NUMERICAL SIMULATION OF TWO-DIMENSIONAL COLLISIONLESS PLASMA FLOWS UNDER THE EFFECT OF ELECTROSTATIC FORCES VIA PARTICLE IN CELL METHOD

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ÖZGÜR TÜMÜKLÜ

IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
AEROSPACE ENGINEERING

JUNE 2013

Approval of the thesis:

NUMERICAL SIMULATION OF TWO-DIMENSIONAL COLLISIONLESS PLASMA FLOWS UNDER THE EFFECT OF ELECTROSTATIC FORCES VIA PARTICLE IN CELL METHOD

submitted by **ÖZGÜR TÜMÜKLÜ** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Ozan Tekinalp
Head of Department, **Aerospace Engineering** _____

Prof. Dr. M. Cevdet Çelenligil
Supervisor, **Aerospace Engineering Dept., METU** _____

Asst. Prof. Dr. Nevsan Şengil
Co-Advisor, **Astronautical Engineering Dept., THK University** _____

Examining Committee Members

Prof. Dr. İ. Sinan Akmandor
Aerospace Engineering Department, METU _____

Prof. Dr. M. Cevdet Çelenligil
Aerospace Engineering Department, METU _____

Assoc. Prof. Dr. Oğuz Uzol
Aerospace Engineering Department, METU _____

Assoc. Prof. Dr. Sinan Eyi
Aerospace Engineering Department, METU _____

Assoc. Prof. Dr. İsmail Rafatov
Department of Physics, METU _____

Date: 14.06.2013

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Özgür Tümüklü

Signature :

ABSTRACT

NUMERICAL SIMULATION OF TWO-DIMENSIONAL COLLISIONLESS PLASMA FLOWS UNDER THE EFFECT OF ELECTROSTATIC FORCES VIA PARTICLE IN CELL METHOD

Tümüklü, Özgür
M.S., Department of Aerospace Engineering
Supervisor: Prof. Dr. M. Cevdet Çelenligil
Co-supervisor: Asst. Prof. Dr. Nevsan Şengil

June 2013, 64 pages

Taking its name from its ability to generate thrust via electricity, the concept of electric propulsion has important space mission applications like station keeping for satellites and deep space probe. However, contamination of plumes in electric propulsion systems with ion beam could hinder communication, and effective neutralization is essential to clear away this predicament. Today, computer simulation is regarded as a powerful tool to investigate plasma behavior in the plumes of electric propulsion devices.

The aim of this thesis is to write a generic simulation code named as SIMPFORT which studies two-dimensional, collisonless plasma flows under the effect of electrostatic forces using the Particle in Cell (PIC) method. The conservation of mass, momentum and energy are checked. In this thesis, the flows with negligible forces from magnetic fields are investigated and electrostatic Poisson's equation is solved in both rectangular and non-rectangular domains. Successive Over Relaxation (SOR) method is implemented in the solution of the Poisson's equation.

SIMPFORT can handle plasma problems using either an accurate full kinetic particle model or an approximate hybrid model which combines kinetic and continuum (fluid) models. In the full kinetic model, both ions and electrons are simulated as particles, but in the hybrid model only ions are simulated as particles while electrons are modeled as fluid using the Boltzmann relation. In the full kinetic model the Poisson's equation is linear, on the other hand, in the hybrid model it becomes nonlinear due to the effect of the Boltzmann relation.

In this thesis, two generic test problems are solved and the results are compared with a demo version of a commercial full kinetic code named VORPAL. The present full kinetic model results compare favorably with those of VORPAL for both test problems. On the other hand, the hybrid model results show some deviations in one of the test problems.

Keywords: plasma simulation, electric propulsion, Particle in Cell (PIC) method, Boltzmann relation

ÖZ

İKİ BOYUTLU ÇARPIŞMASIZ ELEKTROSTATİK KUVVETLER ETKİSİNDEKİ PLAZMA AKIMLARININ HÜCRE YÖNTEMİ KULLANILARAK SAYISAL SİMULASYONU

Tümüklü, Özgür
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. M. Cevdet Çelenligil
Yardımcı Tez Yöneticisi: Yrd. Doç. Dr. Nevsan Şengil

Haziran 2013, 64 sayfa

İsmini elektrikten itki üretebilmesinden alan, electrik itme kavramını uzayda pozisyon koruma ve derin uzay araşatımaları gibi önemli görevleri bulunur. Uzayı kirleten elektrik itme sistminin iyon ışınını haberleşmeye engel olabilir. Etkili nötralizasyon bu problemi ortadan kaldırmak için gereklidir. Günümüzde, bilgisayar simülasyonu elektrikli itki cihazların çıkış ağzında plazma davranışı araştırmak için güçlü bir araç olarak kabul edilir.

Bu tezin amacı hücredeki parçaçık yöntemini kullanrak SIMPFORT olarak adlandırılan özgün iki boyutlu, çarpışmasız plazma simülasyon kodu yazmaktır. Kütle, momentum ve enerjinin korunumu yasaları kontol edildi. Bu tezde, manyetik alandan kaynaklanan ihmal edilebilir kuvvetli akışlar incelenir ve elektrostatik Poisson denklemi dikdörtgen ve dikdörtgen olmayan her iki alanda çözülür. Başarılı fazla dinlenme (SOR) methodu Poisson denklemini çözmek için uygulanmıştır.

SIMPFORT, doğru bir tam kinetik parçacık modeli veya kinetik ve sürekli (sıvı) modellerini birleştiren yaklaşık bir melez modeli kullanılarak plazma problemlerini ele alır. Tam kinetik modelde, iyonlar ve elektronlardan her ikisi de parçacıklar gibi simüle edilir, fakat melez modelde sadece iyonlar parçacıklar olarak simüle edilirken elektronlar ise Boltzmann ilişkisi kullanılarak akışkan olarak modellenir. Tam kinetik modelde, Poisson denklemi doğrusaldır. Diğer taraftan, melez modelde, Boltzmann ilişkisinin etkisiyle doğrusal olmayan hale gelir.

Bu tezde, iki genel test problemleri çözülür ve sonuçları VORPAL adlı ticari tam kinetik kodun demo sürümü ile karşılaştırılır. Mevcut tam kinetik model sonuçları, her iki test problemi için VORPAL sonuçları ile uygun olarak karşılaştırılır. Öte yandan, melez modeli sonuçları test durumlarından birinde bazı sapmalar göstermektedir.

Anahtar Kelimeler: plazma simülasyonu, elektrikli itki sistemleri, hücredeki parçacık yöntemi, Boltzmann ilişkisi

To my family

ACKNOWLEDGMENTS

I would like to express my gratitude to my graduate advisor, Prof. M. Cevdet Çelenligil, for introducing me with this topic and for his valuable guidance and endless help, encouragement throughout my undergraduate and graduate studies.

I am also grateful to my Co-Advisor, Dr. Nevsan Şengil, for his endless support.

I want to appreciate my brother, Oğuz Tümüklü, for his advices and support in all my life. Thanks to him for giving me opportunity to continue my education.

I am always grateful to my mother for her care in my whole life and always remember the memory of my father with God's mercy.

Special thanks go to my colleague, Emre Yılmaz, for helping and supporting me during my graduate years and also during the preparation of this thesis.

I always remember my friends, Burak Akın and Özgür Harputlu for wonderful years spent together.

I must also thank my friends Özgür Yalcin, Kenan Cengiz and Sinem Özçakmak for the enjoyable time spent together.

My sincere thanks are addressed to my friend, Ayşe Usta, for spending her time to cheer me up.

To sum up, I want to express my appreciation to all others not mentioned here who gave me help.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

1.1 Plasma

Plasma is designated as the fourth state of matter. It is a common belief among scientists that it constitutes 99 % of the universe. If the gas molecules are exposed to high temperature or strong electromagnetic fields, they begin to ionize and are named as plasma if relevant criteria are satisfied.

Plasma contains charged particles such as singly or doubly charged ions, electrons and neutral particles. The presence of the charged particles makes ionized gas conductive and causing the gas to be affected by the electromagnetic fields. This makes it difficult to study these complicated problems.

1.1.1 Basic Plasma Characteristic Parameters

One of the main characteristic parameters in plasma physics is the debye length (λ_d) which is defined as the required minimum distance to see collective behavior. In other words, it is the distance where quasi-neutrality behavior of plasma can be seen. The debye length is mathematically defined as:

$$\lambda_d = \sqrt{\frac{\epsilon_0 K T_e}{n e^2}} \tag{1. 1}$$

where T_e and K are temperature of the electrons and Boltzmann constant, respectively. n is number density of electrons or ions, e is elementary charge, and ϵ_0 is permittivity.

The second parameter is the plasma frequency (w_p) that is the frequency of electrons around their equilibrium states, and is given by:

$$w_p = \frac{v_{th}}{\lambda_d} \tag{1. 2}$$

where $v_{th} = \sqrt{\frac{KT_e}{m}}$ is thermal speed of electrons and m is the mass of an electron.

The third parameter is the number of the particles in the debye sphere (N_D) and is given by the formula:

$$N_D = n\frac{4}{3}\pi{\lambda_d}^3. \tag{1. 3}$$

Numbers of particles in the debye sphere determines whether the system is strongly or weakly coupled. A strongly coupled system contains small number of charged particles in debye sphere and the field properties can change abruptly from one point to another. On the other hand, weakly coupled systems contain large number of particles in debye spheres.

Ionized gas can be defined as plasma if the following conditions are satisfied:

$$\begin{aligned} \lambda_d &\ll L \\ N_d &\gg 1 \\ w_p\tau &> 1 \end{aligned} \tag{1. 4}$$

where L is the characteristic length and τ is mean time required to collide two neutral particles [8].

1.1.2 Classification of Plasmas

The degree of ionization determines the type of the plasma. For fully ionized plasma, all gas molecules are ionized. On the other hand, for partially ionized gas, only small percentage is ionized.

The laboratory plasmas can also be classified according to their temperatures which are fusion (or high temperature) plasmas and gas discharge plasmas. Another subdivision can be made according to equilibrium temperature of plasmas: local temperature equilibrium (LTE) and non LTE plasmas. If the temperature of the species in some region is the same, then, it is called LTE. However, in non LTE plasmas, the temperatures of different species are different. For example, electrons have higher temperature compared to heavy particles (ions and neutrals) [6].

1.2 Electric Propulsion

Electric propulsion (widely used in space applications, especially in stationkeeping) offers high specific impulses. Exhaust velocities of particles are very high compared to the conventional propulsion methods. High exhaust velocities from exit result in high specific impulse which means that it significantly reduces the launch weight of vehicle. Unlike chemical thrusters which use internal energy of fuel to get energy, electric propulsion uses external electromagnetic forces to obtain thrust. There are many different types of electric thrusters, however, working principles are quite similar. Gas is ionized by different

methods and is accelerated by electromagnetic forces and is ejected from the exit which provides thrust to vehicles.
In ion and Hall thrusters, xenon gas is generally preferred as fuel due to its large atomic mass, easiness to store, and difficulty to adhere on spacecraft surfaces at low operational temperatures [7].

1.2.1 Background

The background of the electric propulsion dates back to the first decade of the nineteenth century. Robert Goddard from U.S.A, Hermann Oberth from Germany, Shepherd and Cleaver from Britain described early principles for electric propulsion [7]. Many of researches in this field have been conducted by National Aeronautics and Space Administration (NASA), Europan Space Agency (ESA), and various group from Russia and Japan.

The shared aim of these groups was to increase specific impulse, thrust and thruster efficiency. By the beginning of the twenty first century, some private companies have provided electric propulsion devices for specific purposes such as the launch and orbiting of the communication satellites.

1.2.2 Types of Electric Propulsion

Generally speaking, there are three categories of electric propulsion: electrothermal propulsion, electrostatic propulsion, and electromagnetic propulsion.

1.2.2.1 Electrothermal Propulsion

Propellant is heated by electrical energy, and the heated gases are ejected through the nozzle to form thrust force. According to the heating techniques of propellant gas, the electrothermal propulsion devices can be classified as: arc-jets, resistojets, and inductively heated devices [1].

1.2.2.2 Electrostatic Propulsion

In this type of the propulsion, ionized gases are accelerated by the external electric field. Well-known types of this propulsion are ion thruster, and Hall thruster. Main parts of this thruster are ion source module, acceleration channel where external electric field is applied, and the neutralizer. Electrons are ejected from the hollow cathode placed outside of the thruster in order to neutralize the ions.

1.2.2.2.1 Ion Thruster

Inert gas injected on the highly energized electrons from the discharge cathode ionizes the gas by means of bombardment or electron cyclotron resonance (ECR) [28]. Subsequently,

ions are produced and accelerated downstream of the thruster. Potential of the generated ions location is highly positive while the downstream electrode potential is negative which results in the attraction of ions in the downstream direction. The high speed ions are ejected through the nozzle.

1.2.2.2.2 Hall Thruster

The working principle of Hall thruster is quite similar to the ion thruster except that in Hall thrusters, electromagnetic fields are also used for ionization. Also, because electromagnetic effects are not used for acceleration purposes they differ from the electromagnetic propulsion (explained in section 1.2.2.3) [29].

As can be seen in Figure 1-1, the aim of magnetic field lines is to trap electrons. Propellant gas is injected into trapped electron region causing ionization. By the means of strong electric field on the ions, they are repelled and ejected from exit to produce thrust force.

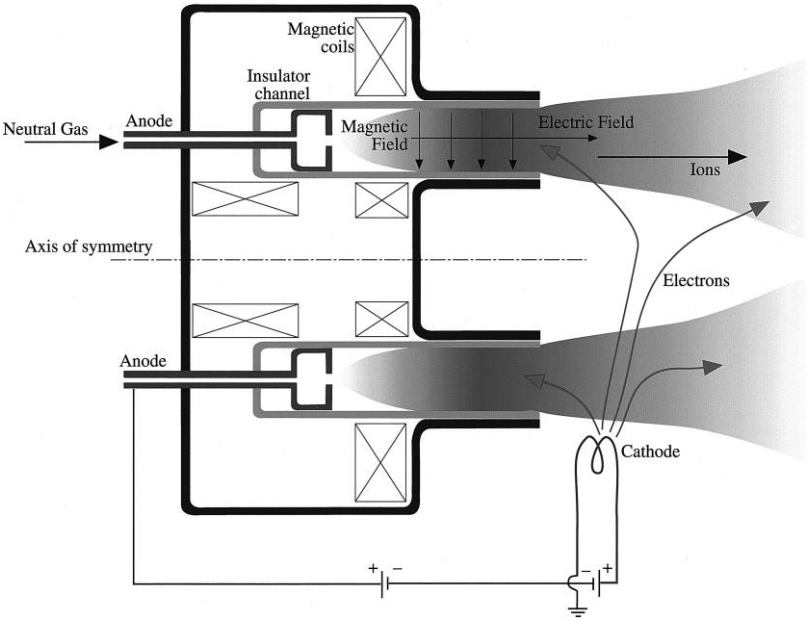


Figure 1-1: General schema of the Hall Thruster[15]

Widely used in space missions, Hall thrusters can be divided into two categories in terms of channel length and channel conductivity: Stationary Plasma Thruster (SPT) and Thruster with Anode Layer (TAL). The SPT has acceleration channel covered with insulator ceramics which can be considered as an advantage from the duration point of view. Particles colliding with boundaries can cause surface erosion which shortens the life time of thruster [9].

In second type of Hall thruster, TAL, the ceramics insulator wall is replaced by metallic conductor wall considered as an advantage of shortening the acceleration channel. This result in decrement of biased voltage values compared to SPT [31].

1.2.2.3 Electromagnetic Propulsion

In electromagnetic propulsion, ionized species are also trapped or accelerated by internal, external magnetic fields as well as electric fields. Utilization of electricity in magnetic medium can result in the creation of a force perpendicular to both magnetic field lines and driven current. If this force is applied to ionized gas with an aim of increasing the exit velocity, then, the goal of the electromagnetic propulsion is achieved.

Most common types of electric propulsion devices and their performance characteristics are presented in Table 1-1 given in [3].

Table 1-1: Characteristics of various electrothermal propulsion devices

Thruster	***Specific Impulse (s)***	***Input Power (kW)***	***Efficiency Range (%)***	***Propellant***
Cold Gas	50-75	-	-	Various
Chemical (Monopropellant)	150-225	-	-	N_2H_4 H_2O_2
Chemical (Bipropellant)	300-450	-	-	Various
Resistojet	300	0.5 – 1.0	65-90	N_2H_4 monoprop
Arcjet	500-600	0.9 - 2.2	25-45	N_2H_4 monoprop
Ion thruster	2500-3600	0.4-4.3	40-80	Xenon
Hall thrusters	1500-2000	1.5-4.5	35-60	Xenon
PPTs	850-1200	< 0.2	7-13	Teflon

1.3 Computer Simulation

Few decades ago, physical problems used to be examined only theoretically or experimentally. Our knowledge about science and nature has been nurtured by the combination of these powerful ways. Recent years brought about another efficient way to handle physical problems. By the aid of hi-tech computations, computer simulation now leads the way. The main advantage of using computer simulation is that it is cheap and applicable over broad ranges of problems. In fact, computer simulations, especially in plasma physics, play a crucial role.

1.4 Modeling of Thruster Plume

The plasma plume is a region where rarefied gases flow to outward space. Complex interaction can occur between charged particles as well as between surfaces and particles. Ions and neutral atoms can be simulated via the direct simulation Monte Carlo (DSMC) method which is a numerical method for modeling rarefied gas flows [2]. Monte Carlo Collisions (MCC) is also a numerical way to simulate collisions between particles. In DSMC, both the target and the source are particles. In MCC, however, the target is a cloud of particles. Moving the charged particles in electromagnetic fields is accomplished by the Particle in Cell (PIC) method [16].

There are three widely used approaches in plasma physics: “kinetic approach”, “fluid approach” and the combination of these known as the “hybrid approach”. These approaches are applied according to the ratio of electron mean free path and plasma length. If the size of plasma is small when compared to mean free path, then the plasma cannot be considered in continuum region. The properties of plasma are determined from the summation of the effects of individual particles in the microscopic aspect. This approach is known as kinetic. In fluid model, plasma is in thermal equilibrium and there is no need to trace individual particles to learn their plasma properties. In fact, the properties of these plasmas can be extracted from the macroscopic aspect [27] and the properties of the system are determined from fluid equations. Lastly, hybrid model is a combination of the both particle and fluid approaches. In plasma plume, the ions and heavy particles are modeled as particles while the electrons are simulated by fluid approaches. This reduces the computational task significantly.

