To appear: *Future Generation Computer Systems* **15**, Elsevier (1999)

A VIRTUAL ENVIRONMENT FOR STEERED MOLECULAR DYNAMICS†

JAN F. PRINS[1], JAN HERMANS[2], GEOFFREY MANN[2], LARS S. NYLAND[1], MARTIN SIMONS[1]

[1]Department of Computer Science,
[2]Department of Biochemistry and Biophysics,
University of North Carolina,
Chapel Hill, NC 27599-3175, USA
(prins@cs.unc.edu)

Abstract

A molecular dynamics simulation approximates the motion of atoms in a system of molecules over short intervals of simulated time, typically on the order of picoseconds to nanoseconds. Such simulations may run for days or weeks on a computer when used to investigate the dynamic behavior of small proteins in biological systems. By adding additional restraints, a simulation may be "steered" to observe the possibility of particular behaviors or to eliminate others over shorter timescales. We have developed the Steered Molecular Dynamics (SMD) system to interactively place and observe the effects of restraints in a running dynamics simulation. In this article we describe an application of SMD to the extraction of small ligands from proteins, and an immersive virtual 3D environment through which the SMD system can be operated. The virtual environment is constructed using the Protein Interactive Theater (PIT) system, a head-tracked stereo workspace for two users.

KEYWORDS: computational steering, virtual environments, molecular dynamics.

1. Introduction

Molecules may be described at many levels of detail ranging from the quantum mechanical to the statistical mechanical. The appropriate level of detail depends on the questions to be investigated. For biological molecules, many important questions concern the dynamic behavior of proteins over time and how these behaviors are changed in the presence of other proteins and drug molecules. Questions of this variety have been investigated using statistical mechanical models in which each atom in a system of molecules reacts to a variety of forces. These include bond length, bond angle and dihedral angle constraints, the relatively short-range van der Waals forces, and the quite longer-range forces due to electrostatic interactions. The combined forces give rise to equations of motion that may be numerically integrated to approximate the energies in the system and provide an account of the molecular dynamics.

However, a number of complications arise in molecular dynamics (MD) simulations based on this approach. The first is that biological molecules of interest are typically proteins of substantial size, containing thousands of atoms. Moreover the natural environment of such proteins consists of other proteins as well as solvent, largely in the form of water molecules. A complete system of atoms to be simulated may number in the tens of thousands and may require periodic boundary conditions or other far-field electrostatic models to obtain consistent behavior. For example, Fig. 1 shows T4-lysozyme, a protein involved in the breakdown of polysaccharides, surrounded by water. The protein contains 2603 atoms while the solvent adds another 5837 water molecules for a total of 20114 atoms. Each integration timestep of this system involves considerable computation, about 10^8 floating point operations in this case.

†This work supported in part by NIH National Center for Research Resources grants RR08102 (UNC/Duke/NYU Computational Structural Biology Resource) and RR02170 (UNC Molecular Graphics Resource).

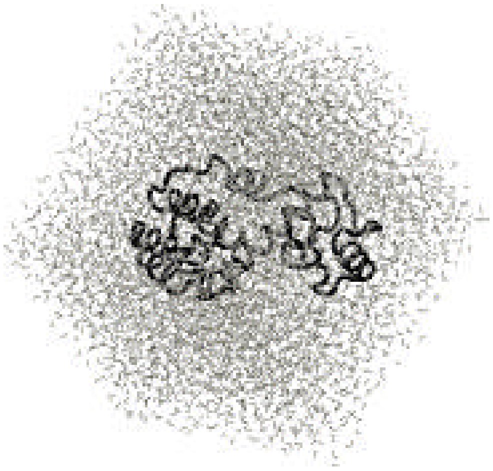


Fig. 1: T4-lysozyme in solvent. The protein backbone is shown as a dark tube; side chains are not shown. The water molecules are shown in light gray. The simulation uses periodic boundary conditions in a simulation volume of size 64Å × 58Å × 58Å, containing 20116 atoms.

The second complication concerns the timescale of the simulation. To accurately model time-dependent behavior requires the integration timestep to be sufficiently small to resolve the highest frequency vibrational modes in the system. These result from stretching vibrations of bonds between hydrogen and other atoms with a 10 fs period (fs = 10^{-15} seconds). Even with sophisticated numerical integration techniques, integration timesteps for molecular dynamics typically remain restricted to 1-2 fs.

The characteristic timescales of different motions found in biomolecules are shown in Fig. 2. For a system of modest size, 10^{-9} seconds of simulated time may take several months of computation on a single-processor scientific workstation. A high-performance parallel computer may reach simulation rates that are 10-1000 times faster, but simulated times above a microsecond, including the timescale of protein folding, remain altogether inaccessible.

In view of the extremely long simulation times required to observe many important behaviors, there is increasing interest in the idea of a *steered* dynamics simulation, in which external forces or restraints are added to the system in order to study behaviors that might take too long to appear, or to appear often enough, in an unrestrained MD simulation.

As an analogy, consider the question of whether an unpowered boat can drift a course from open sea into a harbor. A dynamics simulation that includes wind, currents and wave action in the general direction of the harbor may eventually place the boat in the harbor, but might also sample long periods of time during which the boat has missed the harbor entrance and is bouncing off a nearby shore. We may “steer” this nautical simulation by introducing a small force pulling the boat to the harbor entrance. Great care must be taken in the interpretation of such a steered simulation. Clearly the outcome is suspect if a steering force is introduced that drags the boat over the shore or overwhelms a current that would otherwise be too strong. On the other hand, the magnitude of the force required for the simulation to succeed can give some insight into the likelihood of an unsteered entrance.

