



GENESIS: a hybrid-parallel and multi-scale molecular dynamics simulator with enhanced sampling algorithms for biomolecular and cellular simulations

Jaewoon Jung,^{1,†} Takaharu Mori,^{2,†} Chigusa Kobayashi,¹ Yasuhiro Matsunaga,¹ Takao Yoda,³ Michael Feig⁴ and Yuji Sugita^{1,2,5,6*}

GENESIS (Generalized-Ensemble Simulation System) is a new software package for molecular dynamics (MD) simulations of macromolecules. It has two MD simulators, called ATDYN and SPDYN. ATDYN is parallelized based on an atomic decomposition algorithm for the simulations of all-atom force-field models as well as coarse-grained Go-like models. SPDYN is highly parallelized based on a domain decomposition scheme, allowing large-scale MD simulations on supercomputers. Hybrid schemes combining OpenMP and MPI are used in both simulators to target modern multicore computer architectures. Key advantages of GENESIS are (1) the highly parallel performance of SPDYN for very large biological systems consisting of more than one million atoms and (2) the availability of various REMD algorithms (T-REMD, REUS, multi-dimensional REMD for both all-atom and Go-like models under the NVT, NPT, NPAT, and NP γ T ensembles). The former is achieved by a combination of the midpoint cell method and the efficient three-dimensional Fast Fourier Transform algorithm, where the domain decomposition space is shared in real-space and reciprocal-space calculations. Other features in SPDYN, such as avoiding concurrent memory access, reducing communication times, and usage of parallel input/output files, also contribute to the performance. We show the REMD simulation results of a mixed (POPC/DMPC) lipid bilayer as a real application using GENESIS. GENESIS is released as free software under the GPLv2 licence and can be easily modified for the development of new algorithms and molecular models. © 2015 The Authors. *WIREs Computational Molecular Science* published by John Wiley & Sons, Ltd.

How to cite this article:

WIREs Comput Mol Sci 2015, 5:310–323. doi: 10.1002/wcms.1220

[†]These authors contributed equally to this work.

*Correspondence to: sugita@riken.jp

¹Computational Biophysics Research Team, RIKEN Advanced Institute for Computational Science, Kobe, Japan

²Theoretical Molecular Science Laboratory, RIKEN, Wako-shi, Japan

³Nagahama Institute of Bio-Science and Technology, Nagahama, Japan

⁴Department of Biochemistry and Molecular Biology, and Department of Chemistry, Michigan State University, East Lansing, MI, USA

⁵Interdisciplinary Theoretical Science Research Group, RIKEN, Wako-shi, Japan

⁶Laboratory for Biomolecular Function Simulation, RIKEN Quantitative Biology Center, Kobe, Japan

Conflict of interest: The authors have declared no conflicts of interest for this article.

INTRODUCTION

In 1977, McCammon, Gelin, and Karplus carried out the first molecular dynamics (MD) simulation of a protein.¹ Initially, simulations were carried out for a single protein in vacuum. Since the late 1980s, MD simulations of a protein in explicit water have been possible because of the speedup of computers and advances in MD methodologies.^{2,3} Currently, not only soluble proteins but also membrane proteins in explicit lipid bilayers^{4–6} and protein/nucleic acid complexes like ribosomes or RNA polymerases^{7–9} are routinely subjected to MD simulations. Long MD simulations of various biomolecules are possible using highly optimized and parallelized MD software packages (like CHARMM,^{10,11} AMBER,¹² NAMD,¹³ GROMACS,¹⁴ and others^{15–17}) on different computational platforms. However, on currently available general-purpose supercomputers¹⁸ or accelerators like graphics-processing units (GPUs),¹⁹ the MD simulation length is typically limited to 10–100 μ s, which is shorter than many biologically relevant processes. The development of MD-special purpose computers such as MDGRAPE^{20,21} or Anton/Anton 2^{22,23} has allowed an expansion to milliseconds, although the maximum number of particles.²⁴ Note that reliable simulations require not only good sampling statistics but also accurate force field models, where much progress has been made recently as well.^{25–27}

In this Software Focus, we introduce a new package for MD simulations of biomolecules, which we call GENESIS (Generalized-Ensemble Simulation System). The main motivation for the development of GENESIS in spite of many already existing MD programs is to perform efficient all-atom MD simulation of very large biomolecular systems on general-purpose supercomputers. We believe that one of the future applications of MD simulations involves biomolecules under more realistic cellular environments, such as the cytoplasm, organelles, viruses, biological membranes, and nuclei. In the cytoplasm, a huge number of proteins, RNAs, other macromolecules as well as metabolites co-exist and function. To examine the effect of macromolecular crowding^{28–30} on protein structures and dynamics with all-atom MD simulations, at least 10–100 million atoms in a simulation box are required. To make such large-scale all-atom MD simulations being available, we have developed several efficient computational schemes, namely, the inverse lookup table method,³¹ the midpoint cell method³² both for short-range real-space non-bonded calculations, and the efficient parallelization of three-dimensional (3D) Fast Fourier Transform (FFT)³³ for long-range reciprocal-space calculations

(Jung et al., unpublished data) in the particle mesh Ewald (PME) method.^{34,35} In addition, we utilized OpenMP thread-based parallelization for the communication within a multicore CPU and message-passing interface (MPI) for the communication between different CPUs. This hybrid (MPI + OpenMP) parallelization scheme has become more and more popular in high-performance computing software that runs on modern multicore architectures.

The second key motivation in GENESIS development is to provide a platform for testing new enhanced conformational sampling algorithms or multi-scale/multi-resolution models. To overcome the time-scale gap between MD simulations and experiments, these two have been widely recognized as important approaches in the community of computational biophysics and biochemistry.^{36–46} Our own contributions consist of replica-exchange MD (REMD),⁴⁷ multi-canonical REM (MUCAREM),⁴⁸ replica-exchange umbrella sampling⁴⁹ (REUS, this method is also referred to as Hamiltonian REMD⁵⁰ or window-exchange REMD⁵¹), multi-dimensional REMD (MREMD),⁴⁹ and surface-tension REMD (γ -REMD and γ T-REMD).⁵² Many of these methods were developed in collaboration with Yuko Okamoto. In GENESIS, these enhanced sampling simulations are available along with the high performance MD code based on all-atom/Go-like models^{43,53–56} for standard NVT, NPT, NPAT, or NP γ T ensembles.⁵⁷ The source code of GENESIS is written using the modern Fortran language and released to the community as free software under the GPLv2 licence. Therefore, new sampling methods and molecular models can be easily added by other users.

This paper as Software Focus is organized as follows. First, the software design of GENESIS package and the functions in the two MD simulators (ATDYN and SPDYN) are introduced. Second, high-performance aspects of GENESIS are described in detail. A key aspect is the combination of the midpoint cell method with an efficient parallelization of the 3D FFT method. The hybrid parallelization schemes in ATDYN and SPDYN are also outlined. The implementation of various REMD schemes for the NVT, NPT, NPAT, and NP γ T ensembles in GENESIS is described next. Then, we briefly introduce the basic usage of GENESIS and the input/output files and their formats. One of special features in GENESIS is the option of parallel input/output files for simulations of extremely large systems (>10 million atoms). In the last section, we show performance benchmarks for all-atom MD simulations using GENESIS on a PC-cluster and K computer in RIKEN in comparison with other MD packages.

Simulations of a mixed (POPC/DMPC) lipid bilayer using MD, REMD, γ -REMD, and γ T-REMD methods are shown as an example of a real application with GENESIS.

SOFTWARE DESIGN OF GENESIS

Two MD Simulators: ATDYN and SPDYN

GENESIS consists of two MD simulators, namely ATDYN (ATomic decomposition DYNamics) and SPDYN (SPatial decomposition DYNamics), as well as various analysis and setup tools. ATDYN is designed for all-atom MD simulations based on molecular force-field parameters²⁶ and coarse-grained (CG) MD simulations based on Go-like models^{43,53–56} under periodic and nonperiodic boundary conditions. ATDYN is parallelized based on an atomic decomposition scheme with hybrid parallelization using a replicated-data MD algorithm. Although this implementation is less efficient than SPDYN or other domain decomposition schemes, its advantages are (1) easy modification of source codes for testing new sampling methods and multi-scale models and (2) a good load balance for nonperiodic CG MD simulations. Unlike ATDYN, SPDYN uses a spatial decomposition scheme, where the simulation space is divided into subdomains and cells and a distributed-data MD algorithm is used. Each processor contains information on the atoms assigned to each subdomain. As this scheme requires less communication between processors, the parallel efficiency of SPDYN is much better than that of ATDYN.

