*Available online at* www.pelagiaresearchlibrary.com

**Pelagia Research Library**

**Advances in Applied Science Research, 2015, 6(6):103-113**

**ISSN: 0976-8610**
**CODEN (USA): AASRFC**

# Simulation of heating in space plasma

**[1*]Umar S. A., [2]Ibrahim G. G., [3]Ibrahim I. E., [4]Najib M. U. and [5]Liman M. S.**

*[1]Bauchi State University, Gadau, Nigeria*
*[2]Federal College of Education (Technical), Potiskum, Nigeria*
*[3]Nasarawa State University, Keffi, Nigeria*
*[4]Nigerian Nuclear Regulatory Authority, North West Zonal Office, Kastina, Nigeria*
*[5]Federal University, Lafia, Nigeria*

**ABSTRACT**

*Computer simulation was employed here to study the process of heating in space plasmas. In the simulation, the one-dimensional particle-in-cell simulation method was used and the Maxwell's Equations solved using the Finite Difference Time Domain (FDTD) method. The modified one-dimensional Kyoto University electromagnetic particle code with relativistic effect consideration (MATLAB Version) was used. The simulation results obtained is a plot of energy, which shows heating at the beginning of the simulation with the system reaching its thermodynamic equilibrium later in the simulation. Other results obtained where the velocity vector space plot showing a drift maxwellian nature, indicating collisions between particles and two other Spatial Electric and Magnetic fields plots. The results in the simulation showed that the heating in the space plasmas are caused mainly by binary collisions between the particles.*

## INTRODUCTION

Plasma mostly resembles ionized gas. It is one of the four states of matter after gas (That is Solid-Liquid-Gas-Plasma). Most of the matter in the universe is in plasma state (99.996% of the universe). However, we live in a small region of the universe where matter is predominantly solid, liquid and gas. Interest in the properties of plasma is quite recent and has mainly been stimulated by its importance in space physics and controlled nuclear fusion (Liman, 2008).

It is generated by applying a sufficiently high electric field to a gas, which partially breaks it down, turning some atoms or molecules into positive ions and generating free electrons (When they loses electrons from their outermost shells). These free charges make the plasma electrically conductive, internally attractive and strongly repulsive to electromagnetic fields and surfaces (Fitzpatrick, 2007).

### 2.1 Plasma Simulations

For plasma with a large number of degrees of freedom, particle simulation using high-speed computers can offer insights and information that supplement those gained by traditional experimental and theoretical approaches. The technique follows the motion of a large assembly of charged particles in their self-consistent electric and magnetic fields. With proper diagnostics, these numerical experiments reveal such details as distribution functions, linear and nonlinear behaviour, stochastic and transport phenomena, and approach to steady state. It can also be used in the interpretation of experiments (Dawson, 1983).

Plasma consists of positive charged ions and negative charged electrons. Since $M_i >> m_e$ ($m_i$=ionic mass, $m_e$= electronic mass) , there are many intrinsic time scales in a plasma system even in a uniform background environment. For time scale less than the intrinsic time scales, it is hard for the plasma to reach a thermodynamic equilibrium state. Present plasma simulation codes can be classified based on their phase space resolutions. Note that the so-called particle-code simulation is indeed a multiple-fluid simulation. A simulation particle in the particle-code simulation is indeed a fluid element in the phase space (**x**,**v** ) (Lyu,2007).

Currently, optimum numerical schemes which are run on modern fast CPU, large memory parallelized computer systems led to a revival of kinetic codes which directly solve the Vlasov-equation describing collective collisionless plasmas. For many years due to restricted computer resources, collisionless plasmas have been simulated mainly by re-graining the flow of the distribution functions of phase space via introducing macro-particles in a particle-in-cell (PIC) approach ( Buchner, 2007).

The Vlasov equation describes the evolution of the distribution function of particles in phase space (**x**,**v**), where the particles interact with long-range forces, but where shortrange "collisional" forces are neglected. A space plasma consists of low-mass electrically charged particles, and therefore the most important long-range forces acting in the plasma are the Lorentz forces created by electromagnetic fields. What makes the numerical solution of the Vlasov equation a challenging task is that the fully three-dimensional problem leads to a partial differential equation in the six-dimensional phase space, plus time, making it hard even to store a discretised solution in a computer's memory ( Elliasson, 2002).

