CS 320/CSE 302/ECE 392
Introduction to Parallel Programming for Scientists and Engineers

Machine Problem 4

Assigned: Friday, March 31, 2000
Due: Friday, April 14, 2000
Auto-extension: Monday, April 17, 2000

The objective of this MP is to implement an MPI version of the problem given in Machine Problem 3. The algorithm is the same as the one in MP3 i.e. numerically solving the laplace equation.

As before, write the first parallel version using the wavefront method. For this, the master thread will first initialize the boundary (to the same values as in MP3). It will then send (only) the required values of v(.,.) to the other processors. The processors will update v(i,j) and will then synchronize; the master thread will then send the other set of values, and so on.

The second parallel version will execute the alternate rows/columns in parallel so as to avoid races. So in the first pass the master thread will send elements from rows/columns $1, 3, \ldots$ to the respective processors, and in the second pass will send rows/columns $2, 4, \ldots$

Students taking the course for 1 unit of credit should also implement a blocked version of the algorithm. The number of blocks can be same as the number of processors, or may be more in which case each processor will process more than one block. Also create an appropriate cartesian topology, so that each processor can know its neighbor by the standard MPI commands. Thus if the processor requires some value from its neighboring processor, it will receive the value from only the relevant processor (and not broadcast to all). The blocks themselves will be processed in either of the above two methods, either wavefront order or alternate rows/columns.

Information about running MPI programs

To use MPI you will need to include `/usr/commercial/ptools/mpich/bin` in their path (by adding something like `path=(/usr/commercial/ptools/mpich/bin $path)` to your .login file).

To compile the MPI programs use: `mpif77` *<fortran source file (*.f)>* or `mpicc` *<c source file (*.c))* (There is also `mpiCC` for C++ programs). To run the program use:

`mpirun` -np *<number of processors> <executable file>.*

Please do not use more than 6 processors since no more than these are available, and during high loads please do not use more than 4 processors.

Apart from the tutorials provided on the web page, a comprehensive document describing the MPI commands can be found at `http://www.mcs.anl.gov/mpi.` (The MPI standard being used is 1.1 and not 2.0, so click on MPI Standard 1.1 on the left side).