Both simulators implement PME,^{34,35} and we use FFTE⁵⁸ for the 3D FFT, which is also parallelized based on a hybrid parallelization scheme. In the current version of GENESIS, only the CHARMM force-fields²⁶ (CHARMM19,⁵⁹ CHARMM22/CMAP,^{60,61} CHARMM27,⁶² CHARMM36,^{63,64} and CHARMM37⁶⁵) are available in both simulators. The simulations based on Go-like models utilize the CHARMM input generated by the MMTSB web site.^{66,67} Other force fields are now being implemented and will be available in the next version. As for enhanced conformational sampling methods, REMD simulations are available in both ATDYN and SPDYN. Analysis tools in GENESIS provide basic structural parameters such as distances, angles, RMSDs, and so on, as well as some advanced analysis functions like principal component analysis (PCA)⁶⁸. The usage of GENESIS for MD and REMD simulations is described in later sections and the users' manual.

Acceleration and Parallelization

Inverse Lookup Table

The major bottleneck in MD is the calculation of non-bonded interactions due to van der Waals (vdW) and electrostatic terms. In the PME method,^{34,35} the most time-consuming calculations involve inverse square roots and complementary error functions. Various MD programs utilize lookup tables for energy and gradient evaluation instead of a direct calculation.^{69,70} GENESIS employs a novel lookup table algorithm based on the inverse of the square distance, which we call “inverse lookup table”.³¹ In this approach, the spacing of table entries is decided from the inverse of the squared pair distances. Compared to the lookup table approach in other program,⁷⁰ the “inverse lookup table” approach makes use of larger number of table entries at short pair distances and less number of table entries for larger pair distances. Because the vdW energy changes rapidly at short pair distances, a lot of table entries at short distances improve the accuracy and allow fewer table entries to achieve the same accuracy as in other programs. The computation of nonbonded interactions based on our inverse lookup table is fast due to a small number of points used for 90% of nonbonded interactions in an MD simulation.³¹ For example, let us consider a system with 23,000 atoms and a 12 Å cutoff for nonbonded interactions. Under the assumption of the same total number of table entries, the inverse lookup table uses only 1/30 of the table points for the pairwise distance interval of [6 Å, 12 Å] compared to the lookup tables implemented in the CHARMM program.⁷⁰ Because the distance interval of [6 Å, 12 Å] includes 88% of nonbonded interactions, the inverse lookup table uses much fewer table entries for calculating short-range interactions. This results in fewer L1 cache misses. In benchmarks on K computer,⁷¹ the inverse lookup table improves the performance of nonbonded interactions by up to a 20% speed-up compared to a conventional distance-squared lookup table.³¹

Parallelization of ATDYN

The hybrid parallelization of ATDYN is based on the replicated-data MD algorithm where each MPI processor has a copy of atomic coordinates, velocities, atomic charges, and so on.^{72,73} It starts from a proper distribution of do-loops according to MPI and OpenMP thread numbers. We first assign identification numbers (*id*) to MPI processors and OpenMP thread ranks. Let us assume that the total number of MPI process is N_{MPI} , and the number of OpenMP threads in one MPI process is N_{THREAD} . Here, N_{MPI} and N_{THREAD} are decided such that $N_{\text{MPI}} \times N_{\text{THREAD}}$ is identical to the total number

of CPU cores used. In each core, id is defined as $id = id_{MPI} \times N_{THREAD} + id_{THREAD}$ where the MPI process rank (id_{MPI}) and the thread rank (id_{THREAD}) are automatically decided. Bonded indices are divided into $N_{MPI} \times N_{THREAD}$ blocks and each block is assigned to each CPU core. Nonbonded interactions are parallelized, distributing atom indices according to ids . For each i -th atom, we perform nonbonded calculations on each core only when the remainder after dividing i by $N_{MPI} \times N_{THREAD}$ is equal to id of the core. For the parallelization of reciprocal-space interaction in the PME method,^{34,35} the pencil decomposition scheme of FFT is considered. For this purpose, we make use of the FFTE implementation⁵⁸ because of its efficient hybrid parallelization scheme. In the computation, potential energies in real space (bond, angle, dihedral angle, improper torsion angle, and nonbonded interactions terms) and in reciprocal space could be computed separately on different MPI processors to increase parallel efficiency. The basics of parallelization in the replicated-data MD with PME is described in Ref. 73. After calculating the real-space and reciprocal-space interactions, forces are accumulated by reduction communication. In ATDYN, increasing OpenMP threads within a CPU reduces the MPI communication time and thereby accelerates the MD simulations.

Parallelization of SPDYN

SPDYN is designed for large-scale MD simulations using an efficient hybrid parallelization scheme to match recent hardware trends toward multi-core architectures. In the spatial decomposition scheme, the simulation space is divided into subdomains according to the number of MPI processors (Figure 1(a)). Here, the subdomain size in each dimension is greater than or equal to the cutoff distance of nonbonded interactions. Subdomains are further divided into smaller cells. The cell size of each direction is greater than or equal to half of the cutoff distance. Interactions between particles are distributed over OpenMP threads according to cell pairs for shared-memory parallelization. For efficient cache usage, particle data are sorted according to cell indices. Unlike ATDYN, each processor only contains information of the corresponding subdomain and the buffer region surrounding the subdomain (distributed-data MD algorithm).

The short-range nonbonded interactions in SPDYN are calculated via the midpoint cell method,³² which is an extension of the previously introduced midpoint method⁷⁴ for the hybrid parallelization. In the midpoint cell method, an interaction between two particles is computed in the subdomain containing the midpoint cell of the two cells where each particle

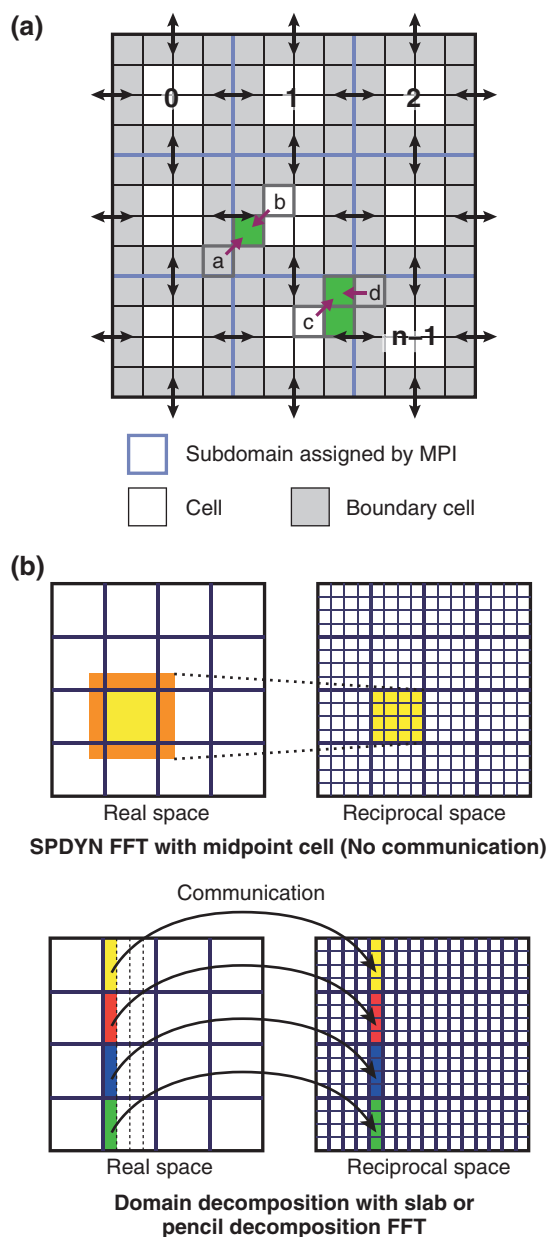


FIGURE 1 | The hybrid (MPI + OpenMP) parallelization scheme in SPDYN. (a) Design of the hybrid parallelization of real space interaction in SPDYN. Adjacent cells for send/receive communication are colored by gray, and MPI communications are shown as black arrows. Midpoint candidate cells for cell pairs (a,b) and (c,d) are colored by green. (b) Two-dimensional views of charge-grid assignment in SPDYN and other MD programs using slab or pencil decomposition FFT.

resides. For example, interactions between particles in cell a and those in cell b are computed in the subdomain containing the midpoint cell of cell a and b (the midpoint cell is designated with a purple arrow in Figure 1(a)). This scheme improves the midpoint method by removing unnecessary calculations of midpoint checking for all of the particle pairs because the

decision of which midpoint cell is appropriate for a given pair is made during the initial setup procedure. If the midpoint cell is not uniquely defined, the cell containing the minimum number of atoms is assigned as the final midpoint cell (the midpoint cell of cell c and d in Figure 1(a)). In the midpoint cell scheme, like midpoint method, sometimes interactions between a pair of particles on a processor on which neither particle resides are calculated (neutral territory method).⁷⁵ For interactions between particles in different subdomains, data of adjacent cells for each subdomain are communicated by MPI send/receive (black arrow in Figure 1(a)). This scheme is very effective in the context of a hybrid parallelization scheme because of the small amount of communication originating from the midpoint scheme and the efficient use of memory by grouping particles cell-wise. In a cutoff-based MD simulation of 1 million atoms, there is a speed up of 2000-fold with 32,768 cores compared to that achieved with 8 cores (parallel efficiency is 50%)³². The good shared-memory parallel efficiency is obtained by avoiding concurrent memory access in multi-thread calculations.

