MVCOMP1, Ex.6

To be submitted by November 29, 2024, 6PM

Particle-mesh and Fourier methods *(24 points)*

Download the toy code NGP_CIC_Fourier.py. The provided code template is incomplete: the exercise consists in adding the missing pieces. Places where you have to complete the code are marked with the comment 'TO BE FILLED IN'.

The code does the following calculations.

- First, it generates an astrophysical N-body system in two dimensions (x and y). There are two options: a Plummer model (often used in astrophysics) and a uniform distribution, which can be chosen with the boolean variable plummer=True/False. In both cases the particles have individual mass $m_i = 1$ for $i = 0, 1, N - 1$. The number N, set to 100000 in the example, can be modified. The only force is gravity.
- Then, the toy code calculates the density on a square mesh of Np X Np points with the nearest grid point (NGP) or the cloud-in-cell (CIC) method. Note that the calculation with the CIC method is left blank.
- Then, it calculates the potential on the mesh by using a Fast Fourier Transform method to solve Poisson's equation. Note that also in this case the code is mostly left blank.

The exercise consists in adding the following features to the code.

(a) Add the calculation of the density on the mesh with the CIC technique. **(10 points)**

(b) Plot the density you calculated with the CIC technique using matplotlib.pyplot.imshow (see the NGP function for an example). Do you see differences between the NGP and the CIC calculation? Why? Please, provide a short explanation. **(4 points)**

(c) Calculate the potential on the mesh with the Fast Fourier Transform method. Assume $G = 1$. Hints: use np.fft.fft2 and then np.fft.ifft2 to go back to the real space. Be careful not to divide by zero for kx=0 and ky=0. **(8 points)**

(d) Plot the potential using matplotlib.pyplot.imshow. **(2 points)**