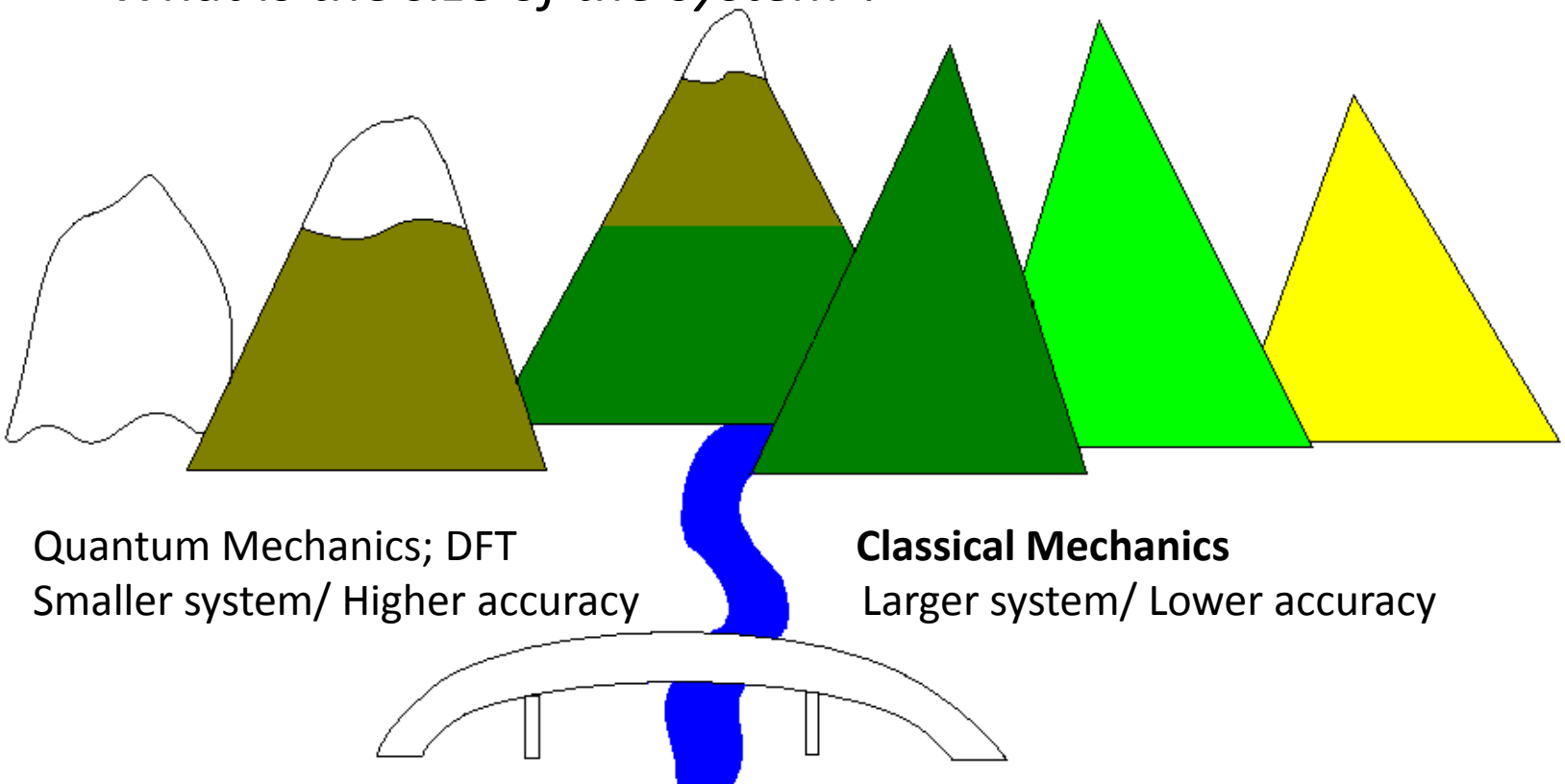
Computational Chemistry Molecular Dynamics: Programming to Production

Matt McKenzie, Ph.D.
IT Consultant
LSU HPC

General Comments on Molecular Simulation

- Three questions you should ask before starting a computational project,
  - What are *properties of interest* ?
  - What *types potentials are needed* ?
  - What is the *size of the system* ?



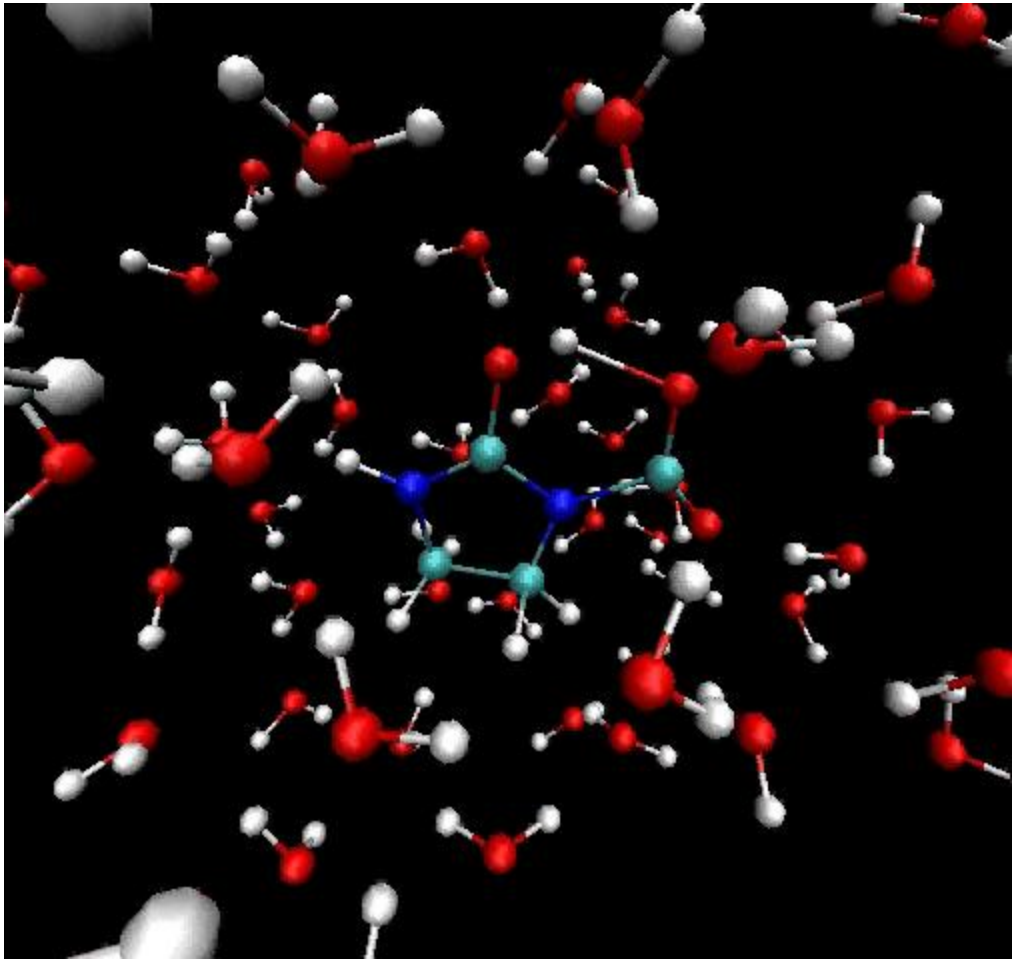
Quantum Mechanics; DFT
Smaller system/ Higher accuracy

Classical Mechanics
Larger system/ Lower accuracy

What is Molecular Dynamics?

General Background

The process in which one generates a system's *trajectory* via *numerical integration of Newton's equations of motion* with an appropriate interatomic potential and proper initial & boundary conditions.



