# CSE 570 Parallel and Distributed Processing

**Ashwin Jadhav**

**ajadhav5**

**50405435**

(A) Algorithm written for a0.cpp to implement fast application of 2D filters.

(1) Once the filter_2D function is called by a0.cpp, the a0.hpp takes the large vector/matrix and extracts a sub-matrix of size-9 (3*3 for kernel).

(2) After that the sub-matrix is then sent to another function called mat_mul.

(3) I am using two nested for loop to accomplish the above task.

Moreover, I have also use # praga omp constructs for the inner for loop to accomplish the extraction of sub-matrix parallely.

(4) Furthermore, I have also used used reduction while matrix multiplication in function mat_mul.

(B) Time Complexity

$$T(n) = O(3^n n^2) + O(n^3)$$

So, $T(n) = O(3^n n^2)$ is

for extraction of sub-matrix and $T(n) = O(n^3)$ is for matrix multiplication.

(c) Is code scalable or not?

(1) Yes, my algorithm is strongly scalable for the number of processors. As for example (please review below graphs as well), while keeping the same data my execution time for 2 and 6 threads decreases substantially.

(2) Further, my algorithm loses its strong scalibity when the threads are large like 12 - threads and for large data like for 35k * 35k vector data.

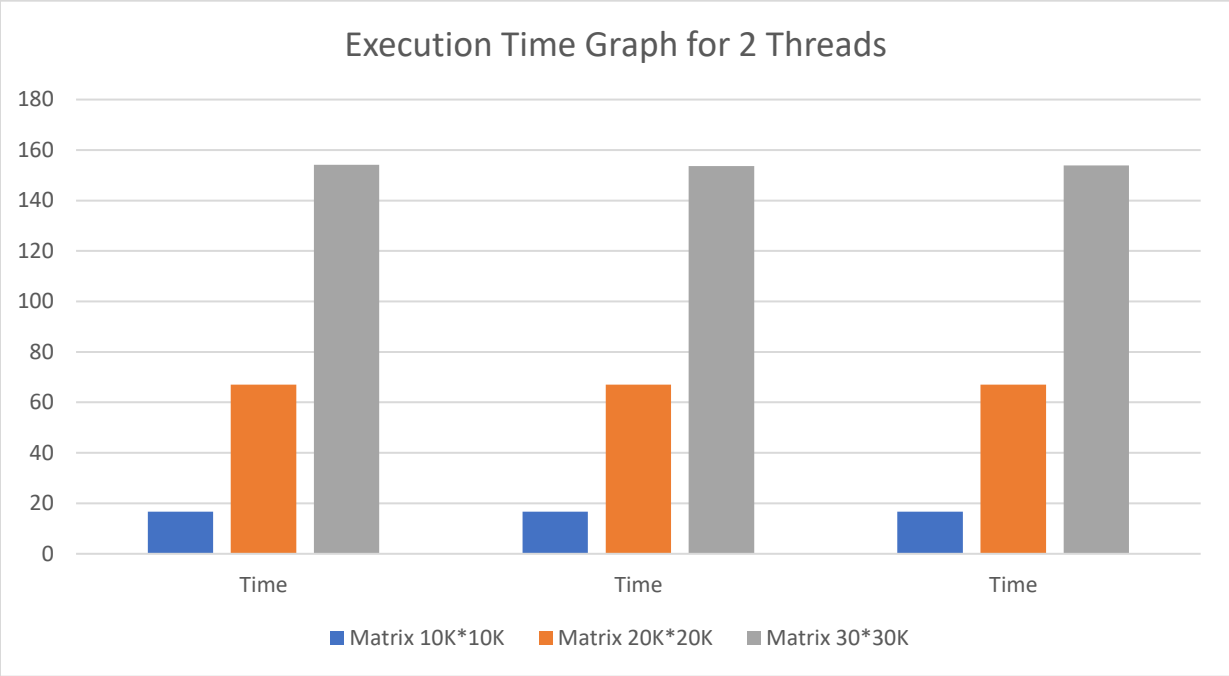
(1) Why my algorith is not scalable?

