2D Particle-In-Cell (PIC) Computer Model for Antimatter Plasma Simulation

Clementine Domine 9973827

Master project performed in collaboration with *Daniel Duque*

School of Physics and Astronomy

University of Manchester

January 2020

Abstract

A particle-in-cell (PIC) computer code aiming at simulating the equilibrium conditions for an antiproton and electron plasma confined in a Penning-Malmberg trap is described in this report. This code is planned to be used for the study of the electron/antiproton separation technique used in the ALPHA experiment at CERN, called the e-kick procedure. In first place, the antiprotons are cooled through collisions with electrons inside the trap which radiate the energy through cyclotron radiation. Removing the electrons is a crucial process that allows the cooled antiprotons to later recombine with positrons to produce antihydrogens. The code presented here was adapted from an existing 2D PIC code written in MATLAB by Federico Peinetti [1]. Different means of establishing appropriate initial conditions are investigated in this report. Using the MATLAB code as a reference, a C++ PIC code was developed in parallel by Daniel Duque to improve the method and efficiency of the calculations.

1 Introduction

The theory of the Big Bang predicts that the universe was created in equal amounts of matter and antimatter. Antimatter was first theorized by Dirac in 1928 [2] and experimentally validated in 1932 by Anderson [3] with the detection of the positron in an experiment detecting cosmic particles. Antiparticles that compose antimatter are defined as the particles that have opposite properties to normal particles [2]. In other words, a particle and its antiparticle have the same mass, but opposite electric charge and quantum numbers. For instance, a proton has a positive charge while an antiproton has a negative charge. However, until now, there has been no observation showing significant antimatter in our universe. The matter and antimatter asymmetry remains one of the grand challenges of physics today. This contradiction between the theory and observation motivates studies on the matter-antimatter asymmetry and the fundamental symmetry of physical laws such as CPT symmetry, the combination of the charge conjugation (C), parity transformation (P), and time reversal (T) symmetry. The ALPHA experiment at CERN produces and traps antihydrogens to experimentally compare their properties (e.g. spectrum, charge, gravitational mass) to hydrogen. The antihydrogen atoms are produced from the recombination of cold plasmas of positrons and antiprotons [4]. Plasmas are complex dynamical and multidimensional systems. This motivates the interest of building computer simulations to gain insights on their dynamical behaviours inside a trap. The current project aims at understanding a particle-in-cell (PIC) computer code written in MATLAB for modeling antimatter plasma inside the ALPHA Penning-Malmberg trap and adapting it to later simulate the crucial separation of the electrons used to cool the antiproton plasma in the antimatter ALPHA experiment.

The following research plan was agreed with Dr. William Alan Bertsche, the supervisor of this project at the University of Manchester. A full understanding and characterization of an existing 2D PIC code written in MATLAB developed by Federico Peinetti for the Penning-Malmberg trap based at the University of California Berkeley should be obtained [1]. The equilibrium conditions for trapped antiproton and electron plasmas should be understood. The different methods should be extensively described in order to justify their physical validity for a plasma phenomenon study. Then, the existing PIC code should be adapted, to simulate and study in detail the electron and antiproton separation process also called the e-kick. In parallel, a C++ PIC code should be developed with the MATLAB code as a reference in order to increase the efficiency and the method. Finally, possible extensions of the model to simulate novel autoresonant separation techniques will be explored. The current report is divided into three sections. The first section describes the physical motivation for building a PIC simulation and in particular the plasma and antimatter theory as well as the electron removal procedure. A general introduction to PIC codes followed by an extensive description of the adapted MATLAB PIC code are detailed in the second section. In the final section, test results of the newly developed program are presented and compared with the results obtained with the newly developed C++ code [5].

2 Theory

2.1 Plasma Physics

Plasmas are often referred to as the fourth state of matter after solid, liquid and gas [6]. A plasma is defined as a quasi-neutral gas which exhibits a collective behaviour [7]. The antiproton and electron plasmas used in the ALPHA experiment at CERN are part of a subcategory of plasmas known as non-neutral plasmas. A non-neutral plasma is made of charged particles that do not overall cancel to neutralise. The net charge produces a self-consistent electric field influencing the dynamical description of the plasma [7]. To characterise a plasma dynamical evolution, the plasma frequency is used as a fundamental time scale. The plasma frequency is given by:

$$\omega_P^2 = \frac{ne^2}{\epsilon_0 m} \tag{1}$$

where m is the effective mass of the particle, ϵ_0 the permittivity constant, e the particle charge and n the density of the plasma [6].

For the plasma to show collective behaviour, it requires that

$$\lambda_D << L \tag{2}$$

where L is a characteristic dimension of the plasma and λ_D is the Debye length giving the range inside of which the particle interactions are dominant and outside of which collective effects prevail. This length scale characterizes the plasma ability to shield out electric potentials applied to it. The Debye length is given by the expression:

$$\lambda_D = \left(\frac{\epsilon_0 T k_b}{ne^2}\right)^{\frac{1}{2}} \tag{3}$$

where T is the temperature of the plasma and k_b is the Boltzmann constant.

The collective behaviour of a plasma is also characterised by N_D, the number of particles inside a Debye sphere (a sphere with a radius of one Debye length) and ensured by [8]:

$$N_D >> 1. \tag{4}$$

Collective behaviour is also ensured by the absence of interparticle collisions as they occur on space and time scales negligible compared to the interactions resulting from the electromagnetic interaction. The Vlasov–Poisson kinetic description of a plasma only considers the electromagnetic interactions between the particles inside the plasma. The interactions between a charged particle and an electromagnetic field are explained by Equation 5, the Lorentz force of electromagnetism $\vec{F}$:

$$\vec{F} = q(\vec{E}_{\text{tot}} + \vec{v} \times \vec{B}) \tag{5}$$

with the total electric field $\vec{E}_{\text{tot}}$ and magnetic field $\vec{B}$ experienced by the particle [9].

We consider a confined plasma inside a cylindrical Penning-Malmberg trap. The geometry of the electromagnetic Penning-Malmberg trap is simplified to a trap made of four identical cylindrical electrodes

of length 13.22 mm and diameter of 29.76 mm. The four electrodes are separated by a gap of 0.5 mm. Each electrode can be set to a chosen voltage to build a trapping voltage potential.
The potentials in the ALPHA trap are not perfectly quadratic rendering their analysis complex but have the advantage to be more technically feasible and extremely flexible in manipulating the particles. The simplified geometry of the Penning-Malmberg trap in the ALPHA experiment is shown in Figure 1 [10].

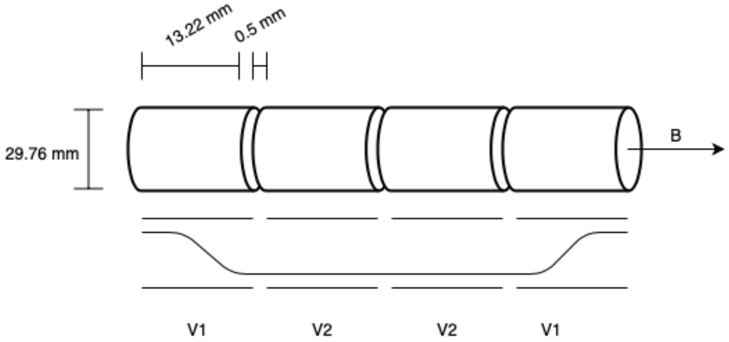


Figure 1: The schematic Penning-Malmberg trap in the ALPHA experiment.