Steering forces have long been incorporated in MD simulations where they are typically referred to as restraints, typically in computations of potentials of mean force, i.e., of free energy profiles along some specified coordinate [1, 20]. Restraints can be extremely difficult to express *a priori* when their direction and strength must vary with the local state of the simulation. Consequently we have constructed the SMD system [16] to enable restraints to be expressed in a graphical model of the system, and to couple the MD simulation to this graphical model, so that the dynamics may be viewed directly and restraints placed interactively.

Interaction with a running MD simulation at rates natural for a human operator correspond to very small simulated time durations. Since such short timescales fall far short of the time needed for a protein to naturally exhibit many organized behaviors, there are two approaches to steered simulations.

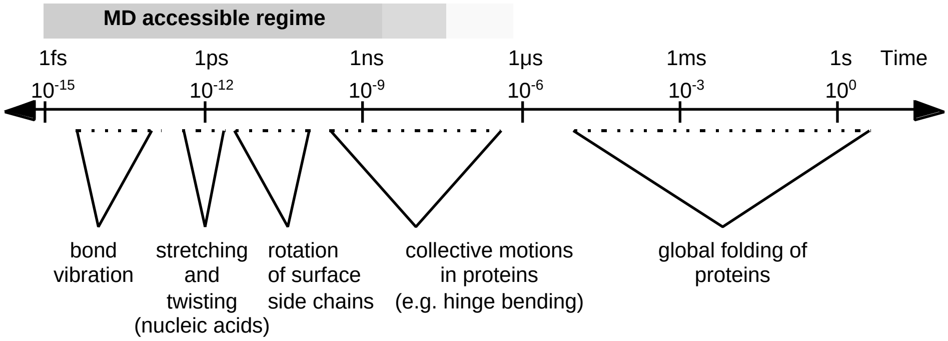

MD accessible regime
1fs
1ps
1ns
1μs
1ms
1s
Time
10^{-15}
10^{-12}
10^{-9}
10^{-6}
10^{-3}
10^{0}
bond vibration
stretching and twisting (nucleic acids)
rotation of surface side chains
collective motions in proteins (e.g. hinge bending)
global folding of proteins

Fig. 2: Timescales of internal motions in biomolecules and the timescales that can be reached using current molecular dynamics simulations.

Trajectory exploration. In this approach relatively large forces are applied over a short time to effect a rapid change in the system (in simulated time) which may be unrepresentative of the typical behavior of the system. What is of interest in such a steered simulation is the trajectory of key components which can be retained as a steering model for a much slower off-line simulation. This approach can also be used to eliminate trajectories that require excessively large steering forces, since this may be an indicator of their physical implausibility.

Incremental Steering. In this approach, a long-running batch simulation is checked from time to time (e.g. once every few days) to view the progress of the simulation and to make incremental adjustments to the restraints. In this approach the restraints are very small forces. As an analogy, this is the process an orthodontist uses to “steer” teeth to new positions over time.

Our work to date has concentrated on the trajectory exploration approach, an example of which will be described in section 2.

One of the early observations in the use of the SMD system was that the display of the simulation on a 2D monitor required considerable skill on the part of the operator to visualize the 3D motions in the system under study, and that placement of a restraint, a force with a 3D orientation, was particularly difficult. Thus we were interested in developing a 3D *virtual environment* for the interaction with SMD. Section 3 describes the adaptation of SMD to operate in the Protein Interactive Theater (PIT), an immersive virtual environment for two collaborating users. Using two head-tracked stereo displays, the PIT enables each user to observe and steer a running MD simulation in a single shared 3D environment.

Section 4 addresses some performance issues for the simulation, and in section 5 we conclude with a discussion of potential applications and future work for the SMD system.

2. Applications

The earliest applications of our interactive steered dynamics system were to steer small proteins through conformation changes and to examine the stability of the resultant system. Subsequently the system was used to extend the work on protein “sculpting” [22] in which protein structure is manipulated and energy minimization is applied to the resultant system. In our case the energy minimization was a consequence of the dynamics simulation. Both of these applications are described further in [16].

In this paper we describe a new class of applications of SMD in which we study the extraction of small ligands from proteins. These studies are interesting because they are related to experimental methods that have now become available such as atomic force microscopy (AFM) and optical tweezers. These micro-manipulation techniques allow experimentalists to manipulate individual molecules in solution. With AFM, experimental biophysicists can measure molecular stress-strain curves or the forces required to extract ligands from proteins, in terms of the deflection of a cantilever that is steadily moved during the experiment, one end of the molecular system being tied to the tip of the cantilever, the other end being tied to a fixed support. Simulated extraction of ligands from proteins using steered molecular dynamics is a means of modeling the extraction process in atomic detail and seeking to determine the critical molecular interactions responsible for producing the observed forces.

The number of extractions performed by experiment and simulation is steadily growing: biotin from avidin by experiment [18] and simulation [8,14], retinal from bacteriorhodopsin, simulation only [13], stress-strain measurements of titin, experiment [15] and simulation [17], and our own simulations of extraction (insertion) of xenon from (into) an artificial cavity in a mutant T4-lysozyme [11].