Molecular Dynamics

- Simulation model or system comprises of
  - *N* particles in a volume, *V*, at temperature, *T*
    - This is an ensemble of parameters
      - NVT a.k.a the Canonical ensemble

$$\{r(t)\} = (r_1(t), r_2(t), r_3(t), \ldots, r_N(t))$$

- The position of the system's *N* particles are specified by a *N* set of vectors at time, *t*

$$E = K + U$$

- Total Energy (*E*) is the sum of Kinetic (*K*) and Potential (*U*) Energies

Kinetic Energy

- Definition,

$$K = \frac{1}{2} m_i \sum_{i=1}^{N} v_i^2$$

- Temperature is computed as the average kinetic energy per degree of freedom, *f*

$$\frac{1}{2} k_B T = \left\langle \frac{1}{2} m v^2 \right\rangle$$

$$k_B T = \frac{\langle 2K \rangle}{f}$$

- Temperature is controlled in this manner

Potential Energy

- Potential Energy is dependent upon particle positions

$$U_{Total} = U(r_1(t), r_2(t), r_3(t), \ldots, r_N(t))$$

- For a three particle system

$$U_{Total} = U_{ij} + U_{jk} + U_{ik} + U_{ijk}$$

U_{ijk} is the three body energy term

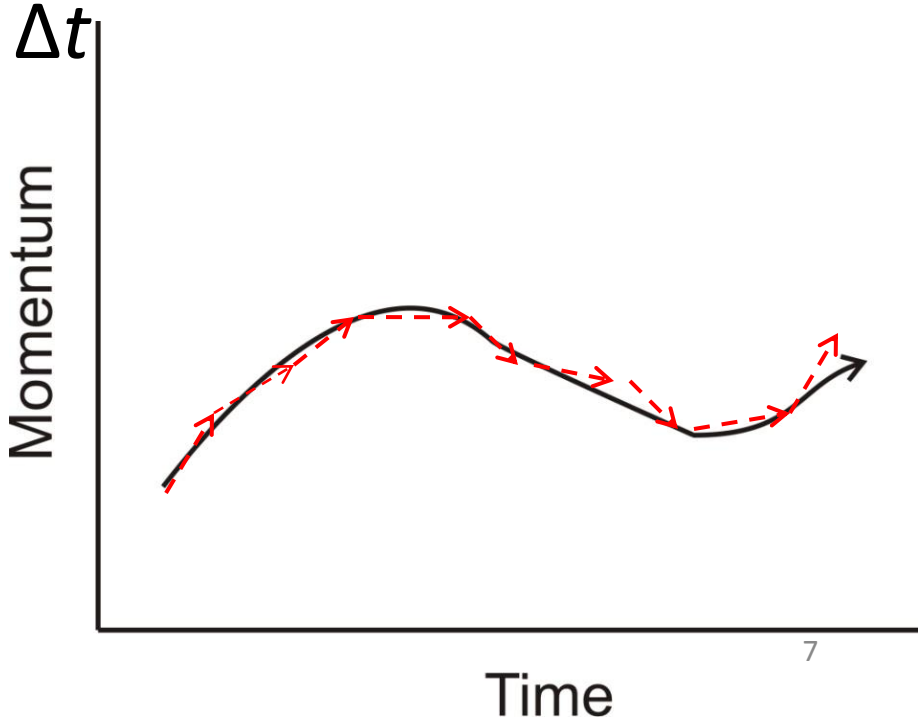
- The true potential would include, pair-, three-body, four-body, … to N^{th}-body
- Truncate to 2 bodies, pair-wise additive approximation

Move the atoms

"Integrate" the equations of motion

- The overall simulation time is divided into smaller time steps
  - This is the integration interval
- Given the initial condition of $\{r(t_0)\}$, advance system by increments of Δt

$$m\frac{d^2 r_i}{dt^2} = -\nabla U(\{r_N\}) \quad i = 1,\ldots, N$$



How to get to the next step?

$$x(t) = x(t_0) + v(t_0)\Delta t + a\Delta t^2$$

- Taylor expand around time, t-Δt and t+Δt, then add the two equations
  - F/m = a

Verlet equation

$$x(t+\Delta t) = 2x(t) - x(t-\Delta t) + a(t)\Delta t^2$$

Where current x is, Where x was, and the Acceleration (force)

The Trajectory

- Trajectory = molecular movie
- List of atomic positions in successive time steps

$$\{r(t_0)\} \rightarrow \{r(t_0 + \Delta t)\} \rightarrow \{r(t_0 + 2\Delta t)\} \rightarrow \ldots \rightarrow \{r(t_0 + t_n \Delta t)\}$$

- Write out all positions every ___ time steps
  - Can contain velocities
- Useful for analysis
  - Diffusion coefficient, radial distribution function

Programming Molecular Dynamics

- For sample program, log onto Queenbee.loni.org
- /home/mmcken6/TRAINING/MD_Prog2Prod/
- File descriptions

md.f	MD code
fort.40*	Input file
fort.77	Output trajectory

Molecular Dynamics Code

- Simple reduced units monatomic Lennard Jonesium MD code in the NVT ensemble

- Illustrate MD and programming fundamentals

- One can further develop the code to include
  - Molecules
  - Force field parameters
  - Other ensembles

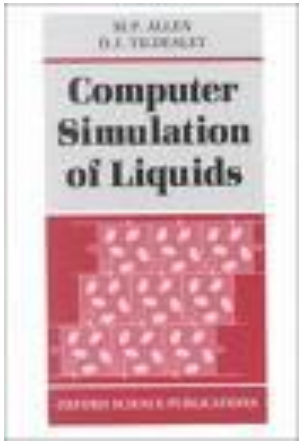
Resources

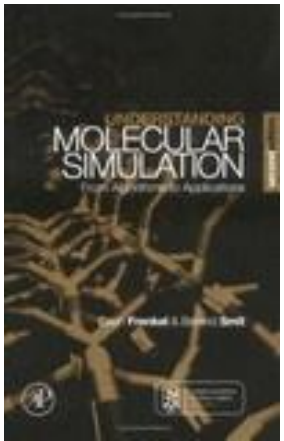
Computer Simulation of Liquids M. P. Allen and D. J. Tildesley

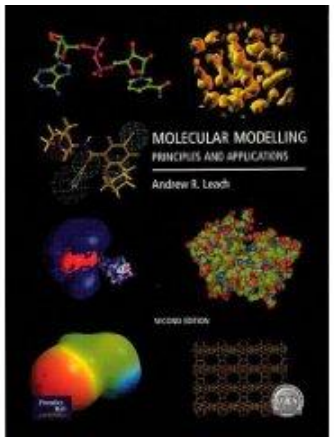
Understanding Molecular Simulation D. Frenkel and B. Smit