SPDYN also introduces good scalability for the reciprocal-space calculation by efficient parallelization of the 3D FFT. The 3D FFT is a key component of the PME scheme that reduces the computational cost of nonbonded interactions to $O(N\log N)$.³⁴ Despite its usefulness, the 3D FFT is the main bottleneck in computation when using a large number of processors due to frequent communications. We developed a novel parallelization strategy in the 3D FFT using a volumetric decomposition scheme. In the midpoint cell method, each MPI processor contains data of its own subdomain and surrounding cells (yellow and orange in the upper panel of Figure 1(b), respectively), which are sufficient to compute charge data using B-spline functions in the subdomain. The communication of the charge data before the 3D FFT calculation is avoided due to the same volumetric decomposition between real and reciprocal spaces (upper panel of Figure 1(b)). In contrast, the slab or pencil decomposition 3D FFT schemes are often utilized in the distributed-data MD algorithms. In this case, all-to-all (or send/receive) communications of charge data between real and reciprocal space are required (bottom panel of Figure 1(b)). Our scheme, a combination between the midpoint cell method and the volumetric 3D FFT, is extremely useful for special network topologies such as torus networks because the number of processors involved in communication is minimized.

The benchmark performance of parallel FFT in SPDYN is as follows: for a $512 \times 512 \times 512$ grid,

it is scalable up to 131,076 cores with the sum of forward/backward FFT costing less than 4 ms on K computer, and, for a $1024 \times 1024 \times 1024$ grid, it is scalable up to 262,152 cores with less than 10 ms execution time. As a result, for a 1 million atom system, the parallel efficiency of FFT in GENESIS provides a 890-fold speed-up on 32,768 cores relative to that achieved on 8 cores, even though the PME electrostatics is calculated at every step.

Replica-Exchange Simulations

Implementation of REMD

The REMD method is one of the enhanced conformational sampling methods widely used for systems with rugged free-energy landscapes.^{47,76} In the original temperature-exchange (T-REMD) method, copies of the original system are prepared and different temperatures are assigned to each replica. Each replica is simulated in a canonical ensemble, and the target temperatures are exchanged between a pair of replicas during a simulation. Exchanging temperature enforces a random walk in temperature space, resulting in the simulation surmounting energy barriers. In the method, atomic momenta are rescaled after replica exchange to satisfy the detailed balance condition. If thermostat and barostat momenta are included in the equations of motion as in the Martyna-Tobias-Klein algorithm,⁷⁷ these variables should also be rescaled.^{78,79} Recently, different types of replica exchanges schemes have been proposed.^{80–83}

In GENESIS, exchangeable parameters are temperature (T-REMD),⁴⁷ pressure (P-REMD),⁸⁴ surface tension (γ -REMD),⁵² and umbrella potentials⁴⁹ (REUS). The simulations can be performed in various ensembles such as NVT, NPT, NPAT, and NP γ T.⁵⁷ For REUS simulations, umbrella potentials for distance, angle, dihedral angle, and positional restraints can be used. We can carry out not only one-dimensional but also multi-dimensional REMD simulations⁴⁹ without limitations on the number of dimensions. All combinations between exchangeable parameters (T , P , and γ), ensembles, and umbrella potentials are possible. For instance, in surface-tension REMD, not only surface tension (γ) but also temperature can be exchanged to enhance the conformational sampling of biological systems (γ -REMD and γ T-REMD). In the current implementation, parameters are exchanged only between neighboring pairs. Let us consider that we have n different parameters assigned to each replica: $\lambda_1, \lambda_2, \dots, \lambda_n$ (n is even). There are two patterns of parameter exchange: $\lambda_1 \leftrightarrow \lambda_2, \lambda_3 \leftrightarrow \lambda_4, \dots, \lambda_{n-1} \leftrightarrow \lambda_n$ (Pattern 1) and $\lambda_2 \leftrightarrow \lambda_3, \lambda_4 \leftrightarrow \lambda_5, \dots, \lambda_{n-2} \leftrightarrow \lambda_{n-1}$ (Pattern 2). Exchanges using Patterns 1 and 2 are alternately repeated during simulations. In the case of

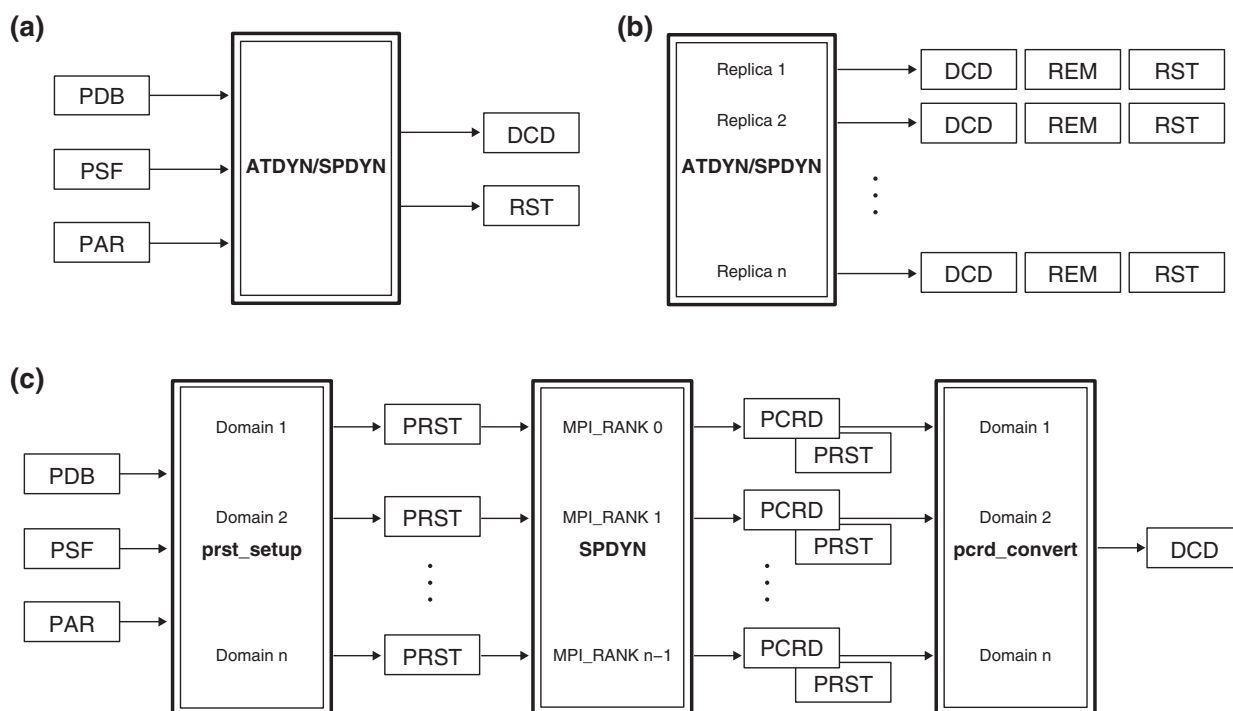


FIGURE 2 | File input/output in GENESIS. (a) Scheme in standard MD simulations. PDB, Protein Data Bank; PSF, Protein Structure File; PAR, Parameter; RST, restart file. Coordinates and velocities are generated in the standard DCD file format. (b) Scheme in REMD simulations. REM, parameter index file. (c) Scheme in large-scale MD simulations. PRST, parallel restart files; PCRD, parallel coordinates files.

two-dimensional REMD simulation, parameters in the first and second dimensions are exchanged alternatively.

The REMD algorithm is suitable for parallel computation using MPI. In GENESIS, the global MPI communicator is split into subgroups, each of which is assigned to a replica. As mentioned above, the MD simulation of each replica is further parallelized using the hybrid parallelization scheme. The communication cost for the replica-exchange scheme is almost negligible, because only the value(s) of one or few variables instead of a set of coordinates is transferred among replicas. In T-REMD, P-REMD, γ -REMD, and REUS (H-REMD), the transferred variables are potential energy, volume, area, and restraint energy, respectively.

