Combinatorial Optimization and Theoretical Computer Science

Interfaces and Perspectives

30th Anniversary of the LAMSADE

Edited by
Vangelis Th. Paschos
This page intentionally left blank
Combinatorial Optimization and Theoretical Computer Science
Combinatorial Optimization and Theoretical Computer Science

Interfaces and Perspectives

30th Anniversary of the LAMSADÉ

Edited by
Vangelis Th. Paschos
Preface

Chapter 1. The Complexity of Single Machine Scheduling Problems under Scenario-based Uncertainty

Mohamed Ali ALOULOU and Federico DELLA CROCE

1.1. Introduction

1.2. Problem MinMax(1|\(\text{prec}_f \leq \theta\), \(\text{max}\)) ........................................ 25
   1.2.1. Uncertainty on due dates................................................... 25
   1.2.2. Uncertainty on processing times and due dates .......... 27
1.3. Problem MinMax(1|| \(\sum w_j C_j, w_j\)) ........................................ 28
1.4. Problem MinMax(1|| \(\sum U_j, \theta\)) ........................................ 29
   1.4.1. Uncertainty on due dates................................................... 29
   1.4.2. Uncertainty on processing times....................................... 29
1.5. Bibliography .............................................................................. 35

Chapter 2. Approximation of Multi-criteria Min and Max TSP(1, 2)........ 37

Eric ANGEL, Evripidis BAMPIS, Laurent GOURVÈS and Jérôme MONNOT

2.1. Introduction.................................................................................. 37
   2.1.1. The traveling salesman problem...................................... 37
   2.1.2. Multi-criteria optimization.............................................. 38
   2.1.3. Organization of the chapter.............................................. 39
2.2. Overview.................................................................................... 39
2.3. The bicriteria TSP(1, 2)................................................................ 40
   2.3.1. Simple examples of the non-approximability.................. 42
   2.3.2. A local search heuristic for the bicriteria TSP(1, 2)............ 43
   2.3.3. A nearest neighbor heuristic for the bicriteria TSP(1, 2)...... 48
   2.3.4. On the bicriteria Max TSP(1, 2)...................................... 52
2.4. $k$-criteria $TSP(1, 2)$ .......................................................... 55
   2.4.1. Non-approximability related to the number of generated solutions .................................................. 56
   2.4.2. A nearest neighbor heuristic for the $k$-criteria $TSP(1, 2)$ ................................................................. 60
2.5. Conclusion .................................................................................................................. 67
2.6. Bibliography .............................................................................................................. 68

Chapter 3. Online Models for Set-covering: The Flaw of Greediness ............... 71
Giorgio AUSIELLO, Aristotelis GIANNAKOS and Vangelis Th. PASCHOS

3.1. Introduction............................................................................................................ 71
3.2. Description of the main results and related work............................................... 73
3.3. The price of ignorance ....................................................................................... 76
3.4. Competitiveness of TAKE-ALL and TAKE-AT-RANDOM ......................... 77
   3.4.1. TAKE-ALL algorithm ........................................................................... 77
   3.4.2. TAKE-AT-RANDOM algorithm ..................................................... 78
3.5. The nasty flaw of greediness.............................................................................. 79
3.6. The power of look-ahead .................................................................................. 82
3.7. The maximum budget saving problem.............................................................. 88
3.8. Discussion ......................................................................................................... 91
3.9. Bibliography .................................................................................................... 91

Chapter 4. Comparison of Expressiveness for Timed Automata and Time Petri Nets ........................................... 93
Béatrice BÉRARD, Franck CASSEZ, Serge HADDAD, Didier LIME and Olivier-Henri ROUX

4.1. Introduction......................................................................................................... 93
4.2. Time Petri nets and timed automata................................................................. 95
   4.2.1. Timed transition systems and equivalence relations ......................... 96
   4.2.2. Time Petri nets .................................................................................... 98
   4.2.3. Timed automata ................................................................................. 101
   4.2.4. Expressiveness and equivalence problems .................................. 103
4.3. Comparison of semantics $I$, $A$ and $PA$ .................................................... 104
   4.3.1. A first comparison between the different semantics of TPNs .......... 104
   4.3.2. A second comparison for standard bounded TPN ....................... 107
4.4. Strict ordering results ....................................................................................... 111
4.5. Equivalence with respect to timed language acceptance ........................... 113
   4.5.1. Encoding atomic constraints .......................................................... 113
   4.5.2. Resetting clocks .............................................................................. 115
   4.5.3. The complete construction ............................................................ 116
   4.5.4. $\Delta(A)$ and $A$ accept the same timed language ........................ 117
   4.5.5. Consequences of the previous results .......................................... 121
4.6. Bisimulation of TA by TPNs ................................................................. 122
  4.6.1. Regions of a timed automaton ..................................................... 122
  4.6.2. From bisimulation to uniform bisimulation ............................... 124
  4.6.3. A characterization of bisimilarity ............................................... 128
  4.6.4. Proof of necessity ...................................................................... 129
  4.6.5. First construction .................................................................... 131
  4.6.6. Second construction .................................................................. 137
  4.6.7. Complexity results ................................................................... 141
4.7. Conclusion ...................................................................................... 142
4.8. Bibliography ................................................................................... 143

Chapter 5. A “Maximum Node Clustering” Problem ............................ 145
Giuliana CARELLO, Federico DELLA CROCE, Andrea GROSSO and Marco LOCATELLI
  5.1. Introduction.................................................................................... 149
  5.2. Approximation algorithm for the general problem ....................... 147
  5.3. The tree case ................................................................................ 150
    5.3.1. Dynamic programming .......................................................... 150
    5.3.2. A fully polynomial time approximation scheme .................... 155
  5.4. Exponential algorithms for special cases ...................................... 156
  5.5. Bibliography ................................................................................ 159