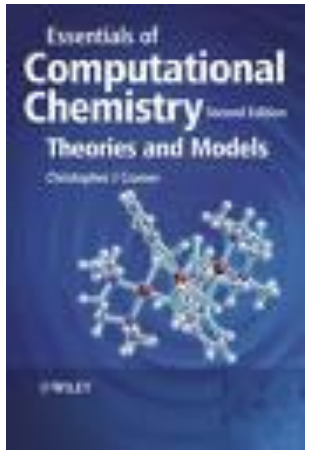
Molecular Modelling A. Leach

Essentials of Computational Chemistry C. J. Cramer







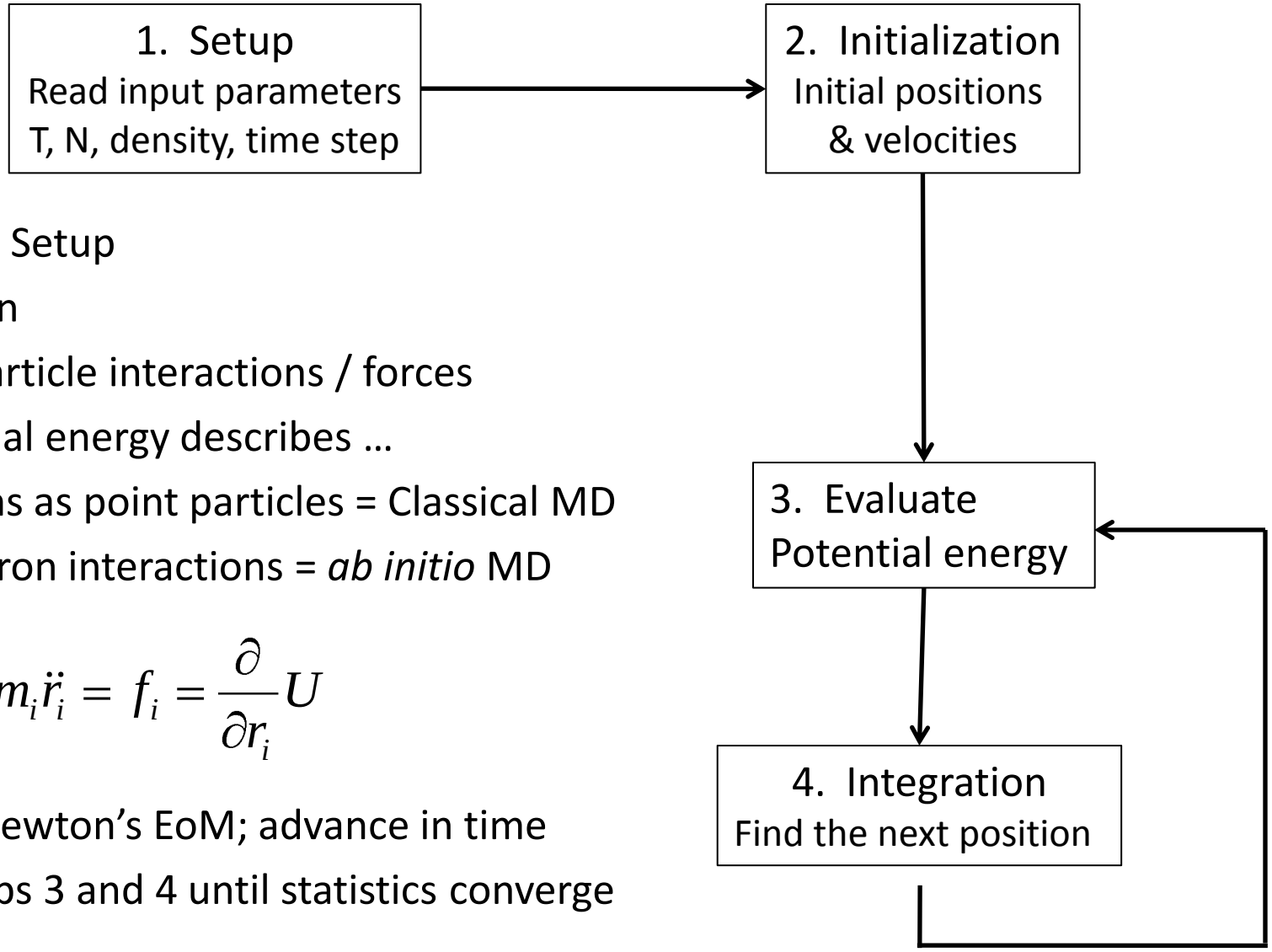


Reduced Units

- Law of Corresponding States
  - infinitely many combinations of density, temperature, mass, energy, etc. all correspond to the same state in reduced units
- $\rho^* = 0.5$ and $T^* = 0.5$ corresponds to both
  - Argon at a state point characterized by T = 60K, $\rho = 840$ kg / m^3
  - Xenon at a state point characterized by T=112K and $\rho = 1617$ kg / m^3

- SI unit simulation computed quantities are usually very large or small compared to 1
- Risk when multiplying such quantities that it will create an over/underflow
- Errors become more difficult to detect

Molecular Dynamics Program



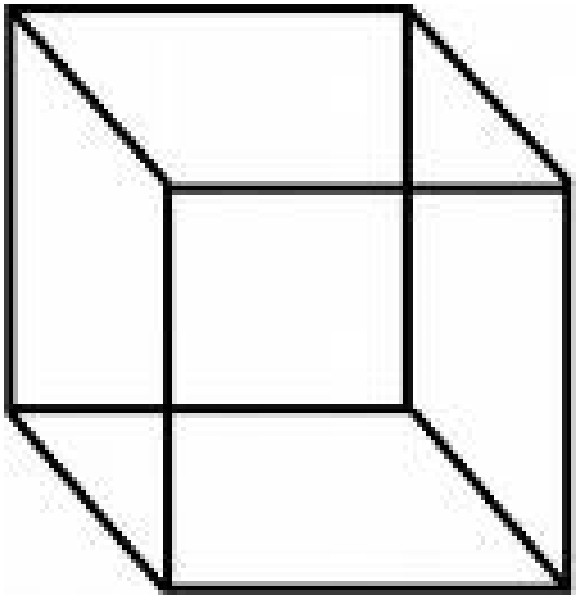
1. Simulation Setup
2. Initialization
3. Evaluate particle interactions / forces
   - If Potential energy describes …
     - Atoms as point particles = Classical MD
     - Electron interactions = *ab initio* MD

$$m_i \ddot{r}_i = f_i = \frac{\partial}{\partial r_i} U$$

4. Integrate Newton's EoM; advance in time
5. Repeat Steps 3 and 4 until statistics converge

1. Setup

- Need to define
  - Creation of the central simulation box
    - Rectangular, cubic
    - Dimensions?
    - Box length = $(N / \rho)^{1/3}$
    - **Density** is very important
      - solid? liquid? gas?
  - Simulation constants
    - For example: N, P, T



Simulation Constants

- The set of constants is called an ensemble
  - Defines the thermodynamic state of a system

Ensemble Name	Constants	Char. State Function	Char. Thermo. Function
Microcanonical	N, V, E	Ω	-TS
Canonical	N, V, T	Z	F or A ("Arbeit")
Isothermal – Isobaric	N, P, T	Δ	G
Grand Canonical	μ, V, T	Ξ	PV

1. Setup: Fortran basics
Reading in constants

```
[user@machine]$ more fort.40
T
0.6   0.9
1200   0.001
```

```
INP = 40
Subroutine INIT  (… , …, …, …, ….)
    READ(INP,*)CUBIC
    READ(INP,*)RHO,TEMP
    IF (CUBIC) THEN
      BOXL(1) = (NPART / RHO)**(1.D0/3.D0)
      BOXL(2) = BOXL(1)
      BOXL(3) = BOXL(1)
    ELSE
      READ(INP,*)
      READ(INP,*)BOXL(1),BOXL(2),BOXL(3)
      READ(INP,*)
    ENDIF
    VOL = BOXL(1) * BOXL(2) * BOXL(3)
    IF (((NPART / VOL) - RHO) .GT. 1.D-5) THEN
      WRITE(OUTP,*) 'DENSITY INCORRECT'
      STOP
    ENDIF
    READ(INP,*)NSTEP, TSTEP
    RETURN
    END
```

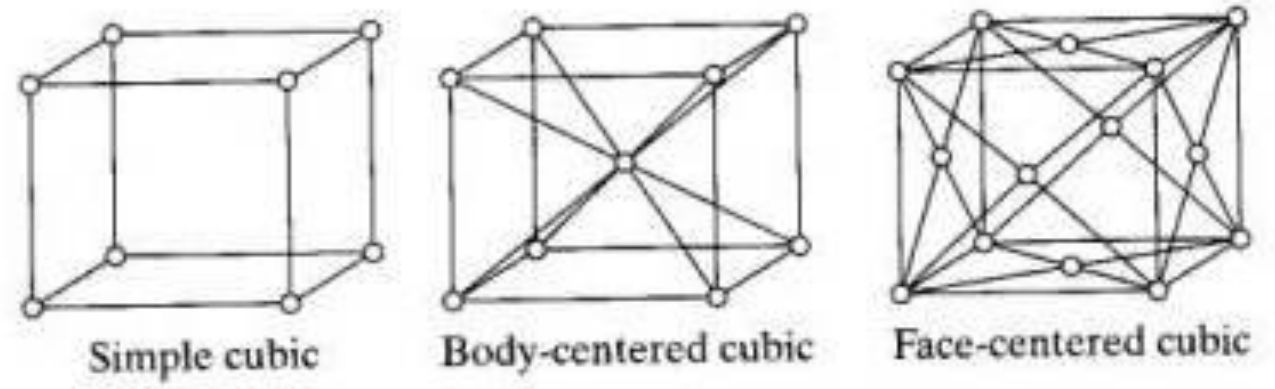
Successively reading in parameters

Is this a cubic simulation box?

Density check

2. Initialization

- How to obtain a starting geometry
  - Start from a solid lattice
    - Melt this structure to become a liquid/gas
  - How to place atoms on a lattice?
  - Look at inorganic crystals



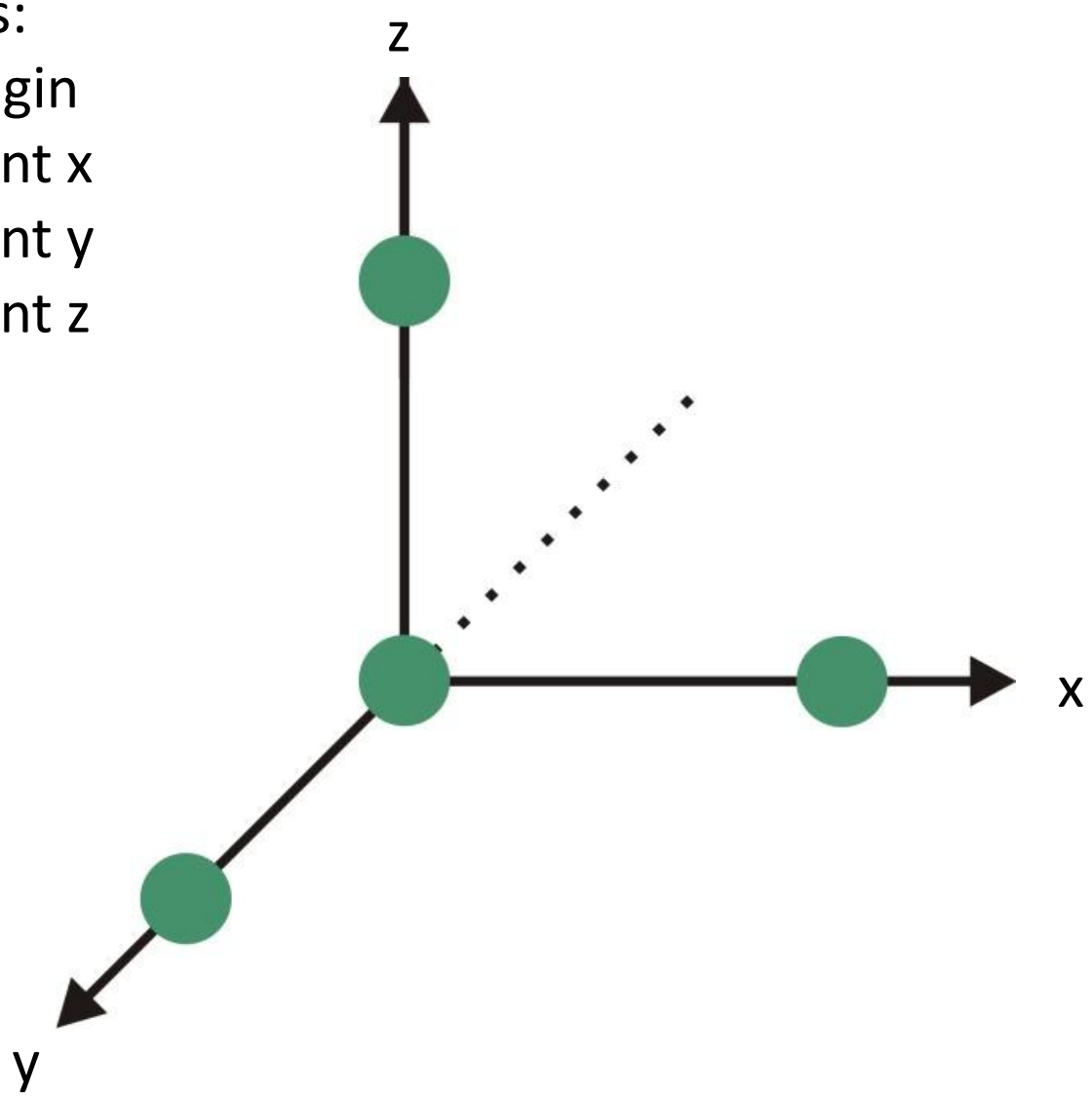
Create a simple grid in which each cell can have 4 atoms:

