



Fundação Getulio Vargas
Escola de Matemática Aplicada
Curso de Ciência de Dados e Inteligência
Artificial

Fabício Dalvi Venturim
Luís Felipe de Almeida Marques
Otávio Augusto Matos Alves
Vinicius Antunes de Sousa

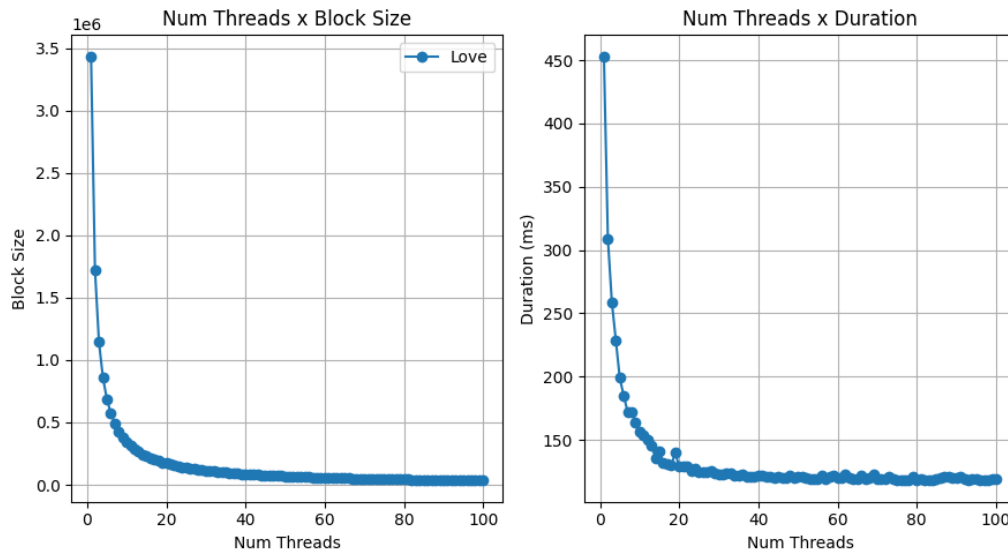
Exercício 1

Computação Escalável
Professor: Thiago Pinheiro de Araújo

Rio de Janeiro
2024

Resultados

O uso de paralelismo representou uma melhora perceptível na performance do programa, chegando a reduzir o tempo de execução a em torno de 20% do tempo sem paralelismo. Pudemos notar também uma estagnação a partir de 12 THREADS, já que é o limite de THREADS em nossa máquina. Apesar disso o tempo continuou a diminuir, mesmo que a um ritmo mais lento, ao pedirmos mais de 12 THREADS, provavelmente devido a otimizações de computação concorrente embutidas no sistema de THREADS.



Observamos 1.500 ocorrências da palavra *hate* e 27.600 de *love*. É importante lembrar que no programa criado, não são contadas instâncias dessas palavras dentro de outras, ou seja, *lovely* ou *hateful* não influenciam em nossas contagens. Nosso *input* foi constituído de 3.433.801 palavras dos textos de Shakespeare.

Podemos observar que, ao aumentarmos linearmente o número de THREADS, o tempo de execução decresce numa taxa sublinear, o que condiz com a ideia de ganhos marginais decrescentes. Simplificadamente, se com 1 THREAD demoramos um tempo x , ao usar n THREADS deveríamos ter um tempo de processamento de x/n , pelo menos até o número de THREADS igualar 12, e ignorando tempos de *overhead* e similares.