The fluid (continuum) approach is commonly used for simulation of plasma phenomena in electrical discharges at moderate to high pressures (>10's mTorr). The description comprises governing equations for charged and neutral species transport and energy equations for electrons and the heavy species, coupled to equations for the electromagnetic fields. The coupling of energy from the electrostatic field to the plasma species is modeled by the Joule heating term which appears in the electron and heavy species (ion) energy equations. Proper numerical discretization of this term is necessary for accurate description of discharge energetics; however, discretization of this term poses a special problem in the case of unstructured meshes owing to the arbitrary orientation of the faces enclosing each cell(Deconinck *et al*, 2009).

A common approach to modeling kinetic problems in plasma physics is to represent the plasma as a set of Lagrangian macro-particles which interact through long-range forces. In the well-known particle-in-cell (PIC) method, the particle charges are interpolated to a mesh and the fields are obtained using a fast Poisson solver. The advantage of this approach is that the electrostatic forces can be evaluated in time , where as the number of macro-particles, but the scheme has difficulty resolving steep gradients and handling nonconforming domains unless a sufficiently fine mesh is used(Christlieb, 2006).

**How Particle-In-Cell Works**

For many types of problems, methods of PIC are relatively intuitive and straight- forward to implement. This probably accounts for much of its success, particularly for plasma simulation, for which the method typically includes the following procedures:

i. Integration of the equations of motion.
ii. Interpolation of charge and current source terms in the field mesh.
iii. Computation of the fields on mesh points.
iv. Interpolation of the fields from the mesh to the particle locations.

Models which include interaction of particles only through the average fields are called PM (particle mesh). Those which include direct binary interactions are PP (particle-particle). In this method, each of the N particles potentially interacts with every other particle, and each force pair is equal and opposite giving N(N-1)/2 unique forces (Nocito, 2010). Models with both types of interaction are called PP-PM or $P^3$M. Since the early days, it has been recognized the PIC method is susceptible to error from so-called discrete particle noise (Okuda, 1972).

**Super-Particle**

The real system studied is often extremely large in terms of particles they contain. In order to make simulations efficient or at all possible, so-called super-particles are used. A super-particle (or macro particles) is a computational particle that represents many real particles; it may be millions of electrons or ions in the case of plasma simulation, or for instance, a vortex element in a fluid simulation.

**The Particle Mover**
Even with super-particles, the number of simulated particles is usually very large (>10s), and often the particle mover is the most time consuming part of PIC, since it has to be done for each particle separately. Thus, the pusher is required to be of high accuracy and speed and much effort is spent on optimizing the different schemes. The schemes used for the particle mover can be split into two categories, implicit and explicit solvers. While implicit solvers calculate the particle velocity from the already updated fields, explicit solver use only the old force from the previous time step, and are therefore simpler and faster, but require a small time step. In this work, the implicit Boris scheme is used.

**The Field Solver**
The most commonly used methods for solving Maxwell's equations (are more generally, partial differential equation (PDE)) belongs to one of the following three categories:

i. Finite Difference Methods (FDM) or Finite Difference Time Domain (FDTD)
ii. Finite Element Methods (FEM)
iii. Spectral Methods

**Particle and Field Weighting**
The name'' particle-in-cell'' originates in the way that plasma macro-quantities (number density, current density,(etc) are assigned to simulation particles (i-e particle weighting. Particles can be situated anywhere on the continuous domain, but macro-quantities are calculated only on the mesh points, just as the fields are. To obtain the macro-quantities, a given "shape'' determined by the shape function S($x$-X) is defined. Where, $x$ is the coordinate of the particle and X the observation point.

The fields obtained from the field solver are determined only on the grid points and can't be used directly in the particles mover to calculate the force acting on particles, but have to be interpolated via the field weighting:

$$E(x) = \sum_i E_i\, S(x_i - X) \tag{3.24}$$

Where the subscript i labels the grid point. To ensure that the forces acting on particles are self-consistently obtained, the way of calculating macro-quantities from particle positions on the grid points and interpolating fields from grid points to particle positions has to be consisted too, since they both appear in Maxwell's equations. Above all, the field interpolation scheme should conserve momentum (Tskhakaya, 2008).

## MATERIALS AND METHODS

**Methodology**
One-dimension electromagnetic particle code is employed in this work. The whole plasma system is considered to have one dimensional distribution of particle with magnetic field influence consideration. This is a particle in cell simulation of the space plasma which is commonly used in the study of plasma behavior. The heating process was studied from the diagnostic plots in the output.

