

**A Three-dimensional Particle-in-Cell Methodology on Unstructured Voronoi Grids
with Applications to Plasma Microdevices**

by

Anton M. Spirkin

A Dissertation

Submitted to the Faculty of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirement for the degree of

Doctor of Philosophy

in

Mechanical Engineering

By

April 2006

APPROVED:

Dr. Nikolaos A. Gatsonis, Advisor
Professor, Mechanical Engineering Department

Dr. Gretar Tryggvason, Committee Member
Professor, Mechanical Engineering Department

Dr. John J. Blandino, Committee Member
Assistant Professor, Mechanical Engineering Department

Dr. George E. Karniadakis, Committee Member
Professor, Applied Mathematics Department, Brown University

Dr. Mark W. Richman, Graduate Committee Representative
Associate Professor, Mechanical Engineering Department

ABSTRACT

The development and numerical implementation of a three-dimensional Particle-In-Cell (3D PIC) methodology on unstructured Voronoi-Delauney tetrahedral grids is presented. Charge assignment and field interpolation weighting schemes of zero- and first-order are formulated based on the theory of long-range constraints for three-dimensional unstructured grids. The algorithms for particle motion, particle tracing, particle injection, and loading are discussed. Solution to Poisson's equation is based on a finite-volume formulation that takes advantage of the Voronoi-Delauney dual. The 3D PIC methodology and code are validated by application to the problem of current collection by cylindrical Langmuir probes in stationary and moving collisionless plasmas. Numerical results are compared favorably with previous numerical and analytical solutions for a wide range of probe radius to Debye length ratios, probe potentials, and electron to ion temperature ratios. A methodology for evaluation of the heating, slowing-down and deflection times in 3D PIC simulations is presented. An extensive parametric evaluation is performed and the effects of the number of computational particles per cell, the ratio of cell-edge to Debye length, and timestep are investigated. The unstructured 3D PIC code is applied to the simulation of Field Emission Array (FEA) cathodes. Electron injection conditions are obtained from a Field Emission microtip model and the simulation domain includes the FEA cathode and anode. Currents collected by the electrodes are compared to theoretical values. Simulations show the formation of the virtual cathode and three-dimensional effects under certain injection conditions. The unstructured 3D PIC code is also applied to the simulation of a micro-Retarding Potential Analyzer. For simple cases the current at the collector plate is compared favorably with theoretical predictions. The simulations show the complex structure of the potential inside the segmented microchannel, the phase space of plasma species and the space-charge effects not captured by the theory.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my academic advisor Professor Nikolaos A. Gatsonis who introduced me to the field of computational plasma physics. Thank you for your help, support, commitment and inspiration. Your patience and confidence helped me to really appreciate my work. I would also like to acknowledge the members of my committee for their insightful comments and suggestions.

My sincere thanks are addressed to Dr. Alexander Fedorov who gave me a great career opportunity. Your friendship and your advices were very helpful and supportive.

To all the guys in the lab including Jurg, Ryan, James, David, Raffaele, Jeff, Andrew and Tom, thanks for the friendship. You made the time I spend in the CGPL very interesting and unforgettable.

Special thanks are due to Barbara Edilberti for her help and attitude.

I want to express my greatest appreciation to my mother for her care and to my father for his priceless advices. Thank you for your love and support (both moral and financial). Thank you for showing me a right direction in life. I would never be able to do it without you.

Support from AFOSR's Computational Mathematics Program is greatly appreciated.

TABLE OF CONTENTS

ABSTRACT.....	i
ACKNOWLEDGEMENTS.....	ii
TABLE OF CONTENTS.....	iii
LIST OF FIGURES	v
CHAPTER 1 INTRODUCTION.....	2
1.1 Research Objectives and Approach.....	8
CHAPTER 2 A 3-D PARTICLE-IN-CELL METHOD ON UNSTRUCTURED VORONOI-DELAUNAY GRIDS.....	11
2.1 Mathematical Description of Collisionless Electrostatic Plasma.....	11
2.2 Unstructured 3-D PIC Methodology	13
2.3 Particle In Cell Computational Cycle.....	13
2.4 Finite Size Particles.....	13
2.5 Unstructured Delaunay Grid Generation.....	15
2.6 Finite Sized Particles on a Grid Particle Weighting on to Grid (Gather).....	17
2.7 Solution of Poisson’s Equation and Electric Potential Evaluation. Finite Volume Formulation.	35
2.8 Electric Field Evaluation.....	39
2.9 Integration of the Equations for Particle Motion	42
2.10 Particle Search-Locate Algorithm (Particle Tracer).....	46
2.11 Electrostatic Boundary Conditions.....	49
2.12 Particle Loading	53
2.13 Particle Injection	55
2.14 Macroscopic Plasma Properties Evaluation.....	58
CHAPTER 3 SIMULATION OF CYLINDRICAL LANGMUIR PROBE IN COLLISIONLESS PLASMAS.....	61
3.1 Review of the Current Collection Theory	61
3.2 Unstructured 3-D PIC Simulations of the Current Collection by a Cylindrical Probes.....	65

CHAPTER 4 HEATING, SLOWING-DOWN AND DEFLECTION TIMES IN 3-D ELECTROSTATIC PIC SIMULATIONS ON UNSTRUCTURED TETRAHEDRAL GRIDS	74
4.1 Overview	74
4.2 Heating, Slowing-Down and Deflection Times in a Collisionless Plasma	77
4.3 Heating, Slowing-Down and Deflection Times in 3-D PIC Simulations	81
4.4 Heating Time.....	85
4.5 Slowing-Down and Deflection Times.....	92
4.6 Conclusions	94
CHAPTER 5 SIMULATIONS OF PLASMA DEVICES AND MICROSENSORS	96
5.1 Simulations of 3-D Space-Charge Effects in Gated Field Emission Array Cathodes	96
5.1.1 Model for Field Emission Array Cathode.....	98
5.1.2 Model for Space-Charge Effects.....	101
5.1.3 PIC Simulations of FEA	103
5.2 3-D PIC Simulations of a Directional Micro Retarding Potential Analyzer..	111
5.2.1 Simulations of the D μ RPA microchannel without the microchannel plate	112
5.2.2 Simulations of the D μ RPA microchannel with the microchannel plate.....	120
CHAPTER 6 CONCLUSIONS AND RECOMMENDATIONS	123
6.1 Summary and Accomplishments.....	123
6.2 Recommendations for Future Work.....	126
REFERENCES	129

LIST OF FIGURES

Figure 1. Particle-In-Cell computational cycle.....	14
Figure 2. Linear weighting functions in 1-D.	21
Figure 3. Example of the 2-D structured Cartesian grid with the computational particle located at (x_p, y_p)	23
Figure 4. Graphical representation of the Linear weighting on the 2-D structured Cartesian grid.	25
Figure 5. Example of the Delaunay triangular computational cell.	27
Figure 6. Graphical representation of the Linear weighting on a 2-D Delaunay grid.	29
Figure 7. Example of the 2-D Voronoi cell.	30
Figure 8. Example of the tetrahedral computational cell.....	31
Figure 9. Graphical representation of the CIC weighting on the 3-D unstructured tetrahedral grid.	34
Figure 10. Example of the 2-D unstructured grid and Voronoi volume associated with the node I	37
Figure 11. Example of the Delaunay mesh and the Voronoi dual in the 2-D case.	38
Figure 12. A control volume used in the node-centered method of the electric field evaluation.	40
Figure 13. The leap-frog integration scheme.....	43
Figure 14. Tetrahedral computational cell and the possible position of the particle at time $t + \Delta t$	47
Figure 15. Graphical representation of the Dirichlet and Neumann boundary conditions (from <i>Hammel (2002)</i>).	49
Figure 16. Coordinate system utilized in the particle injection algorithm.....	55
Figure 17. An example of the computational domain used in the simulations (Thin Sheath regime).	67
Figure 18. An example of the computational domain used in the simulations (OML regime).	67

Figure 19. Potential distribution around a cylindrical probe with $e\Phi_p / kT_e = 5$.	
Simulations parameters are those of Case 1.1.....	68
Figure 20. Electron and ion currents collected by a cylindrical probe with $e\Phi_p / kT_e = 5$.	
Simulation parameters are those of Case 1.1.	69
Figure 21. Influence of the ion to electron temperature ratio on current collection by cylindrical probe. Computational parameters are those of Case1.1 and Case 1.2.	70
Figure 22. Influence of the Debye length on the current collection. Computational parameters ARE those of Case2.1 and Case 2.2.	71
Figure 23. Electron and ion currents measured at $e\Phi_p / kT_e = 5$. Simulations parameters those of Case 3.	71
Figure 24. Non-dimensional electron current as a function of undimensional potential in the OML regime (Case 3).....	72
Figure 25. Electron and ion currents collected by a cylindrical probe as a function of the ion speed ratio.	73
Figure 26. Orbit of the computational particle.....	82
Figure 27. A typical computational domain.	86
Figure 28. Tetrahedral computational cell.	87
Figure 29. Distribution of cell volumes in a computational domain.	87
Figure 30. Distribution of the dihedral angles in a computational domain.	88
Figure 31. Distribution of cell-edge length in a computational domain.	88
Figure 32. Time evolution of the total kinetic energy.	90
Figure 33. Heating time as the function of the timestep, for 20 particles per cell and $\Delta r = \lambda_D$	90
Figure 34. Heating time as a function of the number of particles per cell for $w_p \Delta t = 5 \times 10^{-3}$ and $\Delta r = \lambda_D$	91
Figure 35. Heating time as a function of $\lambda_D / \Delta r$ for $w_p \Delta t = 5 \times 10^{-3}$ and 20 particles per cell.	92
Figure 36. Slowing-down time vs. number of particles per computational cell.	93
Figure 37. Deflection time as the function of the number of particles per cell.	95
Figure 38. FE cathode microtip.	99

Figure 39. Geometry of the space-charge effects model.	102
Figure 40. Fowler-Nordheim current density and tip current as a function of gate potential.....	104
Figure 41. FE array current and current density as a function of gate potential.....	104
Figure 42. Velocity and density of the emitted electrons as a function of gate potential.	105
Figure 43. Simulation domain used in FE cathode simulations.....	105
Figure 44. Anode, gate, analytical maximum and FE array currents as a function of gate potential for $\Phi_{anode} = \Phi_g + 100$ [V].	106
Figure 45. Anode, gate, analytical maximum and FE array currents as a function of gate potential for	107
Figure 46. Schematics of FE array.....	108
Figure 47. Potential distyribution between gate cathode and anode.....	109
Figure 48. Phase space of the emitted electrons.	110
Figure 49. μ RPA schematic (from <i>Partridge et al.</i> (2003)).	112
Figure 50. D μ PRA microchannel computational domain.	112
Figure 51. Collector plate current vs. ion retarding potential.	115
Figure 52. Energy distribution of the ions collected in Case 1 for the computational time 4.15E-8 sec.	116
Figure 53. Side view of the RPA simulation domain, potential distribution and \mathbf{c}_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 0$ V ; Ion (red) and Electron (blue).	117
Figure 54. Side view of the RPA simulation domain, potential distribution and \mathbf{c}_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 50$ V ; Ion (red) and Electron (blue).	118
Figure 55. Side view of the RPA simulation domain, potential distribution and \mathbf{c}_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 80$ V ; Ion (red) and Electron (blue).	119
Figure 56. μ RPA microchannel plate schematic.	120
Figure 57. A part of the computational grid representing the microchannel plate.	121

Figure 58. Collector plate ion current vs. ion retarding potential for the D μ RPA with microchannel plate simulations..... 122

LIST OF TABLES

Table 1. Input conditions and computational parameters for the current collection simulations.	66
Table 2. Molybdenum FE cathode parameters.	103
Table 3. Input conditions and computational parameters for the D μ RPA simulations. .	113
Table 4. Simulation parameters and results of the D μ RPA simulations.	113
Table 5. Input conditions for the simulations of D μ RPA with microchannel plate.	121
Table 6. D μ RPA with microchannel plate simulation results.	122

NOMENCLATURE

A	area	S_i	ion speed ratio
\mathbf{B}_{ext}	external magnetic field vector	s	pertaining to species s
b	impact parameter	T	temperature
\mathbf{C}	peculiar velocity vector	t	time
\mathbf{c}	total velocity vector	\mathbf{u}	drift velocity vector
d_s	sheath thickness	V	volume
e	electron charge	W	weight function
\mathbf{E}	electric field vector	Z	number of electron charges
\mathbf{F}	force vector	β	inverse of most probable velocity
F	computational particle weight	Γ	Voronoi cell volume
f	distribution function, face number	δ	delta function
G	Green's function	ϵ_0	permittivity of free space
I	current, node number	θ	rotation angle, effective temperature
i	particle number	λ	collisional mean free path
J	current density, node number	λ_D	Debye length
Kn	Knudsen number	μ_0	permeability of free space
k	Boltzmann constant	ν	collision frequency
L	distance, length	ρ	charge density
l_p	probe length	σ	surface charge density
m	mass	τ_H	heating time
N_s	number of particles of species s	τ_s	slowing-down time
\dot{N}	number flux	Φ	electric potential
\mathbf{n}	unit normal vector	Φ_p	probe potential
n	number density, time-step counter	χ	non-dimensional potential
p_f	pressure at face f	Ω	computational cell volume
P	probability	w_p	plasma frequency
Q	total charge	$\{c_1, c_2, c_3\}$	total velocity components
q	charge	$\{C_1, C_2, C_3\}$	peculiar velocity components
\mathbf{r}	position vector	$\{u_1, u_2, u_3\}$	drift velocity components
r	radius		
S	shape factor, surface number		

CHAPTER 1 INTRODUCTION

This work is motivated by the need to model bounded plasmas and in particular microdevices and sensors with complex geometries. The goal of this work is to establish a validated 3-D electrostatic particle-in-cell (PIC) methodology on unstructured Voronoi-Delauney grids. The unstructured nature of the spatial discretization allows modeling of plasma devices with surfaces of complex geometries. While the main effort is directed towards the PIC simulation of collisionless plasmas the methods under development can be integrated with a Monte-Carlo approach and deliver a PIC/Monte Carlo methodology for collisional plasmas.

The origin of the PIC or Particle-Mesh (PM) method used in the simulation of collisionless plasmas can be traced to the early work of *Buneman* (1959) and *Dawson* (1960). In these first approaches the space-charge forces were obtained through direct evaluation of Coulomb's law and charges were treated as sheets. Subsequently, the numerical implementation of the velocity distribution function provided the possibility for the simulations of the warm plasmas. By assigning three velocity components to all the computational particles including ions, a one dimensional in space and three dimensional in velocity (1D3V) plasma simulation methodology was developed. This model was successfully applied to the simulation of basic unbounded plasma problems. For example, *Dawson* (1964) demonstrated the effects of Landau damping of the electrostatic wave, which was predicted theoretically but was not yet observed experimentally.

Substantial computational gains in particle simulations of plasmas were harnessed with the introduction of the computational grid and finite-size particles. The simulation

domain was divided into uniform computational cells that were used to trace the charged particles. The charge density was obtained on the nodes of the computational grid by interpolating the charge from the positions of the particles. The solution of the electric and magnetic field equations was implemented numerically providing fast evaluation of the electric fields. *Langdon* (1970) introduced finite-size particles in order to reduce the large inter-particle forces that are associated with collisions while retaining long-range forces that give rise to the collective motion of the computational particles. *Langdon, et al.* (1983) introduced an implicit differencing scheme that allowed the increase of overall time scales for plasma simulations. In the 1980's, the first simulations of real plasma devices were performed with the introduction of external circuits in the boundary conditions *Lawson* (1989). Algorithms for charge-charge and neutral-charge collisions were introduced in PIC to develop powerful PIC/MC and PIC/DSMC approaches (*Birdsall and Langdon* (1991), *Nanbu and Yonemura* (1998)).

Various implementation of PIC have been developed on one, two and three-dimensional structured and mostly uniform grids. The references in *Birdsall and Langdon* (1991), *Hockney and Eastwood* (1988), *Dawson* (1983) show the depth as well as the breath of PIC application. The standardized electrostatic PIC cycle that is commonly accepted can be summarized as follows. In the beginning of a PIC simulation computational particles are loaded into the domain with a specified density and velocity distribution. Each computational particle corresponds to a large number of real charged plasma particles and thus requires the introduction of the particle weight. Charge is accumulated from the position of the particles on the discrete mesh nodes, via the charge assignment step (gather). Then fields are obtained on the nodes of the grid using

Poisson's equation. Electric fields are interpolated from the nodes back to the particles (scatter). The equations of motion are then integrated (particle push) and particles positions and velocities are updated every timestep. In the case of bounded plasmas, boundary conditions that may include circuit equations are integrated into the methodology with the evaluation of surface charges and currents.

Most PIC implementations take advantage of structured uniform and non-uniform grids. This allows for an easy implementation of numerical algorithms such as high order weighting schemes, fast particle movers and tracers and others. Domain decomposition of the structured grid is straightforward which is important in the design of the PIC codes on the parallel platforms (*Ferraro, Liewer, et al., (1993); Qiang, Ryne, et al. (2000)*). As the following review shows there have been very few attempts to implement the PIC methodology on unstructured grids. Challenges relate with all four steps of PIC: gather/scatter, particle tracing and pushing, field evaluation, and formal evaluation of errors and uncertainty. *Celik et al. (2003)* developed a 3-D PIC-DSMC code "AQUILA" on unstructured tetrahedral grids as an extension of the code developed by *Fife et al. (2002)*. The first-order weighting scheme based on the volume coordinates is used for the scatter/gather procedures. The particle mover uses a leapfrog algorithm. In order to define a location of the particle, a search is performed by calculating a volume based shape functions at the particle's new location. The electric potential Φ is obtained by assuming quasi-neutrality in the computational domain and inverting Boltzmann's equation

$$\Phi = \Phi_0 + \frac{kT_e}{e} \ln\left(\frac{n_e}{n_{e0}}\right) \quad (1.1.1)$$

where T_e is an electron temperature and n_e is an electron density obtained on the grid nodes. Another method of calculating potential takes advantage of a polytropic relation between T_e and n_e

$$\frac{T_e}{T_{e0}} = \left(\frac{n_e}{n_{e0}} \right)^{n-1} \quad (1.1.2)$$

where n is between the isentropic value of $5/3$ and isothermal value of 1 . Now the potential can be evaluated as

$$\Phi = \Phi_0 - \frac{kT_{e0}}{e} \frac{1 - \left(\frac{n_e}{n_{e0}} \right)^{n-1}}{n-1} \quad (1.1.3)$$

Both formulations assume quasineutrality but equation (1.1.3) allows for a variation in the electron temperature. In “AQUILA” momentum-exchange and charge-exchange collisions are modeled with the no-time-counter (NTC) method by *Bird* (1994). Since different species might have different computational weights it is important to properly handle the collisions between such particles. In “AQUILA” the velocity components of the lower weight particle is always updated after the collision occurs while the velocity of the higher weight particle is changed with the probability of W_1/W_2 , where W_1 and W_2 are the particles computational weights. Boundary conditions in the “AQUILA” code include reflection, absorption, accommodation and sputtering. This code was applied to the modeling of the hall thruster plume.

A three-dimensional electromagnetic PIC method on the non-uniform hexahedral was developed by *Wang* (1995). The method is parallelized and takes advantage of the hexahedral cells that are connected with cubic cells, distorted to fit the complex geometries. In *Wang’s* work (*Wang, Liewer, et al. (1995), Wang, Kondrashov, et al.*

(1995), *Wang, Lai et al. (1997)*) each hexahedral is mapped one to one to a unit cube in the logical Cartesian space. A tri-linear interpolation is used to map the logical coordinates to the physical space consisting of a hexahedral cells. The approach provided good performance on the parallel platforms. The positions are kept in the logical space while their velocities are defined in the physical space in order to calculate the Lorentz force properly. The gather/scatter procedures are performed in the Cartesian space using a charge conserving weighting scheme by *Villasenor and Buneman (1992)*. The charge accumulation is performed by calculating of how much charge crosses each face of the grid in the logical space. Particles trajectories are updated using a time-centered leapfrog scheme. The position of the computational particle in the logical space is obtained from the velocity \mathbf{c} in the physical space according to

$$\frac{d\mathbf{x}}{dt} = R(\mathbf{r}) \cdot \mathbf{c} \quad (1.1.4)$$

where R is the rotation matrix that maps the physical and logical space. It was shown that such a particle moving technique is second order accurate in time and space and is linearly proportional to the grid distortion magnitude α

$$\text{Error} = O(dt^2) + O(h^2) + O\left(\alpha \left(\frac{h}{\lambda}\right)^2\right) \quad (1.1.5)$$

In the equation (1.1.5) h is grid spacing and λ is the distortion wavelength.

The code takes the advantage of the fact that if charge is rigorously conserved globally and locally, then the electromagnetic fields may be updated by the Maxwell curl equation alone. The electromagnetic fields are updated using an explicit Discrete Surface Integral (DSI) solution of Maxwell's equations, which is an extension of the staggered mesh algorithms by *Yee (1996)*, and *Gedney and Lansing (1995)*. The DSI method is

based on the duality of the hexahedral and Cartesian meshes and uses a discrete approximation of the Stoke's theorem. Fields are obtained at the grid nodes from their normal components at the faces of the cell sharing the node. Fields are then weighted to the nodes by simple vector weighting, full volume weighting or one-sided volume weighting. It was shown by simulating the electromagnetic field propagation, that the DSI field solver is between first and second order accurate and is unstable for certain grid geometries. The simple vector weighting was found to be most stable. Properties of the weak instabilities of the DSI method were also considered. It was also shown that the error in energy conservation in the three-dimensional electromagnetic PIC simulations of the hexahedral grids does not exceed 2%. The code performance was evaluated on the 256-processor Cray computers showing a parallel efficiency of 96%.

Hermeline (1993) developed a method for the solution of Maxwell's equations on the Delaunay meshes and it's Voronoi dual in two and three dimensions. It was shown that electric and magnetic fields may be associated with either mesh. Charge and current weighting schemes were found to be a combination of a piecewise linear function and a least square method. In order to satisfy charge conservation a correction for the electric field was proposed in the form

$$\nabla \cdot (\mathbf{E} - \nabla\Phi) = \frac{\rho}{\epsilon_0} \quad (1.1.6)$$

where Φ is the correction. Error analysis of this method was presented for the eigenmodes of a square cavity on two-dimensional Cartesian and unstructured meshes. The error was shown to be negligible in both cases.

1.1 Research Objectives and Approach

Work on unstructured particle simulations at WPI's Computational Gas and Plasma Dynamics group began with the development of an unstructured grid generator *Kovalev* (2000). The Voronoi-Delaunay tetrahedral grid generator was pursued in parallel with the development of a particle simulation method that can be implemented in a PIC or a Direct Simulation Monte Carlo (DSMC) version. *Hamell et al.* (2001) presented preliminary DSMC simulations of gaseous flows in microchannels and nozzles. In *Hammel* (2002) the basic elements of the DSMC and PIC methodologies were presented. Solution of Poisson's equation were implemented using a finite volume approach. Electric fields are evaluated on the nodes of the grid using the divergence theorem. Integration of the particle equations of motion was done via the leap-frog scheme. Charge and electric field weighting was performed using linear Lagrange polynomials. The PIC methodology was used to model a high-voltage sheath. The preliminary PIC work at CGPL provided the basis of this dissertation that encompasses aspects of computational mathematics, computational implementation, and applications. The objectives are:

- Develop, implement and establish the rigorous mathematical background of the charge assignment, force interpolation, particle tracing and mover algorithms in the unstructured PIC methodology.
- Develop a comprehensive method for evaluation of error and uncertainty in the unstructured 3-D computations.
- Validate the code extensively with theoretical and computational investigations.
- Apply the code to plasma devices and sensors relevant to micropropulsion.

The approach and methodology relevant to the objectives are:

- Pursue a detailed analytical formulation of the scatter/gather weighting functions for the three-dimensional unstructured tetrahedral grids based on the long-constraint approach developed by *Hockney* (1998) for uniform grids.
- Implement a fast particle tracer based on the successive-neighbor search algorithm.
- Implement zero and first-order momentum and energy conserving weighting and interpolation schemes.
- Implement algorithms for the evaluation of nodal and cell-based macroscopic plasma parameters based on linear Lagrange polynomials.
- Implement plasma diagnostics including electrostatic energy, total/drift/thermal energies by species, and surface charge density.
- Compile and test the code on Linux and Windows platforms.
- Develop a methodology for measuring the heating, slowing-down and collision times extending the 2-D work of *Hockney* (1971).
- Validate the methodology using current collection by a cylindrical Langmuir probes in stationary and flowing plasmas in both thin-sheath and the Orbital Motion Limited regimes. Numerical results from an extensive parametric investigation are compared to the numerical results by *Laframboise* (1966) and analytical solutions by *Peterson and Talbot* (1970), *Kanal* (1964) and *Johnson and Murphree* (1969).
- Apply the method to the simulation of plasma microdevices relevant to micropropulsion. The first case involves a multi-scale device simulation of a Gated Field Emission Array cathode. Preliminary results have appeared in *Gatsonis and Spirkin* (2002). The second case, involves the simulation of a micro Retarding Potential Analyzer

(microRPA) that is developed by *Partridge and Gatsonis* (2005) for use in high density flowing plasmas. Results have appeared in *Spirkin and Gatsonis* (2003 and 2004).

Chapter 2 of the dissertation presents the detailed description of the PIC methodology and its numerical implementation. Chapter 3 presents the validation with application of the code to the current collection by a cylindrical Langmuir probes. Chapter 4 discusses the numerical heating, slowing-down and deflection times. Chapter 5 presents the simulations of a micro Retarding Potential Analyzer (RPA) and the Field Emission (FE) cathode. Chapter 6 presents the conclusions and provides suggestions for future research.

CHAPTER 2 A 3-D PARTICLE-IN-CELL METHOD ON UNSTRUCTURED VORONOI-DELAUNAY GRIDS

2.1 Mathematical Description of Collisionless Electrostatic Plasma

We consider plasma occupying a volume V consisting of several species denoted by the species index s each with N_s particles. The mathematical description of the collisionless plasma involves the Vlasov-Maxwell system (*Montgomery and Tidman (1964)*). The single particle distribution function is the particle density in the velocity-space phase space and gives the average number of particles in a volume $d^3r d^3c$ of the phase-space centered at a point (\mathbf{r}, \mathbf{c}) as

$$f_s(\mathbf{r}, \mathbf{c}, t) d^3r d^3c = d^6 N_s(t) \quad (2.1.1)$$

The local number density is then

$$n_s(r, t) = \int f_s(\mathbf{r}, \mathbf{c}, t) d^3c \quad (2.1.2)$$

The equation for the distribution function is,

$$\frac{\partial f_s}{\partial t} + \mathbf{c} \cdot \frac{\partial f_s}{\partial \mathbf{r}} + \frac{q_s}{m_s} [\mathbf{E}(\mathbf{r}, t) + \mathbf{c} \times \mathbf{B}_{\text{ext}}(\mathbf{r}, t)] \cdot \frac{\partial f_s}{\partial \mathbf{c}} = 0 \quad (2.1.3)$$

The self-consistent electric field (and potential) is due to the smoothed distribution (internal) and external distribution of charges given by Maxwell's equation that becomes simply Poisson's equation

$$\varepsilon_0 \nabla \cdot \mathbf{E}(\mathbf{r}, t) = \sum_s q_s \int f_s(\mathbf{r}, \mathbf{c}, t) d^3c - \rho_{\text{ext}}(\mathbf{r}, t) \quad (2.1.4)$$

The formulation assumes that there are no external electric and magnetic fields and assumes that there are no induced magnetic fields due to the particle motion. Those

external fields obey also the Maxwell equations and therefore we can show that in the electrostatic limit.

$$\mathbf{E}(\mathbf{r}, t) = \mathbf{E}_{\text{ext}}(\mathbf{r}, t) + \mathbf{E}_{\text{int}}(\mathbf{r}, t) \quad (2.1.5)$$

$$\mathbf{E}_{\text{int}}(\mathbf{r}, t) = -\frac{\partial}{\partial \mathbf{r}} \left(\sum_s \int \frac{q_s}{4\pi\epsilon_0 |\mathbf{r} - \mathbf{r}'|} f_s(\mathbf{r}', \mathbf{c}', t) d^3r d^3c \right) = -\nabla \Phi_{\text{int}}(\mathbf{r}, t) \quad (2.1.6)$$

and one can show that the internal fields follow

$$\epsilon_0 \nabla \cdot \mathbf{E}_{\text{int}} = \sum_s q_s \int f_s(r, c, t) d^3r \quad (2.1.7)$$

In the Vlasov-Poisson system the discreteness of plasma particles is neglected, and the electric fields are found self-consistently from smoothed charged and current distributions. In addition the motion in the phase-space is incompressible, and the solution to Vlasov equation states that

$$f(\mathbf{r}, \mathbf{c}, t) = f(\mathbf{r}, \mathbf{w}, t = 0) \quad (2.1.8)$$

along characteristics which are the single-particle trajectories in the presence of electromagnetic fields,

$$\frac{d\mathbf{r}(t)}{dt} = \mathbf{c}(t) \quad (2.1.9)$$

$$m_s \frac{d\mathbf{c}(t)}{dt} = \frac{q_s}{m_s} [\mathbf{E}(\mathbf{r}, t) + \mathbf{c} \times \mathbf{B}_{\text{ext}}(\mathbf{r}, t)] \quad (2.1.10)$$

2.2 Unstructured 3-D PIC Methodology

Particle simulation methods for plasmas address the solution utilizing particle trajectories, and then reconstruct the distributions function and its moments. In the computational domain the number of particle (markers or macroparticles) are much fewer than the real system. The particle weight of a species is designated as F_s .

We will provide next the general framework for a PIC implementation in unstructured tetrahedral grids. There are two important parts of the PIC or PM method: the introduction of the finite-size particles and the introduction of the grid.

2.3 Particle In Cell Computational Cycle

A standard PIC method computational cycle is shown in Figure 1. It starts with the charge weighting from the position of the computational particles to the grid nodes (gather). Then, the Poisson's equation is solved on the nodes in order to obtain electric potential and electric field. Electric field from the grid is weighted back to the particles positions (scatter), force acting on the particles is calculated and particles are moved according to their acceleration. Numerical implementation of these four steps on the unstructured tetrahedral meshes is discussed in details in the following chapters.

2.4 Finite Size Particles

The introduction of finite-sized particles can be described by a function designated as the particle shape. *Hockney and Eastwood* (1981) refer to it as the "cloud shape". *Birdsall and Langdon* (1985) use this term in the process of charge-assignment (or particle weighting).

The cloud function shape or simply shape factor $S(\mathbf{r}, \mathbf{r}_i)$ (units of L^{-3}) provides the charge density at the position \mathbf{r} from the center of the particle $\mathbf{r}_p \equiv (x_p, y_p, z_p)$,

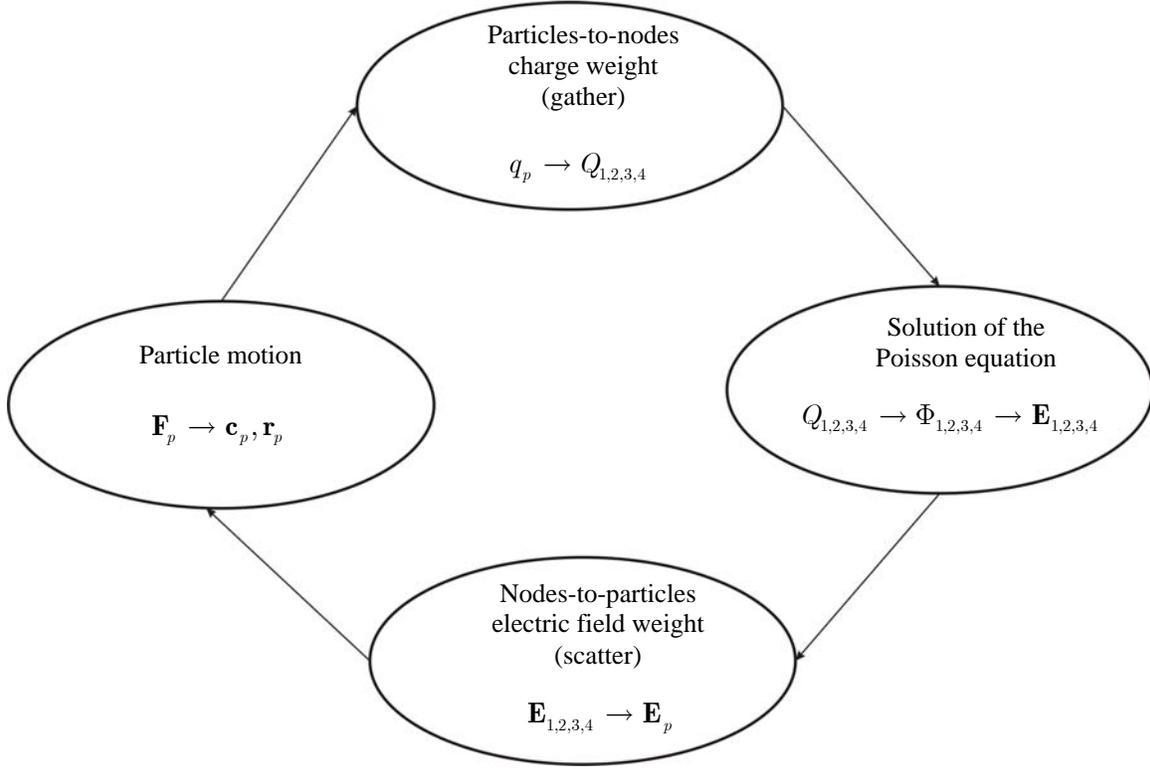


Figure 1. Particle-In-Cell computational cycle.

$$\rho_p(\mathbf{r}, \mathbf{r}_p) = q_p S(\mathbf{r}, \mathbf{r}_p) \quad (2.4.1)$$

The total charge of a particle is

$$\int q_p S(\mathbf{r}, \mathbf{r}_p) d^3r = q_p \quad (2.4.2)$$

The shape factor is therefore normalized

$$\int S(\mathbf{r}, \mathbf{r}_i) d^3r = 1 \quad (2.4.3)$$

An example of a symmetric shape factor provides point particles with charge density

$$q_p S(\mathbf{r}, \mathbf{r}_i) = q_p \delta(\mathbf{r} - \mathbf{r}_i) \equiv q_p \delta(x - x_p) \delta(y - y_p) \delta(z - z_p) \quad (2.4.4)$$

Clouds with uniform charge density, with

$$S(\mathbf{r}, \mathbf{r}_i) = a^{-3} \quad (2.4.5)$$

The distribution function of these finite-sized particles is

$$\tilde{f}_s(\mathbf{r}, \mathbf{v}, t) = \sum_{p=1}^{N_s} S(\mathbf{r} - \mathbf{r}_{sp}) \delta(\mathbf{c} - \mathbf{c}_{sp}) \quad (2.4.6)$$

The units of $\delta(\mathbf{c} - \mathbf{c}_{sp})$ are $[L^3 / T^3]$. Then density, charge density and current due to N_s

particles of species s at \mathbf{r} are

$$n_s(\mathbf{r}, t) = \int f_s(\mathbf{r}, \mathbf{c}, t) d^3 c = \sum_{p=1}^{N_s} S(\mathbf{r} - \mathbf{r}_{sp}) \quad (2.4.7)$$

$$\rho_s(\mathbf{r}, t) = \int f_s(\mathbf{r}, \mathbf{c}, t) d^3 c = \sum_{p=1}^{N_s} q_i S(\mathbf{r} - \mathbf{r}_{sp}) \quad (2.4.8)$$

$$\mathbf{j}_s(\mathbf{r}, t) = \int q_s \mathbf{c}_s(\mathbf{r}, t) f_s(\mathbf{r}, \mathbf{c}, t) d^3 c = \sum_{p=1}^{N_s} q_s c_{sp} S(\mathbf{r} - \mathbf{r}_{sp}) \quad (2.4.9)$$

The total charge is

$$Q_s(t) = \int_V \rho_s(\mathbf{r}, t) d^3 r = \sum_{p=1}^{N_s} q_i S(\mathbf{r}, \mathbf{r}_{sp}) \quad (2.4.10)$$

2.5 Unstructured Delaunay Grid Generation

The unstructured grid generator provides a three-dimensional, tetrahedral, Delaunay mesh for arbitrary geometric configurations. Surface triangulation is the first step of the grid generation. A surface generator was designed by *Hammel* (2001) for axially symmetric objects and objects defined by bi-linear elements. A two dimensional topology is performed by connecting points with lines, arcs or parabolas according to the specified

spacing values. In order to create an axisymmetric surface this topology is analytically tiled about an axis. Two edges of bilinear objects may be created with arbitrary spacing values. It is required for other two edges to have the same number of nodes as the opposing edges. After axisymmetric or bi-linear objects are created they may be rotated and added to a group of objects.

