

Texts in Theoretical Computer Science

An EATCS Series

Editors: W. Brauer G. Rozenberg A. Salomaa

On behalf of the European Association
for Theoretical Computer Science (EATCS)

Advisory Board: G. Ausiello M. Broy C.S. Calude
S. Even J. Hartmanis N. Jones T. Leighton M. Nivat
C. Papadimitriou D. Scott

Juraj Hromkovič

Algorithmics for Hard Problems

Introduction
to Combinatorial Optimization,
Randomization, Approximation,
and Heuristics

Second Edition

With 71 Figures



Springer

Author

Prof. Dr. Juraj Hromkovič
Swiss Federal Institute of Technology
Department of Computer Science
ETH Zürich, ETH Zentrum RZ F2
8092 Zürich, Switzerland

Series Editors

Prof. Dr. Wilfried Brauer
Institut für Informatik, Technische Universität München
Boltzmannstr. 3, 85748 Garching bei München, Germany
Brauer@informatik.tu-muenchen.de

Prof. Dr. Grzegorz Rozenberg
Leiden Institute of Advanced Computer Science
University of Leiden
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
rozenber@liacs.nl

Prof. Dr. Arto Salomaa
Turku Centre for Computer Science
Lemminkäisenkatu 14 A, 20520 Turku, Finland
asalomaa@utu.fi

Corrected printing 2004

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Hromkovič, Juraj: Algorithmics for hard problems / J. Hromkovič. –

(Texts in theoretical computer science)

ISBN 978-3-642-07909-2

ISBN 978-3-662-05269-3 (eBook)

DOI 10.1007/978-3-662-05269-3

ACM Computing Classification (1998):

F.2, F.1.2–3, I.1.2, G.1.2, G.1.6, G.2.1, G.3, I.2.8

ISBN 978-3-642-07909-2

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag Berlin Heidelberg GmbH. Violations are liable for prosecution under the German Copyright Law.

springeronline.com

© Springer-Verlag Berlin Heidelberg 2001, 2003, 2004

Originally published by Springer-Verlag Berlin Heidelberg New York in 2004

Softcover reprint of the hardcover 2nd edition 2004

The use of general descriptive names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and therefore free for general use.

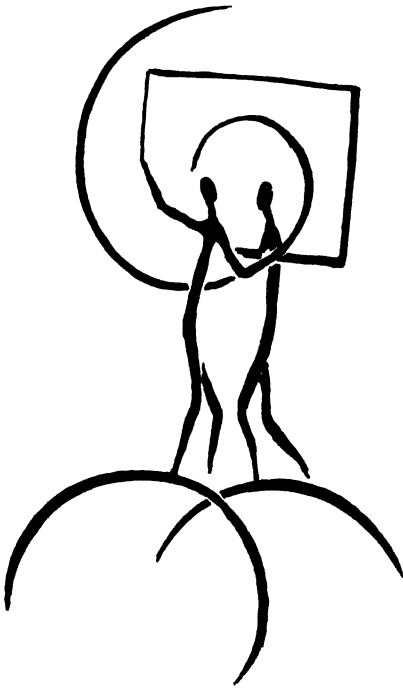
Illustrations: Ingrid Zámečnicková

Cover Design: KünkelLopkaWerbeagentur

Typesetting: Camera ready by author

Printed on acid-free paper 45/3142/PS - 5 4 3 2 1 0

To PETRA and PAULA



You have been told also that life is darkness,
and in your weariness you echo what was said by the weary.

And I say that life is indeed darkness
save when there is urge,

And all urge is blind save when there is knowledge,

And all knowledge is vain save where there is work,

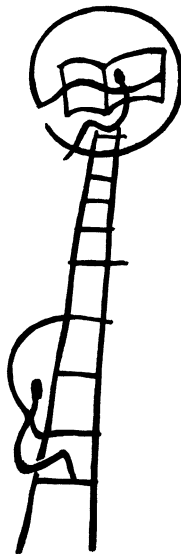
And all work is empty save when there is love;

And when you work with love you bind yourself
to yourself, and to one another, and to God ...