**Position computation:** The particle position X is computed and advanced using velocity $V_x$. In one time step Δ t, the particle position are advance twice each by a half time step Δt/2 as

$$X^{t+\Delta t} = X^{t} + V_x^{t+\Delta t/2} \tag{3.1}$$

$$X^{t+\Delta t} = X^{t+\Delta t/2} + V_x^{t+\Delta t/2}\frac{\Delta t}{2} \tag{3.2}$$

The above iteration is obtained from the numerical solution of the equation.

$$\frac{dx}{dt} = V_x \tag{3.3}$$

**Velocity computation:** For high enery particles with velocities close to the speed of light, we have to solve the relativistic equation of motion

$$\frac{d}{dt}(mv) = q\,(E + v \times B) \tag{3.4}$$

Where m= $\gamma m_0$, $m_0$ is the rest mass, and $\gamma$ is the Lorentz factor given by.

$$\gamma = \frac{1}{\sqrt{1+\frac{v^2}{c^2}}} \tag{3.5}$$

We define u = γ v (3.6)

$$u = \frac{c}{\sqrt{c^2 + |v|^2}}\,v \tag{3.7}$$

Solving for v , we have

$$v = \frac{c}{\sqrt{c^2 + |u|^2}}\,u \tag{3.8}$$

Equation (3.4) is rewritten as

$$\frac{du}{dt} = \frac{q}{m0}\left(E + \frac{c}{\sqrt{c^2 + |u|^2}}u \times B\right) \tag{3.9}$$

Defining a modified magnetic field as

$$B_u = \frac{c}{\sqrt{c^2 + |u|^2}}B \tag{3.10}$$

We obtain $\frac{du}{dt} = \frac{q}{m0}(E + u \times B_u)$ (3.11)

The difference form of equation (3.11) is

$$\frac{U^{t+\Delta t/2} - U^{t-\Delta t/2}}{\Delta t} = \frac{q}{mo}\left(E^t + \frac{U^{t+\Delta t/2} + U^{t-\Delta t/2}}{2} \times B^t_u\right) \tag{3.12}$$

We apply the Bunemann-Boris method as:

Step 1:

$$U^{t-\Delta t/2} = \frac{c}{\sqrt{c^2 - \left|v^{t-\Delta t/2}\right|^2}} V^{t-\Delta t/2} \tag{3.13}$$

Step 2:

$$U_1^t = U^{t-\Delta t/2} + {}^{q}/_{m_0}\,\frac{\Delta t}{2}E^t \tag{3.14}$$

Step 3:

$$B_u^t = \frac{C}{\sqrt{c^2 + |U_1^t|^2}} B^t \tag{3.15}$$

Step 4:

$$U^{t'} = U_1^t + {}^{q}/_{m_0}\,\frac{\Delta t}{2} U_1^t \times B_u^t \tag{3.16}$$

Step 5:

$$U_2^t = U_1^t + \frac{2}{1+(B_u^t \frac{q}{m_0}\Delta t/2)2} U^{t'} \times B_u^t \frac{q}{m_0}\frac{\Delta t}{2} \tag{3.17}$$

Step 6:

$$U^{t+\frac{\Delta t}{2}} = U_2^t + \frac{q}{m_0}\frac{\Delta t}{2}E^t \tag{3.18}$$

Step 7:

$$U^{t+\Delta t} = \frac{c}{\sqrt{c^2 - \left|v^{t+\Delta t/2}\right|^2}} U^{t+\Delta t} \tag{3.19}$$

$B_u^t$ Is computed from $U_1^t$ and $B^t$.

In the relativistic code, we initialize particle velocities so that the following distribution function is realized in the momentum space ($U_x$, $U_Y$, $U_z$)

$$F(U_{\|}, U_{\perp}) \propto \exp\left(-\frac{(U_{\|}-V_{d\|})^2}{2V_{t\|}^2}-\frac{(U_{\perp}-V_{d\perp})^2}{2V_{t\perp}^2}\right) \quad (3.20)$$

Each particle is assigned with a momentum U = ($U_x$, $U_Y$, $U_Z$), which is converted to velocity

