Report

Parallel Systems: Sequential Algorithms

Auer Thomas

January 7, 2023

Contents

1. Introduction

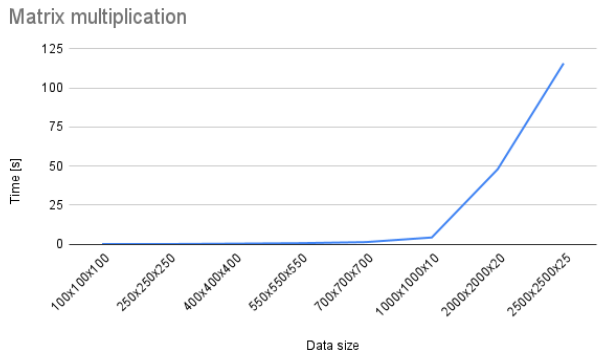
The following report shows performance analytics of several popular algorithms: Matrix multiplication, Gaussian elimination, Dijkstra, and Sieve of Eratosthenes, and several sorting algorithms: Bubble sort, Bucket sort, Counting sort, Insertion sort, Quick sort, and Selection sort. The main goal was to review the time complexity of the algorithms when changing the dimensionality of the input data.

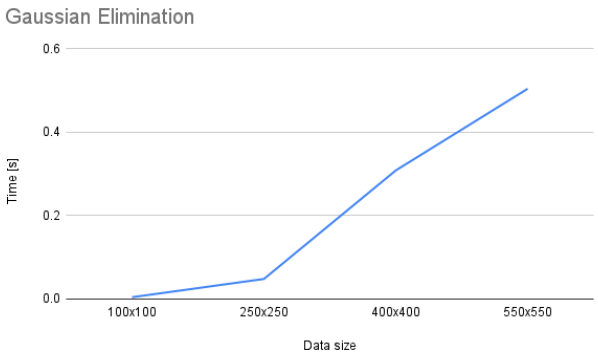
2. Methods and Resources

The algorithms are implemented using the programming language C, compiled using GCC, and executed on a desktop using an Intel® Core™ i7-8700 CPU (3.20GHz x 12). The implementations are available on Github.com.

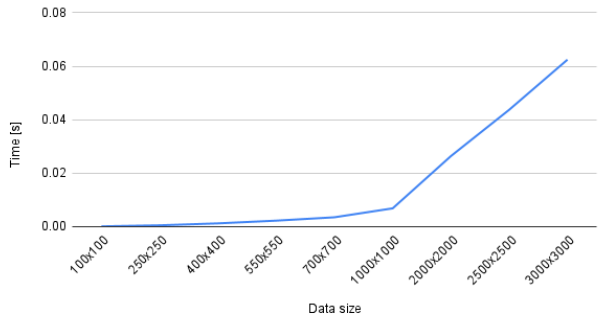
3. Results

The results were aggregated over fields of various dimensions but with an maximum element value of 100.

