NVIDIA GPU Acceleration of Molecular Dynamics Simulations in AMBER

Supercomputing 2009
Portland, OR

Nov 17th 2009

By Ross C. Walker

The Project

- **A collaboration between NVIDIA and the AMBER Development Team.**

San Diego Supercomputer Center	NVIDIA
Ross C. Walker	Scott Le Grand
Mark J. Williamson	Duncan Poole

Molecular Dynamics?

“everything that living things do can be understood in terms of the jiggling and wiggling of atoms.”

Richard Feynman, 1963

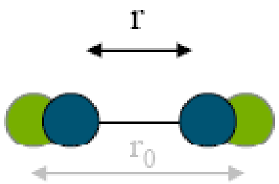
What is Molecular Dynamics?

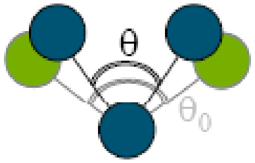
- **In the context of this talk:**
  - The simulation of the dynamical properties of condensed phase biological systems.
    - Enzymes / Proteins
    - Drug Molecules
    - Biological Catalysts
  - Classical Energy Function
    - Force Fields
    - Parameterized (Bonds, Angles, Dihedrals, VDW, Charges…)
    - Integration of Newton's equations of motion.
    - Atoms modeled as points, electrons included implicitly within the parameterization.

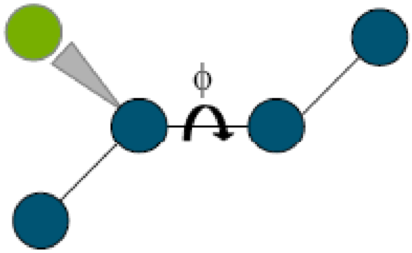
Why Molecular Dynamics?

- **Atoms move!**
  - – We may be interested in studying time dependent phenomena, such as molecular vibrations, phonons, diffusion, etc.
  - – We may be interested in studying temperature dependant phenomena, such as free energies, anharmonic effects,
  - – etc.
- **Ergodic Hypothesis**
  - – Time average over trajectory is equivalent to an ensemble average.
  - – Allows the use of MD for statistical mechanics studies.

Force Fields







$$E_{stretching} = \sum_{1,2\,pairs} K_r (r - r_0)^2$$

$$E_{bending} = \sum_{1,2\,pairs} K_\theta (\theta - \theta_0)^2$$

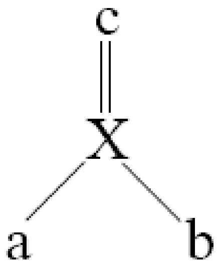
$$E_{torsion} = \sum_{1,4\,pairs} K_\phi (1 - \cos(n\phi - \delta\ ))$$

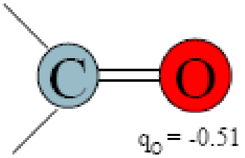
$$\boxed{E = E_{stretch} + E_{bend} + E_{torsion} + E_{impropers} + E_{electrostatic} + E_{vanderWaals}}$$

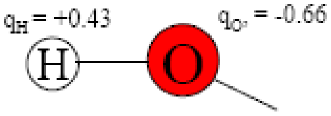
$$E_{impropers} = \sum_{impr.} K_\varphi (\varphi - \varphi_0\ )^2$$

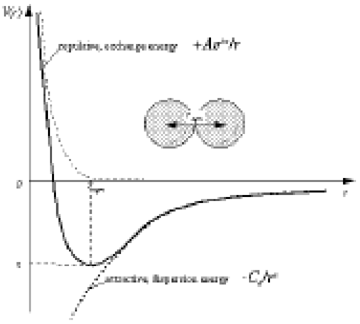
$$E_{electrostatic} = \sum_{\substack{\text{nonbonded} \\ \text{i-k pairs}}} q_i \cdot q_k / D.r_{ik}$$

$$E_{van-der-Waals} = \sum_{\substack{nonbonded \\ pairs}} \left(\frac{\boxed{A_{ik}}}{r_{ik}^{12}} - \frac{\boxed{C_{ik}}}{r_{ik}^{6}} \right)$$









What is AMBER?

An MD simulation package

10 Versions as of 2008

distributed in two parts:

- *AmberTools*, preparatory and analysis programs, free under GPL

- *Amber* the main simulation programs, under academic licensing

independent from the accompanying forcefields

A set of MD forcefields

fixed charge, biomolecular forcefields: ff94, ff99, ff99SB, ff03

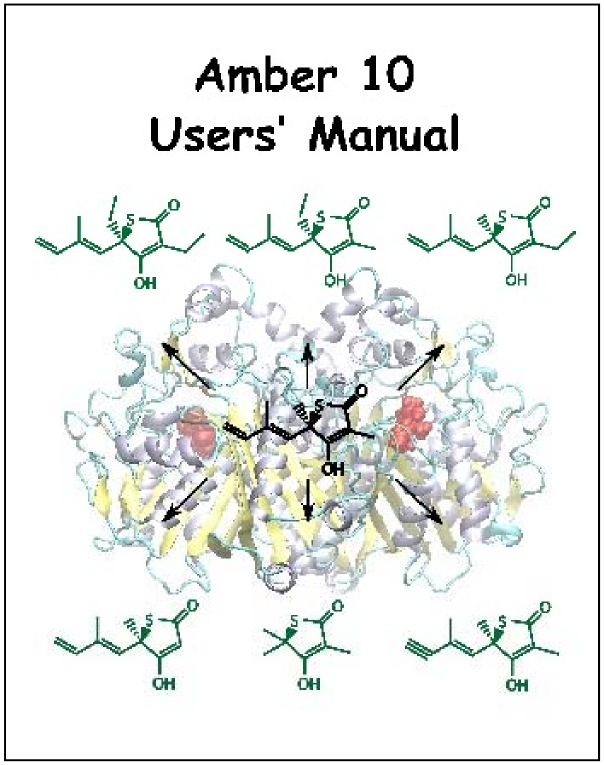
experimental polarizable forcefields e.g. ff02EP

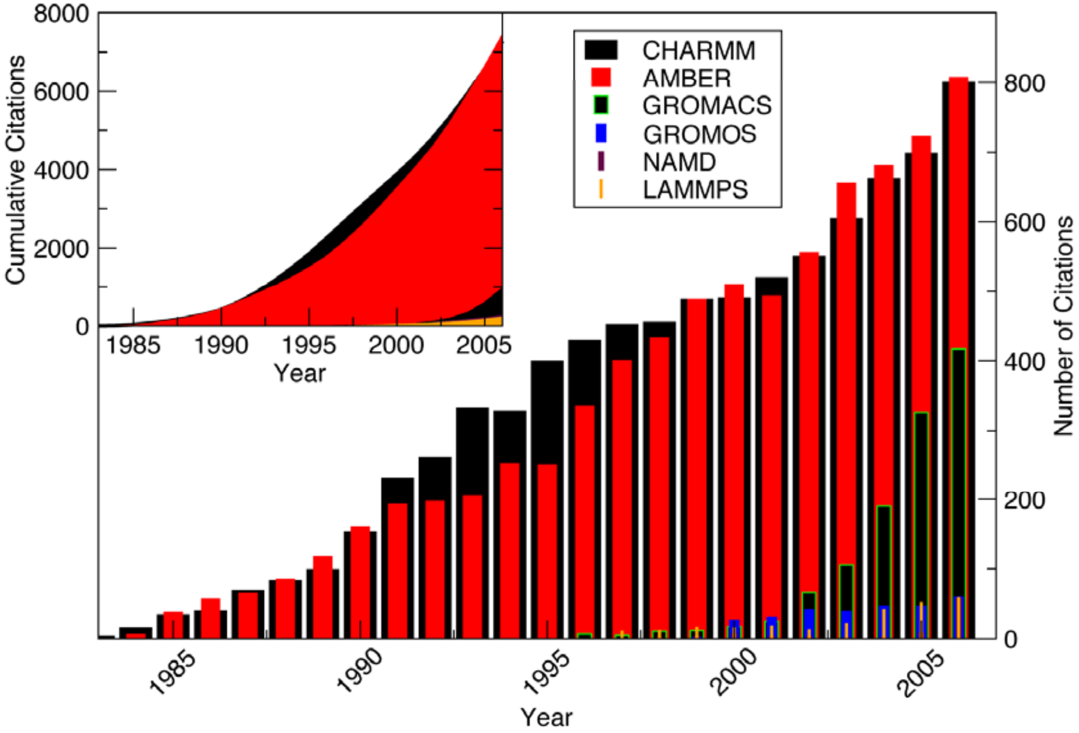
parameters for general organic molecules, solvents, carbohydrates (Glycam), etc.

