MD of Soft Matter

Mark J. Stevens

Sandia National Laboratories

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 Administrationunder contract DE-AC04-94AL85000.

Molecular Dynamics Simulations

- What has been developed?
- Where are we at?
- Where are we going?

Mark Stevens
msteve@sandia.gov

MD & LAMMPS

Mark Stevens
msteve@sandia.gov

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

$$F = ma$$

Mark Stevens
msteve@sandia.gov

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

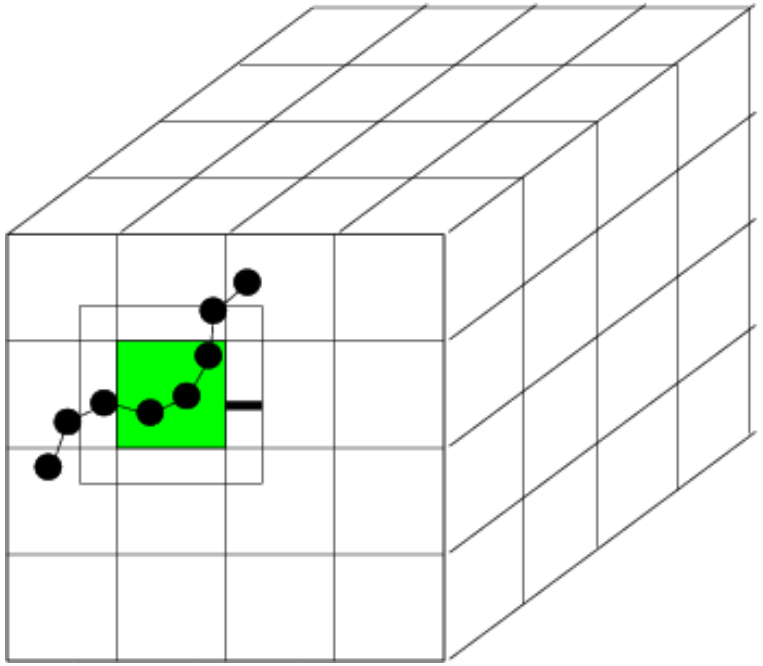
History:

- 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

Mark Stevens
msteve@sandia.gov

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: N/P
  - –so long as load-balanced
- Computation scales as N/P
- Communication scales
  - –sub-linear as $(N/P)^{2/3}$
  - –(for large problems)
- Memory scales as N/P

- Load Balance: cost of computing forces vs. time to communicate updated positions

Mark Stevens
msteve@sandia.gov

LAMMPS ~1997

52 files

communicate.f	integrate.f	pppm.f
diagnostic_PE.f	integrate_pe3.f	pppm2.f
ewald.f	integrate_respa.f	pppm2_coeff.f
ewald_coeff.f	lammps.f	pppm2_remap.f
finish.f	lapack.f	pppm_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

Mark Stevens
msteve@sandia.gov

LAMMPS today

224 files

Multiple force-field types and Hybrid potentials

angle.cpp angle_charmm.cpp angle_cosine.cpp angle_cosine_delta.cpp angle_cosine_squared.cpp angle_harmonic.cpp angle_hybrid.cpp atom.cpp
atom_vec.cpp atom_vec_angle.cpp atom_vec_atomic.cpp atom_vec_bond.cpp atom_vec_charge.cpp atom_vec_dipole.cpp atom_vec_full.cpp
atom_vec_hybrid.cpp atom_vec_molecular.cpp bond.cpp bond_fene.cpp bond_fene_expand.cpp bond_harmonic.cpp bond_hybrid.cpp bond_morse.cpp
bond_nonlinear.cpp bond_quartic.cpp change_box.cpp comm.cpp compute.cpp compute_centro_atom.cpp compute_coord_atom.cpp
compute_displace_atom.cpp compute_erotate_sphere.cpp compute_group_group.cpp compute_ke.cpp compute_ke_atom.cpp compute_pe.cpp
compute_pe_atom.cpp compute_pressure.cpp compute_reduce.cpp compute_stress_atom.cpp compute_temp.cpp compute_temp_com.cpp
compute_temp_deform.cpp compute_temp_partial.cpp compute_temp_ramp.cpp compute_temp_region.cpp compute_temp_sphere.cpp create_atoms.cpp
create_box.cpp delete_atoms.cpp delete_bonds.cpp dihedral.cpp dihedral_charmm.cpp dihedral_harmonic.cpp dihedral_helix.cpp dihedral_hybrid.cpp
dihedral_multi_harmonic.cpp dihedral_opls.cpp displace_atoms.cpp displace_box.cpp domain.cpp dump.cpp dump_atom.cpp dump_bond.cpp
dump_custom.cpp dump_dcd.cpp dump_xyz.cpp error.cpp ewald.cpp fft3d.cpp fft3d_wrap.cpp finish.cpp fix.cpp fix_add_force.cpp fix_ave_atom.cpp
fix_ave_force.cpp fix_ave_spatial.cpp fix_ave_time.cpp fix_com.cpp fix_coord_original.cpp fix_deform.cpp fix_deposit.cpp fix_drag.cpp fix_dt_reset.cpp
fix_efield.cpp fix_enforce2d.cpp fix_gravity.cpp fix_gyration.cpp fix_heat.cpp fix_indent.cpp fix_langevin.cpp fix_line_force.cpp fix_minimize.cpp
fix_momentum.cpp fix_msd.cpp fix_nph.cpp fix_npt.cpp fix_npt_sphere.cpp fix_nve.cpp fix_nve_limit.cpp fix_nve_noforce.cpp fix_nve_sphere.cpp
fix_nvt.cpp fix_nvt_sllod.cpp fix_nvt_sphere.cpp fix_orient_fcc.cpp fix_plane_force.cpp fix_press_berendsen.cpp fix_print.cpp fix_rdf.cpp fix_recenter.cpp
fix_respa.cpp fix_rigid.cpp fix_set_force.cpp fix_shake.cpp fix_shear_history.cpp fix_spring.cpp fix_spring_rg.cpp fix_spring_self.cpp
fix_temp_berendsen.cpp fix_temp_rescale.cpp fix_tmd.cpp fix_viscosity.cpp fix_viscous.cpp fix_wall_lj126.cpp fix_wall_lj93.cpp fix_wall_reflect.cpp
fix_wiggle.cpp force.cpp group.cpp improper.cpp improper_cvff.cpp improper_harmonic.cpp improper_hybrid.cpp input.cpp integrate.cpp kspace.cpp
lammps.cpp lattice.cpp library.cpp main.cpp memory.cpp min.cpp min_cg.cpp min_cg_fr.cpp min_sd.cpp minimize.cpp modify.cpp neigh_bond.cpp
neigh_derive.cpp neigh_full.cpp neigh_gran.cpp neigh_half_bin.cpp neigh_half_multi.cpp neigh_half_nsq.cpp neigh_list.cpp neigh_request.cpp
neigh_respa.cpp neigh_stencil.cpp neighbor.cpp output.cpp pack.cpp pair.cpp pair_airebo.cpp pair_buck.cpp pair_buck_coul_cut.cpp pair_buck_coul_long.cpp
pair_coul_cut.cpp pair_coul_debye.cpp pair_coul_long.cpp pair_dipole_cut.cpp pair_eam.cpp pair_eam_alloy.cpp pair_eam_fs.cpp pair_hybrid.cpp
pair_hybrid_overlay.cpp pair_lj_charmm_coul_charmm.cpp pair_lj_charmm_coul_charmm_implicit.cpp pair_lj_charmm_coul_long.cpp pair_lj_cut.cpp
pair_lj_cut_coul_cut.cpp pair_lj_cut_coul_debye.cpp pair_lj_cut_coul_long.cpp pair_lj_cut_coul_long_tip4p.cpp pair_lj_expand.cpp pair_lj_gromacs.cpp
pair_lj_gromacs_coul_gromacs.cpp pair_lj_smooth.cpp pair_morse.cpp pair_soft.cpp pair_sw.cpp pair_table.cpp pair_tersoff.cpp pair_yukawa.cpp pppm.cpp
pppm_tip4p.cpp random_mars.cpp random_park.cpp read_data.cpp read_restart.cpp region.cpp region_block.cpp region_cylinder.cpp region_intersect.cpp
region_prism.cpp region_sphere.cpp region_union.cpp remap.cpp remap_wrap.cpp replicate.cpp respa.cpp run.cpp set.cpp shell.cpp special.cpp temper.cpp
thermo.cpp timer.cpp universe.cpp update.cpp variable.cpp velocity.cpp verlet.cpp write_restart.cpp

