

Optimisation Algorithms for Planning and Scheduling Workplace Training

by Olivér Gábor Czibula

Supervisors: Hanyu Gu, Feng-Jang Hwang, Mikhail Y. Kovalyov

A thesis submitted in partial fulfilment of the requirements for
the degree of Doctor of Philosophy

University of Technology, Sydney – 2017

CERTIFICATE OF ORIGINAL AUTHORSHIP

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This research is supported by an Australian Government Research Training Program Scholarship.

Signature of Student:

Date:

Abstract

This thesis is concerned with a number of related mathematical optimisation problems in, or closely related to, the fields of scheduling, timetabling, and rostering. The studies are motivated by real world problems routinely faced at an Australian electricity distributor. Due to the computational complexity of the problems considered and typical real-world problem sizes, solution by direct application of mathematical programming is not possible in practically acceptable time. Therefore, we propose a variety of heuristic, metaheuristic, and matheuristic approaches to obtain good quality solutions in acceptable time.

The first study is concerned with a large-scale class timetabling and trainer rostering problem. The problem is formulated as two Integer Programs: one for class timetabling and one for trainer rostering. A three-stage approach is presented, consisting of a timetable construction stage, a timetable improvement stage, and a trainer rostering stage.

The second study investigates a variation of the timetabling problem considered in the first study, but from an analytical perspective. The problem is presented in the context of batch scheduling. Conditions that lead to NP-hardness are shown, and a previously-known NP-hardness result is strengthened. A polynomial time algorithm is presented for a particular case. Simulated Annealing and Genetic Algorithm based metaheuristics are compared by means of extensive computational experimentation.

The third study is of a partitioning problem concerned with optimising the composition of study groups, which is related to, but distinct from, several well-known problems in the literature. The problem is shown to be NP-hard in the strong sense. Four approaches are proposed: the first is based on Lagrangian Relaxation, the second is based on Column Generation, the third encapsulates Column Generation within a fix-and-optimize Large Neighbourhood Search framework, and the fourth is a Genetic Algorithm amalgamated with Integer Programming.

The fourth study is concerned with the timetabling of practice placements. The problem is shown to be NP-hard in the strong sense. Two approaches are presented: the first approach improves an initial timetable by means of a Simulated Annealing metaheuristic, and the second approach constructs and improves a timetable by means of a fix-and-optimize Large Neighbourhood Search procedure.

The aim of this research is to study these related optimisation problems encountered at the electricity distributor from both analytical and practical perspectives, and to design successful solution approaches to them.

The research presented in this thesis has been published in several refereed publications [39, 37, 40, 41, 42, 38, 43].

Acknowledgements

I thank my wife, Anastasia, whose constant support and patience throughout my research made this work possible. I also extend my profound gratitude to my supervisors: Hanyu Gu, Feng-Jang Hwang, and Mikhail Y. Kovalyov, and to my advisor Yakov Zinder. All their encouragement, advice, experience, attention to detail, and expertise helped keep my work to a high standard. Finally, I thank Aaron J. Russell and Tom Emeleus for giving me this opportunity to work on these fascinating real-world problems.

To my son, Maxwell.

Contents

1	Introduction	1
1.1	Demand-Based Course Timetabling	2
1.2	Formation and Sequencing of Classes for a Bottleneck Classroom	3
1.3	Partitioning of Students into Classes	5
1.4	Timetabling of Practice Placements	6
1.5	Contributions	7
2	Literature Review	11
2.1	A Very Brief Introduction to Timetabling	12
2.2	Complexity of Timetabling	13
2.3	Demand-Based Course Timetabling	15
2.3.1	Rostering	18
2.4	Formation and Sequencing of Classes for a Bottleneck Classroom	18
2.5	Partitioning of Students into Classes	19
2.6	Timetabling of Practice Placements	23
3	Demand-Based Course Timetabling	25
3.1	Introduction	28
3.2	Optimisation Procedure	31
3.2.1	Stage 1: Initial timetable construction	32
3.2.2	Stage 2: Timetable improvement	34
3.2.3	Stage 3: Rostering	36
3.3	Timetabling Model	37
3.3.1	Time Discretisation	37
3.3.2	Input Data Set-up	38
3.3.3	List of Symbols	38
3.3.4	Core Timetabling Constraints	41