in the public domain

AMBER Usage

- **Approximately 650 site licenses across most major countries.**





The AMBER Force Field Equation

$$V\left(r^{n}\right)=\sum_{bonds} K_{r}\left(r-r_{eq}\right)^{2}+\sum_{angles} K_{\theta}\left(\theta-\theta_{eq}\right)^{2}$$

$$+\sum_{dihedrals} \frac{V_{n}}{2}\left[1+\cos\left(n\phi-\gamma\right)\right]+\sum_{i<j}\left[\frac{A_{ij}}{R_{ij}^{12}}-\frac{B_{ij}}{R_{ij}^{6}}+\frac{q_{i}q_{j}}{\varepsilon_{r}R_{ij}}\right]$$

Note: 1-2 and 1-3 non-bond interactions are parameterized into the bond and angle terms. Dihedral term also includes some of the non-bond interaction.

1-4 EEL scaled by 1.2
1-4 VDW scaled by 2.0

What can we do with Molecular Dynamics?

- **Can simulate time dependent properties.**
  - Protein domain motions.
  - Small Protein Folds.
  - Spectroscopic Properties.

- **Can simulate ensemble properties.**
  - Binding free energies.
    - Drug Design
    - Biocatalyst Design
  - Reaction Pathways
  - Free Energy Surfaces.

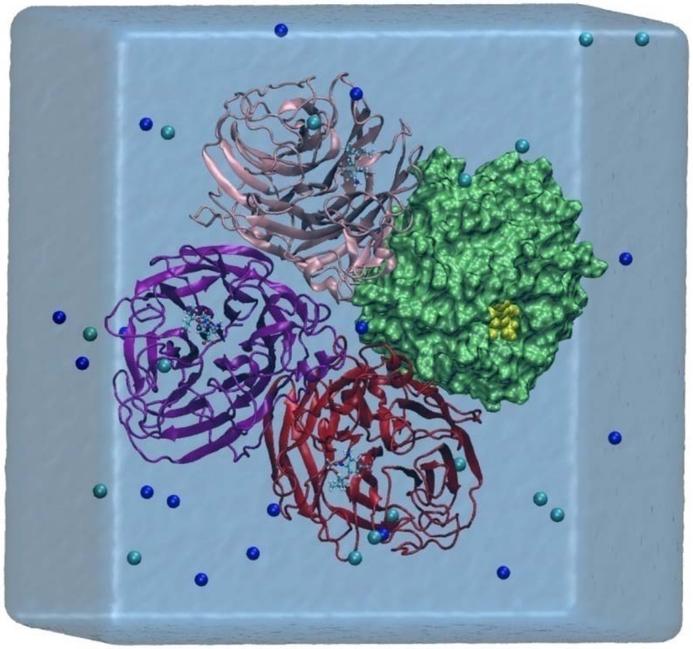
Two Specific Examples

- Design of New Antiviral Flu Drugs

- Improved Bioethanol Production

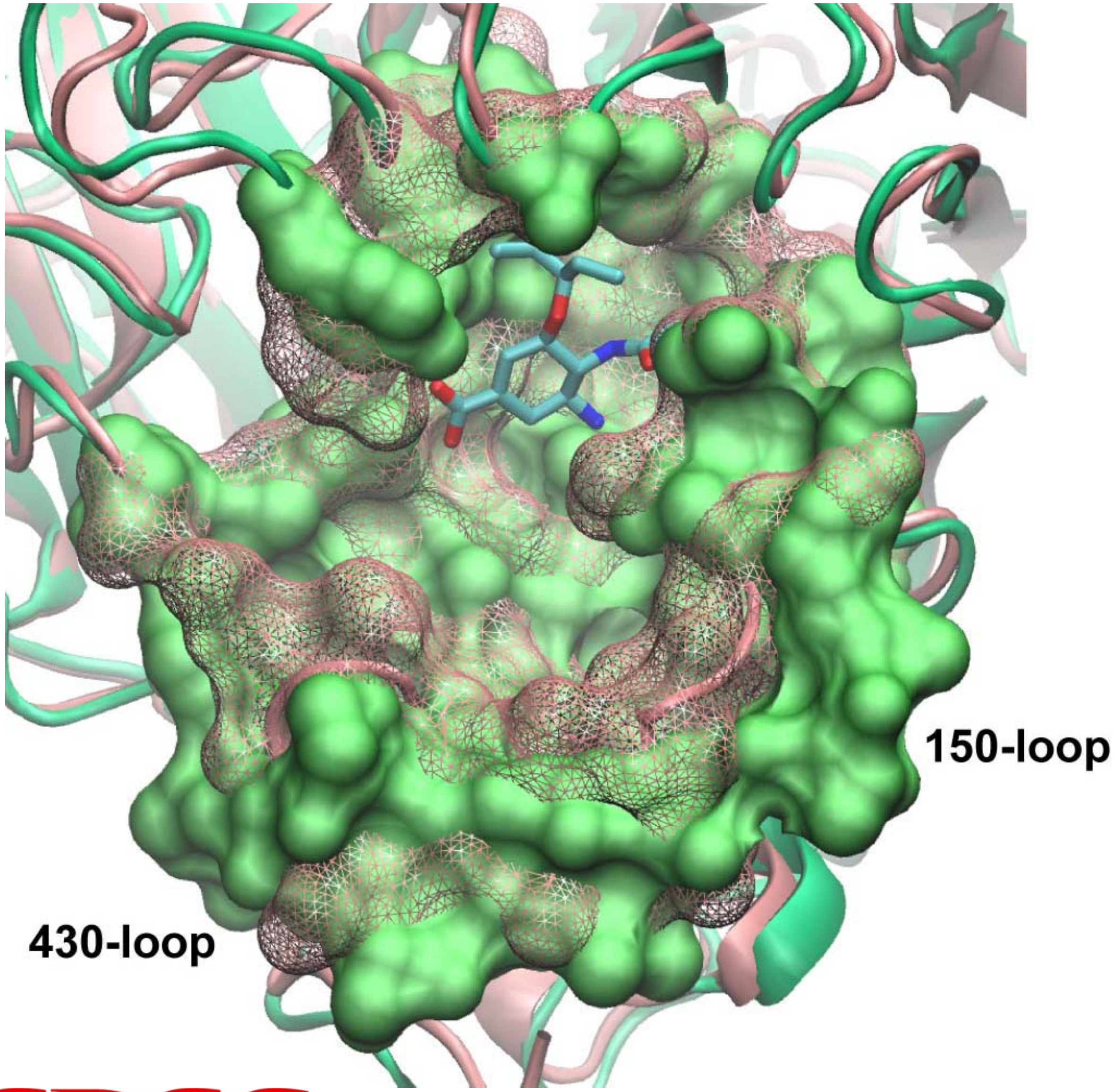
Neuraminidase Simulations

- **H1N1 Swine Flu.**
  - Tamiflu antiviral drug optimization
  - Collaborative project between:
    - Ross Walker (SDSC)
    - Adrian Roitberg (UFL)
    - Rommie Amaro (UC Irvine)
    - Andy McCammon (UCSD)



  - 7 Systems
  - *ca. 125K atoms each.*
  - *100ns per system run in 2 weeks on NICS Athena. (Previously this would have been impossible)*

Implications for Antiviral Drug Design



430-loop and 150-loop very flexible

Structural reorganization reveals new 150- , 430- cavities

These new structural insightsmay help in developing new antivirals that are:

