Automated Molecular Simulation Based Binding Affinity Calculator for Ligand-Bound HIV-1 Proteases

S. Kashif Sadiq, David Wright, Simon J. Watson, Stefan J. Zasada, Ileana Stoica, and Peter V. Coveney*

Centre for Computational Science, Department of Chemistry, University College London, London, WC1H 0AJ, U.K.

Received March 14, 2008

The successful application of high throughput molecular simulations to determine biochemical properties would be of great importance to the biomedical community if such simulations could be turned around in a clinically relevant timescale. An important example is the determination of antiretroviral inhibitor efficacy against varying strains of HIV through calculation of drug−protein binding affinities. We describe the Binding Affinity Calculator (BAC), a tool for the automated calculation of HIV-1 protease−ligand binding affinities. The tool employs fully atomistic molecular simulations alongside the well established molecular mechanics Poisson−Boltzmann solvent accessible surface area (MMPBSA) free energy methodology to enable the calculation of the binding free energy of several ligand−protease complexes, including all nine FDA approved inhibitors of HIV-1 protease and seven of the natural substrates cleaved by the protease. This enables the efficacy of these inhibitors to be ranked across several mutant strains of the protease relative to the wildtype. BAC is a tool that utilizes the power provided by a computational grid to automate all of the stages required to compute free energies of binding: model preparation, equilibration, simulation, postprocessing, and data-marshaling around the generally widely distributed compute resources utilized. Such automation enables the molecular dynamics methodology to be used in a high throughput manner not achievable by manual methods. This paper describes the architecture and workflow management of BAC and the function of each of its components. Given adequate compute resources, BAC can yield quantitative information regarding drug resistance at the molecular level within 96 h. Such a timescale is of direct clinical relevance and can assist in decision support for the assessment of patient-specific optimal drug treatment and the subsequent response to therapy for any given genotype.

1. INTRODUCTION

Molecular dynamics (MD) is a well established computational methodology for studying the time-evolution and conformational dynamics of a diverse array of physicochemical systems at the molecular level, from which a whole host of physical and chemical properties can be determined.[1] The implementation of any physically realistic molecular simulation has, however, always been an involved and multistage process, often requiring the scientist to overcome a large manual overhead in the construction, preparation, and execution protocols needed to complete a set of simulations as well as to invoke various analysis protocols for determining desired properties postproduction. Conversely, the fact that molecular simulation is very computationally expensive, thus limiting the number of simulations that can be achieved in any given period of time, has meant that the manual preparation of a given simulation has not been the overriding issue in implementing a study. Additionally, the complexities involved in preparing a physically realistic molecular model usually require slightly divergent protocols to be implemented between similar molecular systems.

With the advent of grid technology[2] and the availability of geographically distributed high performance computing (HPC) resources, the opportunity to perform large numbers of compute-intensive molecular simulations has become realistic. This opens up a novel modus operandi for molecular simulation, which in turn enables different avenues of scientific enquiry to be pursued. For example, in the biomolecular domain, the substantial variations in binding properties that may exist between only slightly differing receptor−ligand complexes mean that the ability to determine binding affinities for a whole host of slightly varying receptor−ligand combinations in a high throughput manner is not only highly desirable but also now in principle very feasible.

Within the framework of such a new approach, the bottleneck then becomes the manual overhead in the preparation and execution of a large set of distributed simulations. Provided a robust protocol exists for the simulation of a given biomolecular system, a user wanting to study scientific aspects of the system will want to avoid spending time on the repetitive, manual construction and implementation of the required set of molecular dynamics (MD) simulations. Time can be spent more productively if only the varying range of parameters of scientific interest need be specified, while an automated protocol addresses both the construction of the simulation-ready model and the implementation of

* Corresponding author. Address: Centre for Computational Science, Department of Chemistry, University College London, 20 Gordon Street, London, WC1H 0AJ, U.K. Tel.: +44 (0) 207 679 4560. Fax: +44 (0) 207 679 7463. E-mail: p.v.coveney@ucl.ac.uk.

10.1021/ci8000937 CCC: $40.75
Published on Web 08/19/2008

the resulting set of simulations, together with the required postproduction analyses.

Studying the binding of antiretroviral inhibitors to wildtype and mutant strains of HIV-1 protease is of particular importance in the attempt to determine both the optimal inhibitor efficacy for any particular strain and also to uncover the molecular determinants of emergent drug resistant mutants.[3−5] The binding affinity of ligands to HIV-1 protease has been extensively studied using molecular simulation and a number of free energy calculation techniques. These have ranged from highly compute-intensive methods such as thermodynamic integration (TI) on short timescales[6] through moderately intensive continuum solvent methods like the molecular mechanics Poisson−Boltzmann solvent accessible surface area (MMPBSA) method[7−9] on slightly longer timescales, to more empirical methods such as linear interaction energy (LIE).[10] We have also recently reported an accurate and fast protocol, utilizing the MMPBSA and normal-mode based configurational entropy methods, for ranking drug resistant mutants of HIV-1 protease to the inhibitor saquinavir.[11]

In the medical domain, genotypic assaying of patients infected with HIV is routinely implemented to determine patient-specific viral sequences of key antiretroviral targets such as protease and reverse-transcriptase.[12] However, the interpretation of such information, given the complexity of emergent mutational patterns,[13] means that clinicians have had to resort to decision-support software for assistance.[14] Such decision-support software uses existing clinical databases as well as phenotypic information from inhibition studies to rank the susceptibility of a range of inhibitors to a particular viral sequence.

Unfortunately, phenotypic determination of inhibitor efficacy through either experimental or computational means is not trivial and has conventionally taken too long to be of clinical use. The number of drug-resistant mutant sequences of HIV-1 protease[15] thus far outweighs the number of sequences for which drug-binding affinities have been determined or which have even been studied using available experimental and computational techniques. A compelling goal, therefore, would be the realization of a high throughput tool which is accurate in ranking the drug resistance conferred by different protease mutants across a range of inhibitors and rapid enough to return results on clinically relevant timescales (∼2 weeks).

Here, we describe a software tool, the Binding Affinity Calculator (BAC), used to implement our recent study,[11] which establishes an automated platform based on molecular simulation through which the resistance conferred by a vast array of clinically important mutations of HIV-1 protease can be determined for the current array of FDA inhibitors. In principle, BAC is also extendible to novel inhibitors and alternative protein targets. We have recently described the potential for BAC to be utilized in patient-specific decision support for optimizing therapy as well as the assessment of response to therapy.[16,17]

BAC automates the various model construction, MD equilibration, simulation, and postproduction analysis protocols, while requiring the specification of only a few biological input parameters. It extends protocol development to all nine current FDA approved inhibitors of HIV-1 protease as well as seven of the natural substrates cleaved by the protease for an arbitrarily large protease mutation space. Furthermore, the protocols adapt automatically based on initial mutation specification to allow any protease mutant to be robustly equilibrated and then simulated. In its most automated sense, only the identity of the ligand, the protonation state of the catalytic dyad of the protease, and any mutations relative to a designated wildtype need to be assigned. In addition, molecular dynamics simulations of the apo-HIV-1 protease can be implemented using over 200 potential starting crystal structures.

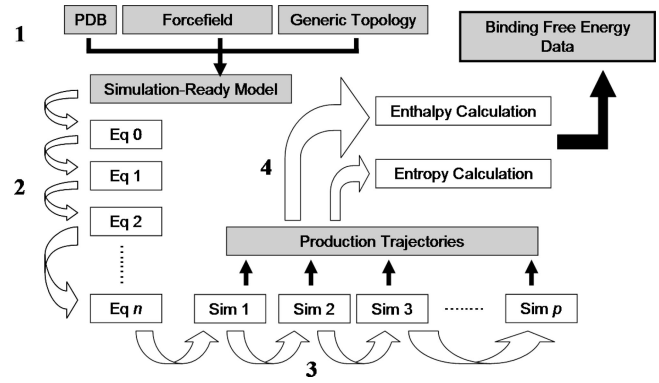

Figure 1. Workflow of an MMPBSA free energy calculation comprising four sequential stages. (1) Preparation of a simulation-ready model from the protein data bank crystal structure (PDB), forcefield parameters, and generic topology information. (2) Linear chain of equilibration simulations. (3) Linear chain of production simulations each generating trajectories for analysis. (4) Postproduction execution of the enthalpy and entropy calculations leading to a determination of the binding free energy. Data files are shown in gray boxes, processes, in white boxes.

