Visualization of the Plasma Frequency by means of a Particle Simulation using a Normalized Periodic Model

Grunwald, Joschka; Mercorelli, Paolo

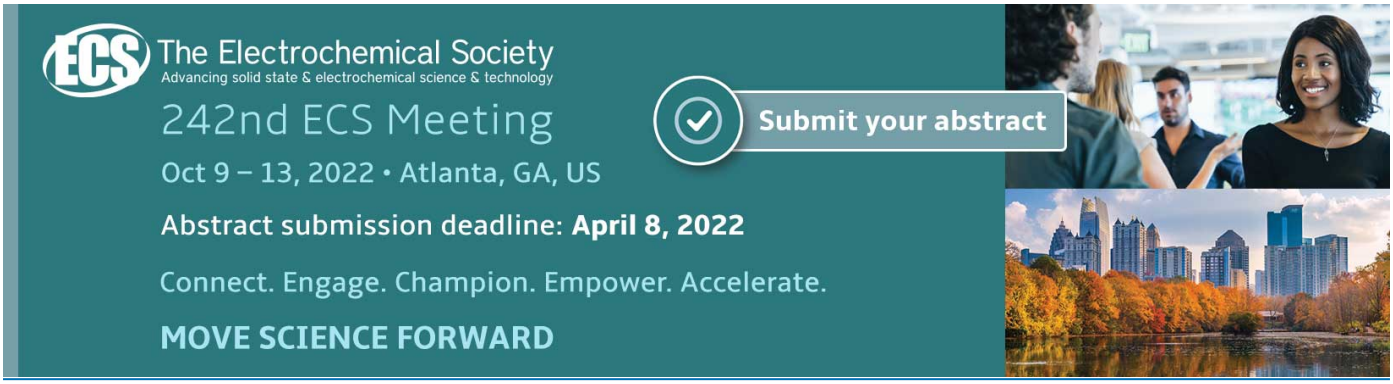
Published in:
Journal of Physics: Conference Series

DOI:
10.1088/1742-6596/2162/1/012016

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

Visualization of the Plasma Frequency by means of a Particle Simulation using a Normalized Periodic Model



Visualization of the Plasma Frequency by means of a Particle Simulation using a Normalized Periodic Model

Joschka Grunwald, Paolo Mercorelli

Institute of Product and Process Innovation, Leuphana University of Lueneburg, Lueneburg, Germany

E-mail: joschkagrunwald@gmail.com, mercorelli@uni.leuphana.de

Abstract. In plasmas the atoms are dissociated into ions and free electrons. Due to the high mobility of the electrons, plasmas are a vibrating medium. As a result, a plasma frequency can be determined, which is an important parameter for characterizing a material. In this study, the plasma frequency is to be visualized. For this purpose, a mathematical model will be set up to describe the electron kinetics. The electrons are briefly deflected with a force against the static ion background in order to examine the interaction between the particles in the form of electrostatic fields. The discretized equations are implemented in a numerical particle simulation to visualize the movement of the electrons. The Particle-In-Cell (PIC) method is used for this. A spatial mathematical grid is created on which the simulated particles are distributed in the cells. The equations of motion and field are then solved on this grid at different times.

Keywords: Plasma Frequency, Particle-In-Cell Method, Periodic Normalized Plasma Model, Particle Simulation, Poisson Equantion, Leapfrog Method

1. Introduction

The matter existing on our earth occurs mainly in the solid, liquid or gaseous aggregate states. These states are extremely rare in the visible universe. More than 99% of matter is in the plasma state, which is also known as the fourth aggregate state [1, cf.: P.227]. The form in which a material exists depends on its thermal energy. If sufficient thermal energy is supplied to a gas, a phase transition occurs. The kinetic energy at the atomic level is so enormous that the binding energy between the nucleus and the electrons is overcome. The atoms dissociate into ions and electrons. This process is called ionization [2, cf.: P.1f.]. Due to the unbound particles, an ionized gas has completely new properties and is therefore no longer called a gas but a plasma.

The free electrons in a plasma create collective phenomena such as the dynamics of space charge fluctuations [3, cf.: P.1]. These charge oscillations have the consequence that electromagnetic radiation with a frequency lower than the natural frequency of the plasma is shielded and reflected on the surface. Due to the lower mass and thus higher mobility of the electrons, they indicate the effective shielding frequency. This shielding frequency or natural frequency is also

referred to as the plasma frequency and is given by

$$\omega_{p_e} = \sqrt{\frac{e^2 \rho_e}{\epsilon_0 m_e}} \quad . \tag{1}$$

The elementary charge is represented by e, ρ_e indicates the electron density, m_e the electron mass and ϵ_0 the electric field constant [4, P.529f.].

