Enhancing the efficiency of animal-alternative in-silico drug cardiotoxicity prediction through CUDA-based parallel processing

Iga Narendra Pramawijaya1*, Aroli Marcellinus1 , Ali Ikhsanul Qauli1 , Ki Moo Lim1
[1]Kumoh National Institute of Technology Industry-academic Cooperation Foundation, Korea
*iga@kumoh.ac.kr

Abstract

Cardiovascular diseases (CVDs) remain the leading cause of mortality worldwide. Current drug discovery pipelines face a significant challenge: the heavy reliance on animal testing. This "species translation" issue can lead to misleading results. In response to these limitations, in-silico (computer-based) methods for drug cardiotoxicity prediction have emerged as promising alternatives. In this research, we propose a novel approach to address the computational bottleneck associated with the calculation of complex in silico models under multi-sample scenario, by leveraging CUDA (Compute Unified Device Architecture)-based parallel processing on GPUs. This approach has the potential to significantly accelerate the execution of the model up to 10 times faster when faced with a population sample for drug-testing, making them more efficient and practical for real-world drug discovery applications.

Method

A. Simulation Protocol

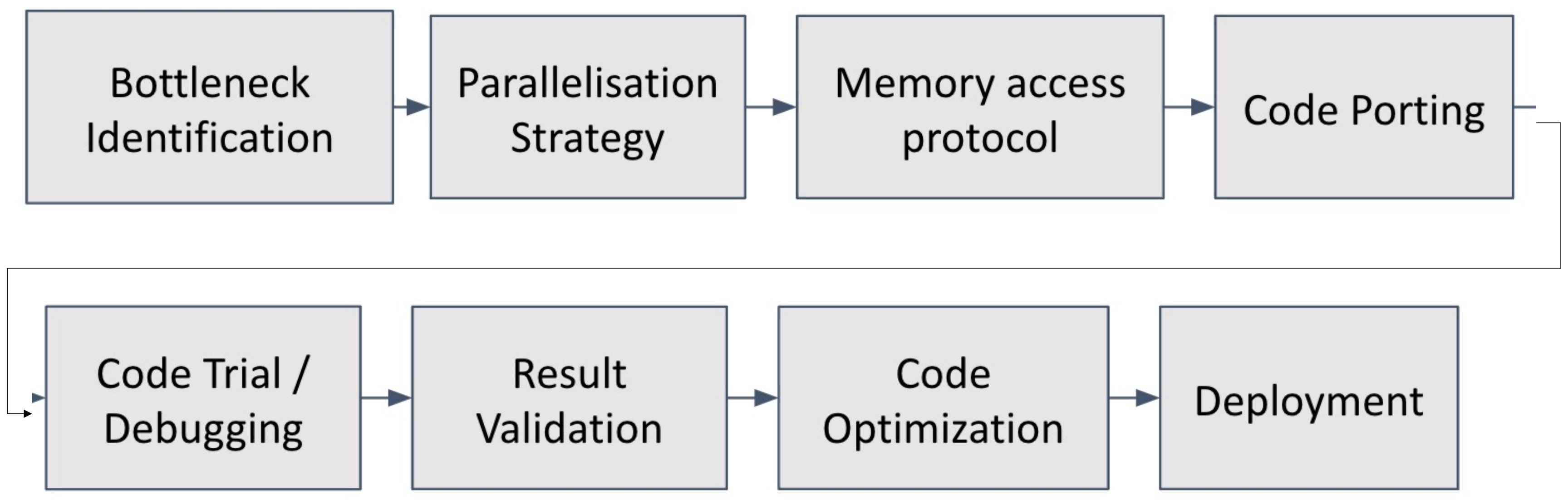
This research builds CUDA-based parallelization model upon existing human cardiac electrophysiology model proposed by O'Hara et. al. [2] to enhance its multi-sample calculation performance. We will also leverages the results from O'Hara et. al. for validation. Our focus is on drug effects at the cellular level, requiring 7 IC50 and 7 Hill coefficient [4] per sample. Key outputs include drug toxicity biomarkers and time-series data from each simulated channel, both in CSV format, valuable for future drug discovery efforts.

