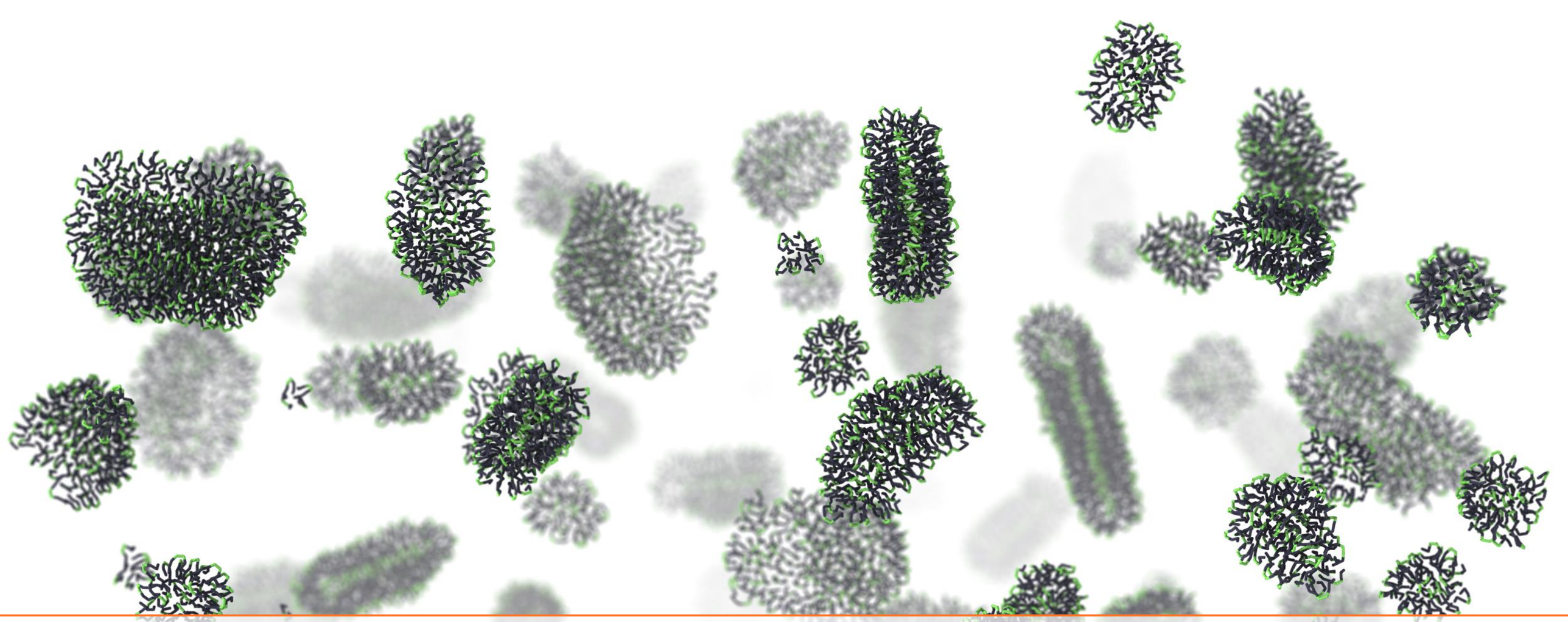


# Practicing Dissipative Particle Dynamics

Yu-Hang Tang

CSRC, September 24, 2015

BROWN

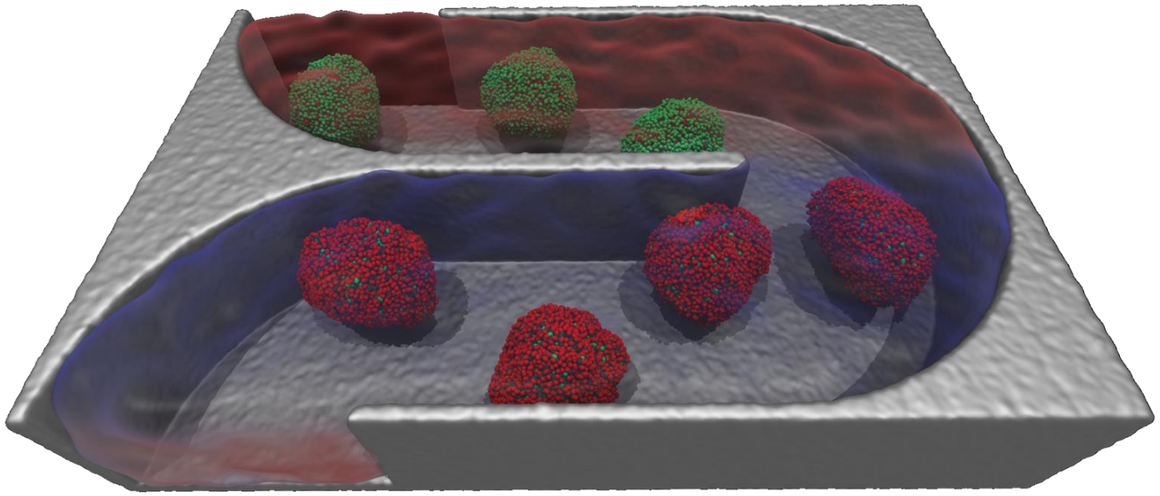
Pacific Northwest
NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

