JOURNAL OF CHEMICAL INFORMATION AND MODELING

Article

pubs.acs.org/jcim

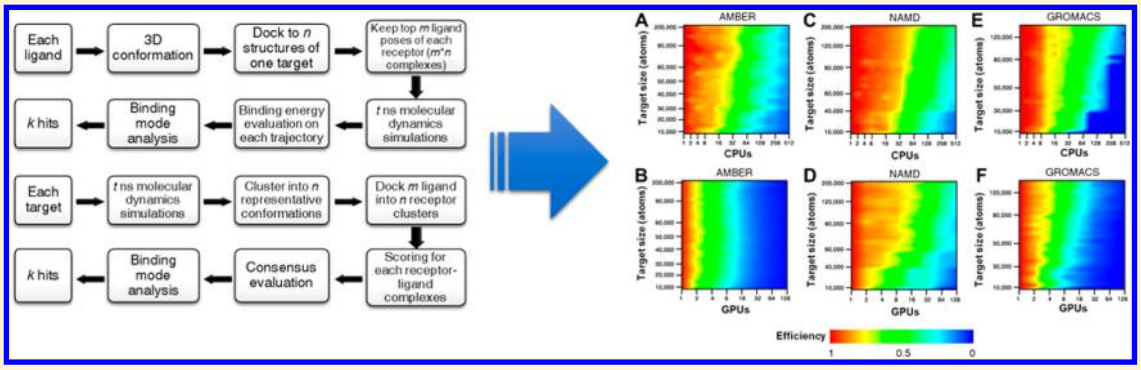
Molecular Dynamics-Based Virtual Screening: Accelerating the Drug Discovery Process by High-Performance Computing

Hu Ge,[†] Yu Wang,[†] Chanjuan Li,[†] Nanhao Chen,[†] Yufang Xie,[†] Mengyan Xu,[†] Yingyan He,[†] Xinchun Gu,[†] Ruibo Wu,[†] Qiong Gu,[†] Liang Zeng,[‡] and Jun Xu*[,†]

[†]School of Pharmaceutical Sciences & Institute of Human Virology, Sun Yat-Sen University, 132 East Circle Road at University City, Guangzhou 510006, China

[‡]College of Computer Sciences, National University of Defense Technology, Changsha 410073, China

Supporting Information



ABSTRACT: High-performance computing (HPC) has become a state strategic technology in a number of countries. One hypothesis is that HPC can accelerate biopharmaceutical innovation. Our experimental data demonstrate that HPC can significantly accelerate biopharmaceutical innovation by employing molecular dynamics-based virtual screening (MDVS). Without using HPC, MDVS for a 10K compound library with tens of nanoseconds of MD simulations requires years of computer time. In contrast, a state of the art HPC can be 600 times faster than an eight-core PC server is in screening a typical drug target (which contains about 40K atoms). Also, careful design of the GPU/CPU architecture can reduce the HPC costs. However, the communication cost of parallel computing is a bottleneck that acts as the main limit of further virtual screening improvements for drug innovations.

INTRODUCTION

High-performance computing (HPC) has developed rapidly worldwide. The race to build faster supercomputers has become a source of national pride; these machines are valued for their ability to solve problems critical to national interests in areas such as defense, energy, finance, and science. Supercomputing has become a fundamental tool in the global competition for economic growth and scientific and technological development.

For the pharmaceutical industry, the question is whether and how HPC can accelerate biopharmaceutical innovation. The difficult, time-consuming process of screening compounds against drug targets is one bottleneck that limits biopharmaceutical innovation. Virtual screening (VS), where molecular dynamics (MD) simulation can play a key role,[1] is an attractive solution to this problem. Applying MD principles within a VS context for both predata and postdata analysis (i.e., MD-based virtual screening, MDVS) is an effective way of reducing false positives (FP) and false negatives (FN) in a VS campaign. In 2008, MD simulation of a protein, for 10 μs, required more than 3 months of supercomputing time.[2] It will require even more HPC time to simulate the interactions between complexes of thousands of ligands and a protein (i.e., a drug target).[3]

Two MDVS protocols that use MD simulations to improve the quality of hits by calculating the binding stabilities of hits with their target are depicted in Figure 1. The protocols reported in this article are readily transferable to other targets with known 3D structures and can be used to screen small molecular libraries. The protocols are mainly for benchmark purposes, without extensive optimization. The user can further optimize the protocol by modifying MD/docking parameters or adding filters, such as Lipinski rules, for a specific virtual screening campaign. The protocols are implemented in Linux scripting language with HPC-specific command lines and can be shared upon the request.

A target-ligand complex can consist of thousands of atoms. In such a situation, MD has to be simulated over a period of nanoseconds for researchers effectively to conclude that the

Received: July 5, 2013
Published: September 3, 2013

   dx.doi.org/10.1021/ci400391s | J. Chem. Inf. Model. 2013, 53, 2757−2764

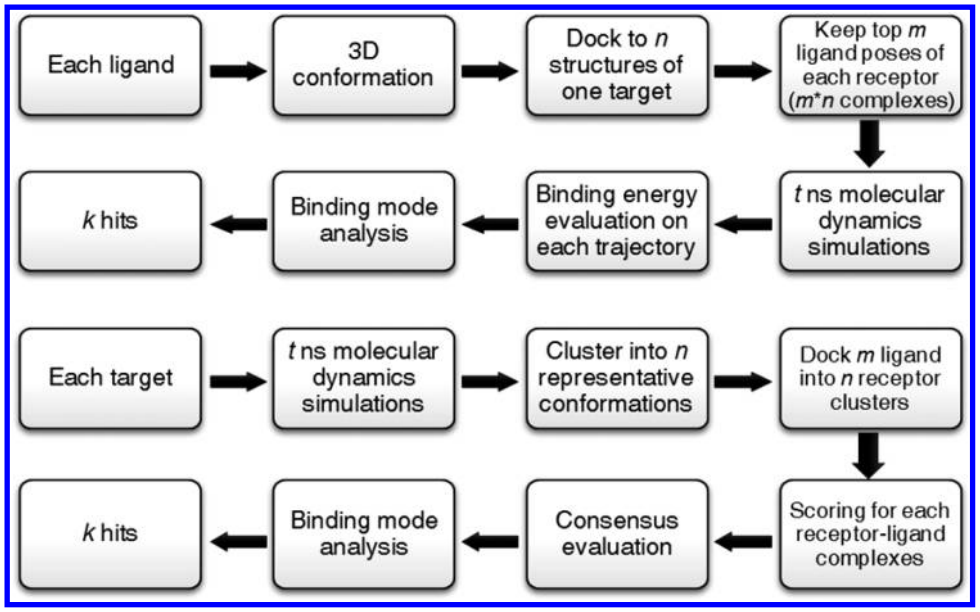


Figure 1. Two typical MDVS protocols.

