HPC4DS - Benchmark

Guglielmo Boi, Marco Piccoli

17/10/2025

Report

For the benchmark exercise we wrote a 2-processes MPI program that does the following:

- Process A iteratively: calls *MPI_Wtime* and stores its result into *start_time*; sends one message of x bytes to process B as a buffer of chars. x starts from 1 and goes up to 1073741824 (approximately 1 GB), which is the maximum number of bytes we managed to send without encountering any issue.
- Process B receives the message and immediately sends it back to process A.
- Process A receives the second message, calls *MPI_Wtime*, and stores its result into *end_time*.

The *elapsed_time* can then be calculated as the difference between *end_time* and *start_time*.

We run the program 100 times for each of the four different PBS placing strategies and calculate the average elapsed times to have a more statistically accurate result.

The throughput rate can be calculated with the following formula, considering that during the time taken the message is sent twice:

$$\text{throughput rate} = \frac{bytes * 2}{average\ elapsed\ time}\ [\text{B/s}]$$

Plots

We managed to visualize the four different graphs altogether in one plot, using a simple script in python. Below you can find 2 plots that describe:

- The average time taken to send and receive the message, over the number of bytes of the message itself, as you can see in Figure 1.
- The bandwidth capacity, measured by the throughput rate over the number of bytes of the message, as you can see in Figure 2.

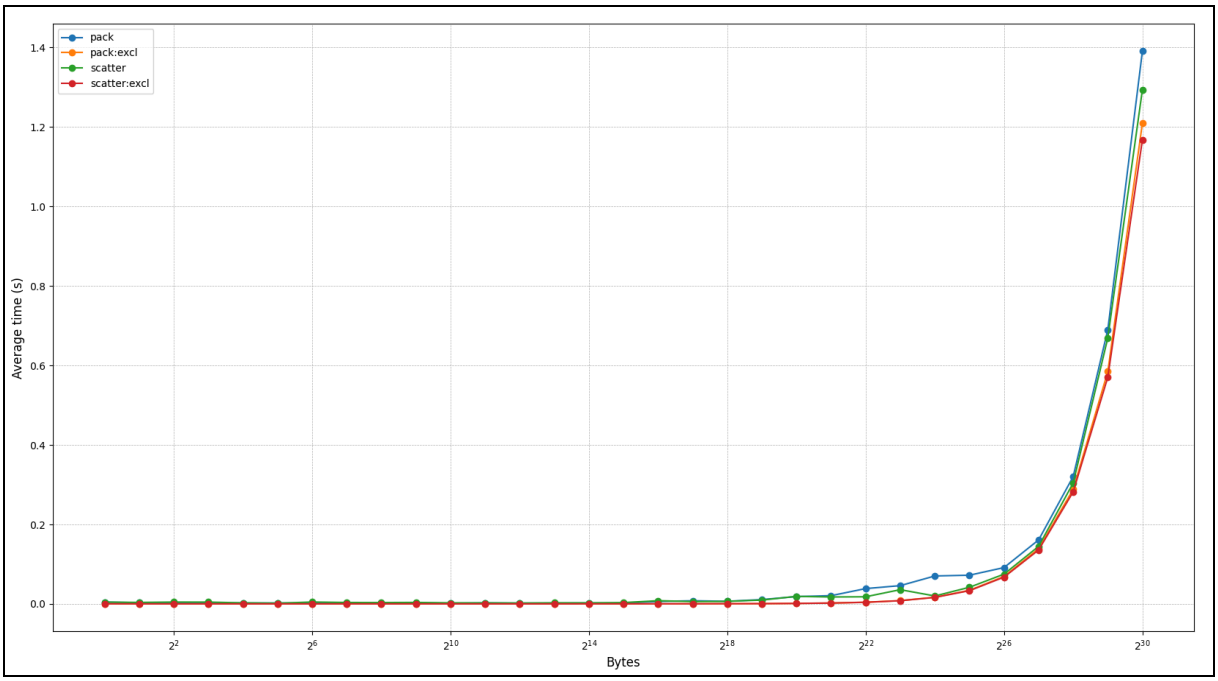

Figure 1: Average time (s)

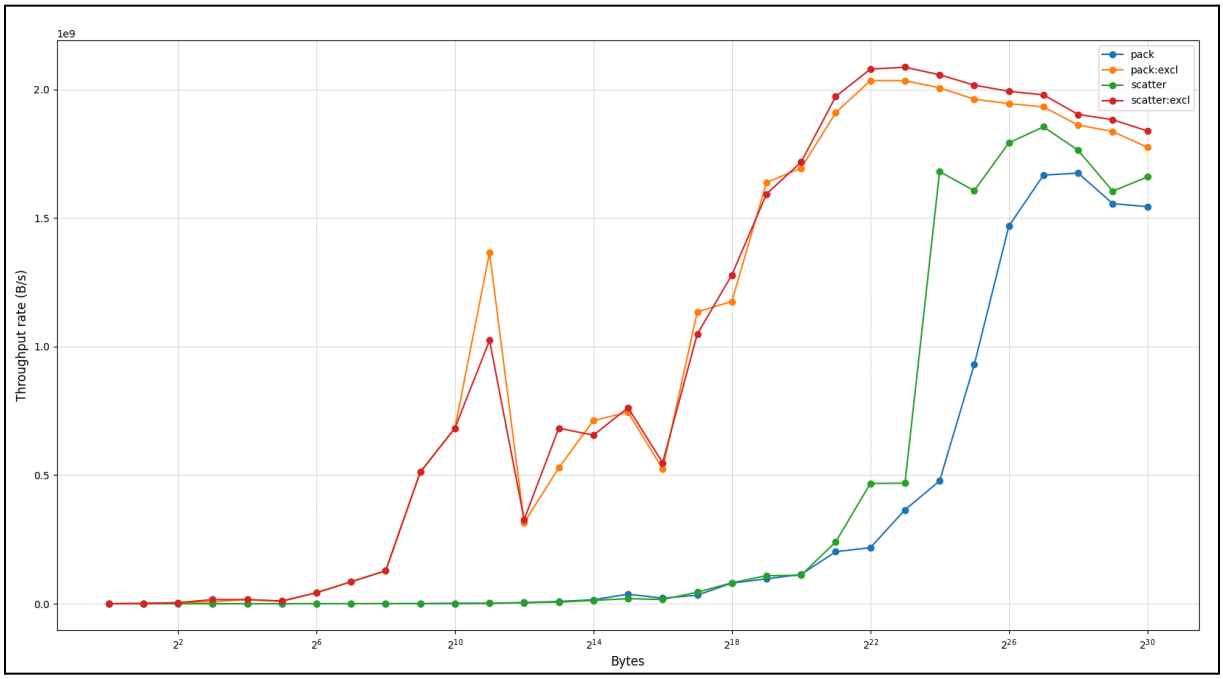

Figure 2: Throughput rate (B/s)

Observations

- The excl placing strategies have a greater rate compared to their non exclusive counterparts, as expected.
- The scatter placing strategies perform slightly better than the pack strategies, which means that the communication between different nodes is a bit faster than the communication in a single node.