

5.1: Review of Lecture 4

Newtonian equations of motion

N atoms $i = 1, \dots, N$ with mass m_i , Cartesian coordinates \mathbf{r}_i and potential

$$\mathcal{V} = \mathcal{V}(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) \equiv \mathcal{V}(\mathbf{r}^N)$$

Dynamics can be found by solving Newton's equations of motion

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i$$

with the force on particle i obtained from the potential as

$$\mathbf{f}_i = -\frac{\partial \mathcal{V}}{\partial \mathbf{r}_i}$$

or, defining linear momentum of particle i as

$$\mathbf{p}_i = m_i \dot{\mathbf{r}}_i$$

Newton's equation of motion is written as

$$\dot{\mathbf{p}}_i = \mathbf{f}_i$$

5.1: Review: Pair potentials

The total potential in "simple" liquids can be resolved in a sum of pair-potentials potentials $v(r_{ij})$, depending only on interparticle distance r_{ij}

$$\mathcal{V}(\mathbf{r}^N) = \sum_{i=1}^N \sum_{j=i+1}^N v(r_{ij})$$

Atom pairs i, j are only counted once ($j > i$). Using $v(r_{ij}) = v(r_{ji})$ and suppressing lower limits gives:

$$\mathcal{V}(\mathbf{r}^N) = \frac{1}{2} \sum_i^N \sum_{j \neq i}^N v(r_{ij})$$

with the condition $j \neq i$ stated below summation sign. In other words you count the energy of each "bond" twice, and then divide by 2 to compensate.

The forces from pair potentials are also a superposition of pair forces i.e. the force on particle i is:

$$\mathbf{f}_i = -\frac{\partial \mathcal{V}}{\partial \mathbf{r}_i} = -\frac{1}{2} \sum_{j \neq i}^N \left(\frac{\partial v(r_{ij})}{\partial \mathbf{r}_i} + \frac{\partial v(r_{ji})}{\partial \mathbf{r}_i} \right) = -\sum_{j \neq i}^N \frac{\partial v(r_{ij})}{\partial \mathbf{r}_i} = \sum_{j \neq i}^N \mathbf{f}_{ij}$$

5.1: Review: Time evolution in molecular dynamics

First you discretize the trajectory in time: timestep δt “small”

$$\dots, \mathbf{r}^N(t_{m-1}) = \mathbf{r}^N(t_m - \delta t), \mathbf{r}^N(t_m), \mathbf{r}^N(t_{m+1}) = \mathbf{r}^N(t_m + \delta t), \dots$$

At time t_m each of the N particles i has a position $\mathbf{r}_i(t_m)$ and a velocity $\mathbf{v}_i(t_m)$. MD gives you a recipe to move those positions and velocities to their new values at time step $t + \delta t$.

1. First calculate the force vectors on each of the N particles:

$$\mathbf{f}_i = \sum_{j \neq i}^N \mathbf{f}_{ij}$$

In principle this is an $N(N - 1)$ operation (expensive).

2. Then update the $(3N)$ positions and $(3N)$ velocities with the **Verlet algorithm**:

$$\mathbf{r}_i(t + \delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \delta t) + \frac{\delta t^2}{m_i} \mathbf{f}_i(t) + O(\delta t^4)$$

$$\mathbf{v}_i(t) = \frac{1}{2\delta t} [\mathbf{r}_i(t + \delta t) - \mathbf{r}_i(t - \delta t)] + O(\delta t^3)$$

5.1: Review: Verlet algorithm derivation

Again start with Taylor expansion from t to $t + \delta t$

$$\mathbf{r}_i(t + \delta t) = \mathbf{r}_i(t) + \delta t \mathbf{v}_i(t) + \frac{\delta t^2}{2m_i} \mathbf{f}_i(t) + \frac{\delta t^3}{6} \mathbf{b}_i(t) + O(\delta t^4)$$

Taylor expansion back in time from t to $t - \delta t$

$$\mathbf{r}_i(t - \delta t) = \mathbf{r}_i(t) - \delta t \mathbf{v}_i(t) + \frac{\delta t^2}{2m_i} \mathbf{f}_i(t) - \frac{\delta t^3}{6} \mathbf{b}_i(t) + O(\delta t^4)$$

