Grid-based Nonequilibrium Multiple-Time Scale Molecular Dynamics/Brownian Dynamics Simulations of Ligand-Receptor Interactions in Structured Protein Systems

Yaohang Li[1], Michael Mascagni[2], Michael H. Peters[3]
[1,2]*Department of Computer Science*
Florida State University
Tallahassee, FL 32306-4530, USA
[1,2,3]*School of Computational Science and Information Technology*
Florida State University
Tallahassee, FL 32306, USA
[3]*Department of Chemical and Biomedical Engineering*
Florida State University and Florida A&M University
Tallahassee, FL 32310, USA
{yaohanli, mascagni}@cs.fsu.edu, peters@eng.fsu.edu

Abstract

In the hybrid Molecular Dynamics (MD)/Brownian Dynamics (BD) algorithm for simulating the long-time, nonequilibrium dynamics of receptor-ligand interactions, the evaluation of the force autocorrelation function can be computationally costly but fortunately is highly amenable to multimode processing methods. In this paper, taking advantage of the computational grid's large-scale computational resources and the nice characteristics of grid-based Monte Carlo applications, we developed a grid-based receptor-ligand interactions simulation application using the MD/BD algorithm. We expect to provide high-performance and trustworthy computing for analyzing long-time dynamics of proteins and protein-protein interaction to predict and understand cell signaling processes and small molecule drug efficacies. Our preliminary results showed that our grid-based application could provide a faster and more accurate computation for the force autocorrelation function in our MD/BD simulation than previous parallel implementations.

1. Introduction

Prediction of the long-time, nonequilibrium dynamics of receptor-ligand interactions for structured proteins in a host fluid is of critical importance to the understanding of infectious diseases, immunology, the development of "target" drugs, and biological separations. However, such processes take place on time scales on the order of milliseconds to seconds, which prevents the "brute-force" real-time molecular or atomic simulations from determining the absolute ligand binding rates to receptor targets. In a previous study [1], we implemented a hybrid Molecular Dynamics (MD)/Brownian Dynamics (BD) algorithm which utilizes the underlying, disparate time scales involved and overcomes the limitations of brute-force approaches. Single and isolated proteins, protein with charge effects, and D-peptide/HIV capsid protein systems were investigated using the hybrid MD/BD algorithm [1].

Within the hybrid MD/BD algorithm, the calculation of the force autocorrelation function to generate the grand particle friction tensor forms the basis of the most computationally costly part, which requires large amount of CPU cycles. Even on an advanced supercomputer, this computation takes from days to months and thus becomes the performance bottleneck of the MD/BD simulation. Fortunately, this part of the computation uses Monte Carlo methods, which is computationally intensive but naturally parallel. It is very amenable to the emerging grid-computing environment, characterized by "large-scale sharing and cooperation of dynamically distributed resources, such as CPU cycles, communication bandwidth, and data, to constitute a computational environment" [2]. A large-scale computational grid can, in principle, offer a tremendous amount of low-cost computational power, which attracts us to utilize the computational grid for our MD/BD application. On the other hand, our previous studies in grid-based Monte Carlo applications [3, 4] showed that Monte Carlo's statistical nature could be applied to improve the performance and enforce the trustworthiness of grid computing at the application level. This paper will study the development of the grid-based nonequilibrium, multiple-time scale simulation application of ligand-receptor interactions. We take advantage of the services of a computational grid and the characteristics of grid-based Monte Carlo applications to provide high-performance and trustworthy computation for predicting

and understanding the dynamics of structured protein systems.

The remainder of this paper is organized as follows. In Section 2, we review the general hybrid MD/BD algorithm, its implementation, and specifically, the computation of the force autocorrelation functions. We discuss the implementation of a grid-based MD/BD simulation and present our preliminary results in Section 3 and Section 4, respectively. Finally, Section 5 summarizes our conclusions and future research directions.

2. Hybrid Molecular Dynamics (MD) / Brownian Dynamics (BD) Algorithm

2.1 Introduction to Hybrid MD/BD Algorithm

In previous study [5] of the behavior of the many-bodied friction tensor for particles immersed in a rarefied, "free-molecule" gas, a molecular dynamics method was used. It was noted that the molecular dynamics method could be used to study the long-time behavior of Brownian particles by a two-step procedure. This procedure is illustrated as following:

1) For a given particle configuration, the many-body friction tensor is determined from MD through the analysis of the force autocorrelation function. In this step, the particle coordinates are kept fixed according to the fluctuation-dissipation type relation that gives the (time-independent) friction tensor in terms of the force autocorrelation function.
2) The Fokker-Planck (FP) equation, which describes the dynamics of a single structured Brownian particle in a molecular fluid, is solved for discrete times assuming that the friction tensor remains constant over the time step. The particles are advanced to new positions according to the integrated FP equation.

