5-2016

INVESTIGATION OF LANGMUIR PROBES IN NON-MAXWELLIAN PLASMA USING PARTICLE-IN-CELL (PIC) MODELING

Densu Aktas Lister
University of Tennessee - Knoxville, daktas@utsi.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Densu Aktas Lister entitled "INVESTIGATION OF LANGMUIR PROBES IN NON-MAXWELLIAN PLASMA USING PARTICLE-IN-CELL (PIC) MODELING." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Trevor M. Moeller, Major Professor

We have read this dissertation and recommend its acceptance:

Gary Flandro, Roy J. Schulz, Christian G. Parigger

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

INVESTIGATION OF LANGMUIR PROBES IN NON-MAXWELLIAN PLASMA USING PARTICLE-IN-CELL (PIC) MODELING

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Densu Aktas Lister

May 2016

To my mother, Sevim (Sara) Tenk, a strong and gentle soul who thought me to trust in God, believe in hard work and who has never let me give up on my dreams.

To my late grandparents: Grandfather Morris C. Early and Grandmother Mildred M. Early who always encouraged me to stay on it. Unfortunately, they were unable to see me through this milestone, as they passed away in 2012 (grandfather) and in 2014 (grandmother). Nevertheless, they have been with me throughout and have provided me with financial security and unflappable confidence.....

This is for you Grandpa!

"The LORD himself will fight for you. You won't
have to lift a finger in your defense!"
Exodus 14:14 (NLT)

Acknowledgements

On the very outset of this dissertation, it is my great pleasure to acknowledge people who have given me guidance, help and encouragement.

I am ineffably indebted to my academic advisor, Dr. Trevor M. Moeller for conscientious guidance and encouragement. He has always been an inspiring professor, a perfect guide, an exceptional faculty advisor and mentor who has gone above and beyond in academic advising. He has always believed in me, and challenged me to go beyond what I think I could do. I would not have been here now if he had not been there for me.

I am extremely thankful and pay my gratitude to the members of my dissertation committee: Dr. Gary Flandro, Dr. Roy Schulz and Dr. Christian G. Parigger for their guidance and support.

I acknowledge with a deep sense of reverence, my gratitude towards the people of First Babtist Church in Nashville: Our wonderful Pastor, Dr. Frank Lewis, Dr. Russ Roach, Uncle Charles and Aunt Barbara Parker, Uncle Fred Linkenhoker, Aunt Grace Swift, Aunt Ruth Bradly, Uncle Charles and Aunt Martia DeMore, Mrs. Carolyn Brakefield, Mr. Bob and Mrs. Richie Whaley, Aunt Julia Smith, Mr. Kyle and Mrs. Becky Cantrell, Mr. Howard and Mrs. Shirley Murphy, Mr. Vern Powers, and my medical doctor, Dr. Donna Scudder; the people of ELS and those in Nashville: Jeff and Rebecca Hutcheson, Uncle Victor and Aunt Judy Davis, Mr. Jimmy and Mrs. Karen Early, and Keudji Roland; the people of MTSU and those in Murfreesboro: Dr. Ron and Mrs. Gloria McBride, Mrs. Caron Case, Ms. Angie Melton, Jeff and Judy Fryer, Mr. David Gore, Dr. Robert Carlton; and those who live far away: Dan and Lois Frasier, Grandpa Floyd and Grandma Carol Moss who provided support and assistance to take care of my dear mother and my little brother, and still create an opportunity to pursue my education over the years. My deepest appreciation is expressed to them for their love, understanding, inspiration and prayers. Without their blessings and encouragement, I would not have been able to finish this work.

Through the years of my being here, Mrs. Susan Hendel (my adopted grandmother) has always remained near me. She has always recognized the value of my work and stood behind the cause. She has encouraged, helped, cheered, criticized and supported me when I needed. I have started my journey arriving on her door steps in a storm ten years ago. I was only seventeen and had just graduated from the university. On my way, many others have joined us; some of them took over her flag, like Grandpa Morris and Grandma Mildred Early. When she rested, and also when my grandpa and grandma had to leave us to be with God, Grandma Susan got back to the course and took over the flag so that I could reach the final. It felt like an athlete preparing an olympics; the nation did everything they could so that the athlete could win. Each time my feet hit a rock and I fell, some of those people joined her to help me to get up and continue my race. I am closing this chapter of my life with you, Grandma Susan. My deepest gratitude and heartfelt thanks to you for your continues support and assistance.

Particular thanks must also be recorded to Mr. Paul Gloyer for his support and encouragement over the years both in terms of completing this and professionally. A specific note of gratitude to the friends at UTSI and those in Tullahoma: Tina Rice, Matthew Duran, LeAnna Truesdale, Josh Batterson, and Lt. Col. Kim Nelson (AAFB), Mrs. Belinda Stuart, Mr. John LaCook and Mrs. Vanessa Grammer (Kia technical service, for their kind support after my accident); staff at UTSI and UTK: Mrs. Dana Hale, Mrs. Robin Nee, Mrs. Courtney Maricle, Mrs. Sylvia Miller, and especially Ms. Jennifer Boyles and Mrs. Charlene Hane (Angle/Miss. Della Street) for going above and beyond their ways to keep me on my track and encouraging me to reach the finish line.

I also would like to acknowledge the contribution of Nicholas Lister; it has been nice knowing you, Nick! We have gone through very hard times, but we also had good times. Thank you for your contribution to my life.

At last but not least gratitude goes to all my friends and family members who directly or indirectly helped me to complete this dissertation. Any omission in this brief acknowledgement does not mean lack of gratitude.

I am also thankful that with my lovely kitty, Pamuk (Cotton), in my life, there is always one thing that is right no matter what else is wrong. I am thankful for the way she helped me heal and took away my pain. I am thankful for the peace and joy she adds to my life on this day, and every day.

Thanking You,

-Densu Aktas

Abstract

This dissertation explores the development of a capability for simulating the plasma dynamics of Langmuir probes (LP) in complex plasmas where the velocity distributions are non-equilibrium and the electron energy spectrum is non-Maxwellian with respect to laboratory and space experiments. The results of this investigation are interpreted to give recommendations for design and use of LPs. This work is conducted using computational techniques to create the exact plasma conditions of the experimental testing environments. The investigations address the following topics:

- development of a technique to model non-Maxwellian physics,
- modification of a baseline-technique and optimization of it for this application,
- creation of three-dimensional PIC code to include non-Maxwellian physics,
- evaluation of effectiveness of enhanced PIC simulations,
- demonstration of use of enhanced PIC code to conduct and simulate LP experiments in non-ideal conditions such as in the EP thruster.

Major results can be summarized as follows: PROBEPIC (PIC code) is modified for interpreting data obtained using an electrostatic-probe in an ion-beam to implement OML (thick-sheath) and SL (thin-sheath) current-collection theories. PROBEPIC was modified to model the non-Maxwellian plasmas, and test-cases are presented to validate the simulation against published empirical data. General equations for current-collection and I-V curves for cylindrical, planar and spherical LP in isotropic and anisotropic non-Maxwellian plasmas are examined. Distribution functions are introduced as a method of measuring the deviation from Maxwellian. Existing non-Maxwellian techniques (i.e., Druyvesteyn, bi-Maxwellian) were modified to model the environment around LP in an EP system. The EEDF has been investigated with LP to overcome some limitations of Druyvesteyn method. The EEDF changes from Druyvesteyn to bi-Maxwellian with decreasing pressure. Therefore, bi-Maxwellian method was also implemented in the system to obtain utmost results in modelling non-isotropic plasmas. This innovative model, which was then integrated into PROBEPIC, was used to simulate operation of LP in a series of validation and demonstration cases. The effective-T_e, n, and V_p were obtained from the LP simulations, and I-V traces were created. The code can predict the high-energy ions, and experimental measurement of the EEDF, providing useful information for the development of a state-of-the-art new plasma and EP diagnostic capabilities.

Table of Contents

List of Tables

List of Figures

Nomenclature

Abbreviations

DMC	Direct Monte Carlo
EED	Electron energy distribution
EEDF	Electron energy distribution function
EP	Electric propulsion
EPT	Electric propulsion thruster
GC	Guiding center
GPU	Graphics processing unit
I-V	Current-Voltage
LP	Langmuir probe
MEMS	Microthruster digital propulsion system
MC	Monte Carlo
MPD	Magnetoplasmadynamic
LISA	Laser interferometer space antenna
LSTF	Large vacuum test facility
OML	Orbital motion limited
PEPL	Plasma electric propulsion laboratory
PIC	Particle-in-cell
PPT	Pulse plasma thruster
SL	Sheath limited

Symbols

A	Normalization constant $[-]$
A_{eff}	Effective collection area of a probe $[m^2]$
A_p	Probe surface area $[m^2]$
A_s	Sheath area $[m^2]$
a	Coefficient dependent of r_p/λ_D (CH-2)
B	Magnetic field vector $[T]$, $(1Torr \approx 133.3Pa)$
b	Coefficient dependent of r_p/λ_D (CH-2)
c	Speed of light in vacuum, 2.9979×10^8 $[m/s]$ (Appendix)
d_C	Effective impact parameter $[m]$
d_o	Effective radius of neutral particle $[m]$
d_{sh}	Sheath width $[m]$
e	Elementary charge, 1.602×10^{-19} $[C]$
$\mathbf{E}$	Electric field vector $[V/m]$

E_{av}	Average kinetic energy [J] or [eV]
E_o	Rate constant for ionization energy (material dependent)
E_s	Initial energy [J] or [eV], $(1eV = 1.6021 \times 10^{-19}$ [J])
E_x	x-directed electric field [V/m],
ϵ	Limiting values (CH-4)
$\langle \epsilon_{kin} \rangle$	Mean kinetic energy [J] or [eV], $(1eV = 1.6021 \times 10^{-19}$ [J])
ϵ_t	Energy where the slop of the distribution change [J] or [eV], $(1eV = 1.6021 \times 10^{-19}$ [J]) (CH-2)
f	Distribution function [−]
$f(\epsilon)$	Normalized EEDF [−]
$f(\upsilon)$	Normalized velocity distribution function $[m^{-2}s]$
f_α	Initial distribution function
$f_M(\upsilon)$	Maxwell distribution of particle speed
$F(\epsilon)$	Electron energy distribution function (EEDF)
F_c	Centripetal force [N]
$F_M(\epsilon)$	Maxwellian EEDF (CH-2)
$F(\upsilon)$	Velocity distribution function [−]
g	Acceleration of gravity, 9.807 $[m/s^2]$
I	Current [A]
I_e	Electron current [A]
I_{es}	Electron saturation current [A]
I_i	Ion current [A]
I_{is}	Ion saturation current [A]
I_p	Probe current [A]
I_{ps}	Probe saturation $(V = V_p)$ current [A]
I_{sp}	Specific impulse [s]
J_e	Electron current density $[A/m^2]$
k_B	Boltzmann constant, 1.38×10^{-23} [J/K] or 0.00008617 [eV/K]
K_i	Ionization rate constant [−]
K_n	Knudsen number [−]
K_o	Rate constant (material dependent) [−] $(1eV = 1.6021 \times 10^{-19}$ [J])
l	Distance between two walls [m]
l_{ins}	Length of probe insulator [m]
l_p	Probe length [m]
L	Characteristic length [m]
m	Mass [kg]
m_e	Mass of electron, 9.109×10^{-31} [kg]
m_i	Mass of ion [kg]

m_p	Mass of proton [kg]
$\dot{m}$	Mass flow rate [kg/s]
M	Magnetic moment of the particle (Appendix)
n	Number density $[m^{-3}]$
n_o	Undisturbed plasma density $[m^{-3}]$
n_e	Electron number density $[m^{-3}]$
n_{es}	Electron density at the sheath edge $[m^{-3}]$
n_i	Ion number density $[m^{-3}]$
n_{is}	Ion density at the sheath edge $[m^{-3}]$
n_n	Number density of neutral particles $[m^{-3}]$
n_s	Number density of a species $[m^{-3}]$; Sheath edge density $[m^{-3}]$
n_{tar}	Number density of target particles $[m^{-3}]$
N	Number of particles
N_e	Number of electrons
N_i	Number of ions
P	Impact parameter [−]
$\vec{p}$	Momentum of particle (Appendix)
p_e	Probe effective radius [m]
q	Charge of a particle [C] (denoted by its subscript)
r_c	Distance of closest approach for two charged particles [m]
r_d	Average distance between particles [m]
r_L	Larmor radius [m]
r_p	Probe radius [m]
r_{ph}	Probe holder diameter [m]
s	Sheath
t	Time [s]
T	Thrust [N]; Temperature [K] or [eV]
T_e	Electron temperature [K] or [eV], $(1eV = 7{,}733\ [K])$
T_i	Ion temperature [K] or [eV], $(1eV = 7{,}733\ [K])$
T_s	Temperature of a species [K] or [eV], $(1eV = 7{,}733\ [K])$
u	Velocity [m/s]
u_{th}	Thermal velocity of distribution [m/s]
U	Total relativistic energy (Appendix)
U_i	Ionization energy [J] or [eV], $(1eV = 1.6021 \times 10^{-19}\ [J])$
υ	Velocity [m/s]
υ_{av}	Average electron speed [m/s]

$\langle \upsilon \rangle$ Average speed of species $[m/s]$ (CH-1)

$\vec{\upsilon}$ Velocity vector (directional) $[m/s]$

υ_B Bohm velocity $[m/s]$

υ_c Critical velocity $[m/s]$

$\vec{\upsilon}_C$ Centrifugal drift velocity $[m/s]$

$\vec{\upsilon}_D$ Total drift velocity $[m/s]$

$\overline{\upsilon}_e$ Mean electron speed $[m/s]$

$\vec{\upsilon}_E$ Drift velocity $[m/s]$

$\vec{\upsilon}_F$ Force drift velocity $[m/s]$

$\vec{\upsilon}_g$ Gravitational drift velocity $[m/s]$

$\vec{\upsilon}_G$ Gradient drift speed $[m/s]$

$\vec{\upsilon}_P$ Polarization drift velocity $[m/s]$

υ_s Particle velocity of a species or initial velocity of a particle $[m/s]$ (CH-1)

υ_T Characteristic velocity $[m/s]$

υ_{th} Thermal velocity $[m/s]$

υ_{th_s} Particle thermal velocity $[m/s]$

υ_{th_e} Electron thermal velocity $[m/s]$

υ_{th_i} Ion thermal velocity $[m/s]$

υ_{rel} Velocity relative to target particles $[m/s]$

$\Delta\upsilon$ Change in velocity $[m/s^2]$

Δt PIC time-step

Δx PIC cell size

$\mathbf{v}$ Velocity vector $[m/s]$

V Potential/Voltage [V]

V_{bias} Bias voltage [V]

V_f Floating potential [V] ($1V = 1\times10^{-18}$ [eV])

V_p Plasma potential [V]

V_s Sheath voltage [V]

W Particle kinetic energy of orbital motion (Appendix); $W \equiv W_\perp + W_\parallel$

x Coordinate variable

y Coordinate variable

z Coordinate variable

Greek Symbols

β Coefficient dependent of r_p/λ_D (CH-2)

β_H Fraction of hot electrons at the mid-plane (CH-2)

Γ Particle flux $[m^{-2}s^{-1}]$; Incomplete gamma function (CH-4)

Γ_i Ion flux $[m^{-2}s^{-1}]$

Γ_e Electron flux $[m^{-2}s^{-1}]$

Γ_r Random flux $[m^{-2}s^{-1}]$

γ Coefficient dependent of r_p/λ_D [m/s] (CH-2); Lorentz factor (Appendix)

δ Sheath thickness [m]

δx Infinitesimal distance [m] (CH-1)

ε Permittivity of free space, 8.854×10^{-12} $[C^2/N.m^2]$ or [F/m]

ε_o Permittivity of vacuum, 8.854×10^{-12} $[C^2/N.m^2]$ or [F/m]

η Non-dimensional probe bias; efficiency

λ Mean free path [m]

λ_e Mean free path of electron [m]

λ_D Debye length [m]

λ_{De} Debye length (for electron) [m]

λ_i Mean free path of ion [m]

λ_{iz} Mean ionization distance [m]

Λ Plasma parameter [−]

$\ln \Lambda$ Coulomb logarithm [−]

ν_{coll} Collision frequency [Hz]

ν_{ee} Electron-electron collision frequency [Hz]

ν_{en} Electron-neutral collision frequency [Hz]

ν_{ei} Electron-ion collision frequency [Hz]

ν_{ie} Ion-electron collision frequency [Hz]

ν_{ii} Ion-ion collision frequency [Hz]

π Mathematical constant, 3.14159 [−]

ρ Charge density $[C/m^3]$

σ Cross section of collision frequency $[m^{-2}]$ (CH-1); reaction cross section $[m^{-2}]$ (CH-3)

σ_C Coulombic cross section $[m^{-2}]$

σ_n Cross section of neutral particles $[m^{-2}]$

τ_p Time scale (plasma period) [s]

ϕ Electric potential [V]

ϕ_s Sheath potential [V]

ϕ_∞ Plasma potential [V]

Φ_B Magnetic flux $[Tm^2]$

ω_c Cyclotron (gyration) frequency [Hz]

ω_p Plasma frequency [Hz]

"Everything we see in the world is the
creative work of women."
- *Mustafa Kemal Atatürk*

Chapter 1

Introduction and General Information

The potential to generate power in significant quantities through thermonuclear fusion [1, 2] and the need to develop advanced electric propulsion (EP) thrusters for future space missions [3] are issues that are driving and dominating research in the field of plasma physics. The extreme conditions associated with these applications can cause mechanical degradation [4]. Due to the high costs associated with testing these systems over long periods of time, it is vitally important that on-board diagnostic systems, including plasma probes for measurement of the internal plasma environment, be integrated into systems to monitor equipment health and provide data to guide the design of future systems.

This dissertation explores the development of a capability for simulating the plasma dynamics of the Langmuir probe in complex plasmas where the velocity distributions are non-equilibrium and the electron energy spectrum is non-Maxwellian with respect to laboratory and space experiments. The results of this investigation are interpreted to give recommendations for the design and use of Langmuir probes. This work is conducted using computational techniques in order to create the exact plasma conditions of the experimental testing environments. This chapter provides an introduction to Langmuir probes, the plasmas in which they are used, and models used to simulate plasma environments. The motivation for the research in this dissertation is also provided.

In general, it is assumed that some quantity measured by the plasma diagnostic can be related back to a desired property in the plasma. For example, the slope of collected current with respect to bias voltage in the I-V characteristic of a Langmuir probe can be related to the electron temperature. A diverse range of diagnostic techniques has been developed to extract information about the flow. For example, the drift current can be measured with Faraday probes, the ion velocity can be visualized with laser induced fluorescence, the electron number density can be found with microwave interferometry, the plasma potential and electric fields can be measured with emissive probes (e.g., Langmuir

probes), the ion energy spectrum can be calculated with retarding potential analyzers, and ion charge states can be identified with $E \times B$ probes [2, 5]. Many of these instruments can also be used to measure more than one measurement technique. However, to properly interpret the sensor measurements, a good understanding of plasma physics is essential.

Traditionally, in the diagnostic theory it is assumed that the probe interacts with an ideal plasma with a Maxwellian velocity distribution, i.e., when the plasma is in equilibrium [6], and the instrument does not significantly affect the properties of the plasma at the point where measurements are taken. In the reality, this assumption is not valid for probe diagnostics, and it is well acknowledged that some disturbance is unavoidable [7]. The theory for the plasma sheath on an immersed surface helps to identify and quantify some of the disturbances caused by the probe. The diagnostic techniques for some instruments take the plasma sheath into account in order to relate the measured properties to undisturbed plasma properties [7]. However, sheath theory is limited to the effects in the electrostatic or electromagnetic sheath within a few Debye lengths of the probe surface. Diagnostic theory does not account for the effects at longer range, since those effects are not well understood.

The diagnostics are interpreted by relating a directly measured property, such as collected current, to the desired plasma property [8]. The relation is often developed from elementary principles by making certain assumptions about the plasma. For the Langmuir probe example, the electrons are assumed to have a Maxwellian velocity distribution so that the logarithmic slope of the I-V characteristic is inversely proportional to the electron temperature [7]. These assumptions commonly include a Maxwellian distribution of electron velocities, the Boltzmann relation between electron density and plasma potential, and a lower temperature for ions than for electrons [7]. The measurements are assumed to accurately obtain the plasma properties, provided the plasma conditions meet the assumptions of the diagnostic theory.

The measurements are very simple and direct when the electron energy distribution (EED) is Maxwellian. The electron energy is characterized by a single scalar value, T_e. If a significant amount of additional high-energy electrons are present in the vicinity, the determination of T_e becomes complicated however, and may not even be possible if the density of these additional components is very high. This may significantly alter the I-V curves and affect the determination of T_e by distorting the electron retarding regions of the I-V curves, which then affect the measurements of the total density (e.g., n_e and n_i).

The plasma is non-ideal and not in equilibrium in many cases, such as in the accelerator of an EP system or in a nozzle [7]. The deviation from the equilibrium limits the effectiveness and accuracy of applying a Maxwellian velocity

distribution in modeling the plasma environments, and it can lead to invalid engineering predictions and inaccurate analysis of empirical data [7]. To meet the diagnostic accuracy requirements for non-equilibrium plasma systems [8], it is essential that non-Maxwellian effects on plasma sensors and plasma models be accurately predicted; this is of particular importance for future EP systems.

1.1 Electric Propulsion

Electric propulsion (EP) refers to spacecraft propulsion systems that utilize electrical processes to accelerate a plasma propellant. EP thrusters use a variety of mechanisms to convert electrical power into thrust and provide superior specific impulse in comparison to chemical systems [9, 10]. Generally an EP device can be grouped into one of three main categories [11]: electrothermal, electrostatic, or electromagnetic. The electrothermal EP (e.g., resistojets and arcjets) uses electrical heaters or an electrical discharge to heat a working gas [11]. That gas is expanded through a nozzle as in a conventional rocket. The electrostatic EP first ionizes the propellant and then accelerates the charged particles via electric fields between one or more extraction and acceleration grids. Other electrostatic devices include Hall thrusters, field emission thrusters, and colloid thrusters [11]. Devices in this category accelerate positive charge particles or ions, so there is a need for an electron-emitting neutralizer to prevent a net charge buildup on the spacecraft [11]. The third category is the electromagnetic EP (e.g., Magnetoplasmadynamic (MPD) thrusters and pulsed plasma thrusters (PPT)) that use both electrical and magnetic fields to accelerate charged particles. In PPT concepts, a strong current is driven through surface material to create plasma that carries the current from anode to cathode. Other concepts may use alternative ionization schemes and they require applied electric fields to drive the plasma. The plasma is accelerated to generate thrust via the Lorentz force of the magnetic fields acting on the plasma current.

EP offers several benefits over conventional chemical rockets, but also suffers from a few drawbacks. The gains in using an EP system can outweigh the losses for long term or high-energy missions. One of the biggest advantages of EP devices is the high exhaust velocity and corresponding high specific impulse. Specific impulse I_{sp} is the ratio of thrust T to propellant weight flow rate as shown in equation (1.1). The weight flow rate is the product of the mass flow rate $\dot{m}$ and standard gravity at the surface of the earth g .

$$I_{sp} = \frac{T}{\dot{m}g} \tag{1.1}$$

EP devices have high I_{sp} ranging from 500-2,000 second for arcjets, to 500-2,000 second for Hall thrusters, to 3,000-10,000 second for ion thrusters. The chemical rockets, however, only range from 250-450 second I_{sp} [12]. Table 1.1 summarizes the typical performance parameters of various types of in-space propulsion with planned or demonstrated extremes included for comparison. The benefit of higher I_{sp} is demonstrated using the ideal rocket equation as in equation (1.2), which is derived from conservation of momentum for a system that is emitting mass [13]:

$$m = m_o \exp\left(-\frac{\Delta\upsilon}{gI_{sp}}\right), \tag{1.2}$$

where m_o is the initial total mass that undergoes a maneuver with a total change in velocity $\Delta\upsilon$. Where m is the final mass after the maneuver, I_{sp} is the specific impulse and g is the gravitational constant. A spacecraft mission requires $\Delta\upsilon$ from orbital mechanics and a limited m_o, so a higher specific impulse translates into more final payload mass after maneuvering. One option for a launch vehicle is to increase the payload mass; an EP system requires less propellant mass, so the mission or operational lifetime could be extended. Another option is to eliminate the propellant mass saved by using the EP system and reduce the total mass allowing a less expensive launch vehicle to be used [12].

Many EP systems are capable of semi-independent control of propellant flow and electrical systems, which allows for throttling of the flow rate, operating voltage, and current to optimize performance at a desired thrust level. This is an improvement over solid rockets, which have no throttle control, and over liquid or hybrid rockets, which can only throttle the flow rate; using EP systems provides less restrictive launch windows than a mission using conventional rockets [12].

The robustness of the propulsive concept is another advantage of EP devices. The acceleration mechanism is not dependent on the particular propellant, so an EP device could operate on a variety of different propellants. This opens the possibility of *in situ* propellant resupply for long duration missions with volatile compounds. The two main drawbacks to using an EP device stem from the practical limit on the maximum current density that are sustained in an electrical arc or plasma current [12]. In electrothermal applications, this limits the rate of energy deposition, while in electrostatic and electromagnetic applications this limits the flow rate of accelerated exhaust. In both cases, EP systems produce much smaller thrust than chemical systems (e.g., thrust levels from EP devices range from a few μN for ion thrusters to less than 10 N for arcjets, while the chemical rockets can achieve $1\text{kN} - 1\text{MN}$ of thrust) [12].

Table 1.1: Typical performance parameters of various in-space propulsion systems.

Thruster Type Units	Min I_{sp} (sec.)	Max I_{sp} (sec.)	Min Power (kW)	Max Power (kW)	Typical Efficiency (Percent)	References
Chemical Monopropellant	150	225	N/A	N/A	N/A	[12]
Chemical Bipropellant	300	450	N/A	N/A	N/A	[12]
Resistojet	150	700	0.5	1.5	80	[12, 14]
Arcjet[1]	450	1,500	0.3	30	25-40	[12, 14, 15]
PPT	1,000	1,500	0.001	200	8-13	[14]
MPD	2,000	5,000	1	4,000	<50	[14]
Gridded Ion[2]	2,800	2,000	0.2	10	55-60	[3, 14]
Hall[3]	1,000	10	0.1	20	45-65	[16-18]

The robustness of the propulsive concept is another advantage of EP devices. The acceleration mechanism is not dependent on the particular propellant, so an EP device could operate on a variety of different propellants. This opens the possibility of *in situ* propellant resupply for long duration missions with volatile compounds. The two main drawbacks to using an EP device stem from the practical limit on the maximum current density that are sustained in an electrical arc or plasma current [12]. In electrothermal applications, this limits the rate of energy deposition, while in electrostatic and electromagnetic applications this limits the flow rate of accelerated exhaust. In both cases, EP systems produce much smaller thrust than chemical systems (e.g., thrust levels from EP devices range from a few μN for ion thrusters to less than 10 N for arcjets, while the chemical rockets can achieve $1\text{kN}-1\text{MN}$ of thrust) [12].

EP systems require much longer firing time than chemical systems. An EP device must operate for months or years to achieve the full Δv (see equation (1.2)) but a chemical rocket operates for only a few minutes to achieve this result [13]. This creates long term performance issues, lifetime concerns and failure for EP systems [13]. Currently, the station keeping and orbit transfers are within the range of thrust provided by EP devices. The orbit transfers take much longer than with a chemical system. Microsatellite formation flight and high precision station keeping maneuvers both benefit from engines producing small but highly repeatable thrust bits (e.g., The planned NASA Laser Interferometer Space Antenna (LISA) mission [19]), requiring μN levels of thrust to maintain proper positioning of the component spacecraft.

[1] Demonstrated maximum specific impulse and power are documented as 1,970sec. and 100kW respectively.
[2] Planned maximum specific impulse and power are documented as 14,000sec. and 100kW respectively.
[3] Demonstrated maximum specific impulse and power are documented as 2,000sec. and 20kW respectively. These values exclude Hall thrusters operating on liquid metal propellants, such as bismuth, which can dramatically increase the specific impulse attained.

An EP device must operate for months or years, so the long duration life tests are carried out in vacuum chambers. It is necessary to understand how the actual performance deviates from the ideal performance by measuring changes in thrust, beam divergence, and efficiency over the lifetime of the device [20, 21]. The characterization of the wear and damage sustained during prolonged operation is also useful for identifying and addressing failure modes, as well as estimating the time to failure.

Research of the sputtering and deposition deals with the physical processes that erode material inside the thruster [22], and the transport and deposition of sputtered material and exhaust particles onto other spacecraft surfaces [23, 24]. Electromagnetic interference [24] is another concern because the charged particles in the exhaust plume interfere with signal transmission to and from the spacecraft. There is strong interest in developing high power EP systems; these systems are able to produce higher thrust levels and maintain the favorable I_{sp} and efficiency. In this area, the research is directed toward developing larger models of a device (monolithic approach) [25] or toward developing clusters of existing smaller models of a device [26-28]. The challenges for monolithic approach are related to fabrication and inadequate facilities for full scale testing. For the clustering approach, the challenges arise from operational difficulties because of the interaction between individual devices in a cluster.

Developing novel EP concepts is an active interest of researchers. New designs incorporate the hybrid systems, as in Hall/ion thrusters [29], or extend EP concepts to new design concepts as in micro-thruster and MEMS (thruster on a chip) concepts [30, 31]. Hybrid designs are able to move quickly to a prototype, since much of the fabrication and operation is obtained from existing designs. When the fabrication issues are resolved, these concepts may offer even more flexible scalability and control than existing EP systems.

The largest academic facility in the United States is the Large Vacuum Test Facility (LVTF)[4] at the plasma dynamics and EP Laboratory (PEPL) at the University of Michigan [32]. Smaller facilities are significantly less expensive to build, maintain, and operate; therefore, the number of large facilities is very limited. Although the LVTF has a very high pumping rate of 240,000 liters per second of xenon, the facility still develops a measurable backpressure during operation of EP thrusters [33]. This residual background pressure is due to the finite pumping capacity and inevitable leaks and outgassing present in any vacuum facility.The interpretation of experimental measurements is very complicated as the secondary effects of the background gas. Collisions between accelerated particles and background particles in the

[4] The Large Vacuum Test Facility (LVTF) is a cylindrical vessel 9m long, and 6m in diameters, which is the largest academic facility in the United States.

exhaust plume can increase beam divergence, broaden the energy distribution, and produce low energy charge exchange ions [33]. This affects measurements of current density and velocity distribution throughout the plume [33].

1.2 Plasma Diagnostics Background

In our universe, 99 percent of the matter is in the form of plasma, which is the fourth state of matter after solid, liquid and gas [2, 34, 35].There are different types of plasmas, which are generated in the laboratories for different applications. These include gaseous discharge, arcs, laser produced plasma, as well as tokamak plasma [36]. These plasmas have various applications in different fields of research and this is the reason why plasma physics has become established as a major research field during the past few decades [2]. In plasma accelerator and EP research, diagnostics measure plasma properties in order to infer performance and operating characteristics. In material processing applications, diagnostics ensure that the desired plasma conditions are sustained in order to produce the desired surface treatment. For each of these examples, plasma diagnostics provide feedback and measurements that help guide the research or operation to achieve a desired goal.

In general, plasma is classified by two factors: electron number density and temperature (hot or cold [37, 38])[5] [37-40]. Electron number density refers to the number of molecules that have ionized (lost one or more electrons) relative to the number of neutral gas species. Plasma temperature, on the other hand, does not refer to whether a plasma feels hot or cold, but rather to the energy of the free electrons. If there are significantly more neutral gas molecules than ionized gas molecules, the plasma may be cold to the touch, but still have very energetic electrons and, therefore, a very high plasma temperature. Material probes such as the most common Langmuir probe can be inserted in the cold plasma [2]. However, in the hot plasma ($T_e \geq$ few keV) [2] the use of intrusive probes is limited to the extreme edge of the plasma ($T_e \leq 50$eV) [2]. Non-contact or non-intrusive techniques [41-43] must be used to diagnose the other high-energy plasma regions (e.g., those encountered in magnetospheric and ionospheric plasmas [44-46] and in cold laboratory plasmas [47]).

Among all of the contact methods, electric probes, such as the Langmuir probe, are the most widely used plasma diagnostic tools [8]. Other methods include passive remote sensing methods, which detects radiation or particles emitted spontaneously by the plasma, such as studying plasma phenomena related to the sun [48]; active non-contact methods,

[5] What is hot or cold plasma? Plasma is called hot when it approaches a state of local thermodynamic equilibrium (LTE), also referred to as thermal plasma. Plasma is called cold when the thermal motion of the ions is ignored. There is no force, the magnetic force is ignored, and only the electric force is considered to act on the particles.

where radiations or particles produced by external sources, such as the study of the absorption and scattering of electromagnetic radiation by the plasma [48]. Langmuir probing determines the properties of a plasma with excellent spatial resolution, enabling the reconstruction of the crossing point of the particle [49]. Additionally, this method successfully describes the plasma's parameters in a variety of situations, such as low-pressure plasmas for material processing, edge plasmas in fusion devices, and ionospheric plasma [1, 46], and it aids the design of ion sources and new plasma chambers.

The simplest application of a Langmuir Probe involves inserting the probe into the plasma region of interest and applying a voltage V with respect to the metallic chamber that contains the plasma. Several configurations are commonly used, including single, double, and triple probes [50-52]. These variations allow for simplifications in the control system or in the analysis of measurements. The bias voltage is typically swept at a specified frequency over a range that is comparable to the electron temperature [53] to determine the current-voltage (I-V) characteristic for the local plasma. The plasma parameters are extracted during a post-processing step, where software can be used to fit the I-V characteristics [54]. This technique allows for faster electron temperature measurements than the typical swept probe while imposing less disturbance to the plasma and to other instruments [53]. This dissertation focuses on the single Langmuir probe theory and its implementation in the computational modeling.

Several theoretical studies have been performed to understand the behavior of Langmuir probes. Irving Langmuir and H. Mott-Smith [55] pioneered both the experimental and theoretical interpretation of probe data. More advanced treatments resulted from a better understanding of plasma sheaths; the work of Bohm [55] in the late 1940s clarified the sheath phenomena. To include the proper potential distribution in the sheath, Allen [56] derived expansions for current collection including the effect of electron potential barriers; ions were assumed to be immobile or cold. Bernstein and Robinowitz [57] expanded Allen's work by allowing for mono-energetic ions. Laframboise [58] completed the picture by including both thermal electrons and ions. All of the theoretical work assumes current collection in collisionless, quiescent plasma by an infinitely cylindrical probe and that the energy distribution is Maxwellian (i.e., ideal).

When the EED is Maxwellian, the electrons and ions have Maxwellian velocity distributions, and the kinetic temperature of electrons is greater than the temperatures of ions or neutrals. The charged particles that reach the surface of the probe do not chemically react with the probe material, and they are always collected and contribute to the probe current. The plasma sheath thickness is small compared with the probe dimension, and thus edge effects may be neglected. The potential around the probe preserves the symmetry (spherical, cylindrical or planar), and the plasma

potential is a monotonically decreasing (or increasing) function between the sheath edge and the probe surface, requiring the particles to maintain some spatial local ordering and assuming nearly Maxwellian velocity distributions. However, if additional electron energy populations are present in significant quantities, the determination of electron temperature is more complicated and may not be possible to determine if the density of the additional component is sufficiently high [59, 60].

1.2.1 Sheaths and Other Probe Disturbances

The insertion of the probe into the plasma results in a perturbation in the vicinity of the instrument. This effect is usually observed over a distance of several Debye lengths[6] [61] from the probe, influencing the charged particle trajectories in the sheath [34, 35, 61]. Thus, the introduction of probes can produce erroneous results due to unexpected differences between the physical geometry and the effective collecting area of the probe [62]. Farther from the probe, the collective shielding effect isolates the bulk plasma from the fields. The transition from bulk plasma through the sheath to a probe surface is shown schematically in Figure 1.1. Figure 1.1 shows the specific geometry to be considered, along with the behavior of the potential and electron and ion densities in the plasma sheath[7] [61]. The bulk plasma regions are also shown, as well as in the transition (presheath [61])[8] region between them. For a positive ion plasma, the electrons are the sole negative charge carriers and are more mobile than the ions. This leads to unequal fluxes to a surface at the same potential as the plasma, with more electrons than ions reaching the surface. If the surface is floating, it will accumulate a net negative charge that repels electrons and attracts ions until the net current is zero.

Sheaths comprise some of the most enduring and widely encountered problems in plasma physics. Langmuir and Tonks investigated sheaths in plasma arc and glow discharges in the late 1920's [63], identifying the major features of the collisionless electrostatic sheath and obtaining analytical solutions for simple geometries. Bohm investigated the current collection of Langmuir probes in the late 1940's, and established the broadly applicable Bohm criterion for the formation of a steady sheath [64]. The criterion has since been generalized for finite temperature ion distributions and arbitrary ion distributions. Bohm proposed that Debye shielding be divided into three phenomenologically different regions (Figure 1.1). Developments in the 1980's and 1990's produced a kinetic description of the sheath and presheath

[6] What is "Debye length"? It is the distance over which the electric field associated with a charge in a plasma is shielded out.
[7] What is "plasma sheath"? It is the region where the transformation from the plasma state to the solid state takes place.
[8] The combined transition and adjacent plasma regions are often referred to as the "presheath".

for application to the strongly flowing plasmas encountered in space environments and fusion plasmas [65-67]. Recently, much attention has been focused on developing a consistent method to span the interfaces between plasma, presheath, and sheath [68-71].

