The Computational Microscope

Computational microscope views at atomic resolution ...

... how living cells maintain health and battle disease
Our Microscope is Made of...

**Chemistry**

\[ U(\vec{R}) = \sum_{\text{bonds}} k_{\text{b}} (r_i - r_0)^2 + \sum_{\text{angles}} k_{\text{a}} (\theta_i - \theta_0)^2 + \sum_{\text{dihedrals}} k_{\text{d}} \left[ 1 + \cos \left( n_i \phi_i + \delta_i \right) \right] + \sum_{\text{other}} \]

**Physics**

\[ m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{F}_i = -\nabla U(\vec{R}) \]

**Math**

\[ \vec{r}_i(t + \Delta t) = 2\vec{r}_i(t) - \vec{r}_i(t - \Delta t) + \frac{\Delta t^2}{m_i} \vec{F}_i(t) \]

(repeat *one billion times* = microsecond)
The Molecular Dynamics Simulation Process

For textbooks see:


More at http://www.biomath.nyu.edu/index/course/99/textbooks.html
Classical Dynamics

at 300K

Energy function: \[ U(\vec{r}_1, \vec{r}_2, \cdots \vec{r}_N) = U(\vec{R}) \]

used to determine the force on each atom:

\[ m_i \frac{d^2\vec{r}_i}{dt^2} = \vec{F}_i = -\vec{\nabla}U(\vec{R}) \]

yields a set of 3N coupled 2\textsuperscript{nd}-order differential equations that can be propagated forward (or backward) in time.

Initial coordinates obtained from crystal structure, velocities taken at random from Boltzmann distribution.

Maintain appropriate temperature by adjusting velocities.
Classical Dynamics

discretization in time for computing

\[ m_i \frac{d^2 \vec{r}_i}{dt^2} = \vec{F}_i = -\vec{\nabla}U(\vec{R}) \]

Use positions and accelerations at time \( t \) and the positions from time \( t-\delta t \) to calculate new positions at time \( t+\delta t \).

\[ \vec{r}(t + \delta t) \approx \vec{r}(t) + \vec{v}(t)\delta t + \frac{1}{2}\vec{a}(t)\delta t^2 \]

\[ \vec{r}(t - \delta t) \approx \vec{r}(t) - \vec{v}(t)\delta t + \frac{1}{2}\vec{a}(t)\delta t^2 \]

“Verlet algorithm”

\[ \vec{r}(t + \delta t) \approx 2\vec{r}(t) - \vec{r}(t - \delta t) + \vec{a}(t)\delta t^2 \]
Potential Energy Function of Biopolymer

- Simple, fixed algebraic form for every type of interaction.
- Variable parameters depend on types of atoms involved.

\[ U(\vec{R}) = \sum_{\text{bonds}} k_i^{\text{bond}} (r_i - r_0)^2 + \sum_{\text{angles}} k_i^{\text{angle}} (\theta_i - \theta_0)^2 + \sum_{\text{dihedrals}} k_i^{\text{dihedral}} [1 + \cos (n_i \phi_i + \delta_i)] + \sum \sum 4\epsilon_{ij} \left[ \left( \frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] + \sum \sum \frac{q_i q_j}{\epsilon r_{ij}} \]

Parameters:
- “force field” like Amber, Charmm; note version number

Heuristic from physics
• Atomic polarizability increases computation by 2x…
• …but, the additional computations are perfectly suited to the GPU!
• For now, NAMD calculates atomic polarizability on CPUs only...soon we will also use GPUs

Atomic polarizability of water, highly accurately simulated through additional particles (shown in green)
Molecular Dynamics Ensembles

Constant energy, constant number of particles (NE)

Constant energy, constant volume (NVE)

Constant temperature, constant volume (NVT)

Constant temperature, constant pressure (NPT)