V = ($V_x$, $V_Y$, $V_Z$) by

$$V = U/\gamma = \frac{c}{\sqrt{c^2+U_x{}^2+U_y^2+U_z^2}}U \quad (3.21)$$

As for the diagnostics, the kinetic energy of each particle is calculated by

$$T_E = mc^2-m_0c^2 \quad (3.22)$$

$$T_E = (\gamma-1)\, m_oc^2 \quad (3.23)$$

The sum of the kinetic energy is taken for all particles in the simulation system and magnetic field energy densities in the system. We divide the sum by the length of the system to obtain the averaged kinetic energy density. The sums of the electric and magnetic field energies are taken for all grid points forming the simulation system. Dividing them by the number of grid points, we obtain the averaged electric and magnetic field energy densities in the system.
**Charge density computation:** The charge density $\rho_i$ is computed at a grid point at x = $x_i$ by

$$\rho_i = \frac{1}{\Delta x}\sum_j^{Np} q_j\, W(x_j - X_i) \quad (3.25)$$

Where $W_x$ is a particle shape function given by

$$\begin{cases} W(x) = 1 - \frac{|x|}{\Delta x}; & |x| \leq \Delta x \\ = 0 \quad ; & |x| > \Delta x \end{cases} \quad (3.26)$$

**Current density computation:** The current density $J_x$ is calculated based on charge conservation method satisfying the continuity equations of charge

$$J_{x,i+1/2}^{t+\Delta t/2} - J_{x,i-1/2}^{t+\Delta t/2} = -\frac{\Delta x}{\Delta t}(\rho_i^{t+\Delta t}+\rho_i^t) \quad (3.27)$$

The current densities $J_y$ and $J_z$ are computed by

$$J_{i+1/2}^{t+\Delta t/2} = \frac{1}{\Delta x}\sum_j^{Np} q_j V\, W(x_j - X_{i+1/2}) \quad (3.28)$$

The $J_y$ computed at the half-integer grids are relocated to full-integer grids by

$$J_{y,i} = \frac{J_{y,i-1/2}+J_{y,i+1/2}}{2} \quad (3.29)$$

With the components of J, we trace the time evolution of electromagnetic fields E and B by solving Maxwell's equations with the finite-difference time-domain (FDTD) method.

**Fields Computations:** Electromagnetic processes in space plasma are governed by Maxwell's equation

$$\nabla \times B = \mu_0 J + \frac{1}{c2}\frac{\partial E}{\partial t} \quad (3.30a)$$

$$\nabla \times E = -\frac{\partial B}{\partial t} \quad (3.30b)$$

$\nabla.E = \rho / \varepsilon_0$ (3.30c)

$\nabla. B = O$ (3.30d)

Where, J, ρ, c, $\varepsilon_0$ and $\mu_0$ are the current density, light speed, electric permittivity and magnetic permeability respectively.

In this simulation, we define $\varepsilon_0$ and $\mu_0$ arbitrarily so as to satisfy the relation

$$\varepsilon_0\mu_0 = \frac{1}{c^2} \quad (3.31)$$

And hence, we adopt

$$\varepsilon_0 = 1, \text{ and } \mu_0 = \frac{1}{c^2} \quad (3.32)$$

**E-Field:** The initial electric field is calculated from

$$\nabla.E = \rho/\varepsilon_0 \quad (3.33)$$

In difference form, one-dimensional equation is written as

$$E_{x,i+1/2} - E_{x,i-1/2} = \frac{\rho_i}{\varepsilon_0}\Delta x \quad (3.34)$$

The electric field is advanced in time by integrating one of the Maxwell's equations

$$\frac{\partial E}{\partial t} = c^2 \nabla \times B - J \quad (3.35)$$

The above equation, we can rewrite for one-dimensional system as

$$\frac{\partial E_x}{\partial t} = - J_x \quad (3.36)$$

$$\frac{\partial E_y}{\partial t} = - c^2\frac{\partial B_z}{\partial x} - J_y \quad (3.37)$$

$$\frac{\partial E_z}{\partial t} = c^2 \frac{\partial E_y}{\partial x} - J_z \quad (3.38)$$

These equations are integrated for one time step of $\Delta t$.

