CS5080: Advanced Topics in High Performance Computing

Fall 2014

Class hours: TTh, 4:45-6:00pm
Classroom: Eng105
Instructor: Qing Yi (qyi@uccs.edu)
Office: Eng176
Office Hours: TTh, 2-3pm; By appointment
Office Phone: 2553066

Reference books

Optimizing Compilers for Modern Architectures: A Dependence-based Approach, by Randy Allen and Ken Kennedy, Morgan-Kauffman Publisheres Inc.

Optimizing And Tuning Scientific Codes, by Qing Yi. SCALABLE COMPUTING AND COMMUNICATIONS: THEORY AND PRACTICE. Edited by Samee U. Khan and Albert Y. Zomaya and Lizhe Wang. Wiley-IEEE Computer Society Press pages 255-276. Jan 2013.
http://www.cs.uccs.edu/~qyi/papers/BookChapter11.pdf.

Structured Parallel Programming Patterns for Efficient Computation. By Michael McCool, James Reinders, and Arch Robison. Morgan Kaufmann. 2012. ISBN 978-0-12-415993-8.
http://parallelbook.com/sites/parallelbook.com/files/SC13_20131117_Intel_McCool_Robison_Reinders_Hebenstreit.pdf

The Art of Multiprocessor Programming, Maurice Herlihy and Nir Shavit. Morgan Kaufmann. 2012. ISBN: 978-0-12-397337-5.
http://www.e-reading.ws/bookreader.php/134637/Herlihy_-_The_art_of_multiprocessor_programming.pdf.

Overview

This course will study ways to efficiently parallelize applications on shared memory multicore platforms and a large collection of optimization techniques to enhance application performance on such platforms. The optimization techniques will be applied at the source level to C++/C applications, focusing on high-level concepts such as loops, arrays and object-oriented abstractions. We will examine techniques to systematically detect opportunities of applying advanced optimizations to effectively improve the degrees of parallelism, synchronization efficiency, and cache reuses using both parallel programming models such as OpenMP, Pthreads, and automated compiler optimization techniques.

Class Objective

This is a research-oriented graduate-level course. By the end of the class, students are expected to have a good grasp of the relevant programming, analysis, and optimization techniques for high performance computing, be able to adequately understand research papers in this area, and be able to implement research projects in this area.

Prerequisites

Students should have a basic understanding of parallel programming, com-

	pilers, and modern computer architectures.
Requirements	Students will be required to present research papers from leading HPC conferences and to develop 2-3 class projects which applies what we learned to improve application performance. A final report is required by the end of the semester.
Grading	50%: project implementations and reports; 30%: paper presentations and reviews; 20% class related exercises and midterm quizzes
Attendance	Students are responsible for all presented materials and assigned readings in class.
Collaboration Policy	Students can work on the assigned projects individually or in groups. For group projects, a clear documentation of the division of work is required.
Email Policy	You may direct class-related questions via email to the instructor, with a response period of 1-24 hours on workdays.