Choose the ensemble that best fits your system and start the simulations, but use NE to check on accuracy of the simulation.
Langevin Dynamics

for temperature control

Langevin dynamics deals with each atom separately, balancing a small friction term with Gaussian noise to control temperature:

\[ m \ddot{\vec{r}} = \vec{F}(\vec{r}) - \gamma m \dot{\vec{r}} + \vec{R}(t) \]

\[ \langle \vec{R}(t) \cdot \vec{R}(t') \rangle = 6k_B T \gamma \delta(t - t') \]
Langevin Dynamics

for pressure control

Underlying Langevin-Hoover barostat equation for all atoms:
Equations solved numerically in NAMD

\[
\frac{d^2 V(t)}{dt^2} = \frac{1}{W_{bs}} [P(t) - P_{\text{target}}] - \frac{1}{\tau_{bs}} \frac{dV(t)}{dt} + R_{bs}(t)
\]

\[
P = \rho k_B T + \frac{1}{V_d} \sum_{i<j} \langle r_{ij} \frac{dU_{\text{tot}}(r_{ij})}{dr_{ij}} \rangle \quad d = \text{dimension}
\]

\[
\langle R_{bs}(t) R_{bs}(t') \rangle = \frac{2 k_B T_{\text{target}} \delta(t - t')}{W_{bs} \tau_{bs}} \quad W_{bs} = d N_{\text{atoms}} k_B T_{\text{target}} \tau_{\text{period}}^2
\]

\[
\dot{r}_i = v_i + s r_i \quad \dot{v}_i = F_i / m_i - s v_i
\]

\[
\dot{V} = dV_s \quad \dot{s} = dV (P - P_{\text{target}}) / W - s / \tau_{bs} + R(t)
\]

\[d - \text{dimension}\]
NAMD Scalability

protein in neural membrane

ns/day

number of cores

40,000 registered users
Reduce fine-grained decomposition overhead. A primary limit on strong scaling of NAMD, preventing small simulations from running faster on larger numbers of processors, is that as work is divided into smaller chunks runtime is dominated by the overhead of control code rather than the simulation calculations. This control code will be audited to purge unnecessary layers, run-time branches, and generality. The goal is to cut the runtime attributed to overhead in half, thereby increasing performance for all simulations, most significantly for those employing GPUs or running with less than 100 atoms per processor core.

![Scaling of NAMD on a VERY Large Machine](image)

NAMD 100M-atom Benchmark on Jaguar XT5

<table>
<thead>
<tr>
<th>Number of Processor Cores</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>2048</td>
<td>1</td>
</tr>
<tr>
<td>4096</td>
<td>2</td>
</tr>
<tr>
<td>8192</td>
<td>4</td>
</tr>
<tr>
<td>16384</td>
<td>8</td>
</tr>
<tr>
<td>32768</td>
<td>16</td>
</tr>
<tr>
<td>65536</td>
<td>32</td>
</tr>
<tr>
<td>131072</td>
<td>64</td>
</tr>
</tbody>
</table>

Maintain current 100 M-atom simulation capability. To enable organelle-scale simulations on the NSF petascale system, NAMD has been extended to support simulations of 100 million atoms and beyond on hundreds of thousands of cores. The memory optimization, load balancing, and parallel I/O features that enable these simulations will be integrated to interoperate with the full range of simulation features. As described above in Innovation, improvements to the NAMD minimizer will increase support for these leading-edge simulations.

Optimize PME with node-aware placement and communication. As shown in Fig. 2.8, PME long-range electrostatics scaling limits NAMD performance and bears improvement. When running on a cluster with many cores per node, the PME reciprocal sum requires communication with many cores that are grouped on a small number of nodes. The resulting messages contain redundant data and their number has been observed to cause normally instantaneous communication calls to extend to milliseconds. Aggregating these messages on each node will reduce both message volume and count, and allow simulations to exploit increasing numbers of cores per node without requiring a proportional increase in inter-node network capacity. We will also reduce the number of PME parallel decomposition elements from one per core to one per shared-memory process and employ multiple threads within each process to perform critical-path FFTs and message processing. As a result, NAMD parallel performance will improve rather than suffer as the number of cores per node increases on new generations of clusters.