The entire process, MD followed by BD is repeated. MD is only performed at the beginning of each BD time step. This MD/BD algorithm is based on a multiple time scales analysis of the total system Hamiltonian, including all atomic molecular structure information for the system: water, ligand, and receptor. The results allow the study of the long-time dynamics of macromolecules in complex systems where complete molecular details of the macromolecule, surface, and solvent can be incorporated. The theoretical background and a detailed review of the hybrid MD/BD algorithm can be found in [5, 6, 7, 8].

2.2 Hybrid MD/BD Algorithm Implementation

The hybrid MD/BD algorithm was implemented in [9] to study the D-peptide/HIV system. The general computational MD/BD algorithm is shown in Figure 1 [1]. The computational scheme begins by reading a standard PDB file from the protein data bank for both ligand and receptor. This file is then converted to a "topology" file that includes computationally critical information on atomic mass, residue charge, and Lennard-Jones interaction force constants. Next, the ligand and receptor must be hydrated using SOLVATE [10], which is using a Monte Carlo method. For the molecular model of water, the so-called modified Simple Point Charge (SPC) model [11] with long-range electrostatic inter-atomic interactions accounted for by a modified Poisson-Bolzmann reaction field method, which using an acceptance-rejection Monte Carlo approach. The center of mass and body fixed axes along the principal axes of inertia for the ligand are initially computed. This sets the body-fixed coordinates and initial Euler angles, the latter of which give the orientation of the body relative to the space fixed frame. MD is then used to determine the particle grand friction tensor. The grand friction tensor is numerically inverted to obtain the grand diffusion tensor. The grand diffusion tensor is then utilized to perform the BD move on a time step of around 10^{-5} seconds. The macromolecule position and orientation change by only a couple of percent or less over this time period. The new atomic positions are updated based on the BD move and the entire process, viz., MD followed by BD is repeated.

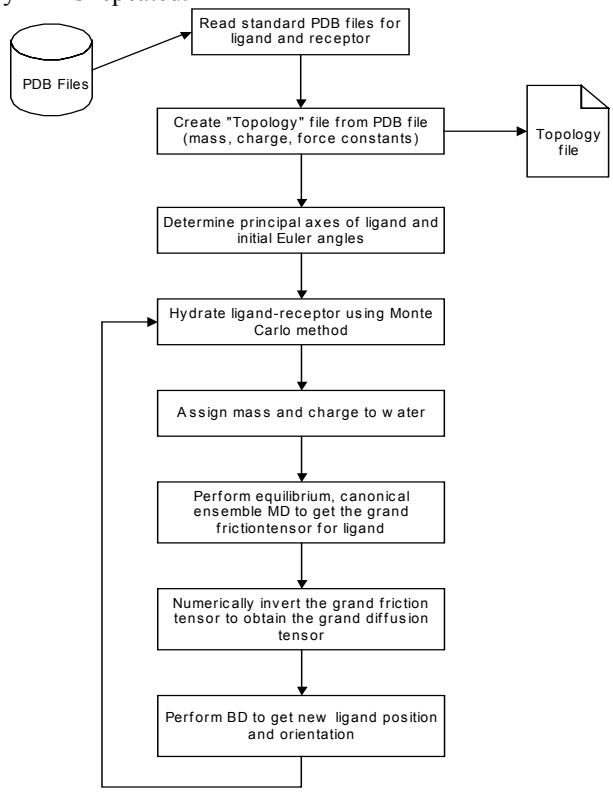

Figure 1: Flowchart of Hybrid MD/BD Algorithm

2.3 The Force Autocorrelation Function

The force autocorrelation function can be obtained by conducting standard canonical, equilibrium molecular dynamics simulations. The particle was considered to be composed of a large number of molecules each interacting with the fluid molecules according to a Lennard-Jones potential [7]. Suppose the Brownian particle is composed of M molecules and the fluid consists of N molecules. The computation of the force autocorrelation function is $O(N^2 + M*N)$ at every time step, which is very computational costly.

