Jeremy Bonnell
CS4230
Programming Assignment #3

README

I was able to successfully parallelize the code into both POINT-TO-POINT and COLLECTIVE Versions. I was able to remove alpha and beta from the 'k-loop'. This was achieved by recognizing that every row of alpha has the same value, and every row of beta is the same. Once the first value of alpha is determined it's sent to core 0 and placed in a whole row of Alphas and a whole column of Betas

The code implementation had two i-loops with nested j-loops. The first i-loop had 2 j-loops. One loop was for calculating alpha and used odd ranked processes. The other was for calculating gamma and used even ranked processes. The results were then sent to the rank 0 core - which was running in the other i-loop (one nested j-loop) - for placing the results into Alphas[][], Betas[][], and Gammas[][]. The odd and even processes operated in a cyclic fashion. For instance, if there were 5 processes running Rank1 and Rank2 got the first alpha and the first gamma, while Rank3 and Rank4 got the second alpha and second gamma. Meanwhile Rank0 took the results and stored their values in the corresponding arrays.

The Collective version was a little more straight forward. I had the threads divide the inner k-loop in a cyclic fashion and combine their results using MPI_Reduce, and MPI_Allreduce.

$ mpicc –g –Wall –o Reduction Reduction.c
is used when compiling C programs. The 'mpicc' says to look in C libraries
$ mpic++ –g –Wall –o Reduction Reduction.cpp
is used when compiling C++ programs. The 'mpic++' says to look in C++ libraries
In order for the Validation_mpi.cpp to compile without errors, I also had to use:
#include <cmath>

TO compile, run, and Validate my code, I used the following commands:

- mpic++ –g –Wall –o Reduction Reduction.cpp
- mpic++ –g –Wall –o Reduction_p2p Reduction_p2p.cpp
- mpic++ –g –Wall –o Reduction_coll Reduction_coll.cpp
- g++ –g –Wall –o Validation_mpi Validation_mpi.cpp

- mpiexec –n 1 ./Reduction 512 512 –t –d

- mpiexec –n 9 ./Reduction_p2p 512 512 –t –d
- ./Validation_mpi –p

- mpiexec –n 8 ./Reduction_coll 512 512 –t –d
- ./Validation_mpi –p

Default code (time output):
Time: 2834.51 ms.

Point-to-Point (time and Validation output):
Time: 1566.51 ms.

VALID!

difference in Alphas: 0
difference in Betas: 0
difference in Gammas: 0

Collective (time and Validation output):
Time: Time: 2644.81 ms..

VALID!

difference in Alphas: 0
difference in Betas: 0
difference in Gammas: 0