1.5 VORPAL

Originally developed by the University of Colorado, VORPAL is a commercial computer code which simulates plasmas, dense and rarefied gases and calculates the dynamics of the electromagnetic and the electrostatic systems. Now, it is being marketed by Tec-X Corporation.

There is an input interface where users can compose their input files in the format of Extensible Markup Language (XML). Composing input file makes it highly flexible to deal with distinct branch of the plasma problems. To simulate plasmas kinetically, PIC method is used. Finite Difference Time Domain (FDTD) method is applied to solve fields. Moreover, output file is in the format of HDF5 (Hierarchical Data Format). In this thesis, a demo version of VORPAL is used to compare the present results.

1.6 SIMPFORT

SIMPFORT is an abbreviation for Simulation of Plasmas in Fortran Language. The code is written by the author of this thesis and it is capable of simulating collisionless plasmas where the effects of magnetic forces are neglected in 2D rectangular and some specific non-rectangular domains. SIMPFORT is a dynamic and flexible program. In other words, it is easy to change inputs of the simulation domain and properties of the particles.

SIMPFORT is composed of three main blocks which are the main program itself, the field solver and the output visualization part. In the main program, users define properties of the plasma species, the boundaries and simulation domain. During the calculations, the main block calls the field solver and at the end, the visualization routine is called to write output data in the format of “dat”.

CHAPTER 2

NUMERICAL CONSIDERATIONS AND THEORY

2.1 Particle in Cell (PIC) Method

The plasma consists of charged particles interacting with the electromagnetic field. In plasma, singly or doubly charged particles can exist, as well as neutral and dust particles. Particles in electromagnetic field are exposed to Lorentz forces. The force applied to a test particle of charge q_p with the position, $\boldsymbol{x_p}$, and the velocity, $\boldsymbol{v_p}$, in an electric field, $\boldsymbol{E}$ **,** and a magnetic field, $\boldsymbol{B}$, is defined as :

$$\boldsymbol{F_p} = q_p \left[\boldsymbol{E}(\boldsymbol{x_p}) + \boldsymbol{v_p} \times \boldsymbol{B}(\boldsymbol{x_p})\right] \tag{2.1}$$

As can be seen in equation (2.1), the fields are dependent on the test particle locations, and advancing the test particles in time can alter the fields.

In nature, the electromagnetic field is a combination of the external field and self-induced field created by the motion and the position of the charge particles. The Maxwell equations given in equation (2.2) deal with the calculation of these fields and are given as follows:

$$\begin{aligned} \boldsymbol{\nabla}.\mathbf{E} &= \frac{\rho}{\epsilon_0}, \\ \boldsymbol{\nabla} \times \mathbf{E} &= -\frac{\partial \boldsymbol{B}}{\partial t}, \\ \boldsymbol{\nabla}.\boldsymbol{B} &= 0, \\ \boldsymbol{\nabla} \times \boldsymbol{B} &= \mu_0 \boldsymbol{J} + \mu_0 \epsilon_0 \frac{\partial \boldsymbol{E}}{\partial t} \end{aligned} \tag{2.2}$$

where ρ and μ_0 are charge density and vacuum perrneability, respectively.

The Maxwell's equations should be solved simultaneously with Lorentz forces which determine the particle trajectories. The equation of motion is simple. However, it requires heavy computational task, but with the aid of some assumptions the procedure of attaining solution can be simplified. If magnetic field is small and there is lack of inductive electric fields, then it can be assumed that the plasma is unmagnetized [10]. In this case, there is no need to solve the whole Maxwell's equations. In fact, the Poisson's equation is sufficient to determine field values. In fact, this is used in the present thesis.

There are basically two ways to simulate plasmas kinetically. Firstly, "Particle-Particle" (PP) approach can be applied to simulate strongly coupled systems. In this case, the number of particles is quite small. For this reason, it is computationally manageable to simulate all particles as real particles. The forces acting on the particles are the summation of the Lorentz

forces from other particles. The particles are advanced in accordance with the values of these forces.

If the number of particles in debye length (N_D) is enormous, then, the system is weakly coupled. In this second case, a group of particles can be represented by a computational particle (superparticle) to increase the computational efficiency. The computational particle has distinct shape function (explained in the next section).

To compute the force on a computational particle, the field equations are solved by the Maxwell's equations (2.2) rather than summing up all effects of the individually charged particles in the region. This method is called "Particle Mesh" (PM). Particle in Cell (PIC) method is a derivation of this method.

Particle in Cell (PIC) method is a technique widely used in kinetic plasma simulations. This method is a combination of the Lagrangian and the Eulerian frames. Particles are traced with an aim of finding their location based on Lagrangian approach, while properties of the field such as charge density and current are computed at fixed Eulerian nodes [18].

Plasma simulation modeling is presented in Figure 2-1.

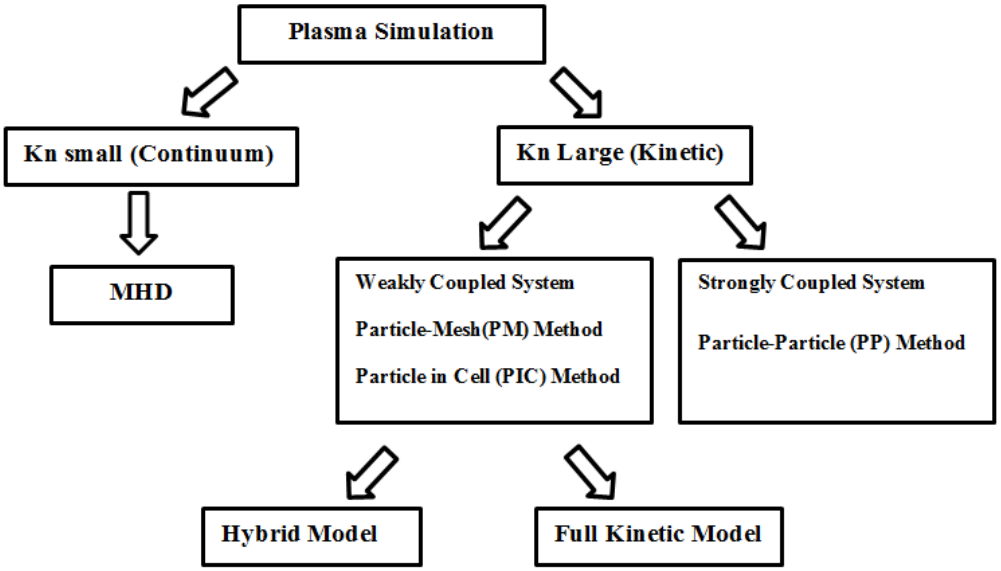


Figure 2-1: Plasma simulation model

2.2 Mathematical Background of PIC

The collisionless plasma can be handled by the Vlasov equation, also known as the collisionless Boltzmann equation. The Vlasov equation gives the rate of change of the distribution functions in physical and phase domains under the presence of external forces. In electrostatic field conditions, the Vlasov equation can be written in one dimension as:

$$\frac{\partial f(x,v,t)}{\partial t} + v\frac{\partial f(x,v,t)}{\partial x} + \frac{q\,E}{m}\frac{\partial f(x,v,t)}{\partial v} = 0 \tag{2.3}$$

where f is the distribution function, v is the speed, x is the position in phase space, E is the electric field and q and m are the charges and masses of particles, respectively.

The distribution function is evaluated from the collection of computational particles which can be mathematically represented as:

$$f(x,v,t) = \sum_p N_p\, S_x\big(x - x_p(t)\big) S_v\big(v - v_p(t)\big) \tag{2.4}$$

where N_p is the number of physical particles represented by the computational particle and S_x, S_v are the position and velocity shape functions, respectively. The mathematical expressions of S_x and S_v are the same. For example, S_v is Dirac function since the velocities of the physical particles in a computational particle are taken as equal [12]

$$S_v\big(v - v_p(t)\big) = \delta\big(v - v_p\big). \tag{2.5}$$

2.2.1 Grid weighting and the shape function

Weight functions in space are derived from spline functions and are used in distributing the particles to neighboring nodes surrounded by the cell. In particle physics, three of them are commonly used. The zero order spline is given as follows:

$$b_0(\xi) = \left\{ \begin{array}{ll} 1 & if\ |\xi| < \frac{1}{2} \\ 0 & else \end{array} \right\}. \tag{2.6}$$

The zero order spline is used in the Nearest Grid Point (NGP) interpolation. Particle location is mapped to the nearest cell. Higher order splines are obtained from the previous order splines by the following formula:

$$b_l(\xi) = \int_{-\infty}^{\infty} d\xi'\, b_0(\xi - \xi') b_{l-1}(\xi'). \tag{2.7}$$

Taking integral of the equation (2.7) results in the weighting function below:

$$S_x(x - x_p) = \frac{1}{\Delta} b_l\left(\frac{x - x_p}{\Delta}\right) \tag{2.8}$$

where Δ is the length of the cell, x_p is the location of the computational particle, and x is the grid point location. The zero, first and second order weighting functions are shown in Figure 2-2. In the present study, the first order weighting function is used.

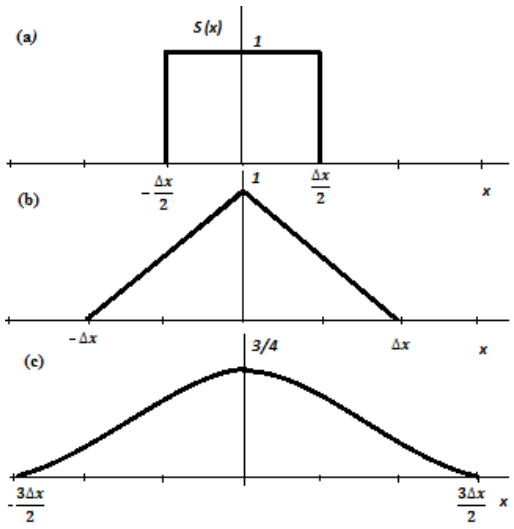


Figure 2-2: Various weighting functions ***a)*** NGP (zero order), ***b)*** CIC or PIC (first order), ***c)*** Parabolic one (second order)

2.2.2 Particle in Cell Method Computation Loop

In order to calculate electromagnetic force, field properties such as charge density and current density should be calculated at the nodes of the grid to solve the Maxwell Equations at these nodes.

PIC method starts with searching the particle location. According to the particle position, the cell in which the particle is located is determined. Then, computational particles are distributed to nodes. This process in PIC method is known as "gathering". After gathering, the charge density is determined at the nodes by solving Poisson's equation. Then, taking the divergence of the potential field, the electric field at the nodes are calculated. Next, the electric field on each computational particle is evaluated. This step is named as "scattering". Finally, computational particle is advanced by the force resulting from the electric field. This procedure is applied at all computational particles in domain.

General schema of the PIC is given in Figure 2-3:

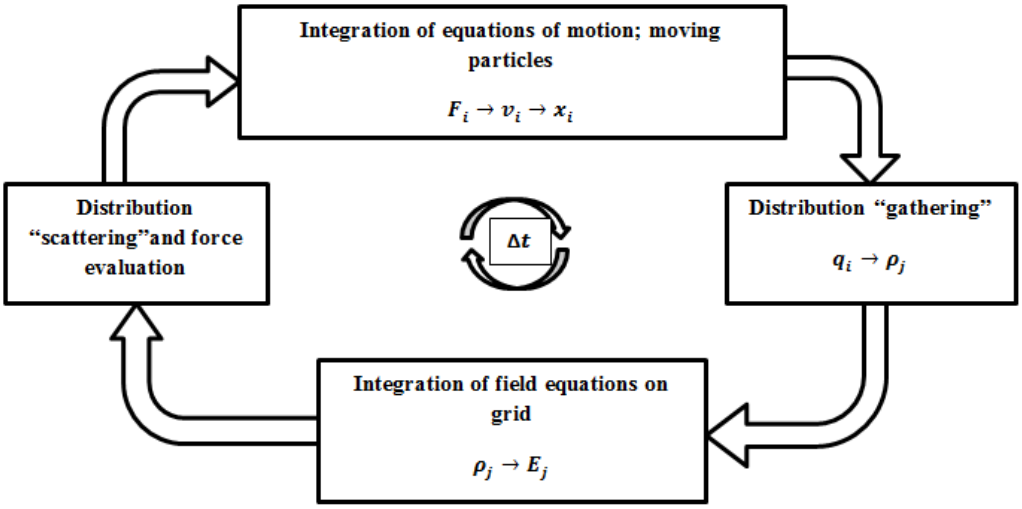


Figure 2-3: Particle in Cell (PIC) procedure

2.2.2.1 Distribution of Particles to Cell Nodes ("gathering")

This starts with finding the cell in which the particle is located. Since a structured mesh is used in this thesis, it is an easy task to find the particle's location and the cell it is in. In this study, to distribute the particles to surrounding nodes of the cell, the first order weighting function presented in the Table 2-1 is used. Weighting process is applied for every computational particle in the domain. The number of computational particles at the nodes can be computed as follows:

$$q_j = \sum_i q_i S_x(x_i - x_j) \tag{2.9}$$

where q_i is number of computational particles in the cells; S_x is the weighting function, x_i is the particle location, and x_j is the cell location. The charge density (ρ_j) at the nodes is given as :

$$\rho_j = \frac{sw * e * q_j}{(dh)^2} \tag{2.10}$$

where dh is the cell size, sw (specific weight) is defined as number of the real particles represented by one computational particle, and e is the elementary charge.

In Figure 2-4, the distribution procedure to the nodes can be observed where the weighting factors are determined according to both the particle position and the node locations.

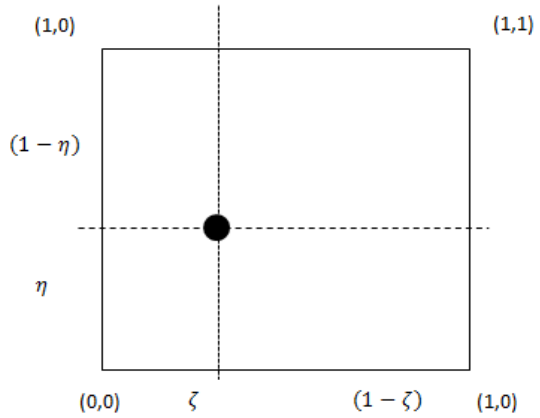


Figure 2-4: Weighting to the nodes

Table 2-1: The weighting factors

Node	***Weight (W)***
(0,0)	$(1-\zeta)(1-\eta)$
(1,0)	$\zeta(1-\eta)$
(0,1)	$(1-\zeta)\eta$
(1,1)	$\zeta\eta$

2.2.2.2 Field Scattering

After the calculation of the electric fields at the nodes, the values of the electric fields on each particle are determined by distributing $\boldsymbol{E}$'s on the particle positions by equation (2.11)

$$\boldsymbol{E_i} = \sum_{j=1}^{4} \boldsymbol{E_j}\, S(x_i - x_j). \tag{2.11}$$

Here, $\boldsymbol{E_i}$ is the electric field on the particles and $\boldsymbol{E_j}$ is the electric field on the nodes.

In the present study, there are four nodes surrounding the particle in 2D. That is the reason why the summation ends with the value 4 in equation (2.11).

2.3 Meshes

In computational methods, discretization of the domain is an essential task in solving a governing equation or tracing the particles. This can be achieved by using structured and unstructured grids.

Structured meshes are generally applicable to simple geometries. Each cell in this type of grid has *i, j* and *k* indices. The location of the cell can be computed by simple mathematical operations.

On the contrary, in unstructured grids, cell data could not be expressed by simple mathematical equations. The coordinates of each node is saved in the memory and if necessary, the particle location is searched to find which cell the particle is in. Hence, this approach requires memory. But, the merit of unstructured grid is that it has the capability to compute complex geometries.

Most of the plasma plume simulations in literature use structured meshes since space domain does not contain relatively complex surfaces. However, surface of a spacecraft may have complex geometry, and it should be gridded with unstructured meshes to investigate sputter or surface erosion [30].

In PIC method, the particles are traced individually. For this reason, the deficiency in memory is a main problem in computations. Obviously, a less demanding method is preferable and for simple geometries structured meshes are used [13].

In this thesis, the computational region is divided into grids by two-dimensional structured meshes. The cell lengths are kept constant throughout the domain. The field equations are solved by using the same grids.

2.4 Particle Boundary Conditions

In molecular dynamics, reflective, open, periodic particle boundaries are being applied for the interactions between the particle and the external boundaries according to the nature of the problem.

The open boundary conditions can be applied to simulate vacuum conditions and the particle is removed from the simulation. This boundary condition can be applied at absorptive walls as well. Particles absorbed by the boundary should be removed from the memory. Otherwise, the memory of computer is wasted unnecessarily. This is performed by replacing the removed particle by the last particle in the domain which cleans out the memory and decreases the array size.

In periodic boundary condition, particles are injected at the opposite side of the region when particles leave the region in steady flows [11].

The reflective boundary conditions are useful at the symmetry boundaries. The particles are reflected specularly at the boundaries. Both specular and periodic boundary conditions are shown in Figure 2-5.

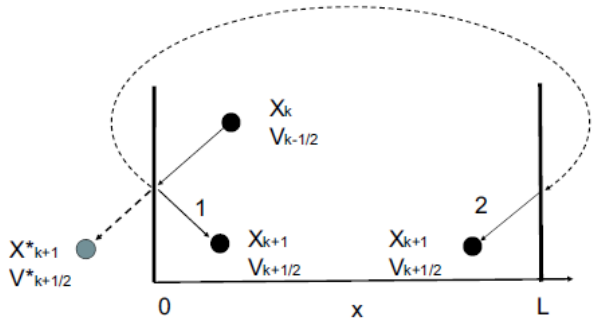


Figure 2-5 Specular and periodic boundary conditions for particles [11].

