

## Chapter 3 MPI topic: Collectives

### 1. broadcast, scatter, reduction and gather.

#### Collectives in MPI

### 2. Reduction

Reduction to all, reduction of distributed data, reduce in place, reduction operations.

### 3. Root collectives: broadcast, reduce

Root process, generate or read in the initial data for a program run, needs to be communicated to all other processes; at the end of a program run, needs to output some summary information.

### 4. Rooted collectives: gather and scatter

Compared to the broadcast/reduce, the gather/scatter calls have: a large buffer on the root, with datatype and size specification; a smaller buffer on each process with its own type and size specification.

The gather call collects all this data from the processors to the root; the scatter call assumes that the information is initially on the root and it is spread to the individual processors.

### 5. All-to-all

The all-to-all operation can be seen as a collection of simultaneous broadcast or simultaneous gathers.

### 6. Reduce-scatter

`MPI_Reduce_scatter` is equivalent to a reduction on an array of data (meaning a pointwise reduction on each array location) followed by a scatter of this array to the individual processes.

### 7. Barrier

A barrier is a routine that blocks all processes until they have all reached the barrier call. Not very useful.

### 8. Variable-size-input collectives

In `MPI_Scatterv` the root process has a different amount of data for each recipient.

In `MPI_Gatherv`, conversely, each process contributes a different sized send buffer to the received result; `MPI_Allgatherv` does the same, but leaves its result on all processes; `MPI_Alltoallv` does a different variable-sized gather on each process.

### 9. Scan operations

The `MPI_Scan` operation also performs a reduction, but it keeps the partial results.

### 10. MPI Operators

MPI operators are used in reduction operators, and it is possible to define a new operator.

### 11. Non-blocking collectives

With a non-blocking barrier: posting the barriers means that a process has reached a certain state; the request being fulfilled means that all processes have reached the barrier.

### 12. Performance of collectives

One way to ameliorate the speed of data movement is to structure the broadcast in a tree-like fashion.

### 13. Collectives and synchronization

## Exercises

### 3.1 a) Broadcast, reduction

b) Broadcast, reduction, scatter

c) Broadcast, gather, reduction

### 3.8 Gather and reduction

// print the maximum random value and the position

Float myrandom, maxrandom;

Int procno, nprocs;

myrandom = (float) rand();

MPI\_MAXLOC();

printf(" Result max random %6.9f, processor number %d \n", maxrandom, procno.);

3.14

$$p*\alpha + n*p*\beta$$

## Questions

1. what is synchronization? Are they operations that are said to be 'collective on their communicator'? Why don't they involve data movement?

2. do not understand the code on page 32.

3. what is the difference between function MPI\_Allgatherv and MPI\_Alltoallv?

4. what is an exclusive scan?