MHPC
Molecular dynamics

Sabine Reißer

Equations of motion

$$\begin{aligned}
\dot{\boldsymbol{r}}_i &= \boldsymbol{p}_i / m_i \\
\dot{\boldsymbol{p}}_i &= -\nabla_{\boldsymbol{r}_i} V_i = \boldsymbol{F}_i
\end{aligned}$$

Integrator

- Conserve energy
- Time reversible
- Long time step Δt
- As exact as possible
- Easy to implement
- Fast

Taylor expansion

$$
\begin{aligned}
r(t+\Delta t) &= r(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 + \ \ldots \\
v(t+\Delta t) &= v(t) + a(t)\Delta t + \frac{1}{2}\dot{a}(t)\Delta t^2 + \ \ldots
\end{aligned}
$$

Taylor expansion

$$
\begin{aligned}
r(t+\Delta t) &= r(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 + \ldots \\
v(t+\Delta t) &= v(t) + a(t)\Delta t + \frac{1}{2}\cancel{\dot{a}(t)}\Delta t^2 + \ldots
\end{aligned}
$$

$$
\begin{aligned}
a &= F/m \\
F(r) &= -\frac{\partial V(R^N)}{\partial r}
\end{aligned}
$$

Euler algorithm

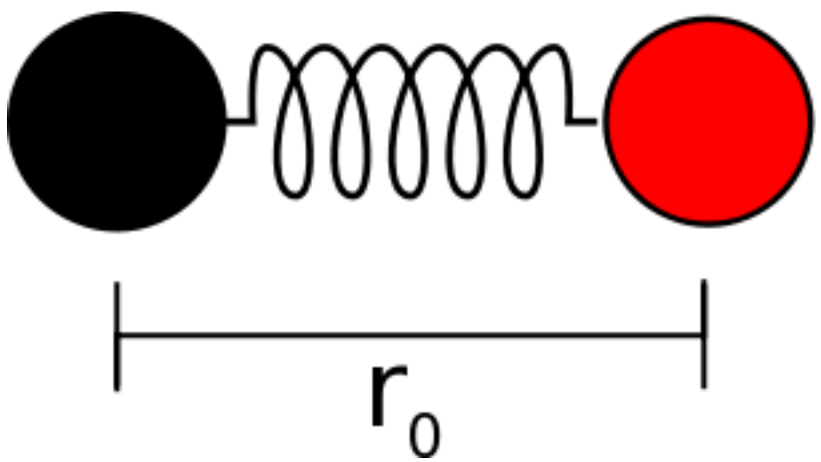
$$\begin{aligned} r(t+\Delta t) &= r(t) + v(t)\Delta t \\ v(t+\Delta t) &= v(t) + a(t)\Delta t \end{aligned}$$

Time step Δt

Vibrational Mode[a]	Wave number $(1/\lambda)$ [cm^{-1}]	Period T_p (λ/c) [fs][b]	T_p/π [fs]
O–H, N–H stretch	3200-3600	9.8	3.1
C–H stretch	3000	11.1	3.5
O–C–O asymm. stretch	2400	13.9	4.5
C≡C, C≡N stretch	2100	15.9	5.1
C=O (carbonyl) stretch	1700	19.6	6.2
C=C stretch			
H–O–H bend	1600	20.8	6.4
C–N–H, H–N–H bend	1500	22.2	7.1
C=C (aromatic) stretch			
C–N stretch (amines)	1250	26.2	8.4
Water Libration (rocking)	800	41.7	13
O–C–O bending	700	47.6	15
C=C–H bending (alkenes)			
C=C–H bending (aromatic)			

Schlick (2010)

O-H vibration



$$V_{bond}(r) = \frac{1}{2}k(r - r_0)^2$$

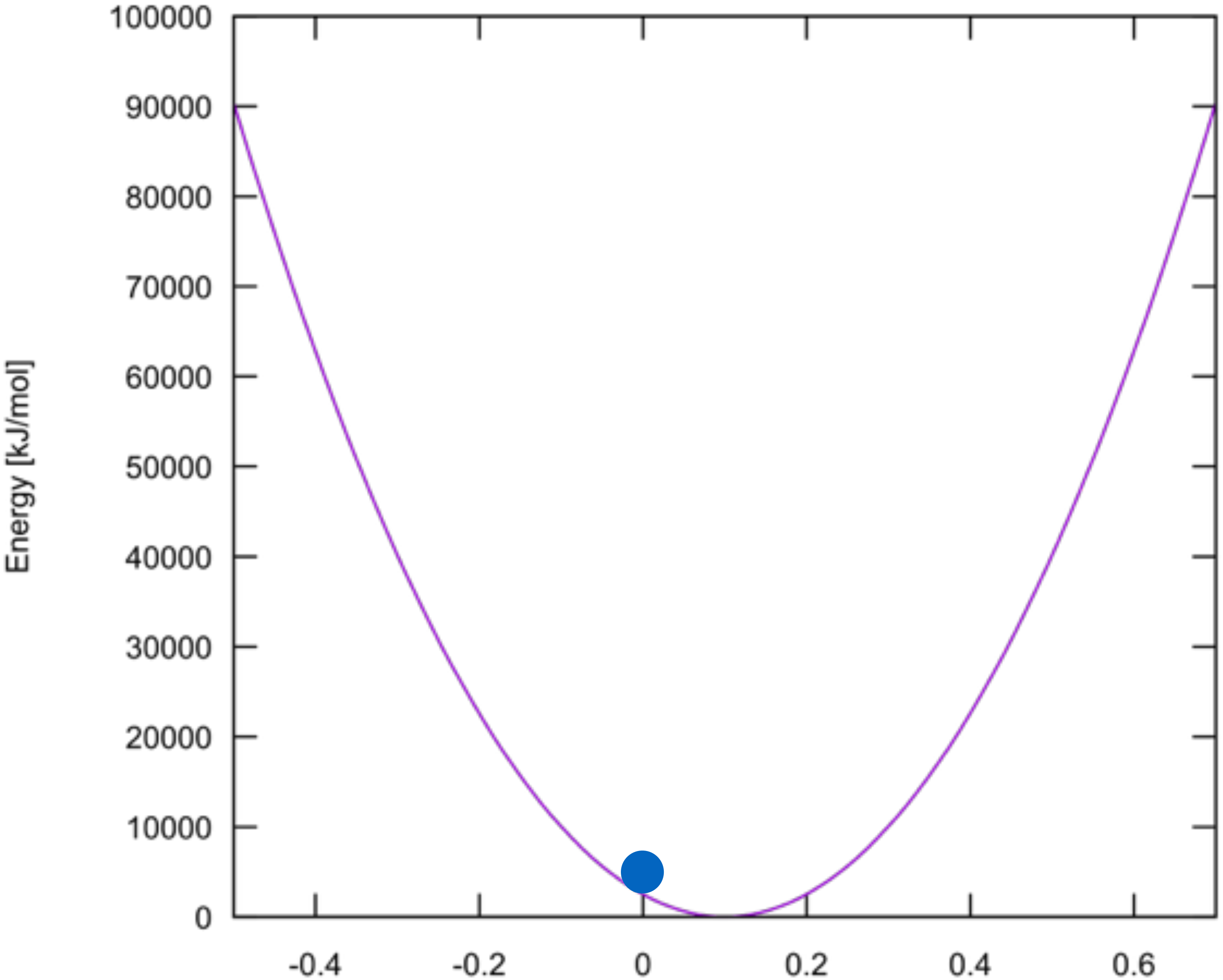
$$\nu = \frac{1}{2\pi}\sqrt{\frac{k}{\mu}}$$

