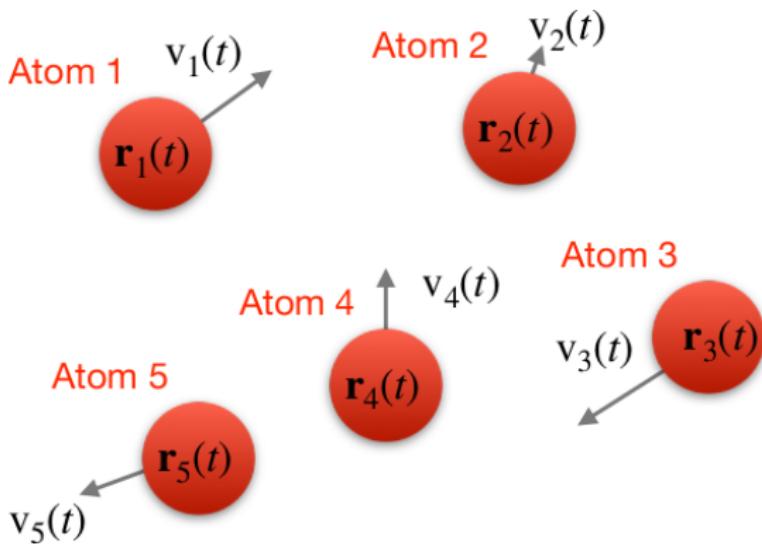


Molecular Modelling and Simulations



Shampa Raghunathan

Recap...

- ▶ Approaches for simulating a finite system to meaningfully represent the actual chemical system, e.g., condensed phases:
 - Periodic boundary conditions
 - Truncating nonbonded interactions using cutoff functions

Outline

1 Energy minimization

2 Molecular dynamics

- Integration Algorithms

Outline

1 Energy minimization

2 Molecular dynamics

- Integration Algorithms

Energy minimization: Statement of the problem

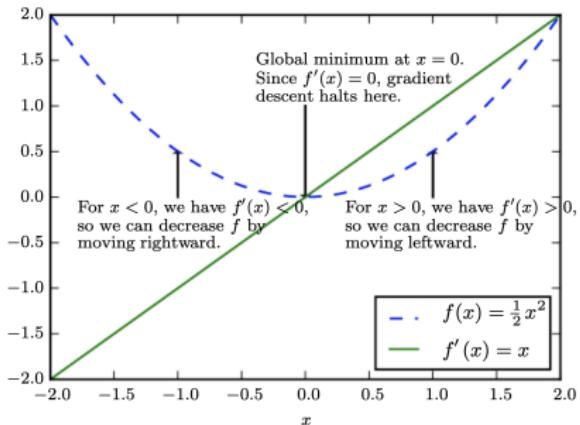
For a given function f which depends on one or more independent variables x_1, x_2, \dots, x_i , find the values of those variables where f has a minimum value. At a minimum point the first derivative of the function with respect to each of the variables is zero, and the second derivatives are all positive:

$$\frac{\partial f}{\partial x_i} = 0 \quad \text{and} \quad \frac{\partial^2 f}{\partial x_i^2} > 0$$

- ▶ First-order minimization algorithm
 - Steepest descent (SD)
 - Conjugate gradient (CONJ)
- ▶ Second-order minimization algorithm
 - Newton Raphson

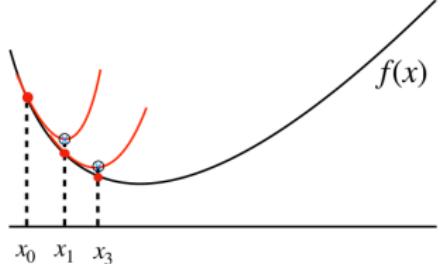
Leach, A. R. (2001). Molecular Modelling: Principles and Applications. Pearson Education.

First-order minimization



First derivative of a function $f'(x)$ is used to follow the function downhill to a minimum. The f is decreased by moving in the direction of the negative gradient, in case of the **steepest descent (SD)** method.

Second-order minimization



Second derivative determines the curvature of a function. Moving to a minimum of quadratic fit at each point in case of the **Newton Raphson (NR)** method.

