# TABLE OF CONTENTS

List of Tables
List of Figures
List of Publications

## CHAPTER-1 : INTRODUCTION

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Parallel Computing Distributed System</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Traditional Scheduling verses Parallel Computing Scheduling</td>
<td>2</td>
</tr>
<tr>
<td>1.2.1</td>
<td>Task partitioning small constraints in parallel computing</td>
<td>2</td>
</tr>
<tr>
<td>1.2.2</td>
<td>Optimization criteria for parallel computing scheduling</td>
<td>2</td>
</tr>
<tr>
<td>1.2.3</td>
<td>Data scheduling</td>
<td>3</td>
</tr>
<tr>
<td>1.2.4</td>
<td>Methodologies</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Types of Parallel Computing Architectures</td>
<td>3</td>
</tr>
<tr>
<td>1.3.1</td>
<td>Flynn’s taxonomy</td>
<td>3</td>
</tr>
<tr>
<td>1.3.1.1</td>
<td>SISD (Single Instruction stream/Single Data stream)</td>
<td>3</td>
</tr>
<tr>
<td>1.3.1.2</td>
<td>SIMD (Single Instruction stream/ Multiple Data stream)</td>
<td>3</td>
</tr>
<tr>
<td>1.3.1.3</td>
<td>MISD (Multiple Instruction /Single Data)</td>
<td>4</td>
</tr>
<tr>
<td>1.3.1.4</td>
<td>MIMD (Multiple Instructions and Multiple Data)</td>
<td>4</td>
</tr>
<tr>
<td>1.4</td>
<td>Important Laws in Parallel Computing</td>
<td>4</td>
</tr>
<tr>
<td>1.4.1</td>
<td>Amdahl's law</td>
<td>4</td>
</tr>
<tr>
<td>1.4.1.1</td>
<td>Limitations of Amdahl's Law</td>
<td>4</td>
</tr>
<tr>
<td>1.4.2</td>
<td>Gustafson's law</td>
<td>6</td>
</tr>
<tr>
<td>1.5</td>
<td>Performance Parameters in Parallel Computing</td>
<td>7</td>
</tr>
<tr>
<td>1.5.1</td>
<td>Throughput</td>
<td>7</td>
</tr>
<tr>
<td>1.5.2</td>
<td>System utilization</td>
<td>7</td>
</tr>
<tr>
<td>1.5.3</td>
<td>Turnaround time</td>
<td>7</td>
</tr>
<tr>
<td>1.5.4</td>
<td>Waiting time</td>
<td>7</td>
</tr>
<tr>
<td>1.5.5</td>
<td>Response time</td>
<td>7</td>
</tr>
<tr>
<td>1.5.6</td>
<td>Reliability</td>
<td>7</td>
</tr>
</tbody>
</table>
1.6 Issues and Challenges in Parallel Computing
1.6.1 Role of server system
1.6.2 Allocation requirements
1.6.3 Resource management
1.6.4 Security
1.6.5 Reliability
1.6.6 Fault tolerance
1.7 NP Hard Scheduling
1.7.1 Classes P and NP in parallel computing
1.8 Thread in Parallel Processing Algorithms
1.8.1 Thread benefits
1.8.1.1 Responsiveness
1.8.1.2 Economy and resource sharing
1.8.1.3 Utilization of multiprocessor architectures
1.9 DAG in Different Scheduling Environments
Reference

CHAPTER-2: COMPARATIVE STUDY OF PARALLEL COMPUTING TOOLS

2.1 Introduction
2.2 PVM (Parallel Virtual Machine)
2.2.1 Evolution of PVM
2.2.2 Portability, interoperability and evolution of PVM
2.2.3 Computing model of PVM
2.3 MPI (Message Passing Interface)
2.3.1 Evolution of MPI
2.3.2 Portability, interoperability and fault tolerance of MPI
2.3.3 Implementation of MPI
2.4 BLACS
2.4.1 ID-Less communication in BLACS
2.5 ScaLAPACK
2.5.1 Portability and scalability of ScaLAPACK
2.5.2 Software component of ScaLAPACK 27
2.6 Parallel Virtual Message Passing Interface 27
2.7 Parallel MATLAB 28
2.8 Compute Unified Device Architecture 29
  2.8.1 NVIDIA tesla GPU architecture 29
  2.8.2 Thread organization and scheduling of CUDA 30
2.9 Comparison of parallel computing tools 30
Reference 32

CHAPTER 3 : CLASSIFICATION OF TASK PARTITIONING 35-49
STRATEGIES IN DISTRIBUTED PARALLEL
COMPUTING SYSTEMS

3.1 Introduction 36
3.2 Deterministic Task Partitioning Strategies 37
  3.2.1 Papadimitriou and Yannakakis scheduling 38
  3.2.2 Linear Clustering with Task Duplication Scheduling 38
  3.2.3 Edge Zeroing Scheduling 38
  3.2.4 Modified Critical Path Scheduling 39
  3.2.5 Earliest Time First Scheduling 39
3.3 Dynamic Task Partitioning Strategies 39
  3.3.1 Evolutionary Task Scheduling 40
  3.3.2 Dynamic Priority Scheduling 40
3.4 Preemptive Task Partitioning Strategies 40
  3.4.1 Rate Monotonic-Decrease Utilization-Next Fit Scheduling 41
  3.4.2 Optimal Preemptive Scheduling 41
  3.4.3 Preemption Threshold Scheduling 41
  3.4.4 Fixed Preemption Point Scheduling 42
3.5 Non-Preemptive Partitioning Strategies 42
  3.5.1 Multiple Strict Bound Constraints scheduling 42
  3.5.2 Earliest Deadline First scheduling 42
Reference 46
# CHAPTER 4: OPTIMAL TASK PARTITIONING MODEL IN DISTRIBUTED HETEROGENEOUS PARALLEL COMPUTING ENVIRONMENT

4.1 Introduction 51  
4.2 PRAM Model 52  
4.3 Proposed Model for Task Partitioning in Distributed Environment 53  
  4.3.1 Proposed algorithm for inter-process communication amongst tasks 56  
  4.3.2 Pseudo code for the proposed algorithm 56  
  4.3.3 Low communication overhead phase 57  
  4.3.4 Priority assignment and start time computing phase 57  
  4.3.5 Procedure for computing the ALAP 58  
4.4 Experimental Phase 59  
Reference 62

# CHAPTER 5: OPTIMAL TASK PARTITIONING STRATEGY WITH DUPLICATION (OTPSD) IN PARALLEL COMPUTING ENVIRONMENTS

5.1 Introduction 65  
5.2 MCP Algorithm 66  
  5.2.1 Design of MCP algorithm 67  
  5.2.2 Efficiency of the algorithm 67  
5.3 HEFT(Heterogeneous Earliest Finish Time algorithm) Algorithm 67  
5.4 Performance Metric for Simulation 68  
5.5 Proposed Model of Task Partitioning Strategies 69  
  5.5.1 Task and processors assignment phase 69  
  5.5.2 Pseudo Code for Grain Pack SubDAG 70  
  5.5.3 Pseudo code for proposed algorithm 73  
5.6 OTPSD Implementation Phase 74  
Reference 80