Chapter 6. The Patrolling Problem: Theoretical and Experimental Results .. 161
Yann CHEVALEYRE
  6.1. Introduction .................................................................................. 161
  6.2. The patrolling task ..................................................................... 162
  6.3. Previous work ............................................................................. 164
  6.4. The cyclic strategies ................................................................... 165
    6.4.1. Patrolling with a single-agent ................................................ 165
    6.4.2. Extending to multi-agent case ................................................. 167
    6.4.3. Optimality of cyclic strategies ............................................... 168
  6.5. Partition-based strategies .............................................................. 169
  6.6. Experiments ................................................................................ 171
  6.7. Conclusion .................................................................................. 172
  6.8. Bibliography ................................................................................ 174

Chapter 7. Restricted Classes of Utility Functions for Simple Negotiation
Schemes: Sufficiency, Necessity and Maximality.................................. 175
Yann CHEVALEYRE, Ulle ENDRISS and Nicolas MAUDET
  7.1. Introduction ................................................................................ 175
  7.2. Myopic negotiation over indivisible resources ............................ 177
    7.2.1. Negotiation problems and deals ............................................. 178
7.2.2. Negotiating with money .............................................................. 179
7.2.3. Negotiating without money ....................................................... 180
7.3. Convergence for restricted classes of utility functions ...................... 180
7.4. Modular utility functions and variants .............................................. 182
7.5. Sufficient classes of utility functions ................................................. 184
  7.5.1. Framework with money .......................................................... 184
  7.5.2. Framework without money ....................................................... 185
7.6. Necessity issues ................................................................................ 185
  7.6.1. Modularity is not necessary ......................................................... 186
  7.6.2. There is no sufficient and necessary class ................................ 186
  7.6.3. Evaluating conditions on profiles of utility functions is intractable ................................................................. 187
7.7. Maximal classes of utility functions ................................................. 191
  7.7.1. Framework with money .......................................................... 192
  7.7.2. Framework without money ....................................................... 196
7.8. Conclusion ....................................................................................... 198
7.9. Bibliography ..................................................................................... 199

Chapter 8. Worst-case Complexity of Exact Algorithms for NP-hard Problems .......................................................... 203
Federico DELLA CROCE, Bruno ESCOFFIER, Marcin KAMIŃSKI and Vangelis Th. PASCHOS

8.1. MAX-CUT ....................................................................................... 204
  8.1.1. Extending a partial partition of vertices ....................................... 206
  8.1.2. An algorithm for graphs with bounded maximum degree ........... 209
  8.1.3. An algorithm for general graphs ................................................. 210
8.2. Pruning the search tree by dominance conditions: the case of MAXCUT-3 .......................................................... 211
  8.2.1. Dominance conditions .............................................................. 212
  8.2.2. The worst-case upper-time bound for MAX-CUT-3 ................... 214
8.3. A more careful analysis for pruning: the case of MIN 3-DOMINATING SET .................................................................................................................. 226
  8.3.1. Analysis .................................................................................... 227
  8.3.2. Counting .................................................................................. 228
  8.3.3. Case n°1 .................................................................................. 228
  8.3.4. Case n°2 .................................................................................. 232
  8.3.5. Case n°3 .................................................................................. 234
  8.3.6. Complexity analysis ................................................................. 237
8.4. Bibliography ..................................................................................... 239
Chapter 9. The Online Track Assignment Problem ........................................ 241
Marc DEMANGE, Gabriele DI STEFANO and Benjamin LEROY-BEAULIEU

9.1. Introduction ................................................................................................ 241
9.1.1. Related works .................................................................................... 242
9.1.2. Results ............................................................................................... 243
9.2. Definitions and notations ........................................................................... 243
9.3. Bounds for the permutation graph in a left-to-right model ....................... 245
9.4. Bounds for overlap graphs ......................................................................... 249
9.5. Bounds for permutation graphs in a more general model ......................... 252
9.6. Conclusion ................................................................................................. 256
9.7. Bibliography .............................................................................................. 256

Chapter 10. Complexity and Approximation Results for the Min Weighted
Node Coloring Problem ............................................................................. 259
Marc DEMANGE, Bruno ESCOFFIER, Jérôme MONNOT, Vangelis Th. PASCHOS and Dominique DE WERRA

10.1. Introduction ............................................................................................. 259
10.2. General results ......................................................................................... 262
10.2.1. Structural properties ...................................................................... 262
10.2.2. Approximation results ...................................................................... 264
10.2.2.1. General graphs ........................................................................ 264
10.2.2.2. k-colorable graphs ............................................................... 266
10.2.2.3. When list coloring is easy ...................................................... 268
10.3. Weighted node coloring in triangle-free planar graphs ......................... 270
10.4. Weighted node coloring in bipartite graphs ......................................... 273
10.4.1. Hardness results ........................................................................... 273
10.4.2. P3-free bipartite graphs .............................................................. 278
10.5. Split graphs .............................................................................................. 280
10.5.1. Complexity result ....................................................................... 280
10.5.2. Approximation result ................................................................... 281
10.6. Cographs .................................................................................................. 284
10.7. Interval graphs ......................................................................................... 285
10.8. Bibliography ............................................................................................ 286

Chapter 11. Weighted Edge Coloring ............................................................... 291
Marc DEMANGE, Bruno ESCOFFIER, Giorgio LUCARELLI, Ioannis MILIS,
Jérôme MONNOT, Vangelis Th. PASCHOS and Dominique DE WERRA

11.1. Introduction ............................................................................................. 291
11.2. Related problems ..................................................................................... 293
11.3. Preliminaries and notation ..................................................................... 294
11.4. Complexity and (in) approximability ..................................................... 295
Chapter 12. An Extensive Comparison of 0-1 Linear Programs for the Daily Satellite Mission Planning

Virginie GABREL

12.1. Introduction

12.2. Different formulations for the daily satellite mission planning problem

12.3. Models comparison

12.4. Experiments and results

12.5. Conclusion

12.6. Bibliography

Chapter 13. Dantzig-Wolfe Decomposition for Linearly Constrained Stable Set Problem

