Simulations in Statistical Physics

Course for MSc physics students

Janos Török

Department of Theoretical Physics

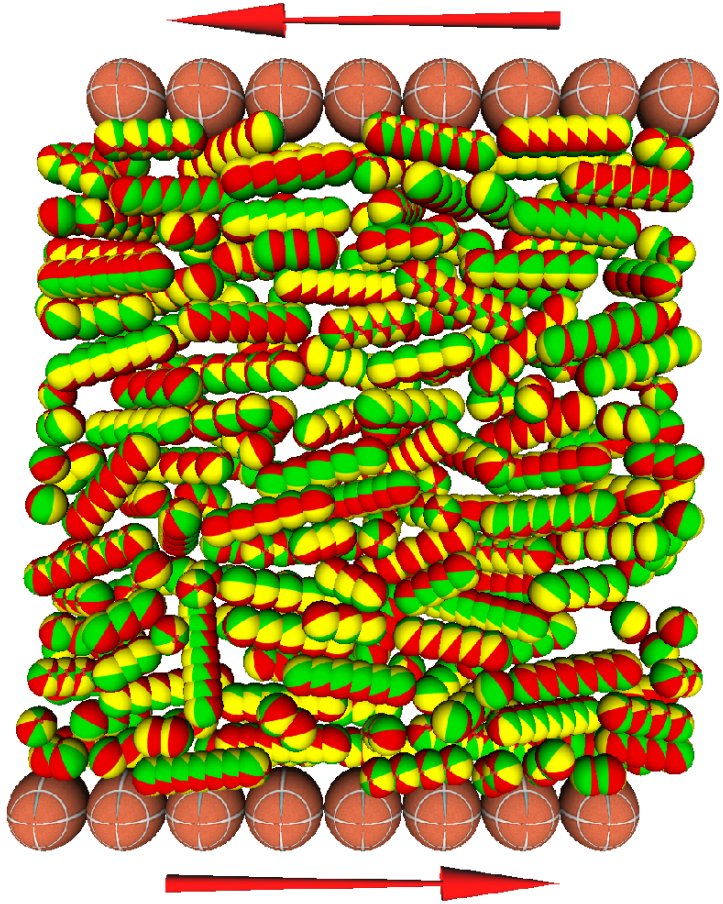
September 30, 2014

Boundary conditions

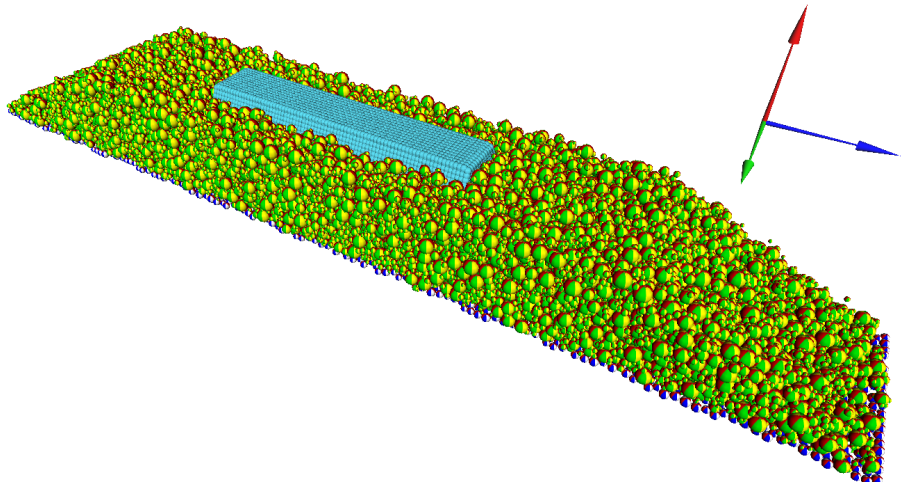
- Real boundary conditions
  - Closed (nothing)
  - Walls (with temperature)
  - Substrate (often too expensive)
- Computer based boundary conditions
  - **Periodic boundary conditions**
  - Absorbing (whatever leaves is gone)
  - Reflecting (everything is reflected back)
  - Walls (some potential)
  - Substrate (fixed basis)
  - ~~Wall with temperature~~

Boundary conditions: Examples

- Periodic boundary conditions
- Walls (some potential)