ligand is a hit for further pharmaceutical study. Also, sometimes the time scale for atomic interaction requires that the time interval for the MD simulation be as short as 1 fs (10^{-15} s).[4] These requirements create additional demand for HPC power; this demand has become one of the driving forces for building faster HPC systems.[5]

Recently, graphics processing units (GPUs), originally developed for computing 3D functions, have provided unprecedented computational power for scientific applications. Examples include quantum Monte Carlo,[6] spectroscopic[7] and gravitational simulations[8] in computational physics, and sequence analysis in computational biology.[9,10] The current peak performance of state of the art GPUs is approximately 10 times faster than that of comparable CPUs. Therefore, more and more parallel MD simulation programs have GPU versions to take advantage of the GPU computing power.

To validate the impact of HPC on virtual screening within the drug innovation process, three GPU-based parallel MD programs—AMBER,[11,12] NAMD,[4] and GROMACS[13,14]—have been employed for experimentation in this report. The MDVS experiments were conducted with both CPU and GPU computer systems. The goals are to determine whether and how HPC can accelerate biopharmaceutical innovation and to identify the limits of such acceleration.

MATERIALS AND METHODS

Data Sets for Molecular Dynamics. One obvious determinant for the performance is the size of the simulated system. To determine the distribution of the size of drug target systems, a comprehensive database containing 1004 3D structures of known and potential drug targets was created using data from the literature and online databases such as the Drug Bank,[15] PDB (Protein Data Bank), SciFinder, TTD (Therapeutic Target Database),[16] PDTD (Potential Drug Target Database),[17] and TRP (Thomson Reuters Pharma) (Supporting Information Table S1). To calculate the number of atoms in solvated drug target systems, we solvated each structure in a rectangular box of TIP3P water molecules with 10 Å solvent layers between the box edges and solute surface. This was done via the tLeap module of AMBER. To test the MD performance with different programs, we selected 30 diverse targets from the above database as the test data set (Supporting Information Table S2). For each target, the topology and parameter files for AMBER, NAMD, and GROMACS were prepared, respectively. All three programs apply the ff99SB force field because it is widely used and is the common force field distributed with AMBER, GROMACS, and NAMD.

Molecular Dynamics Simulations. For comparison, each target-ligand complex in the test data sets was simulated repeatedly using AMBER, GROMACS, and NAMD. Prior to the production step, the data are processed with the following steps: (1) the solutes were fixed, and the solvent molecules with counterions were allowed to move during a 2000-step minimization; (2) all atoms were relaxed by a 2000-step full minimization; (3) after the minimization, each complex was gradually heated from 0 to 300 K in 600 ps, with solutes weakly constrained; (4) periodic boundary dynamics simulations of 20 ns were carried out with an NPT (constant composition, pressure, and temperature) ensemble at 1 atm and 300 K in the production step, using CPU- or GPU-accelerated versions of AMBER, GROMACS, and NAMD. The particle-mesh Ewald[18,19] (PME) method was used to treat long-range electrostatic interactions. A residue-based cutoff of 10 Å was utilized for noncovalent interactions. Algorithms constraining the temperature, pressure, and hydrogen bonds were used and might vary according to the package.

The binding free energy was calculated with the MM/PBSA (molecular mechanics Poisson–Boltzmann surface area)[20] method, using MMPBSA.py.[21] For each protein–ligand complex, 20 snapshots were extracted from the last nanosecond trajectory with an interval of 10 ps; these snapshots were used to calculate ΔG. A normal mode calculation was used to evaluate the entropy contribution to ΔG.

The versions of the MD packages were AMBER 12, GROMACS 4.6, and NAMD 2.9. They were all compiled by Intel C/C++/Fortran compiler 11.1.059. The Nvidia CUDA GPU driver version was 4.0, with ECC on.

Molecular Docking. For enrichment tests, three data sets containing actives and ligands were built, and their structures

were subjected to energy minimization through the MMFF94[22] force field to obtain the lowest-energy conformation prior to docking. The PDB files for the three targets were 1HXB for HIVPR, 2HU4 for NA, and 3VI8 for PPARα. The receptors were protonated at pH 7.4, and their original ligands in the binding pocket were removed. The native ligand binding site was selected as the binding site for docking. Each compound in the data set was docked into the three drug targets with GLIDE, MOE, CDOCKER, and SURFLEX. Other parameters were set with default values. For GLIDE, the SP (standard precision) mode was used. For MOE, the AMBER12/EHT force field and GBVI/WSA ΔG refinement were used. For CDOCKER, the random conformations were increased to 50. The docked poses were ranked according to each program's built-in scoring function, and the top-ranked binding pose for each compound was selected for further MD simulations.

Performance Monitoring and Data Analysis. For each simulation, the average time of the MD simulations was recorded. The backbone RMSD values of the trajectories were calculated to monitor the simulation quality.

To study the scalability of different systems/programs, parallel acceleration and efficiency were calculated according to the following formula

$$P_e = \frac{T_s}{nT_p}$$

where T_s is the time required to finish an MDVS task with a single CPU or GPU, T_p is the time to finish an MDVS task with multiple CPUs or GPUs, and n is the number of CPUs or GPUs used in parallel computing. In this study, a unit of CPU/GPU is defined by a thread. One CPU means one CPU core, and one GPU means one Tesla card.