**B-Field:** The magnetic field B is advanced in time by integrating one of the Maxwell's equations

$$\frac{\partial B}{\partial t} = - \nabla \times E \quad (3.39)$$

Rewriting the one-dimensional form of the above equation as

$$\frac{\partial B_y}{\partial t} = \frac{\partial E_z}{\partial x} \quad (3.40)$$

$$\frac{\partial B_z}{\partial t} = -\frac{\partial E_y}{\partial x} \quad (3.41)$$

Since from Maxwell's equations, $\nabla. B = 0$, (3.42)

With one-dimensional form as; $\frac{\partial B}{\partial x} = 0$, $B_x = 0$ (3.43)

**Input Parameters**

The following input parameters are specified in the program

- ΔX: Grid Spacing
- ΔT: Time step
- CV: Speed of light
- WC1: cyclotron frequency of species 1

- ANGLE: angle between the static $B_O$ is taken in the X-Y plane.
- NX: Number of grid points
- N Time: number of time steps in a simulation run.
- NS: number of particle species.
- QM (i): charge -to- mass ratio $q_i/m_i$ of species.
- WP(i): plasma frequency of species is defined by

$$\omega_{pi} = \sqrt{\frac{n_i \, q^2{}_i}{m_i}} \tag{3.44}$$

Where $n_i$ and $m_i$, are number density charge and mass of species i, respectively.

- VPE (i); perpendicular thermal velocity of species i.
- VPA (i); parallel thermal velocity of species i.
- VD (i): drift velocity of species i.
- PCH (i) pitch angle $\phi$ (degrees) species i defining parallel and perpendicular drift velocities $V_{d//}$ and $V_{d\perp}$ (i-e parallel and perpendicular components).
- NP (i): number of super particles for species i in the simulation system.
- AJAMP: the amplitude of an external current $J_{z,\ ext.}$ placed at the centre of the simulation system.
- WJ: the frequency of the external current $J_{z\ ext}$.
- IEX: control parameter for electrostatic option.
- NPLOT: number of diagnostics to be made through the simulation run.

**Renormalization of Input Parameters**

In order to achieve computational efficiency, it is necessary to reduce the number of operation involved in the difference equations of the fields and particles. Since the operations of multiplication and division are performed frequently with the grid spacing ΔX and the half-time step Δt/2, we renormalize distance and time by ΔX and Δt/2 respectively.

**Simulation Set Up**

The system is set up as unmagnetized plasma with a very small thermal velocity, where the Debye length $\lambda_D$ is made much smaller than the grid spacing and Courant condition, C Δ t < Δx is satisfied.

The input parameters are chosen as follows:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Δx=1 | $\Delta t = 0.04$ | NX =256 | NTIME =512 | CV = 20 | WC =0 | ANGLE =0 | NS =2 | |
| NP = 4096 | WP = 4 | QM = -1 | VPA =1 | VPE =1 | PCH =0, 60 | IEX = 1 | AJAMP =0 | |
| WJ =0 | NPLOT=256 | ICOLOR =1 | IPARAM =1 | $V_{MAX}$ =20 | $E_{MAX}$ =5 | $B_{MAX}$ =0.5 | | |