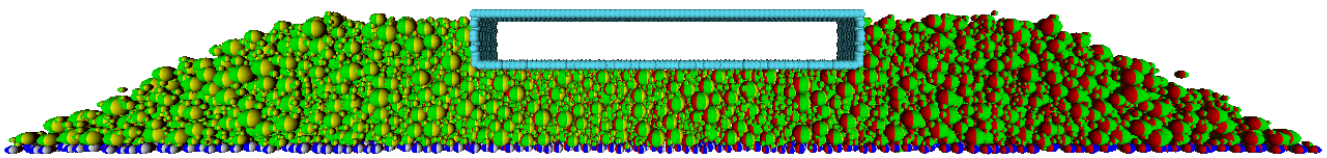


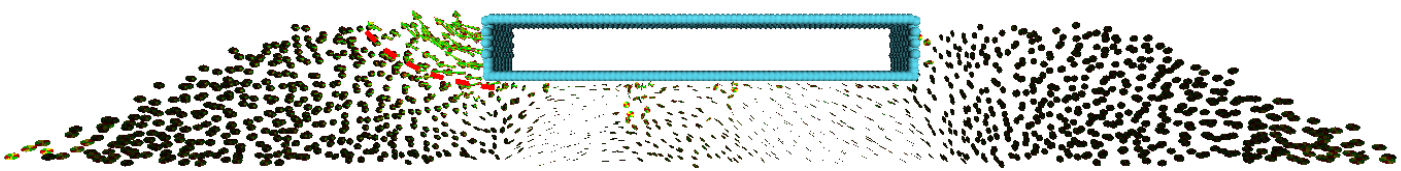
Boundary conditions: Examples



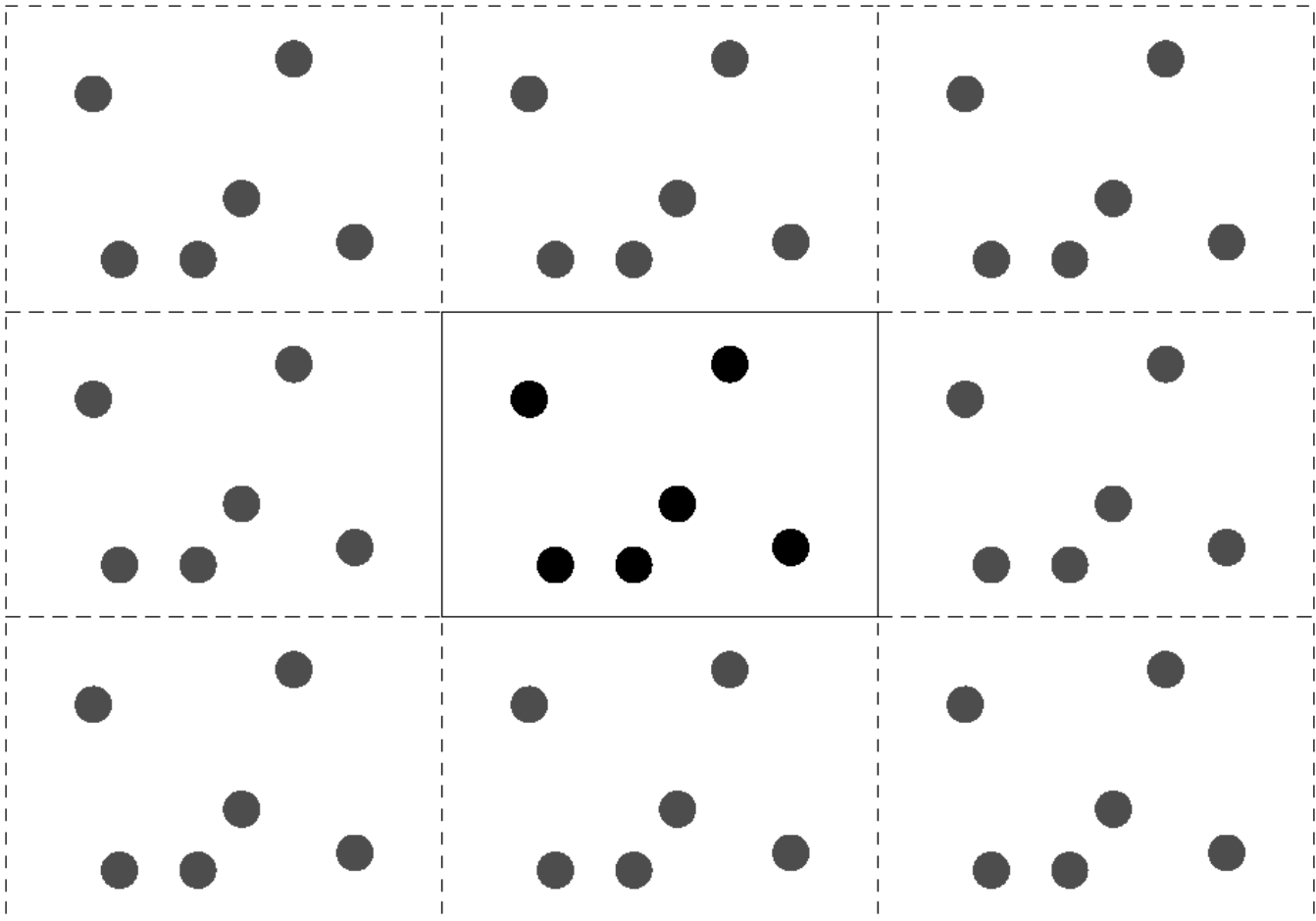
Periodic boundary conditions

Substrate (fixed basis)