(1) My algorithm is not scalable for very large data and threads because my sub-matrix extraction is done using nested for

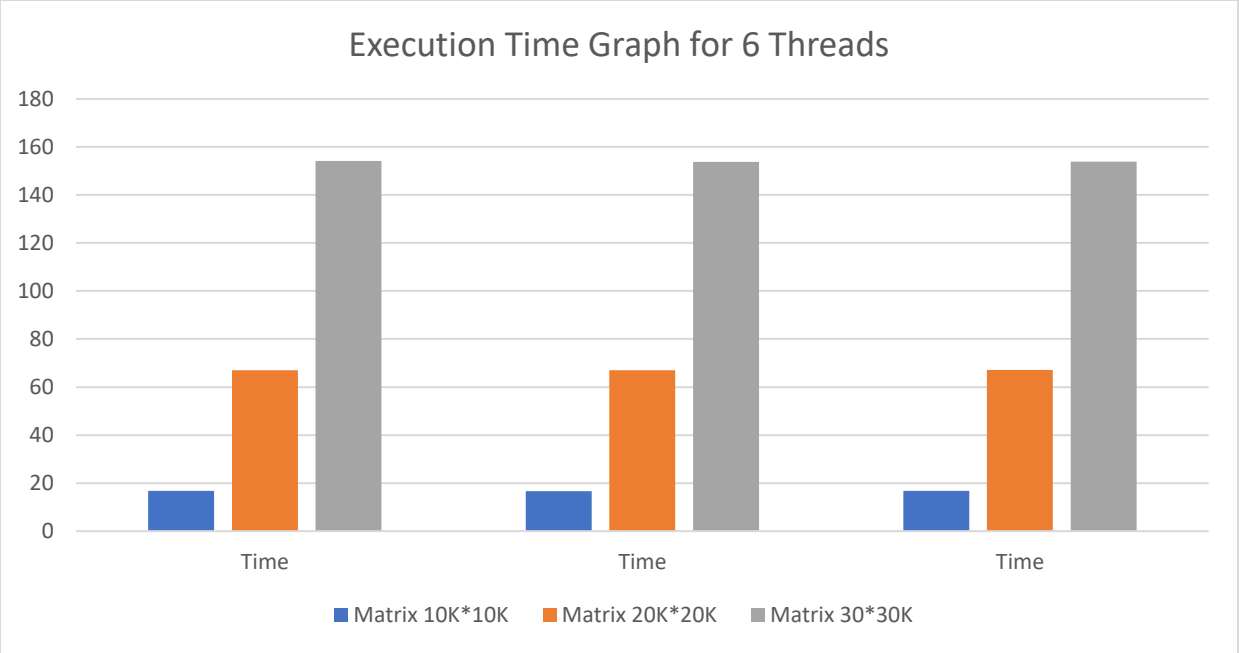
loop and I am using "#pragma omp for" for parallelism.

(2) To make my algorith very strongly scalable, I should have used the "task" constructs.

