Progress in Scaling Biomolecular Simulations to Petaflop Scale Platforms

Blake G. Fitch[1], Aleksandr Rayshubskiy[1], Maria Eleftheriou[1], T.J. Christopher Ward[2], Mark Giampapa[1], Michael C. Pitman[1], and Robert S. Germain[1]

[1] IBM Thomas J. Watson Research Center, 1101 Kitchawan Road/Route 134, Yorktown Heights, NY 10598, USA
[2] IBM Hursley Park, Hursley, Hursley SO212JN, United Kingdom

Abstract. This paper describes some of the issues involved with scaling biomolecular simulations onto massively parallel machines drawing on the Blue Matter application team's experiences with Blue Gene/L. Our experiences in scaling biomolecular simulation to one atom/node on BG/L should be relevant to scaling biomolecular simulations onto larger peta-scale platforms because the path to increased performance is through the exploitation of increased concurrency so that even larger systems will have to operate in the extreme strong scaling regime. Petascale platforms also present challenges with regard to the correctness of biomolecular simulations since longer time-scale simulations are more likely to encounter significant energy drift. Total energy drift data for a microsecond-scale simulation is presented along with the measured scalability of various components of a molecular dynamics time-step.

1 Introduction

IBM's Blue Gene project was announced in December 1999 with the twin goals of advancing the state of the art in all aspects of computer systems while building a petaflop-scale machine and of using the computational power enabled by this work to explore important issues in the life sciences. This paper describes some of the challenges and issues encountered by the Blue Gene application and science team in the course of creating a molecular simulation environment to both support our scientific goals and to facilitate the exploration of parallel algorithms and programming models suitable for massively parallel machines. The largest installation of the first member of the Blue Gene family, Blue Gene/L[13], is a 65,536 node system at Lawrence Livermore National Laboratory with a theoretical peak performance of 360 TFlop/second. Our application development efforts and simulation science within the Blue Gene project target the 20,480 node, 112 TFlop/s peak performance Blue Gene/L installation at the IBM Thomas J. Watson Research Center (BGW) which is currently the largest unclassified supercomputing facility in the world.

Of all the subfields of computational biology, molecular simulation is almost certainly the most mature in its ability to exploit high performance computing. Most of the biology-related work on the Blue Gene/L facility at Watson (BGW) has thus far been in that area, with projects targeting studies of protein folding mechanisms[5] and structural and dynamical studies of membrane proteins[18,16]. All of these projects share

W. Lehner et al. (Eds.): Euro-Par 2006 Workshops, LNCS 4375, pp. 279–288, 2007.

a requirement for very long time-scale simulations (microseconds) of modestly sized molecular systems (10,000-100,000 atoms). The need for long time-scale simulations drives requirements for both (strong) scalability and correctness that will be discussed below.

The original target architecture for the Blue Gene project[1] had characteristics (very small amount of memory per node, millions of processing elements) that drove a specific set of design goals for the Blue Matter application framework that we have developed as part of the Blue Gene project[8]. The design goals included:

- Running only the computationally intensive molecular dynamics core on the massively parallel Blue Gene platform to reduce the memory footprint of the code.
- Leveraging existing applications as much as possible for problem set-up and other non-performance-critical functionality.
- Separating the complexity of domain-specific aspects of molecular dynamics from the complexity of the parallel communications required. The goal was to allow exploration of parallel decompositions without requiring the involvement of the domain experts.

2 Experiences with Blue Gene/L

We believe that our experiences in developing the Blue Matter simulation code and in running simulations on BGW are relevant to discussions about biomolecular simulations on future peta-scale systems since the BGW facility already has a peak capability of over 0.1 PFlop/s. BGW is typically operated in partitioned fashion where most of the partitions comprise 4096 or 8192 nodes. The allocation and usage patterns of the BGW facility reflect the usual tradeoffs between supporting a range of projects, carrying out the ensembles of simulations required for scientific validity, maximizing overall throughput, and the drive to reduce the total time to solution for a single researcher or simulation. Using this resource, we have been able to run a number of large scale simulation experiments including

