1 Simulation Exercise

The **Euler** algorithm shall be applied to the one dimensional classical harmonic oscillator as well as the velocity-**Verlet** to compare their simulation results.

1.1 One dimensional harmonic oscillator

When the harmonic system is replaced from its equilibrium a restoring force is acting on it to get it back to its equilibrium position. The restoring force depends on the displacement and can be described with a positive constant k as

$$m\mathbf{a} = \mathbf{F},$$
$$m\frac{d^2\mathbf{x}}{dt^2} = -k\mathbf{x},$$

where $\mathbf{F}$ is the force acting on the point particle with mass m, and $\mathbf{a} = d^2\mathbf{x}/dt^2$ is the acceleration. This second order differential equation can be solved analytically,

$$x(t) = A\cos(wt + \phi), \tag{1}$$

with the constant amplitude A. The frequency $\omega = 2\pi/T$ with $T = \sqrt{m/k}$ depends on the mass of the system and on the strength of the restoring force k. Given the two initial conditions $x_0 = x(t = 0)$ and $v_0 = v(t = 0)$, the expression for the periodic motion can be obtained uniquely.

1.2 Simulation model

The quality of the molecular dynamic approximation is evaluated by four criteria.

1. **Accuracy**
   The simulated trajectory should obey the equation of motion to good approximation.
2. **Efficiency**
   The algorithm should be simple, fast and use as less as possible memory.
3. **Conservation**
   The quantities which are conserved should remain conserved after applying the method.
4. **Stability**
   The algorithm should not show an energy drift.

1.3 Euler algorithm

In the following, the Euler algorithm is used to simulate the one dimensional classical harmonic oscillator. The Euler algorithm is a numerical method for solving ordinary differential equations with a given initial value and is the first order Runge-Kutta method. The Euler algorithm computes the position $x(t)$ and velocity $v(t)$ step by step via a polygonal curve. We use the definition of the derivative and neglect terms of order higher than linear:

$$\frac{x(t+\Delta t) - x(x)}{\Delta t} = v(t),$$
$$x(t+\Delta t) = x(t) + v(t)\Delta t.$$

Hence, the following two equations are obtained from the the equation of motion

$$\boxed{\begin{aligned} x(t+\Delta t) &= x(t) + v(t)\Delta t, \\ v(t+\Delta t) &= v(t) + \frac{1}{m}F(t)\Delta t. \end{aligned}}$$

Starting with the two initial values x_0 and v_0, the position $x(t)$ can be computed step by step. The quality of the approximation depends strongly on the size of the step Δt.

1.4 Velocity-Verlet algorithm

Within the velocity-Verlet algorithm the solution of the harmonic equations of motion is given by

$$\boxed{\begin{aligned} x(t+\Delta t) &= x(t) + v(t)\Delta t + \frac{1}{2}F(t)\Delta t^2, \\ v(t+\Delta t) &= v(t) + \frac{1}{2m}(F(t) + F(t+\Delta t))\Delta t. \end{aligned}}$$

2 Task

2.1 Implementation in CPP

1. Implement the Euler and velocity-Verlet algorithms for the harmonic oscillator.
2. Choose the units to $m = 1$, $k = 1$.
3. Initial position and velocity are $x(0) = 0$ and $v(0) = 1$.
4. Run the simulations for both algorithms using $\Delta t = 0.1, 0.01$ for $j = 1, \ldots, [10000/\Delta t]$ steps.

2.2 Plotting in Python with Matplotlib

1. Plot $x(j\Delta t)$ and compare with $\sin(j\Delta t)$ for both algorithms.
2. Plot $x(t)$, $v(t)$ and the total energy $E = mv^2(t)/2 + V(x(t))$, where $V(x) = kx^2/2$, and compare to the analytic solution $x(t) = \sin(t)$ and $E = 1/2$ for both algorithms.

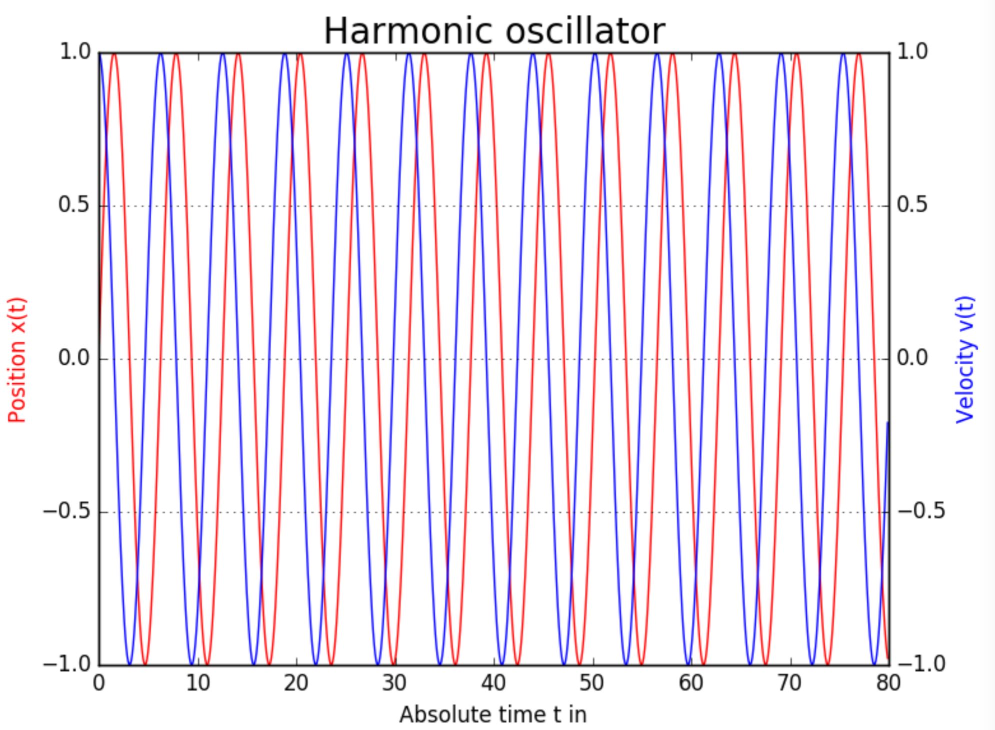

Figure 1: Example plot for the analytic solution of the harmonic oscillator: Position $x(t)$ and velocity $v(t)$.

2.3 Analysis

1. Explain why the energy is not exactly constant, as it should be according to classical mechanics.
2. Discuss the differences between the Euler and the velocity-Verlet implementation based on its algorithms and the obtained simulation results.