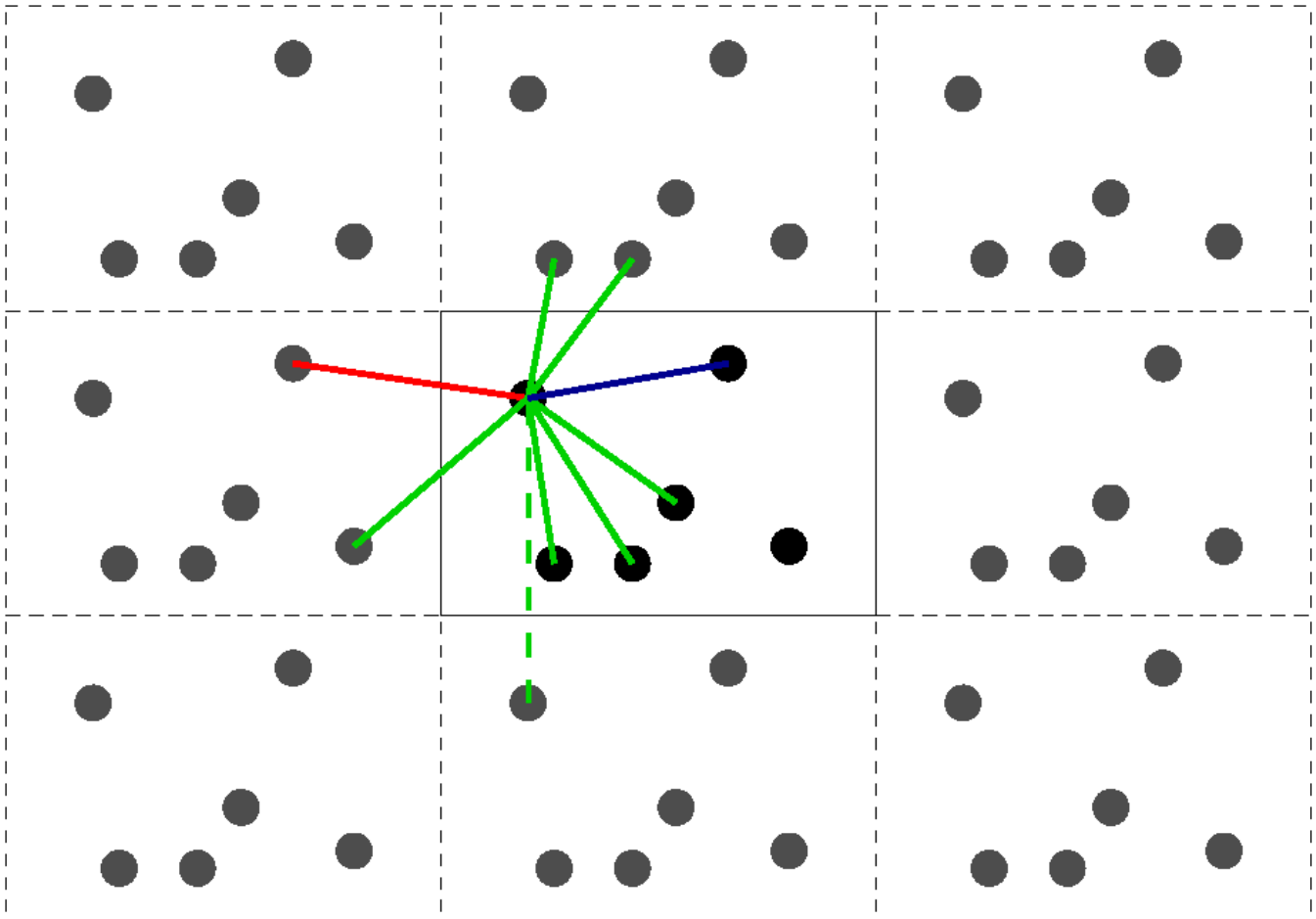




Periodic boundary conditions

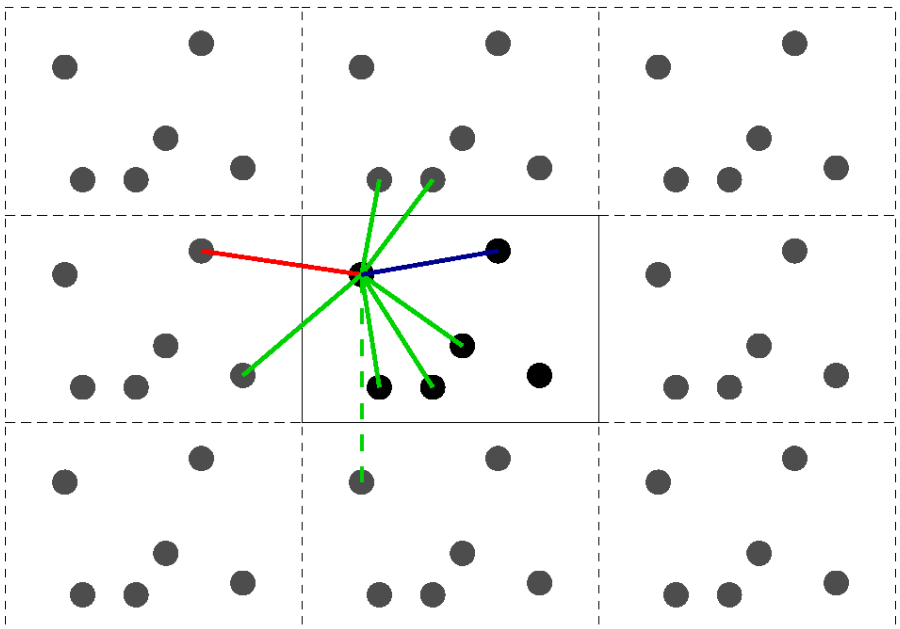


Periodic boundary conditions $\longrightarrow$ contacts



Periodic boundary conditions

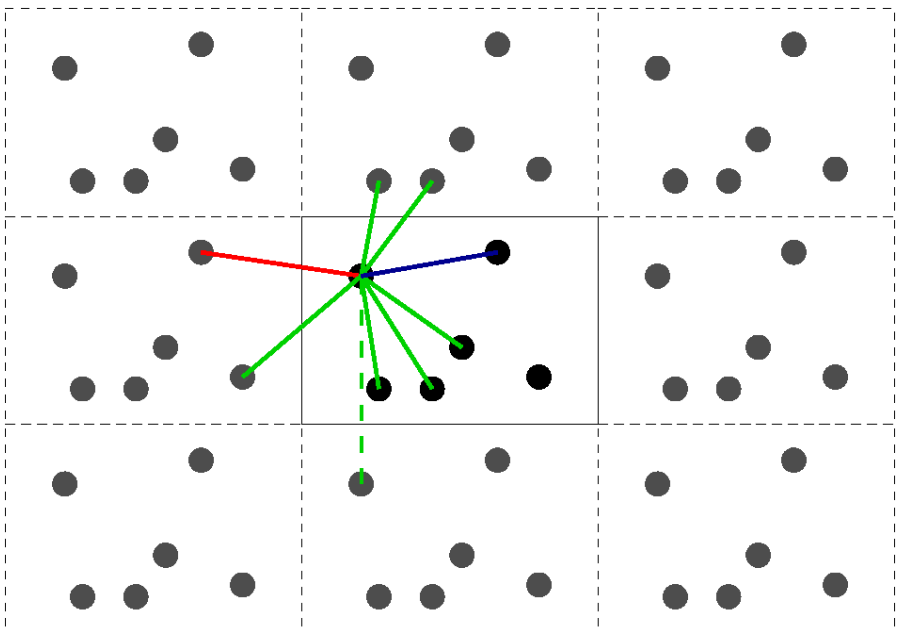
- Infinitely many neighboring cells if long range interactions
- Possibility of self interaction (must be charge neutral)
  - General solution: long range interactions are handled in k-space
- Linear momentum is conserved
- Angular momentum is not conserved