An immersed probe can also affect the flow over longer length scales, since the flow either collides with the probe body or is diverted around the physical obstruction. This can potentially introduce flow features including compression regions upstream of the probe and rarefaction regions in the wake of the probe. Since experimental diagnostic probes are not perfectly absorbing, ions that collide with a probe surface can be neutralized and reflected back into the flow. The neutral gas diffuses away from the probe, extending the region that is perturbed by the probe. Charge exchange or momentum exchange collisions with the neutral gas also have an effect on the overall flow.

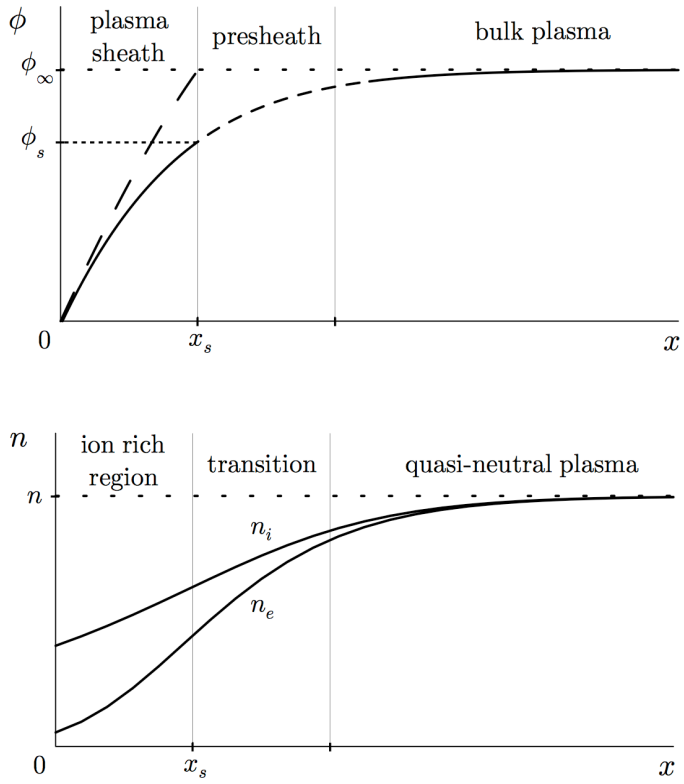

Figure 1.1: Schematic of the variation of electrostatic potential ϕ (sheath potential ϕ_s and plasma potential ϕ_∞) and electron and ion densities (n_e and n_i) in the sheath, transition (presheath) and bulk plasma regions of a plasma near a wall; ϕ is the negative of the potential [72]. Where The ∞ subscript indicates evaluation of the quantities in the bulk plasma region far from the wall (i.e., beyond the plasma sheath and presheath regions.)

1.3 Plasma Modeling and Simulation Background

Experimental investigations will perturb the plasma under study. For example, the physical modification of the chamber to include diagnostic equipment has effects on the plasma generated. Plasma chemistry can be contaminated by techniques such as emissive probes or chamber deposition. Electrostatic probes can produce incorrect results because of the unexpected differences between the physical geometry and the effective collecting area of the probe [62]. Plasma modeling, on the other hand, allows the exploration of the underlying principles and behavior of an idealized version of the plasma, unperturbed by experimental intrusion. Furthermore, it allows for the analysis of phenomena that do not readily lend themselves to experimental scrutiny [73]. A further advantage is that parameters can be changed far more quickly and cheaply than in physical experimentation.

Models of plasma physics generally fall into five categories: analytical models, equivalent circuit models, fluid models, kinetic models and hybrid models [74, 75]. Brief description of these models are given below [74-78]:

A. Analytical Models: Mathematical equations are used to describe the behavior and properties of a particular region of the plasma. Analytical models are constructed using an understanding of the physical mechanisms involved to formulate the appropriate equations. These are then used to further investigate the process or allow the extraction of meaningful information from experimental results [76, 77].

B. Equivalent Circuit Models: They use electrical components (e.g., resistors, capacitors, etc.) arranged into circuits to replicate the electrical properties of the plasma. Different degrees of complexity from whole systems to just sheath behavior can be simulated without a fundamental understanding of the mechanisms at work. This is also their key disadvantage as this simplistic approach means that they cannot provide any insight into those mechanisms [75, 77].

C. Fluid Models: The plasma is treated as a fluid and it is described by the transport equations, which govern continuity, momentum and energy conservation. These are derived from the full Boltzmann equation, and then the closure is made by assuming parameterization of the distribution function. Their biggest drawback is the need for the particle distribution functions to be either assumed or supplied [78].

D. Kinetic Models: The behavior of a plasma is simulated with individual particles rather than the assumed distributions of fluid models. The flows in many EP systems fall into the rarefied regime where collision and plasma length scale are similar to or larger than the size of the thruster. In these cases, the flow is not well represented by a continuum formulation, and instead, a molecular, kinetic approach must be undertaken.

The most well-known example of a kinetic model is the PIC code [79]. The motion of both particle ions and electrons are tracked, from which the particle motion, potential and electric fields are calculated self consistently [77]. It is considered ideal, yet it comes at a far greater computational expense than other methods.

E. Hybrid Models: Hybrid models represent a mix of two or more of the simulation methods [75, 77]. For example, the hybrid PIC model substitutes the particle electrons of the full PIC model with a fluid model. As only the ion motion needs to be tracked, far larger time steps can be taken offering a considerable computational saving.

Out of the all above methods, kinetic models are preferable, as the simulated plasma is based upon a finite number of individual particles rather than the assumed distributions of fluid models. This increased accuracy requires greater computational resources. One of the earliest and most widely used computational example of a kinetic model for simulating plasmas is the PIC method [79]. In this type of model the plasma is represented by a reduced number of macro-particles that obey the standard equations of motion and interact with discrete electric fields calculated on a computational grid. Macro-quantities such as number density, current density, etc., are calculated from the position, the velocity and the interactions of these particles. This method separates the particle motion from the calculation of fields and iterates to achieve a consistent solution. A full description of the PIC method is described in [79].

The PIC method is one of the earliest and most widely used computational techniques for simulating plasmas [79]. The main idea of the PIC code is to model the plasma using a set of super particles (i.e., a computational particle that represents many real particles), which are subject to either Newtonian or relativistic equations of motion and interact with discrete electric fields calculated on a computational grid. The fields in the code are calculated by solving Maxwell's equations using source terms derived from the particle positions and velocities. In the earlier research, the classical PIC procedure was applied to collisionless plasmas. When collisional effects are to be included, they are added via an explicit collision operator, which is commonly some type of Monte Carlo (MC) procedure. This technique also uses macro-particles to simulate gases and incorporates probabilistic models to describe particle collisions [80]. This allows additional physical processes to be considered, including chemistry, ionization and recombination, and surface interactions. MC models can be combined with PIC models to add collisions and wall interactions to a plasma simulation. More in depth data can be obtained from the references [79].

In 1996, Keefer and Markusic presented a 2D/3V electrostatic probe PIC code called PROBEPIC [41]. This code was developed to simulate the behavior of a Langmuir probe in both quiescent and flowing plasmas in regimes for which no closed-form analytical solution exists. It was also able to expand on previous studies by simulating the

behavior of finite length probes [34, 41]. PROBEPIC is a 2D/3V PIC model, meaning that the electric field has two components and that the particles are free to move in three dimensions [34]. The computational domain [34] in PROBEPIC consists of a cylindrical region containing a cylindrical probe [34]. This region provides sufficient space for a plasma sheath, generated by the interaction of the plasma with the probe boundary material, to form around the probe. The program outputs I-V traces, which then can be compared to theoretical/experimental results.

PROBEPIC was mainly used to simulate the plasmas that have the Maxwellian velocity distribution. The expression for this distribution is the product of three distribution functions for the velocity in each of the three principle spatial directions [72]. The distribution is uniform and isotropic in configuration space and is isotropic in velocity space, which means that the position and the velocity of a single particle are specified by providing its three coordinates $(x, y \text{ and } z)$ and that the distributions of particles are isotropic. However, using the same model equations to simulate non-equilibrium plasmas would create large errors in the plasma diagnostics; this is complicated by the fact that all three probe geometries (i.e., cylindrical, planar and spherical) are equally suited for measuring the ionospheric electron temperature and density or for determining the distribution function in the presence of non-Maxwellian electrons [81].

This dissertation adapts PIC codes to allow for non-equilibrated plasma regions with non-Maxwellian velocity distributions. This method can be applied both in collisionless plasmas (regardless of sheath thickness) and in a number of situations in collisional plasmas [81], such as when evaluating the EEDF from the second derivative of I-V curve for low-pressure plasmas. With this greatly improved PIC code model, it is possible to accurately interpret probe data from measure of ionospheric electron temperature and ion and electron number densities on rockets and satellites in a laboratory environment.

1.3 An Overview of Plasma Physics

It has been said that 99 percent of the matter in the universe is in the plasma state in the form of an electrified gas with the atoms dissociated into positive ions and negative electrons [2, 82]. This estimate may not be very accurate, but it is certainly a reasonable one in view of the fact that stellar interiors and atmospheres, gaseous nebulae, and much of the interstellar hydrogen are plasmas. As soon as one leaves the earth's atmosphere, one encounters the plasma comprising the Van Allen radiation belts [83] and the solar wind. However, in our everyday lives plasmas are limited to a few examples: the flash of a lightning bolt, the soft glow of the Aurora Borealis [84], the conducting gas inside a fluorescent

tube or neon sign, and the slight amount of ionization in a rocket exhaust. It seems that we live in the 1 percent of the universe where plasmas do not occur naturally [82].

The reason for this can be seen from the Saha equation [85], which tells us the amount of ionization to be expected in a gas in thermal equilibrium:

$$\frac{n_i}{n_n} \approx 2.4 \times 10^{15} \frac{T^{3/2}}{n_i} e^{-U_i/k_B T} \tag{1.3}$$

where n_i is the density (number per cm^3) of ionized atoms and n_n is the density (number per cm^3) of neutral atoms, T is the gas temperature in ^oK, k_B is Boltzmann's constant, U_i is the ionization energy of the gas–that is, energy required to remove the outermost electron from an atom [85]. As the temperature is raised, the degree of ionization stays low until U_i is a few times $k_B T$. Then the fractional ionization n_i/n_n starts rising, and the gas is in a plasma state. For example, an ordinary air at a room temperature, $n_n \approx 3 \times 10^{25} m^{-3}$, $T \approx 300^oK$ and $U_i = 14.5eV$ (nitrogen), where $1eV = 1.6x10^{-19} J$. Fractional ionization $n_i/(n_n + n_i) \approx n_i/n_n$ predicted by equation (1.3) is very low $(n_i/n_n \approx 10^{-122})$. As the temperature increases n_n becomes less than n_i, and the plasma becomes fully ionized when n_n is zero. This is the reason plasmas exist in astronomical bodies with temperatures of millions of degrees, but not much in our environment (1 percent) [85].

The plasma state is a gaseous mixture of positive ions and negative electrons. Any ionized gas is not a plasma; a plasma is a quasi-neutral gas of charged and neutral particles which exhibits collective behavior [85]. The meaning of quasi-neutrality and collective behavior will be made clear later in this section. This section also introduces some fundamental physics of plasmas. A brief outline of the general properties of plasmas is presented. The knowledge of the general characteristics of the plasma is essential in the context of identifying various categories of plasmas and the changes occurring in them due to the changes in the input parameters related to plasma production and evolution.

1.3.1 Concept of Temperature

At thermal equilibrium, a gas has particles of all velocities, and the most probable distribution of those velocities is the Maxwellian distribution [86]. The one-dimensional Maxwellian distribution is given by [82]

$$f(u) = A \exp\left(-\frac{1}{2} m u^2 \Big/ k_B T\right), \tag{1.4}$$

where f is the number of particles per cm^3 with velocity between u and $u+du$, $mu^2/2$ is the kinetic energy, k_BT is the product of the Boltzmann constant k_B [$1.38\times10^{-23} m^2$ kg s^{-2} K^{-1}] and the temperature T (in this case there is only one temperature for all particles) [87]. A gas is partially characterized by the number of particles per unit volume, known as number density n per cm^3. The coefficient A is defined as a function of density n by [82]

$$A = n\left(\frac{m}{2\pi k_B T}\right)^{1/2}, \tag{1.5}$$

Constant A is derived for a normalized one-dimensional Maxwellian distribution such that $\int_{-\infty}^{\infty}\hat{f}(u)du = 1$ and then integrate as $1 = \int_{-\infty}^{\infty} A e^{-mu^2/k_BT}du = A\sqrt{2\pi k_B T/m}$. The width of the distribution is characterized by the constant temperature T. Knowing the velocity distribution of the particles within the given volume allows the density n, or number of particles per cm^3, to be derived as [82]

$$n = \int_{-\infty}^{\infty} f(u)du, \tag{1.6}$$

and the average kinetic energy within the distribution can be defined by integrating over the velocities [82]:

$$E_{av} = \frac{\int_{-\infty}^{\infty}\frac{1}{2}mu^2 f(u)du}{\int_{-\infty}^{\infty} f(u)du}. \tag{1.7}$$

In order to make evaluating this integral easier, the thermal velocity of the distribution can be defined as [82]

$$\upsilon_{th} = \left(2k_BT/m\right)^{1/2}, \tag{1.8}$$

and then the normalized velocity is $y = u/\upsilon_{th}$. We write equation (1.4) as $f(u) = A\exp\left(-u^2/\upsilon_{th}^2\right)$ and equation (1.7), the average kinetic energy, can now be written in these new terms [82]

$$E_{av} = \frac{\frac{1}{2}mA\upsilon_{th}^3\int_{-\infty}^{\infty}\left[\exp\left(-y^2\right)\right]y^2dy}{A\upsilon_{th}\int_{-\infty}^{\infty}\exp\left(-y^2\right)dy} \tag{1.9}$$

In equation (1.9), the integral in the numerator is integratable by parts [82]

$$\int_{-\infty}^{\infty} y\left[\exp\left(-y^2\right)\right]y\,dy = \left[-\frac{1}{2}\left[\exp\left(-y^2\right)\right]y\right]_{-\infty}^{\infty} - \int_{-\infty}^{\infty} -\frac{1}{2}\exp\left(-y^2\right) \tag{1.10}$$

$$= \frac{1}{2}\int_{-\infty}^{\infty} \exp\left(-y^2\right)dy$$

After the integrals, the average kinetic energy can be written as [82]

$$E_{av} = \frac{\frac{1}{2}mA\upsilon_{th}^3}{A\upsilon_{th}} = \frac{1}{4}m\upsilon_{th}^2 = \frac{1}{2}k_B T . \tag{1.11}$$

A similar process can be explored in order to derive the average kinetic energy in three dimensions. The three-dimensional Maxwellian distribution is defined as [82]

$$f(u,\upsilon,w) = A_3 \exp\left[-\frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)\Big/k_B T\right] \tag{1.12}$$

where [82]

$$A_3 = n\left(\frac{m}{2\pi k_B T}\right)^{3/2} \tag{1.13}$$

Then the average electron kinetic energy is [82]

$$E_{av} = \frac{\int\int\int_{-\infty}^{\infty} A_3 \frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)\exp\left[-\frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)\Big/k_B T\right]du\,d\upsilon\,dw}{\int\int\int_{-\infty}^{\infty} A_3 \exp\left[-\frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)\Big/k_B T\right]du\,d\upsilon\,dw} \tag{1.14}$$

Maxwellian distribution is isotropic, so each of the three terms in the numerator is the same as the others. Therefore, evaluating one of them would be enough, and this can be multiplied by three [82]:

$$E_{av} = \frac{3A_3\int\frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)du\int\int\exp\left[-\frac{1}{2}m\left(\upsilon^2+w^2\right)\Big/k_B T\right]d\upsilon\,dw}{A_3\int\frac{1}{2}m\left(u^2+\upsilon^2+w^2\right)du\int\int\exp\left[-\frac{1}{2}m\left(\upsilon^2+w^2\right)\Big/k_B T\right]d\upsilon\,dw} \tag{1.15}$$

Finally, the general result is

$$E_{av} = \frac{3}{2}k_B T . \tag{1.16}$$

for three degrees of freedom; it is the one-dimensional average kinetic energy times the number of degrees of freedom.

In an ideal gas, number density and temperature determine the pressure: [87]. The average kinetic energy E_{av} of a Maxwellian distribution is $k_B T/2$ [88]. To avoid confusion regarding the number of dimensions, the energy corresponding to $k_B T$ (rather than E_{av}) is used to denote the temperature. For example, if $p = nk_B T$, then

$$T = \frac{1.602x10^{-19}}{1.38x10^{-23}} \approx 11,600^{o}K \ . \tag{1.17}$$

The conversion factor is $1eV = 11,600^{o}K$. In this case, $2eV$ plasma means that $k_B T = 2eV$ or $E_{av} = 3eV$ in three dimensions.

The ions and the electrons have separate Maxwellian distributions with different temperatures. Each species has its own thermal equilibrium. In a magnetic field, the temperature can be different in different directions. Moreover, plasmas are often in a non-equilibrium state with different temperatures (T_e and T_i) [85]. Such two-temperature plasmas are found in gas discharges. The solar plasma, however, is an isothermal plasma with $T_e = T_i$ [85]. When there is a magnetic field B, even a single species (e.g., ions) can have two temperatures. This is because of the forces acting on an ion along B are different from those acting perpendicular to B due to the Lorentz force [85]. The components of velocity perpendicular to B and parallel to B may then belong to different Maxwellian distributions with temperatures (i.e., $T_{\perp}$ and $T_{\parallel}$).

1.3.2 Ionization Processes

Plasmas exist in an environment that provides for a large number of ionization processes of atoms. These can be photoionization by an intense source of ultraviolet radiation or collisional ionization by energetic electrons [85]. Photoionization is important in space plasmas where the electron and atom densities are low but a large number of ultraviolet (UV) photons may be present. Some of the most important causes of ion generation in a plasma are electron impact ionization of neutral atoms, electron impact ionization of an excited atom, and penning ionization by collision with an electronically excited atom according to [89] Table 1.2. In Table 1.2, where $h\upsilon$ is the energy associated with a photon, Atom* refers to the excited state, and the penning ionization is a type of ionization process involving reactions between neutral atoms or molecules [89].

Photoionization is especially important in space plasmas where the electron and atom densities are low but a large number of UV photons may be present. For high density plasmas, i.e., $n_e > 10^{17} m^{-3}$, the electron impact ionization of neutral atoms is the dominant process [89]. The mean ionization distance from electron impact ionization is determined according to [89, 90]

$$\lambda_{iz} = \upsilon_s / K_i(T_e) n_e \ , \tag{1.18}$$

where υ_s is the velocity of the sputtered atom, n_e is the density of electrons, and $K_i(T_e)$ is the ionization rate constant according to

$$K_i(T_e) = K_o \exp(-E_o / k_B T_e), \tag{1.19}$$

where the constants K_o and E_o are material-dependent; values are available in [91-96]. The average distance an atom travels before going through an ionizing collision is calculated by equations (1.18) and (1.19). For example, enough electrical power should be applied to reach electron densities on the order of $10^{19} m^{-3}$, and the electron temperature should be at least a few eV in order to achieve an effective ionization of the deposited material in the sputtering processes [97]. In reality this corresponds to a power on the order of $kWcm^{-2}$ on the target surface. Usually, this much power cannot be applied continuously, as there is not enough cathode cooling (thermionic electron emission to the plasma) [98]. Instead, the power is applied in pulses with a low duty factor [97] (i.e., the ratio between pulse duration and the total cycle time).

1.3.3 Collective Behavior of a Plasma

A constraint of the kinetic theory of gases is that no net forces (e.g., electromagnetic forces) act upon the gas particles. The gas particles carry no net charge and they are neutral. The particles move in straight lines until they collide with each other at velocities with a distribution (gravitational forces are not a concern at this moment) [99]. In plasma, charged particles create long-range electromagnetic forces. This means that each plasma particle interacts with a large number of other particles. Therefore, plasmas show a simultaneous response of many particles to an external stimulus [85]. In this sense, plasmas show collective behavior, which means that the macroscopic result to an external stimulus is the cooperative response of many plasma particles. Wave processes or mutual shielding of plasma particles are good examples of collective behavior of plasmas.

Table 1.2: Different elementary collision processes in the plasma volume [89].

Combinations	Processes
$A + e^- \rightarrow Ion + 2e^-$	(Collisional)
$A^* + e^- \rightarrow Ion + 2e^-$	(Collisional)
$e^- + A^* \rightarrow A + hv + e^-$	(Photoionization)
$A + Atom^* \rightarrow Ion + e^- + A$	(Penning)

At a given time, because of the long-range electrostatic attraction, there are enough electrons near the ions to establish neutrality. When there is a local charge variation, there will be impacts on the whole plasma, i.e., an affected charge distribution will evolve into a new charge distribution [99]. As the electron density is dynamically distorted by the presence of the ions, the charge balance is called quasi neutral [79], which is explained in the sub-section below.

1.3.3.1 Quasi-Neutrality and Thermal Speed

A plasma is a quasi-neutral $(n_e \sim n_i)$ gas consisting of electrons, ions and neutral particles exhibiting collective behavior [79]. Quasi-neutrality is defined as approximately equal numbers of negative and positive charges entering and exiting a system. At thermal equilibrium $(T_e \sim T_i \cong T_s \equiv 1/3 m_s \langle \upsilon_s^2 \rangle)$, the ratio of the thermal electron speed $\upsilon_{th_e} = (2k_B T_e/\pi m_e)^{1/2}$ to the thermal ion speed $\upsilon_{th_i} = (2k_B T_e/\pi m_i)^{1/2}$ is written as the square root of the mass ratio between electrons and ions $(m_p/m_e = 1836)$[9] [79]

$$\frac{\upsilon_{th_e}}{\upsilon_{th_i}} = \left(\frac{m_i}{m_e}\right)^{1/2} \approx 42.9 , \tag{1.20}$$

where it is assumed that the ions are protons, $m_e = 9.11 \times 10^{-31}$ kg, and $m_p = 1.67 \times 10^{-27}$ kg [79]. The higher electron mobility maintains the amount of electrons along their trajectories towards the ions, making it possible for them to escape the trapped orbits domain [79]. Due to the long-range electrostatic attraction, there are enough electrons near the ions. As the electron density is dynamically distorted by the presence of the ions, the charge balance is quasi neutral [79].

1.3.3.2 Plasma (Langmuir) Frequency

Debye shielding is one example for collective behavior of plasma. The second aspect of collective behavior is the time scale, after which the electrons establish a shielded equilibrium positions [85].

When the potential perturbation is small (i.e., $|eV| \ll k_B T_e$), the electron energy is not much changed from its electron thermal velocity $\upsilon_{th_e} \approx \left(2k_B T_e/m_e\right)^{1/2}$. To have a new equilibrium, the electron must reach its new position at a

[9] The rest mass of the proton divided by that of the electron.

typical distance λ_{De} [10][7]. This time is established as $\tau \approx \lambda_{De}/\upsilon_{th_e}$. This reciprocal of the response time [85] is called plasma frequency ω_p,

$$\omega_p = \sqrt{\frac{ne^2}{\varepsilon_o m}}, \tag{1.21}$$

which is the most fundamental time-scale in plasma physics [35]. There is a different plasma frequency for each species; the relatively high electron plasma frequency is determined by [35] $\omega_{pe} = (e^2 n_e/\varepsilon_o m_e)^{1/2}$, where ω_{pe} is the electron plasma frequency, ε_o is the permitivity of a vacuum and n_e is the electron number density, m_e is the electron mass and e is the fundamental electron charge.

Plasma frequency corresponds to the typical electrostatic oscillation frequency of a given species in response to a small charge seperation. For example, a one dimensional situation where a slab consisting entirely of one charge species is displaced from its quasinetral position by an infinitesimal distance δx. The resulting charge density which develops on the leading face of the slab is $\sigma = en\delta x$. The x-directed electric field generated inside the slab is of magnitude $E_x = -\sigma/\varepsilon_o$. Newton's law applied to an individual particle inside the slab yields $m(d^2\delta x/dt^2) = eE_x = -m\omega_p^2\delta x$, giving $\delta x = (\delta x)_o \cos(\omega_p t)$. Above the plasma frequency ω_p, electromagnetic waves propagate through the plasma; below the plasma frequency, however, the electrons screen out the disturbance over a characteristic time (plasma period), $\tau_p = 1/\omega_p$. Note that plasma oscillations will only be observed if the plasma system is studied over time periods τ longer that the plasma period τ_p and if the external actions change the system at a rate no faster than ω_p. In the opposite case, system will no longer be refered to as a plasma. Likewise, if the characteristic length of a plasma L shorter than the distance $\upsilon_{th}\tau_p$ traveled by a typical plasma particle during a plasma period will not detect plasma behavior. Particle will exit the system before completing a plasma oscilation. This distance, which is the spatial equivalent to τ_p, is refered to as Debye length λ_D. If the characteristic length of a plasma L sufficiently large (i.e., $L \gg \lambda_D$), then it behaves in a collective manner.

---

[10] $\lambda_{De} = (\varepsilon_o T_e/en_e)^{1/2}$ is the electron Debye length. For example: $\lambda_{De} = 740\sqrt{T_e/n_e}$ cm, where n_e is in units of cm^{-3} and T_e is in volts.

1.3.3.3 Debye Length

While reducing the size of the observed volume element down to the scale lengths where charge neutrality no longer holds and the local charge density $\rho = e(n_i - n_e)$ is no longer zero, the presence of an electric potential that satisfies the Poisson's equation for electron-proton plasma becomes resolved. Here, e is the elementary charge, n_i is the ion number density and n_e is the electron number density. The density of ions and electrons can still be modeled via the Boltzman's law [34],

$$n_e(r) = n_o \exp\left(\frac{e\phi(r)}{k_B T_e}\right). \tag{1.22}$$

By expanding (1.22) in a Taylor series and combining with the Poisson's equation, one obtains [34]

$$\nabla^2\phi = -\frac{\rho}{\varepsilon_o}, \tag{1.23}$$

resulting in a differential spherical symmetric problem for the potential field ϕ centered on an ion and with radius r [34],

$$\nabla^2\phi = \frac{d^2\phi}{dx^2} = \frac{n_o e^2}{\varepsilon_o k_B T_e}\phi = \left(\frac{1}{\sqrt{\varepsilon_o k_B T_e / n_o e^2}}\right)^2 \phi. \tag{1.24}$$

The solution of equation (1.24) is

$$\phi = \frac{e}{4\pi\varepsilon_o r}\exp\left(-r\Big/\sqrt{\frac{\varepsilon_o k_B T_e}{n_o e^2}}\right). \tag{1.25}$$

By defining [34],

$$\lambda_D = \sqrt{\frac{\varepsilon_o k_B T_e}{n_o e^2}} \tag{1.26}$$

as the characteristic Debye length λ_D, equation (1.25) can now be written as

$$V = \frac{e}{4\pi\varepsilon_o r}\exp\left(-\frac{r}{\lambda_D}\right). \tag{1.27}$$

This characteristic length scale λ_D determines the distance over which the charge is effectively screened by surrounding charges. In a qualitative sense, charges may only detect or sense other charges within their Debye sphere; outside the Debye sphere, the potential generated by a charge falls off and effectively cannot be sensed. This phenomenon is collectively referred to as Debye screening or shielding, which is the ability of the plasma to shield out

externally applied electric fields [88]. The electric potentials can be produced by either local charges or by inserting electrodes inside the plasma. If an electric field is applied to a plasma by inserting two electrodes, these electrodes attract charge carriers of the opposite charge, and a non-neutral cloud of electrons and ions surrounds each electrode [88]. For a plasma with no thermal motion of the charge carriers, the electric potential would be shielded, and no electric field would occur inside the plasma. Equation (1.26) can also be written as

$$\lambda_D = \sqrt{\frac{\varepsilon_o k_B T_e}{n_o e^2}} = \left(\frac{n_o e^2 V}{\varepsilon_o k_B T_e}\right)^{-1/2} \equiv \frac{\upsilon_{th_e}}{\omega_p} , \tag{1.28}$$

the thickness of the region where the electric potential penetrates the plasma. As the potential increases, the potential difference reaches a few times the electron temperature. The Debye length λ_D is numerically related to the electron thermal velocity v_{th_e} and to the plasma (Langmuir) frequency ω_p. T_e is the electron temperature, n_o is the undisturbed plasma density [88]. When the density is increased, λ_D decreases as each layer of plasma contains more electrons. λ_D increases with increasing $k_B T_e$. T_e is important in the definition of λ_D because of the electrons, which are more mobile than the ions, generally do the shielding.

1.3.3.4 The Plasma Parameter

The plasma parameter Λ is referred to as the number of particles in a Debye sphere, a sphere whose radius is the Debye length [34]:

$$\Lambda = n\frac{4\pi}{3}\lambda_D^3 = 1380\, T^{3/2}/n^{1/2} . \tag{1.29}$$

Substituting for λ_D yields

$$\Lambda = \frac{1}{\sqrt{n}}\frac{4\pi}{3}\left(\sqrt{\frac{\varepsilon_o T_e}{e^2}}\right)^3 . \tag{1.30}$$

The average distance between particles is defined as

$$r_d \equiv n^{-1/3} , \tag{1.31}$$

and the distance of the closest approach r_c is the distance at which the kinetic energy of the particle is fully converted into potential energy, i.e., no interaction occurs [34],

$$r_c \equiv \frac{e^2}{4\pi\varepsilon_o T_e} , \tag{1.32}$$

which allows the plasma parameter to be rewritten as

$$\Lambda = \frac{1}{6\sqrt{\pi}}\left(\frac{r_d}{r_c}\right)^{3/2}. \tag{1.33}$$

When plasma parameter $\Lambda \ll 1$, the particles in the plasma are dominated by one another's electrical fields, which is refered to as a strongly coupled plasma. If the Debye sphere is densely populated $(\Lambda \gg 1)$, the particle experiences the electrical influences of particles within its Debye sphere; this is referred to as weakly coupled plasma. As the temperatures are different, coupling states of electrons and ions are also different.

1.3.3.5 Cyclotron Frequency

Charged particles moving in a magnetic field tend to gyrate in a helical path, which is produced by the Lorentz force. For a particle of charge q and velocity $\vec{\upsilon}$ interacting with a magnetic field that is directed along the z-axis, the force is given by [34]

$$\vec{F} = q\vec{\upsilon} \times \vec{B}. \tag{1.34}$$

This equation may be solved for each of three spatial components of the motion. The result is a harmonic oscillator characterized by a cyclotron frequency ω_c [34],

$$\omega_c = \frac{qB}{m}. \tag{1.35}$$

1.3.3.6 The Larmor Radius

The size of the orbit where the particles move in a magnetic field is given by the Larmor radius r_L [34]. This length may be driven by considering the forces acting on a charged particle in a magnetic field. Figure 1.2 shows the free body diagram of such a particle. As the result of applying Newton's Second Law to the particle in the centripetal direction, the following equation [34] is obtained:

$$F_c = q\upsilon_\perp^2 B = \frac{m\upsilon_\perp^2}{r}, \tag{1.36}$$

where r is the Larmor radius r_L and $\upsilon_\perp$ is the velocity component perpendicular to the magnetic field.

$$r = r_L = \frac{m\upsilon_\perp}{qB} = \frac{\upsilon_\perp}{\omega_c}. \tag{1.37}$$

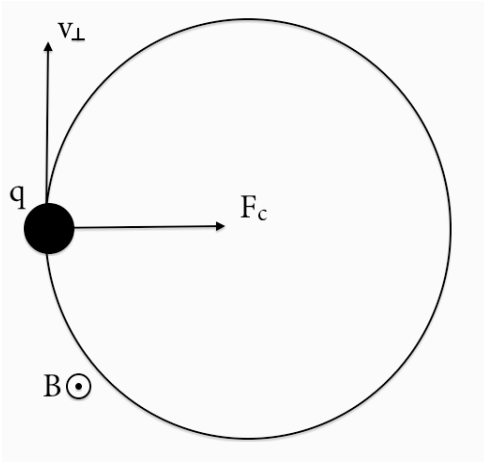

Figure 1.2: Positively charged particle moving in a uniform vertical magnetic field [34].

If an electric field is present in addition to the magnetic field, the motion of the particle will be perpendicular to the directions of both $\vec{E}$ and $\vec{B}$ and will have a velocity given by $\vec{\upsilon} = \vec{E} \times \vec{B} / B^2$ [7]. Equation (1.37) shows that the cyclotron frequency and Larmor radius are hence related.

1.3.3.7 Collision Frequency

The frequency of collisions ν_{coll} is described as the average rate at which particles of two species collide [100]. In weakly ionized plasmas, generally the electrons will collide most frequently with neutral atoms and molecules; the electron-neutral collision frequency ν_{en} will dominate over the electron-ion collision frequency ν_{ei} [101-103]. Since the plasma parameter $\ln\Lambda$ depends as the inverse of the relaxation times between these two different classes of collision, generally a weakly ionized plasma requires only a moderate value of Λ (usually about 10). Larger Λ will tend to indicate that the ion-electron collision frequency ν_{ie} is not insignificant relative to the neutral collision frequency [103].

For elastic collisions, the cross section of the particles is given as $\sigma_n = \pi d_o^2$; where d_o is the effective radius of the neutral particle. Typically this cross section is of the order $\sim 10^{-19} m^2$ and the collision frequency is $\nu_{en} = \sigma_n n_n \langle \upsilon_e \rangle$ [104]. Where n_n is the number density of the neutral collision partners and $\langle \upsilon_e \rangle$ is the average speed of the colliding electrons [104]. If the plasma is highly ionized and the Coulombic collisions dominate, then most of the collisions are small angle scattering collisions between all particles in a Debye sphere. The electron-ion collision frequency $\nu_{ei} = n_e \sigma_C \langle \upsilon_e \rangle$ governs

most of the character of the collisionality in the plasma; where σ_C is the Coulombic cross section [104]. The impact parameter is calculated with $d_C \sim e^2/4\pi\varepsilon_o m_e \langle \upsilon_e \rangle^2$ and the Coulombic cross section is given by $\sigma_C = \pi d_C^2 \sim e^4/16\pi\varepsilon_o^2 m_e^2 \langle \upsilon_e \rangle^4$ [104]. The Coulombic collision frequency is proportional to [104]

$$\nu_{ei} = n_e \sigma_c \langle \upsilon_e \rangle \sim \frac{n_e e^4}{16\pi\varepsilon_o^2 m_e^2 \langle \upsilon_e \rangle^3} \tag{1.38}$$

If there is thermal equilibrium, then the following equations become applicable [104]:

$$\nu_{ee} \sim \nu_{ei} \tag{1.39}$$

$$\nu_{ii} \sim \sqrt{m_e/m_i \nu_{ee}} \tag{1.40}$$

$$\nu_{ie} \sim (m_e/m_i)\nu_{ee} \tag{1.41}$$

1.3.3.8 Mean Free Path

The mean free path is the average distance that a particle (e.g., an atom, a molecule, a photon) travels before a collision is expected, which modifies its direction or energy or other particle properties [104, 105]. It is defined as [104]

$$\lambda = \frac{\langle \upsilon \rangle}{\nu_{coll}} = \frac{1}{\sigma n} \tag{1.42}$$

where $\langle \upsilon \rangle$ is the average speed of the species, ν_{coll} is the collision frequency of interest, σ is the cross section of the collision frequency, n is the number density of the colliding species.

For fully ionized plasma, the dominant collision is the Coulombic long range scattering, which is shown as ν_{ei}. Then the mean free path is shown as [104]

$$\lambda \sim \frac{\langle \upsilon_e \rangle}{\nu_{ei}} \sim 64\pi\lambda_D \frac{\Lambda}{\ln(\Lambda)} \tag{1.43}$$

$\ln\Lambda \geq \sim 10$ and the mean free path λ is very high compared to Debye length λ_D [104].

1.3.4 Existence Regimes

Plasmas are found in a huge parameter space, which covers seven orders of magnitude in temperature and twenty-five orders of magnitude in electron density [85]. Not all of the behavior of the plasma will be preserved with all scale of

changes. Plasma could possess energies we experience on an everyday basis to energies that demand a relativistic treatment [106]. Therefore, it is unlikely to expect a single physical model to adequately address all possible phenomena with equal validity across all ranges of these parameters. A typical definition for a plasma often offers the following conditions [101, 103, 107, 108]:

- The number of charge carriers is sufficiently large to allow particles to interact with several other particles within their Debye sphere, which provides the collective behavior in plasmas. Equation (1.29) represents the number of particles within a Debye sphere, which may be written as $\Lambda \gg 1$. Most common plasmas can be approximated well by assuming $\ln(\Lambda) \simeq 10$.
- The apparatus[11] [107, 109] length is assumed to be much larger than Debye length, $\lambda_D \ll L$. This means that, on the macroscopic scale there is no significant charge separation, which means that the plasma is quasi-neutral [79]. Plasmas that do not satisfy this condition are somewhat referred to as non-neutral plasmas.

Plasma sheaths are intrinsically non-neutral, and occur along the boundary surface in contact with the plasmas. Some plasma devices involve the application of an electric body force to accelerate the plasma flow. Then, the electric body force will be proportional to the net charge and therefore plasma possesses a net electric charge.

Another characteristic quantity in determining the nature of the plasma is the Knudsen number K_n and is defined as [101]

$$K_n = \frac{\lambda}{L}, \tag{1.44}$$

where λ is mean free path and L is the characteristic length of the domain. In Langmuir probe diagnostics, this characteristic length is typically chosen to be the diameter of the probe. The Knudsen number is used to distinguish between the low density[12] [110] and high density plasmas [101, 103]. When the Knudsen number is large, the mean free path is large compared to the overall length scale of interest. In this case, this means that the collisionality is not

[11] What is an apparatus? Apparatus is the technical equipment needed for a particular laboratory activity. A number of important lengths are well defined for ionized gases including the dimensions of the apparatus (e.g., Langmuir probe apparatus for plasma diagnostics).