code capability « algorithmic advance

Mark Stevens
msteve@sandia.gov

Nonbond Interactions

Lennard-Jones (LJ) potential

r_c is cutoff distance

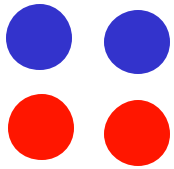
$\varepsilon_{\alpha\beta}$ $\sigma_{\alpha\beta}$ for pair types $\alpha\beta$

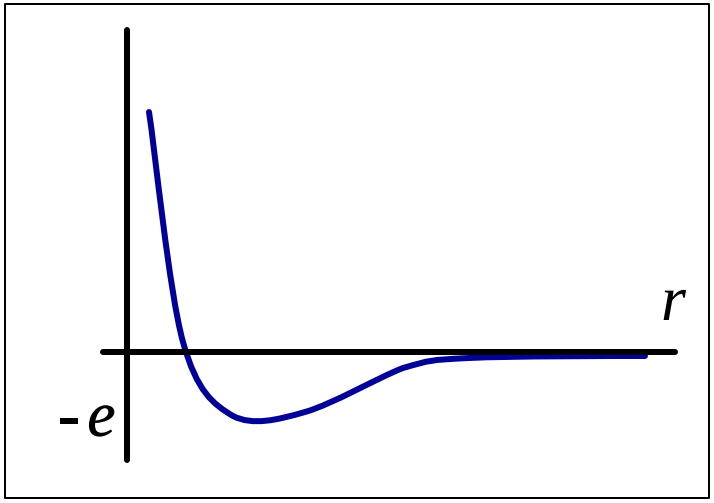
$$4e_{ab}\left[\left(\frac{s_{ab}}{r_{ij}}\right)^{12}-\left(\frac{s_{ab}}{r_{ij}}\right)^{6}\right]$$

Hybrid examples:

•LJ + Embedded Atom Method (EAM)

•integrated LJs





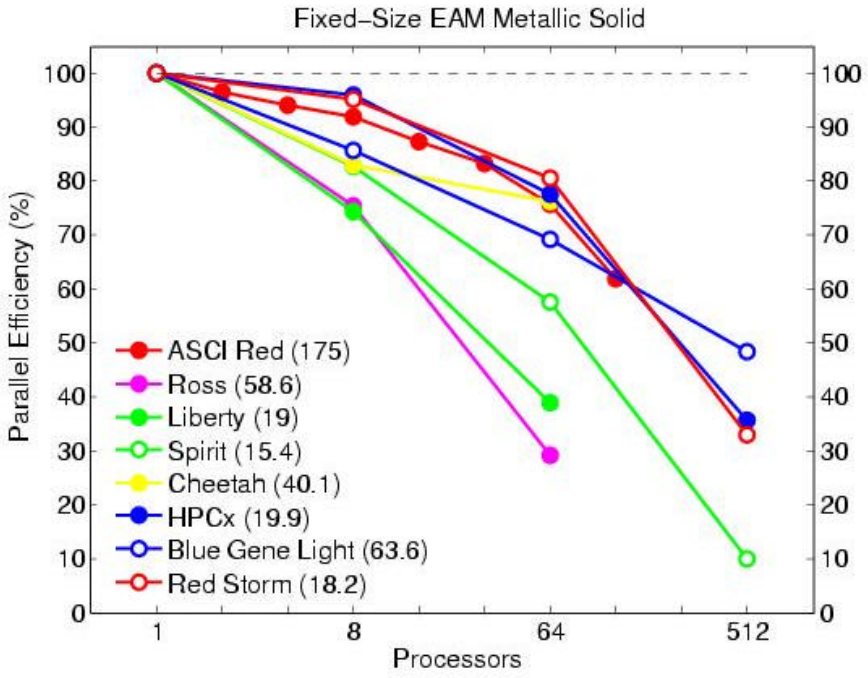
available code is very important asset

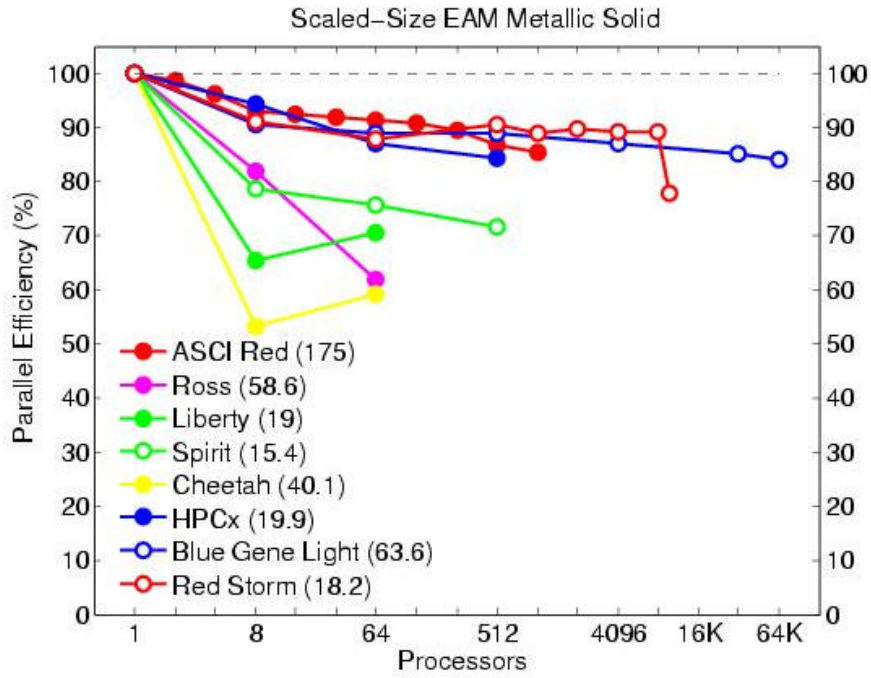
human costs are high

Mark Stevens
msteve@sandia.gov

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)