A confidence interval in the autocorrelation values, CI is obtained from the Tchebycheff inequality as

$$CI = 1 - \frac{\sigma^2}{n_r \varepsilon^2},$$

where ε is the error in the autocorrelation, σ^2 is the variance, and n_r is the number of repeats (ensembles). The error is proportional to the reciprocal square root of the number of repeats n_r, i.e.,

$$\varepsilon \sim 1 / n_r^{1/2}.$$

Thus with increasing number of repeats, the error in the autocorrelation is reduced. Our results show that at least 20 ensembles are minimally necessary and more would be desirable for more accurate results.

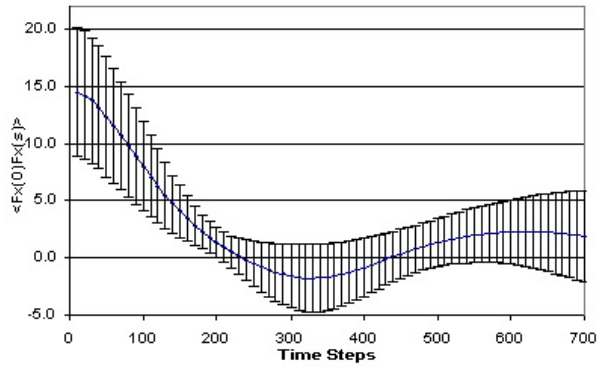

Figure 2: the xx-component of the force autocorrelation function in one standard deviation for a D-peptide

Figure 2 shows the xx-component of the force autocorrelation function for a D-peptide in 1 standard deviation with 40 ensembles. Our experiment of the computation in Figure 2 is implemented using MPI. The computation simulates the D-peptide with 372 molecules in the system with around 10,000 water molecules running in 60,000 time steps. The experiment took almost 8 days on a DEC Alpha DS10 6/466 with 256 DRAM with 4 processors for a single step of MD simulation and more than a month on a serial DEC Alpha DS10. We can expect longer time consumption for a particle with more molecules or system with more host fluid molecules. The high computational cost of evaluating of the force autocorrelation function constrains us to perform more steps of MD/BD simulation to study the behavior of the particles and particle interactions. More importantly, since the friction tensor in BD is the integral of the force autocorrelation, the inaccuracy in MD may mislead the computation of BD. The error can even be propagated in further MD/BD simulation.

Deeper study of the MD part of the algorithm shows that the force autocorrelation function is particularly amenable to multiprocessor systems [1]. In parallel MD simulation, each node can represent one member (3,000 time steps) of the ensemble allowing hundreds and thousands of ensembles to be included. More importantly, once scheduled, each ensemble's computation is based on its own fluid configuration, which is independent with no intercommunication needed. Also, each execution time costs a few hours or less depending on each processor speed. This property of autocorrelation computation motivates us to take advantage of the tremendously large and low-cost computational power in a computational grid for our MD/BD dynamics simulation for structured protein system.

3. Implementation of Grid-based MD/BD Simulation

3.1 Application Overview

To develop a grid-based hybrid MD/BD simulation application, we need to utilize the grid services. First of all, the task split service is used to define the data set and initial conditions for each ensemble computation, e.g., the ligand and receptor configurations, the host fluid configuration, the Lennard-Jones constants, and the parameters for random number stream. Each ensemble computation's data and program are packed into a grid subtask. Secondly, the task schedule service is used to distribute these subtasks to individual computational service providers. During the execution of the subtask, storage service is used to store the checkpointing data, intermediate results, and subtask results. Thirdly, when the partial results are ready, the collection service is responsible of gathering them all and validating each partial result. Finally, based on the computational results of all ensembles, we can assemble the estimation of the force autocorrelation function and estimate the statistical error. After the MD simulation using the grid environment, the BD simulation follows and updates the new atomic position in the particle. The above process can be repeated for next BD moves.

An implicit requirement of the grid-based hybrid MD/BD simulation application is that the underlying

random number streams in each subtask must be independent in a statistical sense to avoid correlation. The SPRNG (Scalable Parallel Random Number Generators) library [12] was designed to use parameterized pseudorandom number generators to provide independent random number streams to parallel processes. Some generators in SPRNG can generate up to $2^{78000}-1$ independent random number streams with sufficient long period and good quality [13], which can meet this requirement.