Obtain from IR spectroscopy

$$\mu = \frac{m_1 m_2}{m_1 + m_2}$$

$$\begin{aligned} T_p &= 1/\nu \\ T_p(O - H) &\approx 10 \text{ fs} \\ r_0(O - H) &\approx 0.1 \text{ nm} \\ k &\approx 502080 \text{ kJ/(mol nm}^2) \end{aligned}$$

Particle in a well



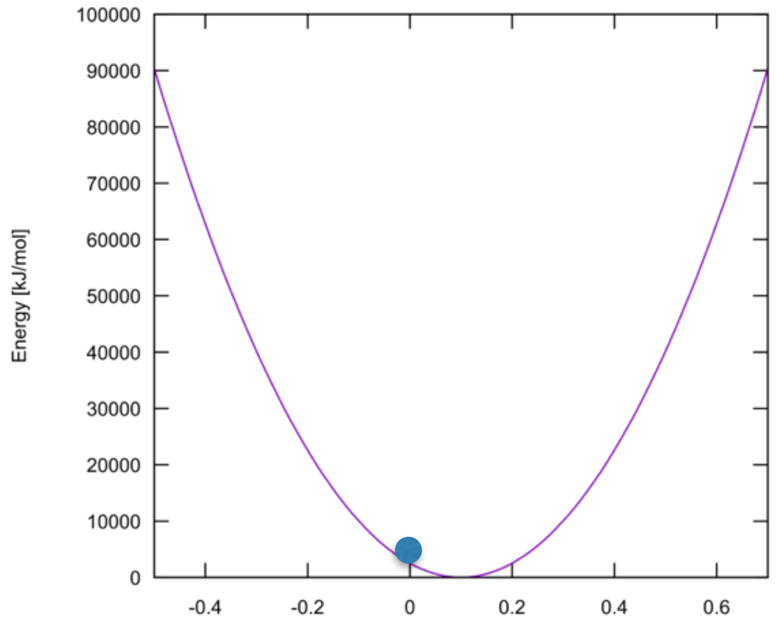
$$V(r) = \frac{1}{2}k(r - r_0)^2$$

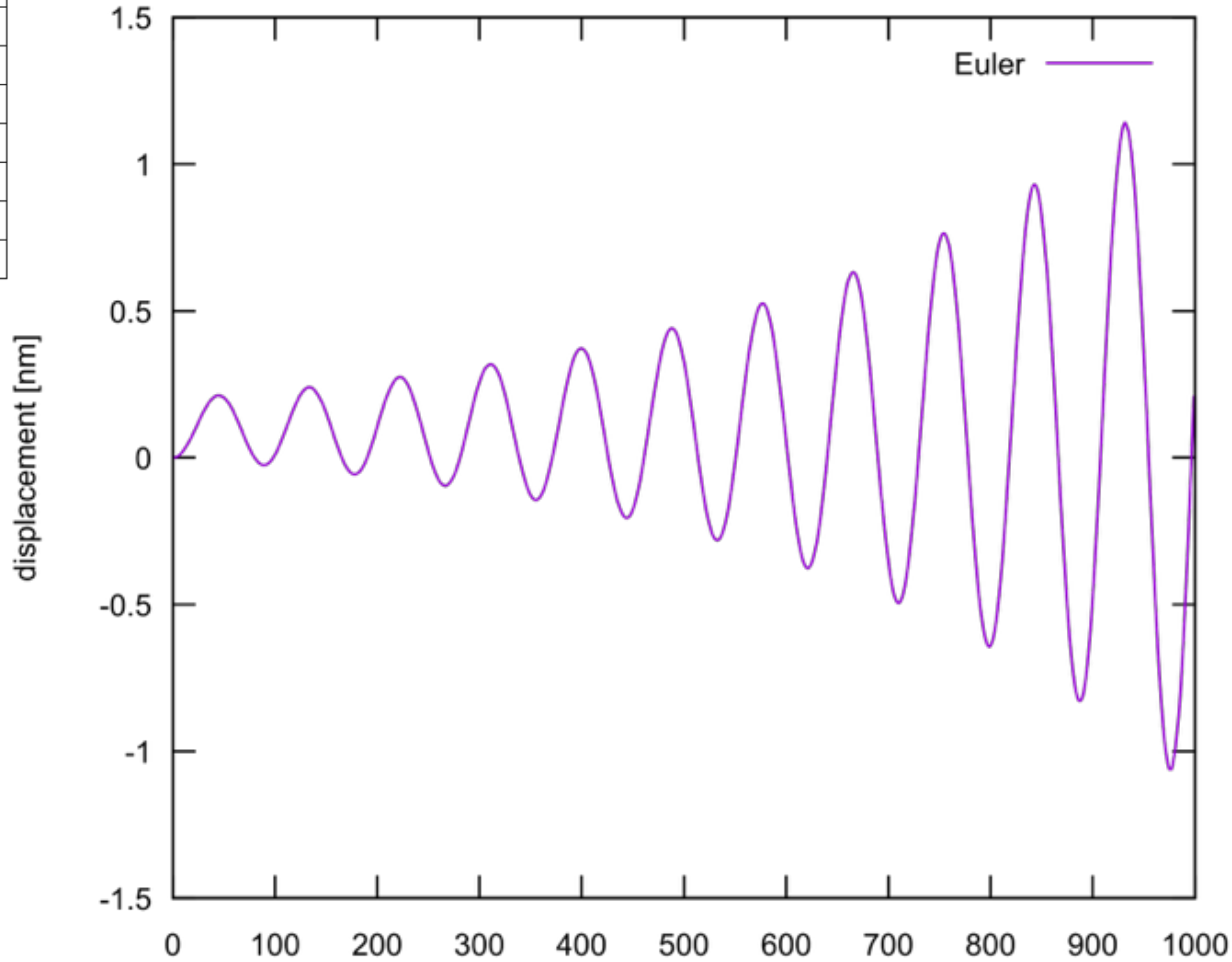
$$\begin{aligned} r_0 &= 0.1 \text{ nm} \\ k &= 502080 \text{ kJ/(mol nm}^2\text{)} \\ T_p &\approx 10 \text{ fs} \end{aligned}$$