1 on origin

+ amount x

+ amount y

+ amount z

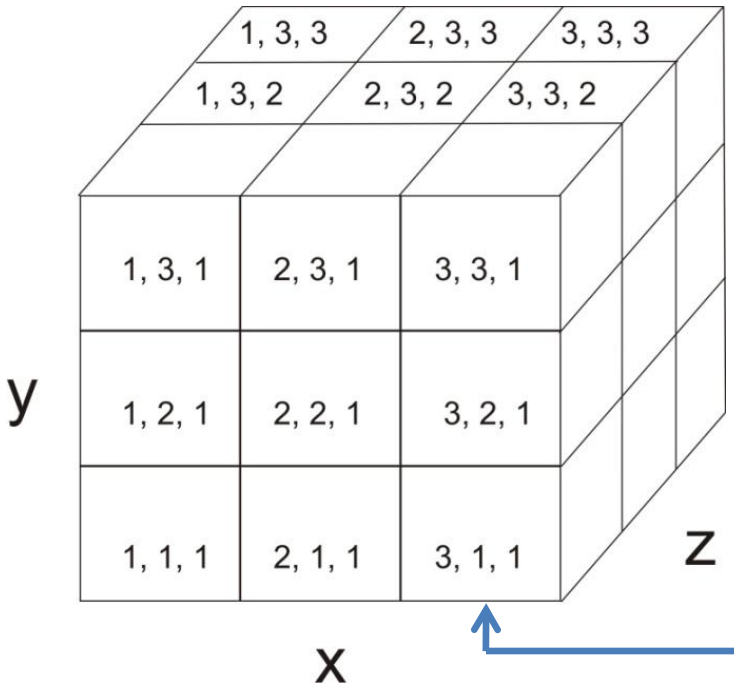


Creating a grid

Place 4 atoms in central cell (PNUM to PNUM+3)
Replicate central cell
Finding positions for 3*3*3*4=108 atoms

PNUM = Particle Number
X(PNUM) = x position of particle #

```
CELLS = 6.D0
CELLSS = 2.D0*CELLS
CELL = 0
DO XX = 1,3
  DO YY = 1,3
    DO ZZ = 1,3
      CELL = CELL + 1
      PNUM  = 4*(CELL-1) + 1
      X(PNUM) = DBLE(XX-1)*(1.D0/CELLS)
      Y(PNUM) = DBLE(YY-1)*(1.D0/CELLS)
      Z(PNUM) = DBLE(ZZ-1)*(1.D0/CELLS)
      X(PNUM+1) = (XX-1)*(1.D0/CELLS)
      Y(PNUM+1) = (1.D0/CELLSS)+DBLE(YY-1)*(1.D0/CELLS)
      Z(PNUM+1) = (1.D0/CELLSS)+DBLE(ZZ-1)*(1.D0/CELLS)
      X(PNUM+2) = (1.D0/CELLSS)+DBLE(XX-1)*(1.D0/CELLS)
      Y(PNUM+2) = (YY-1)*(1.D0/CELLS)
       Z(PNUM+2) = (1.D0/CELLSS) + DBLE(ZZ-1)*(1.D0/CELLS)
       X(PNUM+3) = (1.D0/CELLSS) + DBLE(XX-1)*(1.D0/CELLS)
       Y(PNUM+3) = (1.D0/CELLSS) + DBLE(YY-1)*(1.D0/CELLS)
       Z(PNUM+3) = (ZZ-1)*(1.D0/CELLS)
    ENDDO
  ENDDO
ENDDO
```



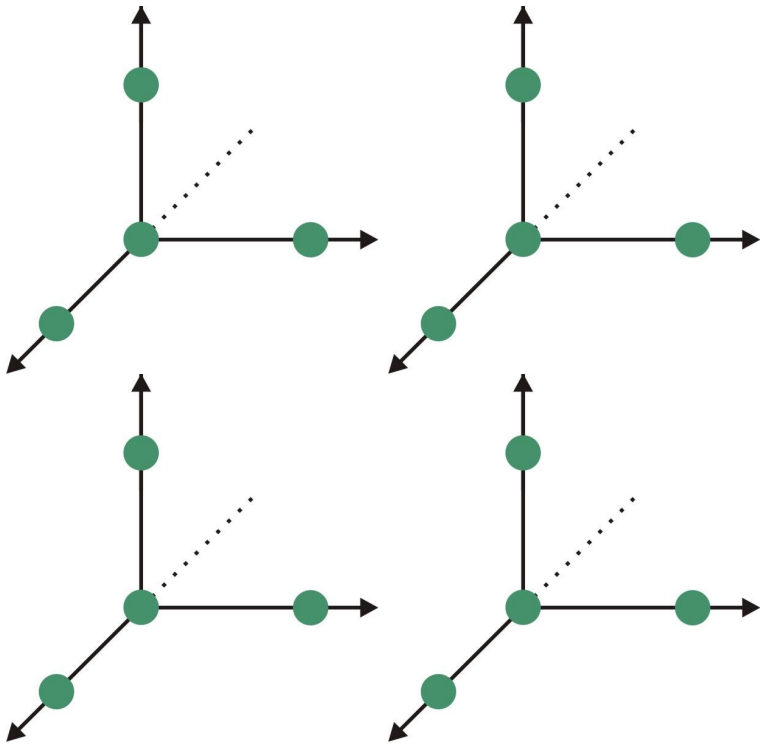
Transforming the Grid Cell to Simulation Box with the proper density

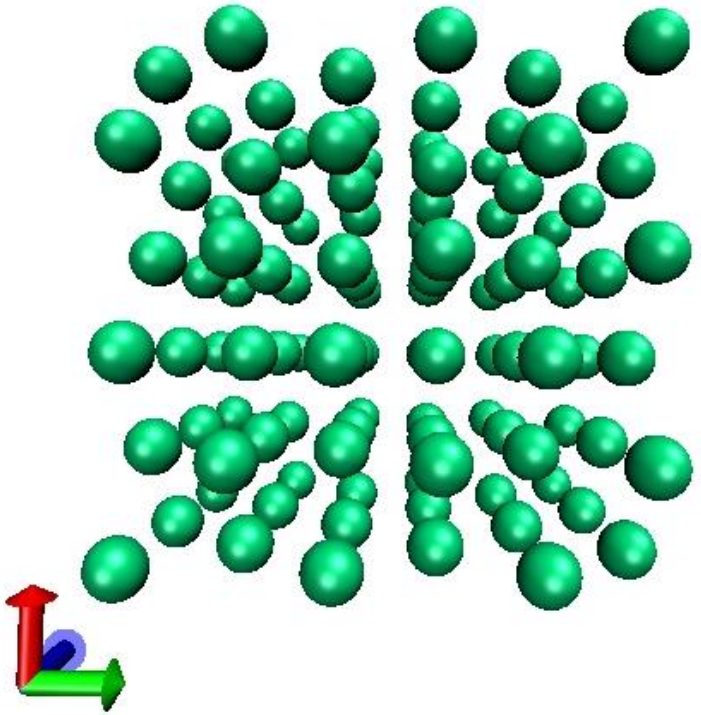
$$Box\ Length = (n_{particles} / density^{*})^{1/3}$$

From the grid points of the initial cells

$x_{i,\ new} = x_{i,\ grid} *$ box length

```
DO I = 1, NPART
      X(I) = X(I) * BOXL(1)
      Y(I) = Y(I) * BOXL(2)
      Z(I) = Z(I) * BOXL(3)
ENDDO
```



