

Scientific Computing II

Molecular Dynamics – Numerics

Michael Bader – SCCS
Technical University of Munich

Summer 2017



TUM Uhrenturm

Recall: Molecular Dynamics – System of ODEs

- resulting force acting on a molecule: $\vec{F}_i = \sum_{j \neq i} \vec{F}_{ij}$
- acceleration of a molecule (Newton's 2nd law):

$$\ddot{\vec{r}}_i = \frac{\vec{F}_i}{m_i} = \frac{\sum_{j \neq i} \vec{F}_{ij}}{m_i} = -\frac{\sum_{j \neq i} \frac{\partial U(\vec{r}_i, \vec{r}_j)}{\partial |\vec{r}_{ij}|}}{m_i}$$

- or, with acceleration $\vec{a}_i := \frac{1}{m_i} \vec{F}_i$: $\ddot{\vec{r}}_i = \vec{a}_i$,
where \vec{F}_i and \vec{a}_i depend on all positions \vec{r}_i
- transfer to $2dN$ coupled ordinary differential equations of 1st order:

$$\begin{array}{lll} m_i \dot{\vec{r}}_i = \vec{p}_i & \text{or} & \dot{\vec{r}}_i = \vec{v}_i \\ \dot{\vec{p}}_i = \vec{F}_i & & \dot{\vec{v}}_i = \vec{F}_i / m_i = \vec{a}_i \end{array}$$

Euler Time Stepping for MD

Explicit Euler Method:

- Taylor series expansion of the positions in time:

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \dot{\vec{r}}(t) + \frac{1}{2} \Delta t^2 \ddot{\vec{r}}(t) + \frac{\Delta t^i}{i!} \vec{r}^{(i)}(t) + \dots \quad (1)$$

(\dot{r} , \ddot{r} , $r^{(i)}$): derivatives)

- neglecting terms of higher order of Δt , and analogous formulation of $\vec{v}(t) := \dot{\vec{r}}(t)$ with $\vec{a}(t) := \dot{\vec{v}}(t) = \ddot{\vec{r}}(t) = \frac{\vec{F}(t)}{m}$ leads to the explicit Euler method:

$$\vec{v}(t + \Delta t) \doteq \vec{v}(t) + \Delta t \vec{a}(t)$$

$$\vec{r}(t + \Delta t) \doteq \vec{r}(t) + \Delta t \vec{v}(t)$$

Euler Time Stepping for MD (cont.)

- explicit Euler method:

$$\vec{v}(t + \Delta t) \doteq \vec{v}(t) + \Delta t \vec{a}(t) \quad (2a)$$

$$\vec{r}(t + \Delta t) \doteq \vec{r}(t) + \Delta t \vec{v}(t) \quad (2b)$$

- similar for implicit Euler method
→ derivatives at the time step end:

$$\vec{v}(t + \Delta t) \doteq \vec{v}(t) + \Delta t \vec{a}(t + \Delta t) \quad (3a)$$

$$\vec{r}(t + \Delta t) \doteq \vec{r}(t) + \Delta t \vec{v}(t + \Delta t) \quad (3b)$$

- disadvantage for both schemes: do not conserve critical properties;
lead to wrong long-term solutions (compare tutorial on circular motion);
plus: low accuracy

Classical Störmer Verlet Method

- the Taylor series expansion in (1) can also be performed for $-\Delta t$:

$$\vec{r}(t - \Delta t) = \vec{r}(t) - \Delta t \dot{\vec{r}}(t) + \frac{1}{2} \Delta t^2 \ddot{\vec{r}}(t) + \frac{(-\Delta t)^i}{i!} \vec{r}^{(i)}(t) + \dots \quad (4)$$

- from (1) and (4) the classical Verlet algorithm can be derived:

$$\begin{aligned} \vec{r}(t + \Delta t) &= 2\vec{r}(t) - \vec{r}(t - \Delta t) + \Delta t^2 \ddot{\vec{r}}(t) + \mathcal{O}(\Delta t^4) \\ &\approx 2\vec{r}(t) - \vec{r}(t - \Delta t) + \Delta t^2 \ddot{\vec{a}}(t) \end{aligned} \quad (5)$$

note: direct calculation of $\vec{r}(t + \Delta t)$ from $\vec{r}(t)$ and $\vec{F}(t)$

