

Wright State University

CORE Scholar

Computer Science & Engineering Syllabi

College of Engineering & Computer Science

Spring 2008

CS 499/699: Introduction to Parallel Programming

Natsuhiko Futamura

Wright State University - Main Campus

Follow this and additional works at: https://corescholar.libraries.wright.edu/cecs_syllabi



Part of the [Computer Engineering Commons](#), and the [Computer Sciences Commons](#)

Repository Citation

Futamura, N. (2008). CS 499/699: Introduction to Parallel Programming. .

https://corescholar.libraries.wright.edu/cecs_syllabi/564

This Syllabus is brought to you for free and open access by the College of Engineering & Computer Science at CORE Scholar. It has been accepted for inclusion in Computer Science & Engineering Syllabi by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.

CS 499/699, Introduction to Parallel Programming

Syllabus: Spring 2008

Time: Tuesday, Thursday, 8:00-9:15PM

Class Room: 153 Russ

Instructor: Professor Natsuhiko Futamura

Office: 387 Joshi Hall

Email: natsuhiko.futamura@wright.edu

Phone: 937-775-5107

Low-cost parallel computers such as PC clusters are becoming available, and many computationally intensive problems can be solved using such computers. It is, however, still not easy to design and implement a software that run fast using multiple processors. This course covers basic software design methods and experiencing programming parallel programming using MPI. After taking this course students will be able to design parallel algorithms, evaluate the speed of the execution, and write MPI codes.

Suggested Reading

Parallel Programming With MPI, Peter Pacheco, Morgan Kaufmann, ISBN: 1558603395

Topics: The topics to be covered include

- Designs of parallel computers.
- Interconnection networks.
- Methods to evaluate the performances of algorithms.
- Benchmarking.
- Speedup and efficiency of parallel algorithm.
- Primitive parallel operations
- Packing messages.
- Overlapping computation and communication.
- Load distributions.