(a) more specific for N1

(b) less amenable to resistance

Improved Bioethanol Production

Biomass

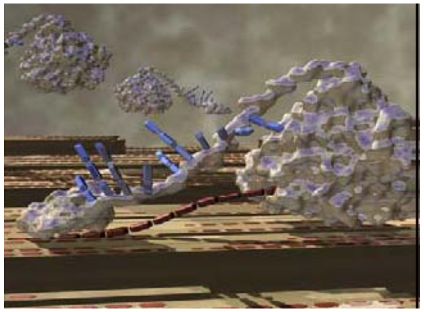
1B Ton annual potential: stover, bagasse, Miscanthus switch grass, poplar trees

Barriers to biofuel production

1. **Dedicated plant crops** to produce cellulose biomass
2. **Efficient enzymes** for digesting cellulose down to sugars
3. **Engineered microbes** that ferment all sugars to ethanol

Pretreatment

biomass to accesssible cellulose



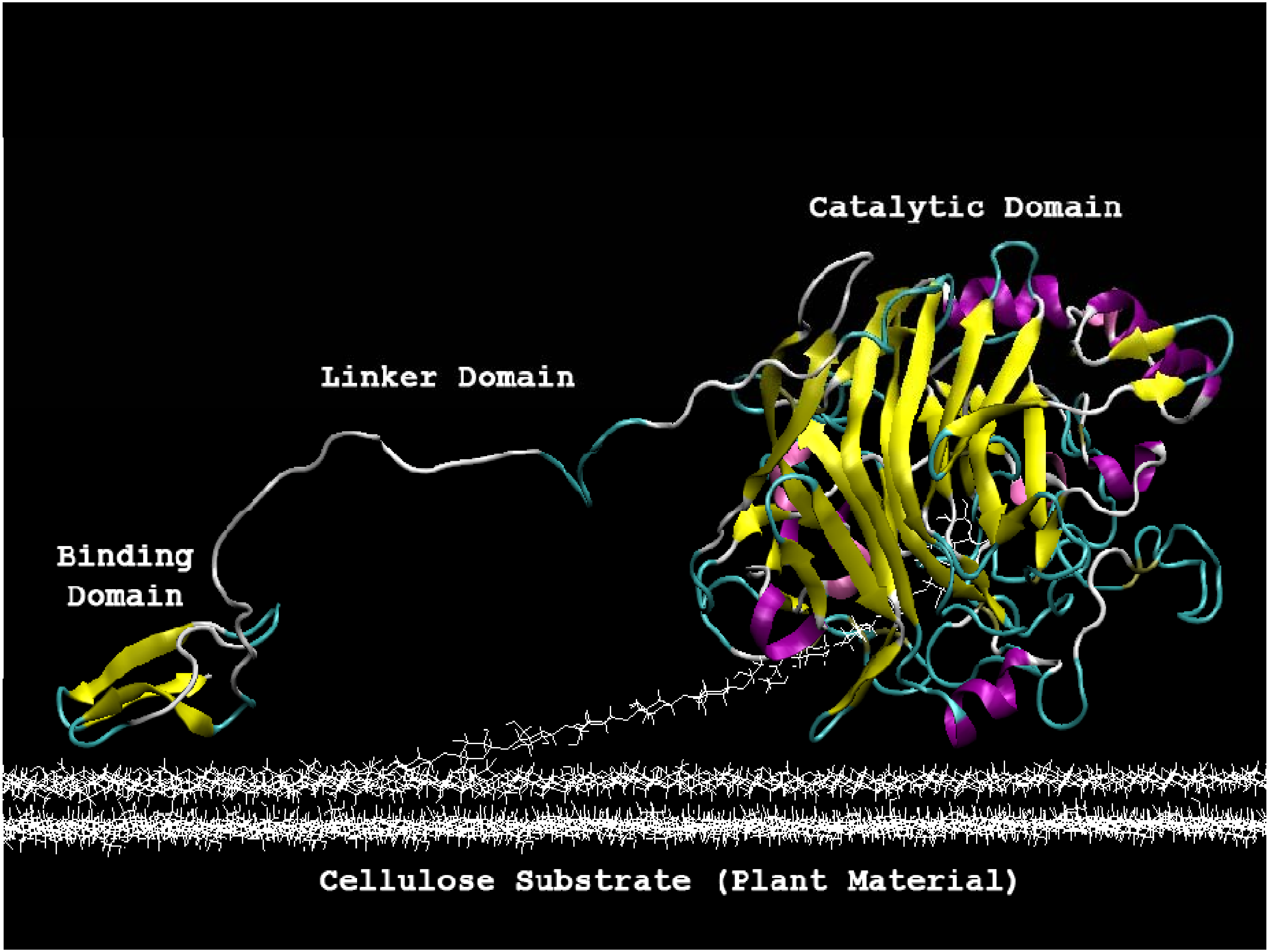
Enzyme Digestion

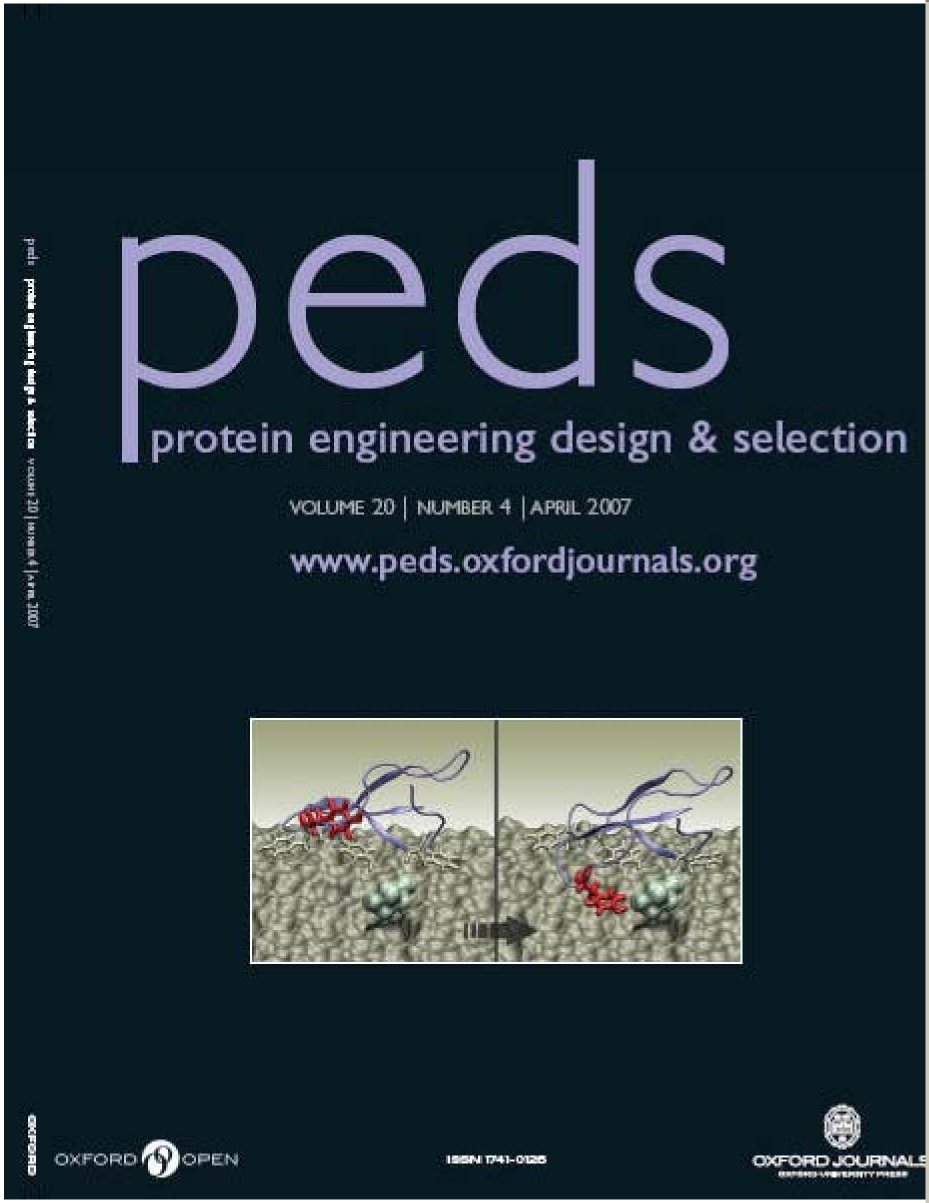
cellulose to glucose

Fermentation

glucose to ethanol & other biofuels

Process goal –conversion of biomass to biofuel faster and cheaper making alternative fuels economically feasible