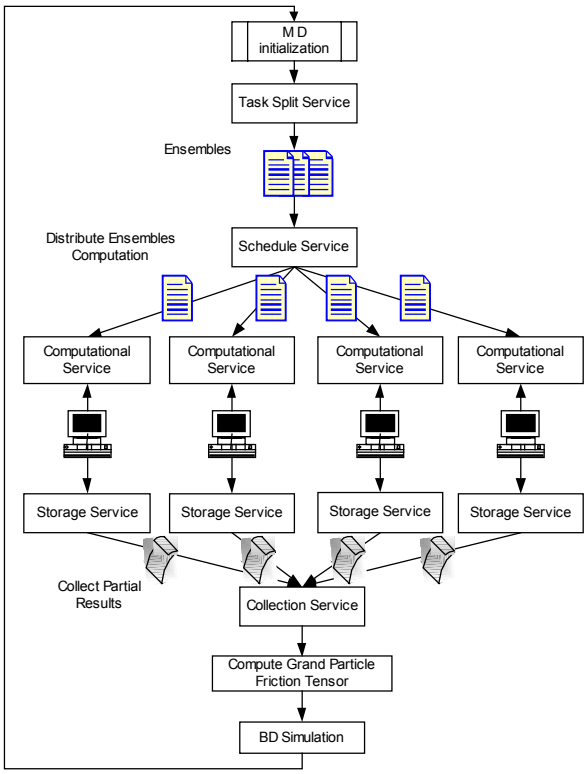

Figure 3: Working Paradigm of Grid-based MD/BD Simulation on Structured Protein System

Figure 3 shows the working paradigm of the grid-based hybrid MD/BD simulation application. Furthermore, the MD part of the computation is a typical grid-based Monte Carlo computation, which exhibits nice characteristics that can be used to improve the application's performance and trustworthiness in the grid [3]. We can take advantage of these properties to optimize the MD computation.

3.2 Subtasks Scheduling

In the grid-computing environment, a node that is assigned an ensemble of the force autocorrelation function might be a high-end supercomputer, or a low-end personal computer, even just an intelligent widget. Also, this node can be a very busy node, or an always-idle node. Therefore, one delayed ensemble computations on a slow node might delay the whole MD simulation. More seriously, one halted ensemble computation may present the MD simulation from completing.

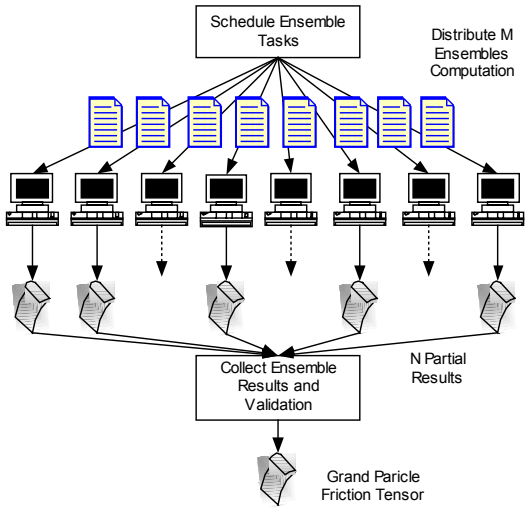

Figure 4: *N-out-of-M* Scheduling Strategy in MD Simulation

Fortunately, carefully studying the statistical nature of the MD simulation using Monte Carlo method, we find that the MD simulation does not care which host fluid configuration (based on the underlying random sample set) is estimated provided that all random samples are independent in a statistical sense. However, the MD simulation cares how many ensembles computation that must obtain to achieve certain accuracy. This enables us to use the *N-out-of-M* scheduling strategy [3] to enhance the application performance. Suppose we need N ensembles. In our MD simulation, we enlarge the ensemble computations from N to M, where $M > N$ and actually schedule M subtasks to the grid. When N partial results are ready, we have enough ensembles computation results to generate the grand particle friction tensor. The *N-out-of-M* strategy can tolerate $M - N$ delayed or halted subtasks in MD simulation. Figure 4 shows the diagram of the *N-out-of-M* scheduling strategy in MD simulation. More analysis of this *N-out-of-M* scheduling strategy and determining the values of M and N can be found in [14].

3.3 Checkpointing in MD Simulation

In the grid-computing environment, a node is probably not dedicated to the computation of ensembles in the MD simulation. It may go down or become inaccessible, causing interruption in the execution of MD simulation. Therefore, checkpointing is necessary to save the previous work for further recovery. Considering the

process-level checkpointing is costly and platform-dependent, we implement the application-level checkpointing in the MD simulation with checkpoint and recover subroutines.

