Sara Gómez, Universidad EAFIT.
Jaime Humberto Hoyos, Universidad de Medellín.

PLASMAPP

USER MANUAL

Particle in cell simulations for electrostatic one-dimensional systems of plasma.

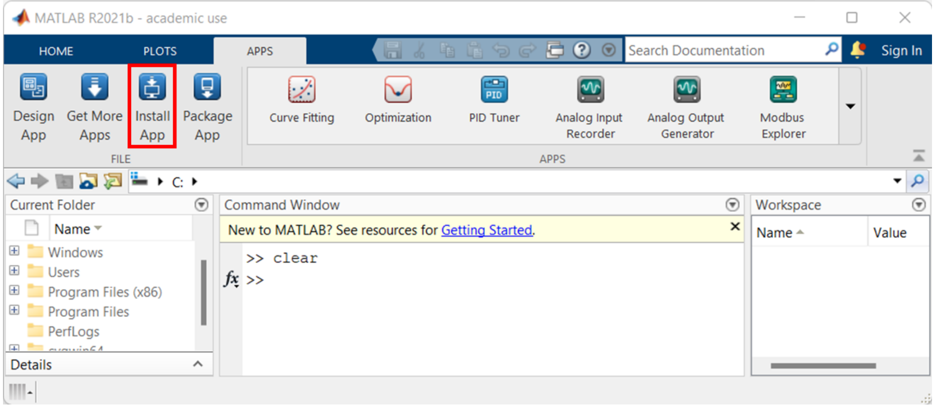
sgomezr3@eafit.edu.co
jhhoyos@udemedellin.edu.co

WHAT IS PLASMAPP?

PlasmAPP is an App designed in MATLAB App Designer that aims to facilitate simulations of plasmas in the one-dimensional electrostatic limit using the particle-in-cell method. This App will allow you to vary the system parameters, the beam parameters, the numerical methods to solve the Poisson's equation and the equations of motion, and you will be able to choose the diagnostics you want to analyze.

HOW TO INSTALL PLASMAPP?

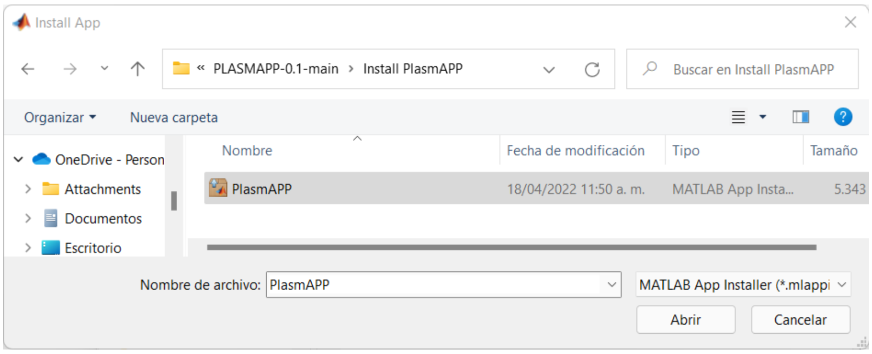
- *Go to the link:* https://github.com/gsara798/PLASMAPP-0.1.git
- *Download the folder "Install PlasmAPP".*
- *Open MATLAB and go to the "APPS" tab and then click on "Install APP".*



New to MATLAB? See resources for Getting Started.

```
>> clear
fx >>
```

- *Go to the folder "Install PlasmAPP" and open the file "PlasmAPP".*

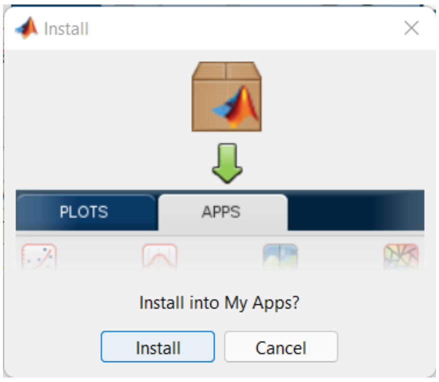


Nombre	Fecha de modificación	Tipo	Tamaño
PlasmAPP	18/04/2022 11:50 a. m.	MATLAB App Insta...	5.343

