Parallel Programming MSBD5009

Course Introduction

Course Background

- An MSBD ELECTIVE
  - Assume CSE basic programming, OS, Algorithms
  - Structured lectures based on reference books
  - Teach parallel programming knowledge
  - Practice parallel programming in three languages
  - Workload
    - Three assignments + final exam
  - Exclusion COMP5112

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

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

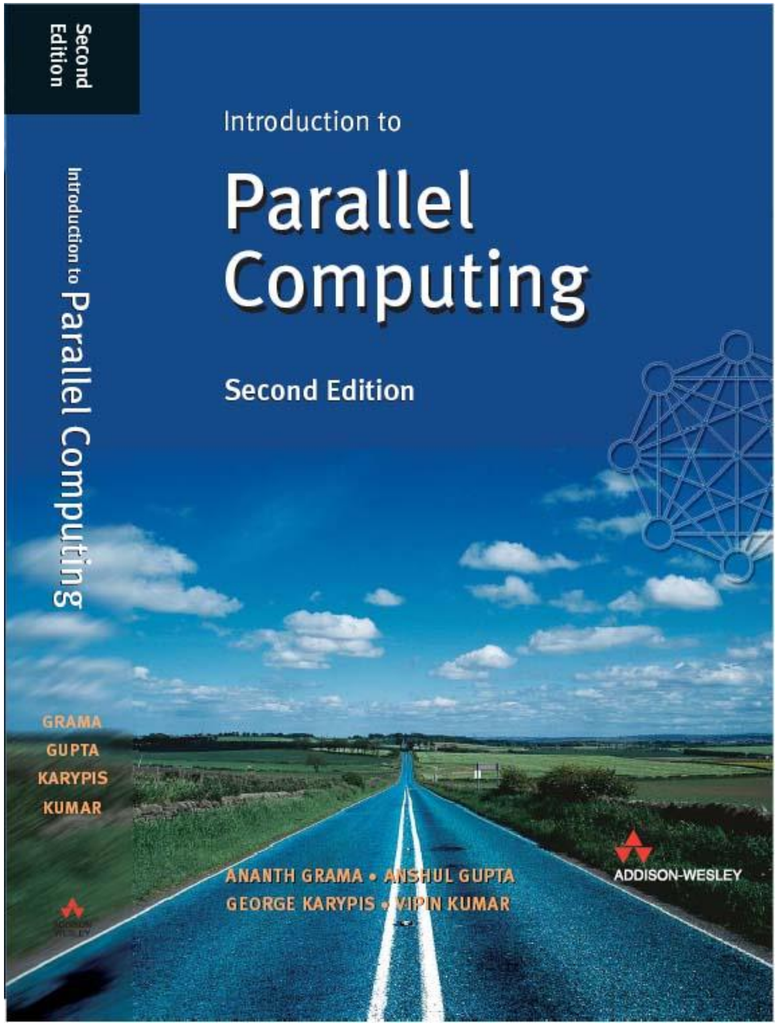
Shared-memory programming models

- 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

Data-parallel programming for GPUs

- CUDA C Language APIs
- CUDA Execution Model
- CUDA Memories
- Performance Considerations
- Parallel Patterns
    - Gather, Scatter, Reduction, Prefix Scan, and so on
- Case Studies