Work is love made visible.

And if you cannot work with love but only with distances,
it is better that you should leave your work
and sit at the gate of the temple and take alms of those
who work with joy.

KAHLIL GIBRAN
The Prophet



Preface to the Second, Enlarged Edition



The term algorithm is the central notion of computer science and algorithmics is one of the few fundamental kernels of theoretical computer science. Recent developments confirm this claim. Hardly any other area of theoretical computer science has been more lively and has achieved comparably deep progress and breakthroughs so fascinating (such as the PCP-theorem and efficient algorithms for primality testing) in recent years. The most exciting development happened exactly in the field of algorithmics for hard problems, which is the topic of this book.

The goal of this textbook is to give a transparent, systematic introduction to the concepts and to the methods for designing algorithms for hard problems. Simplicity is the main educational characteristic of this textbook. All ideas, concepts, algorithms, analyses, and proofs are first explained in an informal way in order to develop the right intuition, and then carefully specified in detail. Following this strategy we preferred to illustrate the algorithm design methods using the most transparent examples rather than to present the best, but too technical, results. The consequence is that there are sections where the first edition of this book does not go deep enough for advanced courses.

To smooth this drawback in the second edition, we extended the materials for some of the topics of central interest – randomized algorithms for primality testing and applications of linear programming in the design of approximation algorithms. This second edition contains both the Solovay-Strassen algorithm and the Miller-Rabin algorithm for primality testing with a selfcontained analysis of their behaviour (error probability). In order to give all related details, we extended the section about algebra and number theory in an appropriate way. To explain the power of the method of relaxation to linear programming, we added the concept of LP-duality and the presentation of the primal-dual method. As an introduction to this topic we used the Ford-Fulkerson pseudo-polynomial-time algorithm for the maximum flow problem, which is in the section about pseudo-polynomial-time algorithms.

In addition to extending some parts of the book, numerous small improvements and corrections were performed. I am indebted to all those who sent

VIII Preface to the Second, Enlarged Edition

me their comments and suggestions. Especially, I would like to thank Dirk Bongartz, Hans-Joachim Böckenhauer, David Buttgereit, Thomas Deselaers, Bernd Hentschel, Frank Kehren, Thorsten Uthke, Jan van Leeuwen, Sebastian Seibert, Koichi Wada, Manuel Wahle, Dieter Weckauf, and Frank Wessel who carefully read and commented on large parts of this book. Special thanks go to Dirk Bongartz and Hans-Joachim Böckenhauer for fruitful discussions on new parts of the book and for their valuable suggestions. The expertise of our \LaTeX experts Markus Mohr and Manuel Wahle was very helpful and is much appreciated. The excellent cooperation with Ingeborg Mayer and Alfred Hofmann from Springer-Verlag is gratefully acknowledged.

Last but not least I would like to express my deepest thanks to Peter Widmayer for encouraging me to make the work on this book a never-ending story.

Aachen, October 2002

Juraj Hromkovič

Preface



Algorithmic design, especially for hard problems, is more essential for success in solving them than any standard improvement of current computer technologies. Because of this, the design of algorithms for solving hard problems is the core of current algorithmic research from the theoretical point of view as well as from the practical point of view. There are many general textbooks on algorithmics, and several specialized books devoted to particular approaches such as local search, randomization, approximation algorithms, or heuristics. But there is no textbook that focuses on the design of algorithms for hard computing tasks, and that systematically explains, combines, and compares the main possibilities for attacking hard algorithmic problems. As this topic is fundamental for computer science, this book tries to close this gap.