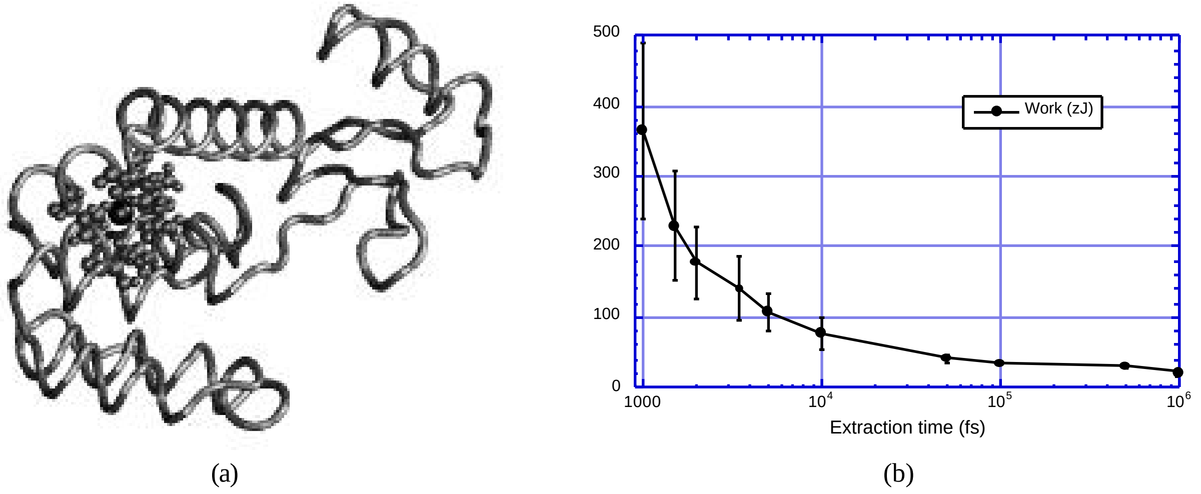
In one of these simulations the force is derived from a harmonic potential the zero of which is moved linearly along a spatial direction, which is a good approximation of the force applied in the AFM experiment. In some simulations the point of application of the force is steadily moved and the force required to achieve the steady displacement is measured (which represents the AFM experiment in the limit of a very stiff cantilever). In simulations of some systems, such as the biotin-avidin complex [8] and the complex of a T4-lysozyme mutant with xenon [11], the displacement is along a straight line. In other simulations, such as the retinal-bacteriorhodopsin system (where no AFM experiment has yet been done), successive restraints were applied along different directions, to avoid destroying the protein's conformation; in fact, this approach could take account of possible rotation of the macromolecule as the ligand is pulled out in an AFM experiment.

What qualitatively does the unbinding experiment report? At the most fundamental level, the forces will reflect the work required to undo the binding, when moving the ligand from the binding site to the less hospitable solution. According to thermodynamics, the net work done in the limit of an infinitely slow extraction process will reflect this binding free energy. Along the path connecting the bound and unbound states, forces nearly always will be needed to elastically deform the protein from its equilibrium state, and the work done in the extraction will then reflect one or more barriers, even in a very slow extraction. At finite extraction speeds, collisions with solvent molecules and atoms of the protein will generate an expenditure of work by the applied force, the accelerations imparted to these being "lost" as thermal motion (friction). In very unfavorable cases, the protein structure may "lock up" as a result of the application of an extraction force, much as an ordinary door will open easily when pushed one way, but fail to budge when pushed in the other direction. A corollary of a large friction component is a large variation of the forces observed in independent experiments [4,9].

Our T4-lysozyme mutant-xenon system is a unique system to test this theory because there are no electrostatic forces involved in the binding of xenon to the cavity of the mutant T4-lysozyme, and when xenon is absent the cavity is empty; unlike in other cases, no water molecules need fill the gap left by the departing ligand. With use of SMD we were able to quickly and efficiently identify pathways for xenon to exit from the hydrophobic cavity of mutant T4-lysozyme, shown in Fig. 3 (a).

A potential exit path is tested by applying a harmonic restraint towards a position far off in the solvent, and the extraction process is followed visually. The whole system is in motion and one exit pathway may look appealing at the beginning of the simulation and then disappear before the xenon reaches the exit. The ability to label amino acids potentially obstructing the xenon's path and to eliminate from the display very packed portions of the hydrophobic cavity where the xenon could not possibly pass, proved to be essential features of the SMD system, helpful in identifying essential detail. After several tries, each of several different informed operators favored the same, unique exit path. Without the SMD system, the search for the exit path would have to be carried out in the "darkness" of a series of more or less random batch simulations.

The interactive experiments with SMD were followed by batch calculations with Sigma, in which the xenon was moved at ever decreasing rate over the same linear path. The work done by the extraction force in extractions over times varying from 1 ps to 1 ns is shown graphically in Fig. 3(b), together with bars representative of the statistical error. (Each time point represents the mean and standard deviation of 35 independent simulations.) One sees that even for the slowest extractions, the net work is still a decreasing function of the extraction time, which indicates that the thermodynamic limit of very slow extraction has not yet been reached. This result is borne out by the persistence of statistical noise even in the slowest extractions. Since the time scale of typical extraction experiments so far has exceeded the time scale of the simulations by three orders of magnitude, direct comparisons are problematic.



(a) (b)

Fig. 3: (a) mutant T4-Lysozyme with xenon atom (center, dark) in non-native binding cavity. To minimize occlusion of the area of interest during extraction of the xenon atom, only the protein backbone and residues near the putative exit pathway are shown. (b) Total work performed during the extraction as steering forces are decreased and extraction time increased. Variability over 35 trials is shown at each point. As the process is slowed, the frictional component should decrease and the work required should converge to the potential of mean force for the extraction.