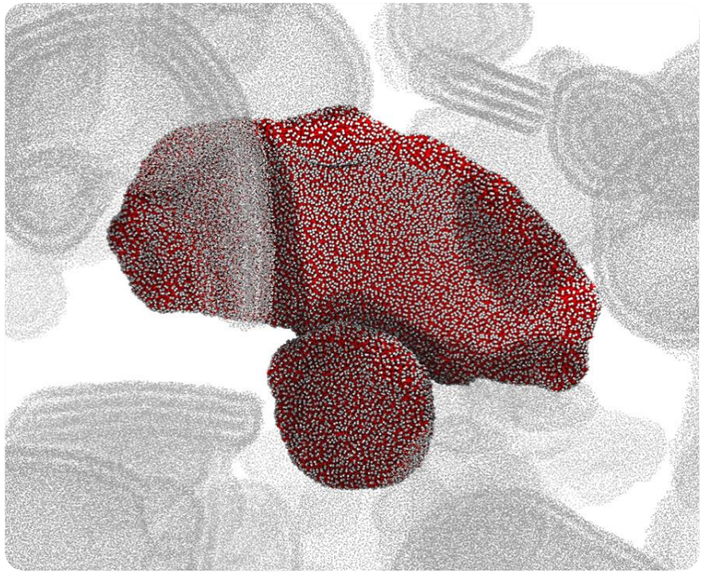
CM4
Collaboratory on Mathematics for
Mesoscopic Modeling of Materials

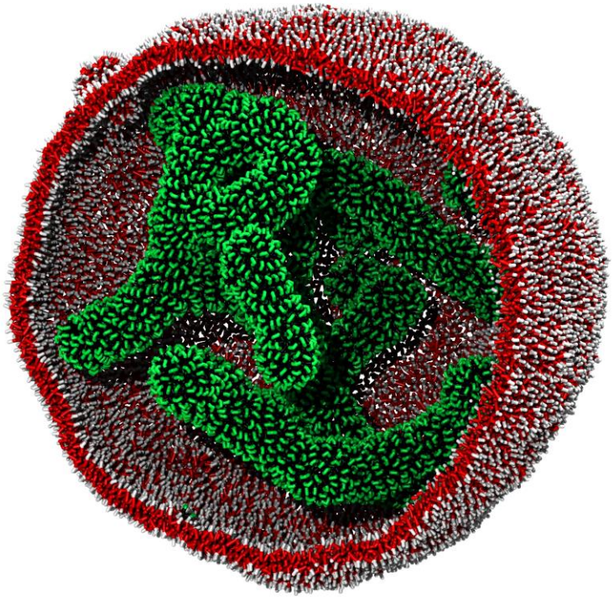
# Format: Discussion-based

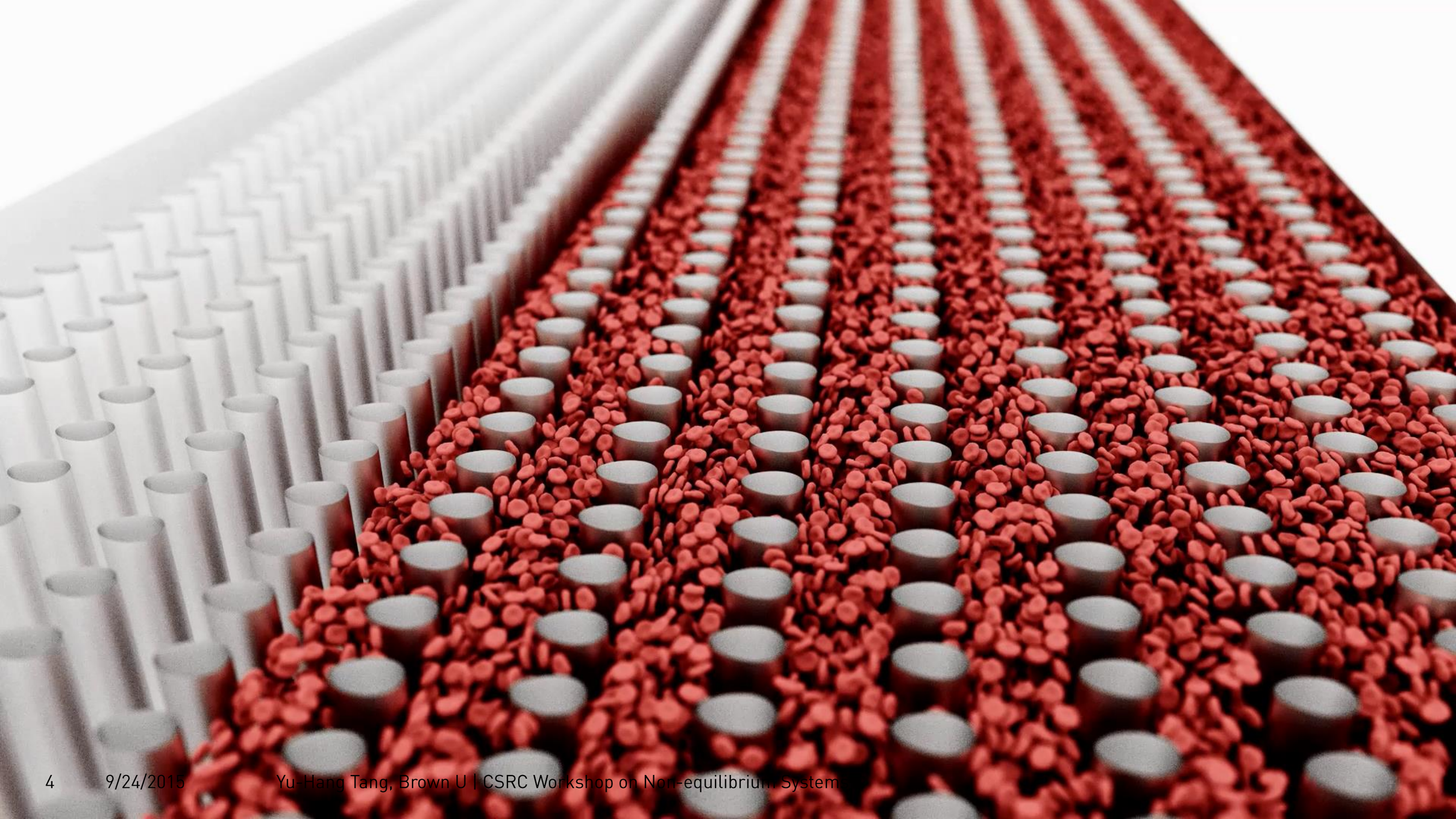
Feel free to interrupt at any time!

