

Análise do Desempenho da Multiplicação de Matrizes por Algoritmo Tradicional contra *Strassen*

Faculdade de Computação, Universidade Federal de Uberlândia (UFU), Uberlândia, Brasil

Dahlan Pereira Gardim (11811BCC030)
Lucas Guimarães Mendes (11811BCC045)
Mateus Oliveira Lemos (11811BCC007)

I. Introdução

A multiplicação de matrizes é uma operação da Álgebra Linear, fundamental para soluções de vários problemas em muitas áreas do conhecimento. Por conta disso, a análise de técnicas mais eficientes é de grande valia para o conhecimento científico.

O principal objetivo deste trabalho é, portanto, apresentar uma análise de desempenho para dois algoritmos de multiplicação de matrizes, isto é, os algoritmos tradicional e o de *Strassen*.

Destaca-se que, se baseando na teoria de estudo da complexidade de tais algoritmos, mais especificamente pelo Teorema Mestre, é possível fazer uma predição para os resultados almejados na execução de ambos. Nesse sentido, sabe-se que a complexidade do Algoritmo Tradicional é $O(n^3)$, enquanto para *Strassen* tem-se $O(n^{2.807})$.

II. Metodologia Aplicada

Para o seguinte experimento foi utilizado a linguagem de programação C. Para fins de simplificação da implementação desenvolvida, as matrizes devem ser necessariamente quadradas ($n \times n$), onde n é uma potência de dois. Isso evita a necessidade de se complementar matrizes não múltiplas de dois com zeros, para realizar a divisão dos grupos necessários para aplicação do método de multiplicação *Strassen*.

Dessa forma, foram realizados experimentos com matrizes 512×512 , 1024×1024 , 2048×2048 , 4096×4096 , sendo que para cada dimensão foram realizados 20 testes (10 para cada um dos dois tipos de algoritmo propostos no trabalho), totalizando 80 testes. Além disso, com fim de evitar discrepância nos dados, todos os testes foram feitos em um único computador com as seguintes especificações:

- Processador: Intel Core i5-7200
- CPU: 2.50 GHz
- Memória RAM: 8 GB
- Sistema Operacional: Windows 10 Home (x64)

III. Resultados Obtidos

Os resultados obtidos nos testes estão exibidos na tabela abaixo.

Análise de Algoritmos de Multiplicação de Matrizes								
N	512		1024		2048		4096	
Algoritmo	Original	Strassen	Original	Strassen	Original	Strassen	Original	Strassen
Tempo em ms	955	316	24740	2763	227673	22203	1719589	176830
	948	323	23572	2795	228606	22391	1720730	182022
	946	322	23989	2717	230810	21974	1713646	177927
	962	314	23804	2736	226319	22151	1712606	178458
	953	344	24135	2749	224010	22057	1747515	183482
	968	328	24994	2765	225728	21948	1756000	180067
	953	312	23870	2749	221620	21964	1752054	177755
	969	312	24619	2796	223323	21901	1714240	196242
	1109	328	24510	2828	226010	21886	1786707	183875
	937	328	24634	2750	225385	21979	1758056	191485
Média	970	322,7	24286,7	2764,8	225948,4	22045,4	1738114,3	182814,3
Desvio Padrão	49,82	9,93	473,25	32,86	2657,58	158,97	25443,32	6407,48

Tabela 1 - Resultados parciais e totais do tempo de execução dos dois algoritmos

No gráfico abaixo, podemos ver a comparação entre as médias de tempo que os algoritmos levam para rodar. O tempo está representado em segundos.

