for a problem. Much of Part A is devoted to model $\alpha$ for a number of reasons. Negative results can often be proven using model $\alpha$. Since these results are independent of a model of computation, they are all the stronger. Furthermore, we can use powerful existing mathematical techniques to determine the optimal error algorithm. Indeed, the optimal error algorithm using optimal information must be determined before we can get good bounds on the optimal complexity algorithm.

We mentioned the difficulty of determining problem complexity. We must almost always settle for upper and lower bounds. This is not surprising. Even in algebraic complexity, which is included as a special case in our framework, it is rare to know the exact problem complexity. However, we often have extremely tight bounds on problem complexity. For example, if the optimal error algorithm is a *linear algorithm*, then the bounds are always very tight. (See Part A, Chapters 5 and 6.) For examples of tight complexity bounds when there is no linear optimal algorithm, see Part A, Chapter 8 and Part B, Section 8.

We list 20 of the general questions to be studied. These 20 questions are completely or partially answered, with the exception of two questions for which we conjecture the answer.

1. What is a lower bound on the error of any algorithm for solving a problem using given information?
2. In general is there an algorithm which gets arbitrarily close to this lower bound?
3. When is the information strong enough to solve a problem to within a given accuracy?
4. What is the optimal information for solving a problem?
5. What is the minimal number of linear functionals to solve a problem to within a given accuracy?
6. For linear problems is there always a linear algorithm with optimal error? If not, is there always a linear algorithm whose error is within a constant factor of the optimal error? In a Hilbert space setting what is the value of this constant?

7. Given a specific problem, how do we characterize and construct an optimal algorithm for its solution?

8. Given a problem, what are tight upper and lower bounds on its complexity?

9. Can it be established that one problem is intrinsically harder than another?

10. Do there exist linear problems with arbitrary complexity? In particular, do there exist linear problems which are arbitrarily hard?

11. What can be said in general about the dependence of complexity upon the regularity of the class of "Problem elements"?

12. Are adaptive algorithms more powerful than nonadaptive algorithms for solving linear problems?

13. Are adaptive algorithms more powerful than nonadaptive algorithms for solving nonlinear problems?

14. What is the power of linear information operators?

15. What is the power of nonlinear information operators?

16. What is the error of the best algorithm using optimal adaptive linear information for computing a simple zero of a nonlinear scalar function?

17. What is the error of the best algorithm using optimal adaptive linear information for searching for the maximum of a unimodal function (generalized Kiefer problem)?

The following questions deal with iterative information and iterative algorithms:

18. What is the maximal order of any algorithm using given information?

19. What is the class of all problems which can be solved by iteration using linear information?

20. What is the minimal number of linear functionals to iteratively solve a system of nonlinear equations in $N$ dimensions?

We describe the overall structure of this monograph. It is divided into three parts. Parts A and B treat, respectively, a general information model and an iterative information model. Part C consists of a short history and an annotated bibliography of well over 300 items.

We end this Overview by describing our system for referring to material within the text. Theorems, equations, remarks, etc. are separately numbered for each section. A reference to material within the same chapter does not name the chapter. A reference to material within the same part but different chapter names the chapter. A reference to material in a different part names the part and chapter.

# PART A

# GENERAL INFORMATION MODEL

## INTRODUCTION

Central to our theory of optimal algorithms is the notion of information. Part A is devoted to the study of "general information," while Part B treats "iterative information."

We develop a theory of optimal algorithms using given information or optimally chosen information. Two types of optimal algorithms will be of interest, "optimal error algorithms" and "optimal complexity algorithms." In Chapter 6, we apply this theory to a wide range of linear problems, and in Chapter 8 to a number of nonlinear problems.

A large portion of Part A (Chapters 2–6) is devoted to the theory of linear problems using linear information and to the applications of this theory. This emphasis is due to a number of factors.

1. Our results in the linear theory are very strong. In some cases, the questions we pose can be completely answered.
2. Many important applications are specified by linear problems and linear information.
3. The *class* of nonlinear information is too powerful (see Chapter 7) in the general information setting.

We broadly summarize the contents of Part A. See the introductions to each chapter for more detailed summaries.

CHAPTER 1 The basic concepts are formalized. An adversary principle leads to sharp lower bounds on algorithm error. Our model of computation is defined. The basic notions of optimal error algorithm and optimal complexity algorithm are introduced. The scope of analytic complexity and of this book are discussed.

CHAPTER 2 Chapters 2–6 treat linear problems and linear information. In Chapter 2, the general theory of linear information is developed. We vary the information operator and study the optimal information for solving a problem. In a Hilbert space setting, we give a complete solution to the problem of optimal information. We study adaptive information and prove the surprising result that adaptive information is not more powerful than nonadaptive information for a linear problem. Relations between Gelfand $n$-widths and $n$th minimal diameters are determined.

CHAPTER 3 We study *linear* algorithms for linear problems; such algorithms must have good time and space complexity. Very tight lower and upper complexity bounds are established for any problem with a linear optimal error algorithm. An especially constructed example establishes the existence of a linear problem for which there is no linear optimal error algorithm; no such example is known for "naturally" occurring problems. We use Smolyak's theorem to establish that algorithms optimal in the sense of Sard and Nikolskij are optimal error algorithms. Relations between optimal linear algorithms and the linear Kolmogorov $n$-widths are established.

CHAPTER 4 Generally, our analysis is worst case over all the problem elements $f$ in a class $\mathfrak{I}_0$. Here we study algorithms for which the *local* error is almost as small as possible for *every* $f$ from $\mathfrak{I}_0$. An algorithm which enjoys this property has small deviation. Do there exist linear algorithms with small deviation which are optimal or nearly optimal error algorithms? We introduce spline algorithms and show they permit us to answer this question.

CHAPTER 5 We specify our model of computation for the linear case. We show that there exist linear problems with essentially arbitrary complexity. This implies the existence of arbitrarily hard linear problems and that there are no "gaps" in the complexity function.

CHAPTER 6 We apply our theory to the solution of many different linear problems. Included are the approximation of a linear functional, interpolation, integration, approximation, and the solution of linear partial differential equations. Results are obtained for various spaces of problem elements. We obtain optimal error and nearly optimal complexity algorithms, optimal information operators, and very tight bounds on problem complexity for many applications. In some cases, our new algorithms are faster than commonly used algorithms by an unbounded factor.