IDC 606A High Performance Computing with Applications

Instructor : Mahendra K. Verma (PHY)

About the course: This course on parallel programming on scientific applications will enable the students to write parallel programs. The course will provide a overview of HPC hardware, and then move to Message Passing Interface (MPI) and OpenMP programming. GPU tools, such as OpenACC, CuPy, and CUDA, will be introduced in the course. These topics would useful to all students irrespective of their branch.

In the course, the algorithms of applications such as Parallel Computational Fluid Dynamics, Parallel Molecular Dynamics would be covered. The HPC tools will then be employed to the above topics.

Prerequisite: Basic knowledge in computer programming and mathematics

Who can take the course: Ph. D., M. Sc., and Advanced UG students.

Course Contents:

1. Importance of HPC in current times [1 lecture]
2. Introduction to HPC hardware: Top-end CPUs and Memory; HPC clusters: Distributed and shared memory architecture; Interconnect; Accelerators—GPUs; Computer taxonomy, e.g., single instruction multiple data (SIMD) [5 lectures]
3. HPC programming paradigms: Single program multiple data (SPMD) & MPMD. [2 lectures]
4. Overview of major scientific and engineering applications: Computational Fluid Dynamics; Monte Carlo methods; Structural mechanics; etc. [2 lecture]
5. Speeding up C/Fortran/Python programs: Vectorization; Compiler options [2 lectures]
6. Programming in Message Passing Interface (MPI): Point-to-point and collective communications; Parallel I/O; MPI for Python and C/Fortran [8 lectures]
7. Programming in OpenMP: [4 lectures]
8. Introduction to GPUs: SIMD architecture; Case study of Nvidia GPUs [3 lecture]
9. Programming GPUs using OpenACC [3 lectures]
10. Programming GPUs using CuPy and CUDA [4 lectures]
11. Handling big data; Introduction to HDF5 and similar formats [2 lectures]
12. Visualisation tools such as Visit and Paraview [2 lectures]
13. Case study of several major application [2 lectures]

Textbooks and References:

1. P. S. Pacheco, An Introduction to Parallel Programming, Elsevier (2011)
2. M. Quinn, Parallel Programming in C and OpenMP, McCraw Hill Education (India) (2003)
3. A. Grama, A. Gupta, G. Karypis, and V. Kumar, Introduction to Parallel Computing, Pearson (2007)
4. G. Zaccone. Python Parallel Programming Cookbook, Packt Publ. (2015)
5. https://cupy.dev (for CuPy)
6. R. Farber, Parallel Programming with OpenACC, Morgan Kaufmann (2016)

Grading policy:

Midsem exam: 20 marks
Final Exam: 20 marks
Midsem project eval: 20 marks
Final project eval: 40 marks
Attendance & class participation: 10 marks

85% attendance is mandatory.