CS633 - Assignment-1

Group - 21

Deven Joshi - 210326 (devenaj21@iitk.ac.in)
Shishir Gujarey - 210977 (shishirg21@iitk.ac.in)
Shreyash Kumar - 211009 (shreyashk21@iitk.ac.in)

August 30, 2024

Introduction

The assignment aims to average out the values in a matrix, making use of **grid decomposition** to parallelize the process. Two types of averaging methods, namely the **5-point stencil** and the **9-point stencil** have been implemented. The **halo-exchange** paradigm in parallel computing has been invoked to carry out the necessary data retrievals between the processes.

Initialization

- **buf:** local matrix for each process; initialized with random seed at the beginning.
- **send_dirn:** the four arrays are used to send the packed arrays for 5-point stencil
- **recv_dirn:** the four arrays are used to receive the packed arrays for 5-point stencil
- **send_dirn_2:** the four arrays are used to send the packed arrays for 9-point stencil
- **recv_dirn_2:** the four arrays are used to receive the packed arrays for 9-point stencil
- **arr_dirn**: the four arrays are used to unpack the received arrays and are finally used for average computation by the cells(except for interior cells)

Communication

The following calls have been used to facilitate communication within the MPI environment:

- **MPI_ISend:** non-blocking function used to initiate send operation between processes
- **MPI_Recv:** blocking function used to receive the data sent by another process
- **MPI_Barrier:** blocking function used to synchronize all processes in a communicator
- **MPI_Reduce:** collective operation used to used to cumulate data from all processes

Code Explanation

- **Initialization:** The code parses command-line arguments, including the number of processes in each dimension (Px), the grid size (N), the number of time steps (steps), the random seed (seed), and the stencil type (stencil). Each process then initializes its sub-domain with random values using the provided seed. Processes then determine whether they have neighboring processes to communicate with using flags.
- **Halo-Exchange:** Each process packs its outlying data in the *send_dirn* using *MPI_Pack*. Packed data is then sent to the corresponding process where the data is required. The packed data is then received in the *Recv_dirn* using *MPI_Recv*. The received data is then unpacked into the *arr_dirn* using *MPI_Unpack*.
- **Computation:** Values in the cells are averaged out using the 5-point or the 9-point stencil techniques. Various edge cases are handled using flags, taking care of the denominators of the average.
- **Reporting:** The code calculates the execution time for 10 time steps of the aforementioned algorithm for each process and reports the maximum time using *MPI_Reduce*. Note that this is the overall execution time for the problem statement.

Optimizations

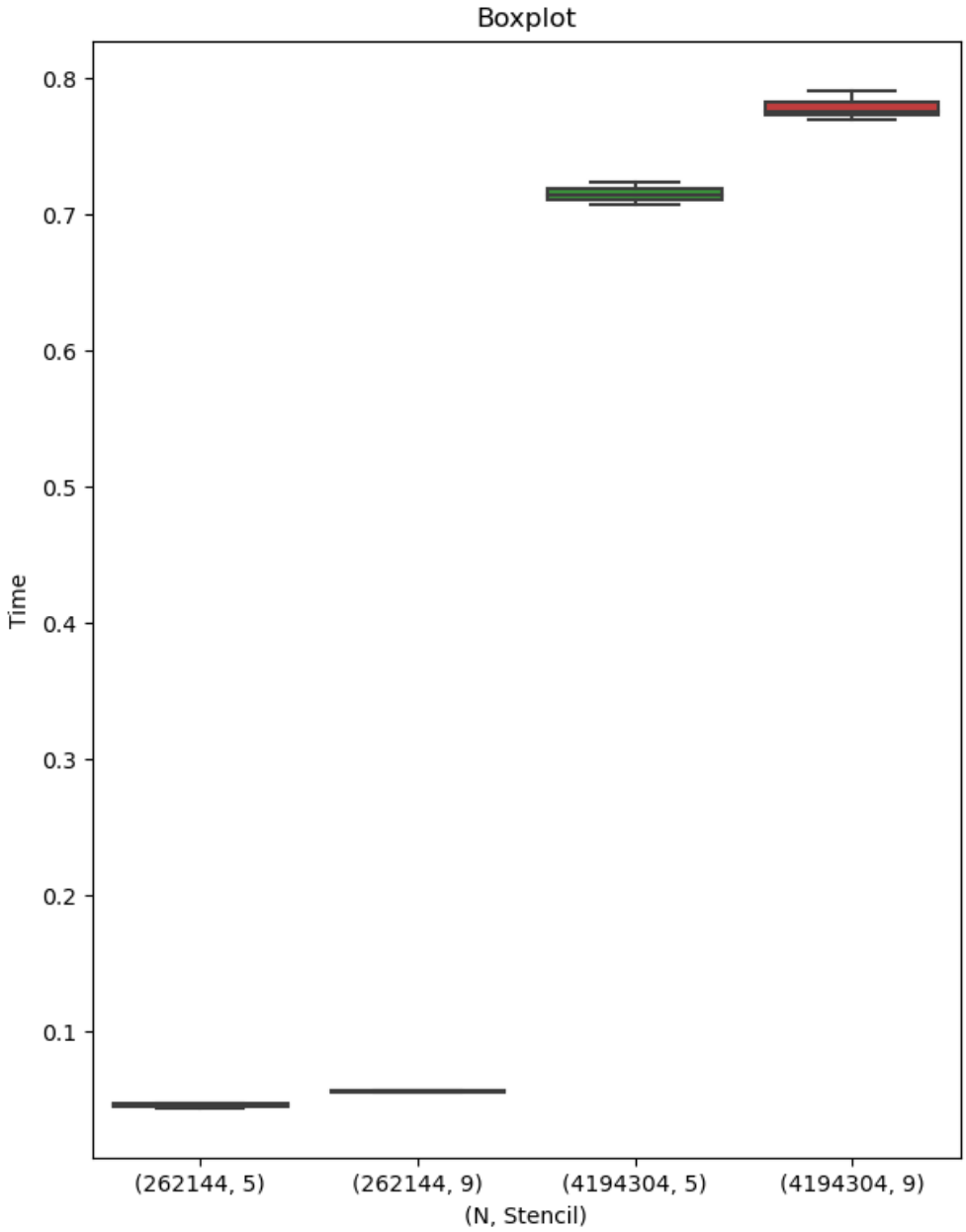
- We have used *MPI_ISend* instead of *MPI_Send*, which allows for asynchronous communication and overlapping communication with computation, potentially improving overall performance by enabling the sending process to perform other tasks while the message is being sent.
- For the 9-point stencil computation, we have packed two arrays and sent them as one, reducing the number of communications between the processes, thereby reducing process execution time.

Timing Data

On executing the above code three times, we get the following output:

N	**Stencil**	**Execution Time**		
		Time 1	**Time 2**	**Time 3**
262144	5	0.045870	0.044295	0.047948
262144	9	0.056290	0.055434	0.056771
4194304	5	0.714495	0.724083	0.707778
4194304	9	0.790445	0.770300	0.775428

Plot and Observations



From the above plot, we can make the following observations:

- There is a huge difference between the time taken for N=262144 and N=4194304. Hence, we can say that the time taken largely depends on the input data size (number of data points per process).
- For the same N, we can see that the time taken for 9-Point Stencil is greater than that for 5-Point Stencil. This is due to greater number of computations and greater size of data sent via ISend in the case of 9-Point Stencil.