Parallel Programming COMP5112

Course Introduction

Course Background

- A CSE 5000-level PG course
  - First offered last year as a CSE non-core PG course
  - BECOMES a CSE PG CORE course this time!
    - A WRITTEN FINAL EXAM
  - Assume CSE basic programming, OS, Algorithms
  - Reference books and structured lectures
  - Parallel programming knowledge and PRACTICE
  - PG level with a little research flavor?

Lecture Time and Venue

- Wed/Fri at 1:30-2:50PM
- Default makeup class time: Mon at 1:30-2:50PM
- Rm 2304, Lift 17-18
  - Subject to change
  - Auditing or sitting in welcome
- Feb 6 Mon - May 5 Fri except Apr 12,14, May 3
  - Feb 6, 8, 10, 13, 15, 17, 22, 24
  - Mar 1, 3, 8, 10, 15, 17, 22, 24, 29, 31
  - Apr 5, 7, 19, 21, 26, 28
  - May 5

Workload & Assessment

- Tentative plan
  - Project 50%, Final Exam 50%
- Your suggestion?

Course Topics

- Introduction to parallel computer architectures
- Principles of parallel algorithm design
- Shared-memory programming models
- Message passing programming models
- Data-parallel programming models for GPUs
- Case studies of parallel algorithms, systems, and applications
- Hands-on experience with writing parallel programs for tasks of interest

Parallel Computer Architectures

- Review on OS and Computer Architecture
  - The von Neumann architecture
  - Processes, multitasking, and threads
  - Modifications to the von Neumann Model
    - Caches
    - Virtual memory
    - Instruction-level parallelism
    - Hardware multithreading
- Parallel Hardware
  - SIMD systems
  - MIMD systems
  - Interconnection networks
  - Cache coherence
  - Shared-memory versus distributed-memory

Principles of parallel algorithm design

- Preliminaries
  - Decomposition, Tasks, and Dependency Graphs
  - Granularity, Concurrency, and Task-Interaction
  - Processes and Mapping
- Decomposition Techniques
- Mapping Techniques for Load Balancing
- Methods for Containing Interaction Overheads
- Parallel Algorithm Models

Shared-memory programming models

- Pthreads
  - Critical sections, busy-waiting, mutexes
  - Producer-Consumer Synchronization and Semaphores
  - Barriers and Condition Variables
  - Read-Write Locks
  - Caches, Cache Coherence, and False Sharing
  - Thread safety
- OpenMP

