

# Contents

List of Figures	ix
List of Tables	xiii
Abbreviations and Acronyms	xv
Abstract	xvii
<i>Resumen</i>	xix
<i>Resum</i>	xxi
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	2
1.1.1 Chip Multiprocessor . . . . .	2
1.1.2 Simultaneous Multithreading . . . . .	4
1.1.3 System Fairness . . . . .	6
1.2 Objectives of the Thesis . . . . .	7
1.3 Main Contributions of the Thesis . . . . .	8
1.4 Thesis Outline . . . . .	9
<b>2 Related Work</b>	<b>11</b>
2.1 Main Memory Contention . . . . .	12
2.2 Cache Hierarchy Contention . . . . .	13
2.2.1 Contention-Aware Scheduling . . . . .	13
2.2.2 Resource Partitioning . . . . .	15
2.2.3 Performance Models . . . . .	16
2.3 SMT Core Contention . . . . .	16
<b>3 Scheduling Framework, Experimental Platforms, and Evaluation Methodology</b>	<b>19</b>
3.1 Scheduling Framework . . . . .	20
3.2 Experimental Platforms . . . . .	21
3.2.1 Single-Threaded Multicore: Intel Xeon X3320 . . . . .	21
3.2.2 SMT Multicore: Intel Xeon E5645 . . . . .	23
3.2.3 IBM POWER8 System . . . . .	24
3.2.3.1 NUMA Effects in the IBM POWER8 . . . . .	25
3.3 Evaluation methodology . . . . .	27

3.3.1	Process Selection Methodology . . . . .	28
3.3.2	Process Allocation Methodology . . . . .	29
3.4	Metrics . . . . .	31
<b>4</b>	<b>Bandwidth-Aware Scheduling on Multicore Processors</b>	<b>35</b>
4.1	Performance Degradation Analysis . . . . .	36
4.1.1	Benchmarks Characterization . . . . .	36
4.1.2	Microbenchmark Design . . . . .	39
4.1.3	Degradation due to Main Memory Contention . . . . .	40
4.1.4	Degradation due to L2 Contention . . . . .	43
4.1.5	Degradation Running in Bandwidth-Aware Scheduling Scenarios . . . . .	44
4.2	Memory-Hierarchy Bandwidth-Aware Scheduling . . . . .	46
4.2.1	Baseline Main Memory Bandwidth-Aware Scheduler . . . . .	46
4.2.2	Memory-Hierarchy Bandwidth-Aware Scheduler . . . . .	47
4.2.3	IPC-Degradation Memory-Hierarchy Bandwidth-Aware Scheduler . . . . .	49
4.3	Evaluation Setup . . . . .	52
4.3.1	Evaluated Algorithms . . . . .	53
4.3.2	Mix Design . . . . .	53
4.4	Experimental Evaluation . . . . .	55
4.4.1	Performance Evaluation . . . . .	55
4.4.2	Profiling the Penalty Coefficient . . . . .	59
4.5	Summary . . . . .	60
<b>5</b>	<b>Bandwidth-Aware Scheduling in SMT Multicores</b>	<b>63</b>
5.1	Performance Degradation Analysis . . . . .	64
5.1.1	Effects of L1 Bandwidth on Performance . . . . .	64
5.1.1.1	Stand-Alone Execution . . . . .	64
5.1.1.2	Analyzing Interference between Co-Runners . . . . .	67
5.1.2	Impact of Cache Space Contention on L1 Bandwidth Consumption . . . . .	69
5.1.3	Performance Degradation due to Main Memory Bandwidth Contention . . . . .	71
5.2	SMT Bandwidth-Aware Scheduling . . . . .	72
5.2.1	Self-Reliant Main Memory Bandwidth-Aware Process Selection . . . . .	73
5.2.2	L1 Bandwidth-Aware Process Allocation . . . . .	76
5.2.2.1	Dynamic L1 Bandwidth-Aware Process Allocation Policy . . . . .	76
5.2.2.2	Static L1 Bandwidth-Aware Process Allocation Policy . . . . .	77
5.3	Evaluation Setup . . . . .	78
5.3.1	Evaluated Algorithms . . . . .	79
5.3.2	Mix Design . . . . .	80
5.4	Experimental Evaluation . . . . .	83
5.4.1	Evaluation of the Process Allocation Policies . . . . .	83
5.4.2	Evaluation of the Process Selection Policies . . . . .	87
5.4.3	Evaluation of the SMT Bandwidth-Aware Scheduler . . . . .	90
5.5	Summary . . . . .	94