Adding gives advanced positions

$$\mathbf{r}_i(t + \delta t) = 2\mathbf{r}_i(t) - \mathbf{r}_i(t - \delta t) + \frac{\delta t^2}{m_i} \mathbf{f}_i(t) + O(\delta t^4)$$

Subtracting gives current velocities

$$\mathbf{v}_i(t) = \frac{1}{2\delta t} [\mathbf{r}_i(t + \delta t) - \mathbf{r}_i(t - \delta t)] + O(\delta t^3)$$

Comparing to Euler algorithm

- We have gained one order in time (at no cost).
- Algorithm is explicitly time reversible.
- Velocity update one step behind position update but **not** needed.

5.2: Coding it all up

Force routine for harmonic potential

```
subroutine force (k, natom, x, y, z, fx, fy, fz, epot)
  epot := 0
  do i := 1, natom
    fx(i) := -k * x(i)
    fy(i) := -k * y(i)
    fz(i) := -k * z(i)
    epot := epot + k * (x(i) ** 2 + y(i) ** 2 + z(i) ** 2) / 2
  enddo
end
```

$x(i), y(i), z(i)$	position particle i at time t (array, $i = 1, \dots, natom$)
$fx(i), fy(i), fz(i)$	force on particle i at time t (array, $i = 1, \dots, natom$)
$epot$	accumulator potential energy
k	spring constant

Basic Verlet subroutine

```
subroutine verlet (natom, dt, ms, fx, fy, fz, x, y, z, xm, ym, zm, ekin)
ekin = 0
do i := 1, natom
  xp := 2 * x(i) - xm(i) + dt ** 2 * fx(i) / ms(i)
  yp := 2 * y(i) - ym(i) + dt ** 2 * fy(i) / ms(i)
  zp := 2 * z(i) - zm(i) + dt ** 2 * fz(i) / ms(i)
  ekin = ekin + ms(i) * ((xp - xm(i)) ** 2 + (yp - ym(i)) ** 2 + (zp - zm(i)) ** 2)
  xm(i) := x(i)
  ym(i) := y(i)
  zm(i) := z(i)
  x(i) := xp
  y(i) := yp
  z(i) := zp
enddo
ekin = ekin / (8 * dt ** 2)
end
```

<i>dt</i>	timestep
<i>ms(i)</i>	mass of particle <i>i</i> (array, $i = 1, \dots, \text{natom}$)
<i>x(i), y(i), z(i)</i>	position particle <i>i</i> at time <i>t</i> (array, $i = 1, \dots, \text{natom}$)
<i>fx(i), fy(i), fz(i)</i>	force on particle <i>i</i> at time <i>t</i> (array, $i = 1, \dots, \text{natom}$)
<i>xm(i), ym(i), zm(i)</i>	position particle <i>i</i> at time $t - \delta t$ (array, $i = 1, \dots, \text{natom}$)
<i>xp, yp, zp</i>	temporary storage for position particle <i>i</i> at time $t + \delta t$
<i>ekin</i>	accumulator kinetic energy

Basic MD program in modular form (3D harmonic oscillator)

```
program mdharmonic
read natom, l, k
read nstep, dt
read v0
call initr (natom, l, x, y, z, )
call initv (natom, dt, v0, x, xm)
call initv (natom, dt, 0, y, ym)
call initv (natom, dt, 0, z, zm)
do mstep := 1, nstep
  call force (k, natom, x, y, z, fx, fy, fz, epot)
  call verlet (natom, dt, ms, fx, fy, fz, x, y, z, xm, ym, zm, ekin)
  etot = ekin + epot
  print mstep, ekin, epot, etot
enddo
end
```

<i>natom</i>	number of atoms
<i>mstep</i>	counter for time steps
<i>nstep</i>	total number of time steps
<i>etot</i>	total energy
<i>l</i>	length for setting up initial configuration
<i>v0</i>	initial velocity

