CS5500 HW6

Chase Mortensen A01535275

October 2019

1 Description

Write your own global_sum function using individual sends and receives. Try several different variants. Compare results.

My implementation is based off of the global_sum.cpp file that was discussed in class. It essentially compares a version of a global sum using `MPI_Send`s and `MPI_Recv`s against one using `MPI_Allreduce`.

2 Program

```
#include <iostream>
#include <mpi.h>

#define MCW MPI_COMM_WORLD
using namespace std;

int main(int argc, char **argv){
    int rank, size;
    int data;
    int powOfTwo = 0;
    int powSize = 1;
    int mask;
    int dest;

    MPI_Init(&argc, &argv);
    MPI_Comm_rank(MCW, &rank);
    MPI_Comm_size(MCW, &size);
    int sum;
    int incoming;
    sum = rank;
    while(powSize<size){
        powSize<<=1;
        powOfTwo++;
```

```
    }

    mask = powSize>>1;

    double t1,t2;

    t1 = MPI_Wtime();

    while (mask){
        dest = rank ^ mask;
        mask>>=1;
        MPI_Send(&sum,1,MPI_INT,dest,0,MCW);
        MPI_Recv(&incoming,1,MPI_INT,dest,0,MCW,MPI_STATUS_IGNORE);
        sum+=incoming;

    }
    t2 = MPI_Wtime();
    if(!rank)cout << "Sends and Receives time: " << t2-t1 <<endl;

    t1 = MPI_Wtime();
    MPI_Allreduce(&rank,&sum,1,MPI_INT,MPI_SUM,MCW);
    t2 = MPI_Wtime();
    if(!rank)cout << "All reduce time: " << t2-t1 <<endl;

    MPI_Finalize();

    return 0;
}
```

3 Output

```
$ mpic++ global_sum.cpp
$ mpirun -np 4 ./a.out
Sends and Receives time: 0.000852108
All reduce time: 0.00106978

$ mpic++ global_sum.cpp
$ mpirun -np 32 ./a.out
Sends and Receives time: 0.0797052
All reduce time: 0.0899656
```