MD of Soft Matter

Mark J. Stevens
msteve@sandia.gov

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.
Molecular Dynamics Simulations

• What has been developed?
• Where are we at?
• Where are we going?
MD & LAMMPS
Molecular Dynamics Basics

• Building the starting state
  – often a time consuming effort for the human
  – for proteins, need a crystal structure (? folding from sequence)

• Integration algorithm
  – Verlet, predictor-corrector
  – multitime step (RESPA)

• Force calculation
  – typically 90% of CPU time
  – Coulomb is most expensive
    ▪ long range requires Ewald
    ▪ particle-mesh methods are fast & parallelizable (N log N)
      – uses global FFTs
  – van der Waals (Lennard-Jones)
    ▪ expensive and scales as cutoff $^3$
  – bonds and other intramolecular terms
    ▪ fast to calculate, but stiffness determines time step

• Ensemble (thermostat, barostat)
  – fundamental issues of statistical mechanics
  – this is where calculations go wrong

\[ F = ma \]
LAMMPS 1995-

LAMMPS

• Massively parallel MD code
  – as system size scales with number of processors, CPU time should remain constant
• Main programmer: Steve Plimpton, Sandia

History:
• parallel Lennard-Jones codes
• CRADA ~1995-
  – spatial decomposition parallelization
  – Nose-Hoover ensemble equations of motion
  – RESPA, multiple time step algorithm
  – Particle mesh Ewald (PPPM), long range Coulomb
  – class 2 atomistic force-field
  – Fortran 77
• Other comparable MD codes
  – 1995: none
  – NAMD, AMBER, CHARMM, GROMACS, DL_POLY
Parallelism via Spatial-Decomposition

- Physical domain divided into 3D boxes, one per processor
- Each proc computes forces on atoms in its box
  - using info from nearby procs
- Each proc owns atoms in its box
  - NO global arrays
- **Communication** occurs every time step
  - update forces between atoms in neighboring boxes
  - via nearest-neighbor 6-way stencil

- Optimal scaling for MD: \( \frac{N}{P} \)
  - so long as load-balanced
- Computation scales as \( \frac{N}{P} \)
- Communication scales
  - sub-linear as \( (\frac{N}{P})^{2/3} \)
  - (for large problems)
- Memory scales as \( \frac{N}{P} \)

- **Load Balance**: cost of computing forces vs. time to communicate updated positions
LAMMPS ~1997

52 files

communicate.f integrate.f ppm.f
diagnostic_PE.f integrate_pe3.f ppm2.f
ewald.f integrate_respa.f ppm2_coeff.f
ewald_coeff.f lammps.f ppm2_remap.f
finish.f lapack.f ppm_coeff.f
fix.f min_algs.f random.f
force.f min_support.f read_data.f
force_bond.f misc.f read_restart.f
force_class2.f neighbor.f setup.f
force_many.f output.f setup_special.f
force_respa.f parlib_c90.f start.f
initialize.f parlib_t3d.f string.f
input.f parlib_t3e.f thermo.f
input_zran.f parlib_unix.f velocity.f
LAMMPPS today

Multiple force-field types and Hybrid potentials

224 files

Mark Stevens
msteve@sandia.gov

Mark Stevens
msteve@sandia.gov
Nonbond Interactions

Lennard-Jones (LJ) potential

\[ r_c \text{ is cutoff distance} \]
\[ \varepsilon_{\alpha\beta}, \sigma_{\alpha\beta} \text{ for pair types } \alpha\beta \]

Hybrid examples:

• LJ + Embedded Atom Method (EAM)
• integrated LJs

available code is very important asset
human costs are high
Parallel Performance

- Fixed-size (32K atoms) and scaled-size (32K atoms/proc) parallel efficiencies
- Metallic solid with EAM potential

- Billions of atoms on 64K procs of Blue Gene or Red Storm
- Opteron processor speed: 5.7E-6 sec/atom/step (0.5x for LJ, 12x for protein)
Other MD codes