Note that in this case the self-consistent electric field emerging from the net charge of the plasma $\vec{E}_{\mathrm{self}}$, as well as the potential well electric field $\vec{E}_{\mathrm{trap}}$, contribute to $\vec{E}_{\mathrm{tot}}$, the total electric field for the trapped plasma. The two electric fields can be added linearly as they are independent. The electric potential of the well can be obtained by solving the Laplace equation given below

$$\nabla^2\phi = 0 \tag{6}$$

where ϕ is the self potential.
The self-consistent electric field can be obtained by solving the Poisson equation for the self-potential φ.

$$\nabla^2\varphi = -\frac{\rho}{\epsilon_0} \tag{7}$$

where φ is the self-potential and ρ the charge density.
The plasma is sitting inside the trap with a large magnetic field. Therefore, the radial transport is assumed to be null due to the shear forces in the plasma and the high intensity of the magnetic field making the electron's Larmor radius negligible in comparison to any relevant length and time scale. Hence, a gyrokinetic description is chosen as an approximation for the motion of the electron in the (r, θ) transverse plane [1]. The $\vec{E}x\vec{B}$ drift induces a slow azimuthal rotation of the plasma column. The $\vec{E}x\vec{B}$ drift can be ignored as the rotation happens on a longer time scale than the one of the research interest. The simulation is thought to be located in the rotational frame of the plasma [7, 11]. In the limit of low velocity of the particle giving negligible currents, the contribution from the magnetic field of the plasma can be neglected without loss of accuracy. The plasma loaded inside the trap is assumed to have an azimuthally symmetric initial distribution, simpliflying the geometry of the system [12]. The above simplifying assumptions and approximations made in accordance with the experimental set up reduce the dimentionality and complexity of the system. Under this hypothesis, each particle has a single spatial

degree of freedom in the axial direction. This is done to simplify the study of the dynamical behaviour of the plasma.

2.2 The Electron Removal Procedure

The antiprotons are created from the collision of protons on a target inside the Proton Synchrotron at CERN. They are sent to the antiproton decelerator which produces bunches of antiprotons with a kinetic energy around 5.3 MeV to be sent to the experiment. Inside the experiment, the antiproton plasma of around 10 million particles with density 10^{12} m^{-3} is confined with a 3 Tesla magnetic field and an electric field inside the trap [12, 13]. The plasma frequency ω 1 and the Debye length are hence calculated to be of the order of 10^6Hz and 10^{-4} m, respectively. When entering the trap, the antiprotons pass through an aluminum foil ('degrader') where they collide with electrons and lose energy. Hence, when they arrive in the trap, the energy distribution of the antiprotons is outside the range of energy for the recombination of positrons and antiprotons to form an antihydrogens. Inside the Penning-Malmberg trap at 4K, the energy of the antiprotons is decreased through Coulomb collisions with a cold electron plasma of approximately 10^7 electrons at a temperature of 10 K [14, 15].

The electron cooling exploits the difference in mass between electrons and antiprotons. The electrons radiate energy on a shorter time scale than the antiprotons as cyclotron radiation energy is inversely proportional to the squared mass as stated by the Lamor formula [9]. This cooling is a crucial step as the energy of the antihydrogen is proportional to the energy of the antiproton and the trapping is limited to antihydrogens with temperature lower than 0.5 K. When the antiproton has reached a thermal equilibrium with the surroundings through cyclotron radiation, the electron has to be sent out of the trap. The removal of electrons is essential to prevent competition with the antihydrogen creation from the positronium (Ps) formation [14].

The electron removal procedure called the e-kick exploits the mass difference between the electron and antiproton when the two are in thermal equilibrium [16] [17]. One side of the well is removed thanks to a quick voltage pulse, which allows the electrons to escape. Antiprotons take a longer time to escape the well as they are more massive and slower. To limit the loss of antiprotons, an additional voltage pulse is applied to re-establish the original well. Since this method leads to plasma heating, the procedure is repeated multiple times, leaving some electrons in the well so that sympathetic cooling can take place in a 'waiting potential' after each pulse. The time between each e-kick is determined experimentally as a compromise between antiproton temperature and density distribution. A schematic diagram of the different steps of the procedure is shown in Figure 2 [18]. In the ALPHA experiment, the temperature obtained at the end of the e-kick procedure is $\sim$ 500 K. After the e-kick procedure, the antiprotons undergo evaporative cooling and adiabatic expansion cooling. The efficiency of antihydrogen production is improved by lower post e-kick temperature [12, 19].

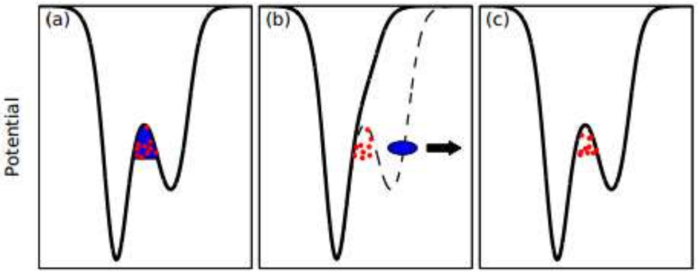


Figure 2: E-kick pulse procedure. The electron-antiproton plasma is initially confined in the e-kick well (a). One side of the well is raised (b), allowing electrons (blue) to escape. The well is restored (c) before antiprotons (red dots) can escape [12].

The influence of the final temperature T by well shapes, duration, timing and voltage of the applied pulses, will be studied using a PIC code simulation in a future study.

3 PIC Codes for Computational Plasma Simulation

Building computer simulations of plasmas provides insights on their complex dynamical behaviours. However, the development of such models is often limited by the available computational power and storage capacities. The method should be chosen to be as fast as possible while retaining a good accuracy of the results. Appropriate simplifying assumptions and approximations reduce the extensive computational resources required. The MATLAB PIC Code developed by Federico Peinetti aims at studying specifically the high amplitude of Bernstein–Greene–Kruskal (BGK) modes of electrostatic waves in an unmagnetized and collisionless plasma [1]. In this report, we describe the adapted version of this code for the ALPHA Penning-Malmberg trap using a loading of one particle.

3.1 Algorithm of the Developed PIC Code

A PIC code solves the multiple-body coupled partial differential equation system of a plasma in order to get an insight into its dynamical behaviour. In this approach, the plasma is discretized as a set of coupled macroparticles using a mean electric field approximation. Hence, the discrete partial differential equations describing the motion of the macroparticles are solved simultaneously on the grid with cells of finite size where the macroparticles are weighted [8]. A flow chart of the characteristic PIC algorithm for a macroparticle referred by the subscript i is shown in Figure 3. The field quantities will be obtained on the spatial grid labeled with the index j [8]. The time loop allows the simulation of the time dependent behaviours of the plasma. The number of iteration steps should be specified in the input section and chosen with respect to the study goal. For instance, in the case of an equilibrium plasma, the time step can be calculated using the frequency of the plasma as a reference. The methods as well as the implementations of the seven steps of the flow diagram in Figure 3 are described in the next seven subsections.