BASIC USAGE OF GENESIS

Input/Output for Standard MD

To perform simulations with GENESIS, users first prepare PDB (Protein Data Bank), PSF (Protein Structure File), and PAR (PARAMeter) files as input. The PSF file can be generated with PSFGEN supplied with NAMD,¹³ VMD,^{8,5} CHARMM,¹¹ or with other tools. The PAR file is expected to be in standard CHARMM format.

During simulations, trajectory files (coordinates and velocities) in standard DCD format are generated as output (Figure 2(a)). A restart file (RST) that contains atomic coordinates, velocities, box size, thermostat momenta, and barostat momenta is also written out.

Input/Output for REMD

In REMD simulations, each replica generates individual trajectory data (energy, coordinates, velocities, and parameter index (REM file)) and restart files (Figure 2(b)). Because trajectory data are not sorted by replica condition (e.g., temperature in T-REMD), users have to convert trajectories to those sorted by a certain condition after the simulation using a trajectory converter tool (remd_convert).

Input/Output for Large-Scale MD

To perform MD simulations of a huge system (>10 million atoms) using SPDYN, the initial setup time may be substantial. Moreover, if single trajectory and restart files are generated during such a simulation, the communication time for gathering coordinates of all atoms on a specific CPU to write the output becomes very long, because each MPI processor has only local structure information in its domain.

To avoid these problems, parallel input/output (I/O) function is available in SPDYN (Figure 2(c)). This function allows file I/O on each MPI processor. To use this feature, users first create parallel (or local) restart files (PRST files) from the initial PDB structure using a setup tool (prst_setup). The MD simulation is then performed using those PRST and PAR files as input. The output data are also written out in parallel with each file containing only local structural information such as coordinates (PCRD) and velocities. After the simulation is finished, users can combine the individual trajectory files into a single DCD file of the whole system or selected atoms by using a converter tool (pcrd_convert).

BENCHMARK PERFORMANCE OF GENESIS

Benchmark Performance on Conventional PC clusters: Comparison with Other MD Programs

Benchmark performance tests were carried out on our in-house PC cluster in which 32 nodes are connected with InfiniBand FDR. Each node has two Intel Xeon E5-2690 CPUs, each with eight 2.9 GHz cores. In total, up to 512 cores were utilized in the benchmark tests. Intel compilers (version 12.1) with OpenMPI (version 1.4.4) were used to compile the MD programs (GENESIS, NAMD version 2.9,¹³ and CHARMM c40a2⁸⁶). The performance was compared for three benchmark systems: DHFR (23,558 atoms in a $62.23 \times 62.23 \times 62.23 \text{ \AA}^3$ box), ApoA1 (92,224 atoms in a $108.86 \times 108.86 \times 77.76 \text{ \AA}^3$ box), and STMV (1,066,628 atoms in a $216.83 \times 216.83 \times 216.83 \text{ \AA}^3$ box). All input files were obtained from the NAMD webpage.⁸⁷ For the CHARMM program, we used domdec that implements a domain decomposition scheme and allows processors to be split between the calculation of real-space and reciprocal-space interactions.⁸⁶ In CHARMM, splitting processors shows better performance for small systems when increasing the number of processors. The processors are split according to a ratio of 3:1 (real-space:reciprocal-space). In the case of ATDYN, we use the ratio 1:1. In all systems, we tried to assign the same conditions: cutoff = 10 Å, pairlist cutoff = 11.5 Å, pairlist update every 10 steps, 2-fs time step with SHAKE/SETTLE^{88,89} constraints with NVE ensemble. The PME grid sizes for DHFR, ApoA1, and STMV are $64 \times 64 \times 64$, $128 \times 128 \times 96$, and $256 \times 256 \times 256$, respectively. In all cases, we used double precision arithmetic for real numbers. Multiple time-step integrators like r-RESPA⁹⁰ were not

used. The performance shown in Figure 3(a–c) was evaluated as the CPU time difference between 1000 and 2000 integration steps. The best performance up to 512 cores is shown in Table 1. On the PC cluster, NAMD shows the best performance for all the three systems. The best performance of SPDYN lies between CHARMM and NAMD. For small numbers of processors, CHARMM shows better performance than SPDYN, but SPDYN scales better with an increasing number of cores. ATDYN shows the worst performance, which is due to an inefficient parallelization scheme, as expected.

Benchmark Performance on K Computer: Comparison with NAMD

We also tested the benchmark performance for STMV on K computer⁷¹ by comparing with NAMD. K computer consists of over 80,000 SPARC64 VIIIfx processors. Each computing node contains a single processor (8 cores, 2.0 GHz), 6 MBL2 cache, 16 GB memory, and 64 GB/s memory throughput, providing 128 GFLOPS at peak performance. The total peak performance is 10.51 PFLOPS, making this system first in the Top500 of 2011. Each node is connected by a 6D mesh/torus interconnect (Tofu network), providing a logical 3D torus network for each job.

The simulation condition of the benchmark is identical to the tests on the PC cluster. However, we noticed that NAMD works better with multiple threads if a large number of processors is assigned. Because one CPU on K computer has eight cores, we assign seven worker threads combined with one communication thread in one MPI processor. In the case of SPDYN, we used eight OpenMP threads. The overall and best performance is described in Figure 3(d) and Table 1. In Figure 3(d), there is a noticeable difference between SPDYN and NAMD: up to 2048 cores, NAMD works better, which is consistent with the benchmark results on the PC cluster. NAMD does not show good scalability after 2048 cores while SPDYN shows good parallel scaling behaviour up to 32,768 cores. In other words, NAMD is more optimized for a small number of processors, whereas SPDYN has better parallel efficiency than NAMD. The benchmark result of STMV using SPDYN is very impressive considering that we calculated the PME reciprocal space interaction at every step.

Performance in Simulations of Large Systems

The larger systems that we used for further benchmark of SPDYN are all-atom models of the cytoplasm of the minimal bacterium *Mycoplasma genitalium*. In one system, 185 proteins, 28 RNAs, 3 ribosomes,

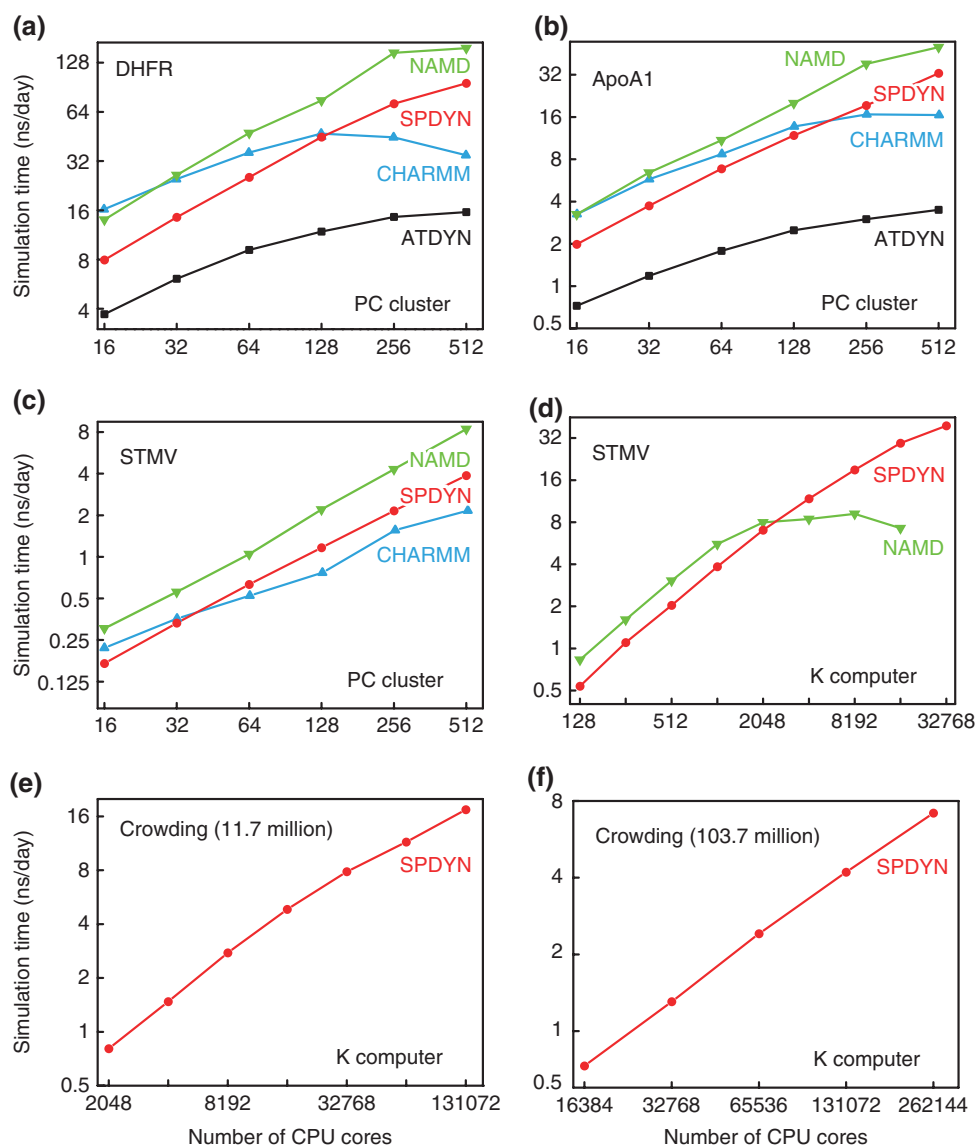


FIGURE 3 | Benchmark performance of MD simulations of (a) DHFR, (b) ApoA1, and (c) STMV on PC clusters, and (d) STMV, macromolecular crowding systems consisting of (e) 11.7 million atoms and (f) 103.7 million atoms on K computer.

