DOI: 10.14529/jsfi230406

High-Performance Computing of Microtubule Protofilament Dynamics by Means of All-Atom Molecular Modeling

Vladimir A. Fedorov[1,2], *Ekaterina G. Kholina*[1], *Nikita B. Gudimchuk*[1], *Ilya B. Kovalenko*[1]

Molecular dynamics (MD) simulation is a useful tool for understanding biological systems at the level of individual molecules and atoms. However, studying such massive biological systems as microtubules and even their constituent components (tubulin protofilaments) takes an enormous amount of processing power. In this paper, using MD calculations of individual microtubule protofilaments, we demonstrate how computational architecture and calculation options affect computing performance. When using the "GPU-resident" option in the GROMACS MD package, you may gain a fantastic computation acceleration by using the newest high-end graphics processing unit (GPU), even in conjunction with a rather outdated central processing unit (CPU). For instance, MD of the biomolecular system containing a tubulin protofilament in an explicitly specified solvent consisting of more than 300 thousand atoms can be investigated with performance of 171 ns/day at time step 2 fs when using a single-node computer with the latest CPU and GPU generation architecture (Intel Core i9-13900K and Nvidia RTX4090 respectively). Nevertheless, high performance computing platforms (e.g., the volta2 partition of "Lomonosov-2" supercomputer) can be very suitable for simulation experiments with a large number of independent calculations, such as the umbrella sampling technique. Obtained results allow one to choose the best price-performance solution to study molecular dynamics of biological systems.

Keywords: molecular dynamics, tubulin, microtubule, CPU, GPU, computing performance.

Introduction

Microtubules are polymers of the tubulin protein that are involved in many critical functions in living cells. One of these functions is the separation of genetic material during cell division (the chromosomes segregation). Search and attachment to chromosomes becomes possible due to an amazing property of microtubules, namely, dynamic instability, their ability to spontaneously polymerize and depolymerize [4]. Despite decades of research, the detailed molecular mechanisms underlying this process remain unknown. The experiments and classical theoretical approaches provide particular facts without explaining the behavior of the entire microtubule at the molecular level.

A powerful tool for studying of biological systems functioning at the level of individual molecules and atoms is molecular dynamics (MD) simulation. However, the study of such large systems as a microtubule and even its individual components (tubulin protofilaments) requires extremely high computational resources. The question arises about choosing the optimal computational architecture for studying such systems using the MD method. Computational capabilities are improved every year with new generations of CPUs and GPUs, so it is important to know what computational capabilities each computer configuration provides. In addition, software packages for MD calculations are being improved by adding options increasing the performance of MD calculations. In this article, using MD calculations of individual microtubule protofilaments, we demonstrate how the choice of computational architecture and calculation options affects their performance.

[1]Lomonosov Moscow State University, Moscow, Russia
[2]Center for Theoretical Problems of Physico-Chemical Pharmacology, Russian Academy of Sciences, Moscow, Russia

The article is organized as follows. Section 1 is devoted to the computing method of all atom molecular dynamics. In Section 2, we described the results comparing the performance of various computing systems and various parameters for molecular dynamics calculations. Conclusion summarizes the study and allows one to make the best choice of MD parameters and computational architecture.

1. Methods

All tests were conducted using the all-atom explicit solvent MD. The calculations were executed using the GROMACS 2022.4 software package [2] which facilitates parallel computing on hybrid architectures and incorporates the CHARMM27 force field [6]. Each benchmark simulation was run for a duration of 30 minutes, employing the TIP3P water model. The initial structure of the taxol-bound tubulin tetramer was retrieved from the Protein Data Bank (PDB id 5SYF [5]). The dimensions of the simulation volume were selected to ensure that the distance from the protein's surface to the nearest boundary of the simulation box was never less than two nanometers. Long-range electrostatic interactions were accounted for using the particle mesh Ewald method [3]. Both Coulomb and Lennard-Jones cutoffs were configured to 1.25 nm. For the water box systems, the time step was chosen to be 1 fs and no restraints were used. For the tubulin protofilament, the time step was chosen to be 2 or 4 fs.

In this work, we used two types of integrators: stochastic dynamics (sd) and molecular dynamics (md). Sd is an accurate and efficient leap-frog stochastic dynamics integrator with constraints; coordinates need to be constrained twice per integration step which can take a significant part of the simulation time. Md is a leap-frog algorithm for integrating Newtons equations of motion. When calculating with an integration step of 2 fs, the constraints were imposed on the bonds of atoms with hydrogens; when calculating with an integration step of 4 fs, they were applied to all bonds. Also, for all systems, when using the md integrator with a step of 1 or 2 fs, the "GPU-resident" option was tested, which updates coordinates on the GPU. This parallelization mode is referred to as "GPU-resident" as all force and coordinate data remain resident on the GPU for a number of steps. The "GPU-resident" scheme however is still able to carry out part of the computation on the CPU concurrently with GPU calculation which helps to support the broad range of GROMACS features not all of which are ported to GPU and also allows improving performance by making use of the otherwise mostly idle CPU [1]. Specifications of MD systems used for benchmarking are summarized in Tab. 1.

