CONTENTS

Abstract.................................................................................................................................................. v

List of Publications..................................................................................................................................... vi

List of Figures........................................................................................................................................... vii

List of Tables............................................................................................................................................ x

1. Introduction........................................................................................................................................... 1
   1.1. Contributions of Dissertation
       1.1.1. Multi-level Cache System for Multi-core Processors ("LogN+1" and "LogN" Cache Models) 2
       1.1.2. Multi-level Cache Simulator for Multi-core processors ("MCSMC") 3
       1.1.3. Multi-threaded Parallel Programming Model for Multi-core processors ("SPC³ PM") 3
   1.2. The Thesis Organization

2. Motivation and Challenges with Multi-Core Processors....................................................................... 5
   2.1. Architectural Challenges
       2.1.1. Memory Hierarchy
           2.1.1.1. Cache Levels
           2.1.1.2. Synchronization
           2.1.1.3. False Sharing
           2.1.1.4. Spinning
           2.1.1.5. Communication Minimization
   2.1.2. Architectural Support for Compilers / Programming Models
   2.2. Software Challenges
       2.2.1. Parallel Programming Models
       2.2.2. Parallel Algorithm Models
           2.2.2.1. Data Parallel Models
           2.2.2.2. Task Graph Model
           2.2.2.3. Work Pool Model
           2.2.2.4. Master-Slave Model
           2.2.2.5. Pipeline or Producer-Consumer Model
       2.2.3. Decomposition Techniques
           2.2.3.1. Recursive Decomposition
           2.2.3.2. Data Decomposition
           2.2.3.3. Exploratory Decomposition
           2.2.3.4. Speculative Decomposition
       2.2.4. Levels of Parallelism
       2.2.5. Compiler Optimization
           2.2.5.1. Parallelism
           2.2.5.2. Removal of Data Dependencies
5.2.2.5. Results Generation
5.2.3. Serial / Parallel Execution of MCSMC
5.2.4. Comparison with CACTI Cache Simulator
5.3. Performance Evolution
5.3.1. Simulation Environment
5.3.2. Result Analysis
5.4. Summary

6. \textit{SPC}^3 PM; A Multithreaded Parallel Software Development Environment for Multi-Core Processors

6.1. Currently Available Parallel Programming Tools
6.1.1. Commercially Available Multi-Core Application Development Aids
6.1.1.1. Intel's Multi-Core Application Development Aids
6.1.1.2. Microsoft's Multi-Core Application Development Aids
6.1.1.3. Sun's Multi-Core Application Development Aids
6.1.1.4. Other Commercial Multi-Core Application Development Aids
6.1.2. Other Standard Shared Memory Programming Approaches Use for Multi-core processors
6.1.2.1. Erlang
6.1.2.2. POSIX Thread (Pthreads)
6.1.2.3. OpenMP
6.1.3. Research Oriented Multi-Core Application Development Tools
6.1.4. Current Multi-Core Research Groups
6.1.5. Summary

6.2. Key Features of SPC$^3$ PM
6.3. Design Concepts
6.3.1. Design Issues with Multi-Core Programming
6.3.2. Task Based Parallelism
6.3.3. Thread Level Parallelism
6.3.4. Decomposition Techniques
6.3.5. Task Scheduling
6.3.6. Execution Modes
6.3.7. Types of Problem Supported
6.3.8. Data Sharing
6.3.9. Compilation

6.4. Programming with SPC$^3$ PM
6.4.1. Rules for Task Decomposition
6.4.2. Properties of a Task
6.4.3. Program Structure
6.4.4. SPC$^3$ PM Library
6.4.4.1. Serial Function
6.4.4.2. Parallel Function
6.4.4.3. Concurrent Function

6.5. Performance Evolution
6.5.1. Matrix Multiplication Algorithm
6.5.2. Serial Function
6.5.3. Parallel Function
6.5.4. Concurrent Function

6.6. Summary
7. Solving Travelling Salesman Problem using SPC3 PM

7.1. Travelling Salesman Problem (TSP)
    7.1.1. TSP applications
    7.1.2. TSP solutions
        7.1.2.1. Exact Algorithms
        7.1.2.2. TSP Heuristics
        7.1.2.3. Meta-Heuristics
        7.1.2.4. Hyper-Heuristics

7.2. Lin-Kernighan Heuristic
    7.2.1. Basic Lin-Kernighan Heuristic Algorithm (LKH)
    7.2.2. Modified Lin-Kernighan Heuristic Algorithm (LKH-1)
    7.2.3. Lin-Kernighan Heuristic Algorithm with General k-opt Sub-move (LKH2)

7.3. LKH-2 Software
    7.3.1. Execution of LKH-2 Software
    7.3.2. Flow Chart for LKH-2 Software Processing

7.4. Parallelization of LKH-2 Software using SPC3 PM
    7.4.1. Flow Chart for Parallel LKH-2 Software Processing Parallelized using SPC3 PM

7.5. Performance Evaluation
    7.5.1. TSP Library (TSPLIB)
    7.5.2. Result Analysis

7.6. Summary

8. Conclusions and Future Work

8.1. Summary
8.2. Future work

Appendix A: List of TSP instances in TSPLIB

References