Virginie GABREL

13.1. Introduction

13.2. The Dantzig-Wolfe decomposition in 0-1 linear programming

13.3. The stable set problem with additional linear constraints

13.4. Dantzig-Wolfe decomposition on stable set constraints: strengthening the LP-relaxation

13.4.1. Gap between LP-relaxations of master and initial problems

13.4.2. The case of co-comparability graph

13.4.3. A decomposition scheme for general graphs
Chapter 14. Algorithmic Games .............................................................. 339
Aristotelis GIANNAKOS, Vangelis Th. PASCHOS and Olivier POTTIÉ

14.1. Preliminaries .................................................................................... 340
  14.1.1. Basic notions on games .............................................................. 340
  14.1.2. Complexity classes mentioned throughout the chapter ........... 343
14.2. Nash equilibria .................................................................................. 345
14.3. Mixed extension of a game and Nash equilibria ............................... 347
14.4. Algorithmic problems ...................................................................... 348
  14.4.1. Games of succinct description .................................................. 349
  14.4.2. Results related to the computational complexity of a mixed
         equilibrium ............................................................................... 349
  14.4.3. Counting the number of equilibria in a mixed strategies
         game ......................................................................................... 355
14.5. Potential games ................................................................................ 355
  14.5.1. Definitions ................................................................................. 356
  14.5.2. Properties ................................................................................ 356
14.6. Congestion games ........................................................................... 360
  14.6.1. Rosenthal’s model ..................................................................... 360
  14.6.2. Complexity of congestion games (Rosenthal’s model) ............. 363
  14.6.3. Other models ........................................................................... 364
14.7. Final note ......................................................................................... 368
14.8. Bibliography .................................................................................... 368

Chapter 15. Flows! ................................................................................... 373
Michel KOSKAS and Cécile MURAT

15.1. Introduction ...................................................................................... 373
15.2. Definitions and notations ................................................................. 374
15.3. Presentation of radix trees ............................................................... 375
  15.3.1. Database management ............................................................. 377
  15.3.2. Image pattern recognition ......................................................... 379
  15.3.3. Automatic translation ............................................................... 380
15.4. Shortest path problem ..................................................................... 380
  15.4.1. Finding paths in $G[r]$ ................................................................. 382
  15.4.2. Complexity ................................................................................ 385
15.5. The Flow problem ........................................................................... 388
  15.5.1. The Ford-Fulkerson algorithm .................................................. 388
15.6. Conclusion ....................................................................................... 390
15.7. Bibliography .................................................................................... 391
Chapter 16. The Complexity of the Exact Weighted Independent Set Problem ................................................................................................................. 393
Martin MILANIĆ and Jérôme MONNOT

16.1. Introduction.............................................................................................. 393
16.2. Preliminary observations.......................................................................... 397
16.3. Hardness results ....................................................................................... 399
   16.3.1. Bipartite graphs .......................................................................... 400
   16.3.2. A more general hardness result.................................................. 403
16.4. Polynomial results.................................................................................... 405
   16.4.1. Dynamic programming solutions............................................... 406
   16.4.1.1. \( mK_2 \)-free graphs ........................................................... 406
   16.4.1.2. Interval graphs ..................................................................... 407
   16.4.1.3. \( k \)-thin graphs ................................................................ 408
   16.4.1.4. Circle graphs .................................................................. 409
   16.4.1.5. Chordal graphs ................................................................ 410
   16.4.1.6. AT-free graphs .................................................................. 413
   16.4.1.7. Distance-hereditary graphs ............................................. 415
   16.4.1.8. Graphs of treewidth at most \( k \) ....................................... 417
   16.4.1.9. Graphs of clique-width at most \( k \) ................................... 422
   16.4.2. Modular decomposition ............................................................. 424
16.5. Conclusion ............................................................................................... 428
16.6. Bibliography ............................................................................................ 429

Chapter 17. The Labeled Perfect Matching in Bipartite Graphs: Complexity and (in)Approximability.................................................................. 433
Jérôme MONNOT

17.1. Introduction.............................................................................................. 433
17.2. The 2-regular bipartite case...................................................................... 436
17.3. Some inapproximation results.................................................................. 438
17.4. The complete bipartite case...................................................................... 446
17.5. Bibliography ............................................................................................ 452

Chapter 18. Complexity and Approximation Results for Bounded-size Paths Packing Problems ....................................................................................... 455
Jérôme MONNOT and Sophie TOULOUSE

18.1. Introduction.............................................................................................. 455
   18.1.1. Bounded-size paths packing problems....................................... 455
   18.1.2. Complexity and approximability status ....................................... 457
   18.1.3. Theoretical framework, notations and organization..................... 458
18.2. Complexity of \( P_6 \)-PARTITION and related problems in bipartite graphs ........................................................................................................... 460
18.2.1. Negative results from the \( k \)-dimensional matching problem .... 460
18.2.1.1. \( k \)-dimensional matching problem ........................................ 460
18.2.1.2. Transforming an instance of \( k \)DM into an instance
of \( P_k \)PACKING ................................................................. 460
18.2.1.3. Analyzing the obtained instance of \( P_k \) PACKING ...... 462
18.2.1.4. \textbf{NP}-completeness and \textbf{APX}-hardness.................. 463
18.2.2. Positive results from the maximum independent set problem ... 466
18.3. Approximating MAX\( WP_3 \)PACKING and MIN3-PATHPARTITION... 466
18.3.1. MAX\( WP_3 \)PACKING in graphs of maximum degree 3 .......... 467
18.3.2. MAX\( WP_3 \)PACKING in bipartite graphs of maximum
degree 3 ...................................................................................... 470
18.3.3. MIN3-PATHPARTITION in general graphs ............................ 473
18.4. Standard and differential approximation of \( P_k \)P ................................... 475
18.4.1. Differential approximation of \( P_k \)P from the traveling salesman
problem ...................................................................................... 475
18.4.2. Approximating \( P_4 \)P by means of optimal matchings .............. 478
18.4.2.1. Description of the algorithm .............................................. 478
18.4.2.2. General \( P_4 \)P within the standard framework ................. 479
18.4.2.3. General \( P_4 \)P within the differential framework .............. 482
18.4.2.4. Bi-valued metric \( P_4 \)P with weights \( a \) and \( b \) within
the standard framework ......................................................... 483
18.4.2.5. Bi-valued metric \( P_4 \)P with weights \( a \) and \( b \) in the
differential framework ....................................................... 486
18.5. Conclusion ....................................................................................... 491
18.6. Bibliography .................................................................................... 492