TABLE 1 | Best Benchmark Performance (ns/day) of DHFR and ApoA1 on PC Clusters and STMV on K Computer

System	DHFR	ApoA1	STMV
SPDYN (PC)	95.78 (512, 8 ¹)	32.74 (512, 4 ¹)	3.88 (512, 4 ¹)
ATDYN (PC)	15.60 (512, 8 ¹)	3.50 (512, 16 ¹)	—
NAMD (PC)	157.10 (512)	50.31 (512)	8.43 (512)
CHARMM(PC)	47.21 (128)	16.65 (512)	2.16 (512)
SPDYN (K)	—	—	39.17 (32,768)
NAMD (K)	—	—	9.18 (8,192)

Numbers in parentheses are number of CPU cores used to get the best performance.

¹Number of OpenMP threads used to get the best performance.

5005 metabolites, 23,049 ions, and 2,944,143 water molecules are included in a $50 \times 50 \times 50 \text{ nm}^3$ box, and the resulting total number of atoms is 11,737,298 (11.7 million). In the second system, 103,708,785 atoms (103.7 million) with 1258 proteins, 284 RNAs, 31 ribosomes, 41,006 metabolites, 214,000 ions, and 26,263,505 water molecules are included in a $105 \times 105 \times 105 \text{ nm}^3$ box. The systems were composed to be biochemically consistent following a metabolic network reconstruction. Molecular concentrations were estimated based on proteomic and metabolomic data for *Mycoplasma pneumoniae*,⁹¹ the closest relative of *M. genitalium*. Details of how the model was constructed are provided in Ref. 92. The

two systems connect detailed structural views of biology to cellular levels and are attractive for investigating effects of macromolecular crowding on protein structures and dynamics in atomic detail. The benchmark MD simulations used CHARMM36⁶⁴ and CGenFF force field⁹³ parameters for proteins, RNAs, and metabolites, and the TIP3P model for water molecules. The PME method³⁵ was used for computing electrostatic interactions with a 12-Å real-space cutoff. SHAKE was applied to all bonds that connect hydrogen atoms and others, and SETTLE was used for treating TIP3P water as a rigid molecule. Due to the constraints, the time step for integrating the Newtonian equation of motion was 2 fs. PME grid sizes were $512 \times 512 \times 512$ and $1024 \times 1024 \times 1024$, for the 11.7 and 103.7 million atom systems, respectively. As a multiple time-step integrator like r-RESPA⁹⁰ was not used here, long-range electrostatics were evaluated at every step.

The benchmark results are shown in Figures 3(e) and 3(f). SPDYN shows scalability up to 130,000 and 260,000 cores for the systems consisting of 11.7 million atoms and 103.7 million atoms, respectively. The resulting simulation time is 17.5 and 6.5 ns/day for these two systems. Production runs of these two systems are now under way on K computer and results will be published elsewhere.

APPLICATION OF GENESIS

Finally, we show an MD simulation using GENESIS to demonstrate the usefulness. Mixed lipid systems are the major structural component of biological membranes. They are involved in a number of cellular processes such as signal transduction, membrane trafficking, and immune responses.⁹⁴ However, little is known about their structure. As such a complex system has many possible configurations, enhanced sampling is essential for reliable simulations. To examine the usefulness of REMD algorithms for mixed lipid bilayer systems, we carried out three different REMD simulations (T-REMD, γ -REMD, and two-dimensional γ T-REMD). The target system contains 40 POPC lipids, 40 DMPC lipids, and 3680 TIP3P waters. In the initial structure, two lipid phases are completely separated. In T-REMD, we used 32 replicas, where temperatures are exponentially distributed between 303.15 and 373.74 K. In γ -REMD, we used four replicas ($\gamma = 0, 6, 12, \text{ and } 18 \text{ dyn/cm}$). In γ T-REMD, temperatures are exponentially distributed between 303.15 and 318.01 K (8 temperatures) and surface tensions are set to 0, 6, 12, and 18 dyn/cm ($8 \times 4 = 32$ replicas in total). We used CHARMM36 force field parameters for the lipids.⁶³ The SHAKE and SETTLE algorithms

were applied for rigid bonds and waters, respectively. We carried out 100 ns simulation (time step = 2 fs) for each replica, and replica exchange was attempted at every 50 ps. We also carried out conventional MD at $T = 303.15 \text{ K}$. All simulations were carried out in the NPT or NP γ T ensemble at $P = 1 \text{ atm}$.

In all simulations, the lipid bilayers maintained structural integrity, that is, there was no collapse in the bilayer structure, even at high temperature and under high surface tension. The acceptance ratio for the Metropolis criteria in all REMD simulations was about 0.3 in temperature space and 0.4 in surface-tension space, indicating frequent parameter exchanges between pairs of replicas. Figure 4(a) shows snapshots obtained from the MD and T-REMD simulations at $T = 303.15 \text{ K}$. In T-REMD, the two lipid components are mixed more compared to MD, simply because high temperature accelerates lipid lateral diffusion. We analysed the mean-square displacement (MSD) of the centre of mass of lipid to examine lipid lateral diffusion (Figure 4(b)). All REMD simulations showed accelerated diffusion, where T-REMD was the most efficient and the order was T-REMD > γ T-REMD > γ -REMD > MD. The enhanced lipid lateral diffusion observed in γ -REMD is due to free-area effects.^{52,95} Note that diffusion and mixing are not identical in the lipid-bilayer systems. To quantify the degree of mixing of two lipid components, we analysed the number of contact pairs between POPC and DMPC. Contact pairs were defined based on the distance between the centres of mass of lipids with a cutoff distance of 10 Å. Figure 4(c) shows a histogram of the number of contact pairs obtained from the snapshots at $T = 303.15 \text{ K}$ and $\gamma = 0 \text{ dyn/cm}$ after 10 ns. We found that the mixing of the two components was enhanced in T-REMD compared to MD, while it was suppressed in γ -REMD. In γ T-REMD, both enhancement and suppression were observed, presumably because diffusion is accelerated at high temperature while it is suppressed under high surface tension. There is a controversy about the effects of membrane tension on the phase formation in mixed lipid bilayers. Some studies suggest that membrane tension induces phase separation,^{96–99} while others suggest mixing^{100–102} or transition to other phases¹⁰³ would result. Our observation that γ -REMD and γ T-REMD suppresses mixing agrees with the former studies. We suggest that REMD methods are useful for exploring structures at phase boundaries in mixed lipid bilayer systems.