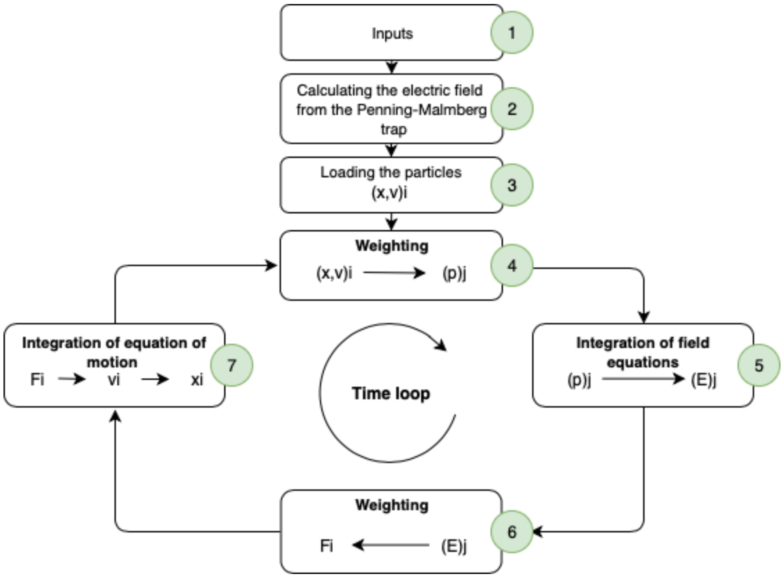

Inputs
1
Calculating the electric field from the Penning-Malmberg trap
2
Loading the particles (x,v)i
3
Weighting
(x,v)i
(p)j
4
Integration of field equations
(p)j
(E)j
5
Weighting
Fi
(E)j
6
Integration of equation of motion
Fi
vi
xi
7
Time loop

Figure 3: Flow diagram of the PIC code simulation for dynamical evolution of macroparticles.

3.1.1 Inputs

The variables that describe the geometry of the trap are manually entered. The trap is built using the geometrical parameters previously specified. New variables such as the length of the trap, the axial and radial grid spacing h_z and h_r are defined. Numerical values such as the number of axial and radial grid points should be specified. The characteristics of each plasma such as the charge, mass, initial, density are also entered manually.

3.1.2 The Laplace Equation Solver

The Laplace equation is solved once per simulation on a grid to evaluate the external electric field experienced by the particle in the simulation. Note that the Laplace equation solver can be run separately in order to insert a driving potential.

Finding the electrostatic potential ϕ inside a Penning-Malmberg trap translates into solving the Laplace equation in cylindrical coordinates. In the case of azimuthal symmetry, this translates into solving Equation 8 [8].

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\phi}{\partial r}\right)+\frac{\partial^2\phi}{\partial z^2}=0 \tag{8}$$

where ϕ is the potential in the vacuum from the electrodes of the trap.

In order to be solved numerically, the problem is discretized onto a grid of size N_r by N_z, corresponding to the number of grid points in the radial and axial directions with spacing of h_r and h_z, respectively. Equation 8 has to be solved over the domain covered by the computational grid such as the one in Figure 4. The size of the grid is defined by $j = (1,...N_r)$, the number of grid points in the radial direction and $i = (1,...N_z)$ the number of grid points in the axial direction.

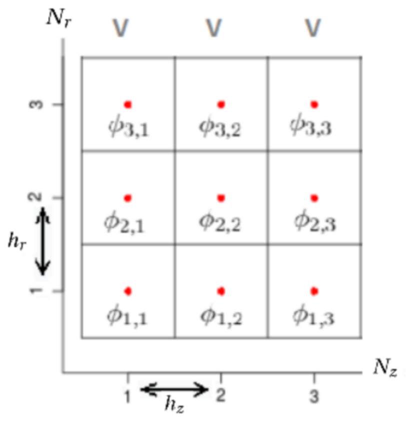


Figure 4: Spatial grid used to solve the Laplace equation. The $N_r = 3$ by $N_z = 3$ grid points are represented by the red dots located at the center of the grid cells of size h_r by h_z.

Using Taylor series, considering small values of h_r and h_z, the Laplace operator can be discretized with respect to the grid, where $\phi_{i,j} = \phi(z = z_i, r = r_j)$. Hence, from the finite difference method [20], Equation 8 can be rewritten as follows where the discretization of the radial dimension is independent of the axial dimension discretization

$$\frac{1}{r_{i,j}}\left[\frac{r_{i,j+\frac{1}{2}}\phi'_{i,j+\frac{1}{2}} - r_{i,j-\frac{1}{2}}\phi'_{i,j-\frac{1}{2}}}{h_r}\right] + \left[\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{h_z^2}\right] = 0. \tag{9}$$

Equation 9 can be rearranged as

$$\frac{1}{r_{i,j}}\left[\frac{r_{i,j+\frac{1}{2}}\phi_{i,j+1} - 2r_{i,j}\phi_{i,j} + r_{i,j-\frac{1}{2}}\phi_{i,j-1}}{h_r^2}\right] + \left[\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{h_z^2}\right] = 0, \tag{10}$$

where

$$\frac{r_{i,j+\frac{1}{2}}}{r_{i,j}} = \frac{j}{j - \frac{1}{2}} \qquad (11) \qquad\qquad \frac{r_{i,j-\frac{1}{2}}}{r_{i,j}} = \frac{j}{j + \frac{1}{2}}. \qquad (12)$$

The two-dimensional Laplace equation for all grid points is reduced to a set of one-dimensional linear equations that can be translated and solved as a matrix product.

$$\begin{bmatrix} M & U & 0 & N_u & 0 & 0 & 0 & 0 & 0 \\ D & M & U & 0 & N_u & 0 & 0 & 0 & 0 \\ 0 & D & M & U & 0 & N_u & 0 & 0 & 0 \\ N_d & 0 & D & M & U & 0 & N_u & 0 & 0 \\ 0 & N_d & 0 & D & M & U & 0 & N_u & 0 \\ 0 & 0 & N_d & 0 & D & M & U & 0 & N_u \\ 0 & 0 & 0 & N_d & 0 & D & M & U & 0 \\ 0 & 0 & 0 & 0 & N_d & 0 & D & M & U \\ 0 & 0 & 0 & 0 & 0 & N_d & 0 & D & M \end{bmatrix} \begin{pmatrix} \phi_{1,1} \\ \phi_{1,2} \\ \phi_{1,3} \\ \phi_{2,1} \\ \phi_{2,2} \\ \phi_{2,3} \\ \phi_{3,1} \\ \phi_{3,2} \\ \phi_{3,3} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{13}$$

Each point on the grid interacts with its four immediate neighbours, translating into four non-zero diagonals inside the linear system. Shown above as an example is the equation for the 3 by 3 grid presented in Figure 4 where each element M, U, D, N_u, N_d is obtained from the factors of the terms $\phi_{i,j}$, $\phi_{i,j+1}$, $\phi_{i,j-1}$, $\phi_{i+1,j}$ and $\phi_{i-1,j}$, respectively, in Equation 10.