Nombre de archivo: PlasmAPP

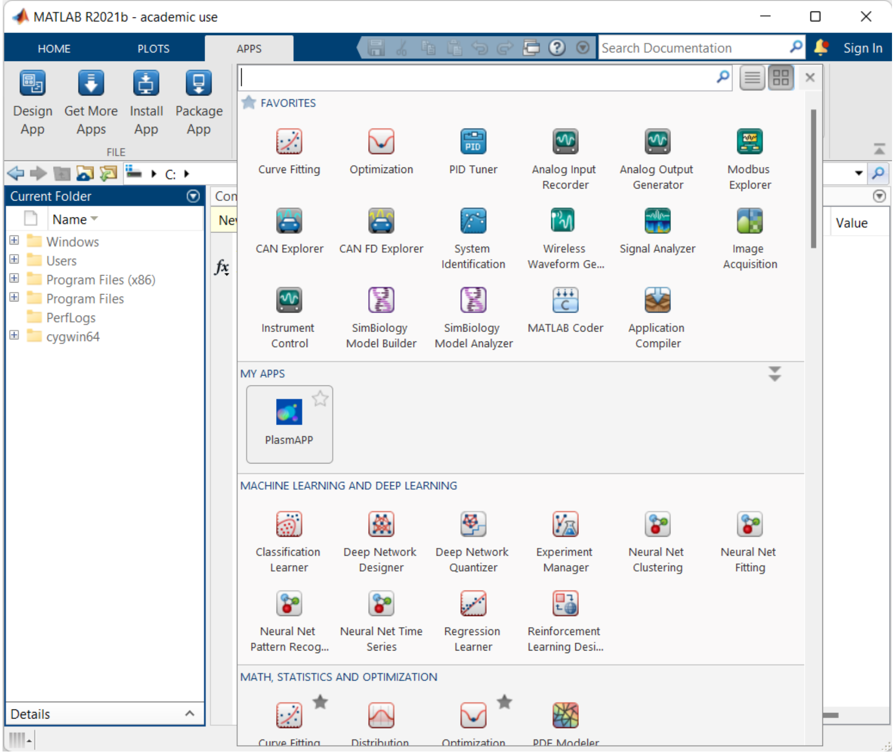
MATLAB App Installer (*.mlappi

- *Click on "Install".*



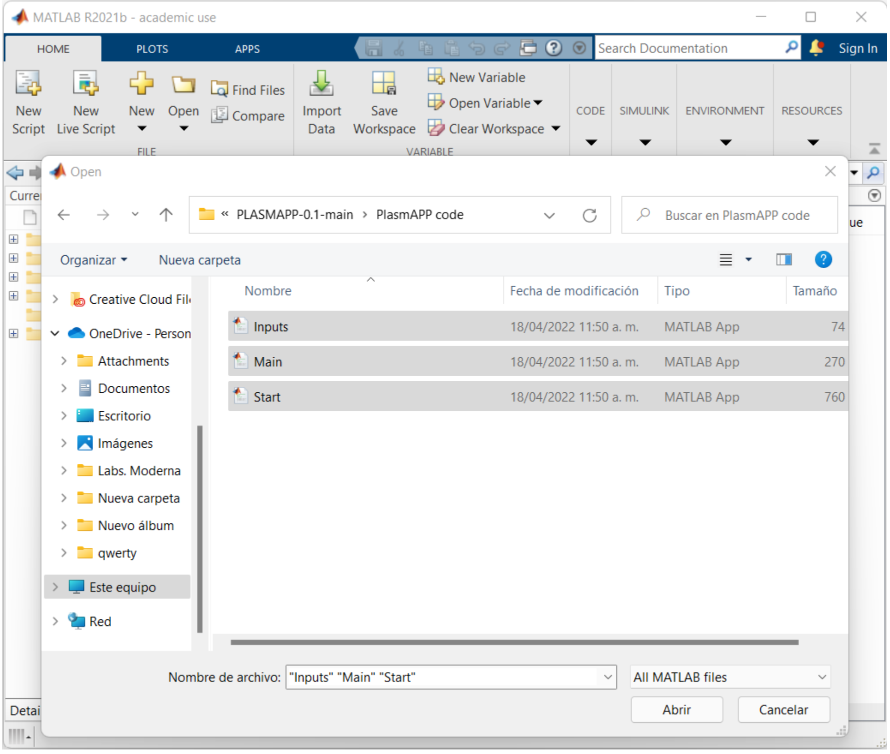
Install into My Apps?