Catalytic Domain
Linker Domain
Binding Domain
Cellulose Substrate (Plant Material)



peds

protein engineering design & selection

VOLUME 20 | NUMBER 4 | APRIL 2007

www.peds.oxfordjournals.org

OXFORD OPEN

ISSN 1741-0126

OXFORD JOURNALS

Protein Engineering, Design & Selection vol. 20 no. 4 pp. 179–187, 2007
Published online April 12, 2007 doi:10.1093/protein/gzm010

Molecular modeling suggests induced fit of Family I carbohydrate-binding modules with a broken-chain cellulose surface

Mark R. Nimlos[1,*], James F. Matthews[1], Michael F. Crowley[1], Ross C. Walker[3], Giridhar Chukkapalli[4], John W. Brady[2], William S. Adney[1], Joseph M. Cleary[4], Linghao Zhong[5] and Michael E. Himmel[1]

[1]National Renewable Energy Laboratory, Golden, CO 80401, USA, [2]Department of Food Science, Cornell University, Ithaca, NY 14853-7201, USA, [3]The Scripps Research Institute, La Jolla, CA 92037, USA, [4]San Diego Supercomputer Center, La Jolla, CA 92093-0505, USA and [5]Pennsylvania State University, Mont Alto, PA 17237, USA

[*]To whom correspondence should be addressed.
E-mail: mark_nimlos@nrel.gov

Cellobiohydrolases are the most effective single component of fungal cellulase systems; however, their molecular mode of action on cellulose is not well understood. These enzymes act to detach and hydrolyze cellodextrin chains from crystalline cellulose in a processive manner, and the carbohydrate-binding module (CBM) is thought to play an important role in this process. Understanding the interactions between the CBM and cellulose at the molecular level can assist greatly in formulating selective mutagenesis experiments to confirm the function of the CBM. Computational molecular dynamics was used to investigate the interaction of the CBM from *Trichoderma reesei* cellobiohydrolase I with a model of the (1,0,0) cellulose surface modified to display a broken chain. Initially, the CBM was located in different positions relative to the reducing end of this break, and during the simulations it appeared to translate freely and randomly across the cellulose surface, which is consistent with its role in processivity. Another important finding is that the reducing end of a cellulose chain appears to induce a conformational change in the CBM. Simulations show that the tyrosine residues on the hydrophobic surface of the CBM, Y5, Y31 and Y32 align with the cellulose chain adjacent to the reducing end and, importantly, that the fourth tyrosine residue in the CBM (Y13) moves from its internal position to form van der Waals interactions with the cellulose surface. As a consequence of this induced change near the surface, the CBM straddles the reducing end of the broken chain. Interestingly, all four aromatic residues are highly conserved in Family I CBM, and thus this recognition mechanism may be universal to this family.

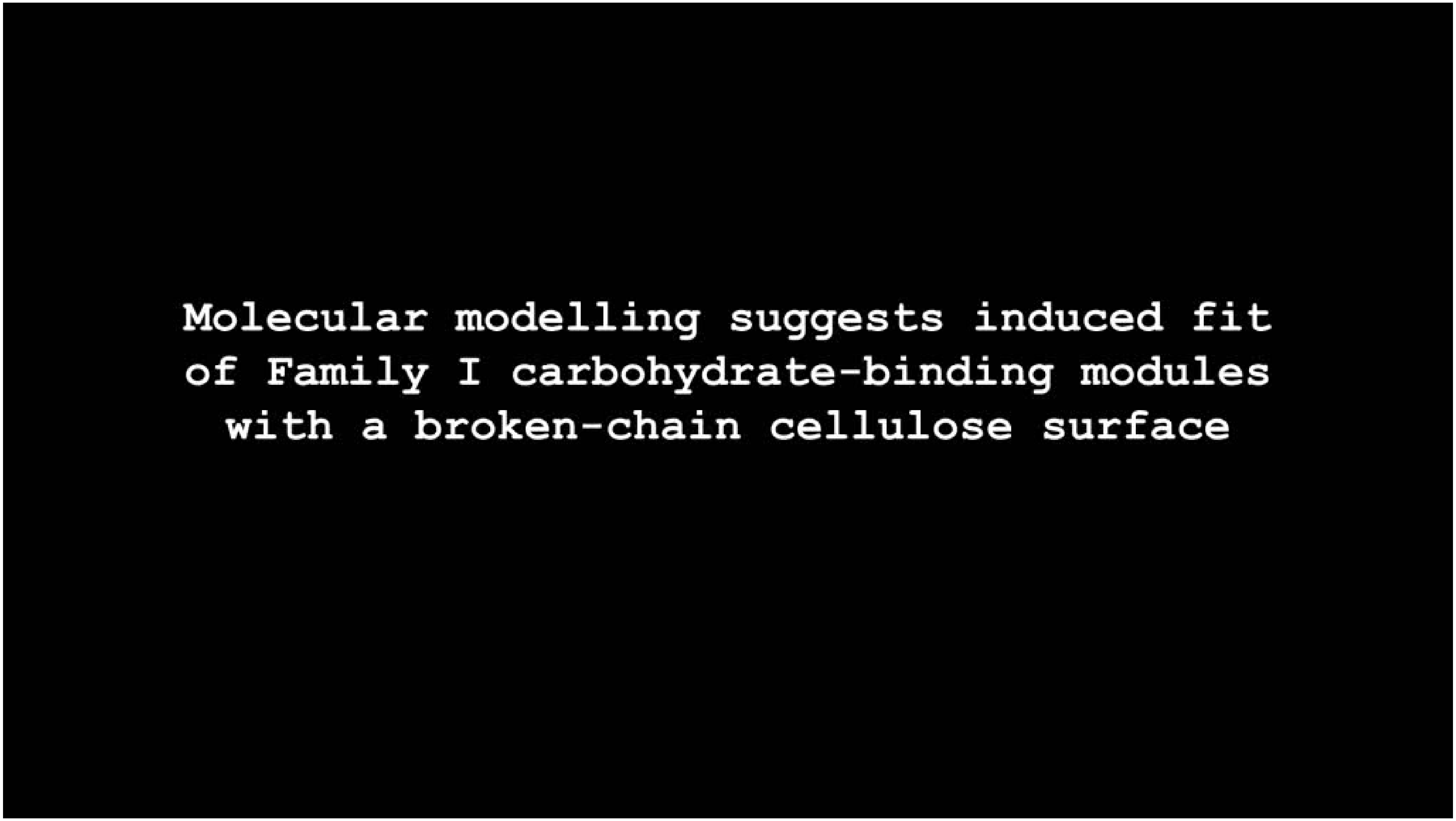
Keywords: biomass/cellulase/cellulose/induced fit/molecular dynamics

Introduction

Cellobiohydrolases (CBH-EC 4.2.1.91) are known to display a multi-domain structure (Teeri *et al.*, 1992) characterized by a catalytic domain (core) and a carbohydrate-binding module (CBM) separated by a linker peptide (Srisodsuk *et al.*, 1993) rich in proline, serine and threonine (Teeri *et al.*, 1987). The non-catalytic CBMs are recognized as being an essential component of effective cellulase action on cellulose substrate (Gilkes *et al.*, 1988; Reinikainen *et al.*, 1991; Reinikainen *et al.*, 1992; Kraus *et al.*, 1995) and are thought to have three primary functions including proximity effects, substrate targeting and microcrystallite disruption (Knowles *et al.*, 1991; Linder and Teeri, 1997; Boraston *et al.*, 2004). CBMs function as a means to promote the association of the enzyme with the substrate and subsequently increase the effective cellulase concentration (proximity effect) (Reinikainen *et al.*, 1991, 1992). CBMs have been shown to have selective affinities for substrates including crystalline and amorphous celluloses, and various soluble and non-soluble polysaccharides (targeting function) (Lamed *et al.*, 1994; Tomme *et al.*, 1995; Creagh *et al.*, 1996; Tormo *et al.*, 1996; Linder and Teeri 1997; Henshaw *et al.*, 2004). In addition, some CBMs appear to disrupt the structures of the carbohydrate ligands to render the substrate more susceptible to enzymatic function (disruptive function) (Din *et al.*, 1994). Of particular interest to biomass conversion are CBMs that target crystalline cellulose (Phelps *et al.*, 1995; Teeri, 1997; Boraston *et al.*, 2004), which forms the core of carbohydrate microfibrils that provide structure and strength to plant cell walls. It has long been recognized that crystalline cellulose is recalcitrant to enzymatic hydrolysis, limiting its conversion rate in many processes based on the production of fermentable sugars from plant biomass.