Chapter 19. An Upper Bound for the Integer Quadratic Multi-knapsack
Problem .............................................................................................. 495
Dominique QUADRI, Eric SOUTIF, Pierre TOLLA

19.1. Introduction ................................................................................... 495
19.2. The algorithm of Djerdjour \textit{et al.} ............................................ 497
19.3. Improving the upper bound .......................................................... 498
19.4. An efficient heuristic to calculate a feasible solution ................. 500
19.5. Computational results ................................................................. 500
19.6. Conclusion .................................................................................... 503
19.7. Bibliography .................................................................................. 503

List of Authors .......................................................................................... 507

Index ........................................................................................................... 511
This page intentionally left blank
Preface

The LAMSADE\(^1\) was established in 1976 as a research laboratory jointly funded by Paris-Dauphine University and the CNRS (the French National Science Foundation) oriented to decision aiding, mainly in the areas of multiple criteria decision aiding and linear programming.

It very soon aggregated the research activities on computer science conducted within Paris-Dauphine University. In 30 years the LAMSADE gained a world-wide reputation in operations research and decision aiding, while developing and strengthening a specific vision of computer science, that is management and decision oriented computer science (from the French term “informatique décisionnelle”). Today the LAMSADE is one of the very few research laboratories showing such originality in its research orientation.

During these years new specific research subjects came to enrich those already existing: multi-agent systems, distributed computing and databases. In this effort, the LAMSADE had to put together different interdisciplinary competencies: decision theory, operations research, mathematics, social sciences and several fields of computer science. At the turning point of its 30 years the LAMSADE is organizing its research activities around four principal areas:

1) decision aiding;
2) optimization and its applications;
3) multi-agent and distributed systems;
4) database systems, information systems and knowledge management.

Under such perspective, the laboratory’s scientific project mainly aims to:

---

– consolidate and extend our international leadership in operations research and
decision aiding;
– strengthen and promote our vision of management and decision-oriented com-
puter science;
– create new large interfaces between operations research and theoretical computer
science.

In particular, research in the intersection of combinatorial optimization and theo-
etical computer science always remains a central key-point of LAMSADE’s research
activity.

Combinatorial optimization and theoretical computer science have been, and still
are, considered as two subjects different from each other. If the difference is quite
evident for some areas of both subjects, it becomes much less so if we think of areas
such as complexity theory, theory of algorithms, solving hard combinatorial problems,
graph theory and, more generally, discrete mathematics, etc. All these matters form a
very large interface between combinatorial optimization and theoretical computer
science. Historically, researchers in the areas mentioned above have been members of
two distinct major scientific communities, namely theoretical computer science and
operations research. They have addressed almost the same problems, worked under
the same paradigms, used parallel formalisms, tools and methods and developed ei-
ther parallel or complementary approaches. The fruits of this “separate research” have
impregnated the entire field of information technology and industry and almost the
whole of what is considered today as management sciences. Moreover, they have been
widespread over numerous scientific disciplines, initially orthogonal to both computer
science and combinatorial optimization, giving rise to new areas of research. However,
if from this “separate attack” we witnessed the emergence of practically all of the tra-
ditional concepts dealing with complexity theory, discrete mathematical modeling and
polynomial approximation of discrete optimization problems, numerous problems and
challenges remain open and without satisfying answers, thus the need for intensive re-
search in the interface of combinatorial optimization and theoretical computer science
becomes not only clear but also extremely challenging. This kind of research is one of
the major directions in the scientific project of the LAMSADE.

With such studies, we expect to advance in the research for new paradigms, get-
ing an insight mainly from the complex system sciences. I strongly believe that in the
near future, the themes of our research will be central to operational research and will
reshape the research landscape in combinatorial optimization. I also believe that they
will influence all the active research for new calculating machine paradigms based
upon properties of natural and human systems that are not exploited by conventional
computers, by providing them with new problems to deal with and new solutions to try
out. Our scientific project can thus be seen as an initiative to drastically renovate the re-
search agenda in combinatorial optimization, by addressing open and novel problems
arising from complex human systems. In order to achieve this objective, we have first to support a research environment that overcomes traditional cluster barriers among communities historically defined as “operations research” and “theoretical computer science”. We have also to work over the common basis of established theories and expertise for studying decidability, complexity, structure and solutions of hard optimization problems, which will definitely serve as the framework for validation of any advances in new research topics.

As stated above, bringing together operations research and theoretical computer science can be the first step in developing close synergies between all the complex systems disciplines, mainly those based upon the study of human systems. Research in the interface of these subjects is the main attempt to build such a broad alliance and to give it a clear scientific status. Moreover, by handling novel problems issued by still unexploited models and working hypotheses, we aim to strongly contribute to the emergence of a new paradigm for both combinatorial optimization, and algorithmic and complexity theory aspects of theoretical computer science.

The main objective of the book is to bear witness to the quality and the depth of the work conducted in the laboratory along the epistemological lines just outlined. In the chapters, the reader will find all the ingredients of a successful matching between combinatorial optimization and theoretical computer science, with interesting results carrying over a large number of their common subjects and going from “pure” complexity theoretical approaches dealing with concepts like \( \text{NP} \)- and \( \text{PSPACE} \)-completeness to “oldies but goodies” and always essential and vital operational research subjects such as flows, scheduling, or linear and mathematical programming, passing from polynomial approximation, online calculation, multicriteria combinatorial optimization, game theory, design of algorithms for multi-agent systems, etc. All of the chapters make a valuable contribution to both the two main topics of the book and any of the areas dealt.

In Chapter 1, Aloulou and Della Croce deal with single machine scheduling. They consider scheduling environments where some job characteristics are uncertain, this uncertainty being modeled through a finite set of well-defined scenarios. They search for a solution that is acceptable for any considered scenario using the “absolute robustness” criterion and present algorithmic and computational complexity results for several single machine scheduling problems.