- **NAMD**
  - Klaus Schulten (U. Illinois)
  - parallel
  - CHARMM FF
  - NIH supported, open source

- **AMBER**
  - >UCSF + many others
  - now parallel
  - associated force-field
  - support ?
  - license

- **CHARMM**
  - Martin Karplus (Harvard) + community
  - weakly parallel
  - associated force-field
  - support ?
  - license

- **GROMACS**
  - Erik Lindahl (Stockholm)
  - David van der Spoel (Uppsala)
  - Berk Hess (Mainz)
  - parallel & fast
  - ~GROMOS force-field
  - open source

- **DL_POLY**
  - W. Smith & others (Britain)
  - parallel
  - license
Hardware & SPEED
Atomistic Simulation of Protein Dynamics

- **Rhodopsin**
  - a membrane protein that absorbs light
  - starts signaling cascade that results in our vision
  - a G-protein coupled receptor (GPCR)
- **GPCR**
  - ligand binds and activates the G-protein on the cytosol and starts a signal
  - big drug target
- **Simulations**
  - 2000 Bovine Rhodopsin crystallized in dark state
  - 2003 MD simulation (and others)
    - 40 ns simulation (only a few competitors)
  - 2007 simulation of photoisomerization (and IBM group)
    - 150 ns simulation
    - changes in transmembrane helices
    - movement of water
    - side chain dynamics
  - just last month crystal structure of light state
    - understand the physical mechanism of the transition between states
- 10 ns standard for major simulators; 100 ns possible
Protein Simulations: Future

• We have reached a new era in protein simulations
  – starting to examine the sequential dynamics of proteins

• BlueGene
  – on the biggest computer, serious attacks on such problems are happening
    ▪ µs simulation
    ▪ many 100 ns simulations
  – statistics of single trajectories (not identical)
    ▪ Is there a way to efficiently obtain good statistics?
    ▪ statistics of a transition path (see also free energy calculation)
  – BlueGene computer
    ▪ optimized code for special processors
    ▪ many many processors ⇒ algorithm must be (and is) fast for few atoms/processor
    ▪ i.e. a lot of work was performed to make it happen & not all code is transferable
  – Alan Grossfield, U. Rochester; Michael Pitman, IBM

• Starting without a measured crystal structure
  – Building a protein is equivalent to folding it.
  – How do you build a GPCR in general? based on rhodopsin?

• Can we simplify the dynamics and make the calculation faster?
  – Do we need to treat the transmembrane helices in full detail all the time?
  – Cost ~ number of interactions to be calculated
Computers of the Future

- GPUs (and other coprocessors)
- Nvidia: CUDA language
- Desktop → 60 processor computer
- resurrection of the workstation?
  - ~$20,000 = ~1000 CPU equivalents?
- Requires code rewrites
  - load imbalanced
  - getting data in/out of GPU efficiently
  - NAMD conversion only get 4x speedup
- nodes on parallel computers will (do!) have GPUs

What's New?

LAMMPS

• C++ code
• many new force-fields
  – typically not funded
  – essential
• ability to combine force-fields
  – Ex: organic-metal system
  – Ex: model systems that use multiple interaction types
• rigid body dynamics
• aspherical potentials
• input script language

Wish list

• combining with continuum mechanics
Coarse-Graining
Coarse-grained models

- polymer
  - standard CG motivation
  - polyelectrolyte examples
    - DNA
      - elliptical interactions for sugar bases
- lipids
  - follow polymer (minimal) models
  - chemically defined models
    - Voth, Klein, Marrink
  - atomistic FFs need work (e.g. CHARMM vs. GROMOS)
- & proteins
  - rigid body
  - same FF issues as lipids

Biomembranes

• Challenges
  – Fusion
  – Effect of lipids on protein activity (e.g. rhodopsin)
  – Organization of multicomponent membranes
    ▪ Domains (rafts) organize proteins
  – transport of particles thru membrane
    ▪ toxins, nanoparticles

• Lipid membranes as a material
  – liquid surface
  – self-assembly
Simulating Biomembranes

We want to simulate lipid membranes & must treat liquid dynamics.