Mark Stevens
msteve@sandia.gov

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

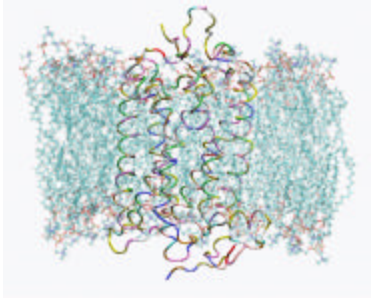
Mark Stevens
msteve@sandia.gov

Hardware & *SPEED*

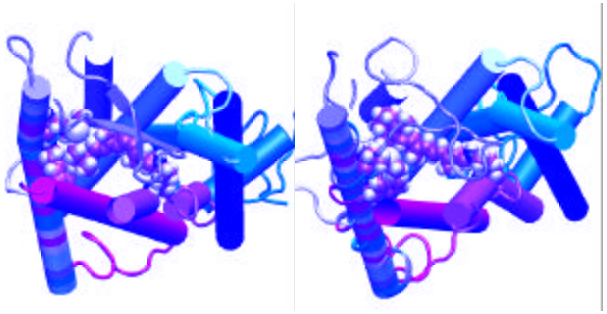
Mark Stevens
msteve@sandia.gov

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 gr
    - 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

Mark J. Stevens
msteve@sandia.gov

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

Mark Stevens
msteve@sandia.gov

Computers of the Future

- GPUs (and other coprocessors)
- Nvidia: CUDA language
- Desktop $\rightarrow$ 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

QuickTime™ and a
TIFF (Uncompressed) decompressor
are needed to see this picture.

QuickTime™ and a
TIFF (Uncompressed) decompressor
are needed to see this picture.

J. Anderson, C. Lorenz, A. Travesset, 2008

Mark Stevens
msteve@sandia.gov

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

Mark Stevens
msteve@sandia.gov

Coarse-Graining

Mark Stevens
msteve@sandia.gov

Coarse-grained models

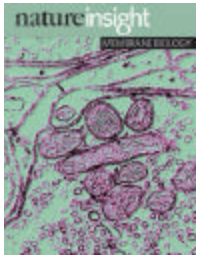
- polymer
  - standard CG motivation
  - polyelectrolyte examples
    - DNA
      - elliptical interactions for sugar bases
- lipids
  - follow polymer (minimal) models
  - chemically defined models
    - Voth, Klein, Marrink
  - atomistic FFs need work (e.g. CHARMM vs. GROMOS)
- & proteins
  - rigid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

see *Coarse-Graining of Condensed Phase and Biomolecular Systems*, ed. G. Voth, CRC Press (2009)

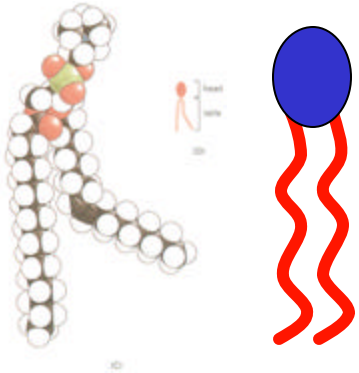
Mark Stevens
msteve@sandia.gov

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



Dec. 2005



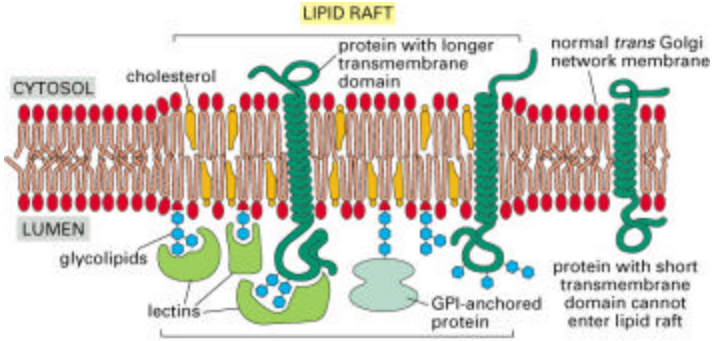


Figure 13-63. Molecular Biology of the Cell, 4th Edition.

Mark J. Stevens
msteve@sandia.gov

Simulating Biomembranes

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

Lipid diffusion is ‘slow’

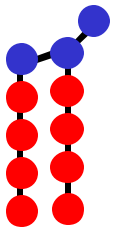
- diffusion constant ~ 10^{-8} cm^2/s = 10^{-3} nm^2/ns
- lipid exchange time ~ 100 ns
- too slow for atomistic simulations

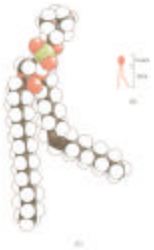
⇒ need to use coarse-grained models

Mark Stevens
msteve@sandia.gov

Coarse-grained Models

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



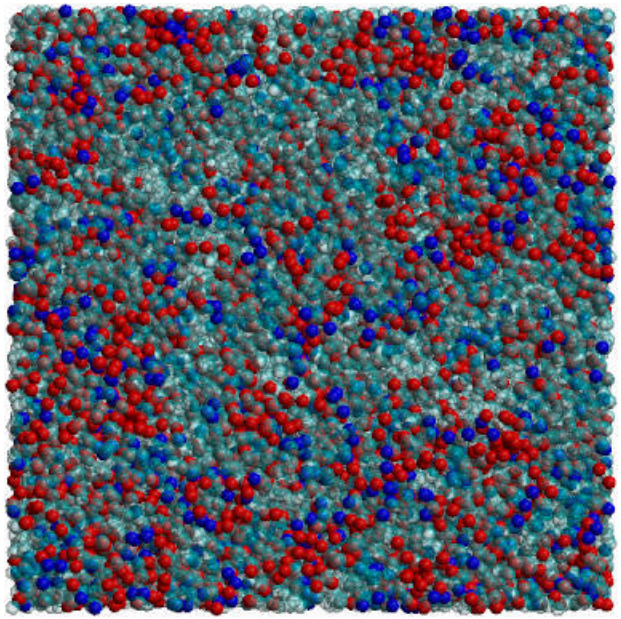


- Can treat essential physical features that drive key phenomena
  - connectivity
  - hydrophobic/hydrophilic interactions
    - self-assembly
  - membrane fluidity

- This is sufficient for more complex phenomena
  - microdomains
  - fusion
  - membrane-protein interactions