- velocity can be estimated via

$$\vec{v}(t) = \dot{\vec{r}}(t) \doteq \frac{\vec{r}(t + \Delta t) - \vec{r}(t - \Delta t)}{2\Delta t} \quad (6)$$

- disadvantage: needs to store two previous time steps

Crank Nicolson Method

- explicit approximation (7a) for half step $[t, t + \frac{\Delta t}{2}]$ inserted into implicit approximation (7b) for half step $[t + \frac{\Delta t}{2}, t + \Delta t]$

$$\vec{v}(t + \frac{\Delta t}{2}) = \vec{v}(t) + \frac{\Delta t}{2} \vec{a}(t) \quad (7a)$$

$$\vec{v}(t + \Delta t) = \vec{v}(t + \frac{\Delta t}{2}) + \frac{\Delta t}{2} \vec{a}(t + \Delta t) \quad (7b)$$

- leads to Crank-Nicolson scheme for v :

$$\vec{v}(t + \Delta t) = \vec{v}(t) + \frac{\Delta t}{2} (\vec{a}(t) + \vec{a}(t + \Delta t)) \quad (8)$$

- key disadvantage: implicit scheme, as $\vec{a}(t + \Delta t)$ depends on $\vec{r}(t + \Delta t)$; needs to solve non-linear system of equations

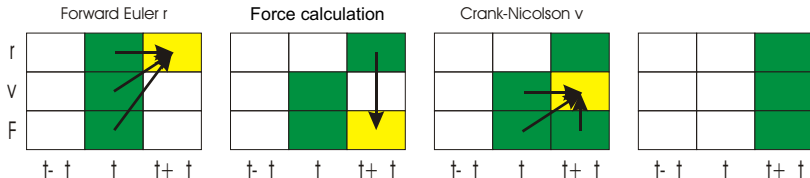
Velocity Störmer Verlet Method

The Velocity Störmer Verlet method is a composition of a

- Taylor series expansion of 2nd order for the positions, as in Eq. (1)
- and a Crank Nicolson method for the velocities, as in Eq. (8)

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}(t) + \frac{\Delta t^2}{2} \vec{a}(t) \quad (9a)$$

$$\vec{v}(t + \Delta t) = \vec{v}(t) + \frac{\Delta t}{2} (\vec{a}(t) + \vec{a}(t + \Delta t)) \quad (9b)$$



Memory requirement: $(3 + 1) \cdot 3N$ (3+1 vector fields)

update of $v(t + \Delta t)$ requires $v(t)$, $r(t + \Delta t)$ and $F(t + \Delta t)$, but also $F(t)$

Velocity Störmer Verlet – Implementation

- reformulate equation for positions \vec{r} :

$$\begin{aligned}\vec{r}(t + \Delta t) &= \vec{r}(t) + \Delta t \vec{v}(t) + \frac{\Delta t^2}{2} \vec{a}(t) \\ &= \vec{r}(t) + \Delta t \left(\vec{v}(t) + \frac{\Delta t}{2} \vec{a}(t) \right)\end{aligned}$$

contains half an Euler time step for \vec{v}

- similar for the velocities \vec{v} :

$$\begin{aligned}\vec{v}(t + \Delta t) &= \vec{v}(t) + \frac{\Delta t}{2} (\vec{a}(t) + \vec{a}(t + \Delta t)) \\ &= \left(\vec{v}(t) + \frac{\Delta t}{2} \vec{a}(t) \right) + \frac{\Delta t}{2} \vec{a}(t + \Delta t)\end{aligned}$$

reuses the result of the half Euler time step for \vec{v}

Velocity Störmer Verlet – Implementation (2)

1. compute half an Euler time step for \vec{v}

$$\vec{v}(t + \frac{\Delta t}{2}) = \vec{v}(t) + \frac{\Delta t}{2} \vec{a}(t)$$

2. update positions \vec{r} :

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}(t + \frac{\Delta t}{2})$$

3. calculate forces $\vec{a}(t + \Delta t)$ from positions $\vec{r}(t + \Delta t)$