Periodic boundary conditions

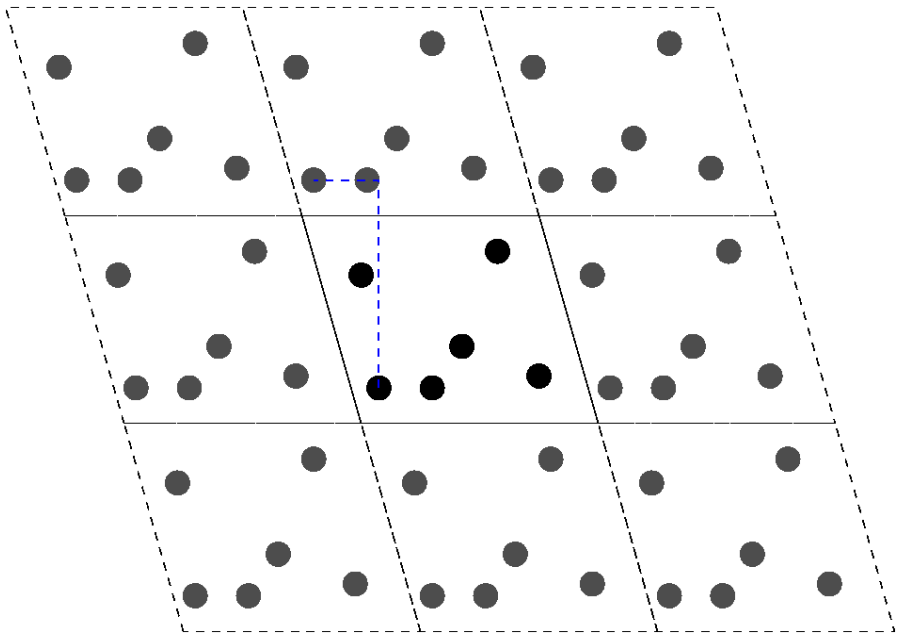
Distance

```
dx = x[i] - x[j]
if (dx < -Lx/2) dx+=Lx;
if (dx >  Lx/2) dx-=Lx;
```



Periodic boundary conditions deformed box

- Box is tilted, positions of particles artificially moved
- Homogeneous shear

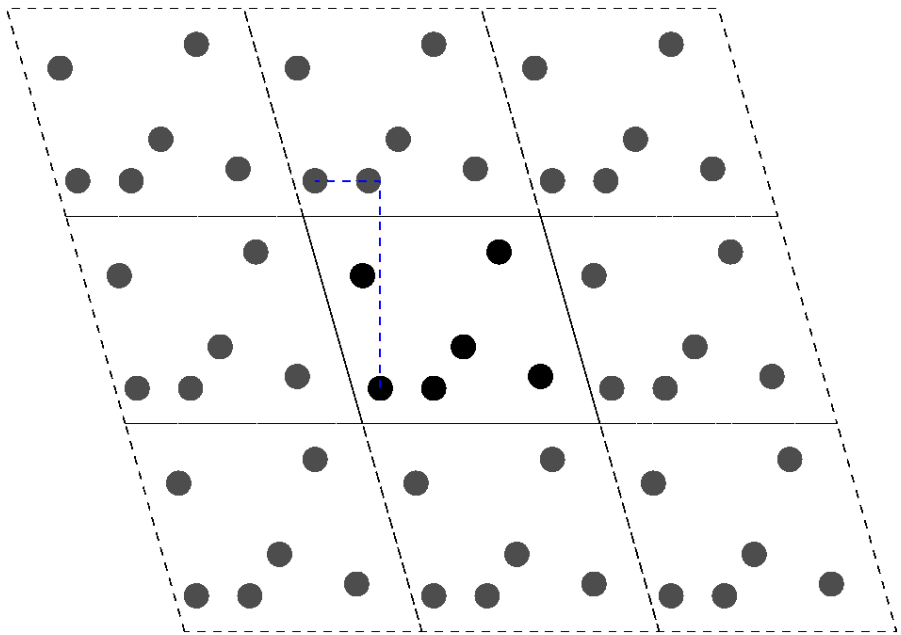


Periodic boundary conditions deformed box

Distance

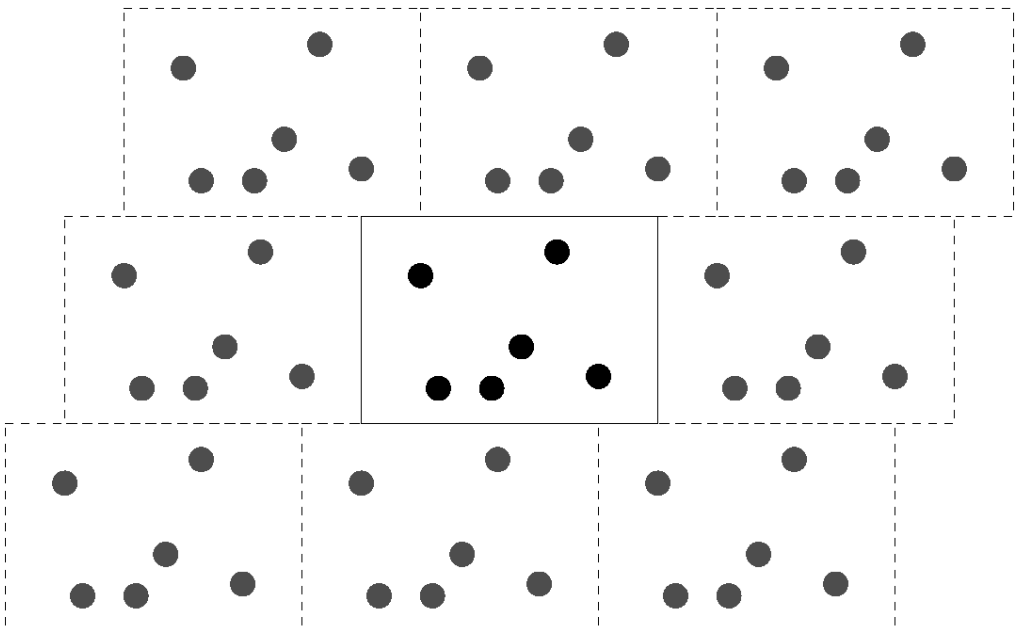
- Order matters
- Tilted: by D_{xy}, D_{xz}, D_{yz}

```
dx = x[i] - x[j]
dy = y[i] - y[j]
dz = z[i] - z[j]
if (dz < -Lz/2) { dz+=Lz; dx+=Dxz; dy+=Dyz; }
if (dz >  Lz/2) { dz-=Lz; dx-=Dxz; dy-=Dyz; }
if (dy < -Ly/2) { dy+=Ly; dx+=Dxy; }
if (dy >  Ly/2) { dy-=Ly; dx-=Dxy; }
if (dx < -Lx/2) dx+=Lx;
if (dx >  Lx/2) dx-=Lx;
```



