Journal of Physics: Conference Series **125** (2008) 012049 doi:10.1088/1742-6596/125/1/012049

Developing improved MD codes for understanding processive cellulases

M F Crowley[1], E C Uberbacher[2], C L Brooks III[3], R C Walker[4], M R Nimlos[1] and M E Himmel[1]

[1]National Renewable Energy Laboratory, Golden, CO 80401,USA
[2]Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA
[3]University of Michigan, Ann Arbor, MI 48109, USA
[4]San Diego Supercomputer Center, La Jolla, CA 92093, USA

Abstract. The mechanism of action of cellulose-degrading enzymes is illuminated through a multidisciplinary collaboration that uses molecular dynamics (MD) simulations and expands the capabilities of MD codes to allow simulations of enzymes and substrates on petascale computational facilities. There is a class of glycoside hydrolase enzymes called cellulases that are thought to decrystallize and processively depolymerize cellulose using biochemical processes that are largely not understood. Understanding the mechanisms involved and improving the efficiency of this hydrolysis process through computational models and protein engineering presents a compelling grand challenge. A detailed understanding of cellulose structure, dynamics and enzyme function at the molecular level is required to direct protein engineers to the right modifications or to understand if natural thermodynamic or kinetic limits are in play. Much can be learned about processivity by conducting carefully designed molecular dynamics (MD) simulations of the binding and catalytic domains of cellulases with various substrate configurations, solvation models and thermodynamic protocols. Most of these numerical experiments, however, will require significant modification of existing code and algorithms in order to efficiently use current (terascale) and future (petascale) hardware to the degree of parallelism necessary to simulate a system of the size proposed here. This work will develop MD codes that can efficiently use terascale and petascale systems, not just for simple classical MD simulations, but also for more advanced methods, including umbrella sampling with complex restraints and reaction coordinates, transition path sampling, steered molecular dynamics, and quantum mechanical/molecular mechanical simulations of systems the size of cellulose degrading enzymes acting on cellulose.

1. Introduction

Lignocellulose could potentially serve today as the dominant renewal biofuel source, capable of meeting our needs for a liquid transportation fuel, if it could be efficiently and economically depolymerized into its component glucose for microbial fermentations [1, 2]. One major obstacle to the efficient and effective use of cellulose is the high resistance of crystalline cellulose to chemical and biological hydrolysis [3]. Even the most efficient cellulase enzymes available are not effective enough to depolymerize cellulose at a rate or cost that can meet the needs of commercial scale biofuel production. Cellobiohydrolase I (Cel7A) from *Trichoderma reesei* is one of the most effective exoglucanases known and in many ways the dominant cellulase in the biosphere. Cel7A is a multidomain enzyme, consisting of a large catalytic domain containing an extensive active site tunnel and a small binding module joined to one another by a linker peptide. Through a series of mechanisms yet unknown, a single cellodextrin chain is removed from the crystal and fed into the active site tunnel of the catalytic domain for hydrolysis to cellobiose. Cel7A is hypothesized to proceed along the target cellodextrin in a processive manner, cleaving one cellobiose unit per catalytic event, see figure 1 below. However, experimental evidence for the processive motion hypothesis is limited. Although much is known about the structure of Cel7A, many questions remain about the functioning of this enzyme, which can be considered a

© 2008 IOP Publishing Ltd

"protein machine." Among them are the roles of the carbohydrate binding module (CBM) and the linker peptide connecting the CBM to the catalytic domain. We have shown in previous work that the CBM recognizes and binds to the cellulose surface [4]. Furthermore, we have postulated that the linker works in a "springlike" motion to enable the enzyme to move on the cellulose surface during catalysis of a cellulose chain; where the catalytic hydrolysis of a cellulose strand by the moving catalytic domain compresses the length of the linker, and once a short enough length is reached, the stored potential energy allows the linker to extend, freeing the binding module from its current position on the cellulose surface [5]. The mechanism by which the catalytic domain translates cellodextrin substrates toward the active site is also not understood. If improved Cel7A enzymes are to be developed using protein engineering, rational design methods require some baseline understanding for structure/function relationships.

