Table of Contents

Foreword xi

Acknowledgements xiii

Chapter 1. Introduction 1

1.1. Overview of the book 2
1.1.1. Constraint programming 3
1.1.2. Ant colony optimization 4
1.1.3. Constraint programming with ant colony optimization . 4

Chapter 2. Computational Complexity 7

2.1. Complexity of an algorithm 8
2.2. Complexity of a problem 10
2.2.1. The $\mathcal{P}$ class 10
2.2.2. The $\mathcal{NP}$ class 11
2.2.3. $\mathcal{NP}$-complete problems 12
2.2.4. $\mathcal{NP}$-hard problems 14
2.2.5. Undecidable problems 14
2.2.6. Complexity of optimization problems 15
2.3. Where the most difficult instances can be found 15
2.3.1. Phase transition 16
2.3.2. Search landscape 19
2.4. Solving $\mathcal{NP}$-hard problems in practice 21
2.4.1. Exploitation of particular cases 23
2.4.2. Approximation algorithms 23
2.4.3. Heuristics and metaheuristics 24
2.4.4. Structuring and filtering the search space 24

PART I. CONSTRAINT PROGRAMMING 27

Introduction to Part I 29

Chapter 3. Constraint Satisfaction Problems 31

3.1. What is a constraint? 31
3.1.1. Definition of a constraint 32
3.1.2. Arity of a constraint and global constraints 33
3.2. What is a constraint satisfaction problem? 33
3.2.1. Complexity of CSPs 34
3.3. Optimization problems related to CSPs 35
3.3.1. Maximizing constraint satisfaction 35
3.3.2. Constrained optimization 36
3.4. The n-queens problem 37
3.4.1. Description of the problem 37
3.4.2. First CSP model 38
3.4.3. Second CSP model 39
3.4.4. Third CSP model 40
3.4.5. Influence of the model on the solution process 41
3.5. The stable marriage problem 43
3.5.1. Description of the problem 43
3.5.2. CSP model 45
3.6. Randomly generated binary CSPs 46
3.7. The car sequencing problem 47
3.7.1. Description of the problem 47
3.7.2. CSP model 49
3.8. Discussion 50

Chapter 4. Exact Approaches 53

4.1. Construction of a search tree 53
4.2. Constraint propagation 57
4.2.1. Forward checking 57
4.2.2. Maintaining arc consistency 59
4.3. Ordering heuristics 60
4.3.1. Heuristics for choosing variables 61
4.3.2. Heuristics for choosing values 62
4.3.3. Randomized restart 63
4.4. From satisfaction to optimization problems 63
4.5. Discussion 65

Chapter 5. Perturbative Heuristic Approaches 69

5.1. Genetic algorithms 70
5.1.1. Basic principles 70
5.1.2. Using GAs to solve CSPs 73
5.2. Local search 73
5.2.1. Basic principles 73
5.2.2. Metaheuristics based on LS 75
5.2.3. Using LS to solve CSPs 77
5.3. Particle swarm optimization 78
5.3.1. Basic principles 78
5.3.2. Using PSO to solve CSPs 80
5.4. Discussion 80

Chapter 6. Constructive Heuristic Approaches 85

6.1. Greedy randomized approaches 86
6.1.1. Basic principles 86
6.1.2. Using greedy randomized algorithms to solve CSPs . . 88
6.2. Estimation of distribution algorithms 88
6.2.1. Basic principles 88
6.2.2. Using EDAs to solve CSPs 90
6.3. Ant colony optimization 90
6.4. Discussion 91

Chapter 7. Constraint Programming Languages 93

7.1. Constraint logic programming 94
7.2. Constraint programming libraries 96
7.3. Constraint-based local search 96
7.4. Discussion 99

PART II. ANT COLONY OPTIMIZATION 101

Introduction to Part II 103

Chapter 8. From Swarm Intelligence to Ant Colony Optimization 105

8.1. Complex systems and swarm intelligence 106
8.2. Searching for shortest paths by ant colonies 108
8.3. Ant system and the traveling salesman problem 111
8.3.1. Pheromone structure 112
8.3.2. Construction of a Hamiltonian cycle by an ant 114

8.3.3. Pheromone updating step 115
8.3.4. Artificial versus real ants 115
8.4. Generic ACO framework 116
8.4.1. Pheromone structure and construction graph 116
8.4.2. Construction of combinations by ants 118
8.4.3. Improving combinations with local search 120
8.4.4. Pheromone updating step 121
8.4.5. Parameters of an ACO algorithm 122