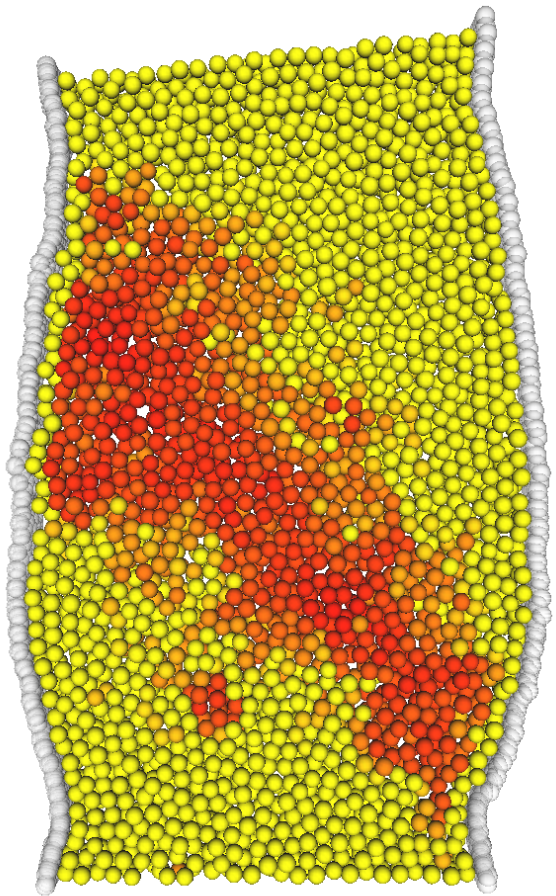
Periodic boundary conditions Lees-Edwards boundary conditions $\rightarrow$ shear

- Images are shifted
- Different from shear by walls
- Different from box tilt
- Stress propagation is important



Molecular dynamics

MD: Molecular dynamics
DEM: Discrete element method

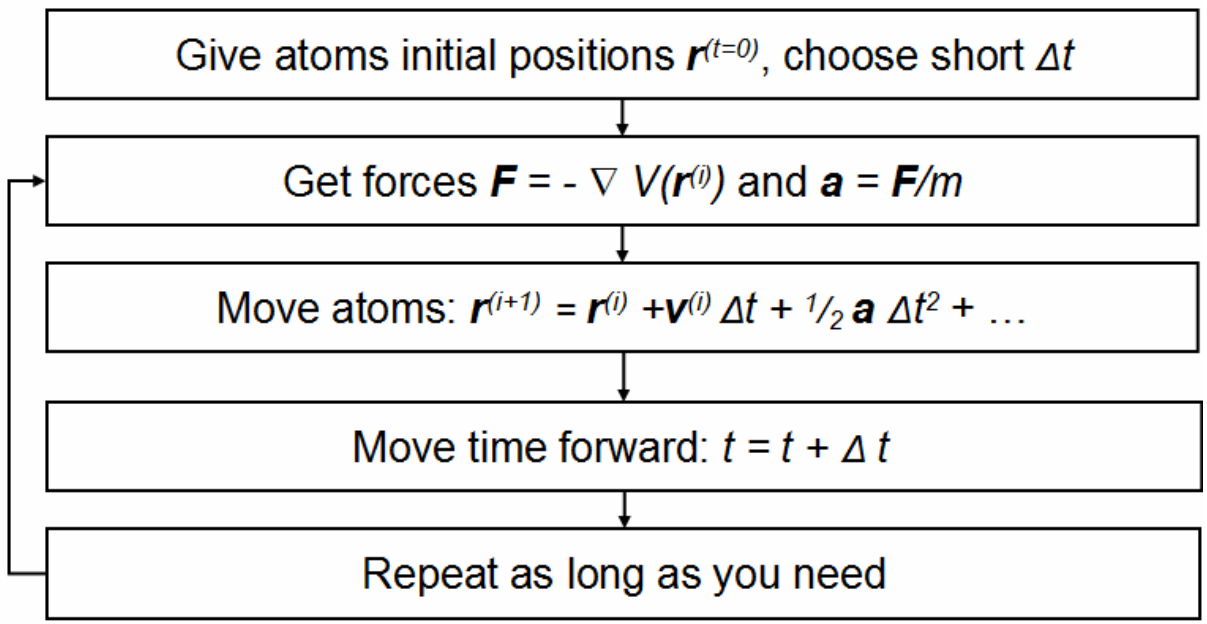


Molecular dynamics

Simulate nature

- Solve Newton's equation of motion

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i = \mathbf{f}_i^{\text{ext}} + \sum_j \mathbf{f}_{ij}^{\text{int}}, \qquad i, j = 1, 2 \ldots N$$



Application of molecular dynamics

- Molecular systems (classic potentials, temperature)
  - Biophysics
  - Structural biology
  - Glasses
  - Amorphous materials
  - Liquids
- Granular materials (hard core, dissipative)
  - Stones, seeds, pills
  - Railbed
- Pedestrians
- Astrological systems (conservative, large scale)

Program

- Have an algorithm to calculate forces
- Get list of interacting particles
- Determine accelerations and velocities; step particles
- (Set temperature)

Forces

Internal forces

- Pair potential:

$$\mathbf{f}_{ij}^{\text{int}} = -\mathbf{f}_{ji}^{\text{int}} = -\nabla V(r_{ij})$$