At the boundary, when $i = N_z$ and $j = N_r$, the generic equation cannot be resolved since $\phi_{i+1,j}$ and $\phi_{i,j+1}$ are not defined, as they are out of the domain. Similarly, when $i = 1$ and $j = 1$, the generic equation cannot be resolved since $\phi_{i-1,j}$ and $\phi_{i,j-1}$ are not defined, as they are out of the domain. Hence, at the boundary, a different method of solving the Laplace equation is used. The Neumann boundary condition at $r = 0$ arises naturally, assuming azimuthal symmetry and the continuity of the plasma in the center of the trap. On the other hand, at $z = 0$ and $z = L$, where L is the length of the trap, the Neumann conditions are assumed for simplicity. This leads to solutions appreciably different from the real field at points close to the boundary. The region of interest is located in the axial center of the trap. It has a comparable size to the length of the plasma and is much shorter than the length of the trap L. Hence, the effect of the axial boundary in this region is negligible. The Dirichlet boundary conditions arise trivially from the potential of electrodes. The potential in the gaps shown in Figure 1 is calculated by linear interpolation of the electrodes's potential. The particles being mostly located at the center of the trap, at the radial boundaries of the trap, the detailed solution of the potential including the gap interpolated potential has a negligible effect.

In Equation 9, at the end of the trap $\phi'_{i+\frac{1}{2},j}$ is defined as

$$\phi'_{N_z+\frac{1}{2},j} = \frac{\partial\phi}{\partial z}\bigg|_{z_w,j} = 0 \tag{14}$$

where z_w is the axial length at the end of the trap.

In Equation 9, $\phi'_{i,j+\frac{1}{2}}$ at the wall is defined as

$$\phi'_{i,N_r+\frac{1}{2}} = \frac{\partial\phi}{\partial r}\bigg|_{i,r_w} \tag{15}$$

where r_w is the radius at the wall. The potential at the N_r grid point can be written as

$$\phi_{i,N_r} = \phi_{i,N_r+\frac{1}{2}} - \frac{h_r}{2}\frac{\partial\phi}{\partial r}\bigg|_{i,r_w} \tag{16}$$

Equation 16 can be rearranged to give an expression for $\phi'_{i,N_r+\frac{1}{2}}$

$$\phi'_{i,N_r+\frac{1}{2}} = \frac{\partial\phi}{\partial r}\bigg|_{i,r_w} = \frac{2}{h_r}(\phi_{i,N_r+\frac{1}{2}} - \phi_{i,N_r}) \tag{17}$$

In this way, for $j = N_r$ and any $i = N_z$, the two-dimensional Laplace equation is reduced to a set of one-dimensional linear equations of the form

$$\frac{1}{r_{i,N_r}}\left[\frac{r_{i,N_r+\frac{1}{2}}\frac{2}{h_r}(\phi_{i,N_r+\frac{1}{2}} - \phi_{i,N_r}) - r_{i,N_r-\frac{1}{2}}\phi'_{i,N_r-\frac{1}{2}}}{h_r}\right] + \left[\frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{h_z^2}\right] = 0. \tag{18}$$

The set of one-dimensional linear equations translated into matrix 13 has to be modified accordingly to the boundary conditions. A similar treatment is used for the boundary at $i = 1$ and $j = 1$.

3.1.3 The Loading

A chosen number of N_l virtual particles are then loaded with a given initial density distribution. The type of loading has to be decided with respect to the goal of the study. Note that the particles in the simulation do not correspond to real-world particles. A one-to-one correspondence would require a huge computational effort due to the large number of particles in a plasma [8]. One computational particle, also called macroparticle, represents a collection of multiple particles. The macroparticle in the simplified geometry translates to what can be thought of as 'macroring' in the three-dimension geometry. The case of a trivial loading of a single particle initially at rest in the trap and a line charge was implemented. In this case, the position, charge, momentum and energy of each particle is determined and manually entered.

An ergodic quiet mode loading was studied and is planned to be used in the C++ version of the code to obtain the necessary equilibrium distribution of the plasma [8]. This method assigns an initial phase-space position to each macroparticle (z_p, v_p) and a charge weight to each computational particle (q_p) by respectively inverting the velocity density distribution and cumulative charge of the plasma. This method allows to reduce the propagation of nonphysical density fluctuations and instabilities inside the plasma. The equilibrium distribution aims at representing to a good degree the experimental conditions in the laboratory [1,16].

3.1.4 The Grid Projection

The charge density on the grid is obtained by projecting the position of each macroparticle on the grid. This process requires a weighting of the charge at the grid points. The weighting depends on each particle position. This is achieved through interpolation among the nearest grid points to the particle. The first-order weighting that uses two grid points per particle is implemented in the code. The same coefficients of the projection are then used later for the 'grid to particle' projection in order to avoid a self force [8].

3.1.5 The Poisson Equation Solver

The calculated charge is used to solve the Poisson equation and determine the self-electric field. The program solves the Poisson equation for the self-consistent potential at each time step. Finding the self-consistent electrostatic potential φ of a plasma inside a Penning-Malmberg trap translates into solving the Poisson equation 7 in cylindrical coordinates. In the case of azimuth symmetry, this translates into solving

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\varphi}{\partial r}\right)+\frac{\partial^2\varphi}{\partial z^2}=-\frac{e}{\epsilon_0}\int f(r,z,v_z,t)\partial v_z \tag{19}$$

where $f(r,z,v_z,t)$ is the electron phase-space distribution function of the plasma [8].

Using the assumption that there is no radial transport on this time scale thanks to the strong magnetic field, the radial dimension of the plasma can be considered to be the superposition of N_R one-dimensional systems, referred to as radial zones. Therefore, the number of radial zones, N_R, can be determined based on the plasma profile, computational cost, and the required accuracy. The numerical algorithm is presented below for a generic radial decomposition. We consider the two-dimensional distribution function f of the plasma such that

$$f(r,z,v_z,t)=\sum_{k=1}^{N_R} f_k(z,v_z,t)S_k(r) \tag{20}$$

where S_k, a shape function for the k_{th} radial layer, is expressed as

$$S_k(r)=\frac{1}{\pi(2r_k\Delta r_k+\Delta r_k^2)}b_0 \tag{21}$$

where r_k is the radius, Δr_k is the width associated with the k_{th} radial zone and where the b_0-spline is a normalised function such that

$$b_0=\begin{cases}1 & \text{if } r_k<r<r_{k+1}\\ 0 & \text{otherwise.}\end{cases}$$

The function S_k is built so that the adjacent zones do not overlap one another. The S_k shape function is normalised as

$$2\pi\int_0^{r_w} S_k(r)rdr=1 \quad k=1,2,..,N_R \tag{22}$$

Under this hypothesis the electric charge Q in the plasma can be calculated as:

$$Q=e\int\sum_{k=1}^{N_R} f_k(z,v_z,t)dzdv_z. \tag{23}$$

The radial average of the two-dimensional total electrostatic potential over the k_{th} radial zone $\bar{\Phi}_k(z,t)$ is defined as:

$$\bar{\Phi}_k(z,t)=\frac{1}{r_k\Delta r_k}\int_{\Delta r_k}\Phi(r,z,t)rdr=\bar{\varphi}_k(z,t)+\sum_l\bar{\phi}_k^l(z,t) \tag{24}$$

where $\bar{\varphi}_k(z,t)$ and $\bar{\phi}_k^l(z,t)$ are the radial averages over the k_{th} radial zone self-potential and potential due to the l_{th} electrodes, respectively.

