

# Contents

<b>1</b>	<b>The Complexity of Optimization Problems</b>	<b>1</b>
1.1	Analysis of algorithms and complexity of problems . . . . .	2
1.1.1	Complexity analysis of computer programs . . . . .	3
1.1.2	Upper and lower bounds on the complexity of problems . . . . .	8
1.2	Complexity classes of decision problems . . . . .	9
1.2.1	The class NP . . . . .	12
1.3	Reducibility among problems . . . . .	17
1.3.1	Karp and Turing reducibility . . . . .	17
1.3.2	NP-complete problems . . . . .	21
1.4	Complexity of optimization problems . . . . .	22
1.4.1	Optimization problems . . . . .	22
1.4.2	PO and NPO problems . . . . .	26
1.4.3	NP-hard optimization problems . . . . .	29
1.4.4	Optimization problems and evaluation problems . . . . .	31
1.5	Exercises . . . . .	33
1.6	Bibliographical notes . . . . .	36
<b>2</b>	<b>Design Techniques for Approximation Algorithms</b>	<b>39</b>
2.1	The greedy method . . . . .	40
2.1.1	Greedy algorithm for the knapsack problem . . . . .	41
2.1.2	Greedy algorithm for the independent set problem . . . . .	43
2.1.3	Greedy algorithm for the salesperson problem . . . . .	47

## Table of contents

2.2	Sequential algorithms for partitioning problems . . . . .	50
2.2.1	Scheduling jobs on identical machines . . . . .	51
2.2.2	Sequential algorithms for bin packing . . . . .	54
2.2.3	Sequential algorithms for the graph coloring problem . . . . .	58
2.3	Local search . . . . .	61
2.3.1	Local search algorithms for the cut problem . . . . .	62
2.3.2	Local search algorithms for the salesperson problem . . . . .	64
2.4	Linear programming based algorithms . . . . .	65
2.4.1	Rounding the solution of a linear program . . . . .	66
2.4.2	Primal-dual algorithms . . . . .	67
2.5	Dynamic programming . . . . .	69
2.6	Randomized algorithms . . . . .	74
2.7	Approaches to the approximate solution of problems . . . . .	76
2.7.1	Performance guarantee: chapters 3 and 4 . . . . .	76
2.7.2	Randomized algorithms: chapter 5 . . . . .	77
2.7.3	Probabilistic analysis: chapter 9 . . . . .	77
2.7.4	Heuristics: chapter 10 . . . . .	78
2.7.5	Final remarks . . . . .	79
2.8	Exercises . . . . .	79
2.9	Bibliographical notes . . . . .	83
<b>3</b>	<b>Approximation Classes</b> . . . . .	<b>87</b>
3.1	Approximate solutions with guaranteed performance . . . . .	88
3.1.1	Absolute approximation . . . . .	88
3.1.2	Relative approximation . . . . .	90
3.1.3	Approximability and non-approximability of TSP . . . . .	94
3.1.4	Limits to approximability: The gap technique . . . . .	100
3.2	Polynomial-time approximation schemes . . . . .	102
3.2.1	The class PTAS . . . . .	105
3.2.2	APX versus PTAS . . . . .	110
3.3	Fully polynomial-time approximation schemes . . . . .	111
3.3.1	The class FPTAS . . . . .	111
3.3.2	The variable partitioning technique . . . . .	112
3.3.3	Negative results for the class FPTAS . . . . .	113
3.3.4	Strong NP-completeness and pseudo-polynomiality . . . . .	114
3.4	Exercises . . . . .	116
3.5	Bibliographical notes . . . . .	119
<b>4</b>	<b>Input-Dependent and Asymptotic Approximation</b> . . . . .	<b>123</b>
4.1	Between APX and NPO . . . . .	124
4.1.1	Approximating the set cover problem . . . . .	124
4.1.2	Approximating the graph coloring problem . . . . .	127