Lipid diffusion is ‘slow’
- diffusion constant $\sim 10^{-8} \text{ cm}^2/\text{s} = 10^{-3} \text{ nm}^2/\text{ns}$
- lipid exchange time $\sim 100 \text{ ns}$
- too slow for atomistic simulations

⇒ need to use coarse-grained models
Coarse-grained Models

• Follow successful coarse-grained models in polymer physics
  – bead-spring model
  – 2 types
    ▪ hydrophobic & hydrophilic

• Can treat essential physical features that drive key phenomena
  – connectivity
  – hydrophobic/hydrophilic interactions
    ▪ self-assembly
  – membrane fluidity

• This is sufficient for more complex phenomena
  – microdomains
  – fusion
  – membrane-protein interactions
Simulations of Bilayer Membranes

Membrane self-assembly? Yes!

1x10^6 time steps
Liquid Membrane

Verifying fluidity of bilayer
• Lipids diffuse across simulation box
• Lipid diffusion not possible presently in atomistic simulations
• Matching diffusion times yields map: LJ time unit $\tau \rightarrow 0.2$ ns.
• $5000 \, \tau = 1.0 \, \mu s$.
• Times in the $\mu s$ to ms range achievable

![Graph showing liquid and gel phases with time in LJ time units and r^2 (\sigma^2) on the y-axis.](image)
Fusion Simulation Setup

- Create single liposome by placing lipids on inner & outer spheres: \( D = 30 \) \( \sigma = 15 \) nm, \( N_T = 4 \).
- Apply constant force to bring liposome together
- Images are slices

\[
\begin{align*}
f & \quad f \\
2158 \text{ lipids/liposome} \\
333680 \text{ total beads}
\end{align*}
\]
Fusion Dynamics
Coarse-grained Models

• Model and FF development of CG lipid
  – Siewaart-Jan Marrink (Netherlands)
    ▪ MARTINI FF (in GROMACS)
    ▪ ~5 LJ types, partial charges
    ▪ different lipid head group
    ▪ cholesterol
  – Greg Voth
    ▪ force matching method (atomistics to CG)
    ▪ atomistic to CG connection
    ▪ well defined mathematics
    ▪ also done work to coarse-grained at higher level (field theory)
  – Michael Klein
    ▪ revised version of early work on surfactants
    ▪ new results on lipids coming
  – Markus Deserno
    ▪ 3-bead lipid model and no solvent
    ▪ for larger scale systems

• atom to CG is not 1:1
  – multiple relevant versions possible
  – best choice may depend on problem

• Need multiple levels of coarse-graining
CG Lipid Simulations

Next Stage

• Different lipid types
  – more complex models

• Free energy calculations
  – very very expensive
  – WHAM
    ▪ constrain system (ex. two vesicles at fixed separation in fusion simulation)
    ▪ do a full simulation at each constraint
    ▪ collect statistics
    ▪ calculate free energy difference from 'WHAM' equations
  – energy barriers
    ▪ slow dynamics → expensive calculations
    ▪ various methods proposed
    ▪ multiple dimensions are a challenge
    – ? who is funded

• Including proteins in membrane simulations
  (next page)
Including Proteins

• Many problems of interest involving lipids and membrane proteins
  – biological problems vs. model systems
  – interactions between proteins in membrane
  – fusion peptides
  – antimicrobial peptides
  – domain organization of proteins (rafts)

• CG protein models
  – Thirumalai (90s)
    ▪ alpha helix & beta sheet
    ▪ recently complex orientational FF
  – Marrink
  – many minimal models developed for protein folding
  – folding vs dynamics
  – rigid body dynamics
    ▪ treat protein as cylinder
    ▪ treat helices as rigid
Coarse-grained Models: Future

• hydrogen bonding
  – directionality important?
  – how incorporate small scale into CG model

• water
  – hydrogen bonding liquid
  – dielectric screening
  – 90% of particles
  – implicit water especially far away