Since at each time step in the MD simulation, the positions of the atoms in the structured proteins remain the same, the only changing data are the configuration of the host fluid, such as the atoms' locations and velocities. Thus, the checkpointing data that the subroutine needs to save are the configuration of the host fluid, the current time step, and the force autocorrelation function values in previous time steps. The checkpointing data are stored into a checkpointing file. Based on the checkpointing data, the recover subroutine can easily restore the interrupted computation. Compared with the process-level checkpointing, this application-level checkpointing is cheap and can be easily migrated to other nodes in the grid to continue the computation.

3.4 Partial Result Validation

In our hybrid MD/BD algorithm, the generation of the particle grand friction tensor depends on the integration of the force autocorrelation function, which is based on all the ensembles computed in the grid. A single erroneous result in an ensemble will lead to an error in later BD simulation. In a computational grid, a node providing computational service is potentially insecure and thus may probably be untrustable. Validation mechanism should be applied to enforce the trustworthiness of the MD simulation performed in the grid.

A partial result validation method for point solution is provided in [3, 4]. This method can be extended and used in our grid-based MD/BD simulation application to validate the force autocorrelation function curve from each ensemble. Based on the force autocorrelation function values at every time step, we calculate its mean, standard deviation, and then the confidence interval. The upper bound endpoints of all these confidence intervals at different time steps construct an upper bound curve and the lower bound endpoints construct a lower bound one. If a force autocorrelation function curve from an ensemble lies in the area between the upper bound and lower bound curves, we consider the partial result of this ensemble computation is trustworthy; otherwise, it is suspicious and we may need to rerun this particular subtask for further verification.

4. Preliminary Results

We performed our experiments on Condor system [15,16]. Figure 5 shows our preliminary results of grid-based MD/BD simulation using GCondor with 40 nodes each carrying the computation of 10 ensembles. The SPRNG [12] library is used to provide parallel random number streams. The D-peptide data in this experiment is the same as the one in Figure 2. Using the *N-out-of-M* subtasks scheduling strategy, this computation took around 8 days when we obtained 400 ensembles computational results of the force autocorrelation function (by scattering the 400 ensembles to more nodes in the grid, we can expect less completion time). We can find that the xx-component of the force autocorrelation function has a much smaller error bound than the one in Figure 2. More specifically, Table 1 shows the statistical error bound estimations of xx-component of the force autocorrelation function with 90% confidence in this experiment. We notice that statistical error decreases while increasing the number of ensembles. The computations of other components of the force autocorrelation function have the similar effects.

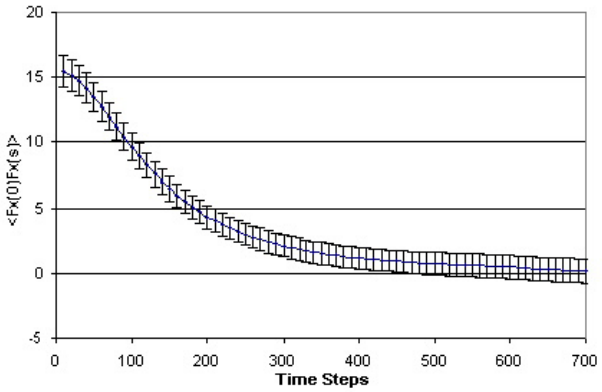

Figure 5: the xx-component of the force autocorrelation function in one standard deviation for a D-peptide on Condor with 400 ensembles

# of Ensembles	Std Deviation	Error Bound
10	4.10	12.97
100	1.39	4.40
200	0.943	2.97
400	0.671	2.11

Table 1: the error of xx-component of the force autocorrelation function with 90% confidence. The error decreases with increasing number of ensembles.

Figure 6 shows the partial result validation mechanism in our grid-based MD/BD simulation. We can find that all of the force autocorrelation function curves obtained from different nodes in the computational grid lie in the area of the upper bound and lower bound curves, which we regard them as trustworthy computations.

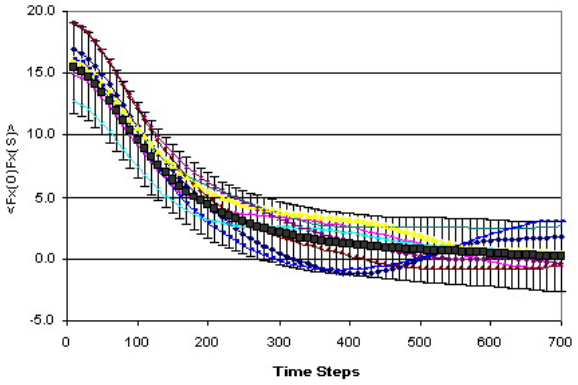

Figure 6: Partial result validation of xx-component of the force autocorrelation function. Each curve lies in the error bounds of 3σ.