$$\begin{aligned} v(0) &= 0 \\ x(0) &= 0 \end{aligned}$$

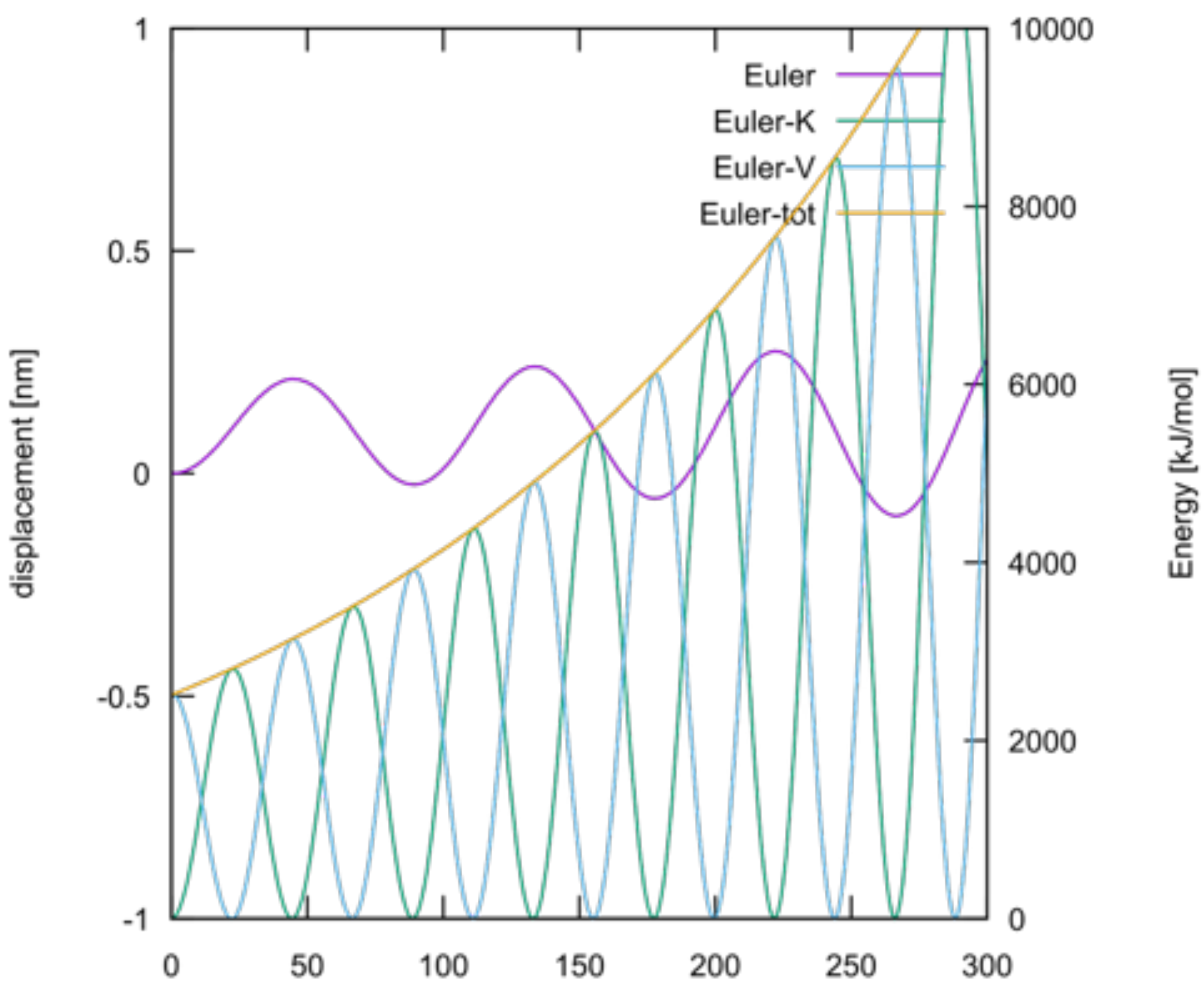
$$\begin{aligned} r(t + \Delta t) &= r(t) + v(t)\Delta t \\ v(t + \Delta t) &= v(t) + a(t)\Delta t \end{aligned}$$

