

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 S. Even
J. Hartmanis N. Jones T. Leighton M. Nivat
C. Papadimitriou D. Scott

Springer-Verlag Berlin Heidelberg GmbH

Juraj Hromkovič

Algorithmics for Hard Problems

Introduction
to Combinatorial Optimization,
Randomization, Approximation,
and Heuristics

With 64 Figures



Springer

Author

Prof. Dr. Juraj Hromkovič
RWTH Aachen
Computer Science I, Algorithms and Complexity
Ahornstraße 55, 52074 Aachen, Germany
jh@I1.informatik.rwth-aachen.de

Series Editors

Prof. Dr. Wilfried Brauer
Institut für Informatik, Technische Universität München
Arcisstrasse 21, 80333 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

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 978-3-662-04618-0

Library of Congress Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Hromkovič, Juraj:

Algorithmics for hard problems: introduction to combinatorial optimization, randomization, approximation, and heuristics / Juraj Hromkovič.

(Texts in theoretical computer science)

ISBN 978-3-662-04618-0 ISBN 978-3-662-04616-6 (eBook)

DOI 10.1007/978-3-662-04616-6

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.

<http://www.springer.de>

© Springer-Verlag Berlin Heidelberg 2001

Originally published by Springer-Verlag Berlin Heidelberg New York in 2001

Softcover reprint of the hardcover 1st edition 2001

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 regulations and therefore free for general use.

Cover design: design & production GmbH, Heidelberg

Typesetting: Camera-ready by the author

Printed on acid-free paper

SPIN: 10702395

45/3142 ud – 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

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 $\text{\LaTeX}$ 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.

Aachen, March 2001

Juraj Hromkovič

Table of 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 46
 - 2.2.4 Algebra and Number Theory 55
 - 2.2.5 Probability Theory 73
 - 2.3 Fundamentals of Algorithmics 86
 - 2.3.1 Alphabets, Words, and Languages 86
 - 2.3.2 Algorithmic Problems 90
 - 2.3.3 Complexity Theory 107
 - 2.3.4 Algorithm Design Techniques 128
- 3 Deterministic Approaches 143
 - 3.1 Introduction 143
 - 3.2 Pseudo-Polynomial-Time Algorithms 146
 - 3.2.1 Basic Concept 146
 - 3.2.2 Dynamic Programming and Knapsack Problem 148
 - 3.2.3 Limits of Applicability 151
 - 3.3 Parameterized Complexity 153
 - 3.3.1 Basic Concept 153
 - 3.3.2 Applicability of Parameterized Complexity 155
 - 3.3.3 Discussion 158
 - 3.4 Branch-and-Bound 159
 - 3.4.1 Basic Concept 159
 - 3.4.2 Applications for MAX-SAT and TSP 161
 - 3.4.3 Discussion 167
 - 3.5 Lowering Worst Case Complexity of Exponential Algorithms . . 168
 - 3.5.1 Basic Concept 168
 - 3.5.2 Solving 3SAT in Less than 2^n Complexity 169

3.6	Local Search	173
3.6.1	Introduction and Basic Concept	173
3.6.2	Examples of Neighborhoods and Kernighan-Lin's Variable-Depth Search	177
3.6.3	Tradeoffs Between Solution Quality and Complexity	182
3.7	Relaxation to Linear Programming	193
3.7.1	Basic Concept	193
3.7.2	Expressing Problems as Linear Programming Problems	195
3.7.3	The Simplex Algorithm	199
3.8	Bibliographical Remarks	209
4	Approximation Algorithms	213
4.1	Introduction	213
4.2	Fundamentals	214
4.2.1	Concept of Approximation Algorithms	214
4.2.2	Classification of Optimization Problems	218
4.2.3	Stability of Approximation	219
4.2.4	Dual Approximation Algorithms	224
4.3	Algorithm Design	226
4.3.1	Introduction	226
4.3.2	Cover Problems, Greedy Method, and Relaxation to Linear Programming	227
4.3.3	Maximum Cut Problem and Local Search	235
4.3.4	Knapsack Problem and PTAS	238
4.3.5	Traveling Salesperson Problem and Stability of Approximation	248
4.3.6	Bin-Packing, Scheduling, and Dual Approximation Algorithms	273
4.4	Inapproximability	281
4.4.1	Introduction	281
4.4.2	Reduction to NP-Hard Problems	282
4.4.3	Approximation-Preserving Reductions	284
4.4.4	Probabilistic Proof Checking and Inapproximability	294
4.5	Bibliographical Remarks	303
5	Randomized Algorithms	307
5.1	Introduction	307
5.2	Classification of Randomized Algorithms and Design Paradigms	309
5.2.1	Fundamentals	309
5.2.2	Classification of Randomized Algorithms	311
5.2.3	Paradigms of Design of Randomized Algorithms	325
5.3	Design of Randomized Algorithms	329
5.3.1	Introduction	329
5.3.2	Quadratic Residues, Random Sampling, and Las Vegas	330

5.3.3	Primality Testing, Abundance of Witnesses, and One-Sided-Error Monte Carlo	335
5.3.4	Some Equivalence Tests, Fingerprinting, and Monte Carlo	342
5.3.5	Randomized Optimization Algorithms for MIN-CUT . . .	348
5.3.6	MAX-SAT, Random Sampling, and Relaxation to Linear Programming with Random Rounding	357
5.3.7	3SAT and Randomized Multistart Local Search	363
5.4	Derandomization	367
5.4.1	Fundamental Ideas	367
5.4.2	Derandomization by the Reduction of the Probability Space Size	369
5.4.3	Reduction of the Size of the Probability Space and MAX- E_k SAT	374
5.4.4	Derandomization by the Method of Conditional Probabilities	376
5.4.5	Method of Conditional Probabilities and Satisfiability Problems	379
5.5	Bibliographical Remarks	383
6	Heuristics	387
6.1	Introduction	387
6.2	Simulated Annealing	389
6.2.1	Basic Concept	389
6.2.2	Theory and Experience	393
6.2.3	Randomized Tabu Search	397
6.3	Genetic Algorithms	400
6.3.1	Basic Concept	400
6.3.2	Adjustment of Free Parameters	408
6.4	Bibliographical Remarks	414
7	A Guide to Solving Hard Problems	417
7.1	Introduction	417
7.2	Taking over an Algorithmic Task or a Few Words about Money	418
7.3	Combining Different Concepts and Techniques	419
7.4	Comparing Different Approaches	422
7.5	Speedup by Parallelization	424
7.6	New Technologies	433
7.6.1	Introduction	433
7.6.2	DNA Computing	434
7.6.3	Quantum Computing	442
7.7	Glossary of Basic Terms	447
	References	459
	Index	481