- 26 separate 100 nanosecond simulations of Rhodopsin in a membrane environment (44K atoms)[16].
- several microsecond-scale simulations of the same membrane protein system including a pair of simulations totaling 3.5 microseconds.
- several 700 nanosecond simulations of Lysozyme (41K atoms)[5].

and additional long time-scale simulations of a fast folding Lambda Repressor protein are currently underway. Although Blue Matter continues to speed up through 16,384 nodes on the systems being studied[11,10,9], these microsecond scale simulations typically use 4096 node partitions since this currently represents the best tradeoff between throughput and total time to solution. Large Replica Exchange[22] simulations running as a single MPI job on up to 8192 nodes[6] have also been run to obtain temperature dependent thermodynamic information about protein systems.

While the I/O bandwidth requirements of molecular dynamics are quite modest since the entire state of the system is represented by the positions and velocities of the particles in the system, the aggregate storage requirement is potentially quite large. Archival storage for the molecular simulation work is provided by a 500 TB capacity tape library which backs approximately 8 TB of disk storage being managed with Tivoli^{TM} Space Manager (hierarchical storage management).

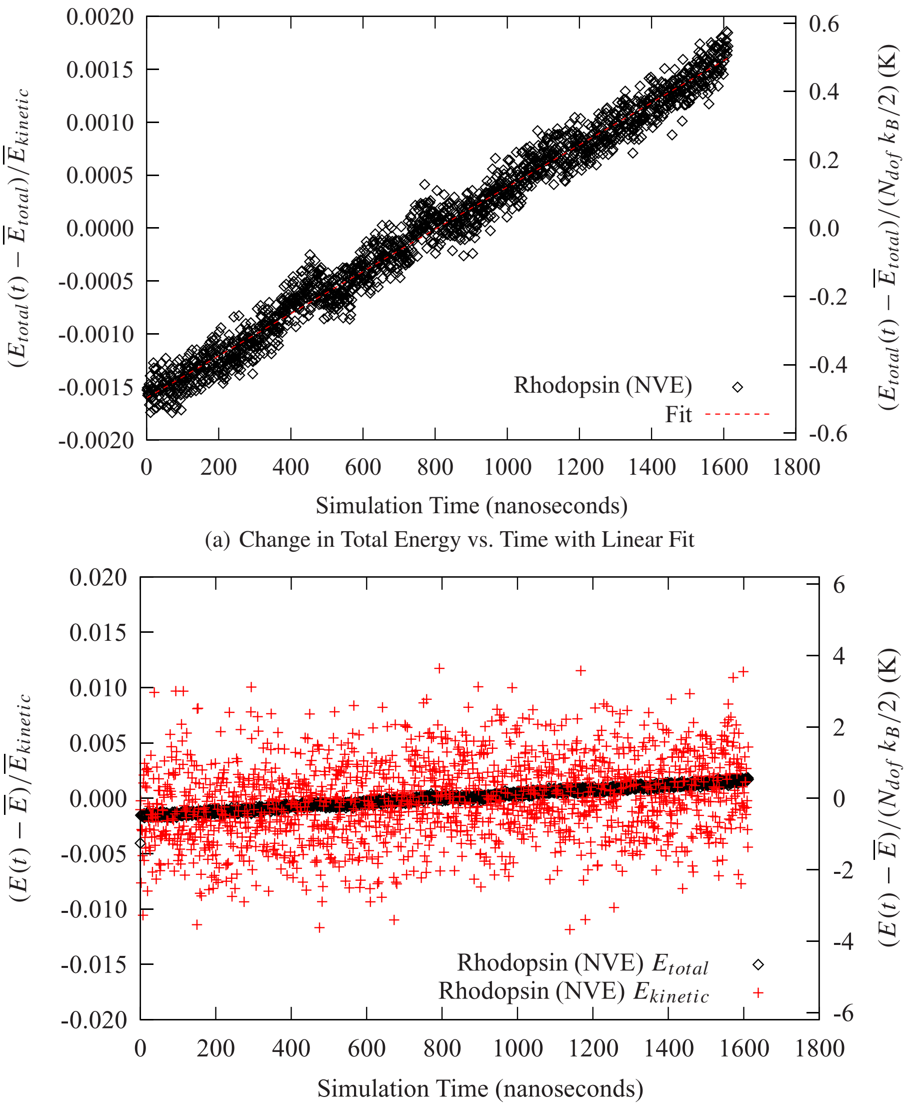
3 Peta-scale Challenges

3.1 Molecular Simulation Validity

As the target time-scale for typical molecular simulations increases from tens of nanoseconds to microseconds or more, the stringency of the requirements on simulations will also increase. In particular, the permissible rate of total energy drift in constant energy, volume, and particle number (NVE) simulations will have to decrease as the length of the NVE simulations increases. An increase in total energy of the system will cause a rise in the instantaneous temperature of the system (defined by the kinetic energy) of the same order. It is useful to measure the energy drift relative to the average kinetic energy in the system (and actually to do so in units of temperature) to make the scale of the effect clear. For example, an energy drift of 7×10^{-2} K/nanosecond results in a an increase in total energy equivalent to 0.7 K over a 10 nanosecond simulation. This is quite small in comparison with biological temperatures on the order of 310 K, but the same energy drift in a 1 microsecond simulation would result in an increase of 70 K in the total energy which is no longer negligible.

One of the principal rationales for believing in the relevance of long term simulations is that for a symplectic integrator such as velocity Verlet[23], used to numerically integrate Hamiltonian systems, there exists a "modified" Hamiltonian whose exact (continuous time) dynamics at integer multiples of the numerical integration time-step coincides with the discrete dynamics generated by the symplectic integrator[17,2,24,19]. This modified Hamiltonian may be "close" to the original in the sense that it can be expressed as a formal expansion in