Family I CBMs, which are entirely fungal, are perhaps the most interesting. Of particular interest is the CBH I CBM produced by *Trichoderma reesei*, the most common source of commercial cellulases today. CBH I is a 'molecular machine', that is thought to be processive (Rouvinen *et al.*, 1990; Vrsanska and Biely, 1992; Barr *et al.*, 1996), moving along a crystalline cellulose chain, 'pulling up' that chain and feeding it into the catalytic domain where cellobiose is formed by hydrolyzing alternate β-(1,4) glycosidic linkages. This exoglucanase is most efficient at hydrolyzing crystalline cellulose when mixed with endoglucanases, whose apparent role is to hydrolyze random single glycosidic linkages. CBH I is known to act upon the reducing end of an already broken chain of cellulose (Vrsanska and Biely, 1992). The processivity of these enzymes makes them highly active and attractive for bioprocessing crystalline cellulose found in plant cell walls. However, details of the functions of the different components of these enzymes during processivity remain unclear, if not controversial.

The structures of Family I CBMs are thought to display specificity for binding to crystalline cellulose through three aromatic amino acid side groups located on a relatively planar surface of the protein. These residues are nearly co-linear, and the distance between them is about the same as the cellobiose unit length (~1.1nm) in the cellulose



Molecular modelling suggests induced fit of Family I carbohydrate-binding modules with a broken-chain cellulose surface

GPU Support in AMBER

GPU Support

- **Collaboration with NVIDIA to produce CUDA version of AMBER.**
  - PMEMD Engine
  - Implicit Solvent GB (V1.0 complete)
  - Explicit Solvent PME (in progress)

- **Focus on accuracy.**
  - It MUST pass the AMBER regression tests.
  - Energy conservation MUST be comparable to double precision CPU code.

Approach

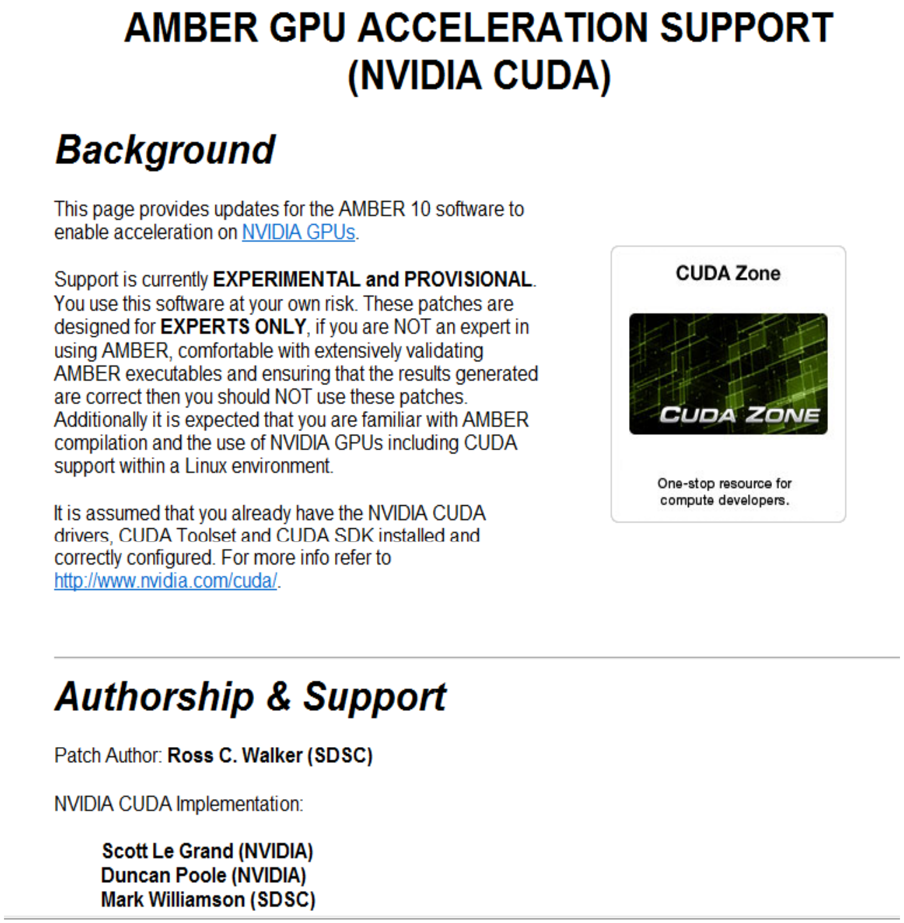
- Utilize current code as ‘Driver Engine’

```
#ifdef CUDA
      call gpu_upload_crd(crd)
#endif
      call gb_force(atm_cnt, crd, frc, gb_pot_ene, irespa)
#ifdef CUDA
      call gpu_download_frc(frc)
#endif
```

Support

- Currently provided as a patch to AMBER 10.
- Will be standard for AMBER 11.
- Designed to be completely transparent to the user. Simply use pmemd.cuda in place of pmemd.

http://ambermd.org/gpus/



AMBER GPU ACCELERATION SUPPORT (NVIDIA CUDA)

Background

This page provides updates for the AMBER 10 software to enable acceleration on NVIDIA GPUs.

Support is currently **EXPERIMENTAL and PROVISIONAL**. You use this software at your own risk. These patches are designed for **EXPERTS ONLY**, if you are NOT an expert in using AMBER, comfortable with extensively validating AMBER executables and ensuring that the results generated are correct then you should NOT use these patches. Additionally it is expected that you are familiar with AMBER compilation and the use of NVIDIA GPUs including CUDA support within a Linux environment.

It is assumed that you already have the NVIDIA CUDA drivers, CUDA Toolset and CUDA SDK installed and correctly configured. For more info refer to http://www.nvidia.com/cuda/.

CUDA Zone

One-stop resource for compute developers.