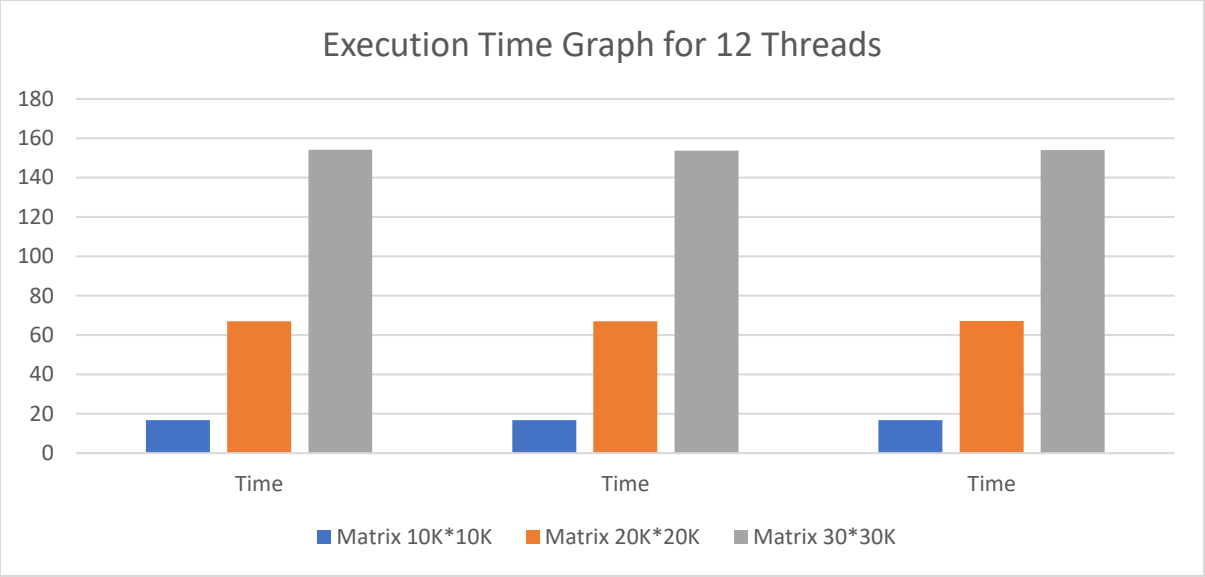
# Parrell Execution




| | Time | Time | Time | |
|---|---|---|---|---|
| **Matrix 10K*10K** | 66.9649 | 66.9227 | 67.1232 | |
| **Matrix 20K*20K** | 152.383 | 152.112 | 152.126 | |
| **Matrix 30*30K** | 327.267 | 337.87 | 341.495 | |
| | | | | |

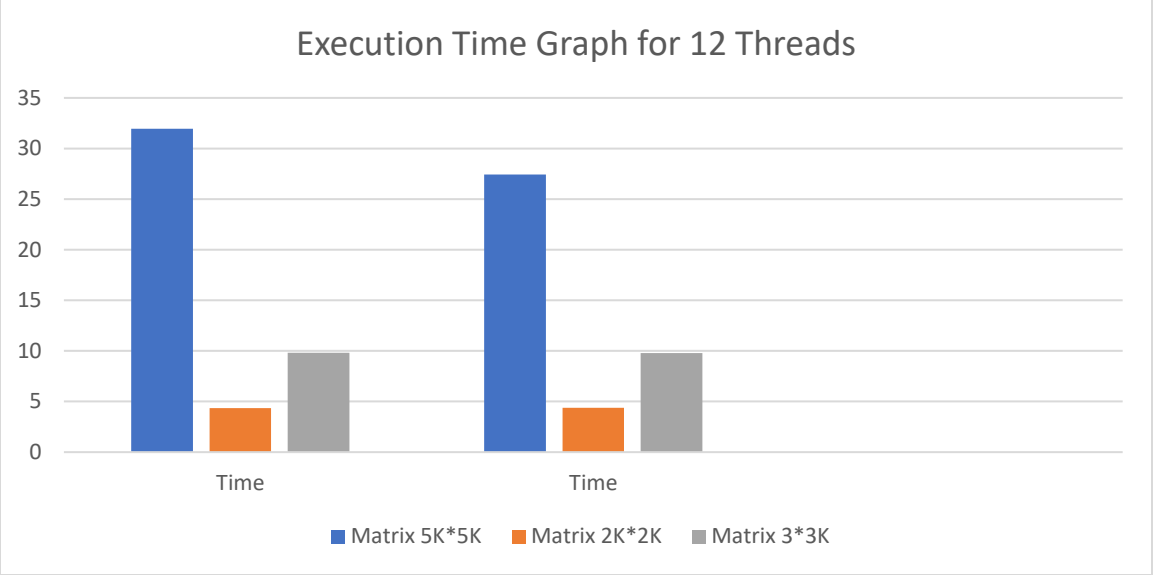



| | Time | Time | Time | |
|---|---|---|---|---|
| **Matrix 10K*10K** | 16.725 | 16.7362 | 16.7276 | |
| **Matrix 20K*20K** | 67.182 | 67.1668 | 67.0347 | |
| **Matrix 30*30K** | 151.927 | 152.399 | 151.834 | |
| | | | | |




| | Time | Time | Time | |
|---|---|---|---|---|
| **Matrix 10K*10K** | 16.7105 | 16.6775 | 16.7198 | |
| **Matrix 20K*20K** | 67.0194 | 67.0529 | 67.0956 | |
| **Matrix 30*30K** | 154.102 | 153.696 | 153.94 | |
| | | | | |

# Serial Execution takes too much time so used smaller data




| | Time | Time |
|---|---|---|
| **Matrix 5K*5K** | 31.942 | 27.42 |
| **Matrix 2K*2K** | 4.3517 | 4.378 |
| **Matrix 3*3K** | 9.808 | 9.788 |
| | | |