powers of the time-step size about the original Hamiltonian. The existence of this modified Hamiltonian means that the trajectory computed by the numerical integrator should exactly conserve the total energy as computed by the modified Hamiltonian (up to numerical roundoff) and hence should approximately conserve the energy as computed by the original Hamiltonian. The popularity of various forms of Verlet integrators for molecular dynamics simulation is largely due to their simplicity and long term energy stability which stems from the symplectic property that these integrators possess[12].

In general, a computational scientist will want to use the largest time-step size possible consistent with "correctness" in order to maximize throughput. Other performance-critical simulation parameters affecting simulation accuracy and stability include the FFT mesh spacing for Particle-Particle-Particle-Mesh (P3ME) methods[3] and the force-splitting scheme and time-step ratios chosen for symplectic multiple time-stepping methods[21,25,26]. Determining the optimal parameters for simulations enabled by multi-teraflop and larger machines that involve billions or tens of billions of time-steps provides a considerable challenge. Figure 1 shows a plot of the change in the total energy in a simulation of a 43,222 atom system containing Rhodopsin running with a velocity Verlet integrator using a 2 femtosecond time-step where all heavy-atom to hydrogen bonds are constrained (eliminating the highest frequency vibrations from the system).



(a) Change in Total Energy vs. Time with Linear Fit

(b) Changes in Total and Kinetic Energies vs. Time

Fig. 1. Energy drift of NVE molecular dynamics simulation of Rhodopsin in a solvated membrane environment over a 1.6 microsecond run using a 2 femtosecond time-step

The energy drift measured by a linear fit to the data is about 6×10^{-4} K/ns where the left-hand axis shows the energy change as a fraction of the average kinetic energy and the right-hand axis expresses the change in energy as the change in instantaneous temperature that would result if all of the change were in the kinetic energy. For the parameters used in this production simulation, the total change in energy over 1.6 microseconds was slightly larger than 1 K. This is smaller than the fluctuations observed in the kinetic energy during the simulation as shown in Figure 1b and uncertainty of the temperature in the experimental data that we compare with. This time-step size was chosen based on experiences with shorter (10-100ns) simulations, but it is entirely possible that those estimates could have been too low. It should also be noted that the execution time required for a time-step is essentially independent of the choice of integration time-step size while the energy drift is a very strong function of time-step size. Therefore, longer simulations could be carried out with acceptably small energy drift simply by reducing the integration time-step size somewhat and our benchmarking data for the amount of wall clock time required per time-step would still be valid.