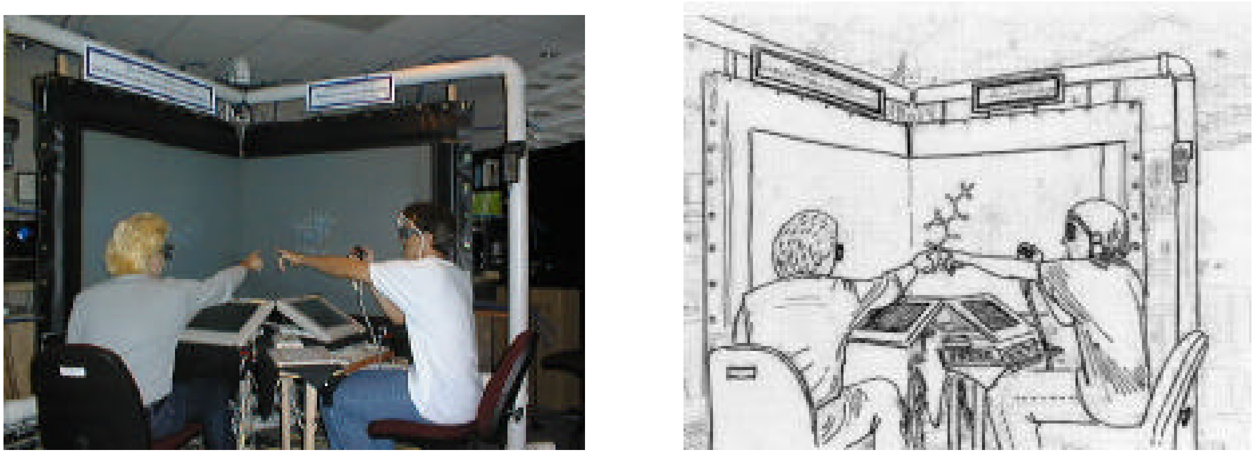
Also, in several insertion experiments performed at the slowest speed (insertion time of 1 ns), the energy change was not the reverse of that observed in the extraction; furthermore, passage of the xenon between the occluding side chains occasioned a much higher energy barrier during insertion than during extraction, and visual analysis indicated that side chains were pushed together by the approaching xenon, and thereby became less yielding to its passage.

In the set of 100 ps extractions, a significant disturbance in the force profile was noted in about one third of the cases, which led us to seek an explanation in mechanical terms. Upon inspection with SMD it was clear that a single solvent-exposed side chain, that of lysine 83, which is relatively large and flexible, was moving at a high frequency at the door of the exit pathway. Approximately one third of the time, this side chain obstructed the xenon's path and the other two thirds of the time it was out of xenon's way in another conformation. We then modeled a modified protein in which this unimportant residue was replaced by a sterically smaller residue, glycine. As a result, the erratic variation of the force profile disappeared. This effect would have been inexplicable with conventional "blind" molecular dynamics simulations.

3. The SMD Virtual Environment

The steered dynamics experiments described in the previous section were carried out using a 2D display with a 2D pointer device (a mouse). In this setting, considerable skill is required to visualize the 3D motions and to set restraints. To set a (simple) restraint, a single atom is "picked". The restraint strength and direction are defined to be the vector between the selected atom and the pointer device. However, as the 2D pointer device is defined to move in a plane perpendicular to the viewing axis, the restraints can only reasonably be set and manipulated by alternating between orthogonal views of the system. The view along the pathway prescribed by the restraint allows the path to be checked for obstacles, while the orthogonal view is used to modify the direction or magnitude of the restraint. While the (highly motivated) SMD operators can master this task, it is tedious at best, and, more significantly, the views and manipulations are quite difficult to follow by observers. This limits the pedagogical utility of the SMD system and makes collaborative work using the system difficult.

It was our feeling that a 3D environment for SMD might simplify the observation of the dynamics and placement of the restraints, and that a 3D environment visible to multiple users would enable a trained and untrained users to work together and permit us to improve our demonstrations. In fact, there exists a capability, provided by the underlying



(a) (b)

Fig. 4: (a) PIT configuration, (b) depiction of virtual environment as seen by PIT operators.

VMD molecular graphics display software, for a single user to view the SMD display in stereo using a Stereographics CrystalEyes™ system [21]. However, steering of the Sigma dynamics simulation is not integrated into this capability.

3.1 Protein Interactive Theater

Fortunately we had an opportunity to create an integrated virtual environment for SMD using the Protein Interactive Theater (PIT) constructed at UNC Chapel Hill [3]. The PIT is a dual-screen, stereo display system for two operators seated at a table as shown in Fig. 4. Each operator wears a pair of CrystalEyes™ liquid crystal shutter glasses with an attached tracking sensor, and views a stereo image projected on the screen across the table. The continuously-updated head-tracked image provides stereo and motion parallax cues to give an illusion of a stable 3D scene located in a shared workspace above the table in front of them. The two operator’s views are in registration, so that the operators agree on the apparent position of objects in the physical workspace, and can augment their discussions using hand gestures as shown in Fig. 4(b). Each operator also has a tracked, 6 degree-of-freedom, handheld controller that provides pointing, picking, and other scene manipulations. Dials and buttons at the table corner shared by the two operators provide top level control over the display parameters and the dynamics simulation. A separate flat-panel LCD monitor in front of each user, visible through their stereo glasses, provides access to a conventional windows, keyboard and mouse interface for detailed control of the visual representation of the molecule and the MD simulation.