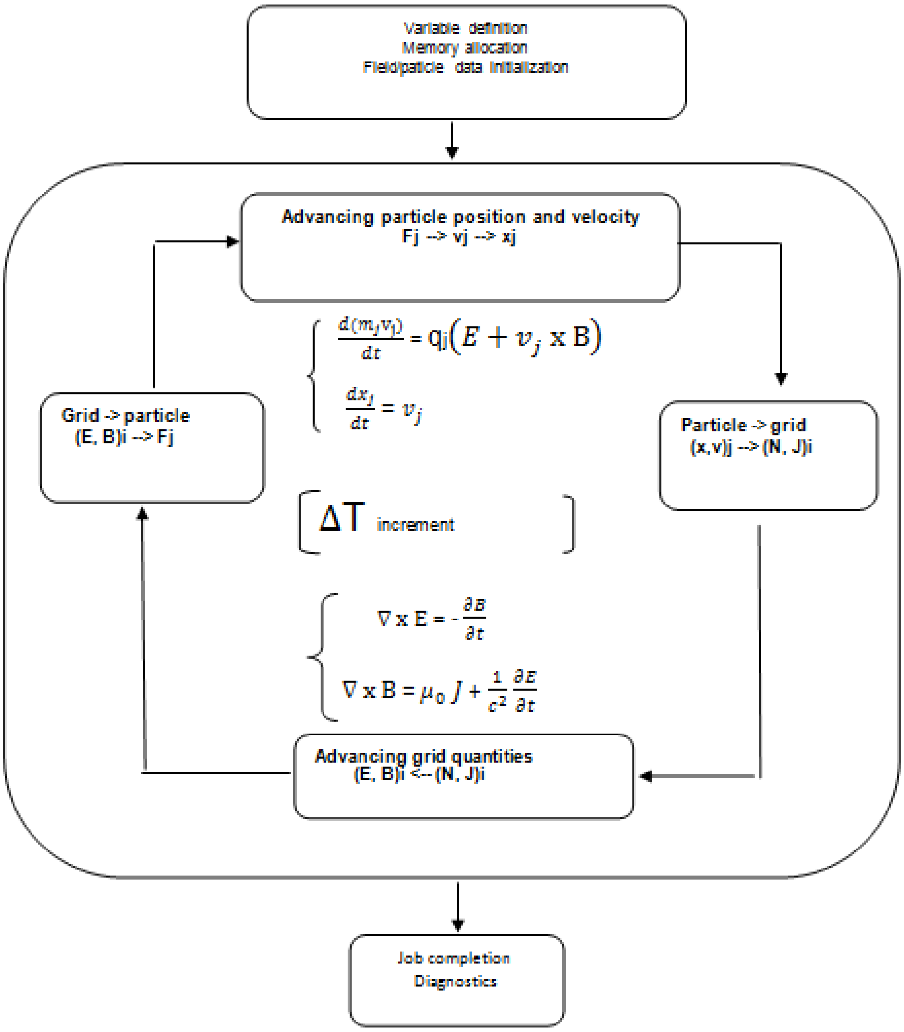
**Grid Assignment**

We define full-integer grids at i Δx ( i =1,2,3,4………..$N_x$) and half integer grids at (i+ 1 ) Δx. The $E_y$, $B_y$, $J_y$ and p are defined at full-integer grids, and $E_x$, $E_z$, $B_z$, $J_x$, $J_z$ at the half -integer grids. This assignment of E and B realizes centered difference form for the spatial derivatives in Maxwell's equations. The components $J_x$, $J_y$, $J_z$ of J must be assigned to the same grid as $E_x$, $E_y$ and $E_z$.

**Time Step Chart**

The equation of the field and particles are advanced in time based on the leap-frog sequence method. We define a full – integer time n Δt and a half- integer time (n+1/2 ) Δt with a time step Δt. Basically, the electric field E at the full-integer and the magnetic field B at the half –integer time are integrated in time by the leap-frog method. However, the magnetic field B is advanced twice by a half- time step Δt/2 to obtain intermediate values for the particles pushing field at the full-integer time. The particle positions x at the full – integer time and velocity v at the half -integer time are also advance by the leap-frog method.

**Flow Chart**




**RESULTS AND DISCUSSION**

The result for the "computer simulation of space plasma heating" using one-dimensional electromagnetic particle code is presented. The simulation runs for a simulation time of 20.480. It consists of four plots; Energy, Velocity vector space, Spatial Magnetic and Electric fields plots as in figures 4.1, 4.2, 4.3 and 4.4 respectively.