CONCLUSIONS

We have developed new MD software, which we call GENESIS. This contains two MD simulators,

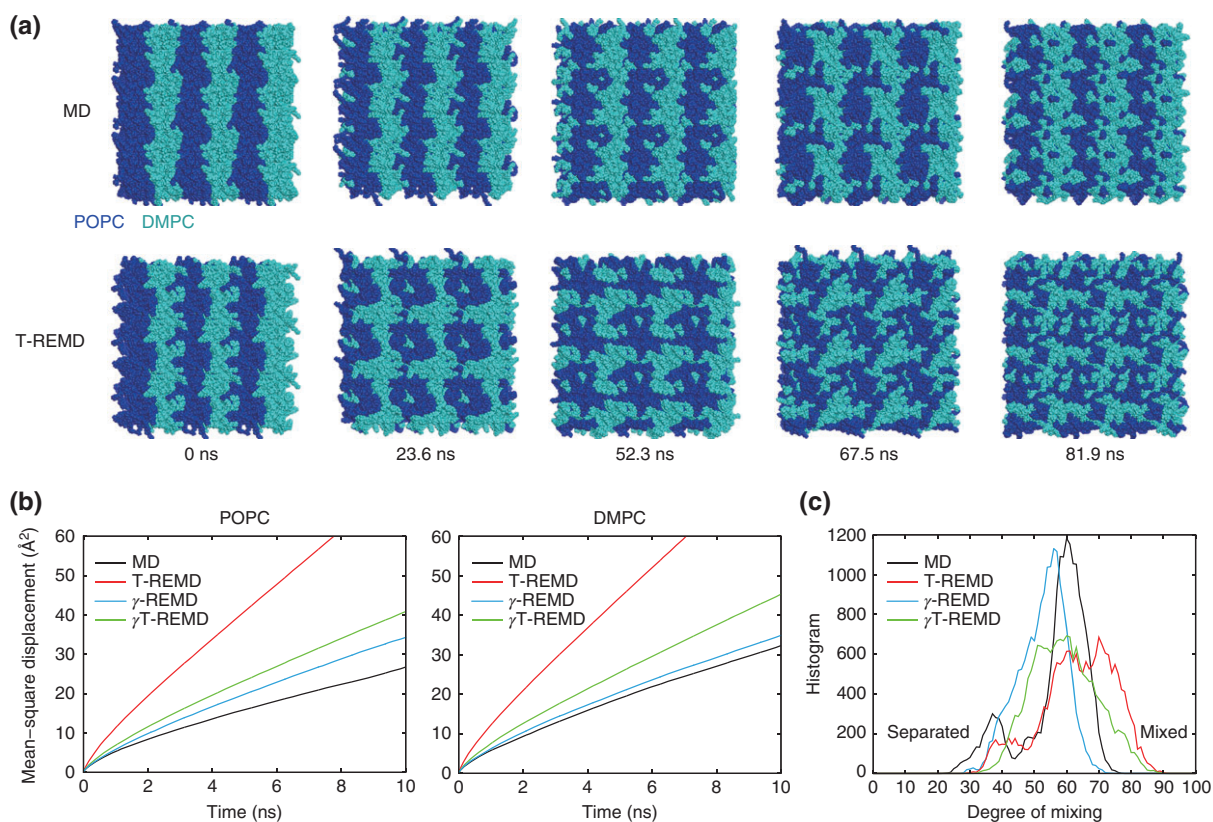


FIGURE 4 | REMD simulations of POPC/DMPC mixed lipid bilayers. (a) Snapshots in the MD simulation at $T = 303.15$ K and $P = 1$ atm (upper panels), and snapshots in the T-REMD simulation at $T = 303.15$ K and $P = 1$ atm (lower panels). POPC and DMPC are coloured by blue and cyan, respectively. Structures in unit and image cells are shown together. (b) Mean-square displacements (MSD) of POPC lipids (left panel) and DMPC lipids (right panel) in MD, T-REMD, γ -REMD, and γ T-REMD. MSD is computed for each replica and averaged over all replicas. (c) Histogram of degree of mixing (number of POPC–DMPC contact pairs) at $T = 303.15$ K and $\gamma = 0$ dyn/cm in MD, T-REMD, γ -REMD, and γ T-REMD.

namely ATDYN and SPDYN, which are parallelized with hybrid (MPI + OpenMP) schemes based on the replicated-data and the distributed-data MD algorithms, respectively. GENESIS also allows various REMD simulations (T-REMD, REUS, and multi-dimensional REMD) both for all-atom and Go-like models in the NVT, NPT, NPAT, and $NP\gamma$ T ensembles. The benchmark performance tests suggest that the scalability of GENESIS in a PC-cluster is comparable to other existing MD programs. For

very large biological systems containing more than one million atoms, the performance on K computer in RIKEN is better than other MD programs due to its good parallel efficiency. GENESIS allows high-performance MD simulations of large biological systems using the realistic representations on modern multicore architectures and advances biological sciences by rigorously connecting physical principles at the molecular level to biological phenotypes at the cellular level.

ACKNOWLEDGMENT

We would like to thank Dr. Isseki Yu at RIKEN for valuable discussions about the cytoplasmic simulations. We also thank Drs. Takashi Imai, Raimondas Galvelis, Ryuhei Harada, Yasuaki Komuro, Wataru Nishima, Naoyuki Miyashita, Yasuhito Karino, Suyong Re, Yasumasa Joti at RIKEN, Hikaru Inoue and Tomoyuki Noda at Fujitsu Ltd., and Norio Takase at Isogo Soft for contributing to the development of GENESIS. This research used the computational resources of K computer provided by the RIKEN Advanced Institute for Computational Science through the HPCI System Research project (Project ID: ra000009, hp120060, hp120309, hp130003, hp140091, hp140169, and hp140229). We thank the RIKEN Integrated Cluster of Clusters (RICC) for additional computer

resources used in the calculations. This research was supported in part by a Grant-in-Aid for Scientific Research on Innovative Areas (No. 26119006) and the Fund from the High Performance Computing Infrastructure (HPCI) Strategic Program of the Ministry of Education, Culture, Sports, Science and Technology (MEXT) (to YS).

REFERENCES

1. McCammon JA, Gelin BR, Karplus M. Dynamics of folded proteins. *Nature* 1977, 267:585–590.
2. Levitt M, Sharon R. Accurate simulation of protein dynamics in solution. *Proc Natl Acad Sci USA* 1988, 85:7557–7561.
3. Duan Y, Kollman PA. Pathways to a protein folding intermediate observed in a 1-microsecond simulation in aqueous solution. *Science* 1998, 282:740–744.
4. Berneche S, Roux B. Energetics of ion conduction through the K⁺ channel. *Nature* 2001, 414:73–77.
5. de Groot BL, Grubmuller H. Water permeation across biological membranes: mechanism and dynamics of aquaporin-1 and GlpF. *Science* 2001, 294:2353–2357.
6. Tajkhorshid E, Nollert P, Jensen MO, Miercke LJW, O'Connell J, Stroud RM, Schulten K. Control of the selectivity of the aquaporin water channel family by global orientational tuning. *Science* 2002, 296:525–530.
7. Gumbart J, Trabuco LG, Schreiner E, Villa E, Schulten K. Regulation of the protein-conducting channel by a bound ribosome. *Structure* 2009, 17:1453–1464.
8. Feig M, Burton ZF. RNA polymerase II with open and closed trigger loops: active site dynamics and nucleic acid translocation. *Biophys J* 2010, 99:2577–2586.
9. Sanbonmatsu KY. Computational studies of molecular machines: the ribosome. *Curr Opin Struct Biol* 2012, 22:168–174.
10. Brooks BR, Bruccoleri RE, Olafson BD, States DJ, Swaminathan S, Karplus M. CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. *J Comput Chem* 1983, 4:187–217.
11. Brooks BR, Brooks CL III, Mackerell AD Jr, Nilsson L, Petrella RJ, Roux B, Won Y, Archontis G, Bartels C, Boresch S, et al. CHARMM: the biomolecular simulation program. *J Comput Chem* 2009, 30:1545–1614.
12. Salomon-Ferrer R, Case DA, Walker RC. An overview of the Amber biomolecular simulation package. *WIREs Comput Mol Sci* 2013, 3:198–210.
13. Phillips JC, Braun R, Wang W, Gumbart J, Tajkhorshid E, Villa E, Chipot C, Skeel RD, Kale L, Schulten K. Scalable molecular dynamics with NAMD. *J Comput Chem* 2005, 26:1781–1802.
14. Pronk S, Pall S, Schulz R, Larsson P, Bjelkmar P, Apostolov R, Shirts MR, Smith JC, Kasson PM, van der Spoel D, et al. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics* 2013, 29:845–854.
15. Bowers KJ, Chow E, Xu H, Dror RO, Eastwood MP, Gregersen BA, Klepeis JL, Kolossvary I, Moraes MA, Sacerdoti FD, et al. Scalable algorithms for molecular dynamics simulations on commodity clusters. In: *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'06)*, Tampa, FL, 2006.
16. Ikeguchi M. Partial rigid-body dynamics in NPT, NPAT and NP γ T ensembles for proteins and membranes. *J Comput Chem* 2004, 25:529–541.
17. Andoh Y, Yoshii N, Fujimoto K, Mizutani K, Kojima H, Yamada A, Okazaki S, Kawaguchi K, Nagao H, Iwahashi K, et al. MODYLAS: a highly parallelized general-purpose molecular dynamics simulation program for large-scale systems with long-range forces calculated by fast multipole method (FMM) and highly scalable fine-grained new parallel processing algorithms. *J Chem Theory Comput* 2013, 9:3201–3209.
18. Service RF. Who will step up to exascale? *Science* 2013, 339:264–266.
19. Owens JD, Houston M, Luebke D, Green S, Stone JE, Phillips JC. GPU computing. *Proc IEEE* 2008, 96:879–899.
20. Narumi T, Ohno Y, Okimoto N, Koishi T, Suenaga A, Futatsugi N, Yanai R, Himeno R, Fujikawa S, Taiji M. A 55 TFLOPS simulation of amyloid-forming peptides from yeast prion Sup35 with the special-purpose computer system MDGRAPE-3. In: *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'06)*, Tampa, FL, 2006.
21. Ohmura I, Morimoto G, Ohno Y, Hasegawa A, Taiji M. MDGRAPE-4: a special-purpose computer system formolecular dynamics simulations. *Philos Trans A Math Phys Eng Sci* 2014, 372:20130387.
22. Shaw DE, Dror RO, Salmon JK, Grossman JP, Mackenzie KM, Bank JA, Young C, Deneroff MM, Batson B, Bowers KJ, et al. Millisecond-scale molecular dynamics simulations on Anton. In: *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'09)*, Portland OR, 2009.
23. Shaw DE, Bank JA, Batson B, Butts JA, Chao JC, Deneroff MM, Dror RO, Even A, Fenton CH, Forte A, et al. Anton 2: raising the bar for performance and programmability in a special-purpose molecular dynamics supercomputer. In: *International Conference for High Performance Computing, Networking, Storage and Analysis (SC'14)*, New Orleans, LA, 2014.