5.3: Sampling in molecular dynamics

In MD simulation time has been discretized in steps of duration δt

$$0, t_1, \dots, t_m, \dots, t_M = \Delta t \quad \text{with} \quad t_m = m\delta t$$

where M the total number of steps and

$$\Delta t = M\delta t$$

is the duration of the “run”. This gives for property A a time series

$$A(t_m) = \mathcal{A}(\mathbf{r}^N(t_m), \mathbf{p}^N(t_m)) \quad , m = 1, \dots, M$$

Time averages over are thus estimated by the averages over time steps

$$\bar{A}_{\Delta t} \approx \frac{1}{M} \sum_m^M A(t_m)$$

Time averages are the link to equilibrium statistical mechanics

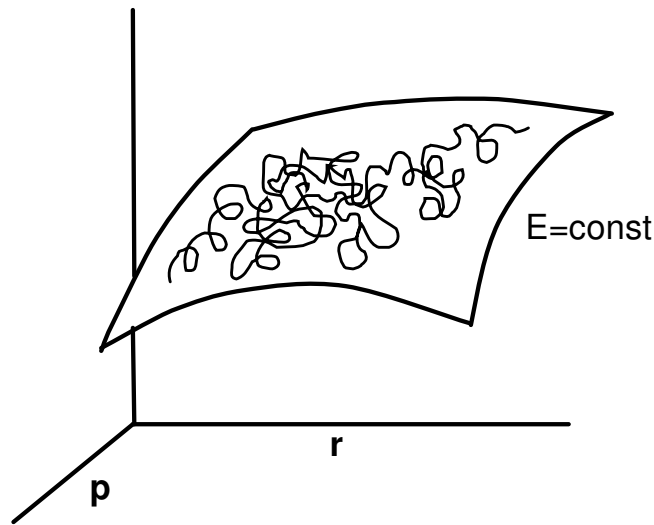
5.4: *Phase space and ergodic hypothesis

- **Phase space** is set of all position momentum state points $(\mathbf{r}^N, \mathbf{p}^N)$
- The hypersurface in phase space at constant energy E is specified by

$$\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = E$$

where \mathcal{H} is the energy phase function called the **Hamiltonian**

$$\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = \sum_i^N \frac{\mathbf{p}_i^2}{2m_i} + \mathcal{V}(\mathbf{r}^N)$$



Dimension phase space is $6N$

Dimension of an energy hypersurface is $6N - 1$

The trajectory in phase space of an interacting many atom system

- is constrained to an constant energy hypersurface in phase space (*energy conservation, rigorous*)
- fills the entire constant energy surface given enough time (*ergodicity, assuming "molecular chaos"*)

5.5: *Microcanonical ensemble (energy fixed)

Ergodic assumption:

Infinite time averages can be replaced by average over equi-energy surface

$$\lim_{\Delta t \rightarrow \infty} \bar{A}_{\Delta t} = \lim_{\Delta t \rightarrow \infty} \frac{1}{\Delta t} \int_0^{\Delta t} dt \mathcal{A}(\mathbf{r}^N(t), \mathbf{p}^N(t)) = \int_{E=\text{cons}} d\mathbf{r}^N d\mathbf{p}^N \mathcal{A}(\mathbf{r}^N, \mathbf{p}^N)$$

or written in terms of an average over phase space distribution $\rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N)$

$$\lim_{\Delta t \rightarrow \infty} \bar{A}_{\Delta t} = \int d\mathbf{r}^N d\mathbf{p}^N \rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N) \mathcal{A}(\mathbf{r}^N, \mathbf{p}^N) \equiv \langle A \rangle_{NVE}$$

$$\rho_{NVE}(\mathbf{r}^N, \mathbf{p}^N) = \frac{f(N)}{\Omega_N} \delta[\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) - E]$$

with normalization factor

$$\Omega_N(V, E) = f(N) \int d\mathbf{r}^N d\mathbf{p}^N \delta[\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) - E]$$

$f(N)$ corrects for indistinguishable particles (*cancels out in averages*)

ρ_{NVE} is called the microcanonical ensemble distribution function

5.6: *Canonical ensemble (temperature fixed)