In this paper, we describe our preparation, mutation-adaptive equilibration, and simulation protocols alongside the workflow we implement to obtain free energies of ligand−protease binding. The robustness of the equilibration protocol contained in BAC is validated by investigating the structural and dynamical properties of a multidrug resistant (MDR) protease with ten dimeric mutations from the wildtype, using two different starting crystal structures. The first is of an existing crystal structure of the MDR protease, the second is an artificially generated structure of the MDR protease derived from the wildtype crystal structure and thus equilibrated differently. We then describe the overall architecture and workflow management of BAC, followed by a description of each of its components. Finally, we conclude with a discussion of possible biomedical applications, in which BAC has the potential to aid in clinical decision making.

2. FREE ENERGY CALCULATION METHODOLOGY

In general, there is no unique way to implement molecular simulations and postproduction free energy analyses required to obtain binding free energies of a given protease−ligand complex. We employ fully atomistic molecular simulations alongside MMPBSA and normal-mode based configurational entropy methods to determine absolute free energies of protease−ligand binding. Here, we provide a general overview of the workflow adopted for such a procedure as well as a subsequent description of the protocols implemented at each stage of the workflow.

2.1. Workflow. Figure 1 shows the various steps required for the execution of the workflow adopted here. We begin

with the assumption that a starting crystal structure of the complex, obtainable from the protein data bank (PDB),[18] exists and that forcefield and charge parameters for the protein and ligand are also provided. In general, however, there are many more protease mutants of interest than available crystal structures, which severely limits the number of varying protease−ligand simulations that can be performed. There are also ligands whose partial charges are not readily available. In section 2.2.1, we discuss how BAC surmounts such limitations.

Prior to any molecular dynamics, a simulation-ready model is generated from the PDB coordinate information together with the generic topology and forcefield parameter information. The process of generating such a model requires the extraction of suitable protease and ligand coordinates, incorporation of any mutations, the addition of charge neutralizing ions and solvation of the target structure. System-specific topology and coordinate files then have to be generated which form the input for subsequent simulation. Thus far we have assumed the availability of forcefield and charge information for the protease and ligand. In general, however, the assignment of ligand partial charges is not readily available and needs to be implemented for each inhibitor as part of the preliminary model preparation stage.

The next stage involves the array of sequential equilibration simulations that need to run before production simulations can commence. These include the stages of minimization, annealing the system, the gradual relaxing of constraints which vary based on the mutations that have been incorporated, and, finally, unrestrained equilibration in a specified thermodynamic ensemble. Each step of this sequential protocol utilizes a separate configuration file containing the exact instructions for that simulation. The output state data of one step in the protocol is then used as the input state of the following step until the end of the equilibration phase.

The production phase is very similar to equilibration and also consists of a chain of sequentially executed simulations. Each stage of the production phase is executed using a separate configuration file, which again reads in the output-state of the previous stage of the simulation. In principle, it is possible to implement only one stage, where a single simulation is run for a sufficiently long time to traverse the entire production phase. In practice however, the queuing regulations for single continuous computations on many HPC resources make it more sensible to decompose the production simulation into several sequentially run and individually queued components.

Finally, the trajectory information that is output in the production phase is postprocessed in order to calculate the enthalpies and entropies of binding using MMPBSA and normal-mode methods respectively. Each part of the calculation uses separate configuration files which contain the specific instructions pertaining to the energy calculation method.

It is clear from the above description that, although the workflow involved in each calculation is rather long, the procedures that need to be implemented across a range of protease−ligand variants are very similar. Automation of such a workflow when studying an array of varying complexes thus saves a substantial amount of time.

2.2. Protocol Specification. We now describe the protocols that are used for each stage of the workflow described above and displayed in Figure 1.

2.2.1. Preparation Protocol. Protease and ligand coordinates are extracted from an initial crystal structure into separate files. Missing hydrogens are inserted on drug coordinates using the PRODRG tool.[19] Gaussian 03[20] is used to perform geometric optimization of all inhibitors at the Hartree−Fock level with 6-31G** basis functions. The restrained electrostatic potential (RESP) procedure, which is part of the AMBER9 package,[21] is used to calculate the partial atomic charges. Mutations on the protease and/or the natural substrate are incorporated using a protocol which invokes the standard "mutate" algorithm from the visualization package VMD[22] and also uses VMD to insert all missing hydrogens on the protease and the natural substrate if liganded. The protonation state of the catalytic dyad specified by the user is also implemented using the mutate algorithm in VMD. Mutations are implemented in ascending numerical order of the amino-acid residue number within this protocol and for the first and second chains of the protease sequentially.

The forcefield parameters for the inhibitors are completely described by the general AMBER force field (GAFF).[23] The standard AMBER force field for bioorganic systems (ff03)[24] is used to describe the protein parameters as well as those for the natural substrates.

The Leap module[25] in the AMBER 9 software package is then used to combine each apo-protease system with the ligand. Leap is also used to electrically neutralize each ligand-bound system, for which the number of ions required varies depending on the unique mutational sequence of the complex. The system is then solvated using atomistic TIP3P water[26] in a cubic box with at least 14 Å distance around the complex, resulting in a fully atomistic system of approximately 40 000 atoms. Finally, Leap is used to generate system-specific topology and coordinate files that are the prerequisites for subsequent simulation.

2.2.2. Mutation-Adaptive Equilibration Protocol. An array of sequential equilibration simulations are then run prior to any production simulation. Table 1 shows all of these stages. The molecular dynamics package NAMD2[27] is used throughout the production simulations as well as for the employment of minimization and equilibration protocols. Although, in principle, several MD codes could be utilized within BAC, the scalability afforded by NAMD2 easily allows larger molecular models to be integrated in the future.

Minimization is conducted using the conjugate gradient and line search algorithms available in NAMD2 for 2000 iterations and achieves a desired gradient tolerance of approximately 10 in each case. The long-range Coulomb interaction is handled using the particle mesh Ewald summation method (PME).[28] A nonbonded cutoff distance of 12 Å is used for all simulations. For the equilibration and subsequent production runs the SHAKE algorithm[29] is employed on all atoms covalently bonded to a hydrogen atom, allowing for an integration time step of 2 fs.

The system is gently annealed from 50 to 300 K over a period of 50 ps and then maintained in the isothermal−isobaric ensemble (NPT) thereafter at a target temperature of 300 K and target pressure of 1 bar using a Langevin thermostat and Berendesen barostat,[30] respectively. The system is equilibrated for 200 ps while maintaining the force constants

Table 1. Mutation-Adaptive Equilibration Protocol for HIV-1 Protease−Ligand Simulations[a]

			force constraint (kcal/(mol Å^2))[b]		
stage	process	duration (ps)	ligand	protease	solvent
eq 0	conjugate gradient minimization	2000 steps	4	4	0
eq 1	annealing 50 → 300 K	50	4	4	0
eq 2	dynamic solvation, NPT[c] ensemble	200	4	4	0
	mutation relaxation protocol (NPT)		M-region[d]	NM-region[e]	solvent
eq (2 + 1)	M1-region relaxation	50	0	4	0
eq (2 + 2)	M2-region relaxation	50	0	4	0
⋮	⋮	⋮	⋮	⋮	⋮
eq (2 + *n*)	M*n*-region relaxation	50	0	4	0
			ligand	protease	solvent
eq (2 + *n*) + 1	constraint relaxation (NPT)	50	3	4	0
eq (2 + *n*) + 2		50	2	4	0
eq (2 + *n*) + 3		50	1	4	0
eq (2 + *n*) + 4		50	0	4	0
eq (2 + *n*) + 5		50	0	3	0
eq (2 + *n*) + 6		50	0	2	0
eq (2 + *n*) + 7		50	0	1	0
eq (2 + *n*) + 8	unrestrained equilibration (NPT)	1400−50*n*	0	0	0