[12] By low density we mean plasmas with neutral gas number densities of approximately 10^{13} to 10^{16} molecules cm^{-3} (pressure between $\sim$ 0.1 to 10^3 Pa) which are weakly ionized between 10^{-6} to 10^{-1}.

significant in the plasma, and sometimes a collisionless model is even appropriate (free molecular flow[13] [111]). When the Knudsen number is small, the collisionality is strongly present in the plasma. When the Knudsen number is of order unity, then the region is referred to as the transitional region and it cannot be characterized properly by either low or high density assumptions [101, 103]. Table 1.3 gives approximate values for the boundaries between flow regimes. It is important to know what regime best describes the plume region plasma, as the validity of any probe theory depends on this heavily [112].

Figure 1.3 shows a schematic that maps some of plasma physics in a logarithmic scale for density and temperature. Typical examples are marked for astrophysical situations, some technical plasmas and the regime of control fusion. To supplement the overall picture of the vastness of these scales, some common parameters describing plasma behavior is listed in Table 1.4.

1.3.5 Sheaths

If quasi-neutral plasma ($n_e \approx n_i$) comes into contact with wall surfaces, it will generate a thin positive layer called a sheath. Quasi-neutral plasma floating in space has the same number of electrons as ions. However, the electron thermal velocity

$$\upsilon_{th_e} = \sqrt{\frac{k_B T_e}{m_e}} \tag{1.45}$$

is at least a hundred times larger than ion thermal velocity

$$\upsilon_{th_i} = \sqrt{\frac{k_B T_i}{m_i}}, \tag{1.46}$$

because $m_e \ll m_i$ and $T_e \geq T_i$.

Table 1.3: Knudsen numbers for various flow regimes [112].

Length Scales	Flow Regimes
$K_n > 10$	Free-molecular flow
$10 > K_n > .01$	Transitional
$1 > K_n > .01$	Slip
$K_n \ll .01$	Continuum

[13] Free molecular flow describes the fluid dynamics of gas where the mean free path of the molecules is larger than the size of the chamber or of the object under test. Knudsen number, in this case, is used to describe how gas flows through a tube in free molecular flow.

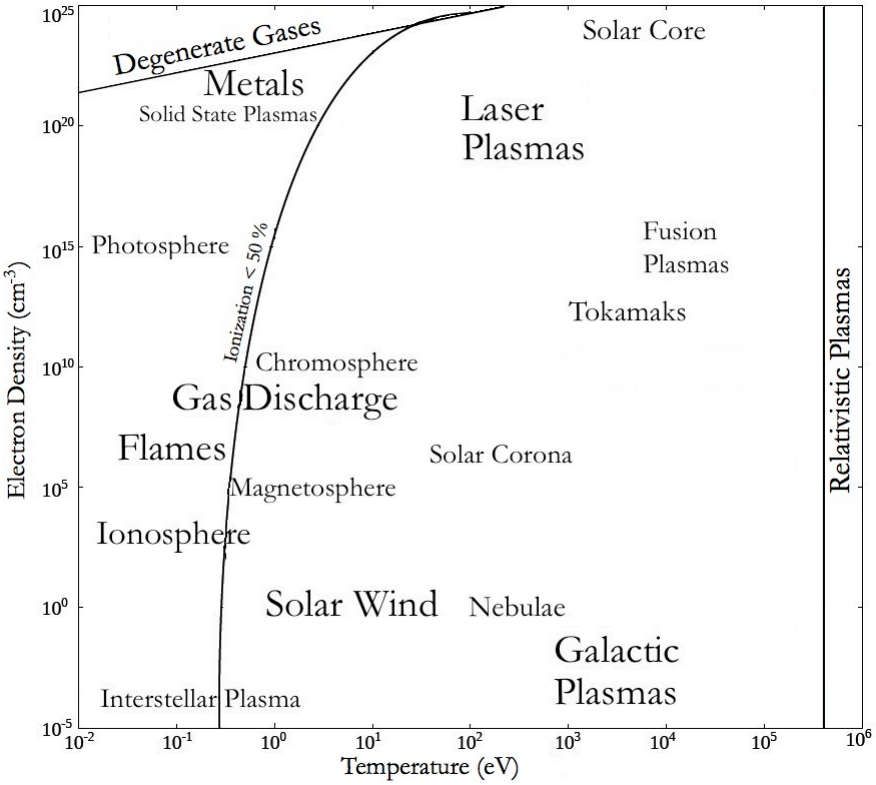

Figure 1.3: Different kind of plasmas in astrophysics, solar physics and solid state physics, nuclear fusion, and for technical applications in a plot of electron temperature T_e versus electron density n_e [103, 113]. Degenerate, relativistic and nonrelativistic plasmas are also denoted on the picture.

If a plasma is held between two grounded walls separated by a length l with a potential $V = 0$, as shown in Figure 1.4 (left), the electric potential V is zero everywhere, as well as the electric field $(E_x = -\nabla V)$, because there is no net charge density $\rho = e(n_i - n_e)$ present. There is no force preventing the faster electrons from escaping to the wall. This results in a charge discrepancy, because the slower ions remain in the plasma bulk, creating a positive charge, as shown in Figure 1.4 (right). Thin $(s \ll l)$ positive sheaths with $n_i \gg n_e$ form near the walls. This results in a potential $\phi(x)$ being positive within the plasma and then falling to zero at the walls. This potential acts like a base for the electrons, confining most of them in the plasma. The ions, however, fall down the potential hill formed by the electric field pointing from the plasma to the wall. If the plasma potential ϕ_p is on the order of a few electron temperatures T_e, most of the electrons will be confined to the plasma. The ions bombard the walls with a few times $k_B T_e$ (energy).

Table 1.4: Comparison of parameters for a wide variety of plasmas [103, 114].

Parameters	n	T	L	λ_D	r_L	$\ln\Lambda$	ω_p	ω_c
Units	(m^{-3})	(eV)	(m)	(m)	(m)		(Hz)	(Hz)
Gas Discharge	10^{16}	1	1	10^{-15}	-	10	10^{10}	-
Magnetic Fusion (Tokamak)	10^{20}	10^{4}	10	10^{-15}	10^{-5}	20	10^{12}	10^{12}
Inertial Fusion	10^{31}	10^{4}	10^{-5}	10^{-15}	-	8	10^{18}	-
Ionosphere	10^{11}	0.1	10^{5}	10^{-15}	10^{-1}	14	10^{8}	10^{6}
Solar Wind (near earth)	10^{7}	10	10^{11}	10	10^{4}	25	10^{5}	10^{2}
Magnetosphere	10^{4}	10	10^{8}	10^{16}	10^{4}	28	10^{5}	10^{3}

The sheaths appear where a plasma meets a solid such as a chamber wall, electrode, substrate, or probe. They arise in a plasma, as the electrons have a temperature at the same or greater order of magnitude as the ions, and they are much lighter [115]. They are faster than the ions by at least a factor of $(m_i/m_e)^{1/2}$ [79]. At the interface with a material surface, therefore, the electrons will fly out of the plasma, charging the surface potential as negative relative to the bulk plasma [115]. The sheath is comparable to the Debye length, but it varies to a certain extent depending on the applied bias voltage on the electrode. Sheaths control the collection of Langmuir probe current by affecting the dynamic of incoming charges via electromagnetic interaction.

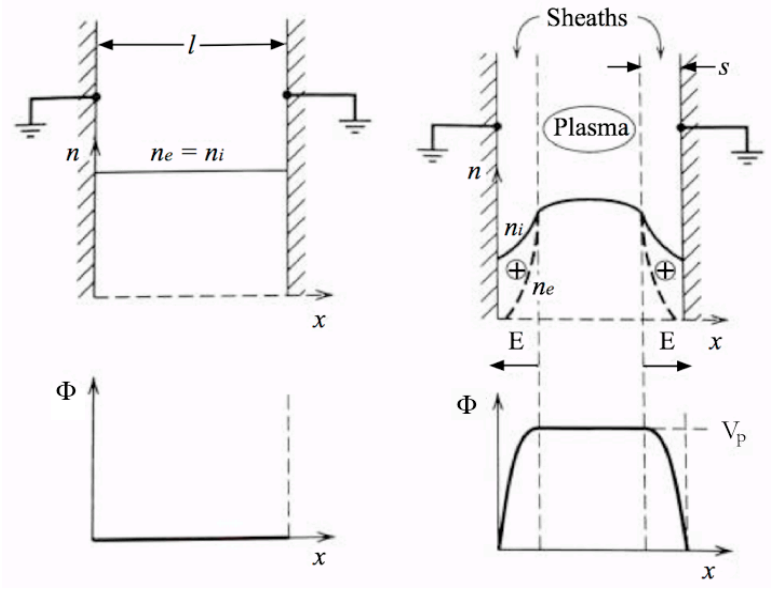

Figure 1.4: (Left) Initial electron densities and potential in the formation of a plasma sheath [7]; (right) densities, electrical field and potential after the formation of a sheath [7].

1.3.6 Collisionless Sheath

Disregarding the collisions, a few predictions can be made regarding the potential and ion velocity at the plasma-sheath interface. Let $x = 0$ be the position of the plasma-sheath interface; the plasma bulk is at $x > 0$. Furthermore, the ions have a velocity υ_s at the interface. The assumptions are presented below:

- No collisions,
- Maxwellian electrons at temperature $k_B T_e$,
- Cold ions ($T_i = 0$) assumes the ions are stationary relative to the electrons,
- $n_e(0) = n_i(0)$ at the plasma sheath interface,
- No ionization in the sheath.

The current density of ions entering the sheath at the edge of the plasma is given by

$$J_i = 0.6ne\upsilon_i \approx \frac{1}{2} n_s e \sqrt{\frac{k_B T_e}{m_i}} \,, \tag{1.47}$$

where n_s plasma number density at the start of the presheath far from the boundary, which was considered to be the center of the plasma by Langmuir for his collisionless plasmas [116]. The convention of approximating the coefficient 0.6 as $1/2$ was made by Bohm in defining what is now called the Bohm current [64].

The ion energy conservation can be postulated as

$$\frac{1}{2} m_i \upsilon^2(x) = \frac{1}{2} m_i \upsilon_s^2 - e\phi(x) \,. \tag{1.48}$$

Because there is no ionization in the sheath, the continuity of the ion flux yields

$$n_i(x)\upsilon(x) = n_{is}\upsilon_s \,, \tag{1.49}$$

with n_{is} being the ion density at the sheath edge. Solving equation (1.48) for $\upsilon(x)$ and substituting into equation (1.49), one arrives at

$$n_i = n_{is}\left(1 - \frac{2e\phi}{m_i \upsilon_s^2}\right)^{-1/2} . \tag{1.50}$$

The electron density is given by the Boltzmann relationship

$$n_e(x) = n_{es} \exp\left(\frac{e\phi(x)}{k_B T_e}\right), \tag{1.51}$$

according to the assumptions $n_{es} = n_{is} \equiv n_s$ at the sheath edge. Substituting n_i and n_e into Poisson's equation

$$\frac{\partial^2 \phi}{\partial x^2} = -\frac{\rho}{\varepsilon_o} = -\frac{e}{\varepsilon_o}(n_i - n_e), \tag{1.52}$$

one obtains

$$\frac{d^2\phi}{dx^2} = \frac{en_s}{\varepsilon_o}\left[\exp\left(\frac{e\phi}{k_B T_e}\right) - \left(1 - \frac{e\phi}{E_s}\right)^{-1/2}\right], \tag{1.53}$$

where $E_s = 1/2 m_i \upsilon_s^2$ is the initial ion energy. Equation (1.53) is the basic non-linear equation that describes the sheath potential and the ion and electron densities. Equation (1.53) can be integrated by multiplying it by $d^2\phi/dx^2$ and integrating over x:

$$\int_o^{\phi} \frac{d\phi}{dx}\frac{d}{dx}\left(\frac{d\phi}{dx}\right)dx = \frac{en_s}{\varepsilon_o}\int_o^{\phi}\left[\exp\left(\frac{e\phi}{k_B T_e}\right) - \left(1 - \frac{e\phi}{E_s}\right)^{-1/2}\right]dx. \tag{1.54}$$

The dx's cancel each other out, and one can integrate over $d\phi$. Corresponding to a field-free plasma, one can set $\phi(0) = 0$ and $d\phi/dx(0) = 0$:

$$\frac{1}{2}\left(\frac{d\phi}{dx}\right) = \frac{en_s}{\varepsilon_o}\left[k_B T_e \exp\left(\frac{e\phi}{k_B T_e}\right) - k_B T_e + 2E_s\left(1 - \frac{e\phi}{E_s}\right)^{1/2} - 2E_s\right]. \tag{1.55}$$

This can be integrated numerically. However, it is clear that the right hand side must be positive in order for a solution to exist. This means that, in the sheath region, the electron density must be less than the ion density [7] when the boundary is able to emit electrons. This can be a problem only for small ϕ. Equation (1.55) can be expanded to the second order in the Taylor series, and one obtains the inequality

$$\frac{1}{2}\frac{(e\phi)^2}{k_B T_e} - \frac{1}{4}\frac{(e\phi)^2}{E_s} \geq 0. \tag{1.56}$$

Substituting $E_s = m_i \upsilon_s^2/2$, this becomes

$$\frac{1}{2}\frac{1}{k_B T_e} - \frac{1}{2}\frac{1}{m_i \upsilon_s^2} \geq 0 \tag{1.57}$$

or

$$\upsilon_s \geq \upsilon_B = \sqrt{\left(\frac{k_B T_e}{m_i}\right)}. \tag{1.58}$$

This is the Bohm sheath criterion, where υ_B is the Bohm velocity. A small electric field must exist inside a plasma in order for the ions to achieve this velocity. This region is much wider than the sheath, and is known as the presheath.

1.4 Motivation and Overview

Computational research in EP overlaps experimental investigations in almost every area. Lifetime issues are addressed by investigating sputtering and erosion processes in the thruster [117, 118]. Integration issues are considered through studies of deposition and implantation [119], plume backflow [120], and through plasma control to mitigate signal interference [121]. High-power cluster configurations are simulated to evaluate the performance, with near and far field studies to characterize the exhaust plume [122]. Preliminary modeling is already underway to characterize prototype MEMS (thruster on a chip) devices [123].

A broader goal is to develop and refine computational models until they become sufficiently accurate to perform reliable assessment and characterization of EP devices. Meeting that goal would help streamline the design process for new devices and also enable more effective interpretation of experimental results. For example, a sufficiently robust computational model could be used to predict on orbit performance of a high power configuration from measurements made in small vacuum chambers where the pumping rate is too low. Experimental and computational researchers continually share results in EP, so it is desirable to have a clear understanding of how experimental measurements relate to undisturbed plasma conditions. EP devices produce highly non-equilibrium plasma flows. The most commonly used electromagnetic and electrostatic devices produce an exhaust plume consisting of high temperature electrons, high energy ions of various charge states formed in the thruster, high energy neutrals and low energy ions formed via charge exchange collisions, and low energy neutrals that diffuse out of the thruster without being ionized or accelerated [11, 34]. The plasma is low density, strongly flowing, and nearly collisionless as a whole [11].

This is markedly more complicated than the simple conditions of isothermal electrons[14] [85] and cold-drifting ions [124] assumed in the theoretical analysis of some plasma probe measurement techniques. One important open question is how well the theoretical probe techniques perform for a more complex, realistic EP plasma flow. A computational setting is especially well suited to evaluating this question, since the plasma conditions can be specified explicitly and

[14] Plasmas are often in a non-equilibrium state with different temperatures, T_e and T_i of electrons and ions. Such two temperature plasmas are typically found in gas discharges. The solar plasma (in the interior and photosphere), on the other hand, is a good example for the isothermal plasma with $T_e = T_i$.

probe measurements can be simulated and analyzed according to standard diagnostic techniques. By moving incrementally from the ideal plasma of the diagnostic theory to a realistic plasma, it is also possible to isolate the effects of each deviation from the ideal assumptions.

A second open question is how the insertion of a diagnostic probe disturbs the plasma flow, and whether that disturbance introduces a systematic error in the measurements the probe makes. In other words, it is not clear whether the disturbed conditions measured by the probe can be related back to the undisturbed plasma conditions. Again, a computational setting is ideal for investigating these effects since the undisturbed conditions are known explicitly as inputs. The remainder of this dissertation describes the development and use of computational models to help address these open questions.

This dissertation focuses on the advances made regarding the theoretical capability for simulating the operation of electrostatic probes, specifically Langmuir probes. Although many plasma diagnostic methods have been developed since the original work of Irving Langmuir [125], the Langmuir probe is still one of the most powerful plasma diagnostic tools due to its simplicity, low cost, adaptability, ability to determine the fundamental characteristics of a plasma (namely, the ion number density n_i, electron number density n_e, electron temperature T_e, EEDF, plasma potential V_p or floating potential V_f) with excellent spatial (of or relating to space) resolution, and simple theory of probe data interpretation [41].

The purpose of this dissertation is to introduce a PIC code, PROBEPIC. PROBEPIC was initially written to elucidate the beam-probe interaction and provide systematic techniques for legitimately interpreting experimental data. The benefits of PROBEPIC were constrained by its inability to model the operation of Langmuir probes in non-Maxwellian plasmas. However, the tool presented in this dissertation can now be used to evaluate the design of Langmuir probes to ensure proper response in non-equilibrium plasmas, as well as to assess the plasma conditions for Langmuir probe data with I-V characteristics that stray from the expected equilibrium results. This means that PROBEPIC is not only capable of simulating the existing Langmuir probe measurements to recreate the conditions and analyze the experimental data, but also capable of conducting, simulating and analyzing own probe measurements/experiments in both Maxwellian and non-Maxwellian plasma environments. I am compelled to use the term experiments because the PIC method, which uses computational particles to represent real electrons and ions, gets as close to reality as we can expect to on a computer. We are not simply numerically integrating a set of differential

equations; the PIC method introduces experimental realities such as statistical deviations. For example, the effect of probe dimensions or specific plasma conditions may be explored.

PROBEPIC can be used to provide better diagnostic capabilities, which can aid in the development of improved EP and other non-equilibrium plasma systems. Modeling using the described updated PROBEPIC model can also facilitate the efficient and cost effective design simulation and evaluation of diverse Langmuir probe geometries under diverse conditions. The modeling also allows for the evaluation of experimental data from Langmuir probes of known geometry when the probe data departs from anticipated current-voltage (I-V) characteristics.

1.5 Dissertation Organization

This dissertation is arranged as a linear sequence that starts with a review of experimental and numerical research related to EP. Chapter 2 introduces the current state of the art and the Langmuir probe theory including analytical techniques that have previously been developed to grapple with the difficult task of interpreting experimental Langmuir probe data. A description of the Maxwellian distribution is presented in sections 2.7; this is necessary because Langmuir probe theory was originally created based on the assumptions using Maxwell's equations. The collisionless plasma presents itself based on these equations, and the I-V curves can be fitted directly to obtain the temperatures and densities of both electron components without resorting to techniques that attempt to derive the plasma distribution from the current by taking derivatives. Following this, section 2.8 introduces non-Maxwellian distribution; this is because this work presents the formulas for and examples of Langmuir probe current collection in such non-Maxwellian plasmas, and shows when the electron energy distribution can be derived from measurements of the first and second derivatives in the retarding region of the I-V curves.

Chapter 3 focuses on numerical methods used to model the flow of plasma. The kinetic and fluid methods are briefly reviewed. The particles are treated kinetically, and their trajectories are followed as they move within a computational grid. All charged particles experience electric and magnetic forces according to Maxwell's equations, and electrons also experience electron-neutral and Coulomb collisional forces. Various other collision processes are also modeled. The simulation is nominally explicit (as opposed to implicit); that is, most quantities that affect the motion of a given particle are sampled from the previous time-step. This chapter also describes the various facets of the PIC and MC simulation kinetic models in detail establishing a theoretical framework for the algorithms used in PROBEPIC.

The Probe PIC computational code is used extensively to perform simulations of the plasma flow field around a cylindrical geometry that represents a Langmuir probe. The results are presented and discussed in Chapter 4. The computational code is profiled and high-resolution multi-grid method is implemented to solve the PIC model, which increased the modeling accuracy. As Langmuir probes are used in space, ionosphere, and in a wide variety of industrial or laboratory plasma devices, probe theories have been developed for very different operation regimes. This is done by implementations of the theoretical approaches that were described previously to evaluate the general validity of PROBEPIC output. This is done by classifying the given plasma conditions and probe properties in comparison with the existing probe theories as well as by empirical arguments regarding the reliability of the theoretical and the measured probe characteristics.

The dissertation is concluded in Chapter 5 with a review and summary of the results and new contributions of this work. This includes the results of the application of PROBEPIC to several problems and summarizes the present study. Finally, a few recommendations for future work in this area are outlined.

Chapter 2

Electrostatic Probes – The Langmuir Probe

The simplest form of electrostatic probe used for measuring the properties of plasma is a Langmuir Probe. The technique was developed by Langmuir and Mott-Smith in 1924 [125]. They discovered that many fundamental plasma parameters can be determined by placing a small conducting probe into the plasma and the current to the probe as a function of the difference between the probe and plasma potentials can be observed [125], i.e., the electron temperature and the electron number density are measured by a single Langmuir probe, and the theoretical form of the Langmuir characteristics are shown in Figure 2.1. A variable biasing voltage is applied on the electrode, which may be positive or negative with respect to the plasma [79].

Although Langmuir probe is a very simple device, it may have a rather complicated theory. This is because probe acts as a boundary and the equations governing the plasma motion change its characteristics. Due to the quasi-neutral behavior of plasma [79] a sheath is formed around the probe, which can sustain a large electric field. The characteristics of the Langmuir probe can be easily understood by plotting the I-V curve. Here I is the current flowing in the probe when a biasing voltage V is applied to it with respect to the plasma [88] or the body of the vessel containing plasma [2].

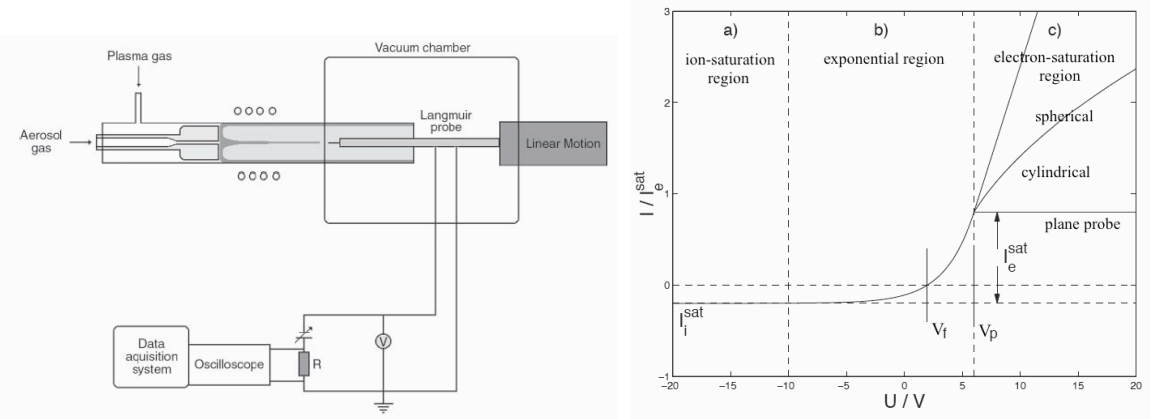

Figure 2.1: (Left) Schematic diagram of single probe measurement [126] and (right) theoretical form of the Langmuir characteristics: a) ion-saturation region, b) exponential region, c) electron-saturation region, V_f floating potential, V_p plasma potential. The increase in the probe current in the electron saturation region is due to sheath expansion effects and depends on the probe shape [127].

Probe configurations can be broken down into three broad groups defined by the shape of the probe tip. These are cylindrical [128-135], spherical [131, 136] and planar [62, 131, 135, 137-142], as shown in Figure 2.2. The cylindrical probe is a straight piece of wire typically made from tantalum or similar materials such as molybdenum, which are chosen for their high melting points and mechanical strength; tungsten and graphite are also popular choices. In contrast, the planar probe is a flat conductor that is typically single-sided, with the rest of the probe being insulated with a simple glass sleeve or coaxial cables built into the sleeve.

The choice of probe depends largely on what characteristics of the plasma are of principal interest. For example, the planar probes, while sensitive to plasma flow, are not of use in measuring the electron saturation current, as the current recorded depends on the probe geometry [143]. The regimes in which the probe will operate and the properties of the plasma that are of interest will decide the choice of probe shape selected. By carefully considering the voltage to be applied, the sheath size can be estimated and an appropriate probe size can be chosen; that way, the plasma will not be perturbed more than necessary [62]. Whether or not the plasma is collisional, along with the type of probe chosen, determines the probe theory, which in turn frames the interpretation of the results.

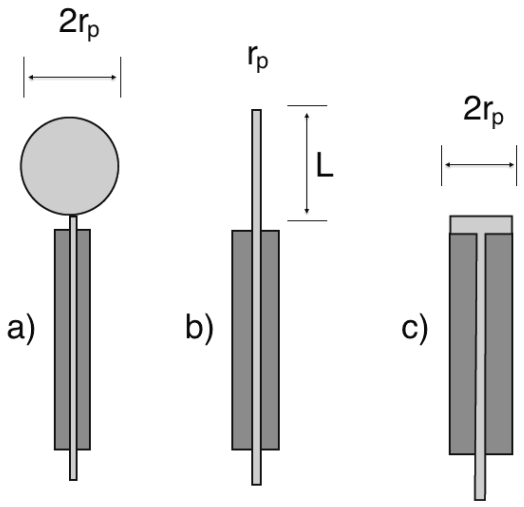

Figure 2.2: An illustration of different types of collecting Langmuir probes with radius r_p; a) spherical, b) cylindrical and c) planar geometries [47] .

2.1 Basic Theory of Operation

The simple approach to the probe theory [125] may be listed under the following assumptions:

- A Maxwell-Boltzmann electron velocity distribution is assumed,
- Plasma perturbations by the probe can be neglected,
- Every particle entering the sheath is collected by the probe.

Under these conditions, the total current I in the electron retarding region of the I-V curve (i.e., below the plasma potential) is given by

$$I = -I_i + I_{es} \exp\left(-\frac{V_p - V_{bias}}{T_e}\right), \tag{2.1}$$

where I_i is the constant ion current; the negative sign indicates positive ion collection. I_{es} is the saturation electron current, which decreases exponentially with sheath voltage ($V_{sh} < 0$, where $V_{sh} = V_p - V_{bias}$).

For an ideal probe, whose sheath expands with sheath voltage and whose sheath area remains constant, the collection area of the probe does not change throughout the I-V characteristics. The electron and ion saturation currents are determined by the ideal probe theory by

$$I_{es} = \frac{1}{4} e n_o A_p \bar{\upsilon}_e \tag{2.2}$$

$$I_{is} = -e n_s A_p \upsilon_B . \tag{2.3}$$

A_p is referred to as the probe surface area, n_o as the plasma density ($n_o = n_e = n_i$), and n_s as the sheath edge density, which is approximately 0.61 times n_o [7]. $\bar{\upsilon}_e$ is the mean electron speed, which is equal to $(8eT_e/\pi m_e)^{1/2}$ [15] [7], and υ_B [16] is the Bohm velocity determined by [7]

$$\upsilon_B = \sqrt{\frac{eT_e}{m_i}}, \tag{2.4}$$

[15] For example: $\bar{\upsilon}_e = 6.7 \times 10^7 \sqrt{T_e}$ cm/s, where T_e is in volts and n_e is in units of cm^{-3}.

[16] For example: $\upsilon_B = 9.8 \times 10^5 \sqrt{T_e / A_R}$ cm/s, where A_R is reduced mass (amu).

which is valid for $T_i \ll T_e$. m_i is the ion mass, and T_e is given in eV. The sheath edge is the point where the ions reach υ_B after they gain energy in the presheath. n_e is less than n_i within the sheath region for the positive charge to exist; the ion must gain energy in the presheath for the current flux to be balanced.

According to equation (2.1), the total current collected by the probe is a combination of ion and electron current. For $V_{bias} < V_p$, the ion current to the probe is constant $(I_{es} = I_i)$. When $(V_{sh} > 0)$ (i.e., the bias voltage is more than the plasma potential), the electron current is not an exponential function of voltage; it begins to saturate. The current above V_p, however, is determined by equation (2.2).

Equations (2.2) and (2.3) are true for an ideal planar probe whose sheath width varies as a function of bias; collection area is constant. For plasma diagnostics, cylindrical probes are usually preferred; their construction is simple, and they cause insignificant plasma perturbation compared with the planar probes [144]. At large voltages, the planar sheath cannot keep its shape and takes on a hemispherical shape [144]; the results are not ideal. For cylindrical probes, however, the collection area is a function of sheath voltage. As the collection area grows with bias, the current does not saturate as it does in the ideal planar case. Instead, it increases with a square-root dependence on V_{sh}. The effective collection area of a cylindrical probe is

$$A_{eff} = A_p\left[1+\left(\frac{d_{sh}}{r_p}\right)\right], \tag{2.5}$$

where A_p is the probe surface area, r_p is the probe radius, and d_{sh} is the sheath width given by Child's law [7],

$$d_{sh} = \frac{\sqrt{2}}{3}\lambda_D\left(\frac{2V_{sh}}{T_e}\right)^{3/4}, \tag{2.6}$$

where λ_D is given as $\left(\epsilon_o T_e/en_s\right)^{1/2}$, and n_s is sheath edge density.

The sheath expands with V_{sh}. As the sheath changes, it shields out the high electric field from reaching the quasi-neutral bulk plasma. Equations (2.2) and (2.3) are modified for sheath expansion associated with cylindrical probes.

For higher negative biases where the ion current is saturated, equation (2.6) becomes valid; an approximate value for the incomplete ion saturation is obtained when equation (2.6) is used with equation (2.5) and substituted into equation (2.3). A more valid equation can be illustrated with the following formula [145, 146],

$$I_i = en_i A_p \sqrt{\frac{eT_e}{2\pi m_i}} \left[a \left(\frac{e\left(V_p - V_{bias}\right)}{T_e} \right)^b \right]. \tag{2.7}$$

The term in the square brackets of equation (2.7) is a correction factor for the saturation ion current. The values a and b depend on the ratio of r_p/λ_D. For cylindrical probes, with $r_p/\lambda_D < 3$, a and b can be approximated by about 1.13 and 0.5, respectively [144].

Equation (2.7) presents the OML ion current to a cylindrical probe in a collisionless, Maxwellian plasma. OML theory was developed by Langmuir and Mott-Smith [147] for cylindrical probes; the ions entering the sheath with a particular angular momentum may actually miss the probe even though they are attracted to it. The effects of collisions are ignored; a sheath is generally considered collisionless if the sheath thickness, $\Delta d_{sh} = d_{sh} - r_p$, is less than the mean free path of either ions λ_i or electrons λ_e. This model is extended [145] to provide a numerical solution for the ion collection in collisionless sheaths. This theory is known [144, 148, 149] to overestimate the ion densities by providing an upper limit to the collected ion current in the low-pressure regime.

The electron collection is not affected by collisions in the same way; the fast electrons and the electron current collection are present near the plasma potential, where the sheath is small. Thus, they suffer less from sheath expansion effects. The equation for the electron saturation is determined by

$$I_e = en_e A_p \sqrt{\frac{eT_e}{2\pi m_e}} \left[\beta \left(\frac{e\left(V_p - V_{bias}\right)}{T_e} + 1 \right)^\gamma \right], \tag{2.8}$$

where the terms in brackets represent a correction factor for the sheath expansion, and the coefficients β and γ depend on the ratio r_p/λ_D. According to [150], as long as the probe potential is below $40\,V$ (positive or negative), collisionless theory is valid for electron current measurements over a wide range of densities [144, 148, 149]. To accurately determine the ion currents from collisionless theory, the pressure must be below $2.67\ Pa$ and a sufficiently high plasma density should be present. For all other cases, collisional effects must be used for accurate results.

2.2 I-V Characteristic

By varying the bias voltage of the probe, different current values can be measured. According to Mott-Smith and Langmuir [125], the floating potential, plasma potential, plasma density and electron temperature could all be measured from the I-V curve consisting of a voltage sweep from negative to positive potentials. A typical trace of the current-voltage (I-V) characteristics of plasma measured with the use of a Langmuir Probe is shown in Figure 2.3 illustrating the typical, somewhat idealized, experimental I-V characteristic for a Langmuir probe. Here negative, or electron, current to the probe is plotted against V, the probe voltage with respect to an arbitrary reference point. The potential V_p is the local zero, or plasma potential. For potentials greater than V_p, the probe attracts electrons; probe potentials less than V_p repel electrons. Ions having opposite charge, are repelled when $V > V_p$, and attracted when $V < V_p$. The qualitative behavior of this curve can be explained as follows.

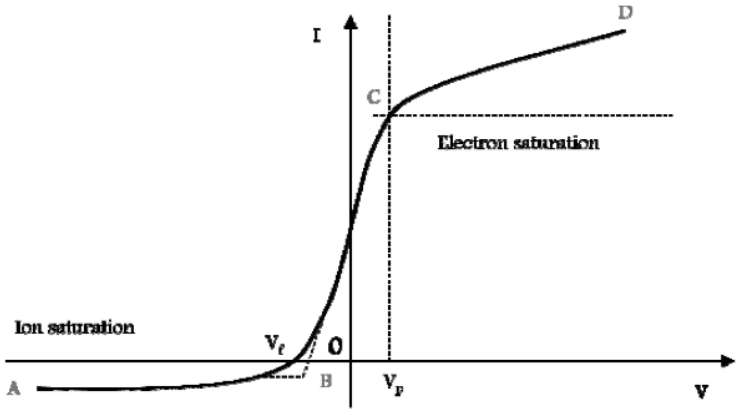

Figure 2.3: Ideal (undisturbed by the probe [151]) current-voltage (I-V) characteristic for a cylindrical Langmuir probe, showing how the probe current varies with probe potential V_p [152]. Electron saturation (right) current occurs when the probe is positively biased, and Ion saturation current (left) occurs in the case of negative biasing [79]. The amount of biasing voltage for which the probe current is zero, is known as floating potential V_f. The probe current increases in between ion and electron saturation current due to Maxwellian distribution in the plasma.

The graph is segmented into three regions [153, 154]. According to Figure 2.3, when the probe potential V is above the plasma potential V_p in region (C-D), the collected electron current reaches a saturated level and the ions are repelled, whereas the opposite occurs in region (A-B). By evaluating the slope of the electron I-V characteristic in region (A-B), the electron temperature T_e is obtained, and by measuring the ion saturation current I_{is} or electron saturation current I_{es} and using the T_e measurement, the density can be computed [153].

Although there is no exact analytical theory to determine the I-V characteristic for an arbitrary plasma condition, there are some simplifying assumptions that can make the problem more manageable. These are two approximate solutions for two limiting cases: a very thin sheath [155] and a very thick sheath [156] plasma condition. For thin sheaths ($\lambda_D \ll r_p$, where λ_D is the Debye length and r_p is the probe radius), space charge-limited current collection is assumed; that is, any particles crossing the sheath are collected, and the resulting I-V characteristic has very sharp knees; see Figure 2.3 at points (B and C). On the other hand, for very thick sheaths ($\lambda_D \gg r_p$), OML current collection is assumed, which is the most relevant theory for analyzing the probe I-V characteristics. The OML current is collected by the probe when none of the undisturbed particles (at infinity) that are capable of reaching the probe on the basis of energy considerations are excluded by the intervening barriers of effective potentials. The influence of the sheath is neglected, and particle orbits are calculated using the space-charge-free electric field of the probe. The orbits intercepting the probe contribute to the current. The I-V characteristic in the OML does not have the sharp knee at points (B and C); rather, there is a very smooth change from positive to negative curvature.

Region (A-B) is considered to be for values of the potential $V < V_f$, where V_f is the floating potential. This occurs when the bias voltage is negative and induces a current of positive ions. Due to the large masses and low mobility of the ions in the plasma, the current is measured to be small and negative. Ion saturation occurs rapidly as V is reduced below V_f between points A and B in Figure 2.3. The I-V curve will exhibit a plateau, and the current measured here is referred to as I_{is}, which is a characteristic ion saturation current that does not increase when further negative voltage is applied [131, 154].

$$I_{is} = n_i e A_p \left(\frac{2k_B T_e}{m_i} \right)^{1/2}, \tag{2.9}$$

where n_i is the ion density, m_i is the ion mass, and A_p is the surface area of the probe.