4. update the velocities \vec{v} :

$$\vec{v}(t + \Delta t) = \vec{v}(t + \frac{\Delta t}{2}) + \frac{\Delta t}{2} \vec{a}(t + \Delta t)$$

Note: memory requirements: $3 \cdot 3N$ (3 vector fields)

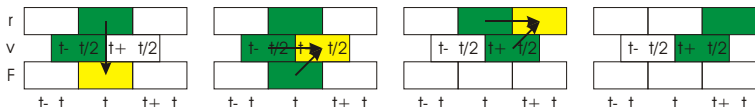
vectors \vec{v} and \vec{r} , as well as forces/accelerations \vec{a} may be updated in-place in each time step

Leapfrog Method

- combine steps 4 (from previous time step) and 1 to a single step
- velocity calculation thus shifted by a half time step:

$$\vec{v}(t + \frac{\Delta t}{2}) = \vec{v}(t - \frac{\Delta t}{2}) + \Delta t \vec{a}(t) \quad (10a)$$

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}(t + \frac{\Delta t}{2}) \quad (10b)$$



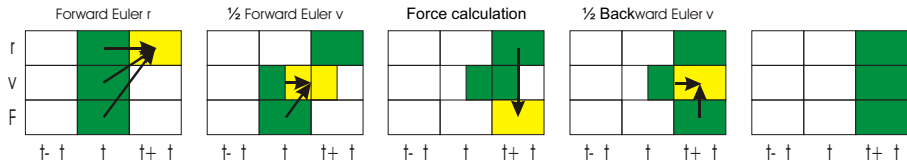
- exact arithmetic: Störmer Verlet, Velocity Störmer Verlet and Leapfrog Scheme are equivalent
- the latter two are more robust w.r.t. roundoff errors

Dimensionless Velocity Störmer Verlet

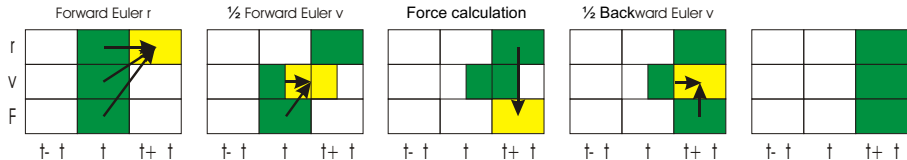
- remember dimensionless formulation:
 $(\vec{r} := \sigma \vec{r}^*, \vec{v} := \frac{\sigma}{\Delta t} \vec{v}^*, \Delta t^2 := \sigma^2 \frac{m}{\epsilon} \Delta t^{*2}, \ddot{\vec{r}} = \frac{1}{m} \vec{F} := \frac{1}{m} \frac{\epsilon}{\sigma} \vec{F}^*)$
- insert into Velocity Störmer Verlet Method to get:.

$$\vec{r}^*(t + \Delta t) = \vec{r}^*(t) + \vec{v}^*(t) + \frac{\Delta t^{*2}}{2} \vec{F}^*(t) \quad (11a)$$

$$\vec{v}^*(t + \Delta t) = \vec{v}^*(t) + \frac{\Delta t^{*2}}{2} \vec{F}^*(t) + \frac{\Delta t^{*2}}{2} \vec{F}^*(t + \Delta t) \quad (11b)$$



Dimensionless Velocity Störmer Verlet (2)



Procedure:

1. calculate new positions (11a),
partial velocity update: $+\frac{\Delta t^*{}^2}{2} \vec{F}^*(t)$ in (11b)
2. calculate new forces, accelerations (computationally intensive!)
3. calculate new velocities: $+\frac{\Delta t^*{}^2}{2} \vec{F}^*(t + \Delta t)$ in (11b)

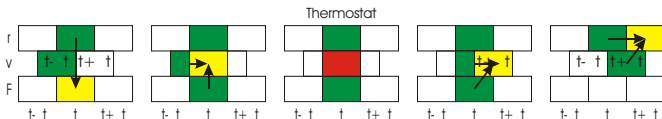
→ memory requirements: $3 \cdot 3N$

Outlook: Leapfrog Method with Thermostat

- Leapfrog method:

$$\vec{v}(t + \frac{\Delta t}{2}) = \vec{v}(t - \frac{\Delta t}{2}) + \Delta t \vec{a}(t)$$

$$\vec{r}(t + \Delta t) = \vec{r}(t) + \Delta t \vec{v}(t + \frac{\Delta t}{2})$$