In this study, the plasma frequency is visualized by means of a Particle-in-Cell (PIC) algorithm, see Fig. 1, which is known for such simulations [5, P.79]. The PIC algorithm is a traditional, simple and valued method for the simulation of plasma phenomena [6]. The kinetics of the electrons are calculated for a one-dimensional periodic plasma model. In a plasma, the positively charged ions and the negatively charged electrons create electrical fields. The particles move according to the forces created by their own and external fields. By the first Maxwell equation, which is also known as Gauss's law, the electric fields can be calculated based on the charge density. The charge density in turn results from the particle positions. The motion of the particles is deduced from the second Newtonian axiom. For this purpose, the forces that result from the electric fields and act on the individual particles are determined. The plasma particles generate the fields, which in turn represent the basis of the particle movements. The PIC method also acts in this way. The electric fields of the charged particles are calculated. Then the electrons are moved slightly and the fields are calculated again. By repeating this computation cycle, the movements of the electrons are simulated step by step.

Since it is not practical to determine the fields or the forces between the charged particles directly using Coulomb's law, a spatial mathematical grid with equidistant grid points is introduced. The charge densities ρ_j of the particles are accumulated at the grid points using an interpolation scheme, see Fig.1, so that Gaussian law can be used to solve the field equations at these discrete points. Then the calculated electric fields E_j are interpolated back, whereby the forces F_i acting on the particles can be calculated. In the following, the interpolations are referred to as weightings. They are to be integrated into the computation cycle already described, as shown in Fig. 1. For a clear differentiation of the various calculation variables, particle-related values are provided with the index i and grid-related values with the index j. The individual computing cycles are also referred to as time steps Δt and can be carried out as often as required. The simulation thus moves in discrete steps z, with $z \in \mathbb{N}$, through time t. The simulation program shows the positions of the electrons between the time steps. In this way, the movement of the electrons is gradually represent and thus the plasma frequency is visualized.

The main nomenclature
z: number of simulated time steps Δt
X_j: $j-th$ grid point
E_j: electric field at the $j-th$ grid point
ρ_j: charge density at the $j-th$ grid point
Φ_j: electric potential at the $j-th$ grid point
Δx: distance between the grid points and width of a super particle
x_i: $i-th$ super particle
F_i: Force that acts on the $i-th$ super particle
q_s: charge of a super particle