Table 1. Specification of molecular dynamics systems used in the benchmarks

MD systems	MD system name	Number of atoms	System size, nm
Water box (WB)	WB-10	10,206	$4.7 \times 4.7 \times 4.7$
	WB-80	80,232	$9.3 \times 9.3 \times 9.3$
	WB-200	203,415	$12.7 \times 12.7 \times 12.7$
	WB-500	500,076	$17.2 \times 17.2 \times 17.2$
	WB-1000	1,000,005	$21.7 \times 21.7 \times 21.7$
Tubulin tetramer	Tub-4	307,453	$11.3 \times 12.5 \times 22.2$

2. Results and Discussion

2.1. System Size Has Very Little Impact on Performance

To find out how calculation performance depends on the system size, we have tested the performance of all-atom MD calculations for water box systems of different sizes: 10, 80, 200, 500, and 1000 thousands of atoms. The calculations were carried out for CPU only and CPU+GPU computer architectures both with sd or md integrator type. The table with all performance tests is available via this link. The results for Intel Core i9-13900K CPU and RTX4090 GPU with "GPU-resident" option are presented in Fig. 1. We normalized the performance to the value for the largest WB1000 system in order to compare MD simulation results across systems of various sizes. We can see that, with the exception of a system of ten thousand atoms on CPU+GPU computer architectures, normalized performance practically never changes with system size. The ineffective use of GPU computing resources can be the cause of the system's relatively poor normalized performance for a small system (less than 80,000 atoms). However, biological systems of interest typically contain larger numbers of atoms. On the other hand, small systems are already computationally fast, the computing performance of a system of 10,000 atoms reaches 1.3 μs/day. So, the actual performance depends linearly on system size, and there is no saturation, at least for molecular systems up to 1 million of atoms.

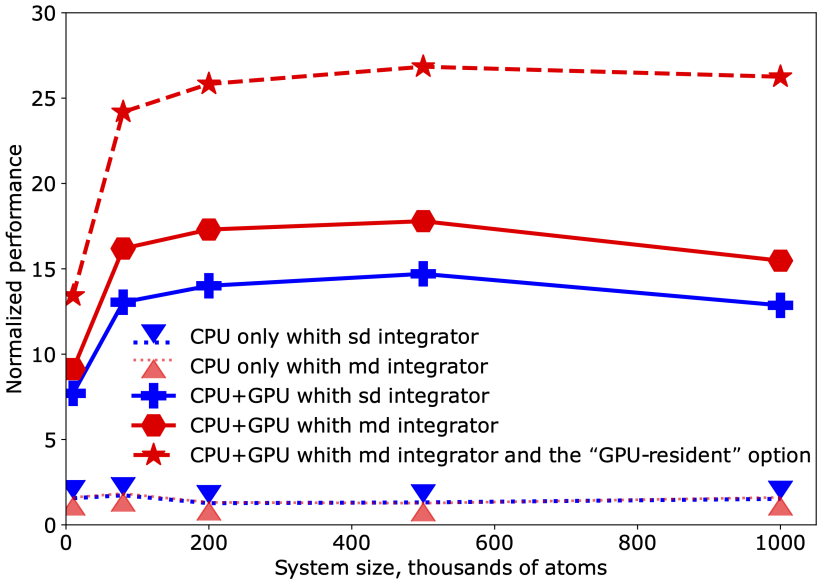

Figure 1. Normalized performance of MD calculation for systems of different sizes both for CPU only and CPU+GPU computer systems. Sd integrator type is colored blue, md – red

2.2. MD Performance is Always Slightly Greater with Md Integrator than with Sd Integrator

We compared MD calculation performance with two integration algorithms, sd and md (Fig. 1). For all our MD calculations, md algorithm gave slightly better performance than sd algorithm. For CPU calculations, the performance of the md algorithm compared to sd was better by no more than 6% (Tab. 2, WB-80 system). For GPU calculations, md algorithm shows

up to 24% better performance than sd (Tab. 2, WB-80 system). Note that the "GPU-resident" option cannot be used with the sd algorithm.