The cylindrical Poisson Equation 19 for the self-potential can hence be rewritten as:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\varphi}{\partial r}\right)+\frac{\partial^2\varphi}{\partial z^2}=-\frac{e}{\epsilon_0}\sum_{k=1}^{N_R}\left[\int f_k(z,v_z,t)\partial v_z\right]S_k(r). \tag{25}$$

Looking at the geometry of the problem, the self-consistent potential can be expanded as the sum of N_b terms of the Fourier Bessel series [1]:

$$\varphi(r,z,t)=\sum_{h=1}^{Nb}\varphi_h(z,t)J_0\left(\frac{j_{0,h}r}{r_w}\right). \tag{26}$$

The number of harmonics from the Bessel expansion N_b can be determined on the basis of the computational cost and the required accuracy.

The newly defined self-potential can be used as a solution to the Poisson Equation 19. This translates into a system of linear one-dimensional, second-order differential equations of the form

rw

$$\frac{\partial^2 \varphi_h}{\partial z^2} - \varphi_h \left(\frac{j_{0,h} r}{R_w}\right)^2 = -\frac{e}{\epsilon_0} \sum_{k=1}^{N_R} \left[\int f_k(z, v_z, t) \partial v_z\right] C_{h,k} \tag{27}$$

in which $C_{h,k}$ is the set of Bessel coefficients

$$C_{h,k} = \frac{2}{\pi(2r_k \Delta r_k + \Delta r_k^2) r_w^2 J_1^2(j_{0,h})} \int_{\Delta r_k} J_0\left(\frac{j_{0,h} r}{r_w}\right) r dr. \tag{28}$$

Equation 27 has to be solved over the domain covered by a computational grid line with N_z grid points in the axial direction such as the one presented in Figure 5.

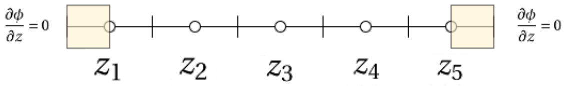

$\frac{\partial \phi}{\partial z} = 0$
$\frac{\partial \phi}{\partial z} = 0$
z_1
z_2
z_3
z_4
z_5

Figure 5: Spatial grid used to solve the Poisson equation with Neumann boundary conditions evaluated at the first and last grid point. The self-consistent potential $\varphi_{h,z}$ is evaluated at the grid points z_i located in the center of the line cell of size h_z delimited by the ticks.

Using the Taylor series and considering small values of h_z, the Poisson Equation 28 can be discretized with respect to the grid, with $\varphi_{h,z}$ where $h = (1, ...N_b)$, is the number of Bessel harmonics and $z = (1, ...N_z)$ is the number of grid points in the axial direction.

$$\left[\frac{\varphi_{h,z+1} - 2\varphi_{h,z} + \varphi_{h,z-1}}{h_z^2}\right] - \varphi_{h,z} \left(\frac{j_{0,h} r}{r_w}\right)^2 = -\frac{e}{\epsilon_0} \sum_{k=1}^{N_R} \left[\int f_k(z, v_z, t) \partial v_z\right] C_{h,k}(r) \tag{29}$$

After solving the above set of equations, the self-potential averaged over the kth radial zone can be evaluated to be

$$\bar{\varphi}_k = \frac{1}{r_k \Delta r_k} \sum_{h=1}^{N_b} \varphi_h(z, t) \int_{\Delta r_k} J_0\left(\frac{j_{0,h} r}{r_w}\right) r dr. \tag{30}$$

The three-dimensional problem can be decomposed as the superposition of N_b linear systems. At the boundary, when $z = 1$ and $z = N_z$ the generic equation cannot be resolved since $\varphi_{h,z-1}$ and $\varphi_{h,z+1}$ are not defined being out of the domain, respectively. Hence, the system is solved for all of the other grid points by a matrix of size $N_{h,z-1} * N_{h,z-1}$ resulting in the self-potential $\phi_{h,2...N_{z-1}}$. Each point in the grid interacts with their two immediate neighbours translating into two diagonals inside this matrix. The grid points at the boundaries are handled separately. The potential at the points at the boundaries in the axial dimension can be calculated using a Taylor expansion of the form:

$$\varphi_2 = \varphi_1 + \varphi_1' h_z + \frac{1}{2} \varphi_1'' h_z^2 \tag{31}$$

and

$$\varphi_3 = \varphi_1 + 2\varphi_1' h_z + 4\frac{1}{2} \varphi_1'' h_z^2. \tag{32}$$

Eliminating φ_n'' from Equations 31 and 32 gives

$$\varphi_1 = \frac{4}{3}\varphi_2 - \frac{1}{3}\varphi_3. \tag{33}$$

The same approach can be used at the opposite boundary of the trap. The expression for ϕ_1 and ϕ_{N_z} are added manually to the potential solutions at the end to give the full solution to the Poisson equation for each grid point.

3.1.6 Calculating the Force Acting on Each Particle

The total electric field $\vec{E}_{\text{tot}}$ is found by adding the self-consistent electric field emerging from the net charge of the plasma $\vec{E}_{\text{self}}$, as well as the potential well electric field $\vec{E}_{\text{trap}}$ contributions. The potential well electric fields are found by evaluating the discrete equation:

$$\vec{E}_{\text{trap}} = \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2h_z}. \tag{34}$$

The self-consistent electric field is calculated in a similar fashion using the self-consistent potential $\varphi_{h,z}$. The total electric field for trapped plasma is used to calculate the force F_{old} acting on each macroparticle in a time step δt corresponding to one loop iteration through [8]:

$$F_{\text{old}} = q_p \vec{E}_{\text{tot,old}} \tag{35}$$

where q_p is the weighted charge and the subscript $'old'$ refers to the variable inside the time loop at time t.

3.1.7 The Particle Pusher

Using the finite difference method, one can find the updated values, $z_{p,\text{new}}$, $v_{p,\text{new}}$ using the calculated force and their current values respectively by integrating the discrete equations of motion given in Equations 36 and 37

$$m\frac{v_{p,\text{new}} - v_{p,\text{old}}}{\delta t} = F_{\text{old}} \tag{36}$$

$$\frac{z_{p,\text{new}} - z_{p,\text{old}}}{\delta t} = v_{p,\text{new}} \tag{37}$$

where z_p is the axial position, v_p the velocity and m the weighted mass of the macroparticle. The system of equations is solved to find the variables noted by the subscript $'new'$ at time $t+\delta t$ which refers to the updated state of the macroparticle calculated using the variable with the subscript $'old'$ defined above. However, the velocity and position cannot be known simultaneously as one is used to calculate the other. Hence, we use the leapfrog iteration method [8] where $v(0)$ is pushed back at $v(\frac{\delta t}{2})$ using the force calculated at t=0. As a result, the kinetic and potential energies are adjusted to be calculated at the same time. The macroparticles are then 'pushed', i.e. their position $z_{p,\text{new}}$, velocity $v_{p,\text{old}}$, energy are updated [8].

4 Results and Discussion

The code was implemented in MATLAB, to profit from the built-in optimised function that solves spare matrix systems such as the ones used for the Laplace and the Poisson equation solvers. The error analysis on both solvers and the consistency of the results of the harmonic motion period between the adapted MATLAB and the C++ version of the PIC code and the theoretical result are presented below. This was done by making use of the strategic saving routines of the charge density, position, momentum and phase space distribution, for instance, implemented throughout the code. Moreover, some error checks to prevent some obvious mistakes such as a different number of electrodes and corresponding voltage value were implemented.