Dissipative Particle Dynamics is a stochastic, particle-based technique for simulating simple and complex fluids.

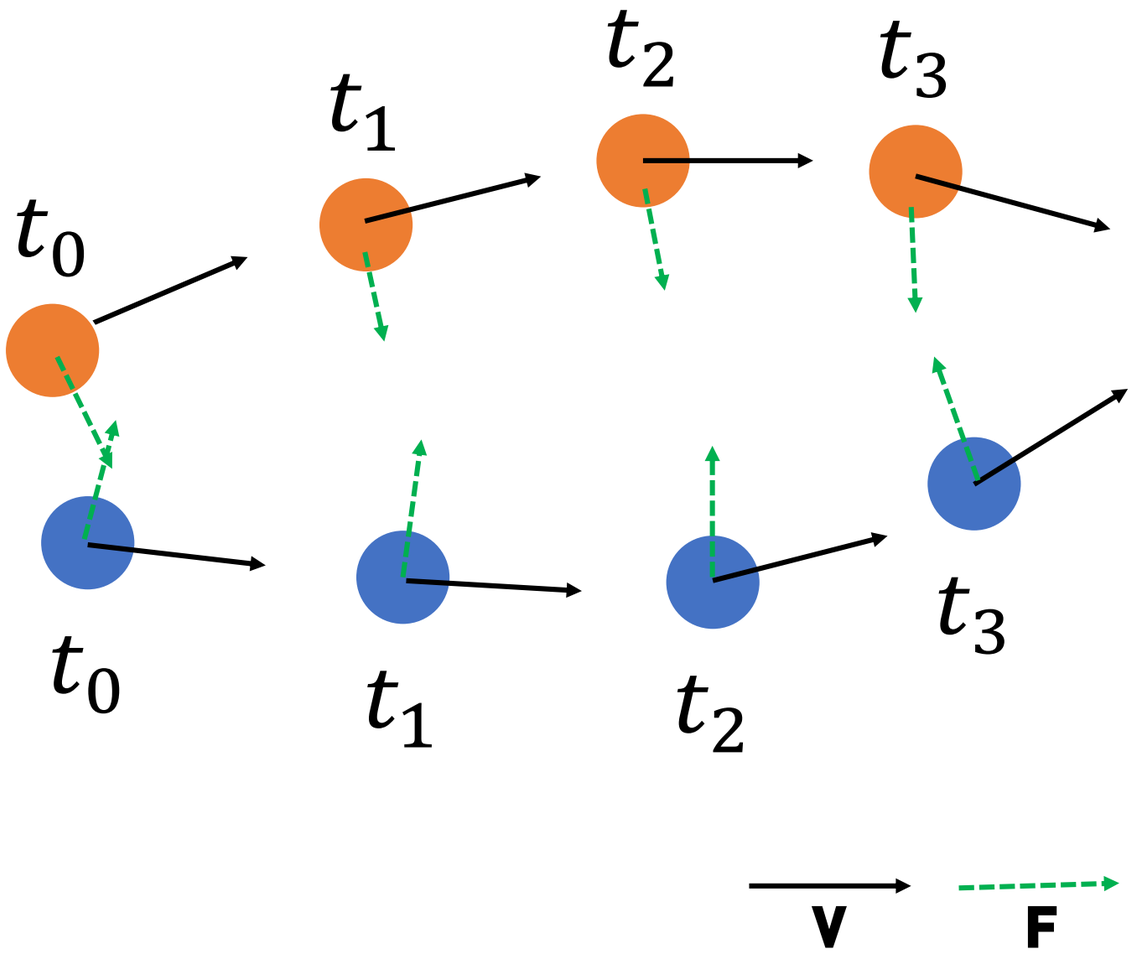








# In DPD we track the **motion of particles**.




$$m \cdot a(t) = f = -\frac{\partial H}{\partial x(t)}$$

$$v\left(t + \frac{\Delta t}{2}\right) = v(t) + \frac{1}{2}a(t)\Delta t$$

$$x(t + \Delta t) = x(t) + v\left(t + \frac{\Delta t}{2}\right)\Delta t$$

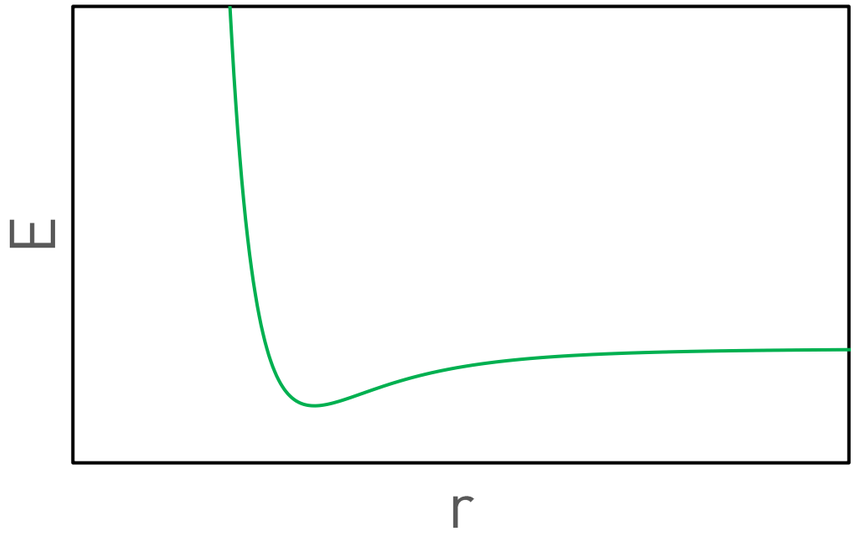
$$a(t + \Delta t) = -\frac{\partial H}{\partial x(t + \Delta t)}$$

$$v(t + \Delta t) = v\left(t + \frac{\Delta t}{2}\right) + \frac{1}{2}a(t + \Delta t)\Delta t$$