3.3.5	Characteristic Constraints	42
3.3.6	Trainer Movement Constraints	42
3.3.7	Resource Movement Constraints	43
3.3.8	Spreading Constraints	44
3.3.9	Objective Function	44
3.4	Rostering Model	46
3.5	Implementation	49
3.6	Computational Results	49
3.7	Conclusions	60
4	Formation and Sequencing of Classes for a Bottleneck Classroom	63
4.1	Introduction	65
4.2	Analytical Results	66
4.2.1	Complexity of ordered bi-criteria cases	67
4.2.2	Fixed batch order	71
4.3	Computational Approaches	72
4.3.1	Integer Programming Approach	72
4.3.2	Simulated Annealing Approach	74
4.3.3	Genetic Algorithm Approach	76
4.4	Computational Results	78
4.5	Conclusions	85
5	Partitioning of Students into Classes	87
5.1	Introduction	89
5.2	Quadratic Programming Formulation	90
5.3	NP-completeness	92
5.4	Lagrangian Relaxation	93
5.5	Lagrangian Heuristic	95
5.6	Column Generation Approaches	97
5.6.1	Reduced Master Heuristic	99
5.6.2	Fix Columns	99
5.6.3	Student Clustering	100
5.7	Large Neighbourhood Search with Column Generation Heuristics	101
5.8	Genetic Algorithm Based Matheuristic	102
5.9	Computational Results	105

5.10	Conclusions	118
6	Timetabling of Practice Placements	121
6.1	Introduction	123
6.2	Problem Formulation	124
6.3	Complexity of Finding a Feasible Solution	125
6.4	Constructive and Improvement Approach	127
6.4.1	Sequential Construction	128
6.4.2	Simulated Annealing	129
6.5	Fix-and-Optimise Large Neighbourhood Search Approach	129
6.6	Computational Results	131
6.7	Conclusions	139
7	Conclusions	141
7.1	Summary	141
7.1.1	Demand-Based Course Timetabling	141
7.1.2	Single-Machine Batch Scheduling with Incompatible Job Families and an Ordered Bi-Criteria Objective	142
7.1.3	Partitioning of Students into Classes	143
7.1.4	Timetabling of Practice Placements	143
7.2	Future Work	144
7.2.1	Demand-Based Course Timetabling	144
7.2.2	Single-Machine Batch Scheduling with Incompatible Job Families and an Ordered Bi-Criteria Objective	146
7.2.3	Partitioning of Students into to Classes	146
7.2.4	Timetabling of Practice Placements	147
	Bibliography	160

List of Figures

- 3.1 A high-level view of the three-stage approach. 31
- 3.2 A buffer added to either end of the reduced planning horizon. 33
- 3.3 An example of a timetable with unnecessary room swaps. 35
- 3.4 Unnecessary room swaps from the example in Figure 3.3 avoided. 35
- 3.5 A sample flow network for some resource t about period p with 2 locations. 44
- 3.6 A sample timetable, simplified for viewing in this format, showing 4 days, 3 locations,
and 7 courses each with 1 or 2 modules. 46
- 3.7 The min-cost flow network corresponding to the sample timetable shown in Figure 3.6.
(Home nodes are hatched, and activity nodes are solid) 47
- 3.8 Typical yearly training volume at the Australian electricity distributor. 50

- 4.1 The composition of the family j in B, corresponding to job j in A. 68
- 4.2 Assigning jobs to a fixed sequence of batches. 71
- 4.3 The average solve time in milliseconds for the test groups using Algorithm 1. 81
- 4.4 Convergence behaviour of SA and GA for test groups A, B, and C. 83
- 4.5 Convergence behaviour of SA and GA for the D test group. 84
- 4.6 Convergence behaviour of SA and GA for the E test group. 84
- 4.7 Convergence behaviour of SA and GA for the F test group. 85
- 4.8 Convergence behaviour of SA and GA for the G test group. 86

List of Tables

- 3.1 The number of variables (Cols) and constraints (Rows) for the timetabling IP model for each test case. 51
- 3.2 The amount of time, in seconds, the algorithm spent in Stage 1 (S1) and Stage 2 (S2) for each test case. 52
- 3.3 The objective value components for stages 1 and 2 for the test cases with low target timetable density. 53
- 3.4 The objective value components for stages 1 and 2 for the test cases with medium target timetable density. 54
- 3.5 The objective value components for stages 1 and 2 for the test cases with high target timetable density. 55
- 3.6 The amount of time, in seconds, the algorithm spent in Stage 3 (S3) for each test case. 56
- 3.7 The number of variables (Cols) and constraints (Rows) for the roster IP model, subject to the final generated timetable, for each test case. 57
- 3.8 The total trainer travel distance and number of trainer swaps for each roster produced. 58
- 3.9 Results related to the optimal solutions for some of the smaller test cases. 59