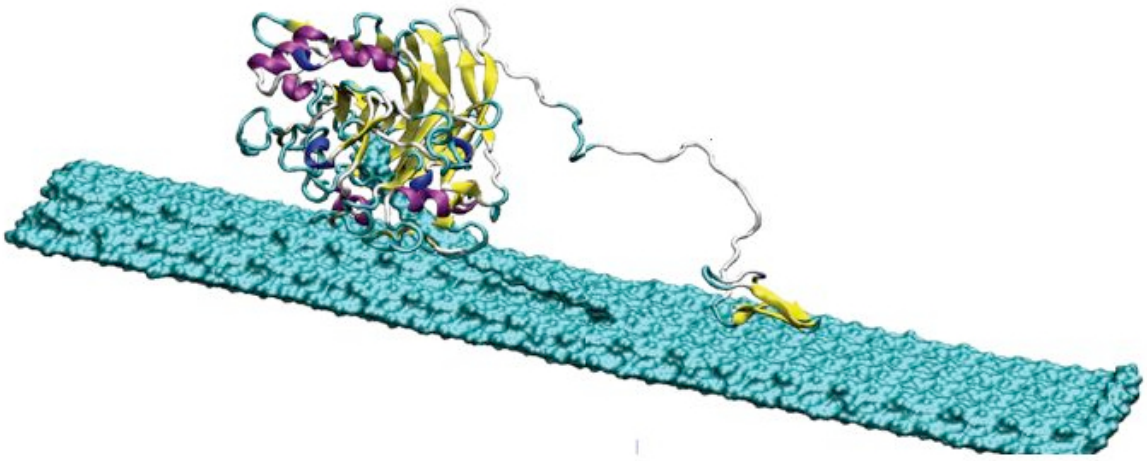

Figure 1. Computational model of Cellobiohydrolase I (Cel7A) from *Trichoderma reesei* on a cellulose 100 surface displaying the large catalytic domain (left), linker (middle single strand), and cellulose binding module (right small domain). A cellodextrin strand is shown peeled out of the surface of the cellulose and threaded into the catalytic tunnel of Cel7A. The solvating water and the lower portion of the cellulose fiber are not shown in this visualization.

Given that there are no experimental methods available to probe this complex behavior at the molecular level, we have proposed using improved molecular dynamics simulations to study this problem from the perspective of the system ensemble, which includes cellulose, the enzyme, and water. To accomplish this work, which focuses on a cellulase-cellulose model of approximately 800,000 atoms, part of the model is shown in figure 1, one must develop improved MD codes capable of taking full advantage of new leadership class computers. Computational codes exist for performing simulations of up to millions of atoms on as many as thousands of processors with some degree of scaling efficiency, namely, Amber [6], LAMMPS [7], and NAMD [8], However, these codes have limited functionality beyond simple and steered MD and some simple sampling techniques. There is a great need for improving the scaling efficiency of MD programs and to migrate more complex simulation and sampling methods into highly scalable programs. Many of the necessary methods for studying the proposed mechanisms of cellulase enzymes and the cellulose substrate are available in CHARMM [9] but are not currently available to be used on systems the size of a cellulase/cellulose system for the length of simulation that would produce meaningful results within a reasonable timeframe. This SciDAC project will design highly parallel and highly efficient programs by both improving the parallel efficiency of existing programs and algorithms and creating new codes and new algorithms for the most important methods for use on current and future terascale and petascale computers.

2. Computational methods for studying cellulose and cellulase

In the most primitive approach to studying at the molecular level of detail the cellulase mechanism for degrading cellulose to cellobiose, simple molecular dynamics is used to produce many nanoseconds and hopefully microseconds of molecular motions. There is likely little to be gained from such simulations due to the low likelihood of important events occurring on that time scale since this enzyme acts on the seconds to minutes timescale. Some of the equilibrium structural and dynamical features of the system have been extracted from multi-nanosecond simulations of the enzyme and substrate including structuring of water around each of the three domains of the enzyme and how it restructures on the cellulose when the enzyme is present. For these simulations to continue to the microsecond time, the improvements in scaling and parallel efficiency are necessary.

The important parts of the mechanism are those responsible for the processive motion of the enzyme and the catalytic event. The processive motion is one or a series of rare events including decrystallization of the cellulose surface, movement of a free cellodextrin strand into the catalytic tunnel, movement of the cellodextrin through the tunnel, the catalytic event, leaving of the cellobiose product, and finally moving of the cellodextrin into reaction position again. Initial simulations using currently available MD programs show that these are both complex and seem to have several very high barriers and are thus rare events requiring much more sophisticated methods than simple MD. Our development wish list for highly parallel and scalable methods specifically targeted at solving the cellulase problem includes umbrella sampling using complex restraints, steered and targeted MD, QMMM, transition path searching and sampling methods, and thermodynamic integration and perturbation methods. Current and new scientific methods and models are also needed to improve the quality of the simulations and to enhance both the speed and sampling of the numerical experiments. These include using new solvent models such as the water models with off-center charges, implicit solvent models, and new polarizable force fields, none of which are available in highly parallel and scalable MD codes.