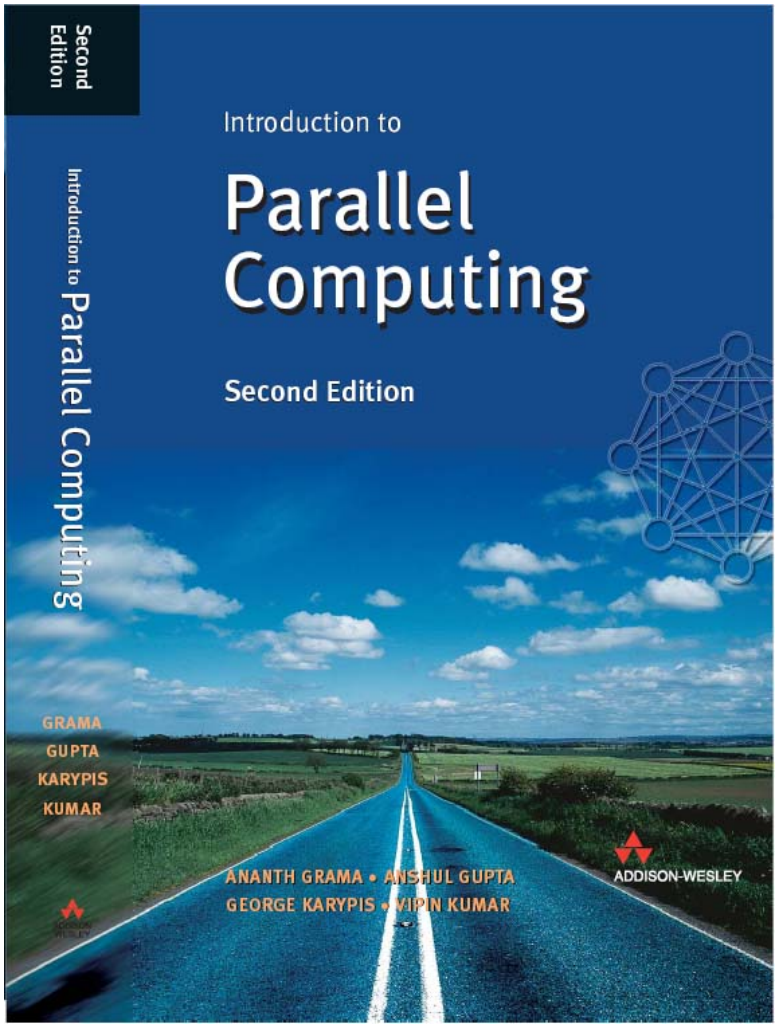
Message passing programming models

- Principles of Message-Passing Programming
- Building Blocks: Send and Receive Operations
- MPI: the Message Passing Interface
- Collective Communication and Computation Operations
  - Gather, Scatter, Prefix, Reduction, Broadcast, Barrier, and so on

Data-parallel programming models for GPUs

- NVIDIA GPU Architecture and CUDA C
- Single-Instruction Multiple-Threads Model
- CUDA Memories
- Performance Considerations
- Parallel Patterns
  - Gather, Scatter, Reduction, Prefix Scan, and so on
- Case Studies

Reference Book 1



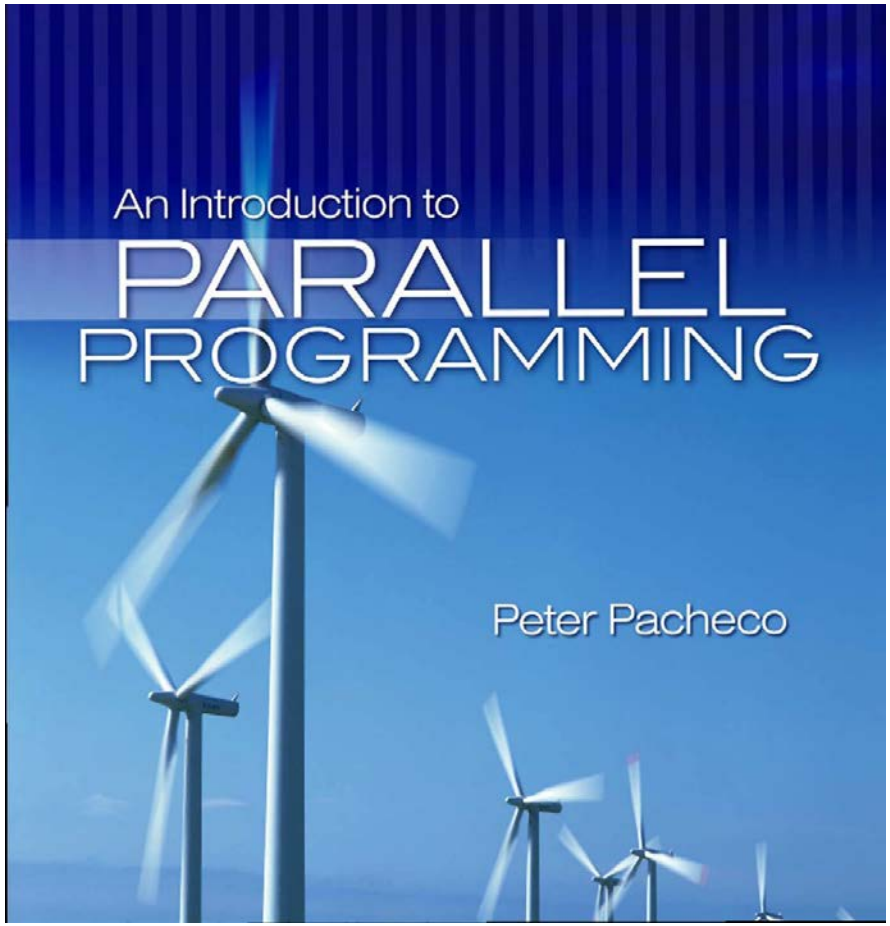
Introduction to Parallel Computing 2nd edition

By Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar.