[a] See step 2 in Figure 1. Equilibration commences from the crystal structure-derived starting structure; the protocol is adapted from the work of Sadiq et al.[32] and Perryman et al.[33] [b] All constraints on hydrogen atoms are set to 0. [c] NPT ensemble: Langevin thermostat, target temperature = 300 K, coupling coefficient = 1/ps. Berendsen barostat, target pressure = 1 bar, pressure coupling constant = 0.1 ps. [d] M-region consists of all non-hydrogen ligand−protease atoms of residues within the 5 Å spheres centered on both mutant residues within the protease dimer. Inhibitors are treated as a single residue. [e] NM-region consists of all non-hydrogen ligand−protease atoms outside the M-region.

on the restrained atoms to allow (see Table 1) for thorough solvation of the complex and to prevent premature flap collapse.[31]

This is followed by a mutation relaxation protocol to allow optimal reorientation of all mutated amino acids.[11] The heavy atoms of each mutated amino acid and those of amino acids within a 5 Å surrounding region (M-region) of the dimeric mutation are completely relaxed sequentially for every dimeric mutation for a duration of 50 ps each. After each mutant region is relaxed for 50 ps, the heavy atoms of that region are again constrained by a force constant of 4 kcal/(mol Å^2) before iterating the procedure on the next mutation region. The mutation regions are selected in ascending numerical order of the mutated amino-acid residue number corresponding to the dimeric mutation. For example, if positions 48/148 and 90/190 are mutated, the first mutation region selected will contain any complete residues that are either partially or wholly within a 5 Å region around positions 48 and 148, while the second mutation region will be an identically defined region around positions 90 and 190.

This procedure is followed by a gradual force constant reduction on the ligand from 4 to 0 kcal/(mol Å^2) over a 200 ps period in equal stages of 1 kcal/(mol Å^2) and then a similar force constant reduction on the protease from 4 to 1 kcal/(mol Å^2) over a period of 150 ps. In the final stage of equilibration, all constraints are removed from the protease and the system allowed to evolve completely unrestrained up to a total duration of 2 ns. The length of this last stage therefore varies only according to the number of mutations that are incorporated in the system. In principle, the 2 ns duration of the protocol implies that up to 28 dimeric mutations can be incorporated into the system. However, this would result in the final stage of the simulation being of zero length. In practice therefore, provided a suitable length of final unrestrained equilibration, such as 200 ps, is maintained, the upper limit to the number of dimeric mutations that can be inserted is 24. This is far larger than the number of mutations that separate different HIV subtypes as well as being sufficient for all mutations within the HIV-1 clade.

2.2.3. Simulation Protocol. The production simulations for each system are also performed in the isothermal−isobaric ensemble (NPT) described above. Although the simulation length can be arbitrary, 10 ns is an indicative length to achieve convergence of binding free energies.[11] In this case, the simulation is decomposed into 10 1 ns simulations run sequentially and labeled sim1−sim10. The coordinates of the trajectories are recorded every 1 ps throughout all equilibration and production runs.

2.2.4. Free Energy Analysis Protocol. Free energy analysis of the production trajectories employs the single-trajectory MMPBSA method combined with a determination of the change in configurational entropy using the harmonic approximation of normal-mode analysis. The principles of these methods are well established and have been discussed by us and others.[11,34] Here, we describe the specific parameters employed in our approach.

The total free energy difference of binding is composed of the following terms:

$$\Delta G_b = \Delta G_{vdW}^{MM} + \Delta G_{ele}^{MM} + \Delta G_{pol}^{sol} + \Delta G_{nonpol}^{sol} - T\Delta S \quad (1)$$

where the first two terms on the right-hand side represent the van der Waals and electrostatic components of the gas-phase molecular mechanics free energy difference, respectively, the third term is the electrostatic/polar component of the solvation free energy, and the fourth term is the nonpolar component of solvation free energy, all calculated using the MMPBSA method. The last term is the contribution from the change in the configurational entropy (ΔS).

The average molecular mechanics free energy difference ΔG^{MM} is calculated using the SANDER module in AMBER 9, with no cutoff for the nonbonded energies. The AMBER PBSA module is used for the evaluation of the electrostatic free energy of solvation ΔG^{sol}_{pol}. A grid spacing of 0.5 Å is employed for the cubic lattice, the internal and external dielectric constants are set to 1 and 80, respectively, and 1000 linear iterations are performed. The nonpolar solvation free energyΔG^{sol}_{nonpol} is calculated from the solvent accessible surface area (SASA) using the MSMS program,[35] with a probe radius of 1.4 Å, the surface tension γ being set to 0.00542 kcal/(mol Å^2), and the off-set β, to 0.92 kcal/mol.

The changes in configurational entropy upon ligand association ΔS are estimated by an all-atom normal-mode analysis performed with the AMBER NMODE module. Prior to the normal mode calculations, the complex, receptor, and ligand are subjected to minimization with a distance-dependent dielectric constant $\varepsilon = 4r$ and convergence tolerance tighter than a root-mean-squared gradient of drms $= 10^{-4}$ kcal/(mol Å). Entropy calculations on all protease−ligand systems are averaged over equally spaced snapshots, extracted over the entire 10 ns of the production phase.

The mean of the binding free enthalpies and entropies of all the snapshots (N) used is computed and the standard error (σ) of the calculation is determined from the standard deviation (σ_{sd}) of the data set, where $\sigma = \sigma_{sd}/N^{1/2}$. Finally, the mean enthalpy and entropy are summed to obtain the binding free energy ΔG_b.

2.3. Protocol Accuracy. The default free energy analysis protocol adopted by BAC has already been used to accurately describe the free energy of binding of a small array of HIV-1 protease variants bound to saquinavir.[11] While being suitable for that study, the above energy analysis protocol cannot be assumed to be accurate, a priori, for any given protease−ligand complex. Indeed a much more extensive study incorporating a larger set of protease−ligand combinations will need to be undertaken to tune and enhance the protocol beyond its current state.

We outline potential enhancements that might be incorporated in the future. First, as mentioned above, BAC currently utilizes the single-trajectory approach in MMPBSA and normal-mode analysis. Such an approach is advantageous due to an exact cancelation of internal energies from the difference between the complex and the sum of the protein and ligand species, resulting in shorter convergence times.[7] However, it does not include the change in binding free energy resulting from possible conformational changes of the protease−ligand complex upon binding. If mutations affect this change to differing degrees, then accurate ranking of binding affinities will be enhanced by factoring in such changes, provided convergence criteria can still be met. To this end, an extensive investigation of the three-trajectory method, which allows for conformational changes to take place, may lead to enhancement of the current protocol.

Second, it is well-known that the accuracy of binding free energies computed from molecular dynamics is very sensitive to the extent of the conformational space explored by the simulations. Increased sampling of the conformational space can be achieved either by extending the timescale of the simulation, allowing a gradual exploration of different conformations, and/or by guiding the initial structure into an ensemble of conformational states, which are all significantly sampled. An example of this is explicit sampling of the multitude of rotameric states that might be adopted by the amino acid residues of the protease. Each distinct rotamer may have significantly differing contributions to the free energy. A suitable averaging of the effects of mutliple rotameric states may therefore lead to more accurate binding affinity results for a number of protease−ligand variants. The current method may therefore be enhanced in the future by incorporating such methodologies.

