Getting familiar with a PIC code

Elisabetta Boella, Giovanni Lapenta

May 20, 2016

The following exercises have been thought such that you can get familiar with using a particle in cell code. In particular several scripts in Python are provided, which implement the PIC algorithm and were written to study electrostatic phenomena. You are asked to look at the scripts, change the input parameters and/or the code and analyse the results, as computational physicists normally do. To run the scripts in a terminal go to the folder where the scripts are (`cd path/to/folder`) and type `python name_of_the_script`. Moreover, you will be given a physical problem and you will be asked to design a simulation to study it.

1 Two-stream instability

The program `twostream_classic.py` is a 1D electrostatic PIC code written in Python to study the non relativistic two-stream instability in the case of two cold counter-propagating electron beams streaming on a background of fixed plasma ions. The beams have equal densities $n_{e,1} = n_{e,2} = \frac{1}{2}n_e$ and opposite velocities $v_{e,1} = -v_{e,2} = v_0$. The instability is triggered by a small perturbation on the particle initial positions and/or by a small thermal spread. The dispersion relation for the two-stream instability in this case is:

$$D(k,\omega) = 1 - \frac{\omega_{pe}^2}{2}\left[\frac{1}{(\omega - kv_0)^2} + \frac{1}{(\omega + kv_0)^2}\right], \tag{1}$$

where $\omega_{pe} = \sqrt{4\pi e^2 n_e/m_e}$ is the electron plasma frequency, e the elementary charge, m_e the electron mass and ω and k the wave frequency and wavenumber respectively. If one sets the dispersion relation (1) equal to zero finds a quartic equation, whose solutions are

$$\omega(k) = \pm\sqrt{\frac{2k^2v_0^2 + \omega_{pe}^2 \pm \left[\omega_{pe}^2\left(8k^2v_0^2 + \omega_{pe}^2\right)\right]^{1/2}}{2}}. \tag{2}$$

Solutions (2) have an imaginary part only when

$$kv_0 < \omega_{pe}. \tag{3}$$

The most unstable solution will have a negative imaginary part and deriving this latter respect to the wavenumber k allows for defining the maximum growth rate, which is given by:

$$\Gamma_{\max} = \frac{\omega_{pe}}{2\sqrt{2}}, \tag{4}$$

corresponding to

$$k_{\max} = \frac{\sqrt{3}\omega_{pe}}{2\sqrt{2}v_0}. \tag{5}$$

Note also that condition (3) sets a constraint on minimum unstable length of the system:

$$\frac{\omega_{pe}L}{v_0} > 2\pi. \tag{6}$$

Therefore if we want to study the instability with `twostream_classic.py`, we need to define the following simulation parameters:

L	$\gg$	$2\pi v_0/\omega_{pe}$ & $2\pi/k$	length of the simulation box
Δx	$\ll$	smallest scale length	cell size
Δt	$\ll$	$1/\omega_{pe}$	time step
t_{end}	$\gg$	$1/\Gamma$	final simulation time

1.1 Tasks

1. Considering $\omega_{pe} = 1$ (`WP` in the code) and $v_0 = 1$ (`V0`), check that the values of `L`, `NG`, `DT`, `NT` (simulation box length, number of grid points, time step and number of simulation steps) set in the Python script allows to observe the instability. Check also the way used to trigger the instability, in particular the variable `mode`. Are there other ways to trigger the instability?

2. Perform a convergence study: vary one by one the number of particles, the number of grid points, the simulation time step and the box size (you can either increase or decrease these values) and run the simulations. Does the simulation get slower? Are the results affected? Which simulations can you trust?

3. Compare the numerical growth rate provided by the simulations with the theoretical one (you can use the script `omega_vs_k.py`): are they matching? (Hint: open the data of the electric field energy "cl_field_ene.txt" using `read_txt.py`, interpolate the values in the linear phase of the instability with a line and compute the line slope. Pay attention that the electric field energy is proportional to the square of the electric field).