3. Program optimization - code development and enhancement

The primary focus of the code development work for this project is on computational methods that are the most useful to scientists studying large biological systems, and in particular on the methods that are initially needed for the biological application chosen, cellulase action. Our primary aim is to make these methods as efficient and parallel as possible without sacrificing versatility and functionality. Many of the most valuable molecular modeling methods such as thermodynamic integration [10], free energy perturbation, pulling or steered MD, umbrella sampling [11], and hybrid quantum mechanical/molecular mechanics (QMMM) [12, 13] have been restricted to smaller molecular systems, 30,000 atoms or less or are very limited in their implementations. These restrictions are a function of both their limited scaling and a lack of knowledge of how these algorithms behave when used for modeling larger systems. These approaches, which largely represent an extension of standard MD methods, are typically limited to around 64 CPUs or to very simple restraints. Not long ago, these CPU counts were considered reasonably large. The development of both the algorithms and the implementations of the computational methods have been largely guided by limitations in the resources available and the characteristics of the existing computers. Many of the existing methods have proven useful scientific tools for studying systems of moderate size. However, the current implementations are not designed for large systems or massively parallel computers by current or projected standards and are only marginally efficient for even the smaller subsystem simulations needed for the cellulase study. We will study the more important algorithms to uncover the limitations to extending to large systems and petascale machines with many thousands of processors in both the implementations and the methods themselves.

Our efforts to improve parallel and scalar efficiency will be combined with a rational approach, to where the effort will be most effective by studying the theory and algorithmic aspects of several computational methods that go beyond simple classical MD and are essential to most studies that model biomolecular systems using MD for more than single trajectory studies. As we extend capabilities of existing tools and methods to run on larger and faster computers and to handle larger molecular models, it is important that we consider which problems are appropriate for the available tools and whether or not the modeling really answers the questions of interest. Our primary focus will be on the methods necessary to answer the questions posed in the Cel7A processive enzyme study, and if, even with petascale computers, these questions can be answered with the approaches that have been traditionally used. Simply running longer trajectories of all-atom models and extending to larger and larger systems for similar time-domain simulations may not yield the increase in information expected from linear-scaling arguments. For instance, larger molecules, because of motions on much larger spatial scales and exponentially larger phase space degrees of freedom, require significantly more time to undergo structural changes and concerted motions than do smaller molecules. For these reasons, we will focus heavily on evaluating the questions that are appropriate for the molecular systems and computational hardware we are targeting. As an essential result of this effort we will provide a gauge of the confidence that can be placed on the results of simulations using the algorithms, methods, theory, and software produced.

The methods we are interested in implementing into an improved code are essential tools of molecular modeling in general and are critical to the cellulase problem. These methods rely on the underlying MD engine of the program in which they are embedded, such as AMBER and CHARMM. The state-of-the-art molecular-modeling technology is well developed in some programs, such as LAMMPS and NAMD, that scale very well to thousands of processors but are limited to simple classical MD and a few simple sampling enhancing methods with limited functionality. The other more traditional simulation packages, such as CHARMM and AMBER, have a wealth of methodologies for detailed studies of the physical properties and behaviors of solvated macromolecular systems. Our primary aim is to enhance the massively parallel computations with the most useful and most widely used methods for studying the physical chemistry of biomolecular systems by implementing the already proven efficient parallel spatial-decomposition methods in the framework of existing less-efficient but highly functional and validated programs. The methods we wish to make highly efficient have not been applied to systems of millions of atoms that are currently considered as appropriate for computers with many thousands of processors. Little is known about whether the methods will produce meaningful results on large systems, or whether there are characteristics of the theory that will prevent either massively-parallel implementations or application to very large systems from being useful. We are implementing algorithms for several methods on massively parallel architectures and evaluating the validity, efficiency, and reliability on the cellulase system as a test bed, but with general use software as the target product. Our studies use existing MD packages, CHARMM]9], AMBER [6], and LAMMPS [7] as starting points both for the studies and code development and for validation of the scientific results produced by the implementations. The specific use and function of each program and the approach to the problem are described below.