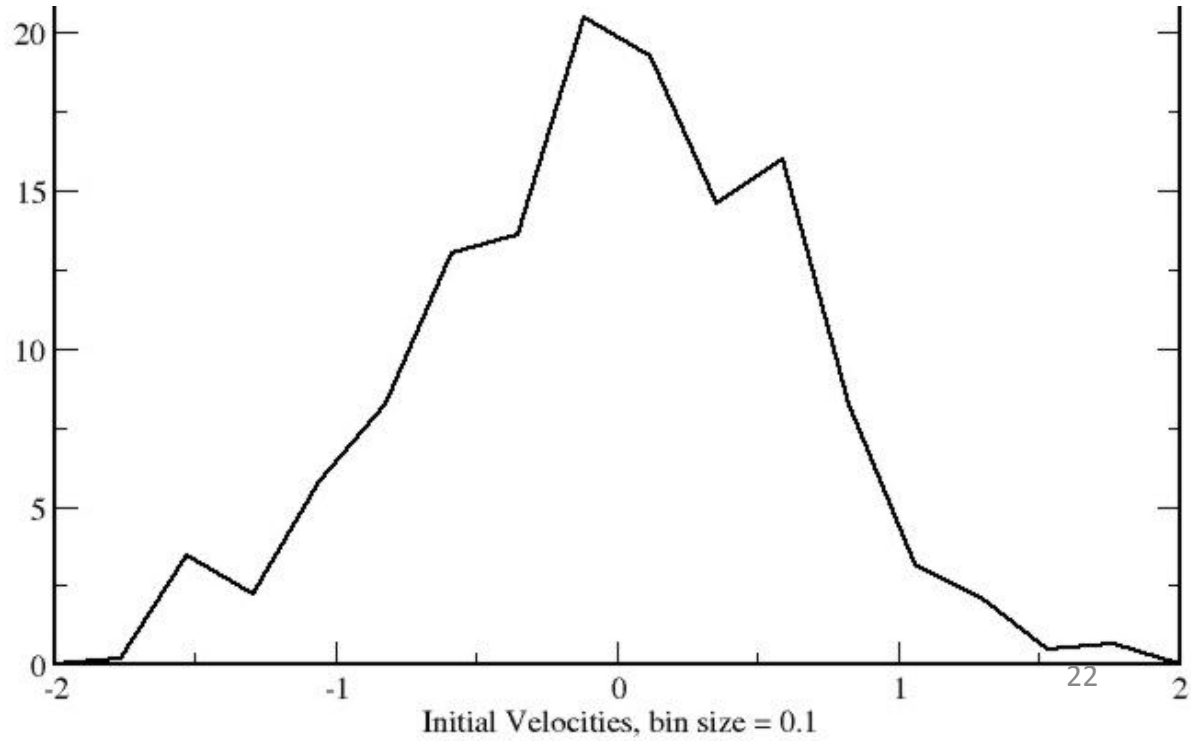


Initialization

- From the gridded atoms, need to know their initial velocities
  - Choose random velocities conforming to the required temperature
  - Ensuring overall momentum, P, is zero

$$P = \sum_{i=1}^{N} m_i v_i = 0$$

For 108 particles;
324 random initial velocities



Temperature Control

```
SUBROUTINE RESCALE(VX,VY,VZ,TEMP,NPART)
    INTEGER NPART,I
    DOUBLE PRECISION VX(NPART),VY(NPART),VZ(NPART),TEMP,SCALEF,V2T
    V2T = 0.D0
    DO I = 1,NPART
        V2T = V2T + VX(I)*VX(I) + VY(I)*VY(I) + VZ(I)*VZ(I)
    ENDDO
    V2T = V2T / DBLE(NPART)
    SCALEF = DSQRT(3.0D0*TEMP / V2T)
    DO I = 1,NPART
        VX(I) = VX(I)*SCALEF
        VY(I) = VY(I)*SCALEF
        VZ(I) = VZ(I)*SCALEF
    ENDDO
    RETURN
    END
```

Declaration of variables

Kinetic Energy

Mean-squared velocity

Velocity Scale Factor

Initial temperature is not terribly important
-> Useful to start at a higher temperature and anneal to one's target temperature

Temperature will change during a simulation

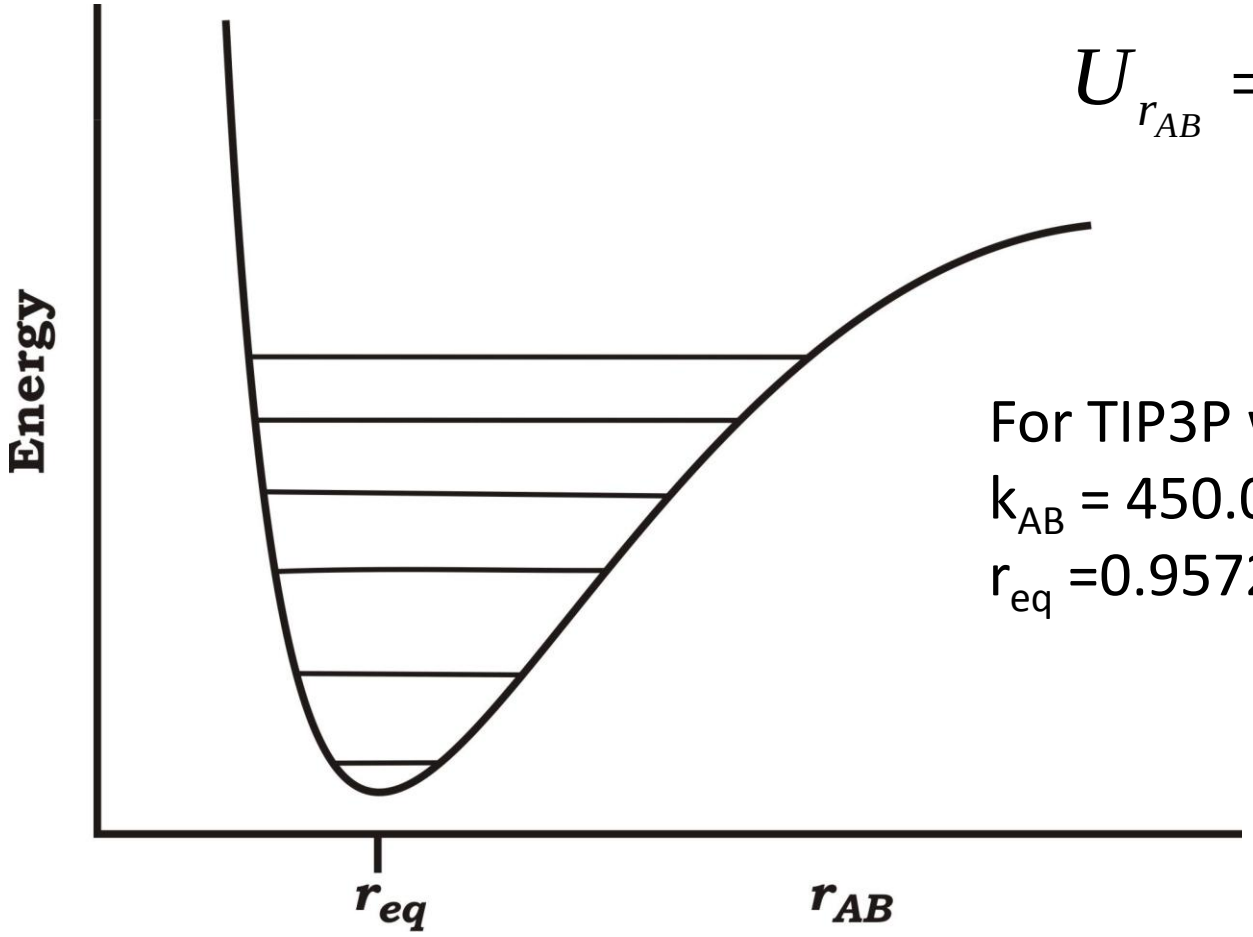
```
IF (MOD (ISTEP,500) THEN
    CALL RESCALE (…)
ENDIF
```

3. Potential Energy Evaluation

Determining the particle forces

- Force field: the functional form and parameter set used to describe the potential energy of a system of particles
  - Typical functional forms may include the following functions
    - non-bonded Lennard Jones interactions, Coulomb, torsion, bending, sometimes polarization
  - Parameter set or calculated constants
    - Force constants, ε, σ
    - Charmm, Amber, TIP3P, SPC/E

Example:
Potential Energy Functional



$$U_{r_{AB}} = \frac{1}{2} k_{AB} (r_{AB} - r_{eq})^2$$

For TIP3P water, O-H potential

k_{AB} = 450.0 kcal/ mol* Å^2

r_{eq} =0.9572 Å

Lennard Jones Potential

- Lennard Jones system in reduced units
- Pair-wise potential to calculate coordinate force directions & contribution to Potential Energy

$$f_x(r) = -\frac{\partial u(r)}{\partial x} = -\left(\frac{x}{r}\right)\left(\frac{\partial u(r)}{\partial r}\right)$$