4.1 Test of the Implementation of the Laplace Equation Solver

An example solution of the potential field inside a Penning-Malmberg trap is shown in Figure 6.

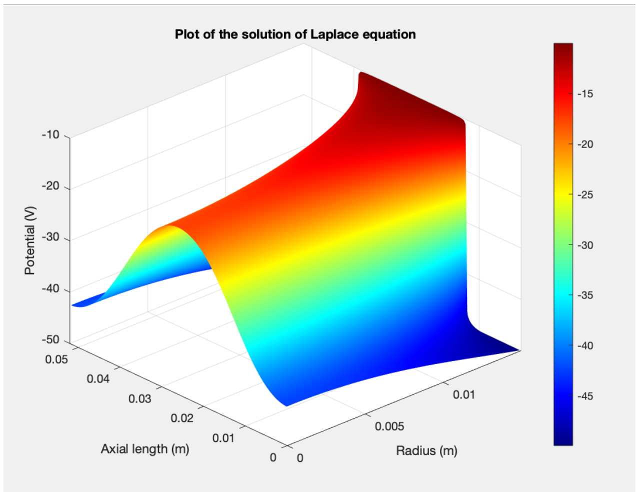


Figure 6: Solution to the Laplace equation using MATLAB with a trap of dimension of the ALPHA Penning-Malmberg trap and voltage $V1 = V4 = -50V$, $V2 = V3 = -10V$ on the electrodes shown in Figure 1. The Neumann and Dirichlet boundary conditions are applied on the grid of size $N_r = 128$, $N_z = 500$ and 3 radial zones.

The finite difference method is said to be consistent with the differential equation 10 if the truncation error $\epsilon_{i,j}$ is small. The value of $\epsilon_{i,j}$ is defined as the difference between the solution to the differential equation 10 for the previously calculated $\phi_{i,j}$ and the exact solution of the Laplace equation 6. The truncation error can be obtained by:

$$\epsilon_{i,j} = \nabla^2\phi_{i,j} - \nabla^2\phi. \tag{38}$$

Figure 7 shows that the magnitude of the truncation error $\epsilon_{i,j}$ increases with the number of radial grid points. In theory, the error due to the finite differential method is of the second order and should converge, gaining in precision from the discretization of the domain in a thinner grid. Hence, we can infer that the other sources of errors such as the loss of precision due to rounding decimal quantities has a higher effect. This can be explained by the large size of the linear system being solved to determine the electrostatic potential. Therefore, errors arise from the loss of precision due to computer rounding of decimal quantities and propagate quickly through the large number of calculations involved. A number of four grid points was determined to be small enough to reduce the loss of precision but sufficiently large to produce accurate results as the truncation errors are negligible, approximately of order 10^{-6}. If a higher-order precision with a lower number of grid points is required, it is suggested to use a 9-stencil third-order scheme. Analogously, the same behaviour was observed when increasing N_z with a fixed value N_r.

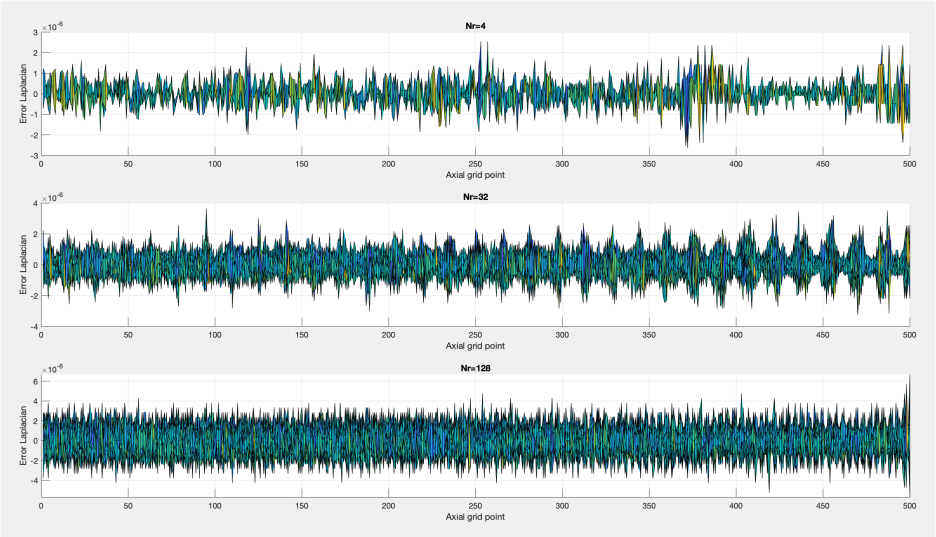


Figure 7: Truncation error through the trap as viewed from the z axis. These were obtained with a fixed value of $N_z = 500$ and varying numbers of radial grid points, $N_r = [4, 32, 128]$ and 3 radial zones. Similar results were obtained for different $N_z = 1000, 2000$.

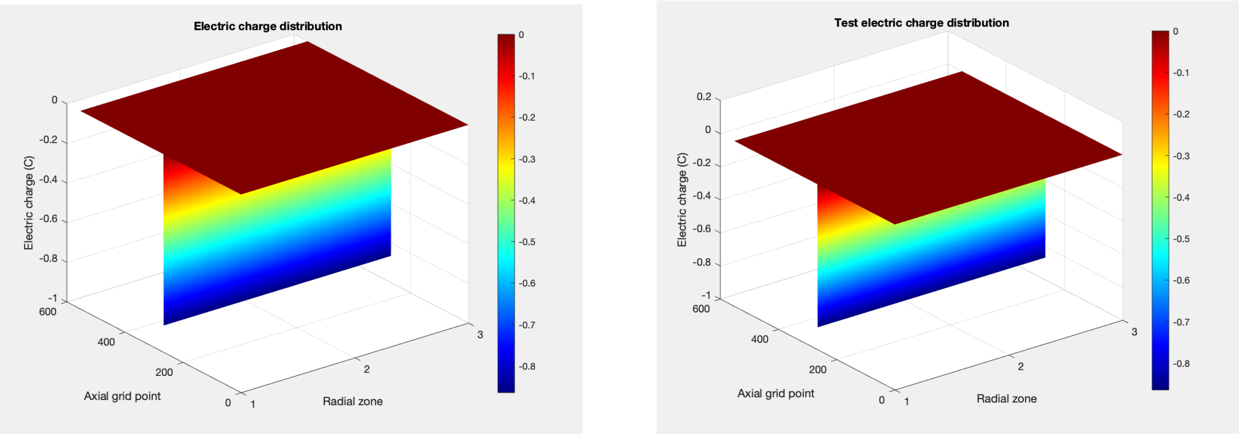
Therefore, we have demonstrated the accuracy and consistency of the method described above. Hence the most convenient number of grid points, in N_r and N_z, can be determined based on the computational cost and the required accuracy. In the case of an equilibrium plasma loading with a chosen density, the length of the grid can be determined to be smaller than the Debye length of the loaded plasma. This choice ensures the resolution of the interaction between each particle and hence the physicality of the results.