Authorship & Support

Patch Author: **Ross C. Walker (SDSC)**

NVIDIA CUDA Implementation:

Scott Le Grand (NVIDIA)
Duncan Poole (NVIDIA)
Mark Williamson (SDSC)

GPU Device Info

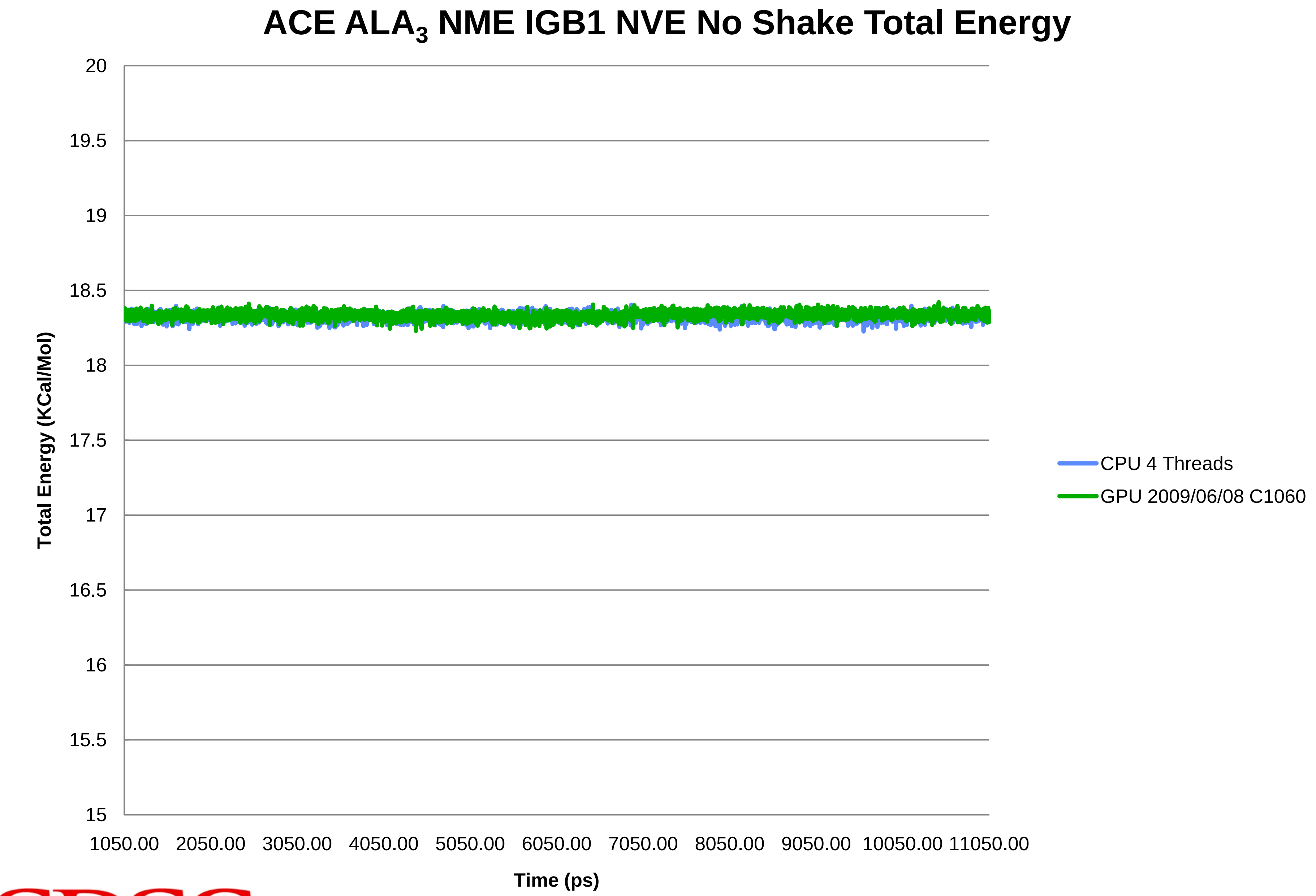
```
|------------------ GPU DEVICE INFO --------------------
|
|   CUDA Capable Devices Detected:      4
|           CUDA Device ID in use:      3
|                CUDA Device Name: GeForce GTX 295
|     CUDA Device Global Mem Size:    895 MB
| CUDA Device Num Multiprocessors:     30
|           CUDA Device Core Freq:   1.24 GHz
|
|------------------------------------------------------
```

Currently Supported Devices:

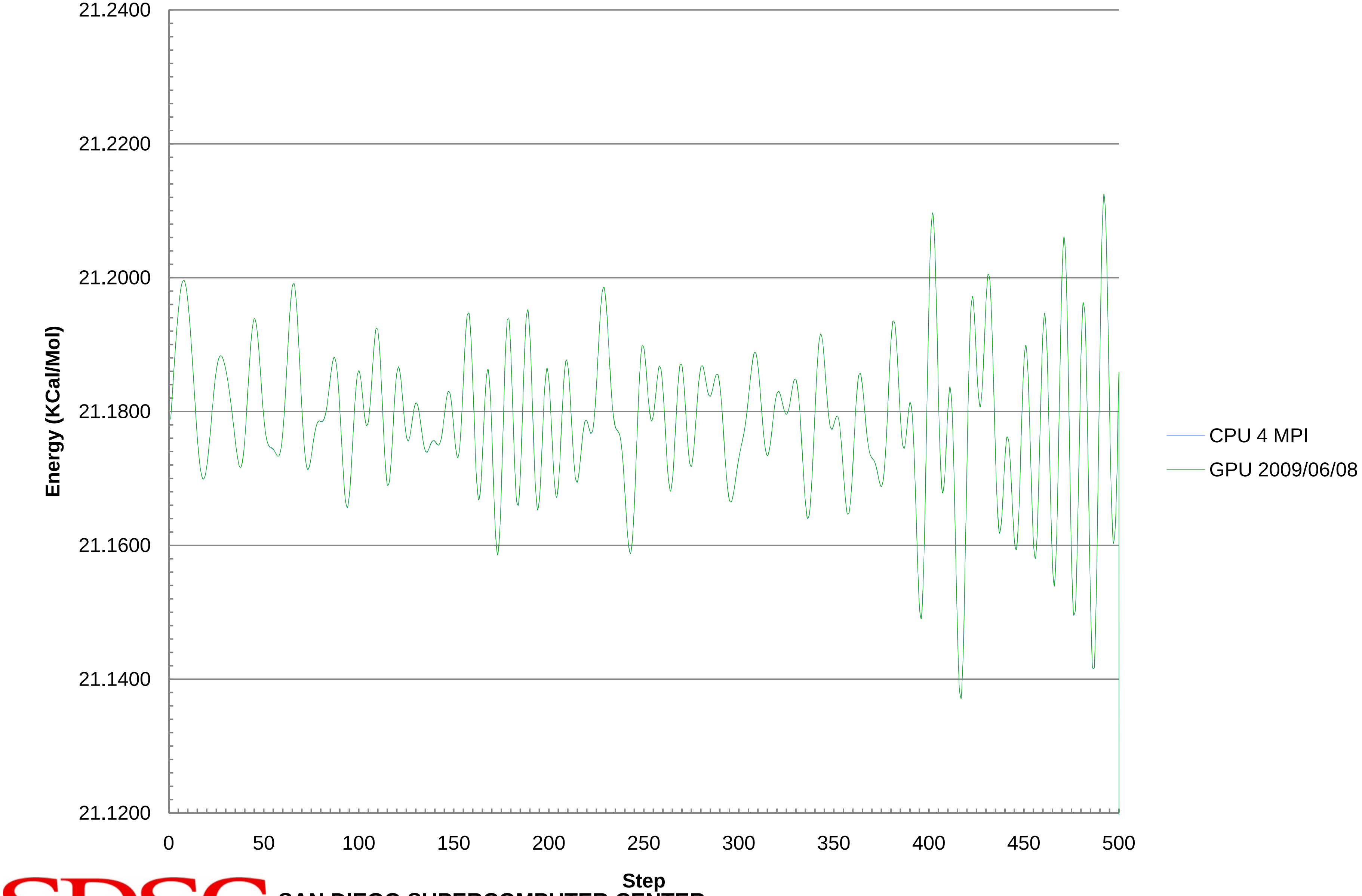
C1060, S1070, GTX295, GTX285, GTX275, FX5400
FX4500 + Newer models as available.

GPU Accuracy

- **Use of double precision in all places severely limits performance.**
  - Make careful use of double precision where needed.
  - Calculate in single precision.
  - Accumulate in double precision.
  - Avoid large dynamic ranges in arithmetic expressions.
  - Switch over to double precision automatically if the dynamic range is too high.