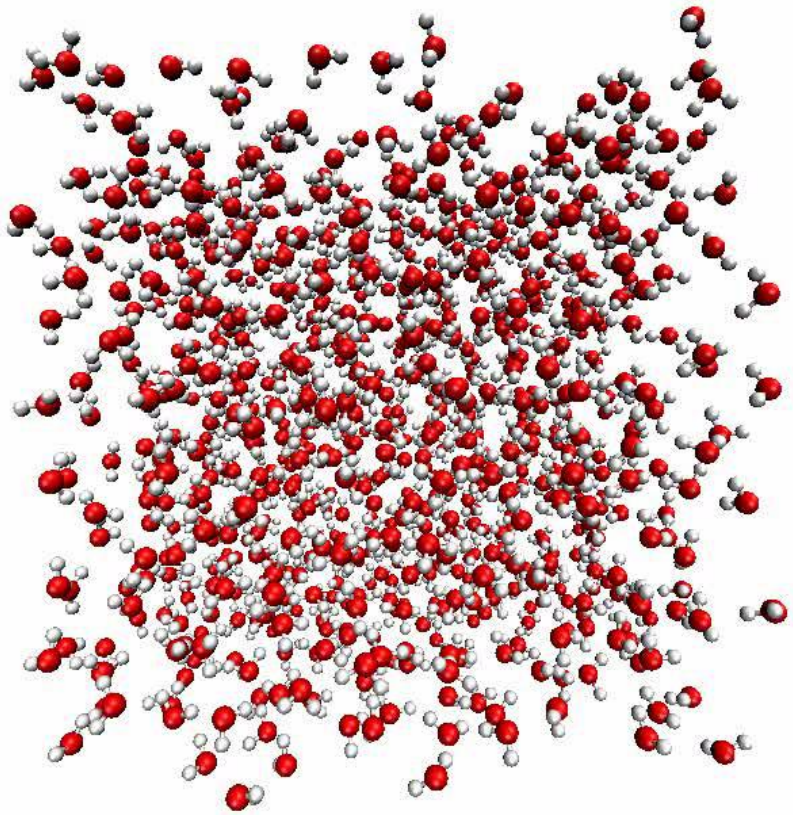
$$f_x(r) = \frac{48}{r^2}\left(\frac{1}{r^{12}} - 0.5\frac{1}{r^6}\right)$$

- How to implement this

Recapping

We have a box with 108 particles

For Classical Mechanical simulations,
typical sizes: 1,000 - 500,000 molecules
For a box containing 1,000 waters, ~480 waters are located on the surface
This is not a liquid… what to do?



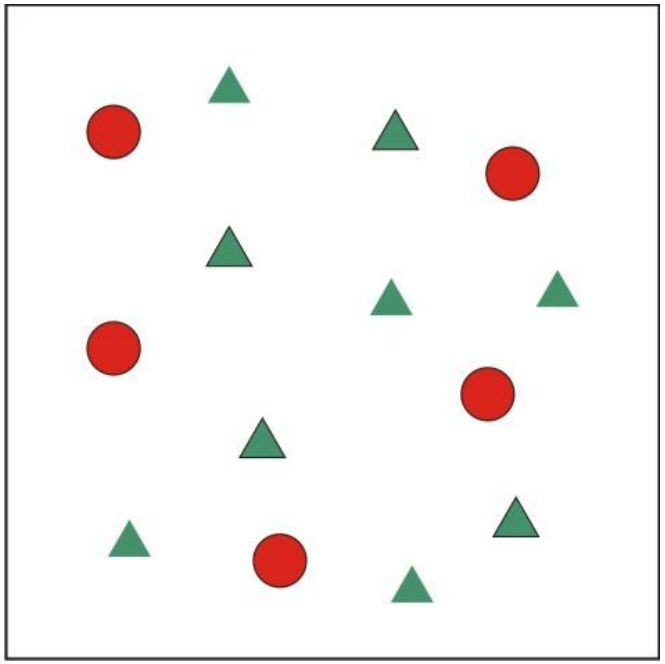
Unit Analysis Question

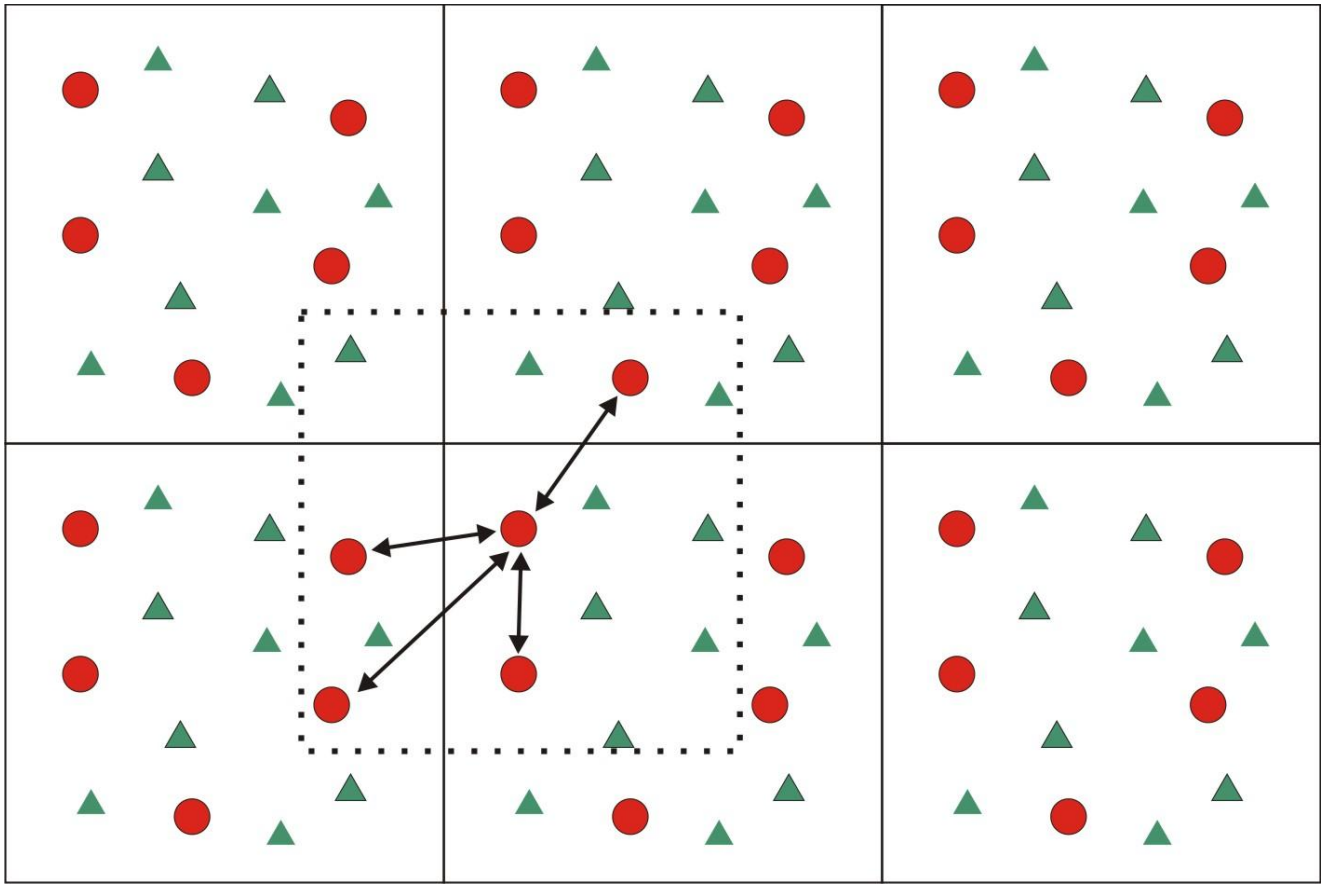
You want to simulate 1,000 waters with a density = 0.99777 g/cm^3.
What would the box length be?

