Analyzing Performance of Matrix Multiplication Algorithms in OpenMP

R Mukesh (CED15I002)

IIITDM Kancheepuram

August 26, 2018

Abstract

Parallelism is the key to improving performance of algorithms. This experiment aims to analyze the performance of parallelized matrix multiplication algorithms in OpenMP, for various matrix storage schemes (row-major, column-major and block matrix storage order).

1 Results [1]

The experiments were performed on double datatype matrices with a dimension of 2048 x 2048 elements. The block size of the block-order matrices was taken as 4 x 4 elements.

Number of Threads	Execution Time (in seconds)		
	Row-Major Matrix	**Column-Major Matrix**	**Block-Order Matrix**
Without OpenMP	39.623860	39.434041	49.235846
1	49.994157	50.452567	49.811989
2	26.935359	27.512718	25.561742
4	24.656942	24.657366	24.169181
6	24.793141	24.703847	24.635781
8	24.742557	24.767288	24.672302
10	24.682174	24.731551	24.779522
12	24.662531	24.803336	24.766994
16	24.713110	24.986742	24.749395
20	24.701823	24.862698	25.155818
24	24.748892	24.868193	25.022288
28	24.765814	24.891739	24.972449
32	24.770109	24.860866	24.869343

2 Calculation

The parallel fraction of the algorithm can be computed using the following equation:

$$f = \frac{(1 - T_p/T_1)}{(1 - 1/p)} \tag{1}$$

For Row-Major Matrix Multiplication Algorithm,

$$f = \frac{(1 - 24.656942/49.994157)}{(1 - 1/4)} = \mathbf{0.675738033\ (67.57\%)}$$

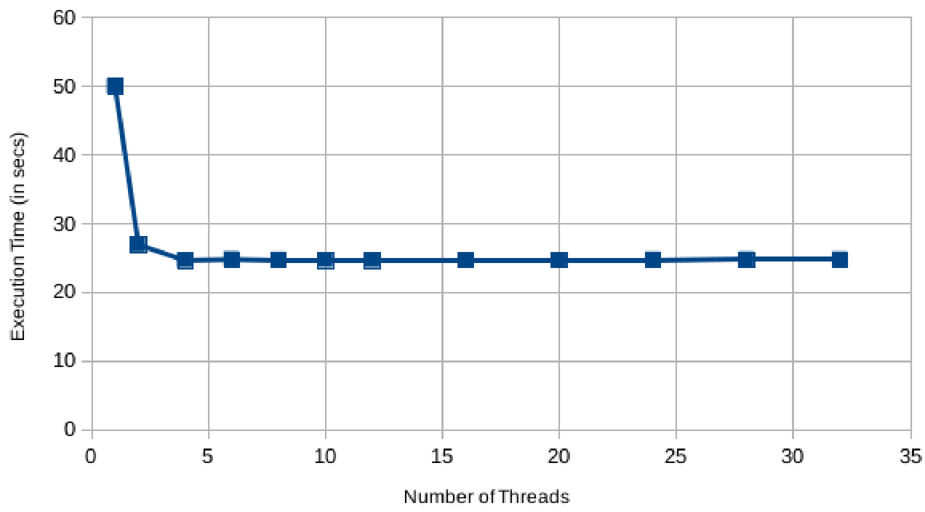
For Column-Major Matrix Multiplication Algorithm,

$$f = \frac{(1 - 24.657366/50.452567)}{(1 - 1/4)} = \mathbf{0.681701713\ (68.17\%)}$$