- *Then you can run the APP.*



HOW TO SEE THE CODE OF PLASMAPP?

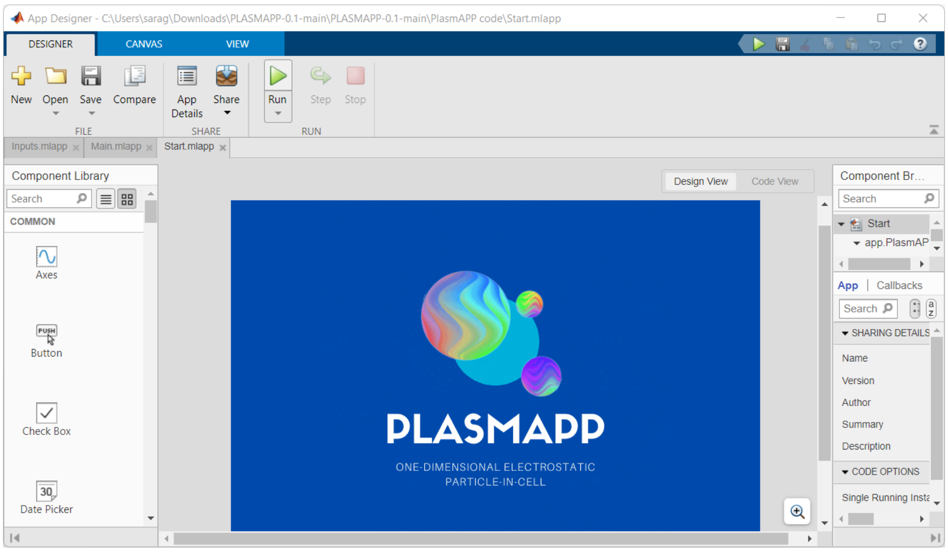
- *Go to the link:* https://github.com/gsara798/PLASMAPP-0.1.git
- *Download the folder "PlasmAPP code".*
- *In MATLAB click on the tab "Open" and select the folder "PlasmAPP code" and open the files: "Start.mlapp", "Inputs.mlapp", and "Main.mlapp".*



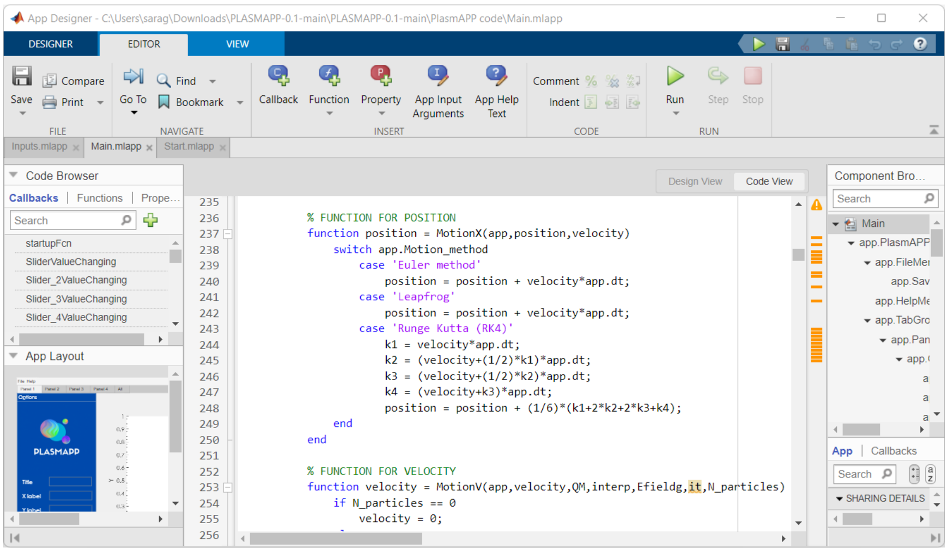
Nombre	Fecha de modificación	Tipo	Tamaño
Inputs	18/04/2022 11:50 a. m.	MATLAB App	74
Main	18/04/2022 11:50 a. m.	MATLAB App	270
Start	18/04/2022 11:50 a. m.	MATLAB App	760