Using the normal system Hamiltonian makes it difficult to estimate the long term energy drift without very long simulation runs because of the short term energy fluctuations observed when using a discrete time integrator. Because of the computational expense involved, it has been impractical to carry out a systematic exploration of the tradeoffs between parameter choices such as time-step size and magnitude of the energy drift. In principle, such a study might have to be carried out for each new molecular system. In practice, a choice of parameters is made based on experience with shorter simulations, the drift is monitored as the simulation progresses, and the simulation would have to be rerun with a less aggressive choice of parameters if excessive energy drift were observed. Recently there have been results reported on the numerical estimation of the modified Hamiltonian from the simulation data[7] and this may allow more extensive explorations of the parameter space affecting tradeoffs between simulation quality and computational throughput without prohibitively large expenditures of computational time.

3.2 Performance and Scalability

It is likely that future peta-scale architectures will achieve their performance through massive concurrency (large numbers of CPUs per chip, massive parallelism). Given that this is the case, the application challenge for biomolecular simulations that require strong scaling will be considerable. Within Blue Matter, we have had to be very careful to root out any non-scalable operations from our implementation. As the scale of hardware available to us grew from a single 512-node prototype to the current 20 rack system we repeatedly went through cycles of identifying previously insignificant non-scalable operations that had to be eliminated.

Our current algorithms as implemented on Blue Gene/L can execute a time-step in fewer than 600,000 processor clock cycles (at 700MHz), including the processing associated with the global data dependency necessitated by the FFTs in the P3ME module. We have found that the velocity Verlet integrator which requires the P3ME operation to be carried out on every time-step enables us to run with very small amounts of energy drift in NVE simulations. If no significant increases in processor clock speeds are anticipated, then each order of magnitude decrease in time to solution will require each time-step to execute in a correspondingly smaller number of cycles. Since our scalability is now limited by the execution speed of the FFTs required for the P3ME method as shown in Figure 2, it is likely that investigations of alternative methods for treatment of the long range electrostatics and/or coarse-graining methods will be required to realize additional improvements to the current strong scaling results.

While the BG/L architecture is a relatively “pure” message-passing machine with two identical processing elements per node, each of which can participate in either communication-related activities or computation, there are other ways to deploy additional processing elements. For example, the use of additional specialized processors for DMA operations or communications could enable more overlap of communications and computation, but it isn’t clear that this would increase the limits of scalability where data dependencies prevent further computation until communication operations complete.

4 Algorithmic Explorations

One way to place bounds on the potential scalability of an algorithm is to determine the amount of concurrency available in principle for that algorithm. As a concrete example, Table 1 enumerates the concurrency available in various portions of the molecular dynamics time-step iteration for a 43,222 atom simulation of Rhodopsin using the Particle-Particle-Particle-Mesh Ewald (P3ME) technique. The P3ME technique requires the evaluation of at least two three-dimensional FFTs on each time-step and development of a highly scalable distributed memory 3D-FFT[4] has been one of the key enablers of Blue Matter’s current scalability. As shown in Figure 2, it appears that the three-dimensional FFT is the limiting factor of performance in the extreme strong scaling limit.

The other major component of the computational load in a typical molecular dynamics simulation comes from the finite-ranged pair interactions between particles. We have explored several different algorithms for parallelizing these operations, from simple replicated data decompositions[8,15], to a geometrically-based interaction decomposition with a minimal communication radius[11,14], and most recently, a set-based

Table 1. Degree of concurrency in computational modules within a single molecular dynamics time-step for a 43,222 atom membrane/protein system (Rhodopsin) using a 128 × 128 × 128 mesh for P3ME. The system parameters were those used in production simulations[18,16]. The last column is the concurrency possible for that computational module based on the number of independent calculations required (assuming a “reasonable” level of granularity). The number of real-space pair interactions to be computed will actually fluctuate somewhat during the course of the simulation because of particle diffusion.