To evaluate the memory requirement for running MD simulations of drug targets, the real-time memory usage for each run was recorded regularly. The real-time memory occupation of every thread was observed and recorded every 10 s. The per-node memory usage was calculated by adding the values of the threads. For a GPU program, the summation was not necessary because there was only one thread per node.

Data Sets for Enrichment Tests. To measure the performance of MDVS, three enrichment tests were carried out. Three drug targets—PPARα, neuraminidase, and HIV protease—were selected. Three data sets of their inhibitors and decoys were collected from different sources (Supporting Information Table S3), and the compounds of the data sets were docked onto the native ligand binding sites of the corresponding targets for virtual screening purposes with GLIDE,[23,24] MOE (Chemical Computing Group), CDOCKER (Accelrys), and SURFLEX (Tripos). The data sets were created as follows.

PPARα. Three hundred forty-three PPARα inhibitors with IC_{50} values of <1 μM were downloaded from BindingDB.[25] Using MACCS fingerprint similarity algorithms, we derived a diverse subset of 100 ligands from it. Then, 1000 decoy compounds were selected by RADER (i.e., Rapid Decoy Retriever, an in-house program).

Neuraminidase. One hundred thirty-two NA inhibitors were collected from the academic literature. One thousand random decoy compounds were selected from the Schrödinger Drug-Like Ligand Decoys Set,[23,24] and a data set of 1132 entries was constructed.

HIV Protease. Sixty-two HIV protease inhibitors were extracted from a DUD[26] (i.e., directory of useful decoys). Using MACCS fingerprint similarity algorithms, we derived 620 decoy compounds from the DUD HIV protease decoys set.

RESULTS

Determinants of the MDVS Computing Power Requirement. The number of atoms in a target-ligand complex, the number of compounds in a virtual library, the time scale for an MD simulation, the biochemical type, and the MD simulation program are the main determinants of the HPC power requirement. A survey of 1004 known and potential drug targets (Supporting Information Table S1) indicates that the most common drug target (Figure 2) has 30K to 40K atoms in a solvated target-ligand complex. The biochemical types covered in this report are listed in Table S2.

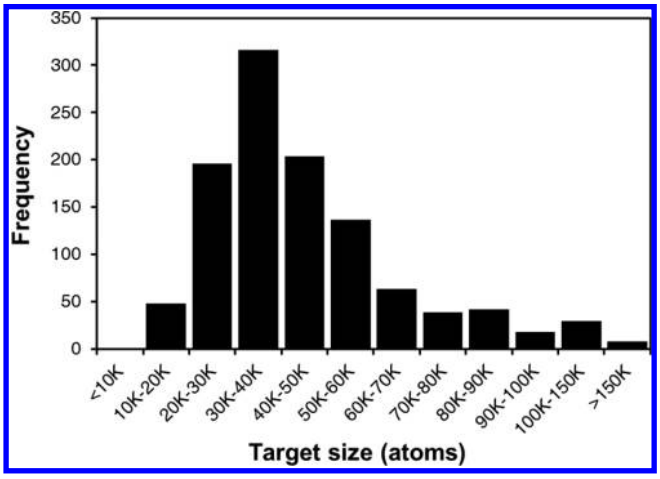


Figure 2. Scales of solvated drug targets and their distributions.

MD Performance and Target Size. MDVS was applied to 30 drug targets using AMBER, GROMACS, and NAMD for 20 ns MD simulations. The computations were executed in parallelized CPU and GPU systems. For each target-ligand complex, the performance (measured in units of ns/day) was calculated for each parallel level, and one peak performance out of all of the gradient parallel levels was obtained (Figure 3). Our study reveals that as the target size increases, the peak performance decreases. No significant difference was observed among the three programs. For the most common target size of about 30K to 40K atoms, a peak performance of about 50 ns/day was achieved.

Different MD simulation parameter settings (e.g., settings regarding the time step integrator, temperature, and pressure) in the three MD simulation packages make comparisons among the programs difficult. Therefore, direct comparisons of the performance among all three MD programs would be unfair and nonsensical. The parallel efficiency, however, may reflect whether the software takes full advantage of the HPC's power.

MD Performance and Parallel Efficiency. The contour maps in Figure 4 demonstrate the relationships among the target size (y axis, number of atoms), parallelization (x axis, number of CPUs or GPUs), and parallel efficiency (P_e), which is calculated via the following equation

$$P_e = \frac{T_s}{nT_p}$$

dx.doi.org/10.1021/ci400391s | J. Chem. Inf. Model. 2013, 53, 2757−2764

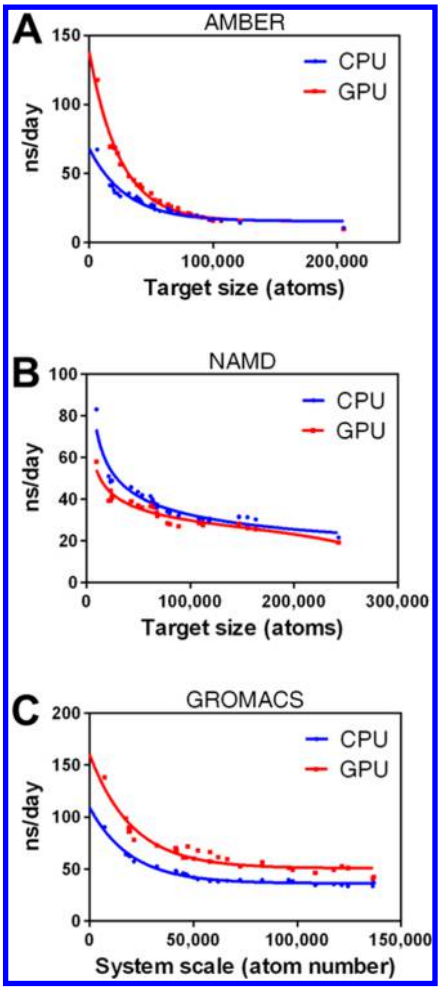


Figure 3. Peak performance for MDVS of 30 targets using AMBER, NAMD, and GROMACS with CPU and GPU computers. The curves are fitted to indicate the trends.

where T_s is the time required to finish an MDVS task with a single CPU or GPU, T_p is the time required to finish an MDVS task with multiple CPUs or GPUs, and n is the number of CPUs or GPUs used in the parallel computing. It should be noted that in Figure 5 blue stands for 0% P_e and red stands for 100% P_e.