<b>6</b>	<b>Progress-Aware Scheduling to Address Fairness in SMT Multicores</b>	<b>97</b>
6.1	Estimating Progress . . . . .	98
6.1.1	Period Length between IPC Estimates . . . . .	99
6.1.2	Process Interference in Low-Contention Schedules . . . . .	100
6.1.2.1	Interference between Pairs of Benchmarks . . . . .	101
6.1.2.2	Cumulative Interference in Low-Contention Schedules . . . . .	103
6.2	Progress-Aware Fair Scheduling . . . . .	104
6.2.1	IPC Estimation-Oriented Process Selection . . . . .	107
6.2.2	Fairness-Oriented Process Selection . . . . .	107
6.2.3	Process Allocation . . . . .	108
6.2.4	Implementation Considerations . . . . .	108
6.3	Progress-Aware Perf&Fair Scheduling . . . . .	109
6.3.1	IPC Estimation-Oriented Process Selection . . . . .	110
6.3.2	Performance- & Fairness- Oriented Process Selection . . . . .	111
6.4	Flexible Progress-Aware Perf&Fair Scheduling: Trading Fairness for Performance . . . . .	112
6.5	Evaluation Setup . . . . .	114
6.5.1	Evaluated Algorithms . . . . .	114
6.5.2	Mix Design . . . . .	115
6.6	Experimental Evaluation . . . . .	115
6.6.1	Evaluation of the Progress-Aware Fair Scheduler . . . . .	117
6.6.1.1	System Fairness Evaluation . . . . .	117
6.6.1.2	Accuracy of the Isolated IPC Estimations . . . . .	119
6.6.2	Evaluation of the Progress-Aware Perf&Fair Scheduler . . . . .	119
6.6.2.1	System Fairness Evaluation . . . . .	119
6.6.2.2	Performance Evaluation . . . . .	121
6.6.2.3	Process Completion in a Mix . . . . .	123
6.6.3	Evaluation of the Flexible Progress-Aware Perf&Fair Scheduler . . . . .	124
6.7	Summary . . . . .	126
<b>7</b>	<b>Symbiotic Job Scheduling on the IBM POWER8</b>	<b>129</b>
7.1	Predicting Job Symbiosis . . . . .	130
7.1.1	SMT Interference Model . . . . .	130
7.1.1.1	Base Component . . . . .	132
7.1.1.2	Resource Stall Components . . . . .	134
7.1.1.3	Miss Components . . . . .	134
7.1.2	Model Construction and Slowdown Estimation . . . . .	136
7.1.3	Obtaining tST CPI stacks in SMT mode . . . . .	138
7.2	SMT Interference-Aware Scheduler . . . . .	141
7.2.1	Reduction of the Cycle Stack Components . . . . .	141
7.2.2	Selection of the Optimal Schedule . . . . .	142
7.2.3	Scheduler Implementation . . . . .	144
7.3	Evaluation Setup . . . . .	145
7.3.1	Evaluated Algorithms . . . . .	146
7.3.2	Mix Design . . . . .	147
7.4	Experimental Evaluation . . . . .	147

---

7.4.1	Model Accuracy . . . . .	148
7.4.1.1	Regression Model Accuracy . . . . .	148
7.4.1.2	Inverse Model Accuracy . . . . .	150
7.4.2	Symbiotic Scheduler Evaluation . . . . .	151
7.4.2.1	System Throughput . . . . .	151
7.4.2.2	Per-Application Performance . . . . .	154
7.4.2.3	Symbiosis Patterns . . . . .	157
7.5	Summary . . . . .	158
<b>8</b>	<b>Conclusions</b>	<b>161</b>
8.1	Contributions . . . . .	162
8.2	Future Directions . . . . .	163
8.3	Publications . . . . .	166
	<b>References</b>	<b>169</b>