Stage	**Major Computational Kernel**	**Independent Computation Count**
Real-space Non-bond (9/1 Å cutoff/switch)	Pairwise forces (L-J and Ewald real-space)	9,113,514
Bonded	bond stretches, angle bends, torsions, Urey-Bradley	126,730
P3ME Meshing/Un-meshing	4 × 4 × 4 stencil	43,222
P3ME Convolution	3D Fast Fourier Transform (FFT)	16,384
Propagation of Dynamics	Verlet integration	43,222

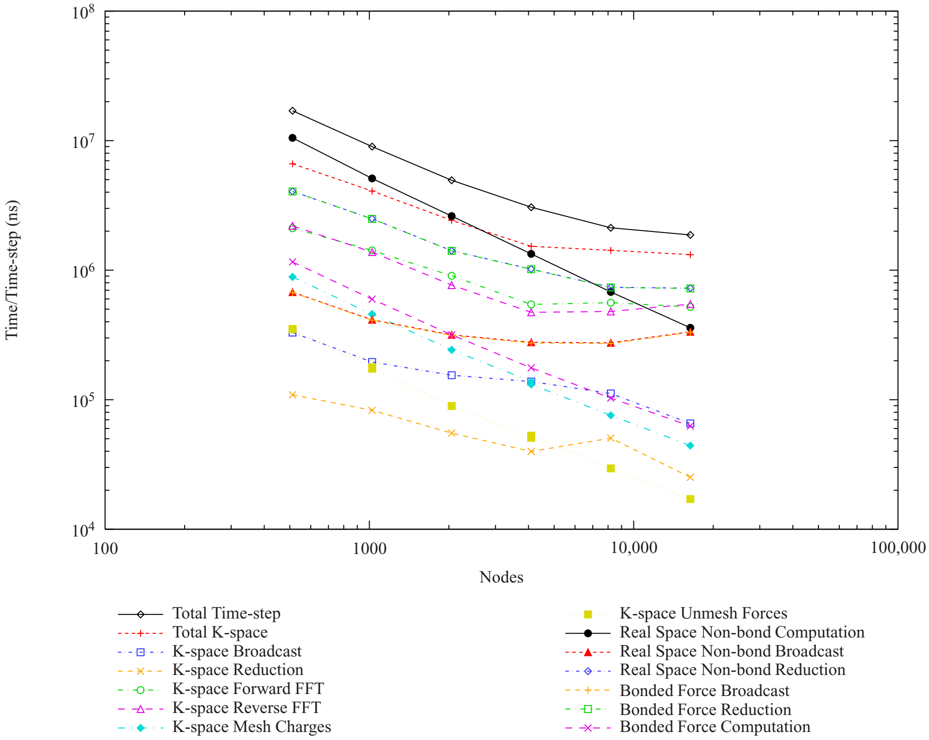


Fig. 2. This is a plot of the scalability of various components of the molecular dynamics time-step as a function of node count. The system is the same Rhodopsin membrane/protein system described in Table 1. There are data dependencies between some of these components and since we schedule modules on both CPUs of the BG/L node, some of the components are executing concurrently.

optimization technique that uses a geometrically derived heuristic as a starting point[9]. The most recent performance results demonstrate time-step execution times below one millisecond for a β-hairpin system and continued speedups through approximately one atom per node[9].

5 Conclusions

Experiences with scaling the Blue Matter biomolecular simulation application to run effectively on the 112 TFlop/s BGW system should be relevant to any efforts to run such codes on future petaflop-scale platforms because the design philosophy of Blue Gene/L required the kind of massive parallelism that is likely to be needed for such platforms. As the development of novel algorithmic techniques was required to realize improved time-to-solution for biomolecular simulations on Blue Gene/L, it is likely that significant additional innovation will be needed in order to continue to increase the time scales accessible via simulation. These innovations will almost certainly be related to the parallelization of the long range electrostatic interactions and may involve the adoption of alternative algorithms for the computation of those interactions such as multi-grid[20].

Even without additional algorithmic improvements, it is likely that increasing the system size studied (weak scaling) will enable effective use of peta-scale platforms to extend the accessible time-scales for those systems into the microsecond regime that Blue Gene/L has opened up for smaller systems ($< 100,000$ atoms). Also, the availability of peta-scale platforms will enable studies involving larger ensembles of long trajectories that can give improved sampling and allow the generation of statistical error estimates[16].