The unstructured mesh generator is based on *Watson's* (1981) incremental node insertion method, which uses properties of the Delaunay triangulation. An initial mesh is required for *Watson's* method, in order to have a domain where point insertion to begin. The initial mesh chosen is a cube divided into six tetrahedra. After the initial mesh is generated, the source geometry is inserted into the domain. This procedure is done in two steps. First, the boundary nodes are successively inserted into the grid via *Watson's* algorithm. The second step is to recover all the boundary facets present in the source geometry using local modifications to the grid. The cells external to the grid are removed as well as cells in any internal cavity.

The underlying sizing function – defined by the source geometry – requires the interior of the grid to be enriched with nodes to the specified density. For this purpose the algorithm by *Borouchaki and George* (1997) is implemented and extended to three dimensions by *Kovalev* (2000). In this algorithm, the characteristic distance between nodes is specified for each grid node as h . Every existing edge of the grid is divided into a number of new prospective nodes, so that the new resulting edge segments vary in length gradually between the h -values of the edge vertices. The prospective nodes are filtered in order to satisfy the spacing and grid quality criteria. Nodes falling too close to existing nodes are eliminated. Nodes that worsen grid quality as specified by the lowest dihedral

angle in a set of cells are also discarded. The nodes that are not rejected are inserted into the grid via Watson's algorithm. The edge division process is repeated while new nodes are inserted. A heuristic quality improvement procedure has been also implemented based on a user-defined minimum dihedral angle.

2.6 Finite Sized Particles on a Grid Particle Weighting on to Grid (Gather)

The introduction of grid where properties are sampled introduces a new level of mathematical description. Particles of any shape assign their charge onto the grid and this process of interpolation results in an effective particles shape or weighing function. We will follow *Hockney and Eastwood* (1981) who described this process by the function W (refereed to as shape factor S in *Birdsall and Langdon* (1985)) *Birdsall and Langdon* use S for this process although in certain instances they discussed particle shape in general.

The area of overlap between the cloud shape (or shape factor) and the grid cell determined the charge assigned to the grid point. In this interpretation particles carry their shape factors with them.

$$\rho_{sk} = \frac{F_s}{V_k} \int_{V_k} n_s(\mathbf{r}) d\mathbf{r} = \frac{F_s}{V_k} \int_{V_k} \left(\sum_{p=1}^{N_p} S(\mathbf{r}, \mathbf{r}_p) \right) d\mathbf{r} \quad (2.6.1)$$

The hierarchy of charge assignment functions (or weights) is derived by *Hockney and Eastwood* (1981) according to the long-range, smoothness, and momentum conservation constraints. We generalize this derivation for structured, non-uniform grids in 1-D, 2-D and 3-D as well as Delauney-Voronoi unstructured grids in 2-D and 3-D.

1-D

We consider first the case of a 1-D domain $\Omega = [x_L, x_R]$ discretized by a structured Cartesian grid of N_X grid points. The grid point is denoted by x_I where the grid-index is $I = 1, N_X$. There are $N_X - 1$ node-centered cells in the region each denoted by $\Gamma_I = [x_{I-1/2}, x_{I+1/2}]$ where $x_{I-1/2} = x_I - \Delta x_I / 2$ and $x_{I+1/2} = x_I + \Delta x_I / 2$. The operators $\Delta x_I = x_{I+1} - x_I$, and $\nabla x_I = x_I - x_{I-1}$. We designate with $W_I(x_i) = W(x_I, x_i)$ the fraction of a charge from a particle located at x_i assigned to grid point x_I .

Following *Hockney and Eastwood* (1981) we assume that the potential at position x due to the unit charge in position $x_p \in \Omega_I = [x_I, x_{I+1}]$ is given by the Green's function $G(x - x_I)$. The potential then due to a unit charge at position x_i is given by the expansion

$$\begin{aligned} \phi(x) = \sum_I W_I(x_p) G(x - x_p) + \left(\sum_I W_I(x_p) (x_p - x_I) \right) \frac{dG(x_p - x)}{dx} + \\ \left(\sum_I W_I(x_p) (x_p - x_I)^2 \right) \frac{d^2 G(x_p - x)}{dx^2} + O(x_p - x_I)^3 \end{aligned} \quad (2.6.2)$$

where the sum is taken over M nodes close to I used to distribute the charge. Charge conservation requires that,

$$\sum_I W_I(x_p) = 1 \quad (2.6.3)$$

For $M > 1$ additional constraint equation appears, by requiring higher-order terms in the expansion of equation (2.6.2) to become grid-independent. Since both $\phi(x)$ and $G(x_p - x)$ are even functions we require that that for values of n

$$\sum_I^M W_I(x_p) (x_p - x_I)^n = \begin{cases} 0 & n \text{ odd} \\ C & n \text{ even} \end{cases} \quad (2.6.4)$$

The process outlined above is equivalent to a multipole expansion (*Birdsall and Langdon* (1985)).

The weights can also be derived using the particle shape. The overlap between the cloud shape from a particle at position x_p and the cell Γ_I gives the fraction of charge to be assigned to a node x_I as

$$W(x_p - x_I) = W_I(x_p) = \int_{x_{I-1/2}}^{x_{I+1/2}} S(x_p - x) dx \quad (2.6.5)$$

The number and charge density at grid point x_I from N_s particles of species s is evaluated using the length of the cell Γ_I following,

$$n_s(x_I) \equiv n_{Is} = \frac{2}{\nabla x_I + \Delta x_I} \sum_{p=1}^{N_s} W(x_{ps}, x_I) \quad (2.6.6)$$

$$\rho_s(x_I) \equiv \rho_{Is} = \frac{2}{\nabla x_I + \Delta x_I} \sum_{p=1}^{N_s} q_{is} W(x_{ps}, x_I) \quad (2.6.7)$$

NGP

In the nearest grid point scheme the charge q_p of the point particle $x_p \in \Omega_I = [x_I, x_{I+1}]$ is distributed to the grid point closest to it ($M = 1$). The shape function in this case is

$$S(x_I, x_p) = \delta(x - x_p) \quad (2.6.8)$$

The charge-assignment interpolation function (or charge weighting function)

$$\begin{aligned} W_I(x_p) &\equiv W(x_p - x_I) = 1 & x_I \leq x_p < x_{I+1/2} \\ W_{I+1}(x_p) &\equiv W(x_p - x_{I+1}) = 1 & x_{I+1/2} \leq x_p < x_{I+1} \end{aligned} \quad (2.6.9)$$

The weighting function can be also described with the left and right-side weights as

$$W_I(x_p) \equiv W(x_p, x_I) = \begin{cases} W_I^-(x_p) & x_I < x_p < x_{I+1/2} \\ W_I^+(x_p) & x_I < x_p < x_{I+1/2} \\ 0 & \text{all other } x \end{cases} \quad (2.6.10)$$

For $|x_p - x_I| \leq \Delta x / 2$ the charge assigned on the grid is

$$q_{Is} = q_{ps} W(x_{ps} - x_I) = q_p \quad (2.6.11)$$

therefore, the charge is conserved. It is also easy to show that

$$W(x_p - x_I) = W_I(x_p) = \int_{x_I - \Delta x / 2}^{x_I + \Delta x / 2} \delta(x - x_p) dx \quad (2.6.12)$$

Linear (CIC or PIC)

In linear interpolation (CIC or PIC) scheme the charge q_p of the particle

$x_p \in \Omega_I = [x_I, x_{I+1}]$ is distributed to two grid points via the charge assignment functions

$W_I(x_p)$ and $W_{I+1}(x_p)$. The two constraints satisfied are

$$W_I(x_p) + W_{I+1}(x_p) = 1 \quad (2.6.13)$$

$$W_I(x_p)(x_p - x_I) + W_{I+1}(x_p)(x_p - x_{I+1}) = 0 \quad (2.6.14)$$

The charge assignment functions for $x_p \in \Omega_I = [x_I, x_{I+1}]$ shown in Figure 2 are

$$W(x_I, x_p) \equiv W_I(x_p) = 1 - \frac{x_p - x_I}{\Delta x_I} = \frac{x_{I+1} - x_p}{\Delta x_I} \quad (2.6.15)$$

$$W(x_{I+1}, x_p) = W_{I+1}(x_p) = 1 + \frac{x_p - x_{I+1}}{\Delta x_I} = \frac{x_p - x_I}{\Delta x_I} \quad (2.6.16)$$

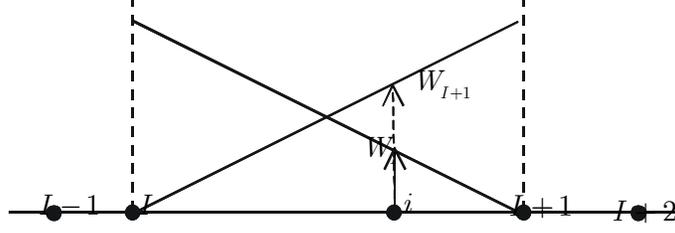


Figure 2. Linear weighting functions in 1-D.

In terms of the left and right charge assignment functions for a particle at

$$x_p \in \Gamma_I = [x_{I-1/2}, x_{I+1/2}]$$

$$W_I(x_p) \equiv W(x_p, x_I) = \begin{cases} W_I^-(x_p) = \frac{x_I - x_p}{\Delta x_I} & x_I < x_p < x_I \\ W_I^+(x_p) = \frac{x_p - x_I}{\Delta x_I} & x_I < x_p < x_{I+1} \\ 0 & \text{all other } x \end{cases} \quad (2.6.17)$$

The charge assigned to nodes X_I, X_{I+1} due to a particle at $x_p \in \Omega_I = [x_I, x_{I+1}]$

$$q(x_I, x_p) \equiv q_I(x_p) = q_p \left(\frac{x_{I+1} - x_p}{\Delta x_I} \right) \quad (2.6.18)$$

$$q(x_{I+1}, x_p) \equiv q_I(x_p) = q_p \left(\frac{x_p - x_I}{\Delta x_I} \right) \quad (2.6.19)$$

2-D, Cartesian, Non-uniform

We consider next a 2-D domain $\Omega = [(x_L, y_L) \times (x_R, y_R)]$ discretized by a structured

Cartesian grid of N_X, N_Y grid points. The grid point denoted with (I, J) has coordinates

(x_I, y_J) where the grid-index is $I = 1, N_X$ and $J = 1, N_Y$. There are $(N_X - 1)(N_Y - 1)$ node-centered cells in the region. A cell around grid point (x_I, y_J) is denoted by

$$\Gamma_{I,J} = \left[(x_{I-1/2}, x_{I+1/2}) \times (y_{J-1/2}, y_{J+1/2}) \right] \quad (2.6.20)$$

where $x_{I-1/2} = x_I - \nabla x_I / 2$, $x_{I+1/2} = x_I + \Delta x_I / 2$, $y_{J-1/2} = y_J - \nabla y_J / 2$ and $y_{J+1/2} = y_J + \Delta y_J / 2$. The forward and backward operators are $\Delta x_I = x_{I+1} - x_I$, $\Delta y_I = y_{J+1} - y_J$ and $\nabla x_I = x_I - x_{I-1}$, $\nabla y_J = y_J - y_{J-1}$ respectively. We designate also with $W_{I,J}(x_p, y_p) = W(x_{I,J}, x_p, y_p)$ the fraction of a charge from a particle p located at (x_p, y_p) assigned to grid point (I, J) . The particle is at position $(x_p, y_p) \in \Omega_{I,J} \left[(x_I, x_I) \times (y_J, y_{J+1}) \right]$ as depicted in Figure 3.

The long-range constraints are derived in a fashion similar to the 1-D case. The potential at $r \equiv (x, y)$ due to the charges at M grid points is given by

$$\phi(x, y) = \sum_{I,J} W_{I,J}(x_p, y_p) G(r - r_{I,J}) \quad (2.6.21)$$

Taylor expanding $G(r - r_{IJ})$ about $(r - r_p)$ and assuming that $G(\mathbf{r}') = G(r')$

$$\begin{aligned} \phi(x, y) = & \sum_{I,J} W_{I,J}(x_p, y_p) G(r - r_p) + \\ & \sum_{I,J} W_{I,J}(x_p, y_p) \left[(x_p - x_I) \frac{dG(r_p - r)}{dx} + (y_p - x_I) \frac{dG(r_p - r)}{dy} + \right. \\ & \left. \mathcal{O}\{(x_p - x_I)^2, (y_p - y_I)^2, \} \right] \end{aligned} \quad (2.6.22)$$

or

$$\phi(x, y) = \sum_{I,J} W_{I,J}(x_p, y_p) \sum_{r,s=0}^{\infty} \frac{\Delta_x^r \Delta_y^s}{r! s!} \frac{\partial^{r+s} G(\mathbf{r} - \mathbf{r}_{IJ})}{\partial x^r \partial y^s} \quad (2.6.23)$$

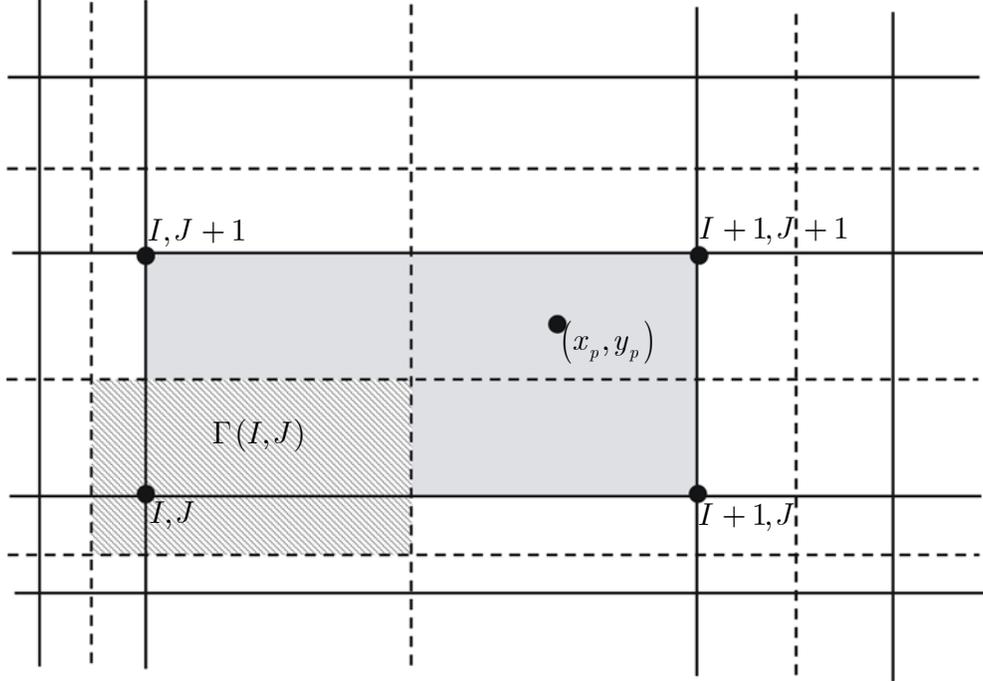


Figure 3. Example of the 2-D structured Cartesian grid with the computational particle located at

$$(x_p, y_p).$$

where the summation of the grid indices (I, J) is carried over the M nodes used in the distribution process, and $\Delta_x = x_p - x_I, \Delta_y = y_p - y_I$.

The charge conservation gives the first constraint as,

$$\sum_{I,J} W_{I,J}(x_p, y_p) = 1 \quad (2.6.24)$$

where the summation of the grid indices (I, J) is carried over the M nodes used in the distribution process. The requirement that the first-order terms in the expansion to be grid independent provides the first-order constraint as

$$\sum_{I,J} W_{I,J}(x_p, y_p)(x_p - x_I)^n = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.25)$$

$$\sum_{I,J} W_{I,J}(x_p, y_p) (y_p - y_I)^n = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.26)$$

In equations (2.6.25) and (2.6.26) n is the order of the weighting scheme used ($n = 0$ for NGP, $n = 1$ for CIC).

NGP (Zero-order)

For a particle in a position $(x_p, y_p) \equiv r_p \in \Omega_{I,J} = [(x_I, x_{I+1}) \times (y_J, y_{J+1})]$ the charge is assigned to the closest node with functions $W_{I,J}, W_{I+1,J}, W_{I+1,J+1}, W_{I,J+1}$. With $M = 1$ the charge conservation constraint becomes

$$W_{I,J}(x_p) = 1 \quad x_I \leq x_p < x_{I+1/2}, y_J \leq y_p < y_{J+1/2} \quad (2.6.27)$$

$$W_{I+1,J}(x_p), W_{I+1,J+1}(x_p), W_{I,J+1}(x_p) = 0$$

The number density and charge density at a node (x_I, y_J) from N_s particles of species s is evaluated using the area of the cell Γ_{IJ}

$$n_s(x_I, y_J) \equiv n_{sI,J} = \frac{4}{(\nabla x_I + \Delta x_I)(\nabla y_J + \Delta y_J)} \sum_{p=1}^{N_s} W_{I,J}(x_{ps}) \quad (2.6.28)$$

$$\rho_s(x_I, y_J) \equiv \rho_{sI,J} = \frac{4}{(\nabla x_I + \Delta x_I)(\nabla y_J + \Delta y_J)} \sum_{p=1}^{N_s} q_{ps} W_{I,J}(x_{ps}) \quad (2.6.29)$$

CIC (First-order)

The charge of a particle in a position $(x_p, y_p) \equiv r_p \in \Omega_{I,J} = [(x_I, x_{I+1}) \times (y_J, y_{J+1})]$ is assigned to the four nodes associated with Ω_{IJ} with functions $W_{I,J}, W_{I+1,J}, W_{I+1,J+1}, W_{I,J+1}$. The charge conservation constraint is

$$W_{I,J}(r_p) + W_{I+1,J}(r_p) + W_{I+1,J+1}(r_p) + W_{I,J+1}(r_p) = 1 \quad (2.6.30)$$

The first-order constraint is

$$W_{I,J}(r_p)(x_p - x_I) + W_{I+1,J}(r_p)(x_p - x_{I+1}) + W_{I+1,J+1}(r_p)(x_p - x_{I+1}) + W_{I,J+1}(r_p)(x_p - x_I) = 0 \quad (2.6.31)$$

$$W_{I,J}(r_p)(y_p - y_J) + W_{I+1,J}(r_p)(y_p - y_J) + W_{I+1,J+1}(r_p)(y_p - y_{J+1}) + W_{I,J+1}(r_p)(y_p - y_{J+1}) = 0 \quad (2.6.32)$$

In addition we require that the function be written as the product

$$W_{I,J}(x_p, y_p) = W_{I,J}(x_p)W_{I,J}(y_p) \quad (2.6.33)$$

The charge assignment functions are then,

$$W_{I,J}(x_p, y_p) = \frac{(x_{I+1} - x_p)(y_{J+1} - y_p)}{\Delta x_I \Delta y_J} = \frac{\Omega_A}{\Omega_{I,J}} \quad (2.6.34)$$

$$W_{I+1,J}(x_p, y_p) = \frac{(x_p - x_I)(y_{J+1} - y_p)}{\Delta x_I \Delta y_J} = \frac{\Omega_B}{\Omega_{I,J}} \quad (2.6.35)$$

$$W_{I+1,J+1}(x_p, y_p) = \frac{(x_p - x_I)(y_p - y_J)}{\Delta x_I \Delta y_J} = \frac{\Omega_C}{\Omega_{I,J}} \quad (2.6.36)$$

$$W_{I,J+1}(x_p, y_p) = \frac{(x_{I+1} - x_p)(y_p - y_J)}{\Delta x_I \Delta y_J} = \frac{\Omega_D}{\Omega_{I,J}} \quad (2.6.37)$$

The area-weighting interpretation of these linear assignment functions is clear and is indicated in Figure 4.

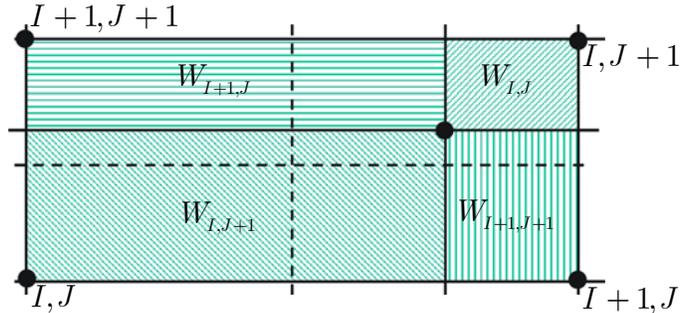


Figure 4. Graphical representation of the Linear weighting on the 2-D structured Cartesian grid.

The number density and charge density at a node (x_I, y_J) from N_s particles of species s is evaluated using the area of the cell Γ_{IJ}

$$n_s(x_I, y_J) \equiv n_{sI,J} = \frac{4}{(\nabla x_I + \Delta x_I)(\nabla y_J + \Delta y_J)} \sum_{p=1}^{N_s} W_{I,J}(x_{ps}) \quad (2.6.38)$$

$$\rho_s(x_I, y_J) \equiv \rho_{sI,J} = \frac{4}{(\nabla x_I + \Delta x_I)(\nabla y_J + \Delta y_J)} \sum_{p=1}^{N_s} q_{ps} W_{I,J}(x_{ps}) \quad (2.6.39)$$

2-D Delauney-Voronoi

We consider next a 2-D domain Ω discretized by an unstructured Delauney grid.

The Delaunay triangular cell is formed by a three nodes located at $\mathbf{r}_1 = (x_1, y_1)$, $\mathbf{r}_2 = (x_2, y_2)$, $\mathbf{r}_3 = (x_3, y_3)$.

The Delauney cell has an area Ω_{123} given by

$$\Omega_{ABC} = \frac{1}{2} |\mathbf{r}_{12} \times \mathbf{r}_{13}| = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix} \quad (2.6.40)$$

where $\mathbf{r}_{12} = \mathbf{r}_2 - \mathbf{r}_1$ and $\mathbf{r}_{13} = \mathbf{r}_3 - \mathbf{r}_1$.

We consider next a particle at position $(x_p, y_p) \equiv r_p \in \Omega_{123}$. Its charge is assigned to the three nodes forming the triangular cell depicted in Figure 5.

The potential at $r \equiv (x, y)$ due to the charges at M grid points of the Delauney is given by

$$\phi(x, y) = \sum_{I=1, M} W_I(x_p, y_p) G(r - r_I) \quad (2.6.41)$$

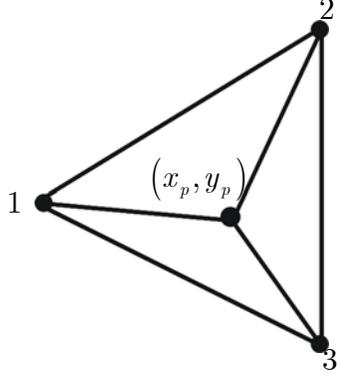


Figure 5. Example of the Delaunay triangular computational cell.

where I designates the local numbering, and the sum is taken over the M nodes of the Delauney used in the charge assignment. Taylor expanding $G(r - r_I)$ about $(r - r_p)$ and assuming that $G(\mathbf{r}') = G(r')$

$$\begin{aligned} \phi(x, y) = & \sum_{I=1, M} W_I(x_p, y_p) G(r - r_p) + \sum_{I=1, M} W_I(x_p, y_p) (x_p - x_I) \frac{dG(r_p - r)}{dx} \\ & + \sum_{I=1, M} W_I(x_p, y_p) (y_p - y_I) \frac{dG(r_p - r)}{dy} + O\{(x_p - x_I)^2, (y_p - y_I)^2\} \end{aligned} \quad (2.6.42)$$

Charge conservation requires that

$$\sum_I W_I(x_p, y_p) = 1 \quad (2.6.43)$$

For $M > 1$ additional constraint equation appears. The requirement that the first-order terms in the expansion to be grid independent provides the first-order constraint as

$$\sum_{I=1, M} W_I(x_p, y_p) (x_p - x_I) = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.44)$$

$$\sum_{I=1, M} W_I(x_p, y_p) (y_p - y_I) = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.45)$$

NGP (Zero-order)

In this case the charge from the particle at position $(x_p, y_p) \equiv r_p \in \Omega_{ABC}$ is assigned to node closest to it.

$$\begin{aligned} W_1(\mathbf{r}_p) &= 1 \text{ if } \min |\mathbf{r}_{1,2,3} - \mathbf{r}_p| = |\mathbf{r}_1 - \mathbf{r}_p| \\ W_2(\mathbf{r}_p) &= W_3(\mathbf{r}_p) = 0 \end{aligned} \quad (2.6.46)$$

CIC (First-order)

In this case the charge from the particle at position P with $\mathbf{r}_p \equiv (x_p, y_p) \in \Omega_{123}$ is assigned to the three nodes of the Delauney cell depicted in Figure 5. The charge conservation provides one constraint equation

$$W_1(r_p) + W_2(r_p) + W_3(r_p) = 1 \quad (2.6.47)$$

The first-order constraints equations are

$$W_1(r_p)(x_p - x_1) + W_2(r_p)(x_p - x_2) + W_3(r_p)(x_p - x_3) = 0 \quad (2.6.48)$$

$$W_1(r_p)(y_p - y_1) + W_2(r_p)(y_p - y_2) + W_3(r_p)(y_p - y_3) = 0 \quad (2.6.49)$$

Solution to the above system of constraints provides the functions

$$W_1 = \frac{x_2 y_3 - x_p y_3 - x_3 y_2 + x_p y_2 + x_3 y_p - x_2 y_p}{x_1 y_3 - x_2 y_3 - x_3 y_1 + x_2 y_1 + x_3 y_2 - x_1 y_2} = \frac{\Omega_{P23}}{\Omega_{123}} \quad (2.6.50)$$

$$W_2 = \frac{-x_1 y_3 + x_p y_3 + x_3 y_1 - x_p y_1 - x_3 y_p + x_1 y_p}{x_1 y_3 - x_2 y_3 - x_3 y_1 + x_2 y_1 + x_3 y_2 - x_1 y_2} = \frac{\Omega_{P13}}{\Omega_{123}} \quad (2.6.51)$$

$$W_3 = \frac{-x_2 y_1 + x_p y_1 + x_1 y_2 - x_p y_2 - x_1 y_p + x_2 y_p}{x_1 y_3 - x_2 y_3 - x_3 y_1 + x_2 y_1 + x_3 y_2 - x_1 y_2} = \frac{\Omega_{P12}}{\Omega_{123}} \quad (2.6.52)$$

In the above expressions Ω_{123} is the cell area

$$\Omega_{123} = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{vmatrix} = \frac{1}{2} |\mathbf{r}_{13} \times \mathbf{r}_{23}| \quad (2.6.53)$$

Ω_{P23} is the area of the triangle formed by the position of the particle and nodes 2 and 3 given by

$$\Omega_{P23} = \frac{1}{2} \begin{vmatrix} 1 & 1 & 1 \\ x_P & x_2 & x_3 \\ y_P & y_2 & y_3 \end{vmatrix} = \frac{1}{2} |\mathbf{r}_{P2} \times \mathbf{r}_{P3}| \quad (2.6.54)$$

where $\mathbf{r}_{P2} = \mathbf{r}_P - \mathbf{r}_2$ and $\mathbf{r}_{P3} = \mathbf{r}_P - \mathbf{r}_3$.

The implementation of the CIC weights is represented graphically in Figure 6 and corresponds to the area-weighting used in the 2D Cartesian case.

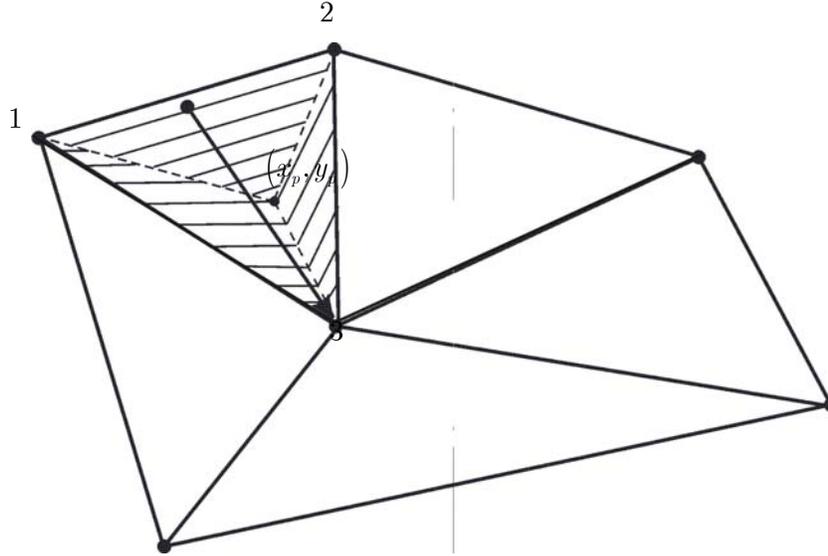


Figure 6. Graphical representation of the Linear weighting on a 2-D Delaunay grid.

The number density and charge density at a node $r_1 = (x_1, y_1)$ from N_s particles of species s is evaluated using the area of the Voronoi cell Γ_1 depicted in Figure 7.

$$n_s(x_1, y_1) \equiv n_{s1} = \frac{1}{\Gamma_1} \sum_{p=1}^{N_s} W_1(x_{ps}, y_{ps}) \quad (2.6.55)$$

$$\rho_s(x_1, y_1) \equiv \rho_{s1} = \frac{1}{\Gamma_1} \sum_{p=1}^{N_s} q_s W_1(x_{ps}, y_{ps}) \quad (2.6.56)$$

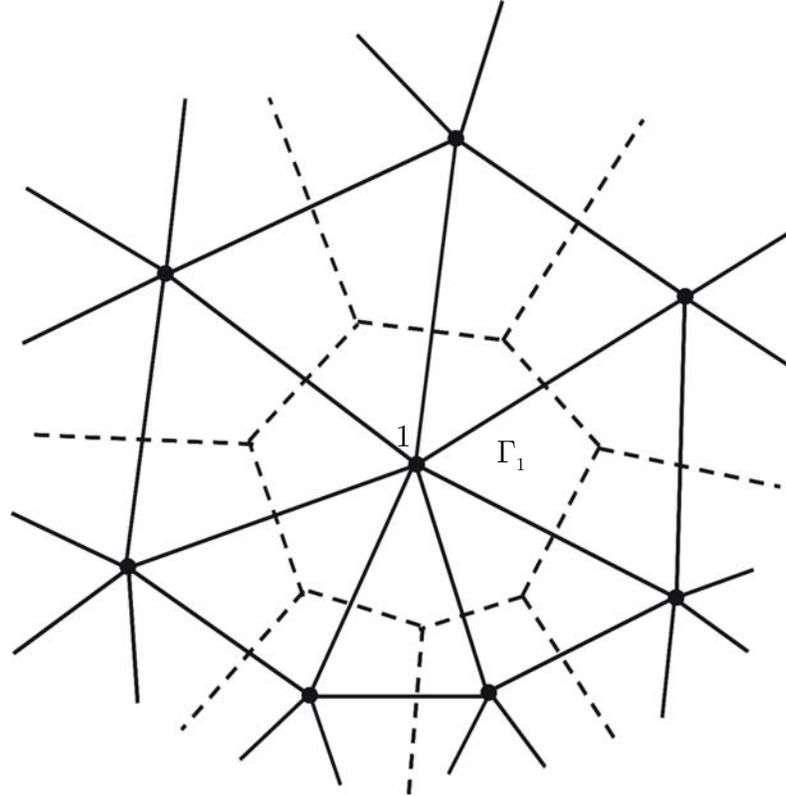


Figure 7. Example of the 2-D Voronoi cell.

3D Delauney-Voronoi

We consider next a 3-D domain Ω discretized by an unstructured Delauney grid. The Delauney tetrahedron formed by the nodes located at $\mathbf{r}_1 = (x_1, y_1, z_1), \mathbf{r}_2 = (x_2, y_2, z_2), \mathbf{r}_3 = (x_3, y_3, z_3), \mathbf{r}_4 = (x_4, y_4, z_4)$ is illustrated in Figure 8.

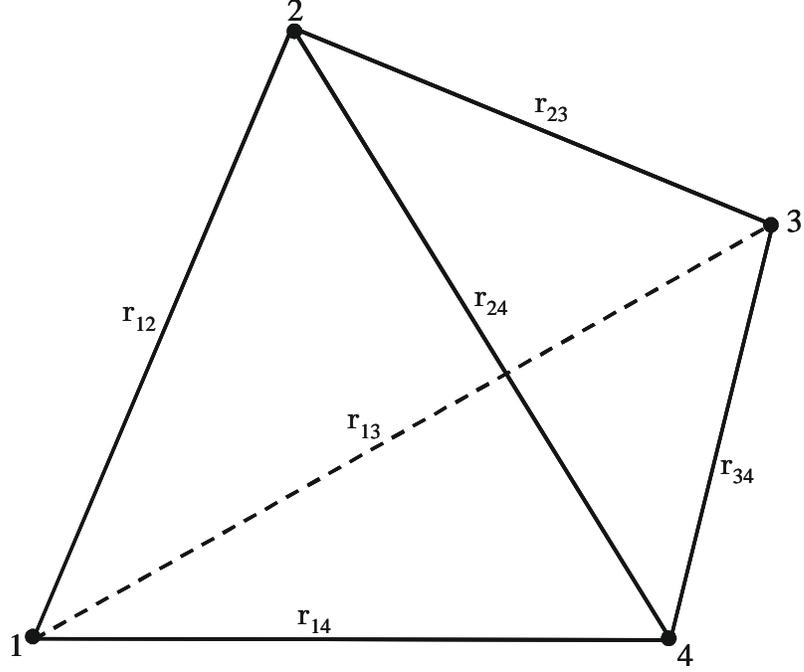


Figure 8. Example of the tetrahedral computational cell.