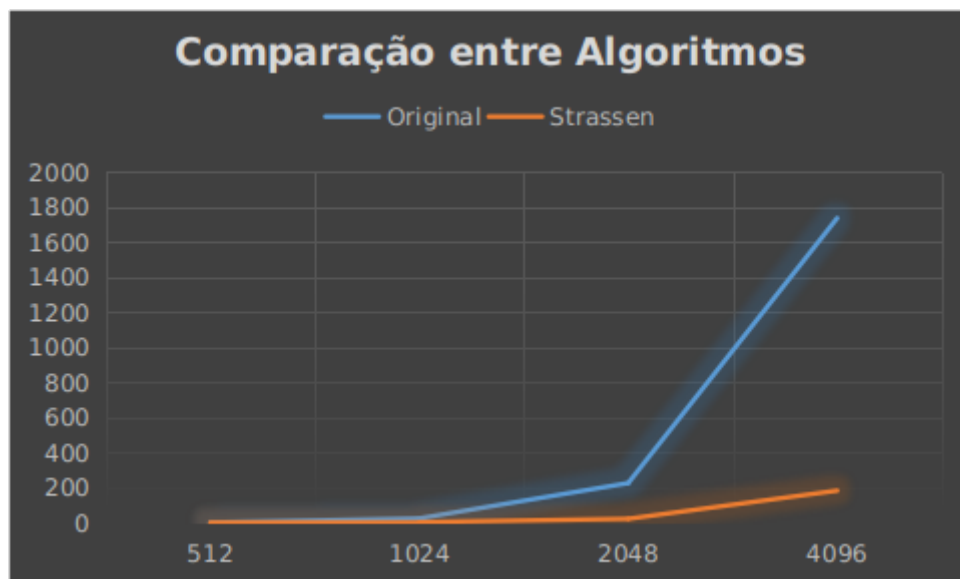


Gráfico 1 - Comparação dos tempos de execução dos dois algoritmos

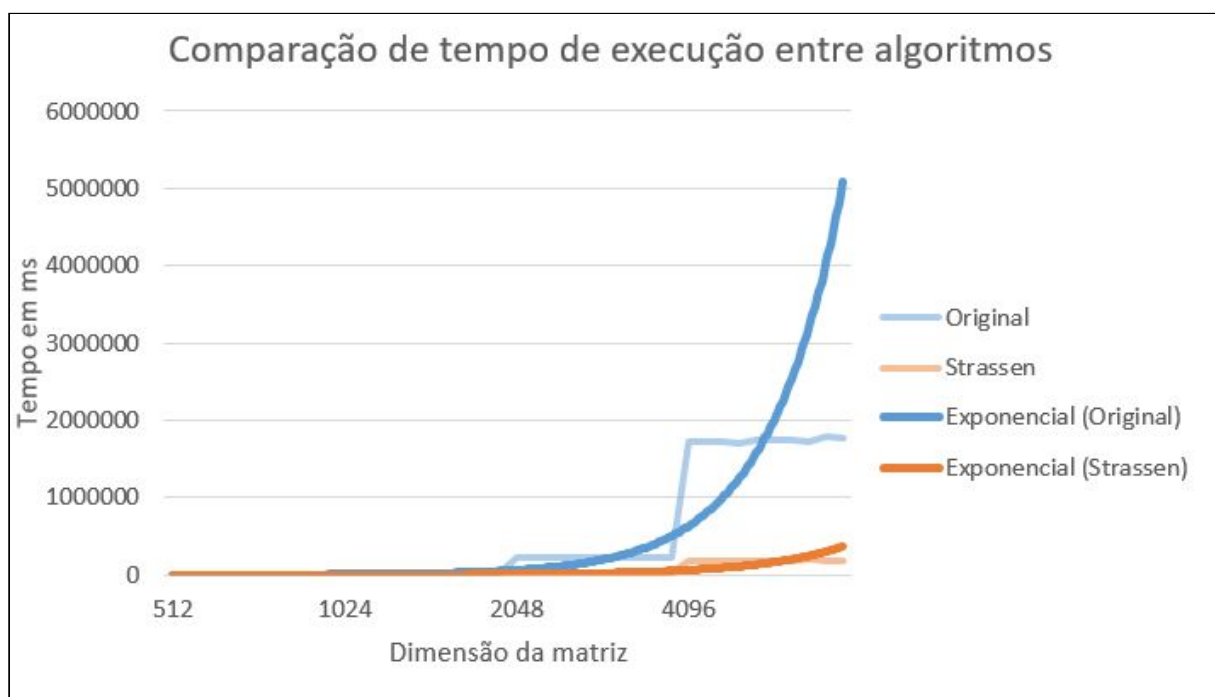
IV. Considerações sobre os Resultados

Analisando os dados da tabela, podemos obter algumas informações relevantes, como por exemplo, quantas vezes mais rápido foi o algoritmo de *Strassen* em comparação com o original. Podemos verificar de forma mais clara esses dados na tabela abaixo:

Dimensão	Matrizes 512 x 512		Matrizes 1024 x 1024		Matrizes 2048 x 2048		Matrizes 4096 x 4096	
Algoritmo	Original	Strassen	Original	Strassen	Original	Strassen	Original	Strassen
Média (ms)	970	322,7	24286,7	2764,8	225948,4	22045,4	1738114,3	182814,3
Razão (Orig. / Strassen)	3,0		8,8		10,2		9,5	

Notamos, então, que na nossa implementação o algoritmo de *Strassen* conseguiu, em seu melhor desempenho, ser mais de 10 vezes mais rápido do que o algoritmo tradicional.

Outra forma que facilita a compreensão do crescimento da função tempo dos dois algoritmos é fazendo a tendência exponencial, demonstrada no gráfico abaixo.



Link para o repositório do código: [Multiplicação de Matrizes \(Strassen\)](#)