Addison Wesley, 2003.

http://www-users.cs.umn.edu/~karypis/parbook/

Reference Book 2



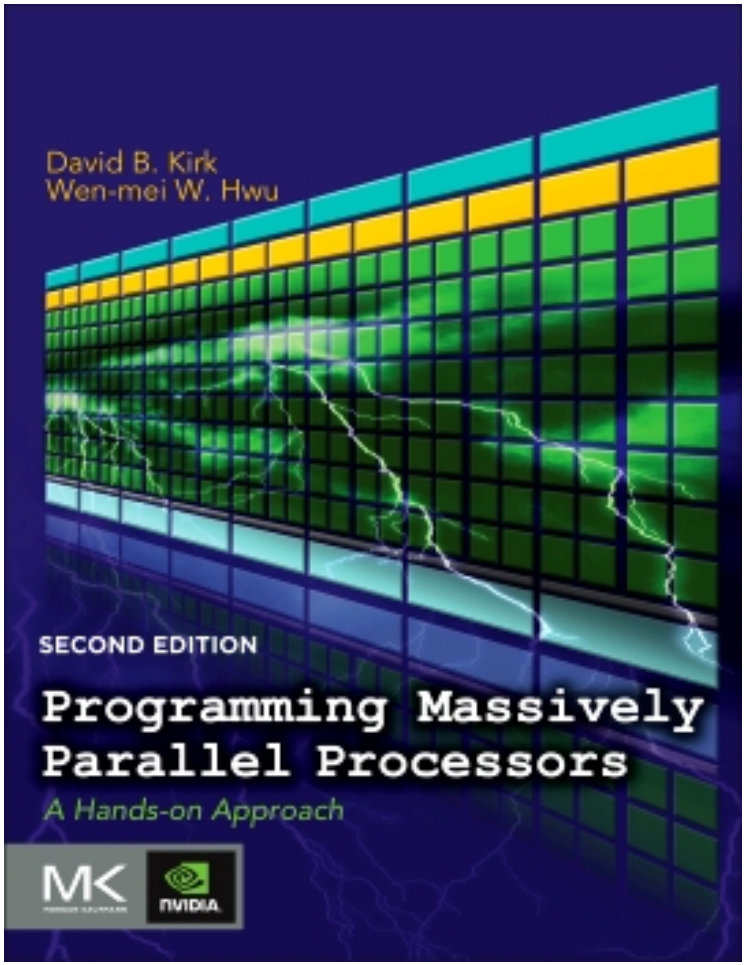
An Introduction to Parallel Programming

By Peter Pacheco

Morgan Kaufmann, 2011

https://www.cs.usfca.edu/~peter/ipp/

Reference Book 3



Programming Massively Parallel Processors:
A Hands-on Approach
2nd Edition
Author(s) : Kirk & Hwu
2013
Morgan Kaufmann

Full-text Online Access:

http://library.ust.hk/cgi/db/24x7.pl?id/51033

Computer Facilities

- Any computers within your access
- CSE Teaching Labs 1-4
- CSE Computer Clusters
  - http://cssystem.cse.ust.hk/Facilities/index.html

Feedback from You

- Optional: Name, Program, Year, Contact
- Enrollment status: enrolled/not enrolled
- Level of interest in the course
  - Will definitely drop/will definitely take/likely to take/likely to drop/not sure
- Background
  - C/C++ programming, OS, Algorithms, Architecture
- Access to computing facilities
- Any other comments
  - workload, assessment, topics, and so on