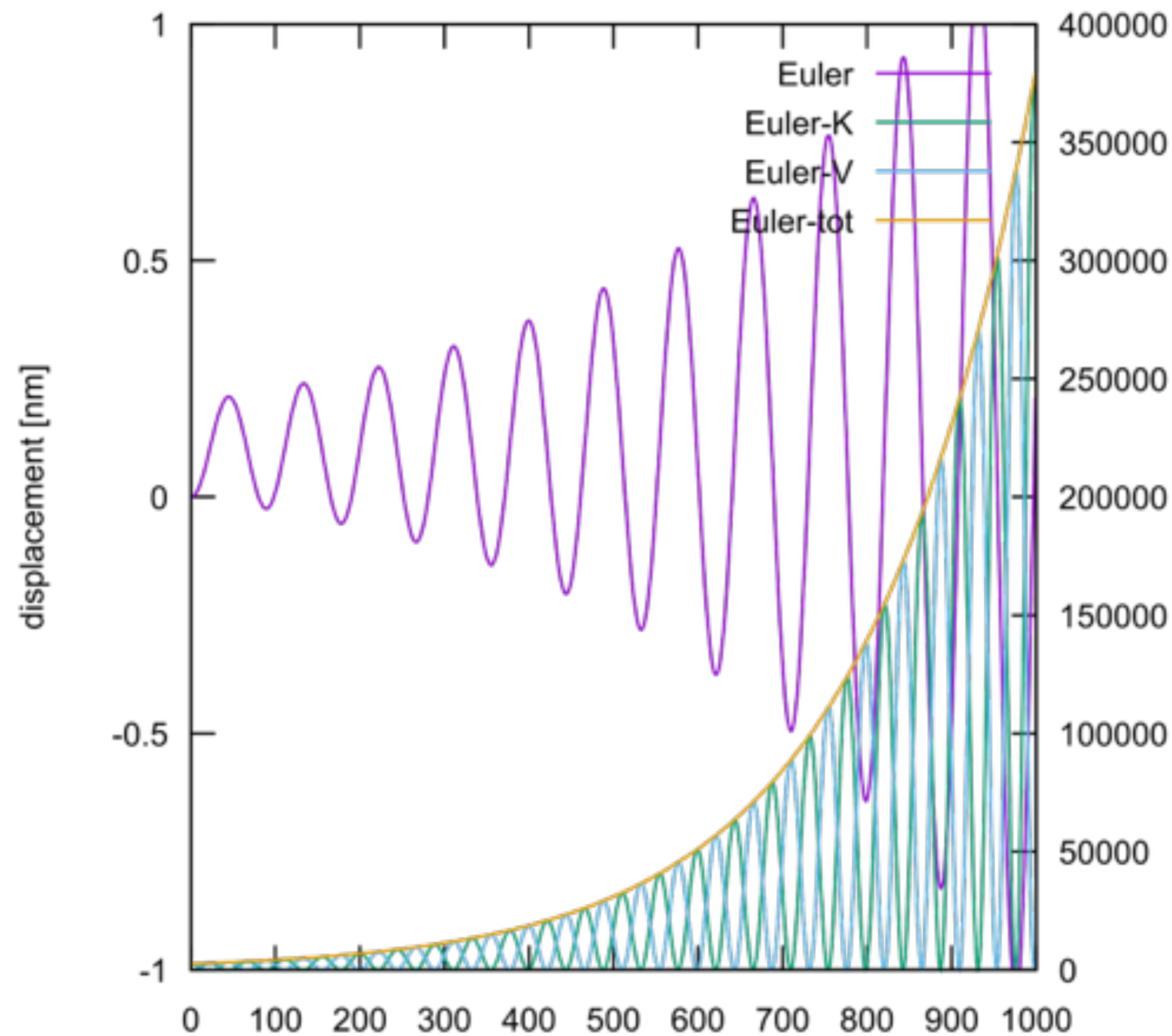
Particle in a well - Euler algorithm





Euler algorithm - energy



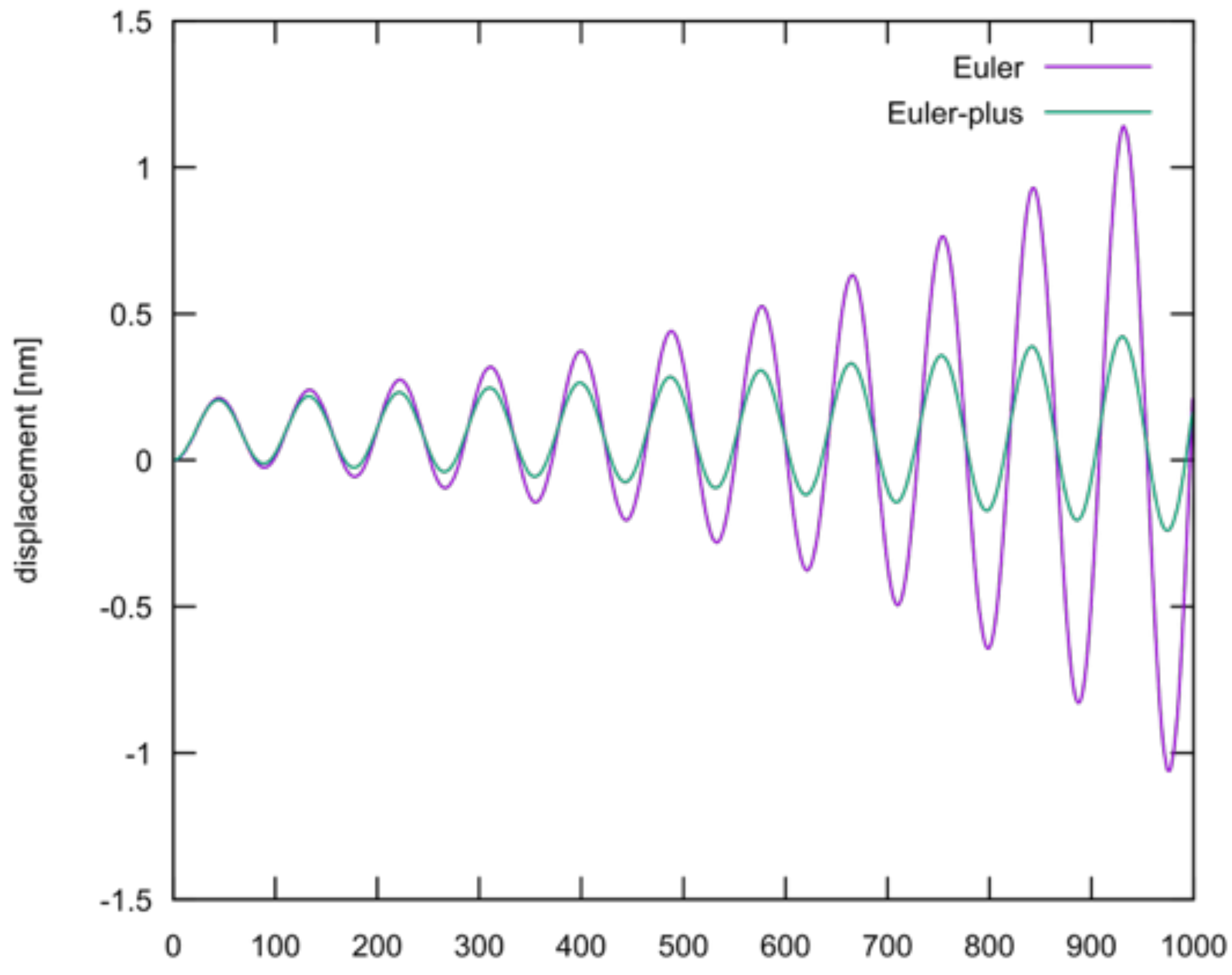


Energy not conserved

Time reversible?

Euler-plus

$$
\begin{aligned}
r(t+\Delta t) &= r(t) + v(t)\Delta t + \frac{1}{2}a(t)\Delta t^2 \\
v(t+\Delta t) &= v(t) + a(t)\Delta t
\end{aligned}
$$



