

Multicore Computing

Project #1, Problem 2

I/ Introduction

This document aims to report the performances of a multithreaded matrix multiplication program. This program uses the **static cyclic** strategy.

II/ How to use

Compile and run the program by typing one of the following command lines within the `problem2/` directory of the project:

```
$ java MatmultD.java <number of cores> < <matrix file>
```

III/ Testing environment

RAM: 18 Go

CPU Type: 11-core Apple M3 Pro (6 performance + 5 efficiency)

Hyperthreading: No (1 thread per core)

Clock Speed: Up to 4.06 GHz (P-cores)

Architecture: ARM64 (Apple Silicon)

OS: Sequoia 15.3.2

IV/ Measurements

The multiplied matrices have a size of 1000x1000.

For this test, the chosen task size is 30. It's a good compromise for ideal effectiveness:

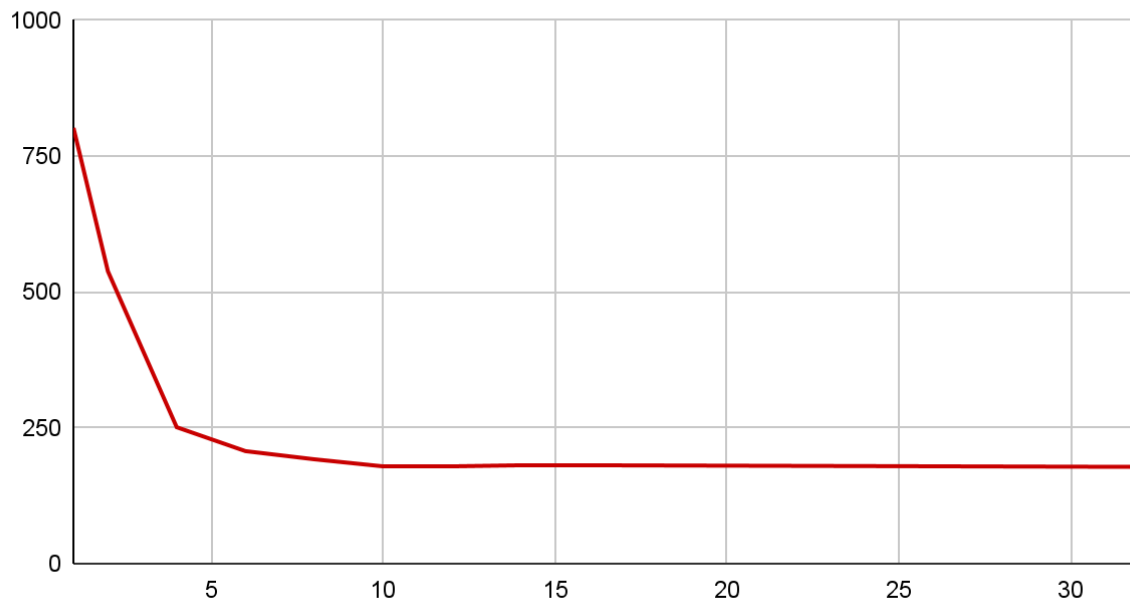
- a too low value would increase the overhead: the ratio between calculation time and task finding time would be too low.
- a too high value may lead to loss of CPU resources: if there are more threads than available tasks, some will remain idle.

The maximum number of threads being 32 here, a matrix of 1000x1000 can be calculated by almost all threads with a task size of 30 ($32 \times 30 = 960$).

Execution time (in ms)

threads	1	2	4	6	8	10	12	14	16	32
exec time (ms)	802	538	251	207	192	179	179	181	181	178