Under exchange energy with a heat bath states are distributed according to

$$\rho_{NVT}(\mathbf{r}^N, \mathbf{p}^N) = \frac{f(N)}{Q_N} \exp \left[-\frac{\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)}{k_B T} \right]$$
$$Q_N(V, T) = f(N) \int d\mathbf{r}^N d\mathbf{p}^N \exp \left[-\frac{\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)}{k_B T} \right]$$

Canonical expectation values are exponentially weighted averages over all points in phase space

$$\begin{aligned} \langle A \rangle_{NVT} &= \int d\mathbf{r}^N d\mathbf{p}^N \rho_{NVT}(\mathbf{r}^N, \mathbf{p}^N) \mathcal{A}(\mathbf{r}^N, \mathbf{p}^N) \\ &= \frac{f(N)}{Q_N} \int d\mathbf{r}^N d\mathbf{p}^N \mathcal{A}(\mathbf{r}^N, \mathbf{p}^N) \exp[-\beta \mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)] \end{aligned}$$

with $\beta = 1/k_B T$. Warning: Newtons equations conserve energy hence ideally

*Trajectories generated by Verlet algorithm are microcanonical, **not** canonical*

5.7: Equilibrium and time averages in MD (in practice)

All quantities in MD (except total energy) fluctuate in time.

$$A(t) = \mathcal{A}(\mathbf{r}^N(t), \mathbf{p}^N(t))$$

*For systems in **equilibrium** the motion is **stationary**:
the system fluctuates in time but there is no real change*

Under these conditions of dynamical equilibrium averages over a period Δt

$$\bar{A}_{\Delta t} = \frac{1}{\Delta t} \int_0^{\Delta t} dt A(t) \approx \frac{1}{M} \sum_m^M A(t_m)$$

approach for “sufficiently” long Δt the microcanonical expectation value

$$\lim_{\Delta t \rightarrow \infty} \bar{A}_{\Delta t} = \langle A \rangle_{NVE}$$

where E is the (constant) energy of the trajectory.

Since run times in MD are finite, observe the following **sampling rules**

- Start sampling only when equilibrium (stationary) conditions have been reached \Rightarrow Sampling must be preceded by an **equilibration run**.
- The length of the **data collection run** (sampling period Δt) must be considerably longer than the characteristic time scale of the fluctuations.

5.8: Temperature in MD

In the canonical ensemble the average of kinetic energy is

$$\left\langle \sum_{i=1}^N \frac{\mathbf{p}_i^2}{2m_i} \right\rangle_{NVT} = \frac{3}{2} N k_B T$$

with the Maxwell-Boltzmann distribution for particle velocity

$$P(v_{x,i}) = \sqrt{\frac{m_i}{2\pi k_B T}} \exp\left[-\frac{m_i v_{x,i}^2}{2k_B T}\right]$$

This suggests to introduce an **instantaneous** temperature

$$\mathcal{T}(\mathbf{p}^N) \equiv \frac{1}{3k_B N} \sum_{i=1}^N m_i \mathbf{v}_i^2$$

\mathcal{T} fluctuates in time, however, when averaged over the MD trajectory

$$T_N = \frac{1}{M} \sum_{m=1}^M \mathcal{T}(t_m)$$

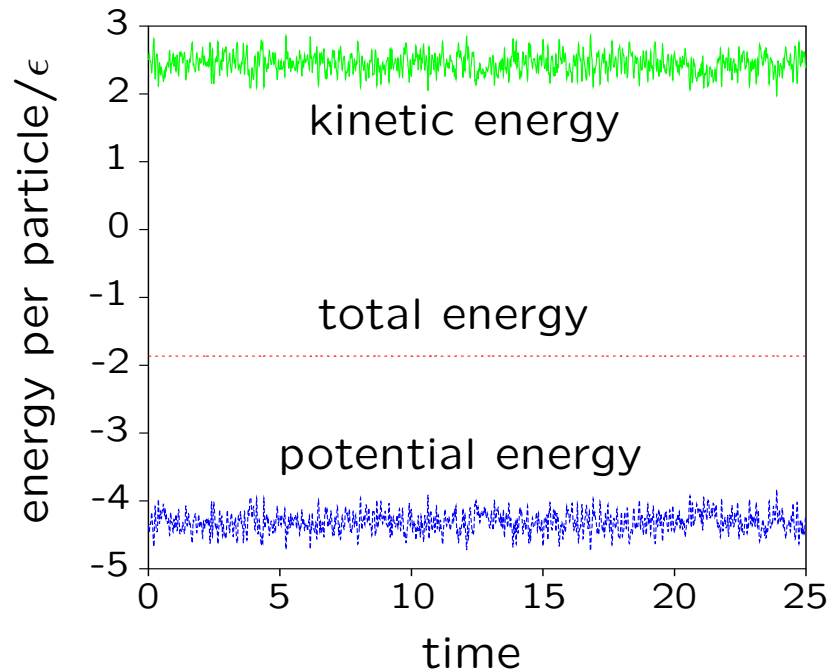
gives the temperature of the corresponding microcanonical N particle system.