The algorithms for parallel MD can be grouped into two general classes: replicated data and spatial decomposition. The traditional MD programs use the replicated data model for several reasons, not least of which is historical since they were originally written before there were parallel computers. The advantage of the replicated data model is that nearly all the important modeling methods can be made parallel when every processor has all the current data. However, there is a low ceiling to the amount of scaling that can be achieved with such an approach. On the other hand, the spatial decomposition method is highly scalable and current MD programs using this method can scale to thousands of processors with tolerable efficiency. The problem with

spatial decomposition is that the more complex MD methods are not spatially localized in character. Many of the important methods break the scaling efficiency of the highly parallel codes and thus have not been implemented. Our aim is to study the scientific methods of standard MD studies and to develop new algorithms that will scale as efficiently as possible in the spatial decomposition framework. Although some of the methods will degrade parallel performance their usefulness outweighs the cost of using them. Some of the methods will not degrade parallel performance but will require the development of new approaches to their parallel implementation. In summary, the aim is to implement the scientific methods, study their scientific and performance behaviors, and develop the most effective and efficient implementations that preserve the most scientific benefit. The particular methods and features for our study are as follows:

- Spatial decomposition implementation in existing replicated-data program
- Spatially decomposed implicit solvent model algorithms
- New fast explicit solvent and non-bond list algorithms
- Polarizable Force Fields in spatial decomposition programs
- Lone pairs/Extra points (TIP4P and TIP5P water)
- General restraints and umbrella sampling
- Thermodynamic integration (TI)
- Nudged elastic band (NEB) and more refined path-finding bead methods
- Steered molecular dynamics (SMD)
- Semi-empirical QMMM methods with Ewald, PME, and implicit solvent models

The need for increased sampling is clear for larger molecular systems. As yet, there are only estimates for what the sampling needs are, and some guesses about what the issues are for both highly parallel approaches and larger systems. The methods of thermodynamic integration, umbrella sampling, and nudged elastic band are critically dependent on converged sampling. We will explore the issues for large systems and generate both analysis tools for the large-scale modeling and highly parallel codes. Many of the methods described above require numerous runs with different starting conditions or different parameters such as umbrella restraints, temperatures, or pulling forces and can be run together to use a large scale machine such as a 100 teraflop to petaflop computer even for the simulations of the subsystems.

4. Program validation and benchmarking

The National Leadership Computing Facility (NLCF) machines at ORNL are soon going to offer computing power on order of hundreds of teraflops. A one-petaflop machine is expected in the next few years. These systems will provide a hardware and software infrastructure that is substantially different from existing cluster supercomputers. One of the critical challenges of such a system is the utilization of extra computing resources. We have demonstrated that the MD calculations are capable of exploiting additional computing resources that are at the expense of reduced memory capacities and bandwidth [14]. Hence we anticipate that the existing application frameworks will exploit the multi-core resource of the next-generation MPP systems effectively. Based on our preliminary profiling studies, we expect to achieve 50-100 ns/day of MD simulation for biomolecular systems with 500K to 1M atoms. By using the portable performance tools such as the Tool and Analysis Utilities (TAU) from the University of Oregon [15] and vendor-specific performance tools like the Cray Performance Analysis Toolset (PAT), we will investigate and explore the factors limiting the achievable performance of petascale software simulations and the underlying systems. Analysis of the performance data reveals the performance and scaling bottlenecks in the code implementation as well as in the algorithms. Collaboration with algorithm designers, scientific library developers, application scientists, and performance engineers will

enable the extreme-scale scaling of these bio-molecular application frameworks. In future, this task scope will also include the possibility of exploring alternative programming models for the application's key kernels in order exploit the underlying architecture efficiently. For example, the use of Co-Array Fortran on the Cray X1 has demonstrated significant performance benefits for Cray systems. The first goal of this strategy is to migrate LAMMPS, CHARMM, and Amber to existing high-performance hardware and verify them for correctness using the CHARMM force field against output from CHARMM on a trusted platform. A set of validation models has been created to compare the results of modified and new code with the trusted output for correctness of energies, forces, and trajectories. Once codes have been verified, optimization for both speed and parallel performance will be performed. A set of benchmarking simulations has been created to monitor the progress and performance of new work. A baseline of performance has been created for several platforms from which our improvements will be measured.

