Fall | 18

[Syllabus]

HBM513E Parallel and Distributed Computing

Prof. Dr. M. Serdar ÇELEBİ
Computational Science and Engineering Program
Istanbul Technical University / Informatics Institute

Course Objective

Parallel computing is a broad field of computer science concerned with the architecture, HW/SW systems, languages, programming paradigms, algorithms, and theoretical models that make it possible to compute in parallel. While parallel programming and execution can be studied in the abstract, performance is parallelism's *raison d'être*. Indeed, parallel computing is the name of the game in high-performance computing. Large-scale parallelism (>10000 processors) lies at the heart of the fastest, most powerful computing systems in the world today. At the same time, small-scale, low-end parallelism is the driving force behind affordable scientific computing and the growing success of computational science.

This class will survey recent advances in parallel algorithms for scientific computing and will aim to discover their impact on improving performance of scientific applications. Since students come from different backgrounds, Lectures will attempt to emphasize an even spread of topics in parallel algorithm design, numerical methods, computer architecture, software and runtime environments. Students will spend time on particular parallel algorithms for scientific problems and their implementations on parallel and distributed machines.

We are going to concentrate upon the message-passing methods of parallel computing and use some standard parallel computing tools such as MPI (Message Passing Interface), OpenMP (Directive-based Threading) for SMP and Distributed Shared-memory Computer Architectures. Finally, students will experience the fundamentals of different approaches to create and manage the parallelism in order to exploit the benefit of state-of-the-art computing technologies.

Instructor: Prof. Dr. M. Serdar Celebi
Phone: (0212) 285 7012 - 285 7077
E-Mail: mscelebi@itu.edu.tr

Office: Informatics Institute, second floor, room 214.

Office Hours: Monday 11:00 –12:00, Friday 10:30 – 12:30

Research Assistants:

Mehmet TUNÇEL (mehmet.tuncel@be.itu.edu.tr)
Ayla OKAN (ayla.okan@be.itu.edu.tr)
Derya BODUR (derya.bodur@be.itu.edu.tr)
Samet Demir (sametdemir@gmail.com)
Gözde İniş (gozdeinis90@gmail.com)

Lecture Hours: Monday 13:30 –16:30

Textbooks:

1) ***Parallel Programming***, Wilkinson Barry & Allen Michael, Prentice Hall, 1999.
2) ***Parallel Programming with MPI***, Peter S. Pacheco, Morgan Kaufmann, 1997.
3) ***Introduction to Parallel Computing***, Petersen and Arbenz, Oxford University Press, 2004.
4) ***Using MPI***, William Gropp and Ewing Lusk, A. Skjellum, MIT Press, 1999.
5) ***MPI- The Complete Reference***, Volume 1, The MPI Core, 2nd Edition. Marc Snir, Steve Otto, S.H. Lederman, David Walker and Jack Dongarra, MIT Press, 1998.
6) ***Using OpenMP***, Barbara Chapman, Gabriele Jost and Ruud Van Der Pas, MIT Press, 2008.
7) ***Introduction to Parallel Computing***, 2nd Edition, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Pearson Education, 2003.
8) ***Parallel Computing in C with MPI and OpenMP***, Micheal J. Quinn, Mc. Graw Hill, 2003.
9) ***Principles of Parallel Programming***, Calvin Lin and Lawrence Snyder, Pearson International, 2009.

Web Page: http://hbmders.be.itu.edu.tr/course/

Lecture Location:

ITU, Ayazaga Campus, Building of Informatics Institute, 4th floor, Room 411.

Lab Location:

ITU, Ayazaga Campus, Building of Informatics Institute, 4th floor, PC Lab.

Grading:

Homeworks (2)	18%
Term Project	20%
Mid-Term Exam	28%
Final Exam	34%

TENTATIVE CONTENT

Week 1: Introduction

- Introduction to computational science
- What is parallelism?
- Why we need powerful computers?
- Units of measure in HPC
- Why powerful computers are parallel?

