Project 1 - Final Specification

Name: Arushi Agarwal
CNet Id: arushiag
Github username: arushiag12

Introduction

The final report for Project 1 implements a distributed-memory parallel Lax-Wendroff scheme to numerically solve the advection equation. The parallelization is achieved using MPI and OpenMP, adhering to the specified initial condition outlined in the project writeup. The execution time measurements are conducted with N = 4000 while plots are generated with N = 1000. The rest of the parameters utilized are:

- L = 1.0
- T = 1.0
- u = sqrt(2) y
- v = - sqrt(2) x
- del_t = 1.25e-4

In terms of code structure, the initialization phase assigns appropriate values to squares in each process based on the initial condition. Subsequently, during the update for each time step, when the code is executed with more than one MPI process, it employs a variant of the 'checkerboard' approach for sending and receiving across processes. Specifically, the checkerboard approach is used for trading the internal ghost cells (excluding those on the boundary) of the final matrix. Then the code exchanges right ghost cells with the left, and vice versa, while similar exchanges occur for the up-down cells.

Code Compilation

The files included in the repository other than this pdf are:

- adv_final.c : The file containing the C code implementing the MPI and OpenMP hybrid model
- project1-plots.ipynb : The python code used to generate plots from the text files produced by adv_final.c

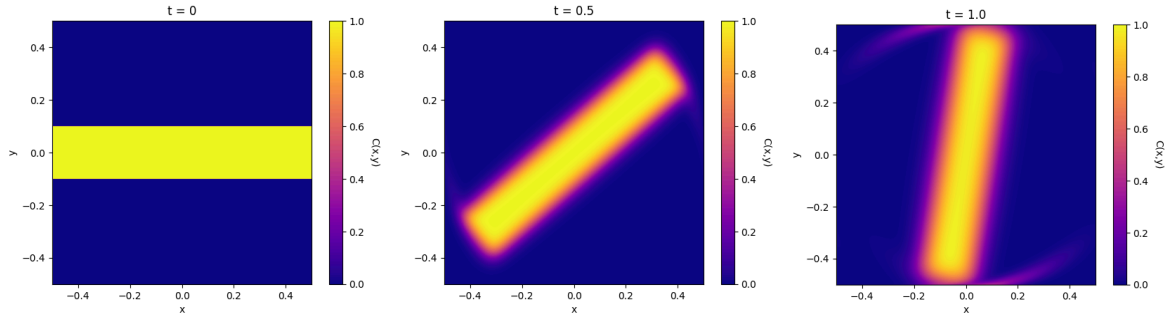
The code was compiled and run on midway using the following specifications:
$ mpicc adv_final.c -o adv_final.exe -lm -fopenmp -O3
$ mpirun –bind-to socket ./adv_final.exe <no of cores per node> <output: true(1)/false(0)>

The nodes and cores were specified using the following lines in the batch script:
#SBATCH --nodes=<no of nodes>
#SBATCH --cpus-per-task=<no of cores per node>

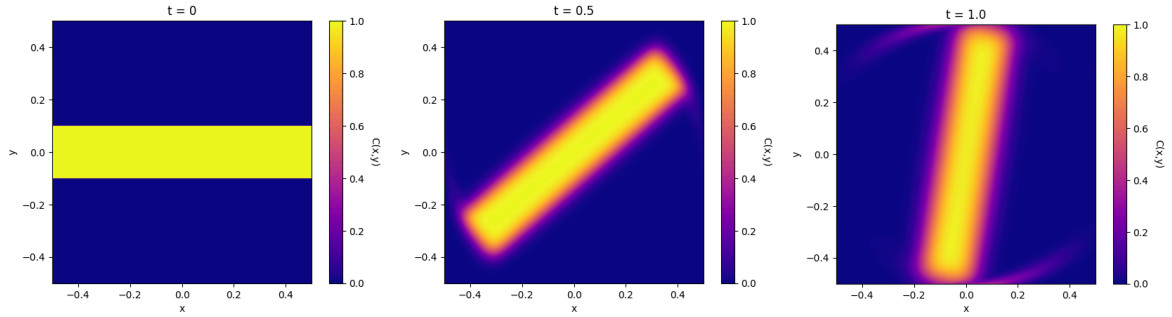
Verification

N = 1000.