Reference Book 1

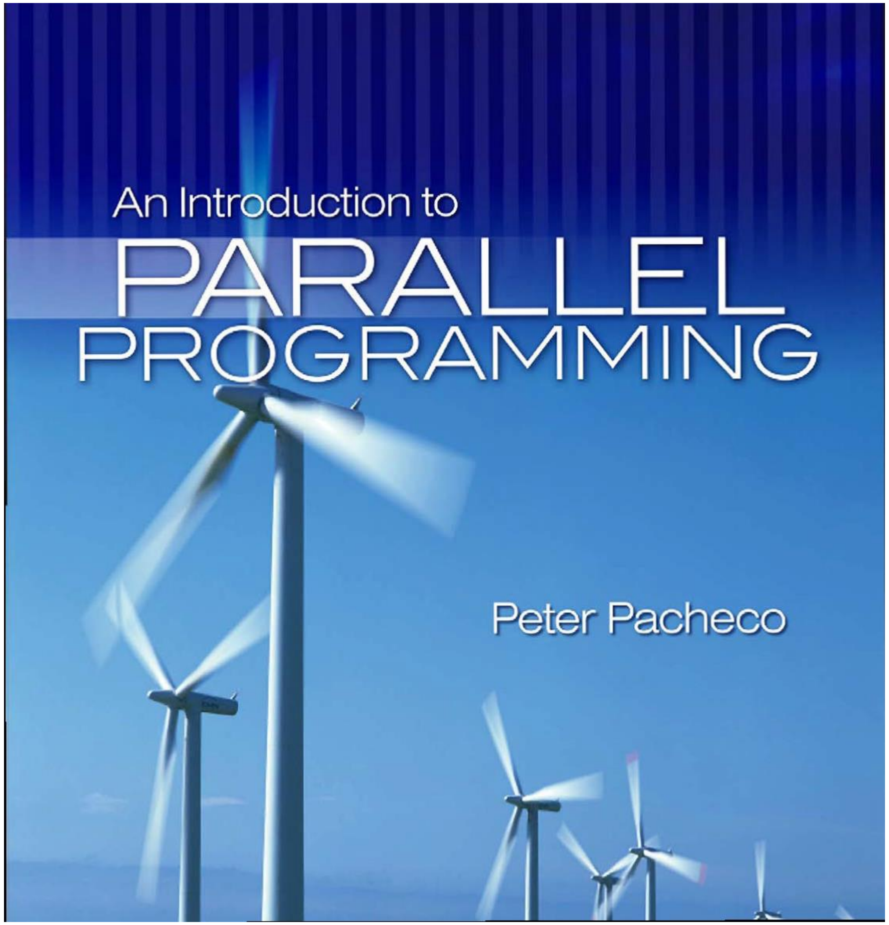


Introduction to Parallel Computing 2nd edition

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

Addison Wesley, 2003.

Reference Book 2



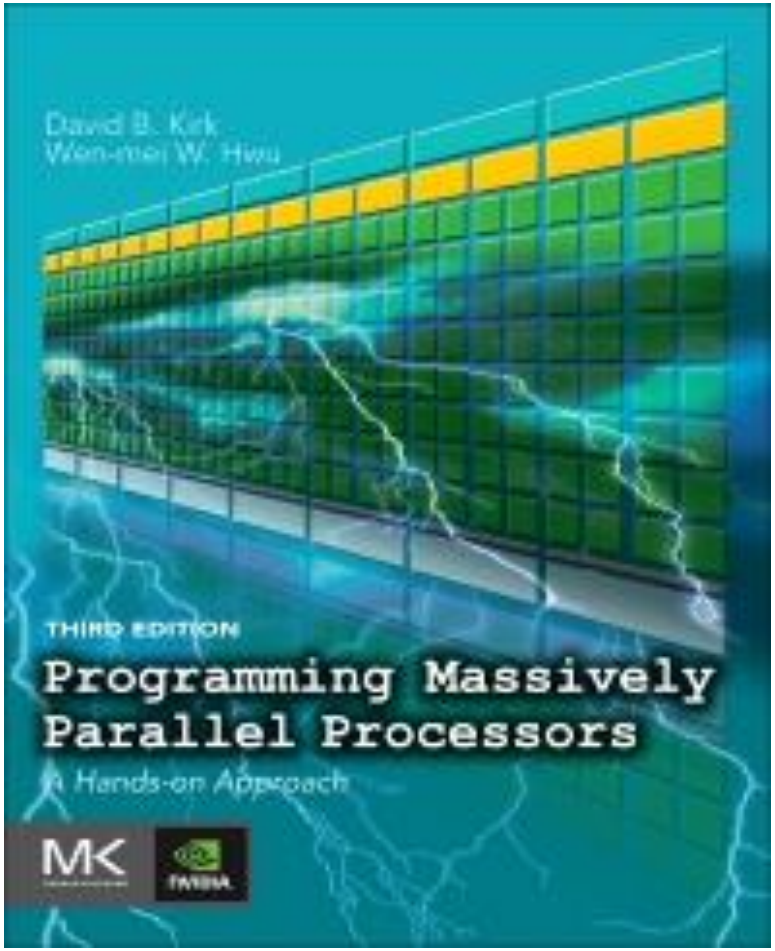
An Introduction to Parallel Programming

By Peter Pacheco

Morgan Kaufmann, 2011

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

Reference Book 3



Programming Massively Parallel Processors:
A Hands-on Approach
3rdd Edition
Author(s) : Kirk & Hwu
2017
Morgan Kaufmann

https://www.elsevier.com/books/programming-massively-parallel-processors/kirk/978-0-12-811986-0

Lecture Time and Venue

- Weekly lectures on Saturdays 3-5:50pm
  - Face-to-Face lectures as situation allows
  - Zoom lectures as necessary

Workload & Assessment

- Tentative plan
  - Three programming assignments 50%
    - Week 4, 7, 9 on MPI, Pthreads, CUDA
    - All assignments on a single topic (e.g., shortest path)
    - Sequential version program given (a few hundred lines of code)
    - Parallel program skeleton given
    - Your task is to fill in parallel processing components
  - One final exam 50%
    - Programming: fill in code, similar to assignments
    - Short answer questions on concepts from course material

Lab Facilities

Microsoft Azure

- Each student has an account.
- TA will guide you to set up virtual machines.
- Each account has about 220 VM hours.
- Get Started with Azure documentations:

https://docs.microsoft.com/en-us/azure/

Academic Integrity

- Code similarity detection enforced
- Assignment demonstration may be requested
- ALL parties in plagiarism penalized