CS-5379: Parallel Processing

Spring Semester 2017
Department of Computer Science
Texas Tech University

Assignment-04

Amitsingh Pardeshi

Sudo code:

If number of processors is divisible by number of vertices in the matrix.

Get the matrix containing vertices and cost

MPI_Bcast Broadcast nodes to each processor

MPI_Scatter distribute matrix among the processors

Start measuring the time.

Call Floyd's algorithm
{

    For k starting from 0 through n-1

    root = k / (n / p);

    determine which process owns the k_{th} row

    if current rank == root;

    if the executing process is the owner of the row, it has to send a copy to all other processes
{

    local_index = k % (n / p);

    computer row k's index in the submatrix owned by process id

    Copy the row from the submatrix owned by this process into the temporary array to be broadcast.

    Since we cannot broadcast the actual row because we will be updating it
}

Broadcasts from root to all other processes

for (local = 0; local < n / p; local++) where n/p gives "n local rows"

for i equals 0 to n

local matrix[local*n + i] = min(local matrix [local * n + i], local matrix[local*n + k] + tmp[i]);

}
End measuring time

MPI_Gather will gathers data

Prints the time it took alone with the shortest path result matrix

---

Submission files –
  1. CS-5379-Assignment-4-Readme
  2. assignment_4

Command to execute –

    mpiexec -n 6 assignment_4.exe