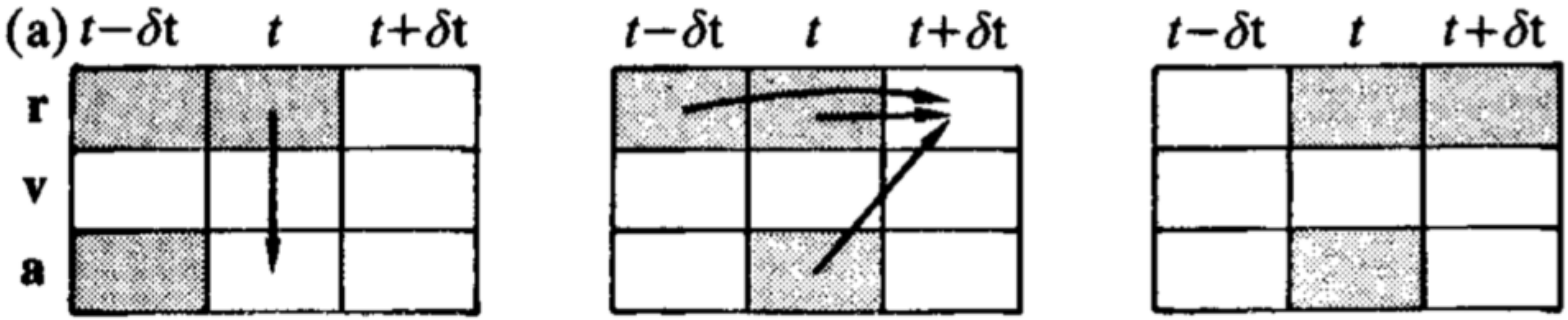
Energy not conserved

Not time-reversible

Verlet algorithm

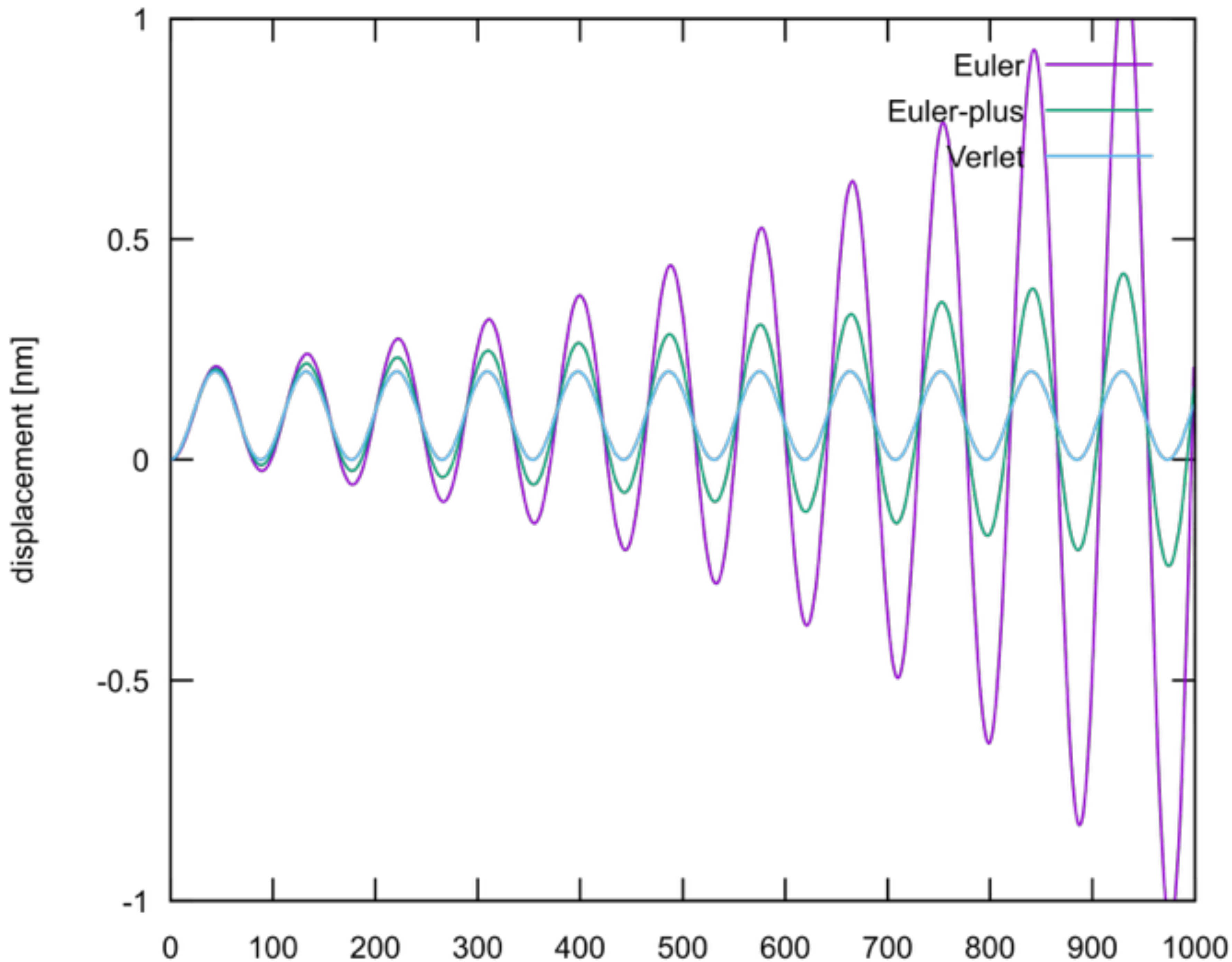
$$r(t+\Delta t) = 2r(t) - r(t-\Delta t) + a(t)\Delta t^2$$

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

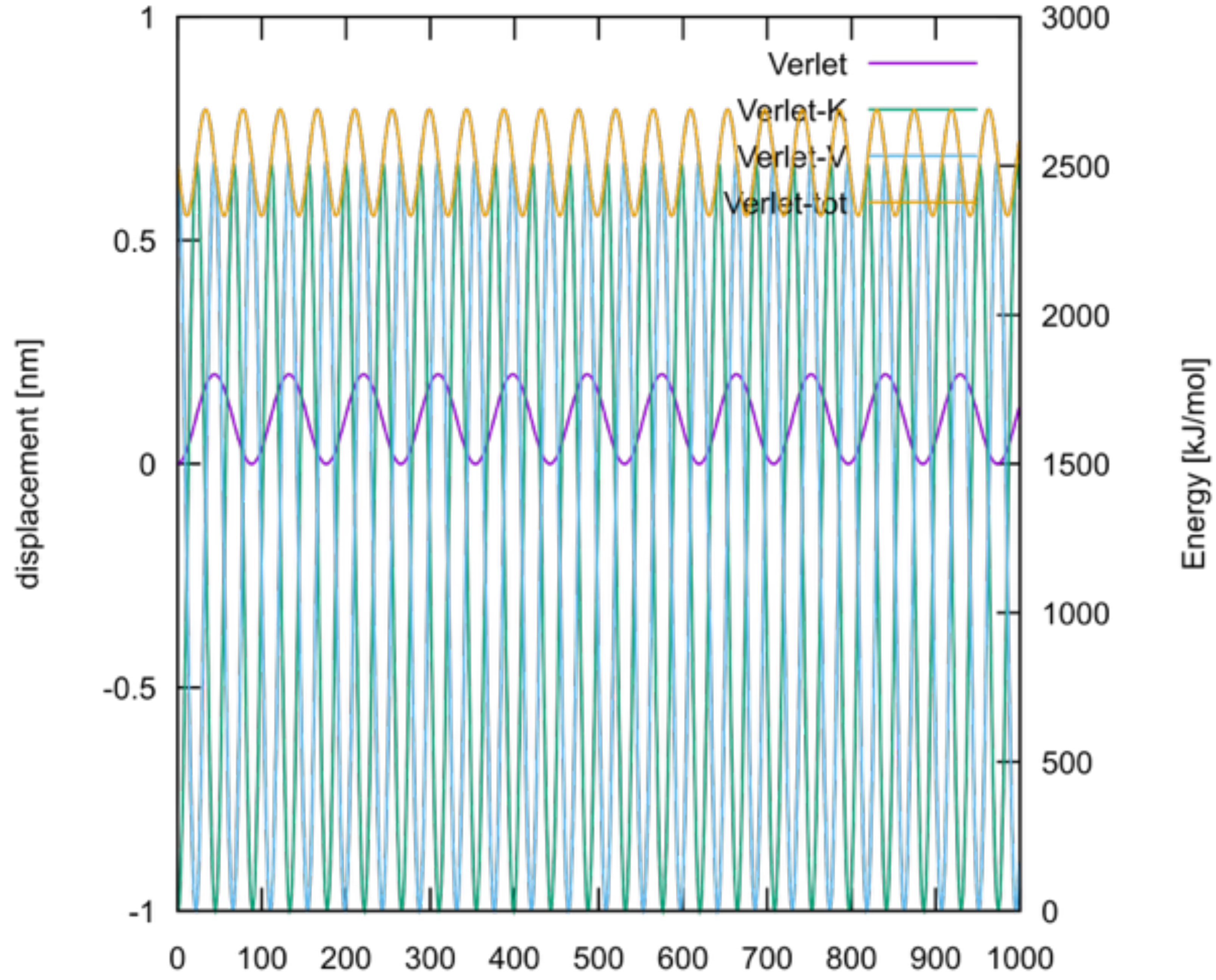


Time reversible?