2.5 Particle Loading

In the simulations, particles are introduced to the domain initially volumetrically and also by specified boundary loadings during the calculations. In the former loading case, particles are distributed randomly throughout the whole region. In the latter type, particles are injected into the domain at specified boundaries. The number of the real particles that are injected at each time step is equal to the multiplication of the particle flux and time step (dt). Flux can be found by multiplying the number density, drift velocity assigned for that direction and the injected surface area. The number of real molecules represented by a computational molecule (sw) is:

$$sw = \frac{flux * dt}{number\ of\ computational\ particles\ injected} \quad . \tag{2.12}$$

2.5.1 Maxwellian Distribution

If a system is in equilibrium, the number of particles in a velocity class must be constant in time [19]. Then, the systems temperature can be defined. The distribution function is Maxwellian in 2D and it is:

$$f(v_x, v_y) = f(v_x)f(v_y) = \frac{1}{\pi {v_t}^2} exp\left[-\frac{({v_x}^2)}{{v_t}^2}\right] exp\left[-\frac{({v_y}^2)}{{v_t}^2}\right], \tag{2.13}$$

$$v_t = \sqrt{\frac{2KT}{m}}, \tag{2.14}$$

$$v^2 = {v_x}^2 + {v_y}^2 \tag{2.15}$$

where v_t is thermal velocity; T is temperature; K is Boltzmann's constant; m is mass; v is speed; v_x is the velocity component in x direction and v_y is the velocity component in y direction.

Also, initial Maxwellian distribution for velocity components can be obtained by using two distinct random variables, $Rand$. It is given as [20] :

$$\begin{aligned} |v| &= v_t\sqrt{-In\,(Rand)}\,, \\ \emptyset &= (2\pi * Rand)\,, \\ v_x &= |v|\sin\emptyset\,, \\ v_y &= |v|\cos\emptyset\,. \end{aligned} \tag{2.16}$$

2.6 Equation of motion

The particles in simulation region are exposed to electromagnetic forces resulting from the self-induced field and the external field applied to the region. The external field results from the field boundary conditions and the self-induced forces arise from the particles and their motions. Simply, the equations of motion of the particle in electromagnetic field are:

$$\frac{d\boldsymbol{v}}{dt} = \frac{q}{m}(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})\,, \tag{2.17}$$

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{v}. \tag{2.18}$$

2.6.1 Particle Movement

In plasma simulations, computational particles have to be advanced at every time step. To advance a particle, Newton's law of motion is applied. In this case, the new position and velocity of the particle is calculated at the same time. In this technique, time step must be small enough so as not to lose the accuracy of the particle trajectories. High order integration techniques can give more accurate results. However, the information of previous time step of the particle should also be stored in memory which is not preferable because it increases the computational load.

The Leap-frog method is highly preferable in particle simulations in terms of amount of data required from the previous steps and the accuracy to preserve the physical properties of the particles. In this method, the velocity and the position are calculated at different times for the particle. The time difference between position and velocity is half of the time step as can be seen in Figure 2-6. By this technique, the equations become centered in time as shown below [11] :

$$\frac{\boldsymbol{x}_{t+\Delta t} - \boldsymbol{x}_t}{\Delta t} = \boldsymbol{v}_{t+1/2\Delta t}\,, \tag{2.19}$$

$$\frac{\boldsymbol{v}_{t+1/2\Delta t} - \boldsymbol{v}_{t-1/2\Delta t}}{\Delta t} = \frac{\boldsymbol{F}_t}{m} \tag{2.20}$$

where t is current time, m is the mass of the particle, $\boldsymbol{F}_t$ is the external force applied on the particle. $\boldsymbol{x}$ and $\boldsymbol{v}$ are the position and velocity of the particle, respectively.

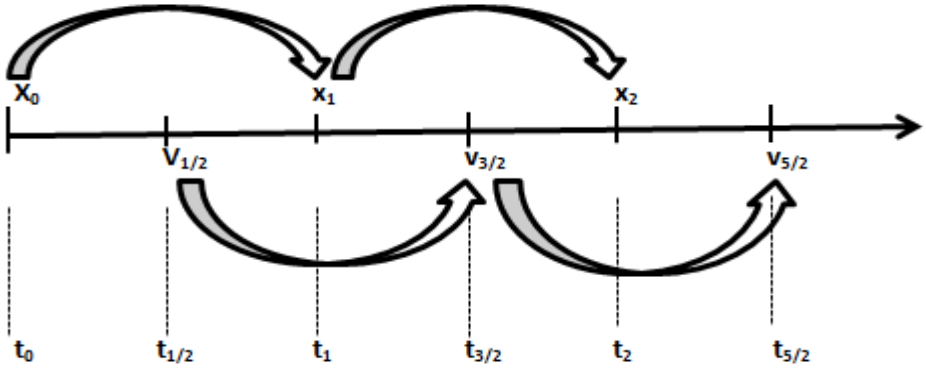


Figure 2-6 Leap-Frog integration schema.

2.7 Field Solver

The electromagnetic field becomes electrostatic if magnetic field does not change in time and space. Only then, the Maxwell's equations can be simplified into the Poisson's equation shown in equation (2.21) and this equation becomes the governing equation for electrostatic case.

$$\nabla^2 \emptyset = -\frac{\rho}{\varepsilon_0} \tag{2.21}$$

where $\emptyset$ is potential and ε_0 is permittivity.

In 2D Cartesian coordinates, this can be stated as:

$$\frac{\partial^2 \emptyset}{\partial x^2} + \frac{\partial^2 \emptyset}{\partial y^2} = -\frac{\rho}{\varepsilon_0}. \tag{2.22}$$

Many methods are available to solve this elliptic partial differential equation [4]. Finite Volume Method (FVM), Finite Difference Method (FDM) and also stochastic methods such as Monte Carlo Method are common in solving engineering problems. Monte Carlo method is applicable in rectangular and nonrectangular domains and can be effectively parallelized [5]. However, it is very slow when compared to FVM and FDM.

Particle methods require fast computations. This can be achieved easily by the finite difference methods in 2D using:

$$\frac{\emptyset_{i-1,j} - 2\emptyset_{i,j} + \emptyset_{i+1,j}}{\Delta x^2} + \frac{\emptyset_{i,j-1} - 2\emptyset_{i,j} + \emptyset_{i,j+1}}{\Delta y^2} = -\frac{\rho}{\varepsilon_0} \tag{2.23}$$

where i and j denote the x and y locations, respectively, and Δx and Δy are cell lengths.

Field boundary conditions in electrostatic case may be of different types:

(i) Dirichlet Boundary Condition: The potential is specified

(ii) Neumann Boundary Condition: The derivative of the potential is specified.

(iii) Robin Boundary Condition: The hybrid method where the above conditions are combined.

2.7.1 Gauss-Seidel Method

Gauss-Seidel method is an iterative way to solve system matrix and can be applied to solve the Poisson's equation. Diagonal system matrix is formed as follows:

$$Au = f\ . \tag{2.24}$$

Then, the unknown value of u at next iteration step is determined from the current data by using a formula which is presented in equation (2.25):

$${u_i}^{k+1} = \frac{1}{a_{ii}}\left(f_i - \sum_{j=1}^{i-1} a_{ij}\, {u_j}^{k} - \sum_{j=i+1}^{N} a_{ij}\, {u_j}^{k} \right). \tag{2.25}$$

The residual value (R) given in equations (2.26) and (2.27) is checked and if the norm (n) is smaller than the tolerance value, the computation is terminated.

$$k_i \equiv \sum_{j=1}^{N} a_{ij}\, {u_j}^{k} \tag{2.26}$$

$$R_i = f_i - k_i \tag{2.27}$$

$$n = \sqrt{\sum_{i=1}^{N} {R_i}^2} \tag{2.28}$$

2.7.2 Successive Over Relaxation (SOR)

To increase the convergence speed of Gauss-Seidel Method, the Successive Over Relaxation (SOR) method can be applied. The main idea of SOR is that it uses the available previous data to calculate the current value. The over relaxation parameter (w) plays an important role in convergence. The value of the relaxation parameter determines whether the system is in over relaxation or under relaxation.

SOR method is convergent only if w is between 0 and 2. If w is between 0 and 1, then it is said to be under relaxation. Otherwise, the system is over-relaxed. If w is equal to unity, the method converges to Gauss-Seidel method [14]. The explicit formulation of SOR method is given in equation (2.29):

$$\begin{aligned} {u_i}^{k+1} = w * \left[\frac{1}{a_{ii}} \left(f_i - \sum_{j=1}^{i-1} a_{ij}\, {u_j}^{k} - \sum_{j=i+1}^{N} a_{ij}\, {u_j}^{k} \right) \right] \\ + (1-w) \\ * \left[\frac{1}{a_{ii}} \left(f_i - \sum_{j=1}^{i-1} a_{ij}\, {u_j}^{k-1} - \sum_{j=i+1}^{N} a_{ij}\, {u_j}^{k-1} \right) \right]. \end{aligned} \tag{2.29}$$

2.7.3 Monte Carlo Method

Monte Carlo method is a stochastic way to solve differential equations. In this technique, first, the differential equations are converted into their finite difference equivalents. Then, according to the coefficients in front of the discretized finite difference formula, the sensitivity coefficients are determined. The direction of the walk is decided according to this sensitivity coefficients and the random number generated. The right hand side values of the differential equations on nodes through which the random walk passes and the potential values at the boundary at which the random walk stops are recorded. At the end, the solution of the differential equation at a given point becomes equal to the average of the recorded values [21].

The linear Poisson's equation can be solved in two steps. Firstly, the Laplace equation $\nabla^2 \emptyset_1 = 0$ is solved with the original boundary conditions and then, Poisson' equation $\nabla^2 \emptyset_2 = -\frac{e(n_i - n_e)}{\epsilon_0}$ is solved with zero boundary conditions. Because of the linearity,

solution of the equation becomes equal to summation of these two solutions $\emptyset = \emptyset_1 + \emptyset_2$ [22].

2.8 Electric Field Calculation

Electric field can be calculated from the derivative of the potential for a given field using :

$$\boldsymbol{E} = -\boldsymbol{\nabla}\emptyset. \tag{2.30}$$

In the present study, forward finite difference formula is used to calculate the electric fields at the starting nodes of the domain because there is no possibility to utilize the data in the backward directions. Similarly, backward finite difference is applied at the end of the domain due to the fact that there is no point in the forward directions. For other nodes, central difference formula is used to reduce numerical error.

2.9 Conservation Laws for Particle in Cell Method

2.9.1 Charge Conservation

Summation of the weighting factors shown in Table 2-1 is equal to 1, and the charge is conserved regardless of using different shape factors in PIC.

2.9.2 Momentum Conservation

Most of the PIC schemes conserve momentum. The time rate of change of the total momentum of the particles $(\boldsymbol{P})$ in the domain can be obtained using the equations numbered from (2.31) to (2.35) [16]. Equation (2.35) shows that this is independent of the shape functions.

The total momentum of the system is given by the Newton's Second Law:

$$\frac{d\boldsymbol{P}}{dt} = \sum_i \boldsymbol{F_i}\,. \tag{2.31}$$

For electrostatic case,

$$\frac{d\boldsymbol{P}}{dt} = \sum_i q_i \sum_j \boldsymbol{E_j} S(X_j - x_i) \tag{2.32}$$

where i and j denote the particle and grid positions, respectively. Rearranging equation (2.32),

$$\frac{d\boldsymbol{P}}{dt} = \sum_j \boldsymbol{E_j} \sum_i q_i S(X_j - x_i). \tag{2.33}$$

After some mathematical operations and using

$$\rho_j(\Delta x) = \sum_i q_i S(X_j - x_i) \tag{2.34}$$

the change of the momentum of system becomes :

$$\frac{d\boldsymbol{P}}{dt} = \Delta x \sum_j \boldsymbol{E_j}\, \rho_j \,. \tag{2.35}$$

2.9.3 Energy Conservation

The energy of the PIC consists of field potential energy and particle kinetic energy. Field potential energy is given by [16] :

$$W_E = \frac{V_c}{2} \sum_j \rho_j \emptyset_j \tag{2.36}$$

where V_c is the volume of the cell and *j* denotes the node points.

Explicit and implicit PIC methods can be utilized numerically. The explicit PIC method introduces an energy increase called "numerical heating" while implicit PIC method gives rise to "numerical cooling". Both of these numerical phenomena are unphysical [25]. Langdon states that there is no PIC scheme at present conserving both energy and the momentum simultaneously [26].

2.10 PIC Stability

The important parameters in PIC methods are:

(i) time step for advancing particles, Δt ,

(ii) the length of the cell, Δx,

(iii) the number of computational particles in debye length, N_D.

To minimize the error, the time step and the cell size should be small. In addition to these, N_D should be large [11], *i.e.*,

$$w_p \Delta t < 0.2 \tag{2.37}$$

and

$$\lambda_D > 2\Delta x \,. \tag{2.38}$$

2.11 Boltzmann Electron Model

The simplest electron distribution can be achieved by using the Boltzmann relation for electrons. The governing equation for this model is derived from the conservation of momentum:

$$mn\left[\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v}\cdot\boldsymbol{\nabla})\boldsymbol{v}\right] = qn\boldsymbol{E} - \boldsymbol{\nabla}p\,. \tag{2.39}$$

The convection term in equation (2.39) is usually small compared to $\frac{\partial \vec{v}}{\partial t}$. Therefore, it can be neglected [8]. Using the ideal gas assumption $p = \gamma nKT$ and assuming the electron temperature is constant along the direction, the equation becomes:

$$\frac{\partial \boldsymbol{v}}{\partial t} = \frac{q}{m}\boldsymbol{E} - \frac{\gamma KT}{mn}\boldsymbol{\nabla}n\,. \tag{2.40}$$

Because the mass of electron is small in the above equation, that is $m \to 0$ and also $q = -e$ and $\boldsymbol{E} = -\boldsymbol{\nabla}\emptyset$, we get :

$$ed\emptyset = \gamma KT\,\frac{dn}{n}. \tag{2.41}$$

Furthermore, the electrons are so mobile that their heat conductivity is almost infinite. As a result, we can assume that electrons are isothermal and taking $\gamma = 1$ and integrating the above equation one can get [8] :

$$e\left(\emptyset - \emptyset_{ref}\right) = KT_e \ln n + C, \tag{2.42}$$

$$n = n_{ref}\exp\left(\frac{e(\emptyset - \emptyset_{ref})}{KT_e}\right). \tag{2.43}$$

The equation (2.43) is known as Boltzmann relation for electrons. The Boltzmann relation has the following assumptions:

- electrons are massless,
- electron flow is isothermal, and collisionless,
- electrons obey the ideal gas law,
- the magnetic field is negligible [9].

The Boltzmann relation for electron is derived only from electron momentum equation. To increase the accuracy of this model, a detailed model including Ohm's law was developed [9].

The electrostatic plume model is governed by the following equation:

$$\nabla^2 \emptyset = -\frac{e}{\varepsilon_0}(n_i - n_e) \tag{2.44}$$

where n_i and n_e are ions and electrons number density, respectively.

Inserting the Boltzmann relation to the above equation, this equation becomes non-linear Poisson equation:

$$\nabla^2 \emptyset = -\frac{e}{\varepsilon_0}\left[n_i - n_{ref} \exp\left(\frac{e(\emptyset - \emptyset_{ref})}{KT_e}\right)\right] \tag{2.45}$$

where n_{ref} and $\emptyset_{ref}$ are number density and potential at the reference point, respectively.

This equation is used in hybrid particle-fluid approach, where is no need to trace electrons. This helps to reduce the memory and shorten runtime.

2.12 Simulation of Plasmas

In plasma simulations, when all species are treated as particles, the approach is called kinetic approach. On the other hand, in fluid approach, the conservation equations of continuum are used. Furthermore, the combination of the particle and fluid approaches is known as hybrid model.

The drawback of the kinetic approach is its computational cost. Fluid approaches save time and memory in computations. However, the kinetic approaches have higher accuracy when compared to the fluid approaches. Also, in some regions like sheath layer near the walls, it is not valid to use Boltzmann electron model [24].

In this thesis, kinetic and hybrid approaches are used to simulate plasmas. In the former approach both ions and electrons are treated as particles. In the latter, the electrons are handled as a fluid.

2.12.1 Kinetic Approach

In the kinetic approach, all ions and electrons in the domain are advanced in time using the same time step. But, due to the fact that electrons move much faster compared to ions, using small time step is necessary. Although this approach increases computational work, it gives more accurate results.

The computer code written in this research uses the FORTRAN 90 language. It starts with the initialization of the parameters such as the arranging the arrays and assigning the initial potential values. After the initialization, ions and electrons are placed in the domain. According to the positions of the particles, the charge on nodes is computed by weighting factors. Then, the potential solver is called and the potentials at the nodes are calculated. Taking the derivatives of the potentials gives the electric field on the nodes. Due to the fact that the particles may not be at the nodes, the electric fields on the nodes are distributed back to the locations of the particles. The particles are moved according to Coulomb forces on the

particles. The particles that reach the boundaries are exposed to boundary conditions of the particles. This process is repeated until the steady state condition is reached. Figure 2-7 shows the structure of the kinetic approach of the code.

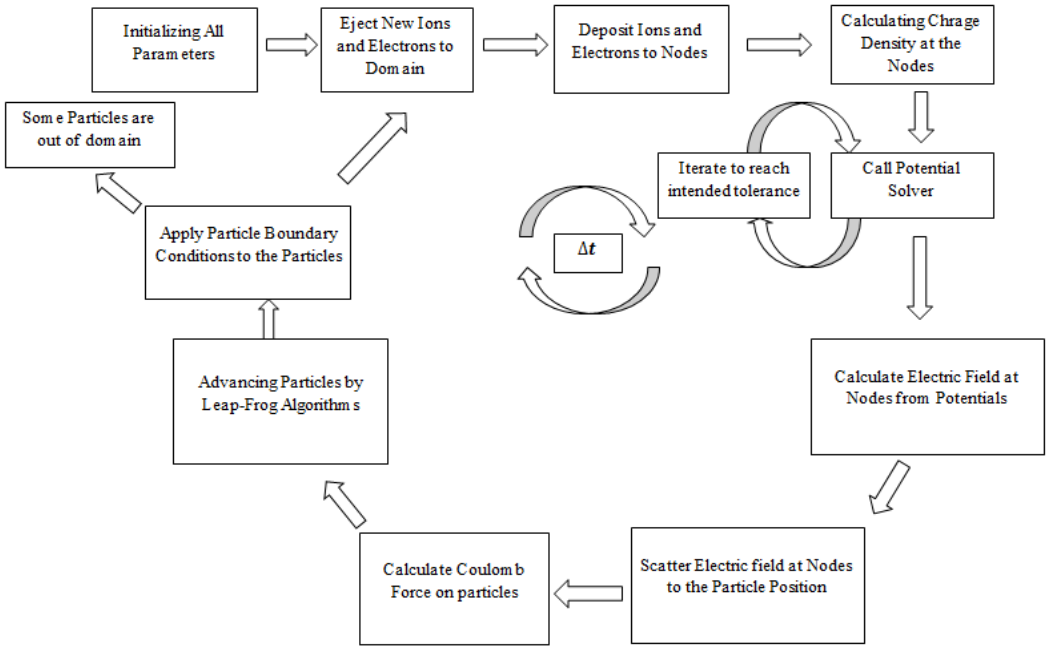


Figure 2-7: Flow chart of the kinetic FORTRAN code

2.12.2 Hybrid Approach

Hybrid model brings the kinetic and fluid approaches together. Since electrons move much faster than heavy particles, they can be treated as fluid whereas ions can be treated as particles. The electrons are distributed through the region by the Boltzmann relation which is usually adopted as the governing equation of the fluid [23].

