

**Scheduling Tasks with
Preemptions and Precedence
Constraints on Identical Parallel
Machines to Minimise the
Maximum Lateness**

by Samuel James Walker

Supervisor: Yakov Zinder

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

University of Technology, Sydney – 2016

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.

Signature of Student:

Date:

Many thanks to my supervisor Yakov Zinder, and to my parents.

Contents

Contents	4
List of Figures	6
List of Tables	7
List of Algorithms	8
Abstract	9
1 Introduction	11
1.1 Scheduling Theory	12
1.2 Problems Considered	15
1.3 Contributions of this Thesis	18
2 Literature Review	25
2.1 NP-Hardness	25
2.2 Polynomially Solvable Cases	29
2.3 Performance Guarantees	37
2.4 Branch and Bound	43
2.5 Order Theory	45
3 Computational Complexity	50
3.1 Classes of Partial Orders	50
3.2 Results	54
3.3 Proofs	57
4 Approximation Algorithms	63
4.1 Families of Algorithms	64
4.2 Scheduling Using Priorities	67
4.2.1 The Modified McNaughton Algorithm	69
4.2.2 Algorithm P	71

4.3	Mu Assignment Algorithms	79
4.3.1	The Brucker-Garey-Johnson Algorithm	79
4.3.2	The Garey-Johnson Algorithm	83
4.3.3	The Zinder-Roper Algorithm	89
4.3.4	The Optimal Recursive Algorithm	92
4.4	Schedule Structure	95
5	Performance Guarantees	101
5.1	The Brucker-Garey-Johnson Algorithm	102
5.1.1	Proving the Bound	103
5.1.2	Proving Tightness	105
5.2	The Garey-Johnson Algorithm	110
5.3	Dominance of Performance Guarantee	117
5.4	The Zinder-Roper Algorithm	138
5.4.1	Analysis of Successors	138
5.4.2	The Remainder of the Proof	148
6	The Domain of an Algorithm	154
6.1	Schedule Structure	155
6.2	The Brucker-Garey-Johnson Algorithm	157
6.2.1	The Class of Partial Orders	158
6.2.2	Proving Sufficiency	159
6.2.3	Proving Necessity	161
6.3	The Zinder-Roper Algorithm	167
6.3.1	The Class of Partial Orders	168
6.3.2	Recognising the Prohibited Substructure	170
6.3.3	Proving Sufficiency	172
6.3.4	Analysis of the Prohibited Substructure	178
6.3.5	Dominance of Solvable Cases	199
6.4	The Garey-Johnson Algorithm	230
6.4.1	The Class of Partial Orders	230
6.4.2	Proving Sufficiency	231
6.4.3	Proving Necessity	234
7	An Optimal Algorithm	257
7.1	The Linear Program	257
7.1.1	Total Weighted Tardiness	262
7.2	An Improvement	264

8 Integer Preemptions	268
8.1 Algorithm IP	270
8.1.1 The First Phase	272
8.1.2 The Second Phase	277
8.1.3 Methods for Assigning μ 's	285
8.2 Schedule Structure	285
8.3 Results	290
Conclusion	292
Summary	293
Future Research	296
Bibliography	298

List of Figures

1.1	The precedence constraints of an example instance.	17
3.1	Partial orders on three elements.	52
3.2	Partial orders on four elements.	53
4.1	Gantt chart of the schedule produced by the Brucker-Garey-Johnson Algorithm when applied to Example 1.1.	82
4.2	Gantt chart of the schedule produced by the Garey-Johnson Algo- rithm when applied to Example 1.1.	87
4.3	Gantt chart of the schedule produced by the Zinder-Roper Algorithm when applied to Example 1.1.	91
5.1	Precedence graph of φ	106
5.2	Precedence graph of φ^2	126
5.3	Precedence graph of φ^3	130
5.4	Precedence graph of φ^S	135
6.1	The two classes of prohibited subgraphs for \mathcal{D}^{BG}	158
6.2	The thirteen prohibited subgraphs of the first kind.	159
6.3	The seventeen prohibited subgraphs of the second kind that do not contain an induced subgraph of the first kind.	160
6.4	A partial order on four elements, describing a new class of partial orders.	171
6.5	Possible subgraphs induced by a singular pair and four independent tasks.	172
7.1	The new partial order $\hat{\rightarrow}$, where $A = \{5, 11\}$ and $B = \{6, 7, 14, 15\}$	267

List of Tables

1.1	The parameters of an example instance.	18
3.1	The four poset class families.	55
3.2	Computational complexity for $P Q, pmtn \gamma$ with $\gamma \in \{C_{max}, L_{max}\}$. . .	56
4.1	Table of completion times and latenesses produced by the Brucker-Garey-Johnson Algorithm when applied to Example 1.1.	83
4.2	Table of completion times and latenesses produced by the Garey-Johnson Algorithm when applied to Example 1.1.	88
4.3	Table of completion times and latenesses produced by the Zinder-Roper Algorithm when applied to Example 1.1.	92

List of Algorithms

McNaughton's Algorithm	29
The Modified McNaughton Algorithm	70
Algorithm P	73
The Brucker-Garey-Johnson Algorithm	80
The Garey-Johnson Algorithm	84
The Zinder-Roper Algorithm	89
The Optimal-Recursive Algorithm	93
The Brucker-Garey-Johnson Chain Algorithm	105
The Garey-Johnson Chain Algorithm	116
The Z-Algorithm	187
Algorithm IP	271
Algorithm IP – First Phase	275
Algorithm IP – Second Phase	281

Abstract

This work examines in depth a collection of related scheduling problems. The problems considered all relate to the processing of tasks on machines over time, with the goal of minimising some function of the tasks' completion times. The tasks being scheduled are subject to preemptions, that is they may be interrupted during their processing and resumed later on the same or a different set of machines. The tasks are subject to precedence constraints, meaning some tasks cannot be processed until others have been completed. The machines on which the tasks are processed are identical, meaning they have the same speed and functionality, and parallel, in contrast to scheduling models where tasks must be processed on multiple machines sequentially. This is a thoroughly investigated area of research in the literature.

Due to the computational complexity of the problems considered, approximation algorithms for these problems are presented and are analysed in several ways. Several well known results are expanded to apply to more general scheduling problems, and a completely new class of results is described. In particular, the solvable cases of some approximation algorithms – i.e. the situations under which the algorithms provide an optimal solution to the considered problem – are examined in more detail than heretofore. Bounds on the worst case of the considered algorithms are also presented, as are results regarding the computational complexity of certain subproblems.

The goal of this work is to advance the understanding of this focused but significant area of research, and present a wide variety of results relating to it.