24. Shaw DE, Maragakis P, Lindorff-Larsen K, Piana S, Dror RO, Eastwood MP, Bank JA, Jumper JM, Salmon JK, Shan YB, et al. Atomic-level characterization of the structural dynamics of proteins. *Science* 2010, 330:341–346.
25. Ponder JW, Case DA. Force fields for protein simulations. *Adv Protein Chem* 2003, 66:27–85.
26. Zhu X, Lopes PEM, MacKerell AD. Recent developments and applications of the CHARMM force fields. *WIREs Comput Mol Sci* 2012, 2:167–185.
27. Piana S, Lindorff-Larsen K, Shaw DE. How robust are protein folding simulations with respect to force field parameterization? *Biophys J* 2011, 100:L47–L49.
28. McGuffee SR, Elcock AH. Diffusion, crowding & protein stability in a dynamic molecular model of the bacterial cytoplasm. *PLoS Comput Biol* 2010, 6:e1000694.
29. Ando T, Skolnick J. Crowding and hydrodynamic interactions likely dominate in vivo macromolecular motion. *Proc Natl Acad Sci USA* 2010, 107:18457–18462.
30. Feig M, Sugita Y. Reaching new levels of realism in modeling biological macromolecules in cellular environments. *J Mol Graph Model* 2013, 45:144–156.
31. Jung J, Mori T, Sugita Y. Efficient lookup table using a linear function of inverse distance squared. *J Comput Chem* 2013, 34:2412–2420.
32. Jung J, Mori T, Sugita Y. Midpoint cell method for hybrid (MPI + OpenMP) parallelization of molecular dynamics simulations. *J Comput Chem* 2014, 35:1064–1072.
33. Eleftheriou M, Moreira JE, Fitch BG, Germain RS. A volumetric FFT for BlueGene/L. *High Performance Computing—Hipc* 2003, 2913:194–203.
34. Darden T, York D, Pedersen L. Particle mesh Ewald: An $N \cdot \log(N)$ method for Ewald sums in large systems. *J Chem Phys* 1993, 98:10089–10092.
35. Essmann U, Perera L, Berkowitz ML, Darden T, Lee H, Pedersen LG. A smooth particle mesh Ewald method. *J Chem Phys* 1995, 103:8577–8593.
36. Mitsutake A, Sugita Y, Okamoto Y. Generalized-ensemble algorithms for molecular simulations of biopolymers. *Biopolymers* 2001, 60:96–123.
37. Hamelberg D, Mongan J, McCammon JA. Accelerated molecular dynamics: a promising and efficient simulation method for biomolecules. *J Chem Phys* 2004, 120:11919–11929.
38. Liwo A, Czaplewski C, Ołdziej S, Scheraga HA. Computational techniques for efficient conformational sampling of proteins. *Curr Opin Struct Biol* 2008, 18:134–139.
39. Huang XH, Bowman GR, Bacallado S, Pande VS. Rapid equilibrium sampling initiated from nonequilibrium data. *Proc Natl Acad Sci U S A* 2009, 106:19765–19769.
40. Barducci A, Bonomi M, Parrinello M. Metadynamics. *WIREs Comput Mol Sci* 2011, 1:826–843.
41. Sutto L, Marsili S, Gervasio FL. New advances in metadynamics. *WIREs Comput Mol Sci* 2012, 2:771–779.
42. Kenzaki H, Koga N, Hori N, Kanada R, Li WF, Okazaki K, Yao XQ, Takada S. CafeMol: a coarse-grained biomolecular simulator for simulating proteins at work. *J Chem Theory Comput* 2011, 7:1979–1989.
43. Takada S. Coarse-grained molecular simulations of large biomolecules. *Curr Opin Struct Biol* 2012, 22:130–137.
44. Saunders MG, Voth GA. Coarse-graining methods for computational biology. *Annu Rev Biophys* 2013, 42:73–93.
45. Kar P, Gopal SM, Cheng YM, Predeus A, Feig M. PRIMO: a transferable coarse-grained force field for proteins. *J Chem Theory Comput* 2013, 9:3769–3788.
46. Ingolfsson HI, Lopez CA, Uusitalo JJ, de Jong DH, Gopal SM, Periole X, Marrink SJ. The power of coarse graining in biomolecular simulations. *WIREs Comput Mol Sci* 2014, 4:225–248.
47. Sugita Y, Okamoto Y. Replica-exchange molecular dynamics method for protein folding. *Chem Phys Lett* 1999, 314:141–151.
48. Sugita Y, Okamoto Y. Replica-exchange multicanonical algorithm and multicanonical replica-exchange method for simulating systems with rough energy landscape. *Chem Phys Lett* 2000, 329:261–270.
49. Sugita Y, Kitao A, Okamoto Y. Multidimensional replica-exchange method for free-energy calculations. *J Chem Phys* 2000, 113:6042–6051.
50. Fukunishi H, Watanabe O, Takada S. On the Hamiltonian replica exchange method for efficient sampling of biomolecular systems: application to protein structure prediction. *J Chem Phys* 2002, 116:9058–9067.
51. Park S, Kim T, Im W. Transmembrane helix assembly by window exchange umbrella sampling. *Phys Rev Lett* 2012, 108:108102.
52. Mori T, Jung J, Sugita Y. Surface-tension replica-exchange molecular dynamics method for enhanced sampling of biological membrane systems. *J Chem Theory Comput* 2013, 9:5629–5640.
53. Taketomi H, Ueda Y, Go N. Studies on protein folding, unfolding and fluctuations by computer-simulation, 1: effect of specific amino-acid sequence represented by specific inter-unit interactions. *Int J Pept Protein Res* 1975, 7:445–459.
54. Clementi C, Nymeyer H, Onuchic JN. Topological and energetic factors: what determines the structural details of the transition state ensemble and "en-route" intermediates for protein folding? An investigation for small globular proteins. *J Mol Biol* 2000, 298:937–953.