The structure of the hybrid code is similar to the kinetic one with an exception of the influence of the electrons on the field. In this model, electrons are fluid and the electron effect on the domain is governed by Boltzmann relation. At this point, it is vital to remind that, only after the addition of the Boltzmann relation, the Poisson's equation becomes non-linear. This equation is solved iteratively by SOR. Because there is no need to trace electrons in hybrid model, this approach decreases the computational time significantly. The structure of the hybrid model is given in Figure 2-8.

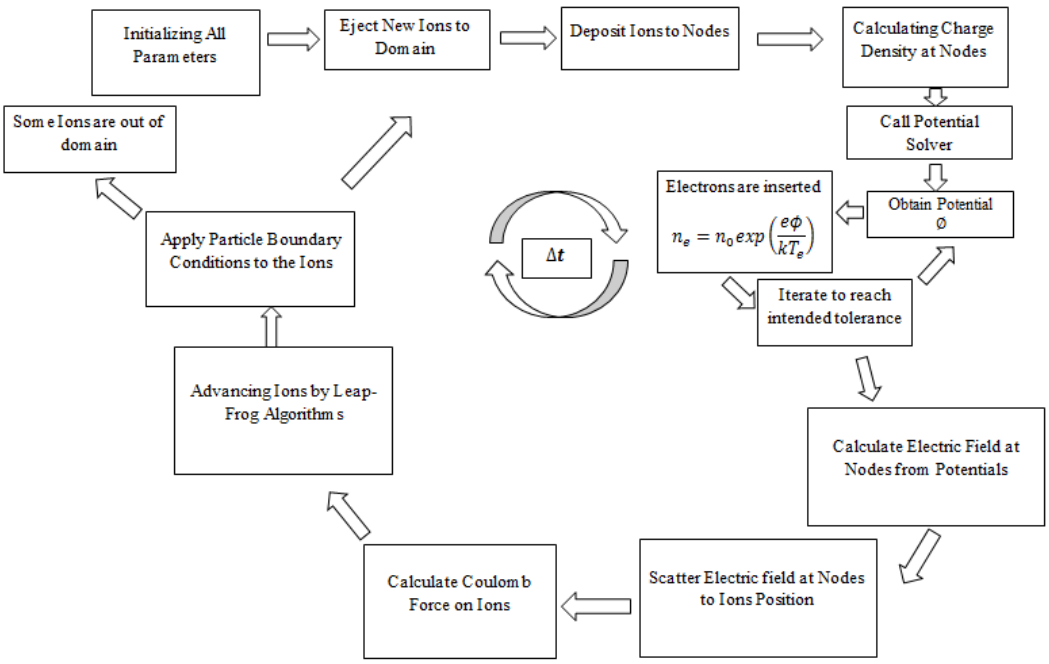


Figure 2-8: Flow chart of code in hybrid model

CHAPTER 3

RESULTS AND VALIDATION

3.1 Comparison of the Potential Solver with Different Solvers

In this thesis, the Poisson's equation is solved in rectangular and non-rectangular domain by different methods and its solutions are investigated and compared in aspects of the exactness and computational efficiency.

3.1.1 Rectangular Domain

The present potential field solver is tested by different solution techniques in rectangular domain and the results and computational efficiency of the solvers are compared. To validate results, equation (3.1) is used in 2D

$$\frac{\partial^2 \emptyset}{\partial x^2} + \frac{\partial^2 \emptyset}{\partial y^2} = -8\pi^2 \sin(2\pi x) \sin(2\pi y). \tag{3.1}$$

The exact solution of the equation (3.1) is given as:

$$\emptyset = sin(2\pi x) sin(2\pi y). \tag{3.2}$$

The solutions of the equation (3.1) with different solvers are shown on Figure 3-1.

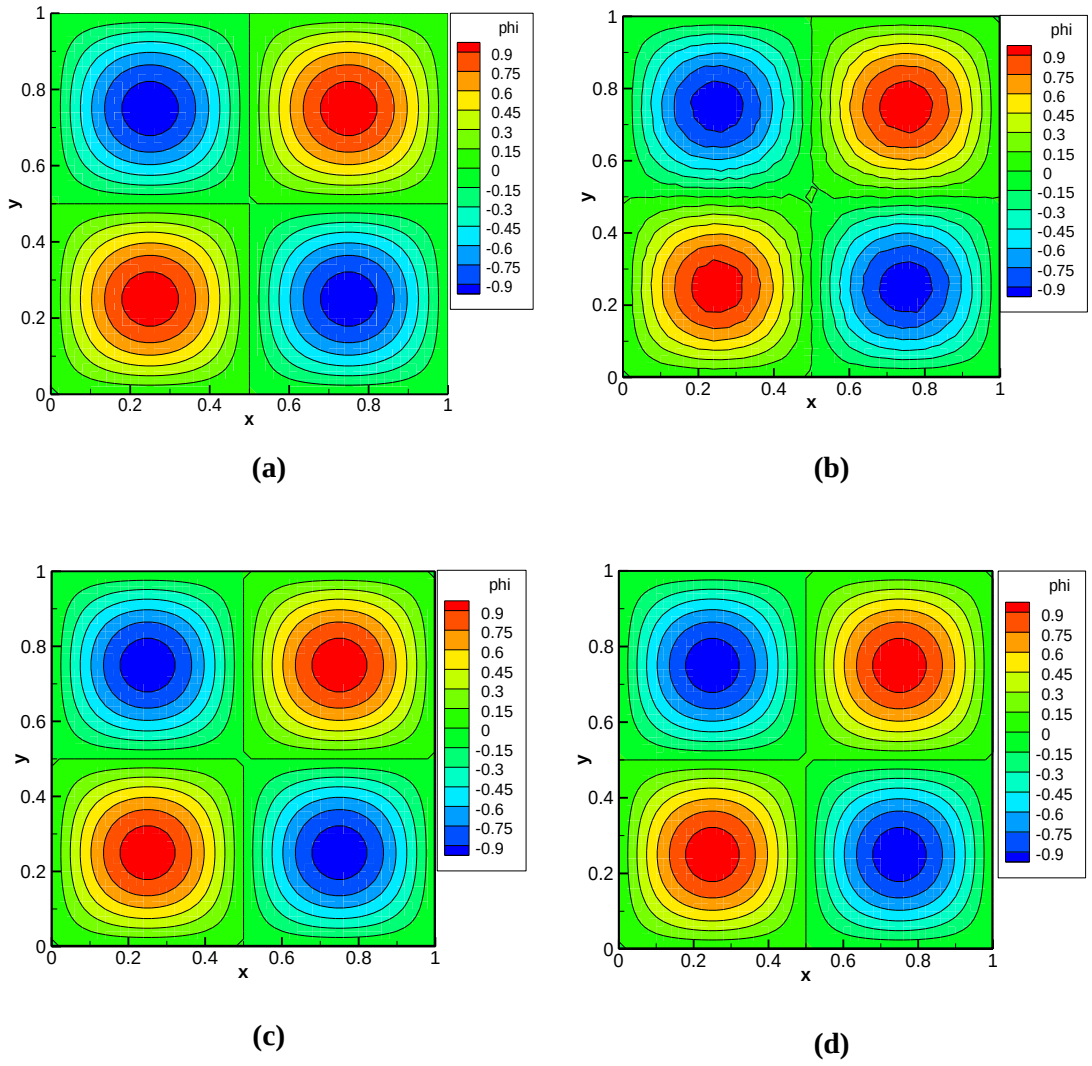


Figure 3-1: The solution of equation (3.1) ***a)*** Exact solution ***b)*** Monte Carlo Method ***c)*** Gauss-Seidel Solution ***d)*** SOR (w=1.92)

In order to investigate accuracy and efficiency of different solvers, some arbitrary points in the domain are chosen. The potential values and the errors produced by different solvers, i.e, Monte Carlo Method (MCM), Gauss-Seidel (G.S), Successive Over Relaxation (SOR) are given in Table 3-1.

Table 3-1: Solution of equation (3.1) for different solvers and errors belonging to them at some arbitrary points (x, y)

(x,y)	*Exact*	*MCM*	*GS*	*SOR*	*Error MCM*	*Error G.S*	*Error SOR*
(0.24, 0.24)	0.9961	0.9817	0.9974	0.9974	0.0143	0.0013	0.0013
(0.24, 0.50)	0.0000	0.0021	0.0000	0.0000	0.0021	0.0000	0.0000
(0.50, 0.50)	0.0000	0.0058	0.0000	0.0000	0.0058	0.0000	0.0000
(0.80,0.60)	0.5590	0.5594	0.5597	0.5598	0.0004	0.0007	0.0007
(0.80,0.80)	0.9045	0.9005	0.9057	0.9057	0.0040	0.0012	0.0012

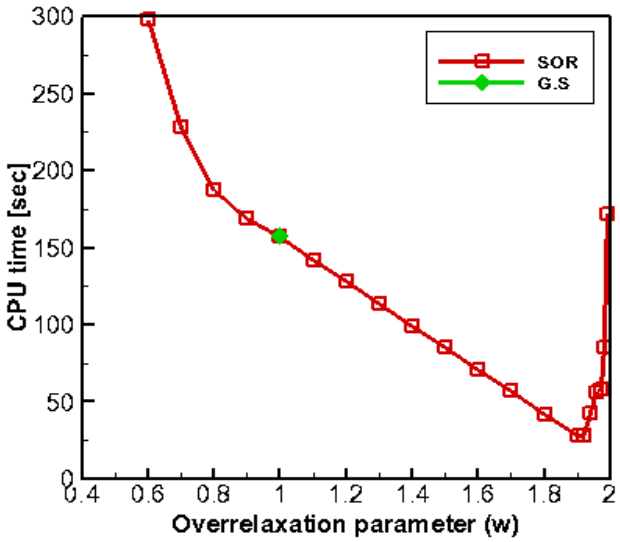


Figure 3-2 : CPU time with different over-relaxation parameter of SOR and Gauss-Seidel Method

As can be seen in Figure 3-2 where the relation between CPU time and over-relaxation parameter is depicted, convergence heavily depends on the over-relaxation parameter. In under-relaxation region where parameter is less than unity, CPU time is higher. In other words, Gauss-Seidel method is more efficient in comparison with SOR in under-relaxation region. The minimum CPU time is obtained for $w = 1.92$ and the regarding time is about 28.1 seconds for a tolerance of 0.01 in a 51x51 domain run with a PC having i5-2400 CPU @3.10 GHz. This over-relaxation parameter is obtained by trial and error. However, in literature, there is an exact formula based on eigenvalues of iterative matrix to it [32].

Table 3-2: Average error and CPU time for different solvers

	MCM	***GS***	***SOR***
Average Error %	0.0053	0.0006	0.0006
CPU Time (sec)	1749.75	156.75	28.1114

Results of Table 3-2 suggest that it is reasonable to use SOR with 1.92 relaxation parameter considering accuracy and efficiency. Therefore, this solver will hereafter be implemented as the chosen solver for the present Poisson's equation in rectangular domain.

3.1.2 Non-Rectangular Domain

The left, the right and the bottom edges of the physical domain are parallel to Cartesian axes, while the upper edge is a quadratic polynomial as shown in Fig. (3.3). Instead of solving the Poisson's equation in this domain, the physical domain (x, y) is mapped to the computational domain (ξ, η) via the analytical relations: $\xi = \frac{x}{d}$ and $\eta = \frac{y}{g(x)}$ where $g(x)$ in the present study is chosen as a polynomial,

$$g(x) = ax^2 + bx + c\,, \tag{3.3}$$

with $a = 0.2, b = 0.1, c = 1$ and $d = 1$ constants.

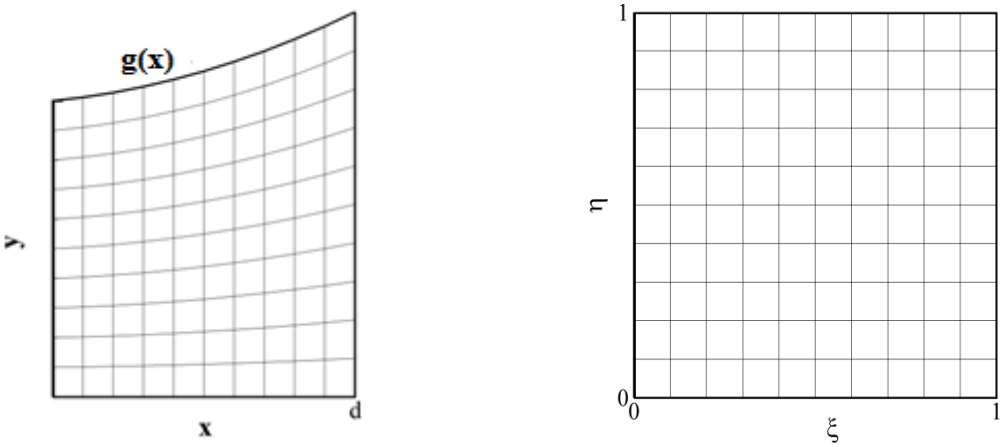


Figure 3-3: ***a)*** 2D non-rectangular physical domain. ***b)*** Square computational domain.

This square shaped computational domain incorporates uniform meshes. In the next step, Poisson's equation is also transformed to the computational domain as follows [17]:

$$\alpha\phi_{\xi\xi} - \beta\phi_{\xi\eta} - \chi\phi_{\eta} + \delta\phi_{\eta\eta} = h(\xi,\eta)\,. \tag{3.4}$$

The coefficients$(\alpha,\beta,\chi,\delta)$ in this equation are:

$$\alpha = \frac{1}{d^2}, \qquad \beta = \frac{2\eta(2ad\xi + b)}{d(ad^2\xi^2 + bd\xi + c)},$$
$$\chi = \frac{2\eta\left[a(ad^2\xi^2 + bd\xi + c) - (2ad\xi + b)^2\right]}{(ad^2\xi^2 + bd\xi + c)^2},$$
$$\delta = \frac{[1 + \eta^2(2ad\xi + b)^2]}{(ad^2\xi^2 + bd\xi + c)^2} \tag{3.5}$$

and the derivative terms in the present study are discretized as:

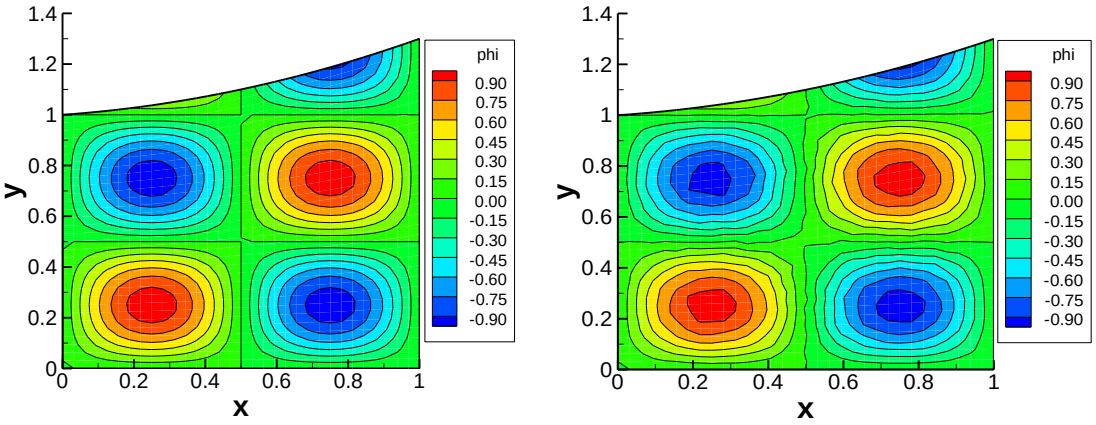
$$\phi_{\xi\xi} = \frac{\left(\phi_{i-1,j} - 2\phi_{i,j} + \phi_{i+1,j}\right)}{(\Delta\xi)^2},$$
$$\phi_{\xi\eta} = \frac{\left(\phi_{i,j+1} - \phi_{i,j} - \phi_{i-1,j+1} + \phi_{i-1,j}\right)}{(\Delta\xi\Delta\eta)},$$
$$\phi_{\eta} = \frac{\left(\phi_{i,j+1} - \phi_{i-1,j}\right)}{(2\Delta\eta)}, \tag{3.6}$$
$$\phi_{\eta\eta} = \frac{\left(\phi_{i,j-1} - 2\phi_{i,j} + \phi_{i,j+1}\right)}{(\Delta\eta)^2}.$$

In order to apply MCM to a finite-difference-scheme, the sensitivity coefficients in front of the $\phi_{i-1,j}, \phi_{i+1,j}, \phi_{i,j-1}, \phi_{i,j+1}$and $\phi_{i-1,j+1}$ terms should meet three important criteria. First, sensitivity coefficients should be smaller than one. Next, none of these coefficients should have negative values. Finally, their sums should be equal to unity [5]. It is verified that these requirements are satisfied successfully at all points in the domain in the present study, and Table 3-3 shows these verifications at some arbitrarily chosen points in the computational domain (ξ,η).

Table 3-3: Sensitivity coefficients at different locations of the computational domain

(ξ, η)	***probability to right***	***probability to up***	***probability to left***	***probability to down***	***probability to up and left***	***total probability***
(1/5,1/5)	0.261441	0.229060	0.243130	0.248057	0.0183111	1.000
(2/5,2/5)	0.280492	0.191722	0.226068	0.247293	0.0544239	1.000
(3/5,3/5)	0.306251	0.137923	0.195871	0.249574	0.1103800	1.000
(4/5, 4/5)	0.336815	0.069020	0.149448	0.257351	0.187367	1.000

The exact solution to Equation (3.1) given in equation (3.2) can be utilized as a comparison tool to investigate the capabilities of the solution of the Monte Carlo method. This can be sighted in Figure 3-4 and Table 3-4.