• nonspherical potentials
  – to treat rings (phenyl, sugars)
  – to treat other rigid or semirigid components
  – need efficient algorithms

• charges
  – explicit (done, but expensive)
  – Debye-Hückel or other effective interaction

• efficiency: complexity or adding features tends to add costs
Next Level of Coarse-grained Models

Schematics have essential features
How do you simulate all this?
• better models
  – cholesterol
  – lipids
• need a model
  – glycolipids
• protein shapes
  – know the shape?
  – model shape better than a collection of points
  – interaction between a shape & particle
    • reduce number of interactions
  – dynamic vs static 'mesh'
  – varying degrees of flexibility
• time scales
  – components
  – multiple levels of coarse-graining
  – fastest frequency → short time step
• can do length scale?
Nanoparticles
Nanoparticles

• What can we do with nanoparticles?
  – most nanoparticles must be coated in order to reside in a system

• In simulations what needs to be done?
  – structure of coated nanoparticles
  – interaction between coated nanoparticles
    ▪ influence of solvent
  – what structures of sets of nanoparticles occur
  – how do you make a desired structure?
  – how do you put the nanoparticles where you want them
    ▪ polymer nanocomposites
    ▪ nanoparticle crystals
SAMs & AFM tips

• Have treated self-assembled monolayers (SAMs)
• primarily on SiO$_2$
• treated a variety of terminations
  – CF$_3$, OH, COOH, ethylene glycol, nylon
  – issue: treating ionization
    ▪ free H$^+$ is problematic
    ▪ dynamics of dissociation/association
    ▪ FF have defined connectivity
• Now doing explicit SiO$_2$ tips

• Nanoparticles
  – coatings on nonplanar surfaces
  – will encompass a big leap in length and time scales
    ▪ want to do more than 2 particles
    ▪ particle dynamics on slower scale than solvent
    ▪ how reduce number of solvent molecules

Mark Stevens
msteve@sandia.gov
PEO-coated silica inter-particle force

- approx. 360,000 water atoms
- approx. 400,000 total atoms
- 90% solvent

- 128 processors for 16 days (slowest case)
- now ~ 2 million atoms to treat hydrodynamics
Force on PEO coated Silica in water

- $v = 1 \text{ m/s}$
- $v = 5 \text{ m/s}$
- $v = 50 \text{ m/s}$

Mark Stevens
msteve@sandia.gov
Interaction Potential

- \( r_c = 10\AA \)
- Sum of LJ ?LJ
- Hybrid potential model
- Solvent represented by standard Lennard-Jones (LJ) particles
- NP-NP interaction treated as interactions between integrated LJ particles\(^1\)
- NP-solvent interaction treats NP as an integrated particle

\[
\begin{align*}
U_{\text{nano-solvent}} &= \frac{2a_1^2 \sigma_{ij}^2 A_{ns}}{9(a_2^3 - r_{ij}^3)^3} \left[ 1 - \frac{(5a_2^6 + 45a_2^3 r_{ij}^2 + 63 a_2^3 r_{ij}^1 + 15 r_{ij}^2) \sigma^*}{15(a_2 - r_{ij})^6 (a_2 + r_{ij})^6} \right] \\

U_{\text{nano-nano}} &= \frac{-A_{nn}}{6} \left[ \frac{2a_1 a_2}{r_{ij}^2 - (a_1 + a_2)^2} + \frac{2a_1 a_2}{r_{ij}^2 - (a_1 - a_2)^2} + \ln \left( \frac{r_{ij}^2 - (a_1 + a_2)^2}{r_{ij}^2 - (a_1 - a_2)^2} \right) \right] \\
U_{\text{R}} &= \frac{A_{nn} \sigma^*}{37800 r_{ij}^2} \left[ \frac{r_{ij}^2 - 7r_{ij} (a_1 + a_2) + 6 (a_1^2 + 7a_1 a_2 + a_2^2)}{(r_{ij} - a_1 - a_2)^7} \\
&+ \frac{r_{ij}^2 + 7r_{ij} (a_1 + a_2) + 6 (a_1^2 + 7a_1 a_2 + a_2^2)}{(r_{ij} + a_1 + a_2)^7} \\
&- \frac{r_{ij}^2 + 7r_{ij} (a_1 - a_2) + 6 (a_1^2 - 7a_1 a_2 + a_2^2)}{(r_{ij} + a_1 - a_2)^7} \\
&- \frac{r_{ij}^2 - 7r_{ij} (a_1 - a_2) + 6 (a_1^2 - 7a_1 a_2 + a_2^2)}{(r_{ij} - a_1 + a_2)^7} \right] \\
U &= U_A + U_R,
\end{align*}
\]