The Delauney cell has a volume given by

$$\Omega_{1234} = \frac{1}{6} [\mathbf{r}_{41} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})] \quad (2.6.57)$$

We consider next a particle charge that is assigned to the four nodes of the tetrahedral cell depicted in Figure 8. For a particle in a position $\mathbf{r}_p = (x_p, y_p, z_p) \in \Omega_{1234}$ the charge is assigned to the M nodes associated with Ω_{1234} with functions W_1, W_2, W_3, W_4 . The long-range constraints are derived in a fashion similar to the 2-D case. The potential at $r \equiv (x, y, z)$ due to the charges at M grid points is given by

$$\phi(x, y, z) = \sum_{I=1,4} W_I(x_p, y_p, z_p) G(r - r_I) \quad (2.6.58)$$

Taylor expanding $G(r - r_I)$ about $(r - r_p)$ and assuming that $G(\mathbf{r}') = G(r')$

$$\begin{aligned} \phi(x, y) = & \sum_{I,J} W_{I,J}(x_p, y_p) G(r - r_p) + \\ & + \sum_{I,J} W_{I,J}(x_p, y_p) \left[\begin{aligned} & (x_p - x_I) \frac{dG(r_p - r)}{dx} + (y_p - y_I) \frac{dG(r_p - r)}{dy} + \\ & + (z_p - z_I) \frac{dG(r_p - r)}{dz} + O\{(x_p - x_I)^2, (y_p - y_I)^2, (z_p - z_I)^2\} \end{aligned} \right] \end{aligned} \quad (2.6.59)$$

Charge conservation requires that

$$\sum_I W_I(x_p, y_p, z_p) = 1 \quad (2.6.60)$$

For $M > 1$ additional constraint equation appears. The requirement that the first-order terms in the expansion to be grid independent provides the first-order constraint as

$$\sum_{I=1,M} W_I(x_p, y_p, z_p) (x_p - x_I) = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.61)$$

$$\sum_{I=1,M} W_I(x_p, y_p, z_p) (y_p - y_I) = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.62)$$

$$\sum_{I=1,M} W_I(x_p, y_p, z_p) (z_p - z_I) = \begin{cases} 0 & \text{even } n \\ c & \text{odd } n \end{cases} \quad (2.6.63)$$

NGP (Zero-order)

The charge from the particle at position $r_p = (x_p, y_p, z_p) \in \Omega_{1234}$ is assigned to closest node.

$$\begin{aligned} W_1(\mathbf{r}_p) &= 1 \text{ if } \min |\mathbf{r}_{1,2,3,4} - \mathbf{r}_p| = |\mathbf{r}_1 - \mathbf{r}_p| \\ W_2(\mathbf{r}_p) &= W_3(\mathbf{r}_p) = W_4(\mathbf{r}_p) = 0 \end{aligned} \quad (2.6.64)$$

CIC (First-order)

In this case the charge from the particle p at position $\mathbf{r}_p \equiv (x_p, y_p, z_p) \in \Omega_{1234}$ is assigned

to the four nodes of the Delauney cell. From the concept of the charge conservation it follows that

$$W_1(r_p) + W_2(r_p) + W_3(r_p) + W_4(r_p) = 1 \quad (2.6.65)$$

The first-order constraints equations are

$$\begin{aligned} W_1(r_p)(x_p - x_1) + W_2(r_p)(x_p - x_2) + W_3(r_p)(x_p - x_3) + \\ + W_4(r_p)(x_p - x_4) = 0 \end{aligned} \quad (2.6.66)$$

$$\begin{aligned} W_1(r_p)(y_p - y_1) + W_2(r_p)(y_p - y_2) + W_3(r_p)(y_p - y_3) + \\ + W_4(r_p)(y_p - y_4) = 0 \end{aligned} \quad (2.6.67)$$

$$\begin{aligned} W_1(r_p)(z_p - z_1) + W_2(r_p)(z_p - z_2) + W_3(r_p)(z_p - z_3) + \\ + W_4(r_p)(z_p - z_4) = 0 \end{aligned} \quad (2.6.68)$$

Solution to the above system of constraints provides the weight functions W_1 in the following form

$$W_1 = \frac{\Omega_{P234}}{\Omega_{1234}} \quad (2.6.69)$$

where Ω_{P234} is the volume of the tetrahedron formed by the particle p and nodes 1,2,3 given by

$$\begin{aligned} \Omega_{P234} = & -x_4y_3z_2 + x_py_3z_2 + x_3y_4z_2 - x_py_4z_2 - x_3y_pz_2 + \\ & + x_4y_pz_2 + x_4y_2z_3 - x_py_2z_3 - x_2y_4z_3 + x_py_4z_3 + x_2y_pz_3 - \\ & - x_4y_pz_3 - x_3y_2z_4 + x_py_2z_4 + x_2y_3z_4 - x_py_3z_4 - x_2y_pz_4 + \\ & + x_3y_pz_4 + x_3y_2z_p - x_4y_2z_p - x_2y_3z_p + x_4y_3z_p + x_2y_4z_p + x_3y_4z_p \\ = & \frac{1}{6} [\mathbf{r}_{4p} \cdot (\mathbf{r}_{p2} \times \mathbf{r}_{23})] \end{aligned} \quad (2.6.70)$$

and the volume of the Delauney cell is

$$\begin{aligned}
\Omega_{1234} &= x_3y_2z_1 - x_4y_2z_1 - x_2y_3z_1 + x_4y_3z_1 + x_2y_4z_1 - \\
&\quad x_3y_4z_1 - x_3y_1z_2 + x_4y_1z_2 + x_1y_3z_2 - x_4y_3z_2 - x_1y_4z_2 + \\
&\quad x_3y_4z_2 + x_2y_1z_3 - x_4y_1z_3 - x_1y_2z_3 + x_4y_2z_3 + x_1y_4z_3 - \\
&\quad x_2y_4z_3 - x_2y_1z_4 + x_3y_1z_4 + x_1y_2z_4 - x_3y_2z_4 - x_1y_3z_4 + x_2y_3z_4 \\
&= \frac{1}{6} [\mathbf{r}_{41} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]
\end{aligned} \tag{2.6.71}$$

The remaining weight functions may be expressed in the similar fashion.

$$W_2 = \frac{\Omega_{p134}}{\Omega_{1234}} = \frac{\frac{1}{6} [\mathbf{r}_{14} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{13})]}{\frac{1}{6} [\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \tag{2.6.72}$$

$$W_3 = \frac{\Omega_{p124}}{\Omega_{1234}} = \frac{\frac{1}{6} [\mathbf{r}_{14} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{12})]}{\frac{1}{6} [\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \tag{2.6.73}$$

$$W_4 = \frac{\Omega_{p123}}{\Omega_{1234}} = \frac{\frac{1}{6} [\mathbf{r}_{13} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{12})]}{\frac{1}{6} [\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \tag{2.6.74}$$

Implementation of the CIC weights on the 3-D unstructured tetrahedral grid is therefore analogous to a volume-weighting and is represented graphically in Figure 9.

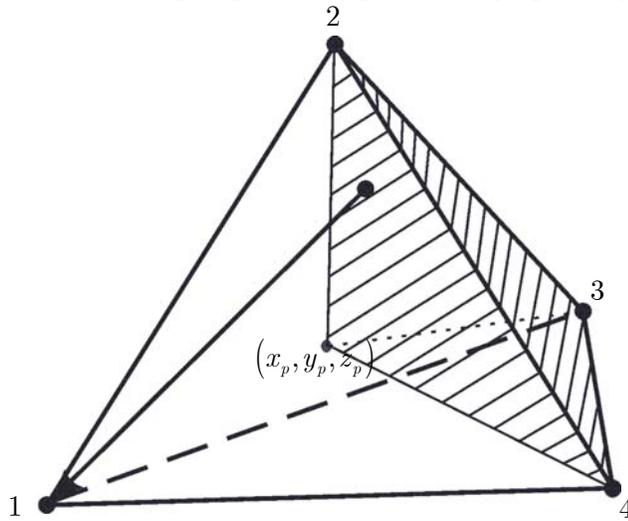


Figure 9. Graphical representation of the CIC weighting on the 3-D unstructured tetrahedral grid.

The number density and charge density at a node $r_1 = (x_1, y_1, z_1)$ from N_s particles of species s are evaluated using the volume of the three dimensional Voronoi cell Γ_1 as

$$n_s(x_1, y_1, z_1) \equiv n_{s1} = \frac{1}{\Gamma_1} \sum_{p=1}^{N_s} W_1(x_{ps}, y_{ps}, z_{ps}) \quad (2.6.75)$$

$$\rho_s(x_1, y_1, z_1) \equiv \rho_{s1} = \frac{1}{\Gamma_1} \sum_{p=1}^{N_s} q_s W_1(x_{ps}, y_{ps}, z_{ps}) \quad (2.6.76)$$

2.7 Solution of Poisson's Equation and Electric Potential Evaluation. Finite Volume Formulation.

The finite volume method separates the domains into discrete control volumes. In each control volume appropriate differential equation is discretized. If possible, integrals over volume involving gradients are transformed into integrals over surfaces using the divergence theorem. Similarly, integrals over surfaces may be transformed into integrals around closed contours and back using Stokes's theorem. The order of error for a finite element formulation is given as the highest derivative kept from a Taylor series expansion. A second order accurate expression for the two evenly spaced points about the point of interest I is

$$\frac{\partial u}{\partial x} = \frac{u_{I+1} - u_{I-1}}{2\Delta x} + O(\Delta x)^2 \quad (2.7.1)$$

Using more points in the Taylor series or reducing the spacing values may increase accuracy of the solution.

In this work, advantage is taken of the Voronoi dual of the Delaunay triangulation in order to formulate a finite volume method for Poisson's equation with accuracy adequate for engineering calculations. First, Voronoi cell corresponding to each Delaunay node

contains the set of points closer to that point than any other. Second, The facets of the Voronoi cell are orthogonal to the lines joining the tetrahedral nodes.

The expressions for electrostatic conditions can be obtained from two of Maxwell's equations

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} = 0 \quad (2.7.2)$$

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \quad (2.7.3)$$

In the electrostatic approximation the electric field may be expressed as negative gradient of the scalar potential

$$\mathbf{E} = -\nabla \Phi \quad (2.7.4)$$

The Poisson equation can be now obtained as

$$\nabla^2 \Phi = -\frac{\rho}{\varepsilon_0} \quad (2.7.5)$$

For a node-centered finite-volume scheme with finite volume associated with a node I with a number of corresponding faces N_f the semi-discrete form of Gauss's law is

$$\sum_{k=1}^{N_f} \mathbf{E} \cdot \mathbf{A}_{I,k} = \frac{Q_I}{\varepsilon_0} \quad (2.7.6)$$

where $A_{I,k}$ is the face area, Q_I is the total charge enclosed by the volume associated with node I and summation is over all the faces of the finite volume Γ_I (example for 2-D case is shown in Figure 10).

Using the potential Φ and the definition of the gradient

$$\nabla \Phi \cdot \hat{\mathbf{n}} = \frac{\partial \Phi}{\partial n} \quad (2.7.7)$$

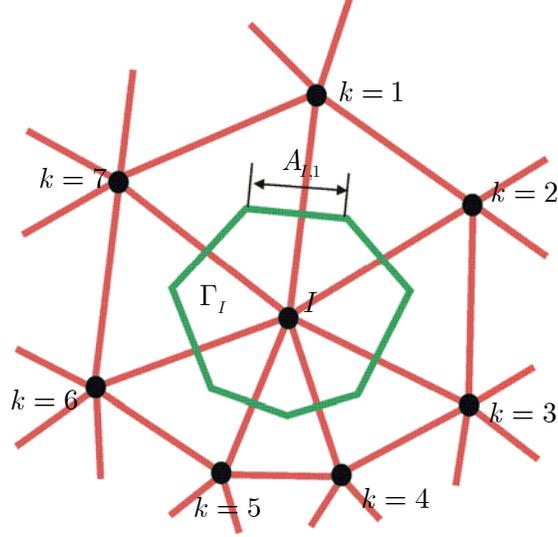


Figure 10. Example of the 2-D unstructured grid and Voronoi volume associated with the node I .

equation (2.7.6) becomes

$$\sum_{k=1}^{N_f} \nabla \Phi \cdot (\hat{\mathbf{n}} A)_{I,k} = \sum_{k=1}^{N_f} \left(A \frac{\partial \Phi}{\partial n} \right)_{I,k} = -\frac{Q_I}{\epsilon_0} \quad (2.7.8)$$

Using the favorable characteristics of the Voronoi dual the derivative at the faces can be obtained from the central difference method

$$\left. \frac{\partial \Phi}{\partial n} \right|_I = \frac{\Phi_k - \Phi_{I_0}}{L} + O(h^2) \quad (2.7.9)$$

Here, I_0 is equivalent to the $I - 1$ in equation (2.7.1) and denotes the number of the node in the Voronoi cell, k is the number of the node at the opposite end of the edge of length L .

An electric flux into a cell Γ_I across the Voronoi face of the edge with nodes I and k is

$$\sum_{k=1}^{N_f} \nabla \Phi \cdot (\hat{\mathbf{n}} A)_{I,k} = \frac{A_{I,k}}{L_{I,k}} (\Phi_k - \Phi_I) \quad (2.7.10)$$

A system of linear equations is formed by summing over all faces of the Voronoi cell corresponding to the node I

$$\sum_{k=1}^{N_f} (\Phi_I - \Phi_k) \frac{A_{I,k}}{L_{I,k}} = \frac{Q_I}{\varepsilon_0} \quad (2.7.11)$$

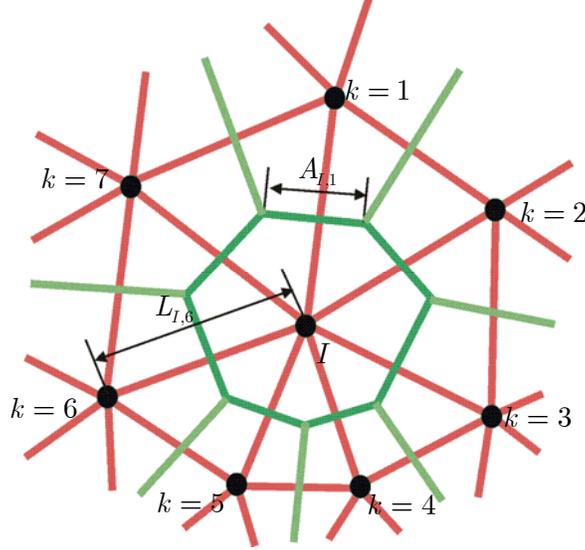


Figure 11. Example of the Delaunay mesh and the Voronoi dual in the 2-D case.

The geometrical variables used in the equation (2.7.11) are shown in the Figure 11.

In matrix form this equation becomes

$$\begin{bmatrix} R_{1,1} & R_{1,2} & R_{1,3} & \cdots & R_{1,N} \\ R_{2,1} & R_{2,2} & R_{2,3} & \cdots & R_{2,N} \\ R_{3,1} & R_{3,2} & R_{3,3} & \cdots & R_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{N,1} & R_{N,2} & R_{N,3} & \cdots & R_{N,N} \end{bmatrix} \begin{Bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \vdots \\ \Phi_N \end{Bmatrix} = \frac{1}{\varepsilon_0} \begin{Bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ \vdots \\ Q_N \end{Bmatrix} \quad (2.7.12)$$

Here N is the number of the mesh points, $R_{I,J}$ are the coefficients which are equal to

$$R_{I,J} = \sum_{k=1}^{N_I} \frac{A_{I,k}}{L_{I,k}} \quad \text{for } I = J, \quad (2.7.13)$$

$$R_{I,J} = -\frac{A_{I,J}}{L_{I,J}} \quad \text{if node } J \text{ is adjacent to } I, \quad (2.7.14)$$

$$R_{I,J} = 0 \quad \text{otherwise.} \quad (2.7.15)$$

2.8 Electric Field Evaluation

Evaluation of the electric fields on the nodes of a computational domain is an essential part of the PIC algorithm. Knowing the electric field it is possible to obtain force acting on the computational particles and thus their acceleration and velocity.

Three methods of evaluating the gradient of the potential are implemented in our code: two algorithms that utilize the divergence theorem with a cell-centered or node centered control volumes and the least square algorithm.

For a volume V surrounded by a surface S composed of n faces, the divergence theorem may be expressed as

$$\int_V \nabla \cdot \mathbf{Y} dV = \oint_S \mathbf{Y} d\mathbf{A} \quad (2.8.1)$$

where Y is a vector and $d\mathbf{A}$ is an outward normal differential surface element. Expressed in the terms of a gradient this equation becomes:

$$\int_V \nabla \cdot (\Phi \mathbf{k}) dV = \oint_S \Phi \mathbf{k} d\mathbf{A} \quad (2.8.2)$$

Since parameter \mathbf{k} is constant, equation (2.8.2) becomes

$$\int_V \nabla \Phi dV = \oint_S \Phi d\mathbf{A} \quad (2.8.3)$$

If the gradient varies slightly over the control volume and the potential is constant for a given face then a discrete formulation of (2.8.3) will be

$$\nabla\Phi = \frac{1}{V} \sum_{f=1}^n \Phi_f \mathbf{S}_f \quad (2.8.4)$$

where \mathbf{S}_f is a normal vector of the face f with the magnitude equal to the area of the face and Φ_f is the potential at that face.

In the case a of cell-centered control volume, the volume of the computational cell Ω is used. For the node-centered method, volume of all tetrahedra, which share the node of interest as shown in Figure 12 is used.

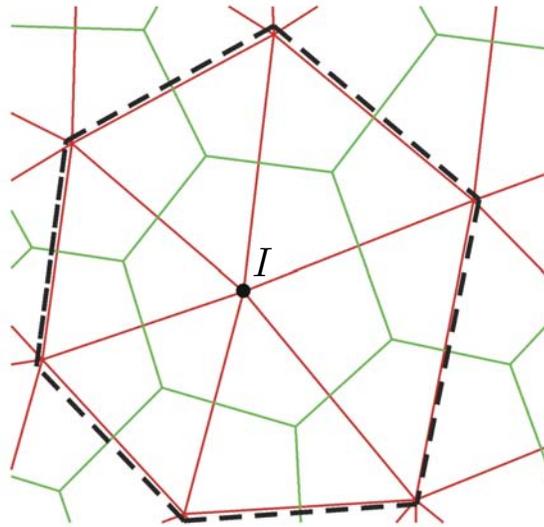


Figure 12. A control volume used in the node-centered method of the electric field evaluation.

The least square algorithm is implemented using nodes that share an edge of the Delaunay mesh with the node of interest. Locally linear variation of the potential is assumed to be

$$\Phi_I + \nabla\Phi \cdot \Delta\mathbf{r} = \Phi_k \quad (2.8.5)$$

where $\Delta\mathbf{r}$ is the vector from node I to node k . For a point with m neighbors above equation may be written in Cartesian coordinates as

$$\Delta x_k \left. \frac{\partial\Phi}{\partial x} \right|_I + \Delta y_k \left. \frac{\partial\Phi}{\partial y} \right|_I + \Delta z_k \left. \frac{\partial\Phi}{\partial z} \right|_I = \Phi_k - \Phi_I \quad (2.8.6)$$

or in matrix form

$$\mathbf{M}\mathbf{d} = \Delta\Phi \quad (2.8.7)$$

where \mathbf{M} is the $m \times 3$ matrix

$$\mathbf{M} = \begin{bmatrix} \Delta x_1 & \Delta y_1 & \Delta z_1 \\ \Delta x_2 & \Delta y_2 & \Delta z_2 \\ \vdots & \vdots & \vdots \\ \Delta x_m & \Delta y_m & \Delta z_m \end{bmatrix} \quad (2.8.8)$$

and \mathbf{d} is the array containing components of the gradient at node I .

This linear system contains m equations and three unknowns. Usually it will be an over-determined system, which means that it is necessary to search for a solution that fits this data in the best possible way.

It is possible to find a solution that minimizes the mean square root value of the error. The error for the point k is given by

$$R_k = \Delta x_k \left. \frac{\partial\Phi}{\partial x} \right|_I + \Delta y_k \left. \frac{\partial\Phi}{\partial y} \right|_I + \Delta z_k \left. \frac{\partial\Phi}{\partial z} \right|_I - (\Phi_k - \Phi_I) \quad (2.8.9)$$

The square of this error over all neighbors of I is

$$R^2 = \sum_k R_k^2 = \sum_k \left[\Delta x_k \left. \frac{\partial\Phi}{\partial x} \right|_I + \Delta y_k \left. \frac{\partial\Phi}{\partial y} \right|_I + \Delta z_k \left. \frac{\partial\Phi}{\partial z} \right|_I - (\Phi_k - \Phi_I) \right]^2 \quad (2.8.10)$$

We want to find the derivatives of the potential that will minimize the error, which means that we need to set the derivatives of R to zero

$$\left\{ \begin{array}{l} \frac{\partial R}{\partial \left(\frac{\partial \Phi}{\partial x} \Big|_I \right)} = 0 \\ \frac{\partial R}{\partial \left(\frac{\partial \Phi}{\partial y} \Big|_I \right)} = 0 \\ \frac{\partial R}{\partial \left(\frac{\partial \Phi}{\partial z} \Big|_I \right)} = 0 \end{array} \right. \quad (2.8.11)$$

The same set of equations may be obtained by multiplying equation (2.8.8) by the transpose of \mathbf{M}

$$\mathbf{M}^T \mathbf{M} \mathbf{d} = \mathbf{M}^T \Delta \Phi \quad (2.8.12)$$

This is a set of three equations, and three unknowns that is solved using a Cramer's rule.

It should be noted that methods described above, applied to the unstructured grids, do not have the same accuracy as they would in the case of structured Cartesian meshes. They are also sensitive to the grid quality.

2.9 Integration of the Equations for Particle Motion

The trajectory of a particle is described by

$$\frac{d\mathbf{c}(t)}{dt} = \frac{q_s}{m_s} [\mathbf{E}(\mathbf{r}, t) + \mathbf{c} \times \mathbf{B}_{\text{ext}}(\mathbf{r}, t)] \quad (2.9.1)$$

$$\frac{d\mathbf{r}(t)}{dt} = \mathbf{c}(t) \quad (2.9.2)$$

Equation (2.9.1) is integrated following *Buneman's* (1967) leap-frog formulation indicated in Figure 13.

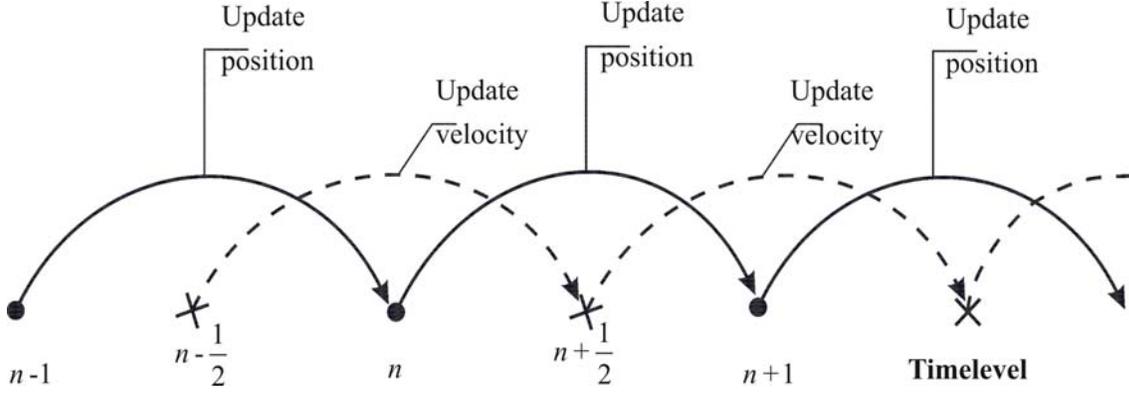


Figure 13. The leap-frog integration scheme.

The discretized form of the equation (2.9.1) is

$$\frac{\mathbf{c}(t + \Delta t/2) - \mathbf{c}(t - \Delta t/2)}{\Delta t} = \frac{q_p}{m_p} (\mathbf{E}(\mathbf{r}, t) + \frac{\mathbf{c}(t + \Delta t/2) - \mathbf{c}(t - \Delta t/2)}{2} \times \mathbf{B}_{\text{ext}}(\mathbf{r}, t)) \quad (2.9.3)$$

$$\mathbf{r}(t + \Delta t) = \mathbf{c}(t + \Delta t/2)\Delta t + \mathbf{r}(t) \quad (2.9.4)$$

The methodology discussed here is adopted from *Birdsall and Lagdon*, (1991) following a method developed by *Boris*, (1970). Let

$$\mathbf{c}(t - \Delta t/2) = \mathbf{c}^- - \frac{q\mathbf{E}(t)}{m} \frac{\Delta t}{2} \quad (2.9.5)$$

$$\mathbf{c}(t + \Delta t/2) = \mathbf{c}^+ + \frac{q\mathbf{E}(t)}{m} \frac{\Delta t}{2} \quad (2.9.6)$$

Substitution of equations (2.9.5) and (2.9.6) into (2.9.3) gives

$$\frac{\mathbf{c}^+ - \mathbf{c}^-}{\Delta t} = \frac{q}{2m} (\mathbf{c}^+ + \mathbf{c}^-) \times \mathbf{B}_{\text{ext}}(t) \quad (2.9.7)$$

Taking the dot product with $\mathbf{c}^+ + \mathbf{c}^-$ we obtain

$$|\mathbf{c}^-|^2 = |\mathbf{c}^+|^2$$

The first algorithmic step is to obtain

$$\mathbf{c}^- = \mathbf{c}(t - \Delta t/2) + \frac{q\mathbf{E}(t)\Delta t}{2m} \quad (2.9.8)$$

Next is to perform a rotation according to (2.9.7) in order to obtain \mathbf{c}^+ . This is accomplished by defining a vector \mathbf{t} parallel to \mathbf{B}_{ext}

$$\mathbf{t} \equiv \frac{q\mathbf{B}_{\text{ext}}}{m} \frac{\Delta t}{2} \quad (2.9.9)$$

and using it along with equation (2.9.8) to produce \mathbf{c}'

$$\mathbf{c}' = \mathbf{c}^- + \mathbf{c}^- \times \mathbf{t} \quad (2.9.10)$$

The vector \mathbf{c}' is perpendicular to $\mathbf{c}^+ - \mathbf{c}^-$ and \mathbf{B}_{ext} . The angle between \mathbf{c}^- and \mathbf{c}' is and

$$\mathbf{t} = \frac{q\mathbf{B}_{\text{ext}}}{m} \frac{\Delta t}{2} \quad (2.9.11)$$

Since $\mathbf{c}^+ - \mathbf{c}^-$ is parallel to $\mathbf{c}' \times \mathbf{B}_{\text{ext}}$, \mathbf{c}^+ can be found from

$$\mathbf{c}^+ = \mathbf{c}^- + \mathbf{c}' \times \mathbf{s} \quad (2.9.12)$$

where \mathbf{s} is parallel to \mathbf{B}_{ext} and its magnitude is determined by the requirement that the square of the velocities (kinetic energy) is unchanged by the rotation according to i.e.

$|\mathbf{c}^-|^2 = |\mathbf{c}^+|^2$. Therefore,

$$\mathbf{s} = \frac{2\mathbf{t}}{1 + t^2} \quad (2.9.13)$$

The magnitude of the angle of rotation can be evaluated from construction of the vectors \mathbf{c}^+ and \mathbf{c}^- according to

$$\left| \tan \frac{\theta}{2} \right| = \frac{|\mathbf{c}_\perp^+ - \mathbf{c}_\perp^-|}{|\mathbf{c}_\perp^+ + \mathbf{c}_\perp^-|} = \frac{q|\mathbf{B}_{\text{ext}}|\Delta t}{m} \frac{\Delta t}{2} = \frac{w_p \Delta t}{2} \quad (2.9.14)$$

The velocity components in this equation are perpendicular to the magnetic field and θ is the angle between the velocity vectors.

Finally, the second half of the electric impulse is added to \mathbf{c}^+ to obtain $\mathbf{c}^{t+\Delta t/2}$ following equation (2.9.6) and the particle position at the new timestep is given by (2.9.4).

In order to derive a stability criterion for the leap-frog scheme equations (2.9.3) and (2.9.4) may be written for the three-dimensional case as (assuming electrostatic simulations)

$$\begin{cases} c_1(t + \Delta t/2) = c_1(t - \Delta t/2) + w_p^2 \Delta t x(t) \\ c_2(t + \Delta t/2) = c_2(t - \Delta t/2) + w_p^2 \Delta t y(t) \\ c_3(t + \Delta t/2) = c_3(t - \Delta t/2) + w_p^2 \Delta t z(t) \\ x(t+1) = c_1(t - \Delta t/2) \Delta t + x(t) (1 - w_p^2 \Delta t^2) \\ y(t+1) = c_2(t - \Delta t/2) \Delta t + y(t) (1 - w_p^2 \Delta t^2) \\ z(t+1) = c_3(t - \Delta t/2) \Delta t + z(t) (1 - w_p^2 \Delta t^2) \end{cases} \quad (2.9.15)$$

A matrix form of the equations (2.9.15) is

$$\begin{bmatrix} c_1(t + \Delta t/2) \\ c_2(t + \Delta t/2) \\ c_3(t + \Delta t/2) \\ x(t + \Delta t) \\ y(t + \Delta t) \\ z(t + \Delta t) \end{bmatrix} = G \begin{bmatrix} c_1(t - \Delta t/2) \\ c_2(t - \Delta t/2) \\ c_3(t - \Delta t/2) \\ x(t) \\ y(t) \\ z(t) \end{bmatrix} \quad (2.9.16)$$

where amplification matrix G is

$$G = \begin{bmatrix} 1 & 0 & 0 & -w_p^2 \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 & -w_p^2 \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 & -w_p^2 \Delta t \\ \Delta t & 0 & 0 & 1 - w_p^2 \Delta t^2 & 0 & 0 \\ 0 & \Delta t & 0 & 0 & 1 - w_p^2 \Delta t^2 & 0 \\ 0 & 0 & \Delta t & 0 & 0 & 1 - w_p^2 \Delta t^2 \end{bmatrix} \quad (2.9.17)$$

The eigenvalues of G are

$$\lambda_{1,2,3} = \frac{1}{2} \left(2 - w_p^2 \Delta t^2 + w_p \Delta t \sqrt{w_p^2 \Delta t^2 - 4} \right) \quad (2.9.18)$$

$$\lambda_{4,5,6} = \frac{1}{2} \left(2 - w_p^2 \Delta t^2 - w_p \Delta t \sqrt{w_p^2 \Delta t^2 - 4} \right) \quad (2.9.19)$$

A scheme is stable if the eigenvalues of the amplification matrix lie on or within the unit circle

$$|\lambda|_{\max} \leq 1 \quad (2.9.20)$$

Following (2.9.20) the stability criterion for the leap-frog scheme in 3-D case is

$$w_p \Delta t < 2 \quad (2.9.21)$$

2.10 Particle Search-Locate Algorithm (Particle Tracer)

Particles are moved between adjacent tetrahedrons using a particle-tracing technique. Three types of algorithms are commonly used on unstructured grids to find the host cell of a computational particle (*Lonher (1990)*). Algorithms that use a Cartesian background grid superimpose an unstructured grid on a Cartesian background grid. Cells of the unstructured grid covering elements of the regular grid are stored. First, the number of the Cartesian cell (i_c, j_c, k_c) containing the new position of the particle position. The next step of the tracing algorithm is to check all elements of the unstructured grid to find a new host element for the particle. Though this method is easy to implement, it is inaccurate when meshes with significant variations in cell sizes are used. Tree-structure search-locate algorithms are an extension of Cartesian background algorithms. By using a hierarchy of Cartesian meshes, they can be used on complex unstructured grids with large cell size

variations. The successive-neighbor search algorithm is used for particle tracing in our Delauney PIC implementation because it is fast and comparatively easy to implement. Also, performance of successive neighbor search algorithms does not suffer for unstructured meshes with greatly varying cells sizes. Successive-neighbor search algorithms are based on the idea that the new particle position is not many cells away from the original cell. This assumption arises from restrictions of stability and accuracy of the PIC simulations as described in Chapter 2. Therefore only cells surrounding the original particle owner should be searched.

Following Figure 14, the particle with $\mathbf{r}_p(t)$ resides in a Delauney cell with volume Ω_{1234} . At $t + \Delta t$ the particle moves to position $\mathbf{r}_p(t + \Delta t)$ and the first step of the algorithm is to evaluate volume-weighted functions N_1, N_2, N_3, N_4 .

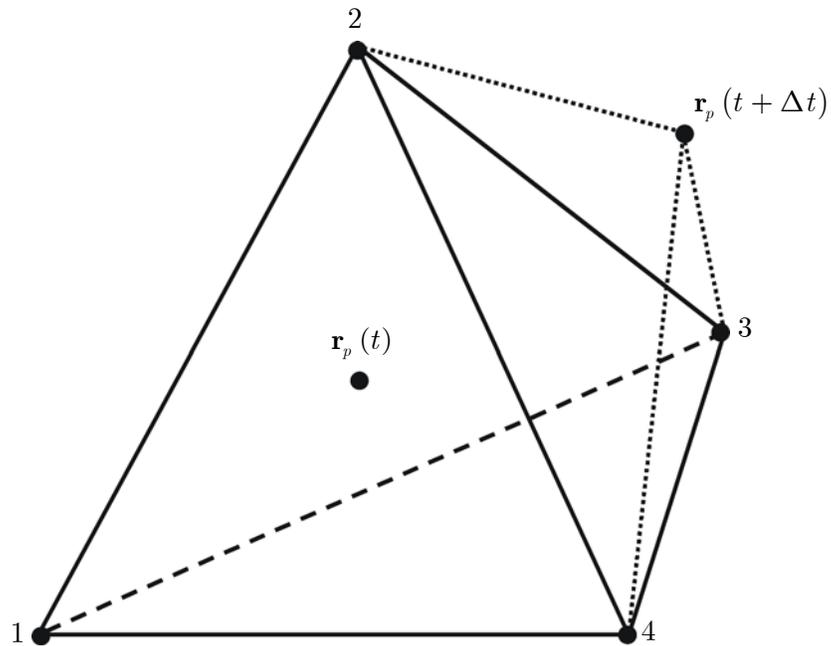


Figure 14. Tetrahedral computational cell and the possible position of the particle at time $t + \Delta t$.

These functions are evaluated from the new particle position $\mathbf{r}_p(t + \Delta t)$ with respect to the nodes of the Delaney cell of origin as

$$N_1(\mathbf{r}_p(t + dt)) = \frac{\Omega_{p234}}{\Omega_{1234}} = \frac{\frac{1}{6}[\mathbf{r}_{24} \cdot (\mathbf{r}_{2p} \times \mathbf{r}_{23})]}{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \quad (2.10.1)$$

where $\mathbf{r}_{p2} = \mathbf{r}_p(t + \Delta t) - \mathbf{r}_2$ etc.

Similarly,

$$N_2(\mathbf{r}_p(t + dt)) = \frac{\Omega_{p134}}{\Omega_{1234}} = \frac{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{13})]}{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \quad (2.10.2)$$

$$N_3(\mathbf{r}_p(t + dt)) = \frac{\Omega_{p124}}{\Omega_{1234}} = \frac{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{12})]}{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \quad (2.10.3)$$

$$N_4(\mathbf{r}_p(t + dt)) = \frac{\Omega_{p123}}{\Omega_{1234}} = \frac{\frac{1}{6}[\mathbf{r}_{13} \cdot (\mathbf{r}_{1p} \times \mathbf{r}_{12})]}{\frac{1}{6}[\mathbf{r}_{14} \cdot (\mathbf{r}_{12} \times \mathbf{r}_{23})]} \quad (2.10.4)$$

As seen from equations (2.10.1) to (2.10.4) $N_1 + N_2 + N_3 + N_4 = 1$ only in the case when $\mathbf{r}_p(t + dt) \in \Omega_{1234}$. If $N_1 + N_2 + N_3 + N_4 > 1$ then cells adjacent to Ω_{1234} should be searched.

The time and exact point of the intersection of the particle with the face of the tetrahedron are obtained through the solution of a system of linear equations involving two edges of the face, the current particle position and its velocity. A solution is obtained in a skewed coordinate system defined by the face edges. Intersection of a computational particle p with a face 1 – 2 – 3

$$\mathbf{r}_p(t) + \mathbf{c}_p(t)\tau = \alpha\mathbf{r}_{21} + \beta\mathbf{r}_{32} \quad (2.10.5)$$

In the equation (2.10.5) τ is a time it takes for the particle to move from its initial position to the face 1 – 2 – 3 and coefficients α and β define the point of intersection in the skewed coordinate system. If τ is negative or $1 < \alpha, \beta < 0$ then the intersection does not occur. If $\alpha + \beta > 1$ then the intersection occurs outside the face. This system of equations may be ill-conditioned if the cell is badly shaped or if the velocity of the particle is very large.

2.11 Electrostatic Boundary Conditions

Three types of boundary conditions are currently implemented: Dirichlet, Neumann and floating conductor. Forcing Dirichlet boundary conditions we set the voltage on simulation domain boundaries, while Neumann boundary conditions specify a normal derivative of electric field. According to *Jackson* (1999), a solution of Poisson equation may be specified uniquely by piece-wise continuous Dirichlet and Neumann boundary conditions.

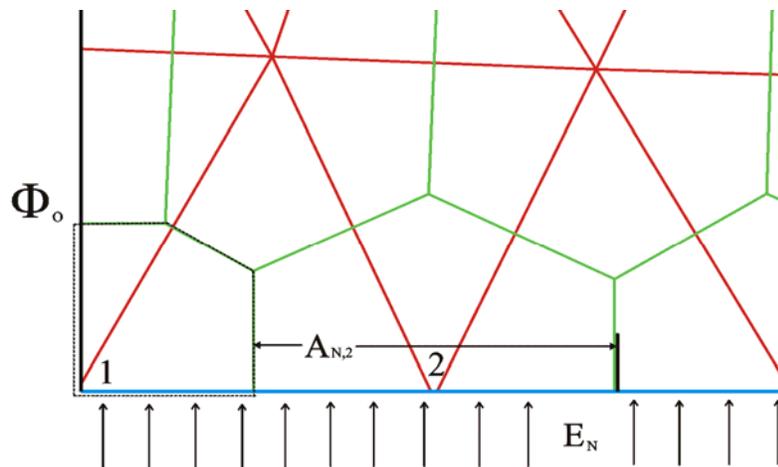


Figure 15. Graphical representation of the Dirichlet and Neumann boundary conditions (from *Hammel* (2002)).