Figure 4 reveals the following trends: (1) P_e declines when the number of CPUs increases; (2) P_e declines more rapidly when the number of GPUs increases; (3) increasing the number of CPUs or GPUs for a larger target size does not proportionally increase the P_e; this may be due to the increasing communication costs of more CPUs or GPUs joining the parallel computing; (4) NAMD seems to have good parallel efficiency for larger targets on both GPUs and CPUs; and (5) AMBER and GROMACS have better parallel efficiency on CPUs than on GPUs.

Memory Concern. All three programs have minimal RAM (random access memory) requirements to run MDVS on a target (Figure S1).

CPU-based programs demand more memory than do GPU-based programs: GPU-based and CPU-based programs require 500 MB and 2 GB of memory for an average-sized drug target, respectively, because GPU has a larger built-in memory. For example, Tesla M2050 GPU has 3 GB of memory, which meets the needs of MDVS. For a TH-1A supercomputer, each parallel node has 24 GB of memory, which almost meets the memory need for any drug target.

Consistency of CPU-Based and GPU-Based MD Programs. To determine whether CPU-based and GPU-based programs produce consistent results, we ran the CPU-based and GPU-based AMBER programs (PMEMD) on 30 drug targets (Table S2) to simulate their molecular dynamics for 20 ns. Then, we calculated the RMSDs of the backbone atoms in the trajectories (of both the CPU-based and GPU-based AMBER programs) and the corresponding crystal structures of the targets. The results are depicted in Figure S2, which indicate that most of the MD simulated trajectories from GPU-based programs are consistent with the targets' corresponding crystal structures (RMSD < 3 Å) and most of the trajectories from GPU-based and CPU-based programs are consistent with one another, except for 1XY1 and 2GBP. With regard to the exception, the CPU-based programs produced results that were inconsistent with the corresponding crystal structures.

MDVS Performance and Parallelization. MDVS was performed to screen a compound library derived from the NCI Open Database (the library was released May 4, 2012) against the MJ0796 ATP-binding cassette target (PDB code 1F3O). The library consists of 265 242 compounds, which were docked to the receptor-binding site of 1F3O with GLIDE. The 10 000 best scored poses were selected, and the corresponding compound-receptor complexes were created for the subsequent 1 ns MD simulations with AMBER, NAMD, and GROMACS. The entire MDVS task was divided into subtasks and distributed to a number of CPUs and GPUs for calculations; the calculations utilized different parallelization strategies. The relationship between the time needed to complete an MDVS campaign (for 10K compounds) and the HPC parallelization strategies (of different programs) is depicted in Figure 5.

Figure 5A,B show that increasing the number of CPUs or GPUs for each MDVS subtask will prolong the time needed to accomplish each subtask. Consequently, it will take a longer time to complete the MDVS campaign. This is because higher P_{CPU} and P_{GPU} values increase the communication costs among calculating units, which in turn causes delays and wastes computing resources. As shown in Figure 5C,D, when an MDVS campaign can employ a greater number of HPC nodes (7168) on TH-1A, the virtual screening performance improves significantly with the increases in P_{CPU} and P_{GPU} (before they reach eight parallelized processing units). After that, the communication costs among calculating units will slowly reduce the MDVS campaign performance; this effect is particularly significant in the case of GPUs because the serial architecture (or low parallelization per subtask) cannot take full advantage of the higher HPC power.

The relationship between the time needed to accomplish an MDVS subtask and its parallelization strategy can be described with the following formula

$$T = \left\lceil \frac{pN_{poses}}{CN_{nodes}} \right\rceil \frac{t}{s}$$

where T is the time required for the entire MDVS task to be completed, p is the number of threads that run on paralleled CPUs or GPUs, N_{poses} is the number of all ligand-binding poses, N_{nodes} is the number of HPC hardware nodes, C is the maximal number of CPU or GPU threads per HPC hardware node, t is the MD simulation time (measured in nanoseconds),