- Many body potentials (molecular bonds)

$$\mathbf{f}_{ijk}^{\text{int}} = \mathbf{F}(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k)$$

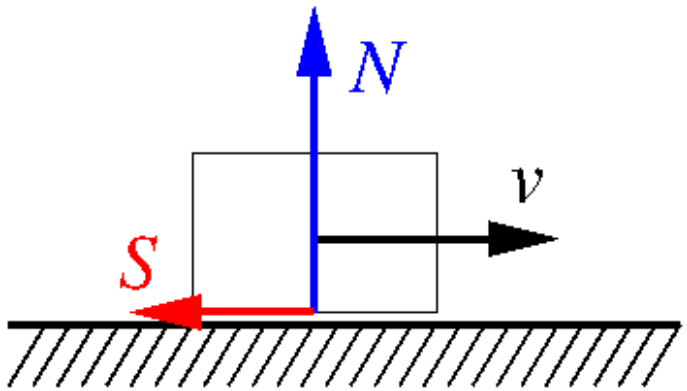
- e.g. 3-body Stillinger-Weber potential:

$$\begin{aligned}
E &= \sum_i \sum_{j>i} \phi_2(r_{ij}) + \sum_i \sum_{j\neq i} \sum_{k>j} \phi_3(r_{ij}, r_{ik}, \theta_{ijk}) \\
\phi_2(r_{ij}) &= A_{ij}\epsilon_{ij}\left[B_{ij}(\frac{\sigma_{ij}}{r_{ij}})^{p_{ij}} - (\frac{\sigma_{ij}}{r_{ij}})^{q_{ij}}\right] \exp\left(\frac{\sigma_{ij}}{r_{ij} - a_{ij}\sigma_{ij}}\right) \\
\phi_3(r_{ij}, r_{ik}, \theta_{ijk}) &= \lambda_{ijk}\epsilon_{ijk}\left[\cos\theta_{ijk} - \cos\theta_{0ijk}\right]^2 \exp\left(\frac{\gamma_{ij}\sigma_{ij}}{r_{ij} - a_{ij}\sigma_{ij}}\right) \exp\left(\frac{\gamma_{ik}\sigma_{ik}}{r_{ik} - a_{ik}\sigma_{ik}}\right)
\end{aligned}$$

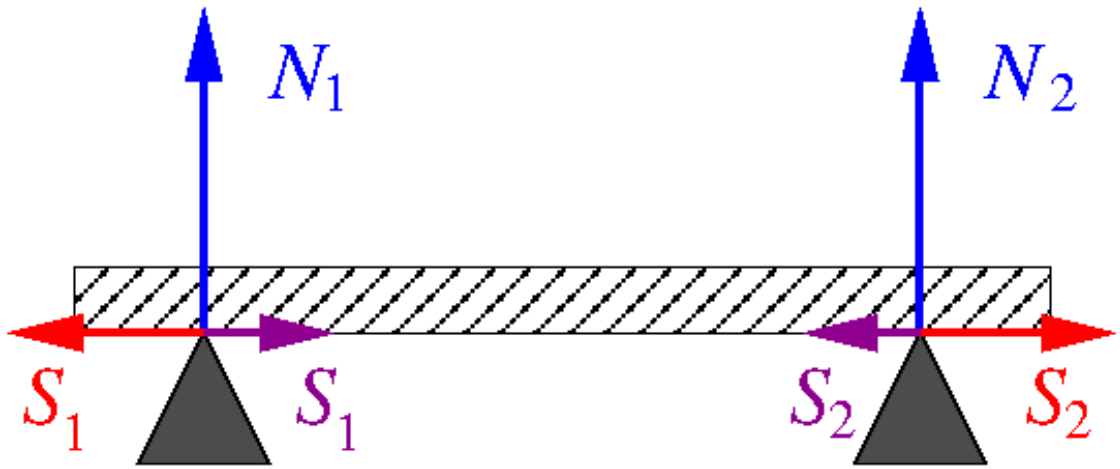
- Friction forces (next slide...)

Friction forces

- Moving:



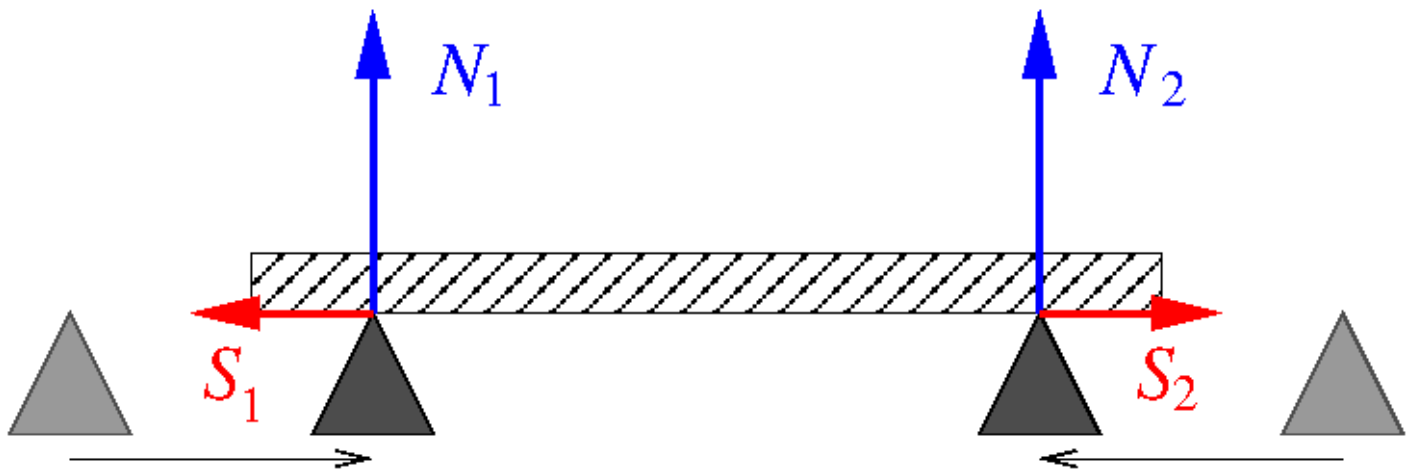
- Stationary:



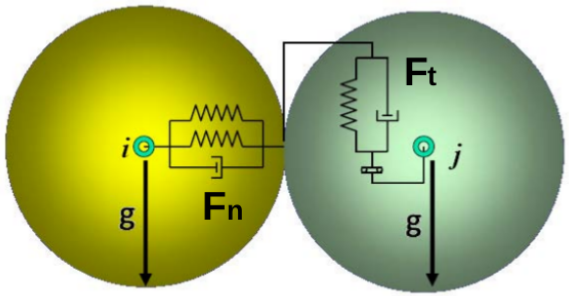
Friction forces

- Position is not enough to set friction forces
- No movement $\rightarrow$ no friction forces
- Solution:

We need history:



Contact history



- Position is not enough to set friction forces
- Normal force:

$$\mathbf{F}_n = k_n \delta \mathbf{n}_{ij} - m_{\text{eff}} \gamma_n \Delta \mathbf{v}_n$$

- Tangential force:

$$\mathbf{F}_t = k_t \Delta \mathbf{s}_t + m_{\text{eff}} \gamma_t \Delta \mathbf{v}_t$$

$$\Delta \mathbf{s}_t = \mathbf{n}_t \int_{t_c}^{t} \left\{ \Delta \mathbf{v}_t(t') + [\omega_i(t') r_i - \omega_j(t') r_j] \right\} dt'$$

- Limit $\Delta \mathbf{s}_t$ to satisfy $|\mathbf{F}_t| \leq \mu \mathbf{F}_n$
- k stiffness, γ damping (critical)

Program

- Have an algorithm to calculate forces
- Get list of interacting particles
- Determine accelerations and velocities; step particles
- (Set temperature)

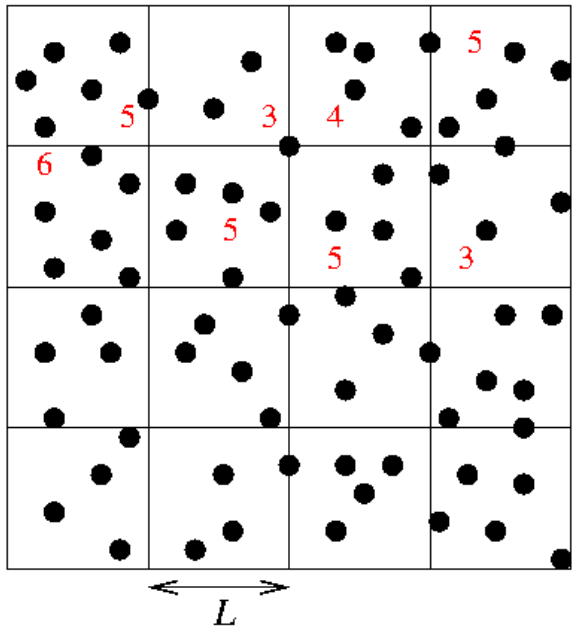
Find pairs

Now we know how to calculate forces. How to get pairs?

- All pairs: $\sim N^2$ calculations. *Only* if there is no other way!
- Short range interactions: box method
- Long range interactions: k-space

Bucketing algorithm

Finite interaction length L



b[0,0]={1,7,9,147,8};
b[0,1]={12,8,99};

- Grid with size L
- Grid of array with particle indexes in box
- Maximum number of neighbors or dynamic array
- If there is $v_{\max}$ then $L' = L + v_{\max}\Delta t$, then reset array every Δt timesteps

k-space solution

- Long reange interactions (e.g. Coulomb) cannot be cut off
- Often more periodic images are needed
- k-space (Fourier-transformation in 3d!)
  - Solution of linear problems by Green's-function
  - Coulomb problem: in Fouier space $\rightarrow$ multiplication!
- Ewald summation:
  - Handle short range in real and long range in k-space

Program

- Have an algorithm to calculate forces
- Get list of interacting particles
- Determine accelerations and velocities; step particles
- (Set temperature)

Euler method

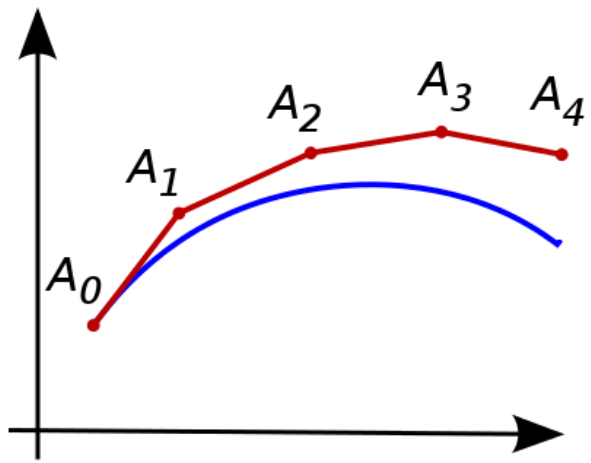
- Velocity:

$$\frac{\Delta v}{\Delta t} = F/m$$

$$\Delta v = F/m\Delta t$$

- Displacement

$$\Delta x = v\Delta t$$



Too bad!