# Short-range pairwise interactions

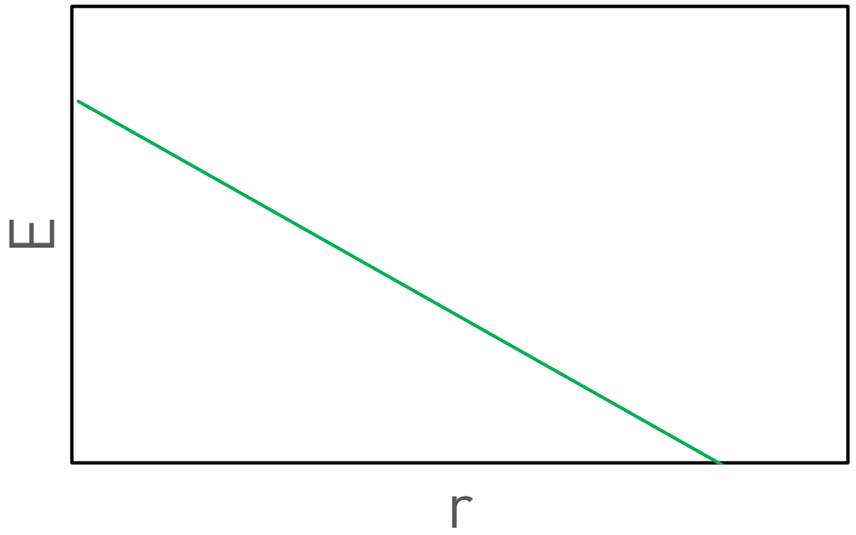
## Lenard-Jones Potential

$$\sigma\left[-2\left(\frac{r}{r_0}\right)^6+\left(\frac{r}{r_0}\right)^{12}\right]$$




## DPD Linear Force

$$a_{ij}\left[1-\frac{r}{r_0}\right]$$



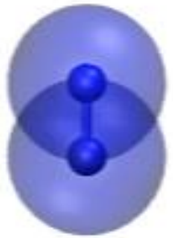
# Bonded potential

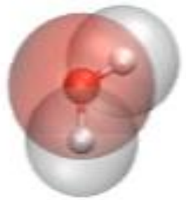
Bond

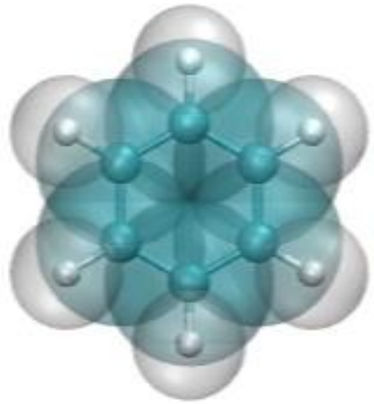
Angle

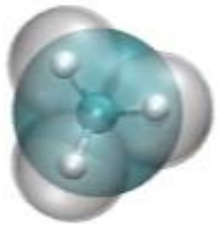
Dihedral

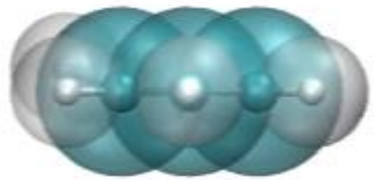
Improper









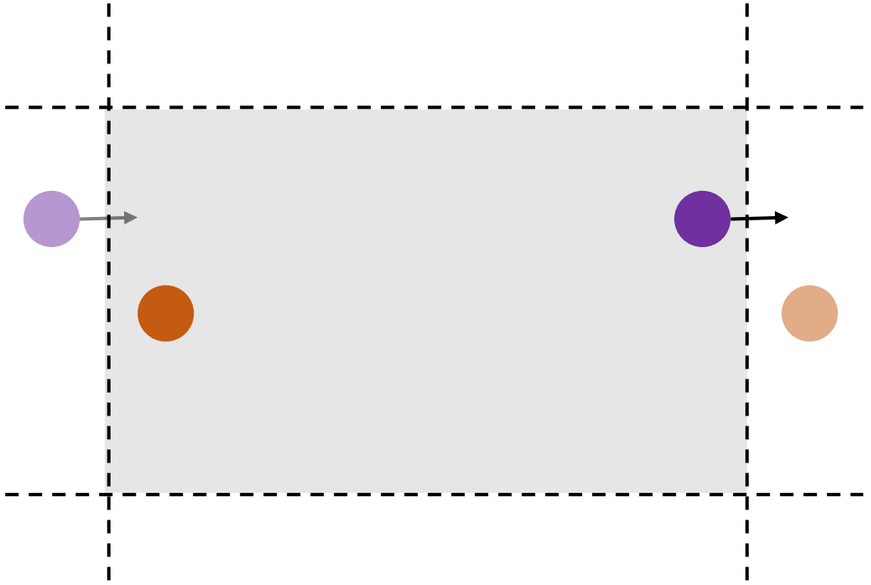


# Periodic boundary conditions

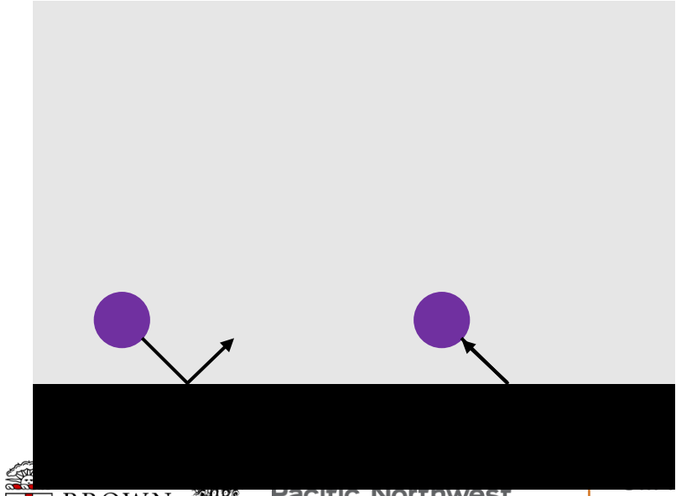
approximate infinite systems by tiling unit cells.