3. VALIDATION OF MUTATION-ADAPTIVE EQUILIBRATION PROTOCOL

The number of mutant sequences of potential clinical interest far exceeds the number of available crystal structures of both the bound and unbound protease. Therefore, in order to model any given protease structure, it is, in general, necessary to be able to routinely incorporate mutations into a given wildtype starting crystal structure. Existing crystal structures are a good template for the structural properties of their corresponding sequences, while dynamical properties are readily obtained from molecular dynamics simulations using these structures as a starting point.[1] However, it cannot be assumed that such crystal structures accurately describe the structural or dynamical properties of a differing mutant strain. Even though the tertiary structure of the HIV-1 protease has been shown to be particularly tolerant to mutations,[36] the structural orientation and flexibility of mutated amino acid side-chains may be significantly different to that of the wildtype structure. Furthermore, these rotameric changes may lead to substantially different structural and dynamical properties and ultimately result in the reduction of binding affinity in the case of drug resistant mutants. Mutation algorithms, such as the one in the VMD package utilized here, can convert amino acids into each other, thus providing an initial derived mutant structure. However, a robust mutation-adaptive equilibration protocol is then needed to ensure that the subsequent equilibration of an artifically generated mutant system from a wildtype results in an equilibrated system that describes similar structural and dynamical properties to that of an equilibrated crystal structure of the same mutant, if it existed.

In order to validate the effectiveness of the mutation-adaptive equilibration protocol described in section 2.2.2, we compared the structural and dynamical properties of an indinavir-bound multidrug resistant (MDR) mutant HIV-1 protease derived from two molecular dynamics simulations. Both simulations were performed using the protocol described in section 2.2.2. However, the first simulation used an existing crystal structure of the mutant (pdb: 1SGU), denoted 1SGUx, while the second used an identical mutant structure artificially derived from the wildtype complex (pdb: 1HSG), denoted 1HSGm. 1SGU differs from 1HSG by the following 10 dimeric mutations: K20R, V32I, L33F, M36I, I54V, L63P, A71V, V82A, I84V, L90M. Figure 2 shows the structure of the HIV-1 protease from 1SGUx, highlighting the positions of the mutations relative to 1HSG. The protease has a C_2-symmetric dimeric structure, each monomer containing 99 residues, labeled 1−99 and 101−199. Ten dimeric mutations therefore correspond to twenty mutations in the structure. The inhibitor has been removed from the image to improve clarity.

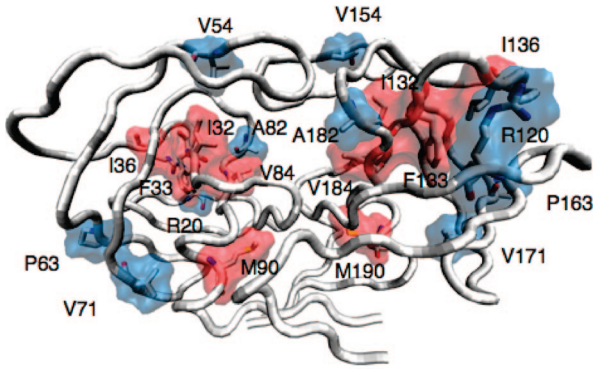

Figure 2. Structure of dimeric indinavir-bound multidrug resistant (MDR) HIV-1 protease as provided by the 1SGU crystal structure. Amino acids from the first and second monomers are labeled from 1−99 and 101−199, respectively. The inhibitor has been removed from the image to improve clarity. 1SGU has 10 dimeric mutations relative to the wildtype structure 1HSG; solvent exposed mutant residues (R20, V54, P63, V71, and A82 from the first monomer and R120, V154, P163, V171, and A182 from the second monomer) are shown in blue while those mutations buried in the protease (I32, F33, I36, V84, and M90 from the first monomer and I132, F133, I136, V184, and M190 from the second monomer) are shown in red.

As well as there being a significant degree of mutational deviation, the two structures are interesting to compare because a subset of the mutants are buried within the protease (red) while others are exposed to solvent (blue). Buried side-chains are in general more configurationally constrained than exposed side-chains; it is then expected that without a suitable relaxation protocol such artificially generated mutant side-chains would be less able to explore their optimal configurational space. Importantly, the choice of these two crystal structures thus allows the degree of rotameric flexibility of buried side-chains in response to the relaxation protocol and the subsequent convergence of side-chain conformations in the mutant-derived (1SGUx) and wildtype derived (1HSGm) structures to be assessed and compared.

3.1. Global and Local Structural Convergence. As mentioned in section 2.2.2, the mutate algorithm in VMD was used to incorporate the mutations within the starting crystal structure. We compared the global and residue decomposed root-mean squared deviations (RMSDs) of 1HSGm relative to 1SGUx both pre- and postequilibration, excluding hydrogen atoms. In the pre-equilibration assessment, the backbone atoms (C_α, N, C) of the 1HSGm and 1SGUx structures were aligned prior to calculating the RMSD. In the postequilibration assessment, the average backbone structures from the 10 ns production trajectories for both 1HSGm and 1SGUx were aligned prior to calculation of RMSD. Table 2 shows the global RMSDs for pre- and postequilibration structures as well as decomposition into mutated and nonmutated residues. The substantial decrease (∼1.5−2 Å) in all RMSDs postequilibration signifies convergence of the 1HSGm and 1SGUx structures as a result of their respective equilibration protocols. Furthermore, structural convergence occurs not only on a global protein level (1.74 Å RMSD reduction) but more importantly for the group of mutated residues (1.81 Å RMSD reduction).

Figure 3 shows the backbone-aligned residue decomposed RMSDs of 1HSGm and 1SGUx pre- and postequilibration depicted by the black and gray graphs, respectively. There is a substantial reduction in RMSD for almost all amino acid residues, especially for each of the mutated residues in both monomers of the protease. Moreover, both solvent exposed (blue) and, importantly, tightly packed (red) mutated residues exhibit reduction in RMSD to the same degree, implying that structural convergence is not restricted only to amino acid residues able to change conformation as a result of solvent interactions.

Table 2. Global, Mutated Residue, and Nonmutated Residue RMSDs of Backbone-Aligned Structures of 1HSGm and 1SGUx before and after Equilibration[a]

	RMSD (Å)	
	pre-equilibration	postequilibration
global	3.18	1.44
mutated residues	3.29	1.91
nonmutated residues	3.17	1.36

[a] In the pre-equilibration assessment, the original crystal structures were aligned, while in the post-equilibration assessment, the average backbone structures from the 10 ns production trajectories were aligned. The substantial decrease (∼1.5−2 Å) in all RMSDs postequilibration signifies convergence of the average structures as a result of the equilibration protocol.

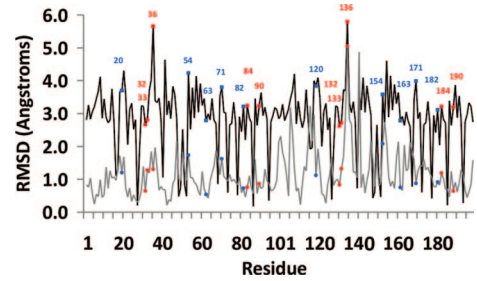

Figure 3. Residue decomposed root-mean squared deviation (RMSD) of the 1SGU crystal structure (1SGUx) with respect to the artifcially constructed mutant from the 1HSG crystal structure (1HSGm). Pre- and postequilibration RMSDs are shown in black and gray, respectively. Postequilbration RMSDs were attained using the average structures from the 10 ns production trajectories. The protein backbones were aligned prior to calculation of the RMSD. There is a substantial decrease in RMSD for most amino acid residues between pre- and postequilibration systems, particularly for both solvent exposed (blue) and protein constrained (red) mutated residues, indicating both global and local structural convergence as a result of the equilibration protocol.

3.2. Comparison of Conformational Flexibility. In addition to the convergence of average structural properties of the proteases, the conformational flexibility of both 1HSGm and 1SGUx was also compared using residue decomposed root-mean squared fluctuations (RMSFs) relative to the respective average structures using the entire 10 ns production trajectories (see Figure 4). Hydrogen atoms were again excluded in the analysis. The residue decomposed RMSF profiles of both 1HSGm (gray) and 1SGUx (black) were almost identical supporting the similar conformational flexibility of each of the amino acid residues across both of the systems. Furthermore, the structural flexibilities of both the solvent exposed (blue) and protein constrained (red) mutated residues were very similar in both systems, indicating converged local dynamics of both of the systems.