Nombre de archivo: "Inputs" "Main" "Start"

- *Select the tab "Start.mlapp" and then you can run the APP.*



- *PlasmAPP is divided into 3 programs, each with a different graphical interface. If you wish to see the code, click on "Code View".*



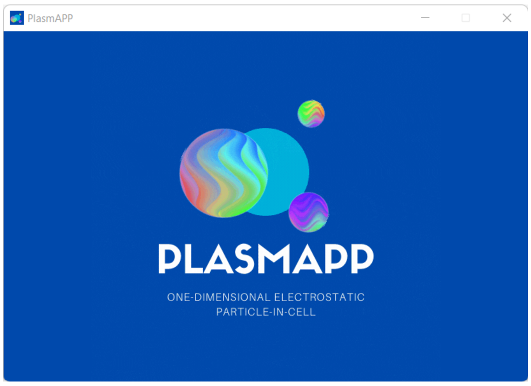
```
        % FUNCTION FOR POSITION
        function position = MotionX(app,position,velocity)
            switch app.Motion_method
                case 'Euler method'
                    position = position + velocity*app.dt;
                case 'Leapfrog'
                    position = position + velocity*app.dt;
                case 'Runge Kutta (RK4)'
                    k1 = velocity*app.dt;
                    k2 = (velocity+(1/2)*k1)*app.dt;
                    k3 = (velocity+(1/2)*k2)*app.dt;
                    k4 = (velocity+k3)*app.dt;
                    position = position + (1/6)*(k1+2*k2+2*k3+k4);
            end
        end

        % FUNCTION FOR VELOCITY
        function velocity = MotionV(app,velocity,QM,interp,Efieldg,it,N_particles)
            if N_particles == 0
                velocity = 0;
```

PLASMAPP STRUCTURE

The implemented algorithm consists of three codes which are interconnected. Each code is described below with its graphical interface.

- ### START.MLAPP

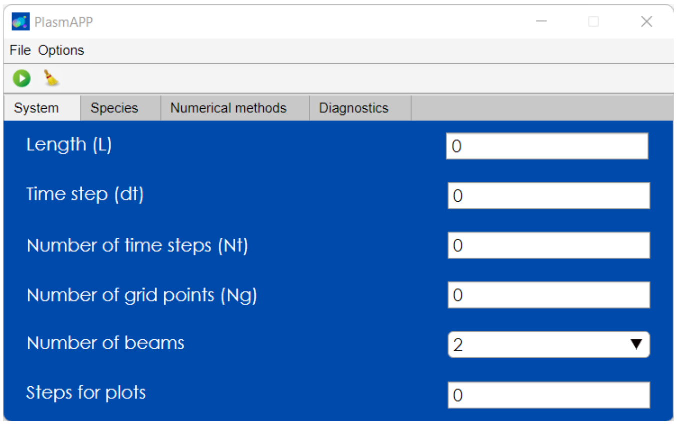
The first program initializes the user by displaying the PlasmAPP logo. Once the logo is clicked, the Inputs.mlapp program opens. The graphical interface of this code is shown below.