5. Conclusions

In this paper, we discussed utilizing the power of the computational grid to implement a grid-based hybrid MD/BD simulation application to study ligand-receptor interactions. The most computationally costly part of the MD simulation, the evaluation of the force autocorrelation function, is computed in the grid-computing environment. Also, we took advantage of the characteristics of MD algorithm that uses Monte Carlo method to improve the task completion time and trustworthiness of the MD/BD simulation. Our preliminary results show a significant performance and accuracy improvement of the simulation results compared with the previous parallel implementation.

The current implementation of the grid-based hybrid MD/BD simulation uses the facilities and services of Condor. In the future, to address the portability and security issues, we plan to adopt the Globus Toolkit [17] in our implementation, which can provide uniform and authorized access to grid resources and security facilities for grid applications. Since we now have a more powerful computational application to study receptor-ligand interactions, we also plan to study more structured protein systems in order to predict and analyze cell signaling processes and small molecule drug efficacies. More aggressively, we expect to develop a "plug-drug" system based on our MD/BD simulation with the purpose of searching for good drug candidates in the long run.

References

[1] Y. Zhang, Y. Li, and M. H. Peters, "Nonequilibrium, Multiple-Time Scale Simulations of Ligand-Receptor Interactions in Structured Protein Systems," submitted to Proteins: Structure, Function, and Genetics, 2002.

[2] I. Foster, C. Kesselman, and S. Tueske, "The Anatomy of the Grid," International Journal of Supercomputer Applications, **15**(3), 2001.

[3] Y. Li and M. Mascagni, "Grid-based Monte Carlo Application," Lecture Notes in Computer Science, **2536**:13-25, GRID2002, Baltimore, 2002.

[4] Y. Li and M. Mascagni, "Analysis of Large-scale Grid-based Monte Carlo Applications," submitted to the special issue of the International Journal of High Performance Computing Applications (IJHPCA), 2003.

[5] M. H. Peters, "Nonequilibrium molecular dynamics simulation of free-molecule gas flows in complex geometries with application to Brownian motion of aggregate aerosols," Physical Review E, **50**(6):4609-4617, 1994.

[6] M. H. Peters, "Fokker-Planck equation and the grand molecular friction tensor for coupled rotational and translational motions of structured Brownian particles near structured surfaces," Journal of Chemical Physics, **110**(1):528-538, 1999.

[7] M. H. Peters, "Fokker-Planck Equation, Molecular Friction, and Molecular Dynamics for Brownian Particle Transport near External Solid Surfaces," Journal of Statistical Physics, **94**:557-586, 1999.

[8] M. H. Peters, "The Smoluchowski diffusion equation for structured macromolecules near structured surfaces," Journal of Chemical Physics, **112**(12):5488-5498, 2000.

[9] Y. Zhang, "Implementation of a Hybrid Molecular-Brownian Dynamics Simulation on a D-Peptide/HIV Protein Macromolecular System," Master's Thesis, Florida State University, 2001.

[10] SOLVATE website, http://www.mpibpc.gwdg.de.

[11] K. Toukau and A. Rahman, "Molecular Dynamics Study of Atomic Motions in Water," Physics Review B, **31**(5):2643-2649, 1985.

[12] M. Mascagni, D. Ceperley, and A. Srinivasan, "SPRNG: A Scalable Library for Pseudorandom Number Generation," ACM Transactions on Mathematical Software, 2000.

[13] SPRNG website, http://sprng.cs.fsu.edu.

[14] Y. Li and M. Mascagni, "Improving Performance via Computational Replication on a Large-Scale Computational Grid," submitted to the IEEE/ACM CCGRID2003, Tokyo, 2003.

[15] M. Litzkow, M. Livny, and M. Mutka, "Condor - A Hunter of Idle Workstations," Proc. of the 8th Intl Conf. of Dist. Comp. Systems, pp. 104-111, 1988.

[16] Condor website, http://www.cs.wisc.edu/condor.

[17] I. Foster and C. Kesselman, "Globus: A Metacomputing Infrastructure Toolkit," International Journal of Supercomputer Applications, **11**(2):115-128, 1997.