High quality stereo is achieved using individual high-resolution fast-phosphor video projectors for each screen. The screens are constructed of vinyl for a bright image with little view-position dependent variation and minimal reflected light between screens. The liquid crystal shutter eye glasses are synchronized with the projectors to ensure that each eye is presented with its own correctly projected view. The projected resolution is 1280 × 492 for each eye, and 1280 × 992 for each user. Subjectively, operators of the PIT report an extremely high-fidelity stereo experience. Additional observers can also view the system in 3D using shutter eye glasses, however they will be seeing a stereo view generated for one of the two operators, and hence their best viewing position is on axis with one of the operators. Demonstration of the system to groups of 6-8 additional observers is entirely reasonable.

3.2 SMD in the PIT

A typical SMD session is started from one of the operator LCD monitors. Once started, the system under study appears motionless in the workspace in front of the operators, typically rendered as a smooth-shaded CPK solid model. The scene may be rotated, translated and scaled using the handheld pointer device to gain the best view.

Using the operator consoles, the parts of the system displayed and their graphical representation can be customized, as shown in Fig. 3(a), for example.

The displayed scene also includes a white and a yellow cone, which are representations of the two pointer devices in virtual space. The pointer's virtual location may be decoupled from the pointer's physical location in the shared workspace to provide a more comfortable laptop operating position. Either operator may translate and rotate the scene as a whole at any time by moving their pointer with the appropriate button depressed. The shared workspace concept enables the operators to easily and intuitively create a view that is suitable for both users. Control over the selection and manipulation of individual components in the scene rests with the operator whose pointer representation displays in white. Control can be passed between the operators using a toggle on the button box. Another toggle on the button box starts and stops the dynamics simulation. When the simulation is running the displayed position of the atoms is continuously updated from the simulation.

To steer the dynamics simulation, restraints are introduced into the computation. A *simple restraint* is defined by selecting an atom with the pointer device. A red line anchored to the selected atom follows the pointer. The direction of the line corresponds to the direction of the restraint, while the magnitude is given by the length of the line. The placement of a restraint in the virtual environment is greatly simplified and considerably faster than the corresponding 2D procedure. However, the selection subtask using the pointer device in 3D is initially somewhat slower than the equivalent procedure in 2D using a mouse, because a depth judgment is required. As an aid, we often suggest that operators "bury" the tip of their cone (the representation of their pointer) inside the sphere of the atom they wish to select, and then press the selection trigger on their pointer device. Restraints may be modified by picking up the end of the line at the restraint point and moving it. A "live" restraint is one whose end is following the pointer device. If the dynamics simulation is running, a live restraint corresponds directly to an interactive steering force controlled by the pointer device.

A *rigid body* restraint is defined by selecting a group of atoms and dragging a line from one of the members of the group to the desired restraint point. In the placement of the rigid body restraint, a frame representation of the rigid body, corresponding to the bonds of the selected group, is shown in green and follows the pointer. In the case of a rigid body restraint it is necessary to show the position and orientation of the frame at the endpoint of the restraint, and here the six degrees of freedom provided by the pointer device are extremely useful. A rigid body restraint behaves as simultaneous, but separate, restraints on all atoms in the group in order to achieve a rigid body transformation of the whole group. This permits, for example, a sidechain rotation to be performed using a single rigid body restraint.

3.2 Software structure of the SMD system

The software structure of the SMD system is shown in Fig. 5 and consists of the following three components.

Dynamics Simulation is provided by the Sigma system developed at the University of North Carolina [10]. This program supports several MD force fields, including Cedar, Charmm, and Amber, and input file formats, including charmm-X-plor and amber formats. The system contains selections for free energy perturbation techniques, utilizes the Ewald method for electrostatics and supports the direct interface with the VMD program when used in SMD mode. In particular, Sigma maintains the current set of restraints, continuously reinterprets rigid body restraints, and incorporates all restraints into the simulation. At present the publicly released code supports single-processor calculations on a variety of platforms, and a shared-memory multiprocessor version is being updated.

Molecular Graphics and User Interface is provided by the VMD system [12]. This system enables molecular models to be displayed in a number of different graphical representations and output formats. Using VMD we may specify a partitioning of the molecular model, with each of the components rendered in its own way. This enables us, for example in the T4-lysozyme extraction, to specify that the solvent should not be shown, to view a backbone representation of the protein, and to show explicitly all residues within a certain radius of the moving xenon atom. The VMD system also controls the pointing and selection user interface for individual components of a scene. With this capability a restraint can be defined in the coordinate system of the displayed system, which is converted to a restraint in the coordinate system of the simulation and passed to Sigma.

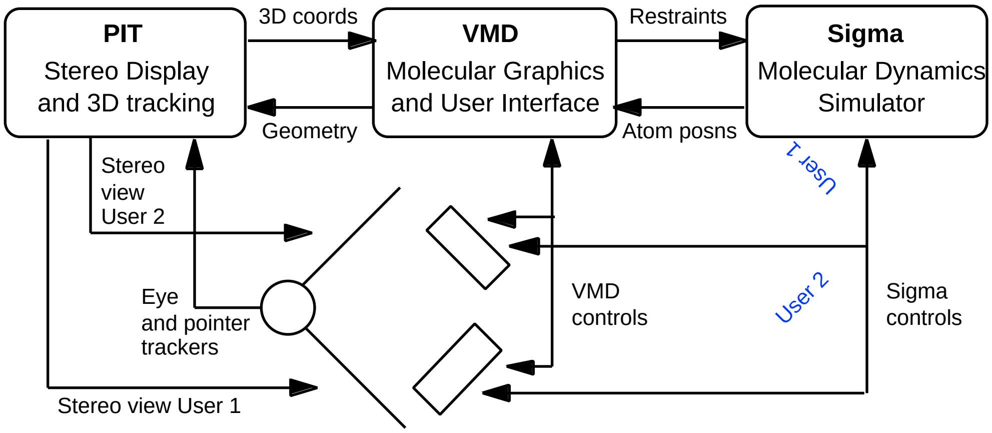


Fig. 5: Software components of the SMD virtual environment.