- ## INPUTS.MLAPP

The Inputs.mlapp code interface, the user enters the initial parameters needed to run the program. It has four tabs which are explained as follows:

- **System:** *In this window, the user chooses the system parameters.*



Length (L) 0

Time step (dt) 0

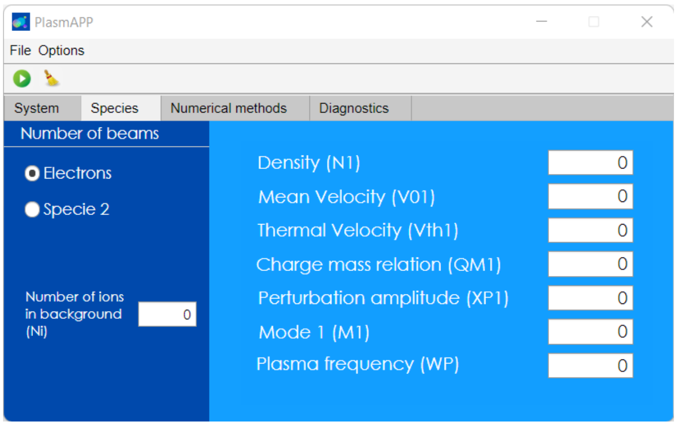
Number of time steps (Nt) 0

Number of grid points (Ng) 0

Number of beams 2 ▼

Steps for plots 0

- **Species:** *Once the user chooses the number of beams to simulate in the System window, the characteristics of each species can be determined in the Species tab.*



Number of beams

● Electrons

○ Specie 2

Number of ions in background (Ni) 0

Density (N1) 0

Mean Velocity (V01) 0

Thermal Velocity (Vth1) 0

Charge mass relation (QM1) 0

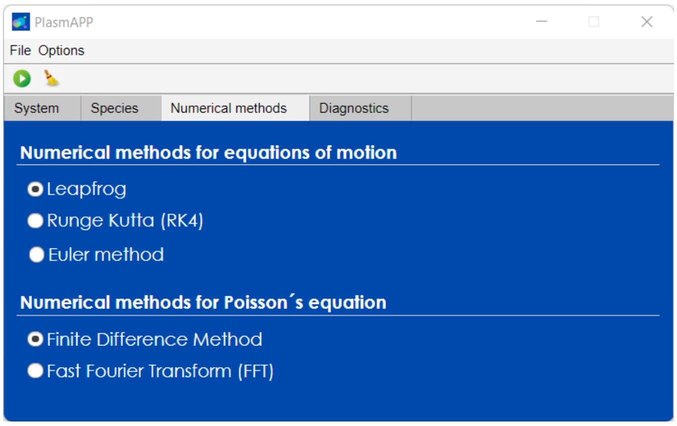
Perturbation amplitude (XP1) 0

Mode 1 (M1) 0

Plasma frequency (WP) 0

If you want to use the background ions, it is important to put the same number of electrons and ions.

- ***Numerical methods:*** *In this tab the numerical methods for solving the equations of motion and Poisson's equation are chosen.*



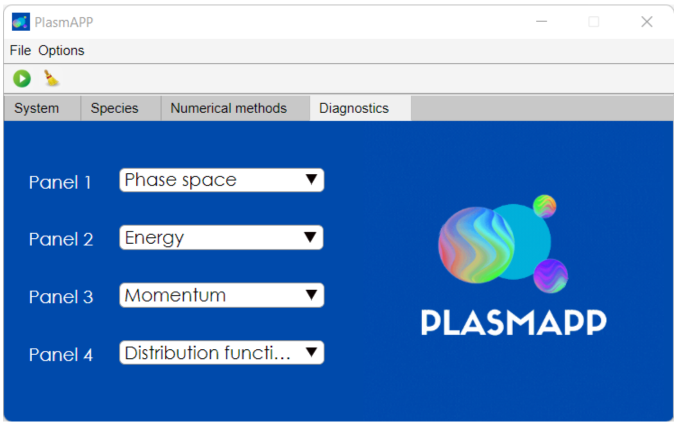
Numerical methods for equations of motion

● Leapfrog
○ Runge Kutta (RK4)
○ Euler method

Numerical methods for Poisson´s equation

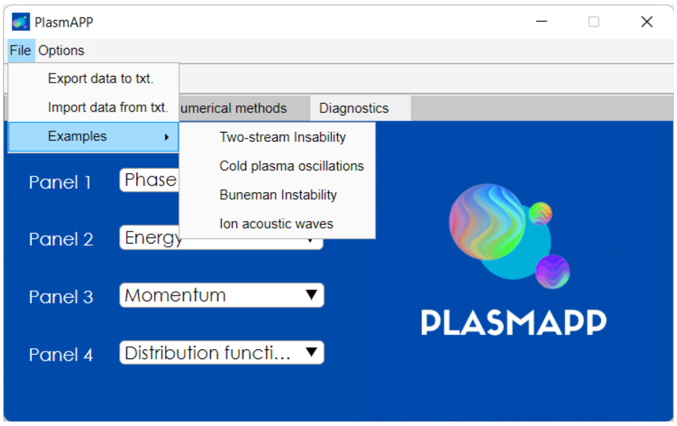
● Finite Difference Method
○ Fast Fourier Transform (FFT)