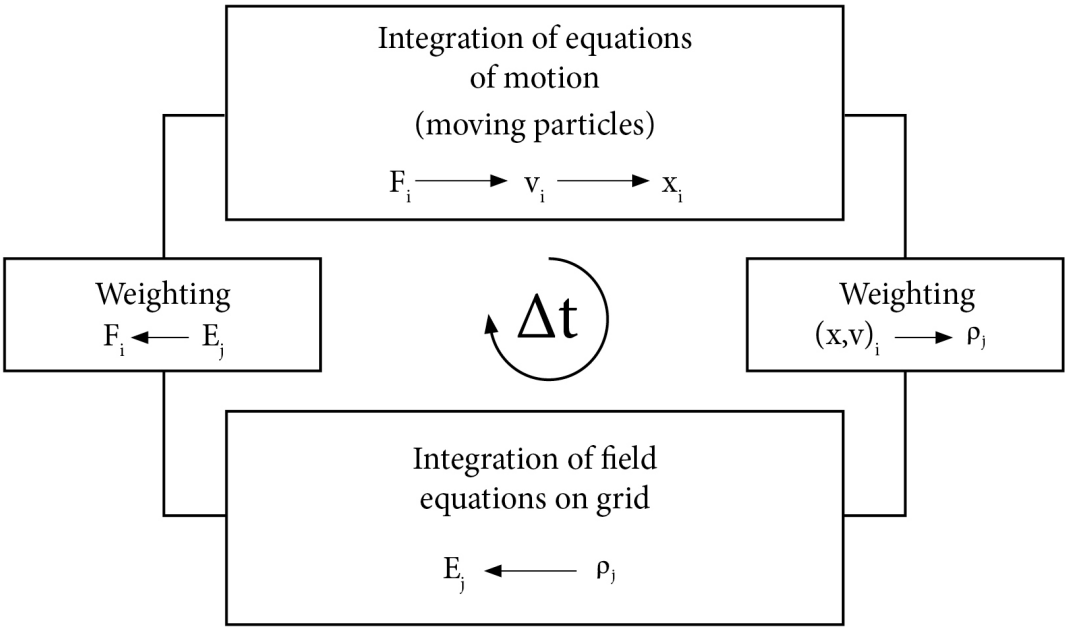


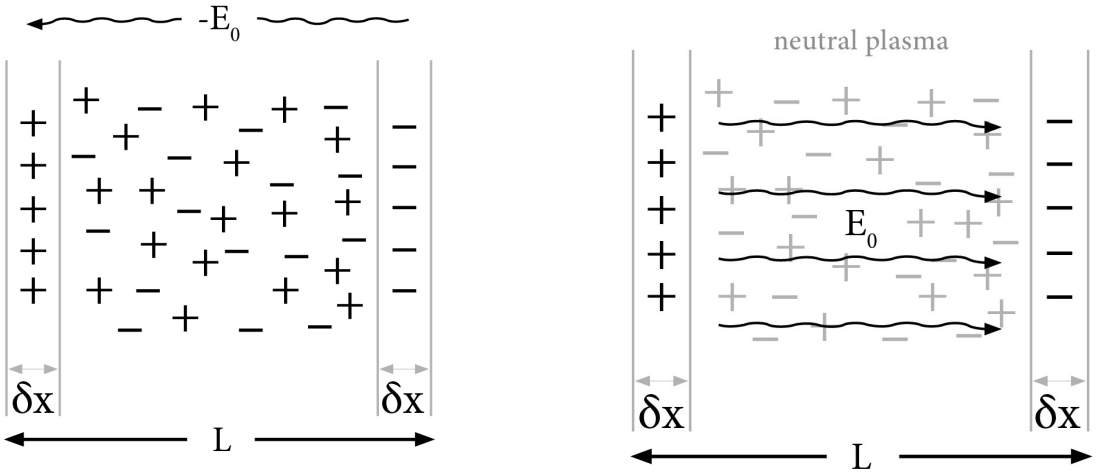
Figure 1: Computing Cycle

2. Model Description

The following model serves as the basis for visualizing the plasma frequency. A layer of width L is considered from a homogeneous, in total neutral plasma. In order to initialize the particle kinetics, the plasma model must be brought out of its equilibrium position. The model is therefore briefly exposed to an external electric field $-E_0$. Due to their higher mobility, the electrons are deflected. The deflaction δx of the electrons is proportional to the strength of $-E_0$. The significantly more massive and therefore more inert ions move only slightly. For a single time step they are regarded stationary. Therefore the ions are only considered as a positively charged, neutralizing background. As shown in Fig. 2a, the deflection of the electrons creates two oppositely charged spatial zones at the edges. After removing the external electric field $-E_0$, an electric field E_0 remains in the interior of the plasma due to these space charge zones, see Fig.2b. The electric field E_0 generates a force on the plasma particles that accelerates them towards their origin. Due to their inertia, the particles move beyond their original position. An opposing electric field arises. The model therefore represents a harmonic oscillator [3, cf.: P.11]. The frequency of the electron oscillation is the plasma frequency.

In the following, the kinetics of the electrons on the basis of the self-generated electrical fields are discussed. The creation of magnetic fields and the collision of particles are neglected. In addition, a reduction in the computational effort is generated by combining the individual electrons into super particles. The super particles can be understood as uniformly charged finite electron clouds. The position of a super particle is represented by the center of mass and the velocity by averaging the individual values. A super particle therefore represents a group of electrons and can be interpreted as a small part of the phase space. So a super particle represents a constant number of electrons.

In the following, the described model is simulated in one-dimensional space with periodic bound-



(a) Deflection of the electrons in a neutral plasma

(b) The resulting electrical field in a neutral plasma

Figure 2: The effect of an external electric field on a neutral plasma

ary conditions and thus the plasma frequency is visualized. No specific plasma matter is assumed here. A standardized, generally valid representation of the plasma frequency is generated.

3. Particle-in-Cell Methode

Figure 1 shows the basic computing cycle of the PIC method. In this paragraph the individual calculations for a simulated time step Δt are presented. After the weightings and thus the interpolations of the charge density ρ_j and the force F_i have been presented, the next two paragraph describe the numerical methods for calculating the electric field $E(x_i(t))$ and the new positions $x_i(t)$ as well as velocities $v_i(t)$. At the end of a paragraph, an implementation option in the Matlab programming language is given.

3.1. Weightings

In the calculation cycle, two calculations are titled as weightings. On the one hand, the force $F_i(z)$ acting on the particle, which is required for the integration of the equations of motion, is calculated. On the other hand, the charge density $\rho_j(z)$, which is required for the integration of the field equations, is determined. These weightings imply a form of interpolation through which generates a connection between the particle quantity and the spatial grid. In order to prevent a particle from accelerating itself, the same interpolation method is used for the weighting of the force $F_i(z)$ and the charge density $\rho_j(z)$ [7, cf.: P.129] .