Verlet algorithm



Verlet - energy



Time reversible

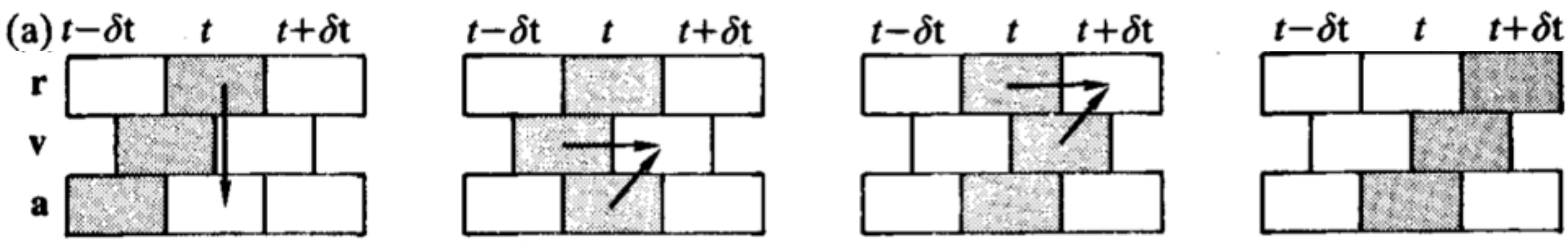
Energy (approx.) conserved

Numerical issues

Flavors of Verlet: leap-frog

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

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



Equivalent Verlet

Time reversible

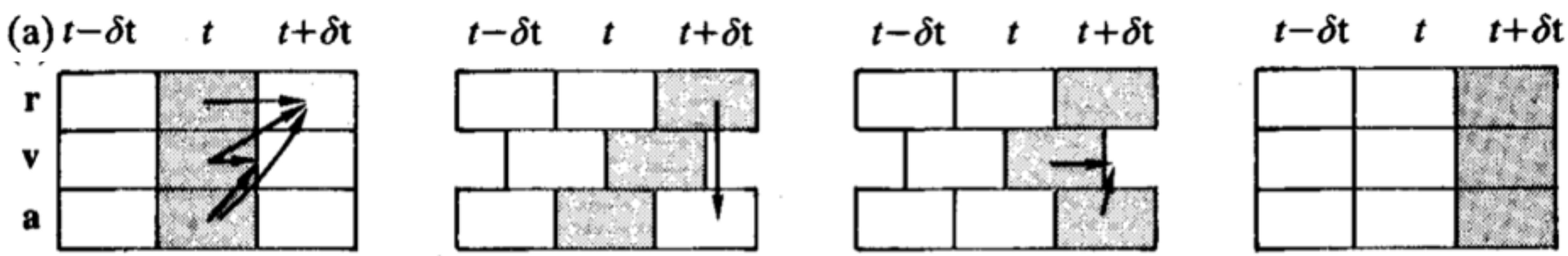
Numerically more accurate

BUT: velocities at different times than positions

Velocity Verlet

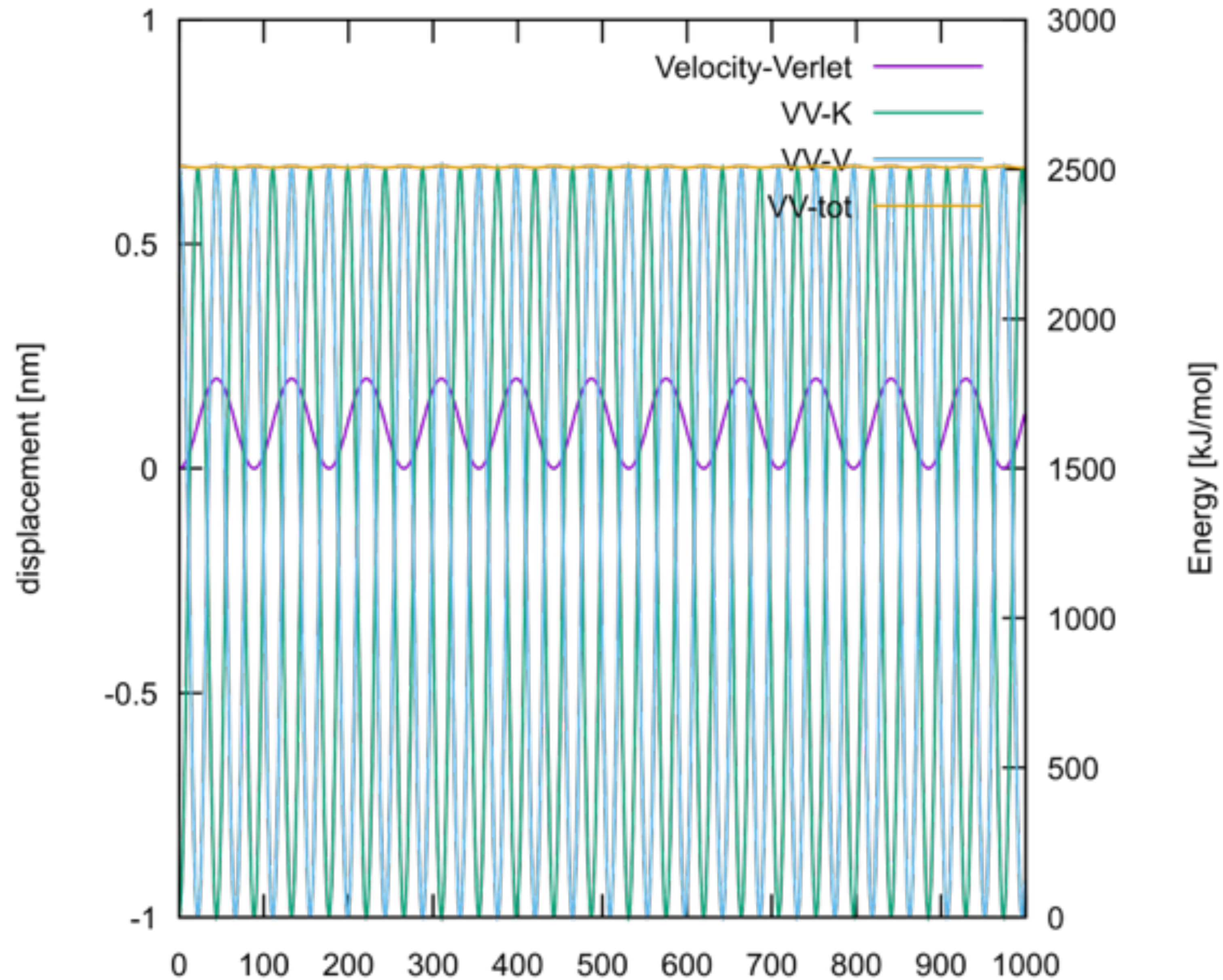
$$r(t+\Delta t) \quad = \quad r(t) + v(t)\cdot\Delta t + \frac{1}{2}a(t)\cdot\Delta t^2$$

$$v(t+\Delta t) \quad = \quad v(t) + \frac{1}{2}(a(t) + a(t+\Delta t))\Delta t$$



Time reversible

Velocity Verlet - energy



Equivalent Verlet
Energy conserved
Time reversible
Numerically stable
Velocities updates half-stepwise