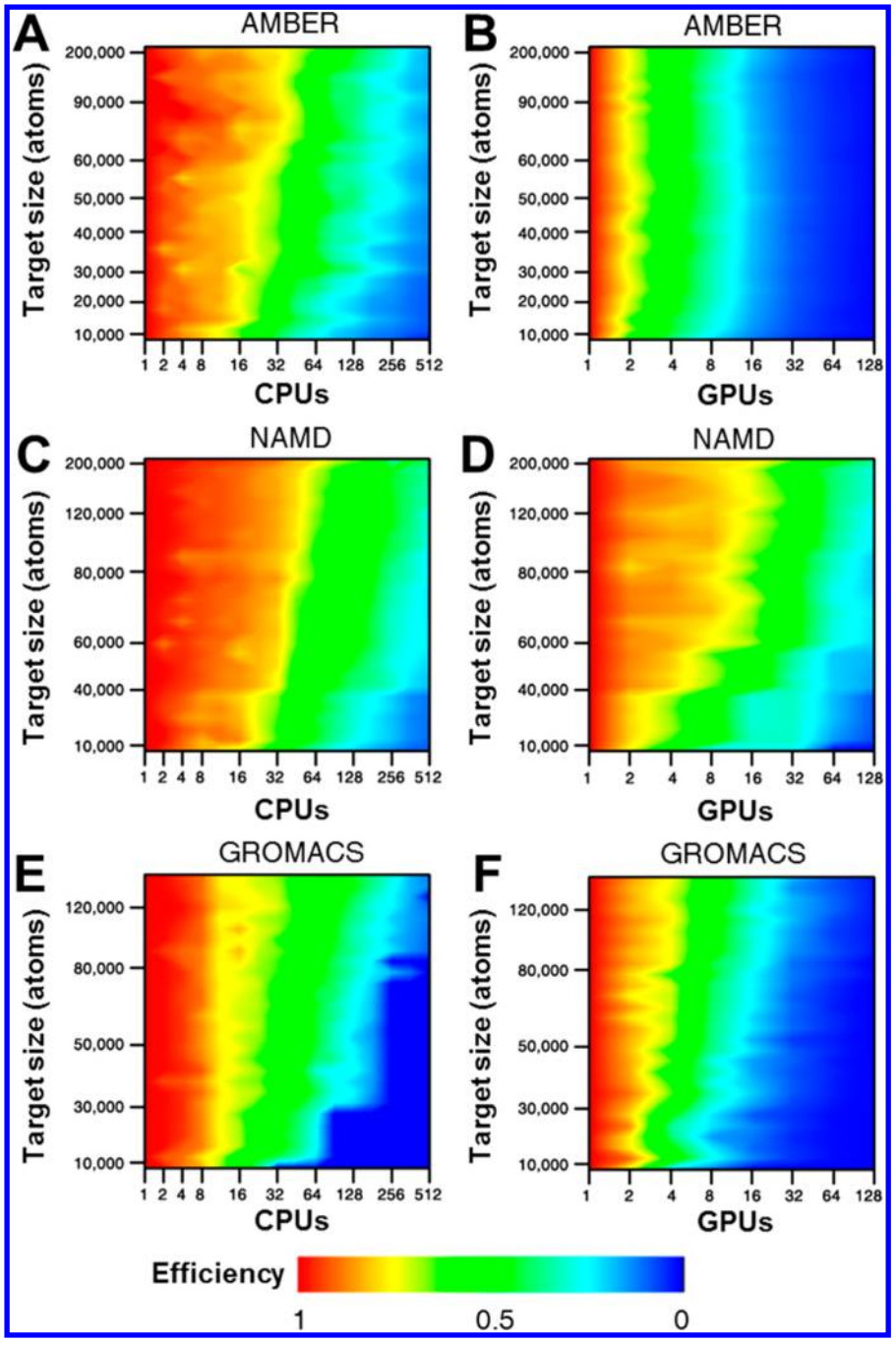


Figure 4. Parallel efficiencies of MDVS for 30 targets, which resulted from using the AMBER, NAMD, and GROMACS programs, all of which ran on both CPUs and GPUs.

and s is the MD simulation throughput (ns/day). The minimal time costs for the same MDVS campaign to run on a PC server as opposed to the time costs of the three HPC systems are compared in Table 1. Field tests were carried out on 200 nodes of TH-1A and 512 nodes of TH-1/GZ (a pilot system of TH-2A in Guangzhou, China) separately. The times needed for a PC to complete the MDVS campaign were extrapolated on the basis of a mainstream PC server with eight CPU cores. It is impossible to use a PC server to complete an MDVS campaign requiring thousands of days. Using 512 nodes of TH-1/GZ, the MDVS campaign can be accelerated more than 600-fold. If the entire computing power of TH-1A is employed, then the MDVS campaign can be carried out in less than 30 min with GROMACS_GPU, which is about 3324 times faster than that of a PC with GPU or 35 430 times faster than a PC without GPU. With the power of HPC, incorporating MD into drug screening no longer seems to be a problem of yes or no but of when and where.

CPU/GPU Ratio and MD Simulation Cost. GPUs are coprocessors and have to work together with CPUs. The CPI (capacity-to-price index) for HPC can be estimated as follows:

$$\mathrm{CPI} = \frac{N_{\mathrm{GPU}}P_{\mathrm{GPU}} + (N_{\mathrm{CPU}} - N_{\mathrm{GPU}})P_{\mathrm{CPU}}}{N_{\mathrm{GPU}}U_{\mathrm{GPU}} + N_{\mathrm{CPU}}U_{\mathrm{CPU}}}$$

CPI is measured in ps/day/dollar. N_{GPU} and N_{CPU} are the numbers of GPUs and CPUs built in an HPC node. P_{GPU} and