Execution time



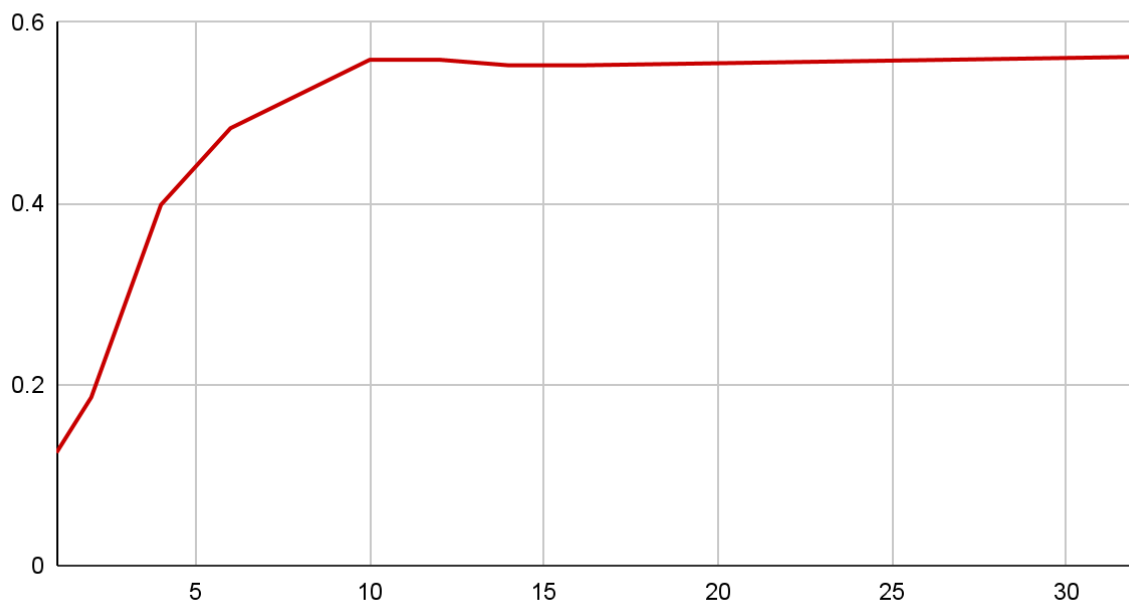
Graph showing the execution time (vertical axis) depending on the number of threads (horizontal axis)

Performance

In order to ease readability, I multiplied the performance by 100. It doesn't affect the relationship between each value.

	1	2	4	6	8	10	12	14	16	32
performance (1/exec time)	0.125	0.186	0.398	0.483	0.521	0.559	0.559	0.552	0.552	0.562

Performance



Graph showing the performance (vertical axis) depending on the number of threads (horizontal axis)