An example, illustrating boundary condition implementation is shown in Figure 15. Potential Φ_0 is directly specified on the face of cell number 1. A voltage, assigned to the boundary faces by the Dirichlet condition is placed in the right hand side of the equation (2.7.12) and a corresponding row of coefficients is set to zero except the diagonal term, which is set to one.

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ R_{2,1} & R_{2,2} & R_{2,3} & \cdots & R_{2,N} \\ R_{3,1} & R_{3,2} & R_{3,3} & \cdots & R_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{N,1} & R_{N,2} & R_{N,3} & \cdots & R_{N,N} \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \vdots \\ \Phi_N \end{bmatrix} = \frac{1}{\varepsilon_0} \begin{bmatrix} \Phi_0 \\ Q_2 \\ Q_3 \\ \vdots \\ Q_N \end{bmatrix} \quad (2.11.1)$$

A Neumann boundary condition is assigned to cell number 2. The value of the inward electric field multiplied by the face area is added to the corresponding row of the matrix, for equation (2.11.1) to become

$$\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ R_{2,1} & R_{2,2} & R_{2,3} & \cdots & R_{2,N} \\ R_{3,1} & R_{3,2} & R_{3,3} & \cdots & R_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_{N,1} & R_{N,2} & R_{N,3} & \cdots & R_{N,N} \end{bmatrix} \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \vdots \\ \Phi_N \end{bmatrix} = \frac{1}{\varepsilon_0} \begin{bmatrix} \Phi_0 \\ Q_2 + \varepsilon_0 E_{N,2} A_{N,2} \\ Q_3 \\ \vdots \\ Q_N \end{bmatrix} \quad (2.11.2)$$

Dirichlet boundary conditions have precedence over Neumann boundary conditions in the case when these two conditions interfere.

Simulations of bounded plasmas often involve different external circuit elements. In this case the solution of the Poisson equation within the plasma must be coupled with external circuit equations.

Following *Vahedi and DiPeso* (1997), time variation of the total charge density on the driven electrode may be evaluated from the Kirchhoff's current loop law as

$$A \frac{d\sigma_T}{dt} = I(t) + Aj_{conv} \quad (2.11.3)$$

where σ_T is a total charge density, $I(t)$ is the external circuit current, A is an electrode surface area and j_{conv} is a convective current density at the electrode due to the plasma. A discrete form of equation (2.11.3) will be

$$A(\sigma_T(t) - \sigma_T(t - \Delta t)) = Q(t) - Q(t - \Delta t) + Q_{conv}(t) \quad (2.11.4)$$

In the above Q is a charge on the capacitor and Q_{conv} is a charge collected by the electrode for the time Δt due to charged particles from the plasma.

A floating potential is a specific case of the conducting boundary with no charge exchange via an external circuit. Under this assumption all terms of equation (2.11.4) that include a capacitor charge will be equal to zero to give

$$A(\sigma_T(t) - \sigma_T(t - \Delta t)) = Q_{conv}(t) \quad (2.11.5)$$

In order to calculate the total charge on the boundary at a specific time we will use a semi-discrete Gauss's law, which can be obtained from the discretization of equation (2.7.11) as

$$\varepsilon_0 \left(E_{N,I} A_{N,I} + \sum_k (\Phi_I - \Phi_k) \frac{A_{I,k}}{L_{I,k}} \right) = Q_{I(plasma)} + \sigma_I A_{I(boundary)} \quad (2.11.6)$$

In equation (2.11.6) an outward electric flux into the Voronoi volume corresponding to the node I is assigned by the $E_{N,I} A_{N,I}$ term, $A_{I(boundary)}$ is a boundary area associated with a node I and $Q_{I(plasma)}$ is the total charge at this node due to the plasma. Summing equation (2.11.6) for all boundary nodes I we will get

$$\sum_I \sigma_I(t) A_{I(boundary)} = \sigma_T(t) \sum_I A_{I(boundary)} = \sum_I \left(\varepsilon_0 \left(E_{N,I}(t) A_{N,I} + \sum_k (\Phi_I(t) - \Phi_k(t)) \frac{A_{I,k}}{L_{I,k}} \right) - Q_{I(plasma)}(t) \right) \quad (2.11.7)$$

Using (2.11.5), equation (2.11.7) becomes

$$A\sigma_T(t) = A\sigma_T(t - \Delta t) + Q_{conv}(t) = \sum_I \left(\epsilon_0 \left(E_{N,I}(t) A_{N,I} + \sum_k (\Phi_I(t) - \Phi_k(t)) \frac{A_{I,k}}{L_{I,k}} \right) - Q_{I(plasma)}(t) \right) \quad (2.11.8)$$

If all boundary nodes have the same constant potential Φ_0 than this potential at a specified time can be found from equation (2.11.8) as

$$\Phi_0(t) = \frac{\frac{1}{\epsilon_0} (A\sigma_T(t - \Delta t) + Q_{conv}(t) + Q_{I(plasma)}(t)) + \sum_p \left(-E_{N,I}(t) A_{N,I} + \sum_k \Phi_k(t) \frac{A_{I,k}}{L_{I,k}} \right)}{\sum_I \sum_k \frac{A_{I,k}}{L_{I,k}}} \quad (2.11.9)$$

From the principle of linear superposition, the total electrostatic potential may be considered as the sum of the potential due to the plasma charge, the imposed electric field and the electrodes.

$$\Phi(t) = \Phi_{plasma}(t) + \Phi_{E-field}(t) + \Phi_{electrodes}(t) \quad (2.11.10)$$

Potential on the driven electrode may be normalized in such a way that field at specific time due to potential drop from that electrode may be calculated as the initial potential multiplied by a constant.

$$\Phi_I(t) = \Phi_0(t) \Phi_{NL,I} + \Phi_{I(plasma)}(t) \quad (2.11.11)$$

where $\Phi_{I(plasma)}(t)$ is the potential due to plasma and imposed electric field, $\Phi_0(t) \Phi_{NL,I}$ is the potential due to driven conductor and $\Phi_{NL,I}$ is normalized potential profile.

By substituting equation (2.11.11) into (2.11.9) a value for the potential Φ_0 at a specified time may be found.

$$\Phi_0(t) = \frac{\frac{1}{\varepsilon_0} \left(A\sigma_T(t - \Delta t) + Q_{conv}(t) + Q_{I(plasma)}(t) \right) + \sum_I \left(-E_{N,I}(t) A_{N,I} + \sum_k \Phi_{k(plasma)}(t) \frac{A_{I,k}}{L_{I,k}} \right)}{\sum_I \sum_k \frac{A_{I,k}}{L_{I,k}} (1 - \Phi_{NL,k})} \quad (2.11.12)$$

2.12 Particle Loading

In the beginning of every simulation involving some type of background plasma, computational domain should be populated with particles according to the local density. It is assumed that the distribution of particles inside each cell is uniform and the number of model particle inside the cell is calculated in accordance to the volume of the cell and total particle weight. Particles are randomly distributed within a cell.

Initial velocities of the particles are prescribed with the assumption of thermal equilibrium and independence of particle velocity components. Therefore, the distribution function for one component of thermal velocity is

$$f_i = \frac{\beta}{\sqrt{\pi}} \exp(-\beta^2 i^2), i = (C_1, C_2, C_3) \quad (2.12.1)$$

where $\beta = \sqrt{\frac{m}{2kT}}$ and C_1 , C_2 , and C_3 are the components of velocity in the x, y, and z directions. The structure of the distribution function can be used in the generation of the particle velocity vector. From a computational point of view, the most economical way to generate initial velocity in this case is sampling of C_1 , C_2 , and C_3 according to distribution function (2.12.1). Due to independence of the velocity components, the distribution function that describes C_1 , C_2 , and C_3 is only a product of type (2.12.1) functions

$$f_1 f_2 f_3 = \frac{\beta^3}{\pi} \exp(-\beta^2 (C_1^2 + C_2^2 + C_3^2)) \quad (2.12.2)$$

In a spherical coordinate system, this product can be rewritten as

$$f_u du f_v dv f_w dw = \frac{1}{2\pi} \exp(-\beta^2 r^2) d(\beta^2 r^2) d\theta d\phi \quad (2.12.3)$$

where r , θ , and ϕ are coordinates in the spherical system of reference. Transition between these coordinate systems is described as

$$\begin{aligned} C_1 &= r \cos(\theta) \sin(\phi) \\ C_2 &= r \sin(\theta) \sin(\phi) \\ C_3 &= r \cos(\phi) \end{aligned} \quad (2.12.4)$$

The advantage of changing to a spherical coordinate system is that the variables r , θ , and ϕ can be sampled much more easily than C_1 , C_2 , and C_3 . Equation (2.12.1) can be represented as product of three distribution functions for variables $\beta^2 r^2$, θ , and ϕ . As the Maxwellian velocity distribution function has no directional preference, θ is uniformly distributed between 0 and 2π , and ϕ is randomly distributed between 0 and π . This means that value of θ can be defined by

$$\theta = 2\pi R \quad (2.12.5)$$

where R is random number distributed uniformly between 0 and 1. Similarly, the value of ϕ may be found by

$$\phi = \pi R \quad (2.12.6)$$

The remaining part of equation (2.12.1) can be presented as distribution function for variable $\beta^2 r^2$

$$f_{\beta^2 r^2} = \exp(-\beta^2 r^2) \quad (2.12.7)$$

Using the standard inverse-cumulative method to sample from this distribution function, we get

$$r = \frac{\sqrt{-\ln(R)}}{\beta} \quad (2.12.8)$$

Transferring back into a Cartesian coordinate system gives values of thermal velocity components for an equilibrium gas.

2.13 Particle Injection

The injection routine is an important part of PIC simulations. It allows a plasma stream to enter a computational domain with prescribed initial parameters.

The number of particles to be added into the simulation can be evaluated on the base of analysis of molecular flux across a surface element. We will choose such a coordinate system where two of coordinate axes are in the injection plane as shown in Figure 16. In this case surface element lies in $y - z$ plane, and mean flow velocity of injected particles is in $x - y$ plane.

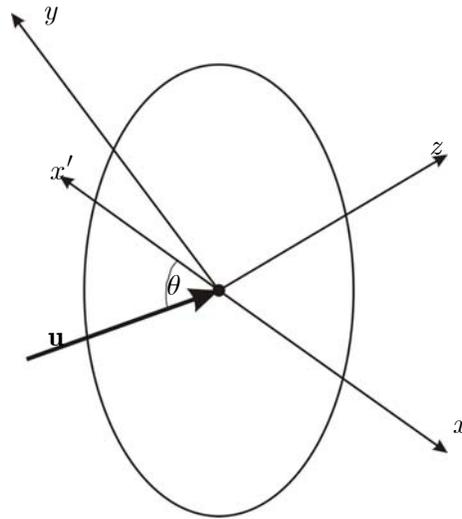


Figure 16. Coordinate system utilized in the particle injection algorithm.

The inward number flux \dot{N} can be defined by integration of the velocity distribution function

$$\dot{N}_s = n_s \frac{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_0^{+\infty} \mathbf{u} f du_1 du_2 du_3}{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f du_1 du_2 du_3} \quad (2.13.1)$$

In this coordinate system, particle velocity can be expressed in terms of mean flow velocity \mathbf{u} and thermal molecular velocity

$$\begin{cases} c_1 = C_1 + \mathbf{u} \cos(\theta) \\ c_2 = C_2 + \mathbf{u} \sin(\theta) \\ c_3 = C_3 \end{cases} \quad (2.13.2)$$

Equation (2.13.1) can be rewritten as

$$\dot{N}_s = n_s \frac{\beta^3}{\pi^{3/2}} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \int_{-\mathbf{u} \cos(\theta)}^{+\infty} (C_1 + \mathbf{u} \cos(\theta)) \exp(-\beta^2 (C_1^2 + C_2^2 + C_3^2)) dc_1 dc_2 dc_3 \quad (2.13.3)$$

After integration, this expression becomes (*Bird, (1998)*)

$$\dot{N}_s = \frac{n_s}{2\beta\sqrt{\pi}} \left(\exp(-s^2 \cos^2(\theta)) + \sqrt{\pi} s \cos(\theta) \{1 + \operatorname{erf}(s \cos(\theta))\} \right) \quad (2.13.4)$$

The value of \dot{N}_s can be interpreted as the number of gas molecules of the species of interest crossing a unit area surface element per unit time with mean flow velocity \mathbf{u} .

The number of model particles to be added to the simulation, ΔN_s , is given by

$$\Delta N_s = \frac{\dot{N}_s}{F_s} \Delta t A \quad (2.13.5)$$

where F_s is the computational weight of the particles of type s , Δt is the time step and A is the area of the surface element.

Once the number of new particles is known, initial parameters must be prescribed to them. Each model particle is characterized by position and velocity vectors. A position vector can be easily generated if we assume uniform distribution of new model particles over the surface element. Generation of a velocity vector is not so straightforward. We will use a flow injection model where normal component of velocity vectors of added particles are distributed due to distribution

$$f \propto (C_1 + \mathbf{u} \cos(\theta)) \exp(-\beta^2 C_1^2) \quad (2.13.6)$$

To apply standard acceptance-rejection method to the distribution, it is necessary to get maximum value of the distribution function. The standard approach used gives:

$$\frac{\partial f}{\partial C_1} \propto \exp(-\beta^2 C_1^2) [1 - 2\beta^2 C_1 (C_1 + \mathbf{u} \cos(\theta))] = 0 \quad (2.13.7)$$

Thermal velocity, which corresponds to the maximum value of the distribution function can be found as a solution of the quadratic equation

$$1 - 2\beta^2 C_1 (C_1 + \mathbf{u} \cos(\theta)) = 0 \quad (2.13.8)$$

which has two solutions

$$C_1 = \frac{-\beta^2 \mathbf{u} \cos(\theta) \pm \beta \sqrt{\beta^2 \mathbf{u}^2 \cos^2(\theta) + 2}}{2\beta^2} \quad (2.13.9)$$

Due to the choice of coordinate system, u' must be more than zero and the final result is given by

$$C_1 = \frac{\sqrt{\mathbf{u}^2 \cos^2(\theta) + 2/\beta^2} - \mathbf{u} \cos(\theta)}{2} \quad (2.13.10)$$

Taking into account the last expression, the ratio of probability to the maximum probability is given by

$$\frac{P}{P_{\max}} = \frac{2(C_1 + \mathbf{u} \cos(\theta))}{\sqrt{\mathbf{u}^2 \cos^2(\theta) + 2/\beta^2 - \mathbf{u} \cos(\theta)}} \times \exp\left(\frac{1}{2} + \frac{\beta^2 \mathbf{u} \cos(\theta)}{2} \left(\mathbf{u} \cos(\theta) - \sqrt{\mathbf{u}^2 \cos^2(\theta) + 2/\beta^2}\right) - \beta^2 C_1^2\right) \quad (2.13.11)$$

Tangential velocity components are generated fitting the equilibrium distribution function (2.12.1) using the same procedure that was used for the loading of particles.

After all necessary vectors are generated, the particle may be moved into computational domain.

2.14 Macroscopic Plasma Properties Evaluation

The species and total mean velocities of the particles in the direction α are

$$\mathbf{u}_{s\alpha I} = \frac{\sum_{p=1}^{N_{sI}} F_s \mathbf{c}_{p\alpha}}{n_{sI} \Gamma_I} \quad (2.14.1)$$

where N_{sI} is the total number of particles of s type in the Voronoi cell Γ_I .

The average temperature of the particles of type s at the node I can then be found using

$$\frac{3}{2} k T_s = \frac{1}{2} m_s \langle \mathbf{C}_s^2 \rangle \quad (2.14.2)$$

to be

$$T_{sI} = \frac{1}{3k} m_s \left(\frac{\sum_{p=1}^{N_{sI}} F_s c_p^2}{N_{sI} F_s} - \sum_{r=1}^3 \left(\frac{\sum_{p=1}^{N_{sI}} F_s c_{pr}}{N_{sI} F_s} \right)^2 \right) \quad (2.14.3)$$

In equation (2.14.3) $r = x, y, z$. The average temperature for the particles located in the Γ_I

will be

$$T_I = \sum_{s=1}^{NS} \frac{N_{sI}}{N_I} T_{sI} \quad (2.14.4)$$

The pressure p_f at the center of the face f from the collisions of plasma particles with the face can be defined as a momentum transported through a surface element as

$$p_f = \frac{\sum_{s=1}^{NS} \sum_{p=1}^{N_s} m_s F_s c_{pn}}{A_f t} \quad (2.14.5)$$

where N_s is the number of particles of type s which hit the face number f , $f = 1, NF$, during the timestep t , c_{pn} is the velocity of the particles in the direction n normal to the face f and A_f is the area of the face f .

The pressure coefficient c_{pf} then will be

$$c_{pf} = \frac{p_f - p_\infty}{q_\infty} \quad (2.14.6)$$

here p_∞ is the freestream pressure which is obtained by the ideal gas law

$$p_\infty = n_\infty k T_\infty \quad (2.14.7)$$

In (2.14.7) the density and temperature at infinity are taken to be equal to the injection parameters. In the equation (2.14.6) q_∞ is the freestream dynamic pressure

$$q_\infty = \frac{1}{2} \rho_\infty \mathbf{c}_\infty^2 \quad (2.14.8)$$

where

$$\rho_\infty = \sum_{s=1}^{NS} m_s n_{\infty s} \quad (2.14.9)$$

The skin coefficient c_{ff} is

$$c_{ff} = \frac{\tau_f}{q_\infty} \quad (2.14.10)$$

here τ_f is the shear stress

$$\tau_f = \frac{\sum_{s=1}^{NS} \sum_{p=1}^{N_{sf}} F_s c_{p\tau}}{A_f t} \quad (2.14.11)$$

Heat transfer coefficient c_h is calculated as follows

$$c_{hf} = \frac{\sum_{s=1}^{NS} \sum_{p=1}^{N_{sf}} \frac{1}{2} F_s m_s \left((\mathbf{c}_p \cdot \mathbf{c}_p) - (\mathbf{c}_{p0} \cdot \mathbf{c}_{p0}) \right)}{A_f t q_\infty \mathbf{c}_\infty} \quad (2.14.12)$$

where \mathbf{c}_p is the velocity of the particle p before it hits the face f and \mathbf{c}_{p0} is its velocity after the collision.

CHAPTER 3 SIMULATION OF CYLINDRICAL LANGMUIR PROBE IN COLLISIONLESS PLASMAS

In order to validate the code we apply it to the problem of current collection by cylindrical probes in collisionless, stationary and drifting, unmagnetized plasma operating in the Thin Sheath and the Orbital Motion Limited (OML) regimes. We are interested in measuring the electron and ion currents collected by the probe as a function of the probe potential for different background plasma parameters. Results are compared to the numerical predictions by *Laframboise (1966)* and analytical solutions by *Peterson and Talbot (1970)*, *Kanal (1964)*, and *Johnson and Murphree (1969)*. An overview of these three works is presented followed by numerical implementation and results.

3.1 Review of the Current Collection Theory

The theory of the current collection by the cylindrical Langmuir probes is developed for plasma consisting of electrons and single-species ions. The distribution function for a species $s \equiv e, i$ is a drifting Maxwellian,

$$f_s(\mathbf{r}, \mathbf{c}, t) = n_s(\mathbf{r}, t) \left(\frac{m_s}{2\pi k T_s} \right)^{3/2} \exp \left[- \frac{m_s (\mathbf{c}_s - \mathbf{u}_s)^2}{2k T_s} \right] \quad (3.1.1)$$

where m_s is the mass, $\mathbf{c}_s \equiv (c_1, c_2, c_3)_s$ is the velocity, $\mathbf{u}_s(\mathbf{r}, t) = \langle \mathbf{c}_s \rangle$ is the average (or drift) velocity (brackets indicate average over the species distribution function) and T_s is the temperature. The most probable thermal speed is $\sqrt{2kT_s/m_s}$ and the speed ratio is

$$S_s = \left(u_s / \sqrt{2kT_s/m_s} \right) \quad (3.1.2)$$

Theory requires that the cylindrical probes are operating in a collisionless plasma, therefore the Knudsen number based on the probe radius r_p must satisfy

$$Kn_{st} = \lambda_{st}/r_p \gg 1 \quad (3.1.3)$$

where $\lambda_{st} \equiv \lambda_{ei}, \lambda_{ee}, \lambda_{en}, \lambda_{ii}, \lambda_{in}$ is the mean free path for collisions between species s and t .

The probe theory requires that the sheath is collisionless, i.e.,

$$d_s \ll \lambda_{ei}, \lambda_{ee}, \lambda_{en}, \lambda_{ii}, \lambda_{in} \quad (3.1.4)$$

The sheath thickness of a probe at a potential Φ_p with respect to the plasma potential can be estimated by

$$d_s = \left(\sqrt{2}/3\right) \lambda_D \left(2e\Phi_p/kT_e\right)^{3/4} \quad (3.1.5)$$

where the Debye length is given by

$$\lambda_D = \sqrt{\epsilon_0 kT_e / e^2 n_e} \quad (3.1.6)$$

Laframboise (1966) developed a method to numerically predict a current collected by an electrically conducting Langmuir probe from the collisionless, stationary Maxwellian plasma. Solution was obtained for the wide range of the ion to electron temperatures T_i / T_e , probe potentials up to 25 times the thermal energy and probe radius to Debye length r_p / λ_D up to 100.

Equations (3.1.7)-(3.1.10) were solved numerically with an iterative numerical scheme using an extension of *Bernstein and Rabinowitz* (1959) method.

Vlasov equation:

$$\frac{df_s(\mathbf{r}, \mathbf{p})}{dt} = \frac{\partial f_s(\mathbf{r}, \mathbf{p})}{\partial \mathbf{r}} \cdot \mathbf{c} + \frac{\partial f_s(\mathbf{r}, \mathbf{p})}{\partial \mathbf{p}} \cdot \mathbf{F}_s = 0 \quad (3.1.7)$$

Poisson's equation:

$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0} \quad (3.1.8)$$

$$\mathbf{F}_s = -Z_s e \frac{\partial \phi}{\partial \mathbf{r}} \quad (3.1.9)$$

$$n_s(\mathbf{r}) = \int f_s(\mathbf{r}, \mathbf{c}) d^3 \mathbf{c} \quad (3.1.10)$$

In the above equations $f_s(\mathbf{r}, \mathbf{p})$ is the distribution function in position \mathbf{r} and momentum \mathbf{p} space and \mathbf{F}_s is the force due to electric field. Results that include potential distribution, charge density and probe current are presented in graphical and tabular form.

In *Peterson and Talbot (1970)* it was shown that for $5 \leq r_p / \lambda_D \leq 100$, $\chi_p > 3$ and $T_i / Z_i T_e \leq 1$ results produced by *Laframboise (1966)* could be reproduced by the following relation

$$I = I_0 (\beta + |\chi|)^\alpha \quad (3.1.11)$$

where I_0 is a random thermal current and χ_p is a non-dimensional potential

$$I_0 = \pi r_p L Z q N \sqrt{\frac{2kT}{\pi m}} \quad (3.1.12)$$

$$\chi_p = \frac{e(\Phi_p - \Phi_s)}{kT_e} \quad (3.1.13)$$

$$\alpha = \frac{a}{\ln\left(\frac{r_p}{\lambda_D}\right) + b} + c \left(\frac{\theta_a}{\theta_r}\right)^m + d \quad (3.1.14)$$

$$\beta = e + \left\{ f + g \left[\ln\left(\frac{r_p}{\lambda_D}\right) \right]^3 - \frac{l}{\ln\left(\frac{r_p}{\lambda_D}\right)} \right\} \frac{\theta_a}{\theta_r} + \frac{l}{\ln\left(\frac{r_p}{\lambda_D}\right)} \quad (3.1.15)$$

In equations (3.1.14) and (3.1.15) θ is an effective temperature, Φ_s is a space potential,

subscripts a and r refer to attracted and repelled particles respectively. Numerical constants a to m are given in a tabulated form for the cases of $\frac{\theta_a}{\theta_r} \leq 1$ and $\frac{\theta_a}{\theta_r} > 1$.

Kanal (1964) theoretically investigated a current collection by stationary and moving Langmuir cylindrical probes. *Kanal* assumed that the sheath around the probe has a cylindrical shape, though this assumption is not valid for the fast probes, and that any effect of the “presheath” is negligible. For the case of $V_0 = \frac{e\Phi_p}{kT_e} > 0$ (accelerated current)

the normalized current as a function of probe velocity and its orientation for the finite sheath thickness was found to be

$$I_{an} = \frac{2}{\sqrt{\pi}} e^{-k^2} \sum_{m=0}^{\infty} \frac{k^m}{m!} \left(e^{V_0} V_0^{-m/2} \Gamma \left(m + \frac{3}{2}, V_0 \left(1 + \frac{r_p^2}{a^2 - r_p^2} \right) \right) J_m (2k\sqrt{V_0}) + \frac{a}{r_p} \frac{k^m}{m!} \gamma \left(m + \frac{3}{2}, \frac{r_p^2}{a^2 - r_p^2} V_0 \right) \right) \quad (3.1.16)$$

In the above equation current is normalized with respect to the random current, Γ and γ are the incomplete gamma functions. In the case of stationary probe ($k = 0$), equation (3.1.16) becomes

$$I_{an}|_{k=0} = e^{V_0} \operatorname{erfc} \left(\sqrt{V_0} \left(1 + \frac{r_p^2}{a^2 - r_p^2} \right) \right) + \frac{a}{r_p} \operatorname{erf} \left(\sqrt{\frac{r_p^2}{a^2 - r_p^2}} \sqrt{V_0} \right) \quad (3.1.17)$$

When $\frac{a}{r_p} \rightarrow 1$ (“shear-area limited” or “Thin Sheath” case), all particles that enter the

sheath are collected by the probe

$$I_{an} = \frac{a}{r_p} \quad (3.1.18)$$

When $\frac{a}{r_p} \rightarrow \infty$ (“orbital motion limited” case), collected current does not depend upon $\frac{a}{r_p}$

and

$$I_{an} = \frac{2}{\sqrt{\pi}} \sqrt{V_0} + e^{V_0} \operatorname{erfc}(\sqrt{V_0}) \quad (3.1.19)$$

Johnson and Murphree (1969) developed an asymptotic expression for the ion current given by

$$I_{\perp i} = A_{\perp} n_e e \left(\frac{kT_e}{2\pi m_e} \right)^{\frac{1}{2}} \frac{2}{\sqrt{\pi}} \exp(-S_i^2) \sum_{n=0}^{\infty} \left[\frac{S_i}{n!} \right]^2 \Gamma\left(n + \frac{3}{2}\right) \quad (3.1.20)$$

In the above expression the sheath is of a negligible thickness, $d_s/r_p \ll 1$ and is independent of the applied potential. Therefore, the current collection area of the probe of length l_p becomes equal to its geometric area given by

$$A_{\perp} = 2\pi r_p l_p \quad (3.1.21)$$

3.2 Unstructured 3-D PIC Simulations of the Current Collection by a Cylindrical Probes

The PIC simulations represent the current collection by the cylindrical Langmuir probes from a stationary and drifting plasmas with parameters specified in Table 1. In the beginning of every simulation computational domain is loaded uniformly with electrons and ions following Maxwellian distribution function (3.1.1). Examples of the computational domains used in the current collection simulations are shown in Figure 17 and Figure 18. Computational particles that reach boundaries of the domain are removed from the simulation. Background particles are also injected at each time-step from the outside boundaries of the domain according to their thermal fluxes. Zero electric field is set at the open boundaries of the domain. A specified (positive or negative) potential is applied at the surface of the cylindrical probe. The size of the computational domain indicated by

the radius R_D in Table 1, was set so that the potential at the boundaries reaches zero, i.e. the unperturbed value of the space (or plasma) potential. Particles that reach the probe are removed from the simulation. The total charge carried by these particles is calculated at every time-step and collected ion and electron currents are measured.

Case #	1.1	1.2	2.1	2.2	3	4
Regime	Thin Sheath		Thin Sheath		OML	Thin Sheath
$n_e = n_i [1/m^3]$	10^{16}		10^{16}		10^{16}	10^{16}
$T_e [eV]$	2		2		2	2
T_i / T_e	1	0.1	1		1	1
$r_p [m]$	10^{-3}		10^{-3}	5×10^{-4}	10^{-5}	5×10^{-3}
$R_D [m]$	10^{-2}		10^{-2}	5×10^{-3}	10^{-2}	5×10^{-2}
r_p / λ_D	10		10	5	0.1	50
$e\Phi_p / kT_e$	-10 to +10		+2 to +9		+1 to +9	-2
S_i	0		0		0	1 to 7
Comparisons	<i>Laframboise (1966), Peterson and Talbot (1970)</i>		<i>Laframboise (1966), Peterson and Talbot (1970)</i>		<i>Laframboise (1966), Kanal (1964)</i>	<i>Johnson and Murphree (1969)</i>

Table 1. Input conditions and computational parameters for the current collection simulations.

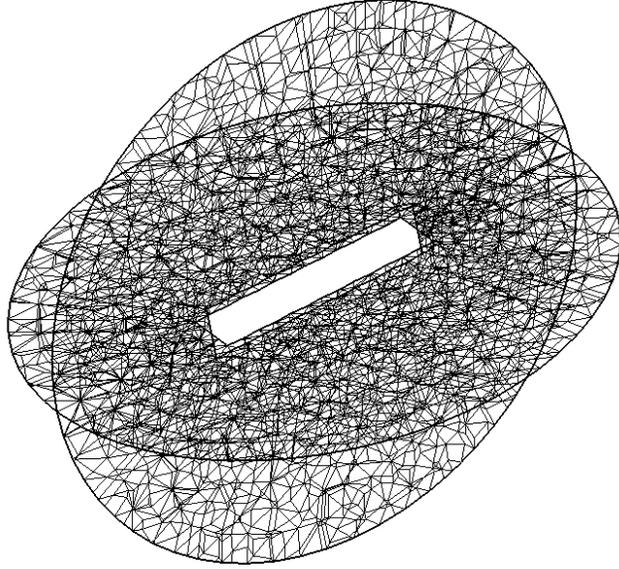


Figure 17. An example of the computational domain used in the simulations (Thin Sheath regime).

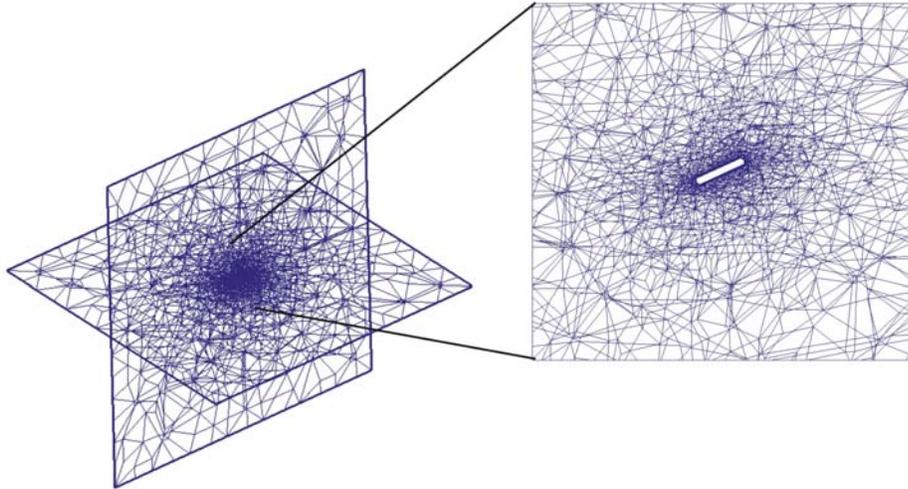


Figure 18. An example of the computational domain used in the simulations (OML regime).

The first set of simulations (Case 1.1 and Case 1.2) was conducted in order to analyze the influence of the ratio T_i/T_e on the electron and ion currents in the Thin Sheath regime ($r_p/\lambda_D > 1$). Potential distributions around a cylindrical probe at $e\Phi_p/kT_e = 5$,

$T_i / T_e = 1$ and $r_p / \lambda_D = 10$ is shown in Figure 19. Formation of the positively charged sheath around the probe can be observed. The electron and ion currents measured in this simulation as a function of number of timesteps are presented in Figure 20. Significant fluctuations in the collected electron current are observed during the sheath formation. Measured electron current is compared with a current predicted by *Peterson and Talbot* (1970) for the computational parameters specified above.

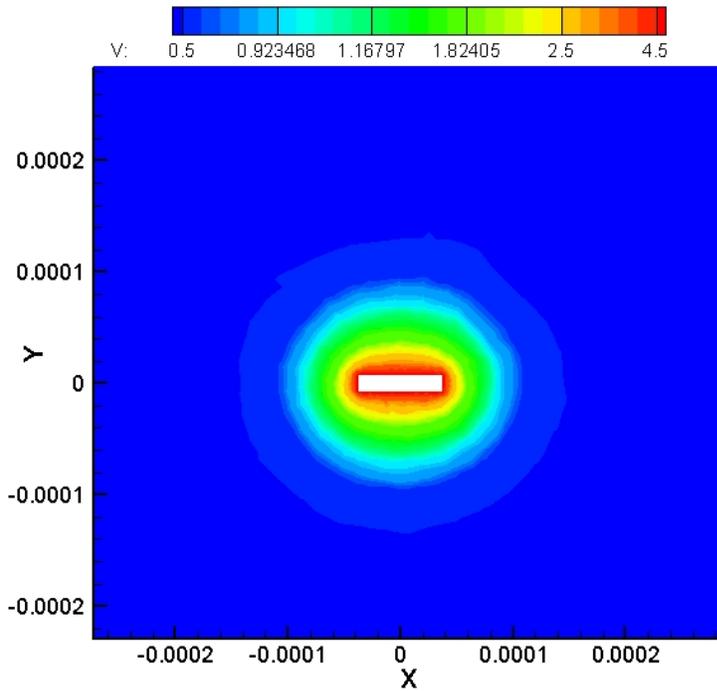


Figure 19. Potential distribution around a cylindrical probe with $e\Phi_p / kT_e = 5$. Simulations parameters are those of Case 1.1.

Figure 21 shows electron and ion currents as a function of probe potential for $r_p / \lambda_D = 10$ and for two ion-to-electron temperature ratios of 1 (Case 1.1) and 0.1 (Case 1.2). The non-dimensional electron current is defined as I/I_{e0} where I_{e0} is the electron current collected by the probe with $\Phi_p = 0$ and $T_i = T_e$.

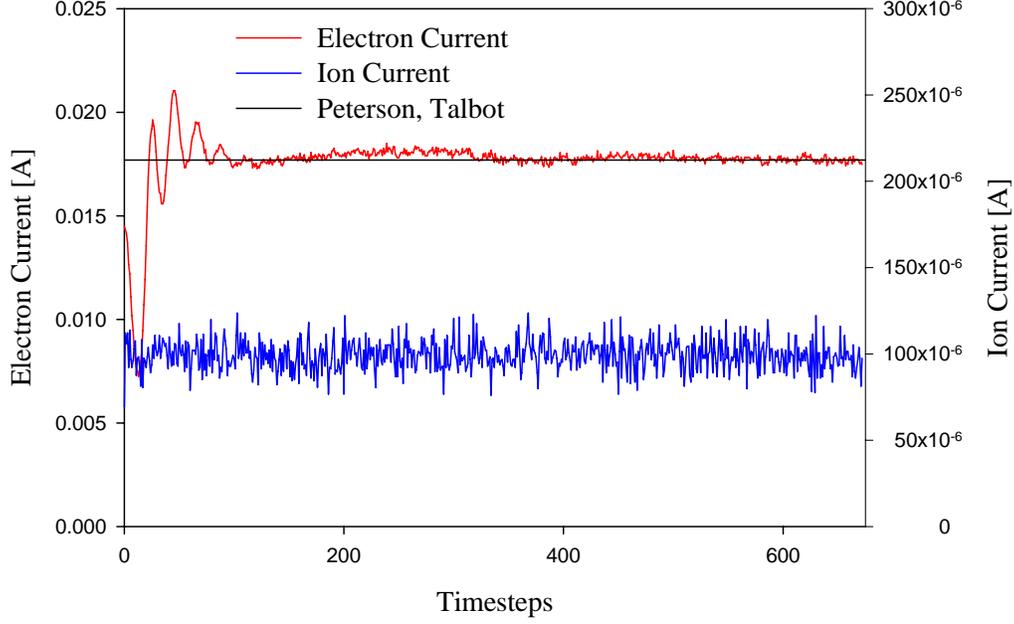


Figure 20. Electron and ion currents collected by a cylindrical probe with $e\Phi_p / kT_e = 5$. Simulation parameters are those of Case 1.1.