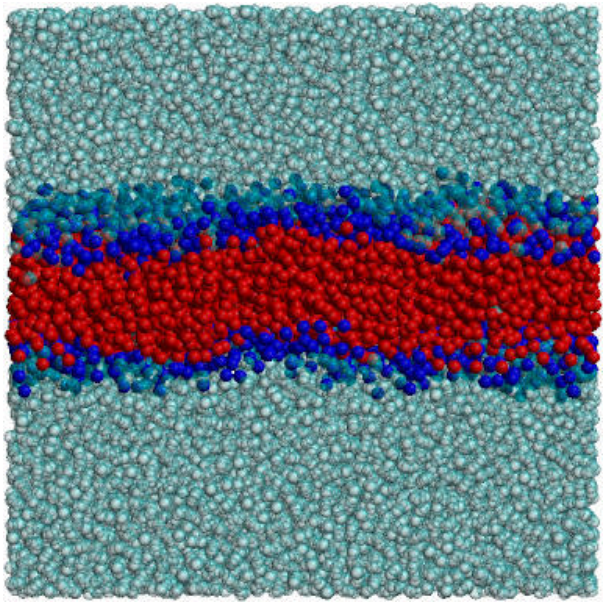
Mark Stevens
msteve@sandia.gov

Simulations of Bilayer Membranes



Membrane self-assembly?

MD



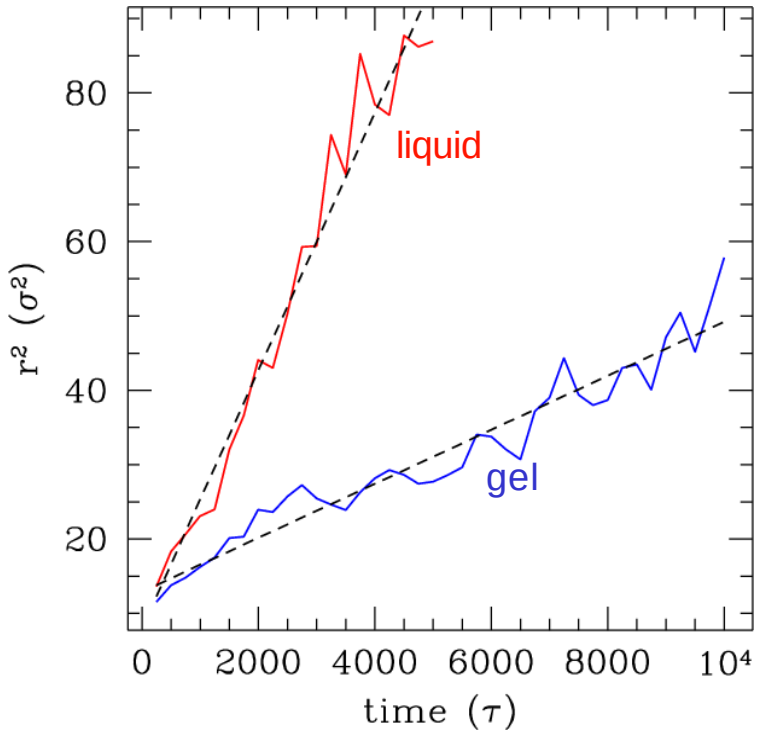
Yes!
$1x10^6$ time steps

Mark Stevens
msteve@sandia.gov

Liquid Membrane

Verifying fluidity of bilayer

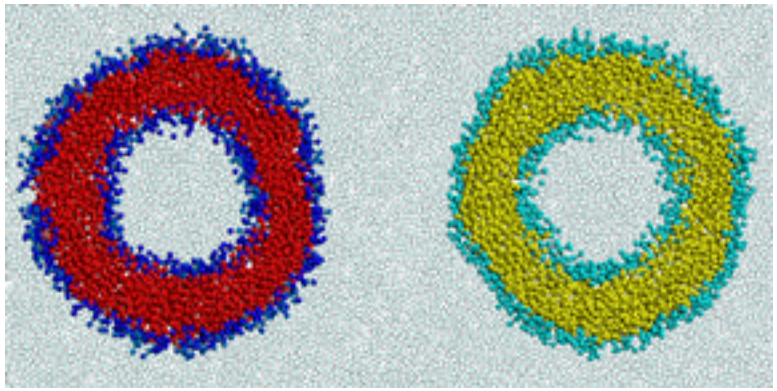
- Lipids diffuse across simulation box
- Lipid diffusion not possible presently in atomistic simulations
- Matching diffusion times yields map: LJ time unit $\tau \rightarrow 0.2$ ns.
- 5000 $\tau = 1.0$ μs.
- times in the μs to ms range achievable