Region (B-C) involves values of V_f and V_p. As $V = V_p$, the electrons and ions diffuse to the probe as if they were unaffected by its presence. This local zero $(V = V_p)$ is characterized by the lack of a plasma sheath. On either side of this point, the difference in voltage introduces electric forces, creating a boundary region known as the plasma sheath, where charge neutrality does not exist. When there is no plasma sheath, the charged particles can migrate to the probe freely due to their thermal velocities. It is assumed that the probe current I_p does not interrupt the plasma equilibrium, and r_p is considered to be less than the electron mean free path [153].

$$I_p = \frac{1}{4} A_p n e \upsilon_{av}, \tag{2.10}$$

where n is the electron density in the immediate vicinity of the probe surface when $V < V_p$. A_p is the surface area of the probe, e is the electron charge, and υ_{av} is average electron speed. n can be determined by

$$n = n_e \exp\left(\frac{-e\left(V_p - V\right)}{k_B T_e}\right) = n_e \exp\left(\frac{\left(V_p - V\right)}{k_B T_{eV}}\right), \tag{2.11}$$

where k_B is Boltzmann's constant, T_e is electron temperature, V_p is plasma potential (space potential), and V is probe potential. The average electron speed can be calculated by

$$\upsilon_{av} = \sqrt{\left(\frac{8 k_B T_e}{\pi m}\right)} = \sqrt{\left(\frac{8 k_B T_{eV}}{\pi m}\right)} = 6.7 \text{x} 10^5 \sqrt{T_{eV}}\, \text{ms}^{-1} \tag{2.12}$$

Taking the natural logarithm of equation (2.11) and using equation (2.10), the following formulas can be calculated:

$$\ln I_p = \ln I_{ps} - \frac{V_p}{T_{eV}} - \frac{V}{T_{eV}}, \tag{2.13}$$

$$I_{ps} = \frac{1}{4} A n_e e \upsilon_{av}, \tag{2.14}$$

where I_{ps} is the probe current when $V = V_p$, and it is calculated by using equation (2.3). Then, the slope of $\ln I_p$ versus probe potential V is determined by

$$\frac{d \ln I_p}{dV} = \frac{1}{T_{eV}}. \tag{2.15}$$

As the potential is decreased from the plasma potential, the electrons are continually repelled as per the Boltzmann relation for the electron density n_e [131, 143]. The reference potential is taken to be at such a value that the electron

density n_e equals the undisturbed plasma density n_o. Starting at the plasma potential, electrons are initially attracted to the positive potential of the probe, and they will form a sheath around it to screen out the electric field. As the sheath is established, a negative space charge gathers around the probe. The electrons are repelled by electrostatic interactions as the probe potential is reduced to lower than the plasma potential. The positive ions are attracted to the negative sheath formed around the probe to preserve the electrical neutrality of the plasma, and some electrons are permitted to reach the probe by the effect of electrostatic interactions. In this region, the electron current is retarded; this region is called the electron retardation region (A-B in Figure 2.3). The theory for a probe in an ideal plasma indicates that the current in the region (A-B) would vary linearly with the probe voltage. However, the current varies exponentially in reality. The electron retardation continues until the potential is sufficient to reduce the electron current to a small fraction of its saturation value. The flow of positive ions and electrons will be equal to each other, yielding a net current of zero. The potential occurring here is referred to as V_f [157],

$$V_f = -\frac{k_B T_e}{e} \ln\left(\frac{m_i}{4\pi m_e}\right)^{1/2}, \tag{2.16}$$

where m_i is the mass of the ion and m_e is the electron mass. V_p is the apparent potential of the bulk region of the plasma and the potential against which the bias to the probe is applied. The geometry of the probe and the nature of the plasma will determine how it is calculated. For example, in the case of a cylindrical probe in a collisionless plasma, the plasma potential can be found by taking the first derivative of the electron current and calculating where the maximum occurs [158].

In Region (C-D), the probe potential is higher than the plasma potential, and the electron current cannot increase any further. Because of a positive potential bias, the electrons are accelerated towards the probe and the positive ions are repelled. As the bias voltage is increased, space charge limited electron current will take place, and electron saturation will occur. Further increases in voltage will not increase the electron current. The current in this section is equal to the electron saturation current. The slope of the curve $\ln I_p$ versus V is calculated when $I_{is} << I_p$, and the following formula is used:

$$I(\upsilon) = -neA_p\left(\frac{k_B T_e}{2\pi m_e}\right)^{1/2} \exp\left(\frac{eV}{k_B T_e}\right) = I_{es} \exp\left(\frac{eV}{k_B T_e}\right). \tag{2.17}$$

There may be some concerns about its accuracy [131, 154] due to the sensitivity of the resulting value to the geometry

of the probe tip. Therefore, the probe geometry should be determined according to the size and shape of the sheath, which also affects the current collected in this region.

As a result, there are two points of asymmetry between saturation ion and saturation electron collection aside from the obvious one of the mass difference, which causes the disparity in the absolute magnitude of the currents. The first point is that the ion and electron temperatures are usually unequal (different than the ideal theory), and it turns out that sheath formation is considerably different when the colder species is collected than when the hotter species is collected. The second point is that when there is a magnetic field, the motion of the electrons is much more affected by the field than the motion of the ions [88]. These two points, which were neglected in the original theory of Langmuir, are responsible for making difficult the simple and straightforward application of probes as originally proposed by Langmuir. It is also evident that the formation of the sheaths plays an important role in the probe theory.

2.3 Enhancements to the Basic Theory

The Langmuir probe current models presented in section 2.1 add significantly to the understanding of probe behavior. However, they are valid for non-drifting, unmagnetized, and collisionless plasma that is being probed with an ideal sensor. The accuracy of the plasma parameters derived from a Langmuir probe is uncertain unless all the non-ideal factors are applied correctly. This section presents an overview of the complications involved with understanding Langmuir probe measurements. For detailed theory, see references [8, 44, 154, 159, 160].

A. Additional Current Sources: The contributions of other current sources to a probe must be identified and removed before calculating the terms in section 2.1 to determine plasma parameters (e.g., photoelectron current). This is a result of exposure of the probe surface to UV radiation, which affects the I-V curve in the ion saturation and the electron retardation region as a constant offset [161]. The electron saturation region is affected until the applied probe potentials are large enough to attract all of the photoelectrons back to the surface [161].

Another type of charging is the result of two objects constructed of insulating material that charge each other as they are rubbed asymmetrically. The sign of charge does not depend on the difference in work function, and it is also correlated with the nature of rubbing experienced by the piece of material [162, 163], which is commonly referred to as frictional charging and is also known as triboelectric charging. This is an important mechanism for Langmuir probes in dusty environments.

B. Mesothermal Plasma: This type of plasma occurs when the thermal speed of ions is less than the spacecraft speed and the electron thermal speed is higher than the spacecraft speed [164], which affects the ion saturation region of the spacecraft in a sweeping motion because the ions are largely collected in the direction of satellite's motion (ram direction). At orbital velocities, this condition creates a rarefied wake region behind the probe [165], changing the collected ion current as the projected area of the probe in the ram direction changes [165].

C. Magnetized Plasma: This is a result of an ambient magnetic field that is strong enough to alter particle trajectories, giving them different velocities parallel and across the field lines [166]. Magnetized plasmas are anisotropic, and they respond differently to forces parallel and perpendicular to the direction of the magnetic field [145]. This affects the collected current as the orientation of the probe changes with respect to the local magnetic field [145].

D. Collisional Plasma: For this type of plasma, there are two limiting cases [72, 152, 167]: OML current collection (known as thick sheath theory) and SL current collection (also known as thin sheath theory) [43, 72, 152, 168]. If the mean free path of the particles h is less than the sheath extent, the equation of motion for particles within the sheath is different from the free-fall collisionless equations used to drive OML current collection equations, which affects the potential profile within the sheath. The collected current increases as the orbital motion of the particles are disrupted. When h is smaller than sheath size but larger than the probe radius r_p, the increase in collected thermal current can be approximated by h/r_p [8]. When the mean free path is not very much larger than r_p, the particle velocity distribution at a distance of h from the probe surface will have its collisional region further away from the probe. The plasma in this area close to the probe surface can be calculated carefully using a numerical approach, e.g., PIC simulation [152].

2.4 Instrument Implementation Issues

In reality, for precise Langmuir probe measurements, it is essential to overcome the limitations of the design by removing the sources of implementation error [44]: contamination of the probe surface, poor positioning of the probe, insufficiently uniform collector surface material, electronics not resolving the I-V characteristics, non-uniformity of the work function on the probe surface, and magnetically induced potential gradient due to movements of the space vehicle [44]. There are several ways to implement an electrical probe, and each of which has certain advantages in deriving a

particular plasma property. However, designing a probe to measure the space environment behaving in an expected way is not easy for researchers [8, 44].

Selecting an appropriate size for the probe is important to ensure its functioning in the desired regime, i.e., thin sheath or thick sheath [43, 72, 152, 168]. Irregular functioning leads to hysteresis in the I-V curve as the probe sweeps up and down, which affects T_e measurements [169, 170]. This may be caused by inherent irregular functioning of the metal surface of the probe because of the non-uniform coating of the probe with inert metals or alloys such as gold, titanium nitride [171], rhenium, or molybdenum [44]. Another approach is to make the probe uniformly dirty by water or alcohol based carbon coatings. Surface contamination due to adsorption of impurities also imparts an insulating patchiness to the probe even if the potential of the underlying metallic probe is highly uniform. This happens when a probe is cleaned, sealed in an airtight container, and released only when the spacecraft reaches space or when an inflight cleaning method is employed, e.g., internal heating to boil off the contaminants [172] or applying high potentials for short periods of time [173, 174].

If the probe collects additional electron current by operating in the electron saturation region, the spacecraft surface closes the loop by collecting additional ions and fewer electrons. As the ion current is more than an order of magnitude smaller than the electron current, the surface area of the spacecraft has to be much larger than that of the Langmuir probe in order to prevent the floating potential of the spacecraft from varying significantly. If the area ratio of spacecraft surface to probe surface is not on the order of 1,000 or greater, then the spacecraft floating potential will change significantly during probe operation [175].

In the case of a sweeping Langmuir probe, a drifting spacecraft potential will lead to oscillations in the spacecraft ground potential as the probe performs its sweep from negative to positive potential and then back to negative potential. When one designs a probe, a large enough ratio of spacecraft surface area to probe area should be carefully considered. The area of the probe also determines the magnitude of current collected, which in turn affects the sensitivity of the electronics used. The performance of electrical printed circuit boards varies with the board temperature. To account for these variations, an accurate calibration is required over the expected temperature ranges. The calibration efforts become more difficult for probes that have small surface areas and are expecting to measure currents in picoAmperes. Such an instrument must be designed in a way to eliminate board leakage currents to critical measurement circuits. More information regarding probe implementation can be obtained from the reference [53].

2.5 Theoretical Approaches

When formulating theoretical models of current collection in plasma, the laws of motion for individual particles quickly grow beyond what is manageable for analysis: i.e., no closed form solutions exist [176]. Thus a practical theoretical description requires assumptions to be made. Two different sets of analytical approximations [72, 152, 167] are used to describe plasma/probe interactions. In both cases, probe current collection and sheath-boundary conditions are uniform, isotropic, collisionless, of single ion species, unmagnetized, and in steady state [125]. Two regimes exist to describe the plasma surrounding the probe: the SL current collection regime, also known as thin sheath, and the OML regime, also known as thick sheath [43, 72, 152, 168]. Both regimes are illustrated in Figure 2.4.

2.5.1 Thin Sheath: Space Charge Limited Current

The physical situation is clarified by Figure 2.4. Suppose that the velocity distribution is essentially Maxwellian at the edge of the sheath. This situation applies, for instance, to the part for the saturation electron current (I-V Characteristic in section 2.2), since in most plasmas $T_e >> T_i$, and the collection of the hotter species does not require a large drift velocity at the edge of the sheath. Suppose that the Debye length λ_D is much smaller than probe radius r_p and that the probe is much smaller than the mean free paths λ (mean distance traveled between collisions by a moving particle) for electrons and ions (λ_e and λ_i, respectively) in the plasma [8, 177]. This corresponds to low-pressure (<1Torr) conditions.

It also is assumed that the electron and ion densities are equal (assumption of quasi-neutrality), that Maxwellian velocity distributions are present far from the probe, that there is no drift velocity associated with the plasma and that the probe is not emitting electrons [8, 43]. The calculation of the ion and electron fluxes is required to determine the current collected by the probe [8, 43]. An I-V trace can then be produced from the electron temperature T_e and the Maxwellian velocity distribution, and the plasma electron density and floating potential can be determined [8, 43, 178].

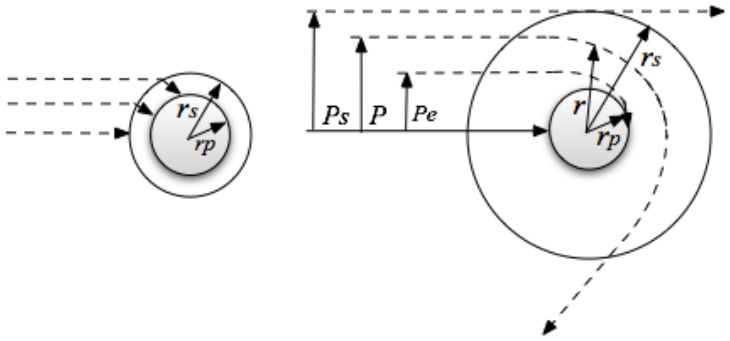

Figure 2.4: Geometry associated with electron attraction: Thick (right) *versus* thin (left) sheathed probes, where r_s is sheath radius, r_p is probe radius and P is the impact parameter (i.e., subscript e refers to effective probe radius and s to sheath radius) [155].

To calculate the electron density, the local velocity distribution function is used. Integrating this function over the relevant electron velocities yields [154]

$$n_e = \int_{-\infty}^{\upsilon_c} n_o \left(m/2\pi k_B T_e\right)^{1/2} \exp\left(\left(-1/2m\upsilon^2 + eV(x)\right)/k_B T_e\right) d\upsilon \,, \tag{2.18}$$

$$\upsilon_c = \sqrt{2e\left[V(x) - V(x_p)\right]/m_e} \,, \tag{2.19}$$

where υ_c is the critical velocity (below this velocity, the electrons will not reach the probe), $V(x)$ is the potential at an arbitrary point in the plasma, $V(x_p)$ is the potential at the probe surface, k_B is Boltzmann's constant, m_e is the electron mass, n_e is the electron density, n_o is the undisturbed plasma density, and T_e is the electron temperature. In equation (2.18), x is taken to be zero at the probe surface. The integration and simplification of equation (2.18) gives the Boltzmann relation for n_e,

$$n_e(x) = n_o \exp\left(eV(x)/k_B T_e\right). \tag{2.20}$$

The calculation of the ion and electron fluxes is required to determine the current collected by the probe [179]. Equation (2.20) also applies to ions because of the quasi-neutrality assumption. Considering the assumption that the ion temperature is low enough such that the kinetic energy of the ions far from the probe is zero, the ion flux is calculated as [180]

$$\Gamma_i(x) = n_i(x)\upsilon_i(x) = n_o \exp\left(eV(x_s)/k_B T_e\right)\left(-2eV(x_s)/m_i\right)^{1/2}. \tag{2.21}$$

For the electrons, the flux is given by the random flux [180],

$$\Gamma_e = (1/4) n_e \bar{\upsilon}_e , \tag{2.22}$$

where $\bar{\upsilon}_e$ is the mean electron speed. Using the Maxwellian relationship for the mean electron speed [180], which is

$$\bar{\upsilon}_e = \left(8 k_B T_e / \pi m_e\right)^{1/2}, \tag{2.23}$$

and the electron flux is calculated by [180]

$$\Gamma_e = (1/4) n_o \exp\left(eV(x_p)/k_B T_e\right)\left(8 k_B T_e / \pi m_e\right)^{1/2} . \tag{2.24}$$

The total current is given by [180]

$$I = e\left[-(1/4) A_p n_o \exp\left(eV(x_p)/k_B T_e\right)\left(8 k_B T_e / \pi m_e\right)^{1/2} + A_s n_o \exp\left(eV(x_s)/k_B T_e\right)\left(-2V(x_s)/m_i\right)^{1/2}\right], \tag{2.25}$$

where A_p and A_s are the probe and sheath areas, and $V(x_s)$ and $V(x_p)$ are the potentials at the edge of the sheath and at the probe [180]. Assuming that the plasma potential is much greater than the probe potential, the result is [180]

$$V_s = -k_B T_e / 2e . \tag{2.26}$$

After substituting this into equation (2.25) and assuming that $A_p \approx A_s$, the total current to the probe is [180] reduced to equation (2.27),

$$I = -eA_p\left[-(1/4) n_o \left(8 k_B T_e / \pi m\right)^{1/2} \exp\left(eV(x_p)/k_B T_e\right) - n_o \left(k_B T_e / m_i\right)^{1/2} \exp(1/2)\right] \tag{2.27}$$

From this equation, the electron and ion saturation currents are found by [180]

$$I_{es} = -(1/4) e A_p n_o \upsilon_e , \tag{2.28}$$

$$I_{is} = e A_p \exp(1/2) n_o \upsilon_i = 0.61 e A_p n_o \upsilon_i . \tag{2.29}$$

The total current is simply given by [180]

$$I = I_{es} \exp\left(eV(x_p)/k_B T_e\right) + I_{is} . \tag{2.30}$$

The electron temperature T_e can be calculated by solving for the slope of equation (2.18) in the exponential region after the ion current has been subtracted. One can calculate the plasma density using equation (2.20), and by setting equation (2.27) equal to zero, one can determine the floating potential (the point at which the applied probe potential results in no probe current [180]). Thin sheath theory makes the implicit assumption that the ion temperature T_i is much less than the electron temperature [181]. However, finding the electron temperature is not limited to equation

(2.27) or to whether or not the model follows thin sheath theory; it can be obtained from the I-V trace [182]. Incremental increases in probe potential and corresponding small increases in current associated with points in the I-V characteristic are often referred to as the electron saturation potential and current.

2.5.2 Thick Sheath: Orbital Motion Limited (OML) Current

In the opposite limit of a thick sheath (known as OML [34, 78]), the probe's effective radius P_e is greater than the probe radius r_p. Consequently, not all particles entering the sheath will hit the probe because of the possibility of orbital motions [177]. The laws of conservation of energy and angular momentum concern only the initial and final values of energy and angular momentum.

The effective radius of the probe increases with the applied voltage due to particle orbit motion. Collection of individual particles is dependent on their impact parameter P, which determines the orbital motion of particles around the curved probe surface [8, 177]. The number of electrons absorbed by the probe is determined solely by energy and angular momentum [8, 177]. It has been shown that the probe current (at a constant temperature) is proportional to the plasma density [8, 177]. Referring to Figure 2.4, the conservation of energy and angular momentum for the electrons gives [177]

$$1/2mv_o^2 = 1/2mv_r^2 - eV_r = -eV_o\,, \tag{2.31}$$

$$Pv_o = rv_r\,, \tag{2.32}$$

where $eV \le 0$, and r is the distance of closest approach to the probe of radius r_p [177].

$$1/2mv_r^2 = 1/2mv_o^2\left(1 + V_r/V_o\right), \tag{2.33}$$

$$P = r\left(v_r/v_o\right) = r\left(1 + V_r/V_o\right)^{1/2}. \tag{2.34}$$

The effective probe radius is shown as P_e (see Figure 2.4).

For mono-energetic particles, the flux to a probe of length l_p [177] is given by

$$\Gamma = 2\pi r_p l_p \left(1 + V_r/V_o\right)^{1/2} \Gamma_r\,, \tag{2.35}$$

where Γ_r is the random flux of ions of such energy. However, this result is extended to energy distributions that are Maxwellian at some large distance $r = s$ from the probe, where s is the sheath edge. The random flux Γ_r is then given by [177]

$$\Gamma_r = n\left(k_B T_i / 2\pi m\right)^{1/2}. \tag{2.36}$$

Because the sheath radius s for OML was taken to be infinite, the density has to be so low that the sheath is much larger than the probe [177].

The current supplied to the probe by electrons in a differential velocity range $d\upsilon$ is given by [183]

$$dI = 2\pi r_p l_p e\left(1 + V_r / V_o\right)^{1/2}\left(\upsilon/\pi\right)dn\,, \tag{2.37}$$

where $(\upsilon/\pi)dn$ is the flux of electrons moving perpendicular to the probe. Those moving parallel to the probe are not collected. Assuming a Maxwellian distribution of velocities, the differential number density dn is the number of electrons with velocity υ and is calculated by [183]

$$dn = n_o\left(m/2\pi k_B T_e\right)\exp\left(-m\upsilon^2/2k_B T_e\right)2\pi\upsilon d\upsilon\,. \tag{2.38}$$

After integrating equation (2.37), the electron current collected by the probe is determined by [177]

$$I_e = 2\pi n_o r_p l_p e\left(k_B T_e / 2\pi m\right)^{1/2}\left[\left(2\sqrt{\eta}/\pi\right) + \exp\left(\eta\right)\mathrm{erfc}\left(\sqrt{\eta}\right)\right], \tag{2.39}$$

where η is the non-dimensional probe bias, which is defined as

$$\eta = eV_r / k_B T_e\,. \tag{2.40}$$

For values of $\eta \geq 2$, the electron current collected by the probe gives more accurate results, calculated by [177]

$$I_e = 2\pi n_o r_p l_p e\left(k_B T_e / 2\pi m\right)^{1/2} 2/\sqrt{\pi}\left(1 + eV_r / k_B T_e\right)^{1/2}. \tag{2.41}$$

The probe voltage appears only once in the square root term. Therefore, the square of the I-V curve in the electron saturation region should be a straight line. Using this plot, n_o is determined from the slope, and T_e is determined from the intercept with the current axis [183]. Then, the ion collection is calculated by [177]

$$I_i = 2\pi n_o r_p l_p e\left(k_B T_i / 2\pi m_i\right)^{1/2} 2/\sqrt{\pi}\left(1 + eV_r / k_B T_i\right)^{1/2}. \tag{2.42}$$

If the equation for ion collection is applied, the only quantity that can be obtained from OML theory is the plasma density [152]. If the plasma potential is known, then the plasma density is derived from the slope of the square of the ion current *versus* the probe bias voltage [152].

2.6 Ideal Plasma Simulations

The approach of the PIC method is to simulate the plasma utilizing a set of super particles, which are subject to Newtonian or relativistic equations of motion, where the fields are calculated by solving Maxwell's equations using the source terms derived from the particle positions and velocities [184]. PROBEPIC was based on these same principles; an algorithm was used to generate random speeds, which after many calls produced a Maxwellian distribution [152]. The Langmuir/Mott-Smith theory [125] was the first to describe OML current collection. OML is the first approximation of the theory that explains the thick sheath limit [8, 152, 185], i.e., where the Debye length is much larger than the probe radius.

A considerable amount of research has been conducted in this area since the establishment of this theory, and several computer codes have been created to simulate Langmuir probes with well-defined geometries (e.g., planar, cylindrical or spherical, as illustrated in Figure 2.1) under ideal plasma conditions (Maxwellian plasmas) [43, 180, 186]. This theory also had an important impact on the understanding of plasma regimes that did not correspond to the OML limit. The complexity of the probe theory was demonstrated by the wide range of probe research [43, 72, 131, 152, 180, 185, 186]. Experimental results often deviate from the ideal I-V characteristic curve [185]. To accurately model experimental measurements, researchers have applied modern computational tools to simulate the interaction of a Langmuir probe with plasma [79, 185, 187]. Using PIC coupled with MC techniques, predicting the probe characteristic curve to some degree of accuracy has been made possible [152, 184, 185, 188]. The first version of the PIC probe simulation code (PROBEPIC) [72, 152] is a notable example of such a tool. PROBEPIC surpasses the work of previous approaches by simulating the behavior of finite length probes [187]. This tool was used to conduct a parametric study on the Langmuir probe and to interpret available experimental data [185].

2.7 Maxwellian Distribution

2.7.1 Maxwell Distribution of Speeds

To describe the distribution of particle speeds without caring for the orientation of motion, Maxwell distribution of particle speeds $\upsilon = |\mathbf{v}|$ is given by

$$f_M(\upsilon) = 4\pi\upsilon^2 n\left(\frac{m}{2\pi k_B T}\right)^{3/2} \exp\left(-\frac{m\upsilon^2}{2k_B T}\right). \tag{2.43}$$

The additional factor $4\pi\upsilon^2$ arises from the fact that the volume element in three-dimensional velocity space is a thin sheet on the surface of a sphere [85]. Hence, for $\upsilon << \upsilon_T$ the distribution of speeds rises like υ^2 because of the increasing number of combinations of υ_x, υ_y and υ_z that lead to the same speed $\upsilon = (\upsilon^2_x + \upsilon^2_y + \upsilon^2_z)^{1/2}$. The characteristic velocity is most probable speed of the gas. For larger speeds, the Maxwellian distribution of speeds decays because the exponential decreases much more rapidly than the growth of the spherical surface in velocity space.

2.7.2 Moments of the Distribution Function

The mean thermal speed of a gas is defined as the first moment of the distribution of speeds

$$\upsilon_{th} = \frac{1}{n}\int_0^\infty f_M(\upsilon)d\upsilon = \left(\frac{8k_B T}{\pi m}\right)^{1/2}, \tag{2.44}$$

where υ_{th} is thermal speed, which is 13 percent larger than υ_T, characteristic velocity. The mean kinetic energy of a gas is defined by the second moment of the distribution of speeds,

$$\langle \epsilon_{kin} \rangle = \frac{1}{n}\frac{m}{2}\int_0^\infty f_M(\upsilon)\upsilon^2 d\upsilon = \frac{3}{2}k_B T. \tag{2.45}$$

To eliminate the confusion between the mean kinetic energy and temperature in plasma physics, it is good to convert temperature units into energy units by the relation, $\epsilon = k_B T$ $(1eV = 11{,}600^o K)$.

2.7.3 Distribution of Particle Energies

For many calculations we need the particle distribution function on an energy scale, $\epsilon = 1/2 m\upsilon^2$. For this purpose, it is necessary to perform a transformation from speed to energy, which conserves the number of particles in a certain velocity and energy interval, respectively, $f_M(\epsilon)d\epsilon = F_M(\epsilon)$. With $d\epsilon = m\upsilon d\upsilon$, the following equation is obtained [81]:

$$F_M(\epsilon) = n\frac{2}{\sqrt{\pi}}\frac{1}{(k_B T)^{3/2}}\epsilon^{1/2}\exp\left(-\frac{\epsilon}{k_B T}\right). \tag{2.46}$$

For gas discharges, the terminology EEDF and electron energy probability function (EEPF) [81] are used. For a Maxwellian, the EEDF $F(\epsilon)$ [81, 88]can be identified with $F_M(\epsilon)$, but in most gas discharges $F(\epsilon)$ is non-Maxwellian distribution. EEPF is defined as

$$g(\epsilon) = \epsilon^{-1/2}F(\epsilon)\ . \tag{2.47}$$

In a semilog plot, $\log[g(\epsilon)]$ *versus* ϵ, we obtain a straight line when the distribution is Maxwellian. The EEPF is the immediate result from evaluating the second derivative of a Langmuir probe characteristic. The particle density can be obtained from the integral of the EEDF

$$n_e = \int_o^\infty F_M(\epsilon)d\epsilon \tag{2.48}$$

and the effective temperature

$$\frac{3}{2}k_B T_e = \frac{1}{n_e}\int_o^\infty \epsilon F_M(\epsilon)d\epsilon\ . \tag{2.49}$$

2.8 Non-Maxwellian Distribution

A Langmuir probe is usually employed under the assumption that the electron velocity distribution is Maxwellian, which makes the calculations simple enough to handle routinely. However, in non-Maxwellian electron velocity distributions, electrons are not in energy equilibrium with ions or neutral particles, and the electron temperatures are much higher than the ion and neutral temperatures [34]. Hence, conventional analysis and the use of a Maxwellian EEDF are not effective methods for the study of non-equilibrium plasma phenomena [189].

The EEDF in low-pressure discharges is generally non-Maxwellian, and the electron temperature is usually thought of as an effective electron temperature corresponding to the mean electron energy determined from the EEDF [81]. In

some cases, non-Maxwellian effects can be ignored when inferring plasma parameters from the probe characteristics and assuming a departure of the actual EEDF from a Maxwellian distribution only for the small number of electrons with energies higher than the energy of the inelastic threshold [81, 190]. This is not always the case, as the EEDF in low-pressure discharges is not Maxwellian even in the low energy range [54, 81, 112, 191].

There are two methods for obtaining the plasma parameters from Langmuir probe measurements. The first of these methods is based on EEDF integrals [158], and the second method employs a fluid model for the modified ion flux [192]. The approach taken in this dissertation uses the EEDF integrals [186], where the EEDF is measured by a double differentiation of the I-V characteristics according to the Druyvesteyn formula, which is considered more accurate for non-equilibrium distributions [186, 193]. For comparison, numerical methods are used to construct simulated probe data that would be obtained by a cylindrical probe in plasma with a two-temperature Maxwellian distribution (detailed in Section 2.8.2). This comparison is important because the distributions transition from a Druyvesteyn to a two-temperature Maxwellian distribution at certain values of pressure [7, 194]. For example, in argon discharges, the EEDF shows a two-temperature Maxwellian structure at low pressures (circa 1mTorr) and becomes Druyvesteyn-like at higher pressures (above 10mTorr). Therefore, to reproduce the I-V characteristics, it is important to implement the two-temperature distribution, which could potentially be used as an alternative to the Druyvesteyn distribution when fitting experimental data. As a test case in the present work, the EEDF integral method is applied to experimental data from a plasma device that has a two-temperature electron distribution, and the results of the methods are then compared to the available experimental data [195].

2.8.1 Druyvesteyn Method

The Druyvesteyn distribution was derived for the EEDF in gas discharges in which the electron temperature is much greater than the ion and neutral temperature. This is the primary distinction between the Druyvesteyn and Maxwellian distributions [112, 193]. The Druyvesteyn EEDF is based on hard sphere collisions between electrons of energy ϵ and stationary neutrals in an electric field $\mathbf{E}$. The amount of kinetic energy lost by an electron in one collision [112, 193] is

$$\Delta\epsilon = -2m_e\epsilon/m_i, \tag{2.50}$$

where ϵ is the energy, m_e is the electron mass and m_i is ion mass. The expression for energy loss in equation (2.50) is expanded to characterize the total energy loss for all electrons per unit volume and per unit time, based on the mean free path λ and electron velocity υ_e [193].

$$F(\epsilon)\left(2m_e/m_i\right)\epsilon\left(\upsilon/\lambda\right)=F(\epsilon)\left(2m_e/m_i\right)\left(\epsilon/\lambda\right)\sqrt{2\epsilon/m_e}\,, \tag{2.51}$$

where $F(\epsilon)$ is the EEDF. For the Druyvesteyn distribution [193] equate equation (2.51) with the energy taken from the electric field E, and introduce the current density $J(\epsilon)$, which is the number of electrons passing through per unit area $(\text{Ampere}/\text{m}^2)$ perpendicular to the electric field,

$$J(\epsilon)\epsilon E=F(\epsilon)\left(2m_e/m_i\right)\left(\epsilon/\lambda\right)\sqrt{2\epsilon/m_e}\,. \tag{2.52}$$

The mobility equation used for the electric current density [193] is

$$J(\epsilon)eE=\left[\begin{array}{r}-\lambda/3\sqrt{2\epsilon/m_e}\,\partial F(\epsilon)/\partial x-\lambda eE/3\sqrt{2\epsilon/m_e}\,\partial F(\epsilon)/\partial\epsilon\\ +\lambda eE/3m_e\sqrt{m_e/2\epsilon F(\epsilon)}\end{array}\right]. \tag{2.53}$$

The energy distribution is independent of the position x, and so the first term on the right of equation (102) equates to zero. Accordingly, equations (2.52) and (2.53) give an expression for $F(\epsilon)$ that is found from the solution to the differential equation [193]:

$$dF(\epsilon)/d\epsilon=F(\epsilon)\left(1/2\epsilon-6m_e\epsilon/m_i\lambda^2e^2E^2\right). \tag{2.54}$$

Using an undetermined coefficient α, an expression for $F(\epsilon)$ is found from the solution to equation (2.54):

$$F(\epsilon)=\alpha\sqrt{\epsilon}\exp\left[3m_e\epsilon^2/m_i\lambda^2e^2E^2\right]. \tag{2.55}$$

Druyvesteyn [193] simplified this equation, leaving in terms for the average energy and eliminating the electric field.

$$F(\epsilon)=\alpha\sqrt{\epsilon}\exp\left[-0.55\left(\epsilon^2/\langle\epsilon\rangle^2\right)\right]. \tag{2.56}$$

Equation (2.56) can be written in terms of the electron temperature and number density by substituting $T_e=2/3\langle\epsilon\rangle=\int_o^\infty\epsilon F(\epsilon)d\epsilon$ and $n_e=\int_o^\infty F(\epsilon)d\epsilon$ (as in basic kinetic theory [196]) as follows:

$$F(\epsilon)=\alpha\sqrt{\epsilon}\exp\left[-0.243\left(\epsilon/k_BT_e\right)^2\right]. \tag{2.57}$$

A specific expression for the Druyvesteyn EED can then be written as [197]

$$f_D(\epsilon) = \left(0.5648 n_e \sqrt{\epsilon} \Big/ \left(k_B T_e\right)^{3/2}\right) \exp\left[-0.243\left(\epsilon / k_B T_e\right)^2\right]. \tag{2.58}$$

The Maxwellian energy distribution and the Druyvesteyn distribution have different powers of energy in the exponential argument: ϵ in the former and ϵ^2 in the latter. Figure 2.5 is a graphical comparison of the Maxwellian and Druyvesteyn distributions.

The Druyvesteyn distribution is specific to electrons interacting with neutrals of comparatively low energy, where electron-neutral collisions behave as hard-sphere collisions [7]. Therefore, it is possible that a Druyvesteyn distribution may be found in low energy plasmas. However, the electric field must be high enough that the electron temperature remains significantly higher than the ion temperature. This is a condition that exists in electrostatic thrusters, where the ratio of electron temperature to ion temperature is typically 10 or more [34].

2.8.2 Two-Temperature Maxwellian Method

The two-temperature Maxwellian (bi-Maxwellian) is an anisotropic distribution model depending on physical orientation. This method has an unstable and non-equilibrium velocity distribution where there is temperature anisotropy, meaning that the temperature characterizing the movement of the particles in one direction is different from that in another direction [198]. The coordinate system usually used consists of the parallel and perpendicular velocities with respect to a background magnetic field. A stable two-dimensional distribution (i.e., one that is Maxwellian in both the parallel and perpendicular directions) is independent of the direction [198].

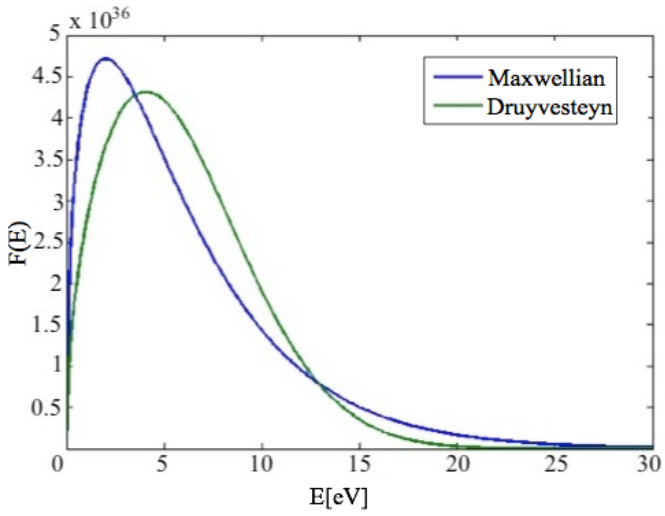

Figure 2.5: Comparison of Maxwellian and Druyvesteyn distributions, plotted for $T_e = 1\text{eV}$.

In many laboratory plasmas, the electrons are confined by a positive plasma potential V_p. Electrons with energy below eV_p may be confined for many periods of self-collision, and the distribution function for these electrons will become nearly Maxwellian [199]. Electrons with energy above eV_p will have a shorter confinement time and will not have equilibrated. Thus, their energy distribution will be determined by the mechanism that created them (e.g., secondary emission from the chamber surfaces) [199]. The slope of the combined distributions is likely to be discontinuous at energy $-eV_p$, where V_p is the plasma potential and q is the absolute value of the charge [199]. For these reasons, the methods of analysis are tested using a two-temperature distribution [199]:

$$f(\upsilon) = n\left(m/2\pi T_1\right)^{3/2} \exp\left(-m\upsilon^2/2T_1\right), \qquad (1/2)m\upsilon^2 < \epsilon_t$$
$$f(\upsilon) = \left[n\left(m/2\pi T_2\right)^{3/2} \exp((-m\upsilon^2/2T_2) + \epsilon_t/T_2 + \epsilon_t/T_1) \right], \qquad (1/2)m\upsilon^2 \geq \epsilon_t \qquad (2.59)$$

where ϵ_t is the energy at which the slope of the distribution changes. As a consequence of the small number of particles in the high-temperature tail of the velocity distribution, the number density is negligibly larger than n. The corresponding speed distribution $(\rho(\upsilon))$ is [199]

$$\rho(\upsilon) = 4\pi\upsilon^2 f(\upsilon). \qquad (2.60)$$

The temperatures associated with the two-temperature distribution are most easily found from the slopes of a semi-logarithmic plot of $f(\upsilon) = \rho(\upsilon)/4\pi\upsilon^2$ [199]. In all cases, the recovered distribution function is nearly identical to the starting distribution [199]. For the EEDF and for the electron density distribution affected by the ambipolar potential, one can obtain [200]

$$f(\epsilon) = A\left[(1-\beta_H)\exp\left\{\left[eV-\epsilon\right]/k_B T_1\right\} + \beta_H \exp\left\{\left[eV-\epsilon\right]/k_B T_2\right\} \right], \qquad (2.61)$$

where T_1 and T_2 represent the temperature of low and high energy electrons of the bi-Maxwellian distribution corresponding to the different velocity distributions that occur along two different axis in the plasma. Non-Maxwellian distribution functions do not represent electrons of the same temperature. Instead, these are represented by their average energy $\langle\epsilon\rangle$, which can be obtained by the total energy divided by the electron density ($3/2k_B T_{eff}$ can be used for non-

Maxwellian plasmas) [201]. In the above equation, ϵ is the electron kinetic energy, β_H is the fraction of hot electrons at the mid-plane, $\beta_H = N_2/N_o$, and A is the normalization constant, $A = (m_e/2\pi k_B T_e)^{3/2}$ [201]; N_2 is the density distribution of hot electrons. From this, the electron number density [200] is found using

$$N_e(x) = A\int_o^\infty \epsilon^{1/2} f(\epsilon) d\epsilon \left[\begin{array}{c} N_o(1-\beta_H)\exp(eV/k_B T_1) \\ +N_o\beta_H \exp(eV/k_B T_2) \end{array} \right]. \quad (2.62)$$

Figure 2.6 illustrates the comparison of the Druyvesteyn and two-temperature Maxwellian distributions. The two-temperature Maxwellian method was implemented similar to the Druyvesteyn distribution by deriving the integral distribution formula. Note that, depending on the electrons' energy, two different values could be assigned to the variable in the code [200].