T_N depends on N and M and (conserved) total energy, but for large N and M converges to the thermodynamic temperature T at energy E .

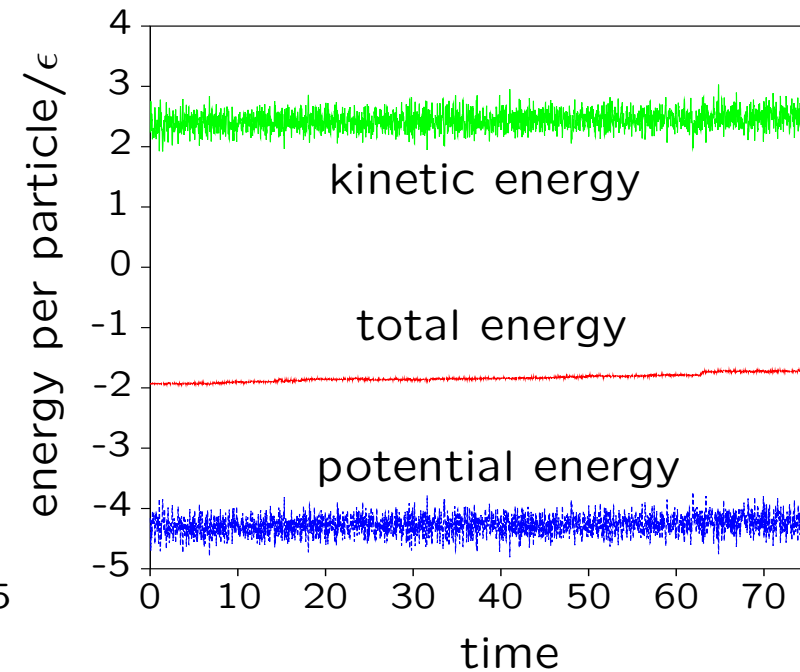
5.9: Energy conservation in Verlet

Lennard-Jones fluid of 108 atoms in cubic box

Energy fluctuations in time



Time step optimized
 $\delta t^* = 0.005$, 5000 steps
Total energy conserved



Time step too **long**
 $\delta t^* = 0.015$, 5000 steps
Small drift in total energy

Time t^* is in reduced units

$$t^* = t \left(\frac{\epsilon}{m\sigma^2} \right)^{1/2}$$

5.10: Temperature control by velocity scaling

To enforce an average kinetic temperature of specified value T :

Rescale all of the current velocities \mathbf{v}_i with correction factor

$$\mathbf{v}'_i = \sqrt{\frac{T}{\mathcal{T}}} \mathbf{v}_i$$

giving a new instantaneous temperature exactly equal to target temperature

$$\mathcal{T}' = \frac{1}{3k_B N} \sum_i^N m_i (\mathbf{v}'_i)^2 = \frac{1}{3k_B N} \sum_i^N m_i \frac{T}{\mathcal{T}} \mathbf{v}_i^2 = T$$

