

# **Impact of GPU Acceleration on Needleman-Wunsch and Smith-Waterman Algorithms**

Sai Prasanthi Kurra

Department of Information Sciences and Technology, George Mason University

AIT 512: Algorithms and Data Structures Essentials

Dr. Mihai Boicu

May 1, 2025

## **Abstract**

This study evaluates the effectiveness of the Needleman-Wunsch and Smith-Waterman algorithms on both CPU and GPU systems. By analyzing nucleotide sequences of various lengths, it assesses execution time and memory consumption across these platforms. The results demonstrate that GPU acceleration significantly enhances performance, particularly for longer sequences, but introduces higher memory usage. This trade-off underlines the need for balancing computational speed with resource efficiency, offering insights for optimizing sequence alignment in bioinformatics.

## **Introduction**

Sequence alignment is crucial in bioinformatics as it allows for the comparison of nucleotide and protein sequences. Among the prominent algorithms for pairwise sequence alignment are the Needleman-Wunsch and Smith-Waterman algorithms (Hung et al., 2015). However, they often require considerable computational power, especially with larger datasets. Traditional CPU methods can be less efficient, leading to longer processing times and higher memory consumption (Hasan et al., 2011). In contrast, GPU computing presents new opportunities for enhancement, allowing for faster execution due to its ability to handle multiple operations at once (Ahmed et al., 2019). This study investigates the impact of GPU acceleration on the performance of Needleman-Wunsch and Smith-Waterman algorithms, focusing on execution time and memory usage. We hypothesize that GPU implementations can outperform CPUs for long sequences, but may introduce overhead for shorter ones. This analysis uses nucleotide sequences to evaluate these trade-offs (Zwaka, 2024).

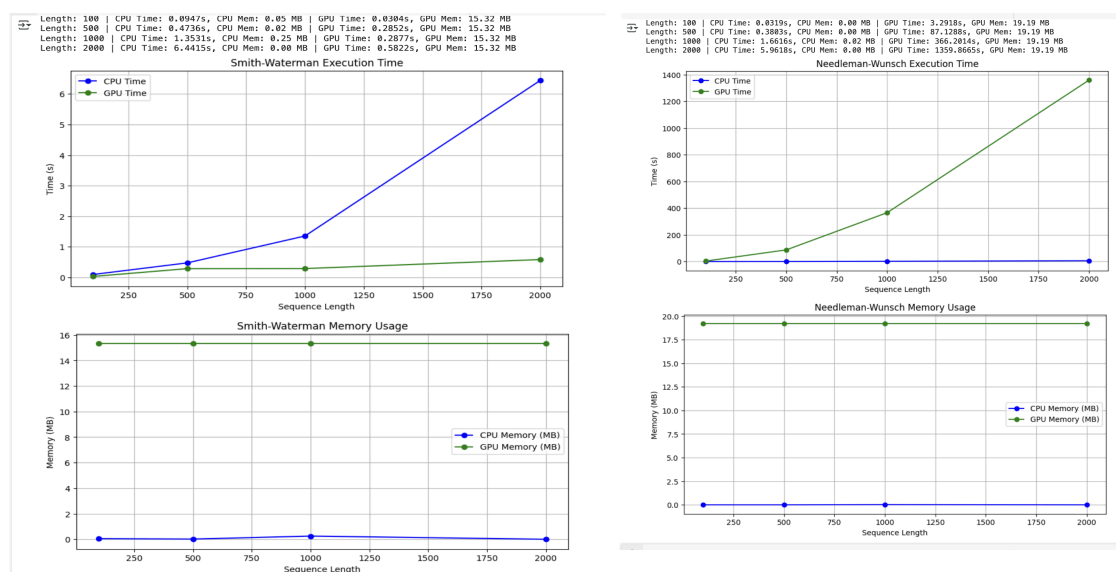
## **Experimental Design**

The experimental design included acquiring two DNA sequences, specifically BRCA1 (NM\_007294 - Breast Cancer Gene) and TP53 (NM\_000546 - Tumor Suppressor Gene), from the NCBI GenBank database through Biopython (FASTA format). These sequences were shortened to lengths of 100, 500, 1000, 2000, and 5000 bases to evaluate scalability. The implementation took place on a personal computer equipped with a standard CPU and a CUDA-enabled GPU, utilizing Google Colab's GPU runtime. The process was developed using Python, incorporating NumPy for CPU operations and CuPy for GPU processing. Execution

times were measured and collected with Python's time module, and memory usage was profiled using memory\_profiler (Hung et al., 2015). Specific comparisons include CPU vs. GPU execution time and memory usage. Performance graphs contrasting CPU and GPU processing times for both algorithms were created using Matplotlib.