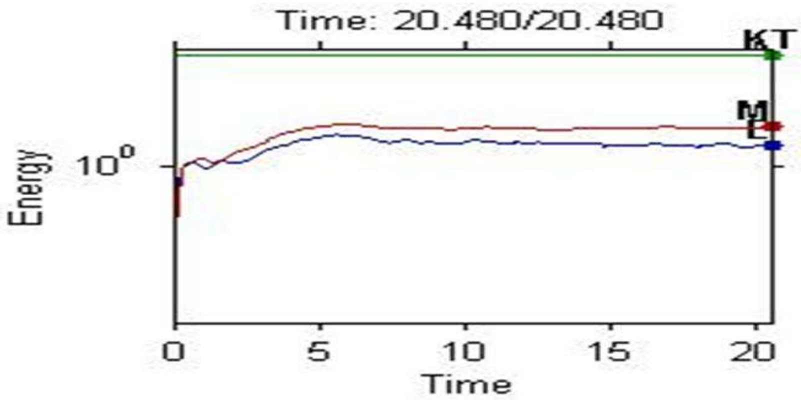



**Figure 4.1: Energy-Time Plot**

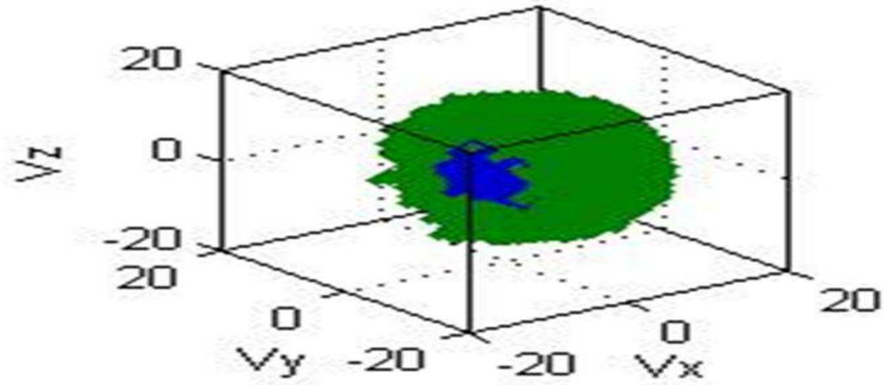



**Figure 4.2: Velocity Vector Space Plot**

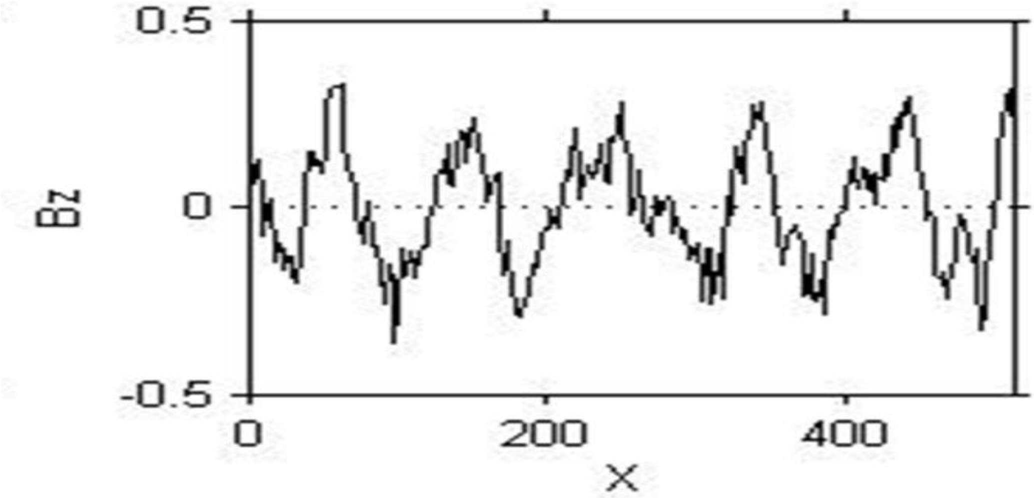



**Figure 4.3: Spatial Magnetic Field Plot**

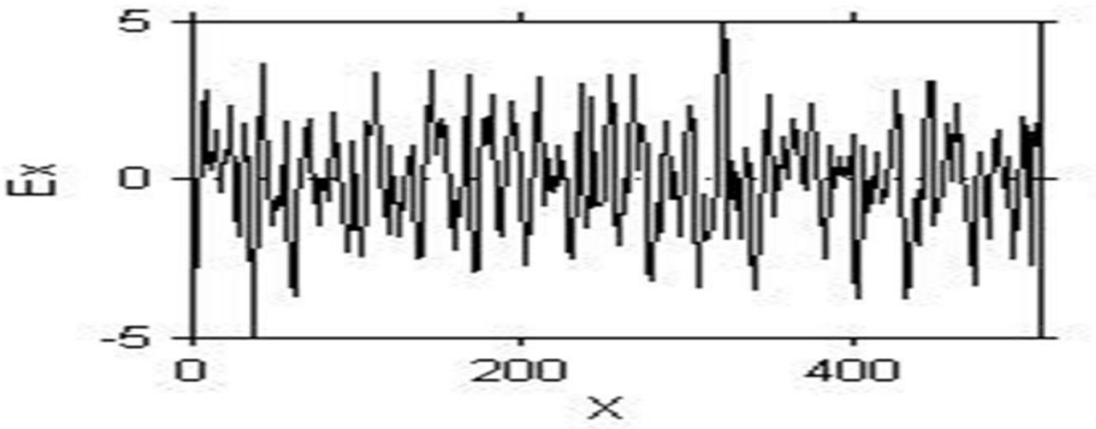



**Figure 4.4: Spatial Electric Field Plot**

## CONCLUSION

In this computer experiment to study the heating in space plasmas, Maxwell's equations were solved numerically with the equations of fields and particles being advanced in time based on the leap-frog sequence method. The whole simulation was carried out using the Modified One-Dimensional Kyoto University electromagnetic Particle code (MATLAB Version), with relativistic effect consideration. The system was initialised as Low-Density plasma with some compromised features (Low temperature, Small Debye length, low thermal velocity).