Velocities “initialized” by sampling from the proper equilibrium distribution

$$P(v_{x,i}) = \sqrt{\frac{m_i}{2\pi k_B T}} \exp\left[-\frac{m_i v_{x,i}^2}{2k_B T}\right]$$

and the same for components $v_{y,i}, v_{z,i}$.

```

subroutine tmpscale (natom, dt, ms, tmpset, x, y, z, xm, ym, zm)
ekin := 0
do i := 1, natom
    ekin := ekin + ms(i) * ((x(i) - xm(i)) ** 2 + (y(i) - ym(i)) ** 2 + (z(i) - zm(i)) ** 2)
enddo
tmpkin := ekin / (3 * natom * dt ** 2)
fact := sqrt(tmpset / tmpkin)
do i := 1, natom
    xm(i) := x(i) - fact * (x(i) - xm(i))
    ym(i) := y(i) - fact * (y(i) - ym(i))
    zm(i) := z(i) - fact * (z(i) - zm(i))
enddo
end

```

Temperature scaling procedure for (position) Verlet

<i>natom</i>	number of atoms
<i>dt</i>	timestep
<i>tempset</i>	target temperature
<i>x(i), y(i), z(i)</i>	position particle <i>i</i> at time <i>t</i> (array, $i = 1, \dots, natom$)
<i>xm(i), ym(i), zm(i)</i>	position particle <i>i</i> at time $t - \delta t$ (array, $i = 1, \dots, natom$)
<i>ms(i)</i>	mass of particle <i>i</i> (array, $i = 1, \dots, natom$)

```

do mstep := 1, nstep
  call force (k, natom, x, y, z, fx, fy, fz, epot)
                                     \* scale temperature once every n scale steps *\  

  if (mod(mstep, n scale) = 0) then
    call tmpscale (natom, dt, ms, tmpset, x, y, z, xm, ym, zm)
  endif
  call verlet (natom, dt, ms, fx, fy, fz, x, y, z, xm, ym, zm, ekin)
  etot = ekin + epot
  tmpkin := ekin / (3 * natom * dt * *2)
  print mstep, tmpkin, epot, etot
enddo

```

Verlet MD loop with periodic temperature scaling

<i>mstep</i>	counter for time steps
<i>nstep</i>	total number of time steps
<i>n scale</i>	period (in time steps) for temperature scaling
<i>tmpkin</i>	instantaneous (kinetic) temperature
<i>epot</i>	potential energy
<i>etot</i>	total energy

```

subroutine myran (n, rndm, iseed)
if (iseed ≤ 0)   iseed := 87654321
do i := 1, n
    rndm(i) := ran(iseed)
enddo
end

```

Generate n uniform random numbers in the interval $[0, 1)$

rndm(i) array for storage random numbers
iseed integer seed (must be propagated!, otherwise you will
 have the same series of random numbers in the next call

```

subroutine initv (natom, dt, iseed, x, xm)
do i := 1, natom
    vrndm := -6.0
    call myran (12, rndm, iseed)
    do m := 1, 12
        vrndm := vrndm + rndm(i)
    enddo
    xm(i) := x(i) - dt * vrndm
enddo
end

```

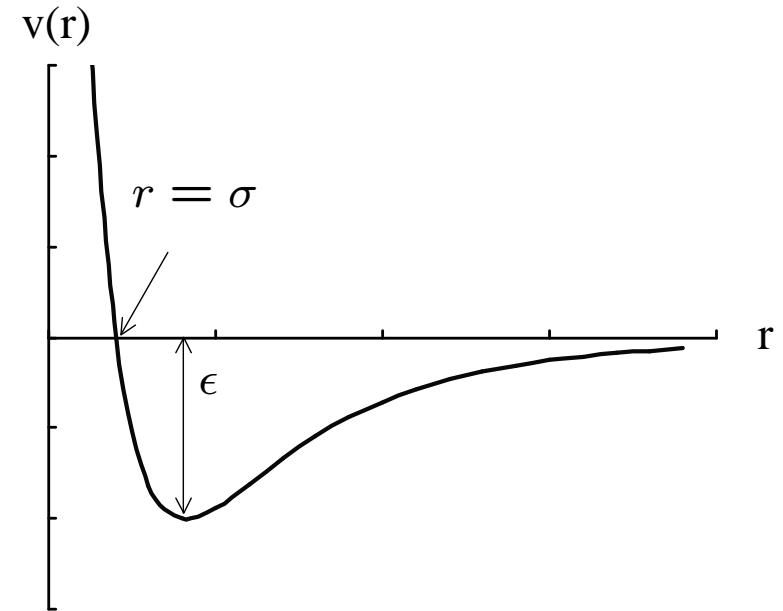
**Sampling a velocity component from Gaussian
with zero mean and unit width: position Verlet modification**