- intermediate step may be introduced for the thermostat
 $\vec{v}(t) := \frac{1}{2} (\vec{v}(t + \frac{\Delta t}{2}) + \vec{v}(t - \frac{\Delta t}{2}))$ to synchronize velocity:

$$\vec{v}_{act}(t) = \vec{v}(t - \frac{\Delta t}{2}) + \frac{\Delta t}{2} \vec{a}(t) \quad (13a)$$

$$\vec{v}(t + \frac{\Delta t}{2}) = (2\beta - 1)\vec{v}_{act}(t) + \frac{\Delta t}{2} \vec{a}(t) \quad (13b)$$

Evaluation of Time Integration Methods

- accuracy (not of great importance for exact particle positions)
- stability
- conservation
 - of phase space density (symplectic)
 - of energy
 - of momentum
(especially with PBC → Periodic Boundary Conditions)
- reversibility of time
- use of resources:
 - computational effort (number of force evaluations)
 - maximum time step size
 - memory usage

Reversibility of Time

- time reversal for a closed system means
 - a turnaround of the velocities and also momentums; positions at the inversion point stay constant
 - traverse of a trajectory back in the direction of the origin
- demand for symmetry for time integration methods
 - + satisfied by Verlet method, e.g.
 - not satisfied by, e.g., Euler method, Predictor Corrector methods (also not by standard Runge-Kutta methods)
- contradiction with
 - the H-theorem (increase of entropy, irreversible processes)? (Loschmidt's paradox)
 - the second theorem of thermodynamics?
 - reversibility in theory only for a very short time
- Lyapunov instability \Rightarrow Kolmogorov entropy

Lyapunov Instability

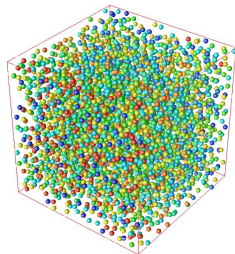
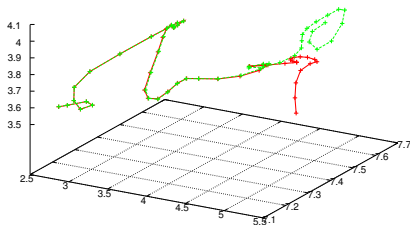
- Basic question: how does a model behave with slightly disturbed initial condition?
- Example of a simple system:
 - stable case:
jumping ball on a plane with slightly disturbed initial horizontal velocity \Rightarrow linear increase of the disturbance
 - instable case:
jumping ball on a sphere with slightly disturbed initial horizontal velocity \Rightarrow exponential increase of the disturbance (Lyapunov exponent)
- for the instable case, small disturbances result in large changes: chaotic behaviour (butterfly leading to a hurricane?)
- non-linear differential equations are often dynamically instable

Lyapunov Instability: A Numerical Experiment

- setup of 4000 fcc atoms
- for a second setup, the position of a single atom was displaced by 0.001
- this atom is traced in both setups

tracing a Molecule (with initial displacement)

Molecule 25, run1 ————
 Molecule 25, run2 - - - - -

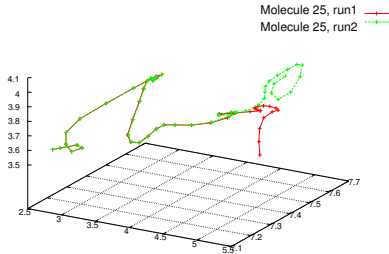


colours indicate velocity

Lyapunov Instability: A Numerical Experiment

- Calculation of the trajectories \rightarrow badly conditioned problem: a small change of the initial position of a molecule may result in a distance to the comparable original position, after some time, in the magnitude of the whole domain!
- Thus: do not target at simulation of individual trajectories \rightarrow numerical simulation of the behaviour of the system is wanted!

tracing a Molecule (with initial displacement)



Molecule deviation (with initial displacement)