The super particle model serves as the basis for weighting the simulated particles. The super particles are viewed as finite electron clouds with a uniform density. As shown in Fig. 3, the electron clouds have a width of Δx around the position of their center of mass $x_i(z)$. Assuming the $i-th$ super particle is in the interval between the $j-th$ and the $(j+1)-th$ grid point, so $X_j < x_i(t) < X_{j+1}$, then the $i-th$ super particle creates a charge density $\rho_{j,i}(z)$ at the grid point X_j

$$\rho_{j,i}(z) = \frac{\Delta x - x_i(z) - X_j}{\Delta x}\frac{q_s}{\Delta x} = \frac{X_{j+1} - x_i(z)}{\Delta x}\frac{q_s}{\Delta x} \tag{2}$$

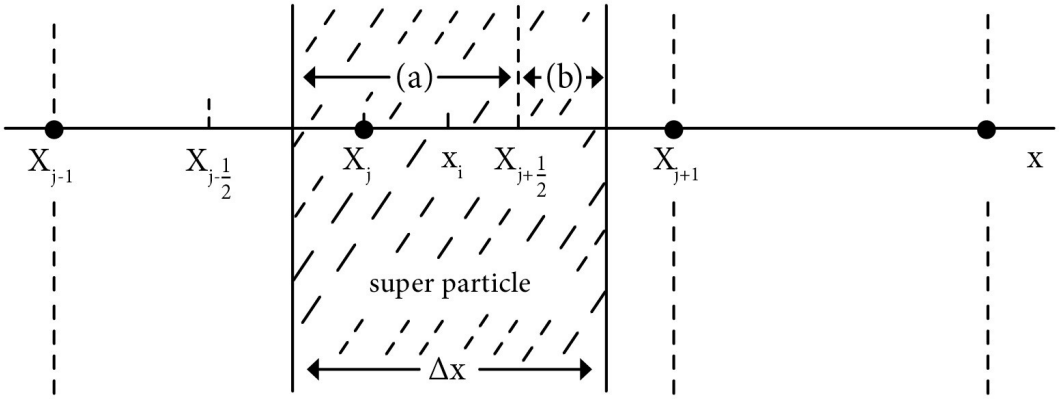


Figure 3: PIC Weighting of the Super Particle x_i

and correspondingly for the charge density $\rho_{j+1,i}(z)$ in the grid point X_{j+1}

$$\rho_{j+1,i}(z) = \frac{x_i(z) - X_j}{\Delta x} \frac{q_s}{\Delta x} \quad , \tag{3}$$

where qs is the charge of a super particle. Δx represents both the distance between two grid points and the width of a super particle. If an equivalent weighting is carried out for all super particles, the charge density $\rho_j(z)$ for the grid point X_j can be summed up by the proportional charge densities of the super particles $\rho_{j,i}(z)$.

$$\rho_j(z) = \sum_{i=1}^{N_{j,s}} \rho_{j,i}(z) \tag{4}$$

Where $N_{j,s}$ represents the number of super particles in the $j-th$ grid cell. This procedure corresponds to a linear interpolation and is referred to as Particle in Cell (PIC) weighting [8, cf.: P.20f.].

For the weighting of the electric fields $E(x_i(z))$ on the super particles, the procedure is equivalent. The electric field for the super particle at position $x_i(z)$ is given by

$$E(x_i(z)) = \Big[\frac{X_{j+1} - x_i(z)}{\Delta x}\Big] E_j(z) + \Big[\frac{x_i(z) - X_j}{\Delta x}\Big] E_{j+1}(z). \tag{5}$$

Here, Δx corresponds to the distance between the grid points X_j and X_{j+1}. $E_j(z)$ or $E_{j+1}(z)$ represent the calculated electric fields in these grid points, see section 3.3). If the electric field $E(x_i(z))$ and the charge q_s of a super particle at position $x_i(z)$ are known

$$F(x_i(z)) = E(x_i(z))q_s \tag{6}$$

can be used to calculate the force F_i acting on the particle $x_i(z)$.

The weighting of the electric fields on the super particles is shown together with the particle movements in paragraph 3.2. The implementation of the particle density in the grid points is implemented as follows.

```
function [] = density(dx, rho0, npart, ngrid)

global x rho_el rho_ion qs;

qs = -rho0 * (ngrid/npart); % normalized electron charge
rho_el = zeros(1,ngrid+1);  % electron density

for i = 1:1:npart
    xp = x(i)/dx;
    j1 = floor(xp);
    j2 = j1 + 1;
    g2 = xp - j1;    % weighting factor
    g1 = 1 - g2;     % weighting factor
    j1 = j1+1;
    j2 = j2+1;
    rho_el(j1)=rho_el(j1)+qs*g1; % accumulated charge density
    rho_el(j2)=rho_el(j2)+qs*g2; % ...in the grid points
end

rho_el(1)=rho_el(1)+rho_el(ngrid+1);
rho_el(ngrid+1)=rho_el(1);  % periodic conditions
rho_ion=rho0;  % ion density (neutralizing background)
end
```

By applying a for-loop, the weighting factors of each individual super particle are calculated for the corresponding grid points. The charge densities of the grid points are then accumulated. Finally, the periodic boundary conditions are implemented and the ion density is defined.