The coupling between VMD and Sigma is through a lightweight protocol built on TCP, hence the simulation may run asynchronously on a different machine. While Sigma generally simulates the full system, the protocol only updates the positions of atoms selected for display with VMD in order to decrease communication costs. With this protocol the bandwidth requirements for communicating with a running simulation (about 250KB/sec for the T4 system without solvent display), are typically within the capabilities of a modern local area network (LAN). However, variation in LAN latency can make the interaction with the simulation jumpy, hence we prefer to run Sigma and VMD on a multiprocessor with a predictable and high-performance interconnect.

Our system currently provides three types of synchronization in the coupling between Sigma and VMD. In *asynchronous* mode, the simulation runs independent of the display. New atom coordinates are provided when requested and restraint changes are incorporated when provided. In *synchronous* mode, Sigma accepts restraint changes (if any), simulates a fixed time interval, and then communicates the new atom coordinates. Typically the simulated time per update is a single integration time step to maximize the smoothness of motion and interactiveness of the system. Finally, the *restraint-driven* mode is a synchronous mode in which the simulation is advanced only when restraints change. This enables the steering to be controlled very precisely, advancing just a single step whenever the pointer is moved. In this mode the operator sets a collection of restraints and "wiggles" one of the restraints to advance the simulation, frequently stopping to examine the system and perhaps checking the energetics in the Sigma window. Clearly this mode is the slowest mode in terms of simulated time. Asynchronous mode provides the highest simulation rate, but the steering control is less accurate. For reasonably-sized systems, synchronous mode provides close to the maximum simulation rate, a smooth display update, and good steering control.

Stereo Rendering and 3D tracking is provided by the PIT system [3]. In our current configuration VMD and the PIT operate as co-routines. VMD handles the user interface and calls the PIT to create the display. The PIT sets up viewpoints and lighting and then calls VMD four times in succession (two eyes for each of the two users) to render the geometry. VMD renders the geometry through a series of calls on OpenGL primitives. The PIT system tracks the location of both operators and both pointers and provides the 3-D coordinates of the pointer device belonging to the operator currently in charge to VMD. In fact VMD is unaware that there are two users. The PIT software also handles the view transformations that result from whole-scene manipulation by either operator. In a future configuration we plan to decouple VMD and the PIT so that the response to operator head position changes and scene manipulation are completely independent of the VMD rendering rate and the Sigma coupling.

4. Performance Considerations

To provide an adequate *interactive* experience , it is necessary to provide visual updates with a frequency of at least 20Hz. We distinguish between the *scene update rate* (in frames per second) and the *simulation update rate* (in integration timesteps per second). Currently we host all three components on a mutiprocessor SGI Onyx, using a single Infinite Reality™ pipe for all graphics. This gives us display update rates of 20 frames per second for small systems. We plan to move the system to an Onyx 2 with multiple graphics pipes and to decouple VMD and the PIT. In this configuration, we expect the scene update rate to be comfortably beyond 20Hz for all systems we might try to visualize. Even with very large systems we expect scene complexity to be limited because the operators will naturally tend to reduce detail in the representation for parts that are far removed from the steering site, in order to improve their understanding of the simulation.

Achieving a high simulation update rate is a more challenging problem because of the size of the computation involved. Most components of the dynamics simulation have computational complexity that scales directly with the number of atoms n in the system. Bond length and angle constraints require a small number of evaluations per atom. Short-range non-bonded forces decay rapidly with increasing distance, hence are computed using a fixed interaction radius, which leads to a constant upper bound on the number of interactions that need to be evaluated per atom. Only the long-range electrostatic force evaluation can scale more rapidly than $O(n)$, although the particle-mesh Ewald formulation used in Sigma has $O(n \log n)$ growth with small constants for the uniform distributions of atoms typically found in simulations of biomolecules. Moreover, a multiple time stepping scheme results in long-range electrostatics being recalculated at a lower frequency than the basic integration timestep. Hence for most purposes we can view the simulation complexity to be linear in n, with a multiplicative factor of about 5,000 flops per atom. To update a 20,000 atom system at 20 integration timesteps per second requires roughly a 2 GFLOP computing rate.

Certain optimizations have been made in Sigma to maximize its performance. To address memory hierarchy optimization the non-bonded force calculations were reordered. Typical force fields require roughly two hundred non-bonded interactions per atom be performed, half of which can be eliminated using $F_{ij} = -F_{ji}$. The remaining interactions may be computed in any order. By computing the forces on atoms in an order that takes into account physical proximity, many of the interacting atoms remain in cache memory for several force calculations. In our case, this led to a factor of 2.5 improvement in performance.

Additional performance gains have also been achieved through parallelization. Our current parallelization scheme relies on data structures maintained in a shared memory. We use an irregular spatial decomposition to assign atoms to processors, again to make use of the fact that nearby atoms interact with similar sets of atoms and to accurately balance the work over the processors. This helps in 2 ways: first, nearby atoms interact with one another, thus using only local data and second, when remote data must be retrieved, it is very likely that it will be used many times, amortizing the cost of remote access [19].

