## CONTENTS

Preface xi  

Acknowledgments xiii  

1. Introduction 1  
1.1 Overview 1  
1.2 Organization 5  

2. Parallel Systems and Programming 7  
2.1 Parallel Architectures 7  
  2.1.1 Flynn’s Taxonomy 7  
  2.1.2 Memory Architectures 9  
  2.1.3 Programming Paradigms and Models 11  
2.2 Communication Networks 13  
  2.2.1 Static Networks 13  
  2.2.2 Dynamic Networks 18  
2.3 Parallelization 22  
2.4 Subtask Decomposition 24  
  2.4.1 Concurrency and Granularity 24  
  2.4.2 Decomposition Techniques 25  
  2.4.3 Computation Type and Program Formulation 27  
  2.4.4 Parallelization Techniques 28  
  2.4.5 Target Parallel System 28  
2.5 Dependence Analysis 29  
  2.5.1 Data Dependence 29  
  2.5.2 Data Dependence in Loops 32  
  2.5.3 Control Dependence 35  
2.6 Concluding Remarks 36  
2.7 Exercises 37  

3. Graph Representations 40  
3.1 Basic Graph Concepts 40  

COPYRIGHTED MATERIAL
CONTENTS

3.1 Computer Representation of Graphs 43
   3.1.1 Computer Representation of Graphs 43
   3.1.2 Elementary Graph Algorithms 46

3.2 Graph as a Program Model 49
   3.2.1 Computation and Communication Costs 50
   3.2.2 Comparison Criteria 50

3.3 Dependence Graph (DG) 51
   3.3.1 Iteration Dependence Graph 53
   3.3.2 Summary 55

3.4 Flow Graph (FG) 56
   3.4.1 Data-Driven Execution Model 60
   3.4.2 Summary 61

3.5 Task Graph (DAG) 58
   3.5.1 Graph Transformations and Conversions 64
   3.5.2 Motivations and Limitations 68
   3.5.3 Summary 69

3.6 Concluding Remarks 69
3.7 Exercises 70

4. Task Scheduling 74
   4.1 Fundamentals 74

4.2 With Communication Costs 76
   4.2.1 Schedule Example 81
   4.2.2 Scheduling Complexity 82

4.3 Without Communication Costs 86
   4.3.1 Schedule Example 87
   4.3.2 Scheduling Complexity 88

4.4 Task Graph Properties 92
   4.4.1 Critical Path 93
   4.4.2 Node Levels 95
   4.4.3 Granularity 101

4.5 Concluding Remarks 105
4.6 Exercises 105

5. Fundamental Heuristics 108
   5.1 List Scheduling 108
      5.1.1 Start Time Minimization 111
      5.1.2 With Dynamic Priorities 114
      5.1.3 Node Priorities 115
   5.2 Scheduling with Given Processor Allocation 118
      5.2.1 Phase Two 119
CONTENTS

5.3 Clustering
   5.3.1 Clustering Algorithms
   5.3.2 Linear Clustering
   5.3.3 Single Edge Clustering
   5.3.4 List Scheduling as Clustering
   5.3.5 Other Algorithms

5.4 From Clustering to Scheduling
   5.4.1 Assigning Clusters to Processors
   5.4.2 Scheduling on Processors

5.5 Concluding Remarks

5.6 Exercises

6. Advanced Task Scheduling
   6.1 Insertion Technique
      6.1.1 List Scheduling with Node Insertion
   6.2 Node Duplication
      6.2.1 Node Duplication Heuristics
   6.3 Heterogeneous Processors
      6.3.1 Scheduling

6.4 Complexity Results
   6.4.1 $\alpha/\beta/\gamma$ Classification
   6.4.2 Without Communication Costs
   6.4.3 With Communication Costs
   6.4.4 With Node Duplication
   6.4.5 Heterogeneous Processors

6.5 Genetic Algorithms
   6.5.1 Basics
   6.5.2 Chromosomes
   6.5.3 Reproduction
   6.5.4 Selection, Complexity, and Flexibility

6.6 Concluding Remarks

6.7 Exercises

7. Communication Contention in Scheduling
   7.1 Contention Awareness
      7.1.1 End-Point Contention
      7.1.2 Network Contention
      7.1.3 Integrating End-Point and Network Contention
   7.2 Network Model
      7.2.1 Topology Graph
      7.2.2 Routing
      7.2.3 Scheduling Network Model
CONTENTS

7.3 Edge Scheduling 203
  7.3.1 Scheduling Edge on Route 204
  7.3.2 The Edge Scheduling 208
7.4 Contention Aware Scheduling 209
  7.4.1 Basics 209
  7.4.2 NP-Completeness 211
7.5 Heuristics 216
  7.5.1 List Scheduling 216
  7.5.2 Priority Schemes—Task Graph Properties 219
  7.5.3 Clustering 220
  7.5.4 Experimental Results 221
7.6 Concluding Remarks 223
7.7 Exercises 224

8. Processor Involvement in Communication 228
  8.1 Processor Involvement—Types and Characteristics 229
    8.1.1 Involvement Types 229
    8.1.2 Involvement Characteristics 232
    8.1.3 Relation to LogP and Its Variants 236
  8.2 Involvement Scheduling 238
    8.2.1 Scheduling Edges on the Processors 240
    8.2.2 Node and Edge Scheduling 246
    8.2.3 Task Graph 247
    8.2.4 NP-Completeness 248
  8.3 Algorithmic Approaches 250
    8.3.1 Direct Scheduling 251
    8.3.2 Scheduling with Given Processor Allocation 254
  8.4 Heuristics 257
    8.4.1 List Scheduling 257
    8.4.2 Two-Phase Heuristics 261
    8.4.3 Experimental Results 263
  8.5 Concluding Remarks 264
  8.6 Exercises 265

Bibliography 269

Author Index 281

Subject Index 285