Chapter 9. Intensification versus Diversification 125

9.1. ACO mechanisms for intensifying the search 125
9.2. ACO mechanisms for diversifying the search 127
9.3. Balancing intensification and diversification 128
9.4. Measures of diversification/intensification 135
9.4.1. The λ-branching factor 136
9.4.2. Resampling ratio 136
9.4.3. Similarity ratio 137

Chapter 10. Beyond Static Combinatorial Problems 141

10.1. Multi-objective problems 141
10.1.1. Definition of multi-objective problems 141
10.1.2. Solving multi-objective problems with ACO 143
10.2. Dynamic optimization problems 145
10.2.1. Definition of dynamic optimization problems 145
10.2.2. Solving dynamic optimization problems with ACO . . . 146
10.3. Optimization problems over continuous domains 147
10.3.1. Definition of continuous optimization problems 147
10.3.2. Solving continuous optimization problems with ACO . 148

Chapter 11. Implementation Issues 151

11.1. Data structures 151
11.1.1. Data structures associated with pheromone factors . . . 151
11.1.2. Data structures associated with heuristic factors 153
11.1.3. Data structures associated with ants 154
11.2. Selection of a component with respect to probabilities 154
11.3. Implementation of a local search procedure 157
11.4. Computation of diversification/intensification measures . . . 157
11.4.1. Resampling ratio 158
11.4.2. Similarity ratio 158

PART III. CP WITH ACO 161

Introduction to Part III 163

Chapter 12. Sequencing Cars with ACO 165

12.1. Notation 165
12.2. A first pheromone structure for identifying good car sequences 166
12.2.1. Pheromone structure 167
12.2.2. Construction of a sequence by an ant 168
12.2.3. Pheromone laying step 170
12.3. A second pheromone structure for identifying critical cars . . 171
12.3.1. Pheromone structure 171
12.3.2. Construction of a sequence by an ant 172
12.3.3. Pheromone updating step 172
12.4. Combining the two pheromone structures 173
12.4.1. First pheromone structure 173
12.4.2. Second pheromone structure 173
12.4.3. Construction of a sequence by an ant 173
12.5. Comparison of the different ACO algorithms 174
12.5.1. Considered algorithms 174
12.5.2. Test suite 175
12.5.3. Parameter settings 175
12.5.4. Experimental results 177
12.6. Comparison of ACO with state-of-the-art approaches 178
12.6.1. Considered approaches 178
12.6.2. IDWalk 179
12.6.3. VFLS 179
12.6.4. Experimental set-up 180
12.6.5. Experimental results 180
12.7. Discussion 182

Chapter 13. Subset Selection with ACO 185

13.1. Subset selection problems 186
13.1.1. Maximum clique 186
13.1.2. Multidimensional knapsack 187
13.1.3. Maximum Boolean satisfiability 187
13.1.4. Maximum constraint satisfaction 187
13.1.5. Minimum vertex cover 188
13.1.6. Maximum common subgraph 188

13.1.7. Edge-weighted k-cardinality tree 189
13.2. Description of Ant-SSP 189
13.2.1. Construction of a combination by an ant 190
13.2.2. Pheromone laying step 192
13.3. Instantiations of Ant-SSP with respect to two pheromone strategies 192
13.3.1. The vertex pheromone strategy 193
13.3.2. The clique pheromone strategy 193
13.3.3. Comparison of the two strategies 194
13.4. Instantiation of Ant-SSP to solve CSPs 196
13.4.1. Heuristic factor 196
13.4.2. Local search 197
13.5. Experimental results 197
13.5.1. Randomly generated binary instances 197
13.5.2. Results on instances of the 2006 solver competition . . 199
13.6. Discussion 202

Chapter 14. Integration of ACO in a CP Language 205

14.1. Framework for integrating ACO within a CP library 206
14.1.1. Pheromone strategy 207
14.1.2. Construction of assignments 208
14.1.3. Pheromone updating step 210
14.2. Illustration of ACO-CP on the car sequencing problem 210
14.2.1. CSP model 210
14.2.2. Variable ordering heuristic 211
14.2.3. Pheromone strategies 211
14.2.4. Heuristic factor 212
14.2.5. Experimental results 213
14.3. Discussion 214

Chapter 15. Conclusion 215

15.1. Towards constraint-based ACO search 215
15.2. Towards a reactive ACO search 216

Bibliography 219

Index 231