4.1.3	Approximating the minimum multi-cut problem . . .	129
4.2	Between APX and PTAS . . . . .	139
4.2.1	Approximating the edge coloring problem . . . . .	139
4.2.2	Approximating the bin packing problem . . . . .	143
4.3	Exercises . . . . .	148
4.4	Bibliographical notes . . . . .	150
<b>5</b>	<b>Approximation through Randomization</b>	<b>153</b>
5.1	Randomized algorithms for weighted vertex cover . . . . .	154
5.2	Randomized algorithms for weighted satisfiability . . . . .	157
5.2.1	A new randomized approximation algorithm . . . . .	157
5.2.2	A 4/3-approximation randomized algorithm . . . . .	160
5.3	Algorithms based on semidefinite programming . . . . .	162
5.3.1	Improved algorithms for weighted 2-satisfiability . . . . .	167
5.4	The method of the conditional probabilities . . . . .	168
5.5	Exercises . . . . .	171
5.6	Bibliographical notes . . . . .	173
<b>6</b>	<b>NP, PCP and Non-approximability Results</b>	<b>175</b>
6.1	Formal complexity theory . . . . .	175
6.1.1	Turing machines . . . . .	175
6.1.2	Deterministic Turing machines . . . . .	178
6.1.3	Nondeterministic Turing machines . . . . .	180
6.1.4	Time and space complexity . . . . .	181
6.1.5	NP-completeness and Cook-Levin theorem . . . . .	184
6.2	Oracles . . . . .	188
6.2.1	Oracle Turing machines . . . . .	189
6.3	The PCP model . . . . .	190
6.3.1	Membership proofs . . . . .	190
6.3.2	Probabilistic Turing machines . . . . .	191
6.3.3	Verifiers and PCP . . . . .	193
6.3.4	A different view of NP . . . . .	194
6.4	Using PCP to prove non-approximability results . . . . .	195
6.4.1	The maximum satisfiability problem . . . . .	196
6.4.2	The maximum clique problem . . . . .	198
6.5	Exercises . . . . .	200
6.6	Bibliographical notes . . . . .	204
<b>7</b>	<b>The PCP theorem</b>	<b>207</b>
7.1	Transparent long proofs . . . . .	208
7.1.1	Linear functions . . . . .	210
7.1.2	Arithmetization . . . . .	214

## Table of contents

7.1.3	The first PCP result . . . . .	218
7.2	Almost transparent short proofs . . . . .	221
7.2.1	Low-degree polynomials . . . . .	222
7.2.2	Arithmetization (revisited) . . . . .	231
7.2.3	The second PCP result . . . . .	238
7.3	The final proof . . . . .	239
7.3.1	Normal form verifiers . . . . .	241
7.3.2	The composition lemma . . . . .	245
7.4	Exercises . . . . .	248
7.5	Bibliographical notes . . . . .	249
<b>8</b>	<b>Approximation Preserving Reductions</b>	<b>253</b>
8.1	The World of NPO Problems . . . . .	254
8.2	AP-reducibility . . . . .	256
8.2.1	Complete problems . . . . .	261
8.3	NPO-completeness . . . . .	261
8.3.1	Other NPO-complete problems . . . . .	265
8.3.2	Completeness in exp-APX . . . . .	265
8.4	APX-completeness . . . . .	266
8.4.1	Other APX-complete problems . . . . .	270
8.5	Exercises . . . . .	281
8.6	Bibliographical notes . . . . .	283
<b>9</b>	<b>Probabilistic analysis of approximation algorithms</b>	<b>287</b>
9.1	Introduction . . . . .	288
9.1.1	Goals of probabilistic analysis . . . . .	289
9.2	Techniques for the probabilistic analysis of algorithms . . . . .	291
9.2.1	Conditioning in the analysis of algorithms . . . . .	291
9.2.2	The first and the second moment methods . . . . .	293
9.2.3	Convergence of random variables . . . . .	294
9.3	Probabilistic analysis and multiprocessor scheduling . . . . .	296
9.4	Probabilistic analysis and bin packing . . . . .	298
9.5	Probabilistic analysis and maximum clique . . . . .	302
9.6	Probabilistic analysis and graph coloring . . . . .	311
9.7	Probabilistic analysis and Euclidean TSP . . . . .	312
9.8	Exercises . . . . .	316
9.9	Bibliographical notes . . . . .	318
<b>10</b>	<b>Heuristic methods</b>	<b>321</b>
10.1	Types of heuristics . . . . .	322
10.2	Construction heuristics . . . . .	325
10.3	Local search heuristics . . . . .	329

10.3.1	Fixed-depth local search heuristics . . . . .	330
10.3.2	Variable-depth local search heuristics . . . . .	336
10.4	Heuristics based on local search . . . . .	341
10.4.1	Simulated annealing . . . . .	341
10.4.2	Genetic algorithms . . . . .	344
10.4.3	Tabu search . . . . .	347
10.5	Exercises . . . . .	349
10.6	Bibliographical notes . . . . .	350
<b>A</b>	<b>Mathematical preliminaries</b>	<b>353</b>
A.1	Sets . . . . .	353
A.1.1	Sequences, tuples and matrices . . . . .	354
A.2	Functions and relations . . . . .	355
A.3	Graphs . . . . .	356
A.4	Strings and languages . . . . .	357
A.5	Boolean logic . . . . .	357
A.6	Probability . . . . .	358
A.6.1	Random variables . . . . .	359
A.7	Linear programming . . . . .	361
A.8	Two famous formulas . . . . .	365
<b>B</b>	<b>A List of NP Optimization Problems</b>	<b>367</b>
	<b>Bibliography</b>	<b>471</b>
	<b>Index</b>	<b>515</b>