Nanoparticles in Solvent

• Simulation details:
  – 10-2000 nanoparticles
  – 0.5-2 million LJ solvent particles
  – $T = \frac{e}{k_B}$, $P = 0.1e/s^3$

• Simulations only feasible due to significant improvements in LAMMPS
  – Multi-region neighbor lists
  – Improved communications

Pieter J. in ’t Veld, Matt K. Petersen, Gary S. Grest, Steve Plimpton
Coarse-Graining of Background Fluid

• Issue
  – too many solvent particles
  – solvent/colloid ratio may be 100:1 or 1000:1
  – interest is colloidal dynamics

• SRD = stochastic rotation dynamics (particle)
  – Hecht et al, PRE, 72 (2005)
  – intermediate Peclet numbers of around 1
  – Pe = ratio of advection to diffusion

• Basic idea:
  – solvent moves by random rotation + streaming flow
  – solvent particles do not interact with each other

• Implementation issues
  – Cheap because no solvent-solvent interactions (LJ) to compute
  – How to add lots of (non-interacting) particles and not slow down
  – How to detect SRD/colloid collisions efficiently?
  – How to thermostat?
Nanoparticles

• We are now considering the 1-2 type nanoparticle systems.

• Proteins are nanoparticles.
• There are interesting materials made of proteins.
• The surfaces of proteins are far more complex
  – many interaction sites
  – many shapes
• The possibilities are far beyond what we are imagining.
Rigid Body Dynamics
Patchy Particles

- Use $\text{C}_{80}$ to define sites on a sphere
- 4 matching sites on nanoparticle
  - attractive: blue:red and cyan:green
  - acid:base binding
  - will make sheets?
- Simulation
  - 500 rigid bodies
- Large particle may reside on multiple processors
  - need efficient parallel methods
  - fast communications

largest cluster
Patchy Particles

- **Modify model**
  - provide orientation within patches
  - yield alignment of bonded particles
  - which yields sheet fragments
  - slow dynamics of fragments forming single sheet

- **Monte Carlo may be more efficient at low densities**
  - Cluster moves
  - No general Monte Carlo codes
  - typically not parallel

- **Mimic MC move in MD?**
  - redefine cluster as rigid unit
    - reduces intra-body calculations
    - allows larger time step ~ rotation of large cluster
    - parallelization issues

- **People are just beginning to study such systems**
  - can use with coarse-grained systems to study proteins in lipids
Atomistic/Continuum Coupling

MD + finite elements for stress/strain response
- boundary conditions for MD
- transfer between MD forces and FE stress
- general issues: applies to solids and liquids

- Rob Hoy and Mark Robbins (JHU)
  - solids, fracture
- Greg Wagner, Reese Jones, Jeremy Templeton (Sandia)
- Jeremy Lechman, Randy Schunk (Sandia)
  - liquids
Acknowledgements

Steve Plimpton
Gary Grest
Jeremy Lechman
Pieter in 't Veld
Matt Lane
Matt Petersen

LAMMPS Funding:
• DoE Office of Science: OASCR/OBER offices, GTL program
• Sandia LDRD program
• original CRADA (Cray, Dupont, Bristol-Myers Squibb)

Stevens Funding:
• Sandia LDRD program
• ASCI program
• DoE Office of Science: BES office