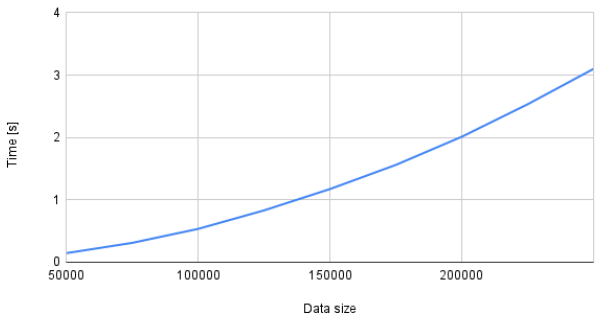




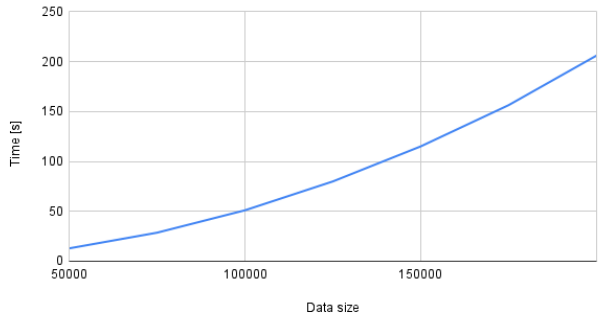
Dijkstra



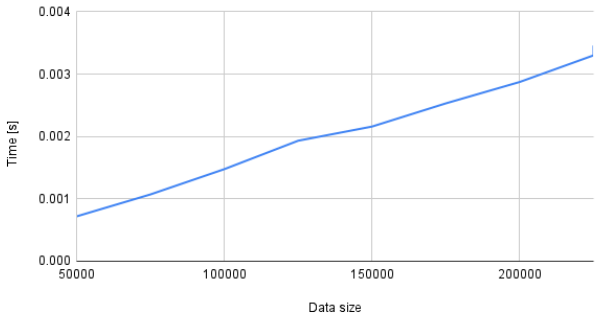
Sieve of Eratosthenes



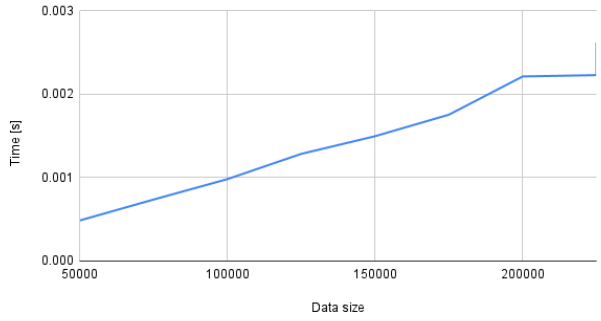
Bubble Sort



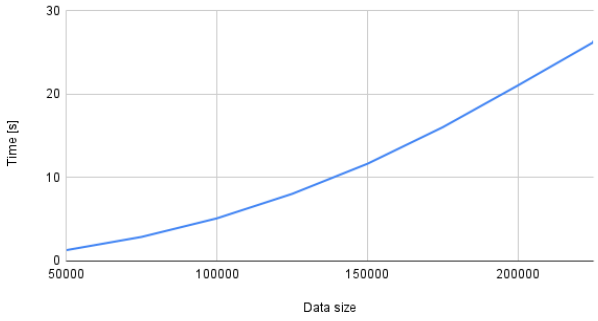
Bucket Sort

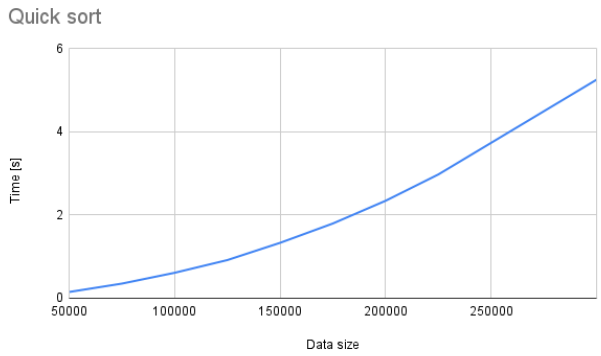


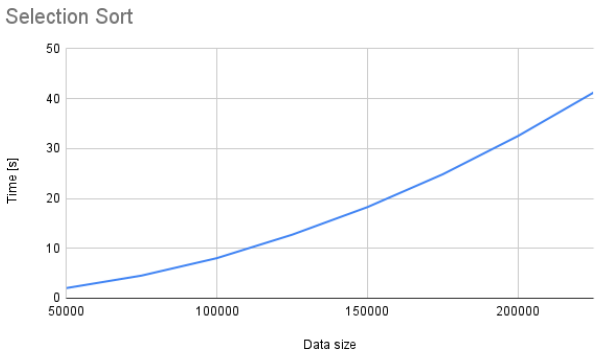
Counting Sort



Insertion sort







4. Discussion

Two sorting algorithms, namely Counting Sort and Bucket Sort, did too well during the performance testing. Despite further inspections, no significant indicator was found why these 2 algorithms would perform that well. It was assumed that this may be due to the maximum element value being 100.
All the other algorithms performed roughly as expected given complexity theory for squared and cubic functions.

5. Conclusion

For future work, it will be necessary to review some of the implemented algorithms for better readability and execution safety as stack smashing and other memory problems occassionally occured on higher dimensions.