Mark Stevens
msteve@sandia.gov

Fusion Simulation Setup

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



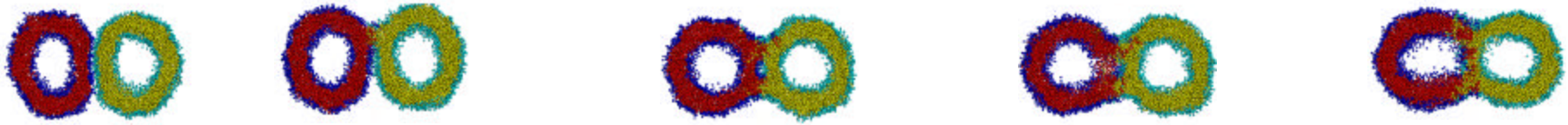
f f

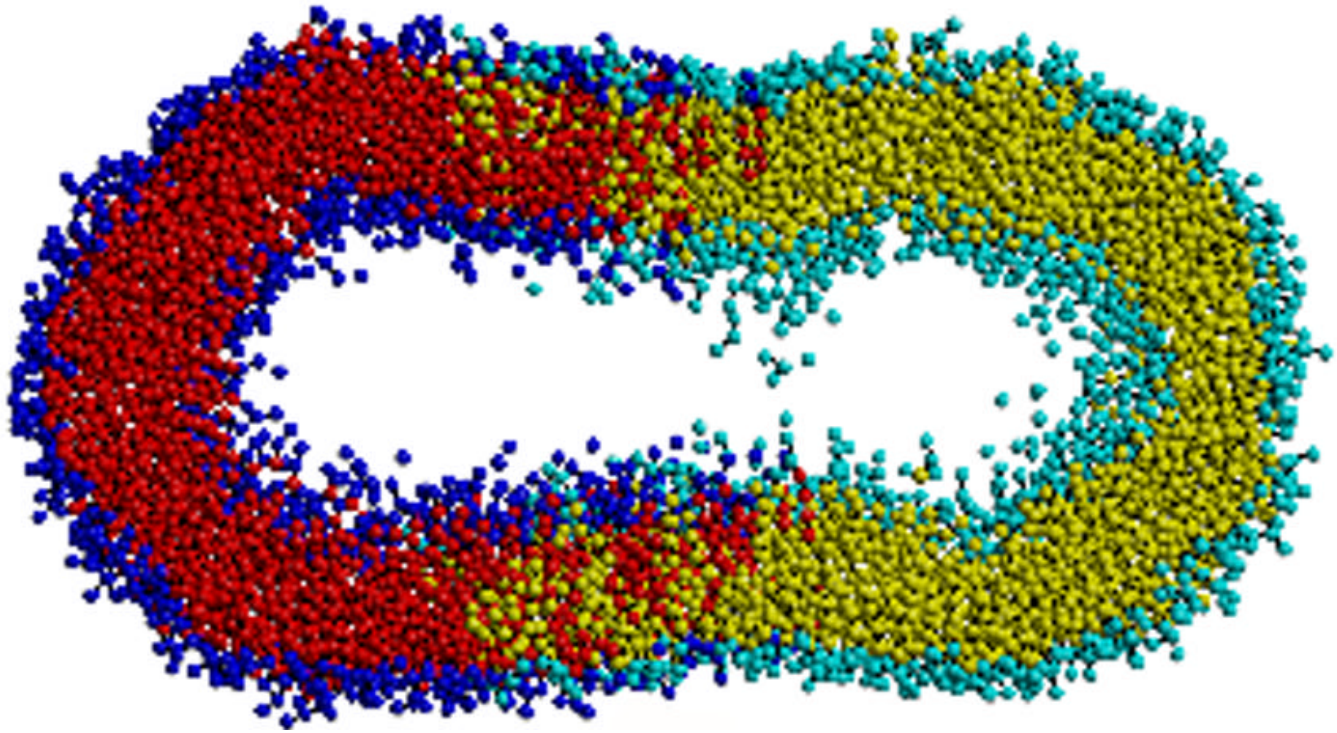
2158 lipids/liposome

333680 total beads

Mark Stevens
msteve@sandia.gov

Fusion Dynamics

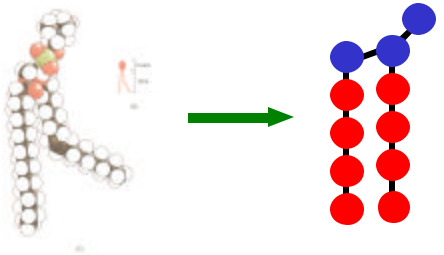




Mark Stevens
msteve@sandia.gov

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



Mark Stevens
msteve@sandia.gov

CG Lipid Simulations

Next Stage

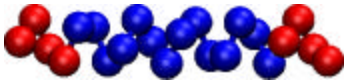
- Different lipid types
  - more complex models
- Free energy calculations
  - very very expensive
  - WHAM
    - constrain system (ex. two vesicles at fixed separation in fusion simulation)
    - do a full simulation at each constraint
    - collect statistics
    - calculate free energy difference from 'WHAM' equations
  - energy barriers
    - slow dynamics $\rightarrow$ expensive calculations
    - various methods proposed
    - multiple dimensions are a challenge
  - ? who is *funded*
- Including proteins in membrane simulations

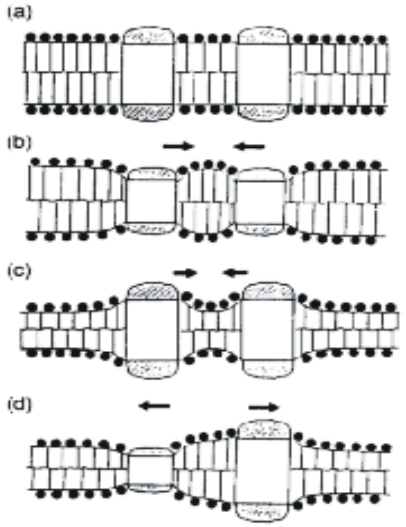
  (next page)

Mark Stevens
msteve@sandia.gov

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



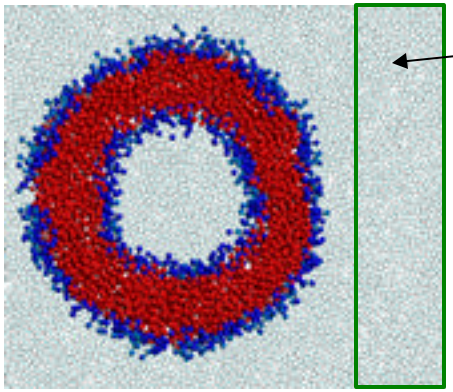


P.A. Kralchevsky et al., (1995)

Mark Stevens
msteve@sandia.gov

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



coarse grain more here

How treat
- interaction between different levels?
- movement of waters between different regions?

Mark Stevens
msteve@sandia.gov

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?