Table 2. Single-node performance (ns/day) for systems of different size depending on various combinations of CPUs and GPUs and protocol of MD calculation

system	Intel Core CPU model	sd CPU	sd CPU+GPU	md CPU	md CPU+GPU	md (CPU+GPU) with the GPU-resident option
WB-10	i9-13900K	156	770	161	913	1343
WB-80	i9-13900K	21.6	163	23	202	302
WB-200	i9-13900K	6.35	70	6.58	86.5	129
WB-500	i9-13900K	2.63	29.4	2.54	35.6	53.7
WB-1000	i9-13900K	1.52	12.9	1.61	15.5	26.3
WB-1000	i9-7900X	0.85	6.53	0.89	9.23	21.6

2.3. "GPU-resident" Option Gives More than Two Times Better Performance

We applied the "GPU-resident" option and it significantly improved MD calculation performance. The larger the molecular system, the greater acceleration was given by this option. For an MD system consisting of a million atoms, the option gave up to 2.3 times better calculation performance (Tab. 2, WB-1000 system, Intel Core i9-7900X processor). Note that the option is not available with sd integrator.

2.4. Latest GPU Gives High Performance Even with Old CPU if the "GPU-resident" Option is Applicable

Surprisingly, newer and more powerful video cards in combination with a central processor from one of the previous generations provided the greatest performance gain. Indeed, the "GPU-resident" option on a computer with 5 years old Intel Core i9-7900X CPU and the newest RTX4090 GPU for an MD system consisting of a million atoms, gave a 2.3 times performance increase, while on the latest Intel Core i9-13900K CPU + RTX4090 GPU it gave only 1.7 times acceleration (Tab. 2, WB-1000 system). This means that there is no need to upgrade the whole computer, it is enough to upgrade only the GPU and this will be the best solution in terms of price-performance ratio, especially if we take into account that the top-end GPU does not exceed 40% of the cost of a new computer.

2.5. Difference in GPUs is Much More Important than Difference in CPUs

CPU upgrade itself from 5 years old Intel Core i9-7900X to the newest Intel Core i9-13900K with RTX2080ti gave only 1.02 times acceleration in the case of tubulin tetramer MD simulation with "GPU-resident" option, while GPU upgrade from 5 years old RTX2080ti to the latest RTX4090 gives up to 2.7 times performance increase (Tab. 2). So the difference in GPUs is

much more important than difference in CPUs when the "GPU-resident" option is available, and there is no sense in putting an old GPU in the newest computer.

Table 3. Single-node performance (ns/day) for tubulin tetramer depending on various combinations of CPUs and GPUs and protocol of MD calculation

time step, fs	integrator	GPU-resident	CPU	GPU	Performance, ns/day
4	sd	No	i9-7900X	RTX2080ti	56.7
4	sd	No	i9-13900K	RTX2080ti	77.0
4	sd	No	i9-7900X	RTX4090	77.1
4	sd	No	i9-13900K	RTX4090	171
4	md	No	i9-7900X	RTX2080ti	69.8
4	md	No	i9-13900K	RTX2080ti	82.4
4	md	No	i9-7900X	RTX4090	116
4	md	No	i9-13900K	RTX4090	211
2	md	No	i9-7900X	no GPU	6.8
2	md	No	i9-13900K	no GPU	12.1
2	md	Yes	i9-7900X	RTX2080ti	46.6
2	md	Yes	i9-13900K	RTX2080ti	47.5
2	md	Yes	i9-7900X	RTX4090	127
2	md	Yes	i9-13900K	RTX4090	171

2.6. Later GPU and CPU Generations Provide Improved Performance

Obviously, a computer with the latest CPU and GPU outperforms any other configuration, so if you need to perform MD calculations at peak performance, it is recommended that you upgrade the entire computer. Compared to the five-year-old i9-7900X CPU + RTX2080ti GPU configuration, the latest i9-13900K CPU + RTX4090 GPU configuration with "GPU-resident" option enabled delivers 3.7× faster tubulin tetramer MD simulation (Tab. 3). At the same time, if you upgrade only the GPU, then the calculations will accelerate only 2.7 times. However, if you must use the sd integrator type, the choice of computing architecture with the best price-performance ratio is to use the latest CPU and GPU. In this case, upgrading only the GPU gives a performance increase of 1.4 times, and if the entire computer is upgraded, the performance will increase by 3 times. Further increase in the productivity of molecular dynamics calculations is possible by increasing the integration step to 4 fs. This, in turn, does not allow the use of the "GPU-resident" option, so in this situation the performance will not increase in proportion to the integration step, that is, by 2 times. Productivity in this case will increase only 1.2 times. Interestingly, when using the sd integrator, GPU upgrade from RTX2080ti to RTX4090 is equivalent to CPU upgrade from i9-7900X to i9-13900K (Tab. 3). When using a computer with the modern architecture (latest GPU and CPU), the sd integrator is two times slower than the md integrator with "GPU-resident" option.

