

Parallel and Distributed Processing

Assignment 03

OpenMP and Distributed Programming

Attique Dawood

10-03-2025

Due: 21-03-2025 2359 Hours

Last Modified: Monday 10th March, 2025, 10:15

Practice: Create a client/server app where the server keeps a list of files (pdf, docx, etc.). The client can then download a file from the list provided by the server. Open downloaded file to check if everything went well. This will help you with data transfer for the distributed implementations.

Question 1: Make a C++ application to sum an array of decently large size. Implement this using:

1. Serial implementation without any parallel processing. You can use this to verify the correctness of parallel implementation.
2. OpenMP on a single machine.
3. Distributed implementation (using sockets) on at least two machines on a network. You can still use OpenMP (or traditional threads) to utilise the distributed machines effectively.

Question 2: Repeat Question 1 for matrix multiplication.

Question 3: Repeat Question 1 by implementing 2-D Laplace's equation solver. For boundary conditions, use $top = 5\text{ V}$, $bottom = -5\text{ V}$, $left = right = 0\text{ V}$. This is effectively a capacitor.

Note 1: You should implement all three parts (single, OpenMP and distributed) as one programme/exe for each question.

Note 2: Please create three folders for these three questions named 'ArraySum', 'MatMul' and 'Laplace.' Add these folders to a zip file with your name and submit that (along with a pdf containing graphs etc). For example, `Attique_Dawood.zip`.

Note 3: Please use the makefile from assignment 02 for these questions and remove any object/executable files before submission. You can change makefiles for added functionality (e.g., adding `-fopenmp`).

Note 4: Please do mention the hardware specs and plot graphs to show performance speed-up. Show 2-D contour plots of MatMul and Laplace as well. Visualisation can help check if the solution is correct.

Note 5: Extra marks if you can show that the distributed implementation can solve a large problem that cannot fit in the memory of a single PC.