(a) **(b)**

Figure 3-4: Comparisons of the results ***a)*** Exact solution ***b)*** Solution of the Poisson's equation with MCM

Table 3-4: Exact and MC simulation results on the computational domain

ξ	***η***	***Exact***	***MCM***	***Error***
0.25	0.25	0.998266	0.981847	0.016
0.25	0.75	-0.95694	-0.950862	0.006
0.5	0.5	-1.94011E-007	0.007513	0.008
0.75	0.25	-0.984427	-0.970429	0.014
0.75	0.75	0.634394	0.639635	0.005

Table 3-4 indicates that the results attained by the MCM equation solver are in quite good agreement with the exact solution.

3.2 Momentum and Energy Consideration for the Overall System

To test the momentum conservation in Particle in Cell (PIC) method, a simple problem is devised in which the movements of ions in square domain are investigated. The particles are initially inserted into domain with Maxwellian distribution. The field boundary conditions at left, right, upper and lower edges are all Dirichlet Boundary Conditions with zero potentials. There is no external force in the simulation region because of the fact that no gradient results from these field boundary conditions. It is known that in periodic or infinite systems, if the net force acting on the particle is zero, then the momentum of the system should be constant [16].

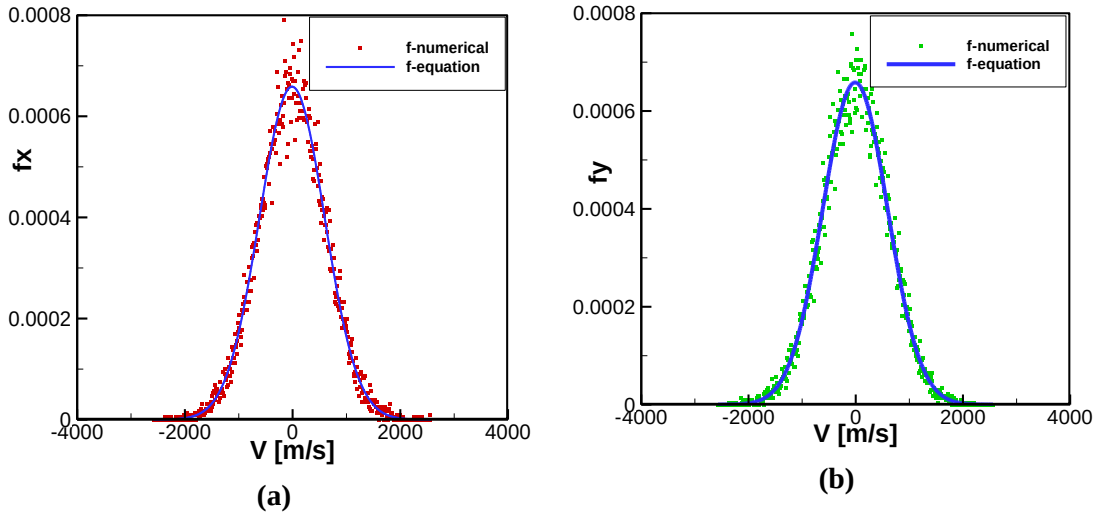


Figure 3-5:Initial distribution functions ***a)*** x direction, ***b)***y direction

Due to the fact that Xenon is widely used in electric propulsion, Xenon ions are inserted into the simulation region. The initial distributions of these ions in x and y directions are presented in Figure 3-5.

Different simulation parameters are used to investigate the effect of the spatial and time resolutions. The simulation region parameters are shown in Table 3-5. Case 1 has a coarse mesh with large time step. Fine mesh with large time step is used in the second case. In the third case, coarse mesh with small time step is used.

Table 3-5: Different computational parameters

	Properties of field and Ions	***Properties of field and Ions***	***Properties of field and Ions***	***Properties of field and Ions***
	Case 1	***Case 2***	***Case 3***	***Case 4***
Grid (N_x ,N_y)	*16 x 16*	*31x31*	*16x16*	*31x31*
Domain size (L_x ,L_y)	(0.558*mm*,0.558 *mm*)	(0.558*mm*,0.558*mm*)	(0.558*mm*,0.558 *mm*)	(0.558*mm*,0.558 *mm*)
Cell size	0.0372 *mm*	0.0186 *mm*	0.0372 *mm*	0.0186 *mm*
Mass of Particle	2.181e-25 *kg*	2.181e-25 *kg*	2.181e-25 *kg*	2.181e-25 *kg*
Charge of Particle	1.602e-19 *C*	1.602e-19 *C*	1.602e-19 *C*	1.602e-19 *C*
Weight of macro-particles, w	138384	138384	138384	138384
time step	1.e-8 s	1.e-8 s	1.e-9 s	1.e-9 s
Number density	1.e16 #/m^2	1.e16 #/m^2	1.e16 #/m^2	1.e16 #/m^2
# of particles in region	22500	22500	22500	22500
Temperature of ions	1.0*eV*	1.0*eV*	1.0*eV*	1.0*eV*

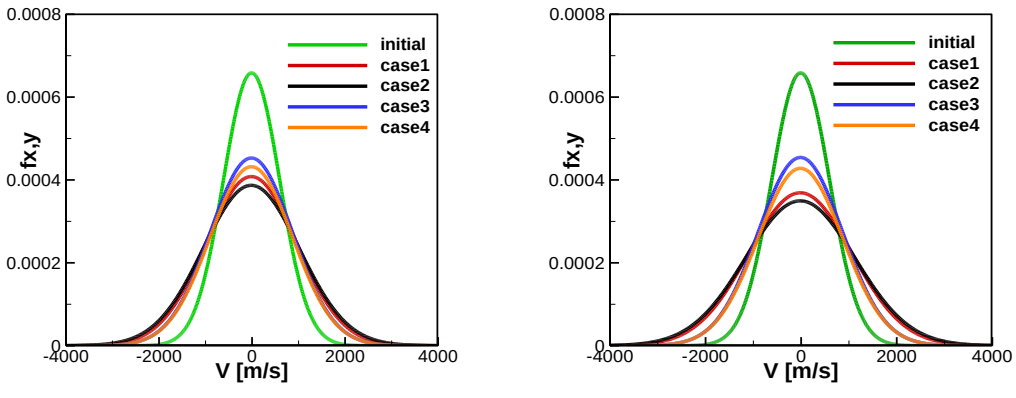
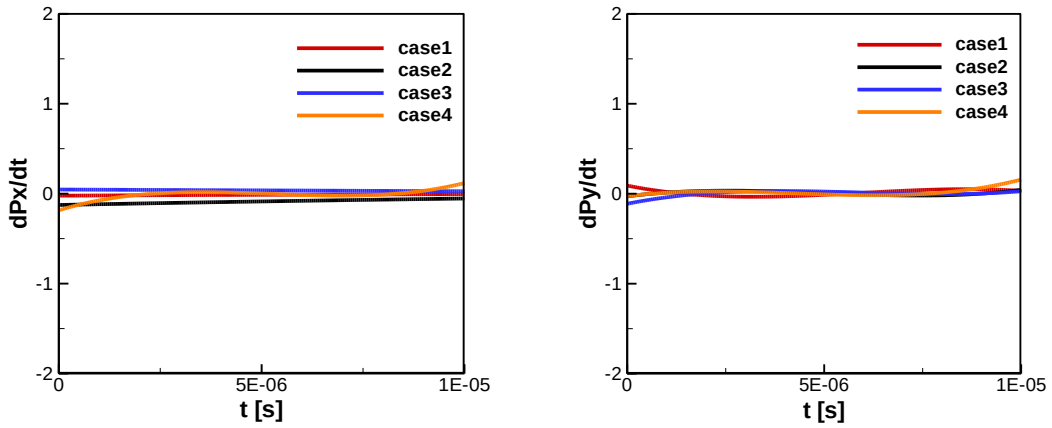


Figure 3-6 Distribution functions at ***a)*** 5*10^{-6}s and ***b)***10^{-5}s

The smearing of the distribution function in time shows the numerical heating produced in PIC and this can be investigated in Figure 3-6. Clearly, decreasing time step causes less numerical heating. It is interesting that using fine mesh increases the numerical heating as can be seen from Cases 1 and 2. Figure 3-6 also shows that numerical heating continues to increase with time in cases 1 and 2, but remains almost constant in case 3.
Since there is no external electric field on the system, the total momentum should be zero. In Figure 3-6, the velocity distribution function is symmetric about y axis which shows that the total momentum of the system is zero. In the present study, in one calculation, one boundary condition was changed and it was observed that the total moment was no longer zero.



(a) **(b)**

Figure 3-7: Time rate of change of momentum ***a)*** x direction ***b)*** y direction

As can be seen from Figure 3-6 and Figure 3-7, total momentum of the system and its rate of change with time is nearly zero for all cases. There exists a small deviation from zero resulting from numerical error.

The potential and kinetic energy of the system is also considered for these cases. It is observed that the kinetic energy increases, while potential energy of the system reduces in time and total energy of the system is not conserved. The reduction in potential energy does not compensate for the increment in kinetic energy. This phenomenon is shown in Figure 3-8. This is consistent with the statement given in section (2.9.3) by Langdon [26].

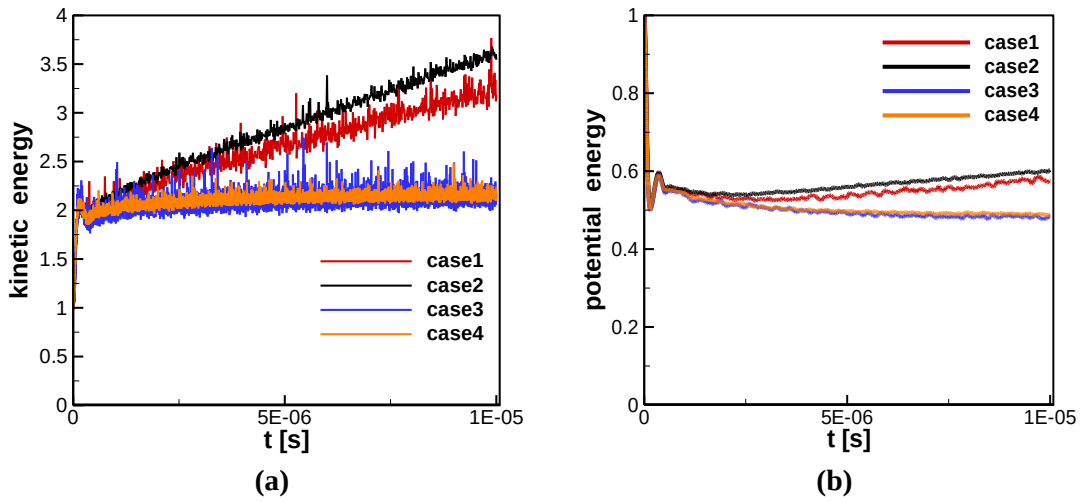


Figure 3-8***: a)*** Kinetic energy ***b)*** Potential energy of the domain normalized with initial kinetic and potential energies, respectively.

3.3 Test Problem 1 " Expansion into vacuum from a nozzle exit area"

In this test case, a region in front of a nozzle is simulated by means of both kinetic and hybrid approaches. The simulation domain is square as can be seen in Figure 3-9. The R1 boundary is nozzle exit, and particles are injected to the simulation domain from R1. The other boundaries simulate the vacuum conditions. Thus, the particles reaching these boundaries are removed from the memory of computer.

3.3.1 Boundary Conditions

The simulation domain shown in Figure 3-9 with boundaries R2, R3 and R4 for vacuum conditions are used with Neumann boundary conditions with zero derivatives. For the left boundary, R1, Dirichlet boundary condition is used with zero potential. The field boundaries are formulated mathematically as follows:

$$R_1 \rightarrow \phi = 0, \qquad R_3 \rightarrow \frac{\partial \phi}{\partial x} = 0, \qquad R_2, R_4 \rightarrow \frac{\partial \phi}{\partial y} = 0. \tag{3.7}$$

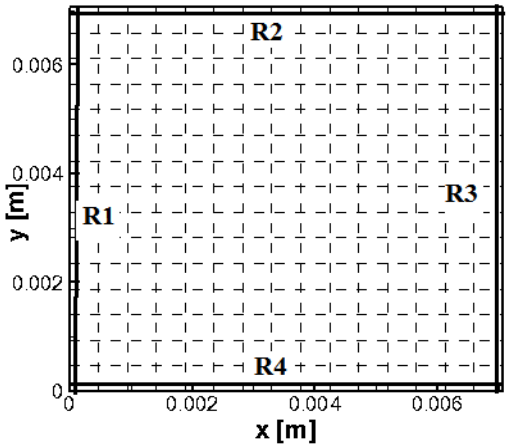


Figure 3-9: Simulation domain

The simulations parameters and properties of the species in simulation domain are shown in Table 3-6.

Table 3-6: Initial properties of ejected particles

	Electrons	***Ions***
Grid (N_x ,N_y)	*16 x 16*	*16 x 16*
Domain size (L_x ,L_y)	(7.053*mm* , 7.053*mm*)	(7.053*mm* ,7.053*mm*)
Cell size	0.47 *mm*	0.47 *mm*
Mass of Particle	9.109e-31 *kg*	2.181e-25 *kg*
Charge of Particle	-1.602e-19 *C*	1.602e-19 *C*
time step	1.e-11 s	1.e-11 s
# of injected particles per time step	75	75
Time step required to reach steady state	22500	22500
Temperature of injected particles	0.25 *eV*	-
Drift in x direction velocity of injected particles	0	30000 *m/s*
Drift in y direction velocity of injected particles	0	0

For this case, the hybrid code is capable of measuring the electron temperatures. Calculations show that the electrons keep their Maxwellian distribution but, the temperatures of the electrons change as time progresses. Several calculations are performed and it is observed that this variation depends on the field boundary conditions and the number density.

As can be seen the Figure 3-10, the electron temperature first shows a decreasing behavior and then increases. The initial temperature, 0.25 eV, is given in Table 3-6. Eventually, it reaches the equilibrium value, 0.78 eV.

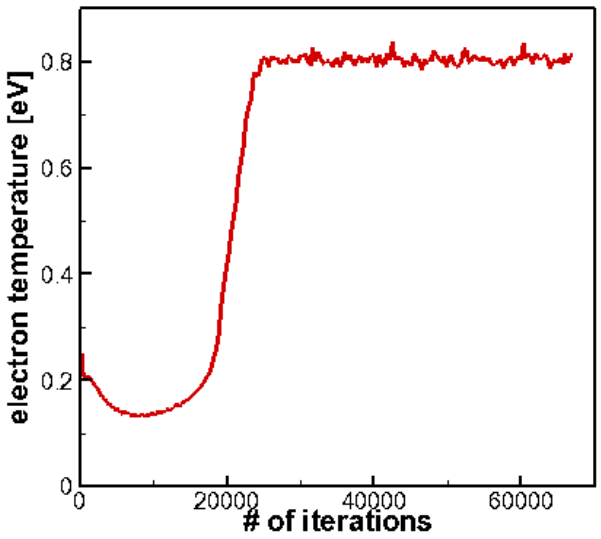


Figure 3-10: Variation of electron temperature with time

The governing equation of the particle-fluid hybrid model is derived in section 2.11. It can be seen that the non-linear term of the equation (2.45) is a function of the electron temperature. To investigate the effect of the electron temperature on the potential, the results of the hybrid model using initial electron temperature 0.25 eV, and equilibrium temperature, 0.78 eV are compared with those of the kinetic model steady state solution. This can be seen in Figure 3-11. The potential values along the x axis at $y = 0.0033$ are also plotted in Figure 3-12.

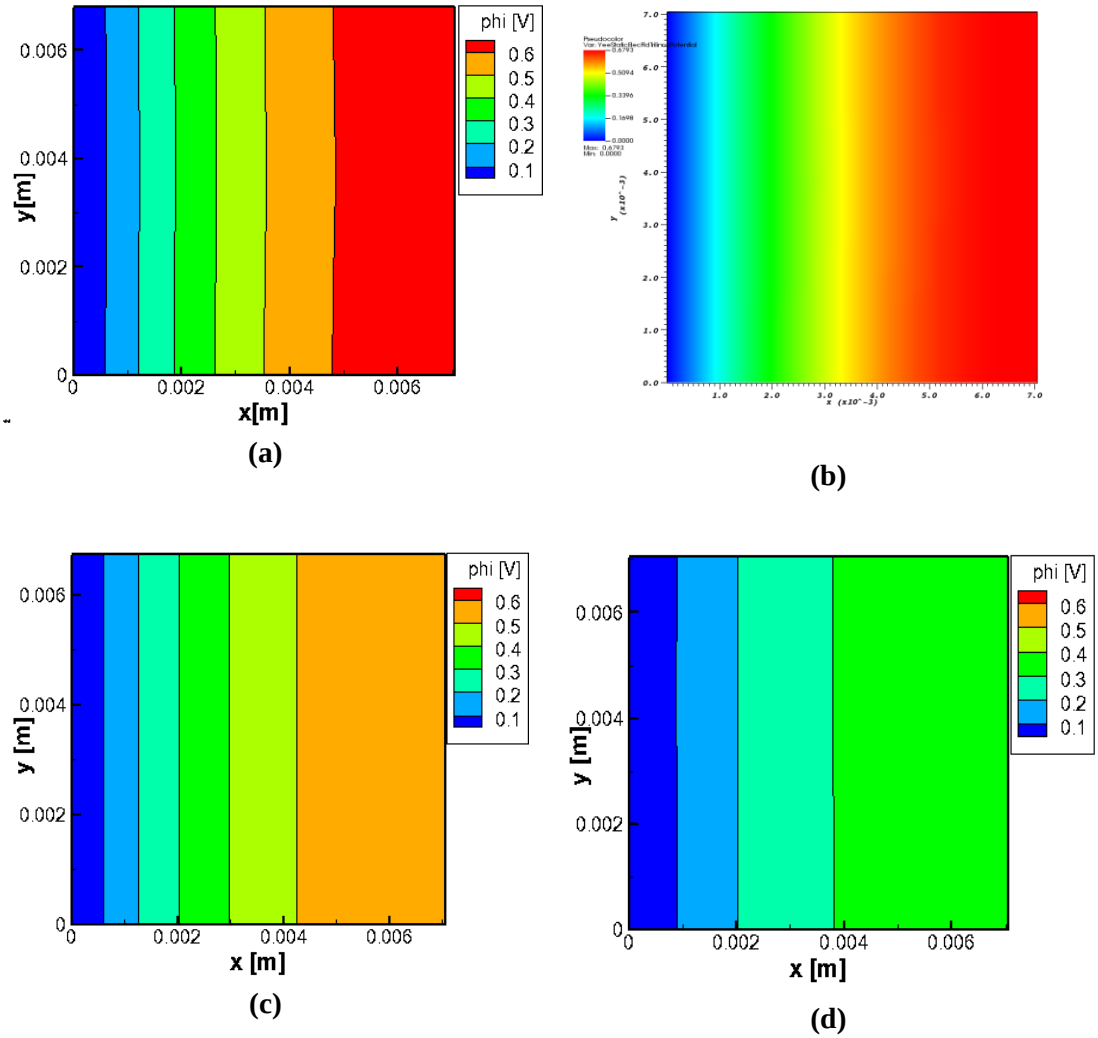


Figure 3-11 ***a)*** Kinetic model SIMPFORT ***b)*** Kinetic model VORPAL ***c)*** Fluid approach 0.78 *eV* ***d)*** Fluid approach 0.25 *eV*

Clearly, the potential of whole domain heavily depends on the electron temperature. In Figure 3-12, it can be observed that the hybrid model using equilibrium electron temperature gives closer results to the kinetic model compared to the results of the hybrid model that uses the initial electron temperature.