Figure 6 shows scaling characteristics of a parallel version of Sigma on the SGI O2000. The graph shows performance on two different simulations. For the larger T4-lysozyme system, a simulation rate of better than 10 integration steps per second is achieved using 8 processors. As we scale to larger number of processors, there must necessarily be a limit in performance scalability for fixed size systems. However, aggressive implementation efforts can still improve considerably on current practice. Analytically, the principal impediment to scaling a fixed size system to a large number of processors is load imbalance, which can be improved using modifications of the decomposition strategy for non-bonded interactions [19]. In the recent 1μs simulation of a villin subdomain with 12,000 atoms using a 256 processor Cray T3E, detailed load-balance techniques of this sort were applied to reach a simulation update rate of 115 integration time steps per second or 230 fs simulated time per second [7].

Another approach to improving simulation rates is via a *molten zone*. Such a simulation only updates the positions of atoms within some region, typically a sphere centered on the steering locus. The atoms in the molten zone benefit from a full calculation of the forces, including force contributions from atoms outside the molten zone. The work saved lies in the force computations for atoms outside of the molten zone. In the T4-lysozyme studies, the molten zone technique provides a factor of 4-8 improvement in the simulation rate.

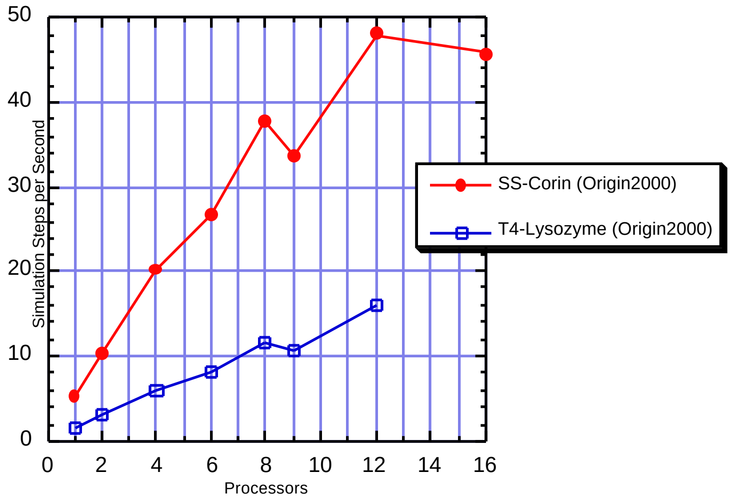


Fig. 6: Parallel performance of Sigma in integration timesteps per second on two simulations. The SS-Corin system has a total of 3,913 atoms; the T4-lysozyme system in this experiment has 13,642 atoms. The simulations were performed using periodic boundary conditions, a 10Å cutoff radius for long-range interactions and 2fs integration timesteps using SHAKE bond length constraints. The machine used was an SGI Origin 2000, with 195Mhz R100000 processors and 1MB secondary cache per processor.

In summary , there are a number of available techniques that lead us to expect that adequate interactive simulation update rates can be achieved even on relatively large systems using a modern mid-sized multiprocessor. In an absolute sense, however, the simulated time scales will remain very small compared with the steering forces that can be introduced, so that great care must be taken in the interpretation of the results.

5. Conclusions

We are approaching a time when experimental measurements of extraction forces for ligands from proteins using atomic force microscopy, optical tweezers and other techniques may be made at timescales that are also accessible to computational methods. The SMD system can provide a valuable tool for biophysicists to perform the computational analog of such experiments to refine the accuracy of both experimental and computational techniques and to improve the understanding of forces and dynamics of the binding and dissociation processes.

We are considering the use of a force-feedback input modality in the PIT [5] to offer resistance proportional to the local temperature in the system to reflect the dissipation of work through friction. We are also examining the possibility for simultaneous steering of a simulation by both operators so that complex steering motions can be orchestrated.

6. Acknowledgments

Jon Leech originally developed the SMD system [16] interfacing Sigma and VMD. Michael Lappe assisted with the adaptation of SMD to operate in the PIT. Jane Marshall drew Fig. 4(b). Access to the PIT system was provided by the GRIP project led by Frederick P. Brooks at the Dept. of Computer Science at UNC Chapel Hill; Kevin Arthur, Zac Kohn, Tim Preston, Russ Taylor and Mary Whitton provided information on its use. Thanks also to Robert Belleman and Jaap Kaandorp at the University of Amsterdam for their help in arranging a demonstration of SMD in the SARA Cave at HPCN 98.

7. References

[1] P. M. Allen, D. J. Tildesley, *Computer Simulation of Liquids.*, Clarendon Press, Oxford, 1987.

[2] A. G. Anderson, J. Hermans, “Microfolding: Conformational Probability Map for the Alanine Dipeptide in Water from Molecular Dynamics Simulations”, *Proteins* **3**:262--265, 1988.

[3] K. Arthur, T. Preston, R. M. Taylor II, F. P. Brooks, Jr., M. C. Whitton, and W. V. Wright, “Designing and Building the PIT: a Head-Tracked Stereo Workspace for Two Users”, in Proc. of 2nd International Immersive Projection Technology Workshop (Ames, Iowa, May 11-12, 1998).

[4] M. Balsera, S. Stepaniants, S. Izrailev, Y. Oono, K. Schulten, “Reconstructing potential energy functions from simulated force-induced unbinding processes”, *Biophys. J.* **73**:1281--1287, 1997.

[5] F.P. Brooks et al., “Project GROPE – Haptic Displays for Scientific Visualization”, *Computer Graphics*, **24**:177--185, 1990.

[6] M. Nelson et al., “MDScope: A visual computing environment for structural biology”, *Comput. Phys. Comm.* 91:111-134, 1995.

[7] Y. Duan, P.A. Kollman, “Pathways to a Protein Folding Intermediate Observed in a 1-Microsecond Simulation in Aqueous Solution”, *Science* **282**:740-744, 1998.