2.8.3 Significance of Non-Maxwellian Distributions: Empirical Data

Experiments involving non-equilibrium distributions have been conducted in a vacuum test facility [202, 203]. For these experiments, argon was used for the majority of testing, and a Langmuir probe was used to collect data to determine the EEDF. The results were used to compare non-Maxwellian to Maxwellian distributions for non-equilibrium plasma applications. Results from the Langmuir probe experiment are presented for the Druyvesteyn method for the non-Maxwellian distribution [202]. As the distribution functions obtained from a single I-V curve [202] were not clear enough to make a comparison with Maxwellian and non-Maxwellian distributions, the EEDF was obtained to preserve important features in the curve.

Figure 2.7 [202] (also Table 2.1) summarizes the results of the analysis considering three variables (n_e, T_e and V_p) for two cases, where Case 2 has higher electron temperature ($T_e = 3.9eV$) and number density ($1.7x10^{16} m^{-3}$) but lower plasma potential (20.2V) compared with Case 1 ($n_e = 1.6x10^{16} m^{-3}$, $T_e = 3.7eV$, $V_p = 21.2V$). Exemplifying the non-Maxwellian nature of the plasmas, the results show that, for both cases, the Druyvesteyn distribution provides a better description of the EEDF than the Maxwellian distribution. The Druyvesteyn distribution places more weight on higher electron energies than the Maxwellian; the reason for the discrepancy in electron temperature is that Langmuir theory [43, 72, 152] assumes a Maxwellian distribution. The average energy, which is directly proportional to the temperature

for a Druyvesteyn distribution, is slightly higher than that of the Maxwellian distribution. Langmuir theory might have provided more accurate electron temperatures if it were derived again assuming a Druyvesteyn distribution [41, 196].

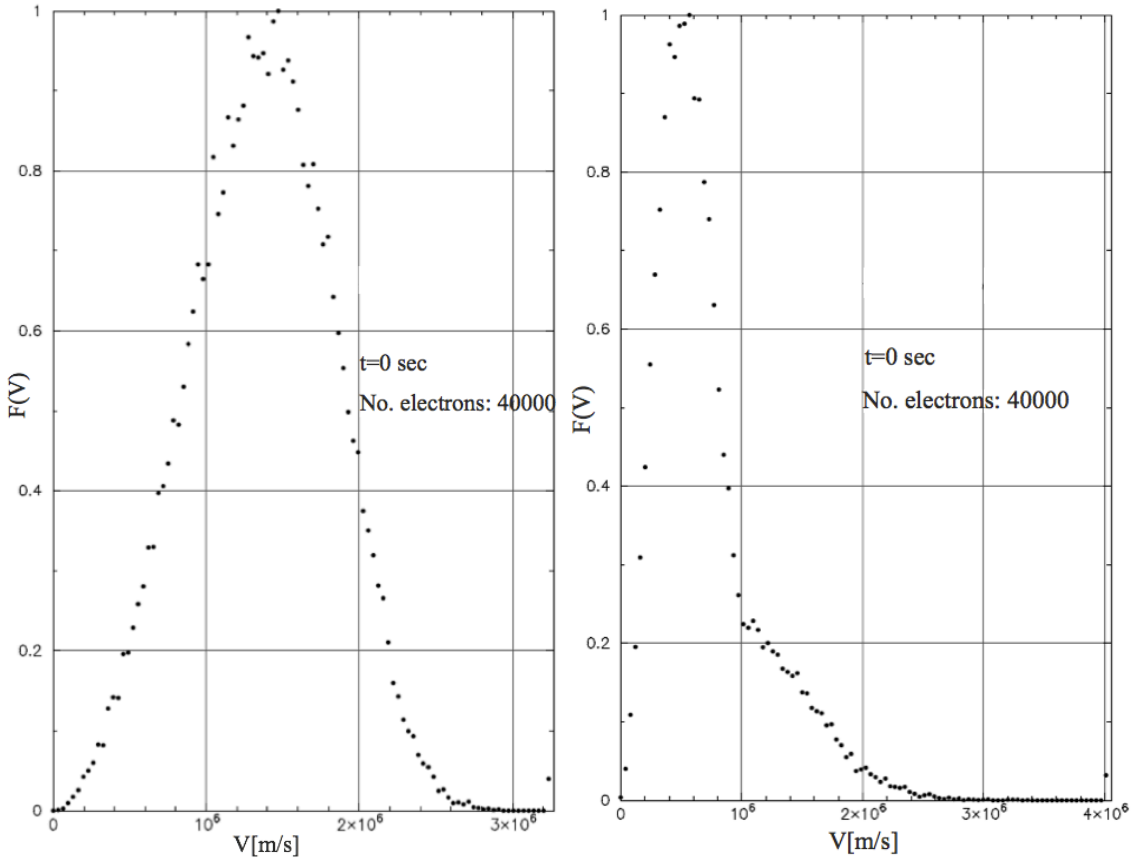

Figure 2.6: The comparison of Druyvesteyn (left) and two-temperature Maxwellian (right) velocity distributions during the particle injection (distributions are normalized to one).

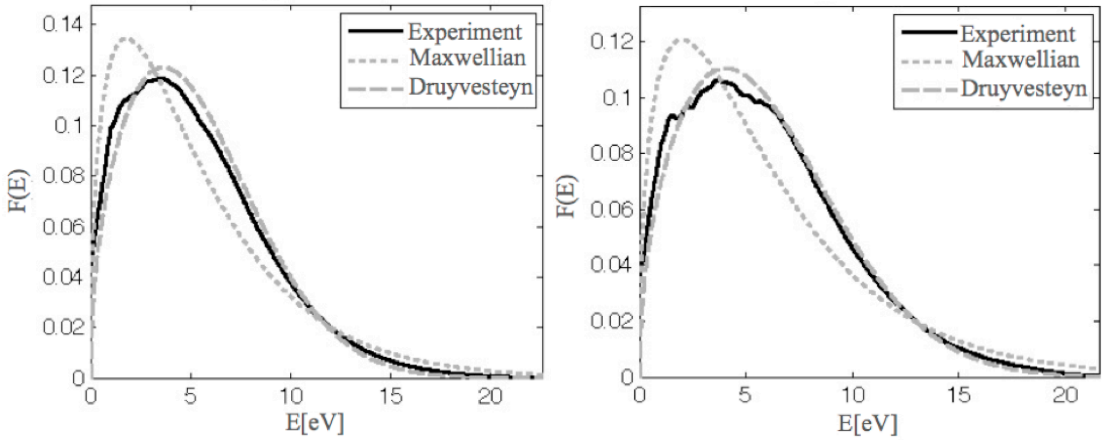

Figure 2.7: EEDF on Axis for Case 1 (top): $n_e = 1.6x10^{16} m^{-3}$, $T_e = 3.7eV$, $V_p = 21.2V$; EEDF on Axis for Case 2 (bottom): $n_e = 1.7x10^{16} m^{-3}$, $T_e = 3.9eV$, $V_p = 20.2V$ [202]. Distributions are averaged.

Table 2.1: Case analysis for plasmas of different characteristics.

Variables	Case 1	Case 2
n_e [m^{-3}]	$1.6x10^{16}$	$1.7x10^{16}$
T_e [eV]	3.7	3.9
V_p [V]	21.2	20.2

Chapter 3

Numerical Methods

There are two fundamental types of numerical approaches for modeling the gas and plasma flows: (1) continuum or fluid methods, and (2) kinetic methods. The choice of method for modeling a specific thruster should be dictated by the physical characteristics of the flow in the device, and by the level of accuracy required from the simulation. There is a wide range of plasma conditions, which means that different methods and computer codes must be developed for each of these conditions. The most important numerical methods that are employed are reviewed in this chapter.

3.1 Review of the Continuum/Fluid Methods

The gas or plasma may be considered as continuum or fluid when length scales associated with the most important physical phenomena (e.g., spatial gradients of flow properties, distance between collisions, charge separation distance) are small in comparison to the size of the plasma environment. In general, these conditions are encountered in the relatively high-density devices such as resistojets and arcjets [204-207]. Continuum methods are usually based on a set of partial differential equations describing conservation of mass (or charge), momentum, and energy. For example, the viscous flow encountered in a resistojet may be modelled using the following equation set (often referred to as the Navier-Stokes equations) [208, 209]:

$$\text{Continuity:} \quad \frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} = S_C \tag{3.1}$$

$$\text{Momentum:} \quad \frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ik}}{\partial x_k} + \rho F_i + S_{M_i} \tag{3.2}$$

$$\text{Energy:} \quad \frac{\partial \rho e}{\partial t} + \frac{\partial \rho u_j e}{\partial x_j} = \frac{\partial p}{\partial t} + \frac{\partial}{\partial x_k}\left[u_j \tau_{jk} - q_k \right] + \rho F_i u_i + S_E , \tag{3.3}$$

where ρ is the mass density, u_i is a velocity component, p is the pressure, e is the total specific internal energy, τ_{ik} and τ_{jk} are shear stress tensors, q_k is the heat flux vector, F_i is an external body force, and S_C, S_{M_i}, and S_E are source terms caused by chemical reactions in the mass, momentum, and energy equations, respectively.

First order models for the shear stress and heat flux vector use temperature dependent transport coefficients (e.g., viscosity and thermal conductivity). For plasma flows, additional equations supplement the above set. These additional equations are highly specific to the particular plasma environment. In general, it is necessary to include an additional set of conservation equations for the electrons and to add terms to the heavy particle momentum equation to account for the electromagnetic fields and friction with the electrons. These types of equation can be solved using a wide variety of numerical approaches. In finite difference formulations, each partial derivative is evaluated to some order using Taylor expansions. In finite volume formulations, the integral forms of the conservation equations are solved. The issues regarding these formulations are discussed in a variety of texts [208, 209].

3.2 Review of the Kinetic Methods

The flows in many EP systems fall into the rarefied regime in which collision and plasma length scales are similar to or even larger than the size of the thruster. In such cases, the flow is not well represented by a continuum formulation, and instead, a molecular, kinetic approach must be undertaken. Let us discuss separately the numerical simulation of rarefied gas-dynamic and plasma-dynamic effects.

3.2.1 Gas Dynamics

The rarefied flow regime is defined to exist for Knudsen numbers above 0.01 [80]. The continuum approach fails under rarefied conditions because there is an insufficient rate of collisions to maintain the velocity distribution functions anywhere near to the small departures from the Maxwellian equilibrium form implicitly assumed in continuum formulations. While the Boltzmann equation of dilute gas dynamics should is solved under high Knudsen number conditions, this turns out to be a formidable mathematical and computational task. The Direct MC (DMC) [80] method is a highly successful numerical technique for simulating rarefied gas flow.

In the Direct Simulation MC (DSMC) technique, a large number of model particles are simulated. Each model particle represents a much larger number of real atoms or molecules. The model particles possess unique velocity components. A key aspect of the DSMC technique is that particle motion is decoupled from particle collisions. This is only reasonable if the time step employed in the simulation is significantly smaller than the mean time between collisions. During the each iteration of the DSMC algorithm, all the particles are first moved according to the product of

the time step and their individual velocity vectors. Any interactions between the particles and boundaries are then processed. The particles are next collected into cells for computation of collisions. Within each cell, the positions of the particles are ignored. The particles are paired up at random and collision probabilities are computed for each pair to determine whether a collision will take place. This statistical approach is only accurate when the size of the computational cell is of the order of a mean free path.

The collision probability is proportional to the product of the collision cross section and the relative velocity of the colliding particles. Many different types of collision phenomena can be simulated using the DSMC technique including momentum exchange, charge exchange, internal energy exchange, and chemical reactions. Average flow properties such as density, velocity, and pressure are obtained by time averaging of the particle properties. Unsteady flows may be simulated by averaging over small periods of time, or by ensemble averaging. A related method, the MC collision technique [210], is also often used in EP simulations. This approach is useful in hybrid particle-fluid simulations to determine the effects of collisions of electrons (modeled as a fluid) on the heavy species that are represented as particles.

3.2.2 Plasma Dynamics

Plasma cannot be described accurately by a continuum formulation under high Knudsen number conditions. For EP devices, the most important behavior to simulate is the plasma dynamic effects on the momentum of the heavy ions since these are the main source of thrust. The PIC method [79] is a particle technique for simulating plasmas. In most EP systems, ions are treated as particles using PIC and the electrons are treated using a continuum, fluid approach. This approach is usually justified, because the time scales associated with the much lighter electrons are orders of magnitude smaller than those for the heavy ions. Even if the velocity distribution function of the electrons is not Maxwellian, it is usually compact allowing meaningful interpretation of continuum quantities such as temperature and pressure.

Plasma physics has been described as the art of approximation, and this is an important aspect of applying the PIC method to simulate the special plasma environment such as inside the EP thruster. In this assumption charge quasi-neutrality is employed. This approach allows the spatial distribution of the ions to give the spatial distribution of the electrons. This is a good assumption provided the Debye length is much smaller than the length scale of the thruster. For a typical plasma environment, electron temperature of several electron volts, the assumption of quasi-neutrality is good down to plasma densities of about $10^{14}\mathrm{m}^{-3}$, which is much lower than the values encountered in most EP thrusters (see values listed in Table 1.1) except for the inter-grid regions of ion thrusters. A variety of approximations of

the electron momentum and energy equations are invoked for different types of plasmas of EP thrusters. The electric field is usually obtained from spatial differentiation of the plasma potential that in turn is derived from the electron momentum equation. This field is used to change the velocity vector of each ion particle in the PIC simulation according to Newton's 2nd Law:

$$m_i \frac{d\vec{\upsilon}}{dt} = q_i \vec{E} \tag{3.4}$$

where m_i is the ion mass, $\vec{\upsilon}$ is the velocity vector, q_i is the ion charge, and $\vec{E}$ is the electric field. The steps in the PIC algorithm are to obtain the plasma density from the spatial distribution of the ions, calculate the electric field, accelerate the ions, and finally move the ions. The MC calculation of collisions is very easily introduced into the PIC algorithm after the movement step. The following sections employ the PIC and MC collision techniques for simulating the plasma dynamics [211], and establishes a theoretical framework for the algorithms used in PROBEPIC, which is an enhanced simulation code for modeling both Maxwellian and non-Maxwellian plasma dynamics.

3.3 The Particle-In-Cell (PIC) Technique

Methods for adapting PIC kinetic plasma calculations are highly valuable in the study of multiple-length scale quandaries [188]. Usually, multiple-length scale problems include small regions with increased gradients embedded in large systems. For these conditions, computational efficiency can be achieved most successfully by focusing attention on the regions of interest. PIC simulations are well suited to such systems and contain three main components: particles, grids and boundaries. PIC codes track the motion of both ions and electrons (neutral particles can be included in the simulation when applicable), from which the particle motion, potential, and electric fields are calculated self-consistently [79]. The method simulates the motion of plasma particles and calculates all macro-quantities (such as number density and current density) from the positions and the velocities of these particles [79].

In PIC, instead of solving equations of motion and Maxwellian equations (e.g., Coulomb's law, Poisson's equation) in continuous space and time, the physical volume is divided into cells by lines that run parallel to the boundaries. The intersections of these lines define a set of points called mesh points or grid points. In these points, the charge fields are calculated and are moved relative to the particles associated with these points [79]. The space coordinates of the charge fields are continuous, and they can occupy positions anywhere within the mesh. The local charge density is simply the number of particles in an individual cell divided by the volume of the same cell. Particles are shielded from other

particles beyond the Debye length, which determines the grid spacing [79]. The general concept is illustrated in Figure 3.1.

Mesh points are also used in standard numerical solutions of differential equations. The details of close encounters of particles are not important. The electrical field is not obtained by summing the contributions of all particles, but rather it is obtained by calculating the local charge density, potential and then the electrical field. The mesh sets the lower limit of the spatial resolution of particle-particle and particle-field interactions [79]. It is ideal to keep the length of individual spatial cells smaller than the Debye length. If a spatial cell is larger, then the effects are not seen in the calculated electrical fields [79].

3.1.1 Iteration Steps in the Simulation

A PIC program contains some modules that are used to move the particles through the phase space. These modules include the particle mover, the charge weighter, the field solver, and the force weighter [79]. A typical run is divided into several periods and may include thousands of time steps. The order of these modules in the PIC code is shown in Figure 3.2. Once initial conditions, particle positions and velocities are read by the system, the particles are weighted to a computational grid at time t to determine the charge density and other particle moments. Next, the electric potential and field are calculated on the grid. Then, both electric and (pre-computed) magnetic fields are weighted back to the particles, which are time-stepped forward to $t+1$ according to a leapfrog scheme [79]. Then, the cycle begins again. Convergence of the simulation is determined by the rate at which the parameters of interest change.

The size of the time step is chosen based on the physical phenomena of interest. If the interest is in the small mass particles, a shorter time step may be required because the plasma frequency is larger than for plasma with heavier particles, which allows a longer time step for a shorter simulation time. Thus, the choice of numerical method is very important because it has a substantial impact on the time required to run the simulation. Computer memory is another important consideration when choosing the numerical method; for example, high-order methods (e.g., Runge-Kutta) require more memory [79]. More information regarding the PIC technique can be found in [79].

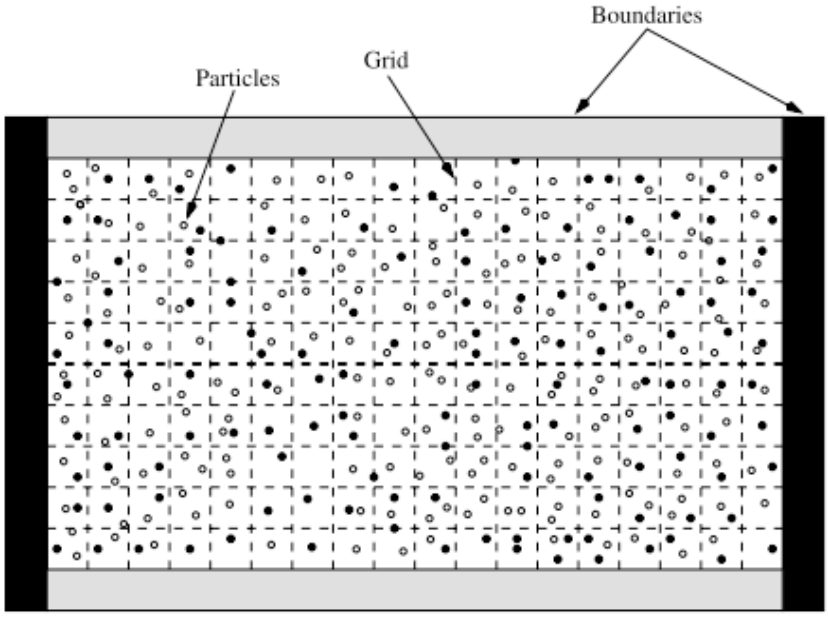

Figure 3.1: A view of the objects in a typical PIC code [72, 152].

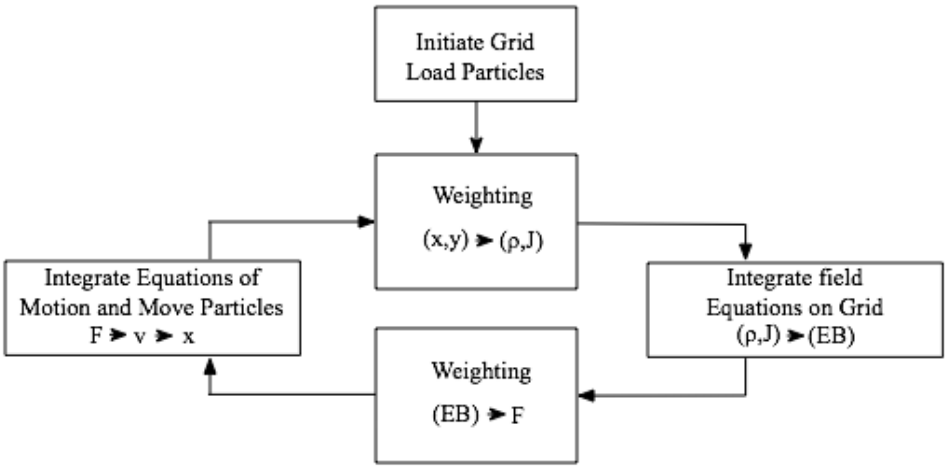

Figure 3.2: A typical cycle within a single time step (dt) in a PIC particle simulation program.

3.1.2 Boundary and Initial Conditions

General considerations at the beginning of the simulation process include the size of the computational domain, the number of grid points, the number of particles and the positions of objects (conductors, insulators, etc.) relative to the boundaries [79]. The size of the computational domain must be large enough to encompass the interaction that is to be observed. As the perturbations of the plasma result in electron oscillations on the order of a Debye length [212], the grid must contain at least two grid points per Debye length to resolve these oscillations.

The number of particles in a simulation determines its statistical accuracy. The influence of the self-force (resulting from the charge weighting) can be excluded progressively as more particles are added to the system. To obtain acceptable results, ten or more particles should be introduced for each computational grid cell (e.g., a 10x10 computational grid would require 1,000 or more particles). Increasing the number of heavy super-particles results in an increase in the frequency of collisions.

Objects should not be placed near the boundaries. If current collection on an electrode is being simulated, the electrode should not be placed near a computational boundary where particles are introduced (fluxed across) to the system [213]. Instead, the electrode should be placed near the center of the computational domain so that the particles can have a chance to thermalize and so that the effects of the (non-physical) computational boundary have time to relax.

The boundary conditions determine how the plasma interacts with its surroundings, and they include particle fluxing across boundaries of the computational domain, surface interactions or material interactions between the plasma and objects in the computational domain, and electromagnetic field boundary conditions [79]. Initializing a PIC code involves careful determination of the initial distribution function for each species, $f_\alpha(\vec{x},\vec{v},\vec{t}=0)$, including any initial perturbation, as well as distributing the particles throughout the computational domain to initiate the simulation.

3.2 Addition of the Monte Carlo (MC) Technique to PIC Codes

The MC method is a technique that involves using random numbers and probability to solve problems. Through the MC method, many approximate solutions for a variety of mathematical problems can be performed via statistical sampling experiments on a computer. The method applies to problems with absolutely no probabilistic content, as well as to those with inherent probabilistic structure [214].

PIC codes involve deterministic classical mechanics, which generally move all particles simultaneously using the same time step. The part left to chance is usually limited to choosing the initial velocities and positions and the injected particles. The objective of a simulation is to seek collective effects due to self and applied fields. The Δt values of the time steps are most commonly uniform, but some exceptions include multi-scale codes [215] (moving the slower particles less frequently than the faster particles, as well as allowing a non-uniform grid) and electron sub-cycling [216] (moving the electrons more frequently than the ions).

In orbit averaging [217], the particles are all advanced at small time steps (allowing for resolution of good cyclotron orbits), but the fields are advanced less frequently, using averaged orbit data. Because the number of calculations per particle is not reduced, the gain in speed comes from reducing the number of particles required, as the averaged contribution of each particle can substitute for many conventionally handled particles.

As MC codes are probabilistic in nature, they mostly seek collisional effects in relatively weak electric fields. For example, a given charged particle is known by its (total) kinetic energy E and its velocity relative to some target particles. This information produces a collision frequency $(\nu_{coll} = n_{tar}\sigma(E)\upsilon_{rel})$ and a probability that a collision will occur. Where σ is reaction cross section, n_{tar} is the number density of target particles and υ_{rel} is the velocity relative to target particles. There are several PIC-MCC code applications, and more information regarding this technique can be obtained from [210].

In the simulation, electron diffusion across magnetic field lines occurs primarily through a series of large-angle collisions. Because electrons are much faster than neutrals, the MCC method is appropriate for electron-neutral scattering. The MCC method may also be used to model ion-neutral scattering, electron-ion and electron-electron scattering.

3.3 PROBEPIC Simulation Code

PROBEPIC is a notable example of a PIC simulation code, and it has expanded on the work of previous treatments [72, 152]. The PROBEPIC [72, 152] code was originally created to investigate the OML current collection theory describing the thick sheath limit in a plasma with a Maxwell-Boltzmann particle velocity distribution. PROBEPIC simulates the kinetic processes of ions and electrons in the vicinity of finite length Langmuir probes [152]. This code was later updated to improve its accuracy and operation by exploiting the capabilities of modern computers [152]. It can now be used to

simulate the space charge limited electron emitting sheaths, as it accurately addresses the loss of plasma electrons to the boundary and the velocity distribution of the emitted electrons [185]. Currently, the most recent version of PROBEPIC can produce I-V characteristics for both OML current collection (thick sheath theory) and SL current collection (thin sheath theory) [185].

PROBEPIC was originally developed to simulate plasmas that have a Maxwellian velocity distribution [72, 152]. The distribution is uniform and isotropic in configuration space and also isotropic in velocity space, which means that the position and the velocity of a single particle are designated by its three coordinates, x, y and z. However, the energy distribution of a plasma is not always described by the Maxwellian distribution theory. Using the same equations to simulate non-Maxwellian plasmas creates enormous errors in the plasma diagnostics; this is complicated by the fact that all probe geometries used are not equally suited to characterizing the properties of non-equilibrium plasmas [202, 218], such as ionospheric electron temperature and density, or for determining the distribution function in the presence of non-Maxwellian background electrons.

PROBEPIC is a 2D/3V PIC code that designates that the electric field has two components, radial and axial, and that the particles are free to move in three dimensions [152]; 2D represents the physical space and 3D represents the velocity space. PROBEPIC uses a standard leapfrog mover [72, 79, 152], which moves the particles in Cartesian coordinates, and the new particle position is transformed back into cylindrical coordinates. The field solver, charge weighting, etc. are carried out in cylindrical coordinates. This technique was implemented to avoid the singularity at $r = 0$ [72, 79, 152, 185]. The computational domain in PROBEPIC consists of a cylindrical region containing a cylindrical Langmuir probe, as shown in Figure 3.3.

The probe consists of a cylindrical conducting wire partially covered by an alumina insulator [72, 152, 158]. The computational domain provides enough room for a plasma sheath to naturally form around the probe, which means that the plasma sheath is generated by the interaction of the plasma with the boundary material. In PROBEPIC, it is surmised that the plasma is axisymmetric, permitting the calculation only for the fields in the r and z directions on the half plane [152]. The computational grid plane can be divided into two sections: interior grid points and surface grid points. Here, the interior grid points lie within the plasma, probe conductor and insulator, whereas the surface grid points lie at the edges of the computational domain and on the surface of the conductor and the insulator of the probe [219].

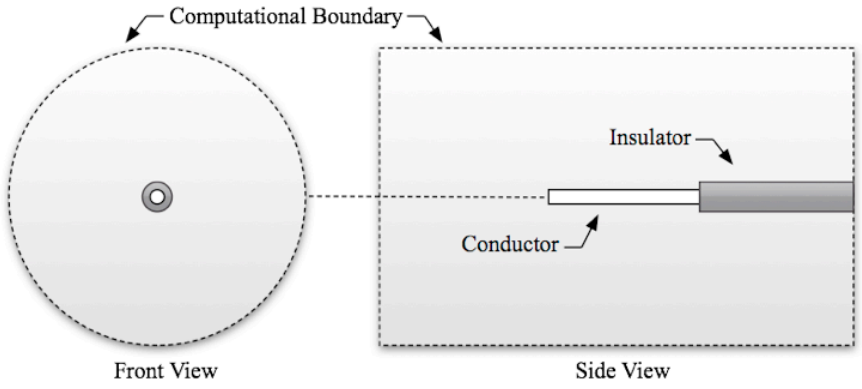

Figure 3.3: Schematic of the computational domain layout in PROBEPIC.

During a cycle, the region is divided by a suitably sized grid. According to the physical parameters of the plasma, the grid is then loaded with a proper distribution of super-particles, each of which represents a set number of real particles. The next step involves the assignment of the charge densities to the grid, where a weighting is used to distribute a proportion of each particle's charge to the surrounding grid points. The interpolation used for this depends on the system's tolerance to numerical noise and on the accuracy required *versus* considerations for the time required and computational resources available. Although the nearest grid point method [112] is computationally efficient, it does not yield satisfactory macroscopic results for PROBEPIC. Therefore, first order linear particle weighting is used. This allows the particle's charge to be distributed not only to one adjacent grid-point, but also to its two nearest neighbors [34, 41, 112].

Next, the field equations are integrated across the grid using an appropriate method. For a one-dimensional electrostatic model without magnetic fields, the Poisson equation solved in finite-difference form is sufficient. However, for higher dimensions with electromagnetic fields, the use of the full set of Maxwell's equations is required [34, 112]. PROBEPIC makes direct use of the Gaussian (normal) distribution to eliminate the need for any transformation [34], yielding a simple, intuitive algorithm. The electric field boundary conditions used for the various surfaces in PROBEPIC are further explained by Aktas [34]. The fields are then applied back onto each particle using the same weighting scheme used to distribute charges to the grid points, and the equations of motion are integrated to determine the force on each particle [34, 112].

The particles are moved in Cartesian coordinates, and the new particle position is subsequently transformed back into cylindrical coordinates, as the calculations in some modules (the field solver, charge weighter, *etc.*) are carried out in

cylindrical coordinates. Finally, the particle mover updates the position and velocity of each super-particle given its previous state and the electric field intensity at the particle position. The whole process is then repeated for another time-step [34, 112]. In PROBEPIC, the electron time-step (dt_e) and the ion time-step (dt_i) are defined separately [34]. The electron time-step is much smaller than the ion-time-step because of the high mean electron velocity.

The modules were developed independent of any particular problem, and the simulation code uses physics in the form of boundary conditions. These conditions are dependent upon the size of the computational domain, the number of grid-points, the number of particles and the positions of objects relative to the boundaries [34, 41, 112]. The initial plasma parameters in PROBEPIC (e.g., temperature and number density) are set in a subprogram [34] and are user-defined conditions, where quantities such as a plasma frequency and Debye length are calculated for use in later subroutines. The computational grid of PROBEPIC involves the assignment of particle position, charge density, radial and axial electric field intensity, and electrostatic potential for each grid point [34]. PROBEPIC uses a non-uniform grid [34], and its density is increased in the vicinity of the probe to resolve the field within the sheath region [34]. The grid dimensions are also set in a subprogram [34], and this information is then used to set the x and y coordinates of the grid-points. To flux particles into the computational domain, PROBEPIC determines how many particles enter the computational domain in a given time step, as well as the angular and speed distribution of these particles [34, 41].

PROBEPIC includes a (pseudo) random number generator (an important element of any PIC/MC simulations [34]), which is used to calculate random particle speeds. This type of generator is well suited for massively parallel simulations on graphics processing unit (GPU) that performs rapid mathematical calculations. An algorithm inside the code allows one to send it an array of any size, which upon return will be filled with random uniform deviates. PROBEPIC can store many values for later use rather than calculating them each time they are needed. The algorithm generates random particle speeds for particles in a quiescent plasma according to the Boltzmann H-theorem, reproducing a Maxwellian distribution over many calls. A mathematical description of the generation of a set of random speeds with a Maxwellian distribution is given in [41].

Chapter 4

Simulations and Results

4.1 Implementation of Theoretical Approaches

There are several practical considerations to make when trying to apply the simple Langmuir probe (LP) theory to experimental data. When any of the classical LP assumptions are violated or when the experimental setup is flawed, departures from the nominal probe characteristic occur. To make the problem more tractable, some approximate solutions were proposed for the two limiting cases [72] (see Section 2.5). These include the thin sheath case, where the plasma-sheath boundary can be identified from the plasma side as a point of electric field singularity [72], and the thick sheath case, where the boundary is defined as the point of vanishing electric field [152].

Both of these approximations are reasonable for many plasma systems. For example, thicker sheaths tend to induce more distortions in particle trajectories, as their shape tends to be more affected by surrounding objects, and incoming particles have more possibilities for crossing isopotential surfaces that are not parallel to the surface at the point of impact [220]. In that case, sheath distortions would be most significant in low density, high temperature plasmas [220]; therefore, the thin sheath approximation would be the less applicable case. On the contrary, the thin sheath approximation would be the most accurate for high density, low temperature plasmas [220]. The aim of this section is to extend the simulation to enable the production of I-V traces that describe both regimes.

4.1.1 PROBEPIC for the Thin Sheath Theory

For a very thin sheath, where the Debye length is smaller than the probe radius, space-charge limited current collection is assumed, where any charged particles crossing the sheath are collected [72, 152]. As the name suggests, the thin sheath model assumes that the sheath thickness is much smaller than the characteristic length scale of other surrounding structures. In that case, the change in position as a particle crosses the sheath is negligible, and the only effect of the sheath's electric field is to accelerate particles in the direction perpendicular to the surface at the point of impact [220]. To enable the modeling of the SL current collection, the PROBEPIC code was modified to allow the adjustment of the simulation parameters listed in Table 4.1. The updated PROBEPIC enables the user to specify a range of probe voltages

over which the I-V characteristic will be determined without further user intervention. The code also was modified so that multi-core parallel processing could be exploited, improving the efficiency 20-fold.

PROBEPIC was created on the original assumption that a particle was considered to hit and be collected by the probe only when it fell exactly in a cell on the probe [152]. When this occurs, probe current is generated. However, with this assumption, the high-speed electrons and ions that have trajectories intersecting the probe but not falling precisely in a probe cell at the end of a time-step do not contribute to the probe current. This issue was resolved by extending the particle trajectory in a straight line to determine whether it intersects with the cylinder representing the probe during a given time step. A screen shot from the PROBEPIC animation window showing the electron and ion trajectories, the locations where the particles are injected, and where they hit the probe is presented in Figure 4.1. The figure shows the Langmuir probe in full domain, illustrating the ion and electron trajectories: purple dots are for the electrons, and green dots are for the ions. The new particles that are injected into the domain and the particles that hit the probe are also shown. Note that only a few of the approximately 224,000 particles in the simulation are shown in this picture.

Table 4.1 Probepic user-controlled input parameters.

Parameters	Desscription and Units
0.1	Percent of particles displayed in animation; 0.1,...,100.0; 0.0 means NO animation
0.0035	Probe radius [m]
57.142	Probe Aspect Ratio [length/radius]
3	Number of cells per probe radius
112,000	Initial number of ions in domain
5.6	Number of injected ions from BACK at each time step
5.6	Number of injected ions from FRONT at each time step
112,000	Number of injected ions from SIDE at each time step
112,000	Initial number of electrons in domain
14	Number of injected electrons from BACK at each time step
14	Number of injected electrons from FRONT at each time step
22.4	Number of injected electrons from SIDE at each time step
0.5	Temperature, T [eV]
1×10^{15}	Number density [cm^{-3}]
1.67×10^{-27}	Mass of ion [kg]
1.602×10^{-19}	Charge of ion [C]
9.11E-31	Mass of electron [kg]
-1.60×10^{-19}	Charge of electron [C]
0.009	Plasma frequency [sec]
0.0	Plasma potential [V]

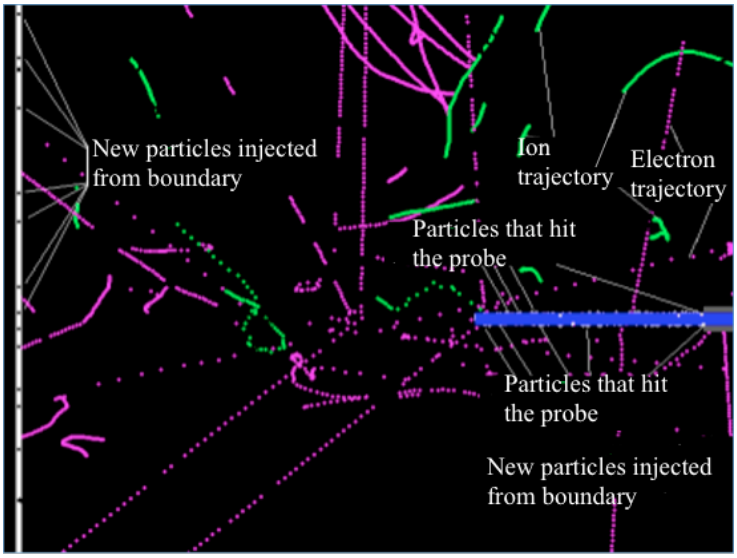

Figure 4.1: PROPEPIC screenshot for full domain, showing the electron and ion trajectories, the locations where the particles are injected, and where they hit the probe. Green represents the ion trajectories, and purple represents electron trajectories.