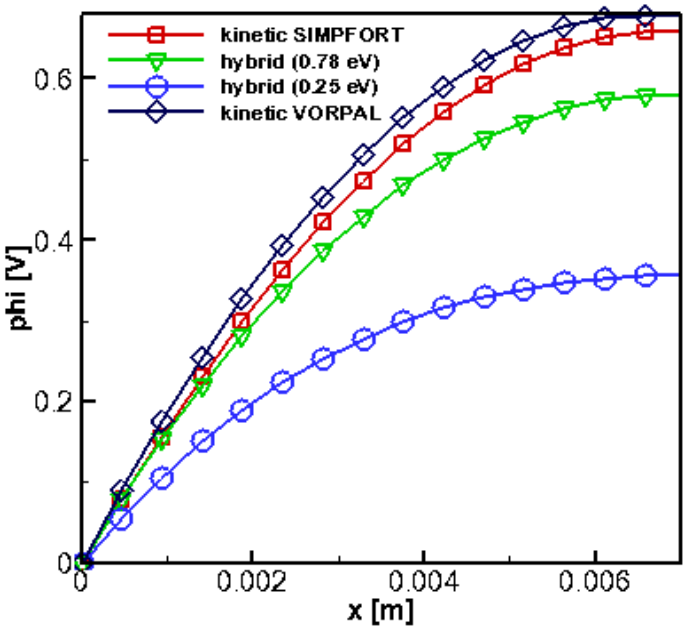
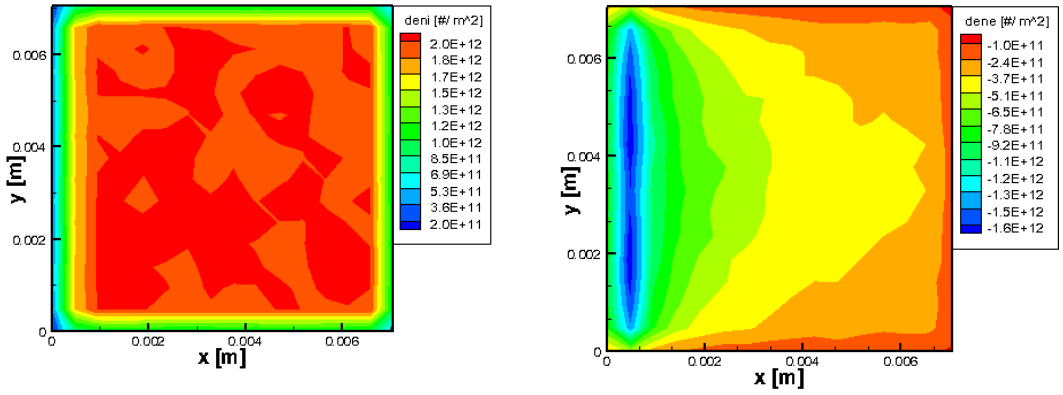


Figure 3-12: Potential variation along x-axis at $y = 0.0033\ [m]\ constant$

In Figure 3-12, the potential values of kinetic models are reasonably close and are somewhat higher than those produced by the hybrid models. This may be partly caused by the numerical heating of the electrons inherent in the kinetic models.

The number density distributions of ions and electrons are shown in Figure 3-13 (a) and (b), respectively. Clearly, the ions are distributed reasonably homogenously because the acceleration term is small to deviate them from their directions. The electron number density, however, decreases along the x-axis. The electrons are exposed to strong electric field near $x = 0$ and are accelerated in the x-direction. Therefore, the number density of electrons decreases in the x-direction.



(a) **(b)**

Figure 3-13: Kinetic (SIMPFORT) results for ***a)*** ion density in, ***b)*** electron density

3.4 Test Problem 2 "The thruster Plume Model"

In this part, a simple collisionless two-dimensional thruster plume is modeled shown in Figure 3-14. In the simulations, collisions among the simulation particles and electromagnetic effects are neglected as opposed to the real case.

In this model, the cell size is fixed and the value is assigned as $\Delta x = \Delta y = 0.4\lambda_d$. The cathode-neutralizer layer is inserted on the sixth cells in the x direction and on top of the fourth cell in y direction and electrons are injected at this layer. Ions are injected at anode layer which is located at the right hand side of the third cells in the x-direction and the first two cells in the y-direction.

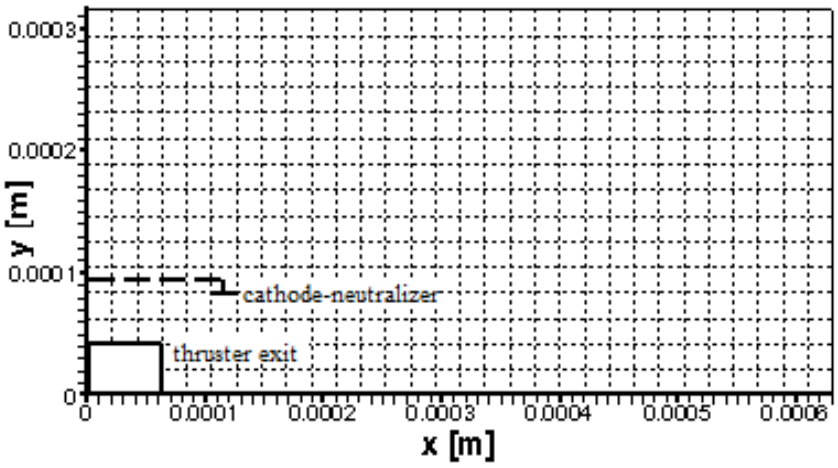


Figure 3-14: Simple Thruster Model

3.4.1 Boundary Conditions

3.4.1.1 Particle boundary conditions

The boundary conditions are shown in Figure 3-15. The particles reaching S1, S2 and R1, R2, R3 are absorbed and are removed from the calculations. The particles which reach R4 are reflected specularly due to symmetry.

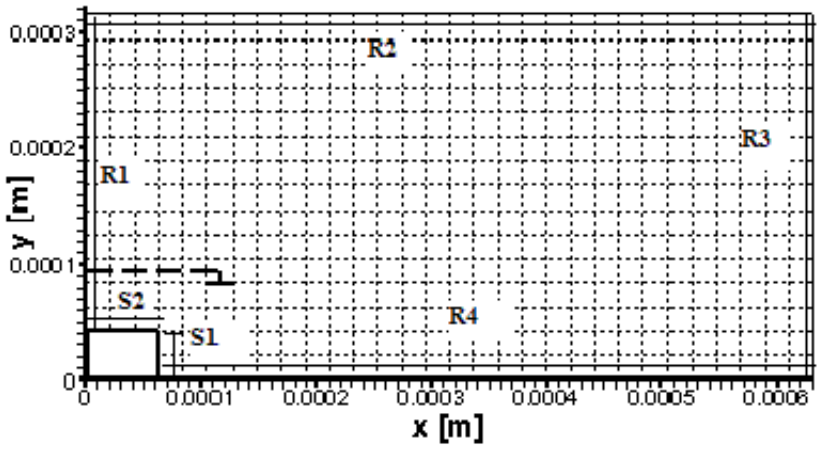


Figure 3-15: Boundary conditions of the thruster

3.4.1.2 Field boundary conditions

In the present calculations, Dirichlet B.C are used in S1 and S2, and Neumann B.C are applied at R1, R2, R3, R4 as follows:

$$s_1 \rightarrow \phi\ const. = 10V, \quad S_2 \rightarrow \phi\ const. = 10V,$$
$$R_1, R_3 \rightarrow \frac{\partial \phi}{\partial x} = 0, \qquad R_2, R_4 \rightarrow \frac{\partial \phi}{\partial y} = 0 \tag{3.8}$$

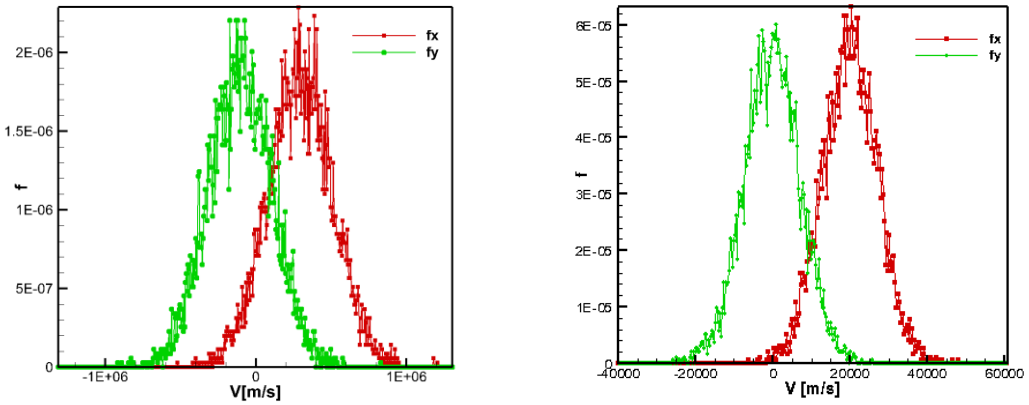
3.4.2 Loaded Particle Properties

In this test problem, ions are injected at the thruster exit whereas the electrons are emitted from the cathode layer. In the present calculations, neutral particles are not injected into the domain. This stems from the fact that collisions are excluded and neutral particles do not interact with electromagnetic fields.

In Table 3-7, the properties of the injected particles are given. The number of computational electrons and ions in the simulation region are assumed to be the same having a value of 6750. The ions and electrons have both thermal speeds and drift velocities. Different distributions are chosen for the x- and y- directions for electrons and ions. These values are chosen consistent with the real thruster data. Note that the distribution functions match with the Gaussian distribution as shown in Figure 3-16. As can be observed in this figure, the peaks of the distribution functions occur at the values of the drift velocities.

Table 3-7: Properties of loaded particles and simulation domain

	Electrons	***Ions***
Grid (N_x ,N_y)	*31 x 16*	*31 x 16*
Domain size (L_x ,L_y)	(0.631*mm* , 0.316 *mm*)	(0.631*mm* , 0.316*mm*)
Cell size	0.021 *mm*	0.021 *mm*
Mass of Particle	9.109e-31 *kg*	2.181e-25 *kg*
Charge of Particle	-1.602e-19 *C*	1.602e-19 *C*
Weight of macroparticles, w	2.e9	2.e10
time step	1.e-12 s	1.e-12 s
Number density	1.e16 #/m^2	1.e16 #/m^2
# of injected particles per time step	15	15
Time step required to reach steady state	22500	22500
Temperature of injected particles	0.5 *eV*	136.15 *eV*
Drift in x direction velocity of injected particles	300000 *m/s*	30000 *m/s*
Drift in y direction velocity of injected particles	-100000 *m/s*	0



(a) **(b)**

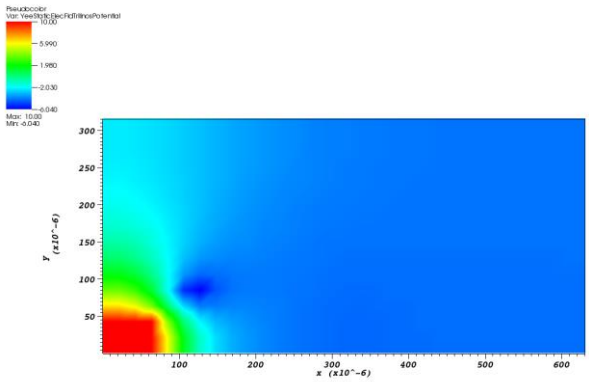
Figure 3-16: Initial distribution function of ***a)*** electrons ***b)*** ions

The time evolution of the computed results can be seen in Figure 3-17 to Figure 3-31. These figures show the contours for the potentials and the electric fields (in both x- and y-directions), and also the positions of the ions and the electrons in the simulation domain. The VORPAL and the SIMPFORT results are labelled as "***(a)***" and "***(b)***" in these figures, respectively. To check the time accuracy of the calculations, an extra run has been performed using half time step in the SIMPFORT code, and its results are presented with labels "***(c)***" in these figures. Clearly, all the results are reasonably close.

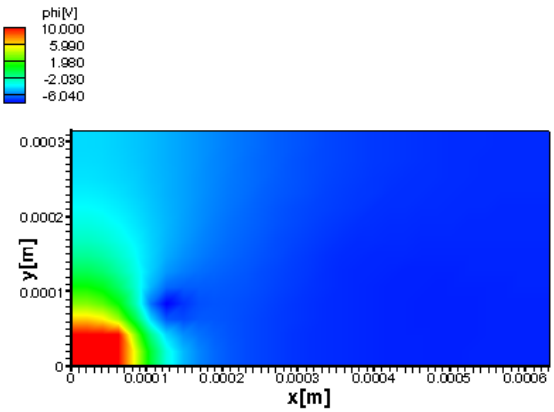
In the present study, for accurate calculations, it is made sure that each computational particle does not travel more than the cell length in each time step, *i.e.*

$$\frac{u\,\Delta t}{\Delta x} < 1\,, \qquad \frac{v\,\Delta t}{\Delta y} < 1 \tag{3.9}$$

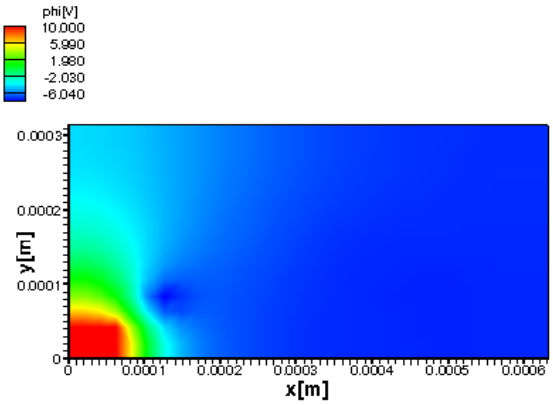
where u and v are speed of particles in the x- and y- directions, respectively.



(a)

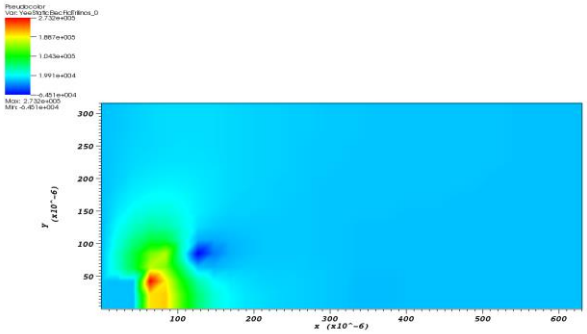


(b)

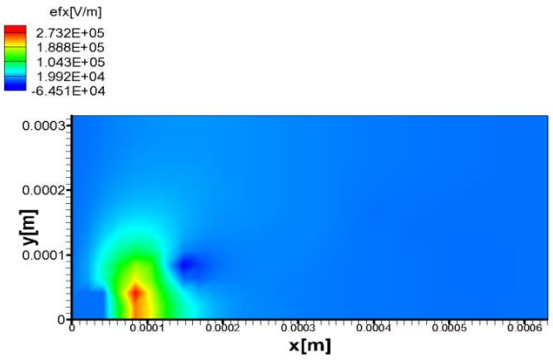


(c)

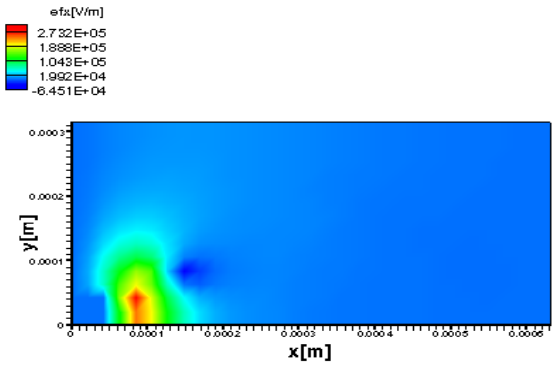
Figure 3-17: Potential contours at 500 time steps ***a)*** VORPAL, ***b)*** SIMFORT ***c)*** SIMPFORT using half Δt



(a)



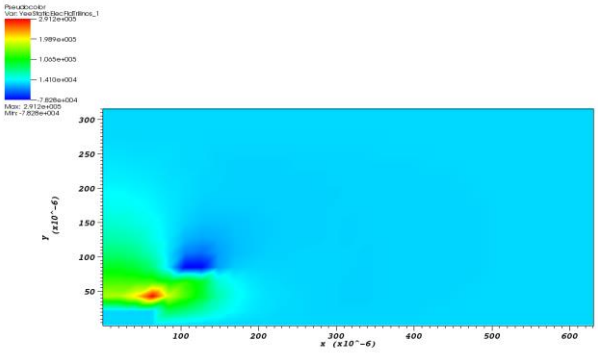
(b)



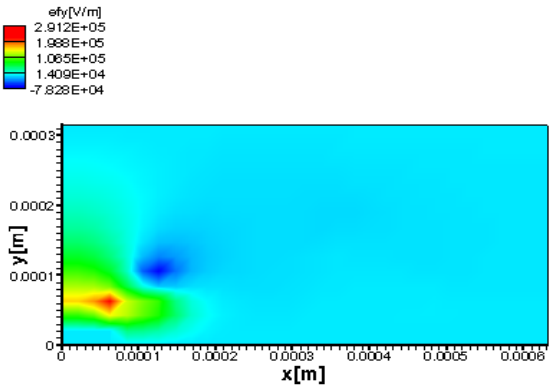
(c)

Figure 3-18: Electric field contours in the x-direction at 500 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt

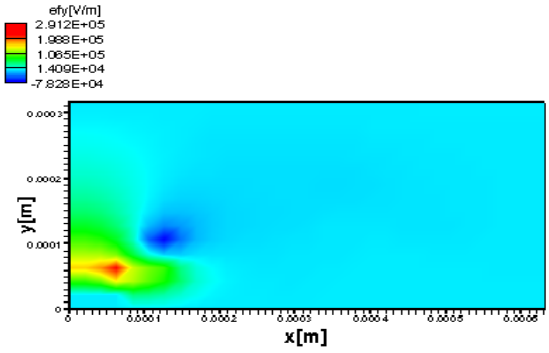
.