4. BAC ARCHITECTURE AND WORKFLOW MANAGEMENT

Conventionally, there are two general obstacles that impede automation of the workflow described in the previous

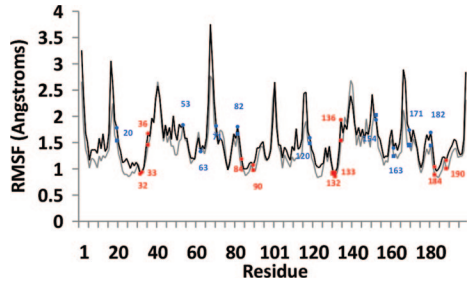

Figure 4. Residue decomposed root-mean squared fluctuations (RMSFs) relative to the average structure from the 10 ns production runs for 1SGUx (black) and 1HSGm (gray). The RMSF profiles were almost identical indicating a similar conformational flexibility of all residues postequilibration as well as, importantly, both the solvent exposed (blue) and protein constrained (red) mutated residues.

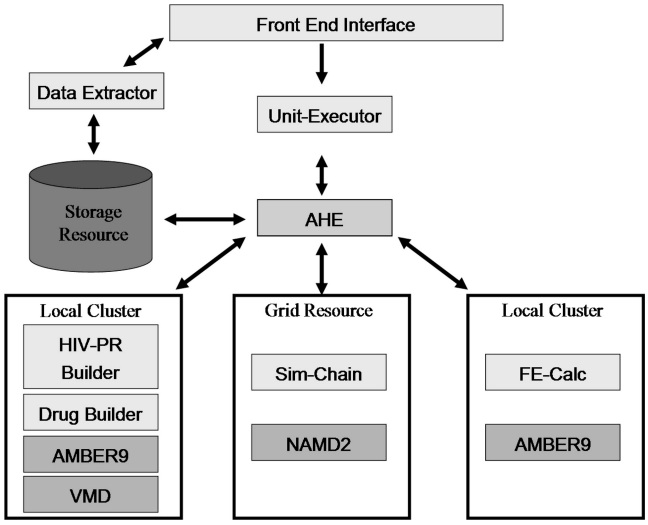

Figure 5. Architecture of the BAC. Simulation workflow is managed by the Unit-Executor, a Perl script designed to utilize the application hosting environment (AHE) middleware. The components of the workflow, namely model construction, simulation, and postproduction analyses, are implemented by the HIV-PR Builder, Sim-Chain, and FE-Calc applications, respectively. AHE fully automates the workflow through the execution of each component and the subsequent marshaling of files across distributed HPC resources.

section. First, all of the preparation files required for a simulation-ready model and associated configuration files necessary for the execution of the chain of simulations, as well as for the postproduction calculation of the binding free energy, need to be generated. Second, the intensive computational requirement of molecular simulations, in general, requires them to be implemented on HPC resources. Once the set of files required for a simulation has been generated, they are manually transferred to a HPC resource, where simulations need to be submitted using an appropriate job submission script. After the computation has completed, subsequent output data then needs to be marshalled back to an appropriate storage resource for postprocessing.

The architecture of BAC has been designed in a manner which overcomes these two obstacles to automation (see Figure 5) and which facilitates its use, in general, with HPC and across grid resources. Essential to the full automation conferred by BAC is the utilization of the application hosting environment (AHE),[37] which manages the workflow around various computational resources. One aspect of the functionality afforded by AHE is that job submission can be handled through a command line interface on the client-side resource. Perl scripts can then be used to construct workflows to manage the order in which a series of simulations is conducted. Within BAC, the "Unit-Executor" is an example of such a Perl script and is responsible for managing the calculation of a single, uniquely defined protease−ligand sequence, termed a "unit". The Unit-Executor is executed from the front-end command line interface to the client-side resource.

BAC decomposes the workflow of a complete free energy calculation into three main components: (a) building of a model, (b) MD equilibration and simulation of the model, and (c) postproduction analysis through which the free energy is calculated. These are implemented by the "HIV-PR Builder", "Sim-Chain", and "FE-Calc" applications, respectively (see Figure 5). In addition, the BAC contains a module, the "Drug Builder", run one time only for each inhibitor, which can be used to assign the partial charges and topological information for a novel ligand. We will describe each of these applications in more detail later, but for now, we turn our attention to the overall workflow management of a single calculation.

Upon initiating the Unit-Executor, the AHE is used to run the HIV-PR Builder program on a resource that has the AMBER 9[21] and VMD[22] software applications installed. The HIV-PR Builder subsequently builds all the presimulation and configuration files necessary for all stages of the equilibration and production simulations, prior to any simulation taking place. In addition to this it spawns the Sim-Chain program. AHE then stages all of the required files to a compute resource, including the spawned instance of the Sim-Chain program, which is subsequently executed on that resource. It is necessary for the compute resource to already have the NAMD2[27] molecular dynamics software, used by Sim-Chain, compiled on it. When each component of the equilibration/simulation run is complete, output data is staged back to a storage resource; the Unit-Executor then checks for successful completion before re-executing the Sim-Chain program for the next component of the simulation. When all stages in the simulation are complete and have been staged back to the data storage resource, AHE then executes the FE-Calc program. The FE-Calc program generates the input and execution files required for the enthalpy and entropy calculations, implemented respectively using MMPBSA and normal-mode analysis methods described in the previous section, and then submits them for calculation. Once the calculations are complete, the calculation output files are staged back to the storage resource and the Unit-Executor terminates. The binding free energy data can then be directly viewed from storage or extracted in a convenient way using the "Data Extractor" program, which runs on the front-end command line interface.

The modular design of BAC allows specific components, such as HIV-PR Builder, Sim-Chain, and FE-Calc applications to be used independently at the cost of complete automation. In such a scenario, the presimulation and configuration files required for a specified HIV-1 protease−ligand variant are still automatically generated, affording considerable speed up over manual preparation. However, the user then needs to marshall data from resource to resource and submit jobs manually. For scientists interested in implementing changes to the default equilibration, simulation

and/or free energy calculation protocols, but who wish the relational structure of a set of simulations to be preserved, this may nevertheless be of considerable benefit.

5. HIV-PR BUILDER, DRUG BUILDER, AND SIM-CHAIN

The HIV-PR Builder application automates the preparation of a simulation-ready molecular dynamics model of HIV protease, either in complex with a ligand or in the apo form. It consists of a set of Perl scripts which include the generation and execution of "tcl" scripts in the VMD application and "tleap" commands in the AMBER 9 software package.

To correctly run the HIV-PR Builder, which is executed from the command line, it is necessary to specify the forcefield, the initial pdb crystal structure, the complexed status of the protease (either drug-bound, substrate-bound, or apo), the ligand identity, if bound, and the protonation state of the catalytic dyad. In addition, optional parameters with default values may be specified, such as any desired mutations relative to the crystal structure chosen and/or mutations relative to the peptide substrate selected, as well as the size of the periodic solvation box and the nonbonded cutoff distance.

The builder contains a host of over 200 premodified pdb structures of HIV protease with atomic nomenclature in the AMBER format, the two chains of the protease being designated A and B sequentially. These are listed in the Supporting Information. Atomic coordinates have been left unaltered. All of these pdbs can be used as the basis for apo-protease simulations. For simulations of protease−ligand complexes, a subset of these structures is used. For the protease complexed to the nine FDA inhibitors, saquinavir (SQV), ritonavir (RTV), indinavir (IDV), nelfinavir (NFV), lopinavir (LPV), amprenavir (APV), atazanavir (AZV), tipranavir (TPV), and darunavir (DRV), the crystal structures 1FB7, 1HXW, 1HSG, 1OHR, 1MUI, 1HPV, 2AQU, 2O4P, and 2HS1 are used respectively. For the protease complexed to the natural substrates MA-CA, CA-p2, p2-NC, NC-p1, p1-p6, RT-RH, and RH-IN the crystal structures 1KJ4, 1F7A, 1KJ7, 1TSU, 1KJF, 1KJG, and 1KJH are used, respectively.