The simulations proposed here are expected to execute for weeks to months, a situation that requires additional support and capabilities from the target supercomputing systems as well as the applications. The mean time between failures (MTBF) for the latest generation of supercomputers with tens to hundreds of thousands of processors is projected to be a few hours. To benefit most from the computing power and to avoid loss of vital simulation data for long-running experiments, we are developing a new low-level optimized MD kernel that is able to self-monitor and is failure-tolerant. This work is being performed in collaboration with several hardware vendors including IBM, Cray, and SRC, as well as in collaboration with a SciDAC proposal "Center for Reliability, Availability and Serviceability for Petascale High-End Computing" (PI: Stephen Scott, ORNL). The novel capabilities of this MD kernel will include memory image check pointing for faster restart and progress migration to new processors. One of the novel features of this new MD kernel will be the ability to dynamically resize avoiding any lost productivity due to stopping and restarting. On the petascale machines failure of a single node will lead to the failure of the whole simulation. With this functionality the simulation will be able to continue on a decreased number of processors. In another situation, even on an oversubscribed machine there are idle processors when the scheduler is waiting for more processors to become available so that they can be assigned to the next job in queue. In such cases, as additional processors become available this novel MD kernel will expand on the additional processors; similarly shrink to fit a decreased number of processors and release processors to the system to be assigned other jobs. Such functionality will allow efficient utilization of the petascale computing resources. This powerful kernel will be designed to be modular and flexible. It will be able to be plugged with the functionalities of different programs such as CHARMM and AMBER, thus allowing the user to take advantage of these popular packages on the peta-scale systems. Moreover, we are working closely with hardware vendors to perform low-level optimization of each platform. thereby allowing science applications to benefit most from the available hardware.

Acknowledgments
This work is performed with support from the U.S. Department of Energy Office of Science through a SciDAC award titled "Understanding the Processivity of Cellobiohydrolase I."

[1] Schubert C 2006 Can biofuels finally take center stage *Nature Biotechnol.* **24** 777-784
[2] Lynd L R, Laser M S, Bransby D, Dale B E, Davison B, Hamilton R, Himmel M E, Keller M, McMillan J D, Sheehan J and Wyman C E 2008 How biotech can transform biofuels *Nature Biotechnol.* **26** 169-172
[3] Himmel M E, Ding S-Y, Johnson D K, Adney W S, Nimlos M R, Brady J W and Foust T D 2007 Biomass Recalcitrance: Engineering Plants and Enzymes for Biofuels Production *Science* **315** 804-807

[4] Nimlos M R, Matthews J F, Crowley M F, Walker R C, Chukkapalli G, Brady J W, Adney W S, Cleary J M, Zhong L and Himmel M E 2007 Molecular modeling suggests induced fit of Family I carbohydrate binding modules with a broken chain cellulose surface *Protein Engineering and Design* **20** 179-187
[5] Zhao X, Rignall T R, McCabe C, Adney W S and Himmel M E 2008 Molecular simulation evidence for processive motion of Trichoderma reesei Cel7A during cellulose depolymerization *Chem. Physics Lett.* (in press**)**
[6] Case D A, Cheatham III T E, Darden T, Gohlke H, Luo R, Merz Jr. K M, Onufriev A, Simmerling C, Wang B and Woods R J 2005 The Amber biomolecular simulation programs *J. Computational Chemistry* **26** 1668-1688
[7] Plimpton S 1995 Fast parallel algorithms for short-range molecular-dynamics *J. Computational Physics* **117** 1-19
[8] Phillips J C, Braun R, Wang W, Gumbart J, Tajkhorshid E, Villa E, Chipot C, Skeel R D, Kale L and Schulten K 2005 Scalable molecular dynamics with NAMD *J.f Computational Chemistry* **26** 1781-1802
[9] Brooks B R, Bruccoleri R E, Olafson B D, States D J, Swaminathan S and Karplus M 1983 CHARMM: A program for macromolecular energy, minimization, and dynamics calculations *J. Computational Chemistry* **4** 187-217
[10] Simonson T 2001 *Computational Biochemistry and Biophysics,* ed. O M Becker*, et al.* (New York: Marcel Dekker, Inc.)
[11] Torrie G M and Valleau J P 1977 Non-physical sampling distributions in Monte-Carlo free-energy estimation - umbrella sampling *J. Computational Physics* **23** 187-199
[12] Walker R C, Crowley M F and Case D A 2006 The implementation of a fast and efficient hybrid QM/MM potential method within Amber 9.0 Sander module *J. Comput. Chem.*
[13] Warshel A and Levitt M 1976 Theoretical studies of enzymic reactions - dielectric, electrostatic and steric stabilization of carbonium-ion in reaction of lysozyme *J. Molecular Biology* **103** 227-249
[14] Alam S, Vetter J S, Agarwal P K and Geist A 2006 Performance characterization of molecular dynamics techniques for biomolecular simulations. *Proc. 11th ACM SIGPLAN symposium on Principles and practice of parallel programming* (New York: ACM)
[15] Shende S and Malony A D 2006 The TAU parallel performance system *International J. High Performance Computing Applications* **20** 287-331