B. Solving Ordinary Differential Equation (ODE)

The model built upon algebraic calculations and dynamic functions served in the form of ODEs. We were able to trace the computational procedures and create semi-analytical method to be implemented in the CUDA-based model [2]. This transformation is required to simplify the computational process, and let the parallelization process focuses on processing multiple samples instead of multiple equations.

ODE solver in CUDA-based model is quite similar to Euler style of solving, where the next value determined by adding the previous value with rate of change multiplied by time difference. We also provided function that dynamically update the time difference in each pace to minimize error from this method.

C. Developing GPU-based Parallel Process Using GPU Parallelisation Framework


Bottleneck Identification → Parallelisation Strategy → Memory access protocol → Code Porting → Code Trial / Debugging → Result Validation → Code Optimization → Deployment

Result

A. Time Performance Comparison

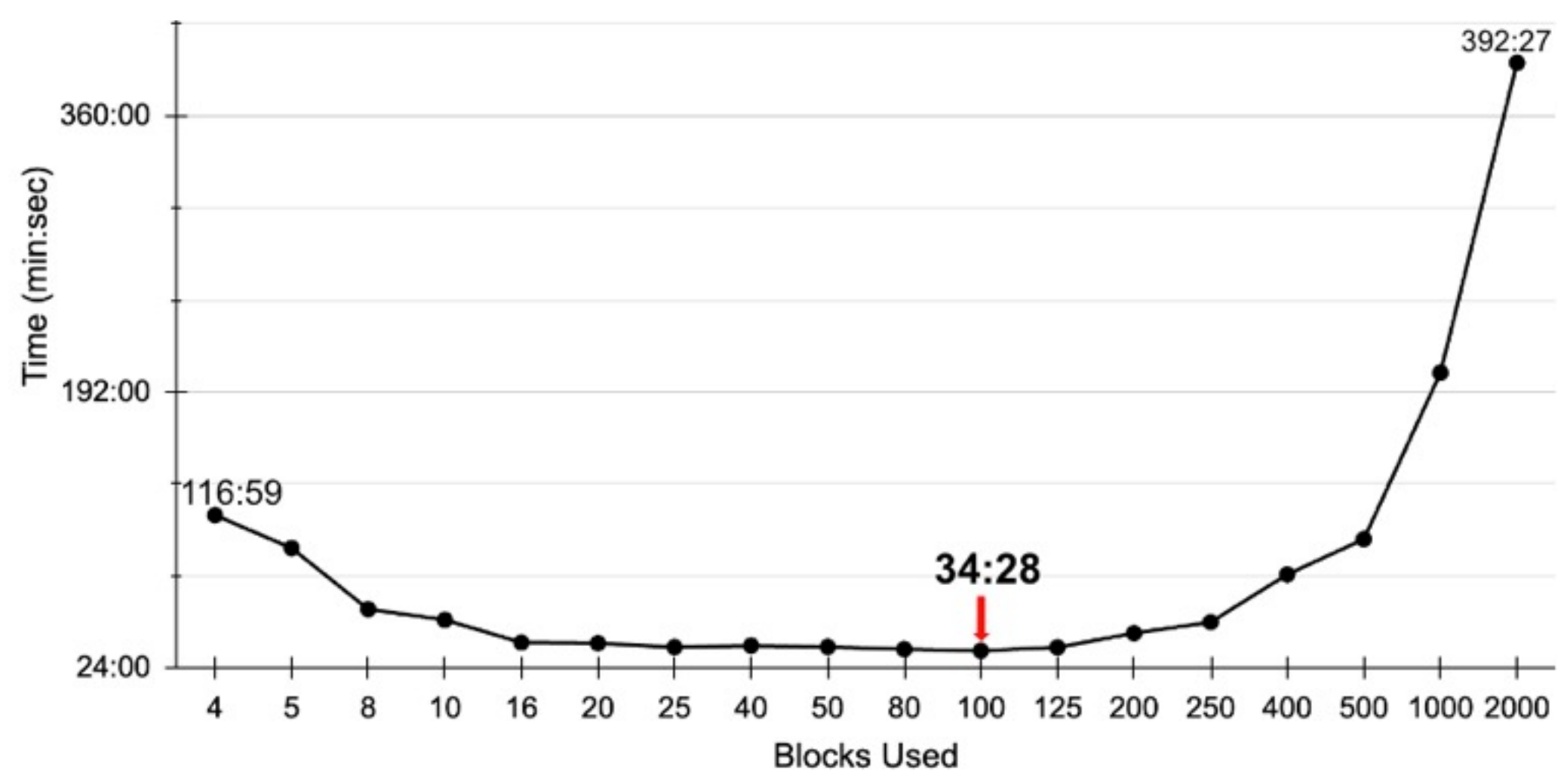


Figure 3. 2 Comparison of Time and Block Used in GPU-based Simulation

Initially, we need to pick the most optimal GPU core usage per GPU computing block. We used an Nvidia RTX 3080 TI with 8GB of GPU memory for this optimization trials. Figure 3.2 shows all of block configuration we tried, highlighting extremities and most optimal configuration. In total, we tested the model for 2000 samples. As we are creating the parallelization based on each sample, each sample has their own “computing core”, making the configuration as factors of 2000 (2 blocks = 1000 core/block; 10 blocks = 200 core/blocks, etc.). Significant trials bring us to a conclusion where each computing block optimally consist of 20 cores.