Acknowledgements. We would like to thank the members of the Blue Gene systems team including P. Heidelberger, A. Gara, M. Blumrich, J. Sexton as well as others who have made use of and contributions to the Blue Matter code over the course of its development, particularly Y. Zhestkov, Y. Sham, F. Suits, W. Swope, J. Pitera, A. Grossfield, and R. Zhou.

References

1. F. Allen, G. Almasi, W. Andreoni, D. Beece, B. J. Berne, A. Bright, J. Brunheroto, C. Cascaval, J. Castanos, P. Coteus, P. Crumley, A. Curioni, M. Denneau, W. Donath, M. Eleftheriou, B. Fitch, B. Fleischer, C. J. Georgiou, R. Germain, M. Giampapa, D. Gresh, M. Gupta, R. Haring, H. Ho, P. Hochschild, S. Hummel, T. Jonas, D. Lieber, G. Martyna, K. Maturu, J. Moreira, D. Newns, M. Newton, R. Philhower, T. Picunko, J. Pitera, M. Pitman, R. Rand, A. Royyuru, V. Salapura, A. Sanomiya, R. Shah, Y. Sham, S. Singh, M. Snir, F. Suits, R. Swetz, W. C. Swope, N. Vishnumurthy, T. J. C. Ward, H. Warren, and R. Zhou. Blue Gene: a vision for protein science using a petaflop supercomputer. *IBM Journal of Research and Development*, 40(2):310–327, 2001.
2. Giancarlo Benettin and Antonio Giorgilli. On the hamiltonian interpolation of near-to-the-identity symplectic mappings with application to symplectic integration algorithms. *J. Statist. Phys.*, 74:1117–43, 1994.
3. Markus Deserno and Christian Holm. How to mesh up Ewald sums. ii. an accurate error estimate for the particle-particle-particle-mesh algorithm. *J. Chem. Phys.*, 109(18): 7694–7701, 1998.
4. M. Eleftheriou, B. Fitch, A. Rayshubskiy, T.J.C. Ward, and R.S. Germain. Performance measurements of the 3d FFT on the Blue Gene/L supercomputer. In J.C. Cunha and P.D. Medeiros, editors, *Euro-Par 2005 Parallel Processing: 11th International Euro-Par Conference, Lisbon, Portugal, August 30-September2, 2005*, volume 3648 of *Lecture Notes in Computer Science*, pages 795–803. Springer-Verlag, 2005.
5. M Eleftheriou, R Germain, A Royyuru, and R Zhou. Thermal denaturing of mutant lysozyme with both oplsaa and charmm force fields. to appear in J. Am. Chem. Soc., 2006.
6. M. Eleftheriou, A. Rayshubskiy, J. W. Pitera, B. G. Fitch, R. Zhou, and R. S. Germain. Parallel implementation of the replica exchange molecular dynamics algorithm on Blue Gene/L. In *Fifth IEEE International Workshop on High Performance Computational Biology*, April 2006.
7. Robert D. Engle, Robert D. Skeel, and Matthew Drees. Monitoring energy drift with shadow hamiltonians. *Journal of Computational Physics*, 206(2):432–452, 2005.