3.2. Integration of the equation of motion - Leapfrog method

The Leapfrog method is a fast, second-order numerical integration method based on little information [9, cf.: P.2f.]. By integrating Newton's second axiom using the Leapfrog method, the new velocity and the new position of the modeled particles can be calculated from the force F_i. In one-dimensional space

$$F(x_i(z)) = m_s \frac{dv_i(z)}{dt} \tag{7}$$

and

$$v_i(z) = \frac{dx_i(z)}{dt} \tag{8}$$

are the relevant equations for calculating the trajectories of the electrons. $F(x_i)$ represents the force that acts on the particle x_i, m_s indicates the mass of a super particle, v_i represents the velocity of the $i-th$ particle and t the time.

The Leapfrog method can be interpreted as an extension of the Euler method. But the Leapfrog method generates higher accuracy by considering a variable not at the beginning but in the middle of an interval [9, cf.: P.2f.]. Takes into account that the particles move gradually through time with Δt, then

$$x_{z+1} = x_z + \Delta t v_{z+\frac{1}{2}} \qquad (z = 1, 2, ..., n) \tag{9}$$

for the new position and

$$v_{z+\frac{3}{2}} = v_{z+\frac{1}{2}} + \Delta t \frac{F(x_{z+1})}{m_e} \qquad (z = 1, 2, ..., n) \tag{10}$$

for the new velocity are the approximated result from Eq. (7) and (8). The number of time steps is given by z with $z \in \mathbb{N}$ and runs from zero (initial position) to n (end of simulation) with $n \in \mathbb{N}$. The step length is essential for the accuracy of the approximation. The longer the time step Δt is chosen, the more imprecise the process becomes.

Due to the fact that v is postponed over time, x and v are not known at the same time. They must be calculated alternately, as shown in Fig. 4. Furthermore, the Leapfrog method achieves

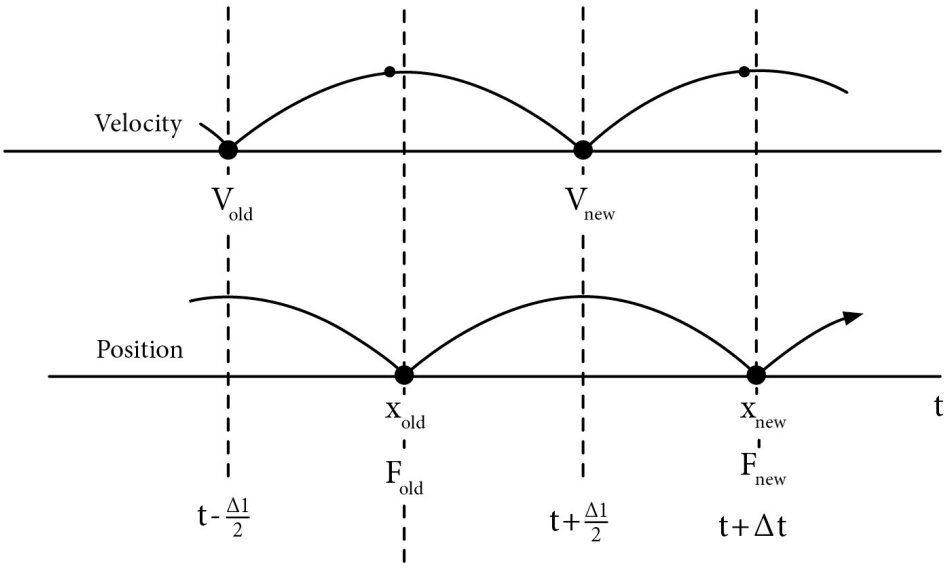


Figure 4: Leapfrog Method

improved stability through the time deflection of v [8, cf.: P.56.]. Compared with alternative second order integration methods it is a stable and relatively precise method for calculating the trajectories of the electrons.

