USC Viterbi
School of Engineering
Ming Hsieh Department of Electrical Engineering

Course Announcement

EE/CSCI 451: PARALLEL AND DISTRIBUTED COMPUTATION

TTH 200-320, LAB/DISCUSSION F 200-320

FALL 2023

The course will focus on broad principles of parallel and distributed computation. The Lab associated with the course will illustrate the principles through parallel programming examples.

INSTRUCTOR: VIKTOR K. PRASANNA

Prerequisite: (EE 355x or CSCI 201L) or consent of the instructor.
Text: Introduction to Parallel Computing, Second edition, Grama, Karypis, Kumar, Gupta, Addison-Wesley.
Course Grade: based on home works, class participation, parallel programming assignments, midterm(s), final and project.

Course Outline:

1. **Introduction (1):** Architectural advances, technology perspectives, motivating examples, challenges.
2. **Architectural Principles for Application Developers (2): 1.** Pipelined processor organization: data and control hazards, ILP, out of order execution, multithreading. **2.** Memory systems: DRAM organization, cache organization. Impact on software performance, locality, multithreading. **3.** Interconnection networks: static, dynamic networks.
3. **Analytical Models for Parallel Systems (4): 1.** Architecture performance metrics: CPI, MIPS, SpecMark. Software performance benchmarks: Peak performance, sustained performance, LinPack, Bandwidth benchmarks. **2.** Limits on achievable performance, Amdhal's Law, Gustafson's Law, Scaled speed-up, scalability definitions, work optimality, Iso efficiency function, Order notation. **3.** Communication costs in parallel machines: start-up cost, throughput, latency. Routing mechanisms: packet routing, cut through, virtual channels. Modeling message passing and shared address space machines. Data layouts and graph embeddings. **4.** Multi-core, many-core architectures.
4. **PRAM and Data Parallel Algorithms (4): 1.** PRAM model of computation, Brent's theorem, illustrative examples. **2.** Max, Scan operations. **3.** Recursive doubling, graph algorithms. **4.** Performance analysis, scalability. **5.** FFT.
5. **Basic Communication Primitives (4): 1.** Broadcast and all to all, communication costs on various topologies. **2.** Personalized communication. **3.** Reduce, prefix sum and scatter and gather**. 4.** Graph embeddings.
6. **Message Passing Programming Model (2): 1.** Message passing abstraction, send receive primitives, blocking and non-blocking commands, collective operations. **2.** Illustrative examples: Canon's algorithm, overlapping computation and communication, Odd even merge sort.
7. **Shared Address Space Programming Models (2): 1.** Pthreads, OpenMP. **2.** Illustrative examples.
8. **Data Parallel Programming Abstraction of GPUs (2): 1.** GPU architecture, SIMT execution model, CUDA programming model. **2.** Illustrative examples and application mapping, optimizations, OpenCL.
9. **Parallel Dense Algebra (2): 1.** Matrix vector, matrix matrix computations. **2.** Solution to linear systems.
10. **Parallel Search and Sorting (2): 1.** Parallel search, illustrative example applications, throughput optimization. **2.** Multi-dimensional search, decision trees. **3.** Sorting techniques, bitonic sort. **4.** Mapping onto parallel architectures.
11. **Cloud, Big Data and Map Reduce (2): 1.** Cloud as a computing platform, Large data sets and organization. **2.** Map Reduce as a parallel programming model, Hadoop. **3.** Frameworks for graph analytics. **4.** Illustrative examples.
12. **Heterogeneous Computing (1): 1.** Accelerators. **2.** Spatial Computing (FPGAs) **3.** Parallel Programming Models **4.** Examples. DPC++, OneAPI, Sycl, etc.

Professor Viktor K. Prasanna
Email: prasanna@usc.edu
Ext: 0-4483
Office: EEB-200C

USC Viterbi School of Engineering
Ming Hsieh Department of Electrical Engineering

Course Announcement

EE 451
Some student Course Projects completed in earlier semesters

Parallelizing Hessenberg reduction on real square matrices
Multi-core Accelerated AlphaZero using Adaptive Parallelism
Evaluation of Parallel Gradient Descent Methods
Triangle counting on large graphs using SpMM on GPU
Accelerating CNN training using the Log Number System (LNS)
Ray tracer with Pthreads and CUDA: Evaluating CUDA performance on control-intensive applications
Event Driven Multi-threaded Parallel Fault Simulation
Spatial Separable Convolutional Neural Networks Parallelization and Acceleration
Parallel first-order logic inference
CUDA Acceleration and Memory Optimization on Attention
Accelerating the Application of Gabor Filter Banks to Images using GPU
Parallel Genetic Algorithm to solve Traveling Salesman Problem using MapReduce
Accelerated Matrix Factorization using CUDA
Parallelization of Fast Fourier Transform
Parallel Implementation of histogram based object detection
Evaluation of Connected Components Problem with Shared Memory and Message Passing
Implementation and Analysis of Parallel Delaunay Triangulation
Comparing HW and SW Acceleration for Batch Gradient Decent Algorithms
Parallel Border Tracking using GPU based Lookup Tables
Implementation and analysis of parallel LZW algorithm
Implementation and Analysis of Parallel Algorithms for the Maximum Flow Problem in a Network

Professor Viktor K. Prasanna
Email: prasanna@usc.edu
Ext: 0-4483
Office: EEB-200C