A particle passing through the boundary re-appears on the opposite side with the same velocity.

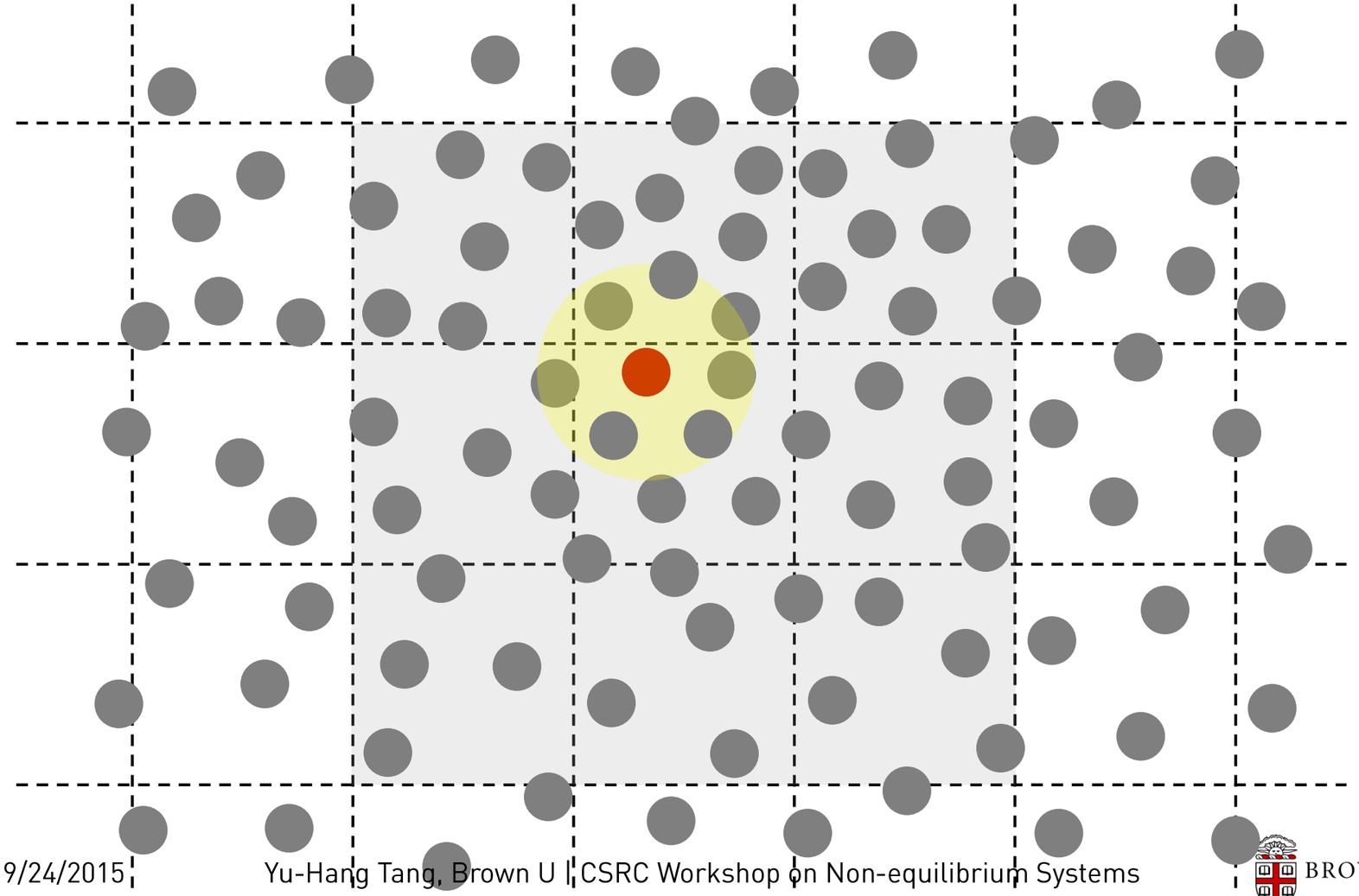
minimum image convention



# Bounce forward/back conditions

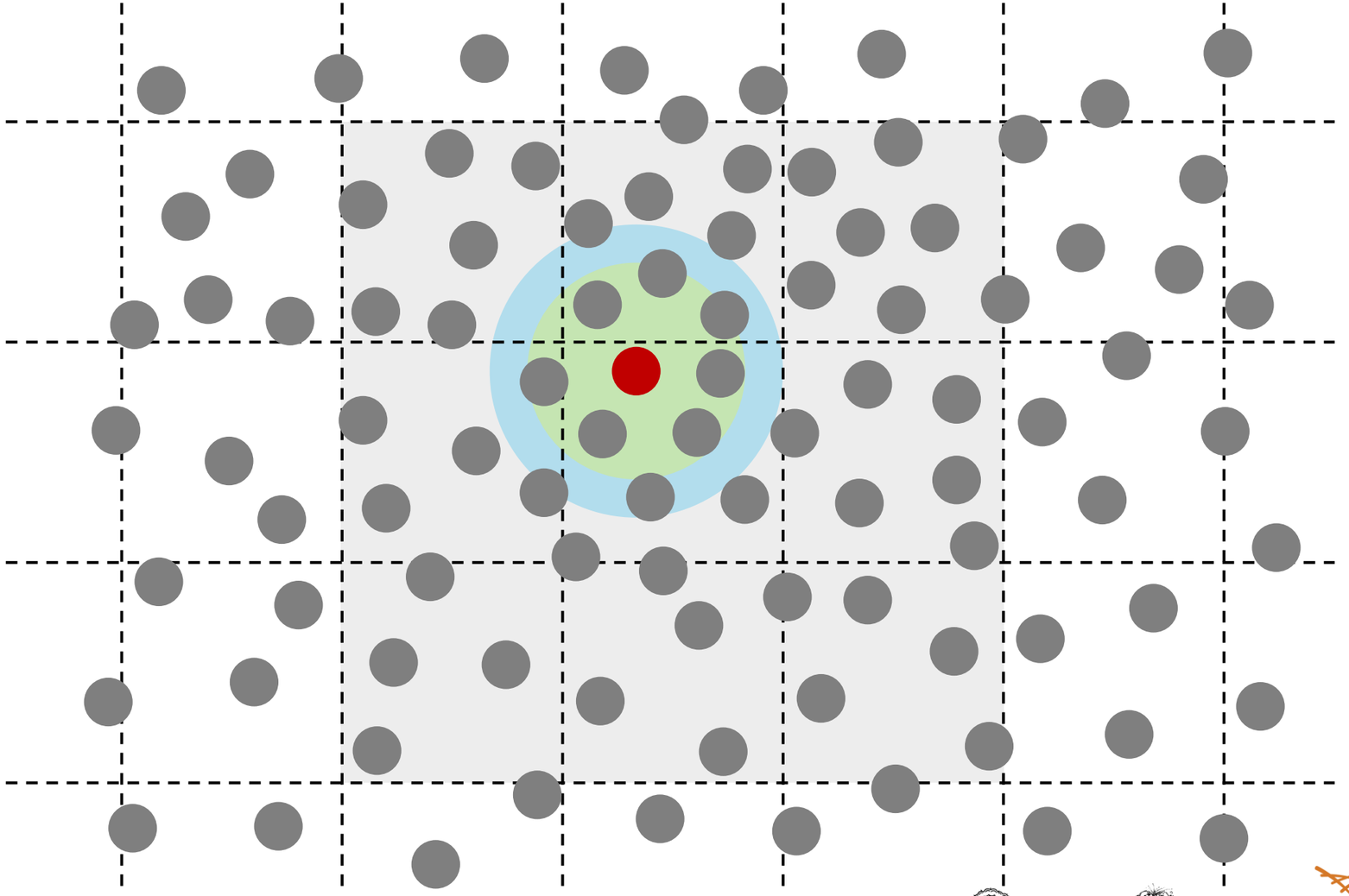


