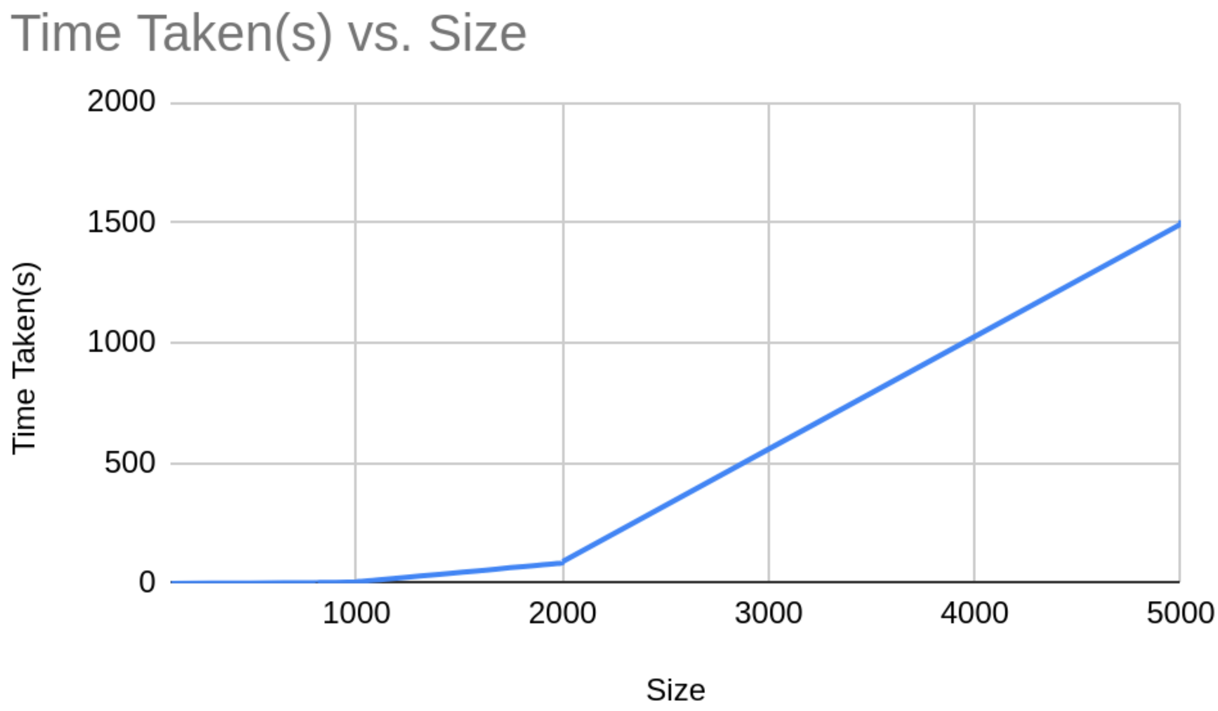
MTCS-103P Lab Work- 5 & 6

Parallel Matrix Multiplication

Regd no: 22555

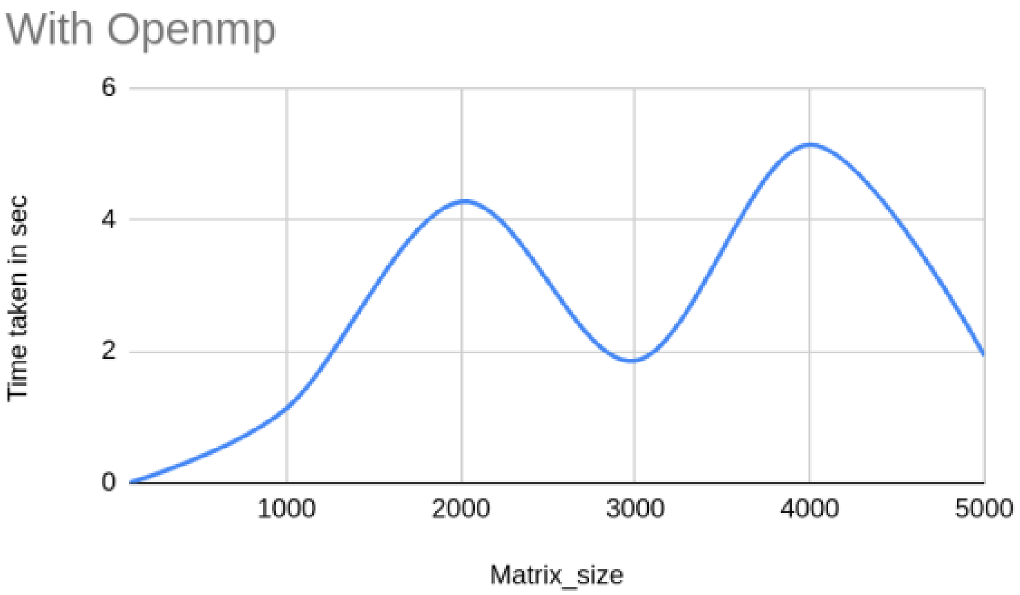
Given 2 matrices of the same size, we need to write a parallel code of matrix multiplication. Allocating memory and then performing the multiplication using **#pragma omp parallel for collapse**, we have seen that it does much better than the serial program of the Matrix Multiplication. The average number of executions that were considered was 3. Thereafter the average was taken to plot the graphs.

The graph of the time taken v/s size of the matrix that has been taken for the serial case is given below



From the above graph, we can see that the time that is calculated in seconds is very high and it keeps increasing as we increase the size of the matrix.

In the parallel program, we see that the time is much lesser than in the serial program. The graph for the parallel program is given below.



The above Graph shows that the parallel code takes much lesser time than the serial code of the matrix multiplication. In the graph, we see that there is a steep rise in the time around the matrix size 2000 and 4000 as there is false sharing of the data that is occurring.

The performance also depends on how the matrices are stored in the hardware. By default, C language stores 2-dimensional arrays in row-major order. If we transpose the second matrix and then multiply the matrices with the changes accordingly, we might have a much better result.

If we use sub-matrix multiplication for the serial code, the time taken for multiplication will decrease but it will not be comparable to the time taken by the parallel code as there are multiple threads that are handling the multiplication simultaneously.