CS406: Parallel Computing
Homework 3 — MPI All-to-All

Sabancı University

December 22, 2025

Problem Definition

In this homework you will implement the behavior of `MPI_Alltoall` with pairwise communication primitives `MPI_Send` and `MPI_Receive`. Your function will be used to broadcast a number of integers and the number of processors will always be a power of 2. You can find the function signature and the testing environment in the accompanying source code file main.cpp. Performance is **not graded** this time; however, you must:

- Produce **correct results** (required for credit),
- Write a short report that compares the runtime of your implementation against `MPI_Alltoall` for **small vs. large messages**.

What is All-to-All? Assume there are P MPI processes. Each process has a send buffer that consists of P blocks, and each block is destined to a different process. After an all-to-all, every process receives one block from every other process.

What is Provided (Starter Code)

We provide a single file `main.cpp` that already includes:

1. A baseline run using `MPI_Alltoall` (for comparison),
2. A call to your function `MPI_Alltoall_int` (the part you implement),
3. Timing measurements (milliseconds),
4. A strict correctness check.

Where to Edit

You must edit the following function in `main.cpp`:

```
void MPI_Alltoall_int(const int* sb, int* rb, int msg_size, MPI_Comm comm);
```

Rules / Constraints

- **Do not use collectives** inside `MPI_Alltoall_int`: do NOT call `MPI_Alltoall`, `MPI_Allgather`, `MPI_Allreduce`, etc.
- You **may** use pairwise routines such as:
  - `MPI_Send`, `MPI_Recv`
- Avoid deadlocks.

Program Arguments and Output

The executable accepts the following optional arguments:

- `--iters N` : number of iterations per message size (default: 20)
- `--sizes a,b,c` : comma-separated list of message sizes (in **ints**)
  Example: `--sizes 1,128,4096,100000`

  Your submission must print `RESULT: PASS`**.**

Build & Run (gandalf.sabanciuniv.edu)

Example workflow on the department cluster:

Compiling (example)

```
mpicxx main.cpp -O3 -o all2all.out
```

Running (example)

```
mpirun -np 16 ./all2all
mpirun -np 16 ./all2all --iters 50 --sizes 1,128,4096,100000,500000
```

What to Report

Include:

- A runtime table comparing:
  - baseline: `MPI_Alltoall`
  - custom: your `MPI_Alltoall_int`
- Build and execution details – e.g., how did you compile the code (which instruction set etc.) and how I can reproduce your results.
- Charts and tables must have correct and appropriate captions.
- A discussion: why results change with message size and process count.
- Process counts: $P \in \{4, 8, 16, 32\}$
- Message sizes (ints): $\{1, 128, 4096, 100000, 500000\}$

Deliverables

Submit a single `zip` file named:

`surname_name.zip`

containing:

1. `main.cpp` (with your implementation),
2. `report.pdf`.

Good luck.