Another motivation, and probably the main reason for writing this book, is connected to education. The considered area has developed very dynamically in recent years and the research on this topic discovered several profound results, new concepts, and new methods. Some of the achieved contributions are so fundamental that one can speak about paradigms which should be included in the education of every computer science student. Unfortunately, this is very far from reality. This is because these paradigms are not sufficiently known in the computer science community, and so they are insufficiently communicated to students and practitioners. The main reason for this unpleasant situation is that simple explanations and transparent presentations of the new contributions of algorithmics and complexity theory, especially in the area of randomized and approximation algorithms, are missing on the level of textbooks for introductory courses. This is the typical situation when principal contributions, whose seeping into the folklore of the particular scientific discipline is only a question of time, are still not recognized as paradigms in the broad community, and even considered to be too hard and too special for basic courses by non-specialists in this area. Our aim is to try to speed up this transformation of paradigmatic research results into educational folklore.

This book should provide a “cheap ticket” to algorithmics for hard problems. Cheap does not mean that the matter presented in this introductory

material is not precisely explained in detail and in its context, but that it is presented as transparently as possible, and formalized by using mathematics that is as simple as possible for this purpose. Thus, the main goal of this book can be formulated as the following optimization problem.

Input: A computer science student or a practitioner

Constraints:

- To teach the input the main ideas, concepts, and algorithm design techniques (such as pseudo-polynomial-time algorithms, parameterized complexity, local search, branch-and-bound, relaxation to linear programming, randomized algorithms, approximation algorithms, simulated annealing, genetic algorithms, etc.) for solving hard problems in a transparent and well-understandable way.
- To explain the topic on the level of clear, informal ideas as well as on the precise formal level, and to be self-contained with respect to all mathematics used.
- To discuss the possibilities to combine different methods in order to attack specific hard problems as well as a possible speedup by parallelization.
- To explain methods for theoretical and experimental comparisons of different approaches to solving particular problems.

Costs: The expected time that an input needs to learn the topic of the book (particularly, the level of abstractions of mathematics used and the hardness of mathematical proofs).

Objective: Minimization.

I hope that this book provides a feasible solution to this hard optimization problem. To judge the quality (approximation ratio) of the solution provided in this book is left to the reader.

I would like to express my deepest thanks to Hans-Joachim Böckenhauer, Erich Valkema, and Koichi Wada for carefully reading the whole manuscript and for their numerous comments and suggestions. I am indebted to Ivana Černá, Vladimír Černý, Alexander Ferrein, Ralf Klasing, Dana Pardubská, Hartmut Schmeck, Georg Schnitger, Karol Tauber, Ingo Wegener, and Peter Widmayer for interesting discussions or their comments on earlier drafts of this book. Special thanks go to Hans Wössner and the team of Springer-Verlag for their excellent assistance during the whole process of the manuscript preparation. The expertise and helpfulness of our L^AT_EX expert Alexander Ferrein was very useful and is much appreciated.

Last but not least I would like to thank Tanja for her patience with me during the work on this book.

Contents

- 1 Introduction** 1
- 2 Elementary Fundamentals** 11
 - 2.1 Introduction 11
 - 2.2 Fundamentals of Mathematics 13
 - 2.2.1 Linear Algebra 13
 - 2.2.2 Combinatorics, Counting, and Graph Theory 30
 - 2.2.3 Boolean Functions and Formulae 45
 - 2.2.4 Algebra and Number Theory 54
 - 2.2.5 Probability Theory 80
 - 2.3 Fundamentals of Algorithmics 93
 - 2.3.1 Alphabets, Words, and Languages 93
 - 2.3.2 Algorithmic Problems 97
 - 2.3.3 Complexity Theory 114
 - 2.3.4 Algorithm Design Techniques 134
- 3 Deterministic Approaches** 149
 - 3.1 Introduction 149
 - 3.2 Pseudo-Polynomial-Time Algorithms 152
 - 3.2.1 Basic Concept 152
 - 3.2.2 Dynamic Programming and Knapsack Problem 154
 - 3.2.3 Maximum Flow Problem and Ford-Fulkerson Method 157
 - 3.2.4 Limits of Applicability 167
 - 3.3 Parameterized Complexity 169
 - 3.3.1 Basic Concept 169
 - 3.3.2 Applicability of Parameterized Complexity 171
 - 3.3.3 Discussion 174
 - 3.4 Branch-and-Bound 175
 - 3.4.1 Basic Concept 175
 - 3.4.2 Applications for MAX-SAT and TSP 177
 - 3.4.3 Discussion 183