Goodfellow, I et al. (2016). Deep Learning. The MIT Press.

SD	NR
1. Computationally fast	Expensive
2. Good for minimizing initial structures	Unstable far from minimum
3. Does not require initial structure to be near minimum; slow to converge especially near minimum at low gradient values	Quickly converges near minimum

Outline

- 1 Energy minimization
- 2 Molecular dynamics
 - Integration Algorithms

Newton's equation of motion—for every i^{th} atom

$$F_i = m_i a_i \quad (1)$$

N-coupled differential equations

$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \mathbf{F}_i = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) \quad i = 1, N \quad (2)$$

Here m_i and \mathbf{r}_i are the mass and position of particle i , and $V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N)$ is the potential energy function, that depends on the positions of all the N particles in the system. The negative gradient $-\nabla_i V(\dots)$ of the potential with respect to the position of particle i is the force \mathbf{F}_i acting on that particle.

- ▶ To solve eq. 2 it is necessary to define the initial conditions from which the integration starts. These initial conditions are the initial positions of N atoms and their velocities at the initial time.

Taylor expansion of position about time t at forward time $t + \Delta t$

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \dot{\mathbf{r}}(t)\Delta t + \frac{1}{2}\ddot{\mathbf{r}}(t)\Delta t^2 + \dots \quad (3)$$

alternatively this can be written as

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\Delta t + \frac{1}{2}\mathbf{a}(t)\Delta t^2 + \dots \quad (4)$$

where $\mathbf{v}(t)$ is the velocity vector and $\mathbf{a}(t)$ is the acceleration

Verlet Integrator

Time is discretized into time-step of Δt . Let, x_k be the position after k time-steps, i.e., $x_k = x(t = k\Delta t)$. The central finite difference approximation for the second derivative is

1-D along x spatial coordinate

$$\ddot{x}_k = \frac{x_{k+1} + x_{k-1} - 2x_k}{\Delta t^2}$$
$$\implies x_{k+1} = 2x_k - x_{k-1} + \frac{F(x_k)}{m} \Delta t^2 \quad (5)$$

Since the differential equation is second order, we need to know two points to get the recursion started.

Verlet Integrator

Derivation using Taylor's expansions, along x coordinate

$$x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 + \frac{1}{3}x^{(3)}(t)\Delta t^3 + O(\Delta t^4) \quad (6)$$

$$x(t - \Delta t) = x(t) - \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 - \frac{1}{3}x^{(3)}(t)\Delta t^3 + O(\Delta t^4) \quad (7)$$

Adding two equations 6 and 7, the odd terms cancel

$$x(t + \Delta t) + x(t - \Delta t) = 2x(t) + \ddot{x}(t)\Delta t^2 + O(\Delta t^4) \quad (8)$$

Verlet Integrator

The Verlet algorithm is obtained by neglecting the fourth and higher order terms.

$$\begin{aligned}x(t + \Delta t) &= 2x(t) - x(t - \Delta t) + \ddot{x}(t)\Delta t^2 \\&= 2x(t) - x(t - \Delta t) + \frac{F(x(t))}{m}\Delta t^2\end{aligned}\quad (9)$$

Here, $\ddot{x}(t)$ was replaced by $F(x(t))/m$ (recall Eq. 1). Positions at the next time-step $t + \Delta t$ is calculated based on the positions at the two previous steps, and force on the current step t .

Verlet Integrator

Velocities as such do not appear in the Verlet algorithm, however, can be computed as: Eliminating terms from 2^{nd} and higher orders in equations 6 and 7,

$$\dot{x} = \frac{x(t + \Delta t) - x(t - \Delta t)}{2\Delta t}$$

To compute velocities both positions at $(t + \Delta t)$ and $(t - \Delta t)$ should be available, also subject to error of the order of $O(\Delta t^2)$

- Velocities are needed, for instance, while computing the kinetic energy of a system

Verlet Integrator

- ▶ Algorithm is simple, easy to program does not require any velocity information, only positions are taken into account
- ▶ Single force evaluation per integration cycle
- ▶ Reversible in time

There are various integration algorithm available, like, Velocity Verlet algorithm where, at each step velocity is calculated, and it is 'self-starting' from the position and velocity at the initial time.