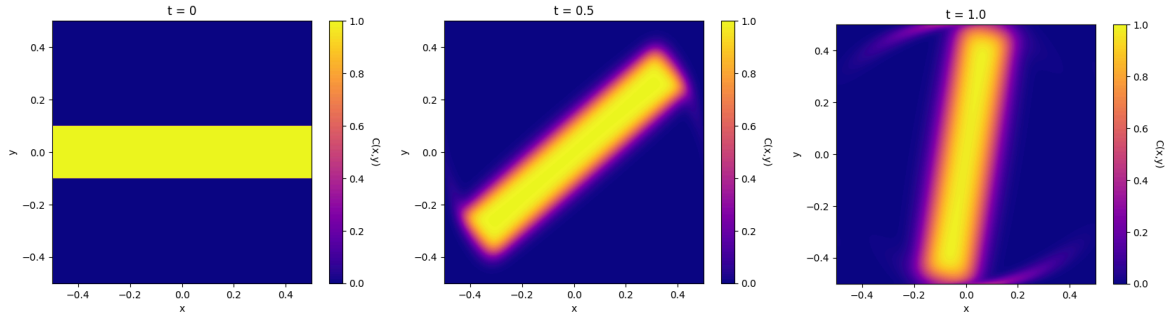
- Serial Lax.



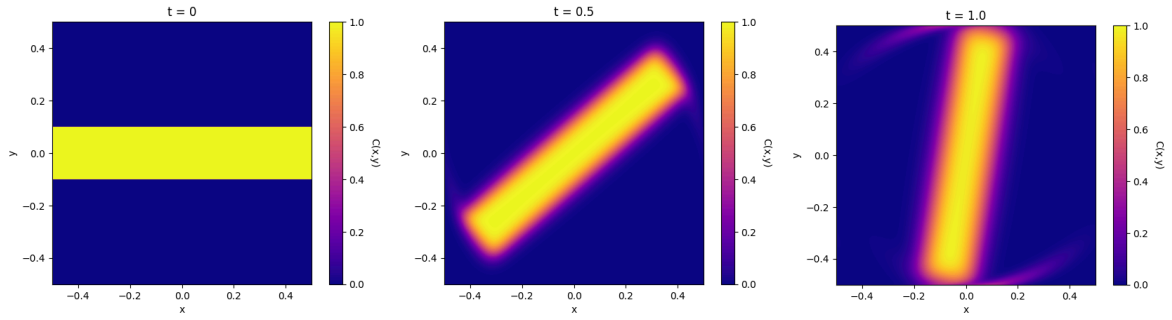
- Shared Memory Parallel Lax (16 cores).



- Distributed Memory Parallel Lax (4 nodes, 1 core per node).



- Distributed Memory Parallel Lax (16 nodes, 16 cores per node).



Performance

N = 4000.

Model	Nodes	Cores per Node	Execution Time (secs)	Grind Rate (cells/sec)
MPI	1	1	755.2	1.69e8
MPI	2x1	1	478.1	2.68e8
MPI	2x2	1	156.6	8.17e8
MPI	3x3	1	79.4	1.61e9
MPI	4x4	1	51.3	2.49e9
OpenMP	1	1	755.2	1.69e8
OpenMP	1	2	328.9	3.89e8
OpenMP	1	4	181.4	7.05e8
OpenMP	1	9	88.3	1.44e9
OpenMP	1	16	51.6	2.48e9
Hybrid	1	16	51.6	2.48e9
Hybrid	2x1	16	30.1	4.02e9
Hybrid	2x2	16	11.8	1.08e10
Hybrid	3x3	16	6.4	1.99e10
Hybrid	4x4	16	5.0	2.54e10