The **cell list** algorithm is ubiquitously employed to accelerate the evaluation of pairwise forces.

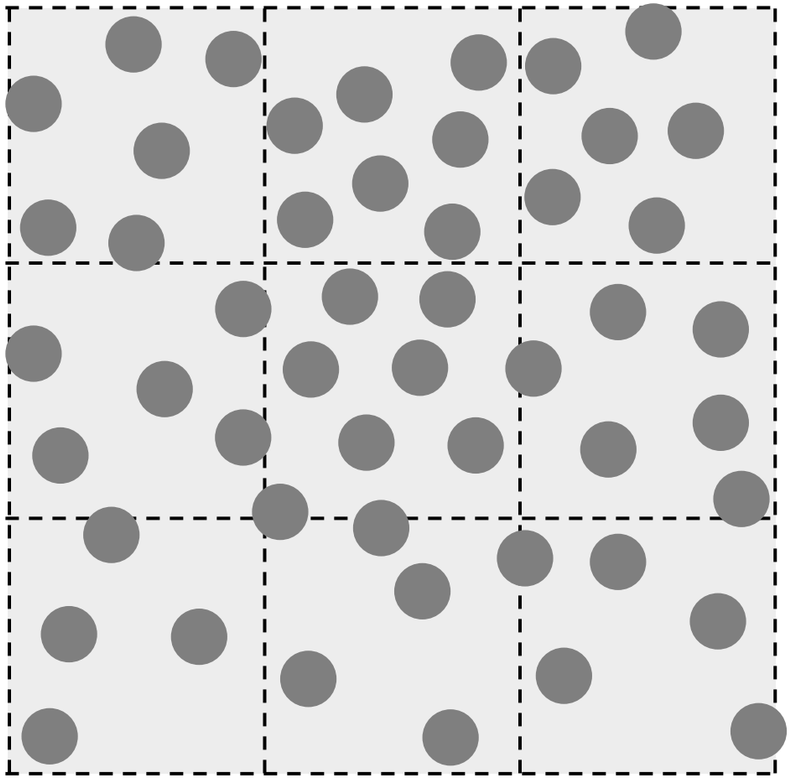


O(N^2) to O(N)

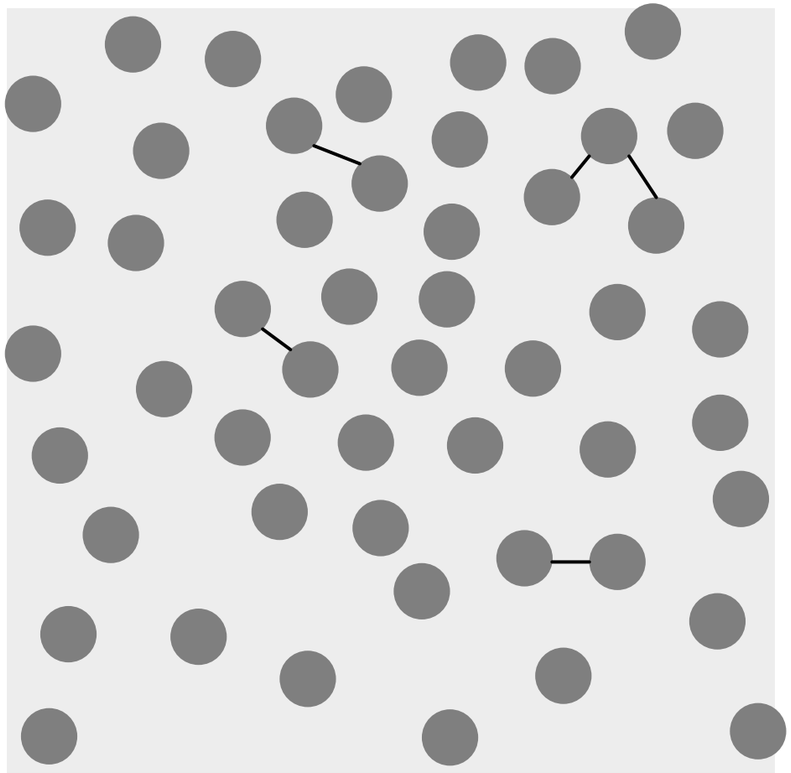
# A **neighbor list** with skin distance may be even faster.