Although the approximability of multi-criteria combinatorial problems has been the inspiration for numerous articles, the non-approximability of these problems seems to have never been investigated until now. Angel et al. in Chapter 2 propose a way to get some results of this kind that work for several problems. Then, they apply their method on a multi-criteria version of the traveling salesman problem in graphs with edge-distances one and two. Furthermore, they extend existing approximation results
for the bi-criteria traveling salesman problem in graphs with edge-weights 1 or 2 to any number \( k \) of criteria.

In Chapter 3, Ausiello et al. study online models for minimum set cover problem and minimum dominating set problem. For the former problem, the basic model implies that the elements of a ground set of size \( n \) arrive one-by-one; we assume that with any such element, arrives also the name of some set containing it and covering most of the still uncovered ground set-elements. For this model they analyze a simple greedy algorithm and prove that its competitive ratio is \( O(\sqrt{n}) \) and that it is asymptotically optimal for the model dealt. They finally deal with a new way to tackle online problems by using what they call “budget models”. For the case of the minimum set cover problem the model considered generates the so-called maximum budget saving problem, where an initial budget is allotted that is destined to cover the cost of an algorithm for solving set-covering and the objective is to maximize the savings on the initial budget.

In Chapter 4 by Bérard et al., Merlin-like time Petri nets (TPN) and timed automata (TA) are considered. The authors investigate questions related to expressiveness for these models: they study the impact of slight variations of semantics on TPN and compare the expressive power of TA and TPN with respect to both time language acceptance and weak time bisimilarity. On the one hand, they prove that TA and bounded TPNs (enlarged with strict constraints) are equivalent w.r.t. timed language equivalence, by providing an efficient construction of a TPN equivalent to a TA. On the other hand, they exhibit a TA such that no TPN (even unbounded) is weakly bisimilar to it. Motivated from this latter result, they characterize the subclass \( TA^- \) of TA that is equivalent to the original model of Merlin-like TPN and show that both the associated membership problem and the reachability problem for \( TA^- \) are \( \text{PSPACE} \)-complete.

Carello et al., in Chapter 5, introduce a graph problem which is called maximum node clustering. They prove that it is strongly \( \text{NP} \)-hard, but it can be approximated, in polynomial time, within a ratio arbitrarily close to 2. For the special case where the graph is a tree, they prove that the associated decision problem is weakly \( \text{NP} \)-complete as it generalizes the 0-1 knapsack problem and is solvable in pseudo-polynomial time by a dynamic programming approach. For this case they devise a fully polynomial time approximation schema for the original (optimization) problem.

In Chapter 6, Chevaleyre tackles the problem of multi-agent patrolling dealt with as a combinatorial optimization problem. More precisely, territory (one of the inputs of the problem) is modeled by means of a suitable edge-weighted graph \( G(V, E) \) and then the exploration strategies for this graph are based upon particular solutions of the traveling salesman problem. With this method, when the graph is metric, he obtains, in polynomial time, an exploration strategy with value bounded above by \( 3\text{opt}(G) + 4\max\{w(i, j) : (i, j) \in E\} \), where \( \text{opt}(G) \) is the value of the optimal exploration strategy and \( w(i, j) \) is the weight of the edge \( (i, j) \in E \). It is also proved that, using
another approach for the patrolling problem, based on a particular graph-partitioning problem, the multi-agent patrolling problem is approximable within approximation ratio 15, even in the case where the underlying graph is not metric.

In Chapter 7, Chevaleyre et al. investigate the properties of an abstract negotiation framework where, on the one hand, agents autonomously negotiate over allocations of discrete resources and, on the other hand, reaching an optimal allocation potentially requires very complex multilateral deals. Therefore, they are interested in identifying classes of utility functions such that, whenever all agents model their preferences using them, any negotiation conducted by means of deals involving only a single resource at a time is bound to converge to an optimal allocation. They show that the class of modular utility functions is not only sufficient (when side-payments are allowed) but is also maximal in this sense. A similar result is proved in the context of negotiation without money.

In Chapter 8, Della Croce et al. study two very well-known hard combinatorial problems, the maximum cut problem and the minimum dominating set restricted to graphs of maximum degree 3 (minimum 3-dominating set). For the former, they mainly focus on sparse graphs, i.e., on graphs having bounded maximum degree. They first use a technique based upon enumeration of cuts in a properly chosen subgraph of the input graph and then an extension of them in an optimal way to produce a cut for the whole instance. By means of this method they produce an exact algorithm for the weighted maximum cut problem with improved upper complexity bound in the case of sparse graphs. Next, they restrict themselves to the unweighted maximum cut problem in graphs of maximum degree 3 and devise a tree-search based exact algorithm. Exploiting some simple and intuitive dominance conditions that efficiently prune the search-tree, they provide a fairly competitive upper complexity bound for the case settled. Finally, they refine the search tree’s pruning by introducing a counting procedure, based upon the introduction of weights for the fixed data, which allows them to measure in a more precise way the progress made by the algorithm when it fixes them. They apply this method to min 3-dominating set.

In Chapter 9, Demange et al. study the computational complexity of online shunting problems. They consider a depot consisting of a set of parallel tracks. Each track can be approached from one side only and the number of trains per track is limited. The departure times of the trains are fixed according to a given timetable. The problem is to assign a track to each train as soon as it arrives to the depot and such that it can leave the depot on time without being blocked by any other train. They show how to solve this problem as an online bounded coloring problem on special graph classes. They also study the competitiveness of the first fit algorithm and show that it matches the competitive ratio of the problem.

Chapter 10, by Demange et al., surveys complexity and approximation results for the minimum weighted vertex coloring problem. This is a natural generalization of the
traditional minimum graph coloring problem obtained by assigning a strictly positive integer weight for any vertex of the input graph, and defining the weight of a color (independent set) as the maximum of the weights of its vertices. Then, the objective is to determine vertex coloring for the input graph minimizing the sum of the weights of the colors used. Complexity and approximation issues for this problem are presented for both general graphs and for graphs where the traditional minimum graph coloring problem is polynomial.

