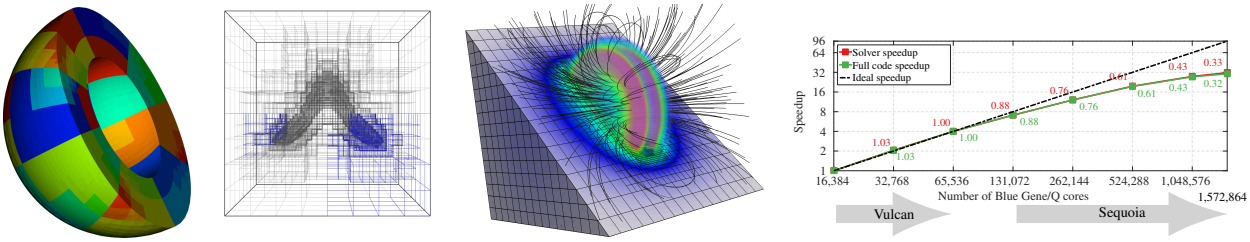
Spring 2019: Advanced Topics in Numerical Analysis: High Performance Computing

Cross-listed as MATH-GA 2012.001 and CSCI-GA 2945.001



Lectures: Monday 5:10–7:00pm, WWH 1302

Website: https://nyu-hpc19.github.io/

Instructors: Georg Stadler,
WWH #1111
E-mail: stadler@cims.nyu.edu

Dhairya Malhotra,
WWH #1008
E-mail: dm4340+hpc19@nyu.edu

Description: This class is an introduction to the fundamentals of parallel scientific computing. We will establish a basic understanding of modern computer architectures (CPUs and accelerators, memory hierarchies, interconnects) and of parallel approaches to programming these machines (distributed vs. shared memory parallelism). Issues such as load balancing, communication, and synchronization will be covered and illustrated in the context of numerical algorithms. Since a prerequisite for good parallel performance is good serial performance, this aspect will also be addressed. Along the way you will be exposed to important tools for high performance computing such as debuggers, schedulers, visualization, and version control systems.

Prerequisites: Prerequisites for the course are some (serial) programming experience with C, C++ (we will mostly use C and some basic C++ in class) or FORTRAN, some familiarity with numerical methods and knowledge of basic UNIX commands. In case of doubt, please come by or send an email to the instructors.

Required work: This will be a hands-on class, with several parallel (and serial) computing assignments. Your active participation is crucial since you will have to explore material by yourself and try things out. There will be individual larger final projects at the end, which you can tackle by yourself or with a partner. Students who have code they want to parallelize or speed up can use that for their final project.

Intended topics: Algorithms (matrix-matrix and matrix-vector multiplication, sorting, tree algorithms, Jacobi smoothing and multigrid); Computer architectures (CPUs, accelerators, memory hierarchy, parallel programming models, networks); Programming (single-core performance optimization, OpenMP, MPI, Cuda); Tools (make, git, valgrind, shell scripting, visualization).

Literature and Organization: We will regularly post links to online reading material and resources which you are required to study. The class will be organized using Slack. If you are signed up for the class you will get an invitation for the Slack group, if you are auditing let us know and we will add you.