These results show that a larger electron current is collected in the case of hotter ions. Almost no electrons were collected by the probe at $e\Phi_p / kT_e < -5$.

The non-dimensional ion current is defined as I_i / I_{i0} where I_{i0} is the ion current that collected by the probe at $\Phi_p = 0$ and $T_i = T_e$. Higher non-dimensional ion currents are predicted for the case when $T_i / T_e = 1.0$. No ion current was measured for $e\Phi_p / kT_e > 0$ for the temperature ratio of 0.1. Also, ion current is negligible for $e\Phi_p / kT_e > 5$ when $T_i / T_e = 1.0$.

The non-dimensional electron current is plotted in Figure 22 as a function of non-dimensional probe potential for $T_i / T_e = 1.0$ and $r_p / \lambda_D = 10$ (Case 2.1) and $r_p / \lambda_D = 5$ (Case 2.2). Higher non-dimensional electron current is predicted for the case of $r_p / \lambda_D = 5$.

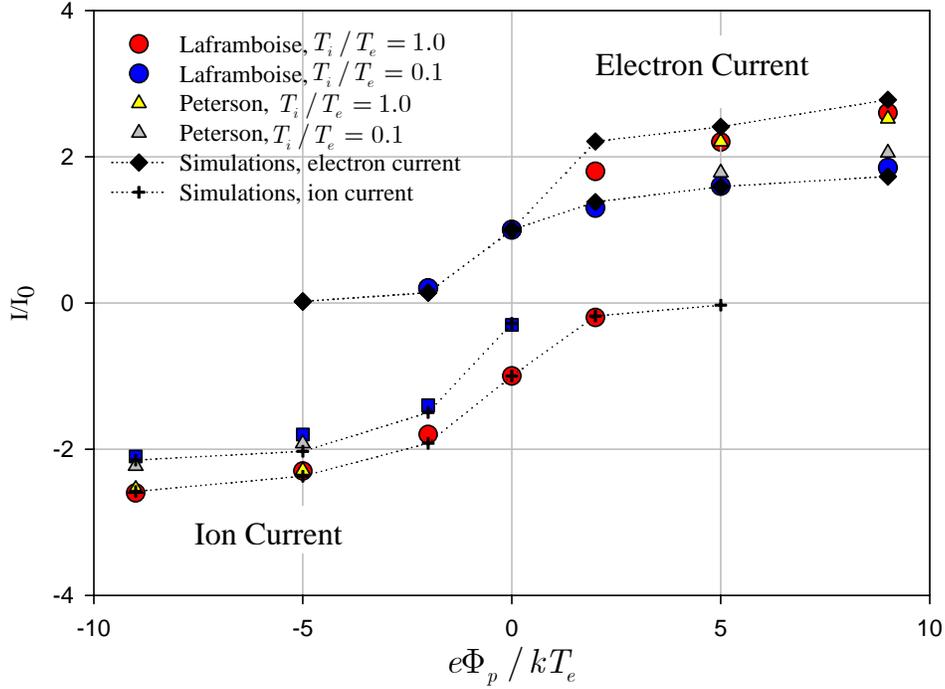


Figure 21. Influence of the ion to electron temperature ratio on current collection by cylindrical probe.

Computational parameters are those of Case1.1 and Case 1.2.

The electron current collected by a cylindrical probe in the OML regime for the $r_p / \lambda_D = 0.1$, $T_i / T_e = 1.0$ and $e\Phi_p / kT_e = 5$ is presented in Figure 23. A process of the sheath formation leads to the fluctuations in the collected electron current before it reaches a steady state. The measured electron current compares favorably to the analytical prediction by *Kanal* (1964).

The non-dimensional electron current measured by a cylindrical probe in the OML regime ($r_p / \lambda_D = 0.1$) for $T_i / T_e = 1.0$ is presented in Figure 24 as a function of the non-dimensional potential $e\Phi_p / kT_e$ (Case 3). Results show good agreement with the numerical results by *Laframboise* (1966) and analytical solution by *Kanal* (1964).

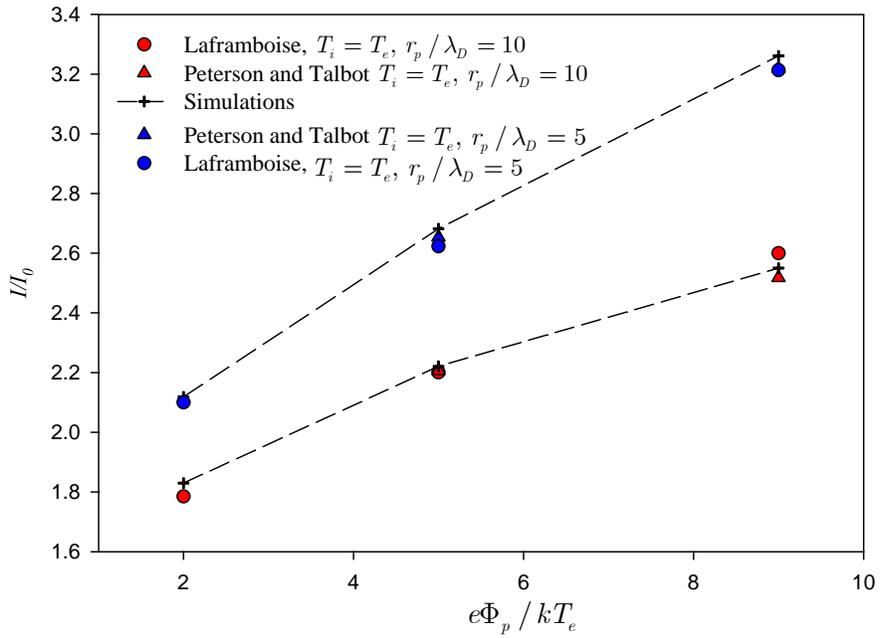


Figure 22. Influence of the Debye length on the current collection. Computational parameters ARE those of Case2.1 and Case 2.2.

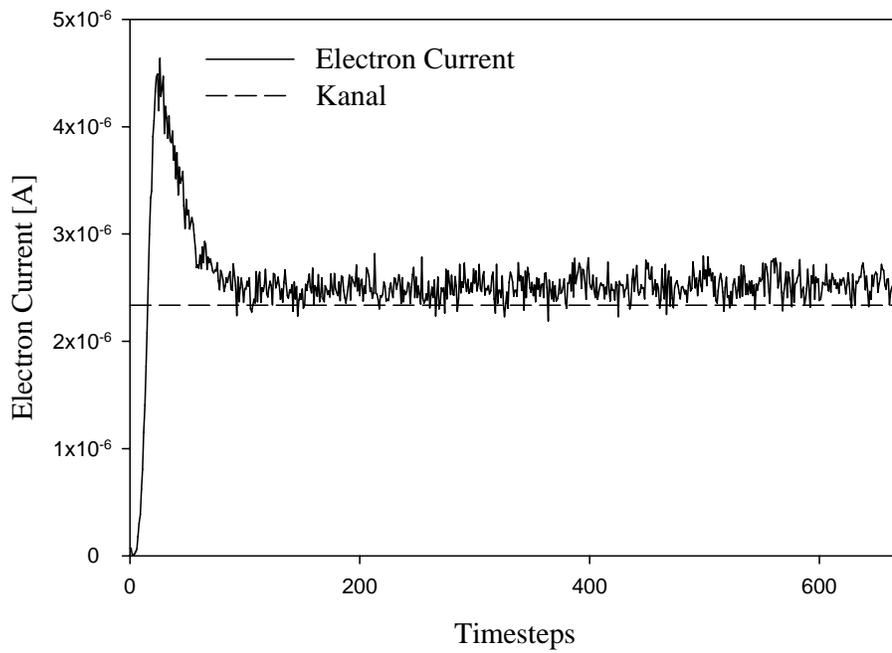


Figure 23. Electron and ion currents, measured at $e\Phi_p / kT_e = 5$. Simulations parameters those of Case 3.

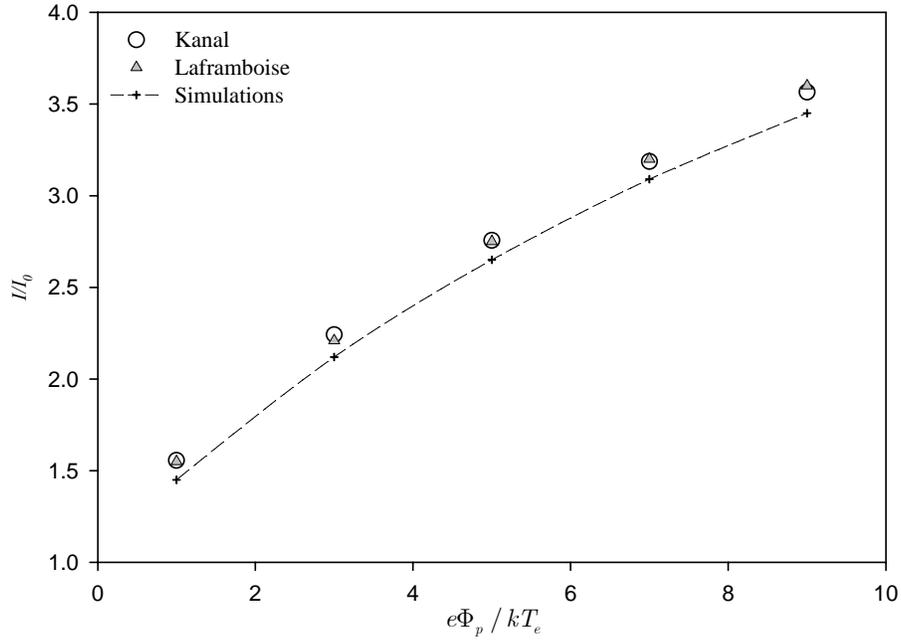


Figure 24. Non-dimensional electron current as a function of undimensional potential in the OML regime (Case 3).

Electron and ion currents collected by a cylindrical probe from drifting plasma are plotted in Figure 25 for the range of ion speed ratios S_i (Case 4). Currents are presented as a function of the ion speed ratio defined by the equation (3.1.2) for $T_i/T_e = 1.0$ and $r_p/\lambda_D = 50$. An increase of the ion current from 1.19 [A] for the ion speed ratio of 1 to 4.86 [A] for $S_i = 7$ is observed. Measured electron current slightly fluctuates about 1.8 [A]. Results are in a good agreement with the equation (3.1.20) by *Johnson and Murphree* (1969).

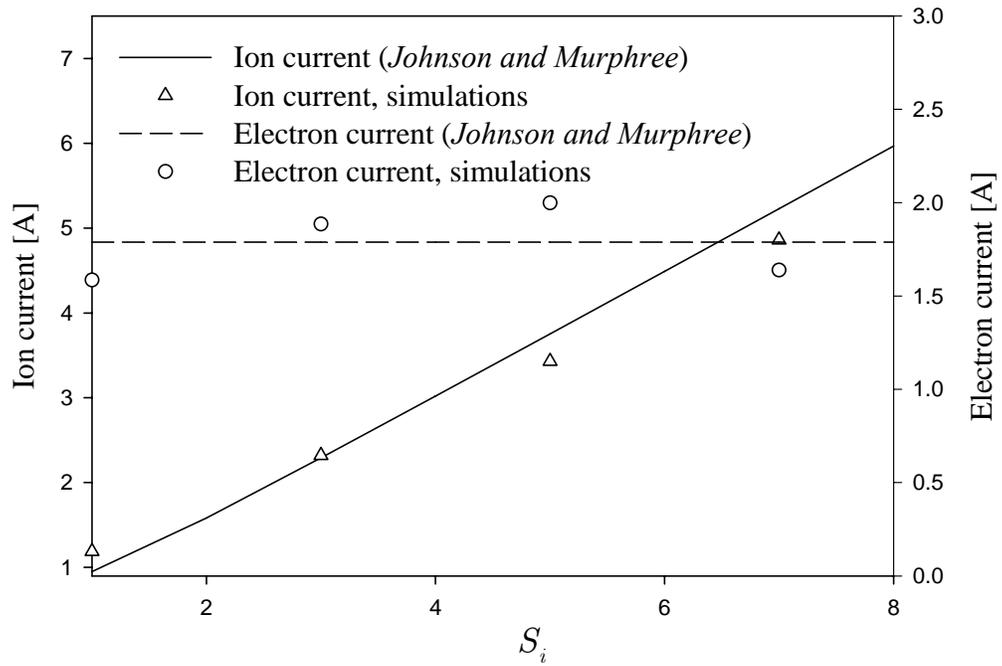


Figure 25. Electron and ion currents collected by a cylindrical probe as a function of the ion speed ratio.

CHAPTER 4 HEATING, SLOWING-DOWN AND DEFLECTION TIMES IN 3-D ELECTROSTATIC PIC SIMULATIONS ON UNSTRUCTURED TETRAHEDRAL GRIDS

The numerical heating and collisions in 3-D electrostatic PIC simulations on unstructured Voronoi-Delaunay tetrahedral grids are investigated. The dependence of heating is evaluated for the near-grid-point and volume weighting and electric field interpolation schemes, for various sizes of the timestep, the cell-edge length and the number of computational particles per cell. Optimum choices of the simulation parameters for reducing numerical heating are discussed. The nonphysical collision-like effects in the collisionless PIC simulations are determined by estimating the slowing-down and deflection times.

4.1 Overview

In the Vlasov limit of an infinite number of particles the collision rates, electric fields and fluctuations reach asymptotically to zero and particle orbits in a uniform plasma in equilibrium are straight lines. In particle-in-cell (PIC) simulations particle the orbits are perturbed and collisions in the simulation occur at rates different that the real plasma. This is a result of the finite size of computational particles, the smaller of number of particles in the simulation than in the real space, the effects of spatial discretization, the charge and force interpolation, and time integration.

Hockney (1971) provided the background theory for measuring the numerical collisional effects in 2D2V PIC simulations with uniform grids. He measured collision

times that exist in real plasma but are accentuated in PIC simulation. Specifically, the slowing-down time, related to the reduction in the velocity along the original direction, and the deflection time related to the deflection in the velocity perpendicular to the original direction. Hockney characterized also the non-physical increase in the kinetic energy of the system in terms of the heating time τ_H . In a real collisionless, uniform plasma in equilibrium there is no heating and collective thermal fluctuations become zero. Hockney attributed the numerical heating to the stochastic electric fields that are finite in the simulation. *Hockney* (1971) measured the heating time using NGP and CIC (Cloud in Cell) interpolating schemes and their modifications. For the 2-D uniform grid case the optimum path for obtaining the time step Δt was found to be $\frac{1}{2} \frac{H}{\lambda_D}$, where H is a cell size and λ_D

is a Debye length. This defines the most sensitive choice of the Δt for a given $\frac{H}{\lambda_D}$ which

means that if Δt is decreased, there is little increase in the heating time and if Δt is increased the rapid degradation of the model is observed. It was also found that CIC heating time is about 20 times longer than NGP heating time except near the $\frac{C\Delta t}{H} = 1$ line

where CIC results degrade much rapidly than results for NGP. It means, that CIC model can be used for much denser or colder plasmas. It was also mentioned that the main parameters controlling the heating time are: the product of the plasma frequency and the time step $\omega_p \Delta t$, the number of computational particles per cell and the ratio λ_D / H .

Hockney also found that the slowing-down time τ_s is as a function of $n(\lambda_D^2 + W^2)$, where W is a computational particle width. No dependence on H or Δt was observed. It was shown that a ratio of the heating time to the slowing-down time is proportional to the

$\left(\frac{H}{\lambda_D}\right)^{-2}$. The deflection time was measured and the square root growth of τ_c with time was predicted.

Birdsall and Maron (1980) considered the issue of the self-heating in the 1-D periodic system and the cold beam instability caused by the numerical grid. It was found that these instabilities are self-quenching due to self-heating and trapping of the beam particles. In the case when $\frac{\mathbf{u}}{w_p \Delta x} > 1$, where \mathbf{u} is a particles mean velocity, regime instability becomes negligible for $C \ll \mathbf{u}$. It was shown that because these instabilities have a very large growth rate, the effect of self-heating may be negligible.

A study of the numerical heating in electrostatic 1-D PIC simulations with periodic boundaries was performed by *Ueda et al.* (1994). Equation for the variation of the heating time when $\lambda_D / \Delta x > 0.1$ was formulated for the CIC weighting scheme. Expression for the ratio of the total field energy to the kinetic energy density, F_E / K_E , was developed. This equation is important in estimating the level of thermal fluctuations. It was shown that increasing the number of particles in the computational domain linearly reduces F_E / K_E .

The issue of numerical heating in a hybrid plasma simulations was considered by *Rambo* (1997). The results for one and two-dimensional cases are presented for the hybrid particle-ion fluid-electron simulations. The dependence of the heating rate from the number of particles per grid cell, time step and the parameter ZT_e / T_i , where Z is an ion charge, was investigated. The comparisons between linear interpolation and NGP were performed, indicating an approximate factor of 6 increase in heating rate for the zero order scheme. The linear dependence on the number of particles per cell was shown. For the “energy conserving” weighting scheme it has been shown that the ion heating can be

significantly reduced at the price of nonconservation of momentum. A 3-point digital smoothing applied both to the density and field for the momentum conservation, gave satisfactory results.

Mardahl (2001) analyzed the dependence of the heating time on the 1-D uniform grid from the several parameters. They are: number of the grid cells, timestep, charge and electric field weighting schemes (NGP, linear, quadratic and cubic splines), number of particles per cell, using a k-space and digital filtering and single or double precision. It was found that the most cost-effective means of reducing numerical heating is k-filtering, followed by digital filtering, increasing the order of the weighting scheme and the number of particles per computational cell. It was also shown that for the best result other parameters should be chosen in the way that $C\Delta t / \Delta x \leq 1/2$. The energy conserving scheme did not give any advantage over the momentum conserving schemes. Using of the higher order schemes reduce the heating significantly but their implementation especially near the domain boundaries is complex.

In this investigation a numerical study of heating and collisions is performed for 3d3V PIC simulations on unstructured tetrahedral grids. The numerical methodology of measuring the heating, slowing-down and deflection times is presented first followed by results from an extensive set of computations.

4.2 Heating, Slowing-Down and Deflection Times in a Collisionless Plasma

We consider a plasma occupying a volume V consisting of electrons and ions denoted by the species index $s \equiv i, e$ each with N_s particles. The mathematical description of the collisionless plasma involves the Vlasov-Maxwell system (*Montgomery and Tidman*

(1964)). If we define single-particle distribution function as the particle density in the velocity-space phase space then the average number of particles in a volume $d^3r d^3c$ of the phase-space centered at a point (\mathbf{r}, \mathbf{c}) is

$$f_s(\mathbf{r}, \mathbf{c}, t) d^3r d^3c = d^6 N_s(t) \quad (4.2.1)$$

and the number density for species s is then

$$n_s(\mathbf{r}, t) = \int f_s(\mathbf{r}, \mathbf{c}, t) d^3c \quad (4.2.2)$$

The total plasma density is

$$n = n_e + n_i \quad (4.2.3)$$

The equation for the distribution function is

$$\frac{\partial f_s}{\partial t} + \mathbf{c} \cdot \frac{\partial f_s}{\partial \mathbf{r}} + \frac{q_s}{m_s} \mathbf{E}(\mathbf{r}, t) \cdot \frac{\partial f_s}{\partial \mathbf{c}} = 0 \quad (4.2.4)$$

The self-consistent electric field (and potential) due to the smoothed distribution (internal) of charges is given by Poisson's equation

$$\varepsilon_0 \nabla \cdot \mathbf{E}(\mathbf{r}, t) = \sum_s q_s \int f_s(\mathbf{r}, \mathbf{c}, t) d^3c \quad (4.2.5)$$

For a plasma in equilibrium the distribution function is a Maxwellian

$$f_s(\mathbf{c}) = \left(\frac{m_s}{2\pi k T_s} \right)^{3/2} \exp\left(-\frac{m_s \mathbf{c}^2}{2k T_s} \right) \quad (4.2.6)$$

The velocity of a particle is $\mathbf{c} \equiv (c_1, c_2, c_3)$ and the equilibrium temperature T

$$\frac{3}{2} k T = \frac{1}{n} \sum_s \int f_s(\mathbf{c}) \frac{1}{2} m_s c_s^2 d^3c \quad (4.2.7)$$

Collisional processes between a test-particle α and a species s of the background plasma are described in terms of basic relaxation times: the longitudinal slowing down, $\tau_s^{\alpha/\beta}$ the transverse $\tau_{\perp}^{\alpha/\beta}$ and parallel deflection time $\tau_{\parallel}^{\alpha/\beta}$ and the energy-exchange

(thermalization) time $\tau_t^{\alpha/s}$ (Trubnikov 1965)). Formulations for the relaxation times depend on the distribution function of the test and field particles.

For the numerical experiments considered in this investigation the field particles are described by a Maxwellian distribution shown in equation (4.2.6). For a test particle moving with velocity \mathbf{c}_a with energy $\varepsilon_a = \frac{1}{2}m_a c_a^2$ we designate as parallel the direction at

$t = 0$, so that

$$\mathbf{c}_{a\parallel}(t = 0) = \mathbf{c}_a(t = 0) \quad (4.2.8)$$

Following Trubnikov (1965) we introduce

$$x_s = \frac{m_s \varepsilon_a}{m_a k_B T_s} \quad (4.2.9)$$

and the Maxwell integral given by

$$\psi(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t} \sqrt{t} dt \quad (4.2.10)$$

The slowing-down time is associated with the process of the average momentum transfer

$$\begin{aligned} \frac{d\mathbf{p}_a}{dt} &= \frac{d}{dt}(m_a \mathbf{c}_a) = \sum_s -\nu_s^{a/s} m_a \mathbf{v} \equiv \sum_s m_a \frac{\langle \Delta v \rangle^{a/s}}{\Delta t} \\ &= \sum_s \left[\left(1 + \frac{m_a}{m_s} \right) \psi(x) \right] \nu_0^{a/s} m_a \mathbf{v} = \nu_s^a m_a \mathbf{v} \end{aligned} \quad (4.2.11)$$

The transverse deflection time is associated with the increase in the spread in the velocity component transverse to the original direction

$$\frac{d}{dt} |\mathbf{c}_a - \bar{\mathbf{c}}_{a\perp}|^2 = \sum_s \mathbf{c}_{\perp}^{a/s} c_a^2 \equiv \sum_s \frac{\langle \Delta c_{\perp}^2 \rangle^{a/s}}{\Delta t} = \sum_s \left[\psi(x) + \psi' - \frac{\psi}{2x} \right] \nu_0^{a/s} c^2 = \nu_{\perp}^a c_a^2 \quad (4.2.12)$$

$$\frac{d}{dt} |\mathbf{c}_a - \bar{\mathbf{c}}_{a\parallel}|^2 = \sum_s \mathbf{c}_{\parallel}^{a/s} c_a^2 \equiv \sum_s \frac{\langle \Delta c_{\parallel}^2 \rangle^{a/s}}{\Delta t} = \sum_s \left[\frac{\psi}{2x} \right] \nu_0^{a/s} c^2 = \nu_{\parallel}^a c_a^2 \quad (4.2.13)$$

The energy loss time

$$\frac{d}{dt} \varepsilon_a \equiv \sum_s \frac{m_a}{2} \frac{\langle \Delta c^2 \rangle^{a/s}}{\Delta t} = \sum_s -2 \left[\frac{m_a}{m_s} \psi - \psi' \right] \nu_0^{a/s} \varepsilon_a = \nu_\varepsilon^a \varepsilon_a \quad (4.2.14)$$

The reference collision frequency is defined as

$$\nu_0^{a/s} = \frac{n_s q_a^2 q_s^2 \ln \Lambda_{as}}{4\pi \varepsilon_0^2 m_a^2 \mathbf{c}_a^3}$$

where $\ln \Lambda_{as}$ is the Coulomb logarithm (*Trubnikov, (1965)*).

The above relations obtained for a test particle can be integrated to provide relaxation times for the general case of a drifting Maxwellian of test particles in a drifting Maxwellian population of background particles. For example, the average momentum transfer electron-ion collision frequency between two Maxwellian can be expressed as (*Mitchner and Kruger (1975)*)

$$\bar{\nu}_{ei} \equiv n_i \frac{4\sqrt{2\pi}}{3} \left(\frac{m_e}{kT_e} \right)^{3/2} \left(\frac{e^2}{4\pi\varepsilon_0 m_e} \right)^2 \ln \Lambda \quad (4.2.15)$$

The slowing-down time

$$\tau_{ei} = - \frac{\langle c_{e\parallel}(0) \rangle}{\frac{d\langle \Delta c_{e\parallel} \rangle}{dt}} = \frac{1}{\bar{\nu}_{ei}} \quad (4.2.16)$$

describes an average time between electron-ion collisions in a plasma. It describes the rate at which the average parallel velocity of the electrons is decreased by encounters with ions and will be compared with the slowing-down time measured in PIC simulations.

The average transverse deflection of a Maxwellian population of test electrons is

$$\frac{d\langle (\Delta c_\perp)^2 \rangle}{dt} = \frac{n_i Z_i^2 e^4 \ln \Lambda}{2\pi \varepsilon_0^2 m_e^2} \int \frac{f_e(\mathbf{c})}{\mathbf{c}_e} d^3 \mathbf{c}_e = \frac{n_i n_e Z^2 e^4 \ln \Lambda}{\sqrt{2\pi}^{3/2} \varepsilon_0^2 m_e^2 C_e} \quad (4.2.17)$$

where C_e is the electron thermal velocity.

We can define also as an average deflection time as the time for the root mean square value of the deflection angle to reach a value of $\frac{\pi}{2}$. The average change in the perpendicular velocity component of electrons measured in the PIC simulations may be compared to the analytical result

Montgomery and Tidman (1964) described also the average energy fluctuation of the electric field per unit volume in 3d as

$$\frac{\langle E^2 \rangle}{8\pi} = \int_{-\infty}^{+\infty} \frac{dk_x dk_y dk_z}{(2\pi)^3} \frac{k_B T}{2} \frac{1}{1 + (k_x^2 + k_y^2 + k_z^2) \lambda_D^2} \quad (4.2.18)$$

where $\mathbf{K} = (k_x, k_y, k_z)$ is the wave number vector. Following (4.2.18) electric field fluctuations are inversely proportional to the number of particles per unit volume. Therefore, in the Vlasov limit of infinite number of particles with uniform density distribution and no external fields, electric field and collision rate become zero and particles orbits are straight lines. In our simulations we will be interested in the artificial acceleration and deflections of the particles (primarily electrons) from their initial directions due to the numerical effects.

4.3 Hating, Slowing-Down and Deflection Times in 3-D PIC Simulations

We consider next the same volume of plasma populated with computation particles. Following *Hockney* (1971) we designate the initial velocity of the particle p as the parallel direction for that particle $\mathbf{c}_p(0) = \mathbf{c}_{p\parallel}(0)$. In the case of no external electric fields and no collisional effects particles would follow that direction. In the PIC simulations at the time t we can measure for the particle p its velocity $\mathbf{c}_p(t)$, its velocity component in the

parallel direction $\mathbf{c}_{p\parallel}(t)$, its velocity component in perpendicular direction $\mathbf{c}_{p\perp}(t)$ and the deflection angle $\varphi_p(t)$, as shown in Figure 26.

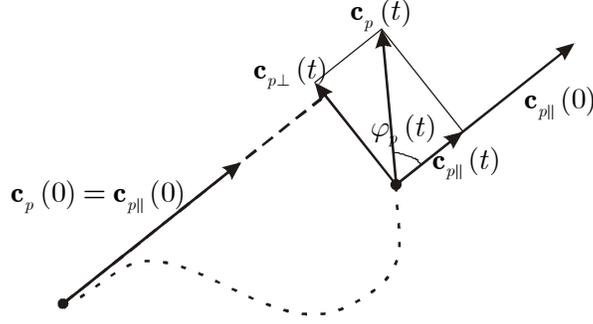


Figure 26. Orbit of the computational particle.

The change in the kinetic energy of a particle p of species s at the time t from its initial value is

$$\Delta E_{sp} = \frac{1}{2} m_{sp} (\mathbf{c}_{sp}^2(t) - \mathbf{c}_{sp}^2(0)) \quad (4.3.1)$$

The average change of the kinetic energy of a particle of species s in the ensemble

$$\langle \Delta E_s(t) \rangle = \frac{1}{N_s} \sum_{p=1}^{N_s} \Delta E_p(t) = \frac{3}{2} k_B T_s(t) - \frac{3}{2} k_B T_s(0) \quad (4.3.2)$$

The heating time τ_{Hs} is defined as the time for the average kinetic energy per particle of species s to increase its energy by $\frac{1}{2} k_B T_s$, i.e.,

$$\langle \Delta E(\tau_{Hs}) \rangle = \frac{1}{2} k_B T_s(0) = \frac{3}{2} k_B T_s(\tau_{Hs}) - \frac{3}{2} k_B T_s(0) \quad (4.3.3)$$

In order to understand how the numerical errors influence the average kinetic energy of the system of particles it is convenient to consider all the errors as giving rise to a stochastic error field $\delta\mathbf{E}$. When this error field is applied to a computational particle it results in a change in the particle velocity $\delta\mathbf{c}_p$ given by

$$\delta \mathbf{c}_p = \frac{q_p \delta \mathbf{E} \Delta t}{m_p} \quad (4.3.4)$$

For n time steps

$$\Delta \mathbf{c}_p = \mathbf{c}_p^n - \mathbf{c}_p^0 = \sum_n \delta \mathbf{c}_{pn} \quad (4.3.5)$$

Following *Hockney and Eastwood* (1999) the error field is assumed to be constant in magnitude $|\delta \mathbf{c}_{pn}| = |\delta \mathbf{c}_p|$ and random in direction. The mean velocity spread of species s particles may be expressed as

$$\langle \Delta \mathbf{c}_s \rangle = \frac{1}{N_s} \sum_{p=1}^{N_s} \Delta \mathbf{c}_{sp} = 0 \quad (4.3.6)$$

For the specific species of computational particles (electrons or ions) the change in the velocity $\delta \mathbf{c}_s$ is also constant in magnitude and random in direction, so that the mean square value of $\Delta \mathbf{c}_s$ is the linear function of the number of timesteps

$$\begin{aligned} \langle |\Delta \mathbf{c}_s|^2 \rangle &= \frac{1}{N_s} \sum_{p=1}^{N_s} |\Delta \mathbf{c}_{ps}|^2 = \frac{1}{N_s} \sum_{p=1}^{N_s} \sum_n |\delta \mathbf{c}_n|^2 = \frac{1}{N_s} \sum_{p=1}^{N_s} n |\delta \mathbf{c}_p|^2 = \\ &= n |\delta \mathbf{c}_p|^2 = n \frac{q_s^2}{m_s^2} (\Delta t)^2 |\delta \mathbf{E}|^2 \end{aligned} \quad (4.3.7)$$

Following (4.3.7) the average change in the kinetic energy of species s particles or heating is

$$\langle \Delta E_s(t) \rangle = \frac{1}{2} \frac{q_s^2}{m_s} (\Delta t)^2 |\delta \mathbf{E}|^2 n \quad (4.3.8)$$

The average change in the kinetic energy expressed by the equation (4.3.8) is independent of the particle's initial velocities, is directly proportional to the number of timesteps, and is inversely proportional to the mass of the computational particles. Therefore, the main contribution to the stochastic error is the change in the temperature of the electrons.

In the heating time investigation we will neglect the plasma potential energy in comparison to the kinetic energy. The plasma as a whole has initially energy of

$$N_e \frac{3}{2} k_B T_e(t) + N_i \frac{3}{2} k_B T_i(0) = (N_e + N_i) \frac{3}{2} k_B T(0) = N \frac{3}{2} k_B T(0) \quad (4.3.9)$$

At $t = \tau_{He}$ the energy of system using equation (4.3.9) and assuming that ions have negligible change in their energy, is

$$N_e \frac{4}{2} k_B T_e(0) + N_i \frac{3}{2} k_B T_i(0) = (N_e + N_i) \frac{7}{2} k_B T(0) \quad (4.3.10)$$

Therefore, the system energy increases by 16% over its initial value. In 2D3V the increase is 25% and in 1d3V is 50%.

We compute also the ensemble average of the parallel velocity component as

$$\langle c_{\parallel s}(t) \rangle = \frac{1}{N_s} \sum_{p=1}^{N_s} c_{\parallel ps}(t) \quad (4.3.11)$$

The slowing-down time T_s is defined as the time it takes for the average electron parallel

velocity to reach $\frac{\langle c_{e\parallel}(0) \rangle}{e}$, i.e.,

$$\langle c_{s\parallel}(T_s) \rangle = \frac{1}{N_s} \sum_{p=1}^{N_s} c_{p\parallel}(T_s) = \frac{\langle c_{s\parallel}(0) \rangle}{e} \quad (4.3.12)$$

This process is modeled by

$$\frac{d}{dt} \langle c_{e\parallel}(t) \rangle = \frac{\langle c_{e\parallel}(0) \rangle}{T_s} \quad (4.3.13)$$

We also calculate the root-mean-square average deflection for a particle of species s at time t

$$\sqrt{\langle \varphi_s^2(t) \rangle} = \sqrt{\frac{1}{N_s} \sum_{p=1}^{N_s} \varphi_{sp}^2(t)} \quad (4.3.14)$$

The deflection time τ_Φ is the time for the average deflection to reach 90 degrees. i.e.

$$\langle \phi_s^2(\tau_\Phi) \rangle^{1/2} = \frac{\pi}{2} \quad (4.3.15)$$

The slowing-down and deflection times can be ascribed to the presence of collisional effects in PIC simulations and are finite in real plasma. The heating time τ_H characterizes a lack of the energy conservation in the PIC model due to the numerical errors and is infinite in real plasma.

4.4 Heating Time

A typical computational domain used in the heating calculations is shown in Figure 27. The plasma is loaded initially into the spherical domain following a Maxwellian distribution with zero drift velocity. Electrons and ions are also injected at each time-step from the open boundaries according to their thermal fluxes. Zero electric field is set at the open boundaries of the domain. Particles that reach the domain surface are removed from the simulation.

Previous 1-D and 2-D investigations on uniform grids have shown that τ_H strongly depends on ratio of the size of the computational cell over the Debye length, $\Delta r / \lambda_D$. A grid-parameter for the unstructured tetrahedral grid used in our investigation is Δr , which corresponds to edge length of a tetrahedron as shown in Figure 28.

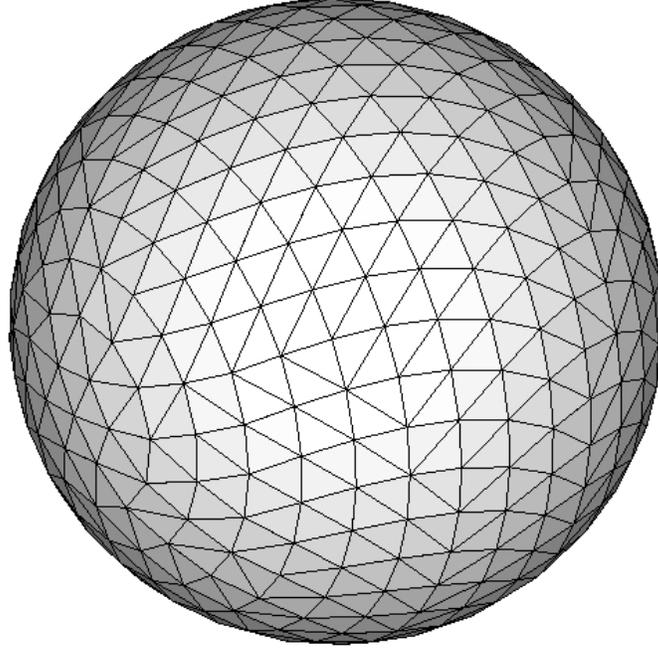


Figure 27. A typical computational domain.

It is therefore important to construct a grid for which Δr will be the same (or as close as possible) for all the cells. To satisfy this requirement on a tetrahedral mesh all cells should have approximately the same volume and dihedral angles as close to 70.53 degrees (1.23 radians). Figure 29 and Figure 30 show the histograms for the cell volumes and dihedral angles obtained for a typical grid used in the simulations. The histogram for Δr is shown in Figure 31.

