In-Class Work: Molecular Dynamics Simulation

1. Numerical Integration

I will first give a short introduction.

1a. For $f(t, y) = Ax$ write a program which uses the Euler-step for integration (and therefore use the flowchart of the introduction) to integrate

$$\frac{dx}{dt} = f(t, x) = Ax$$

Use $A = 0.3, t_0 = 0.0, x_0 = 0.6, \Delta t = 0.2, n_{\max} = 100$ (that means $t_{\max} = 20.0$). Print t and $x(t)$ for every time-step. Save the data in a file, e.g. `out1sim0.2.dat`. To help you with the writing the data into a file of specified name, you may start with copying the following program into your working directory:
`~kvollmay/share.dir/inclass2025.dir/md1_start.py`

1b. What do you expect for $x(t)$ (we can solve the DE analytically).

1c. Add to your program that the exact solution for $t, x(t)$ is printed for every time-step into another file, e.g. named `out1theory.dat` Look at the comparison of the numerical solution and the theoretical solution with
`xmgrace out1theory.dat out1sim0.2.dat`
and if you would also like to use a logarithmic vertical axis and also set the x-range and y-range (and also show the legend) then you may use
`xmgrace out1theory.dat out1sim0.2.dat  -log y -world 0 0.2 20 250 -legend load`
When you get to this, I can also show you how to set the y-axis and x-axis settings within xmgrace.

1d. Now rerun the program for $\Delta t = 0.1$ and adjust $n_{\max}$ to get the same $t_{\max} = 20.0$ and print into another file, e.g. `out1sim0.1.dat`. Then rerun the program again this time for $\Delta t = 0.01$ and $n_{\max}$ again adjusted. Look at your data with
`xmgrace out1theory.dat out1sim0.*.dat -log y -world 0 0.2 20 250 -legend load`
When you got this, please get me, I will remind you about a few tools with `xmgrace` I summarize here in footnote. [1]

2. Newton's Second Law

2a. Make sure to get me, before you continue. What are the Euler step updates for $x(t+\Delta t)$ and for $v_x(t + \Delta t)$?

2b. Numerically integrate for $F_x^{\text{net}} = -mg$. Use $g = 9.8, \Delta t = 0.2, t_{\max} = 20.0, x_0 = 5.0, v_{x0} = 2.3$. Print into a file $t, x(t), v_x(t)$. As above, also determine the analytical solution and rerun the numerical solution also for $\Delta t = 0.1$ and $\Delta t = 0.01$. Look at your comparison using `xmgrace`

[1] To save xmgrace session: `File` → `Save as` and in Selection entry give filename, for example `md1dfig.xmgr` To save eps-file `File` → `Print setup` then change `Postscript` to `EPS`. In case you had previously used Save as the eps-filename is already suggested and then click `Accept`. To get the eps-file printed use `File` → `Print`

3. Harmonic Oscillator & Surprise

3a. Numerically integrate this time for the case of an harmonic oscillator, so $F_x^{\rm net} = -kx$. We can also analytically solve this equation. Let's choose $t_0 = 0.0, x_0 = 5.0, v_{x0} = 0.0$, then the theoretical solution is

$$x(t) = x_0 \cos(\omega_0 t) \qquad\qquad v_x(t) = -x_0\omega_0 \sin(\omega_0 t)$$

where $\omega_0 = \sqrt{(k/m)}$. So we know the period $T = 2\pi/\omega_0$. Let's choose $\Delta t = T/n_{\rm div}$. Integrate $F_x^{\rm net} = -kx$ for $k = m = 1$ and for $n_{\rm div} = 100$ and do $n_{\rm max} = 10n_{\rm div}$ MD steps. Print also the analytical solution and compare. **Note: Before you update $x(t)$, you need to copy the value of $x(t)$ into a temporary variable for example `xold=x` only then you can update x and then v. For v you need to use `xold`.** Try also with $n_{\rm div} = 1000$. What happens?

3b. Also determine the theoretical and numerical total energy

$$E_{\rm tot} = \frac{1}{2}kx^2 + \frac{1}{2}mv_x^2$$

as function of time, so $E_{\rm tot}(t)$. If you print for example into `out3sim100.dat` $t, x, v_x, E_{\rm tot}$ and similarly into the other files, you can compare the $E(t)$ results with

```
xmgrace -block out3theory.dat -bxy 1:4 -block out3sim100.dat -bxy 1:4 -block out3sim1000.dat  -bxy 1:4
```

Get me, when you have the results.

4. Euler-Cromer

Read in the Gould et al. book the first page of chapter 3. Change your program from 3b to use the Euler-Cromer step instead of the Euler step. Repeat the integration and compare again with the theoretical solution.