- ***Diagnostics:*** *In the last tab, the user selects the diagnostics to be observed.*



Panel 1: Phase space
Panel 2: Energy
Panel 3: Momentum
Panel 4: Distribution functi...

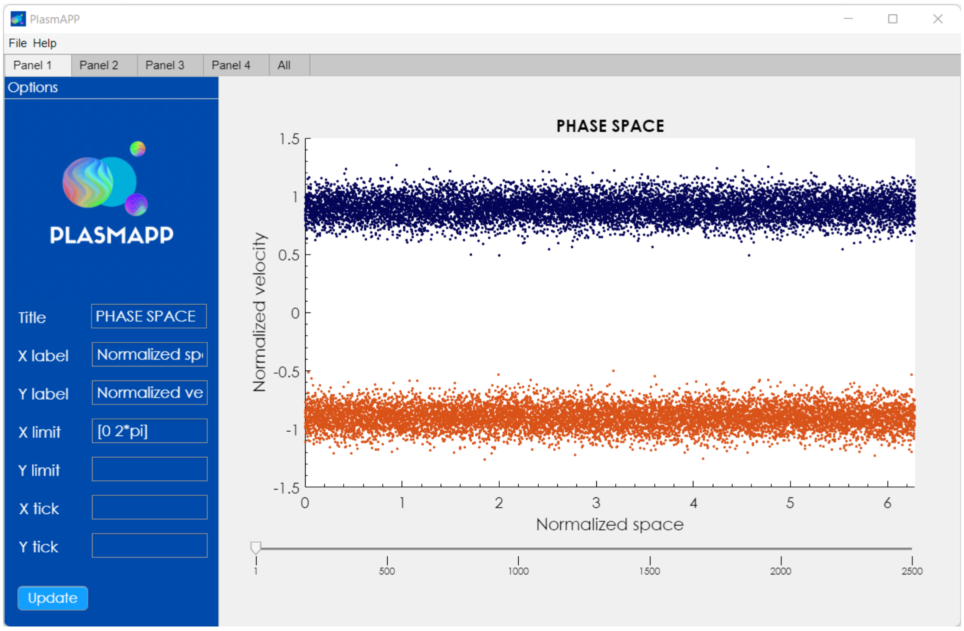
Additionally, the user has the option to export data in text format, import data in the same format, and can load the selected examples by clicking on the file button in the menu. Once the user presses the button to run the program, all variables are saved in a data vector. Subsequently, the Main.mlapp program is opened and this vector is sent to it.



Panel 1: Phase
Panel 2: Energy
Panel 3: Momentum
Panel 4: Distribution functi...

- ## MAIN.MLAPP

The last code is used to process the variables and with these to implement the PIC algorithm, additionally, in the graphical interface the diagnostics selected by the user must be observed. The interface contains a tab for each diagnostic chosen by the user and a last tab that collects all the diagnostics. The user is able to edit the title of the graph and the labels, limits, and ticks of the axis. In addition, the user can observe the simulations in the iterations available in the slider. Finally, in the last tab, you can see the maximum error percentage for energy, the maximum error percentage for momentum, and the execution time.