4. Comment the line `xp+=XP1*np.cos(2*np.pi*mode/L*xp)` in the code (e.g. ignore the perturbation on the particle position) and change the thermal velocity value v_{th} to `VT=0.001`. What do you observe? Compute a Fourier transform of the electric field (Use the commands `scipy.fft` and `scipy.fftpack.fftfreq`) and the growth rate. Which mode is growing faster? Do the values match with the theory?

5. Consider one of the electron beams. Compute their average velocity at $t = 0$ and at $t = t^*$ when the instability saturates. Compute also their variance at $t = 0$ and at $t = t^*$. What do you observe?

6. Remove the perturbation on the initial particle positions and initialise the beams of particles with an increasing thermal spread (`VT`). What do you observe?

7. Modify the code to simulate the stream of an electron beam in a "static" plasma. Consider a beam with mass density $n_b = 0.1n_e$, velocity v_b and thermal velocity $v_{th} \ll v_b$ (v_{th} alone should be enough to start the instability, comment the position displacement for this task). Make sure to guarantee charge and current neutrality at $t = 0$ (e.g. choose the plasma electron density n_e such that $n_b + n_e = n_p$, with n_p ion plasma density, and the plasma electron velocity v_e such that $v_e = -n_b v_b/n_e$). Repeat points 3 and 4. Note that, assuming $n_b \ll n_e$ and $v_e \simeq 0$, the dispersion relation reduces to:

   $$D(k,\omega) = 1 - \left(\frac{\omega_{pe}}{\omega}\right)^2 - \left(\frac{\omega_{pb}}{\omega - kv_b}\right)^2, \tag{7}$$

   with $\omega_{pb} = \sqrt{(4\pi e^2 n_b/m_e)}$. Defining the quantity $\alpha = n_b/n_e$, the maximum growth rate is given by:

   $$\Gamma_{\max} = \frac{\sqrt{3}}{2}\left(\frac{\alpha}{2}\right)^{1/3}\omega_{pe} \tag{8}$$

   and occurs at the resonance $kv_b \simeq \omega_{pe}$.

2 Relativistic two-stream instability I

Recently, it has been pointed out that two identical systems evolving one according to Newton's laws and the other according to special relativity, relax to equilibrium in different ways [Lapenta *et al.*, Astrophys. J. 666, 949 (2007)].

Consider a beam of electrons counter-streaming with a beam of positrons and verify the conclusions of the paper. The beams have the same density n and opposite drift velocities $v_p = -v_e$, where the subscripts p and e stay for positrons and electrons respectively.

2.1 Tasks

1. The code `twostream_e+e-_classic.py` was written to study the classical two-stream instability in the electron-positron case. Compare it with `twostream_classic.py` and check the differences. In `twostream_e+e-_classic.py`, is there an ion background?

2. Compare the scripts `twostream_classic.py` and `twostream_relativistic.py` to see what are the differences.

3. Having in mind `twostream_e+e-classic.py` and `twostream_relativistic.py`, write your own script `twostream_e+e-_relativistic.py` to compute the relativistic two-stream instability in the electron-positron case.

4. After testing `twostream_e+e-_relativistic.py`, repeat point 3 of section 1 and compare the results with the theory. Pay attention that in the relativistic case the dispersion relation becomes:

$$D(k,\omega) = 1 - \frac{\omega_p^2}{2}\left[\frac{1}{\gamma_e^3(\omega - kv_e)^2} + \frac{1}{\gamma_p^3(\omega + kv_p)^2}\right], \tag{9}$$

   where $\omega_p = \sqrt{4\pi e^2 n/m_{e,p}}$ is the plasma frequency, $\gamma_{e,p} = 1/\sqrt{1 - v_{e,p}^2/c_2}$ is the Lorentz factor and c the speed of light in vacuum.

5. In `twostream_e+e-_classic.py`, define the particle velocities (drift and thermal) as $v/\sqrt{1-v^2}$. Run a simulation with the same initial conditions of `twostream_e+e-_relativistic.py` . Compare the phase space at the equilibrium in the two cases. Are there differences?

3 Relativistic two-stream instability II

It is well knows that beam-plasma instabilities can decelerate electron beams in very short distances through the collective fields that they produce. Malkin and Fisch [Phys. Rev. Lett. 89, 125004 (2002)] proposed that such instabilities could be exploited to deposit energy in the core of an inertial fusion target, contributing to the realisation of fast-ignition [Tabak *et al.*, Phys. Plasmas 1, 1626 (1994).].