ACE ALA$_3$ NME IGB1 NVE No Shake Total Energy
20
19.5
19
18.5
18
17.5
17
16.5
16
15.5
15
Total Energy (KCal/Mol)
1050.00
2050.00
3050.00
4050.00
5050.00
6050.00
7050.00
8050.00
9050.00
10050.00
11050.00
Time (ps)
CPU 4 Threads
GPU 2009/06/08 C1060

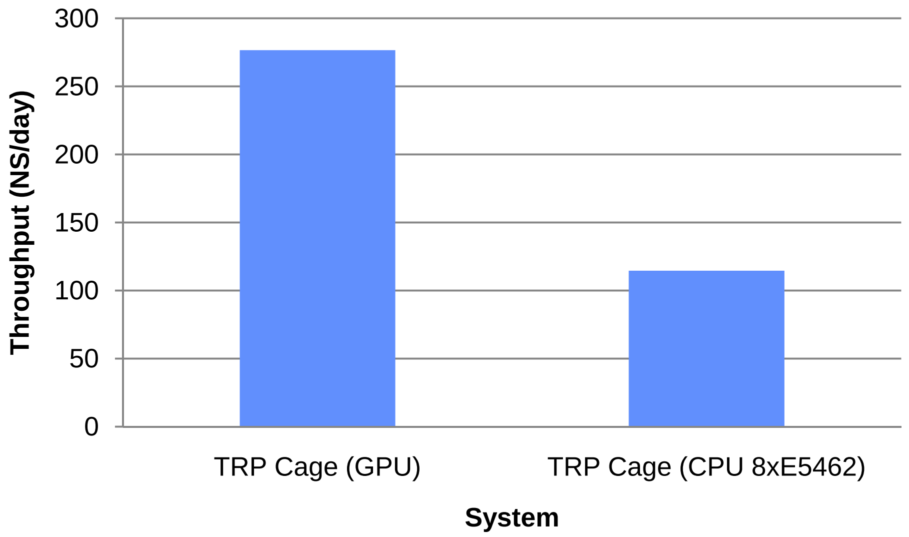
IGB1 NVE No Shake Total E 0.5fs Time Step (NSCM = 1000)



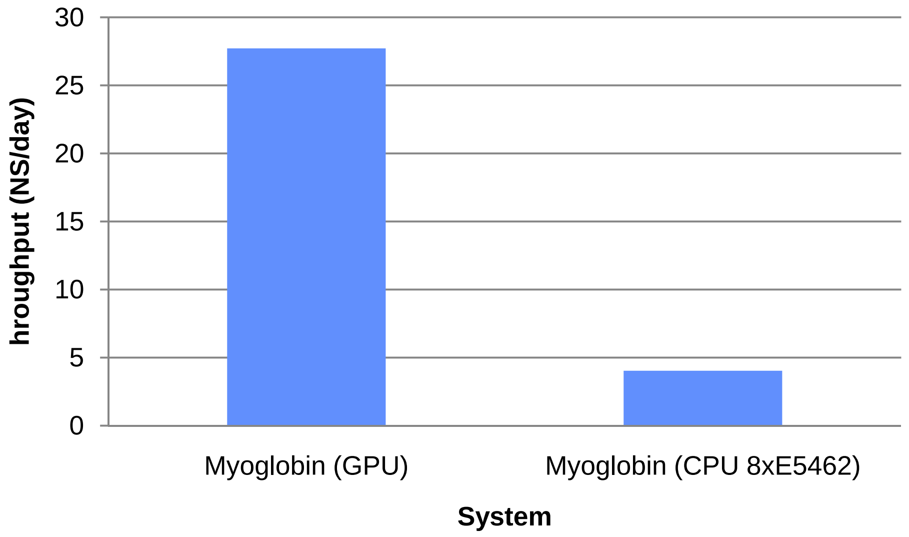
Provisional Performance

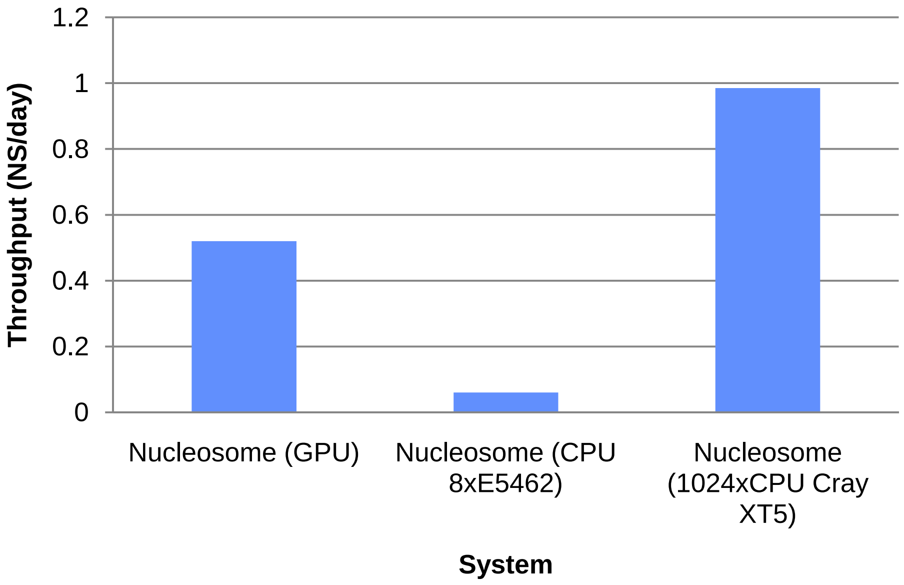
- **1 x Tesla C1060**
  - Implicit Solvent Generalized Born Calculations.
  - Multi-GPU support is work in progress.

```
Timings
-------
System               Wall Time (s)      NS/day
TRPCage(GPU)          62.48             276.57
TRPCage(8xCPU)       150.84             114.56

Myoglobin(GPU)        62.22              27.72
Myoglobin(8xCPU)     392.34               4.04

Nucleosome(GPU)      332.03               0.520
Nucleosome(8xCPU)   2877.60               0.060
Cray XT5 (1024xCPU)  175.49               0.985
```



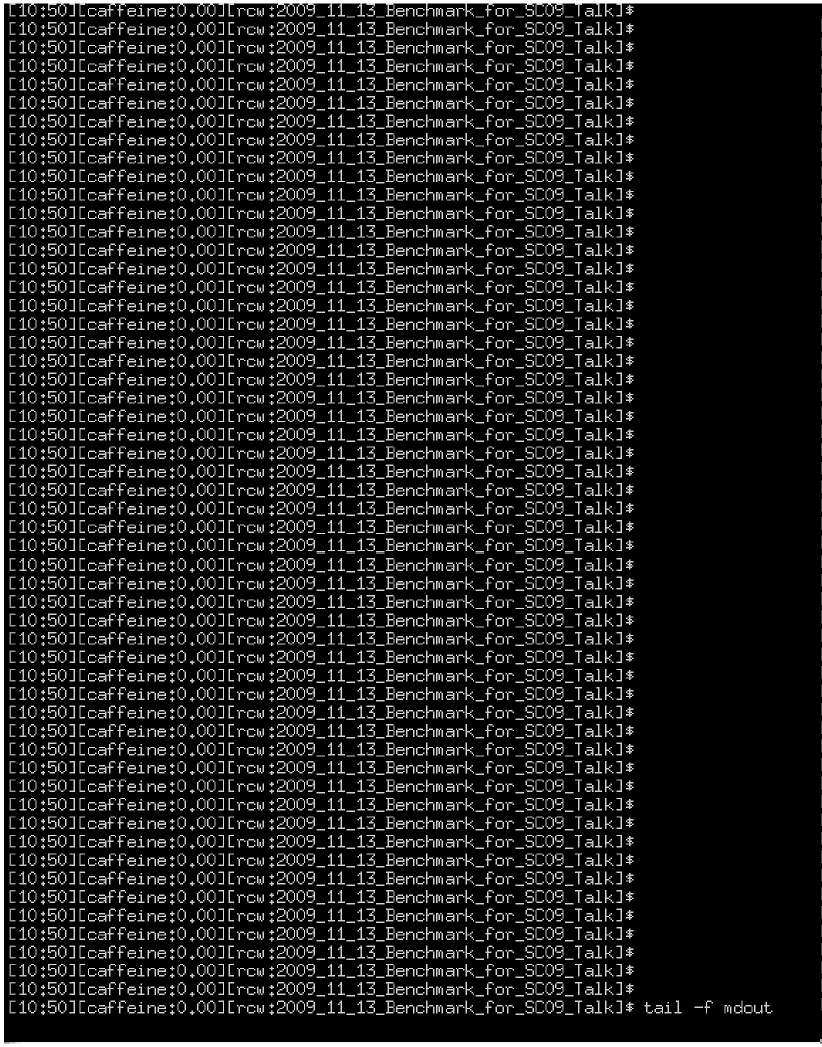
As expected the performance differential is larger for bigger systems.