Chapter 11 is a complement of Chapter 10 where, along the same lines, complexity and approximation issues are addressed for the minimum weighted edge coloring problem where, instead of vertices, edges are now to be legally colored.

In Chapter 12, Gabrel considers the Dantzig-Wolfe decomposition for 0-1 linear programming when a subset of constraints defines a independent set polytope. She compares linear relaxations of both the initial and master program (obtained by decomposing on independent set constraints) with respect to various independent set polytope representations. For perfect graphs (in particular for co-comparability graphs), the linear relaxation of the master program is easy to solve while for general graphs its optimal value cannot be calculated in polynomial time. Consequently, she proposes to decompose only on a subset of the independent set constraints (those associated with “polynomial” independent set problems) in order to define another master program for which the LP-relaxation is easy to solve and remains stronger than the traditional LP-relaxation of the initial program.

In Chapter 13, Gabrel compares several 0-1 linear programs for solving the satellite mission planning problem. She considers two models and explains why one of them systematically calculates lower upper bounds. Her explanation is based upon independent set polytope formulations for perfect graphs. Then, she proposes new upper bounds for some large-size benchmark instances.

Chapter 14, by Giannakos et al., is a survey on some of the main results dealing with the problem of finding a Nash equilibrium in a game. After reporting several questions concerning complexity of general games (how many equilibria exist?, what are the conditions of the existence of an equilibrium verifying some given property?), the authors focus on games having pure Nash equilibria, as potential games and congestion games, for which they present several models.

In Chapter 15, entitled “Flows!”, Koskas and Murat give another novel interface between operational research and theoretical computer science by showing how tools from combinatorics of words can be very efficiently used in order to devise “divide and conquer” algorithms in a number of operational research and computer science fields, like database management, automatic translation, image pattern recognition, flow or shortest path problems, etc. The current contribution details one of them, dealing with maximum flow in a network.
Milanič and Monnot, in Chapter 16, introduce the exact weighted independent set problem, consisting of determining whether a weighted graph contains an independent set of a given weight. They determine the complexity of this problem as well as the complexity of its restricted version, where the independent set is required to be of maximum size, for several graph-classes. Furthermore, they show that these problems can be solved in pseudo-polynomial time for chordal graphs, AT-free graphs, distance-hereditary graphs, circle graphs, graphs of bounded clique-width, and several subclasses of $P_5$-free and fork-free graphs. Monnot, in Chapter 17, deals with complexity and approximability of the labeled perfect matching problem in bipartite graphs, as well as with minimum labeled matching and maximum labeled matching in 2-regular bipartite graphs, i.e., in collections of pairwise disjoint cycles of even length.

In Chapter 18, Monnot and Toulouse present several standard- and differential-approximation results for the $P_4$-partition problem for both minimization and maximization versions.

Finally, in Chapter 19, Quadri et al. present an improvement of a well-known method, based upon surrogate relaxation and linearization of the objective function, for calculating an upper bound of integer separable quadratic multi-knapsack and report computational experiments that seem to confirm the efficiency of their approach.

I think that all these contributions show the vitality and the originality of the research carried out by the LAMSADE. I do hope that the reader will really appreciate the depth and the richness of all the presented contributions.

To conclude, let me say once more that it is always a pleasure for me to work with Chantal, Sami and Raphael Menasce, Jon Lloyd and their colleagues at ISTE Ltd.
This page intentionally left blank
1.1. Introduction

This chapter deals with single machine scheduling problems where some job characteristics are uncertain. This uncertainty is described through a finite set $S$ of well-defined scenarios. We denote by $p_j^s$, $d_j^s$ and $w_j^s$, respectively, the processing time, the release date, the due date and the weight of job $j$ under scenario $s \in S$.

Consider a scheduling problem, denoted by $\alpha|\beta|\gamma$ according to Graham et al. notation [GRA 79]. Let $\Pi$ be the set of feasible schedules with respect to the problem constraints. For each scenario $s \in S$, we denote by $\text{OPT}(\alpha|\beta|\gamma, s)$ the problem of finding an optimal schedule $\pi^*_s$ satisfying:

$$F(\pi^*_s, s) = \min_{\pi \in \Pi} F(\pi, s).$$  \[1.1\]

When problem parameters are uncertain, it is appropriate to search for a solution that is acceptable for any considered scenario. For this purpose, several criteria can be applied in order to select among solutions. In [KOU 97], Kouvelis and Yu proposed three different robustness criteria: the absolute robustness or maximal cost, the maximal regret or robust deviation and the relative robustness. In this chapter, we focus on the absolute robustness criterion.
To the best of our knowledge, the absolute robustness in single machine scheduling problems has only been considered in [DAN 95] and [YAN 02] where, for the $1||\sum C_j$ problem with uncertain processing times, two distinct proofs of the NP-hardness even for $|S| = 2$ were provided (notice that the corresponding deterministic version is well known to be polynomially solvable [SMI 56]). The maximal regret criterion was instead much more studied (see, for instance, [YAN 02] but also [AVE 00, KAS 05, KOU 00]).

The absolute robustness of schedule $\pi$ over all scenarios $s \in S$ is denoted by $\bar{F}(\pi)$. We have:

$$\bar{F}(\pi) = \max_{s \in S} F(\pi, s)$$ [1.2]

We denote by $MinMax(\alpha|\beta|\gamma, \theta)$ the problem of finding a schedule $\pi^A$ minimizing the absolute robustness $\bar{F}(\pi)$ among all schedules $\pi \in \Pi$. Field $\theta$ indicates the set of uncertain problem parameters. For the problems considered here, $\theta \subseteq \{p_j, d_j, w_j\}$. Sequence $\pi^A$ is called in [KOU 97] absolute robust sequence. It satisfies:

$$\bar{F}(\pi^A) = \min_{\pi \in \Pi} \bar{F}(\pi) = \min_{\pi \in \Pi} \max_{s \in S} F(\pi, s)$$ [1.3]