V/ Screenshots

```
+ problem2 git:(main) x java MatmultD.java 1 < mat1000.txt
[Thread-0] Execution Time: 798ms
[thread_no]: 1, [Time]: 802 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 2 < mat1000.txt
[Thread-1] Execution Time: 519ms
[Thread-0] Execution Time: 537ms
[thread_no]: 2, [Time]: 538 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 4 < mat1000.txt
[Thread-3] Execution Time: 222ms
[Thread-2] Execution Time: 222ms
[Thread-1] Execution Time: 233ms
[Thread-0] Execution Time: 251ms
[thread_no]: 4, [Time]: 251 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 6 < mat1000.txt
[Thread-5] Execution Time: 166ms
[Thread-4] Execution Time: 175ms
[Thread-3] Execution Time: 179ms
[Thread-1] Execution Time: 201ms
[Thread-0] Execution Time: 206ms
[Thread-2] Execution Time: 207ms
[thread_no]: 6, [Time]: 207 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 8 < mat1000.txt
[Thread-6] Execution Time: 150ms
[Thread-7] Execution Time: 154ms
[Thread-3] Execution Time: 158ms
[Thread-5] Execution Time: 152ms
[Thread-4] Execution Time: 157ms
[Thread-1] Execution Time: 158ms
[Thread-2] Execution Time: 166ms
[Thread-0] Execution Time: 192ms
[thread_no]: 8, [Time]: 192 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 10 < mat1000.txt
[Thread-7] Execution Time: 127ms
[Thread-9] Execution Time: 138ms
[Thread-4] Execution Time: 146ms
[Thread-8] Execution Time: 148ms
[Thread-5] Execution Time: 151ms
[Thread-6] Execution Time: 151ms
[Thread-3] Execution Time: 159ms
[Thread-1] Execution Time: 169ms
[Thread-0] Execution Time: 174ms
[Thread-2] Execution Time: 179ms
[thread_no]:10, [Time]: 179 ms, [Matrix sum]: 1001895301

+ problem2 git:(main) x java MatmultD.java 12 < mat1000.txt
[Thread-11] Execution Time: 95ms
[Thread-10] Execution Time: 101ms
[Thread-4] Execution Time: 104ms
[Thread-0] Execution Time: 138ms
[Thread-9] Execution Time: 138ms
[Thread-8] Execution Time: 145ms
[Thread-1] Execution Time: 152ms
[Thread-3] Execution Time: 159ms
[Thread-5] Execution Time: 174ms
[Thread-7] Execution Time: 177ms
[Thread-2] Execution Time: 179ms
[Thread-6] Execution Time: 179ms
[thread_no]:12, [Time]: 179 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 14 < mat1000.txt
[Thread-12] Execution Time: 106ms
[Thread-13] Execution Time: 107ms
[Thread-10] Execution Time: 129ms
[Thread-8] Execution Time: 131ms
[Thread-11] Execution Time: 129ms
[Thread-5] Execution Time: 137ms
[Thread-7] Execution Time: 141ms
[Thread-9] Execution Time: 140ms
[Thread-6] Execution Time: 142ms
[Thread-1] Execution Time: 163ms
[Thread-2] Execution Time: 163ms
[Thread-0] Execution Time: 165ms
[Thread-3] Execution Time: 169ms
[Thread-4] Execution Time: 181ms
[thread_no]:14, [Time]: 181 ms, [Matrix sum]: 1001895301
+ problem2 git:(main) x java MatmultD.java 16 < mat1000.txt
[Thread-14] Execution Time: 110ms
[Thread-2] Execution Time: 147ms
[Thread-8] Execution Time: 136ms
[Thread-13] Execution Time: 135ms
[Thread-12] Execution Time: 143ms
[Thread-7] Execution Time: 142ms
[Thread-4] Execution Time: 151ms
[Thread-3] Execution Time: 150ms
[Thread-5] Execution Time: 152ms
[Thread-9] Execution Time: 153ms
[Thread-11] Execution Time: 151ms
[Thread-6] Execution Time: 162ms
[Thread-0] Execution Time: 161ms
[Thread-1] Execution Time: 163ms
[Thread-15] Execution Time: 163ms
[Thread-10] Execution Time: 180ms
[thread_no]:16, [Time]: 181 ms, [Matrix sum]: 1001895301

+ problem2 git:(main) x java MatmultD.java 32 < mat1000.txt
[Thread-2] Execution Time: 75ms
[Thread-5] Execution Time: 76ms
[Thread-12] Execution Time: 67ms
[Thread-3] Execution Time: 69ms
[Thread-7] Execution Time: 77ms
[Thread-11] Execution Time: 68ms
[Thread-9] Execution Time: 72ms
[Thread-10] Execution Time: 58ms
[Thread-8] Execution Time: 84ms
[Thread-4] Execution Time: 119ms
[Thread-13] Execution Time: 98ms
[Thread-6] Execution Time: 123ms
[Thread-17] Execution Time: 73ms
[Thread-23] Execution Time: 57ms
[Thread-21] Execution Time: 75ms
[Thread-19] Execution Time: 85ms
[Thread-28] Execution Time: 56ms
[Thread-24] Execution Time: 68ms
[Thread-27] Execution Time: 62ms
[Thread-25] Execution Time: 74ms
[Thread-14] Execution Time: 121ms
[Thread-1] Execution Time: 164ms
[Thread-22] Execution Time: 86ms
[Thread-31] Execution Time: 56ms
[Thread-30] Execution Time: 59ms
[Thread-29] Execution Time: 72ms
[Thread-20] Execution Time: 101ms
[Thread-26] Execution Time: 79ms
[Thread-16] Execution Time: 107ms
[Thread-18] Execution Time: 105ms
[Thread-15] Execution Time: 119ms
[Thread-0] Execution Time: 177ms
[thread_no]:32, [Time]: 178 ms, [Matrix sum]: 1001895301
```

VI/ Analysis of the results

As in problem 1, the performance grows as the number of threads increases until it reaches a ceiling - 10 threads here. The shape of the curves look similar to a logarithm function. This ceiling is probably due to the fact that the machine has a limited number of physical cores (11), in addition to overhead and memory bandwidth limitations.

Additionally, we can observe - as in problem 1 - that performance gain slows down from 6 threads. It confirms that the 6 first threads are run by performance cores, and the 5 other ones are run by efficiency cores, which are slower than p-cores.