- Principles of parallel computing
- Locality and parallelism
- The Flynn Taxonomy
- Custer Server Systems
- I/O Needs on Parallel Computers, RAID Concept
- Parallel File Systems
- Driving issues/Trends on CPU
- Introduction to Eclipse

Week 2: Parallel Computing Architectures

- Parallel Programming Models
- Performance
- Speedup factor and efficiency, Amdahl's and Gustafson's Laws
- Types of Parallel Computers: Shared Memory Multiprocessor
- Programming Shared Memory Multiprocessors
- Multiprocessor Interconnection Networks
- Distributed Shared Memory
- Network: Latency and Bandwidth

LAB 1: Introduction to HPC Systems: Login and Environmental Setup

Week 3: OS and Memory Hierarchy

- Operating System and Distributed OS
- Linux System process
- Memory hierarchies
- Cache basics
- Application: Matrix multiplication
- Latency
- Impact of memory bandwidth
- Multilevel caches
- Introduction to cache coherence

LAB 2: Introduction to HPC Systems: Unix/Linux Development Tools for HPC

Week 4: Caches, Virtual Memory and Programming Performance

- Cache mapping
- Direct-Mapped cache
- Set Associative cache
- Fully Associative cache
- Replacement Algorithms
- Cache Performance Metrics
- Virtual memory

- Address translation
- Accelerating with TLB's
- How architecture impacts your programs
- How (and how not) to tune your code

LAB 3: Eclipse I: Software Development Environment

Week 5: Message Passing Computing I

- SIMD and MIMD model
- Point-to-point send and receive communications
- Synchronous and asynchronous message passing
- Simple MPI examples
- Basics of collective communications
- Evaluating parallel programs

LAB 4: Eclipse II: Parallel Tool Platform

Week 6: Message Passing Computing II

- Buffered and non-buffered point-to-point communication
- Collective communications

LAB 5: Intel Software Tools I: Compiler, debugger, profiler and analyzer

Week 7: Parallel Techniques I

- Performance issues
- Embarrassingly Parallel Computations
- Ideal Parallel Computation
- Embarrassingly Parallel Examples

LAB 6: MPI I: Introduction, simple algorithms, send & receive

Week 8: Parallel Techniques II

- Partitioning and Divide-and-Conquer Strategies
- Pipelined Computations

LAB 7: MPI II: Introduction, simple algorithms, send & receive

Week 9: Parallel Techniques III

- Synchronous Computations
- Synchronization

- Data Parallel Computations
- Synchronous Iteration Program Examples
- Solving a System of Linear Equations by Iteration
- Heat Distribution Problem

LAB 8: MPI III: One-sided communication

Week 10: Parallel Techniques IV

- Load Balancing and Termination Detection
- Dynamic Load Balancing (DLB)
- Centralized DLB
- Decentralized DLB
- Load Balancing Using a Line Structure
- Distributed Termination Detection Algorithms

LAB 9: MPI IV: One-sided communication, collective communication

Week 11: Programming with shared memory I

- Basic shared memory architecture
- Differences between a process and threads
- Accessing shared data
- Shared data in systems with caches
- Cache coherence problem

LAB 10: MPI V: Collective communication

Week 12: Programming with shared memory II

- Introduction to cache coherency
- Snoopy cache coherence
- Directory based cache coherence
- Distributed Directory based cache coherence

LAB 11: Intermediate Compiler, debugger, profiler and analyzer tools

Week 13: Programming with shared memory III

- Introduction to OpenMP
- Directives and variable sharing
- Programming samples
- Multi-core systems and programming

LAB 12: OpenMP I: Introduction

Week 14: Programming with shared memory IV

- Advanced OpenMP
- OpenMP and Machine Architecture
- Performance: Vectorization
- OpenMP for Heterogeneous Computing

LAB 13: OpenMP II: Introduction & Applications

LAB 14: MPI and OpenMP: Applications, Performance and Tuning

LAB & SUPPLEMENTARY NOTES

1. Int. to HPC Systems I: Login and Environmental Setup Notes (One Lab)
2. Int. to HPC Systems II: Unix/Linux Development Tools for HPC Notes (One Lab)
3. Supplementary Notes for Code Optimization Techniques
4. Supplementary Notes for Batch Processing
5. Intel Software Tools I and II: Compiler, Debugger, Profiler and Analyser (Two Labs)
6. Eclipse Software Development Environment Notes (One Lab)
7. MPI I, II, III, IV and V Lab Notes (Five Labs)
8. Open MP I and II Lab Notes (Two Labs)
9. MPI Applications and Performance Tuning (One Lab)

