Parallel Computing
Exercise Sheet 10: Programming MPI II
30.01.2018

1 Collective Communication

Given is a 2-dimensional torus of size $N \times N$. Nodes are labelled from 1 to $N$ in both dimensions where node $(1, 1)$ resides in the upper left corner and node $(N, N)$ in the lower right corner of this topology. For implementing a broadcast, each node – after receiving – always forwards the received information both to its right and lower neighbour. This procedure – starting from one dedicated root node – successively continues until all nodes have been informed. In order to keep this procedure ‘symmetric’, the root node should also receive from its other two neighbours.

a) Sketch the single steps of this algorithm (drawing arrows between sender and receiver nodes) for a $4 \times 4$ torus with node $(1, 1)$ as root!

b) Give an MPI implementation (a communication skeleton is enough) of the above algorithm using correct send/receive statements and also think about dependencies! To keep communication simple, you may refer to a node’s neighbours with left, right, up, and down instead of MPI ranks.

2 Communication Pattern

Given is the following MPI code fragment (with mod denoting the division with remainder and div the integer division).

```plaintext
for i ← N to 1 do
  off ← 2^i
  shift ← (rank div off) * off
  dest ← ((rank mod off) + off/2) mod off + shift
  MPI_Send (val, 1, MPI_INT, dest)
  MPI_Recv (tmp, 1, MPI_INT, dest)
  if val < tmp then
    min ← val; max ← tmp
  else
    min ← tmp; max ← val
  fi
  if rank < dest then
    val ← min
  else
    val ← max
  fi
od
```
a) In totally, there are $N$ communication stages where $p = 2^N$ processes are involved. Sketch the communication pattern using arrows (MPI_Send only) for $N = 3$ stages with $p = 8$ processes.

![Communication Pattern Diagram]

b) Processing the data within the loop body takes $T_{EX}$ time, exchanging two elements (i.e. both send and receive) between processes $p_i$ and $p_j$ takes $T_{COM} = 3 \cdot T_{EX}$ time. Assuming $T(1) = p^2 \cdot T_{EX}$, compute the speed-up depending on $N$ only and sketch it for different values of $N$! What do you observe?

c) Sketch for $N = 3$ and the given initialisation of the variable $val$ how this value changes during program execution and describe what the program computes!

![Value Changes Table]