# Domain decomposition



# Task decomposition



# LAMMPS

# =

# **L**arge-scale **A**tomic/**M**olecular **M**assively **P**arallel **S**imulator

http://lammps.sandia.gov/

- ✓ model particles at the atomic, meso, or continuum scale.
- ✓ serial or parallel using message-passing spatial decomposition
- ✓ easy to modify or extend
- ✓ open source (GPL)

# Installation

## Step 1. Download source

```
svn co svn://svn.lammps.org/lammps-ro/trunk lammps
```

## Step 2. Compile

```
cd lammps/src

make serial OR make mpi
```

## Step 3. Verify compilation

```
cd lammps/bench

../src/lmp_serial -in in.lj

mpirun -np 2 ../src/lmp_mpi -in in.lj
```

# Enable optional packages

## Molecule package

```
make yes-molecule
```

## USERMESO package (requires CUDA and MPI)

```
svn co http://www.cfm.brown.edu/repo/release/USER-MESO/ meso
cd meso
make yes-molecule
make yes-user-meso
make meso
```

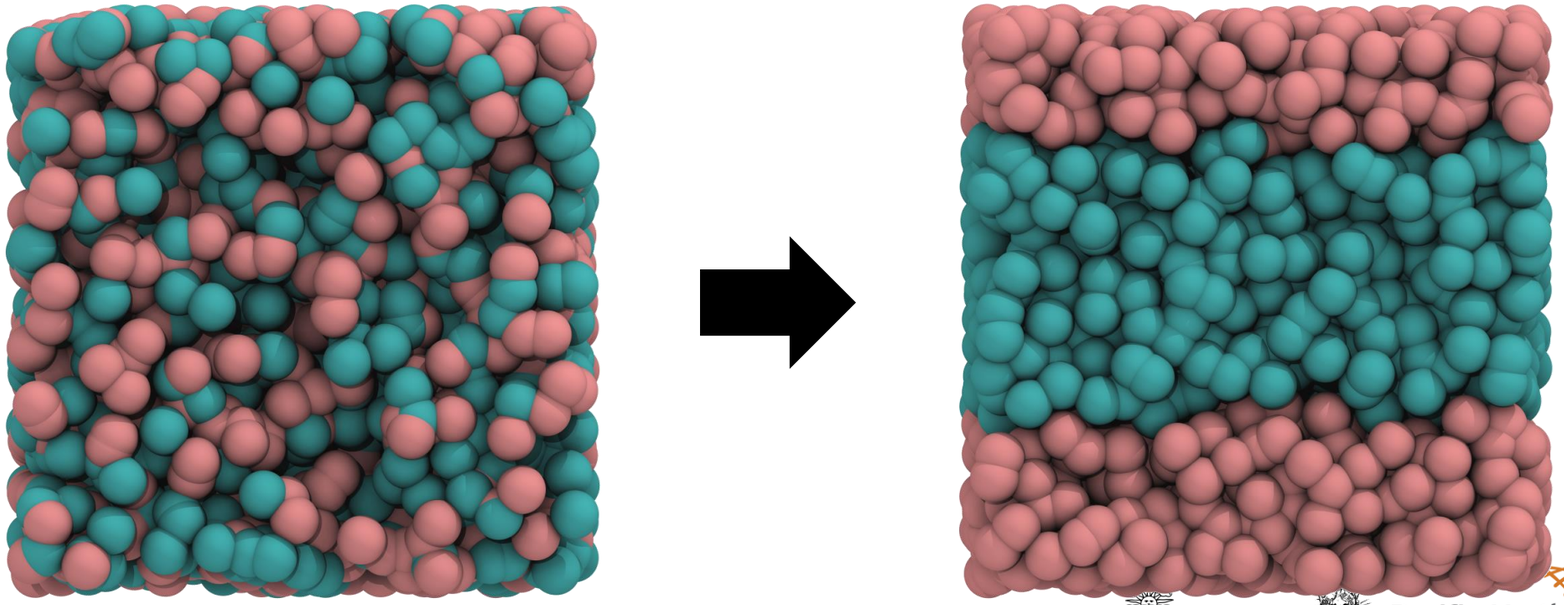
# Example I: Binary Immiscible liquid

Script at

http://www.cfm.brown.edu/repo/release/DPD-workshop/binary.lmp

Once finished, visualize with VMD

vmd binary.lammpstrj



# Line-by-line: **# DPD binary mixture**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

A comment. Specifically, the # sign may appear anywhere in a line, and all characters from there onward are treated as comment and discarded.

# Line-by-line: **dimension 3**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set the dimensionality of the simulation

syntax: dimension N

N: 2 or 3

# Line-by-line: **units lj**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set the unit used in the simulation to be 'lj', i.e. reduced units.

# Line-by-line: **comm_modify vel yes**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Exchange velocity data of particles during MPI parallel execution; necessary for DPD.

# Line-by-line: **newton off**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Whether or not to exploit Newton's 3rd law to accelerate computation.

Syntax: newton on|off

Setting newton to on accelerates serial computation but may lower performance at high parallelism; newton off mandatory for GPU acceleration.

# Line-by-line: **atom_style  atomic**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Select style of atom container.

Syntax: atom_style style

style = atomic, bond, full, etc.

