

Ex 10. Parallel Conjugate

Gaurav Kukreja, Evangelos Drossos

We have been unable to solve the exercise 10. Our parallel implementation of the algorithm does not work as expected.

This document explains the approach we took, and the issues we have identified.

To partition the problem among the threads, we created blocks of the inner cells of the grid, and provided 2 rows and 2 columns extra, for holding the values from the adjacent block of the matrix, handled by another thread.

Initialization of the matrices should be done in parallel by each thread. We will initialize the edges of the matrices. Due to some logical issues, we are unable to get this to work. We are looking into the issue currently. We have identified that because of the choice of dividing the the matrix in the way we have chosen, the mesh width is generally an irrational number. Which is `grid_points_1d` is calculated incorrectly, which is leading to problems.

Most operations like copy, and dot_product are trivial. `MPI_Allreduce` was used for dot_product.

The `g_product_operator` function has been modified such that each thread first exchanges the corresponding columns with processes on the left and right, and corresponding rows with processes on top and bottom. Vectors have been transferred. Hereafter, we can perform the computation trivially once, we have all the required data.

Again, we haven't been able to verify our solution. I think if we change the topology for dividing the problem in our approach, we may be able to solve the problem.