Notice that if problem $\alpha|\beta|\gamma$ is NP-hard, then, in the presence of uncertainty, the corresponding problems $MinMax(\alpha|\beta|\gamma, \theta)$ are also NP-hard. However, if problem $\alpha|\beta|\gamma$ is polynomially solvable, then, the corresponding problems $MinMax(\alpha|\beta|\gamma, \theta)$ are not necessarily polynomially solvable. In this chapter we establish the complexity status for the absolute robustness versions of the most well known non-preemptive polynomial-time single machine scheduling problems, namely problems $1||\text{prec}|f_{\text{max}}$ (with $f_{\text{max}} \in \{C_{\text{max}}, L_{\text{max}}, T_{\text{max}}\}$), $1||\sum w_j C_j$ and $1||\sum U_j$. Notice that all these problems present regular cost functions which are non-decreasing in the job completion times. In this context any schedule $\pi \in \Pi$ is completely characterized by the corresponding job sequence. Given a schedule $\pi \in \Pi$, the completion time of job $j$ under scenario $s$, denoted by $C_j(\pi, s)$, $j = 1, \ldots, n, s \in S$, can easily be determined and the quality of the schedule $\pi \in \Pi$ under scenario $s$ is then evaluated using the regular cost function $F(\pi, s)$. We consider the following cost functions:

- the general maximum cost function $f_{\text{max}} = \max_j \{f_j(C_j)\}$ with $f_{\text{max}} \in \{C_{\text{max}}, L_{\text{max}}, T_{\text{max}}\}$: hence we deal with the maximum completion time (makespan) $C_{\text{max}} = \max_j \{C_j\}$, the maximum lateness $L_{\text{max}} = \max_j \{L_j\}$ with $L_j = C_j - d_j$ and the maximum tardiness $T_{\text{max}} = \max_j \{T_j\}$ with $T_j = \max\{0, L_j\}$;
- the total weighted completion time $\sum_j w_j C_j$;
– the number of late jobs \( \sum_j U_j \) with \( U_j = 0 \), if job \( j \) is on-time \((C_j \leq d_j)\) and \( U_j = 1 \) if \( j \) is late \((C_j > d_j)\).

Using the set \( S \) of scenarios, we construct a scenario \( s^w \) in which parameters \( k_j \) take their worst case value, denoted by \( k^w_j \). In our case, we have \( p^w_j = \max_{s \in S} p^s_j \), \( d^w_j = \min_{s \in S} d^s_j \) and \( w^w_j = \max_{s \in S} w^s_j \). Notice that in the context of a discrete set of scenarios, the constructed scenario is not necessarily feasible, i.e. we can have \( s^w \notin S \); \( s^w \) is called worst-case artificial scenario.

REMARK 1.1. – When parameters are interval-uncertain, \( s^w \) is a feasible scenario. In this case, an absolute robust solution \( \pi^A \) of problem \( \text{MinMax}(1|\beta|\gamma, \theta) \) is such that:

\[
\bar{F}(\pi^A) = \min_{\pi \in \Pi} \max_{s \in S} F(\pi, s) = \min_{\pi \in \Pi} \max_{s \in S} F(\pi, s^w) [1.4]
\]

Hence, \( \pi^A \) is also optimal for problem \( \text{OPT}(1|\beta|\gamma, s^w) \). This means that the problem of finding an absolute robust sequence can be solved straightforwardly by the algorithm solving the problem without uncertainty applied to the worst-case artificial scenario.

When uncertainty is scenario-based, we cannot apply the same reasoning because scenario \( s^w \) is not necessarily feasible. Nevertheless, we show in this chapter that problem \( \text{MinMax}(1|\text{prec}|f_{\text{max}},\theta) \) can be solved by Lawler’s algorithm applied to the worst case artificial scenario. We also prove that an extension of Lawler’s algorithm, called here \( \text{MinMax-Lawler} \), solves problem \( \text{MinMax}(1|\text{prec}|f_{\text{max}}, p_j, d_j) \) in polynomial time. On the other hand, problems \( \text{MinMax}(1|\sum w_j C_j, w_j) \) and \( \text{MinMax}(1|\sum U_j, p_j) \) are proved to be \( \text{NP} \)-hard even when \( |S| = 2 \). However, problem \( \text{MinMax}(1|\sum U_j, d_j) \) is still open.

Table 1.1 summarizes the above results presenting the complexity status for the absolute robustness versions of the most well known non-preemptive polynomial-time single machine scheduling problems, where an entry “-” indicates that the considered case does not apply (for instance, problem \( 1|\sum w_j C_j \) cannot have uncertainty on due dates as due dates are not present in the problem).

1.2. Problem \( \text{MinMax}(1|\text{prec}|f_{\text{max}}, \theta) \)

1.2.1. Uncertainty on due dates

We consider problem \( \text{MinMax}(1|\text{prec}|f_{\text{max}}, d_j) \) where processing times are deterministic and due dates are uncertain (here \( f_{\text{max}} \in \{L_{\text{max}}, T_{\text{max}}\} \) as for \( C_{\text{max}} \) no
uncertainty holds). In this case the worst-case artificial scenario $s^w$ is such that, for all $j \in N$, $d_j^w = \min_{s \in S} d_j^s$.

We recall that problem $1|\text{prec}|f_{\text{max}}$ can be solved in $O(n^2)$ time by Lawler’s algorithm [LAW 73]. This algorithm constructs an optimal schedule backwards. At the points in time where the unscheduled jobs should be completed, starting with point $t = P = \sum_{j \in N} p_j$, Lawler’s algorithm chooses among the unscheduled jobs having no successors a job with minimum cost to be completed at $t$. Notice that as processing times are deterministic, we have:

$$\forall \pi \in \Pi, \forall s \in S, \forall j \in N, C_j(\pi, s) = C_j(\pi)$$ [1.6]

The following theorem holds.

**Theorem 1.1.**— Problem $\text{MinMax}(1|\text{prec}|f_{\text{max}}, d_j)$ can be optimally solved in $O(n^2 + n|S|)$ time by means of Lawler’s algorithm applied to the worst-case artificial scenario $s^w$.