The coordinates of drugs and substrates have been pre-extracted from these structures. Furthermore, partial drug charges have also been predetermined using the "Drug Builder" application which semiautomates the partial charge assignment protocol described in the previous section. Therefore, even though the charge information has been predetermined by the Drug Builder for the above-specified set of inhibitors, in principle, the Drug Builder can be used to generate such information for any given ligand, provided a crystal structure of the ligand exists. Once generated, the generic topology, structure and charge information for the ligand is transferred to the HIV-PR Builder where it can be accessed repeatedly to build any protease variant bound to the novel ligand, provided the protease variant is generated from the corresponding crystal structure from which the ligand structure is extracted. The Drug Builder is described in more detail in the Supporting Information.

Figure 6 shows a schematic representation of the processes implemented by the HIV-PR Builder. Once executed, the HIV-PR Builder splits the two monomeric chains (A and B) of the protease in the pdb specified into separate

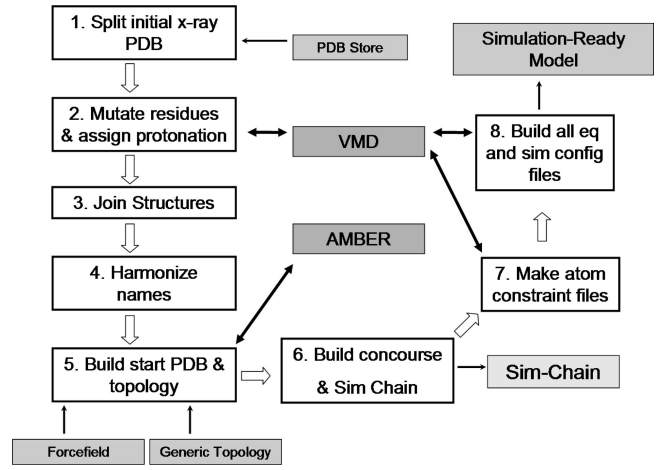

Figure 6. Schematic representation of the HIV-PR Builder application. The workflow processes (labeled 1−8) utilize a premodified store of PDB structures, together with forcefield and topology files and interface with the VMD and AMBER applications to construct the core set of files necessary for subsequent simulations.

coordinate files alongside any ions and all crystallographic water molecules. If a substrate has been specified, this too is extracted into another file. Each of these files is then subject to the incorporation of mutations by means of a tcl script run on the VMD command line interface and generated by a Perl script. The protonation of the dyad is similarly assigned. The separate coordinate files are merged and atomic nomenclature, previously assigned by VMD, is converted back into AMBER nomenclature. A source file, with instructions to add a drug, neutralizing ions and water molecules as well as to write starting topology and coordinate files, is generated by a Perl script and subsequently executed using the "tleap" module of AMBER 9. Following this, the main directory, termed the "concourse", and its subdirectories are generated, into which all subsequent data corresponding to the specified unique HIV-1 protease−ligand combination will be stored. Simulation start files are transferred to a concourse subdirectory. The Sim-Chain application resides within the HIV-PR Builder and consists of a collection of modified job submission scripts for a range of HPC resources. It is subsequently copied to another concourse subdirectory.

The minimization, equilibration, and production simulations implemented by the BAC make use of NAMD2. All equilibration and simulation stages require individual configuration files in the NAMD format. Furthermore, several components of the equilibration stage require constraint files to be accessible. These specify the atoms in the system that will be constrained with a certain force constant (see Table 1). However, as all of the details of equilibration and simulation configuration are predeterminable and follow the protocol described in Table 1, the builder generates all constraint files and configuration files at this stage. The tcl scripts are generated and executed by a Perl script using VMD to construct the appropriate constraint files.

Generation of equilibration and simulation configuration files proceeds as follows. The cell basis vectors are computed using a tcl script in VMD; these are then used to determine optimal PME values for the initial stage of the equilibration. Temperature, pressure, and constraint settings are automatically written to each configuration file as well as the number

of simulation timesteps. The number of equilibration stages varies according to the number of mutants incorporated into a system. For each mutation, there is an additional equilibration configuration file during execution of which the constraints around the mutation are relaxed (see Table 1). However, the total number of timesteps for the whole equilibration phase remains constant (2 ns). Input and output files are specified in the configuration file in a systematic manner which ensures that the output of one stage is named as the input of the following and all file-paths are assigned relative to the concourse directory. The only differences in the simulation configuration files are the names of the input and output files which are written in a similar systematic manner.

As mentioned before, 10 ns has thus far been an indicative production length to achieve accurate binding affinities. However, in general, it may not be enough to achieve adequate sampling for any given mutant protease−ligand complex. The facility to perform simulations by default up to 100 ns has therefore been incorporated into the BAC through construction of the relevant simulation configuration files as well as job-submission scripts within the Sim-Chain application, while simulations for any further period of time can easily be implemented by small modifications to the Sim-Chain application and further generation of more simulation configuration files.

Once generation of all presimulation files is complete, the modular design of each specific protease−ligand unit, contained entirely within its respective concourse directory, facilitates its transfer to different compute resources. The Sim-Chain application can then be run by executing each individual job submission script from within the appropriate concourse subdirectory. This is possible as the job submission scripts also utilize a naming scheme relative to the concourse unit. Each submission script is designed to sequentially submit a range of equilibration/simulation stages, executed by NAMD2 and, by default, using 32 processors on the host compute resource. Submission scripts for the Oxford, Manchester, and Leeds computing nodes of the UK National Grid Service (www.ngs.ac.uk) as well as the Lonestar and Ranger machines at the Texas Advanced Computing Center on the US TeraGrid (www.teragrid.org) are currently available within Sim-Chain. Current processor speeds on the NGS yield a time of approximately 6 h/ns of simulation using 32 processors, leading to a total compute time of around 72 h for each system. When not called by AHE, the user must initiate each script manually after checking that the set of simulations has terminated correctly. When interfaced with AHE, the Unit-Executor uses AHE to check for successful completion, before automated execution of the following job submission script.

6. FE-CALC APPLICATION

The FE-Calc application executes MMPBSA and normal-mode analysis calculations using the MMPBSA module of the AMBER 9 software package. It consists of a Perl script that generates all the input files necessary for a calculation, subsequently submitting these calculations to the designated compute resources.

Figure 7 shows the processes implemented by the application. FE-Calc takes in similar input to the HIV-PR Builder

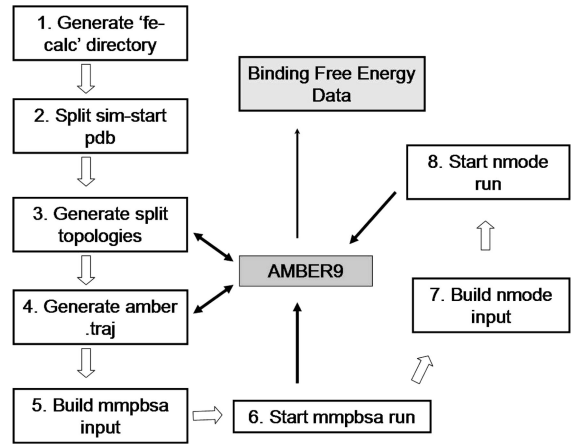

Figure 7. Schematic representation of the FE-Calc application. The workflow processes (labeled 1−8) utilize the AMBER application to (i) generate topologies, (ii) generate trajectories in the AMBER format, and (iii) implement the MMPBSA and normal-mode analyses, resulting in output of binding free energy data.

application and uses this to identify the unique concourse unit to process. An "fe-calc" concourse subdirectory is produced (step 1) and all subsequently generated files are written therein. The MMPBSA module requires separate topology files to be written for the complex, ligand, and receptor as well as input trajectories written in the AMBER "traj" format. The simulation-ready starting pdb for the original molecular dynamics simulation is split into three separate pdbs, for the complex, ligand, and receptor (step 2). These are used to generate separate topologies using a tleap source file written by the Perl script (step 3). Amber trajectories are generated by executing source files written by FE-Calc for the PTRAJ module of AMBER 9 (step 4). MMPBSA and normal-mode analyses use different parameters and are thus implemented from separate input files. These are generated from existing templates (steps 5 and 7) and subsequently modified by the application. FE-Calc next determines appropriate atom numbers for the beginning and end of each molecular species and assigns these along with the snapshot frequency and output filenames to each input file. Generic job submission scripts are then used to launch both the MMPBSA and NMODE calculations on the compute resource (steps 6 and 8). Steps 4−8 are executed once for each of the 10 1 ns trajectories obtained in the production simulations. In this way, it is possible to parallelize the MMPBSA and normal mode computation for the entire trajectory across 20 simultaneously used processors, each one carrying out either an MMPBSA or normal mode calculation for a specific nanosecond of trajectory data.