B. Time-Series Result Validation

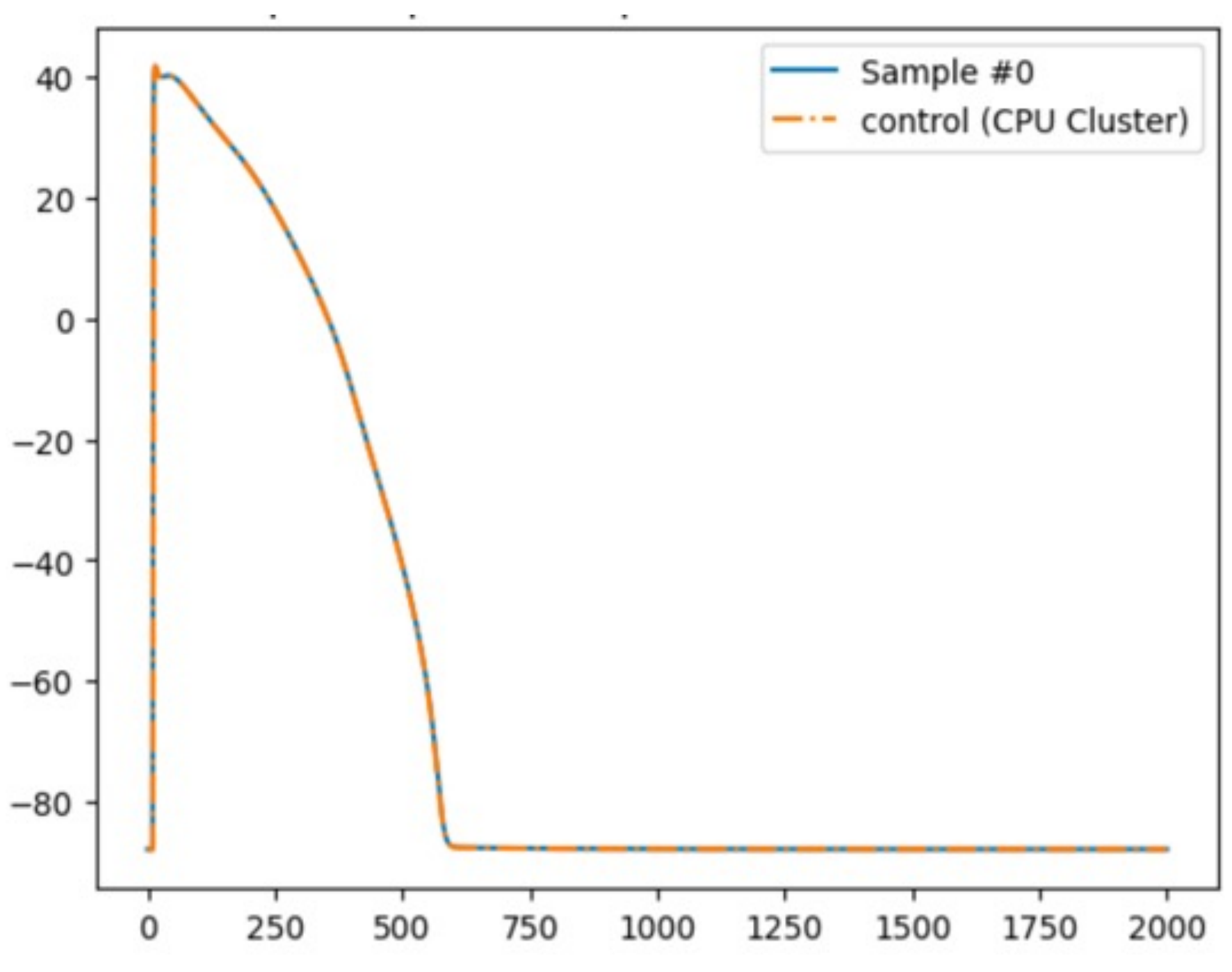


Figure 3. 5 Action Potential Shape of Both CPU and GPU Simulation Result

Result inaccuracy can cause invalid drug cardiotoxicity prediction. The simplest way to validate the result is by comparing both of action potential shape from CPU and GPU simulation. Figure 3.5 shows action potential curve from both CPU and GPU simulation. As shown, little to no difference from both of the result, indicating a valid result from the GPU-based simulation. Promising more efficient in-silico drug cardiotoxicity prediction.