55. Karanicolas J, Brooks CL. Improved Go-like models demonstrate the robustness of protein folding mechanisms towards non-native interactions. *J Mol Biol* 2003, 334:309–325.
56. Whitford PC, Noel JK, Gosavi S, Schug A, Sanbonmatsu KY, Onuchic JN. An all-atom structure-based potential for proteins: bridging minimal models with all-atom empirical forcefields. *Proteins* 2009, 75:430–441.
57. Zhang YH, Feller SE, Brooks BR, Pastor RW. Computer-simulation of liquid/liquid interfaces, I: theory and application to octane/water. *J Chem Phys* 1995, 103:10252–10266.
58. Takahashi D. An implementation of parallel 3-D FFT with 2-D decomposition on a massively parallel cluster of multi-core processors. In: Wyrzykowski R, Dongarra J, Karczewski K, Wasniewski J, eds. *PPAM 2009*. LNCS, vol. 6067. Berlin Heidelberg: Springer; 2010, 606–614.
59. Neria E, Fischer S, Karplus M. Simulation of activation free energies in molecular systems. *J Chem Phys* 1996, 105:1902–1921.
60. MacKerell AD Jr, Bashford D, Bellott M, Dunbrack RL Jr, Evanseck JD, Field MJ, Fischer S, Gao J, Guo H, Ha S, et al. All-atom empirical potential for molecular modeling and dynamics studies of proteins. *J Phys Chem B* 1998, 102:3586–3616.
61. MacKerell AD Jr, Feig M, Brooks CL III. Extending the treatment of backbone energetics in protein force fields: Limitations of gas-phase quantum mechanics in reproducing protein conformational distributions in molecular dynamics simulations. *J Comput Chem* 2004, 25:1400–1415.
62. MacKerell AD, Banavali N, Foloppe N. Development and current status of the CHARMM force field for nucleic acids. *Biopolymers* 2001, 56:257–265.
63. Klauda JB, Venable RM, Freites JA, O'Connor JW, Tobias DJ, Mondragon-Ramirez C, Vorobyov I, MacKerell AD Jr, Pastor RW. Update of the CHARMM all-atom additive force field for lipids: validation on six lipid types. *J Phys Chem B* 2010, 114:7830–7843.
64. Best RB, Zhu X, Shim J, Lopes PEM, Mittal J, Feig M, MacKerell AD. Optimization of the additive CHARMM all-atom protein force field targeting improved sampling of the backbone phi, psi and side-chain chi(1) and chi(2) dihedral angles. *J Chem Theory Comput* 2012, 8:3257–3273.
65. Raman EP, Guvench O, MacKerell AD. CHARMM additive all-atom force field for glycosidic linkages in carbohydrates involving furanoses. *J Phys Chem B* 2010, 114:12981–12994.
66. Feig M, Karanicolas J, Brooks CL III. MMTSB tool set: enhanced sampling and multiscale modeling methods for applications in structural biology. *J Mol Graph Model* 2004, 22:377–395.
67. Available at: <http://mmtsb.org/webservices/gomodel.html>. (Accessed March 9, 2015).
68. Kitao A, Hirata F, Go N. The effects of solvent on the conformation and the collective motions of protein—normal mode analysis and molecular-dynamics simulations of melittin in water and in vacuum. *Chem Phys* 1991, 158: 447–472.
69. Lindahl E, Hess B, van der Spoel D. GROMACS 3.0: a package for molecular simulation and trajectory analysis. *J Mol Model* 2001, 7:306–317.
70. Nilsson L. Efficient table lookup without inverse square roots for calculation of pair wise atomic interactions in classical simulations. *J Comput Chem* 2009, 30:1490–1498.
71. Available at: <http://www.aics.riken.jp>. (Accessed March 9, 2015).
72. Plimpton S. Fast parallel algorithms for short-range molecular-dynamics. *J Comput Phys* 1995, 117:1–19.
73. Crowley MF, Darden TA, Cheatham TE, Deerfield DW. Adventures in improving the scaling and accuracy of a parallel molecular dynamics program. *J Supercomput* 1997, 11:255–278.
74. Bowers KJ, Dror RO, Shaw DE. The midpoint method for parallelization of particle simulations. *J Chem Phys* 2006, 124:184109.
75. Shaw DE. A fast, scalable method for the parallel evaluation of distance-limited pairwise particle interactions. *J Comput Chem* 2005, 26:1803.
76. Hansmann UHE, Okamoto Y. Generalized-ensemble Monte Carlo method for systems with rough energy landscape. *Phys Rev E* 1997, 56:2228–2233.
77. Martyna GJ, Tobias DJ, Klein ML. Constant pressure molecular dynamics algorithms. *J Chem Phys* 1994, 101:4177–4189.
78. Mori Y, Okamoto Y. Replica-exchange molecular dynamics simulations for various constant temperature algorithms. *J Phys Soc Jpn* 2010, 79:074001.
79. Mori Y, Okamoto Y. Generalized-ensemble algorithms for the isobaric-isothermal ensemble. *J Phys Soc Jpn* 2010, 79:074003.
80. Paschek D, Garcia AE. Reversible temperature and pressure denaturation of a protein fragment: a replica exchange molecular dynamics simulation study. *Phys Rev Lett* 2004, 93:238105.
81. Liu P, Kim B, Friesner RA, Berne BJ. Replica exchange with solute tempering: a method for sampling biological systems in explicit water. *Proc Natl Acad Sci USA* 2005, 102:13749–13754.
82. Itoh SG, Okumura H, Okamoto Y. Replica-exchange method in van der Waals radius space: overcoming steric restrictions for biomolecules. *J Chem Phys* 2010, 132:134105.

83. Itoh SG, Okumura H. Coulomb replica-exchange method: handling electrostatic attractive and repulsive forces for biomolecules. *J Comput Chem* 2013, 34:622–639.
84. Okabe T, Kawata M, Okamoto Y, Mikami M. Replica-exchange Monte Carlo method for the isobaric-isothermal ensemble. *Chem Phys Lett* 2001, 335:435–439.
85. Humphrey W, Dalke A, Schulten K. VMD: visual molecular dynamics. *J Mol Graph Model* 1996, 14:33–38.
86. Hynninen AP, Crowley MF. New faster CHARMM molecular dynamics engine. *J Comput Chem* 2014, 35:406–413.
87. Available at: <http://www.ks.uiuc.edu/Research/namd>. (Accessed March 9, 2015).
88. Ryckaert JP, Ciccotti G, Berendsen HJC. Numerical integration of cartesian equations of motion of a system with constraints: molecular-dynamics of n-alkanes. *J Comput Phys* 1977, 23:327–341.
89. Miyamoto S, Kollman PA. SETTLE: an analytical version of the shake and rattle algorithm for rigid water models. *J Comput Chem* 1992, 13:952–962.
90. Tuckerman M, Berne BJ, Martyna GJ. Reversible multiple time scale molecular dynamics. *J Chem Phys* 1992, 97:1990–2001.
91. Fraser CM, Gocayne JD, White O, Adams MD, Clayton RA, Fleischmann RD, Bult CJ, Kerlavage AR, Sutton G, Kelley JM, et al. The minimal gene complement of *Mycoplasma genitalium*. *Science* 1995, 270:397–403.
92. Feig M, Harada R, Mori T, Yu I, Takahashi K, Sugita Y. Complete atomistic model of a bacterial cytoplasm for integrating physics, biochemistry, and systems biology. *J Mol Graph Model* 2015, 58:1–9.
93. Vanommeslaeghe K, Raman EP, MacKerell AD. Automation of the CHARMM general force field (CGenFF) II: assignment of bonded parameters and partial atomic charges. *J Chem Info Model* 2012, 52:3155–3168.
94. Mukherjee S, Maxfield FR. Membrane domains. *Annu Rev Cell Dev Biol* 2004, 20:839–866.
95. Galla HJ, Hartmann W, Theilen U, Sackmann E. 2-Dimensional passive random-walk in lipid bilayers and fluid pathways in biomembranes. *J Membr Biol* 1979, 48:215–236.
96. Li L, Cheng JX. Coexisting stripe- and patch-shaped domains in giant unilamellar vesicles. *Biochemistry* 2006, 45:11819–11826.
97. Akimov SA, Kuzmin PI, Zimmerberg J, Cohen FS. Lateral tension increases the line tension between two domains in a lipid bilayer membrane. *Phys Rev E* 2007, 75:011919.
98. Ayuyan AG, Cohen FS. Raft composition at physiological temperature and pH in the absence of detergents. *Biophys J* 2008, 94:2654–2666.
99. Hamada T, Kishimoto Y, Nagasaki T, Takagi M. Lateral phase separation in tense membranes. *Soft Matter* 2011, 7:9061–9068.
100. Uline MJ, Schick M, Szleifer I. Phase behavior of lipid bilayers under tension. *Biophys J* 2012, 102:517–522.
101. Portet T, Gordon SE, Keller SL. Increasing membrane tension decreases miscibility temperatures: an experimental demonstration via micropipette aspiration. *Biophys J* 2012, 103:L35–L37.
102. Aydin F, Ludford P, Dutt M. Phase segregation in bio-inspired multi-component vesicles encompassing double tail phospholipid species. *Soft Matter* 2014, 10:6096–6108.
103. Chen D, Santore MM. Large effect of membrane tension on the fluid–solid phase transitions of two-component phosphatidylcholine vesicles. *Proc Natl Acad Sci USA* 2014, 111:179–184.

FURTHER READING

GENESIS's website is available at: <http://www.riken.jp/TMS2012/cbp/en/research/software/genesis/index.html>. A user manual for instruction of installation, usage, and tutorials is provided here. GENESIS is distributed under the GNU General Public License version 2 (<http://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html>; accessed March 9, 2015). Further information (bug reports, release updates, lectures, and workshops) is announced from the GENESIS developer group.