PROJECT IDEAS

HPC1 Fall 2013

Project proposals due Thursday, October 10. Final projects are due Tuesday, December 17.

Projects should involve investigating a substantial topic of interest in HPC. Some suggested topics will be included below, but by all means feel free to propose something of interest to you (research-related or otherwise). Your **proposal** should be approximately one page, and should have a description of the project goals along with a detailed plan for implementation. The final well-organized report should be at least 10 pages in length. The project report should contain the background description of the problem that you are trying to solve, and the computational solution (using some form of parallel processing), including validation of the code and performance analysis utilizing the tools and methods covered in class.

Some project ideas:

1. Parallelize an existing scientific/engineering application to achieve scalable parallel performance (if it related to your thesis work that is even better).
2. Develop a detailed performance model of an existing parallel scientific/engineering application, and improve its performance using the resulting model.
3. Compare a “new” parallel API (e.g., UPC or CAF) with more established parallel APIs within the context of an HPC application, or suite of applications.
4. Utilize graphics cards (GPUs) for doing calculations.
5. Develop or extend an application using ScaLAPACK solvers, and model/evaluate the resulting performance.
6. Develop or extend an application using the PETSc solvers, and model/evaluate the resulting performance.
7. Investigate the “randomness” (e.g. using the statistical tests in Knuth’s volume 2 of *The Art of Computer Programming* of stock pseudo-random number generators, and/or develop a “quality” parallel RNG.
8. “Unofficial” MPI bindings (e.g., Java, Python, etc.) - availability, comparative performance, and real applicability.
9. Develop or port a scientific/engineering application using a dynamic programming language such as Python.