To implement the particle movements, the electrical fields must be weighted. As in paragraph 3.1 a for-loop has been used.

```
function [] = part_move(dt, dx, npart, q_over_m)

global x v E;

for i = 1:1:npart
    xp = x(i)/dx;
    j1 = floor(xp);
    j2 = j1 + 1;
    g2 = xp - j1;  % weighting factor
    g1 = 1 - g2;   % weighting factor
```

```
    j1 = j1+1;
    j2 = j2+1;
    E_xi = g1*E(j1) + g2*E(j2); % electric field at x_i
    v(i)=v(i)+q_over_m*E_xi*dt; % calculation of the new velocity
end
x = x + dt*v;  % calculation of the new position
end
```

The new velocity and new positions can then be calculated using the electric fields and the normalized ratio of charge to mass.

3.3. Integration of field equation - Gaussian law

In order to be able to deduce the forces acting on the individual super particles from the charge density $\rho_j(z)$, the electric fields at the grid points must be calculated. With the first Maxwell equation, which is also known as Gauss's law, the divergence of the electric field $E_j(z)$ at the grid points X_j are determined.

$$divE_j(z) = \frac{\rho_j(z)}{\epsilon_0} \qquad \text{or} \qquad \vec{\nabla}E_j(z) = \frac{\rho_j(z)}{\epsilon_0} \tag{11}$$

The Nabla differential operator $\vec{\nabla}$ represents a vector whose components contain the partial derivative operators [10, cf.: P.54]. Conservative fields can be expressed in terms of the negative gradient of the potential. The electrostatic and thus conservative field considered here is therefore also given by

$$E_j(z) = -grad\Phi_j(z) \quad \text{or} \quad E_j(z) = -\vec{\nabla}\Phi_j(z) \tag{12}$$

The charge density in the grid points is given by $\rho_j(z)$ and the electrical potential by $\Phi_j(z)$. The electric field constant is represented by ϵ_0. A combination of Eq. 11 and 12 yields to

$$div(grad\Phi_j(z)) = -\frac{\rho_j(z)}{\epsilon_0} \qquad \text{or} \qquad \vec{\nabla}^2\Phi_j(z) = -\frac{\rho_j(z)}{\epsilon_0} \quad , \tag{13}$$

which is called the Poisson Equation [8, cf.: P.16]. The discretization of Eq. (12) and (13) results in

$$E_j(z) = \frac{\Phi_{j-1}(z) - \Phi_{j+1}(z)}{2\Delta x} \tag{14}$$

and

$$\frac{\Phi_{j-1}(z) - 2\Phi_j(z) + \Phi_{j+1}(z)}{(\Delta x)^2} = -\frac{\rho_j(z)}{\epsilon_0} \tag{15}$$

according to the finite difference method [8, cf.: P.16]. Equation (15) can also be written in matrix form.

$$\begin{bmatrix} -2 & 1 & 0 & ... & 1 \\ 1 & -2 & 1 & ... & 0 \\ 0 & 1 & \ddots & \ddots & \vdots \\ \vdots & ... & \ddots & \ddots & \vdots \\ 1 & 0 & ... & 1 & -2 \end{bmatrix} \begin{bmatrix} \Phi_1(z) \\ \Phi_2(z) \\ \vdots \\ \Phi_{n-1}(z) \\ \Phi_n(z) \end{bmatrix} = \begin{bmatrix} p_1(z) \\ p_2(z) \\ \vdots \\ p_{n-1}(z) \\ p_n(z) \end{bmatrix} \quad \text{with} \quad p_j(z) = -\frac{(\Delta x)^2\rho_j(z)}{\epsilon_0} \quad . \tag{16}$$

The vectors of the matrix in Eq. (16) are linearly dependent. Therefore the additional information that the charge density of the entire system $\langle\rho\rangle$ is

$$\langle\rho\rangle = \sum_{j=1}^{m} \rho_j(z) = 0 \tag{17}$$

must be inserted. Where m indicates the number of grid points. The system of equations can thus be solved uniquely. The Gaussian elimination method is used to calculate the potentials $\Phi_j(z)$. Then the electric fields in the grid points can be determined with the potentials according to Eq. (14).