4.2 Test of the Implementation of the Poisson Equation Solver

The Bessel expansion and averaged radial zone method described above is said to be consistent with the differential equation 29 if the truncation error in the kth radial zone, ε_k, is small. The value of the truncation error is defined as the difference between the electric charge distribution retreived from the calculated self-potential $\bar{\varphi}_k$ and the exact electric charge distribution ρ_k used to calculate $\bar{\varphi}_k$ in the first place by solving Equation 29. The truncation error for the k_{th} radial zone can be obtained by:

$$\varepsilon_k = \epsilon_0 \nabla^2 \bar{\varphi}_k - \rho_k \tag{39}$$

The truncation error for the electric charge distribution and the retrieved electric charge distribution, shown in Figures 8a and 8b, respectively, was calculated. Figure 9 shows that the magnitude of the calculated truncation error ε_k for a grid of size of $N_r = 128$, $N_z = 600$ and 3 radial zones is negligible (approximately of the order 10^{-14} C and hence small compared to the 1C charge density). This result demonstrates the accuracy and consistency of the Bessel expansion and averaged radial zone numerical method. Hence, the most convenient number of grid points and radial zone, in N_r and N_z, can be determined based on the computational cost and the required accuracy.



(a)

(b)

Figure 8: (a) Electric charge distribution with voltage $V1 = V4 = -50V$ and $V2 = V3 = -10V$ on the electrodes of a trap of dimension of the ALPHA Penning-Malberg trap as shown in Figure 1. The Neumann and Dirichlet boundary conditions are applied on the grid of size $N_r = 128$, $N_z = 600$ and 3 radial zones. (b) Retrieved electric charge distribution from the calculated self-consistent potential solution to the Poisson equation.

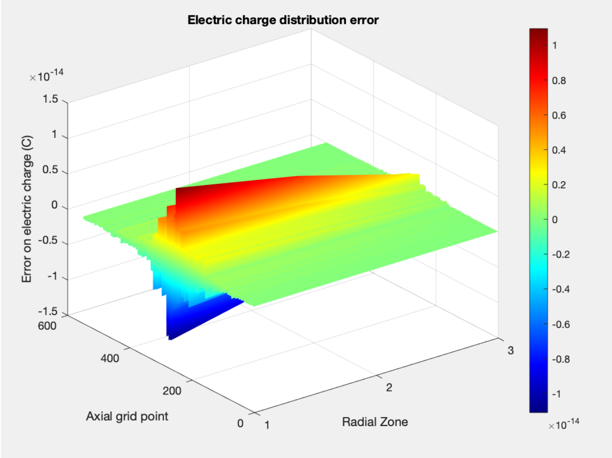


Figure 9: The truncation error of the electric charge distribution from the electric charge distribution shown in Figures 8a and 8b.

The motivation for investigating the above method is to decrease the computational cost and maintain the required accuracy compared to the finite difference method described in this report for the Laplace equation solver and used in the Poisson equation solver of the C++ code from Daniel Duque [5]. The results need to be assessed with respect to the programming language used and the method of implementation inside the model. The error associated with the Bessel expansion is of the order of $\frac{1}{N_b}$ in Equation 26 [21]. For a sufficiently large number of Bessel harmonics, the result for the potential should remain of tolerable accuracy. Moreover, using this method for plasma profile close to the functional shape of Bessel harmonics could lead to a faster convergence of the result for the potential. Finally, the averaged value over each radial zone is used. It improves the computational efficiency and speed of the model by lowering the number of calculations involved compared to the finite difference method.

4.3 Simple Harmonic Motion of a Particle in a Well

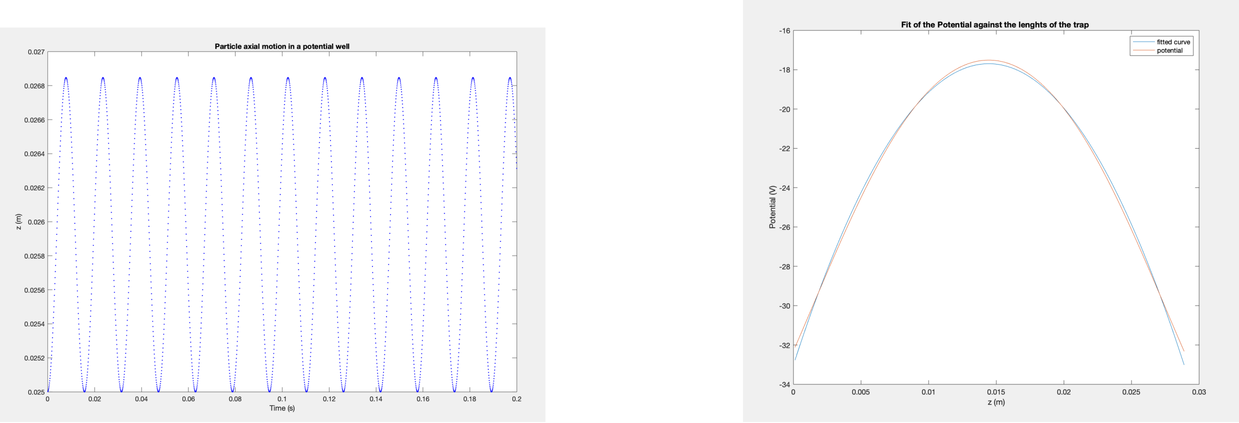
To test the implemented algorithm, the code was run with hypothetical values and the result compared to the expected values from the equations. In the case of a single particle loading, the self-potential can be ignored. The potential well can be approximated in the center of the trap with good accuracy by quadratic polynomial. Under this hypothesis, the particle is expected to undergo simple harmonic motion with a frequency

$$\omega^2 = -\frac{q}{m}\nabla\phi \tag{40}$$

where q is the charge, m the mass, and ϕ the potential of the well.
The quadratic equation of the fitted curve is given by

$$\phi_{fit} = (-7.49 * 10^4)z^2 + (2.03 * 10^3)z - 31.33 \tag{41}$$

where z is the number of grid cells. The quadratic fit of the potential is shown in Figure 10b.



(a)

(b)

Figure 10: (a) Harmonic motion of a particle loaded in a potential with a trap of dimension of the ALPHA Penning-Malmberg trap and voltage $V1 = V4 = -50V$, $V2 = V3 = -10V$ on the electrodes shown in Figure 1. The Neumann and Dirichlet boundary conditions are applied on the grid of size $N_r = 128$, $N_z = 500$ for three radial zone. The graph was obtained using MATLAB with 1000 loop iterations and $\delta t = 10^{-5}$s (b) Quadratic fit of the potential well in the first radial zone.

Under this hypothesis, for a particle with an hypothetical mass of $1K_g$ and charge $1C$ (values chosen for the sake of simplifying the calculation and for the validation of the implemented algorithm) in the trap, the angular frequency of oscillation is $\omega = 3.85 * 10^2$Hz and the corresponding frequency $\nu = 62$Hz. The frequency of oscillation of the particle with mass $1K_g$ and charge $1C$ shown in Figure 10a was found to be equal to 65Hz. The error on the result is believed to be due to the accuracy of the quadratic fit. The frequencies of the modes of oscillation were found using a Fast Fourier Transform function (FFT) written in MATLAB. The results of the period are consistent with the theoretical prediction of ω^2. This result confirms the validity of the algorithm without the self interaction field. The developed C++ code of Daniel Duque also shows simple harmonic motion of a particle in a well [5] .

