01-Report

July 9, 2020

1 Matrix Multiplication: Homework

1.1 Exercise 1

- Generalize the implementation to deal with non-square matrices.

Solution: Padding method is used to generalize the implementation. First, `strassen_aux_rec` function changed to deal with rectangular matrices. However this method didn't work most probably due to the bug and unfortunately I couldn't find the bug. Although S matrices were correct, P matrices return 0 so there was a problem with implementation. Implementation can be found in `strassen_rec.c` file.

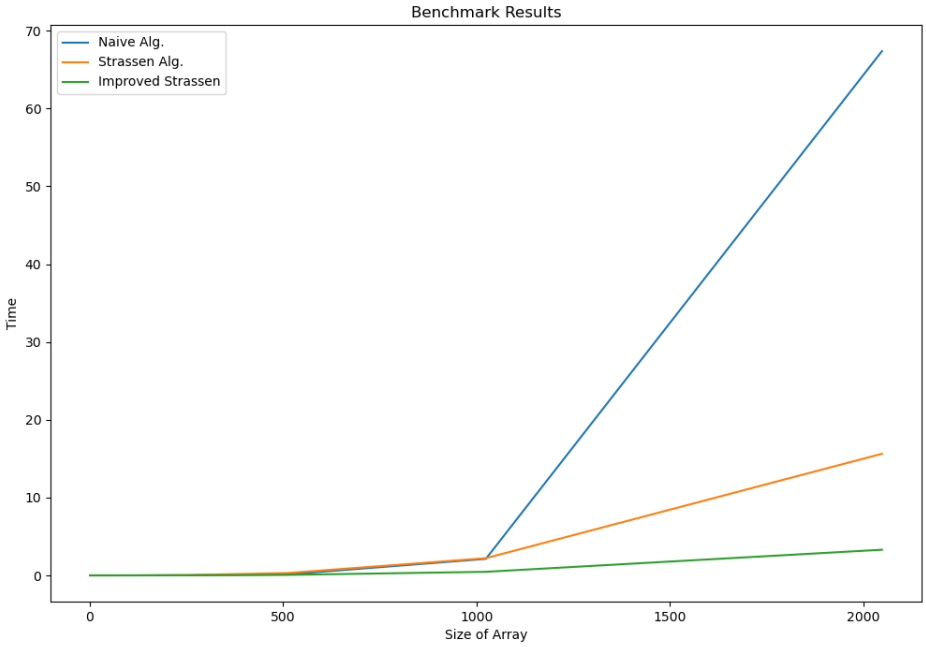
As an alternative method, which is very similar to first idea, matrices padded to closest square matrices so that one can use the existing template. This implementation can be found in `improved_strassen.c`

1.2 Exercise 2

- Improve the implementation of the Strassen's algorithm by reducing the memory allocations and test the effects on the execution time.

Solution: One can easily notice that some of the allocations of matrices S and P are redundant. In other words same operations can be performed by using less allocation. Therefore in `improved_strassen.c` file memory efficient version of strassen algorithm can be found. Basically, allocation on `S` matrices reduced from 10 to 2 since we use at most 2 matrices in one operation. Same idea could be implemented to the `P` matrices because same operations can be done with 3 allocation. However for simplicity, allocations of P matrices were left as it is.

Belove figure you can find the comparasion of naive, strassen and improved strassen.


Benchmark Results
Naive Alg.
Strassen Alg.
Improved Strassen
Time
Size of Array
0
10
20
30
40
50
60
70
0
500
1000
1500
2000