dx.doi.org/10.1021/ci400391s | J. Chem. Inf. Model. 2013, 53, 2757−2764

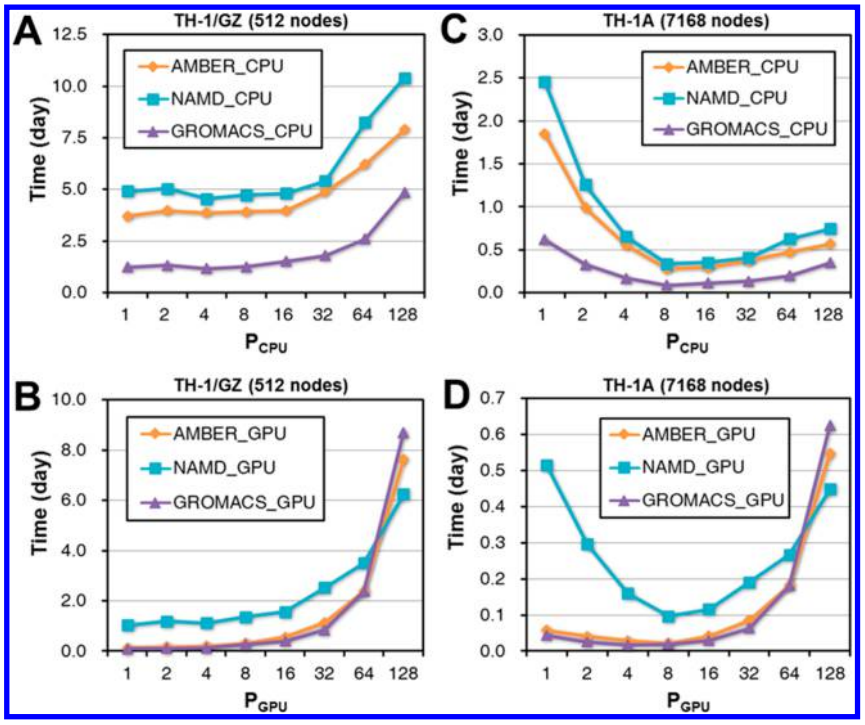


Figure 5. Relationship between the time needed to complete an MDVS campaign and the HPC parallelization strategy (of different programs). The MDVS campaign involves screening 10K compounds. P_{CPU}: number of parallelized CPUs called by each MDVS subtask. P_{GPU}: number of parallelized GPUs called by each MDVS subtask. (A) MDVS campaign run by three programs in the 512-node parallelized CPUs of TH-1/GZ. (B) MDVS campaign run by three programs in the 512-node parallelized GPUs of TH-1/GZ. (C) MDVS campaign run by three programs in the 7168-node parallelized CPUs of TH-1A (theoretical projections). (D) MDVS campaign run by three programs in the 7168-node parallelized GPUs of TH-1A (theoretical projections).

Table 1. Total Times for the Same MDVS Campaign as Run on a PC Server and Three HPC Systems

time spent (days)	PC[a]	TH-1A (200 nodes)[b]	TH-1/GZ (512 nodes)[b]	TH-1A (7168 nodes)[a]
AMBER_CPU	2314	8.91	3.70	0.279
GROMACS_CPU	771	2.84	1.17	0.089
NAMD_CPU	3069	11.06	4.55	0.337
AMBER_GPU	72	0.29	0.12	0.022
GROMACS_GPU	53	0.22	0.09	0.017
NAMD_GPU	643	2.57	1.03	0.097

[a]Extrapolated measures. [b]Experimental measures.

P_{CPU} are the maximal performance values contributed by a thread from a CPU and GPU card, as measured in ns/day. U_{CPU} and U_{GPU} are the unit prices of a CPU core and a GPU card.

At the time of this writing, the market price for an NVIDIA Tesla M2050 (Fermi GPU Card-3GB GDDR5) is $1090; the price for an Intel Xeon X5670 processor (2.93 GHz 12 MB Cache Socket LGA1366) is $1120; and the price for a CPU core is about $187. An average drug target with 30K−40K atoms, such as 1F3O, requires no more than 24 CPU cores and 4 GPU cards. Therefore, the relationships among CPI, the number of CPUs, and the number of GPUs are depicted in Figure 6.

Because each GPU card needs at least one CPU core to function, the architecture of one CPU with multiple GPUs is not valid. Figure 6A indicates that (1) the CPI drops when the CPU/GPU ratio increases and (2) the optimized CPU/GPU ratio is 4:4 for parallelized AMBER. GROMACS shows a similar trend to that of AMBER (Figure 6C). NAMD has a different trend: CPI will increase with increasing numbers of CPUs and GPUs (Figure 6B). In short, the CPU/GPU ratio should be greater than 1, and the higher the ratio, the better.

Hit Enrichment of MDVS. To measure the performance of MDVS, three drug targets—PPARα, neuraminidase, and HIV protease—were selected. Three data sets of their inhibitors and

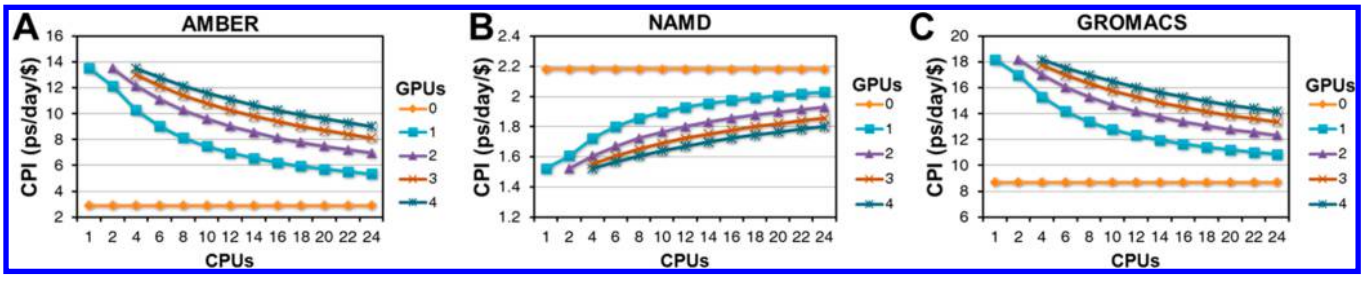


Figure 6. Relationship between of CPI (cost performance index) and the CPU/GPU ratio. (A) AMBER, (B) NAMD, and (C) GROMACS.

dx.doi.org/10.1021/ci400391s | J. Chem. Inf. Model. 2013, 53, 2757−2764

decoys were collected from different sources (Table S3), and the compounds of the data sets were docked onto the native ligand binding sites of the corresponding targets for virtual screening purposes with GLIDE,[23,24] MOE (Chemical Computing Group), CDOCKER (Accelrys), and SURFLEX (Tripos). The docked poses were ranked according to each program's built-in scoring function. The performances of various virtual screening methods, including MDVS, are demonstrated with receiver operating characteristic curves (ROCs)[27] (specifically through the corresponding areas under the curves (AUCs) in Figure 7).

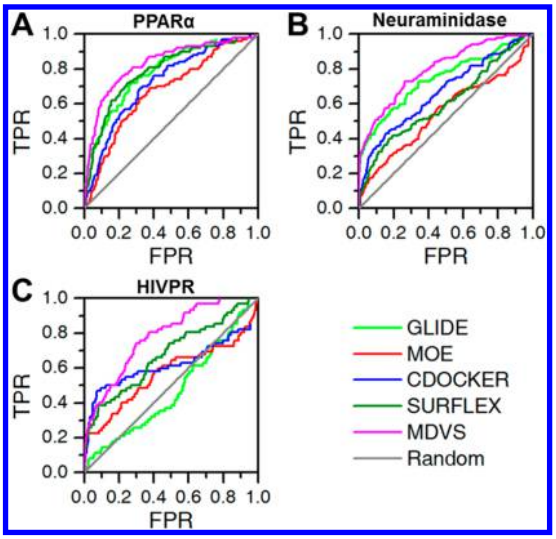


Figure 7. Receiver operating characteristic curves (ROC) for virtual screening campaigns. FPR, false positive rate (% decoys); TPR, true positive rate (% actives). The ROC for MDVS is in magenta. (A) PPARα, (B) neuraminidase, and (C) HIV protease.