Title PHASE SPACE

X label Normalized sp

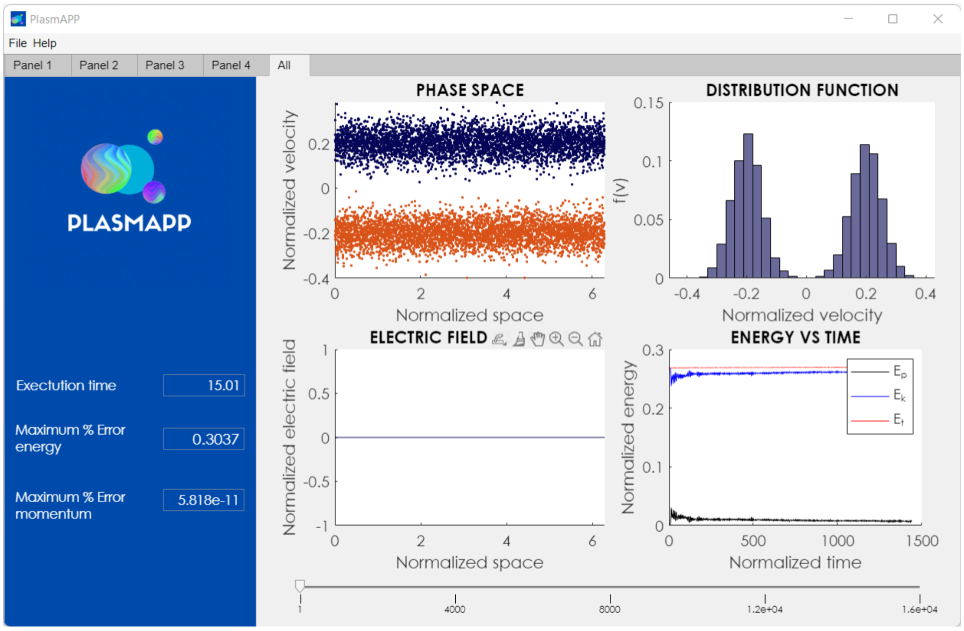
Y label Normalized ve

X limit [0 2*pi]

Y limit

X tick

Y tick

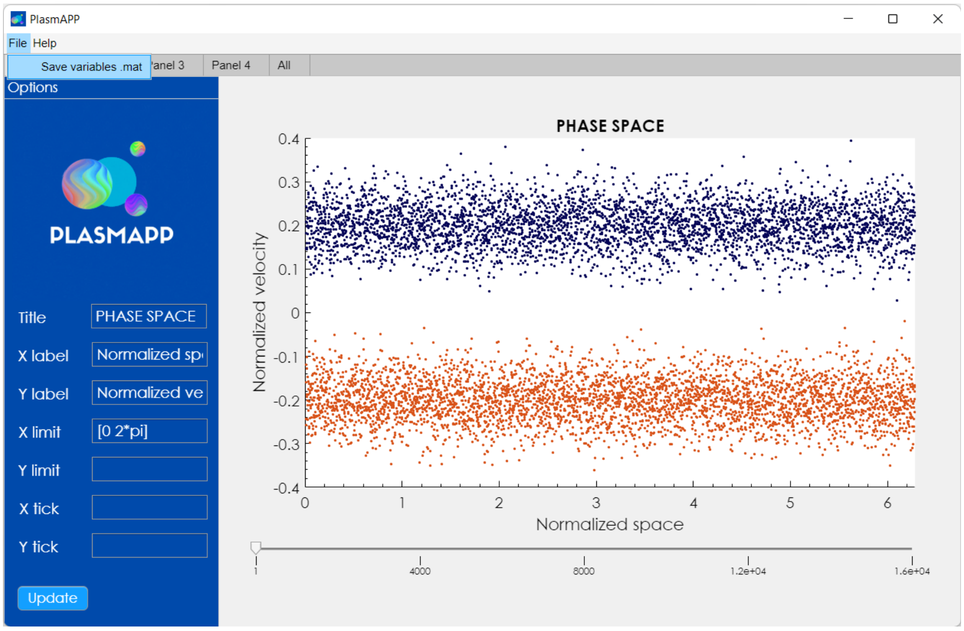


Exectution time 15.01

Maximum % Error energy 0.3037

Maximum % Error momentum 5.818e-11

In order to post-process the simulation, the results of the variables can be saved by clicking on "file" and then on "Save variables .mat".



Title PHASE SPACE

X label Normalized sp

Y label Normalized ve

X limit [0 2*pi]

Y limit

X tick

Y tick