TERM PROJECT

A Term Project will be assigned to you for writing a parallel algorithm using MPI and/or OpenMP. In the Project, an initial and/or boundary value problem or your choice of pre-approved problem will be solved using parallel algorithms and tools toughed in the class. Profiling, Analyzing, Performance and Scalability tests are going to be implemented for each assigned problem.

ADDITIONAL REFERENCES

1) G.Golub and J. M. Ortega, *Scientific Computing with an Introduction with Parallel Computing,* Academic Press Inc., 1994.

2) Workshop notes at www.hlrs.de (stutgard university, Germany)

3) J. M. Ortega, *Introduction to Parallel and Vector Solution of Linear Systems*, Plenum Press, 1988.

4) D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Compuation: Numerical Methods,* D. P. Bertsekas and J. N. Tsitsiklis, Prenctice Hall, 1989.

5) http://www.cs.berkeley.edu/cs267/

6) J. Dongarra et. al., *Solving Linear Systems on Vector and Shared Memory Computers*, , SIAM, 1991.

7) David W. Walker's Parallel Processing Lecture Notes

8) Wim Bohm's Parallel Processing Lecture Notes, Colorado State University

9) Rajat P. Garg and I. Sharapov, ***Sun BluePrints: Techniques for Optimizing Applications***, Sun Microsystem Press, 2002.

10) Seyed H. Roosta, ***Parallel Processing and Parallel Algorithms***, Springer Verlag, 2000.

11) Peter Pacheco, ***Parallel Programming with MPI***, Morgan Kaufmann Pub.,1997.

12) Xavier C., Iyengar S.S. and Lyengar S.S., ***Introduction to Parallel Algorithms,*** Wiley Series, 1998.

13) Dongarra, Foster, Fox, Gropp, Kennedy, Torczon, W., ***Sourcebook of Parallel Computing***, Morgan Kauffman, 2003.

14) David E.K., Ahmed S. and Venkatakrishnan, ***Parallel Numerical Algorithms,*** Icase/Larc Interdisciplinary Series in Science and Engineering, Vol 4, 1997.

15) Kumar V., ***Introduction to Parallel Computing: Design and Analysis of Parallel Algorithms***, Addison-Wesley, 1994.

16) Van Der Welde, ***Concurrent Scientific Computing****,* Springer, 1994.

17) Almasi G.S. and Gottlieb A., ***Highly Parallel Computing***, 2nd ed., Benjamin/Cummins, CA, 1994.

18) Akl S.G., ***The Design and Analysis of Parallel Algorithms***, Prentice Hall, Englewood Cliffs, NJ, 1989.

19) Akl S.G., ***Parallel Computation: Models and Methods***, Prentice Hall, Upper Saddle River, NJ, 1997.

20) Berman K.A. and Paul J.L., ***Fundamentals of Sequential and Parallel Algorithms****,* PWS Publishing, Boston, 1997.

21) Chaudhuri P., ***Parallel Algorithms: Design and Analysis***, Prentice Hall, Englewood Cliffs, NJ, 1992.

22) Gibbons A. and Rytter W., ***Efficient Parallel Algorithms***, Cambridge University Press, New York, 1988.

23) Jaja J., ***An Introduction to Parallel Algorithms***, Addison-Wesley, Reading, MA, 1992.

24) Smith J.R., ***The Design and Analysis of Parallel Algorithms***, Oxford University Press, New York, 1993.

25) Quinn M.J., ***Designing Efficient Algorithms for Parallel Computers***, McGraw-Hill, New York, 1987.
26) Freeman T.L. and Phillips C., ***Parallel Numerical Algorithms***, Prentice Hall, Englewood Cliffs, NJ, 1992.