8. B.G. Fitch, R.S. Germain, M. Mendell, J. Pitera, M. Pitman, A. Rayshubskiy, Y. Sham, F. Suits, W. Swope, T.J.C. Ward, Y. Zhestkov, and R. Zhou. Blue Matter, an application framework for molecular simulation on Blue Gene. *Journal of Parallel and Distributed Computing*, 63:759–773, 2003.
9. Blake G. Fitch, Aleksandr Rayshubskiy, Maria Eleftheriou, T.J. Christopher Ward, Mark Giampapa, Michael C. Pitman, and Robert S. Germain. Blue matter: Approaching the limits of concurrency for molecular dynamics. Research Report RC23956, IBM Research Division, April 2006. To appear in the Proceedings of the 2006 ACM/IEEE conference on Supercomputing.
10. Blake G. Fitch, Aleksandr Rayshubskiy, Maria Eleftheriou, T.J. Christopher Ward, Mark Giampapa, Yuri Zhestkov, Michael C. Pitman, Frank Suits, Alan Grossfield, Jed Pitera, William Swope, Ruhong Zhou, Scott Feller, and Robert S. Germain. Blue Matter: Strong scaling of molecular dynamics on Blue Gene/L. In V. Alexandrov, D. van Albada, P. Sloot, and J. Dongarra, editors, *International Conference on Computational Science (ICCS 2006)*, volume 3992 of *LNCS*, pages 846–854. Springer-Verlag, 2006.
11. Blake G. Fitch, Aleksandr Rayshubskiy, Maria Eleftheriou, T.J. Christopher Ward, Mark Giampapa, Yuri Zhestkov, Michael C. Pitman, Frank Suits, Alan Grossfield, Jed Pitera, William Swope, Ruhong Zhou, Robert S. Germain, and Scott Feller. Blue matter: Strong scaling of molecular dynamics on Blue Gene/L. Research Report RC23688, IBM Research Division, August 2005.
12. D. Frenkel and B. Smit. *Understanding Molecular Simulation*. Academic Press, San Diego, CA, 1996.
13. A. Gara et al. Overview of the Blue Gene/L system architecture. *IBM Journal of Research and Development*, 49(2/3):195–212, 2005.
14. Robert S. Germain, Blake Fitch, Aleksandr Rayshubskiy, Maria Eleftheriou, Michael C. Pitman, Frank Suits, Mark Giampapa, and T.J. Christopher T.J. Christopher Ward. Blue Matter on Blue Gene/L: massively parallel computation for biomolecular simulation. In *CODES+ISSS '05: Proceedings of the 3rd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*, pages 207–212, New York, NY, USA, 2005. ACM Press.
15. R.S. Germain, Y. Zhestkov, M. Eleftheriou, A. Rayshubskiy, F. Suits, T.J.C. Ward, and B.G. Fitch. Early performance data on the Blue Matter molecular simulation framework. *IBM Journal of Research and Development*, 49(2/3):447–456, 2005.
16. Alan Grossfield, Scott E. Feller, and Michael C. Pitman. A role for direct interactions in the modulation of rhodopsin by omega-3 polyunsaturated lipids. *PNAS*, 103(13):4888–4893, 2006.
17. Benedict Leimkuhler and Sebastian Reich. *Simulating Hamiltonian Dynamics*, volume 14 of *Cambridge Monographs in Applied and Computational Mathematics*. Cambridge University Press, 2004.
18. Michael C. Pitman, Alan Grossfield, Frank Suits, and Scott E. Feller. Role of cholesterol and polyunsaturated chains in lipid-protein interactions: Molecular dynamics simulation of rhodopsin in a realistic membrane environment. *Journal of the American Chemical Society*, 127(13):4576–4577, 2005.
19. Sebastian Reich. Backward error analysis for numerical integrators. *SIAM Journal on Numerical Analysis*, 36(5):1549–1570, 1999.
20. C. Sagui and T. Darden. Multigrid methods for classical molecular dynamics simulations of biomolecules. *Journal of Chemical Physics*, 114(15):6578–6591, April 2001.
21. J.C. Sexton and D.H. Weingarten. Hamiltonian evolution for the hybrid Monte Carlo algorithm. *Nuclear Physics B*, 380:665–677, 1992.

22. Y. Sugita and Y. Okamoto. Replica-exchange molecular dynamics method for protein folding. *Chem. Phys. Lett.*, 314:141–151, 1999.
23. W.C Swope, H.C. Andersen, P.H. Berens, and K.R. Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *Journal of Chemical Physics*, 76:637–649, 1982.
24. Søren Toxvaerd. Hamiltonians for discrete dynamics. *Phys. Rev. E*, 50(3):2271–2274, Sep 1994.
25. M. Tuckerman, B.J. Berne, and G.J. Martyna. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.*, 97(3):1990–2001, August 1992.
26. R. Zhou, E. Harder, H. Xu, and B.J. Berne. Efficient multiple time step method for use with Ewald and particle mesh Ewald for large biomolecular systems. *Journal of Chemical Physics*, 115(5):2348–2358, August 2001.