4.4 The Poisson Boltzmann Equation for Plasma in Equilibrium

An equilibrium distribution that resembles the plasma inside the ALPHA trap is planned to be implemented in the C++ code. A cold non-neutral plasma such as the antiproton plasma used in ALPHA, is considered to have a plasma density following the Boltzmann distribution [16, 22, 23]. The particles inside a pure plasma may come to thermal equilibrium with each other and be confined by a static electric and magnetic field. The Boltzmann distribution predicts the collective behaviour of systems at thermodynamic equilibrium as a function of their constituent parts. The Poisson-Boltzmann equation is used to calculate the electric potential created by charged particles inside a non-equilibrium thermal plasma described by the Boltzmann distribution to determine the different electrostatic interactions. The Poisson-Boltzmann equation for a cylindrical geometry and derived via mean-field assumption is given by

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\varphi}{\partial r}\right)+\frac{\partial^2\varphi}{\partial z^2}=4\pi n_0 e^{\frac{-[-e\varphi(r,z)+\frac{1}{2}mw(\Omega-w)r^2]}{KT}} \tag{42}$$

where Ω is the cyclotron frequency, φ the electric potential, $-e$, m and v are the particle charge, mass and velocity, respectively. The parameters n_0, T and w are the total number of particles, the energy and the canonical angular momentum in the system [22]. In the case of cylindrical trap and a finite plasma, numerical solution of Equation 42 is obtained through the convergence of an iteration procedure. The solution of the Laplace equation is used as a starting guess for the potential. At the end of the iterative procedure the macroparticle number density $n(r,z)$ and the potential $\varphi(r,z)$ are calulated and extracted as outputs. These methods are planned to be implemented in the C++ code of Daniel Duque.

5 Conclusion

A PIC code model for the simulation of the equilibrium conditions for an antiproton and electron plasma trapped in a Penning-Malmberg trap was developed and described in this report. This code is planned to be used for the study of the electron/antiproton separation technique used in the ALPHA experiment at CERN, called the e-kick procedure. The study of the PIC code developed by Federico Peinetti provided a basis for the general understanding of the functioning of PIC codes [1]. Moreover, the context of the project and the physics motivation were clearly defined. In particular, a good understanding of the finite difference numerical method implemented in the Laplace equation solver was obtained. Similarly, the Bessel expansion and averaged radial zone numerical method implemented in the Poisson equation solver was understood in detail. The error analysis on both solvers and the consistency of the results of the harmonic motion period between the two developed versions of the PIC code and the theoretical result confirms the validity of the implemented model. Some challenges were faced in the investigation of Peinetti's PIC code due to the lack of documentation of the existing code [1]. The code being intrinsically linked to a complex loading method prevented trivial modifications of the code for different loads and processes. Hence, further investigations should be performed on the C++ code developed by Daniel Duque. The C++ code, as of today, can load a set of particles and update their position and velocity at each time step, giving the time evolution of the plasma inside a trap. The C++ code uses the finite difference method described in this report for both the Poisson and Laplace equation solvers [5]. We plan to implement the Bessel expansion and averaged radial zone method for possible improvements of the computational efficiency while retaining result accuracy. We also plan to load an equilibrium distribution that resembles the plasma inside the ALPHA trap to perform the e-kick procedure described in this report, using an ergodic quiet start loading [8].

6 Acknowledgements

I would like to express my special thanks to my supervisor Dr. William Bertsche who guided and supported me in the research. I would like to express my gratitude to Federico Peinetti for his preliminary work of building the model, writing non-formal documentation and supporting me throughout the project.

References

[1] F. Peinetti, F. Peano, G. Coppa, and J. Wurtele, "Particle-in-cell method for parallel dynamics in magnetized electron plasmas: Study of high-amplitude bgk modes," *Journal of computational physics, Vol.218*, pp. 102–122, 2006.

[2] P. Dirac, "The Quantum Theory of the Electron," *Royal Society A: Mathematical, Physical and Engineering Sciences, Vol.117*, pp. 610–622, 1928.

[3] C. D. Anderson, "The Apparent Existence of Easily Deflectable Positives," *Science, Vol.76*, pp. 238–239, 1932.

[4] W. Bertsche, E. Butler, M. Charlton, and N. Madsen, "Physics with antihydrogen," *Journal of Physics B: Atomic, Molecular and Optical Physics, Vol.48*, pp. 1–25, 2015.

[5] D. Duque, "Plasma PIC code for use in Antihydrogen Physics," *Master Thesis at the University of Manchester, School of Physics and Astronomy*, 2020. C++ code can be found at: `https://github.com/Daniel32Duque/PIC-Trapped-Plasma`.

[6] F. Chen, *Introduction to Plasma Physics and Controlled Fusion.* Springer, 3rd ed., 1984.

[7] R. C. Davidson, *Physics of non-neutral plasmas.* London: Imperial College Press, 2nd ed., 2001.

[8] C. Birdsall and A. Langdon, *Plasma Physics via Computer Simulation.* CRC Press, 1st ed., 1975.

[9] J. Jackson, *Classical Electrodynamics.* Wiley, 3rd ed., 1999.

[10] C. Amole *et al.*, "The ALPHA antihydrogen trapping apparatus," *Nuclear Instruments and Methods in Physics Research, Vol.735*, pp. 319–340, 2013.

[11] S. Prasad and T. O'Neil, "Waves in a cold pure electron plasma of finite length," *The Physics of Fluids, Vol.26*, pp. 278–281, 1982.

[12] E. Butler, "Antihydrogen formation dynamics and trapping," *PHD Thesis from the Department of Physics at Swansea University*, 2011.

[13] X. Huang *et al.*, "Steady-State Confinement of Non-neutral Plasmas by Rotating Electric Fields," *Physical Review Letters, Vol.78*, pp. 875–878, 1997.

[14] G. Baur *et al.*, "Production of Antihydrogen," *Physics Letters, Vol.368*, pp. 251–258, 1996.

[15] G. B. Amsler *et al.*, "Production and detection of cold antihydrogen atoms," *Nature, Vol.419*, pp. 419–456, 2002.

[16] C. Driscoll, J. Mamlmberg, and K. Fine, "Observation of Transport to Thermal Equilibrium in Pure Electron Plasmas," *Physical Review Letters, Vol.60*, pp. 1290–1293, 1987.

[17] T. O'Neil and P. Hjorth, "Collisional dynamics of a strongly magnetized pure electron plasma," *The Physics of Fluids, Vol.28*, pp. 3241–3252, 1985.

[18] B. Bottura, "E-kick potential well modifications in the ALPHA experiment at CERN," *Internship report, University of Manchester*, 2017.

[19] G. Andersen *et al.*, "Evaporative Cooling of Antiprotons To Cryogenic Temperatures," *Physics Letters, Vol.105*, pp. 1–5, 013003, 2010.

[20] P. Frey, "The finite difference method ," *Université Sorbone Paris, Lecture notes*, pp. 79–92, 2017. `https://www.ljll.math.upmc.fr/frey/cours/UdC/ma691/ma691_ch6.pdf`.

[21] S. Storey, " The convergence of Fourier-Bessel expansions," *Computer Journal, Vol.10*, pp. 402–405, 1968.

[22] A. Peurrung and J. Fajans, "Non-neutral plasma shapes and edge profiles," *Physics of Fluids B: Plasma Physics 2, Vol.693*, pp. 693–699, 1989.

[23] S. Prasad and T. O'Neil, "Finite length thermal equilibria of a pure electron plasma," *The Physics of Fluids 22, Vol.278*, pp. 278–281, 1987.