(a)

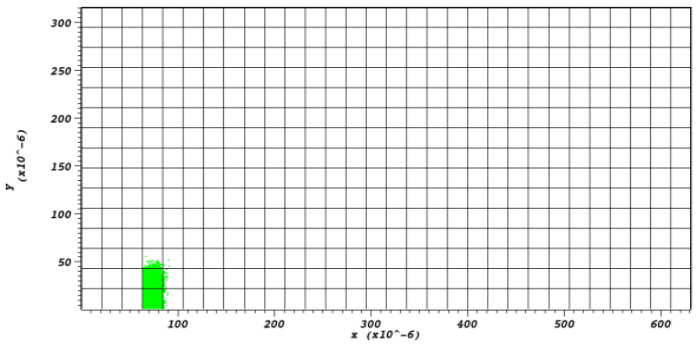


(b)

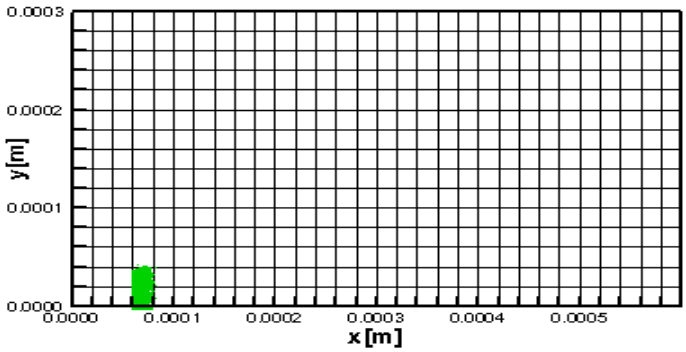


(c)

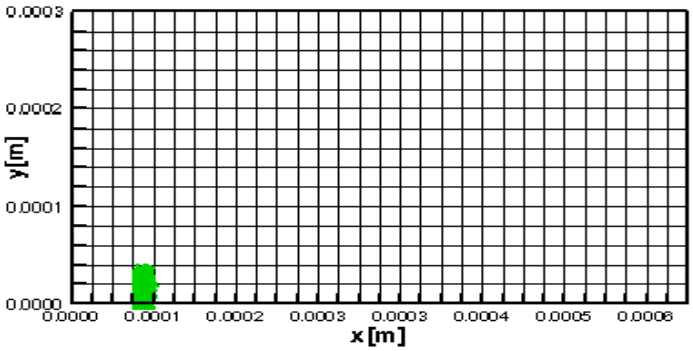
Figure 3-19: Electric field contours in the y-direction at 500 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

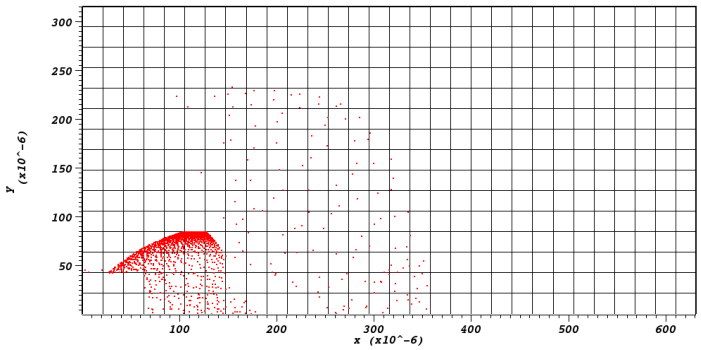


(b)

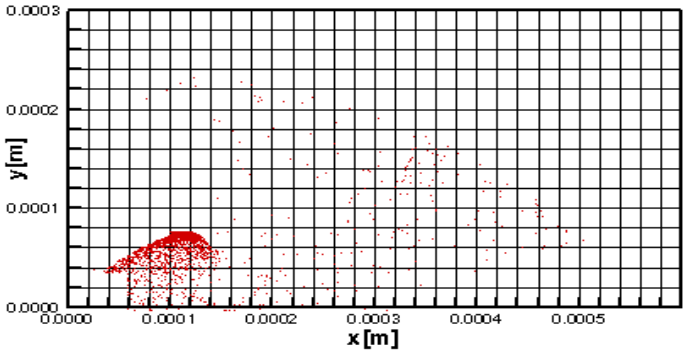


(c)

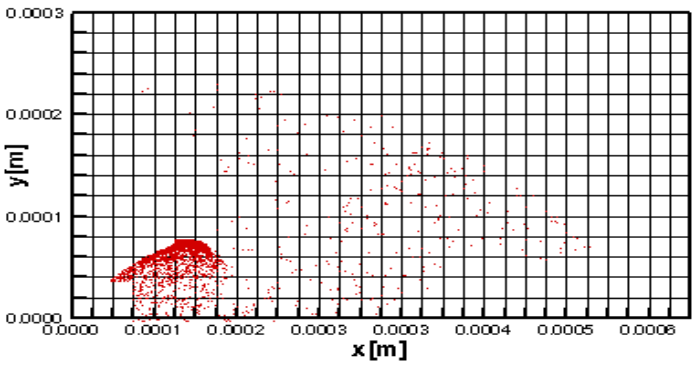
Figure 3-20: Ion positions at 500 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

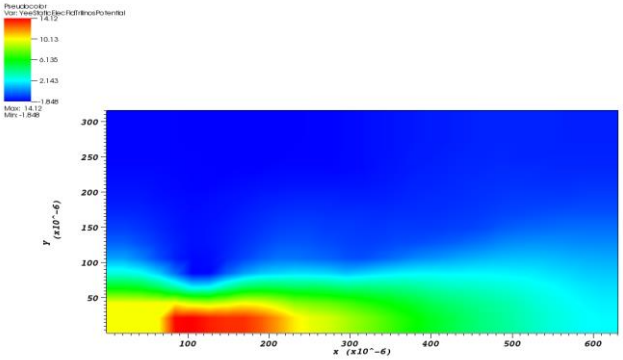


(b)

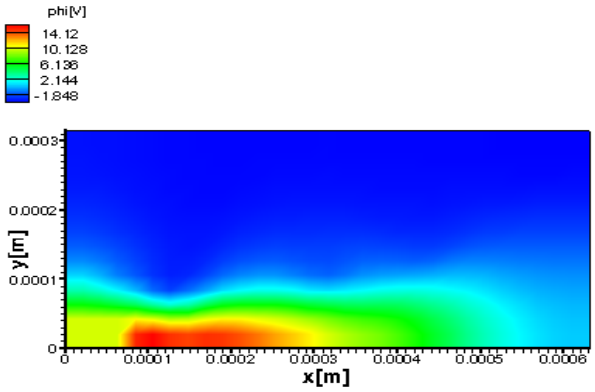


(c)

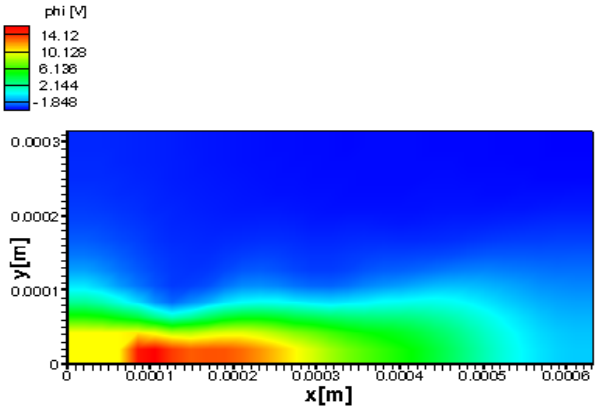
Figure 3-21: Electrons positions at 500 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

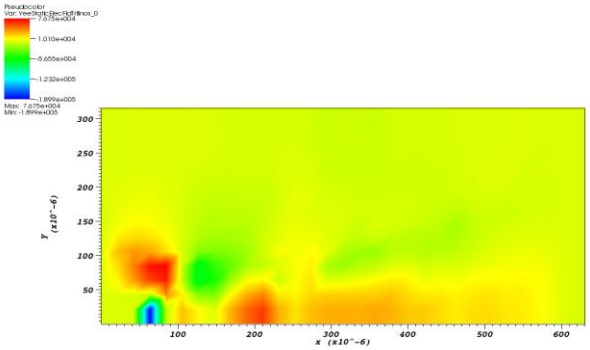


(b)

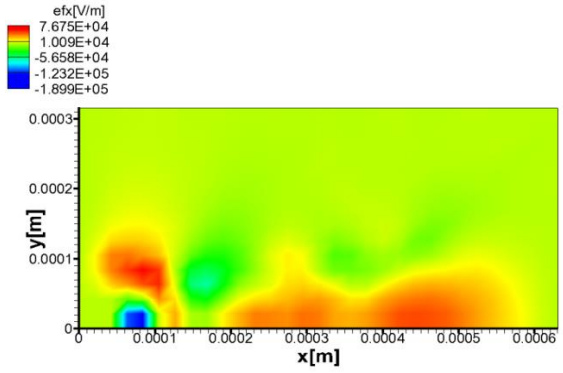


(c)

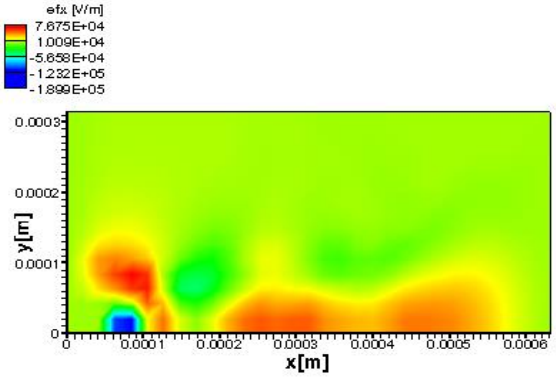
Figure 3-22: Potential contours at 12000 time steps ***a)*** VORPAL, ***b)*** SIMFORT ***c)*** SIMPFORT using half Δt



(a)

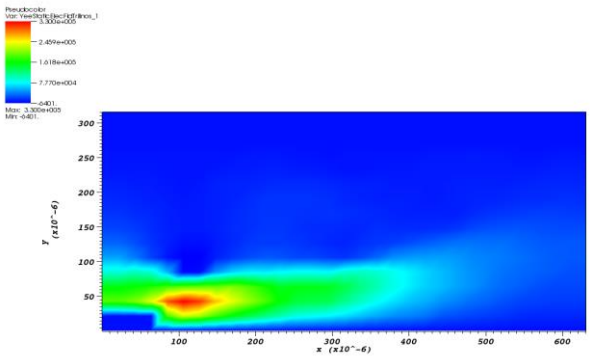


(b)

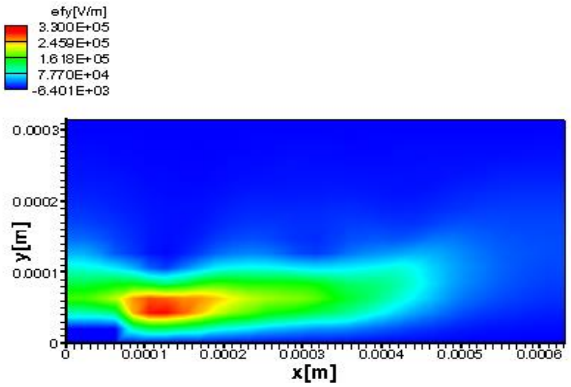


(c)

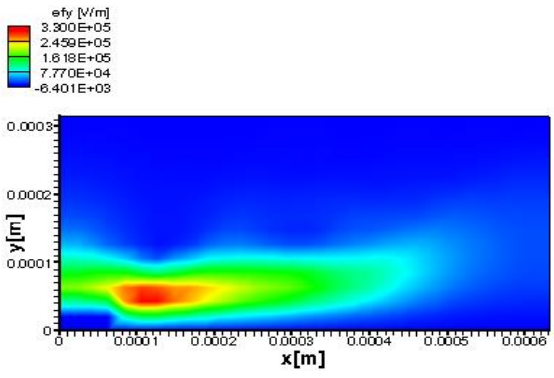
Figure 3-23: Electric field contours in the x-direction at 12000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

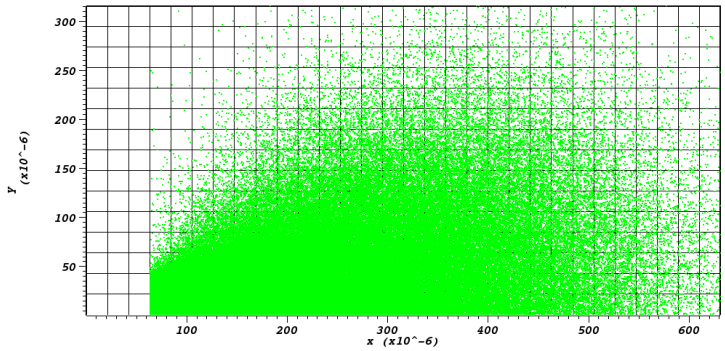


(b)

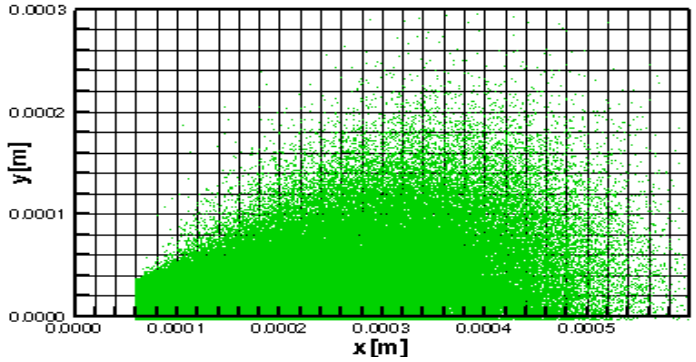


(c)

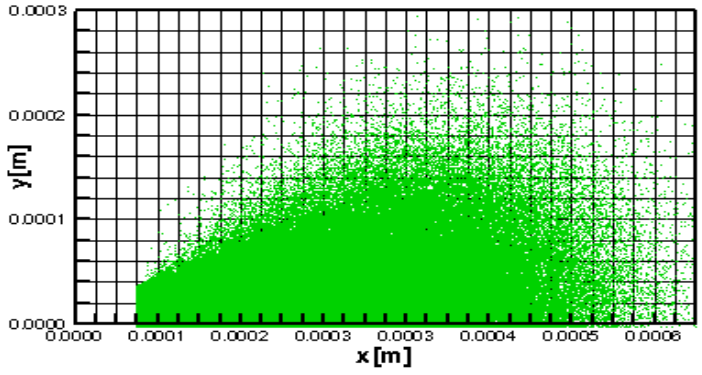
Figure 3-24: Electric field contours in the y-direction at 12000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

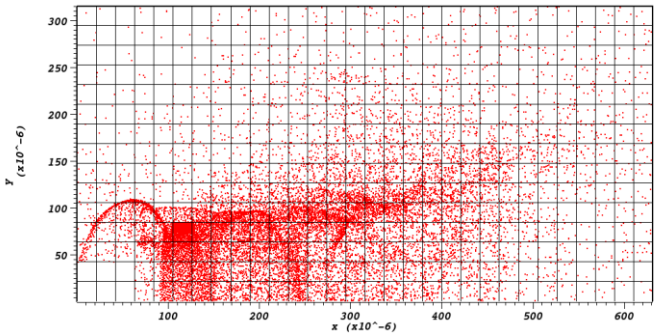


(b)

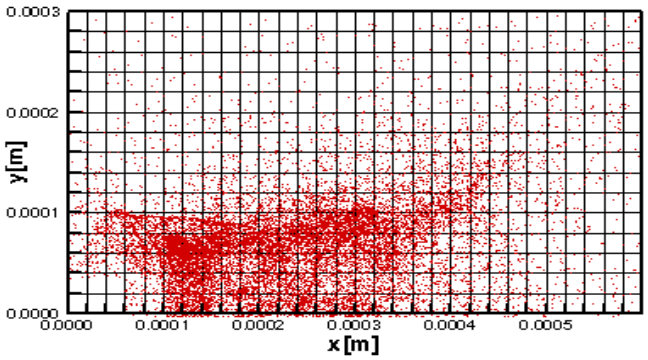


(c)

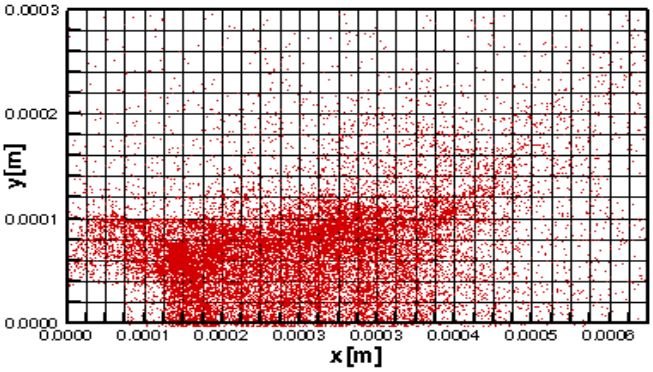
Figure 3-25: Ions positions at 12000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

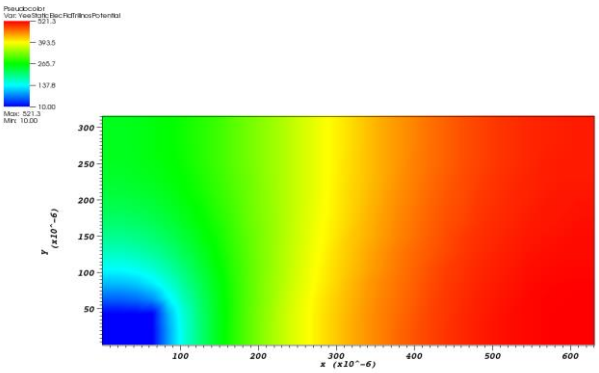


(b)