**Proof.** For the sake of clarity we consider that $f_{\text{max}} = L_{\text{max}}$, but the same analysis holds for $f_{\text{max}} = T_{\text{max}}$. An absolute robust solution $\pi^A$ of problem $\text{MinMax}(1|\text{prec}|L_{\text{max}}, d_j)$ is such that:

$$\bar{L}(\pi^A) = \min_{\pi \in \Pi} \max_{s \in S} L_{\text{max}}(\pi, s)$$ [1.7]

$$= \min_{\pi \in \Pi} \max_{s \in S} \max_{j \in N} (C_j(\pi, s) - d_j^s)$$ [1.8]

$$= \min_{\pi \in \Pi} \max_{s \in S} \max_{j \in N} (C_j(\pi) - d_j^s)$$ [1.9]

$$= \min_{\pi \in \Pi} \max_{j \in N} \max_{s \in S} (C_j(\pi) - d_j^s)$$ [1.10]

$$= \min_{\pi \in \Pi} \max_{j \in N} (C_j(\pi) - d_j^w)$$ [1.11]

$$\bar{L}(\pi^A) = \min_{\pi \in \Pi} L_{\text{max}}(\pi, s^w)$$ [1.12]
Hence, $\pi^A$ is also an optimal solution for problem $\text{OPT}(1|\text{prec}|L_{\max}, s^w)$. For the complexity, the construction of the worst-case scenario requires $O(n|S|)$ time and the application of Lawler’s algorithm requires $O(n^2)$ time, hence the overall complexity is $O(n^2 + n|S|)$.

We observe that the proof of Theorem 1.1 can be applied, as it is, to any scheduling problem $\alpha|\beta|f_{\max}$. Hence, we have the following result.

**Corollary 1.1.** Any algorithm optimally solving problem $\alpha|\beta|f_{\max}$ provides an absolute robust solution for problem $\text{MinMax}(\alpha|\beta|f_{\max}, d_j)$, when applied to the worst-case artificial scenario $s^w$.

### 1.2.2. Uncertainty on processing times and due dates

We consider problem $\text{MinMax}(1|\text{prec}|f_{\max}, p_j, d_j)$ where we suppose now that both processing times and due dates are uncertain. A robust solution $\pi^A$ is such that:

$$\bar{F}(\pi^A) = \min_{\pi \in \Pi} \max_{s \in S} F_{\max}(\pi, s) \quad [1.13]$$

$$= \min_{\pi \in \Pi} \max_{s \in S} \max_{j \in N} f_j(C_j(\pi, s)) \quad [1.14]$$

We propose an algorithm, called $\text{MinMax-Lawler}$, which is an extension of Lawler’s algorithm. This algorithm constructs a sequence $\pi$ in reverse order. Let $U$ be the set of unscheduled jobs. Define $p^s(U) = \sum_{j \in U} p_j^s$ for all $s \in S$. The rule is the following: schedule last the job $j \in U$, which has no successor in $U$ and such that $\max_{s \in S} f_j(p^s(U))$ is minimal. It is immediately clear that the complexity of $\text{MinMax-Lawler}$ is $O(n^2|S|)$.

We have the following result.

**Theorem 1.2.** Problem $\text{MinMax}(1|\text{prec}|f_{\max}, p_j, d_j)$ is optimally solved by algorithm $\text{MinMax-Lawler}$.

**Proof.** The proof is very similar to the proof of Lawler’s algorithm optimality for problem $1|\text{prec}|f_{\max}$.

Enumerate the jobs in such a way that $(1, 2, \ldots, n)$ is the sequence constructed by the proposed algorithm. Let $\pi^A$ be an absolute robust sequence for problem $\text{MinMax}(1|\text{prec}|f_{\max}, p_j, d_j)$ with $\pi^A(i) = i$ for $i = n, n - 1, \ldots, r$ and $\pi^A(r) = j < r$. 

Notice that it is possible to schedule \( r - 1 \) immediately before \( r \). Hence, we can construct a sequence \( \pi' \) in which we shift to the left the block between jobs \( r - 1 \) and \( r \) and process \( r - 1 \) immediately before \( r \). Clearly:

\[
\forall i \in N - \{ r - 1 \}, \forall s \in S, C_i(\pi', s) \leq C_i(\pi^A, s)
\] [1.15]

Hence:

\[
\bar{F}(\pi') = \max_{s \in S} F(\pi', s) = \max \{ \bar{F}(\pi^A), \max_{s \in S} f^s_{r-1}(C_{r-1}(\pi', s)) \} \] [1.16]

\[
= \max \{ \bar{F}(\pi^A), \max_{s \in S} f^s_{r-1}(C_j(\pi^A, s)) \} \] [1.17]

\[
\leq \max \{ \bar{F}(\pi^A), \max_{s \in S} f^s_j(C_j(\pi^A, s)) \} \] [1.18]

\[
\leq \bar{F}(\pi^A)
\] [1.19]

Consequently, \( \pi' \) is also an absolute robust sequence.

We can reiterate the same reasoning and transform sequence \( \pi^A \) into sequence \((1, 2, \ldots, n)\) without increasing the objective function value.

Correspondingly, the following corollary also holds.

**Corollary 1.2.** Problem \( \text{MinMax}(1|\text{prec}|f_{\text{max}}, p_j) \) is optimally solved by algorithm \( \text{MinMax-Lawler} \).

**1.3. Problem** \( \text{MinMax}(1||\sum w_j C_j, w_j) \)

We consider problem \( \text{MinMax}(1||\sum w_j C_j, w_j) \) where processing times are deterministic and weights are uncertain. We prove that this problem is \( \text{NP} \)-hard even when \(|S| = 2 \) and \( p_j = 1 \) \( \forall j \). To this extent we need to prove the following instrumental lemma.

**Lemma 1.1.** The \( 1||\sum C_j \) problem and the \( 1|p_j = 1||w_j C_j \) problem are equivalent.

**Proof.** Given any instance of the \( 1||\sum C_j \) problem where each job \( j \) has processing time \( p'_j \), generate an instance of the \( 1|p_j = 1||w_j C_j \) problem where each job \( j \) has weight \( w''_j = p'_{n-j+1} \). Consider a generic sequence \((1, 2, \ldots, n - 1, n)\). For the