- 4.1 Outline of the 7 difficulty groups. 78
- 4.2 The SA parameters used across the 7 test groups. 80
- 4.3 The GA parameters used across the 7 test groups. 80
- 4.4 The probabilities π_k associated with swapping k pairs of batches in the neighbourhood function 80
- 4.5 The minimum, mean, and maximum reported time, in seconds, for CPLEX to find an optimal solution. 81
- 4.6 The performance of SA and GA with respect to the optimal or best-known solutions. . 82

- 5.1 The number of classes (Classes), number of students (Students), and density (Density) for the 27 test cases (Case). 106

5.2	The number of columns (Cols) and rows (Rows) for the QP model (QP) and LQP model (IP) for the 27 test cases (Cases).	107
5.3	The lower bound provided by the linear relaxation of LQP (LinRel), Lagrangian Relaxation (LagRel), and Column Generation (ColGen) for the 27 test cases (Case).	108
5.4	The time, in seconds, required to compute the lower bound by means of linear relation of LQP (LinRel), Lagrangian Relaxation (LagRel), and Column Generation (ColGen) for the 27 test cases (Case).	110
5.5	Objective values from the LR-based heuristic using 10% of students paired with and without large cliques (10% C) and (10%), 20% of students paired with and without large cliques (20% C) and (20%), and 30% of students paired with and without large cliques (30% C) and (30%) for the 27 test cases (Case).	111
5.6	Objective values from the Column Generation-based heuristic obtained by using the reduced master heuristic (MP), the column fixing (PF), and the student clustering (SC) methods for the 27 test cases (Case).	113
5.7	The time, in seconds, taken by the Column Generation-based heuristic obtained by using the reduced master heuristic (MP), the column fixing (PF), and the student clustering (SC) methods for the 27 test cases (Case).	114
5.8	Objective values from the fix-and-optimise LNS heuristic (LNS) using the student clustering methods for the 27 test cases (Case).	115
5.9	Minimum (Min), Average (Avg), and Maximum (Max) objective value for GA-based matheuristic with tournament sizes N , $5N$, and $10N$ for the 27 test cases (Case).	116
5.10	Poorest (Max), average (Avg), and best (Min) objective values from the LR-based heuristic (LR), the CG-based heuristic (CG), the GA-based matheuristic (GA), the LNS heuristic (LNS), and QP model (QP) for the 27 test cases (Case).	117
6.1	The number of apprentices, placement groups, and placements for the 36 test cases	132
6.2	The time taken, in minutes, for each tested procedure (1 of 2).	133
6.3	The time taken, in minutes, for each tested procedure (2 of 2).	134
6.4	$\sum_{j=1}^M \sum_{t=1}^T (U_{j,t} + V_{j,t})$ for solutions produced by each tested procedure.	136
6.5	$\sum_{i=1}^N \sum_{j=1}^M \sum_{t=1}^T c_{i,j} (X_{i,j,t} - Y_{i,j,t})$ for solutions with no bound violations produced by each tested procedure (1 of 2).	137
6.6	$\sum_{i=1}^N \sum_{j=1}^M \sum_{t=1}^T c_{i,j} (X_{i,j,t} - Y_{i,j,t})$ for solutions with no bound violations produced by each tested procedure (2 of 2).	138

List of Refereed Publications

- [39] O. Czibula, H. Gu, Y. Zinder, and A. Russell. A multi-stage IP-based heuristic for class timetabling and trainer rostering. In *International Conference of the Practice and Theory of Automated Timetabling*, 2014
- [37] O. Czibula, H. Gu, A. Russell, and Y. Zinder. A multi-stage IP-based heuristic for class timetabling and trainer rostering. *Annals of Operations Research*, pages 1–29, 2014
- [40] O. G. Czibula, H. Gu, F.-J. Hwang, M. Y. Kovalyov, and Y. Zinder. Bi-criteria sequencing of courses and formation of classes for a bottleneck classroom. *Computers & Operations Research*, 65:53–63, 2016
- [41] O. G. Czibula, H. Gu, and Y. Zinder. A Lagrangian relaxation-based heuristic to solve large extended graph partitioning problems. In *WALCOM: Algorithms and Computation*, pages 327–338. Springer, 2016
- [42] O. G. Czibula, H. Gu, and Y. Zinder. Scheduling personnel retraining: Column generation heuristics. In *International Symposium on Combinatorial Optimization*, pages 213–224. Springer, 2016
- [38] O. Czibula, H. Gu, and Y. Zinder. Timetabling of workplace training: A combination of mathematical programming and simulated annealing. In *International Conference of the Practice and Theory of Automated Timetabling*, 2016
- [43] O. G. Czibula, H. Gu, and Y. Zinder. Lagrangian relaxation versus genetic algorithm based matheuristic for a large partitioning problem. *Theoretical Computer Science*, 2017