In the TH-1/GZ HPC system, 1 ns of MD simulation was applied to each top receptor–ligand complex, and its binding free energy was evaluated. The enrichment for binding free energy results were compared as an individual series of ROC plots (Figure 7). Although different programs demonstrate different virtual screening performances for three targets, the ROC curves of MDVS are always the best. To measure the performance further,[28,29] AUCs, enrichment factors (EF), robust initial enhancements (RIE), areas under the accumulation curve (AUAC), and the Boltzmann-enhanced discrimination of receiver operating characteristics (BEDROC)[29] were calculated (Supporting Information Table S4). The enrichment test of HIV protease was also reported in the literature,[30] where AutoDock 4 and Vina demonstrated AUCs of 0.40 and 0.66, respectively. The AUC of MDVS is 0.78, which is 18.2% greater than reported. All of these metrics indicate that the MDVS approach is superior to conventional virtual screening approaches. These results are consistent with the observations of our colleagues.[31]

DISCUSSION

One of the bottlenecks in biopharmaceutical innovation is biological screening. Virtual screening has been an attractive solution. MDVS can significantly improve the performance of a virtual screening campaign. Reducing the HPC cost and increasing the computational speed will significantly accelerate biomedical innovation. There are no disease-modifying drugs against, for example, Alzheimer's, osteoarthritis, metabolic syndromes, and important cancers, and the MDVS campaigns described in this article may shorten the time for preclinical studies for disease-modifying drugs and antipandemic drugs.

MDVS involves validating the interactions of thousands or millions of individual compounds against a drug target. The task is parallelizable by dividing a great number of compound–protein complexes into smaller computational packages, which are distributed to GPU-CPU units for calculations. MDVS does not require massive memory but demands only greater HPC power.

Many supercomputers adopt GPUs as coprocessors to reduce manufacturing costs and increase calculation speed. However, the architecture of combining CPUs and GPUs is one of the key factors in computing performance. Another key factor is the application program. Today, more and more MD simulation programs offer their parallel or GPU versions, but they have different ways of exploiting HPC power. Consequently, their performances are quite different.

HPC technology may continue to obey Moore's law, but the updating of application programs can be a bottleneck. Therefore, it is time to invest more in developing HPC-based application programs while we are building a supercomputer with a higher Linpack Benchmark score.

ASSOCIATED CONTENT

Supporting Information

Minimal memory requirements of the AMBER, GROMACS, and NAMD programs. RMSDs of the backbone atoms in the trajectories of both CPU-based and GPU-based AMBER programs and the corresponding crystal structures for the 30 drug targets. Numbers of solvated drug targets. Thirty drug targets being MD simulated at 20 ns. Enrichment tests for MDVS. Statistical data for the MDVS enrichment tests. This material is available free of charge via the Internet at http://pubs.acs.org.

AUTHOR INFORMATION

Corresponding Author

*E-mail: junxu@biochemomes.com.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was funded in part by the National High-Tech R&D Program of China (863 Program) (2012AA020307), the Introduction of Innovative R&D Team Program of Guangdong Province (no. 2009010058), and the National Natural Science Foundation of China (nos. 81001372 and 81173470). The National Supercomputing Centers in Guangzhou (2012Y2-00048 and 201200000037), Tianjin, and Shenzhen have provided the computing resources necessary for this report.

REFERENCES

(1) Huang, D. N.; Gu, Q.; Hu, G.; Ye, J. M.; Salam, N. K.; Hagler, A.; Chen, H. Z.; Xu, J. On the value of homology models for virtual screening: discovering hCXCR3 antagonists by pharmacophore-based and structure-based approaches. *J. Chem. Inf. Model.* **2012**, *52*, 1356–1366.

(2) Freddolino, P. L.; Liu, F.; Gruebele, M.; Schulten, K. Ten-microsecond molecular dynamics simulation of a fast-folding WW domain. *Biophys. J.* **2008**, *94*, L75–L77.

