CS410: Parallel Computing

Nikhil Hegde
Milind Chabbi

Course Overview

ಭಾರತೀಯ ತಂತ್ರಜ್ಞಾನ ಸಂಸ್ಥೆ ಧಾರವಾಡ
भारतीय प्रौद्योगिकी संस्थान धारवाड
INDIAN INSTITUTE OF TECHNOLOGY DHARWAD

CS410: Parallel Computing

- Description:

High performance computing is needed in all branches of engineering. This course introduces HPC applications, architectures, platforms, and programming.

- Credit structure (L-T-P-C) : 3-0-0-6

3 contact hours (three 55min lectures) per week. 6 credits.

Full-semester (14 week + 2 exam week) elective course.

- Prerequisites: None

Exposure to C/C++/Fortran desired. If you are familiar with only MATLAB and/or Python, need to put in extra effort.

CS410: Parallel Computing - Syllabus

- Principles of Parallel Algorithm Design
  - Decomposition techniques
  - Mapping and scheduling computation
  - Templates
- Programming shared-address space systems
  - OpenCilk
  - OpenMP
  - Pthreads
  - Synchronization
- Parallel Computer Architectures
  - Shared-memory systems and Cache coherence
  - Distributed-memory systems
  - Interconnection networks and routing

CS410: Parallel Computing - Syllabus

- Programming Scalable Systems
  - Message passing and MPI
- Collective Communication
- Analytical modeling of program performance
  - Speedup, efficiency, scalability, cost optimality, isoefficiency
- Parallel Algorithms
  - Non numerical algorithms – sorting, graphs
  - Numerical algorithms – dense and sparse matrix algorithms
  - Interconnection networks and routing
- Performance measurement and analysis of parallel programs
- GPU programming with CUDA and OpenMP
- Domain-specific languages - Halide

CS410: Parallel Computing - Lecture Topics

1-2 Motivations for Parallel Computing (technology push, application pull)	25-27 GPU architecture and programming (using CUDA and OpenMP)
3-5 Introduction to Parallel Algorithm Design (decomposition techniques, metrics, limits on parallel performance, units of concurrency and their mapping)	17-24 Parallel algorithms and applications (Embarrassingly parallel, Monte-carlo methods, Stencil computation, matvec, matmul, Gaussian elim. Graph algorithms)
6-10 Shared-memory programming with OpenCilk (tasks, complexity measures, scheduling, performance and granularity, examples: vector-add, N-queens, Cilkscale and Cilksan)	28-30 Cache coherence, memory models and weak ordering, Java memory model, synchronization
11-13 Shared-memory programming with POSIX threads (data structures and functions for thread management, synchronization, producer-consumer problem.	31-35 algorithms for scalable synchronization, Message passi[illegible] programming using MPI, P[illegible] analysis using HPCToolkit
14-16 Shared-memory programming with OpenMP	36 – Domain specific languages, Halide 37 – Summary and conclusion

From MIT 6.172 lecture

From CMU 15-418 lecture

CS410: Parallel Computing

- Reference Text(s):
  - Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar: **Introduction to Parallel Computing**, Addison Wesley 2003
  - Eric Aubanel, **Elements of Parallel Computing**, CRC Press, 2017.
  - **Using OpenMP:** Portable Shared Memory Parallel Programming - Barbara Chapman, Gabriele Jost, Ruud van der Pas (2008)
  - **Using MPI:** Portable Parallel Programming with the Message-Passing Interface, 3rd Ed - William Gropp, Ewing Lusk, Anthony Skjellum
  - **Programming Massively Parallel Processors:** A Hands-on Approach, 3rd Ed. - David B. Kirk, Wen-mei W. Hwu

CS410: Parallel Computing

- Assessment Plan
  - 2 Programming Assignments (30%)
  - Exams-Written: Mid-sem (5%), End-sem exam (20%)
  - Exams-Programming: Mid-sem (20%), End-sem (20%)
  - Class participation – 5% (based on in-class quizzes administered via Google form. Roughly 1 quiz per class.)

- Sample assignments
  - Matmul on GPU, implementing MCS-K42 lock
  - Implementing All-Reduce in MPI
  - Matmul using OMP and Pthreads
  - Parallel mergesort using OpenCilk. Demonstrate use of Cilksan and Cilkscale

Required to submit a detailed report for each programming assignment