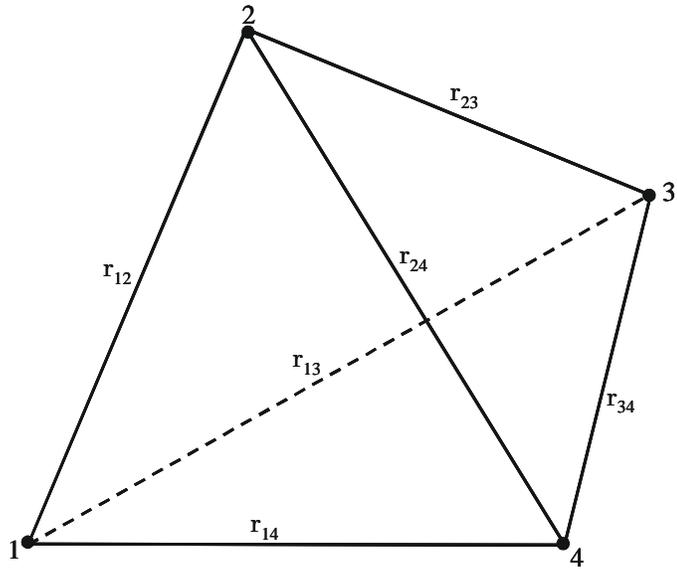


Figure 28. Tetrahedral computational cell.

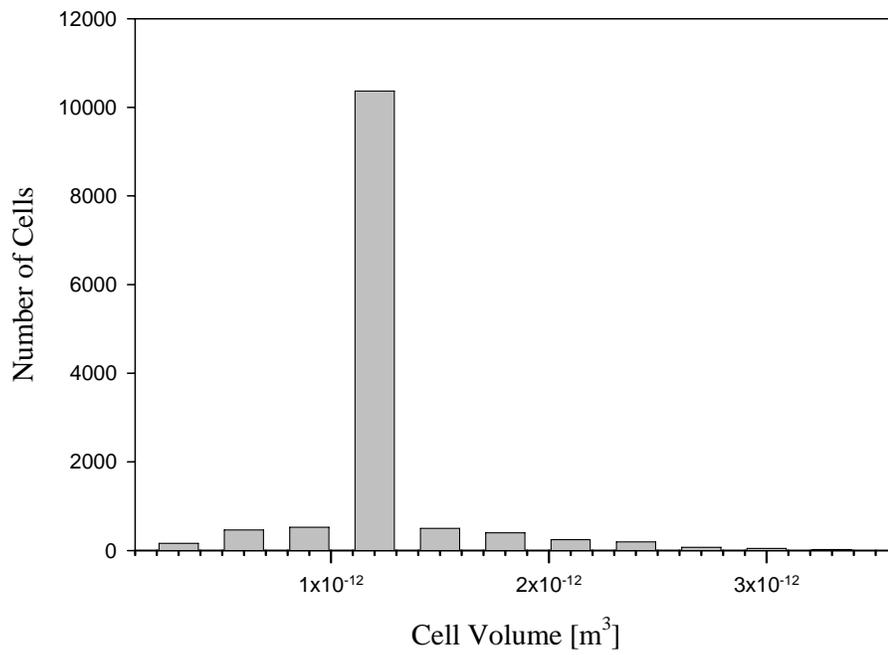


Figure 29. Distribution of cell volumes in a computational domain.

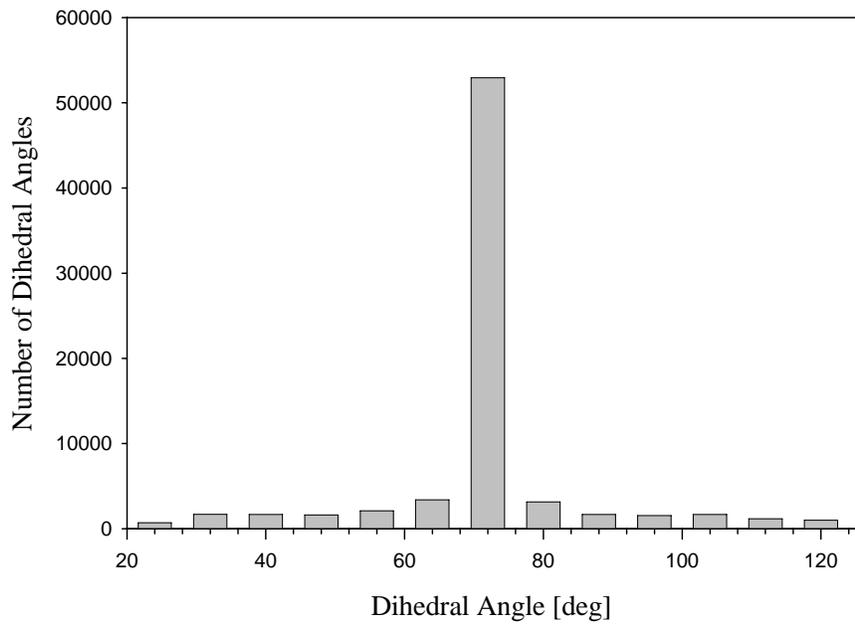


Figure 30. Distribution of the dihedral angles in a computational domain.

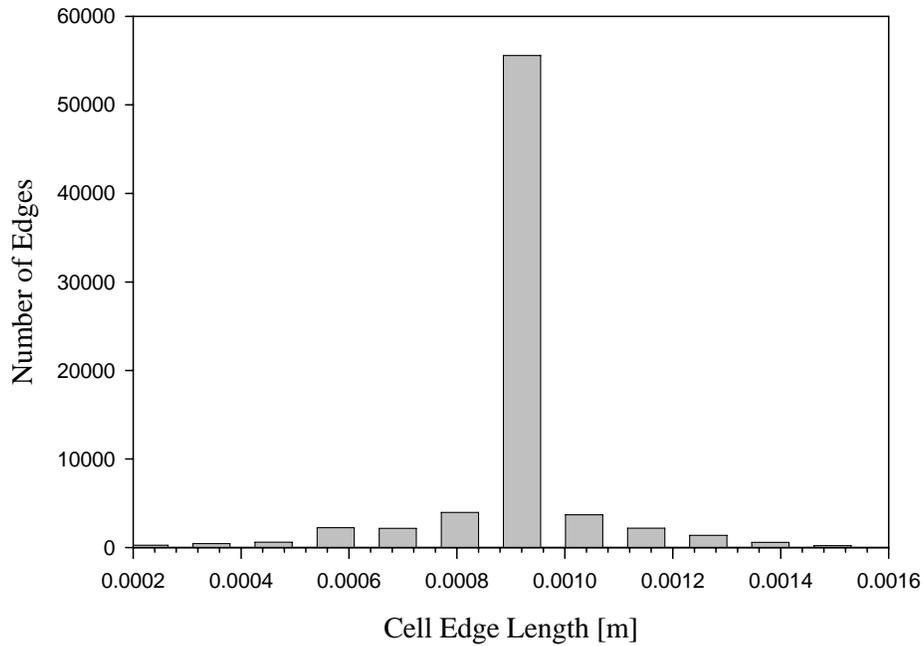


Figure 31. Distribution of cell-edge length in a computational domain.

In order to measure the heating time we vary the simulation parameters and observe how fast the average kinetic energy of the electrons is growing. In many cases an increase in energy is so small that we have to obtain τ_H from the slope of the energy vs. time plot. Equation (4.3.8) predicts that the average kinetic energy of the system is a linear function of the number of timesteps. This has been demonstrated for 1-D and 2-D PIC simulations. Figure 32 shows that for 3-D PIC simulations on unstructured grids expression (4.3.8) holds for both the zero-order (NGP) and first-order (linear) weighting/interpolation schemes.

Figure 33 shows the effects on heating time from changing the parameter $C\Delta t / \Delta r$, while keeping the number of particles per cell constant. The heating time is normalized with the plasma frequency $w_p = \sqrt{\frac{ne^2}{\epsilon_0 m_e}}$. One can observe that when $C\Delta t / \Delta r$ is in the range of 10^{-4} to 5×10^{-4} the heating time remains almost constant. For both weighting schemes the simulations show a significant drop in $w_p t_H$ from 5×10^{-4} to 10^{-3} . Further increase of the timestep has no impact on $w_p t_H$.

The results show that the linear-weighting heating time is about 20% larger than the NGP heating time for small Δt ($C\Delta t / \Delta r < 5 \times 10^{-4}$). For large timesteps ($C\Delta t / \Delta r > 5 \times 10^{-4}$) this difference is more than 100%.

Another parameter in the heating investigation is the number of particles per cell. Increasing the number of particles per computational cell leads to smoother electric fields and therefore larger heating times.

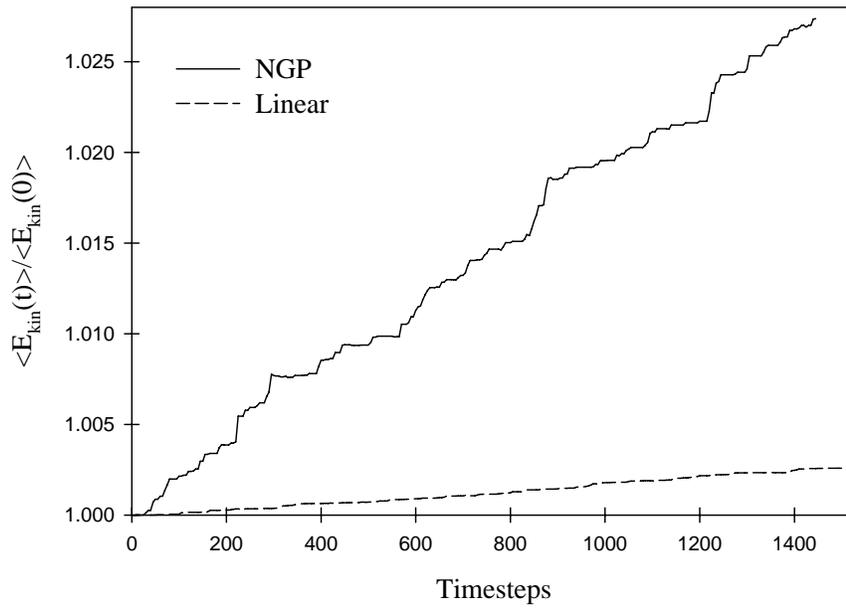


Figure 32. Time evolution of the total kinetic energy.

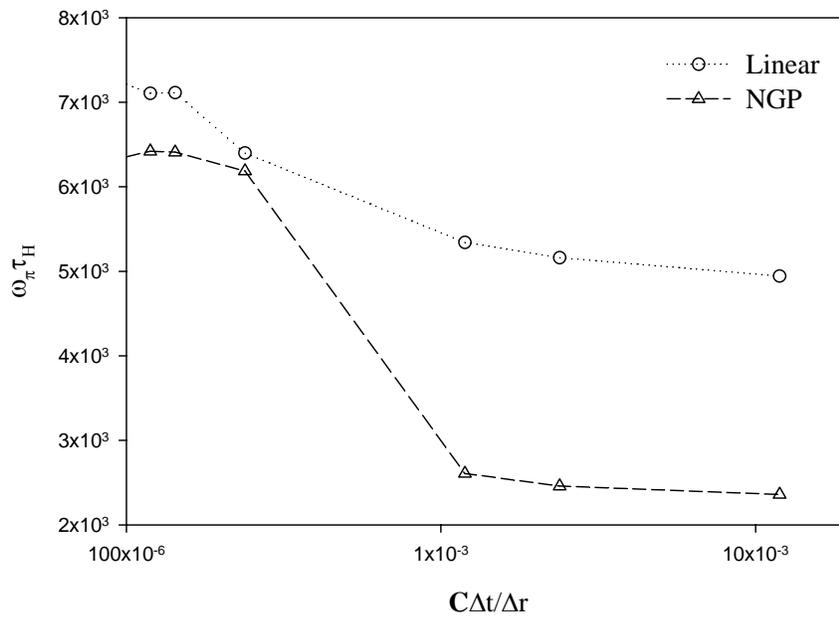


Figure 33. Heating time as the function of the timestep, for 20 particles per cell and $\Delta r = \lambda_D$.

Figure 34 shows the effect of a number of particles per cell on the heating time for NGP and linear weighting schemes. The increase in the heating time shown is about 10 times for both weighting schemes while the number of particles per cell changes from 15 to 90. It is important to mention that the computational time for particle move rises linearly with the number of particles. Therefore, a tradeoff is required between the overall computational time and the acceptable quality of the computation.

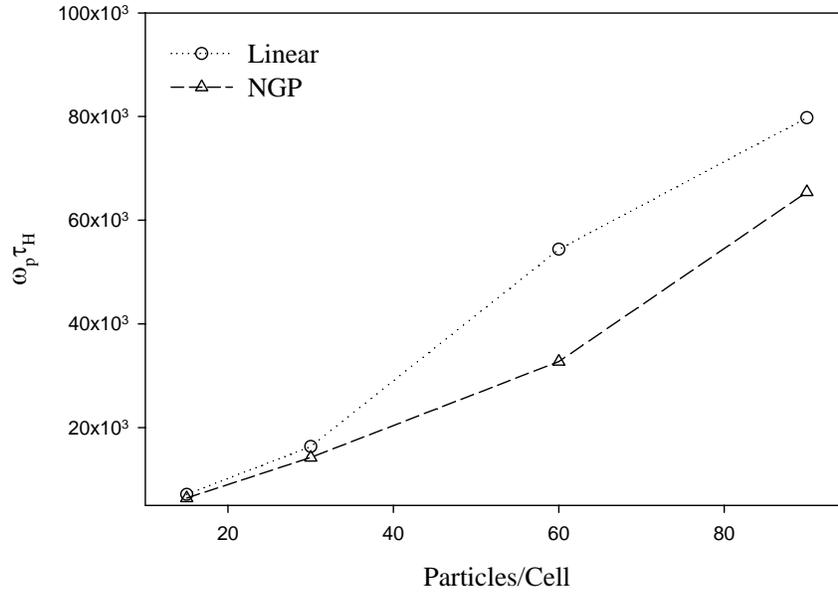


Figure 34. Heating time as a function of the number of particles per cell for $w_p \Delta t = 5 \times 10^{-3}$ and $\Delta r = \lambda_D$.

Changing the ratio of the Debye length to the cell edge length $\lambda_D / \Delta r$ has a strong effect on the heating time as shown in Figure 35. A decrease of the heating time by an order of magnitude is observed when $\lambda_D / \Delta r$ changes from 0.2 to 2. There is no appreciable effect on heating in the range $2.0\lambda_D \leq \Delta r \leq 5.0$. When $\lambda_D / \Delta r$ is increased further Figure 11 shows a drop in the heating time by an almost order of magnitude.

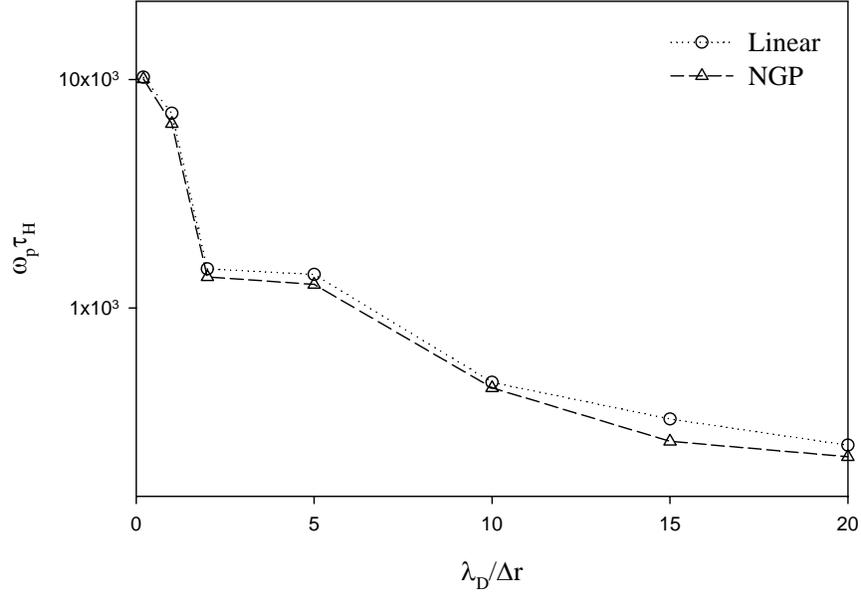


Figure 35. Heating time as a function of $\lambda_D / \Delta r$ for $w_p \Delta t = 5 \times 10^{-3}$ and 20 particles per cell.

4.5 Slowing-Down and Deflection Times

At a time t in the computation we calculate a parallel velocity component from the dot product of the vectors $\mathbf{c}_{\parallel}(0)$ and $\mathbf{c}(t)$. We will treat the plasma as collisionless until the simulation time is less than the numerical slowing-down time T_s . Since the average electron slowing-down time is much smaller than the one for heavier ions, it will be used to define the limiting conditions.

Figure 36 shows the T_s / τ_{ei} as the function of the number of particles per computational cell for both weighting/interpolation schemes.

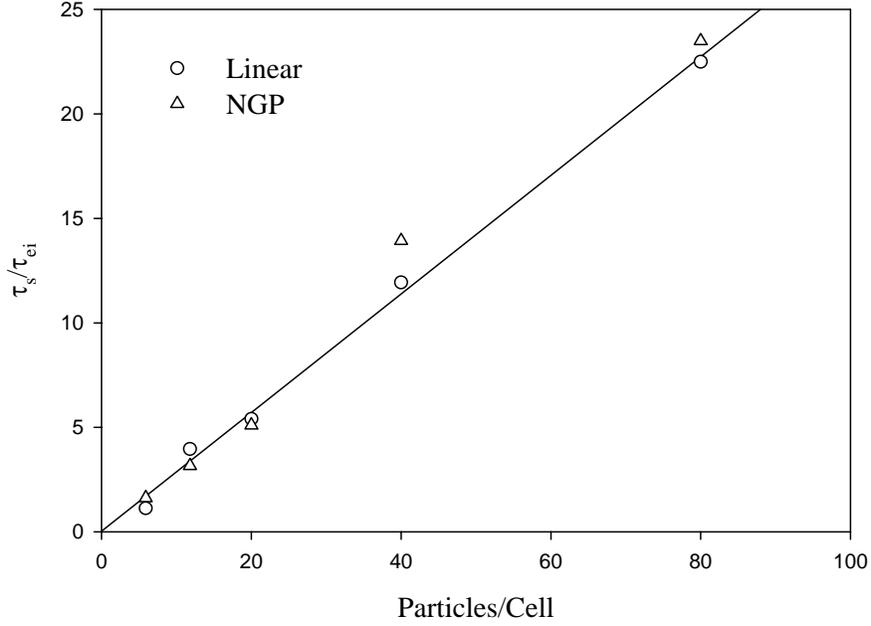


Figure 36. Slowing-down time vs. number of particles per computational cell.

A linear growth of the slowing-down time can be observed as the number of computational particles in the domain increases. The magnitude of the slowing-down time does not have any significant dependence on the order of the weighting scheme used.

In the PIC simulation, an electron deflection angle at the time t is measured as

$$\varphi(t) = A \cos \left(\frac{\mathbf{c}(0) \cdot \mathbf{c}(t)}{|\mathbf{c}(0)||\mathbf{c}(t)|} \right) \quad (4.5.1)$$

The average change in the perpendicular velocity component is evaluated when the average value of the deflection angle of an electron reaches 90 degrees. This value is compared to the theoretical value.

Figure 37 shows $\langle \mathbf{c}_{\perp}^2 \rangle_{simulation} / \langle \mathbf{c}_{\perp}^2 \rangle_{theory}$ as a function of the number of particles per computational cell. The rate of change of the perpendicular component of the velocity decreases significantly with an increase of number of particles. Since the deflection time is

much smaller than heating time, no significant dependence upon type of weighting scheme is observed.

4.6 Conclusions

For the three dimensional electrostatic plasma simulations on unstructured tetrahedral grids the most cost effective way to reduce numerical heating is to choose the size of the computational cells in such a way that $0.2 < \lambda_D / \Delta r < 5.0$. However, Δr is not a free parameter but relates to stability requirements as well as the Debye length.

The proper choice of the weighting/interpolation schemes can reduce the heating time significantly. Difference in measured heating times using NGP or Linear weighting schemes is not as significant as in the case of 1-D PIC simulations. Linear weighting will always produce better results, is easy to implement and is only slightly slower than NGP. Higher order weighting/interpolation schemes should be able to further increase the heating time. However, construction of such schemes on 3-D unstructured grids could lead to more complex analytical formulations that will increase the computational time.

In order to obtain a high-quality simulation, the numerically stable timesteps should be chosen so that $C\Delta t / \Delta r < 5 \times 10^{-4}$. Using longer timesteps reduces the computational time but it also reduces the heating time significantly.

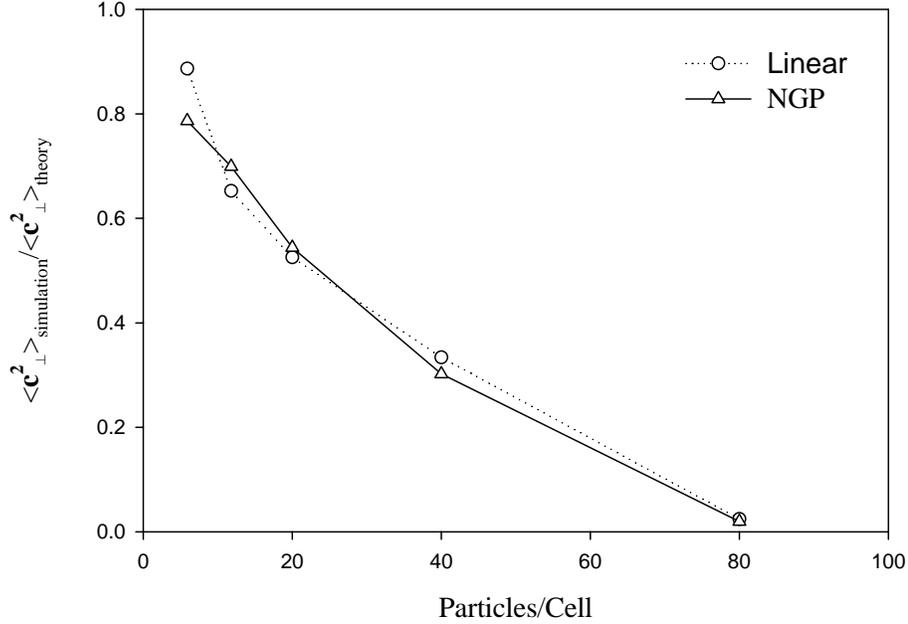


Figure 37. Deflection time as the function of the number of particles per cell.

Increasing the number of computational particles per cell is the easiest way to reduce numerical heating. i.e. increase the heating time but at the cost of increasing the overall computational time.

The slowing-down and deflection times measured in the PIC computations are compared to the analytical expressions for the average change of the electron velocity due to interactions with heavy ions. It is shown, that increasing the number of computational particles in the domain significantly reduces these artificial collision-like effects in collisionless PIC simulations.

CHAPTER 5 SIMULATIONS OF PLASMA DEVICES AND MICROSENSORS

In this chapter we present two novel applications of the unstructured PIC code. The first involves the simulation of the operation of a Field Emission Array (FEA) and the second, involves the simulation of a micro Retarding Potential Analyzer. The simulations provide an insight of the operation of these and demonstrate the ability to use advanced PIC simulations as design tools.

5.1 Simulations of 3-D Space-Charge Effects in Gated Field Emission Array

Cathodes

Field Emission Array (FEA) cathodes are electron sources that can provide high electron current densities for various applications including mass spectrometers, scanning electron microscopes and different space-based applications of electric propulsion. FEA's are considered as an alternative to the thermionic cathodes because they are easily scalable in size, chemically inert and do not require propellant or heater. FE cathodes use electric fields to emit electrons and are significantly more effective than other types of cathodes. FEA cathode consists of many tiny emitters with gates (as shown in Figure 46) that are produced with micro fabrication methods. According to *Marrese* (1999), tips packing densities up to 10^{12} tips/m² can be achieved producing emission current densities of more than 10^7 A/m². Two most commonly used types of FEA cathodes are silicon or molybdenum micro fabricated and carbon thin-film FEA's.

A FE cathode tip configuration is shown in Figure 38. Tip and gate electrode radii r_t and r_g are on the order of 1 and 100 nm respectively. An electric field is applied to the gate electrode in order to deform the potential barrier and allow the electrons to leave the surface of the tip. Following *Marrese* (1999), electric fields larger than 10^9 V/m are required for the field emission.

Simulations of the electron beam emitted by the FEA have been performed by *Candel et al.* (2005) using the 3-D Maxwell PIC code “Capone”. The FEA a cathode consists of about 2000 emitting tips. Convergence tests for the emission from an ideal FEA (all emitting tips are identical) are performed. Also properties of the electron beam are analyzed as a function of the FE geometrical parameters and different gap voltages. Importance of the different degradation effects on the FEA operation is investigated in a parametric study. Simulation results define ideal parameters for the design of the LEG DC gun.

In this chapter we present a multi-scale model and simulation of a FEA. The electron emission by the elliptic field emitters is modeled following *Jensen and Zaidman* (1995). The current emitted by a single molybdenum microtip is estimated using these analytical expressions. Subsequently, the current and current density by the FEA cathode are estimated for a wide range of gate potentials. These estimates serve as inputs to the unstructured PIC simulations that include the FEA cathode and the anode. The simulations predict the formation of the virtual cathode for different gate potentials and fixed cathode-to-anode potential differences. FEA operation limitations due to space-charge effects are analytically predicted and the maximum current collected by the anode located at a fixed

distance from the emitting cathode is estimated following *Jensen et al. (1997)*. Measured electron anode and gate currents are compared to the analytical predictions.

5.1.1 Model for Field Emission Array Cathode

Jensen and Zaidman (1995) formulated an analytical model for a hyperbolic shape FE tip that predicts an electric field at the tip apex to be

$$F_{tip} = \left(\frac{\pi}{\ln\left(\frac{kr_g}{r_t}\right)} - (\tan \beta_c)^2 \right) \frac{\Phi_g}{r_t} \quad [\text{V}/\text{\AA}] \quad (5.1.1)$$

In above r_t is a tip radius, r_g is a gate electrode radius in [\AA], Φ_g [V] is a gate potential and

$$k = \frac{1}{54} \left(86 + \frac{r_g}{r_t} \right) \cos \beta_c \quad (5.1.2)$$

Variable $\beta_g = \frac{1}{r_t} \left(\frac{\pi}{\ln\left(\frac{kr_g}{r_t}\right)} - (\tan \beta_c)^2 \right)$ in equation (5.1.1) is referred to as “field

enhancement factor”. Geometrical parameters used on equation (5.1.1) are shown in Figure 38.

The emitted electron current density is evaluated by the Fowler-Nordheim expression:

$$J_{FN}(F, \beta, \mu) = \alpha_{fn} F^2 \exp\left(-\frac{\beta_{fn}}{F}\right) \frac{c_{fn} \pi / \beta}{\sin(c_{fn} \pi / \beta)} \left(1 - (c_{fn} \mu + 1) \exp(-c_{fn} \mu)\right) [e/\text{\AA} \text{Fsec}] \quad (5.1.3)$$

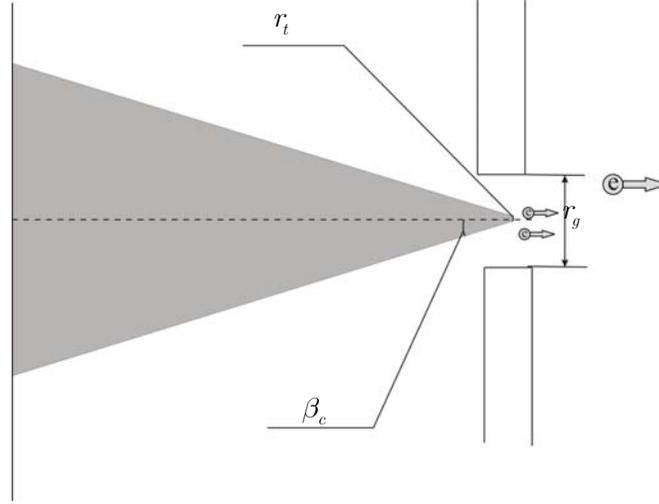


Figure 38. FE cathode microtip.

In equation (5.1.3) μ [eV] is the Fermi level of the electron gas and

$$\beta = \frac{1}{k_B T} \quad (5.1.4)$$

Fowler-Nordheim coefficients are estimated as follows:

$$\alpha_{fn} = \frac{\exp\left(\frac{16Q}{3\hbar} \sqrt{\frac{2m}{\Phi}}\right)}{16\pi^2 \hbar \Phi t^2} \quad [e/eV^2 \text{fsec}] \quad (5.1.5)$$

$$\beta_{fn} = \frac{4}{3\hbar} \sqrt{2m\Phi^3} \nu_0 \quad [\text{eV}/\text{\AA}] \quad (5.1.6)$$

$$c_{fn} = \frac{2}{\hbar F} \sqrt{2m\Phi} t \quad [1/\text{eV}] \quad (5.1.7)$$

In the equations (5.1.5) - (5.1.7)

$$Q = \frac{\alpha_{fs} \hbar c}{4} \quad [\text{eV} \text{\AA}] \quad (5.1.8)$$

where $\alpha_{fs} = 1/137.04$ is the fine structure constant, m [eV/c²] is the electron mass in vacuum, Φ [eV] is the work function, $t \approx 1.05657$ and $\nu_0 = 0.93685$.

The current density, obtained as a function of F_{tip} , should be integrated over the emitter surface in order to obtain a current emitted by a single tip.

$$I_{tip} = b_{area} J_{FN} \quad [\text{e/fsec}] \quad (5.1.9)$$

where the area factor b_{area} [\AA^2] is

$$b_{area} = 2\pi r_t^2 (\cos \beta_c) \left(\frac{F_{tip}}{\beta_{fn} + (\sin \beta_c) F_{tip}} \right) \quad (5.1.10)$$

The current produced by the number of tips N_{tips} is then

$$I_{array} = N_{tips} \Sigma(\Delta s, \Phi_g) b_{area} J_{FN} \quad (5.1.11)$$

In equation (5.1.11) $\Sigma(\Delta s, \Phi_g)$ is the distribution factor associated with a spread Δs in tip radii. For a single tip Δs equals zero but for a nonuniform array of tips $\Delta s > 0$. In order to estimate $\Sigma(\Delta s, \Phi_g)$ we assume that the tip radii $a(s)$ are uniformly distributed for $0 \leq s \leq \Delta s$.

Setting

$$F(s) \approx F_{tip} (1 + c_0 s) \quad (5.1.12)$$

the distribution function becomes

$$\Sigma(\Delta s, \Phi_g) = \frac{e^{\Delta s b} - 1}{\Delta s b} \quad (5.1.13)$$

where

$$b = \frac{c_0 \beta_{fn}}{\beta_g V_g} + (3c_0 + 2) - \frac{c_0 \beta_g \Phi_g}{\beta_{fn}} \tan^2(\beta_c) \quad (5.1.14)$$

$$c_0 = \frac{1}{\pi \beta_g r_t} (\tan^2(\beta_c) + \beta_g r_t)^2 - 1 \quad (5.1.15)$$

5.1.2 Model for Space-Charge Effects

At high current densities space charge effects significantly affect the emission process and give rise to a large currents collected by a gate electrode. *Jensen, Kodis et al.* (1997) developed a model to analyze such an effects assuming that all emitters emit

electrons every $\tau = \frac{1}{I_{tip}}$ and at point z_g electrons have a velocity equal to $\sqrt{\frac{2\Phi_g}{m}}$.

$$z_g = \frac{\Phi_g}{\sqrt{2F_{tip}F_0}} \quad [\text{\AA}] \quad (5.1.16)$$

$$F_0 = \frac{\Phi_{anode} - \Phi_g}{D} \quad [\text{eV/\AA}] \quad (5.1.17)$$

In equation (5.1.17) Φ_{anode} is anode voltage and D is a distance between cathode and anode.

In this model it is also assumed that the anode is uniformly charged, emitter is a spherical and gate to anode distance is mach larger then gate radius, as shown in Figure 39.

Equation for the flight time of a sheet of charge moving from the gate electrode to the anode, taking into account changes in the anode current I_{anode} is then found to be

$$\Delta t = \frac{\Delta t_0}{1 - 4\pi\alpha_{fs}\hbar c I_{anode}\sigma\Psi} \quad [\text{fsec}] \quad (5.1.18)$$

In equation (5.1.18)

$$\Psi = \frac{(D - z_g)^2}{35(\nu_a + \nu_g)^5 m\nu_a} (47\nu_a^3 + 235\nu_a^2\nu_g + 285\nu_a\nu_g^2 + 105\nu_g^3) \quad [\text{\AA fsec/eV}] \quad (5.1.19)$$

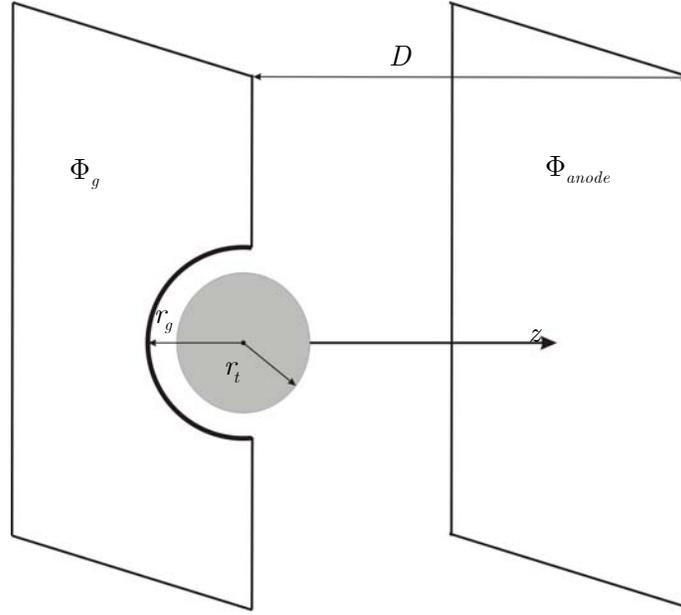


Figure 39. Geometry of the space-charge effects model.

At a specific gate potential anode current reach a maximum possible value, which can be estimated as

$$I_{\max} = \frac{m\nu_a}{18\alpha_{fs}\pi\hbar c\sigma(D-z_g)^2} \quad [\text{e/fsec}] \quad (5.1.20)$$

In (5.1.20) σ is a sheet charge density

$$\sigma = \frac{\eta \Sigma(\Delta s, \Phi_g)}{d_{tt}^2} \quad [1/\text{\AA}^2] \quad (5.1.21)$$

In above d_{tt} is a tip-to-tip distance.

Nondimensional parameter η in equation (5.1.21) is the ratio of the mean sheet charge density with the sheet charge density at the emitter surface and is a function of the spread angle φ , the emitter width W and the cathode to anode distance.

$$\eta^{-1} = \frac{1}{3} \left[\left(\frac{D}{W} \right)^2 \tan^2(\varphi) + \frac{D}{W} \tan(\varphi) + 1 \right] \quad (5.1.22)$$

Parameters for the molybdenum FEA cathode used in further investigations are shown in Table 2.

$r_t[\text{\AA}]$	$r_g[\text{\AA}]$	$\beta_c[\text{rad}]$	$\mu[eV]$	$\Phi[eV]$
40	2500	0.262	5.873	4.41
Packing. Dens. [tips/m ²]	$\varphi[\text{rad}]$	$D[\text{\AA}]$	Cathode Area [\AA^2]	$d_{tt}[\text{\AA}]$
10^{10}	0.698	2.24×10^8	10^{16}	10^5

Table 2. Molybdenum FE cathode parameters.

Figure 40 and Figure 41 show that a tip current up to 10^{-4} [A] and FE array current up to 20 [A] may be theoretically reached assuming high gate potentials (about -100 [V]). Velocities and densities of the emitted electrons are shown in Figure 42 as a function of Φ_g .

The operation of a FEA was simulated using the unstructured 3d PIC code. The objective is to predict the formation of the virtual cathode and the role of space-charge effects in the operation of the FEA.

5.1.3 PIC Simulations of FEA

The cylindrical simulation domain is shown in Figure 43. The left-side boundary represents a circular FE cathode surface with applied potential Φ_g . Cold electrons are emitted with a Drifting Maxwellian distribution from cathode with the velocity corresponding to the gate potential.