In thin sheath theory, a Langmuir probe is characterized by a high ratio of probe radius to Debye length [72, 152]. This means that, to more closely approximate the thin sheath properties, one can either increase the probe radius or decrease the Debye length. According to Chen, the ratio of probe radius to Debye length (r_p/λ_D) should be greater than 10:1 to ensure the collection of charged particles entering the sheath [8, 181, 183]. Initially the ratio was adjusted by decreasing the Debye length and keeping the probe radius fixed. This it led to slightly flatter electron saturation regions [8, 221], as expected for the thin sheath case [8, 221]. However, it was difficult to obtain a sufficiently small Debye length, as the probe radius was already very small. Also, a higher T/n ratio increased the super-particle counts in the domain, resulting in longer simulation runtimes. For these reasons, attempts to approximate thin sheath properties by decreasing the Debye length alone were abandoned. Subsequently, the ratio was altered by varying the probe radius.

For larger probe sizes, a greater computational domain and more super-particles were required, increasing the runtime. Figure 4.2 shows the simulation results for a probe radius of 3.5mm and a domain radius of 7mm (roughly equivalent to 10s over the probe surface [8, 221]). The probe length was then reduced to 2mm to reduce computational domain volume. A careful coalescing process conserves the particle and mesh charge and current densities as well as the particle energy [222] to ensure the proper calculation of the $\mathbf{E}$ field. Accordingly, the number of the particles was scaled

proportionally to the change in domain volume, and the grid cell counts were scaled proportionally to the change of domain radius.

As Figure 4.2 shows, when the domain was adjusted to new probe sizes, the measured current was still much greater than that predicted by thin sheath theory. This was found to be due in part to the disabling of the insulator interaction in the original PROBEPIC code. Neglecting this process did not impact the PROBEPIC modeling for thick sheath case, as the probe radii were sufficiently small. However, when modeling thin sheath cases, disregarding this interaction means that particles can freely travel through the insulator and hit the probe conductor from the back, leading to a higher current than expected. In theory, this problem could be resolved by simply changing the velocity of particles when they are in the insulator volume. However, if the new velocity is not high enough to allow the particles to leave the insulator, they could become trapped within it. This may cause them to jump back and forth endlessly, never being able to leave the computational domain. To avoid this situation, particles were always moved to the surface of the insulator volume following a collision. In an effort to more accurately reproduce the theoretical results, the electrical field grid was aligned with the probe, and the effects of varying the grid size were investigated. It was found that making the number grid cells too high decreased the accuracy, as the number of particles in each cell was too low for statistical accuracy. However, by halving the cell size, the accuracy was increased (Figure 4.2). By lowering the temperature to 0.5eV, V_f increased to approximately -2V. As shown in Figure 4.3, the simulated results are in good agreement with those obtained theoretically.

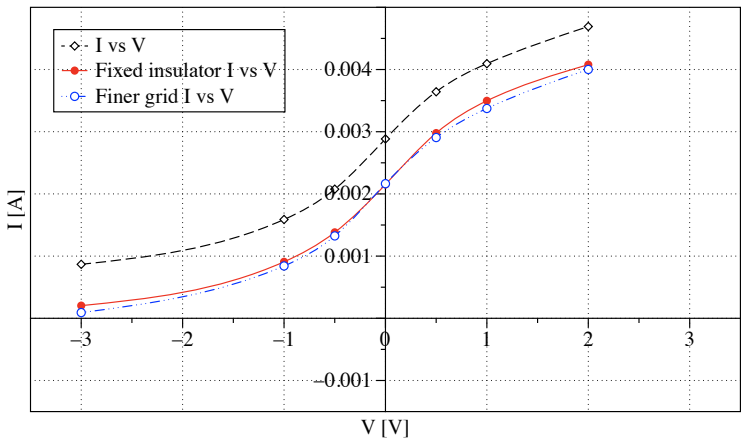

Figure 4.2: I-V trace for the thin sheath case: comparison of the results before and after the author added the insulator interaction. $T_e = 1\text{ eV}$, $n = 10^{15}\text{m}^{-3}$, $r_p = 3.5\text{ mm}$, $l_p = 2\text{mm}$.

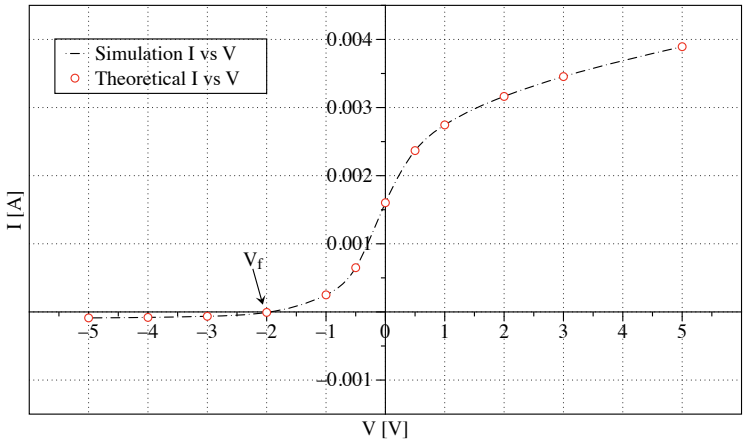

Figure 4.3: I-V trace for the thin sheath case: comparison of theoretical and simulation results for a large-probe size. $T_e = 0.5\text{eV}$, $n = 10^{15}\text{m}^{-3}$, $r_p = 3.5\text{mm}$, $l_p = 2\text{mm}$.

These results demonstrate the importance of the accurate determination of the simulation grid size. The grid cell width is generally chosen to be of the order of the Debye length [210, 223]. By setting the cell width equal to the Debye length, short-range particle interactions are reduced. However if the cells are made much larger than the Debye length, important electric field gradients can be overlooked because the fields are determined only at the grid points. Furthermore, as the cell widths define the particle size, values larger than the Debye length lead to unphysical particles. A very high grid density might provide the closest fit to the theoretical data. However, this high grid density requires more particles, and therefore is more computationally demanding.

4.1.2 PROBEPIC for the Thick Sheath Theory

For a very thick sheath, where the Debye length is larger than the probe radius, OML current collection is assumed. The OML current is the current collected by the probe when none of the undisturbed particles that are capable of reaching the probe (on the basis of energy considerations) are excluded by the intervening barriers of effective potentials [72, 152]. OML theory neglects the influence of the sheath, and instead particle orbits are simply computed using the space-charge free electric field of the probe [72, 152].

To observe what effects the changes made to PROBEPIC (particularly re-enabling the insulator collision) have on the simulation of thick sheath case, for which it was initially designed, a run with the small probe geometry was conducted. Initially, the same grid and domain size as used in the large probe scenario was applied. However, because the smaller probe occupied less of the domain volume, the overall super-particle density was low. This resulted in

smaller current measurements than expected. Accordingly, the domain and grid sizes were updated in the subsequent run based on the small-probe state [152]. The accuracy of the updated PROBEPIC was evaluated by comparing the new results with those obtained from the original PROBEPIC simulations and the analytical results. Figure 4.4 reports the updated simulation results in an I-V characteristic plot for the thick sheath case.

When the probe potential is less than the plasma potential, the PROBEPIC results show good agreement with the Langmuir theory. On the other hand, the PROBEPIC results diverge from the expected theoretical results in the electron saturation region (voltages above the plasma potential [152]). This is because Langmuir theory always produces the maximum current that could potentially occur (because an ideal, infinite-length probe is assumed) [72, 152, 187]. The probe aspect ratio may produce a significant difference in the I-V characteristic [104]. In addition, the sharp infinite-length probe tips on the conductors produce an intense electric field in the vicinity of the tip, which creates a large difference between infinite- and finite-length probes [152]. Note that, because both probe types exhibit an OML current-collection case, the resulting I-V characteristic does not have a sharp knee when the probe potential is equal to the plasma potential (as would be expected in the SL regime [152]). Rather, a very smooth curve [156] from positive to negative is evident, as is expected [152]. The code modifications also resulted in simulated currents that more closely approximate those predicted analytically in the electron saturation region of the I-V characteristic, which is the region that demonstrates the largest departure from theory.

4.2 Langmuir Probes in Non-Maxwellian Plasmas

This section presents probe simulations for non-equilibrium plasmas and the electron energy spectrum. This includes the modification of existing non-Maxwellian modeling techniques (e.g., the Druyvesteyn procedure) to improve the ability to model non-equilibrium plasma. The resulting enhanced software is used to simulate the operation of Langmuir probes in a series of validation and demonstration cases. The results of these simulations are compared with available experimental data to assess the effectiveness of the enhanced simulation tool.

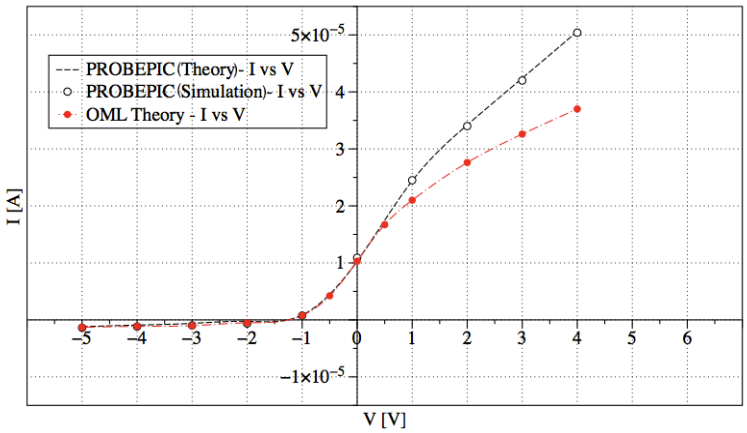

Figure 4.4: I-V trace for the thick sheath case: comparison of the results obtained for the small probe state. The conditions used for testing are $T_e = 0.5\text{eV}$, $n = 10^{15}\text{m}^{-3}$, $r_p = 0.3125\text{mm}$, and $l_p = 2.87\text{mm}$. For PROBEPIC (theoretical [152] and simulation), a finite-length probe is used, whereas the probe length is infinite for OML theory [152, 155] .

4.2.1 Application of the Druyvesteyn Method

Before applying the new method, a test was conducted to compare the Maxwellian particle distribution from PROBEPIC with a theoretical Maxwell-Boltzmann distribution. The aim of this was to confirm that the updated code was functioning correctly. To make a comparison, a test simulation was created, and a thick sheath plasma simulation was used as an initial reference point. This was done because OML theory, introduced by Langmuir for particles that have a Maxwellian distribution, is the most common experimental analysis approach. Then, the current was derived for a Maxwellian EEDF to verify that the code was error-free [185]. After determining the relevant theory for OML, the electron temperature was extracted from the slope of the I-V curve at retarding voltages in logarithmic coordinates (less than the plasma potential). This retarded part of the curve weakly depends on the probe geometry but more strongly depends on the plasma parameters [200]. The electron density is extracted from the high voltage part of this curve using OML theory [131]:

$$I_{OML} = A_p ne\sqrt{k_B T/2\pi m}\left(\left(2/\sqrt{\pi}\right)\sqrt{\pi} + \exp(\eta)\text{Erfc}(\eta)\right), \tag{4.1}$$

where $\eta = eV/k_B T$ and

$$I_o = A = A_p ne\sqrt{k_B T/2\pi m}\,, \tag{4.2}$$

where A_p is the probe area. More information regarding the implementation of OML theory in PROBEPIC can be found in [185]. To compare the particle distribution with the theoretical Maxwell-Boltzmann distribution, the current was derived using the Maxwellian EEDF,

$$F(\epsilon) = n\left(2/\left(k_B T\right)^{3/2}\right)\sqrt{\epsilon}\exp\left(-\epsilon/k_B T\right). \tag{4.3}$$

These comparisons confirmed that the updated version of PROBEPIC properly simulates equilibrium plasmas, as demonstrated in Ref [185]. A sample of raw Langmuir probe data for a non-equilibrium plasma is shown in Figure 4.5, which is the result of an experiment conducted in the vacuum test facility with argon gas using a cathode that was part of a Hall thruster on a U.S. spacecraft [202]. The experimental data were recorded via LabVIEW using a data acquisition system. The most apparent feature of the curve is that it lacks a well-defined knee. Determination of the plasma potential is not easy as there is no distinct knee in the raw I-V curve. Therefore, it is necessary to examine the first and second derivative for the plasma potential. In the ideal case, the plasma potential corresponds to the absolute maximum of the derivative. As this curve is not ideal the first derivative does not have the absolute minimum and second derivatives do not have a zero.

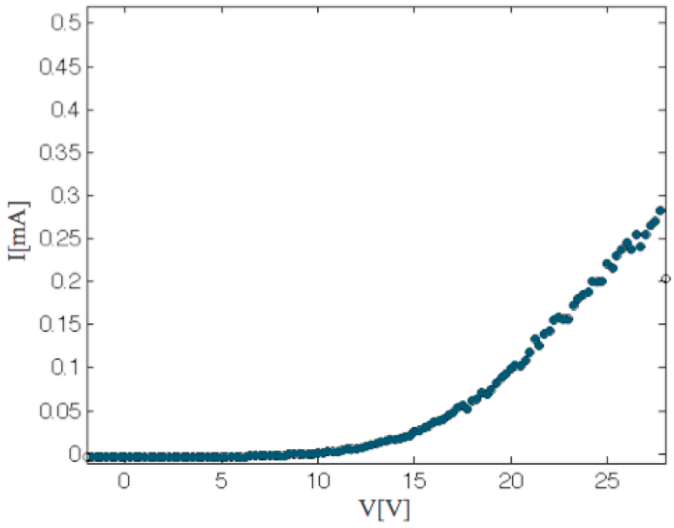

Figure 4.5: A sample of raw data collected during testing. Experimental I-V characteristic: Plasma potential $V_p = 21.05\text{V}, n_e = 1.8 - 2.3\text{x}10^{16}\text{m}^{-3}$, $T_e = 4.0 - 4.1\text{eV}$, $T_i = 0.4 - 0.41\text{eV}$, $m_i = 6.62\text{x}10^{-26}\text{kg}$ for argon [202].

The PROBEPIC simulation code requires input parameters to specify the probe geometry and the plasma parameters. The appropriate probe theory is chosen once the initial conditions and plasma environment are determined. Determining the appropriate theory helps to identify the correct set of plasma parameters. Then, the geometry, test conditions, and the method are determined for this particular theory. The Druyvesteyn method was found to be the most appropriate procedure and was subsequently used to determine the plasma parameters, which were then used as the input for PROBEPIC. The EEDF was derived using the Druyvesteyn equation,

$$F(\epsilon) = 2\left(\sqrt{2m}/e^3 I_o\right)\sqrt{eV}\left(d^2 I/dV^2\right), \tag{4.4}$$

and then the experimental EEDF was fitted to a theoretical Druyvesteyn distribution.

Each PROBEPIC simulation is initialized with a distribution of particles in the phase space. The velocity of every particle is determined via a random (MC procedure [210]) process that is constrained to result in the appropriate velocity distribution, whether it is equilibrium or non-equilibrium. The same constrained distribution is used to randomly select the velocity of each particle injected during the simulation. To implement the non-equilibrium plasma model, the PROBEPIC code was extended to accommodate the non-Maxwellian velocity distribution. PROBEPIC requires the integral EEDF to be calculated. This is normally done by the Druyvesteyn method (equation (2.48)). The PROBEPIC code requires the integral EEDF to be calculated as

$$\int_o^{\epsilon} F(\epsilon)\,d\epsilon = n\left[1 - \Gamma\left(3/4, 0.243\epsilon^2/\left(k_B T\right)^2\right)\right], \tag{4.5}$$

where $\Gamma(\epsilon, z)$ is the incomplete gamma-function [224, 225], and the limiting values of this integral are 0 for $\epsilon = 0$ and 1 for $\epsilon = \infty$. The functions (i.e., limiting values) are defined to eliminate error in the distribution function. The limiting values of the distribution function should be calculated numerically; otherwise, an error in the distribution function could drastically affect the calculations in the code. The distribution model was applied to all particles in the initial configuration space and all particles injected during the simulation process. The results were compared with the theory to ensure that the theoretical Druyvesteyn and simulated particle distributions were in agreement. Figure 4.6 shows a comparison between theory and simulation, with 3.5×10^5 super particles. Due to the high electron density and high number of super particles, the program works several orders of magnitude slower than simulations that have less than 2×10^5 super particles.

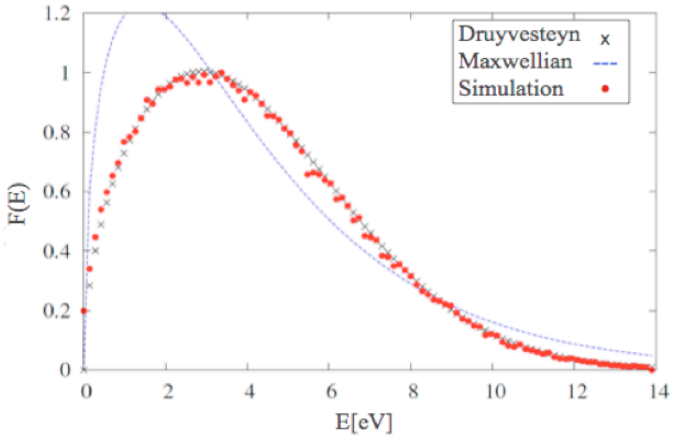

Figure 4.6: A comparison of the Maxwellian and Druyvesteyn distributions in a PROBEPIC simulation.

In Figure 4.7, I-V characteristics of the experimentally measured and simulated data are compared. The comparison of the I-V curves highlights some differences, such as discrepancies at high voltages. At negative voltages, the current is low and these data are in good agreement, with some minor differences. These slight differences at negative voltages appear to be due to the change in distributions; in this case, the distribution becomes Druyvesteyn. This is expected, because the velocity distribution of electrons in weakly ionized plasmas often deviate from Maxwellian to Druyvesteyn [193], where the collision frequency is velocity dependent but the mean free path length of electrons is constant [226]. This also confirms that the deviation from a Maxwellian distribution can be large enough that using a Maxwellian function for all plasma modeling is not a justified approximation.

For non-Maxwellian distribution functions, ions and electrons are not represented by the same temperature. Thus, for an accurate fit to the experimental measurements, the temperatures of the ions and the electrons should be determined separately. For non-thermal plasmas, the electron temperature is greater than the ion temperature; hence, $T_i = 0.1T_e$ [34, 41, 227] has been applied to these simulations. A decrease in the ion temperature results in a decrease in the ionic current. However, the electron current dominates at higher voltages, and a change in the ion temperature does not significantly alter the current, but it should improve the results at retarding voltages where the ion current dominates. There are several reasons for current change at high voltages, including chemical reactions inside the plasma, the impact of a modified electron population (i.e., electrons having different temperatures and density ratios) such as a non-Maxwellian electron population, and dependence on probe geometry. To test this hypothesis, experimental and theoretical curves were compared for two types of probe geometries, cylindrical and spherical, and the differences are seen in Figure 4.8. The finite length probe [228] could also be the reason for this current change. To test this, several

simulations of a probe with different aspect ratios (l_n/r_p) in the range of 5 to 150 (5, 10, 50 for short probe and 100 and above for long probe) were performed. Small probe aspect ratios yield different I-V characteristics, especially at voltages less than the plasma potential [229] (see Figure 4.8 right).

Small (short) probes reach electron saturation faster than longer probes. Ion-ion and ion-atom collisions, as well as the effect of a finite length probe, significantly affect ion orbital motion and tend to destroy it. As a result, the ion current to the probe under such conditions corresponds to radial ion motion rather than to orbital motion. This is more readily observed at voltages less than the plasma potential where the ions are more dominant. As the retarding current region of the I-V curve acts like an energy analyzer and carries information about the electron distribution function (EDF), choosing the right size and geometry of a probe is very important [230]. For a Maxwellian plasma, geometries such as cylindrical, spherical and planar are suitable for the measurement of T_e [41]. However, the electron saturation regions differ appreciably with collector geometry. The electron saturation region of the cylindrical probe is nearly independent of T_e, whereas this region is highly temperature-dependent for the planar and spherical probes [41], highlighting the practical advantage of the cylindrical probe.

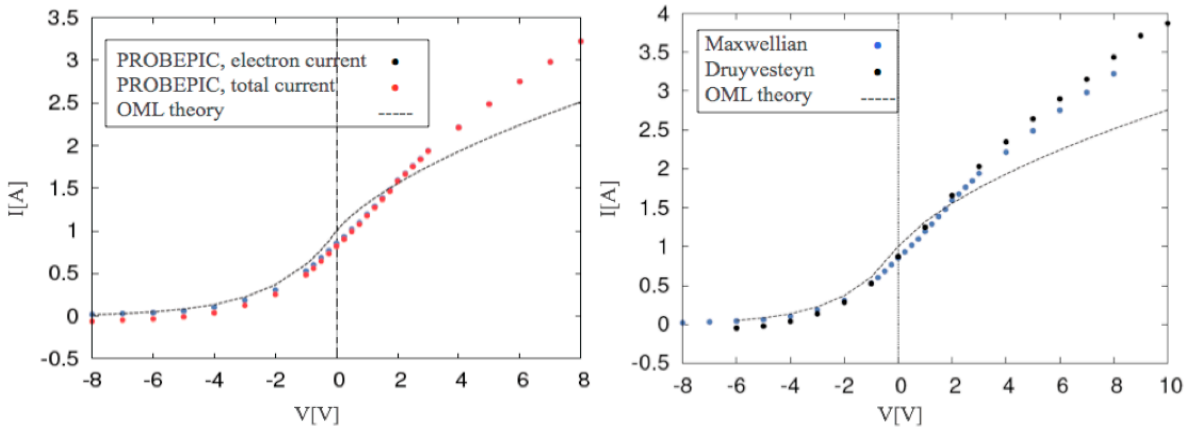

Figure 4.7: Graph (left) compares the results of the thick sheath plasma simulation with the theory: Maxwellian distribution. $V_p = 21.05V$, $n_e = 1.0x10^{15} m^{-3}$, $T_e = 2eV$, $r_p = 0.00003125m$, $l_p = 0.00287m$, $l_{ins} = 0.004305m$. Graph(rightcompares I-V relationships for different distributions. $n_e = 1.8 - 2.3x10^{16} m^{-3}$, $T_e = 4.0 - 4.1eV$, $T_i = 0.4 - 0.41eV$, $m_i = 6.62x10^{-26} kg$ for argon. The vertical line represents the plasma potential.

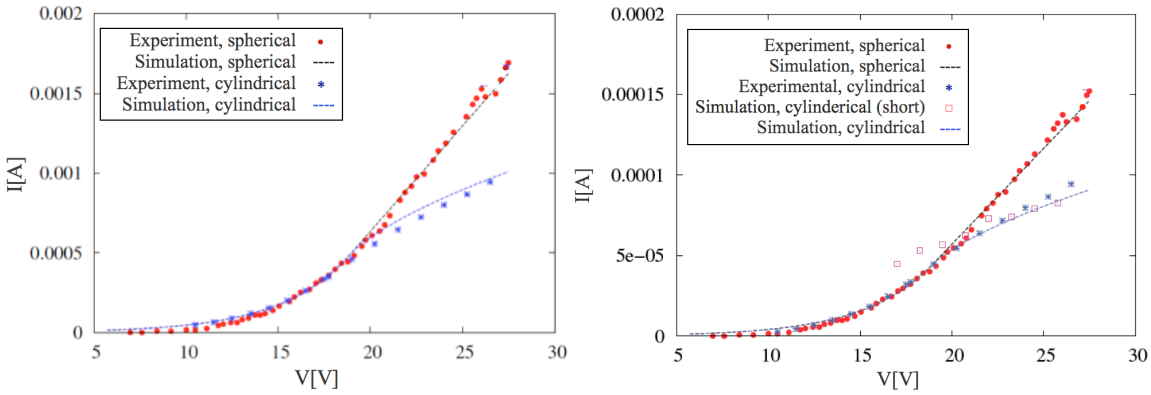

Figure 4.8: Left: comparisons of OML theory for spherical and cylindrical geometries with experimental [202] measurements. The simulation was conducted for Druyvesteyn distribution. Right: comparison of the results for aspect ratios of 100 and 10. $V_p = 16.6\text{V}$, $n_e = 6.5 - 7.0\text{x}10^{16}\,\text{m}^{-3}$, $T_e = 2.9\text{eV}$, $T_i = 0.29\text{eV}$, $m_i = 6.62\text{x}10^{-26}\,\text{kg}$ for argon, $l_p = 0.002\text{m}$, $r_p = 44.45\mu\text{m}$.

4.2.2 Application of the Bi-Maxwellian Method

In addition to Druyvesteyn's method, another distribution, called bi-Maxwellian, was implemented in the simulation code. This distribution was used as an alternative to the Druyvesteyn in an attempt to more closely replicate the experimental data [202]. The probe geometry used in this experiment is cylindrical because these data are more directly related to the distribution of energies projected onto the plane perpendicular to the probe axis. This method has the advantage of being a good alternative for precisely testing the PROBEPIC code [200]. First, the EEDF is determined using Druyvesteyn's method. Next, the EEDF is applied to the bi-Maxwellian distribution to extract the plasma parameters (e.g., two densities, two temperatures) [200]. When the two curves are in agreement, the correct implementation of the new distribution is verified.

As indicated earlier in Section 2.8.2, electron velocity distributions can be of different types for different electron energy ranges. For example, an electron distribution can be bi-Maxwellian at low electron energies and Maxwellian at high-electron energies. PROBEPIC simulations result in different I-V curves when the distribution transitions between Maxwellian and bi-Maxwellian. For a particular reference point, the results of the simulation using a Maxwellian distribution were compared with simple OML theory. When the results prove that both OML theory and the Maxwellian distribution demonstrate different behaviors at small voltages, where the contribution of high-energy electrons is important, the non-Maxwellian distribution is shown to be the more accurate model. Figure 4.9 shows the

simulation results for a bi-Maxwellian (two-temperature) distribution; both curves are in good agreement, verifying that the new distribution was correctly implemented.

The PROBEPIC simulation results in very different I-V curves when transitioning between Maxwellian and bi-Maxwellian simulations [194]. Both OML theory and the Maxwellian distribution produced incorrect behavior at small voltages. For these cases, the Debye length is similar to the probe radius, and there may be sheath effects influencing the currents in this region [231]. However, there is an important contribution of high-energy electrons in the region, and the distribution method used is non-Maxwellian.

An additional experiment is illustrated in Figure 4.10. This experiment was conducted with a cylindrical probe in a vacuum test facility consisting of a stainless steel vacuum chamber. The plasma source used for this experiment is the Busek BHT-1500 [232], which flew in space as part of the first successful Hall thruster demonstration on a U.S. spacecraft. The reported data (Figure 4.10) are digitized to obtain the current-voltage characteristic associated with the plasma. Note that the plasma parameters are obtained with an estimated error of 10 percent [202, 203], which induces differences in I-V curves. However, the simulated results still show good agreement with the experimental data (Figure 4.10).

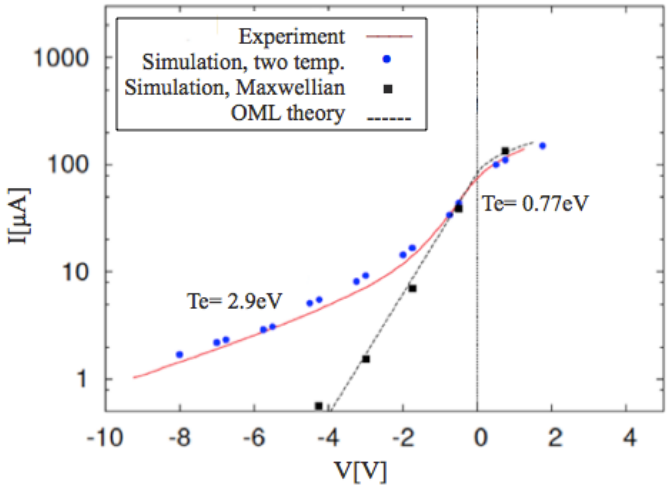

Figure 4.9: Comparison of experiment [202] and simulation using a two-temperature distribution. The other curves are for OML theory and simulation for the Maxwellian distribution. The dashed vertical line corresponds to plasma potential. $V_p = 16.6V$, $n_e = 6.5 - 7.0x10^{16} m^{-3}$, $T_e = 2.9eV$, $T_i = 0.1T_e eV$, $m_i = 6.62x10^{-26} kg$ for argon. Temperature of high-energy electrons: $T_h = 6.0eV$; energy of transitioning from low energy distribution to high: $\epsilon_t = 4.0eV$. As the distributions transition from one point to another, the graph appears to look different than the others. The vertical line represents plasma potential.

An additional experiment is illustrated in Figure 4.10. This experiment was conducted with a cylindrical probe in a vacuum test facility consisting of a stainless steel vacuum chamber. The plasma source used for this experiment is the Busek BHT-1500 [232], which flew in space as part of the first successful Hall thruster demonstration on a U.S. spacecraft. The reported data (Figure 4.10) are digitized to obtain the current-voltage characteristic associated with the plasma. Note that the plasma parameters are obtained with an estimated error of 10 percent [202, 203], which induces differences in I-V curves. However, the simulated results still show good agreement with the experimental data (Figure 4.10).

In the electron saturation region where the maximum is reached for the flux of electrons to the probe, the sheath continues to grow as the probe voltage is increased while the flux remains constant. The electron current increases with increasing sheath surface area. Because the current varies in the electron saturation region for spherical and cylindrical probes, the electron saturation current can be complex at first. With a spherical probe, no knee in the voltage-current characteristic is observed, but for a cylindrical probe, the knee is often rounded-off, providing a smooth transition to the electron saturation region [43]. Also, the theoretical cylinder yields a much closer approximation than the theoretical sphere.

The probe voltage is given with respect to the facility ground, and the plasma is at the plasma potential. When the probe voltage is well below the plasma potential, only the highest energy electrons reach the probe. As the probe voltage approaches the plasma potential, additional lower energy electrons are collected. A low probe voltage with respect to ground corresponds to a high electron energy with respect to the plasma. Therefore, a change of variables is necessary to express the EED with respect to the plasma potential. This also changes the reference voltage of the probe from the facility ground to the plasma potential. The Druyvesteyn equation, given in equation (2.53), is applied with the second derivative of the probe current with respect to the probe voltage. The data are also plotted on a logarithmic scale (Figure 4.11, right) to better visualize the plasma potential.

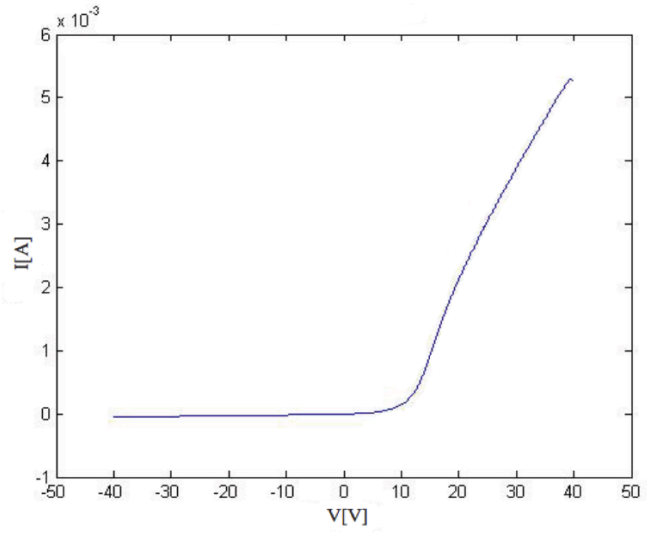

Figure 4.10: Experimental data [232] obtained using a cylindrical probe, showing probe current as a function of voltage. $V_p = 16.6\text{V}$, $n_e = 6.5 - 7.0\text{x}10^{16}\text{m}^{-3}$, $T_e = 2.85 - 3.0\text{eV}$, $T_i = 0.3\text{eV}$, $m_i = 6.62\text{x}10^{-26}\text{kg}$ for argon.

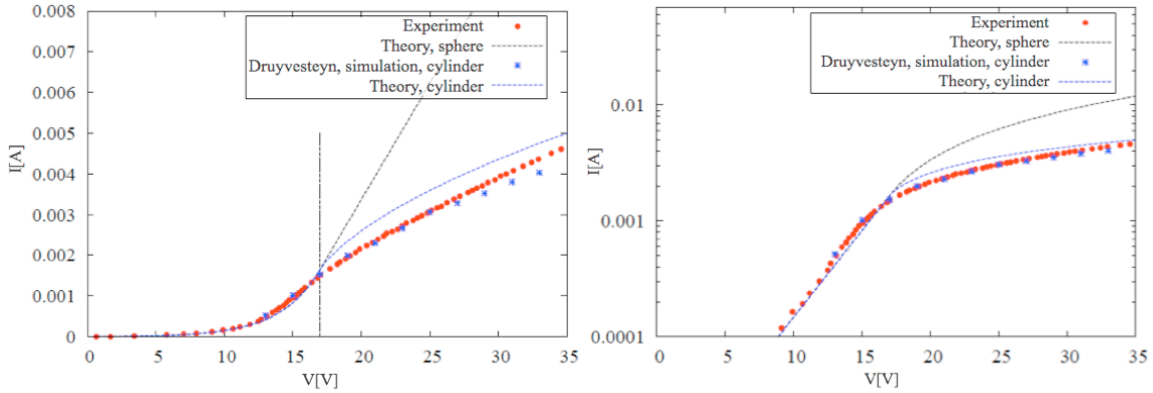

Figure 4.11: Comparison of computer simulations with experimental data. The vertical line represents the plasma potential. The figure on the right is plotted with a logarithmic scale. $V_p = 16.6\text{V}$, $n_e = 6.5\text{x}10^{16}\text{m}^{-3}$, $T_e = 2.9\text{eV}$, $T_i = 0.29\text{eV}$, $m_i = 6.62\text{x}10^{-26}\text{kg}$ for argon. The vertical line represents the plasma potential.

"Victory is for those who can say "Victory is mine". Success is for those who can begin saying "I will succeed" and say "I have succeeded" in the end."
- Mustafa Kemal Atatürk

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

Although the Langmuir probe technique for *in situ* measurement of plasma parameters has been around for eight decades, deriving the parameters with accuracy from the data acquired by a Langmuir probe immersed in space plasma is still a challenging task. This dissertation demonstrated that, although the vast majority of I-V characteristics can be fitted very well using the original Langmuir probe theory for an isotropic plasma with uniform temperature and density (isotropic), the use of non-Maxwellian distributions in such methods does not work.

This dissertation created a new, more powerful, more physically correct and more general PROBEPIC code for simulating plasma dynamics in both the SL and OML current collection regimes. For the thin sheath theory, good agreement between the simulated results and those derived analytically was obtained. The implementation of PROBEPIC for thick sheath theory was also successful, although a disparity was observed between the PROBEPIC models with a finite length probe and Langmuir probe theory with infinite length probe, which was expected [152] and demonstrates the need for general diagnostic applications of Langmuir probes. This results from the ideal state and infinite probe length assumed in Langmuir probe theory. The results of the implementations of both cases, SL and OML current collection methods, show good agreement with equilibrium Langmuir probe theory and provide validation for using particle simulations to accurately model Langmuir probes.

After expansion of the model to include non-Maxwellian distribution methods (i.e., Druyvesteyn and bi-Maxwellian methods), Langmuir probe theory associated with non-equilibrium plasmas [42, 46] was implemented into the PROBEPIC code. The EEDF is obtained using Druyvesteyn and bi-Maxwellian procedures, and parameters such as electron temperature, plasma density and plasma potential are also obtained. The dependence of EEDF and other

parameters is examined. Three test cases were presented to establish an algorithm through which the described PIC code can be used to analyze experimental data. Through this approach, important parameters of non-equilibrium plasmas can be obtained, and theoretical I-V traces can be created for comparison with experimental traces. The plasma conditions are determined when the experimental and theoretical I-V characteristics agree within a desired level of accuracy.

The results show that EEDF is not necessarily Maxwellian everywhere in the plasma. Considerable differences in electron density and temperature are observed, and these values differ dramatically in many cases. According to these results, the EEDF is best described by the Druyvesteyn distribution because its validity is independent of the sheath size and its determination of the EEDF provides a more detailed description of the plasma. However, results also show that the EEDF changes from one distribution to another at different levels of gas pressure (e.g., EEDF is Druyvesteyn-like when the pressure $P_{argon} = 0.3\text{Torr}$ and becomes bi-Maxwellian when the pressure $P_{argon} = 0.03\text{Torr}$).

Three distribution techniques, Maxwellian, bi-Maxwellian and Druyvesteyn methods, were carefully implemented in PROBEPIC. As a result of this procedure, we can determine the energy levels even at the points where two distributions transition [179]. Another reason for including bi-Maxwellian method in PROBEPIC is that, this method can be used as a reference distribution to be able to confirm whether the correct method is applied for the plasma condition. Methods for finding the electron distribution function and the electron density were compared for a cylindrical probe. The reference method for analyzing the probe data is to assume that these data are derived from the sum of bi-Maxwellian distributions and to find the two temperatures and densities that fit the data. Both I-V curves were in good agreement for the test case, verifying that the new distribution was correctly implemented and that Druyvesteyn's analysis for the speed distribution agrees with the bi-Maxwellian analysis.

This dissertation also introduced and demonstrated that three probe geometries are not equally suitable for the measurement of electron temperature. Distinct methods for deriving temperature tend to yield different temperature values when the velocity distribution is non-Maxwellian. Distinct techniques or instrument types also lead to different plasma probe results. Therefore, for anisotropic distributions, not all probe geometries are equally suited for measuring the electron temperature and density or for determining the distribution function in the presence of non-Maxwellian background electrons. To test this hypothesis, the experimental curve was compared with the theory for spherical and cylindrical geometries, and the differences were illustrated.