C. Computational Unit Comparison

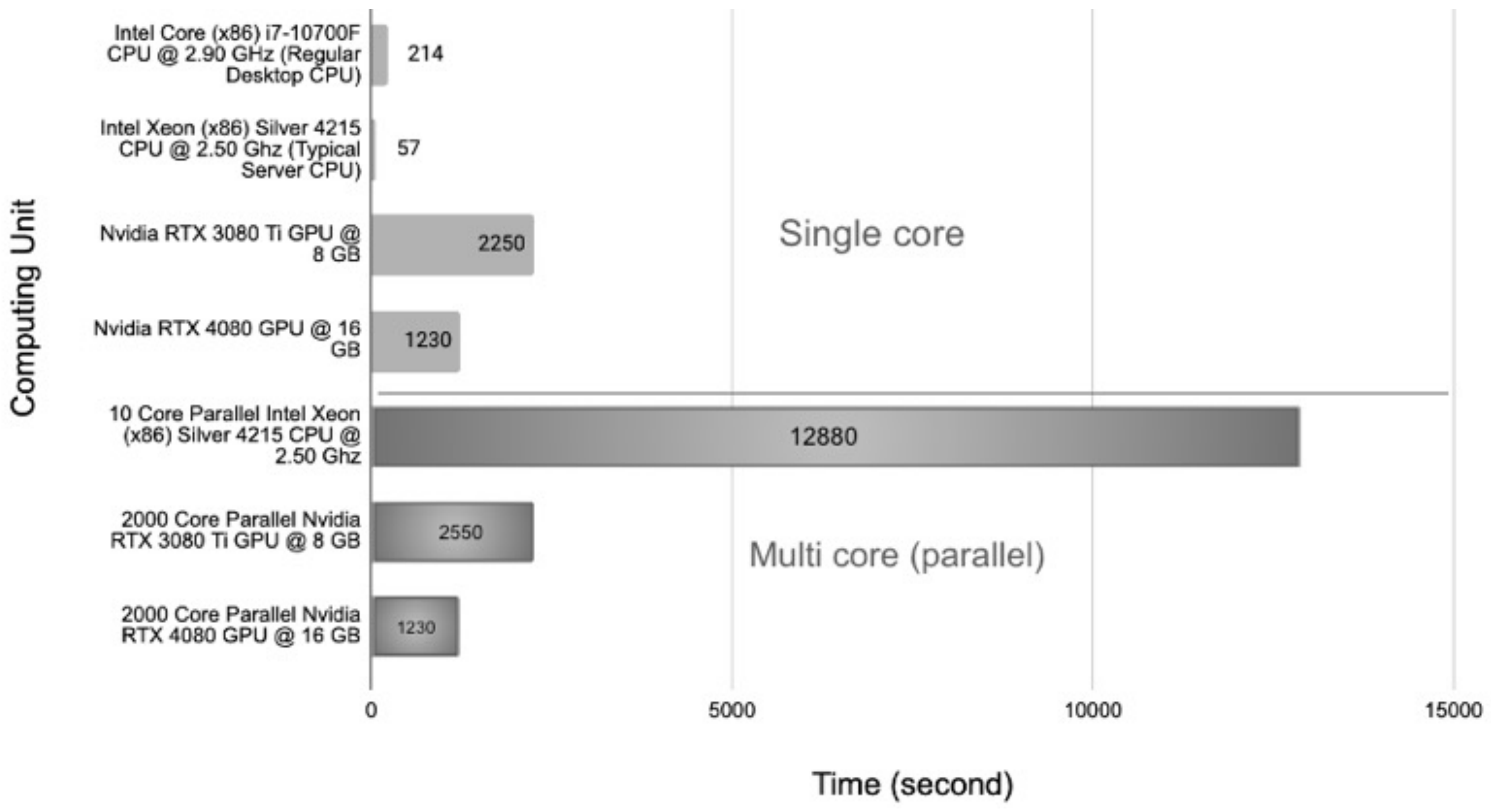


Figure 3. 3 Computational Speed Of Different Resources for Single and Multi Core

CPU Calculation time is linear with the sample size and pacing. This linearity makes CPU computation time grow as sample grows. In GPU computing unit, this linearity does not affect the computing speed due to GPU parallelization. In other words, the time it takes to compute 1 sample will be similar to any number of samples. We compared our CUDA-based GPU approach with a parallel computing CPU using OpenMPI [2]. For 2000 samples simulation, Lower part of Figure 3.3 shows GPU processing achieved up to a 10x speedup, demonstrating significant efficiency gains. All results shown simulated under Bepridil drug effect, with concentration of 99.0 mMol. Experiment shows little to no performance difference between under drug and no-drug simulation.

Acknowledgements

This research was partially supported by the Ministry of Food and Drug Safety (22213MFDS3922), the NRF (National Research Foundation of Korea) under the Basic Science Research Program (2022R1A2C2006326), and the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2022-2020-0-01612) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

Reference

[1] Jason Sanders and Edward Kandrot. 2010. CUDA by Example: An Introduction to General-Purpose GPU Programming (1st. ed.). Addison-Wesley Professional.

[2] O'Hara T, Virág L, Varró A, and Rudy Y (2011) “Simulation of the Undiseased Human Cardiac Ventricular Action Potential: Model Formulation and Experimental Validation”. PLoS Comput Biol 7(5): e1002061. https://doi.org/10.1371/journal.pcbi.1002061

[3] R. L. Graham, G. M. Shipman, B. W. Barrett, R. H. Castain, G. Bosilca and A. Lumsdaine, "Open MPI: A High-Performance, Heterogeneous MPI," 2006 IEEE International Conference on Cluster Computing, Barcelona, Spain, 2006, pp. 1-9, doi: 10.1109/CLUSTR.2006.311904.

[4] Mirams G. R., Cui Y., Sher A., Fink M., Cooper J., Heath B. M., et al. (2011). Simulation of multiple ion channel block provides improved early prediction of compounds' clinical torsadogenic risk. Cardiovasc. Res. 91 (1), 53–61. doi:10.1093/cvr/cvr044