(3) Schneider, G. Virtual screening: an endless staircase? *Nat. Rev. Drug Discovery* **2010**, *9*, 273−276.
(4) Kale, L.; Skeel, R.; Bhandarkar, M.; Brunner, R.; Gursoy, A.; Krawetz, N.; Phillips, J.; Shinozaki, A.; Varadarajan, K.; Schulten, K. NAMD2: greater scalability for parallel molecular dynamics. *J. Comput. Phys.* **1999**, *151*, 283−312.
(5) Shaw, D. E.; Maragakis, P.; Lindorff-Larsen, K.; Piana, S.; Dror, R. O.; Eastwood, M. P.; Bank, J. A.; Jumper, J. M.; Salmon, J. K.; Shan, Y.; Wriggers, W. Atomic-level characterization of the structural dynamics of proteins. *Science* **2010**, *330*, 341−346.
(6) Anderson, A. G.; Goddard, W. A.; Schroder, P. Quantum Monte Carlo on graphical processing units. *Comput. Phys. Commun.* **2007**, *177*, 298−306.
(7) Collange, S.; Daumas, M.; Defour, D. Line-by-line spectroscopic simulations on graphics processing units. *Comput. Phys. Commun.* **2008**, *178*, 135−143.
(8) Belleman, R. G.; Bedorf, J.; Zwart, S. F. P. High performance direct gravitational N-body simulations on graphics processing units II: An implementation in CUDA. *New Astron.* **2008**, *13*, 103−112.
(9) Liu, W.; Schmidt, B.; Voss, G.; Muller-Wittig, W. Streaming algorithms for biological sequence alignment on GPUs. *IEEE Trans. Parallel Distrib. Syst.* **2007**, *18*, 1270−1281.
(10) Schatz, M. C.; Trapnell, C.; Delcher, A. L.; Varshney, A. High-throughput sequence alignment using graphics processing units. *BMC Bioinf.* **2007**, *8*, 474.
(11) Case, D. A.; Cheatham, T. E.; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668−1688.
(12) Weiner, P. K.; Kollman, P. A. Amber-assisted model-building with energy refinement - a general program for modeling molecules and their interactions. *J. Comput. Chem.* **1981**, *2*, 287−303.
(13) Hess, B.; Kutzner, C.; van der Spoel, D.; Lindahl, E. GROMACS 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation. *J. Chem. Theory Comput.* **2008**, *4*, 435−447.
(14) Pronk, S.; Pall, S.; Schulz, R.; Larsson, P.; Bjelkmar, P.; Apostolov, R.; Shirts, M. R.; Smith, J. C.; Kasson, P. M.; van der Spoel, D.; Hess, B.; Lindahl, E. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinformatics* **2013**, *29*, 845−854.
(15) Knox, C.; Law, V.; Jewison, T.; Liu, P.; Ly, S.; Frolkis, A.; Pon, A.; Banco, K.; Mak, C.; Neveu, V.; Djoumbou, Y.; Eisner, R.; Guo, A. C.; Wishart, D. S. DrugBank 3.0: a comprehensive resource for 'Omics' research on drugs. *Nucleic Acids Res.* **2011**, *39*, D1035−D1041.
(16) Zhu, F.; Shi, Z.; Qin, C.; Tao, L.; Liu, X.; Xu, F.; Zhang, L.; Song, Y.; Liu, X. H.; Zhang, J. X.; Han, B. C.; Zhang, P.; Chen, Y. Z. Therapeutic target database update 2012: a resource for facilitating target-oriented drug discovery. *Nucleic Acids Res.* **2012**, *40*, D1128−D1136.
(17) Gao, Z. T.; Li, H. L.; Zhang, H. L.; Liu, X. F.; Kang, L.; Luo, X. M.; Zhu, W. L.; Chen, K. X.; Wang, X. C.; Jiang, H. L., PDTD: a web-accessible protein database for drug target identification. *BMC Bioinformatics* **2008**, *9*.
(18) Darden, T.; York, D.; Pedersen, L. Particle mesh Ewald: an N·log(N) method for Ewald sums in large systems. *J. Chem. Phys.* **1993**, *98*, 10089−10092.
(19) Kholmurodov, K.; Smith, W.; Yasuoka, K.; Darden, T.; Ebisuzaki, T. A smooth-particle mesh Ewald method for DL_POLY molecular dynamics simulation package on the Fujitsu VPP700. *J. Comput. Chem.* **2000**, *21*, 1187−1191.
(20) Hou, T. J.; Wang, J. M.; Li, Y. Y.; Wang, W. Assessing the performance of the MM/PBSA and MM/GBSA Methods. 1. The accuracy of binding free energy calculations based on molecular dynamics simulations. *J. Chem. Inf. Model.* **2011**, *51*, 69−82.
(21) Miller, B. R.; McGee, T. D.; Swails, J. M.; Homeyer, N.; Gohlke, H.; Roitberg, A. E. MMPBSA.py: an efficient program for end-state free energy calculations. *J. Chem. Theory Comput.* **2012**, *8*, 3314−3321.
(22) Halgren, T. A. Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *J. Comput. Chem.* **1996**, *17*, 490−519.
(23) Friesner, R. A.; Banks, J. L.; Murphy, R. B.; Halgren, T. A.; Klicic, J. J.; Mainz, D. T.; Repasky, M. P.; Knoll, E. H.; Shelley, M.; Perry, J. K.; Shaw, D. E.; Francis, P.; Shenkin, P. S. Glide: a new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy. *J. Med. Chem.* **2004**, *47*, 1739−1749.
(24) Halgren, T. A.; Murphy, R. B.; Friesner, R. A.; Beard, H. S.; Frye, L. L.; Pollard, W. T.; Banks, J. L. Glide: a new approach for rapid, accurate docking and scoring. 2. Enrichment factors in database screening. *J. Med. Chem.* **2004**, *47*, 1750−1759.
(25) Liu, T.; Lin, Y.; Wen, X.; Jorissen, R. N.; Gilson, M. K. BindingDB: a web-accessible database of experimentally determined protein-ligand binding affinities. *Nucleic Acids Res.* **2007**, *35*, D198−201.
(26) Huang, N.; Shoichet, B. K.; Irwin, J. J. Benchmarking sets for molecular docking. *J. Med. Chem.* **2006**, *49*, 6789−6801.
(27) Linden, A. Measuring diagnostic and predictive accuracy in disease management: an introduction to receiver operating characteristic (ROC) analysis. *J. Eval. Clin. Pract.* **2006**, *12*, 132−139.
(28) Nicholls, A. What do we know and when do we know it? *J. Comput. Aided Mol. Des.* **2008**, *22*, 239−255.
(29) Truchon, J. F.; Bayly, C. I. Evaluating virtual screening methods: Good and bad metrics for the "early recognition" problem. *J. Chem. Inf. Model.* **2007**, *47*, 488−508.
(30) Chang, M. W.; Ayeni, C.; Breuer, S.; Torbett, B. E. Virtual screening for HIV protease inhibitors: a comparison of AutoDock 4 and Vina. *PLoS One* **2010**, *5*, e11955.
(31) Li, Z.; Cai, Y. H.; Cheng, Y. K.; Lu, X.; Shao, Y. X.; Li, X.; Liu, M.; Liu, P.; Luo, H. B. Identification of novel phosphodiesterase-4D inhibitors prescreened by molecular dynamics-augmented modeling and validated by bioassay. *J. Chem. Inf. Model.* **2013**, *53*, 972−981.