MMPBSA analysis then takes approximately 3 h while normal-mode analysis takes up to 24 h while using 20 opteron CPUs. Compounded with the total simulation time (72 h), the turn around time for obtaining a binding affinity value for a single protease−ligand complex using BAC, when the necessary resources are available, is approximately 96 h.

7. DATA EXTRACTOR

As mentioned before, once the calculations are complete, the data can be viewed directly from storage or extracted in a convenient way using the Data Extractor program. The Data Extractor is composed of a set of perl scripts which

further manipulates the raw data ouptut from both the MMPBSA and normal-mode analyses into a variety of formats. These include the following:

• A decomposed output of each energetic/entropic component from every snapshot analyzed, concatenated into a time series. This facilitates the observation of fluctuations in the energy components.

• A cumulative forward and reverse time-average of both the enthalpic and entropic components of binding, including the standard error. This facilitates the assessment of convergence of the free energy as well as allowing the user to see if the latter part of a trajectory may be more well equilibrated than the earlier part.

• A mean and standard error for each component of the calculation across the maximum production timescale used in the simulation.

The variety of data generated by the scripts contained within the Data Extractor are examples of data types that may be relevant to the user in the analysis of a given system. Indeed many other scripts can be envisaged which further analyze the trajectories both in qualitative and quantitative ways. However, given the diversity of methods by means of which a trajectory may be investigated, the user is at liberty to further develop any array of potential extraction scripts, utilizing both the free-energy data as well as the raw conformation data across the production trajectory for subsequent analysis.

8. EXTENDIBILITY OF THE BAC

In principle, the BAC is extendible to any set of biomolecular species. However, there are several obstacles to constructing an all-purpose binding affinity calculator for any possible protein−ligand complex that stem from a number of sources. First, the nonstandard component nomenclature for structures in the protein data bank[18] means that the naming convention for the components of any novel biomolecular structure has to be altered and standardized before the structure can be subsequently processed by the BAC software. For example, there is no standard chain-naming convention across all crystal structures of HIV-1 protease. As a result of this, all the crystal structures available in the BAC had to be first modified such that the first and second monomers were all labeled A and B, respectively. It is currently impossible to design an all-purpose calculator for any general protein in the data bank and instead a certain amount of work must be done to standardize the naming scheme for any designated protein structure.

Second, the preparation protocol utilized for the HIV-1 protease is by no means generic. In general, the degree of minimization and the amount of time required to equilibrate a protein are highly dependent on the type and size of protein as well as its proximity to its native state. The preparation protocol which has been carefully devised for HIV-1 protease and validated in section 3 is only applicable to the protease and a priori is not applicable to any other system. A significant degree of investigation needs to be undertaken to determine the optimal preparation protocol for each novel system that might be integrated into BAC in the future. Once this is done, alongside the standardization of naming components of that system, BAC could easily be extended to incorporate MD simulations and/or binding affinity calculations for novel protein−ligand complexes. Indeed work is currently underway to extend BAC to ligand-binding studies and apo-simulations of the HIV-1 reverse transcriptase and integrase enzymes as well as the tyrosine kinase domain of the epidermal growth factor receptor (EGFR).

9. CONCLUSION

We have developed a Binding Affinity Calculator (BAC), a grid based tool that automates all of the stages of model preparation, equilibration, simulation, postprocessing, and data-marshaling around available computing resources required to compute free energies of binding for HIV-1 protease−ligand complexes. Such automation enables the molecular dynamics methodology to be used at a level of throughput not realistically achievable by manual methods.

BAC has already been used to rapidly determine and accurately rank the binding affinities of saquinavir bound to wildtype and mutant HIV-1 proteases[11] Although not so far tested on a clinically large array of protease−ligand variants, the excellent quantitative ranking of drug resistant mutants exhibited in that study is encouraging for future studies on different drug-bound protease variants. Furthermore, the infrastructure afforded by BAC means that there is now a way to rapidly perform large numbers of compute-intensive molecular simulations of HIV-1 protease as well as automated protease−ligand binding affinity calculations.

Given access to sufficient compute resources, BAC is able to confer quantitative information regarding drug resistance at the molecular level within 96 h. Such a timescale opens the way for binding affinity calculations to have a direct impact on biomedical/clinical decision making. We have recently described the potential for BAC to be utilized in patient-specific decision support for optimizing therapy as well as the assessment of response to therapy.[16] The BAC can be integrated with existing[14] and future[38] "decision-support" software that interpret the complexity of emergent mutational patterns[13] from genotypic assays of patients already on antiretroviral therapy as well as from treatment-naive patients. As the efficacy of such software is limited by both the size of existing clinical databases and the failure to use the genotypic information optimally, BAC can deductively confer additional information at the molecular level on sequences not characterized well by such software, within a relevant clinical timescale.

Future work is likely to require fine-tuning of the parameters used in the methodology encapsulated by BAC and may call for the investigation of other dynamical aspects of the HIV-1 protease system. The changes required for this can be straightforwardly implemented within BAC without having to manually construct an entirely new set of simulations each time a new study is conceived. Further development of BAC may also extend the protein−ligand space simulated to other target proteins such as reverse transcriptase and/or integrase.

ACKNOWLEDGMENT

We are grateful to EPSRC for funding much of this research through RealityGrid grant GR/R67699. Our work was partially supported by the National Science Foundation under NRAC grant MCA04N014, utilizing the Lonestar and Ranger machines at the Texas Advanced Computing Center

(TACC). We are grateful to Jay Boisseau and the TACC support team for facilitating our early friendly user access to and subsequent discretionary allocation of service units on Ranger in late 2007 and early 2008. We also wish to thank the UK NGS for providing access to their resources. This research has been partially supported by the EU-funded ViroLab project (IST-027446).

Supporting Information Available: Tables displaying the crystal structures that can be used for binding affinity calculations within BAC and the crystal structures available for apo-protease simulations and a description of the Drug Builder application. Reference 20 is also cited in full. This material is available free of charge via the Internet at http://pubs.acs.org.