5.11: Spherical cutoff of pair interactions

The Lennard-Jones 12-6 pair potential

$$v(r) = 4\epsilon \left(\left(\frac{\sigma}{r} \right)^{12} - \left(\frac{\sigma}{r} \right)^6 \right)$$

$r < \sigma$	strongly repulsive (hard core)
$r > \sigma$	attractive
$r_0 \approx 1.12\sigma$	minimum $v_0 = -\epsilon$
$r = 3\sigma$	effectively zero $v(3\sigma) \approx -0.005\epsilon$



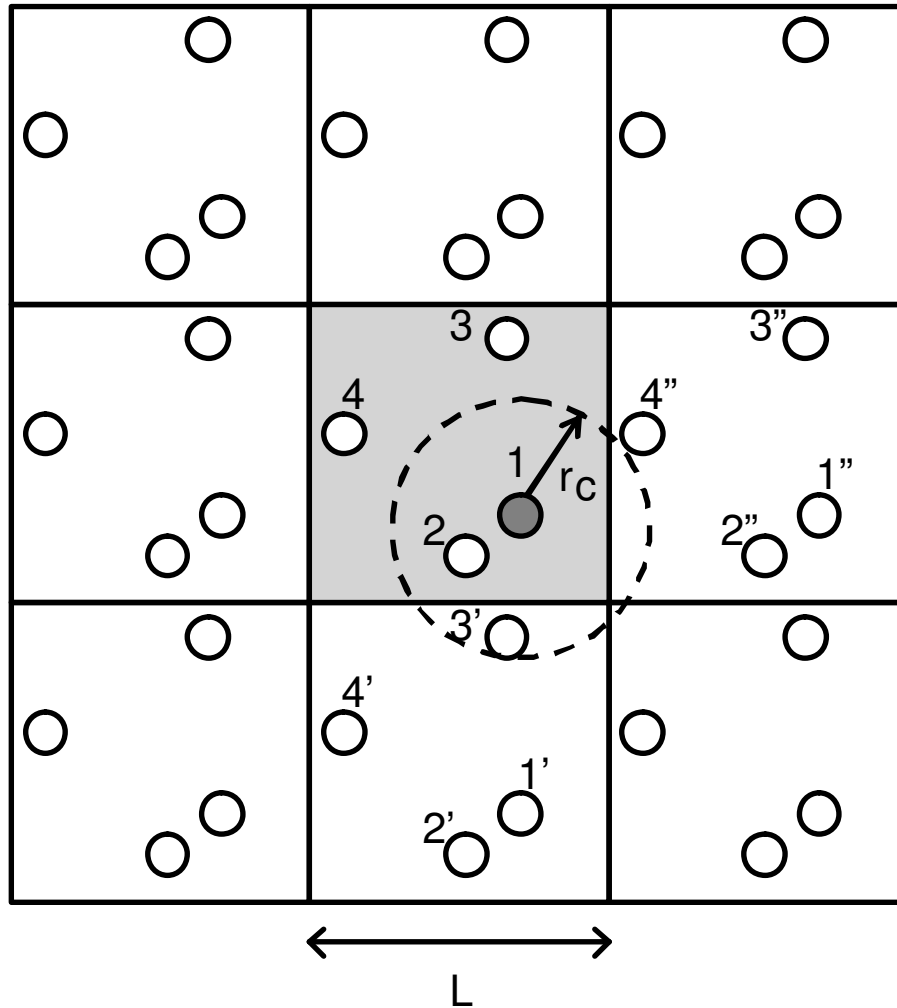
Spherical cutoff: Neglect interaction beyond radius r_c

$$v_c(r) = \begin{cases} v(r) & r \leq r_c \\ 0 & r > r_c \end{cases}$$

Limits the number of pair interactions to be computed (saves computer time)

5.12: Periodic boundaries

Modeling an **extended system** with a finite number of atoms



*Repeat cubic cell of size L
periodically in all directions*

Combined with spherical cutoff

$$r_c < L/2$$

minimum image convention

Inside cutoff sphere particle 1

- particle 2
- image 3' of particle 3

Outside cutoff sphere particle 1

- all images of particle 4.