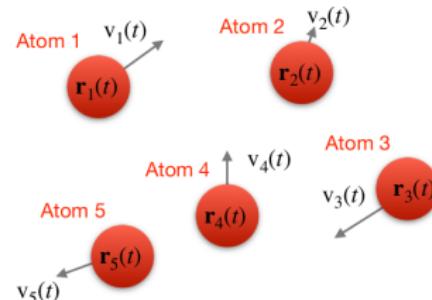
MD and integrators so far today...

- ▶ Propagation of time: Position at time $t + \Delta t$ is determined by position at time t and $t - \Delta t$, and by the forces at time t
- ▶ The equations of motion are *deterministic*, i.e., the positions and the velocities at *time zero* determine the positions and velocities at all other times, t .

Lecture 5

MD simulations: Objective and theory

- ▶ All atoms are moving differently though their motions are coupled
- ▶ Forces acting on them are functions of time
- ▶ Solve equations of motion for a system of N atoms interacting via a potential V



N-coupled differential equations

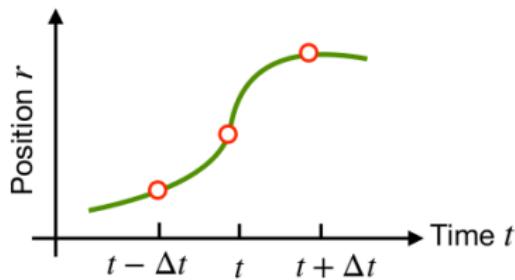
$$m_i \frac{d^2 \mathbf{r}_i}{dt^2} = \mathbf{F}_i = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) \quad i = 1, N$$

- ▶ Analytical solutions of equations of motion to get $x(t)$ and $v(t)$ not possible
- ▶ Numerically solved via finite difference approach

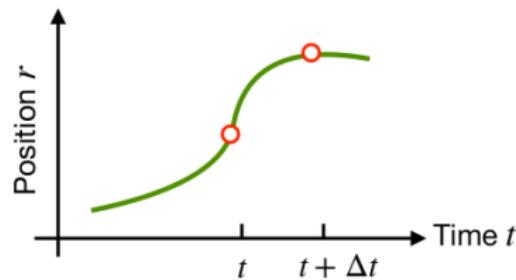
MD simulations: Objective and theory

- ▶ The idea is: Given the positions, velocities at time t , we try to obtain the positions, velocities at a later time $t + \Delta t$
- ▶ If the classical trajectory of a particle is continuous then the estimate of position and velocities at time $t + \Delta t$ may be obtained by Taylor expansion about t
- ▶ Integration algorithm for solving equations of motion must follow:
 - Energy conservation law: Total energy ($T_{\text{kinetic}} + V_{\text{potential}}$) must not change during simulation
 - Time reversible
 - Use of long time-step Δt
 - Fast, easy to program, also less memory-usage

MD simulations: Objective and theory



Verlet “position” integrator

Velocity Verlet
“position” and “velocity” integrator

Velocity Verlet Integrator

Derivation using Taylor's expansions, along x coordinate

$$x(t + \Delta t) = x(t) + \dot{x}(t)\Delta t + \frac{1}{2}\ddot{x}(t)\Delta t^2 + O(\Delta t^3)$$

Recall, $\ddot{x}(t) = \frac{F(x(t))}{m}$; $\dot{x}(t) = v(t)$; $\dot{v}(t) = \frac{F(x(t))}{m}$

$$x(t + \Delta t) = x(t) + v(t)\Delta t + \frac{1}{2}\frac{F(x(t))}{m}\Delta t^2 + O(\Delta t^3) \quad (10)$$

Velocity Verlet Integrator

Now, expanding the second function, $v(t + \Delta t)$

$$v(t + \Delta t) = v(t) + \dot{v}(t)\Delta t + \frac{1}{2}\ddot{v}(t)\Delta t^2 + O(\Delta t^3) \quad (11)$$

Now, we need to develop an expression for \ddot{v} in terms of known quantities. This can be done by expanding $\dot{v}(t + \Delta t)$

$$\dot{v}(t + \Delta t) = \dot{v}(t) + \ddot{v}(t)\Delta t + O(\Delta t^2) \quad (12)$$