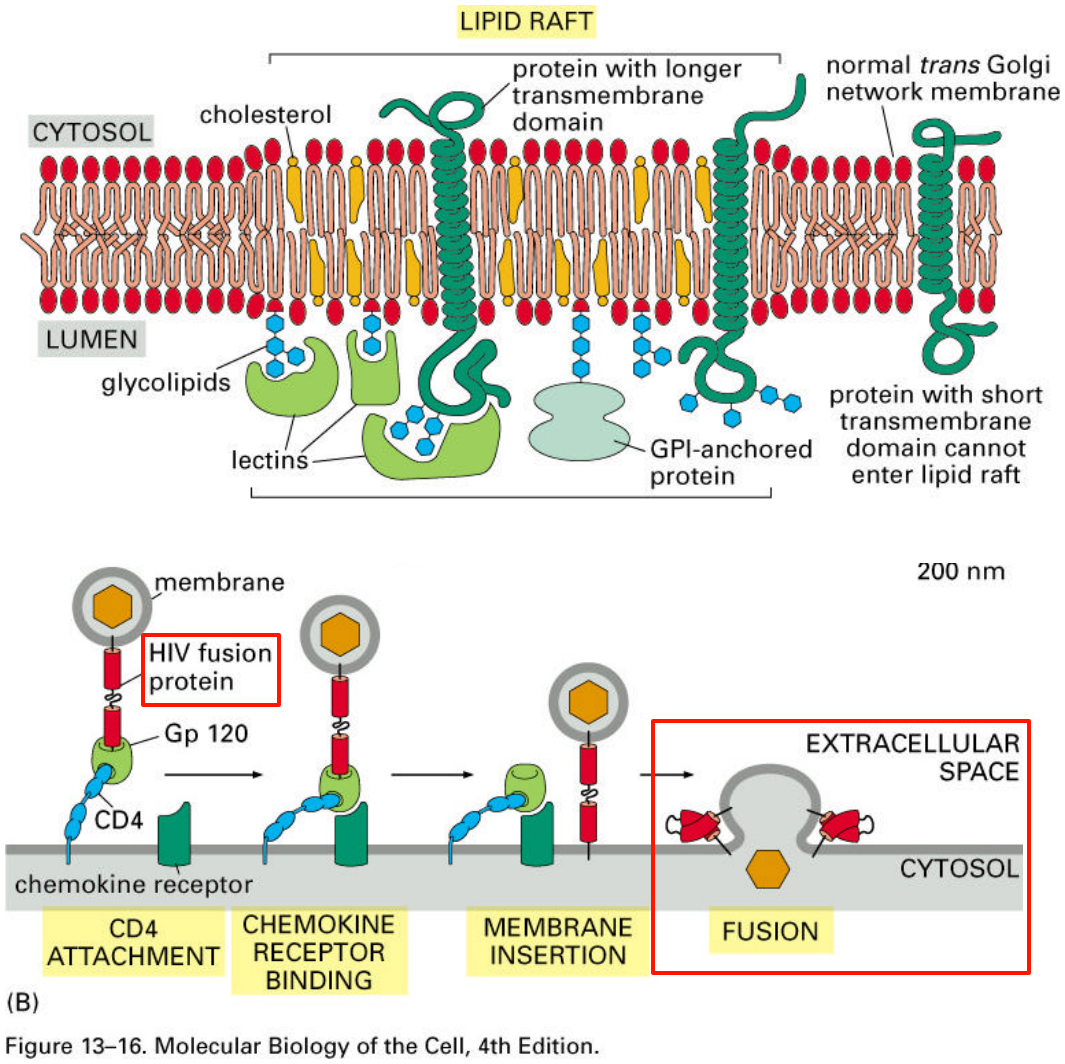


Figure 13–16. Molecular Biology of the Cell, 4th Edition.

Mark Stevens
msteve@sandia.gov

Nanoparticles

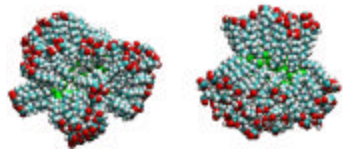
Mark Stevens
msteve@sandia.gov

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

1
2
2+N
NxN

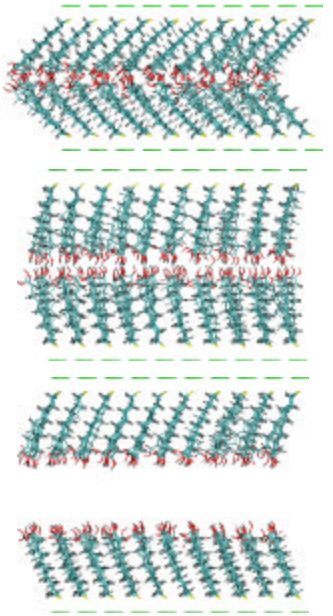


COOH terminated alkanethiols
air-decane air-water

Mark Stevens
msteve@sandia.gov

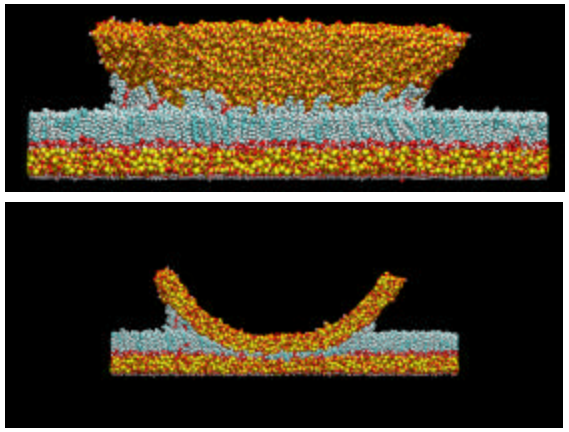
SAMs & AFM tips

- 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



coupling between terminal groups (COOH) and SAM structure as a function of separation

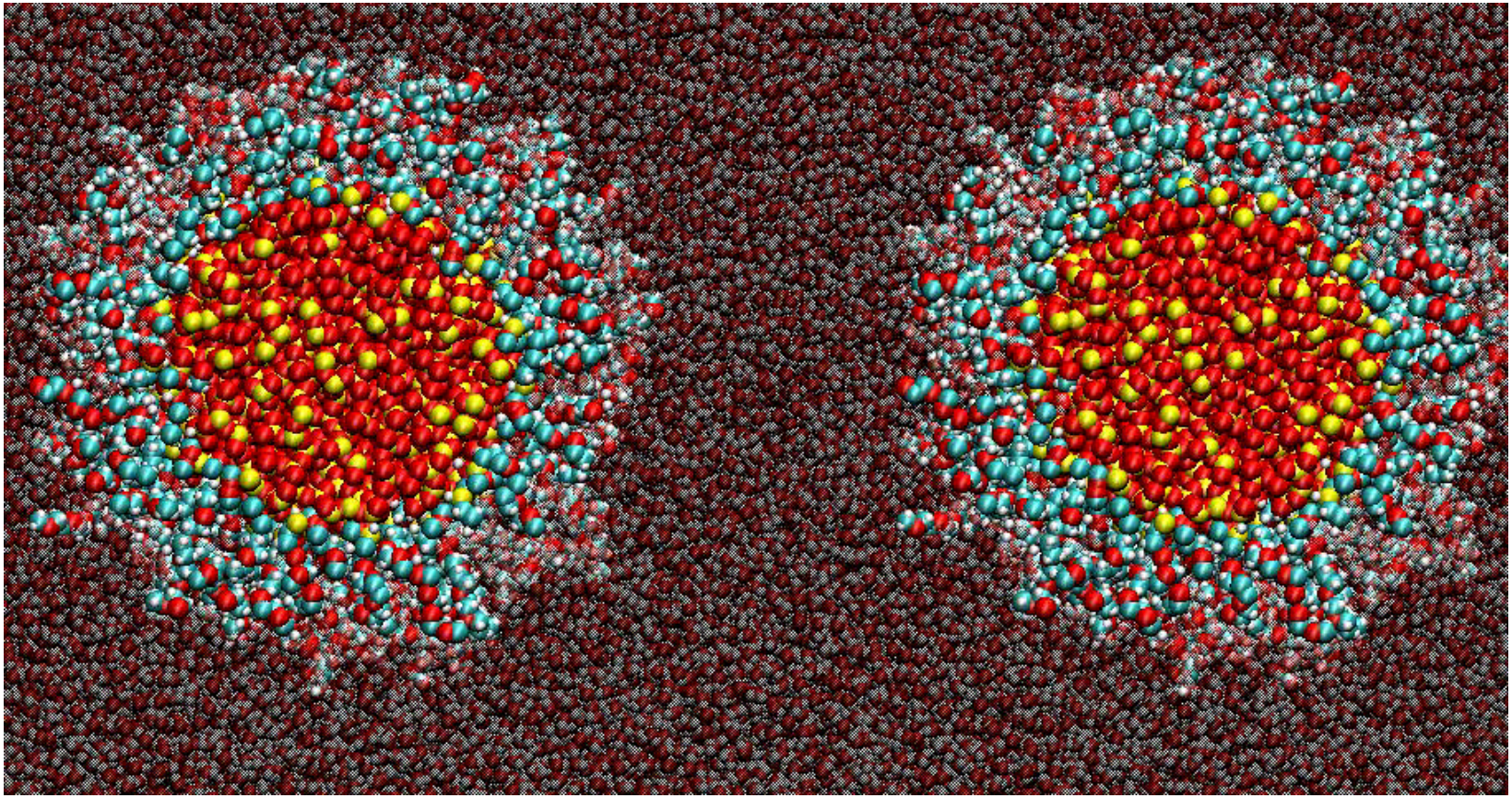
- Nanoparticles
  - coatings on nonplanar surfaces
  - will encompass a big leap in length and time scales
    - want to do more than 2 particles
    - particle dynamics on slower scale than solvent
    - **how reduce number of solvent molecules**