[8] H. Grubmüller, B. Heymann, P. Tavan, “Ligand Binding: Molecular Mechanics Calculation of the Streptavidin-Biotin Rupture Force”, *Science* **271**:997--999, 1996.

[9] J. Hermans, “A simple analysis of noise and hysteresis in free energy simulations”, *J. Phys. Chem.*, 9029-9032, 1991.

[10] J. Hermans, R. H. Yun, J. Leech, D. Cavanaugh, Sigma documentation. University of North Carolina, `http://femto.med.unc.edu/SIGMA/`

[11] J. Hermans, G. Mann, L. Wang, L. Zhang, “Modeling macromolecule-small molecule interactions with dynamics simulations”, in *Algorithms for Molecular Modelling* (Deuflhard, P., Hermans, J., Leimkuhler, B., Mark, A., Skeel, R. D. & Reich, S., eds.). Springer, Berlin, 1998.

[12] W. Humphrey, A. Dalke, K. Schulten, “VMD - Visual Molecular Dynamics” *J. Molec. Graphics* 14(1):33--38, 1996.

[13] B. Isralewitz, S. Izrailev, K. Schulten, “Binding pathway of retinal to bacterio-opsin: A prediction by molecular dynamics simulations”, *Biophys. J.* **73**:2972--2979, 1997.

[14] S. Izrailev, S. Stepaniants, M. Balsera, Y. Oono, K. Schulten, “Molecular dynamics study of unbinding of the avidin-biotin complex” *Biophys. J.* **72**:1568--1581, 1997.

[15] M. Kellermeyer, S. Smith, H. Granzier, C. Bustamante, “Folding-unfolding transition in single titin modules characterized with laser tweezers”, *Science* **276**:1112-1116, 1997.

[16] J. Leech, J. F. Prins and J. Hermans, “SMD: Visual Steering of Molecular Dyanmics for Protein Design”, *IEEE Computational Science and Engineering* **3**(4):38--45, 1996.

[17] H. Lu, B. Isralewitz, A. Krammer, V. Vogel, K. Schulten, “Unfolding of titin immunoglobulin domains by steered molecular dynamics”, *Biophys. J.* **75**, 662-671.

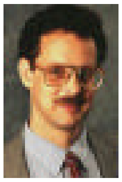
[18] V.T. Moy, E. Florin and H. Gaub, “Intermolecular Forces and Energies Between Ligands and Receptors”, *Science* **266**:257--259, 1994.

[19] L. Nyland, J. Prins, R.H. Yun, J. Hermans, H.-C. Kum, L. Wang, “Achieving Scalable Parallel Molecular Dynamics Using Domain Decomposition Techniques”, *Journal of Parallel and Distributed Computing* **47**(2):125--138**,** 1997.

[20] S. Stepaniants, S. Izrailev, W. Wriggers, K. Schulten, “Steered molecular dynamics”, in *Algorithms for Molecular Modelling* (Deuflhard, P., Hermans, J., Leimkuhler, B., Mark, A., Skeel, R. D. & Reich, S., eds.). Springer, Berlin, 1998.

[21] StereoGraphics Corporation. `http://www.stereographics.com/`

[22] M. Surles et al., “Sculpting Proteins Interactively: Continual Energy Minimization Embedded in a Graphical Modelling System”, *Protein Science* **3:**198-210, 1994.



Jan F. Prins is associate professor of computer science at the University of North Carolina at Chapel Hill. He obtained a BS in Mathematics from Syracuse University and an MS and PhD (1987) in computer science from Cornell. His research interests center on high-performance computing, including algorithm design, computer architecture, and programming languages. He is a participant in the UNC/Duke/NYU Parallel Computing Resource for Structural Biology sponsored by NIH. Prins is a member of the IEEE and ACM. E-mail: prins@cs.unc.edu

Jan Hermans is professor of biochemistry and biophysics in the School of Medicine at the University of North Carolina at Chapel Hill. He obtained a BS in chemistry from the University of Groningen and a PhD (1958) in physical chemistry from the University of Leiden. His research interests center on the application of computer modeling to structural biology. He directs the UNC/Duke/NYU Parallel Computing Resource for Structural Biology. Hermans is a member of the American Chemical Society, the Biophysical Society, the American Crystallographic Association and AAAS. E-mail: hermans@med.unc.edu

Geoffrey Mann is a doctoral candidate in the department of Biochemistry and Biophysics at the University of North Carolina at Chapel Hill. His research interests lie in the area of protein dynamics in the presence of ligands. E-mail: gmann@femto.med.unc.edu

Lars S. Nyland is research associate professor of computer science at the University of North Carolina at Chapel Hill. He obtained his BS in Computer Science from the Pratt Institute in Brooklyn, NY (1981), and his AM (1983) and PhD (1991) from Duke University in Durham, NC. His research interests focus on high-performance computing, with specific interests in programming languages, optimizations, algorithms and their implementation and high-performance hardware. He is a participant in the UNC/Duke/NYU parallel computing resource for Structural Biology sponsored by NIH. E-mail: nyland@cs.unc.edu

Martin Simons is a postdoctoral researcher in computer science at the University of North Carolina. He is on leave from TU Berlin and is funded by a scholarship from the German research foundation (DFG). He obtained a Diplom in Computer Science from Universität Karlsruhe and a Dr.rer.nat. (1996) in computer science from TU Berlin. His research interests focus on high-performance computing, including formal semantics, programming languages, and algorithm design. Simons is a member of IEEE Computer Society and ACM. E-mail: simons@cs.unc.edu.