Langmuir probes are widely used for plasma diagnostics. Their small size and relatively simple theory of operation make them indispensable tools. Expanding the capabilities of PROBEPIC to simulate Langmuir probes within the non-

Maxwellian plasmas greatly enhances the utility of this software tool, allowing the exploration of the underlying principles and behavior of an idealized version of the plasma unperturbed by experimental intrusion. Furthermore, the plasma parameters can be changed far more quickly and cheaply than in a physical experimental setup. This capability directly aids in diagnostic efforts using Langmuir probes.

The physical regimes created in the EP thrusters indicate that enhance numerical methods are required for accurate numerical simulation. It is concluded that numerical simulations can be expected to play a more prominent role in the design and evolution of future EP thrusters. This work provides an opportunity to expand the code to produce simulations of the entire plasma inside EP devices, providing insight to help make them more efficient and increasing their performance, as well as enabling the identification of the life-limiting factors of these devices.

5.2 Recommendations

The possibilities for PROBEPIC's future applications are virtually limitless. The purpose of this work was to verify accurate modeling of the physical processes; to this end, widely used, robust algorithms were employed. We now have a code that apparently models reality, to the first and second order, quite well.

Further work should include modeling different plasma experiments and developing PROBEPIC as a diagnostic aid for both evaluating experimental data and simulating realistic plasmas to conduct virtual experiments, e.g., space environment plasmas. With additional work, the tool could be expanded to provide simulations of the gas and plasma flows in various EP thrusters, thereby aiding in the enhancement of these high-performance devices.

PROBEPIC should also implement other electrostatic probes, e.g., Faraday cups and retarding potential analyzers, to simulate plasma diagnostics to provide insight to improving conventional designs. This would provide user an interface to a variety of sensor hardware in order to create desired space and laboratory testing environment.

Bibliography

[1] R. J. Goldston, L. M. Kovrizhnykh, P. K. Kaw, J. VMeyer-ter-Vehn, and Y. Seki, "Plasma Physics and Controlled Nuclear Fusion Research," *Conference Summeries*, 1995.

[2] V. N. Rai, "Basic Concept in Plasma Diagnostics," *Bulletin of Laser and Spectroscopy Society of India,* vol. 13, pp. 55-70, 2004.

[3] J. W. Dunning, S. Benson, and S. Oleson, "NASA's Electric Propulsion Program," *IEPC,* vol. 01, no. 002, 2001.

[4] N. Farid, S. S. Harilal, O. El-Atwani, H. Ding, and A. Hassanein, "Experimental simulation of materials degradation of plasma-facing components using lasers," *Nuclear Fusion,* vol. 54, no. 1, pp. 012002, 2014.

[5] E. A. Oks, "Principles of Spectroscopic Diagnostics of Plasma with Oscillating Electric Fields," *Measurement Techniques,* vol. 29, no. 9, pp. 805-811, 1986, 1986.

[6] J. H. Malmberg, C. F. Driscoll, and T. M. O'Neil, *Pure Electron Plasmas Near Thermal Equilibrium*, University of California, San Diago, La Jolla, CA, 1990.

[7] M. A. Lieberman, and A. J. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, 2nd Edition ed., New Jersey: John Wiley and Sons, Inc., 2005.

[8] F. F. Chen, "Electric Probes," *Plasma Diagnostic Techniques*, pp. 113-200, New York: Academic Press, 1965.

[9] E. Y. Choueiri, "A critical history of electric propulsion: the first 50 years (1906-1956)," *Journal of Propulsion and Power,* vol. 20, no. 2, pp. 193-203, 2004.

[10] R. H. Goddard, *The Green Notebooks*: The Dr. Robert H. Goddard Collection at Clark University Archives, Clark University, Worcester, MA.

[11] R. G. Jahn, *Physics of electric propulsion*, 1st ed., New York: McGraw-Hill, 1968.

[12] R. L. Sackheim, R. S. Wolf, and S. Zafran, *Space Mission Analysis and Design*, Torrance, CA and Norwell, MA: Microcosm, Inc. and Kluwer Academic Publishers, 1995.

[13] R. A. Serway, *Physics for Scientists and Engineers with Modern Physics*, Chicago, IL.: Saunders College Publishing, 1996.

[14] M. Martinez-Sanchez, and J. E. Pollard, "Spacecraft Electric Propulsion – An Overview," *Journal of Propulsion and Power,* vol. 14, no. 5, pp. 688-699, 1998.

[15] M. Auweter-Kurtz, T. Golz, H. Habiger, F. Hammer, H. Kurtz, M. Riehle, and C. Sleziona, "High-Power Hydrogen Arcjet Thrusters," *Journal of Propulsion and Power,* vol. 14, no. 5, pp. 764-773, 1998.

[16] V. Hruby, J. Monheiser, B. Pote, P. Rostler, J. Kolencik, and C. Freeman, "Development of Low-Power Hall Thrusters,," in 30th Plasmadynamics and Lasers Conference,, Norfolk, VA, 1999.

[17] R. R. Hofer, and A. D. Gallimore, "The Role of Magnetic Field Topography in Improving the Performance of High-Voltage Hall Thrusters," in 38th AIAA / ASME / SAE / ASEE Joint Propulsion Conference, Indianapolis, IN, 2002.

[18] D. T. Jacobson, and D. H. Manzella, "50 kW Class Krypton Hall Thruster Performance," in 39th AIAA / ASME / SAE / ASEE Joint Propulsion Conference, Huntsville, AL, 2003.

[19] L. Biagioni, F. Ceccanti, M. Saverdi, M. Saviozzi, and M. Andrenucci, "Qualification status of the FEEP-150 electric micropropulsion subsystem," in AIAA-2005-4261 , 2005, presented at the 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Tucson, AZ., 2005.

[20] C. E. Garner, J. R. Brophy, J. E. Polk, and L. C. Pless, "Cyclic endurance test of a SPT-100 statonary plasma thruster," in 30th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 1994.

[21] E. J. Pencil, T. Randolph, and D. Manzella, "End-of-life stationary plasma thruster far-field plume characterization," in 32nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Buena Vista, FL., 1996.

[22] M. Britton, D. Waters, R. Messer, E. Sechkar, and B. Banks, *Sputtering erosion measurement on boron nitride as a Hall thruster material,* NASA TM-2002-211837, 2002.

[23] T. Randolph, E. Pencil, and D. Manzella, "Far-field plume contamination and sputtering of the stationary plasma thruster," in 30th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 1996.

[24] J. M. Fife, W. A. Hargus, D. A. Jaworske, R. Jankovsky, L. Mason, C. Sarmiento, J. S. Snyder, S. Malone, J. Haas, and A. Gallimore, "Spacecraft Integration test results of the high performance Hall system SPT-140," in 36th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Huntsville, AL., 2000.

[25] D. Manzella, R. Jankovsky, and R. Hofer, "Laboratory model 50 kW Hall thruster," in 38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 2002.

[26] W. A. Hargus, and G. Reed, "The Air Force clustered Hall thruster program," in 38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 2002.

[27] B. E. Beal, A. D. Gallimore, and W. A. Hargus, "Preliminary plume characterization of a low-power Hall thruster cluster," in 38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 2002.

[28] B. E. Beal, A. D. Gallimore, and W. A. Hargus, "The effects of clustering multiple Hall thrusters on plasma plume properties," in 39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Huntsville, AL., 2003.

[29] P. Y. Peterson, and A. D. Gallimore, "The performance and plume characterization of a laboratory gridless ion thruster with closed electron drift acceleration," in 40th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Fort Lauderdale, FL., 2004.

[30] T. Ito, Gascon, N., W. S. Crawford, and M. A. Capelli, "Experimental characterization of a micro-Hall thruster," *Journal of Propulsion and Power,* vol. 23, no. 5, pp. 1068–1074, 2007.

[31] D. W. Youngner, S. T. Lu, E. Choueiri, J. B. Neidert, R. E. Black, K. J. Graham, D. Fahey, R. Lucus, and X. Zhu, "MEMS mega-pixel micro-thruster arrays for small satellite stationkeeping," in 14th AIAA/USU Small Satellite Conference, North Logan, UT., 2000.

[32] "Vacuum Facilities at the Plasmadynamics and Electric Propulsion Laboratory," http://aerospace.engin.umich.edu/spacelab/facilities/facilities.html.

[33] M. L. R. Walker, A. L. Victor, R. R. Hofer, and A. D. Gallimore, "Effects of backpressure on ion current density measurements in Hall thruster plumes," *Journal of Propulsion and Power,* vol. 21, no. 3, pp. 408-415, 2005.

[34] D. M. Goebel, and I. Katz, *Fundamentals of Electric Propulsion: Ion and Hall Thrusters*, New Jersey: John Wiley and Sons, Inc., 2008.

[35] R. Fitzpatrick, *Plasma Physics*, Texas, 2012.

[36] T. F. R. Equipe, "Tokamak Plasma Diagnostics," *Nuclear Fusion,* vol. 18, pp. 647-731, 1978.

[37] M. I. Boulos, P. Fauchais, and E. Pfender, *Thermal Plasmas: Fundamentals and Applications*, NewYork, NY: Springer, 1994.

[38] S. Wu, *Polymer Interface and Adhesion CRC Press*, p.^pp. 299, New York, NY: CRC Press, 1982.

[39] K. Wasa, and S. Hayakawa, *Handbook of Sputter Deposition Technology: Principles, Technology and Applications (Materials Science and Process Technology Series)*, Westwood, NJ: William Andrew Inc., 1992.

[40] M. Goossens, *Plasma Astrophysics and Magnetohydrodynamics*, p.^pp. 216, Charlottesville, VA: Springer, 2003.

[41] W. R. Hoegy, and L. H. Brace, "Use of Langmuir Probes in Non-Maxwellian Plasmas," *Rev. Sci. Instrum,* vol. 70, no. 7, pp. 3015-3024, 1999.

[42] J. Cooper, "Plasma Spectroscopy," *IOP Science,* vol. 29, no. 35, 1966.

[43] F. F. Chen, "Lecture Notes on Langmuir Probe Diagnostics," Mini-Course on Plasma Diagnostics, IEEE-ICOPS meeting. Jeju, Korea, 2003.

[44] L. H. Brace, "Langmuir Probe Measurements in the Ionosphere," *Measurement Techniques in Space Plasmas: Particles*, Geophysical Monograph Series R. F. Pfaff, J. E. Borovsky and D. T. Young, eds., pp. 23-25, Washington, D.C.: American Geophysical Union, 1998.

[45] L. H. Brace, R. L. Boggess, and N. W. Spencer, "Langmuir Probe Measurements in the Ionesphere," *Journal of Geophysical Research,* vol. 64, no. 10, 1959.

[46] G. N. Taylor, and G. L. Wrenn, "Comparisons of simultaneous satellite and ground-based measurements of ionospheric parameters," *Planetary and Space Science,* vol. 18, pp. 1663-1666, 1970.

[47] L. Conde, "An introduction to Langmuir probe diagnostics of plasmas," Universidad Politécnica de Madrid, 2011.

[48] T. N. Academies, "Plasma Science: From Fundamental Research to Technological Applications," National Academy Press 1995.

[49] ImpedansLtd. "Plasma Measurement to Understand and Control the Future," 2014; http://www.impedans.com/langmuir_probes.

[50] J. G. Laframboise, *Theory of spherical and cylindrical Langmuir probes in a collisionless, Maxwellian plasma at rest,* Report No.100, UTIAS, 1966.

[51] D. A. Herman, and A. D. Gallimore, "A high-speed probe positioning system for interrogating the discharge plasma of a 30 cm ion thruster," in 38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Indianapolis, IN., 2002.

[52] R. Eckman, L. Byrne, E. Cameron, and N. A. Gatsonis, "Triple Langmuir probe measurements in the plume of a pulsed plasma thruster," in 34th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Cleveland, OH, 1998.

[53] R. F. Pfaff, J. E. Borovsky, and D. T. Young, *Measurement Techniques in Space Plasmas: Particles*, Washington, DC: American Geophysical Union, 1998.

[54] E. V. Shun'ko, *Langmuir Probe in Theory and Practice*, Boca Raton: Universal Publishers, 2008.

[55] D. Bohm, "Minimum Ionic Kinetic Energy for a Stable Sheath," *The Characteristics of Electrical Discharges in Magnetic Fields*, A. Guthrie and R. Wakerling, eds., pp. 77-86, New York: McGraw-Hill, 1949.

[56] J. E. Allen, R. L. F. Boyd, and P. Reynolds, "The Collection of PositiveIonsby a Probe Immersed in a Plasma," *IOP Science: Proc. Phys. Soc. B,* vol. 70, pp. 297, 1956.

[57] I. B. Bernstein, and I. N. Rabinowitz, "Theory of Electrostatic Probes in a Low-Density Plasma," *AIP Physics of Fluids,* vol. 2, no. 2, pp. 112-121, 1959.

[58] J. G. Laframboise, "Probe design for orbit-limited current collection," *Physics of Fluids,* vol. 16, no. 5, pp. 629, 1973.

[59] Y. A. Ivanov, Y. A. Lebedev, and L. S. Polak, *Methods of In Situ Diagnostics of Nonequilibrium Plasma-Chemistry*, Moscow: Nauka, 1981.

[60] H. Amemija, and Y. Sakamoto, "Diagnostics of Low Temperature Plasma by the Probe Method," *Journal of Vacuum Society of Japan,* vol. 28, no. 4, pp. 177-192, 1985.

[61] R. J. Goldston, and P. H. Rutherford, *Introduction to Plasma Physics*, p.^pp. 13-15, New York: Taylor and Francis, 1995.

[62] T. E. Sheridan, "How big is a small Langmuir probe?," *Physics of Plasmas,* vol. 7, no. 7, pp. 3084, 2000.

[63] I. Langmuir, "The interaction of electron and positive ion space charges in cathode sheaths," *Physical Review,* vol. 33, pp. 954-989, 1929.

[64] D. Bohm, *The Characteristics of Electric Discharges in Magnetic Fields*, p.^pp. 1-76, New York, NY: McGraw-Hill, 1949.

[65] G. A. Emmert, R. M. Wieland, A. T. Mense, and J. N. Davidson, "Electric sheath and presheath in a collisionless, finite ion temperature plasma," *Physics of Fluids,* vol. 23, no. 4, pp. 803-812, 1980.

[66] I. H. Hutchinson, "A fluid theory of ion collection by probes in strong magnetic fields with plasma flow," *Physics of Fluids,* vol. 30, no. 12, pp. 3777-3781, 1987.

[67] K. U. Riemann, "Kinetic theory of the plasma sheath transition in a weakly ionized plasma," *Physics of Fluids,* vol. 24, no. 12, pp. 2163-2172, 1981.

[68] I. D. Kaganovich, "How to patch active and collisionless sheath: a practical guide," *Physics of Plasmas,* vol. 9, no. 11, pp. 4788-4793, 2002.

[69] V. Godyak, and N. Sternberg, "Good news: you can patch active plasma and collisionless sheath," *IEEE Transactions on Plasma Science,* vol. 31, no. 2, pp. 303, 2003.

[70] M. Keidar, and I. I. Beilis, "Transition from plasma to space-charge sheath near the electrode in electrical discharges," *IEEE Transactions on Plasma Science,* vol. 33, no. 5, pp. 1481-1486, 2005.

[71] K. U. Riemann, "Plasma-sheath transition in the kinetic Tonks-Langmuir model," *Physics of Plasmas,* vol. 13, no. 063508, 2006.

[72] T. E. Markusic, "Particle Simulation of a Langmuir Probe in Quiescent and Flowing Plasmas," M.S. thesis, M.A.B.E. Dept., The University of Tennessee, Knoxville, TN., 1996.

[73] M. Chiaretta, "Numerical Modelling of Langmuir Probe Measurements for the Swarm Spacecraft," Diploma Thesis, Department of Astronomy and Space Physics, Swedish Institute of Space Physics, Uppsala University, Sweden, 2011.

[74] S. Sharma, "Investigation of Ion and Electron Kinetic Phenomena in Capacitively Coupled Radio-Frequency Plasma Sheaths: A Simulation Study," Ph.D. thesis, School of Physical Sciences, Dublin City University, Dublin, Ireland, 2012.

[75] P. E. Sheerin, "Development of Planar Langmuir Probes For Supersonic Plasma Flow," Ph.D. thesis, School of Physical Sciences, Dublin City University, Dublin, Iralend, 2009.

[76] K. U. Riemann, and T. Daube, "Analytical model of the relaxation of a collionless ion matrix sheath," *Journal of Applied Physics,* vol. 86, no. 1202, 1999.

[77] S. A. Ledvina, Y. J. Ma, and E. Kallio, "Modeling and Simulating Flowing Plasmas and Related Phenomena," *Space Science Reviews,* vol. 139, no. 1-4, pp. 143-189, 2008.

[78] E. Havlickova, "Fluid Model of Plasma and Computational Methods for Solution." pp. 180-186.

[79] C. K. Birdsall, and A. B. Langdon, *Plasma Physics via Computer Simulation*, New York, NY: Adam Hilger: IOP Publishing Ltd., 1991.

[80] G. A. Bird, *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford, UK: Oxford University Press, 1994.

[81] V. A. Godyak, and V. I. Demidov, "Probe measurements of electron-energy distributions in plasmas: what can we measure and how can we achieve reliable results?," *Journal of Physics D: Applied Physics,* vol. 44, no. 26, pp. 269501, 2011.

[82] F. F. Chen, and G. Bekefi, "Introduction to Plasma Physics," *American Journal of Physics,* vol. 44, no. 112, 1976.

[83] S. I. Bahtiyarov, *Handbook of Cosmic Hazards and Planetary Defense*, Socorro, NM: Springer International Publishing, 2015.

[84] L. Combs, and R. Viereck, "Aurora," *NORA: Space Environment Topics,* vol. SE-12, 1996.

[85] A. Piel, *Plasma Physics: An Introduction to Laboratory, Space, and Fusion Plasmas*, Kiel, Germany: Springer-Verlag Berlin Heidelberg, 2010.

[86] R. Fitzpatrick, *Thermodynamics & Statistical Mechanics: An intermediate level course*, Austin, TX: University Reprints, 2012.

[87] F. Reif, *Fundamentals of Statistical and Thermal Physics*, New York, NY: Waveland Pr Inc, 2008.

[88] F. F. Chen, *Introduction to Plasma Physics and Controlled Fusion*, 2nd ed., New York, NY: Plenum Press, 1984.

[89] J. Hopwood, "Ionized physical vapor deposition of integrated circuit interconnects," *Physics of Plasmas,* vol. 5, no. 5, pp. 1624-1631, 1998.

[90] S. M. Rossnagel, "Thin Solid Films," *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films,* vol. 263, pp. 1-12, 1995.

[91] F. J. d. Heer, R. H. J. Jansen, and W. v. d. Kaay, "Total cross sections for electron scattering by Ne, Ar, Kr and Xe," *J. Phys. B: At. Mol. Phys.,* vol. 12, no. 6, pp. 979, 1979.

[92] L. Vriens, "Calculation of absolute ionisation cross sections of He, He*, He+, Ne, Ne*, Ne+, Ar, Ar*, Hg and Hg*," *Physics Letters,* vol. 8, no. 4, pp. 260-261, 1964.

[93] D. Rapp, and P. Englander-Golden, "Total cross sections for ionization and attachment in gases by electron impact. I. Positive ionization," *Journal of Chem. Phys.,* vol. 43, pp. 1464-1479, 1965.

[94] L. L. Shimon, E. I. Nepiipov, and I. P. Zapesochnyi, "Effective total electron-impact ionization cross sections for aluminum, gallium, indium and thallium," *Sov. Phys. Tech. Phys.,* vol. 20, no. 3, pp. 434-437, 1975.

[95] M. A. Lennon, K. L. Bell, H. B. Gilbody, J. G. Hughes, A. E. Kingston, M. J. Murray, and F. J. Smith, "Recommended Data on the Electron Impact Ionization of Atoms and Ions: Fluorine to Nickel," *J. Phys. Chem.,* vol. 17, pp. 1285-1363, 1988.

[96] W. Lotz, "Electron-Impact Ionization Cross-Section for Atoms up to Z=108*," *Z. Physik,* vol. 232, pp. 101-107, 1970.

[97] M. W. Thompson, "II. The energy spectrum of ejected atoms during the high energy sputtering of gold," *Philosophical Magazine,* vol. 18, no. 152, pp. 377-414, 1968.

[98] K. D. Goodfellow, "A Theoretical and Experimental Investigation of Cathode Processes In Electric Thrusters," Doctor of Philosophy, University of Southern California, May 1996.

[99] A. Grill, *Cold Plasma in Metarials Fabrication: from Fundamentals to Applications*, Piscataway, NJ: Institute of Electrical and Electronics Engineers Press, Inc., 1994.

[100] D. A. Gurnett, and A. Bhattacharjee, *Introduction to Plasma Physics: With Space and Laboratory Applications*, Cambridge, U.K.: Cambridge University Press, 2005.

[101] T. Boyd, and J. Sanderson, *The Physics of Plasmas*, UK: Cambridge University Press, 2003.

[102] U. Inan, and M. Galkowski, *Principles of Plasma Physics for Engineers and Scientists*, New York, NY: Cambridge University Press, 2011.

[103] P. M. Bellan, *Fundamentals Of Plasma Physics*: Cambridge University Press, UK, 2006.

[104] E. H. Holt, and R. E. Haskell, *Foundation of plasma dynamics*, 1st ed., New York: Macmillan, 1965.

[105] M. Brünglinghaus. "Mean Free Path," http:// www.euronuclear.org/info/encyclopedia/m/mean-fee-path.htm.

[106] M. S. Livingston, and J. Blewett, *Particle Accelerators*, New York, NY: McGraw-Hill, 1969.

[107] G. W. Sutton, and A. Sherman, *Engineering Magnetohydrodynamics*, New York, NY: McGraw-Hill Book Company, 1965.

[108] A. F. Aleksandrov, L. S. Bogdankevich, and A. A. Rukhadze, *Principles of plasma electrodynamics*, 1st ed., Moscow, Russia: Springer-Verlag, 1984.

[109] J. M. Burgers, W. M. Elsasser, F. Hoyle, A. Kantrowitz, and M. Rosenbluth, *Magnetohydrodynamics*, Stanford, CA: Stanford University Press, 1985.

[110] L. G. Christophorou, and J. K. Olthoff, *Fundamental Electron Interactions With Plasma Processing Gases*, p.^pp. 39, 2004.

[111] K. Yamamoto, and D. C. Pack, "Transient free molecular flow through a tube." pp. 207-218.

[112] W. G. Vincenti, and C. H. Kruger, *Introduction to Physical Gas Dynamics*, New York: Krieger, 1975.

[113] G. P. Sutton, and O. Biblarz, *Rocket Propulsion Elements*, 7 ed.: Wiley, 2001.

[114] I. H. Hutchinson, *Plasma Diagnostics For Tokamaks and Stellarators*, Madrid, Spain: Department of Energy: Office of Scientific and Technical Information 1994.

[115] A. Anders, "Fundamentals of pulsed plasmas for materials processing," *Surf. Coat. Technol.,* vol. 183, pp. 301-311, 2004.

[116] I. Langmuir, "The Effect of Space Charge and Residual Gases on Themionic Currents in High Vacuum," *Physical Review,* vol. 2, pp. 450, 1913.

[117] E. Sommier, M. K. Allis, and M. A. Capelli, "Wall erosion in 2D Hall thruster simulations," in 29th International Electric Propulsion Conference, Princeton, NJ., 2005.

[118] G. Fairweather, A. R. Gourlay, and A. R. Mitchell, "Some high accuracy difference schemes with a splitting operator for equations of parabolic and elliptic type," *Numerische Mathematik,* vol. 10, no. 156-66, 1967.

[119] J. T. Yim, M. Falk, M. Keidar, and I. D. Boyd, "Calculation of Boron Nitride Sputter Yields Under Low Energy Xenon Ion Bombardment," in 43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Cincinnati, OH., 2007.

[120] I. D. Boyd, K. Scavezze, C. Cai, J. E. Polk, and E. Y. Choueiri, "Analysis of plume backflow from a lithium magnetoplasmadynamic thruster," in 29th International Electric Propulsion Conference, Princeton, NJ., 2005.

[121] M. Kim, M. Keidar, I. D. Boyd, and D. Morris, "Plasma density reduction using electromagnetic ExB field during reentry flight," in 43rd ITC/USA Conference, Las Vegas, NV., 2007.

[122] C. Cai, and I. D. Boyd, "3D Simulation of Plume Flows from a Cluster of Plasma Thrusters," in 36th AIAA Plasmadynamics and Lasers Conference, Toronto, Ontario., 2005.

[123] A. A. Alexeenko, D. A. Fedosov, S. F. Gimelshein, D. A. Levin, and R. J. Collins, "Transient heat transfer and gas flow in a MEMS-based thruster," *Journal of Microelectromechanical Systems,* vol. 15, no. 1, pp. 181-194, 2006.

[124] F. F. Cap, *Handbook of Plasma Instabilities*, London, UK: Academic Press, 1978.

[125] H. Mott-Smith, and I. Langmuir, "The Theory of Collectors in Gaseous Discharges," *Physical Review,* vol. 28, no. 4, pp. 727-763, 1926.

[126] Y. I. Sung, H. B. Lim, and R. S. Houk, "Diagnostic studies of a low-pressure inductively coupled plasma in argon using a double Langmuir probe," *Journal of Analytical Atomic Spectrometry,* vol. 17, pp. 565-569, 2002.

[127] M. Wendt, "Experimental Investigation of Wave Motion and Electric Resistance in Collisionfree Plasmas," Division of Plasma Physics, Royal Institute of Technology, Stocholm, Sweeden, 2001.

[128] B. Toftmann, J. Schou, T. N. Hansen, and J. G. Lunney, "Angular distribution of electron temperature and density in a laser-ablation plume," *Phys Rev Lett.,* vol. 84, no. 17, pp. 3998-4001, 2000.

[129] T. N. Hansen, J. Schou, and J. G. Lunney, "Langmuir probe study of plasma expansion in pulsed laser ablation," *Applied Physics A,* vol. 69, no. 1, pp. 601-604, 1999.

[130] J. M. Hendron, C. M. O. Mahony, T. Morrow, and W. G. Graham, "Langmuir probe measurements of plasma parameters in the late stages of a laser ablated plume," *Journal of Applied Physics,* vol. 81, no. 5, pp. 2131, 1997.

[131] V. I. Demidov, S. V. Ratynskaia, and K. Rypdal, "Electric probes for plasmas: The link between theory and instrument," *Review of Scientific Instruments,* vol. 73, no. 10, pp. 3409, 2002.

[132] Z. Zhang, P. A. VanRompay, J. A. Nees, and P. P. Pronko, "Multi-diagnostic comparison of femtosecond and nanosecond pulsed laser plasmas," *Journal of Applied Physics,* vol. 92, no. 5.

[133] S. B. Segall, and D. W. Koopman, "Application of cylindrical Langmuir probes to streaming plasma diagnostics," *Physics of Fluids,* vol. 16, no. 7, pp. 1149-1156, 1973.

[134] C. S. MacLathchy, "Charge Depletion Downstream of a Cylindrical Langmuir Probe Immersed in a Flowing, High-pressure Plasma," *Plasma Science, IEEE Transactions on,* vol. 17, no. 1, pp. 29-33, 1989.

[135] A. Bergmann, "Two-dimensional particle simulation of Langmuir probe sheaths with oblique magnetic field," *Physics of Plasmas,* vol. 1, no. 1, pp. 3598-3606, 1994.

[136] I. H. Hutchinson, "Ion collection by a sphere in a flowing plasma: I. Quasineutral," *IOP Science,* vol. 44, no. 9, pp. 1953-1977, 2002.

[137] B. Doggett, C. Budtz-Joergensen, J. G. Lunney, P. Sheerin, and M. M. Turner, "Behaviour of a planar Langmuir probe in a laser ablation plasma," *Appl. Surf. Sci,* vol. 247, no. 1-4, pp. 134-138, 2005.

[138] J. P. Booth, N. S. J. Braithwaite, A. Goodyear, F. Neuilly, and J.-M. Francou, "Ion flux and electron temperature measurements with an insulator-covered planar electrostatic probe."

[139] N. S. J. Braithwaite, J. P. Booth, and G. Cunge, "A novel electrostatic probe method for ion flux measurements," *Plasma Sources Science and Technology,* vol. 5, pp. 677-684, 1996.

[140] R. M. Clements, and P. R. Smy, "The positively biased planar Langmuir probe in high-pressure plasma," *IOP Science,* vol. 26, no. 11, pp. 1916-1920, 1993.

[141] A. Skoelv, R. J. Armstrong, and J. Trulsen, "Ion-beam diagnostics by means of an electron-saturated plane Langmuir probe," *Physics of Fluids,* vol. 27, no. 11, pp. 2744-2751, 1984.

[142] S. G. Lee, D. Diebold, N. Hershkowitz, T. Intrator, X. Wang, and G. H. Kim, "Direct measurement of plasma flow velocity using a Langmuir probe," *AIP Rev. Sci. Instrum.,* vol. 64, no. 2, pp. 481, 1993.

[143] D. N. Ruzic, *Electric probes for low temperature plasmas (AVS monograph series)*, 1st ed. ed., New York: American Vacuum Society, 1994.

[144] I. D. Sudit, and R. C. Woods, "Study of the Accuracy of Various Langmuir Probe Theories," *Journal of Applied Physics,* vol. 76, no. 8, pp. 4488-4498, 1994.

[145] J. Rubinstein, and J. G. Laframboise, "Theory of a spherical probe in a collisionless magnetoplasma," *Physics of Fluids,* vol. 25, pp. 1174, 1982.

[146] C. Steinbruchel, "A new method for analyzing Langmuir probe data and the determination of ion densities and etch yields in an etching plasma," *Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films,* vol. 8, no. 3, pp. 1663-1667, 1990.

[147] I. Langmuir, and H. M. Mott-Smith, "Studies of electric discharges in gases at low pressures," *General Electric Rev.,* vol. 27, pp. 449-455, 1924.

[148] M. B. Hopkins, "Langmuir probe measurements in the gaseous electronics conference rf reference cell," *J. Re. Natl. Inst. Stand. Technol.,* vol. 100, no. 4, pp. 415-425, 1995.

[149] M. B. Hopkins, and W. G. Graham, "Langmuir probe technique for plasma parameter measurement in a medium density discharge," *Review of Scientific Instruments,* vol. 57, no. 9, pp. 2210, 1986.

[150] V. Pletnev, and J. G. Laframboise, "Current collection by a cylindrical probe in a partly ionized, collisional plasma," *Physics of Plasmas,* vol. 13, no. 7, 2006.

[151] E. V. Shun'ko, *Langmuir Probe in Theory and Practice* Boca Raton, FL: Universal Publishers, 2009.

[152] D. Aktas, "Enhancement of PROBEPIC Code and Application to Langmuir Probes," M.S. Thesis, M.A.B.E. Dept., The University of Tennessee, Knoxville, TN., 2010.

[153] J. B. Hoag, and S. A. Korf, *Electron and Nuclear Physics*, New York: Van Norstrand Co. Inc., 1952.

[154] N. Hershkowitz, "How Langmuir Probes Work," *Plasma Diagnostics*, O. Auciello and D. L. Flamm, eds., pp. 1-72, Boston: Academic Press, 1989.

[155] D. Aktas, and T. M. Moeller, "Implementing Thin Sheath Theory for Langmuir Probe PIC Simulations," *IEEE Transactions on Plasma Science*, 2015.

[156] D. Aktas, "Simulations of a Langmuir Probe," in 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, Nashville, TN., 2012.

[157] W. L. Holtgreven, *Plasma Diagnostics*, New York: American Institute of Physics, 1995.

[158] R. L. Merlino, "Understanding Langmuir probe current-voltage characteristics," *American Journal of Physics,* vol. 75, no. 12, pp. 1078, 2007.

[159] L. Schott, "Electrical probes," *Plasma Diagnostics*, W. Lochte-Holtgreven, ed., pp. 668-731, New York: AIP Press, 1995.

[160] J. D. Swift, and M. J. R. Schwar, *Electrical probes for plasma diagnostics*: lliffe Books, 1970.

[161] D. Hastings, and H. Garrett, *Spacecraft-Environment Interactions*, Cambridge Atmospheric and Space Science Series, Cambridge: Cambridge University Press, 1996.

[162] W. R. Harper, *Contact and Frictional Electri¯cation*, Monographs on the Physics and Chemistry of Materials, London: Oxford University Press, 1967.

[163] J. Lowell, "Contact electrification of metals," *Journal of Physics D: Applied Physics,* vol. 8, pp. 53, 1975.

[164] É. Coggiola, and A. Soubeyran, "Mesothermal plasma flow around a negatively wake side biased cylinder," *Journal of Geophysical Research: Space Physics,* vol. 96, no. A5, pp. 7613–7621, 1991.

[165] G. J. I. Katz, V. Davis, T. Morton, and D. Ferguson, "Current collection by a large Langmuir probe in a mesothermal (ram) plasma," *Measurement Techniques in Space Plasmas: Particles*, Geophysical Monograph Series R. F. P. et.al., ed., pp. 37-41, Washington, D.C.: AGU, 1998.

[166] B. Roberts, *Solar System Magnetic Fields*, E. Priest, ed., p. 37: D. Reidel Publishing Company, 1985.

[167] A. Barjatya, "Langmuir Probe Measurements In The Ionosphere," Ph.D. thesis, Department of Electrical Engineering, Utah State University, Logan, UT, 2007.

[168] L. Schott, *Plasma diagnostics using electrical probes*, New York: John Wiley and Sons, Inc., 1971.

[169] A. Piel, M. Hirt, and C. T. Steigies, "Plasma diagnostics with Langmuir probes in the equatorial ionosphere: I. The influence of surface contamination," *Journal of Physics D: Applied Physics,* vol. 34, pp. 2643-2649, 2001.

[170] K. I. Oyama, "A systematic investigation of several phenomena associated with contaminated Langmuir probes," *Planetary and Space Science,* vol. 24, pp. 183, 1976.

[171] M. Wahlstrm, E. Johansson, E. Veszelei, P. Bennich, M. Olsson, and S. Hogmark, "Improved Langmuir probe surface coatings for the Cassini satellite," *Thin Solid Films,* vol. 220, pp. 315, 1992.

[172] W. E. Amatucci, P. W. Schuck, D. N. Walker, P. M. Kintner, S. Powell, B. Holbeck, and D. Leonhardt, "Contamination-free sounding rocket Langmuir probe," *Reviews of Scientific Instruments,* vol. 72, no. 4, pp. 2052, 2001.

[173] W. E. Amatucci, M. E. Koepke, T. E. Sheridan, M. J. Alport, and J. J. C. III, "Self cleaning Langmuir probe," *Reviews of Scientific Instruments,* vol. 68, no. 5, pp. 1253, 1993.

[174] D. Strele, M. Koepke, R. Schrittwieser, and P. Winkler, "Simple heatable Langmuir probe for alkali plasmas," *Review of Scientific Instruments,* vol. 68, no. 10, pp. 3751, 1997.

[175] E. P. Szuszczewicz, "Area influences and floating potentials in Langmuir probe measurements," *Journal of Applied Physics,* vol. 43, no. 3, pp. 874, 1972.

[176] J. G. Laframboise, and L. J. Sonmor, "Current collection by probes and electrodes in space magnetoplasmas: A review," *Journal of Geophysical Research,* vol. 98, pp. 337-357, 1993.

[177] F. F. Chen, and J. P. Chang, "Plasma Diagnostics," *Lecture Notes on Principles of Plasma Processing*, pp. 75-97, New York: Springer Science by Kluwer Academic, 2003.

[178] R. V. Kennedy, and J. E. Allen, "The floating potential of spherical probes and dust grains. II: Orbital motion theory," *Journal of Plasma Physics,* vol. 69, no. 6, pp. 485-506, 2003.

[179] I. G. Mikkellides, I. Katz, D. M. Goebel, and J. E. Polk, "Hollow Cathode Theory and Experiment. II. A Two-Dimensional Theoretical Model of the Emitter Region," *Journal of Applied Physics,* vol. 98, no. 11, 2005.

[180] B. Lipschultz, I. H. Hutchinson, B. LaBombard, and A. Wan, "Electric probes in plasmas," *J. Vac. Sci. Technol. A,* vol. 4, no. 3, pp. 1810, 1986.

[181] I. Katz, I. G. Mikellides, D. M. Goebel, and J. Polk, E., "Insert Heating and Ignition in Insert-Gas Hollow Cathodes," *IEEE Transactions on Plasma Science,* vol. 36, no. 5, pp. 2199-2206, 2008.

[182] W. G. Tighe, K. Chien, D. M. Goebel, and R. T. Longo, "Hollow Cathode Emission and Ignition Studies at L-3 ETI," in 30th International Electric Propulsion Conference, Florence, 2007.

[183] I. G. Mikeelides, I. Katz, D. M. Goebel, and J. E. Polk, "Model of a Hollow Cathode Insert Plasma," in 40th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, 2004.

[184] M. M. Turner, "Kinetic properties of particle-in-cell simulations compromised by Monte Carlo collisions," *Physics of Plasmas,* vol. 13, no. 3, 2006.

[185] D. Aktas, and T. M. Moeller, "Use of PROBEPIC for Particle Simulations of Langmuir Probes in Non-Maxwellian Plasmas," *IEEE Transactions on Plasma Science*, 2015. Manuscript is under review.

[186] T. H. Chung, Y. M. Shin, and D. C. Seo, "Comparison of Two Methods of Interpretation of LangmuirProbe Data for an Inductively Coupled Oxygen Plasma," *Contributions to Plasma Physics,* vol. 46, no. 5-6, pp. 348-353, 2006.

[187] D. Keefer, and V. Semak, "Measurements of radial and axial distributions of ion thruster plasma parameters using a Langmuir probe," 1996.