modeling of interaction between a tip and a SAM

Mark Stevens
msteve@sandia.gov

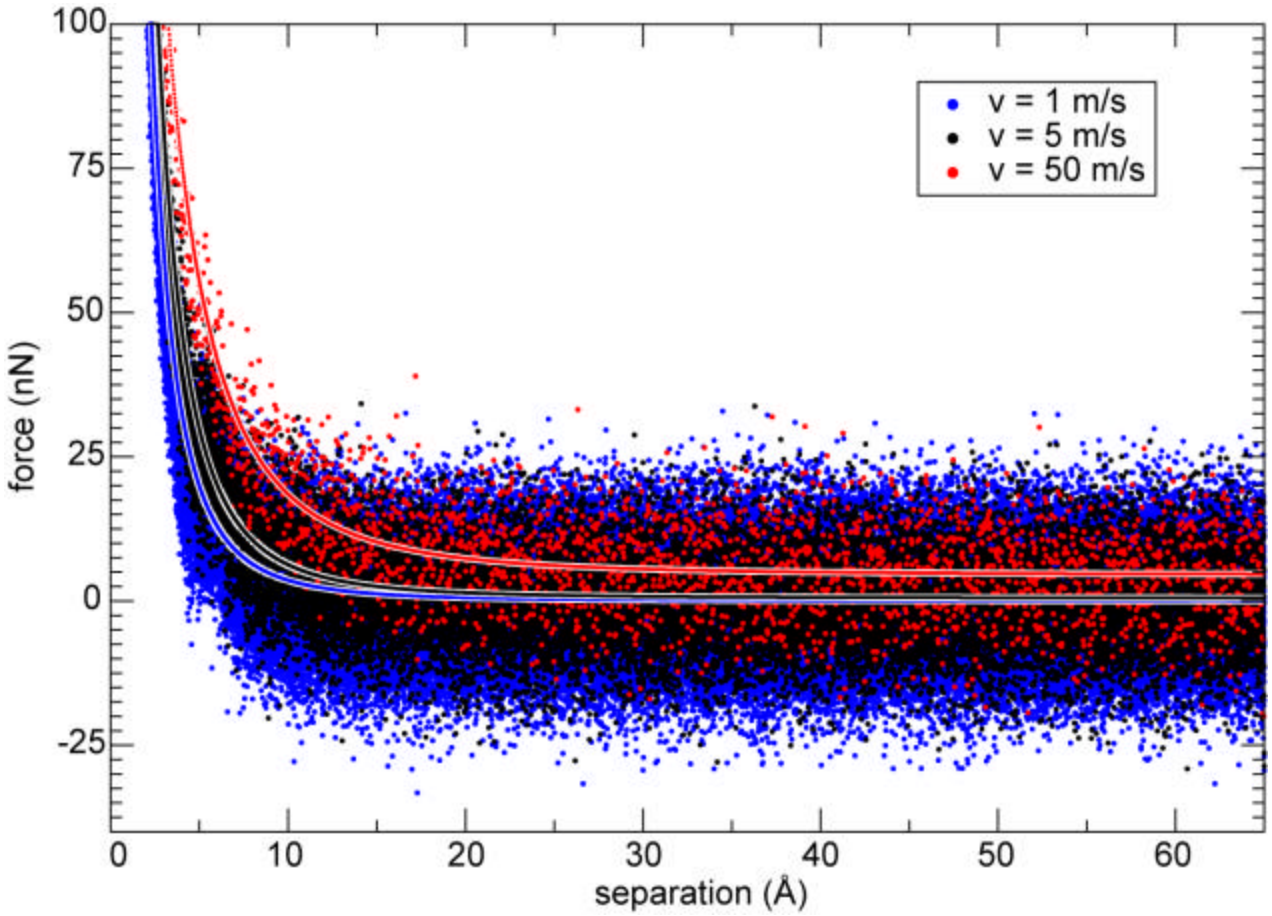
PEO-coated silica inter-particle force



- 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

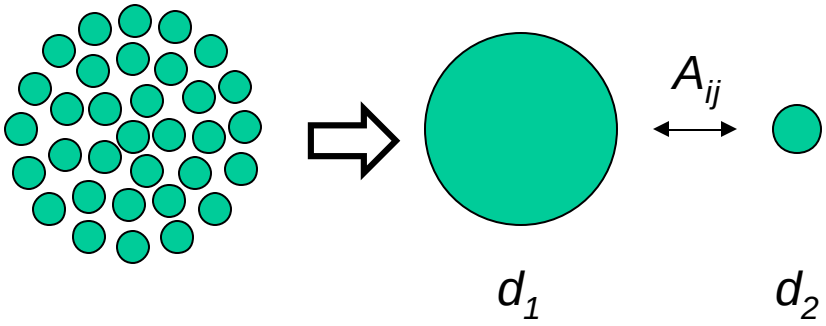
Mark Stevens
msteve@sandia.gov

Force on PEO coated Silica in water



Mark Stevens
msteve@sandia.gov

Interaction Potential



- r_c = 10Å
- 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

nano-solvent

$$U = \frac{2\,a_2^3\,\sigma_{12}^3\,A_{ns}}{9\,(a_2^2 - r_{12}^2)^3}\left[1 - \frac{(5\,a_2^6 + 45\,a_2^4\,r_{12}^2 + 63\,a_2^2\,r_{12}^4 + 15\,r_{12}^6)\,\sigma^6}{15\,(a_2 - r_{12})^6\,(a_2 + r_{12})^6}\right],$$

nano-nano

$$U_A = -\frac{A_{nn}}{6}\left[\frac{2a_1a_2}{r_{12}^2 - (a_1+a_2)^2} + \frac{2a_1a_2}{r_{12}^2 - (a_1-a_2)^2} + \ln\left(\frac{r_{12}^2 - (a_1+a_2)^2}{r_{12}^2 - (a_1-a_2)^2}\right)\right]$$