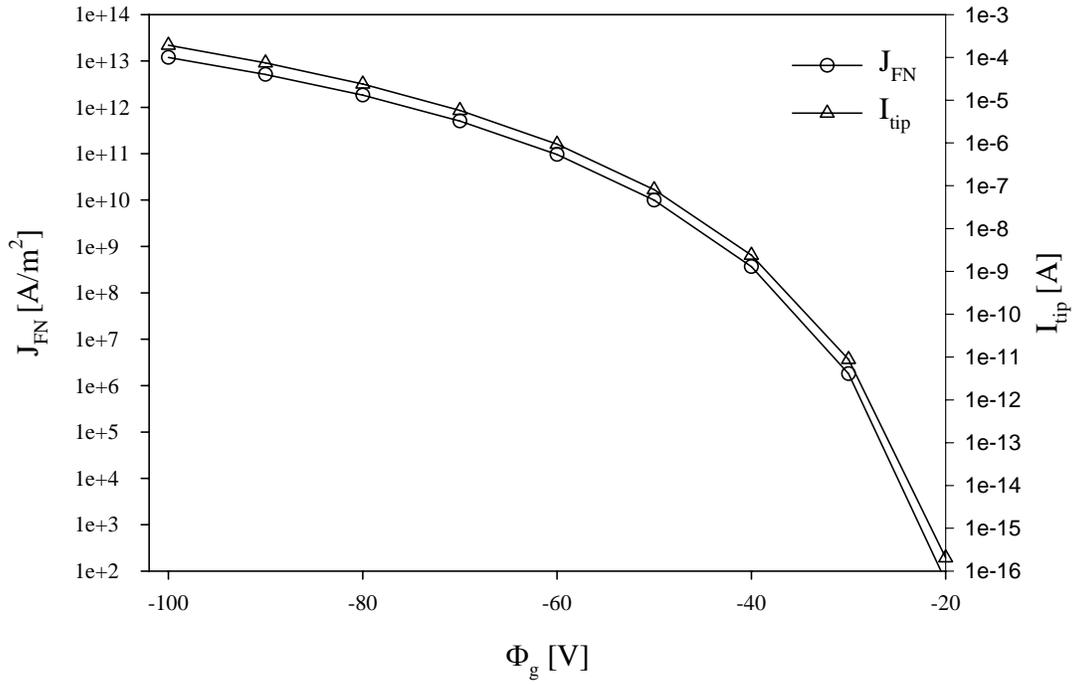


Figure 40. Fowler-Nordheim current density and tip current as a function of gate potential.

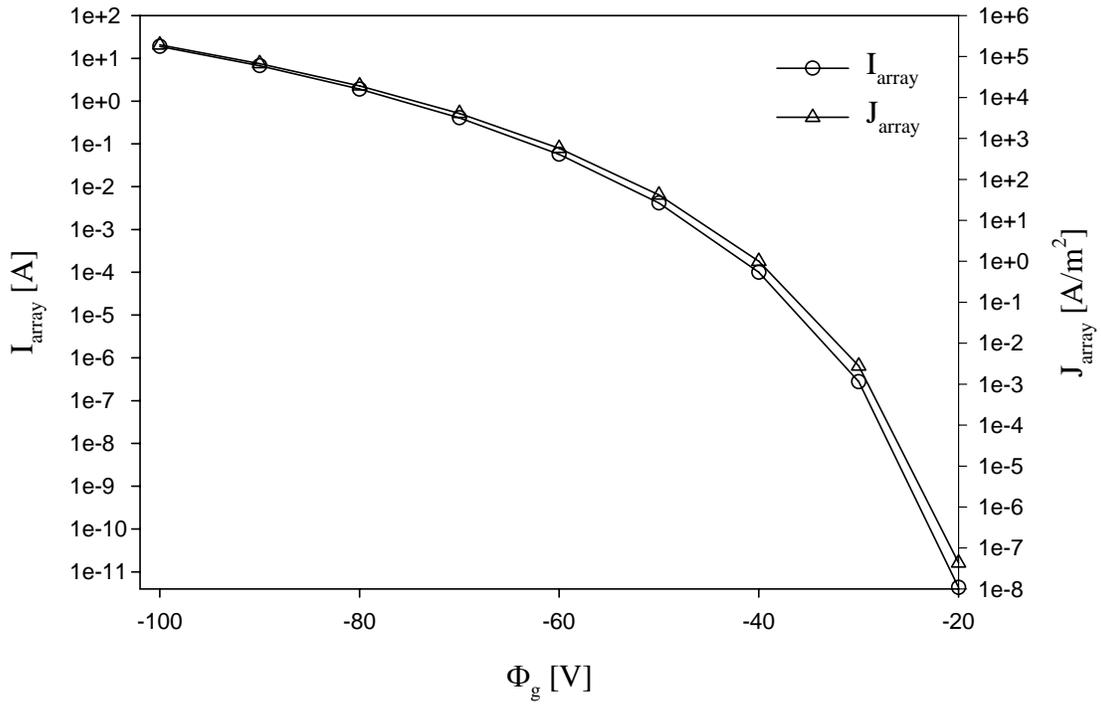


Figure 41. FE array current and current density as a function of gate potential.

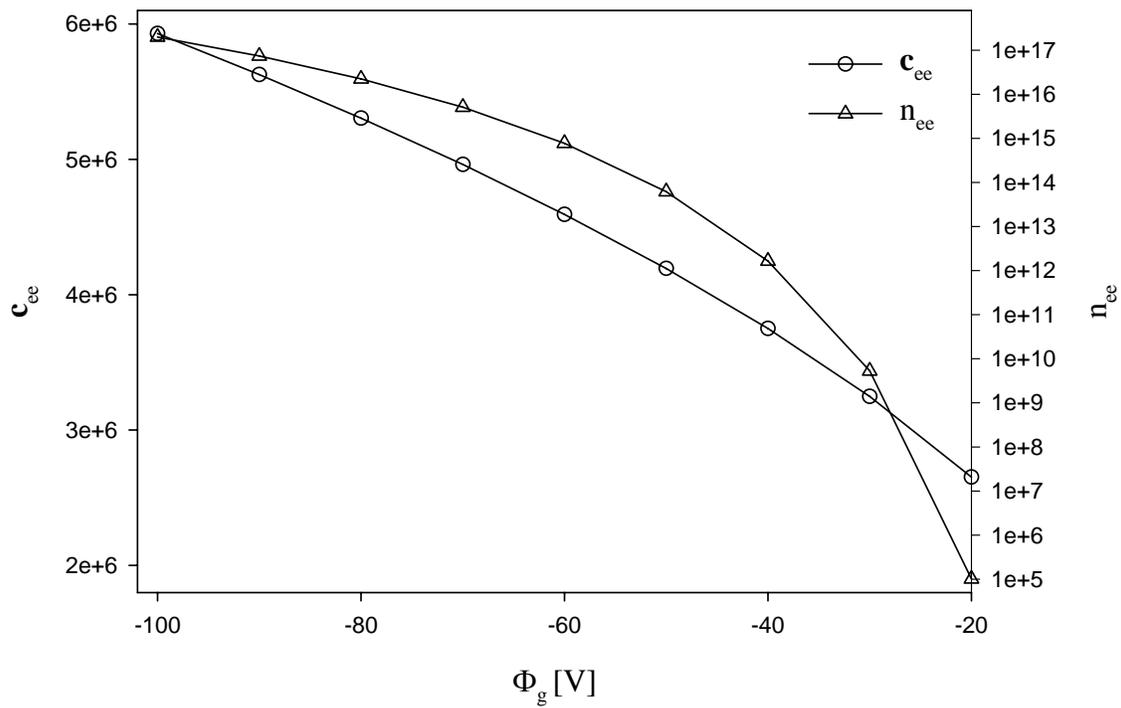


Figure 42. Velocity and density of the emitted electrons as a function of gate potential.

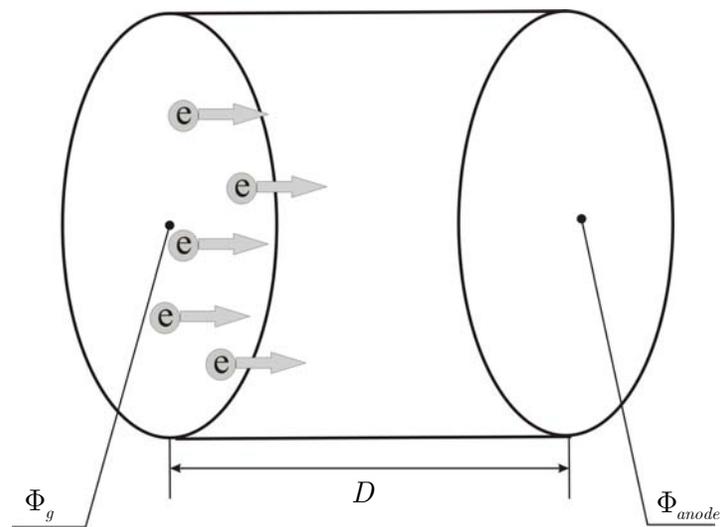


Figure 43. Simulation domain used in FE cathode simulations.

Injection parameters for the FE electrons are shown in Figure 42. Two sets of simulations were performed where cathode-to-anode potential drop was set to 100 and 150 Volts. Results are shown in Figure 44 and Figure 45. The anode current I_{anode} is measured as a function of Φ_g . Some electrons will be reflected by the virtual cathode and will be collected by the emitter. The gate current I_{gate} is also measured since reflected electrons may limit a FE cathode emission.

As Figure 44 shows, formation of virtual cathode is expected for the gate potentials between -35 and -40 Volts. Comparably large gate currents are observed for Φ_g lower than -40 Volts. At a potential of about -50 Volts the maximum possible current is achieved. This prediction agrees with analytical predictions, though the anode current obtained in the simulation is slightly larger.

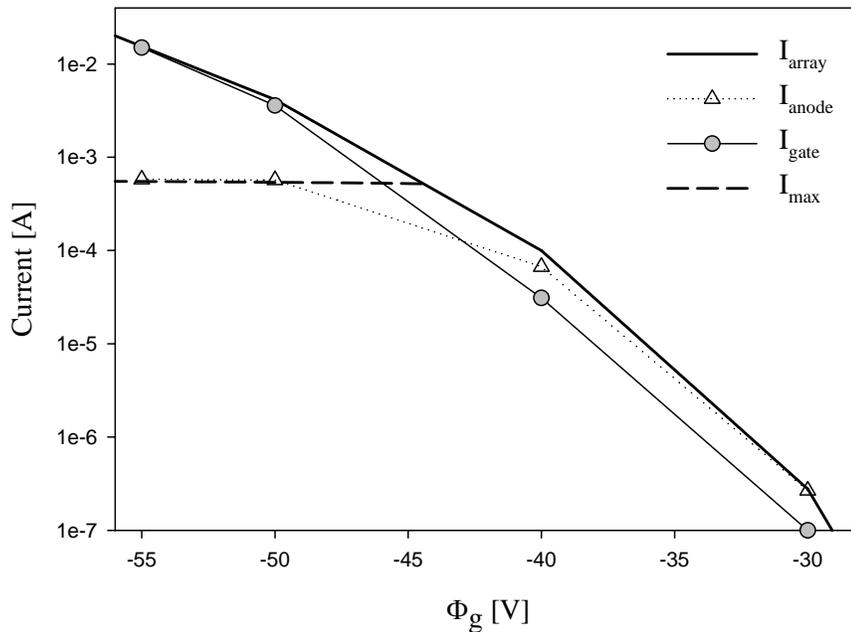


Figure 44. Anode, gate, analytical maximum and FE array currents as a function of gate potential for $\Phi_{anode} = \Phi_g + 100$ [V].

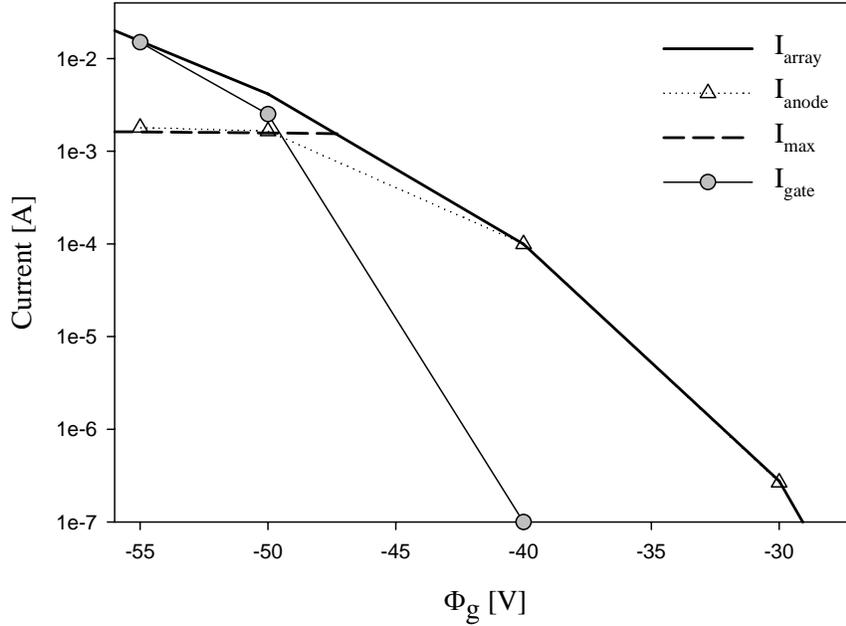


Figure 45. Anode, gate, analytical maximum and FE array currents as a function of gate potential for

$$\Phi_{anode} = \Phi_g + 150 [V]$$

Results for the second set of simulations ($\Phi_{anode} = \Phi_g + 150 [V]$) are shown in

Figure 45. In this case significant gate current is observed for the gate potentials lower than -45 Volts. Maximum anode current is achieved at approximately 50 Volts. Maximum anode current is larger than in the $\Phi_{anode} = \Phi_g + 100 [V]$ case.

Figure 47 represents a potential distribution in the computational domain. At the gate potential of 30 Volts all injected particles are collected by the anode, saturation current is not reached and virtual cathode is not formed.

For the case of $\Phi_{anode} = \Phi_g + 100 [V]$, virtual cathode is observed when gate potential reaches 40 Volts. At this voltage some portion of injected electrons is already reflected back to the gate electrode. This can be seen in Figure 48, that represents a phase space of the injected electrons. For the case of $\Phi_{anode} = \Phi_g + 150 [V]$, a layer with a potential of about -50 Volts is formed in front of the gate at $\Phi_g = -40 [V]$. But, as can be

seen from Figure 48, energy of the emitted electrons is high enough to penetrate through the potential well and reach the anode surface. For both sets of simulations formation of the virtual cathode can be clearly observed at $\Phi_g = -50$ [V]. Significant portion of electrons is reflected and then collected by the gate electrode.

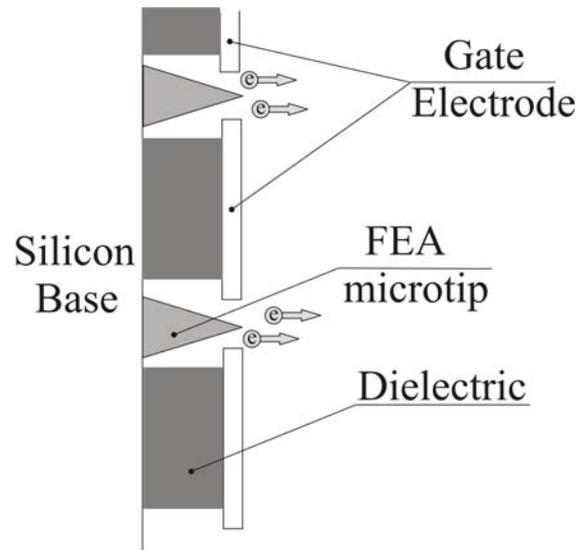


Figure 46. Schematics of FE array.

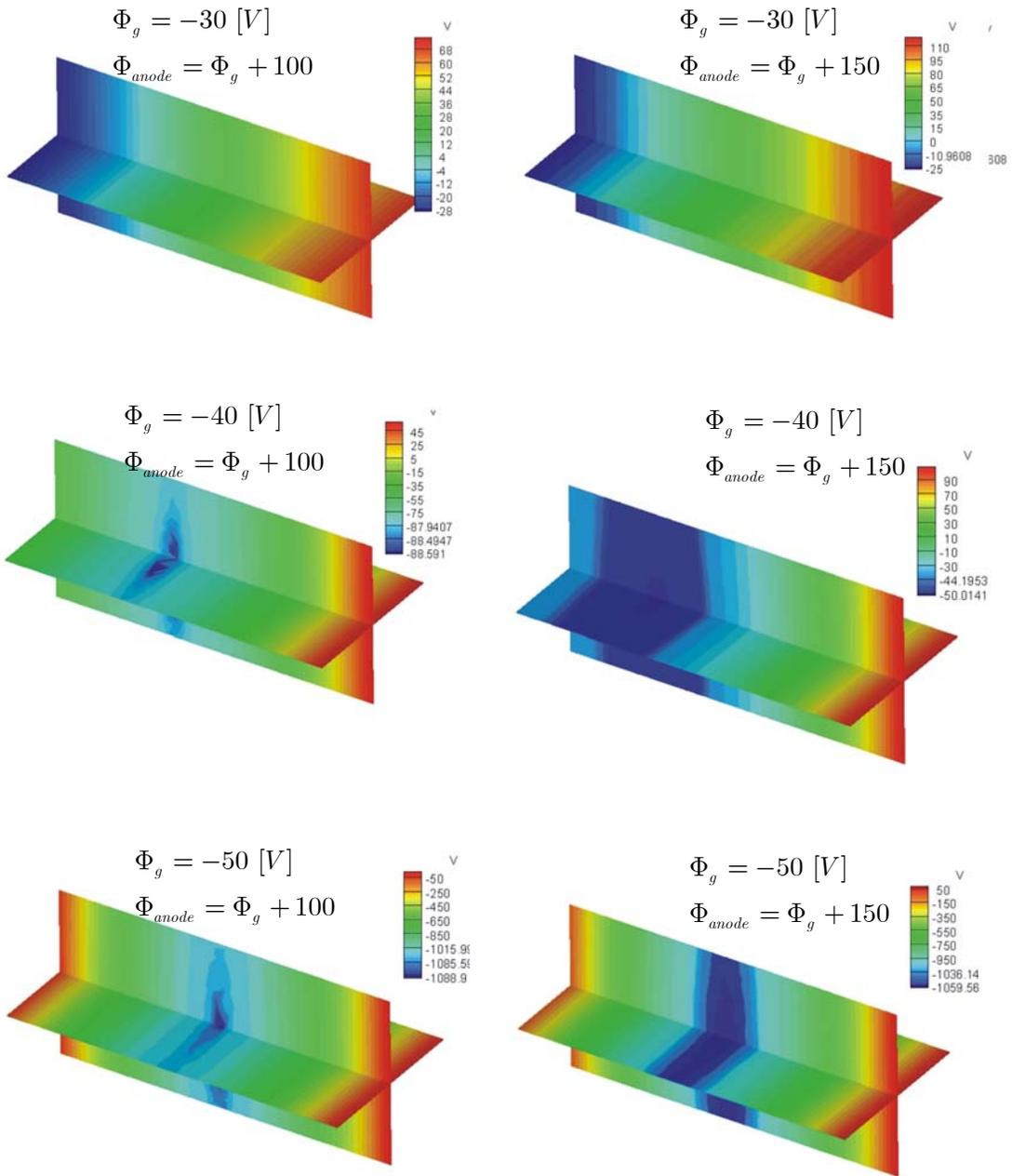


Figure 47. Potential distyribution between gate cathode and anode.

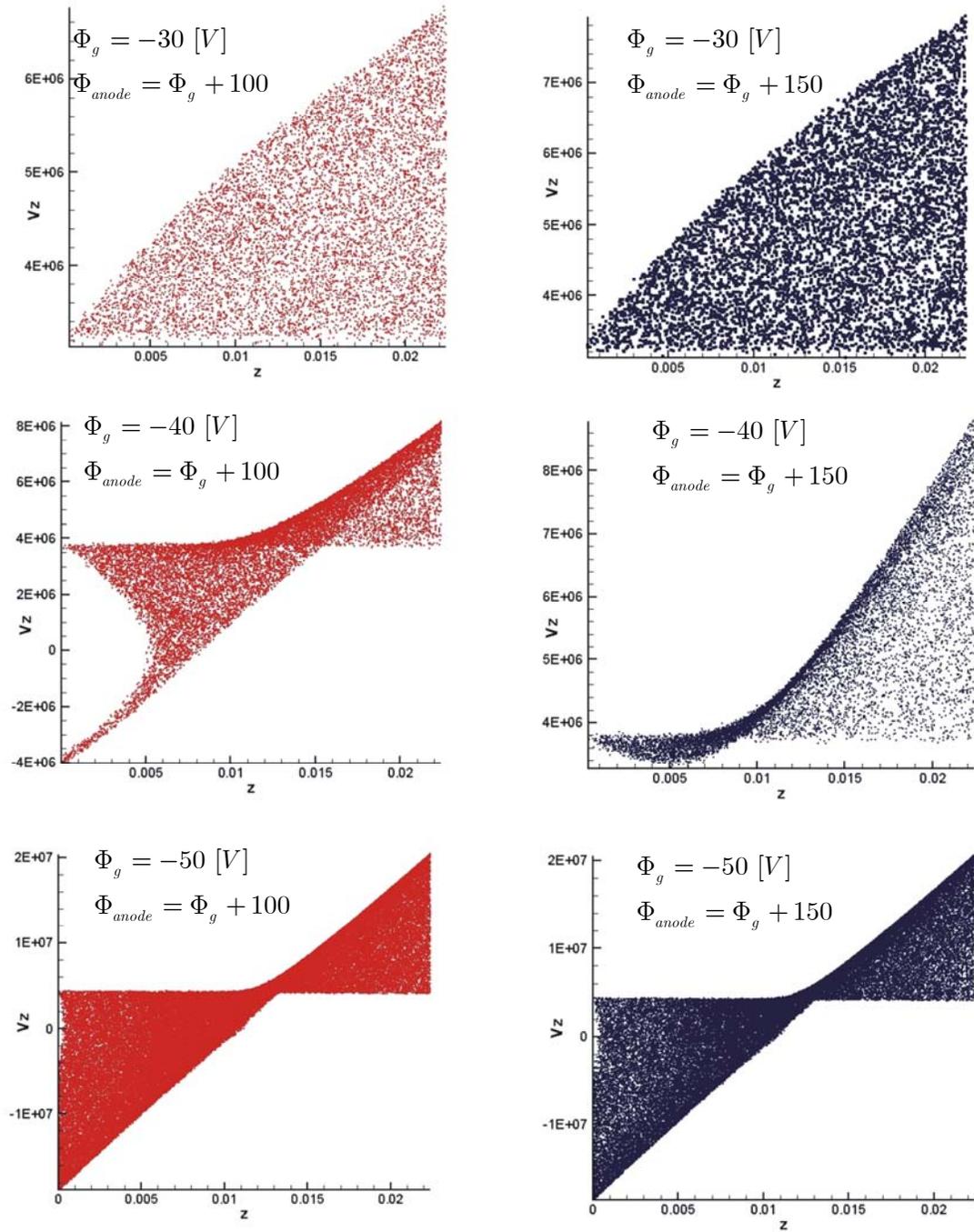


Figure 48. Phase space of the emitted electrons.

5.2 3-D PIC Simulations of a Directional Micro Retarding Potential Analyzer

A Retarding Potential Analyzer (RPA) is a plasma plume diagnostics device consisting of a series of biased electrodes. An electric potential applied to these electrodes selectively filters out plasma particles, changing an ion current collected by a collector plate. A directional microRPA (D μ RPA) under development (*Partridge and Gatsonis (2005)*; *Partridge and Gatsonis (2005)*; *Partridge (2005)*) is designed to operate in high-density flowing plasmas ($n \sim 10^{21} \text{ m}^{-3}$) and eliminate space-charge effects. It comes with and without a microchannel plate attached over the floating electrode orifice. It consists of a floating electrode (FE), a negatively biased electron retarding electrode (ERE), two positively biased ion retarding electrodes (IRE) and a negatively biased secondary emission suppression electrode (SESE), as shown in Figure 49. Energy distribution of ions may be obtained by analyzing the collector plate currents collected for different potential applied to the electrodes. The theory of operation of the D μ RPA (*Partridge (2005)*) assumes an equipotential surface inside the microchannel and neglects space charge effects. PIC simulations of the D μ RPA can elaborate the plasma processes inside the channel and offer an insight on the operation of the sensor that can lead to optimization of its design.

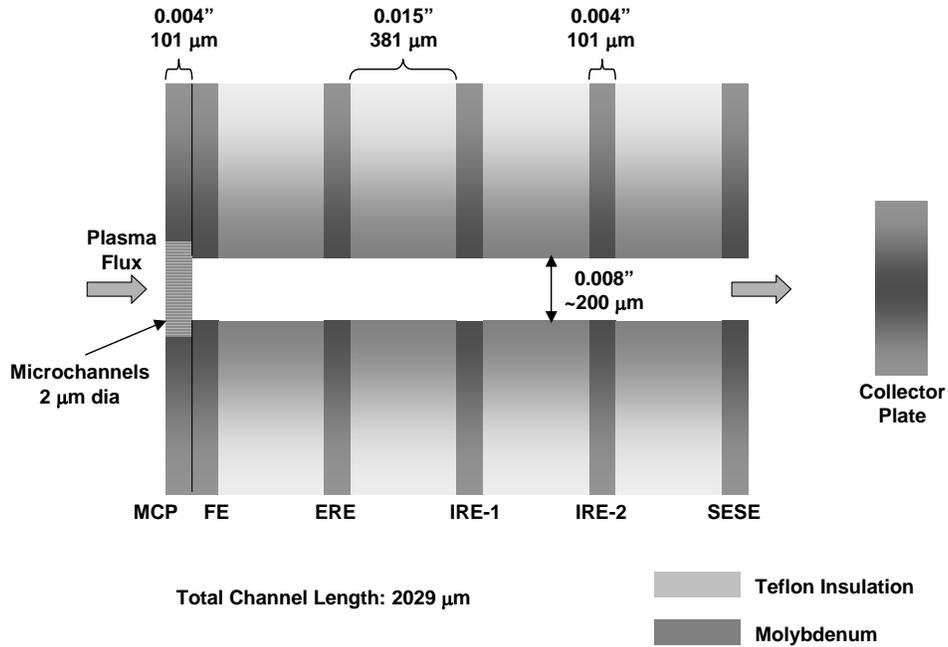


Figure 49. μRPA schematic (from Partridge *et al.* (2003)).

5.2.1 Simulations of the $\text{D}\mu\text{RPA}$ microchannel without the microchannel plate

First set of simulations corresponds to the operation of the $\text{D}\mu\text{RPA}$ in low-density plasma with no microchannel plate attached. A cylindrical computational domain representing the $\text{D}\mu\text{RPA}$ microchannel is shown in Figure 50.

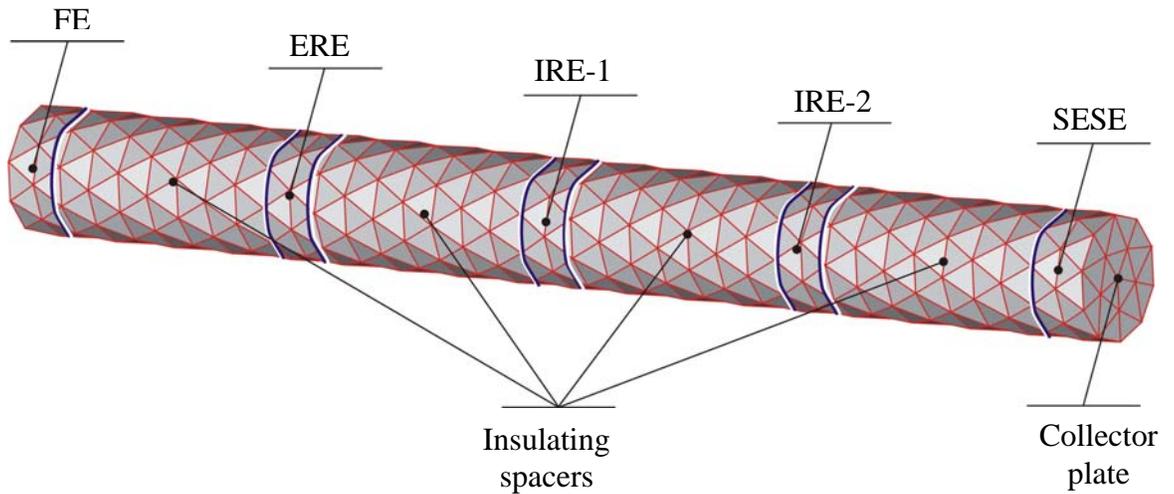


Figure 50. $\text{D}\mu\text{RPA}$ microchannel computational domain.

The domain length and radius are 2100 and 100 [μm] correspondingly. The potential of the ERE and SESE electrodes is fixed at -50 [V] for all simulation cases listed in Table 4. The potential of IRE-1 and IRE-2 electrodes changes from zero to 80 [V]. The plasma with parameters listed in Table 3 is injected from the left side of the domain into vacuum. Particles that reach any (conductor or dielectric) boundary are removed from the simulation as in the case of 100% wall absorption. Neumann boundary condition is applied at all insulating surfaces.

Species	Ion	Electron
Mass [kg]	2.0E-26	9.1E-31
Temperature [eV]	10	10
Density [m^{-3}]	10^{16}	10^{16}
Drift velocity [m/sec]	15000	0
Number of computational cells	3502	

Table 3. Input conditions and computational parameters for the D μ RPA simulations.

Case #	1	2	3	4	5	6
$\Phi_{\text{ERE}}, \Phi_{\text{SESE}}$ [V]	-50	-50	-50	-50	-50	-50
$\Phi_{\text{IRE-1}}, \Phi_{\text{IRE-2}}$ [V]	0	14.1	30	50	64.1	80
I_{CP} [A]	8.26E-8	4.72E-8	1.50E-8	1.08E-9	1.18E-10	~ 0

Table 4. Simulation parameters and results of the D μ RPA simulations.

The ion current that reaches the right side of the domain (collector plate) is evaluated. Figure 51 shows the dependence of the collector current on the ion retarding potential. Comparison is made with the analytical expression for the ion current at an applied

effective retarding potential, φ_{eff} , based on a Maxwellian ion distribution with a drift velocity \mathbf{u}_i (*Kelley (1989)*)

$$I_i = \chi \frac{A q_i n_i}{2\pi^{3/2}} \left[\frac{\exp\left(-\beta^2 (\mathbf{c}_{eff} - \mathbf{u}_i)^2\right)}{\beta} + \mathbf{u}_i \sqrt{\pi} \left\{1 - \text{erf}\left(\beta (\mathbf{c}_{eff} - \mathbf{u}_i)\right)\right\} \right] \quad (5.2.1)$$

In equation (5.2.1) \mathbf{c}_{eff} is the ion velocity corresponding to the effective retarding potential φ_{eff} , β is the inverse of the ion most probable velocity, A is the RPA orifice area and χ_{RPA} is the transmission fraction defined as

$$\chi_{RPA} = \frac{N_{cc}(S_i, D_{RPA})}{N_{cc}(S_i, \infty)} \quad (5.2.2)$$

In equation (5.2.2) S_i is the ion speed ratio, D_{RPA} is the μ RPA orifice diameter to length ratio and N_{cc} is defined from the cylindrical flux theory (*Patterson (1971)*)

$$N_{cc}(S_i, D) = \frac{n_i}{2\sqrt{\pi}\beta} \pi r^2 \left\{ \exp(-S_i^2) + S_i \sqrt{\pi} (1 + \text{erf}(S_i)) \right\} - \left[\frac{2}{D^2} (\sqrt{1+D^2} - 1) \right] - \frac{4S_i}{\sqrt{\pi}} \left[\frac{1}{D} \int_0^1 dY \int_0^{\tan^{-1}(D\sqrt{1-Y^2})} (1 + \text{erf}(S_i \cos \varphi)) \cos \varphi \exp(-S_i^2 \sin^2 \varphi) d\varphi \right] \quad (5.2.3)$$

where r is the orifice radius, φ and Y are the geometrical integration parameters.

A decrease of the current from 8.26×10^{-8} [A] in Case 1 to 1.18×10^{-10} [A] in Case 5 is observed. Practically no current is collected in Case 6. The energy distribution of the ions collected in Case 1 (steady state) is shown in Figure 52. The energy of the registered ions differs from about 70 [eV] to 120-130 [eV]. The potential distribution inside of the domain and \mathbf{c}_x vs. x phase-plots for ions and electrons at the steady state are presented in Figures 53, 54 and 55, where x is the axis along the cylindrical computational domain. The acceleration of the ions after they pass the ERE electrode as

well as between and after the ion retarding electrodes is observed in all cases. The deceleration of the ions in Case 1 and the formation of a population of ions with negative c_x velocities in Case 3 and Case 6 is evident. These effects make it possible to control the amount of the current collected by changing the ion retarding potential. In all cases presented only a few of the high-speed electrons were able to pass the negative 50 [V] barrier of the ERE. A decrease in the number of ions along the computational domain is observed because the particles that hit the surface of the RPA microchannel are removed from the simulation.

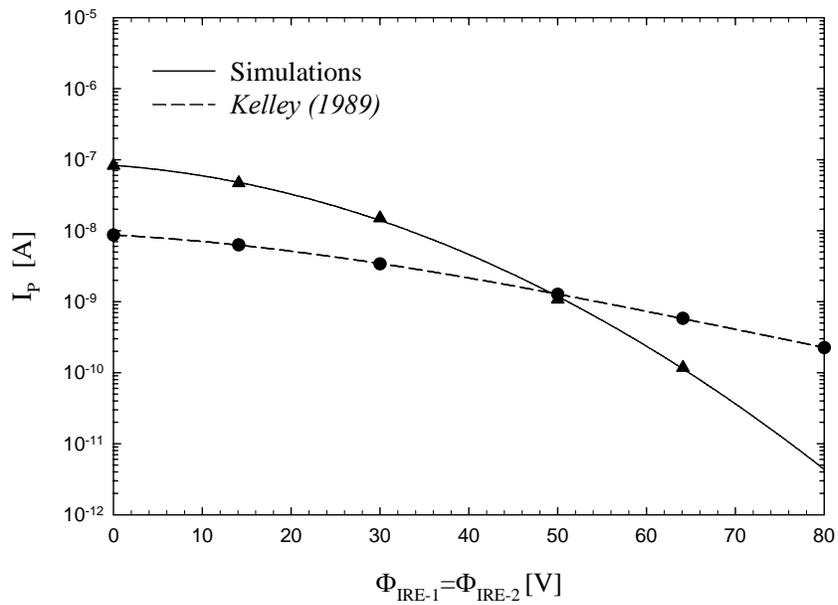


Figure 51. Collector plate current vs. ion retarding potential.

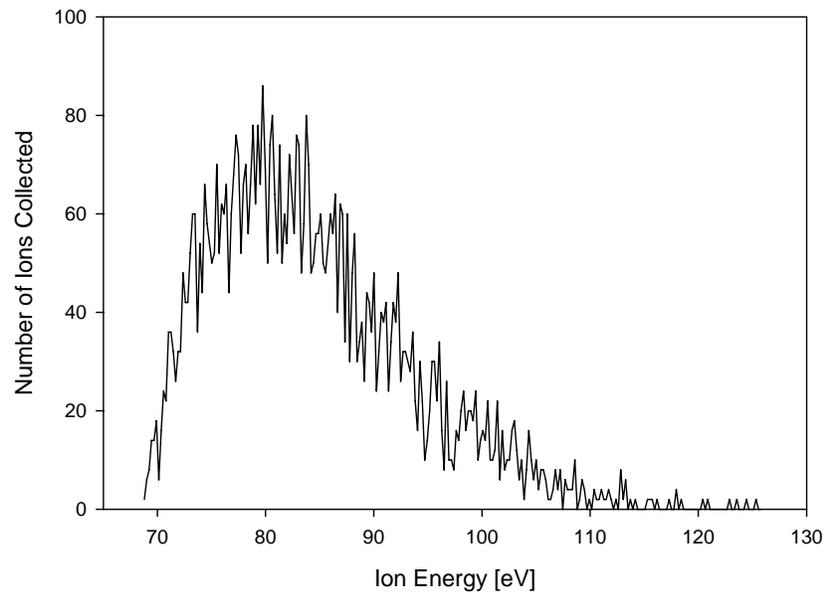


Figure 52. Energy distribution of the ions collected in Case 1 for the computational time 4.15E-8 sec.