2.7. Lomonosov-2 Supercomputer Architecture Test

We also tested "Volta2" section of Lomonosov-2 supercomputer launched in 2018. Using just the CPU on a single node, we were able to achieve 12.8 ns/day performance (Tab. 4), which is comparable to the latest Intel i9-13900K CPU and more than double the performance of our i9-7900X test processor released in 2018 (Tab. 3). However, if four nodes are used in parallel in CPU-only mode, then performance increases by 2.3 times (Tab. 4).

When using one supercomputer node in CPU+GPU mode, the performance of MD calculations became an impressive 62 ns/day, that is, using GPU speeds up calculations by almost 5 times. However, the modern combination of i9-13900K CPU + RTX4090 GPU provides performance of 114 ns/day, which is almost 2 times faster (Tab. 3). Moreover, enabling the "GPU-resident" option increases the performance of the latter system to the stunning 171 ns/day, whereas the supercomputer node gave only 34.1 ns/day when this option was enabled. Thus, the GPU-resident option provides the greatest performance benefit when using a relatively old CPU and a newer GPU. However, when combined with powerful CPUs and relatively outdated GPUs, this option causes an imbalance of calculations owing to higher data interchange within the system, resulting in a drop in system performance.

Table 4. Performance of "Lomonosov-2" supercomputer (ns/day), dependending on the number of computing nodes for MD simulations of tubulin tetramer with md integrator and 2 fs time step with "GPU-resident" option

Number of nodes	GPU	GPU-resident	Perfomance, ns/day
1	no GPU	No	12.8
4	no GPU	No	29.7
1	2×NVIDIA Tesla V100	No	62.1
4	2×NVIDIA Tesla V100	No	58.7
5	2×NVIDIA Tesla V100	No	96.4
1	2×NVIDIA Tesla V100	Yes	34.1
4	2×NVIDIA Tesla V100	Yes	57.1
5	2×NVIDIA Tesla V100	Yes	81.8

Conclusion

Our comparative performance analysis of GPU-based architectures for all-atom MD simulations of realistic biomolecular systems provides the guidelines for selection of the best computing solution in terms of price-performance ratio. When using the "GPU-resident" option in the GROMACS MD package, you may gain fantastic computation acceleration by employing a current GPU, even in conjunction with a rather outdated CPU. Namely, changing only the GPU allows you to gain nearly three times quicker performance. However, if you also update the CPU, this will speed up the calculation by almost 4 times. MD of the biomolecular system containing a tubulin protofilament in an explicitly specified solvent consisting of more than 300 thousand atoms can be studied with performance of 171 ns/day at time step 2 fs when using single-node computer with the latest CPU and GPU generation architecture (Intel Core i9-13900K and Nvidia RTX4090 respectively). Nevertheless, HPC platforms (e.g., the volta2 partition of

"Lomonosov-2" supercomputer) can be very suitable for simulation experiments with a large number of independent calculations, such as the umbrella sampling technique. For calculations where one task is computed on one node, it would give effective performance in linear proportion with the number of nodes, i.e., it is possible to achieve more than 1 μs/day calculation performance for a tubulin tetramer using the entire volta2 partition (18 nodes), which exceeds the capabilities of even the most advanced one-node computer by an order of magnitude.

Acknowledgments

The study was supported by the Russian Science Foundation, project No. 22-74-00119, https://rscf.ru/en/project/22-74-00119/. GPUs Nvidia RTX4090 were purchased under the Development Program of Moscow State University (order No. 99 of February 6, 2023, contract No. 0183-44-2023).

References

1. Abraham, M., Alekseenko, A., Bergh, C., *et al.*: Gromacs 2023.2 manual (Jul 2023). https://doi.org/10.5281/zenodo.8134388

2. Abraham, M.J., Murtola, T., Schulz, R., *et al.*: Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX 1, 19–25 (2015). https://doi.org/10.1016/j.softx.2015.06.001

3. Ewald, P.P.: Die berechnung optischer und elektrostatischer gitterpotentiale. Annalen der physik 369(3), 253–287 (1921). https://doi.org/10.1002/andp.19213690304

4. Gudimchuk, N.B., McIntosh, J.R.: Regulation of microtubule dynamics, mechanics and function through the growing tip. Nature reviews Molecular cell biology 22(12), 777–795 (2021). https://doi.org/10.1038/s41580-021-00399-x

5. Kellogg, E.H., Hejab, N.M., Howes, S., *et al.*: Insights into the distinct mechanisms of action of taxane and non-taxane microtubule stabilizers from cryo-em structures. Journal of molecular biology 429(5), 633–646 (2017). https://doi.org/10.1016/j.jmb.2017.01.001

6. MacKerell Jr, A.D., Feig, M., Brooks, C.L.: Improved treatment of the protein backbone in empirical force fields. Journal of the American Chemical Society 126(3), 698–699 (2004). https://doi.org/10.1021/ja036959e