Avoids physical boundaries but finite size effects of periodicity remain

5.13: Periodic boundaries (formal view)

Periodic boundaries: include interactions in neighboring cells

$$\mathcal{V}(\mathbf{r}^N) = \frac{1}{2} \sum_i^N v_i(\mathbf{r}^N)$$
$$v_i(\mathbf{r}^N) = \sum_{j \neq i}^N \sum_{l, m, n = -\infty}^{+\infty} v(|\mathbf{r}_j + l\mathbf{a} + m\mathbf{b} + n\mathbf{c} - \mathbf{r}_i|)$$

$\mathbf{a}, \mathbf{b}, \mathbf{c}$ span MD cell. Cubic cell: $\mathbf{a} = L\mathbf{e}_x, \mathbf{b} = L\mathbf{e}_y, \mathbf{c} = L\mathbf{e}_z$.

periodic boundaries, box size L
spherical cutoff $r_c \leq \frac{L}{2}$ } \rightarrow minimum image approximation

OK for **short range** interactions $\sigma \ll L$ (e.g. LJ)

```

subroutine force (fmodel, l, rc, natom, x, y, z, fx, fy, fz, epot)
  feps := 4 * fmodel(1)
  sigm2 := fmodel(2) ** 2
  rc2 = rc ** 2
  do i := 1, natom
    fx(i) := 0,    fy(i) := 0,    fz(i) := 0
  enddo
  epot := 0
  do i := 1, natom - 1
    do j := i + 1, natom
      dx := x(j) - x(i),    dy := y(j) - y(i),    dz := z(j) - z(i)
      dr2 := dx ** 2 + dy ** 2 + dz ** 2
      if (dr2 ≤ rc2) then
        dr2nv := 1/dr2
        rm6 := (sigm2 * dr2nv) ** 3
        rm12 := rm6 ** 2
        epot := epot + feps * (rm12 - rm6)
        frad := feps * dr2nv * (12 * rm12 - 6 * rm6)
        fx(j) := fx(j) + frad * dx,    fy(j) := fy(j) + frad * dy,    fz(j) := fz(j) + frad * dz
        fx(i) := fx(i) - frad * dx,    fy(i) := fy(i) - frad * dy,    fz(i) := fz(i) - frad * dz
      endif
    enddo
  enddo
end

```

Force routine for 12-6 potential with spherical cutoff

fmodel(1) LJ 12-6 parameter $\epsilon \rightarrow feps = 4\epsilon$
fmodel(2) LJ 12-6 parameter $\sigma \rightarrow sigm2 = \sigma ** 2$

Minimum image vector for cubic periodic boundary conditions

```
subroutine pbc (l, x1, y1, z1, x2, y2, z2, dx, dy, dz)
  dx = x2 - x1
  dy = y2 - y1
  dz = z2 - z1
  if (l > 0) then
    dx = dx - l * nint (dx/l)
    dy = dy - l * nint (dy/l)
    dz = dz - l * nint (dz/l)
  endif
end
```

l side of periodic cubic box
x1, y1, z1 components \mathbf{r}_1 , position particle $i = 1$
x2, y2, z2 components \mathbf{r}_2 , position particle $j = 2$
dx, dy, dz components minimum image difference vector $\mathbf{r}_{21} = \mathbf{r}_2 - \mathbf{r}_1$

Force loop with minimum image periodic boundary conditions

```
do i := 1, natom - 1
  do j := i + 1, natom
    call pbc (l, x(i), y(i), z(i), x(j), y(j), z(j), dx, dy, dz)
    dr2 := dx ** 2 + dy ** 2 + dz ** 2
    if (dr2 ≤ rc2) then
      ...                                    \* calculations of forces as before * \
    endif
  enddo
enddo
```