~31 Å

Periodic Boundary Conditions

- Instead of modeling a large system, model a small part far from an edge
  - When an atom passes through one face, it enters the central simulation cell on the opposite face



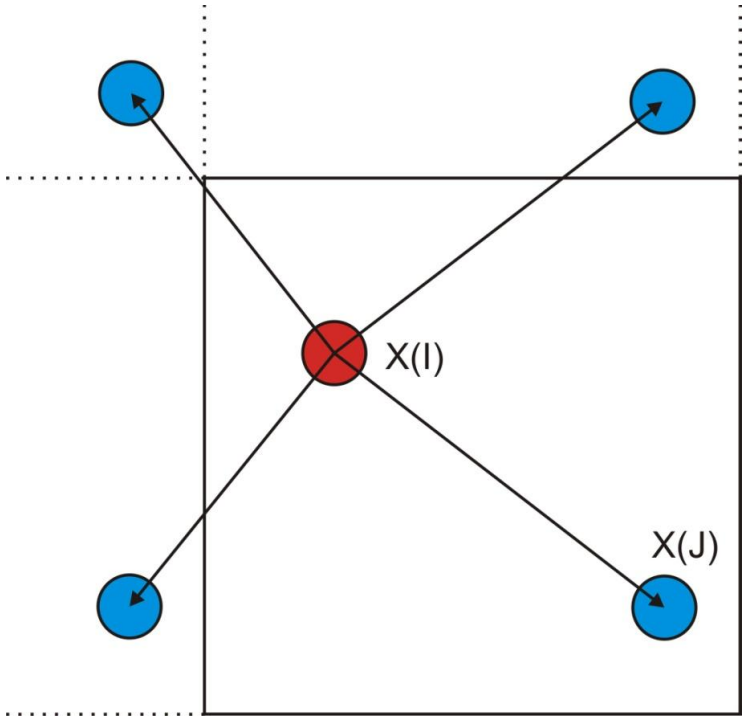


A common PBC technique is the *minimum image convention*: individual particles interact with the closest image of the remaining particles

Need to locate whether which pair is closest – interacting with a periodic image or of the particle in the central cell

Finding the nearest i-j neighbor

- Programming function, NINT
- Nearest INTeger
  - NINT(2.784) = 3
  - NINT(2.135) = 2



Difference between 2 particle's x positions
RX = X(I) – X(J)

Box Length, BOXL(1) = 14

RX = RX - NINT(RX/BOXL(1)) * BOXL(1)

RX	RX new
-11	3
20	6
35	-7

Calculating Lennard Jones Interactions

```
DO I=1,NPART-1
    DO J=I+1,NPART
      RX = X(I) - X(J)
      RX = RX - NINT(RX/BOXL(1))*BOXL(1)

      R2 = 1.0D0 / (RX*RX + RY*RY + RZ*RZ)
      R6 = R2 * R2 * R2
      F = 48*R2*R6*(R6 - 0.5D0)
      FX(I) = FX(I) + RX*F
      FX(J) = FX(J) - RX*F
      PENER = PENER + 4*R6*(R6 - 1.D0)
    ENDDO
  ENDDO
```

$$f_x(r) = \frac{48}{r^2}\left(\frac{1}{r^{12}} - 0.5\frac{1}{r^6}\right)$$

Constants can be placed outside of the loop to save time

Summing x-direction forces on particle I and J

<u>Parallel Molecular Dynamics Programming</u>

Threads/ processes can be:

Grouping of atoms

Neighbor Lists

Sub-volume grouping

4. Integration

Verlet Algorithm

$$x(t+\Delta t) = 2x(t) - x(t-\Delta t) + a(t)\Delta t^2$$

Twice the amount where you are

Minus where you have been

Plus the force times double the time step

Calculate and update new positions and velocities

```
DO I=1,NPART
   XN = 2*X(I) - XO(I) + FX(I)*TSTEP*TSTEP
   VX(I) = ( XN - XO(I) ) / (2*TSTEP)
```

Solve for next position
Calculate new velocity

```
   IF (XN .GT. BOXL(1)) THEN
      XN = XN - BOXL(1)
   ENDIF
   IF (XN .LT. 0.D0) THEN
      XN = XN + BOXL(1)
   ENDIF
```

Enforce PBC
If particle leaves the box, insert on the opposite side

```
   XO(I) = X(I)
   X(I) = XN
ENDDO
```

Update time step information

```
WRITE(77,*)NPART
WRITE(77,*)
DO I=1,NPART
    WRITE(77,*)"H", X(I),Y(I),Z(I)
ENDDO
```

For the trajectory save atomic positions at this time step

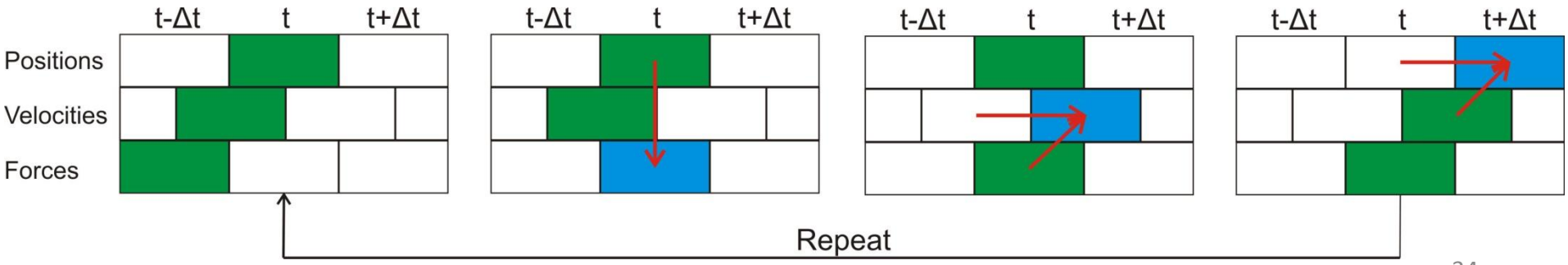
Will write to file fort.77

Alternative to Verlet
Leap-frog Time Integration

Velocities calculated every half time step
Positions calculated every time step
Velocities and positions *"leap"* over each other
*Velocities are explicitly calculated
*Larger time steps can be used

$$v(t+\tfrac{1}{2}\Delta t) = v(t-\tfrac{1}{2}\Delta t) + a(t)\Delta t$$

$$x(t+\Delta t) = x(t) + v(t+\tfrac{1}{2}\Delta t)\Delta t$$



Equilibration to Production

- Equilibration
- The initial grid system needs to be equilibrated to the target system of interest
  - Solid lattice to liquid phase

- Determine when it is equilibrated and the model is ready for Production (data collecting)
  - Instantaneous Potential energy INCREASES
  - Instantaneous Kinetic energy DECREASES
  - Oscillates around steady mean values

- Ergodic hypothesis: over a long period of time the statistical averages are equal to the time averages of the system

