Computational Physics (SS 2020)

Prof. Dr. Riccardo Mazzarello

Sheet 2 (20 points)
Submission: 23 April 2020, 10:00 a.m.

Remarks

- The exercise sheets can be solved in teams of two students.
- Provide a single PDF-file containing the written solutions of all exercises, i.e. analytical derivations, numeric results, their interpretation, etc. If it contains a scan, please make sure it is well readable. Plots can either be embedded into this file or, alternatively, given separately (e.g. as PNG-files) and referred to in the PDF.
- Use a current version of Python 3 for your implementations.
- Please hand in your solutions (PDF-file with answers, source code files, and, if not embedded in the PDF, PNG-files of plots) as a **single** .zip-file named

  Surname1-MatNum1-Surname2-MatNum2-SheetX.zip,

  where MatNum is your matriculation number, by uploading it to RWTH Gigamove (`https://gigamove.rz.rwth-aachen.de`). Please send the download link using your RWTH address to

  brancaccio@physik.rwth-aachen.de,

  and write in the object of the email

  CP SheetX - Surname1-MatNum1-Surname2-MatNum2.

1. Euler-Cromer method (8 points)

Write a program that calculates the trajectory of the 1D harmonic oscillator, starting with a maximum displacement $x = x_{\mathrm{max}}$ as the initial configuration. Compare the results from using

(a) the Euler method

(b) the velocity Verlet method

(c) the Euler-Cromer method.

The Euler-Cromer method is defined by the following scheme:

$$
\begin{aligned}
v(n+1) &= v(n) + \tau a(n) \\
x(n+1) &= x(n) + \tau v(n+1)
\end{aligned}
$$

Compute the total energy as well. Why does the Euler-Cromer method perform better than the Euler method?

2. Ideal Pendulum (7 points)

Write a program that calculates the trajectory of the ideal pendulum using the velocity Verlet method. The deflection $x \in [-\pi, \pi)$ of the pendulum shall evolve by

$$
\ddot{x} = -\frac{g}{l} \sin(x)
$$

with the initial configuration $(x, \dot{x}) = (x_0, v_0)$. Determine the period T of oscillations as a function of maximal deflection of $x_{\text{max}} \in (0, \pi)$ for initial configrations $(x, \dot{x}) = (x_{\text{max}}, 0)$. Make a plot of T vs. x_{max}. Compare this dependence $T(t_{\text{max}})$ with the harmonic approximation for small values of x_{max}.

3. Leap-frog scheme with friction (5 points)

In the lecture we derived a Verlet algorithm for a specific velocity dependent acceleration, namely for a linear dependence of the acceleration on the velocities:

$$
\vec{a}(\vec{x}, \vec{v}, t) = \vec{a}^0(\vec{x}, t) - \Gamma \vec{v}
$$

Derive a leap-frog version of the same algorithm.