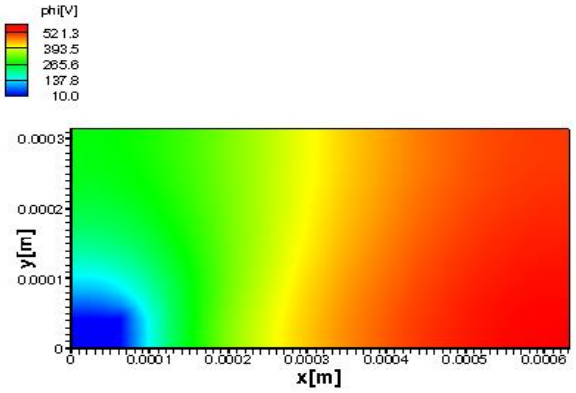


(c)

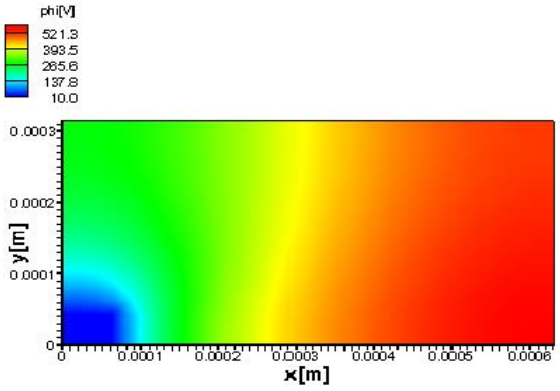
Figure 3-26: Electrons positions at 12000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

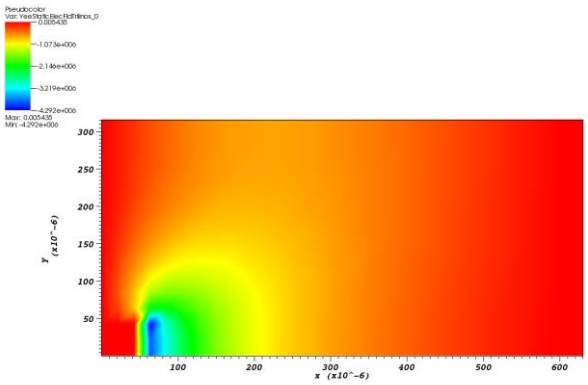


(b)

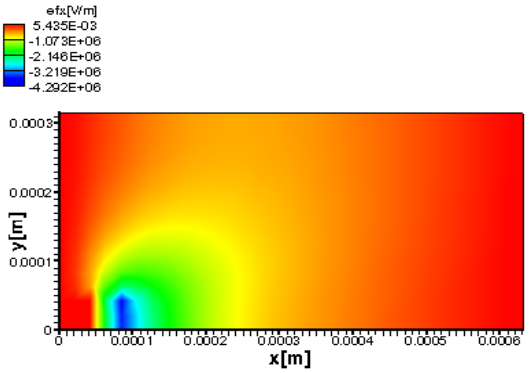


(c)

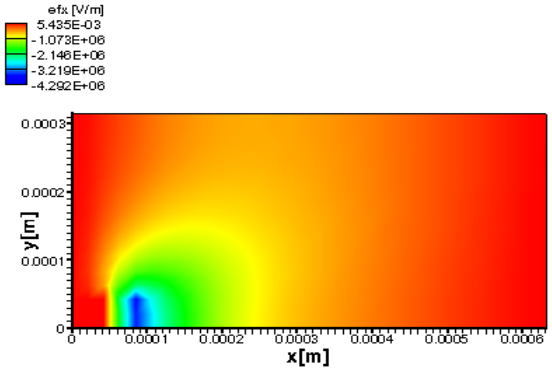
Figure 3-27: Potential contours at 30000 time steps ***a)*** VORPAL, ***b)*** SIMFORT ***c)*** SIMPFORT using half Δt



(a)

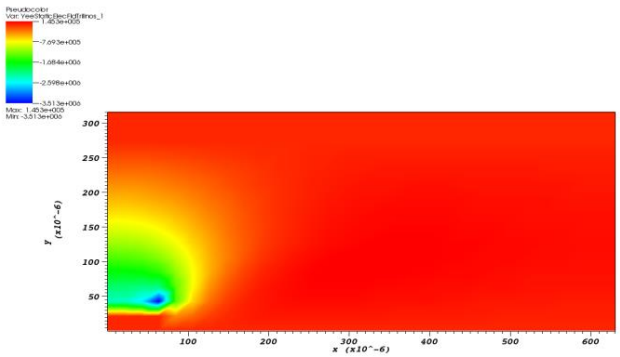


(b)

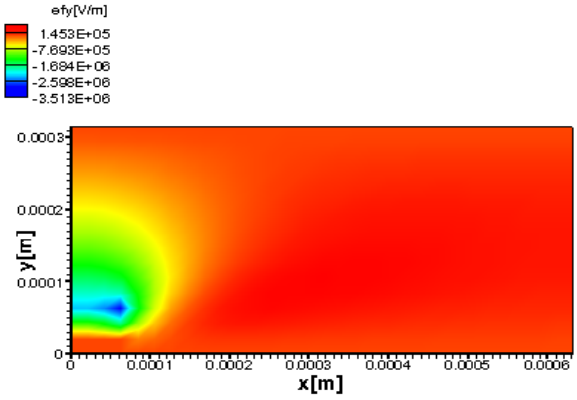


(c)

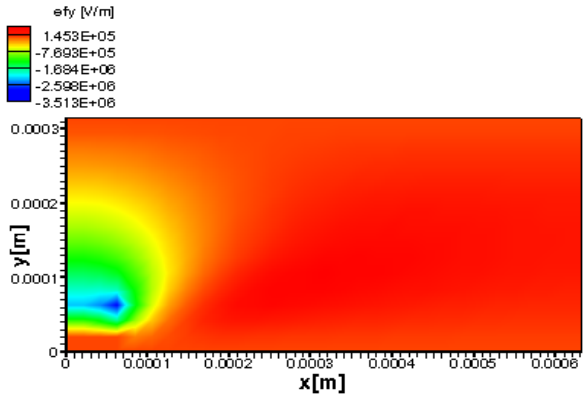
Figure 3-28: Electric field contours in the x-direction at 30000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

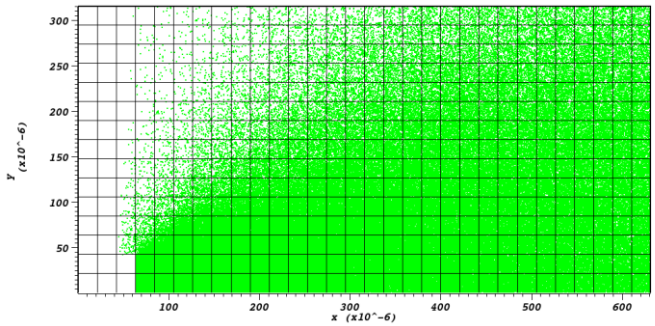


(b)

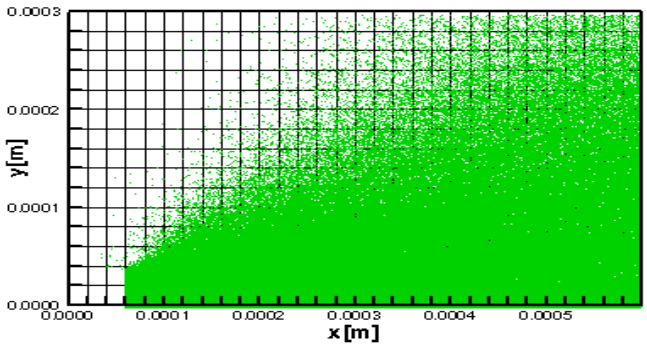


(c)

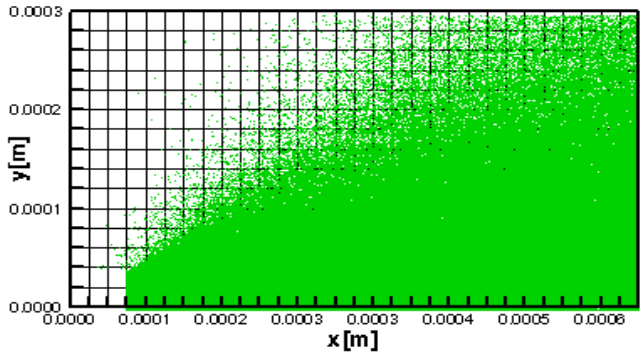
Figure 3-29: Electric field contours in the y-direction at 30000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)

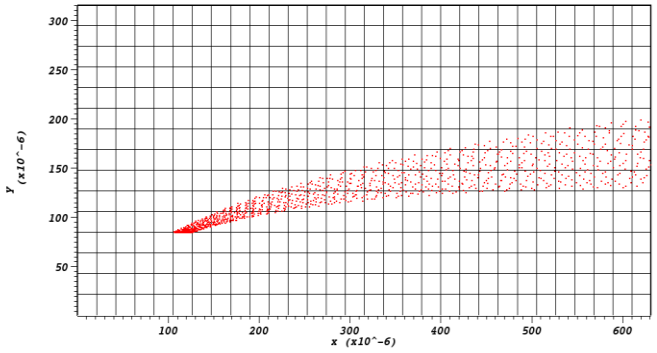


(b)

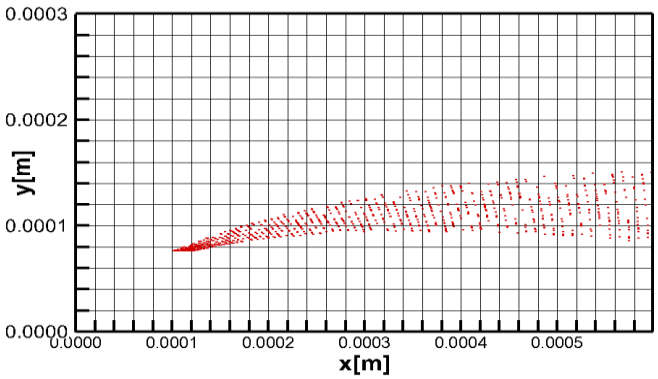


(c)

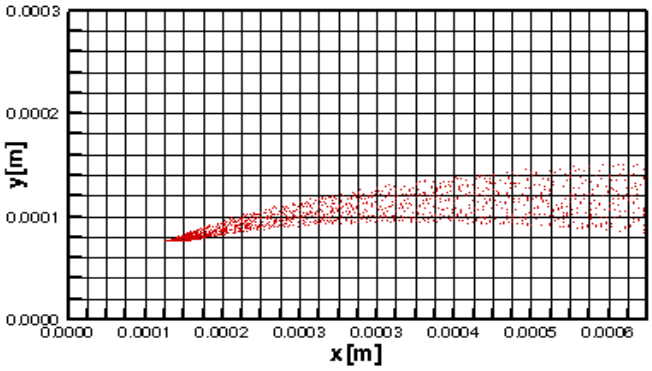
Figure 3-30: Ions positions at 30000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt



(a)



(b)



(c)

Figure 3-31: Electrons positions at 30000 time steps ***a)*** VORPAL, ***b)*** SIMPFORT, ***c)*** SIMPFORT using half Δt

CHAPTER 4

CONCLUSION

4.1 Summaries and Conclusions

In this thesis, a two-dimensional computer code (SIMPFORT) is written to study collisionless plasma flows under the effect of electrostatic forces. The simulations are performed using the Particle-in-Cell (PIC) method. The results are validated by using different plasma simulation approaches and comparing with commercial code, VORPAL.

In the early stages of this study, different types of the Poisson's equation solvers were sought with careful investigation over their accuracies and efficiencies. The most accurate and efficient solver was picked up and embedded into SIMPFORT.

By performing momentum and energy computations, the effects of the spatial and time step resolutions on PIC method were investigated. It was realized that time step is a crucial parameter in reducing unphysical numerical heating.

Subsequently, two different approaches that are extensively used in literature were inserted into SIMPFORT. These are the kinetic and hybrid approaches. In the kinetic approaches, all particles are treated as particle. On the other hand, the hybrid approaches combine the continuum and kinetic models.

Next, two test problems are used to simulate plasma flows. Results of test case 1"expansion into vacuum from a nozzle" are validated by kinetic and hybrid codes, SIMPFORT, and kinetic code, VORPAL. Test case 2 "simple thruster model" are modeled by the two kinetic codes, but results of hybrid model are not close to those of kinetic models due to the fact that electrons are not in the thermal equilibrium state.

4.2 Future Works

SIMPFORT is only capable of doing two-dimensional flow simulations in both rectangular and non-rectangular domains. The code should be developed to perform simulations with complicated geometries in 3-D.

To improve the physical simulation power of the code, the collisions should be taken into consideration. DSMC and MCC methods could be applied to perform collision calculations.

SIMPFORT assumes that the field is electrostatic and only obtains the solution of the Poisson's equation. However, the code should be improved to deal with electromagnetic Lorentz forces as well by solving the full Maxwell's equations.

SIMFORT is a sequential code and to increase computational efficiency, it should be parallelized.

REFERENCES

[1] R. G. Jahn, E. Y. Choueiri, "*Electric Propulsion"*, Encyclopedia of Physical Science and Technology, Third Edition, Volume 5, 2002

[2] G.A Bird, "*Molecular Gas Dynamics and the Direct Simulations of Gas Flows"*, Oxford Press, New York,1994

[3] J.R Wertz and W. J. Larson, "*Space Mission Analysis and Design"*, Torrance, California Press, 1999

[4] U. Schumann, R.A. Sweet, "*Direct Poisson Equation Solver for Potential and Pressure fields on a Staggered grid with Obstacles"*, Computational Phys. 20 (1976), pp.171-182

[5] A.Haji-Sheikh, "*Handbook of Numerical Heat Transfer"*, Wiley 1988, New York, pp. 673-722.

[6] A. Bogaerts, E. Neyts, R. Gijbels, "*Gas Discharge Plasmas and their Applications"*, Spectrochimica Acta Part B 57 (2002) pp. 609–658

[7] D. M. Goebel, I. Katz, "*Fundamentals of Electric Propulsion"*, John Wiley & Sons,2008

[8] F. F. Chen, "*Introduction to Plasma Physics and Controlled Fusion"*, Springer, 2006

[9] C. Yongjun, M. Keidar, I. D Boyd, *"Particle Simulation of Plume Flows From an Anode Layer Hall Thruster"*, AIAA 2006-5026

[10] P. M. Ballen," *Fundamentals of Plasma Physics"*, Cambridge University Press, 2006

[11] D. Tskhakaya, K. Matyash, R. Schneider, and F. Taccogna, "*The Particle-In-Cell Method* ",Contrib. Plasma Phys. 47, No. 8-9(2007), pp. 563 – 594

[12] G. Lapenta, "*Particle In Cell Methods with Apllication to Simulations in Space Weather"* https://perswww.kuleuven.be/~u0052182/pic/book.pdf, as accurate of June, 2013

[13] L. Brieda, *Development of the DRACO ES-PIC Code and Fully-Kinetic Simulation of Ion Beam Neutralization*, MS thesis, Virginia Polytechnic Institute and State University, 2005

[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, "*Numerical Recipes in Fortran 77"*, Cambridge University Press, 2nd Edition, 1992 , pp. 858-860

[15] Y. Choi, "*Modeling an Anode Layer Hall thruster and its Plume"*, Phd thesis, The University of Michigan, 2008

[16] Birdsall, C., and Langdon, A., "*Plasma Physics Via Computer Simulations"*, McGraw-Hill, New York, 1985

[17] N. Sengil, O. Tumuklu, M.C. Celenligil, "*Implementation of a Monte Carlo Method to a two-dimensional Particle-in-Cell solver using algebraic meshe"s,* Nukleonika 57 (2) (2012), pp. 313-316

[18] U. N. Grigoryev, V. A. Vshivkov,M. P. Fedoruk,"*Numerical Particle in Cell Methods",* Netherlands Ridderprint Press, 2002

[19] W. G. Vincenti, C. H. Kruger," *Intoduction to Physical Gas Dynamics",* John Wiley & Sons, Inc.1965

[20] G. A. Bird, "*Molecular Gas Dynamics",* Clarendon Press (Oxford), 1994

[21] N. Matthew O. Sadiku, "Monte Carlo Methods for Electromagnetics", CRC Press, 2009

[22] V.Vahedi, G.Dipeso, *"Simultaneous Potential and Circuit Solution for Two-Dimensional Bounded Plasma Simulation Codes",* Computational Phys.131(1997) pp. 149-163

[23] Boyd I.D., "*Computational of the Plume of an Anode-Layer Hall Thruster",* Journal of Propulsion and Power , Vol.16, No.5, 2000, pp. 902-909

[24] A. M. Spirkin, "*A 3D PIC Methology on Unstructured Voroni Grids with Applications to Plasma Microdevices",* Phd thesis, Worcester Polytechnic Institute, 2006

[25] J. Brackbill, D. Forslund, "*Simulation of Low-Frequency Electromagnetic Phenomena in Plasmas",* in: J. U. Brackbill, B. Cohen (Eds.), Multiple Time Scales, Academic Press Orlando, 1985

[26] A.B. Langdon, *"Energy-Conserving Plasma Simulation Algorithms",* Journal of Computational Physics 12 (2) (1973) 247 – 268

[27] R. J. Goldston and P. H. Rutherford, "*Introduction to Plasma Physics*", Institute of Physics Publishing, Bristol and Philadelphia, 1995

[28] *"Ion propulsion",* NASA Facts, FS-2004-11-021- GRC

[29] S. D. Grishin and L. V. Leskov, "*Electrical Rocket Engines of Space Vehicles",* Moscow, Russia: Mashinostroyeniye Publishing House , 1989

[30] J. T. Yim, "*Computational Modelling of Hall Thruster Channel Wall Erosion",* Phd Dissertation, The University of Michigan, 2008

[31] E. Y. Choueiri, "*Fundemental Difference between the Two Variants of Hall Thruster: SPT and TAL",* AIAA-2001-3504.

[32] Hafez M., Parlette E., Salas M. "Convergence acceleration of iterative solutions of Euler equations for transonic flow computations" Computational Mechanics (1986) 1, pp. 165-176