REFERENCES AND NOTES

(1) Karplus, M.; Kuriyan, K. Molecular dynamics and protein function. *Proc. Natl. Acad. Sci. U.S.A.* **2005**, *102*, 6679–6685.
(2) Coveney, P. V. Scientific grid computing. *Philos. Trans. R. Soc. A* **2005**, *363*, 1707–1713.
(3) Ohtaka, H.; Schon, A.; Freire, E. Multidrug resistance to HIV-1 protease inhibition requires cooperative coupling between distal mutations. *Biochemistry* **2003**, *42*, 13659–13666.
(4) Todd, M. J.; Luque, I.; Velazquez-Campoy, A.; Freire, E. Thermodynamic basis of resistance to HIV-1 protease inhibition: Calorimetric analysis of the V82F/I84V active site resistant mutant. *Biochemistry* **2000**, *39*, 11876–11883.
(5) Maschera, B.; Darby, G.; Palu, G.; Wright, L. L.; Tisdale, M.; Myers, R.; Blair, E. D.; Furfine, E. S. Human immunodeficiency virus: Mutations in the viral protease that confer resistance to saquinavir increase the dissociation rate constant of the protease-saquinavir complex. *J. Biol. Chem.* **1996**, *271*, 33231–33235.
(6) Rick, S. W.; Topol, I. A.; Erickson, J. W.; Burt, S. K. Molecular mechanisms of resistance: Free energy calculations of mutation effects on inhibitor binding to HIV-1 protease. *Protein Sci.* **1998**, *7*, 1750–1756.
(7) Lepsik, M.; Kriz, Z.; Havlas, Z. Efficiency of a second-generation HIV-1 protease inhibitor studied by molecular dynamics and absolute binding free energy calculations. *Proteins: Struct., Funct., Bioinf.* **2004**, *57*, 279–293.
(8) Zoete, V.; Michielin, O.; Karplus, M. Protein-ligand binding free energy estimation using molecular mechanics and continuum electrostatics. Application to HIV-1 protease inhibitors. *J. Comput.-Aided Mol. Des.* **2003**, *17*, 861–880.
(9) Wang, W.; Kollman, P. A. Computational study of protein specificity: The molecular basis of HIV-1 protease drug resistance. *Proc. Natl. Acad. Sci. U.S.A.* **2001**, *98*, 14937–14942.
(10) Chen, X.; Weber, I. T.; Harrison, R. W. Molecular dynamics simulations of 14 HIV protease mutants in complexes with indinavir. *J. Mol. Model.* **2004**, *10*, 373–381.
(11) Stoica, I.; Sadiq, S. K.; Coveney, P. V. Rapid and accurate prediction of binding free energies of saquinavir-bound HIV-1 proteases. *J. Am. Chem. Soc.* **2008**, *130*, 2639–2648.
(12) Snoek, J.; Riva, C.; Steegen, K.; Schrooten, Y.; Maes, B.; Vergne, L.; Laethem, K. V.; Peeters, M.; Vandamme, A. M. Optimization of a genotypic assay applicable to all human immunodeficiency virus type 1 protease and reverse transcriptase subtypes. *J. Virol. Methods* **2005**, *128*, 47–53.
(13) Wu, T. D.; Schiffer, C. A.; Gonzales, M.; Taylor, J.; Kantor, R.; Chou, S.; Israelski, D.; Zolopa, A. R.; Fessel, W. J.; Shafer, R. W. Mutation patterns and structural correlates in human immunodeficiency virus type 1 protease following different protease inhibitor treatments. *J. Virol.* **2003**, *77*, 4836–4847.
(14) Kantor, R.; Machekano, R.; Gonzales, M. J.; Dupnik, K.; Schapiro, J. M.; Shafer, R. W. Human immunodeficiency virus reverse transcriptase and protease sequence database: an expanded data model integrating natural language text and sequence analysis programs. *Nucleic Acid Res.* **2001**, *29*, 296–299.
(15) Johnson, V. A.; Brun-Vezinet, F.; Clotet, B.; Conway, B.; Kuritzkes, D. R.; Pillay, D.; Schapiro, J.; Telenti, A.; Richman, D. Update of the drug resistance mutations in HIV-1: 2005. *Int. AIDS Soc.—USA* **2005**, *13*, 51–57.
(16) Sadiq, S. K.; Mazzeo, M. D.; Zasada, S. J.; Manos, S.; Stoica, I.; Gale, C. V.; Watson, S. J.; Kellam, P.; Brew, S.; Coveney, P. V. Patient-specific simulation as a basis for clinical decision-making *Philos. Trans. R. Soc. A* **2008**, *366*, 3199.
(17) Sloot, P.; Coveney, P.; Bubak, M.; Vandamme, A.-M., Nuallin, B. Ã.; van de Vijver, D.; Boucher, C. Multi-science decision support for HIV drug resistance treatment *Global Healthgrid: e-Science Meets Biomedical Informatics—Proceedings of HealthGrid 2008*; IOS Press: Amsterdam, 2008; pp 188−198.
(18) Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyalov, I. N.; Bourne, P. E. The protein data bank. *Nucleic Acid Res.* **2000**, *28*, 235–242.
(19) Schuettelkopf, A. W.; van Aalten, D. M. F. PRODRG - a tool for high-throughput crystallography of protein-ligand complexes. *Acta Crystallogr. Sect D: Biol. Crystallogr.* **2004**, *D60*, 1355–1363.
(20) Frisch, M. J *Gaussian 03*; revision C.02 2004, Gaussian, Inc.: Wallingford, CT, 2004.
(21) Case, D. A.; Cheatham, T.; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M., Jr.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668–1688.
(22) Humphrey, W.; Dalke, A.; Schulten, K. VMD - Visual molecular dynamics. *J. Mol. Graphics* **1996**, *14*, 33–38.
(23) Wang, J.; Wolf, R. M.; Case, D. A.; Kollman, P. A. Development and testing of a general AMBER force field (GAFF). *J. Comput. Chem.* **2004**, *25*, 1157–1174.
(24) Duan, Y.; Wu, C.; Chowdhury, S.; Lee, M. C.; Xiong, G.; Zhang, W.; Yang, R.; Cieplak, P.; Luo, R.; Lee, T. A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations. *J. Comput. Chem.* **2003**, *24*, 1999–2012.
(25) Schafmeister, C. E. A. F.; Ross, W. S.; Romanovski, V. *LEaP 1995*; University of California, San Francisco, CA, 1995.
(26) Jorgensen, W. L.; Chandrasekhar, J.; Madura, J. D.; Impey, R. W.; Klein, M. L. Comparison of simple potential functions for simulating liquid water. *J. Chem. Phys.* **1983**, *79*, 926–935.
(27) 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.
(28) Essmann, U.; Perera, L.; Berkowitz, M. L.; Darden, T. A smooth particle mesh Ewald method. *J. Chem. Phys.* **1995**, *103*, 8577–9593.
(29) Ryckaert, J. P.; Ciccotti, G.; Berendsen, H. J. C. Numerical integration of the Cartesian equations of motion of a system with constraints: Molecular dynamics of n-alkanes. *J. Comput. Phys.* **1977**, *23*, 327–341.
(30) Berendsen, H. J. C.; Postma, J. P. M.; van Gunsteren, W. F.; DiNola, A.; Haak, J. R. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.* **1984**, *81*, 3684–3690.
(31) Meagher, K. L.; Carlson, H. A. Solvation influences flap collapse in HIV-1 protease. *Proteins: Struct., Funct., Bioinf.* **2005**, *58*, 119–125.
(32) Sadiq, S. K.; Wan, S.; Coveney, P. V. Insights into a mutation-assisted lateral drug escape mechanism from the HIV-1 protease active site. *Biochemistry* **2007**, *46*, 14865–14877.
(33) Perryman, A. L.; Lin, J.; McCammon, J. A. HIV-1 protease molecular dynamics of a wild-type and of the V82F/I84V mutant: Possible contributions to drug resistance and a potential new target site for drugs. *Protein Sci.* **2004**, *13*, 1108–1123.
(34) Wang, W.; Donini, O.; Reyes, C. M.; Kollman, P. A. Biomolecular simulations: Recent developments in force fields, simulations of enzyme catalysis, protein-ligand, protein-protein, and protein-nucleic acid noncovalent interactions. *Annu. Rev. Biophys. Biomol. Struct.* **2001**, *30*, 211–243.
(35) Sanner, M. F.; Olson, A. J.; Spehner, J. C. Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers* **1996**, *38*, 305–320.
(36) Zoete, V.; Michielin, O.; Karplus, M. Relation between sequence and structure of HIV-1 protease inhibitor complexes: A model system for the analysis of protein flexibility. *J. Mol. Biol.* **2002**, *315*, 21–52.
(37) Coveney, P. V.; Saksena, R. S.; Zasada, S. J.; McKeown, M.; Pickles, S. The application hosting environment: Lightweight middleware for grid-based computational science. *Comput. Phys. Commun.* **2007**, *176*, 406–418.
(38) Sloot, P. M. A.; Tirado-Ramos, A.; Altintas, I.; Bubak, M.; Boucher, C. From molecule to man: Decision support in individualized e-health. *Computer* **2006**, *39*, 40–46.

CI8000937