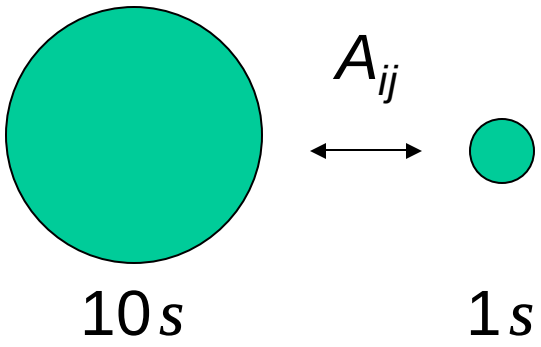
$$U_R = \frac{A_{nn}}{37800}\frac{\sigma^6}{r_{12}}\left[\frac{r_{12}^2 - 7r_{12}\,(a_1+a_2) + 6\left(a_1^2 + 7a_1a_2 + a_2^2\right)}{(r_{12} - a_1 - a_2)^7} + \frac{r_{12}^2 + 7r_{12}\,(a_1+a_2) + 6\left(a_1^2 + 7a_1a_2 + a_2^2\right)}{(r_{12} + a_1 + a_2)^7} - \frac{r_{12}^2 + 7r_{12}\,(a_1-a_2) + 6\left(a_1^2 - 7a_1a_2 + a_2^2\right)}{(r_{12} + a_1 - a_2)^7} - \frac{r_{12}^2 - 7r_{12}\,(a_1-a_2) + 6\left(a_1^2 - 7a_1a_2 + a_2^2\right)}{(r_{12} - a_1 + a_2)^7}\right]$$

$$U = U_A + U_R,$$

[1]R. Everaers and M.R. Ejtehadi, Phys. Rev. E **67**, 41710 (2003)

Mark Stevens
msteve@sandia.gov

Nanoparticles in Solvent



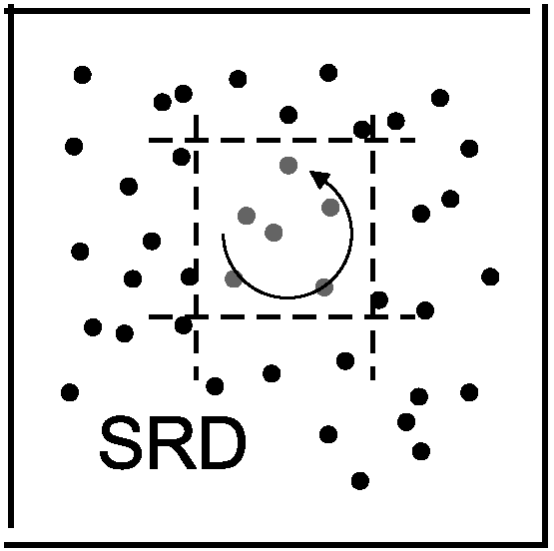
- Simulation details:
  - 10-2000 nanoparticles
  - 0.5-2 million LJ solvent particles
  - T= e/k_B , P=$0.1e/s^3$
- Simulations only feasible due to significant improvements in LAMMPS
  - Multi-region neighbor lists
  - Improved communications

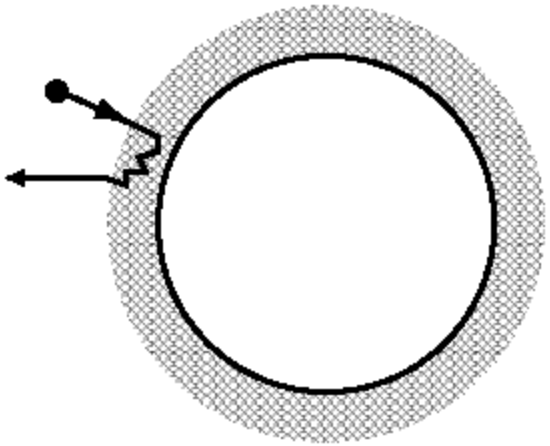
Pieter J. in 't Veld, Matt K. Petersen, Gary S. Grest, Steve Plimpton

Mark Stevens
msteve@sandia.gov

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)
  - Malevanets & Kapral, J Chem Phys, 110 (1999)
  - Padding & Louis, PRL, 93 (2004)
  - 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?



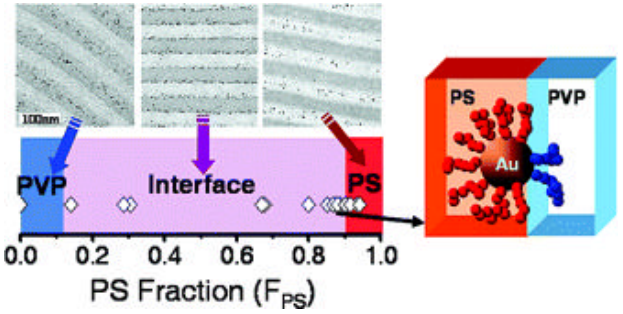


Mark Stevens
msteve@sandia.gov

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.



E. Kramer 2007

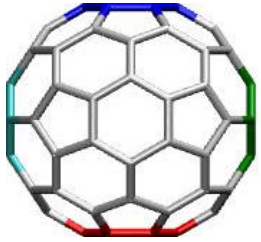
Mark Stevens
msteve@sandia.gov

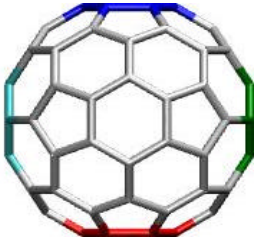
Rigid Body Dynamics

Mark Stevens
msteve@sandia.gov

Patchy Particles

- Use 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





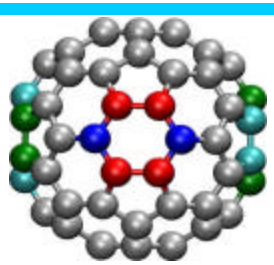
largest cluster

QuickTime™ and a
YUV420 codec decompressor
are needed to see this picture.

Mark Stevens
msteve@sandia.gov

Patchy Particles

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



clusters > 4

QuickTime™ and a
YUV420 codec decompressor
are needed to see this picture.

- 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

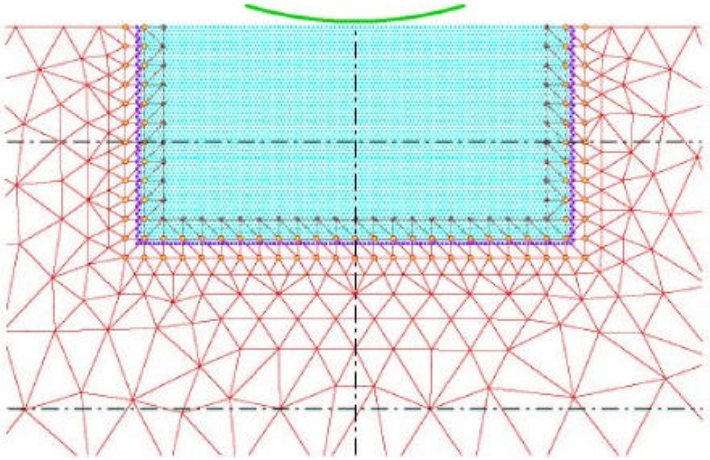
Mark Stevens
msteve@sandia.gov

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



Mark Stevens
msteve@sandia.gov

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

Mark Stevens
msteve@sandia.gov