For the implementation of Poisson's equation, the matrix or the system of equations shown in Eq. (16) is created. The electrical potentials are then determined with a function using the Gaussian elimination method. The electrical fields are then calculated by using a for-loop. Finally, the periodic boundary conditions are implemented.

```
function [] = E_Poisson(dx, ngrid)

global rho_el rho_ion E Phi;
rho = rho_el + rho_ion; % total charge density

D = diag(-2*ones(ngrid,1)); % Matrix
for i = 1:1:ngrid-1
    D(i+1,i)= 1;
    D(i,i+1)= 1;
end
D(ngrid,1)= 1;
D(1,ngrid)= 1;

p = -(dx^2)*rho;
Phi = gaussAlgorithmus(D,p,ngrid); % Gaussian elimination
Phi = Phi(1:ngrid); % el. potentials
E = zeros(1,ngrid+1);
sum_E = 0;

for j = 1:1:ngrid % calculation of the el. fields
    if j==1
        E(j) = 0.5*(Phi(ngrid)- Phi(j+1))/dx;
    elseif j==ngrid
        E(j) = 0.5*(Phi(j-1)- Phi(1))/dx;
    else
    E(j) = 0.5*(Phi(j-1)- Phi(j+1))/dx;
    end
    sum_E = sum_E + E(j);
end
E = E - sum_E/ngrid;
E(ngrid+1) = E(1);  % implementation of boundary conditions
end
```

4. Simulation Results

For the visualization and the plasma frequency there is the problem that a one-dimensional model with several thousand super particles cannot be clearly represented. For the visualization of the plasma frequency, the direct representation of the kinetics of the super particles on a line is not functional. For this reason, the movements of the particles are shown using their density functions in the grid points. For each time step, the particle densities are visualized in the grid points, creating a "film" that reflects the particle distribution of the entire system over all time steps.

The relevant complexity of the PIC algorithm is given by the calclation of the elctrical potentials Φ_j. To calculate Φ_j in accordance with Gaussian elemination method the complexity of the calcutaions is proportional to n^3, where n is the number of the grid points.

In Fig. 5 the electron densities in the grid points are visualized. On a grid of length 2π with 100 grid points, 10^4 super particles were implemented.

Figure 5 shows that the system, after it has been deflected once, follows the oscillations of a harmonic oscillator.

5. Conclusion

This paper deals with the visualization of the plasma frequency using a standardized one-dimensional model. The applied PIC algorithm is an excellent method for simulating the electron kinetics. Using Poisson equation the electric fields are calculated from the electral potentials. Alternatively, the electric fields can also be calculated directly from the electron densities using the Gaussian law and the trapezoidal method as an integration method. The simulation confirms the results to be expected from the model.

References

[1] Heuermann, H.: *Mikrowellentechnik: Feldsimulation, nichtlineare Schaltungstechnik, Komponenten und Subsysteme, Plasmatechnik, Antennen und Ausbreitung*. Springer Fachmedien Wiesbaden, 2020. http://dx.doi.org/10.1007/978-3-658-29023-8. http://dx.doi.org/10.1007/978-3-658-29023-8. – ISBN 9783658290238

[2] Kegel, W.H.: *Plasmaphysik: Eine Einführung*. Springer Berlin Heidelberg, 2013. – ISBN 9783642587719

[3] Stroth, Ulrich: *Plasmaphysik: Phänomene, Grundlagen und Anwendungen*. 2. Springer Spektrum, 2018. http://dx.doi.org/10.1007/978-3-662-55236-0. http://dx.doi.org/10.1007/978-3-662-55236-0. – ISBN 978–3–662–55235–3,978–3–662–55236–0

[4] Grimsehl, E. ; Lösche, A.: *Struktur der Materie*. Vieweg+Teubner Verlag, 1990. – ISBN 9783663014522

[5] Lapenta, Giovanni: Kinetic Plasma Simulation:Particle In Cell Method / Center for mathematical Plasma Astrophysics, KU Leuven,Belgium. 2015. – Forschungsbericht

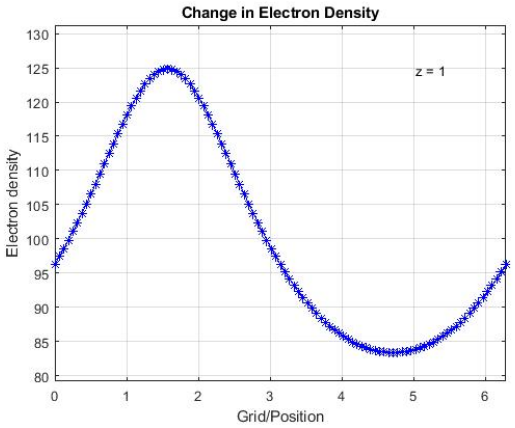
[6] Philip L. Pritchett, Manfred S. Jörg Büchner B. Jörg Büchner ; Dum, Christian T. (Hrsg.): *Space Plasma Simulation*. 1. Springer-Verlag Berlin Heidelberg, 2003 (Lecture Notes in Physics 615). – ISBN 3540006982,9783540006985

[7] Hockney, R.W. ; Eastwood, J.W.: *Computer Simulation Using Particles*. CRC Press, 1988. – ISBN 9781439822050