Visualization

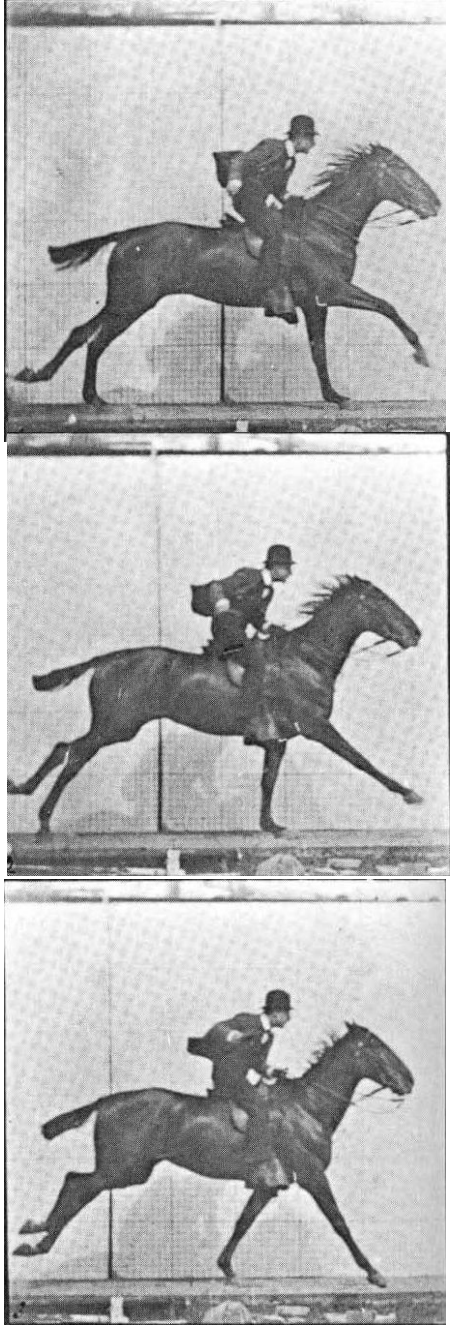
xyz, pdb, psf, dcd

VMD, pymol, rasmol

Number of atoms
<blank/title line>
Element x y z
Element x y z
Element x y z

3			
O	4.2140	-1.6693	9.5508
H	4.1355	-2.5202	9.1195
H	3.5973	-1.7838	10.3204

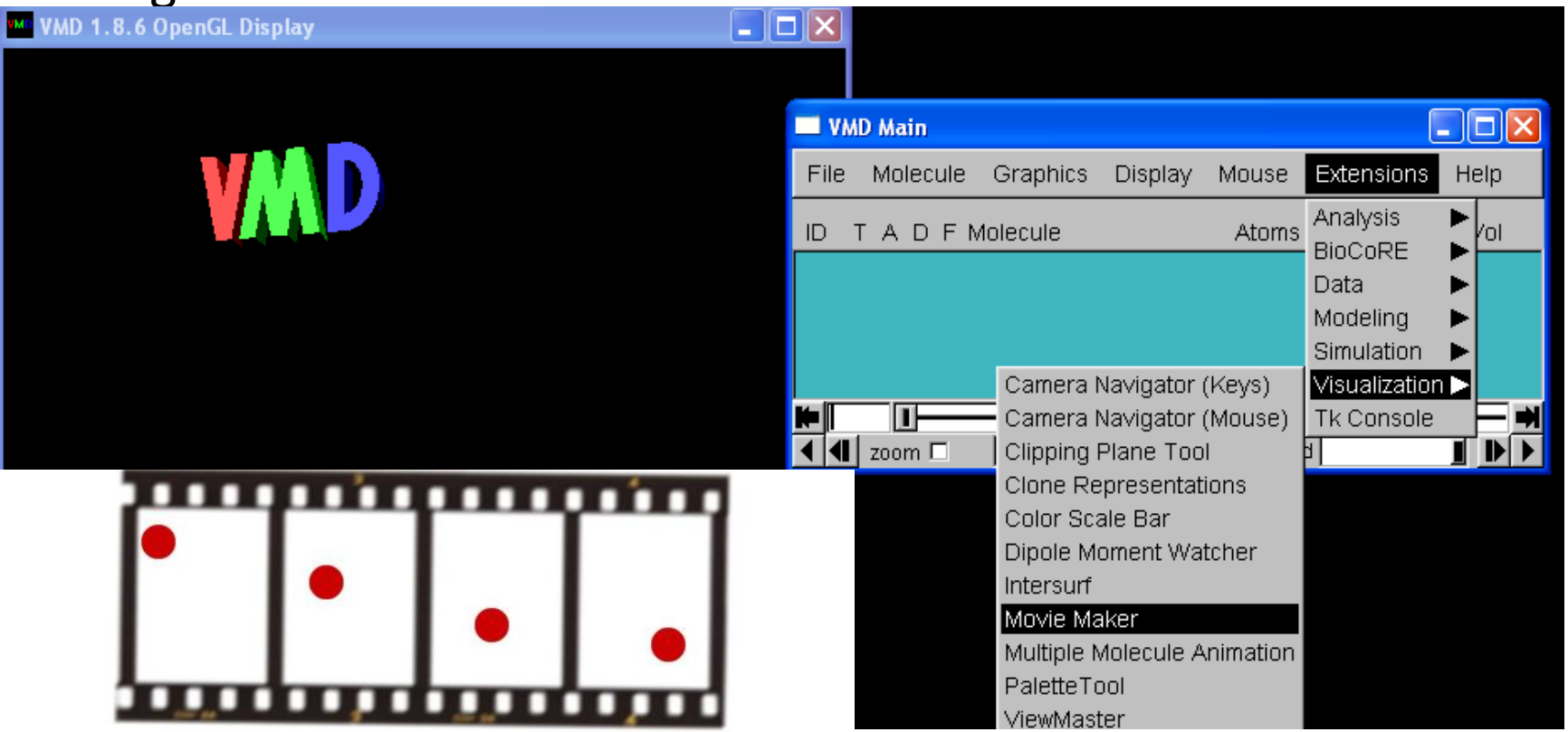
3			
O	4.2140	-1.6693	9.5508
H	4.1355	-2.5202	9.1195
H	3.5973	-1.7838	10.3204
3			
O	4.2091	-1.6647	9.5463
H	4.1219	-2.5004	9.1348
H	3.6245	-1.7658	10.3104
3			
O	4.2048	-1.6562	9.5430
H	4.1132	-2.5120	9.1307
H	3.6479	-1.7460	10.3110
3			
O	4.2016	-1.6455	9.5386
H	4.1103	-2.5412	9.1153
H	3.6601	-1.7267	10.3309

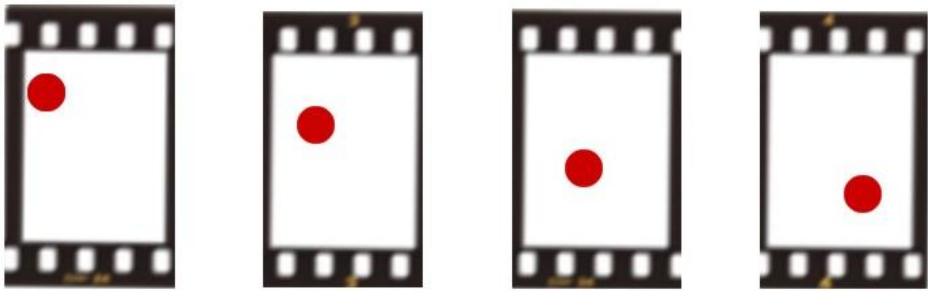


Muybridge's *The Horse in Motion*

.XYZ Trajectory to .MPEG

Using VMD and VideoMach



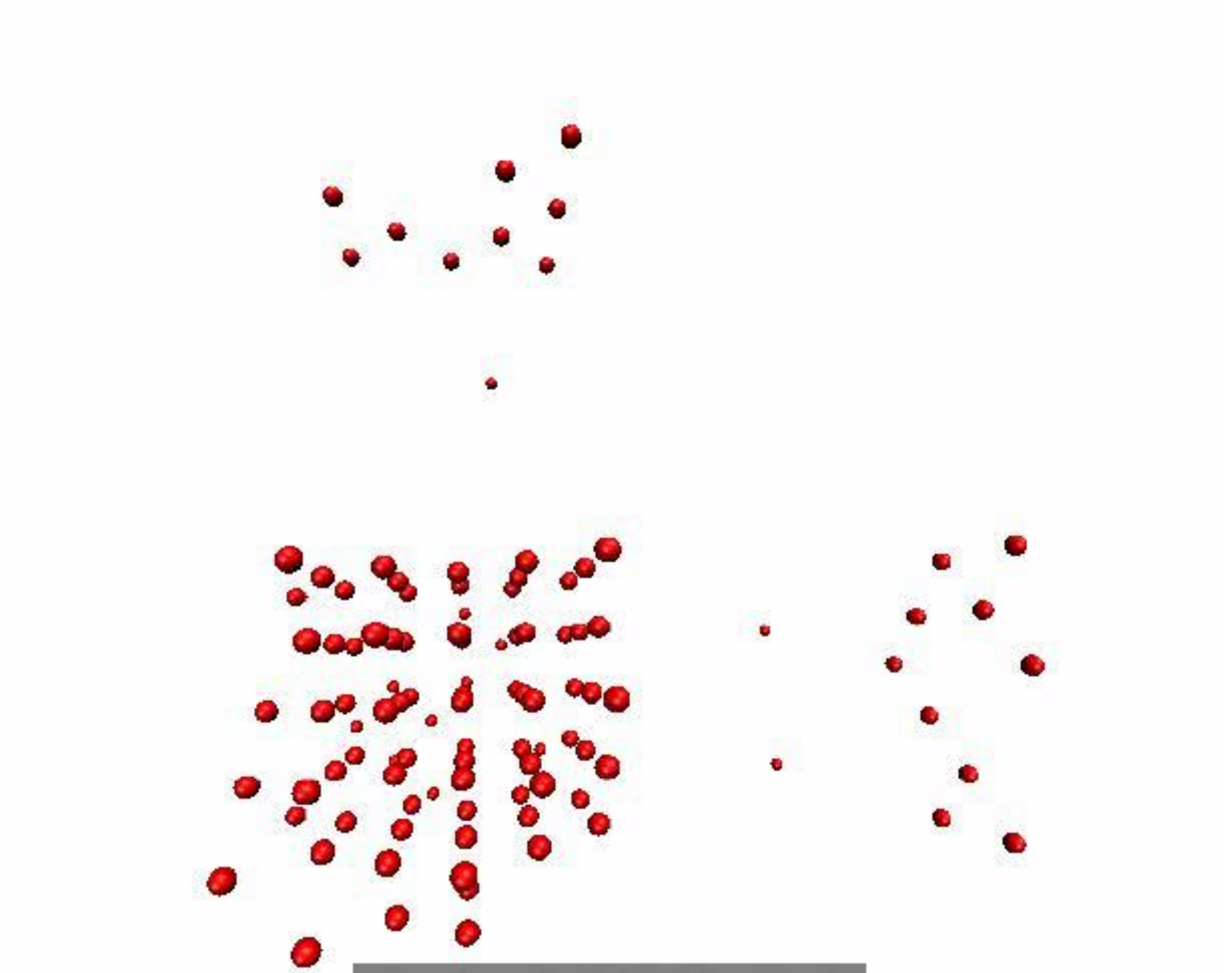


VMD- Movie Maker: Each time step/frame is rendered and saved to a bmp/pov-ray file

VideoMach: Reconnects and transforms these pictures into a movie file (mpeg/mov)

VideoMach http://gromada.com/main/products.php

VMD http://www.ks.uiuc.edu/Research/vmd/


Produced with VideoMach
www.videomach.com

To better ourselves and our HPC center
Please fill out the survey

http://www.surveymonkey.com/s/ZJBSDVH