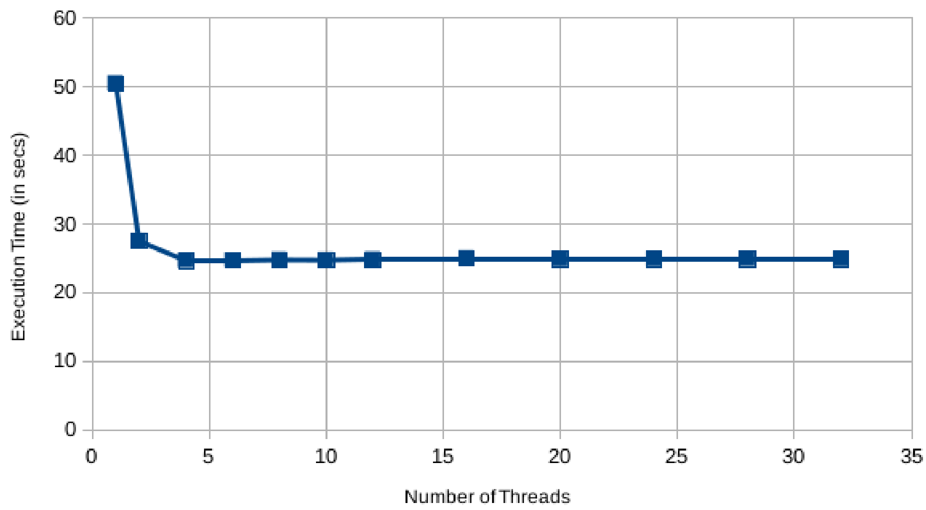
For Block Matrix Multiplication Algorithm,

$$f = \frac{(1 - 24.169181/49.811989)}{(1 - 1/4)} = \mathbf{0.686389188\ (68.63\%)}$$

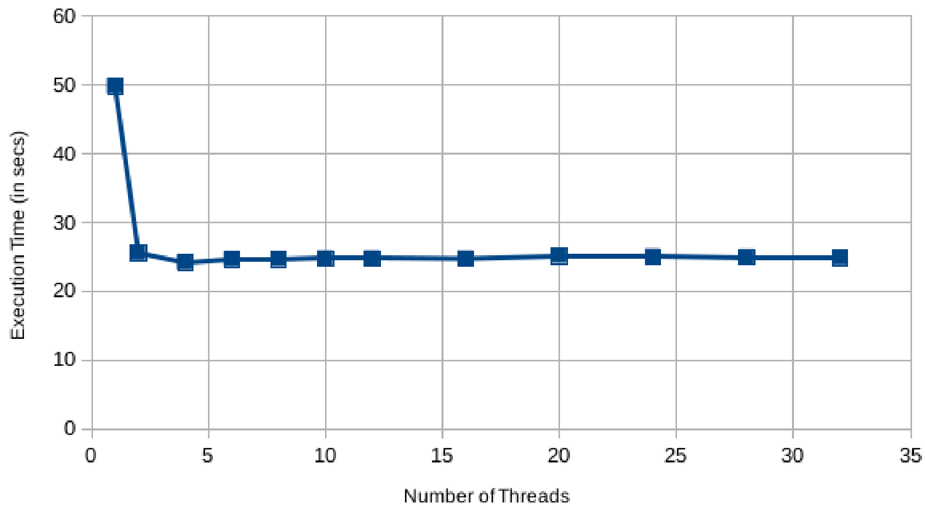
[1] The experiments were conducted on a computer with 6^{th} Generation Intel(R) Core(TM) i7-6500U Processor (4M Cache, upto 3.10 GHz) and 4GB Single Channel DDR3L 1600M Hz (4GBx1) RAM.



(a) Performance of Row-Major Matrix Multiplication



(b) Performance of Column-Major Matrix Multiplication



(c) Performance of Block Matrix Multiplication

Figure 1: Number of Threads vs Execution Time of Row-Major, Column-Major and Block Matrix Multiplication in OpenMP

3 Inferences

1. The matrix multiplication algorithms for the various matrix storage schemes (row-Major, column-Major and block-order) have same parallel fraction (f) of approximately 68%.

2. The execution time of matrix multiplication algorithms drops until the number of threads is 4 and then saturates as the computer hardware has only four cores and therefore supports only four simaltaneous threads.

3. The slight increase in execution time for number of threads beyond four is attributed to the overheads associated with creation and switching of threads.

4. The slightly improved performance of block matrix multiplication over row-major and column-major matrix multiplication is due the reduced number of cache misses that can be attributed to the memory access patterns of the algorithm.