Consider an electron beam with density $n_b = 5 * 10^{22}$ particles/cm^3 moving at $\gamma = 50$ (where $\gamma = 1/\sqrt{1 - v_b^2/c^2}$ is the Lorentz factor). The beam is Maxwellian with temperature $T_b = 0.1\,\text{eV}$. The beam is supposed to propagate through the pellet core having electron density $n_e = 10^{26}$ particles/cm^3 and temperature $T_e = 5\,\text{KeV}$.

3.1 Tasks

1. Design a 1D simulation to verify if the two-stream instability is going to stop the beam as Malkin and Fisch proposed, considering that the dispersion relation for relativistic beams is

$$D(k,\omega) = 1 - \left(\frac{\omega_{pe}}{\omega}\right)^2 - \frac{1}{\gamma^3}\left(\frac{\omega_{pb}}{\omega - kv_b}\right)^2, \tag{10}$$

with a maximum growth rate of

$$\Gamma_{\max} = \frac{\sqrt{3}}{2\gamma}\left(\frac{\alpha}{2}\right)^{1/3}\omega_{pe} \tag{11}$$

occurring at the resonance $kv_b \simeq \omega_{pe}$.

2. Normalise the quantities in order to use them as input parameters in `twostream_relativistic.py`. Use the following definitions: $\hat{t} = t\omega_{pe}$, $\hat{v} = v/c$ and $\hat{x} = x/(c/\omega_{pe})$ and remember that $v_{th} = \sqrt{k_B T_e/m_e}$, where k_B is the Boltzmann constant. Given these definitions, what is the normalisation for the electric field and the energy?

3. Run the simulation and analyse the results

4 Landau damping

Landau damping was predicted analytically for the first time in 1964 by the soviet scientist Landau. We will recall here the explanation of the phenomenon given in the book D. R. Nicholson, *Introduction to plasma theory*, John Wiley & Sons 1983, p. 83: *"Consider a wave with phase speed $v_\varphi = \omega/k$ in a Maxwellian plasma. Those particles with speeds u very close to v_φ interact strongly with the wave. Particles withs speeds slightly faster than v_φ are grabbed by the wave and slowed down, giving up energy to the wave, while particles with speeds slightly slower than the wave are sped up, taking energy from the wave. Since in a Maxwellian plasma there are more particles with speeds slightly less than v_φ than with speeds slightly greater than v_φ, the net result is an energy gain by the particles and an energy loss by the wave; this is Landau damping."*. It appears important to underline that Landau damping is a collisionless process, which does not involve collisions and/or dissipation and it is therefore a reversible process.

The script `Landau_damping.py` will allow you to visualise the effect of the Landau damping on Langmuir waves in a Maxwellian plasma.

4.1 Tasks

1. Perform a convergence study and define the simulation parameters that will allow you to reduce the simulation computational cost, but to trust the results.

2. The damping frequency for Langmuir waves in a Maxwellian plasma is given by

$$\omega_D = -\omega_{pe}\sqrt{\frac{\pi}{8}}\left(\frac{1}{k\lambda_D}\right)^3 \exp\left(-\frac{3}{2}\right)\exp\left(-\frac{1}{2k^2\lambda_D^2}\right) \tag{12}$$

where $\lambda_D = \sqrt{k_B T_e/\left(4\pi e^2 n_e\right)}$ is the Debye length. Do the results agree with the theory? (Hint: use the information contained in the file "norm_phi.txt").

3. Increase the number of time steps in the simulation (choose `NT` at least four times bigger) and check the field energy. What do you observe? This phenomenon is called plasma echoes and demonstrates that Landau damping is a reversible process (a good introduction to plasma echoes is given in the book F. F. Chen *Introduction to Plasma Physics and Controlled Fusion*, Premium Press, New York 1984, p. 324).

4. Initialise the electron velocities according to a flat-top distribution function between `[-VT, VT]` and repeat the simulation. Can you observe a strong damping?