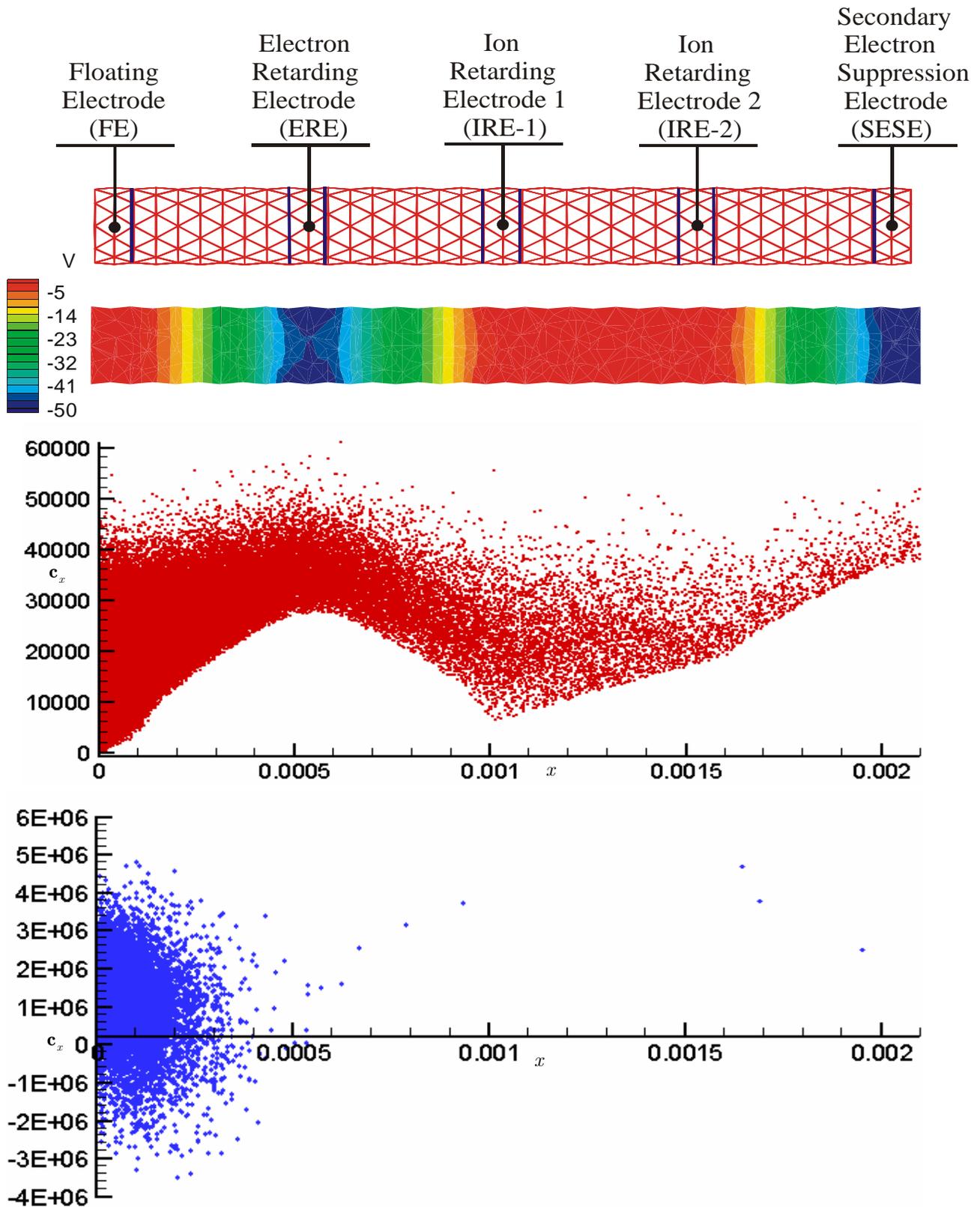


Figure 53. Side view of the RPA simulation domain, potential distribution and c_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 0$ V ; Ion (red) and Electron (blue).

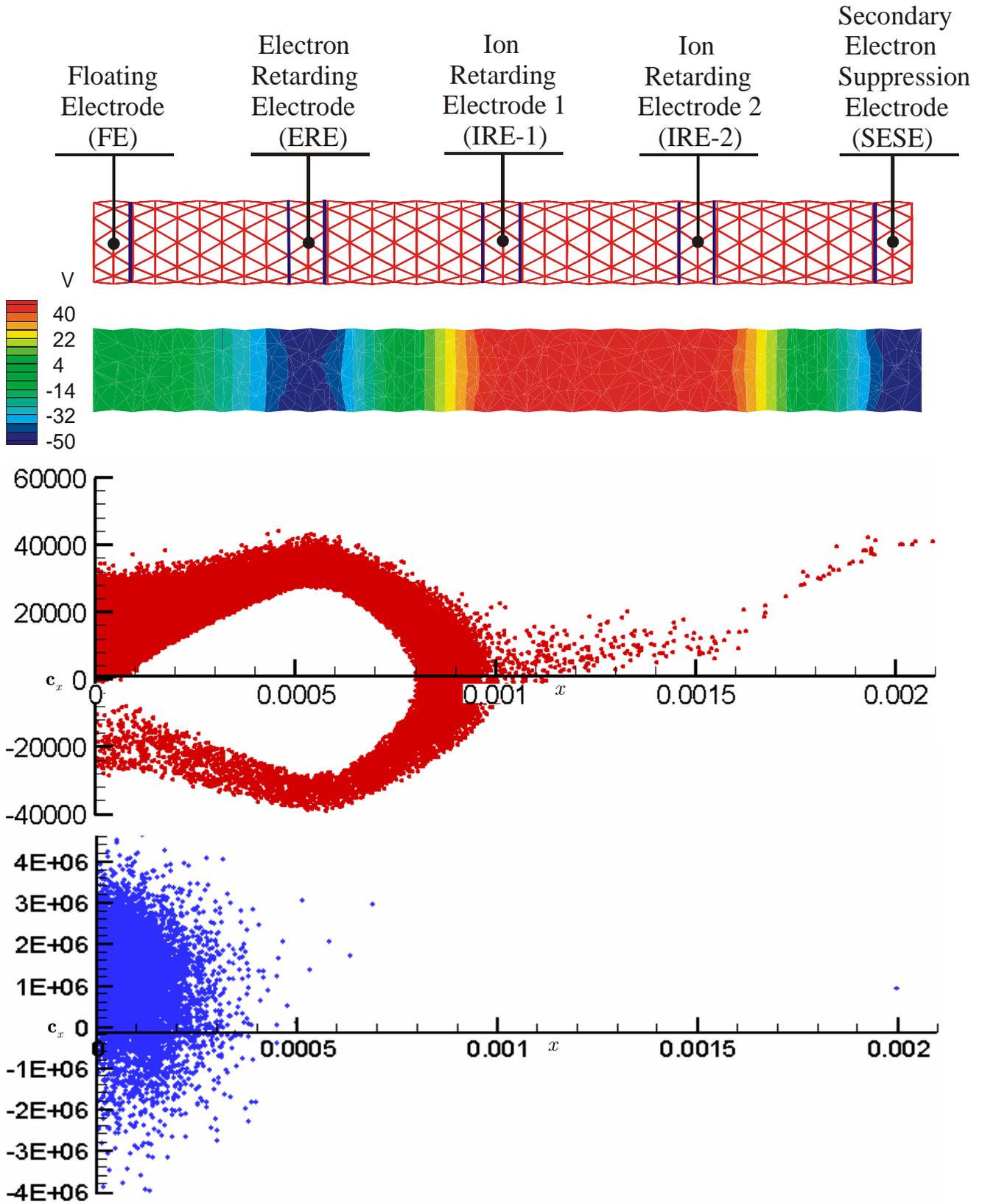


Figure 54. Side view of the RPA simulation domain, potential distribution and c_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 50$ V ; Ion (red) and Electron (blue).

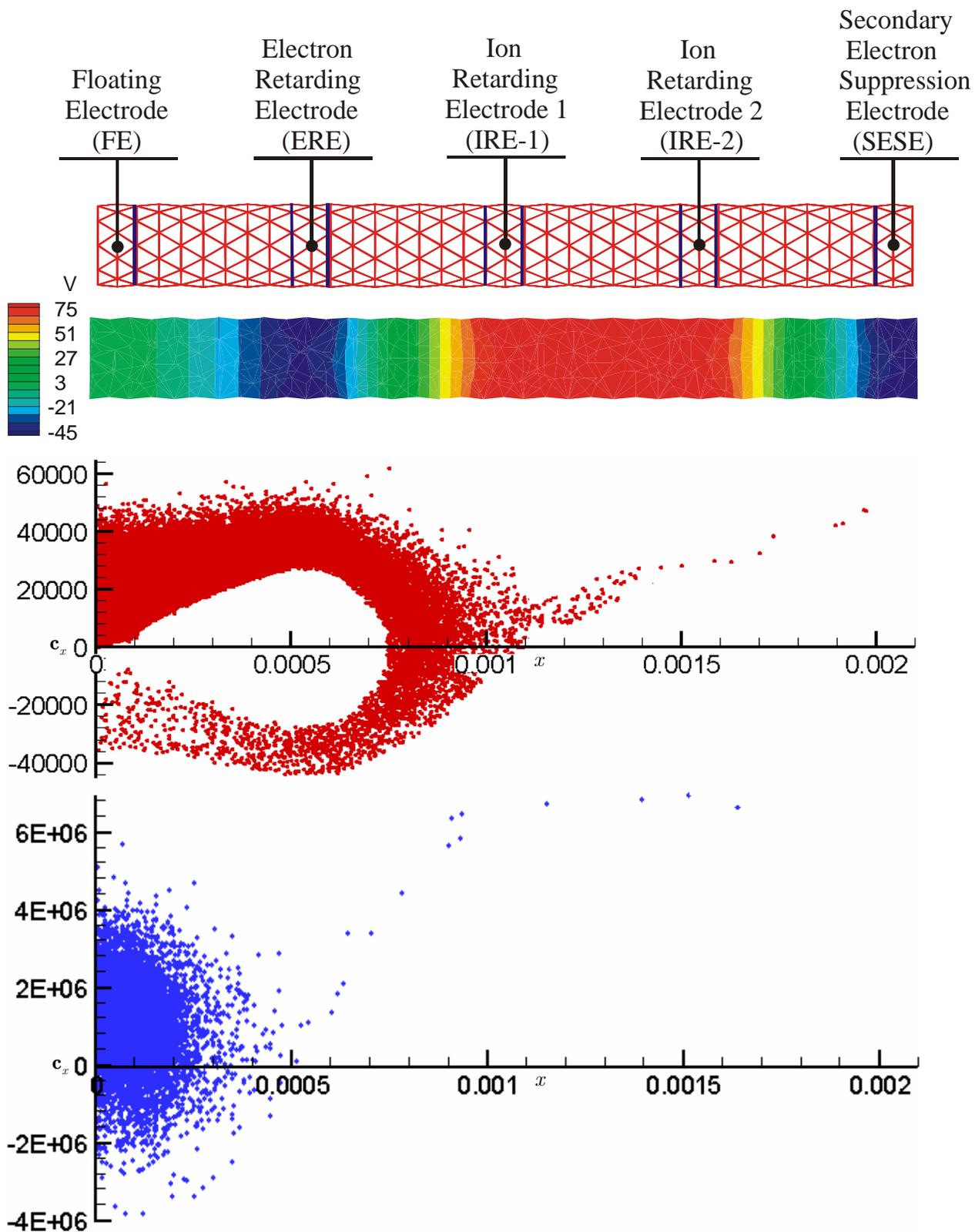


Figure 55. Side view of the RPA simulation domain, potential distribution and c_x vs x phasespace plots for $V_{ERE} = V_{SESE} = -50$ V, $V_{IRE-1} = V_{IRE-2} = 80$ V ; Ion (red) and Electron (blue).

5.2.2 Simulations of the D μ RPA microchannel with the microchannel plate

In order for the D μ RPA to operate in a high-density plasma plumes a microchannel plate of low transparency is attached in front of the floating electrode. Following *Partridge et al. (2003)*, the microchannel plate was manufactured using a 100 μm thick molybdenum plate with 3 μm diameter holes. The hole-to-hole spacing is 50 μm with 30 $^\circ$ offset as shown in Figure 56.

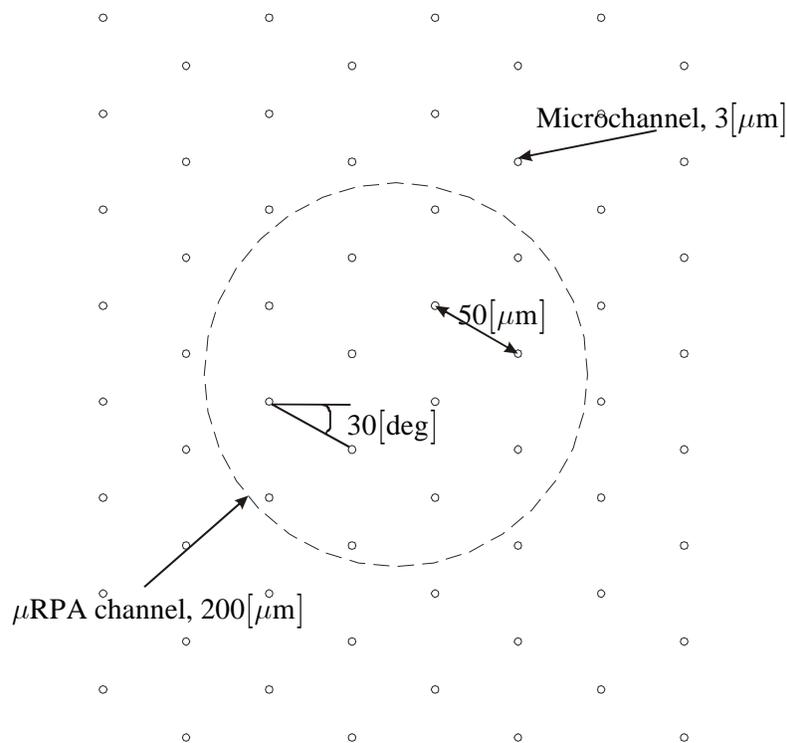


Figure 56. μ RPA microchannel plate schematic.

The microchannel plate is represented in the simulations by the array of cylinders attached to the floating electrode as shown in Figure 57. Cylinders diameter and length are 3 μm and 100 μm correspondingly.

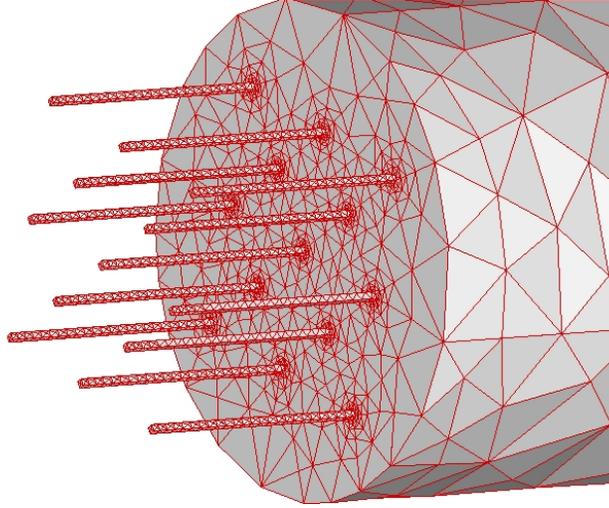


Figure 57. A part of the computational grid representing the microchannel plate.

At every timestep computational particles are injected into the domain through the microchannels with the parameters listed in Table 5. The ion current measured at the collector plate surface is presented in Table 6 and in Figure 58 for various ion retarding potentials. Comparison is made with the analytical expression (5.2.1), assuming χ to be a transmission fraction of the microchannel plate. A decrease of the current from 1.83×10^{-10} [A] to 2.65×10^{-11} [A] is observed.

Species	Ion	Electron
Mass [kg]	2.0E-26	9.1E-31
Temperature [eV]	10	10
Density [m^{-3}]	10^{18}	10^{18}
Drift velocity [m/sec]	15000	0
$\Phi_{\text{ERE}}, \Phi_{\text{SESE}}$ [V]	-50	

Table 5. Input conditions for the simulations of D μ RPA with microchannel plate.

V_{IRE-1}, V_{IRE-2} [V]	Simulations	Theory
0	1.83E-10	2.18E-10
14.1	1.71E-10	1.59E-10
30	1.48E-10	8.58E-11
50	4.97E-11	3.20E-11
64.1	2.65E-11	1.46E-11

Table 6. D μ RPA with microchannel plate simulation results.

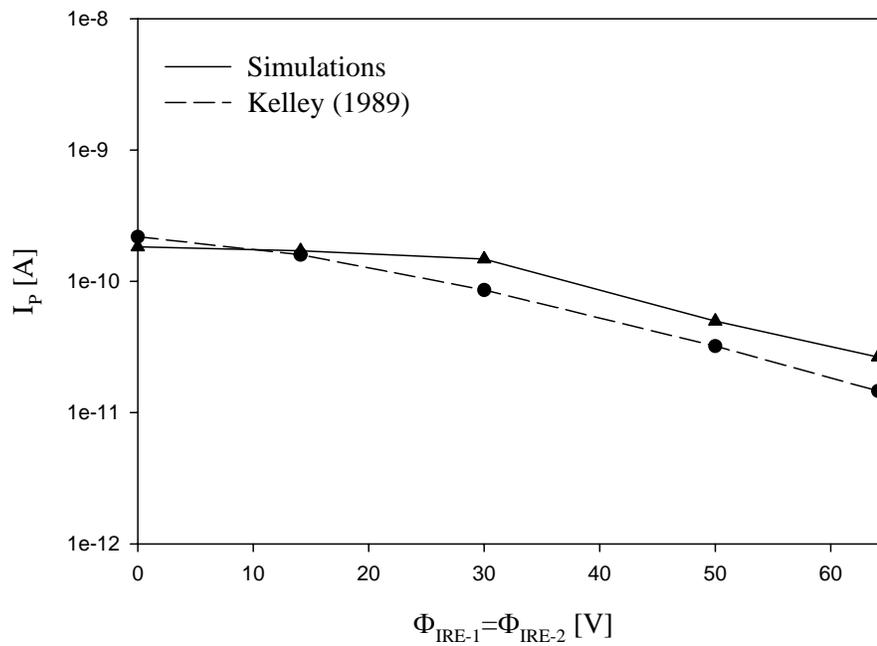


Figure 58. Collector plate ion current vs. ion retarding potential for the D μ RPA with microchannel plate simulations.

CHAPTER 6 CONCLUSIONS AND RECOMMENDATIONS

This dissertation further developed and validated a three-dimensional, electrostatic, Particle-In-Cell (3D PIC) method on unstructured Voronoi tetrahedral grids. The accomplishments regarding the computational mathematics aspects of this work are outlined below.

6.1 Summary and Accomplishments

Computational Mathematics

A particle mover and particle tracing algorithm based on the successive-neighbor search algorithm were implemented. The particle tracer takes advantage of the local coordinates and allows to save a computational time significantly.

Momentum and energy conserving schemes of zero (NGP) and first-order (Linear) were implemented in the charge assignment and force interpolation procedures. The implementation of these weighting schemes on the 3-D unstructured grids was formulated based on the theory of the long-range and smoothness constraints developed by *Hockney et al.* (1988) for uniform grids. This approach required an exact analytical and graphical (when possible) representation of the weight and shape functions on different types of computational grids starting with one-dimensional structured up to the three-dimensional unstructured.

Macroscopic plasma parameters are evaluated as cell-based and nodal-based quantities using linear Lagrange polynomials. Use of the identical weighting schemes for

both charge/electric field interpolation and plasma parameters evaluation allows for saving in computational time.

Solution of Poisson's equation is based on a finite-volume discretization. Advantage is taken of the Voronoi-Delaunay dual. Three methods of evaluating the gradient of the potential were implemented. Two algorithms utilize the divergence theorem with a cell-centered or node-centered control volume and the third algorithm is based on least-squares. In the case a of cell-centered control volume, the volume of a single computational cell is utilized in the utilized. For the node-centered method the volume of all tetrahedral that share the node of interest is used. The least-squares algorithm is implemented using nodes that share an edge of the Delaunay mesh with the node of interest.

A methodology for evaluating the heating, slowing-down and deflection times in unstructured tetrahedral meshes PIC simulations was developed. The effects of size of the timestep, number of the computational particles per cell and tetrahedral cell-edge length were investigated. Most cost-effective means of reducing numerical heating was formulated.

The code was compiled and tested successfully on Linux and Windows platforms.

Methodology and Code Validation

Validation was performed using the problem of current collection by cylindrical Langmuir probes in stationary and moving plasmas. The dependence of the collected current on the physical parameters of the background plasma and probe parameters was investigated. Current collection simulations were conducted for both the Thin Sheath

($5 \leq r_p / \lambda_D \leq 50$) and OML ($r_p / \lambda_D = 0.1$) regimes. Cases with ion to electron temperature ratios of 0.1 and 1 were considered. Cases of stationary and drifting (ion speed ratios up to 7) background plasmas were considered. Results were obtained for a wide range of the non-dimensional probe potentials ($-10 \leq e\Phi_p / kT_e \leq 10$). Simulation results were compared favorably to numerical results of *Laframboise* (1966) and analytical solutions by *Peterson and Talbot* (1970), *Kanal* (1964) and *Johnson and Murphree* (1969).

Simulation of Plasma Microdevices

The code was applied to the problem of the space-charge limited electron emission from Field Emission Array (FEA) cathodes. The operation of such FEAs can be hindered by formation of a virtual cathode that limits the current emission and forms a return current to the device. The multi-scale modeling approach used a statistical FEA emission model by *Jensen* (1994) order to obtain the required simulation parameters for the molybdenum FEA cathode. For the FEA with the cathode area of $10^{16} [\text{\AA}^2]$ and packing densities of $10^{10} [\text{tips}/\text{m}^2]$ a possible emitter current of several Amperes was predicted. Such a current correspond to an electron density of 10^{16} - $10^{17} [1/\text{m}^3]$. Formation of a potential well in front of the emitter was observed under certain injection conditions. The currents collected by the electrodes and the maximum current that may be collected by the anode were compared to the analytical predictions that take into account space-charge effects *Jensen* (1997).

The code was also used to the simulation of a micro-Retarding Potential Analyzer (RPA) under development by *Partidge and Gatsonis* (2005). The simulations aided in the

design of this new diagnostic and provided a virtual operation of the new microsensor in a regime that could not be addressed by the theory. A segmented microchannel (with the length of 2100 μm and the diameter of 200 μm with and without microchannel plate (14 microchannels with the length of 100 μm and the diameter of 3 μm) was simulated for different incoming plasma conditions. The potential applied to the Ion Retarding Electrodes was varied from 0 to 80 V. Currents collected by a collector plate were compared favorably to the theoretical predictions based on a Maxwellian ion distribution with a drift velocity (*Partridge (2005)*). While the theory is based on the equipotential assumption inside the microRPA, the 3d PIC simulations predict the complex structure of the potential inside the microRPA and the distribution functions of the accelerated ions and retarding electrons.

6.2 Recommendations for Future Work

1. Zero and first-order weighting schemes currently implemented in the code are fast and comparably easy to implement. In order to reduce numerical heating and improve the quality of the simulations a higher order weighting may be used. Implementation of the second order interpolation scheme on the uniform structured grids is straightforward. Since in nonuniform unstructured grids, mesh nodes are not aligned, it is difficult to obtain proper assignment functions based on the coordinates of the nodes outside the cell containing the particle of interest. Because it is not desirable to use mesh points outside the computational cell, containing a particle, it is possible to obtain a higher order assignment functions by introducing new shadow mesh nodes on each edge of the cell. For the second order weighting scheme it is necessary to

introduce one shadow mesh point for each cell edge, which in the case of unstructured tetrahedral mesh will be represented by a points of the intersection of Delaunay and Voronoi meshes. Poisson's equation will have to be solved on the shadow nodes, which will slow down simulations significantly. Implementation of the higher order weighting schemes also suffer increased complexity near the boundaries of the computational domain. Digital filtering and other techniques may produce a good result.

2. New external circuit boundary conditions may be implemented using the theory by *Vahedi and DiPeso* (1997) discussed in Chapter 2 for the floating potential boundary condition. A capacitive circuit consisting of an ideal voltage source and capacitor is useful for the simulations of the capacitative discharges. A backwards difference expression for the series RLC circuit which is required in many plasma discharges simulations can be implemented following *Verboncoeut et al.* (1993) with the second order of accuracy in time. A problem with implementation of the no-flux boundary conditions with the external circuits is discussed in *Vahedi and DiPeso* (1997). A capacitance matrix may be formulated in terms of charge in order to obtain an electrostatic potential in the system with the external circuit. New particle-surface interaction algorithms are required in order to properly model boundary conditions.
3. Several applications involve plasmas with species of significantly different number densities. In order to have a reasonable number of computational particles for each species and properly simulate collisions between them it is important to develop algorithms for the particles with different computational weights.

4. In order to run a large-scale simulations of plasma devices one should have a possibility to run the code on the parallel platform. This will require a mesh partitioning technique as well as faster and more flexible grid generator. Significant changes will have to be made in the way the global information is currently stored in the code. Most of the subroutines must be updated using MPI or OpenMP techniques in order to properly exchange information between the processors. Using variable timesteps and computational particles weights between processors may significantly increase computational efficiency of the code.

REFERENCES

- Aguero, V., "Experimental Results and Considerations on Use of Field Emission Devices in Space", 37th Joint Propulsion Committee Conference and Exhibit, 8-11 July, 2001, Salt Lake City, Utah.
- Allievi, A., Bermejo, R., "A Generalized Particle Search-Locate Algorithm for Arbitrary Grids", *Journal of Computational Physics*, 132, 157-166, 1997.
- Bird, G.A., "Molecular Gas Dynamics and the Direct Simulation of Gas Flows", Clarendon Press, Oxford, 1994.
- Birdsall, C.K., A. B. Langdon, "Plasma Physics via Computer Simulations", *Plasma Physics Series*, 1991.
- Birdsall, C.K., Fuss, D., "Clouds-in-Clouds, Clouds-in Cells Physics for Many-Body Plasma Simulation", *Journal of Computational Physics*, 3, 494-511, 1969.
- Birdsall, C.K., Maron, N., "Plasma Self-Heating and Saturation due to Numerical Instabilities", *Journal of Computational Physics*, 36, 1-19, 1980.
- Boris, J.P., "Relativistic Plasma Simulation-Optimization of a Hybrid Code", *Proceedings of the Fourth Conference on Numerical Simulation of Plasma*, Naval Res. Lab, Washington D.C., 3-67, 2-3 November, 1970.
- Brodie, I., Schwoebel, P.R., "Vacuum Microelectronic Devices", *Proceedings of the IEEE*, Vol.82, No.7, July, 1994.
- Catwright, K.L., Verboncoeur, J.P., Birdsall, C.K., "Loading and Injection of Maxwellian Distribution in Particle Simulations", *Journal of Computational Physics*, 162, 483-513, 2000.

- Celik, M., Santi, M., Cheng S., et al., “Hybrid-PIC Simulation of a Hall Thruster Plume on an Unstructured Grid with DSMC Collisions”, IEPC-03-134.
- Dawson, J.M., “Particle Simulation of Plasmas”, Reviews of Modern Physics, Vol.55, No.2, April, 1983.
- Ferraro, R.D., Liewer, P.C., et al. “Dynamic Load Balancing for a 2D Concurrent Plasma PIC Code”, Journal of Computational Physics, 109, 329-341, 1993.
- Gatsonis, N.A., Spirkin A., “Unstructured 3D PIC Simulations of Field Emission Array Cathodes for Micropropulsion Applications”, AIAA-2002-3687, 38th Joint Propulsion Conference, July 7-10, 2002, Indianapolis, IN.
- Gatsonis, N.A, Byrne, L.T., Zwahlen, J.C., et al., “Current-Mode Triple and Quadruple Langmuir Probe Methods With Applications to Flowing Pulsed Plasmas”, IEEE Transactions on Plasma Science, Vol.32, No.5, October, 2004.
- Gatsonis N.A., Yin, X., “Hybrid (Particle-Fluid) Modeling of Pulsed Plasma Thruster Plumes”, Journal of Propulsion and Power, Vol.17, No.5, September-October, 2001.
- Hammel, J., K. Kovalev, N.A. Gatsonis, “Unstructured Adaptive Monte Carlo Simulations of Flows in Micronozzles”, 35th AIAA Thermophysics Conference, Anaheim, CA, June, 2001.
- Hammel, J., “Development an Unstructured 3-D Direct Simulation Monte Carlo/Particle-in-Cell Code and the Simulation of Microthruster Flows”, Master’s Thesis, WPI, 2002.
- Hockney, R.W., “Measurements of Collision and Heating Times in a Two-Dimensional Thermal Computer Plasma”, Journal Of Computational Physics 8, 19-44, 1971.

- Hockney, R.W., Eastwood J.W., "Computer Simulations Using Particles", IOP Publishing, 1999.
- Hoegy, W.R., Wharton, L.E., "Current to a Moving Cylindrical Electrostatic Probe", Journal of Applied Physics, Vol.44, No.12, December, 1973.
- Jensen, K.L., "Field Emitter Arrays for Plasma and Microwave Source Applications", Physics of Plasmas, Vol.6, No.5, May, 1999.
- Jensen, K.L., Zaidman, E.G, "Analytic Expressions for Emission Characteristics as a function of Experimental Parameters in Sharp Field Emitter Devices", Journal of Vacuum Science Technology, B 13(2), Mar/Apr, 1995.
- Jensen, K.L., "Theory and Simulation of Field Emission From Microstructures", Naval Research Laboratory: Cathode Workshop, May 10-11, 2001, Washington, DC.
- Jensen, K.L., Kodis, M.A., Murphy, R.A., Zaidman, E.G., "Space Charge Effects on the Current-Voltage Characteristics of Gated Field Emitter Arrays", Journal of Applied Physics, 82(2), 15 July, 1997.
- Jensen, K.L., Zaidman, E.G., Kodis, M.A., et al., "Analytical and Seminumerical models for Gated Field Emitter Arrays. Theory", Journal of Vacuum Science Technology, B 14(3), May/Jun, 1996.
- Jensen, K.L., Abrams, R.H., Parker, R.K., "Field Emitter Array Development for High Frequency Applications", Journal of Vacuum Science Technology, B 16(2), Mar/Apr, 1998.
- Kanal, M, "Theory of Current Collection of Moving Cylindrical Probes", Journal of Applied Physics, Vol.35, No.6, June, 1964.

- Karmesin, S.R., Liewer, P.C., Wang, J., “A Parallel Three-Dimensional Electromagnetic Particle-in-Cell Code for Non-Orthogonal Meshes”, CRPC-TR96731, September, 1996.
- Kiel, R.E., “Electrostatic Probe Theory for Free Molecular Cylinders”, AIAA Journal Vol.6, No.4, 1967.
- Kovalev, K., “Development of a 2-D/3-D Unstructured Adaptive Grid Generator for Direct Simulation Monte Carlo Computations, Master’s Thesis, WPI, 2000.
- Laframboise, J.G., “Theory of Spherical and Cylindrical Langmuir Probes in a Collisionless, Maxwellian Plasma at Rest”, UTIAS Report No. 100, June, 1966.
- Langdon, A.B., ““Energy-Conserving” Plasma Simulation Algorithm”, Journal of Computational Physics, 12, 247-268, 1973.
- Lawson, W.S., “Particle Simulation of Bounded 1d Plasma Systems”, Journal of Computational Physics, 80, 253-76, 1989.
- Lohner, R., Ambrosiano, J., “A Vectorized Particle Tracer for Unstructured Grids”, Journal of Computational Physics, 91, 22-31, 1990.
- Mardahl, P.J., “PIC Code Charge Conservation, Numerical Heating, and Parallelization; Application of XOOPIC to Laser Amplification via Raman Backscatter”, Ph.D. dissertation, University of California, Berkeley, 2001.
- Marrese, C.M., Wang, J.J., “Space-Charge-Limited Emission from Field Emission Cathodes for Electric Propulsion and Tether Applications”, AIAA, 2000.
- Marrese, C.M., Polk, J.E., “Molybdenum Field Emission Array Cathode Performance in Primarily Xenon Environments: Preliminary Experimental Results at Low Operating Voltages”, AIAA-2000-3266.

- Marrese, C.M., “Compatibility of Field Emission Cathode and Electric Propulsion Technologies”, Ph.D. dissertation, University of Michigan, 1999.
- Marrese, C.M., Gallimore, A.D., Polk, J.E., et al., “Field Emission Array Cathodes for Electric Propulsion Systems”, AIAA 98-3484.
- Miller, R.H., Combi, M.R., “A Coulomb Collision Algorithm for Weighted Particle Simulations”, Geophysical Research Algorithm, Vol.21, No.16, 1735-1738, August 1, 1994.
- Mitchner, M., Kruger, C.H., “Partially Ionized Gases”, Wiley Series in Plasma Physics, 1992.
- Montgomery, D.C., Tidman, D.A., “Plasma Kinetic Theory”, McGraw-Hill Book Company, 1964.
- Morris, D., Gilchrist, B., Gallimore, A., “Development of Field Emitter Array Technology for ED Tether Propulsion”, 2001 International Electric Propulsion Conference, October, 2001, Pasadena, California.
- Nanbu, K., Yonemura, S., “Weighted Particles in Coulomb Collision Simulations Based on the Theory of a Cumulative Scattering Angle”, Journal of Computational Physics, 145, 639-654, (1998).
- Partridge, J., Gatsonis, N. A. et al. “Design and Analysis of a Directional Micro-Retarding Potential Analyzer for High-Density Plumes”, AIAA-2003-5172, 39th Joint Propulsion Conference, July 20-23, 2003, Huntsville, AL.
- Pepper, D.W., Heinrich, J.C., “The Finite Element Method: Basic Concepts and Applications”, Hemisphere Publishing Corporation, 1992.

- Peterson E.W., Talbot L., "Collisionless Electrostatic Single-Probe and Double-Probe Measurements", AIAA Journal, Vol.8, No.12, 1970.
- Rambo, P.W., "Numerical Heating in Hybrid Plasma Simulations", Journal of Computational Physics, 133, 173-180, 1997.
- Spirkin, A., Gatsonis, N.A., "Unstructured 3D PIC Simulations of the Flow in a Retarding Potential Analyzer", Computer Physics Communications, 164, 2004, 383-389.
- Spirkin, A., Gatsonis, N.A., "Unstructured 3D PIC Simulations of Plasma Flow in a Segmented Microchannel", 36th AIAA Thermophysics Conference, July, 23-26, 2003, Orlando, Florida.
- Trubnikov, B.A., "Particle Interactions in a Fully Ionized Plasma", Reviews of Plasma Physics, Consultants Bureau, 1965.
- Ueda, H., Omura, Y., Matsumoto, H., Okuzawa, T., "A Study of the Numerical Heating in Electrostatic Particle Simulations", Computer Physics Communications, 79, 1994, 249-259.
- Vahedi, V., DiPeso, G., "Simultaneous Potential and Circuit Solution for Two-Dimensional Bounded Plasma Simulation Codes", Journal of Computational Physics, 131, 149-163, 1997.
- Verboncoeur, J.P., "Particle Simulation of Plasmas: Review and Advances", Plasma Physics and Controlled Fusion 47, 2005, A231-A260.
- Wang, J., Liewer, P., Decyk, V., "3D Electromagnetic Plasma Particle Simulations on a MIMD Parallel Computer", Computer Physics Communications, 87, 1995, 35-53.

Wang, J., Kondrashov, D., Liewer, P.C., Karmesi, S.R., “Three-Dimensional Deformable Grid Electromagnetic Particle-in-Cell for Parallel Computers”, *Journal of Plasma Physics*, 1995.

Wang, J., Lai, S.T., “Virtual Anode in Ion Beam Emissions in Space: Numerical Simulations”, *Journal of Computational Physics*, Vol.34, No.6, November-December, 1997.

Watson, D. F., “Computing the Delaunay Tessellation with Application to Voronoi Prototypes,” *The Computer Journal*, Vol.24 (2),