Runge-Kutta method

$$\dot{y} = f(t, y), \quad y(t_0) = y_0.$$

$$y_{n+1} = y_n + \frac{1}{6}h\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$

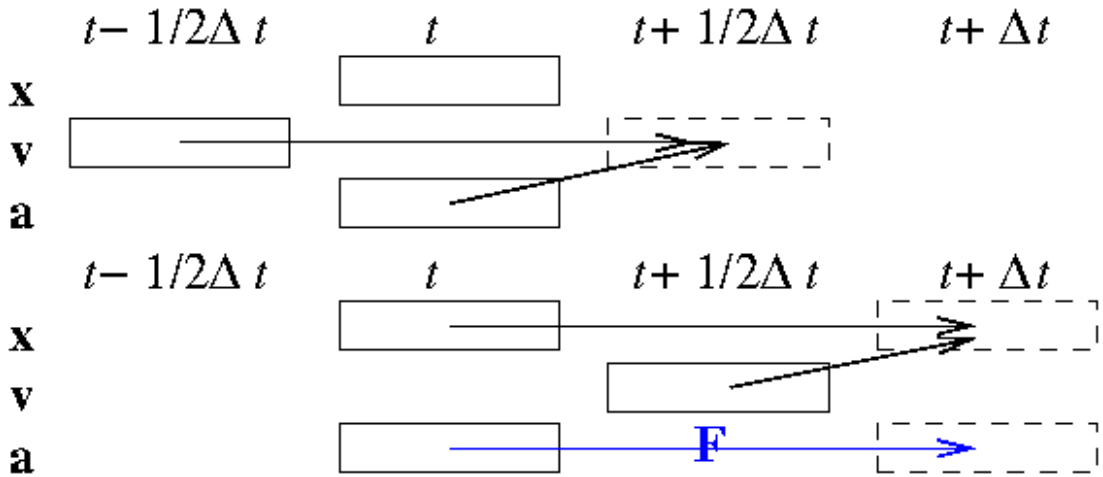
$$t_{n+1} = t_n + h$$

$$k_1 = f(t_n, y_n),$$
$$k_2 = f(t_n + \tfrac{1}{2}h, y_n + \tfrac{h}{2}k_1),$$
$$k_3 = f(t_n + \tfrac{1}{2}h, y_n + \tfrac{h}{2}k_2),$$
$$k_4 = f(t_n + h, y_n + hk_3).$$

- Fourth order method
- Very precise but
  - Four times force calculation
  - No energy conservation (non-sympletic)

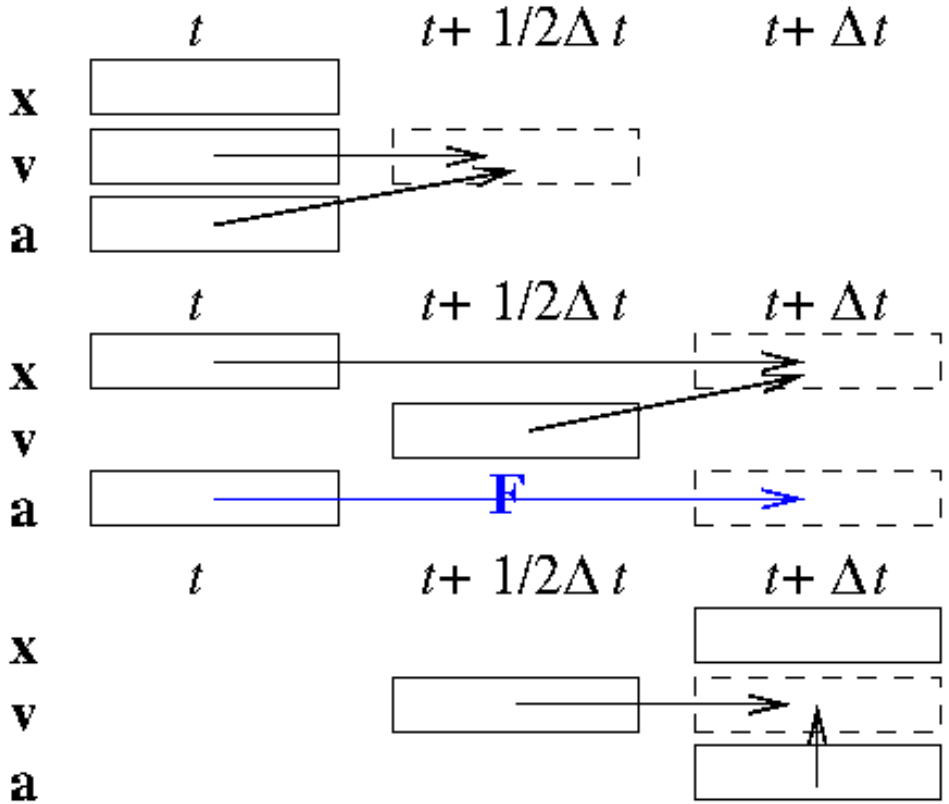
Leapforg method

- Calculate $\mathbf{v}(t + \frac{1}{2}\Delta t) = \mathbf{v}(t - \frac{1}{2}\Delta t) + \mathbf{a}(t)\Delta t$
- Calculate $\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \mathbf{v}(t + \frac{1}{2}\Delta t)\Delta t$



Verlet method

- Calculate $\mathbf{v}(t + \frac{1}{2}\Delta t) = \mathbf{v}(t) + \frac{1}{2}\mathbf{a}(t)\Delta t$
- Calculate $\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \mathbf{v}(t + \frac{1}{2}\Delta t)\Delta t$
- Derive $\mathbf{a}(t + \Delta t)$ from the forces
- Calculate $\mathbf{v}(t + \Delta t) = \mathbf{v}(t + \frac{1}{2}\Delta t) + \frac{1}{2}\mathbf{a}(t + \Delta t)\Delta t$



Sympletic integrator

- Energy (slightly modified) is conserved
- Time reversibility
  - Verlet
  - Leapfrog
- Most molecular dynamics methods use Verlet!
  - Forces are calculated once per turn
  - Microcanonical (NVE) modelling can be only done with these

Multiple time scale integration

- Different force range
  - Short range change fast
  - Long range change slowly
- Recalculate long range forces only in every *n*th times-step
  - Forces are calculated once per turn
- Typical examples:
  - Intramolecular forces: strong, high frequency
  - Intermolecular forces (e.g. Lennard-Jones, Coulomb) slow

Error

Method	Error	Cumulative error
Euler:	Δt^3	Δt
Runge-Kutta:	Δt^5	Δt^4
Verlet:	Δt^4	Δt^2
Leapfrog:	Δt^4	Δt^2