

## Homework: Performance Analysis

(A)

Parallelize the block algorithm for matrix multiplication described at:

<http://www.netlib.org/utk/papers/autoblock/node2.html>

Implement a master / worker scheme using MPI in either C or Fortran. The program should be able to run on any number of processors and multiply square matrices of order 512. The master (rank zero) should distribute the data to the workers and receive the results.

Now, analyze the performance behavior of your application. Write MPI wrappers in either C or Fortran for all MPI functions you are using to measure the following quantities:

- o Time spent in each MPI function individually
- o Data received by each MPI function individually
- o Data sent by each MPI function individually
- o Time spent in all MPI functions together
- o Data received by all MPI functions together
- o Data sent by all MPI functions together
- o Time spent in the entire program (i.e., between MPI\_Init() and MPI\_Finalize())

The wrappers should calculate the results for each process individually and aggregate across all processes. At the end, the master is supposed to print a human-readable performance report listing the desired quantities.

Deliverables:

- o Program source code including wrappers
- o Makefile (please specify your platform)
- o Performance reports for 1,4, and 16 processes
- o Documentation

Due: April 25th before the class.

(B)

Read "Scalable Analysis Techniques for Microprocessor Performance Counter Metrics" by D.H. Ahn and J.S. Vetter:

[http://www.llnl.gov/CASC/people/vetter/pubs/SC02\\_HWC\\_stat\\_analysisDec02.pdf](http://www.llnl.gov/CASC/people/vetter/pubs/SC02_HWC_stat_analysisDec02.pdf)