Each atom style carries a different set of degrees of freedom.

# Line-by-line: **neighbor 0.3 bin**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Select style of neighbor list.

Syntax: neighbor skin style

skin = skin distance

style = bin | nsq

# Line-by-line: **neigh_modify delay 0 every 4 check no**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Configure neighbor list update frequency.

Syntax: every M delay N check yes|no

delay: wait for N steps before rebuilding

every: rebuild neighbor list every M steps

check: only rebuild if some atom has moved more than half the skin.

# Line-by-line: **region domain block -5 5 -5 5 -5 5 units box**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Define a region in the 3D space for later use; multiple regions can be define within each simulation

Syntax: region ID style args

In this case we built a cubic region of size 10x10x10 centered at the origin

# Line-by-line: **create_box 2 domain**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Create a simulation box based on the region previously defined.

Syntax: create_box N region

N: max number of atom types used in the simulation

# Line-by-line: **lattice fcc 4**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Define a lattice for the subsequent 'create_atoms' command

Syntax: lattice style scale

style = none | sc | bcc | fcc etc...

Note: this line is actually unused in the presented script but will be part of an exercise.

# Line-by-line: **create_atoms 1 random 1500 15255 domain**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Create atoms randomly in the simulation box

Syntax: create_atoms type style args

Specifically: create_atoms type random N seed region

type: type of the created particles

N: number of particles to create

seed: seed for random number generator

region: only create atoms in the region

# Line-by-line: **mass * 1.0**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set the mass for each type of particle

Syntax: mass type value

type: number or range or *

m: type m

n*: all types from n to N

*n: all types from 1 to n

m*n: all types from m to n

*: all types

# Line-by-line: **pair_style dpd 1.0 1.0 419084618**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Use the DPD style for pairwise force evaluation.

Syntax: pair_style dpd $k_BT$ $r_c$ seed

$k_BT$: temperature

$r_c$: cutoff distance

seed: seed for random number generator

# Line-by-line: **pair_coeff 1 1 25.0 4.5**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set up pairwise coefficients

Syntax: pair_coeff type1 type2 A gamma [$r_c$]

A: conservative force constant

gamma: dissipation strength

$r_c$ (optional): cutoff for a specific pair

# Line-by-line: **thermo 100**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

display diagnostics every N steps

# Line-by-line: **dump dump0 all atom 5 binary.lammpstrj**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1   25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2   25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Configure trajectory output

Syntax: dump ID group style args

ID: ID for the dump; multiple dumps with different IDs may be defined within a simulation.

group: the groups of particles to dump; a group 'all' containing all particles is predefined in LAMMPS

style: atom – LAMMPS's own trajectory format; may also be dcd, xyz etc.

args|atom = N filename

# Line-by-line: **velocity all create 1.0 8178251**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set particle initial velocity

Syntax: velocity group style args

Specifically:

velocity group create T seed

velocity group set vx vy vz

# Line-by-line: **fix  integrator all nve**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Uses the NVE integrator as a 'LAMMPS fix'.

Syntax: fix ID group style

# Line-by-line: **timestep 0.01**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Set time step size

# Line-by-line: **run 10000**

```
# DPD binary mixture

dimension       3
units           lj
comm_modify     vel yes
newton          off

atom_style      atomic
neighbor        0.3 bin
neigh_modify    delay 0 every 4 check no

region          domain block -5 5 -5 5 -5 5 units box
create_box      2 domain
lattice         fcc 4
create_atoms    1 random 1500 15255 domain
create_atoms    2 random 1500 47214 domain
mass            * 1.0

pair_style      dpd 1.0 1.0 419084618
pair_coeff      1 1  25.0 4.5
pair_coeff      1 2 150.0 4.5
pair_coeff      2 2  25.0 4.5

thermo          100
dump            dump0 all atom 5 binary.lammpstrj

velocity        all create 1.0 8178251
fix             integrator all nve

timestep        0.01
run             10000
```

Run the simulation for 10000 steps

Instead of 'creating' particles we can also load them through **data file**.

Each atom style takes a slightly different format, i.e. different # of columns in the Atoms section, etc.

Check LAMMPS website for the specifics

```
# data file sample

1000 atoms

1 atom types

0 10 xlo xhi
0 10 ylo yhi
0 10 zlo zhi

Masses

1 1.000000

Atoms

1 1 8.401877172 3.943829268 7.830992238
2 1 7.984400335 9.116473579 1.975513693
3 1 3.352227557 7.682295948 2.777747108
...
```

# Style

Styles is the mechanism used by LAMMPS to allow scalable functionality extension.

From a user's perspective they serve as options to various LAMMPS commands.

From a developer's perspective they are C++ derived classes of a virtual base class. Overloading the virtual interfaces allows the custom functionalities to be implemented.

# Groups

A group is a collection of particles. The group ID can be used in other commands such as fix, compute, dump, or velocity to act on those atoms together.

Syntax:

group ID style args

style = region, type, etc...

# Fix

A 'fix' is any operation that is applied to the system during time stepping. Examples include updating of atom positions and velocities due to time integration, controlling temperature, applying constraint forces to atoms, enforcing boundary conditions, computing diagnostics, etc.

Fixes perform their operations at different stages of the timestep.

http://lammps.sandia.gov/doc/fix.html

# Advanced topic: implementing custom fixes

## Verlet scheme, stages and fix class

# Thank you for attending!

Y.H.T. appreciates invitation and support from **Dr. Zhen Li**, **Dr. Zhewei Zhou**, **Dr. George Karaniadakis**, and **Shanghai Institute of Applied Mathematics and Mechanics**, **Shanghai Center for Nonlinear Sciences**, **Pacific Northwest National Lab**, and **Beijing Computational Science Research Center**.

BROWN

Pacific Northwest
NATIONAL LABORATORY
*Proudly Operated by* Battelle *Since 1965*

CM4
Collaboratory on Mathematics for Mesoscopic Modeling of Materials