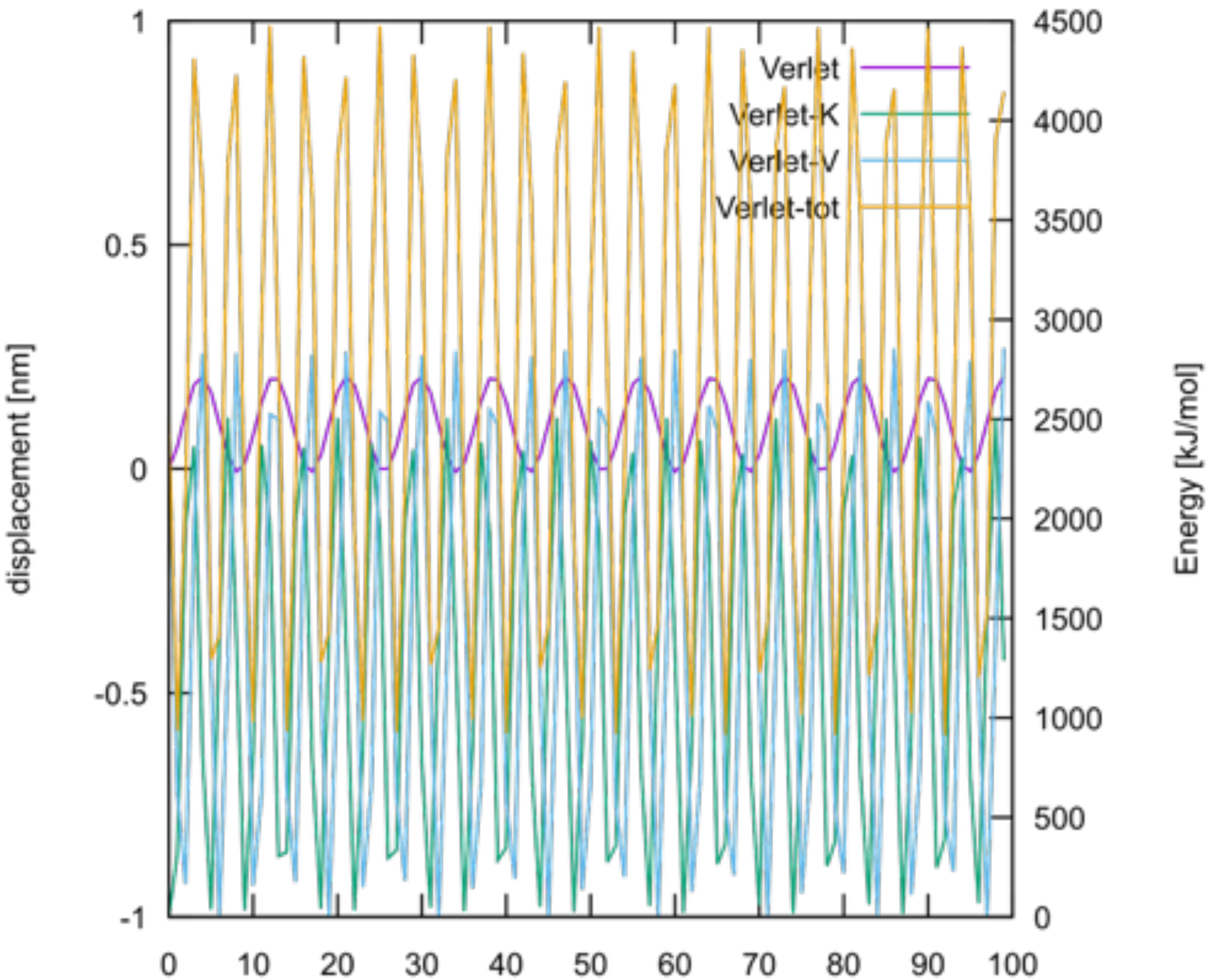
Increase the time step

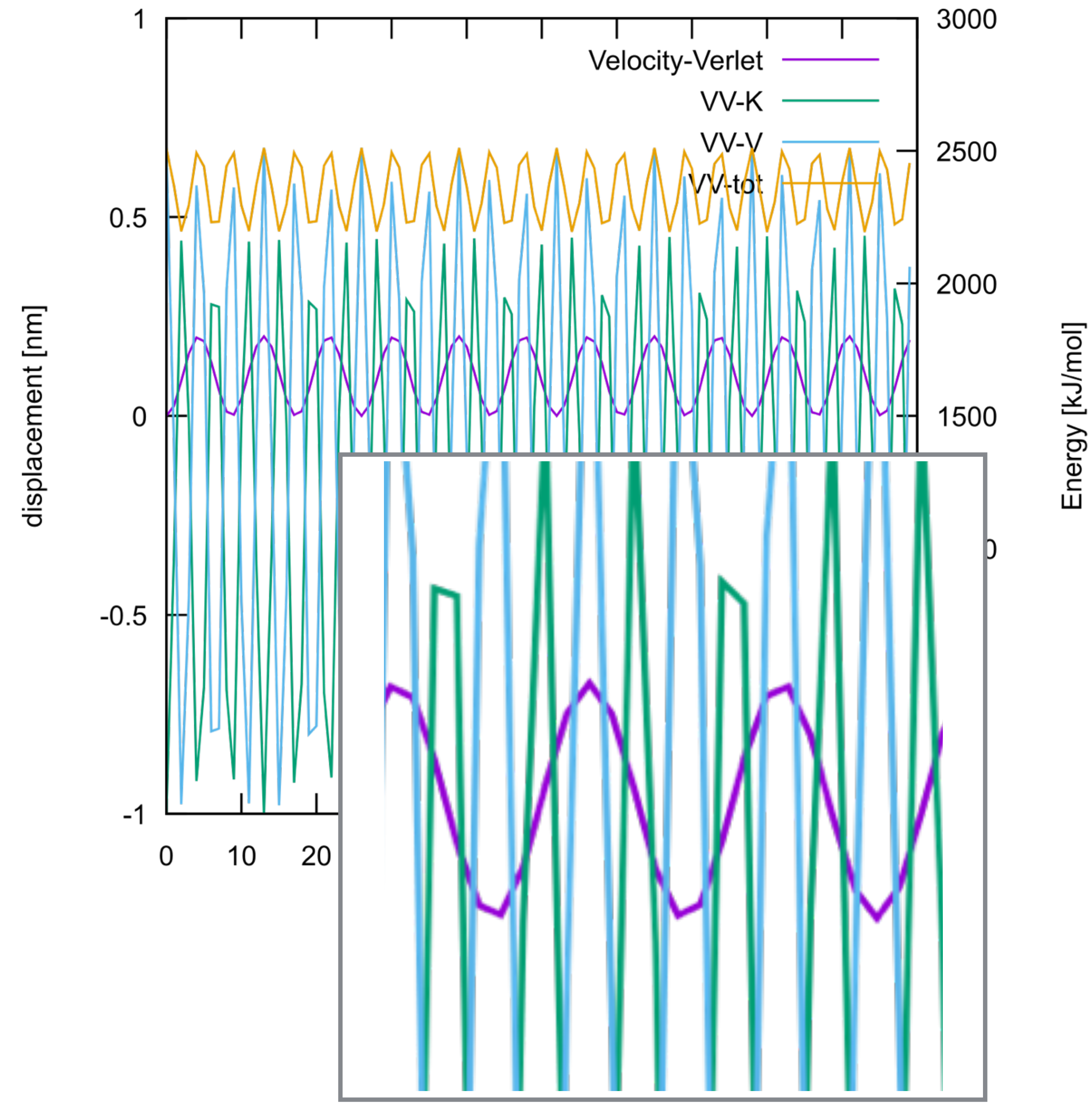
- Until now: $\Delta t \;=\; 10^{-16}$ s
- 100 steps/ period
- What if we increase the time step?