Optimize and evaluate multilevel summation method. The Center will optimize the parallel implementation of the multilevel summation method (MSM) in NAMD to enable full-electrostatics simulations without the parallel scalability limits of PME for very large simulations shown in Fig. 2.8. The performance of MSM in NAMD will be further enhanced by adapting the GPU-accelerated MSM implementation that has already been released in VMD for the evaluation of electrostatic potentials. The employed mapping of the gridded MSM calculations to GPUs is already compatible with the spatial decomposition of NAMD, and global communication requirements will be reduced by applying multiple-time-stepping techniques to the different grid levels of the MSM algorithm. As a result, both large simulations and non-periodic simulations of all sizes will run efficiently in parallel with full electrostatics.
Recent Tsubame (Tokyo) Test Run of GPU Accelerated NAMD

<table>
<thead>
<tr>
<th></th>
<th>4 nodes</th>
<th>8 nodes</th>
<th>32 nodes</th>
<th>64 nodes</th>
<th>128 nodes</th>
<th>256 nodes</th>
<th>512 nodes</th>
<th>700 nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU 12 cores</td>
<td>0.02</td>
<td>0.04</td>
<td>0.14</td>
<td>0.23</td>
<td>0.41</td>
<td>0.92</td>
<td>1.69</td>
<td>2.23</td>
</tr>
<tr>
<td>CPU 12 cores + 1 GPU</td>
<td>N/A</td>
<td>N/A</td>
<td>0.42</td>
<td>0.80</td>
<td>1.50</td>
<td>2.93</td>
<td>5.01</td>
<td>5.98</td>
</tr>
<tr>
<td>CPU 12 cores + 2 GPUs</td>
<td>N/A</td>
<td>N/A</td>
<td>0.62</td>
<td>1.17</td>
<td>2.00</td>
<td>3.68</td>
<td>6.18</td>
<td>7.38</td>
</tr>
<tr>
<td>CPU 12 cores + 3 GPUs</td>
<td>0.10</td>
<td>0.19</td>
<td>0.65</td>
<td>1.25</td>
<td>2.11</td>
<td>3.93</td>
<td>6.42</td>
<td>8.00</td>
</tr>
</tbody>
</table>

AFM image of flat chromatophore membrane (Scheuring 2009)
Figure 2.2: The size of biomolecular systems that can be studied using all-atom MD simulations has steadily increased from that of Lysozyme (40,000 atoms) in 1990 to the million-atom STMV virus capsid in 2006, and now 100 million atoms as in the spherical chromatophore model shown above. Atom counts include aqueous solvent, not shown.
Large is no problem. But …

Molecular dynamics simulation of alpha-hemolysin with about 300,000 atoms; 1 million atom simulations are routine today, 20 million atom simulations are possible.
But long is still a problem!

*biomolecular timescale and timestep limits*

Rotation of buried sidechains
Local denaturations
Allosteric transitions

Hinge bending
Rotation of surface sidechains
Elastic vibrations

Bond stretching
Molecular dynamics timestep

\[ \delta t = 1 \text{ fs} \]

<table>
<thead>
<tr>
<th>Unit</th>
<th>Steps</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>fs</td>
<td>$10^0$</td>
<td>(15 min)</td>
</tr>
<tr>
<td>ps</td>
<td>$10^3$</td>
<td>(10 days, 2 hrs)</td>
</tr>
<tr>
<td>ns</td>
<td>$10^6$</td>
<td>(30 years, 2 months)</td>
</tr>
<tr>
<td>ms</td>
<td>$10^{12}$</td>
<td></td>
</tr>
<tr>
<td>s</td>
<td>$10^{15}$</td>
<td></td>
</tr>
</tbody>
</table>

(NSF center, Shaw Res.)
Protein Folding

- Protein **misfolding** responsible for diseases:
  - Alzheimer’s
  - Parkinson’s
  - Huntington
  - Mad cow
  - Type II diabetes
  - ...

Observe folding process in unprecedented detail

* villin headpiece
  3 months on 329 CPUs
PDB Files
gives one the structure and starting position

- Simulations start with a crystal structure from the Protein Data Bank, in the standard PDB file format.
- PDB files contain standard records for species, tissue, authorship, citations, sequence, secondary structure, etc.
- We only care about the atom records…
  - atom name (N, C, CA)
  - residue name (ALA, HIS)
  - residue id (integer)
  - coordinates (x, y, z)
  - occupancy (0.0 to 1.0)
  - temp. factor (a.k.a. beta)
  - segment id (6PTI)
- No hydrogen atoms!
  (We must add them ourselves.)
Potential Energy Function of Biopolymer