## Results and Analysis

The analysis showed that the CPU execution time for both the Needleman-Wunsch (NW) and Smith-Waterman (SW) algorithms increased in line with the length of the sequences. In comparison, utilizing GPU acceleration significantly shortened the execution time for both algorithms, with GPU execution times remaining fairly stable even as sequence lengths grew. When it comes to memory usage, CPU memory requirements rose predictably alongside the input sequence sizes, as indicated by the scaling of the scoring matrix. Conversely, GPU memory usage reached higher peak levels than the CPU, especially with longer sequences. Nevertheless, through CuPy's memory pool management system, efficient memory allocation was maintained during execution, helping to alleviate potential inefficiencies (Perez-Serrano et al., 2018).



**Note.** The left graph shows CPU vs. GPU execution time (in seconds) and memory usage (in megabytes) for Smith-Waterman, while the right graph shows the same for Needleman-Wunsch.

## Conclusion and Limitations

This research highlighted that GPU acceleration significantly enhances the performance of sequence alignment tasks using the Needleman-Wunsch and Smith-Waterman algorithms. For instance, the GPU-optimized version of the Needleman-Wunsch algorithm showed remarkable efficiency in aligning lengthy sequences, making it especially beneficial for situations that require global sequence alignment (Ahmed et al., 2019). However, this speed gain is accompanied by an increase in memory usage, which might restrict its effectiveness with very large datasets. GPUs excel over CPUs for specific tasks due to their thousands of smaller cores designed for parallel processing. Additionally, GPUs offer greater memory bandwidth, which enables quicker data transfers. On the other hand, CPUs are tailored for general-purpose (Perez-Serrano et al., 2018). The study has some limitations, including the use of fixed scoring parameters and specific hardware configurations, which may not apply to different systems. Future research should focus on optimizing GPU memory usage to reduce peak loads, exploring advanced hardware options (like GPUs with larger memory capacities), and broadening the analysis to include additional sequence alignment algorithms (Hung et al., 2015).

### References

1. Ahmed, N., Levy, J., Ren, S., Mushtaq, H., Bertels, K., & Al-Ars, Z. (2019). GASAL2: a GPU accelerated sequence alignment library for high-throughput NGS data. *BMC Bioinformatics*, 20(1), 520–520. <https://doi.org/10.1186/s12859-019-3086-9>
2. Hasan, L., Kentie, M., & Al-Ars, Z. (2011). DOPA: GPU-based protein alignment using database and memory access optimizations. *BMC Research Notes*, 4(1), 261–261. <https://doi.org/10.1186/1756-0500-4-261>
3. Hung, C.-L., Lin, C.-Y., Hong, Y., Liu, Y., & Jiang, H. (2015). Accelerating Smith-Waterman Alignment for Protein Database Search Using Frequency Distance Filtration Scheme Based on CPU-GPU Collaborative System. *International Journal of Genomics*, 2015(2015), 1–12. <https://doi.org/10.1155/2015/761063>
4. Perez-Serrano, J., Sandes, E., Magalhaes Alves de Melo, A. C., & Ujaldon, M. (2018). DNA sequences alignment in multi-GPUs: acceleration and energy payoff. *BMC Bioinformatics*, 19(Suppl 14), 421–421. <https://doi.org/10.1186/s12859-018-2389-6>
5. Zwaka, L. (2024). *Parallel DNA Sequence Alignment on High-Performance Systems with CUDA and MPI*. <https://doi.org/10.48550/arxiv.2412.21103>