Verlet vs. Velocity-Verlet

Increase timestep by factor 10: $\Delta t \;=\; 10^{-15}\ \mathrm{s}$

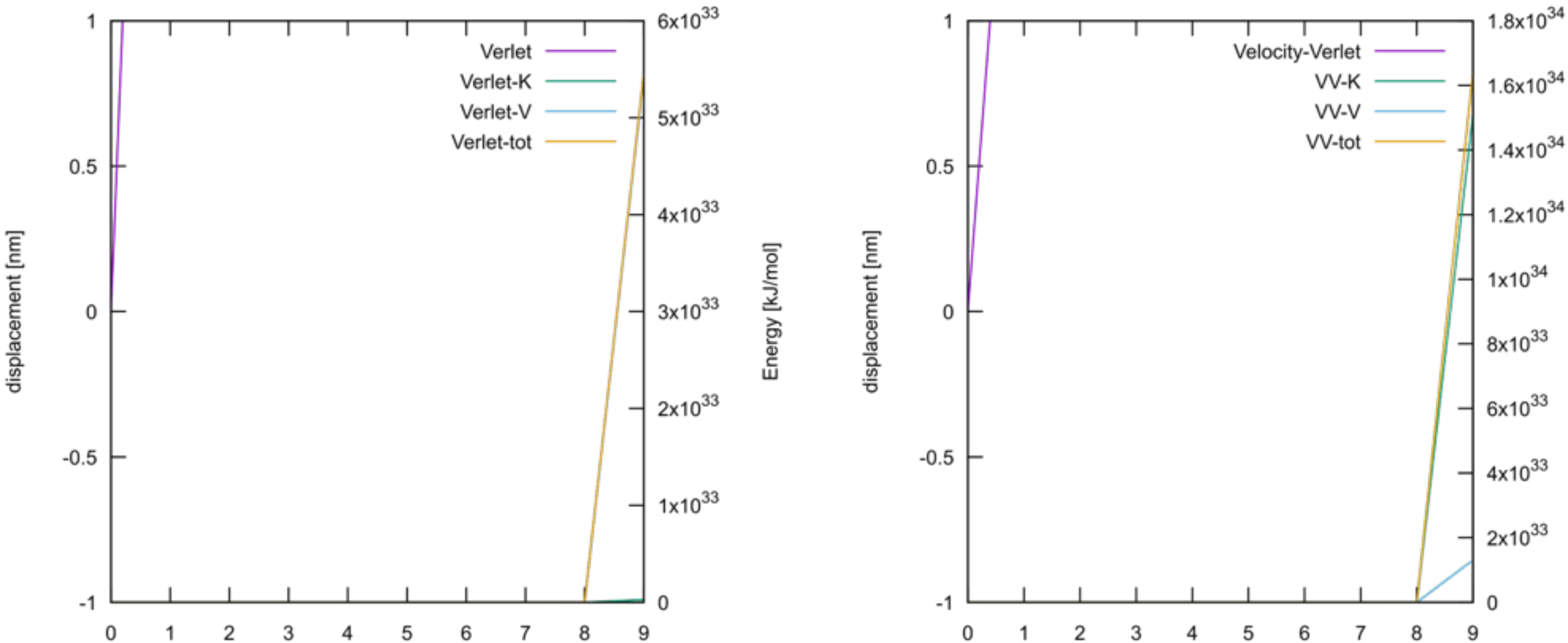
10 steps per period





Verlet vs. Velocity-Verlet

Increase timestep by factor 100: $\Delta t \;=\; 10^{-14}\,\mathrm{s} \approx T_p$



Stability condition for Verlet

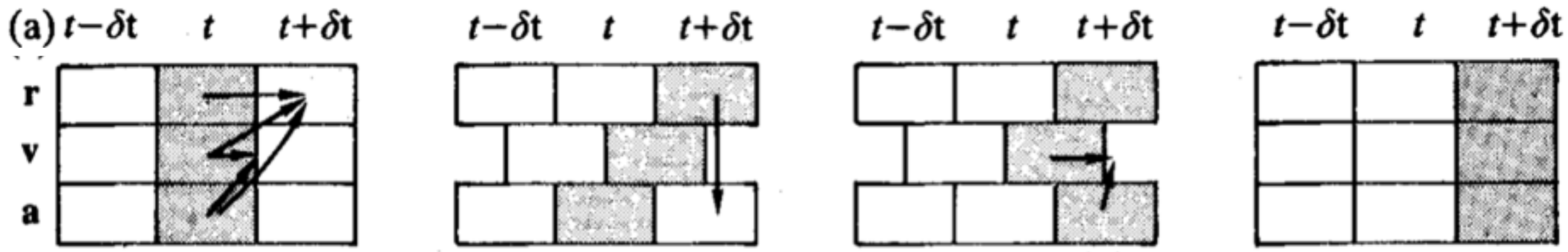
$$\Delta t \quad < \quad T_p/\pi$$

$$T_p(\text{stretch}) \quad < \quad T_p(\text{bend})$$
$$T_p(\text{light atoms}) \quad < \quad T_p(\text{heavy atoms})$$

Vibrational Mode[a]	Wave number $(1/\lambda)$ [cm^{-1}]	Period T_p (λ/c) [fs][b]	T_p/π [fs]
O–H, N–H stretch	3200-3600	9.8	3.1
C–H stretch	3000	11.1	3.5
O–C–O asymm. stretch	2400	13.9	4.5
C≡C, C≡N stretch	2100	15.9	5.1
C=O (carbonyl) stretch	1700	19.6	6.2
C=C stretch			
H–O–H bend	1600	20.8	6.4
C–N–H, H–N–H bend	1500	22.2	7.1
C=C (aromatic) stretch			
C–N stretch (amines)	1250	26.2	8.4
Water Libration (rocking)	800	41.7	13
O–C–O bending	700	47.6	15
C=C–H bending (alkenes)			
C=C–H bending (aromatic)			

Generally 1-2 fs

Velocity-Verlet




```
// Read initial positions
read_positions
// Set random velocities
randomize_velocities

// Compute initial neighbor list
 compute_list
// compute initial forces
compute_forces

for(istep=0;istep<nstep;istep++){

    thermostat
    velocity += (force*dt/2)/mass
    position += velocity*dt

    // Check whether the neighbour list has to be recomputed
    check_list

    if(recompute_list){
      compute_list
    }

    compute_forces

    velocity += (force*dt/2)/mass
    thermostat
}
```

$$r(t+\Delta t) = r(t) + v(t)\cdot\Delta t + \frac{1}{2}a(t)\cdot\Delta t^2$$

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