- Simple, fixed algebraic form for every type of interaction.
- Variable parameters depend on types of atoms involved.

\[
U(\vec{R}) = \sum_{\text{bonds}} k_i^{\text{bond}} (r_i - r_0)^2 + \sum_{\text{angles}} k_i^{\text{angle}} (\theta_i - \theta_0)^2 + \sum_{\text{dihedrals}} k_i^{\text{dihedrals}} \left[ 1 + \cos (n_i \phi_i + \delta_i) \right] + \sum_{\text{nonbonds}} 4\varepsilon_{ij} \left( \frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{ij}}{r_{ij}} \right)^6 + \sum_{i,j \neq i} q_i q_j \epsilon_{ij}
\]
PSF Files

• Every atom in the simulation is listed.
• Provides all static atom-specific values:
  – atom name (N, C, CA)
  – atom type (NH1, C, CT1)
  – residue name (ALA, HIS)
  – residue id (integer)
  – segment id (6PTI)
  – atomic mass (in atomic mass units)
  – partial charge (in electronic charge units)

• What is not in the PSF file?
  – coordinates (dynamic data, initially read from PDB file)
  – velocities (dynamic data, initially from Boltzmann distribution)
  – force field parameters (non-specific, used for many molecules)
PSF Files

*molecular structure (bonds, angles, etc.)*

Bonds: Every pair of covalently bonded atoms is listed.

Angles: Two bonds that share a common atom form an angle. Every such set of three atoms in the molecule is listed.

Dihedrals: Two angles that share a common bond form a dihedral. Every such set of four atoms in the molecule is listed.

Impropers: Any *planar* group of four atoms forms an improper. Every such set of four atoms in the molecule is listed.
Preparing Your System for MD Solvation

Biological activity is the result of interactions between molecules and occurs at the interfaces between molecules (protein-protein, protein-DNA, protein-solvent, DNA-solvent, etc).

Why model solvation?
• many biological processes occur in aqueous solution
• solvation effects play a crucial role in determining molecular conformation, electronic properties, binding energies, etc

How to model solvation?
• explicit treatment: solvent molecules are added to the molecular system
• implicit treatment: solvent is modeled as a continuum dielectric or so-called implicit force field

mitochondrial bc1 complex
Biological activity is the result of interactions between molecules and occurs at the interfaces between molecules (protein-protein, protein-DNA, protein-solvent, DNA-solvent, etc).

Why model solvation?
- many biological processes occur in aqueous solution
- solvation effects play a crucial role in determining molecular conformation, electronic properties, binding energies, etc

How to model solvation?
- explicit treatment: solvent molecules are added to the molecular system
- implicit treatment: solvent is modeled as a continuum dielectric
Preparation Your System for MD
Solvation

Biological activity is the result of interactions between molecules and occurs at the interfaces between molecules (protein-protein, protein-DNA, protein-solvent, DNA-solvent, etc).

Why model solvation?
• many biological processes occur in aqueous solution
• solvation effects play a crucial role in determining molecular conformation, electronic properties, binding energies, etc

How to model solvation?
• explicit treatment: solvent molecules are added to the molecular system
• implicit treatment: solvent is modeled as a continuum dielectric

(Usually periodic! Avoids surface effects)
From the Mountains to the Valleys

*how to actually describe a protein*

Initial coordinates have bad contacts, causing high energies and forces (due to averaging in observation, crystal packing, or due to difference between theoretical and actual forces)

Minimization finds a nearby local minimum.

Heating and cooling or equilibration at fixed temperature permits biopolymer to escape local minima with

Initial dynamics samples thermally accessible states.
From the Mountains to the Valleys

*a molecular dynamics tale*

Longer dynamics access other intermediate states; one may apply external forces to access other available states in a more timely manner.
Cutting Corners

cutoffs, PME, rigid bonds, and multiple timesteps

• Nonbonded interactions require order $N^2$ computer time!
  – Truncating at $R_{\text{cutoff}}$ reduces this to order $N R_{\text{cutoff}}^3$
  – Particle mesh Ewald (PME) method adds long range electrostatics at order $N \log N$, only minor cost compared to cutoff calculation.