The plasma was seen to heat up in the beginning of the simulation in figure 4.1 and the spherical velocity vector plot in figure 4.2 indicates collision behaviour of the plasma particles. The spatial electric and magnetic fields variations are observed to be sinusoidal as figures 4.3 and 4.4 show.

Thus, the heating in the space plasmas is caused by the collisions between the plasma particles and hence the spatial variations in the electric and magnetic fields.

The One-Dimensional PIC method was chosen for its simplicity and convenience, not because it is physical. A real system is a three-Dimensional system. Though it is more complex to simulate a Three-Dimensional system considering the time and memory requirements, the One-Dimensional PIC Plasma heating simulated in this work gives the foundation for the simulation in Three-Dimensional PIC and other simulation methods like MHD and Hybrid simulation methods

## REFFERENCES

[1] Abdalla, A., Dutton M. & Dutton, D.A. (**1985**) "*Space Plasma Simulation*". D. Reidel Pub. Co. 1985.
[2] Baumjohann, W. & Treumann, R. A. (**1996**). "*Basic space plasma physics*". Max-Plank-Institut fur extraterrestrische physik, Garching and Institut fur Geophysik, Ludwig-Maximilians-Universitat Munchen.
[3] Brackbil, l J. U. & Lapenta G. (**2008**). *Communications in computational physics* Vol. **4**, No. 3, pp. 433-4.
[4] Buchner, J. (**2007**). "*Vlasov-code simulation*". Max-Planck-Institut fur Sonnensystemforschung, 37191 Katlenburg-Lindau, Germany.
[5] Christlieb, A. J. (**2006**). *IEEE Transactions on Plasma Science*, VOL. 34, NO. 2.
[6] Dawson, J. M. (**1983**). "*Particle Simulation of plasma*" Department of Physics, University Of California, Los Angeles, California 90024.
[7] Deconinck, T., Mahadevan, S. & Raja, L. L. (**2009**). *Journal of computational physics* 228(2009) 4435-4443.
[8] Eliasson, B. (**2002**). "*Numerical Vlasov-Maxwell Modelling of Space Plasma*". Alta Universitatis Upsaliensis, Uppsala.
[9] Filbet, F., Negulescu C. & Yang C. (**2011**). "*Numerical study of a Nonlinear Heat Equation for Plasma Physics*"
[10] Fitzpatrick, R. (**2007**). "*Plasma Physics Introduction*". University of Texas at Austin.
[11] Greengard, L. & Rokhlin V. (**1987**) *Journal of computational physics* 73. 325-348(1987).
[12] Hockney, R.W. & Eastwood, J. W. (**1981**). "*Computer Simulation Using Particles*". McGraw-Hill, New York.

[13] Lapenta, G. (**2009**) *"A brief description of the PIC method"* Centrum Voor Plasma Astofysica, Kotholieke Universitiet, Leuven.
[14] Liman, M. S. (**2008**) *"Proposal for the establishment of plasma science and technology Kaduna state university, Kaduna."* Private submission.
[15] Lyu, L. H. (**2007**) "*Numerical Simulation of Space Plasmas"* (I) [AP-4036] Lecture 5.
[16] Matsumoto, H. (**1993**) "*Chapter 1: Computer Experiments of Space Plasma*". Kyoto University Japan.
[17] Mitharwal R. (**2001**). "*A Two-dimensional numerical simulation of plasma wake around a CUBESAT"*. Master of Science thesis, Utah State University, Logan, Utah.
[18] Nocito, N. W. (**2010**) *Journal of computational science education. Volume , issue 1.*
[19] Okuda, H. (**1972**). *Journal of computational physics* 10(3):475.
[20] Schunk, R. (**2009**). **"***Ionospheres: Physics, Plasma Physics, and Chemistry*". Cambridge University Press, Pp. 59-66.
[21] Spitkvosky, A. (**2008**). "*Kinetic Plasma Simulation*". Princeton (Retrieved). Pp.20-46.
[22] Tskhakaya, D. (**2008**). *Computational physics* 25(3):205-219.