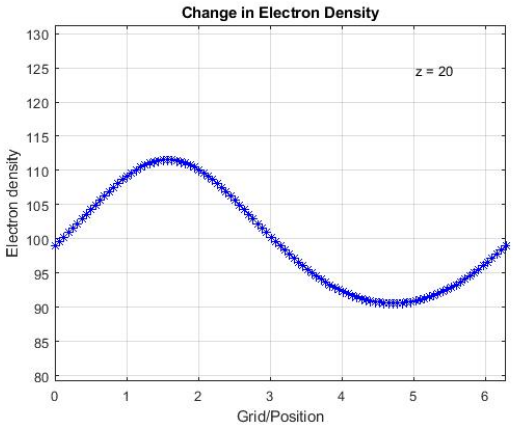
[8] Birdsall, A.B C.K; L. C.K; Langdon: *Plasma Physics via Computer Simulation*. CRC Press, 2004 (Series in Plasma Physics and Fluid Dynamics). – ISBN 978–1–4822–6306–0,1482263068,277–277–277–2

[9] Young, Peter: *The leapfrog method and other "symplectic" algorithms for integratingNewton's laws of motion*, April 2014. – Physics 115/242

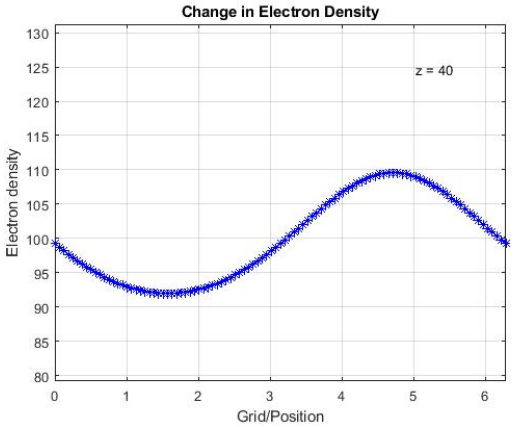
[10] Schwab, A.J.: *Begriffswelt der Feldtheorie: Elektromagnetische Felder, Maxwell-Gleichungen, Gradient, Rotation, Divergenz*. Springer Berlin Heidelberg, 2019. – ISBN 9783662583920



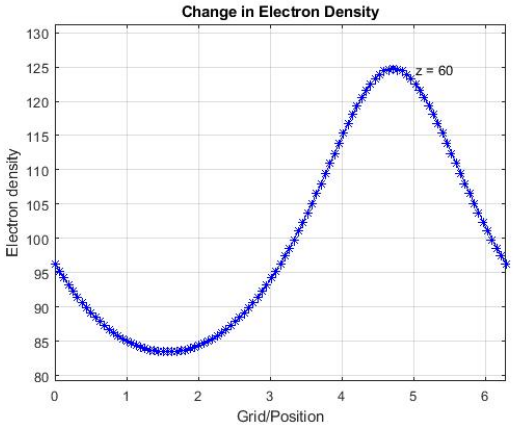
(a) $z = 1$



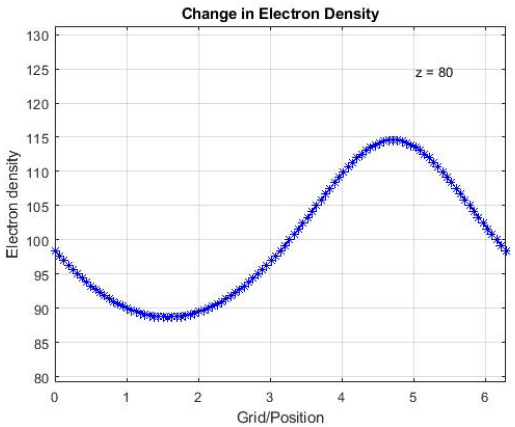
(b) $z = 20$



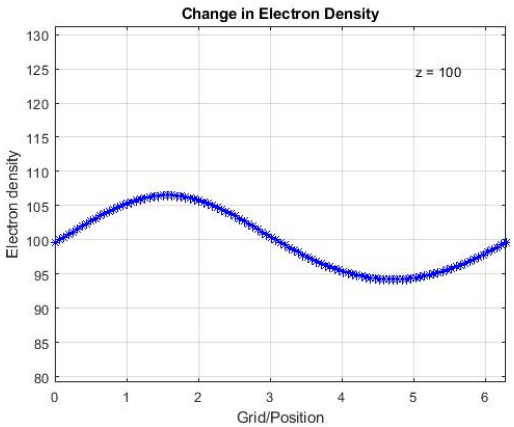
(c) $z = 40$



(d) $z = 60$



(e) $z = 80$



(f) $z = 100$

Figure 5: Electron Density according to different Number of Time Steps