• Can we extend the timestep, and do this work fewer times?
  – Bonds to hydrogen atoms, which require a 1fs timestep, can be held at their equilibrium lengths, allowing 2fs steps.
  – Long range electrostatics forces vary slowly, and may be evaluated less often, such as on every second or third step.

• Coarse Graining
Residue-Based Coarse-Grained Model

- Lipid model: MARTINI
- Level of coarse-graining: ~4 heavy atoms per CG bead
- Interactions parameterized based on experimental data and thermodynamic properties of small molecules
- Protein model uses two CG beads per residue
- One CG bead per side chain another for backbone

Nanodisc Assembly CG MD Simulation

- 10 µs simulation
- Assembly proceeds in two steps:
  - Aggregation of proteins and lipids driven by the hydrophobic effect
  - Optimization of the protein structure driven by increasingly specific protein-protein interactions
- Formation of the generally accepted double-belt model for discoidal HDL
Validation of Simulations

*reverse coarse-graining* and *small-angle X-ray scattering*

Reverse coarse-graining:
1. Map center of mass of the group of atoms represented by a single CG bead to that bead's location
2. MD minimization, simulated annealing with restraints, and equilibration to get all-atom structure

Small-angle X-ray scattering:
Calculated from reverse coarse-grained all-atom model and compared with experimental measurements
Shape-Based Coarse-Grained (CG) model

- Fully automatic
- Number of CG beads is chosen by a user (we used ~200 atoms per CG bead)


Virus Capsid Mechanics
Atomic Force Microscope

— Hepatitis B Virus —

![Graph showing force vs. indentation for Hepatitis B Virus. The graph compares experiment (gray diamonds) and simulation (green circles). The x-axis represents indentation in Ångstroms (Å), and the y-axis represents force in piconewtons (pN). The data points show a linear relationship between force and indentation.](image-url)
Example: MD Simulations of the K\(^+\) Channel Protein

Ion channels are membrane-spanning proteins that form a pathway for the flux of inorganic ions across cell membranes.

Potassium channels are a particularly interesting class of ion channels, managing to distinguish with impressive fidelity between K\(^+\) and Na\(^+\) ions while maintaining a very high throughput of K\(^+\) ions when gated.
Setting up the system (1)

• retrieve the PDB (coordinates) file from the Protein Data Bank

• add hydrogen atoms using PSFGEN

• use psf and parameter files to set up the structure; needs better than available in Charmm to describe well the ions

• minimize the protein structure using NAMD2
Simulate the protein in its natural environment: solvated lipid bilayer
Setting up the system (3)

Inserting the protein in the lipid bilayer

Automatic insertion into the lipid bilayer leads to big gaps between the protein and the membrane — long equilibration time required to fill the gaps. Solution: manually adjust the position of lipids around the protein. Employ constant (lateral and normal) pressure control.
The system

Kcsa channel protein (in blue) embedded in a (3:1) POPE/POPG lipid bilayer. Water molecules inside the channel are shown in vdw representation.
Simulating the system: Free MD

Summary of simulations:
• protein/membrane system contains 38,112 atoms, including 5117 water molecules, 100 POPE and 34 POPG lipids, plus K$^+$ counterions
• CHARMM26 forcefield
• periodic boundary conditions, PME electrostatics
• 1 ns equilibration at 310K, NpT
• 2 ns dynamics, NpT

Program: NAMD2

Platform: Cray T3E (Pittsburgh Supercomputer Center) or local computer cluster; choose ~1000 atoms per processor.
RMS deviations for the KcsA protein and its selectivity filter indicate that the protein is stable during the simulation with the selectivity filter the most stable part of the system.

Temperature factors for individual residues in the four monomers of the KcsA channel protein indicate that the most flexible parts of the protein are the N and C terminal ends, residues 52-60 and residues 84-90. Residues 74-80 in the selectivity filter have low temperature factors and are very stable during the simulation.
Simulation of Ion Conduction (here for Kv1.2)

Theoretical and Computational Biophysics Group
Beckman Institute
University of Illinois at Urbana-Champaign
Theoretical and Computational Biophysics Group Developers

- focus on systems biology
- focus on quantum biology
- theoretical biophysics
- computational biophysics
- develops renewable energy
- guides bionanotechnology

Funding: NIH, NSF

L. Kale
J. Stone
J. Phillips