Homework 1 [TI8419 - Multilinear Algebra]

Lucas Abdalah

Professors: André Lima e Henrique Goulart

Table of Contents

Problem 1

For randomly generated $\mathbf{A}$ and $\mathbf{B} \in \mathbb{C}^{N \times N}$, create an algorithm to compute the Hadamard Product $\mathbf{A} \odot \mathbf{B}$. Then, compare the run time of your algorithm with the operator A.*B of the software Octave/Matlab ®. Plot the run time curve as a function of the number of rows/columns $N \in \{2, 4, 8, 16, 32, 64, 128\}$.

Results

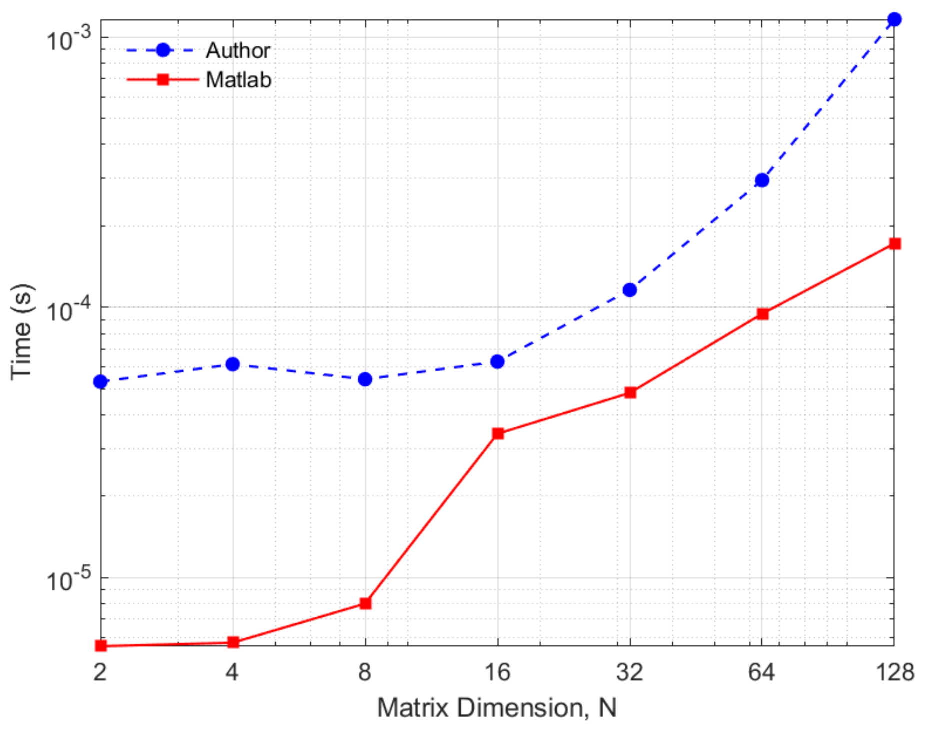
Simulation setup

- 500 Monte Carlo Runs;
- Each Monte Carlo iteration uses a new matrix initialization from a Normal distribution $\mathcal{N}(0,\, 1)$;
- Compute the mean for each value, for $N = \{2, 4, 6, 8, 16, 32, 64, 128\}$.

Discussion

We can see that for all values of N, Matlab's method outperforms the Author's. For small values of N, the gap between them, 6×10^{-5}s vs 6×10^{-6}s, approximately ten times faster. However as the N increases, that performance gap becomes more subtle.

Problem 1 script



Problem 2

For randomly generated $\mathbf{A}$ and $\mathbf{B} \in \mathbb{C}^{N \times N}$, create an algorithm to compute the Kronecker Product $\mathbf{A} \otimes \mathbf{B}$. Then, compare the run time of your algorithm with the operator kron(A, B) of the software Octave/Matlab ®. Plot the run time curve as a function of the number of rows/columns $N \in \{2, 4, 8, 16, 32, 64, 128\}$.

Results

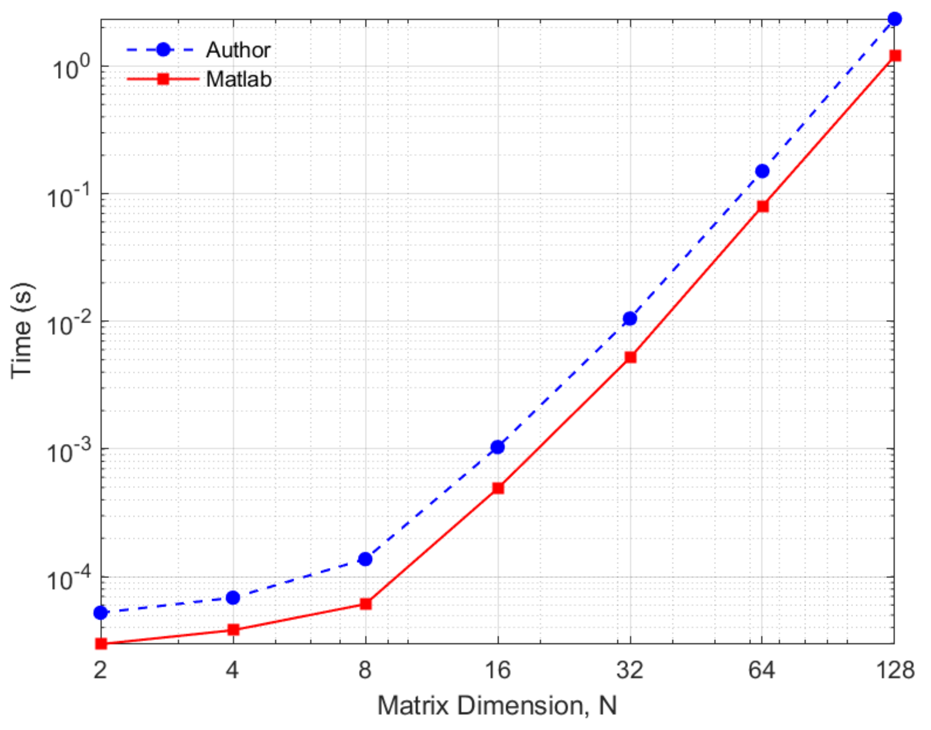
Simulation setup

- 500 Monte Carlo Runs;
- Each Monte Carlo iteration uses a new matrix initialization from a Normal distribution $\mathcal{N}(0,\, 1)$;
- Compute the mean for each value, for $N = \{2, 4, 6, 8, 16, 32, 64, 128\}$.

Discussion

We can see that for all values of N, Matlab's method outperforms the author's. There's a narrow performance gap between them, up to three times faster. The difference varies very little regardless the value of N increase.

Problem 2 script



Problem 3

For randomly generated $\mathbf{A}$ and $\mathbf{B} \in \mathbb{C}^{N \times N}$, create an algorithm to compute the Khatri-Rao Product $\mathbf{A} \diamond \mathbf{B}$ according with the following prototype function:

$$R = kr(A, B).$$

Results

Simulation setup

- 500 Monte Carlo Runs;
- Each Monte Carlo iteration uses a new matrix initialization from a Normal distribution $\mathcal{N}(0,\, 1)$;
- Compute the mean for each value, for $N = \{2, 4, 6, 8, 16, 32, 64, 128\}$.

Discussion

The method developed by the author present similar behavior to Kronnecker product and a predictable trend for all values of N.

Problem 3 script