Multiplying by $\frac{\Delta t}{2}$ and rearranging, gives

$$\frac{1}{2}\ddot{v}(t)\Delta t^2 = \frac{\Delta t}{2}(\dot{v}(t + \Delta t) - \dot{v}(t)) - O(\Delta t^3) \quad (13)$$

Velocity Verlet Integrator

Now the Eq. 11 for $v(t + \Delta t)$ becomes,

$$v(t + \Delta t) = v(t) + \dot{v}(t)\Delta t + \frac{\Delta t}{2}(\dot{v}(t + \Delta t) - \dot{v}(t)) \quad (14)$$

Using the equation of motion, this can finally be rewritten as

$$v(t + \Delta t) = v(t) + \frac{\Delta t}{2m}(F(x(t + \Delta t)) + F(x(t))) \quad (15)$$

Velocity Verlet Integrator

Quick schematic:

Step 1

$$x_{k+1} = x_k + v_k \Delta t + \frac{1}{2} \frac{F(x_k)}{m} \Delta t^2$$

Step 2

Calculate $F(x_{k+1})$

Step 3

$$v_{k+1} = v_k + \frac{\Delta t}{2m} (F(x_{k+1}) + F(x_k))$$

Recap from the derivatives of the potential energy functions

Force acting on every i^{th} atom

$$\mathbf{F}_i = -\nabla_i V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) \quad i = 1, N$$

Let's derive the force acting on atoms interacting via a pairwise additive potential

$$V(\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_N) = \sum_{i=1}^N \sum_{j=i+1}^N v(r_{ij})$$

r_{ij} is the distance between atoms i and j

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$$

Recap from the derivatives of the potential energy functions

Using the chain rule for differentiation, the derivative of the pair potential can be written as

$$\frac{\partial v}{\partial x_i} = \frac{\partial v}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial x_i}$$

$$\frac{\partial r_{ij}}{\partial x_i} = \frac{(x_i - x_j)}{r_{ij}}$$

$$\frac{\partial v}{\partial r_{ij}} = \frac{24\epsilon}{r_{ij}} \left[-2 \left(\frac{\sigma}{r_{ij}} \right)^{12} + \left(\frac{\sigma}{r_{ij}} \right)^6 \right]$$

Force in the x direction acting on atom i interacting with j

$$F_{x_i} = (x_i - x_j) \frac{24\epsilon}{r_{ij}^2} \left[2 \left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right]$$

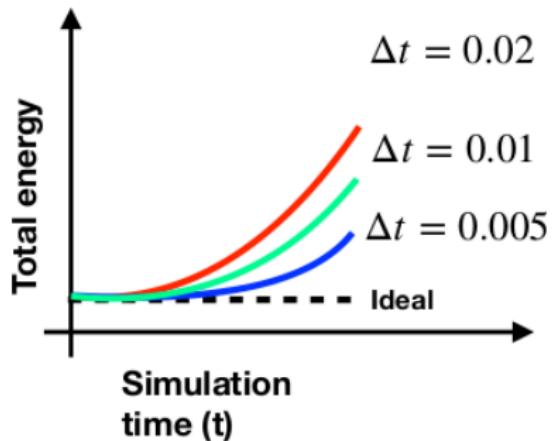
Typical MD simulation

- ➊ Update all positions
 - ➋ Update all forces
 - ➌ Update all velocities
 - ➍ Apply boundary conditions (including temperature, pressure controls)
 - ➎ Move time forward by time-step (dt)
 - ➏ Calculate any output, e.g., total energy
 - ➐ Repeat steps 1–6 as many times as necessary
- Proper use of a position integration algorithm is a robust starting point for MD.

1D application of the Velocity Verlet: $\Delta t=0.005$

1D application of the Velocity Verlet: $\Delta t=0.01$

1D application of the Velocity Verlet: $\Delta t=0.02$

Discretization error, time-step Δt ?

Maxwell-Boltzmann distribution

$$T(t) = \frac{1}{k_B M} \sum_{i=1}^M m_i v_i^2$$

All atoms are assigned initial velocities at random from a Maxwell-Boltzmann distribution of kinetic energy