3.5	Lowering Worst Case Complexity of Exponential Algorithms	184
3.5.1	Basic Concept	184
3.5.2	Solving 3SAT in Less than 2^n Complexity	185
3.6	Local Search	189
3.6.1	Introduction and Basic Concept	189
3.6.2	Examples of Neighborhoods and Kernighan-Lin's Variable-Depth Search	193
3.6.3	Tradeoffs Between Solution Quality and Complexity	198
3.7	Relaxation to Linear Programming	209
3.7.1	Basic Concept	209
3.7.2	Expressing Problems as Linear Programming Problems . . .	211
3.7.3	The Simplex Algorithm	218
3.7.4	Rounding, LP-Duality and Primal-Dual Method	227
3.8	Bibliographical Remarks	243
4	Approximation Algorithms	247
4.1	Introduction	247
4.2	Fundamentals	248
4.2.1	Concept of Approximation Algorithms	248
4.2.2	Classification of Optimization Problems	253
4.2.3	Stability of Approximation	253
4.2.4	Dual Approximation Algorithms	258
4.3	Algorithm Design	260
4.3.1	Introduction	260
4.3.2	Cover Problems, Greedy Method, and Relaxation to Linear Programming	261
4.3.3	Maximum Cut Problem and Local Search	269
4.3.4	Knapsack Problem and PTAS	272
4.3.5	Traveling Salesperson Problem and Stability of Approximation	282
4.3.6	Bin-Packing, Scheduling, and Dual Approximation Algorithms	308
4.4	Inapproximability	316
4.4.1	Introduction	316
4.4.2	Reduction to NP-Hard Problems	317
4.4.3	Approximation-Preserving Reductions	319
4.4.4	Probabilistic Proof Checking and Inapproximability	329
4.5	Bibliographical Remarks	337
5	Randomized Algorithms	341
5.1	Introduction	341
5.2	Classification of Randomized Algorithms and Design Paradigms . .	343
5.2.1	Fundamentals	343
5.2.2	Classification of Randomized Algorithms	345
5.2.3	Paradigms of Design of Randomized Algorithms	359

5.3	Design of Randomized Algorithms	363
5.3.1	Introduction	363
5.3.2	Quadratic Residues, Random Sampling, and Las Vegas . . .	364
5.3.3	Primality Testing, Abundance of Witnesses, and One-Sided-Error Monte Carlo	369
5.3.4	Equivalence Tests, Fingerprinting, and Monte Carlo	385
5.3.5	Randomized Optimization Algorithms for MIN-CUT	392
5.3.6	MAX-SAT and Random Rounding	400
5.3.7	3SAT and Randomized Multistart Local Search	406
5.4	Derandomization	411
5.4.1	Fundamental Ideas	411
5.4.2	Derandomization by the Reduction of the Probability Space Size	413
5.4.3	Probability Space Reduction and MAX-E k SAT	418
5.4.4	Derandomization by the Method of Conditional Probabilities	420
5.4.5	Conditional Probabilities and Satisfiability	422
5.5	Bibliographical Remarks	426
6	Heuristics	431
6.1	Introduction	431
6.2	Simulated Annealing	433
6.2.1	Basic Concept	433
6.2.2	Theory and Experience	437
6.2.3	Randomized Tabu Search	441
6.3	Genetic Algorithms	444
6.3.1	Basic Concept	444
6.3.2	Adjustment of Free Parameters	452
6.4	Bibliographical Remarks	457
7	A Guide to Solving Hard Problems	461
7.1	Introduction	461
7.2	Taking over an Algorithmic Task or a Few Words about Money . . .	462
7.3	Combining Different Concepts and Techniques	463
7.4	Comparing Different Approaches	466
7.5	Speedup by Parallelization	468
7.6	New Technologies	477
7.6.1	Introduction	477
7.6.2	DNA Computing	479
7.6.3	Quantum Computing	486
7.7	Glossary of Basic Terms	492
	References	503
	Index	525