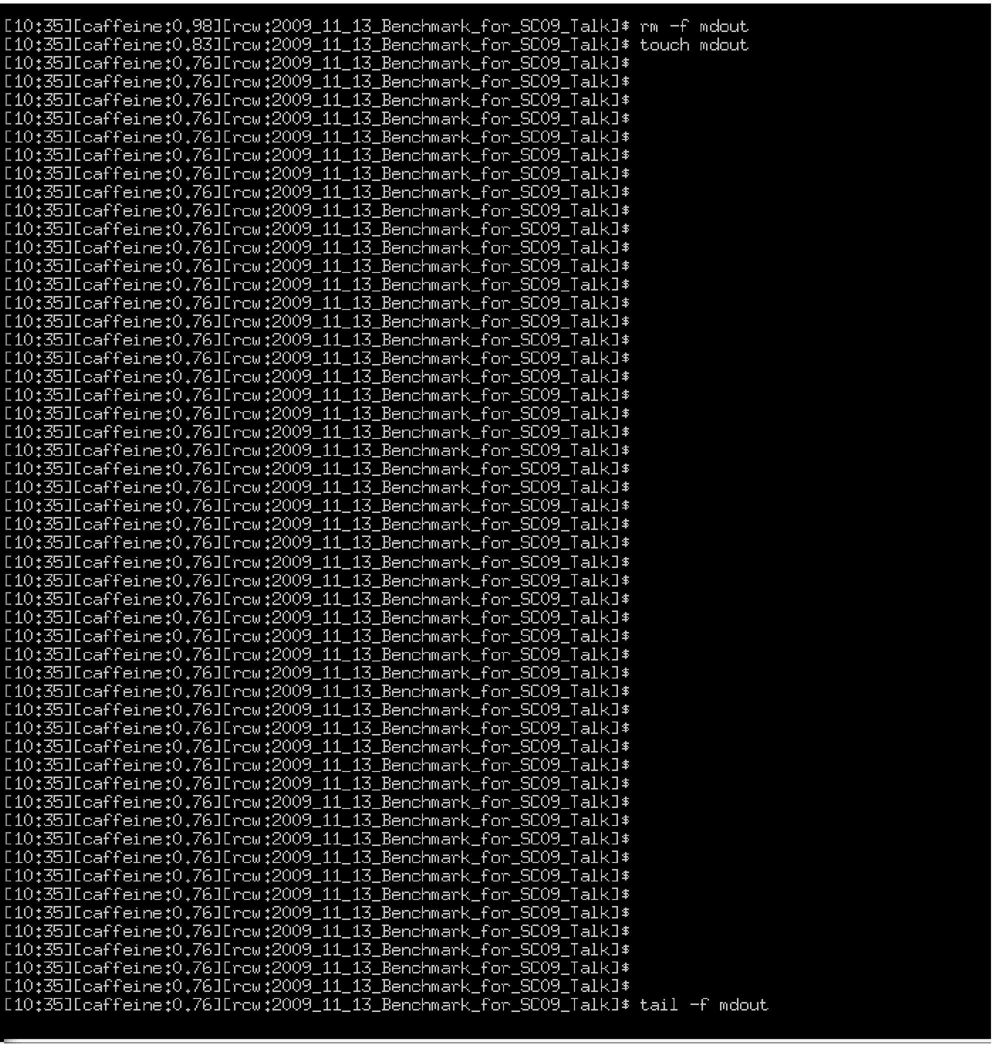
Performance Example (TRP Cage)

CPU E5462



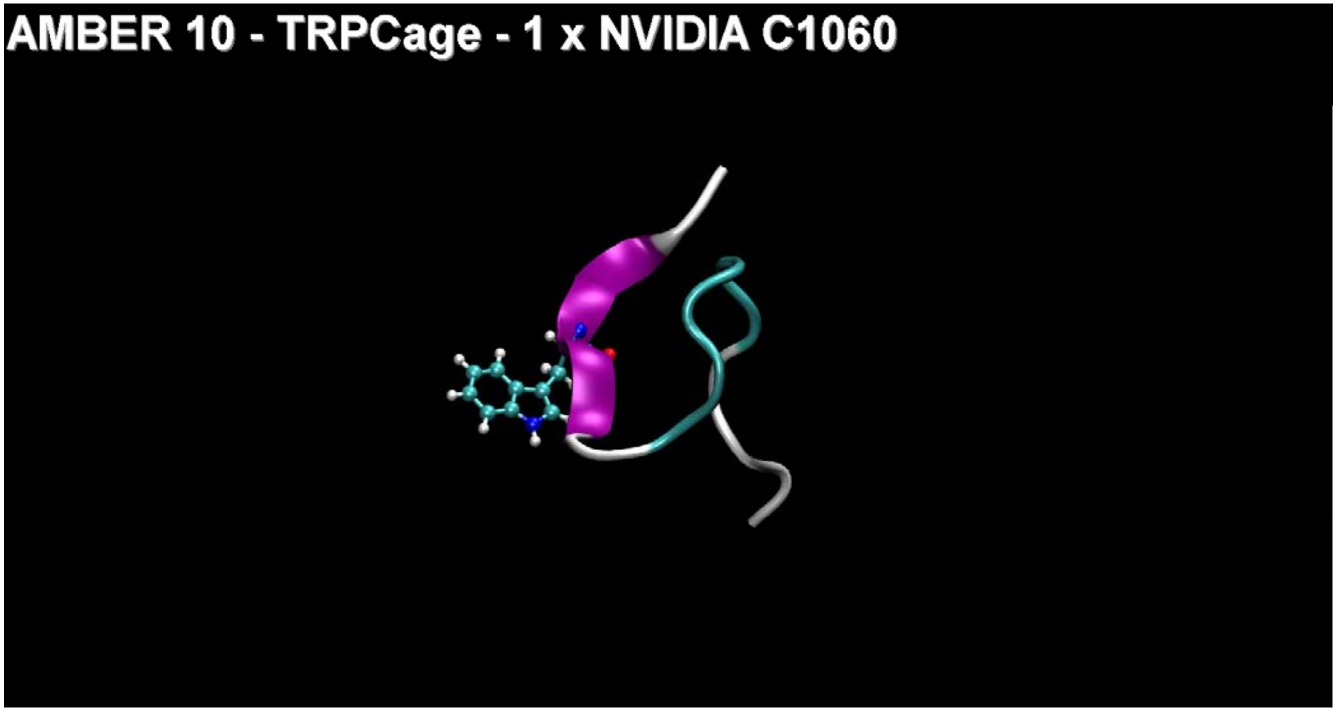
```
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:50][caffeine:0.00][rcw:2009_11_13_Benchmark_for_SC09_Talk]$ tail -f mdout
```

GPU C1060



```
[10:35][caffeine:0.98][rcw:2009_11_13_Benchmark_for_SC09_Talk]$ rm -f mdout
[10:35][caffeine:0.83][rcw:2009_11_13_Benchmark_for_SC09_Talk]$ touch mdout
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$
[10:35][caffeine:0.76][rcw:2009_11_13_Benchmark_for_SC09_Talk]$ tail -f mdout
```

AMBER 10 - TRP Cage - 1 x E5462 cpu


AMBER 10 - TRPCage - 1 x NVIDIA C1060

Future Work

- **Further optimization.**
  - Explicit Solvent Particle Mesh Ewald.
    - Currently equivalent to 16 x E5462 cores.
    - Expect 3x performance improvement over the next month.
  - Implicit Solvent Generalized Born.
    - Speedups *ca.* 23x.
    - Expect another 2x performance improvement over the next few months.
  - Support for Multiple GPUs.
    - C1070’s (4 x tesla)
    - Fermi Optimizations.