[188] G. Lapenta, "Automatic Adaptive Multi-Dimensional Particle In Cell," *Advanced Methods for Space Simulations*, pp. 61-76, Tokyo: Terrapub, 2007.

[189] B. Crowley, and S. Dietrich, "A Langmuir probe system incorporating the Boyd-Twiddy method for EEDF measurement applied to an inductively coupled plasma source," *IOP Science: Plasma Sources Sci. Technol.,* vol. 18, no. 014010, 2009.

[190] A. El Saghir, "Improvements to EEDF Analysis from Langmuir Probes Using Integral Methods," Doctor of Philosophy, Nuclear Engineering, North Carolina State University, North Carolina, 2011.

[191] F. Skiff, H. Gunell, A. Bhattacharjee, C. S. Ng, and W. A. Noonan, "Electrostatic degrees of freedom in non-Maxwellian plasma," *Physics of Plasmas,* vol. 9, no. 5, pp. 1931, 2002.

[192] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd ed.: Cambridge University Press, 2007.

[193] M. J. Druyvesteyn, and F. M. Penning, "The Mechanism of Electrical Discharges in Gases of Low Pressure," *Reviews of Modern Physics,* vol. 12, no. 2, pp. 87-174, 1940.

[194] H. Li, "Measurements of electron energy distribution function and neutral gas temperature in an inductively coupled plasma," Master's Thesis, Contributions to Plasma Physics, University of Saskatchewan, Saskatoon, Saskatchewan, 2006.

[195] I. D. Sudit, and F. F. Chen, "RF Compensated Probes for High Density Discharges," *Plasma Sources Science and Technology,* vol. 3, pp. 162-168, 1994.

[196] V. A. Godyak, R. B. Piejak, and B. M. Alexandrovich, "Probe diagnostics of non-Maxwellian plasmas," *Journal of Applied Physics,* vol. 73, no. 8, pp. 3657, 1993.

[197] M. Li, S. K. Dew, and M. J. Brett, "Effects of electron distribution functions on the floating potential of particles in the plasma: thin plasma sheaths," *J. Phys. D: Appl. Phys.,* vol. 32, no. 16, pp. 2056-2059, 1999.

[198] M.-B. Kallenrode, "Space Physics-An Introduction to Plasmas and Particles in the Heliosphere and the Magnetosphere," *Space Science Reviews,* vol. 88, no. 3-4, pp. 612, 1999.

[199] S. Robertson, Z. Sternovsky, and B. Walch, "Reduction of asymmetry transport in the annular Penning trap," *Physics of Plasmas,* vol. 11, no. 5, pp. 1753, 2004.

[200] S. Knappmiller, S. Robertson, and Z. Sternovsky, "Method to find the electron distribution function from cylindrical probe data," *Physical Review,* vol. 73, no. 6, 2006.

[201] S. V. Singh, "Investigation of ICP RF discharges by means of a Langmuir probe," Ph.D. thesis, Fakultät für Physik und Astronomie, Fakultät für Physik und Astronomie der Ruhr-Universität Bochum, Bochum, 2004.

[202] N. J. Behlman, "Electron Energy Distribution Measurements In The Plume Region Of A Low Current Hollow Cathode," M.S. thesis, Mechanical Engineering, Worcester Polytechnic Institute, Worcester, MA., 2009.

[203] D. J. Asselin, "Characterization of the near-plume region pf a low-current hollow cathode," Masters of Science, Mechanical Engineering, Worcester Polytechnic Institute, Worcester, MA., 2011.

[204] S. C. Kim, "Calculations of Low-Reynolds-Number Resistojet Nozzles," *Journal of Spacecraft and Rockets,* vol. 13, pp. 259--264, 1994.

[205] I. D. Boyd, P. F. Penko, D. L. Meissner, and K. J. DeWitt, "Experimental and Numerical Investigations of Low-Density Nozzle and Plume Flows of Nitrogen," *AIAA,* vol. 30, no. 10, pp. 2453-2461, 1992.

[206] I. D. Boyd, D. B. VanGilder, and E. J. Beiting, "Numerical and Experimental Investigations of Rarefied Flow in a Small Nozzle," *AIAA,* vol. 34, pp. 2320-2326, 1996.

[207] D. Gibbon, I. Coxhill, D. Nicolini, R. Correia, and J. Page, "The Design, Development and In-flight Operation of a Water Resistojet Micropropulsion System," *AIAA*, no. 2004-3798, 2004.

[208] J. C. Tannehill, D. A. Anderson, and R. H. Pletcher, *Computational Fluid Mechanics and Heat Transfer*, Boca Raton, FL: Taylor and Francis, 1997.

[209] C. A. J. Fletcher, *Computational Techniques for Fluid Dynamics*, Berlin: Springer-Verlag, 1991.

[210] C. K. Birdsall, "Particle-in-cell charged-particle simulations, plus Monte Carlo collisions with neutral atoms, PIC-MCC," *IEEE Transactions on Plasma Science,* vol. 19, no. 2, 1991.

[211] J. J. Szabo, M. Martinez-Sanchez, and O. Batishchev, "Particle-in-Cell Modeling of Thruster with Anode Layer (TAL)," in IEPC 99-100, 26th International Electric Propulsion Conference, Kitakyushu, Japan, 1999.

[212] J. D. Callen, "Collective Plasma Phenomena," *Fundamentals of Plasma Physics*, University of Wisconsin, Madison, 2006.

[213] B. M. Annaratone, M. Glier, T. Stuffler, M. Raif, H. M. Thomas, and G. E. Morfill, "The plasma-sheath boundary near the adaptive electrode as traced by particles," *IOP Science,* vol. 5, no. 1, pp. 92, 2003.

[214] J. Wang, and J. Brophy, "3-D Monte Carlo Particle-in-Cell Simulations of Ion Thruster Plasma Interactions."

[215] A. Friedman, S. E. Parker, S. L. Ray, and C. K. Birdsall, "Multi-scale Particle-in-Cell Plasma Simulation," *J. Comput. Phys.,* vol. 96, no. 1, 1991.

[216] J. C. Adam, A. G. Serveniere, and A. B. Langdon, "Electron sub-cycling in particle simulation of plasmas," *J. Comput. Phys.,* vol. 47, pp. 229-244, 1982.

[217] B. I. Cohen, T. A. Brengle, D. B. Conley, and R. P. Freis, "An orbit averaged particle code," *J. Comput. Phys.,* vol. 38, pp. 45-63, 1980.

[218] J. Boerner, and I. D. Boyd, "Comparison of Detailed Electron Fluid Model Formulations for Non-equilibrium Plasma Simulations," in 38th AIAA Plasmadynamics and Lasers Conference, Miami, FL, 2007.

[219] R. W. Hockney, and J. W. Eastwood, *Computer Simulation Using Particles*, Philadelphia: Adam Hilger, 1988.

[220] R. Marchand, J. K. Burchill, and D. J. Knudsen, "Modelling Electrostatic Sheath Effects on Swarm Electric Field Instrument Measurements," *Space Science Reviews,* vol. 156, no. 1-4, pp. 73-87, 2010.

[221] F. F. Chen, "Langmuir probe analysis for high density plasmas," *Physics of Plasmas,* vol. 8, no. 3029, 2001.

[222] F. Assous, T. P. Dulimbert, and J. Segre, "A new method for coalescing particles in PIC codes," *Journal of Computational Physics,* vol. 187, pp. 550-571, 2003.

[223] H. B. Smith, "Computational Studies of an Asymmetric rf Plasma using Particle-in-cell Techniques," Research School of Physical Science and Engineering, Australian National University, 1994.

[224] W. Gautschi, "A Computational Procedure for Incomplete Gamma Functions," *ACM Transections on Mathematical Software,* vol. 5, no. 4, 1979.

[225] C. Lanczos, "A Precision Approximation of the Gamma Function," *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis,* vol. 1, pp. 86-96, 1964.

[226] T. Holstein, "Energy Distribution of Electrons in High Frequency Gas Discharges," *Physical Review,* vol. 70, pp. 367, 1946.

[227] J. Gruenwald, D. Tskhakaya, J. Kovačič, M. Čerček, T. Gyergyek, C. Ionita, and R. Schrittwieser, "Comparison of measured and simulated electron energy distribution functions in low-pressure helium plasmas," *Plasma Sources Science and Technology,* vol. 22, no. 1, pp. 015023, 2013.

[228] E. F. Jaeger, L. A. Berry, and D. B. Batchelor, "Characteristics of a cylindrical Langmuir probe of finite length," *Journal of Applied Physics,* vol. 69, no. 10, pp. 6918, 1991.

[229] D. N. Walker, R. F. Fernsler, D. D. Blackwell, and W. E. Amatucci, *Plasma Diagnostics with a High Aspect Ratio Cylindrical Probe Used as an Impedance Probe*, Naval Research Laboratory, Washington, DC., 2011.

[230] I. Langmuir, and H. M. Mott-Smith, "The Theory of Collectors in Gaseous Discharges," *Physical Review,* vol. 28, pp. 727, 1926.

[231] C. Weyde, "Plasma parameters from the Rosetta LAP instrument," Masters of Thesis, Space Physics, Swedish Institute of Space Physics, Uppsala, Sweden, 2006.

[232] B. C. Inc., "Hall Effect Thruster Systems," http:// www.busek.com/technologies__hall.htm.

[233] W. Baumjohann, and R. A. Treumann, *Basic Space Plasma Physics*, London, UK: Imperial College Press, 2004.

Appendix

A.1 Single Charged Particle Motion

Plasma transport can be described as [88]:

- single particle motion gives the step size
- collision frequency tells us how often steps are taken.

If the magnetic field geometry is not optimized carefully, particles have trajectories causing them to leave the system without collisions; high-energy particles are the most susceptible and damaging. The charged particles move freely along a constant magnetic field, but any velocity perpendicular to the field causes them to orbit around the field lines. When the magnetic field strength isn't constant, it changes parallel velocity of the charged particle and it is velocity along field line to change. Finally, a force perpendicular to the field lines will cause the particle to move perpendicular to both the force and the magnetic field lines [88].

The Lorentz force describes the forces on a charged particle moving in the presence of an electric field and magnetic field [88]:

$$\vec{F} = q\left(\vec{E} + \vec{\upsilon} \times \vec{B}\right), \tag{A.1}$$

where $\vec{F}$ is the force on the particle, q is the charge on the particle, $\vec{E}$ is the electric field, $\vec{\upsilon}$ is the velocity of the particle, and $\vec{B}$ is the magnetic field. Newton's second law of motion is another fundamental equation as shown below [88]:

$$\vec{F} = m\frac{d\vec{\upsilon}}{dt}, \tag{A.2}$$

where $\vec{F}$ is force on the particle, m is mass of the particle, and t is time, and $\vec{\upsilon}$ is velocity of the particle. The equation of motion for a charged particle q, under the action of the Lorentz force $\vec{F}$ due to electric $\vec{E}$ and magnetic induction $\vec{B}$ fields can be written as in the following form [88]:

$$\frac{d\vec{p}}{dt} = \vec{F} = q\left(\vec{E} + \vec{\upsilon} \times \vec{B}\right), \tag{A.3}$$

where $\vec{p}$ represents the momentum of particle. This equation is relativistically correct if we take

$$\vec{p} = \gamma m\vec{\upsilon}, \tag{A.4}$$

where m is the rest mass of the particle and γ is the Lorentz factor defined by

$$\gamma = \left(1 - \frac{\upsilon^2}{c^2}\right)^{-1/2}, \tag{A.5}$$

where c is the speed of light in vacuum [88]. In relativistic case, (A.3) can be written in the form of [88]

$$\gamma m\frac{d\vec{\upsilon}}{dt}+q\left(\frac{\vec{\upsilon}}{c^2}\right)\left(\vec{\upsilon}\cdot\vec{E}\right)=q\left(\vec{E}+\vec{\upsilon}\times\vec{B}\right). \tag{A.6}$$

Note that the time rate of change of the total relativistic energy $(U=\gamma mc^2)$ is given by $dU/dt=q\left(\vec{\upsilon}\cdot\vec{E}\right)$ and that $d\vec{p}/dt=d\left(U\vec{\upsilon}/c^2\right)/dt$; the term υ^2/c^2 is negligible compared to unity. For $\upsilon^2/c^2\ll 1$, $\gamma\simeq 1$ and m can be considered constant (independent of υ), so that (A.6) is reduced to the following nonrelativistic expression [88, 233]:

$$m\frac{d\vec{\upsilon}}{dt}=q\left(\vec{E}+\vec{\upsilon}\times\vec{B}\right). \tag{A.7}$$

If the velocity obtained from equation (A.7) does not satisfy the condition $\upsilon^2\ll c^2$, the corresponding result is not valid and the relativistic expression (A.6) is used instead of (A.7). Relativistic effects become important for highly energetic particles (e.g., a $1MeV$ proton has a velocity of $1.4\times10^7\,m/s$, with $\upsilon^2/c^2\simeq 0.002$). It is assumed that the restriction $\upsilon^2\ll c^2$, implicit in equation (A.7), is not violated. Note that all radiation effects are also neglected [88].

Based on the given information above, we can consider series examples of different external magnetic and electric field configurations. These configurations [88, 233] are listed below.

A.1.1 Case with $\vec{E}=\text{const}$ and $\vec{B}=0$

The equation of motion for a particle with mass m and charge q is [88]

$$m\frac{d\vec{\upsilon}}{dt}=q\vec{E}, \tag{A.8}$$

whose solution for the particle velocity [88]

$$\vec{\upsilon}=\frac{q}{m}\left(t-t_o\right)\vec{E}+\vec{\upsilon}_o, \tag{A.9}$$

which represents a uniformly accelerated particle motion along a constant $\vec{E}$ field. The particles with charges of different sign move in opposite directions resulting in electric currents [88]:

$$\vec{j}=q_e\vec{\upsilon}_e+q_p\vec{\upsilon}_p\ . \tag{A.10}$$

A.1.2 Case with $\vec{E}=0$ and $\vec{B}=\text{const}$

Assume that $\vec{E}=\vec{g}=0$ and $\vec{B}$ is constant. The equation of motion of a charged particle in a constant magnetic field is described by [88]

$$m\frac{d\vec{\upsilon}}{dt}=q\vec{\upsilon}\times\vec{B}\,, \tag{A.11}$$

which yields a condition of

$$\vec{\upsilon}\cdot\frac{d\vec{\upsilon}}{dt}=0 \qquad \Rightarrow \qquad |\vec{\upsilon}|=\upsilon=\text{const.} \tag{A.12}$$

i.e., constancy of the velocity vector intensity [88]. This means that the magnetic field does no work on the particle and charged particle does not gain any kinetic energy form the considered magnetic field [88].

Decomposing the velocity vector into two components [88],

$$\vec{\upsilon}=\vec{\upsilon}_{\parallel}+\vec{\upsilon}_{\perp}\,, \tag{A.13}$$

in directions parallel and normal to $\vec{B}$, the equation of motion (A.11) reduces to two equations for each of the velocity components [88]:

$$m\frac{d\vec{\upsilon}_{\perp}}{dt}=q\vec{\upsilon}_{\perp}\times\vec{B} \qquad \text{and} \qquad \frac{d\vec{\upsilon}_{\parallel}}{dt}=0\,. \tag{A.14}$$

Velocity component $\vec{\upsilon}_{\parallel}$ along $\vec{B}$ remains constant in this case, i.e., unaffected by the presence of magnetic fields and its magnitude $\upsilon_{\parallel}$ is simply prescribed as the initial condition [88]. If $\vec{\upsilon}_{\parallel}=\text{const.}$, the same must also be true for the magnitude of the normal component $\vec{\upsilon}_{\perp}$ since $\vec{\upsilon}^2\equiv\vec{\upsilon}_{\parallel}^2+\vec{\upsilon}_{\perp}^2=\text{const.}$ as it is already obtained in equation (A.12). Therefore [88],

$$\vec{\upsilon}_{\perp},\vec{\upsilon}_{\parallel},\upsilon\equiv\sqrt{\vec{\upsilon}_{\perp},\vec{\upsilon}_{\parallel}}=\text{const.} \tag{A.15}$$

To find the shape of the trajectory the particle moves along, we take the Cartesian geometry with $\vec{B}=(0,0,B)$, $\vec{\upsilon}_{\perp}=(\upsilon_x,\upsilon_y,0)$ and $\vec{\upsilon}_{\parallel}=(0,0,\upsilon_z)$ and the equation (A.14) can be written as [88]:

$$m\frac{d\upsilon_x}{dt}=qB\upsilon_y, \qquad m\frac{d\upsilon_x}{dt}=qB\upsilon_x, \qquad m\frac{d\upsilon_x}{dt}=0\,. \tag{A.16}$$

After some elementary rearrangements, we obtain the following set of equations:

$$m\frac{d^2\upsilon_x}{dt^2} = -\omega_L^2\upsilon_x, \quad \upsilon_x^2 + \upsilon_y^2 = \upsilon_\perp^2, \quad \upsilon_z = \upsilon_\parallel, \tag{A.17}$$

where $\upsilon_\parallel$ and $\upsilon_\perp$ are constants given as initial conditions. Three velocity components follow from equation (A.17); they are given as [88]:

$$\begin{aligned} \upsilon_x &\equiv \frac{dx}{dt} = \upsilon_\perp \cos(\omega_L t) \\ \upsilon_y &\equiv \frac{dy}{dt} = -\upsilon_\perp \sin(\omega_L t) \\ \upsilon_z &\equiv \frac{dz}{dt} = \upsilon_\parallel , \end{aligned} \tag{A.18}$$

where

$$\omega_L \equiv \frac{qB}{m} \tag{A.19}$$

is known as Larmor frequency (also referred to as gyro or cyclotron frequency; see section 1.3.3.5). The following equations yield the particle trajectory equation after one time integration [88, 233]:

$$\begin{aligned} x - x_o &= r_L \sin(\omega_L t) \\ y - y_o &= r_L \cos(\omega_L t) \\ z - z_o &= \upsilon_\parallel t \end{aligned} \tag{A.20}$$

with

$$r_L \equiv \frac{\upsilon_\perp}{\omega_L}, \tag{A.21}$$

which is referred to as Larmor radius or gyroradius [88, 233] (also known as gyro-radius; see section 1.3.3.6).

The trajectory (A.20) is a helix along the z axis (i.e., in the direction of $\vec{B}$) with the pitch angle α given by the relation $\upsilon_\perp = \upsilon_\parallel \tan\alpha$. The vector of gyration angular velocity $\vec{\omega}_L$ follows directly from the equation (A.14) integrated over time:

$$m\int \frac{d\vec{\upsilon}_\perp}{dt} dt = q\int \vec{\upsilon}_\perp \, dt \times \vec{B} \quad \Rightarrow \quad \vec{\upsilon}_\perp = \vec{\omega}_L \times \vec{r}_\perp , \tag{A.22}$$

where

$$\vec{\omega}_L \equiv -\frac{q\vec{B}}{m}. \tag{A.23}$$

The charged particle motion is therefore a superposition of gyration (with the gyration radius and gyration frequency r_L and ω_L respectively) in a plane normal to magnetic field lines, and a uniform motion of the center of gyration, which is referred to as guiding center (GC), along the field lines. For GC, such charged particle motion is identified as the motion of its GC. According to equation (A.15), the positively charged particles gyrate about the magnetic field line in the clockwise direction while particles with a negative charge move in the opposite direction.

A.1.3 Case with Constant $\vec{E}$ and $\vec{B}$

Assume that both $\vec{E}$ and $\vec{B}$ are constant and non-zero. Then the dynamics of a charged particle is governed by the following equation [88]:

$$m\frac{d\vec{v}}{dt} = q\vec{E} + q\vec{v}\times\vec{B}\,, \tag{A.24}$$

which can be analyzed and solved in a way as done in Case A.1.2. We decompose $\vec{v}$ and $\vec{E}$ as

$$\vec{v} = \vec{v}_\perp + \vec{v}_\parallel, \qquad \vec{E} = \vec{E}_\perp + \vec{E}_\parallel, \tag{A.25}$$

and substitute these into equation (A.24) and obtain two equations for propagations parallel and normal to the magnetic field [88]:

$$m\frac{d\vec{v}_\parallel}{dt} = q\vec{E}_\parallel, \quad \text{and} \quad m\frac{d\vec{v}_\perp}{dt} = q\vec{E}_\perp + q\vec{v}_\perp\times\vec{B}\,. \tag{A.26}$$

The equations (A.26) shows that the particle is accelerated along the magnetic field by the parallel component of the electric field in the same way as it happens in Case A.1.1 [88]:

$$\vec{v}_\parallel = \frac{q}{m}\left(t - t_o\right)\vec{E}_\parallel + \vec{v}_o\,. \tag{A.27}$$

To solve the equation (A.11), we switch to a new frame of reference moving with some constant speed $\vec{v}_E$ in a direction normal to $\vec{B}$, so that

$$\vec{v}_\perp = \vec{v}_E + \vec{u}_\perp\,, \tag{A.28}$$

where $\vec{u}_\perp$ is the normal velocity component relative to the moving frame. The equation (A.26) now becomes [88]:

$$m\left(\frac{d\vec{u}_\perp}{dt} + \frac{d\vec{v}_E}{dt}\right) = q\vec{E}_\perp + q\vec{v}_E\times\vec{B} + q\vec{u}_\perp\times\vec{B}\,, \tag{A.29}$$

where $d\vec{\upsilon}_E/dt = 0$; since $\vec{\upsilon}_E$ has not been specified, we have first two terms on the right hand side of equation (A.29) cancel out [88, 233]:

$$q\vec{E}_\perp + q\vec{\upsilon}_E \times \vec{B} = 0 \qquad \Rightarrow \qquad \vec{\upsilon}_E = \frac{\vec{E}_\perp \times \vec{B}}{B^2} = \frac{\vec{E} \times \vec{B}}{B^2} . \tag{A.30}$$

The remaining part of equation (A.29) is then written as:

$$m\frac{d\vec{u}_\perp}{dt} = q\vec{u}_\perp \times \vec{B} , \tag{A.31}$$

which is the same type of equation as (A.14) in Case A.1.2. The velocity component $\vec{u}_\perp$ describes an orbiting motion with Larmor frequency ω_L and Larmor radius r_L around magnetic field lines as viewed in the frame of reference moving with a constant velocity $\vec{\upsilon}_E$. The particle velocity components in the rest frame are given by equations (A.27) and (A.28) indicating an accelerated GC motion along magnetic field lines with $\vec{\upsilon}_\parallel$ and a superimposed perpendicular drift motion with

$$\vec{\upsilon}_E = \frac{\vec{E} \times \vec{B}}{B^2} . \tag{A.32}$$

The drift, also called $\vec{E} \times \vec{B}$ drift, is a charge independent and therefore induces no electric currents as both the positive and negative charges move in the same direction as seen in the equation (A.32) for $\vec{\upsilon}_E$ [88, 233].

A.1.4 Case with Constant $\vec{F}$ and $\vec{B}$

If an additional constant external force $\vec{F}$ acts on a charged particle moving in a constant magnetic field, then the equation of motion is written as [88]:

$$m\frac{d\vec{\upsilon}}{dt} = \vec{F} + q\vec{\upsilon} \times \vec{B} . \tag{A.33}$$

Comparing this equation with the equation of motion (A.24) in Case A.1.3, the only divergence between them is the replacement of $q\vec{E}$ by $\vec{F}$. This means that all derivations performed in Case A.1.3 can be repeated here by taking $\vec{F}/q$ instead of $\vec{E}$ [88]. A charged particle spirals around magnetic field lines, and its GC velocity has two components describing an accelerated motion along the magnetic field lines due to $\vec{F}_\parallel$, and a drift motion across the field lines with velocity $\vec{\upsilon}_F$ [88, 233]:

$$\vec{\upsilon}_F = \frac{\vec{F}_\perp \times \vec{B}}{qB^2} = \frac{\vec{F} \times \vec{B}}{qB^2}, \tag{A.34}$$

whose orientation is a charge dependent [88]. This drift is also called force drift, and produces electric currents as charges with opposite signs drift in opposite directions. One of the interesting examples of a force drift is the gravitational drift occurring in the presence of a uniform gravitational field $\vec{g}$, when $\vec{F} = m\vec{g}$. In this case, the gravitational drift velocity $\vec{\upsilon}_g$ follows from equation (A.34) as [88]:

$$\vec{\upsilon}_g = \frac{m\vec{g} \times \vec{B}}{qB^2}. \tag{A.35}$$

In astrophysical plasmas, the drift contributes to the formation of ring currents among other things [233].

A.1.5 Case with $\vec{E} = \vec{E}(t)$ and $\vec{B} = \text{const}$

By taking the uniform electric field from Case A.1.3 varies in time, $\vec{E} = \vec{E}(t)$, we can examine the effects on the plasma particle motions. If time variation is assumed small on the time scale of one gyration period $\tau_L = 2\pi/\omega_L$ $(q > 0)$ [233], and only the first time derivative is retained in series expansions meaning that $d\vec{E}/dt \approx \text{const.}$, then going back to the Case A.1.3 we can repeat the whole analytical procedure up to equation (A.29) [233]:

$$m\left(\frac{d\vec{u}_\perp}{dt} + \frac{d\vec{\upsilon}_E}{dt}\right) = q\vec{E}_\perp + q\vec{\upsilon}_E \times \vec{B} + q\vec{u}_\perp \times \vec{B}. \tag{A.36}$$

The term $d\vec{\upsilon}_E/dt$ remains in the equation and its presence represents the effects of a slowly time varying electric field. This is the same as in Case A.1.3, we go to a new coordinate system moving with drift velocity $\vec{\upsilon}_E$ given by the equation (A.30)

$$\vec{\upsilon}_E = \frac{\vec{E} \times \vec{B}}{B^2}, \tag{A.37}$$

which reduces equation (A.36) to:

$$m\frac{d\vec{u}_\perp}{dt} = -m\frac{d\vec{\upsilon}_E}{dt} + q\vec{u}_\perp + \vec{B}. \tag{A.38}$$

This equation is equivalent to equation (A.33) with the external force $\vec{F}$ given by:

$$\vec{F} = -m\frac{d\vec{\upsilon}_E}{dt}, \tag{A.39}$$

which promotes an additional drift motion of the GC with the velocity $\vec{\upsilon}_F \equiv \vec{\upsilon}_P$ called the polarization drift shown as [233]:

$$\vec{\upsilon}_P = -\frac{m}{qB^2}\frac{d\vec{\upsilon}_E}{dt}\times\vec{B} = -\frac{m}{qB^2}\frac{d}{dt}\left(\frac{\vec{E}_\perp \times \vec{B}}{B^2}\right)\times\vec{B} \tag{A.40}$$

or

$$\vec{\upsilon}_P = \frac{m}{qB^2}\frac{d\vec{E}_\perp}{dt}\,. \tag{A.41}$$

The polarization drift $\vec{\upsilon}_P$ is charge sign dependent and therefore it induces electric currents in plasmas. The total drift velocity $\vec{\upsilon}_D$ is given by [233]:

$$\vec{\upsilon}_D = \vec{\upsilon}_P + \vec{\upsilon}_E \tag{A.42}$$

with the following ordering [233]:

$$\frac{\upsilon_P}{\upsilon_E} \equiv \frac{\left|\frac{m}{qB^2}\frac{d\vec{E}_\perp}{dt}\right|}{\left|\frac{\vec{E}_\perp \times \vec{B}}{B^2}\right|} \sim \frac{\tau_L}{E_\perp}\frac{dE_\perp}{dt} \ll 1\,. \tag{A.43}$$

A.1.6 Case with Non-Uniform Magnetic Field: $\vec{B} = (0,0,B(x))$

Consider a charged particle with $q > 0$ moving in a magnetic field with straight lines parallel to the z - axis whose density varies in the x - direction: $\vec{B} = B(x)\hat{e}_z$. Let this x - dependence be sufficiently weak as to allow the first order series expansion as an acceptable approximation [233]

$$B(x) = B(0) + x\left.\frac{dB}{dx}\right|_{x=0} \tag{A.44}$$

and $B(x)$ changes over the distance of the gyration radius r_L is small relative to $B(x)$: $r_L dB/dx \ll B$. We consider only the normal component of the particle velocity vector $\vec{\upsilon}$ as the parallel component remains unaffected by the $\vec{B}$ field; we take $\vec{\upsilon}_\parallel = 0$ by the initial condition so that $\vec{\upsilon}\cdot\vec{B} = 0$. The particle velocity has only two components:

$$\upsilon_x = \frac{dx}{dt} \quad \text{and} \quad \upsilon_y = \frac{dy}{dt}\,, \tag{A.45}$$

and the vector equation of motion (A.11) can be expressed as a system of two scalar equations in the following way [233]:

$$\begin{aligned} m\frac{d\upsilon_x}{dt} &= q\left(B(0) + x\left.\frac{dB}{dx}\right|_0 \right)\frac{dy}{dt} \\ m\frac{d\upsilon_y}{dt} &= -q\left(B(0) + x\left.\frac{dB}{dx}\right|_0 \right)\frac{dx}{dt}. \end{aligned} \tag{A.46}$$

The trajectory of a particle motion described by the equation (A.46) that would be circular with the radius r_L and the gyration period τ_L if the magnetic field were uniform: $dB/dx = 0$. The presence of a weak magnetic field non-uniformity dB/dx slightly modifies the trajectory in the sense that it is not a closed circle any more, and the particle position shifts by some Δy along the y-axis after each gyration time τ_L. This results into a drift motion along the y-axis known as the magnetic gradient drift with speed $\vec{\upsilon}_G = \Delta y/\tau_L$. As the particle motion remains periodic in the x-direction, we have:

$$m\int_0^{\tau_L} \frac{d\upsilon_x}{dt} dt = m\upsilon_x(\tau_L) - m\upsilon_x(0) \approx 0 , \tag{A.47}$$

which yields:

$$B(0)\int_0^{\tau_L} \frac{dy}{dt} + q\left.\frac{dB}{dx}\right|_0 \int_0^{\tau_L} x\frac{dy}{dt} = 0 \tag{A.48}$$

or

$$qB(0)\Delta y - \pi r_L^2 \left.\frac{dB}{dx}\right|_0 = 0 \quad \Rightarrow \quad \Delta y = \frac{\pi r_L^2}{B}\frac{dB}{dx} , \tag{A.49}$$

with the expressions (A.20) for $x(t)$ and $y(t)$ taken into account. The gradient drift velocity is now:

$$\vec{\upsilon}_G \equiv \frac{\Delta y}{\tau_L}\hat{e}_y = \frac{\omega_L r_L^2}{2B}\frac{dB}{dx}\hat{e}_y = \frac{m\upsilon^2}{2qB^2}\frac{dB}{dx}\hat{e}_y , \tag{A.50}$$

or in a full vector form:

$$\vec{\upsilon}_G = \frac{m\upsilon^2}{2qB^3}\vec{B}\times\nabla B . \tag{A.51}$$

As the magnetic field gradient drift (A.51) depends on the sign of charge q it includes electric currents in the plasmas (e.g., in planetary magnetospheres [233]).

A.1.7 Case of Stationary $\vec{B}$ Field with Curved and Parallel Field Lines

A uniform and stationary magnetic field configuration studied in Case A.1.2 is modified in this section by adding a small curvature to its field lines that cause corrections to particle motion of the first order of smallness [233]: The curved magnetic field lines are parallel and uniformly distributed through any perpendicular plane. The magnetic field intensity B does not change in the direction along the magnetic field vector $\vec{B} = B\hat{e}_s$.

The basic type of a plasma particle motion in this case is a gyration with its GC moving along slightly curved magnetic field lines with velocity $\vec{v}_{\parallel} = v_{\parallel}\hat{e}_s$, which introduces a centrifugal force $\vec{F}_C$. This results into a force drift described in Case A.1.4 with the drift velocity [233]:

$$\vec{v}_C = \frac{\vec{F}_C \times \vec{B}}{qB^2}, \tag{A.52}$$

which is referred to as the centrifugal drift. The explicit expression for the centrifugal force and the related drift follows from [233]:

$$\vec{F}_C \equiv -m\frac{d\vec{v}_{\parallel}}{dt} = -mv_{\parallel}\frac{d\hat{e}_s}{dt} = -mv_{\parallel}\frac{d\hat{e}_s}{ds}\frac{ds}{dt} = -\frac{mv_{\parallel}^2}{B}\frac{d\vec{B}}{ds}, \tag{A.53}$$

where:

$$v_{\parallel} = \frac{ds}{dt}, \tag{A.54}$$

while the derivative of the magnetic field along the curved filed line s can be written as:

$$\frac{d\vec{B}}{ds}(\hat{e}\cdot\nabla)\vec{B} = \frac{1}{B}(\vec{B}\cdot\nabla)\vec{B}. \tag{A.55}$$

Equation (A.53) for the centrifugal force $\vec{F}_C$ then takes its final form:

$$\vec{F}_C = -\frac{mv_{\parallel}^2}{B^2}(\vec{B}\cdot\nabla)\vec{B}, \tag{A.56}$$

while the centrifugal drift velocity equation (A.52) becomes:

$$\vec{v}_C = \frac{mv_{\parallel}^2}{qB^4}\vec{B}\times\left[(\vec{B}\cdot\nabla)\vec{B}\right]. \tag{A.57}$$

Equation (A.57), the centrifugal drift $\vec{\upsilon}_C$, is charge dependent and therefore produces electric currents. These effects of curved magnetic fields are commonly present in the ring current formation mechanisms in the planetary magnetospheres.

A.1.8 Stationary $\vec{B}$ Field with Slightly Convergent Field Lines

The magnetic field lines from Case A.1.2 are slightly convergent in this case and axially symmetric with respect to the z-axis. It is more convenient to apply a cylindrical coordinate system $(\hat{e}_r, \hat{e}_\theta, \hat{e}_z)$ oriented along the axis of symmetry, so that the considered weakly convergent magnetic field $\vec{B}$ is given by [233]

$$\vec{B} = (B_r(r,z), 0, B_z(r,z)) \quad \text{with} \quad |B_r(r,z)| \ll |B_z(r,z)| . \tag{A.58}$$

The positively charged particle motion in such a magnetic field configuration retains two basic properties from Case A.1.2, where the magnetic field is uniform [233]: The GC motion with velocity $\vec{\upsilon}_\parallel$ along a magnetic field line, it is the line of the axis of symmetry, i.e., the z-axis, and a gyration with velocity $\vec{\upsilon}_\perp$ about the same field line. The difference here is that the uniform $\vec{B}$ field is considered in Case A.2.1 has no effect on $\vec{\upsilon}_\parallel$ while the magnetic field with slightly convergent field lines exerts a parallel force $\vec{F}_\parallel$ along the z-axis, which effects the GC velocity $\vec{\upsilon}_\parallel$ [233]:

$$\vec{F}_\parallel \equiv F_z \hat{e}_z = qB_r \vec{\upsilon}_\perp \times \hat{e}_r . \tag{A.59}$$

To calculate the radial magnetic field component B_r we start from Gauss' law:

$$\nabla \cdot \vec{B} = \frac{1}{r}\frac{\partial}{\partial r}(rB_r) + \frac{\partial}{\partial z}B_z = 0 \quad \Rightarrow \quad \frac{\partial}{\partial r}(rB_r) = -r\frac{\partial}{\partial z}B_z , \tag{A.60}$$

we then perform an integration over the coordinate r between $r = 0$ and $r = r_L$ with conditions $\partial B_z / \partial z \approx \text{const}$ and $B_r(0,z) = 0$, which yields:

$$r_L B_r(r_L, z) = -\frac{\partial B_z}{\partial z}\int_0^{r_L} r\,dr \tag{A.61}$$

or

$$B_r(r_L, z) = -\frac{1}{2} r_L \frac{\partial B_z}{\partial z} . \tag{A.62}$$

Taking equation (A.22) for the gyration velocity, i.e., [233]:

$$\vec{\upsilon}_{\perp} = \vec{\omega}_L \times \vec{r}_L = -\omega_L r_L \hat{e}_z \times \hat{e}_r = \upsilon_{\perp} \hat{e}_r \times \hat{e}_z \,, \tag{A.63}$$

from the equation (A.59), we obtain the following expression for $\vec{F}_{\parallel}$ [233]:

$$\vec{F}_{\parallel} = -\frac{1}{2} q r_L \upsilon_{\perp} \frac{\partial B_z}{\partial z} (\hat{e}_r \times \hat{e}_z) \times \hat{e}_r = -\frac{m\upsilon_{\perp}^2}{2B_z} \frac{\partial B_z}{\partial z} \hat{e}_z \tag{A.64}$$

or

$$\vec{F}_{\parallel} \approx -\frac{W_{\perp}}{B} \frac{dB}{dz} \hat{e}_z \quad \text{with} \quad W_{\perp} \equiv \frac{1}{2} m\upsilon_{\perp}^2 \,, \tag{A.65}$$

where $W_{\perp}$ is the particle kinetic energy of orbital motion, and with the assumption $B_z \gg B_r$ equivalent to $B \approx B_z$ taken into account.

According to equation (A.65), the parallel force $\vec{F}_{\parallel}$ is oriented in the direction of decreasing of the magnetic field intensity. This force tends to slow down and eventually to stop the GC motion toward the region with stronger magnetic field. The details of this process follow directly from the equation of motion under the action of the parallel force equation (A.65). Expressing the parallel acceleration term in equation (A.65) as [233]:

$$m\frac{d\vec{\upsilon}_{\parallel}}{dt} = m\frac{d\vec{\upsilon}_{\parallel}}{dz} \hat{e}_z = \frac{dW_{\parallel}}{dz} \vec{e}_z \,, \tag{A.66}$$

so that equation (A.65) reduces to:

$$\frac{dW_{\parallel}}{dz} = -\frac{W_{\perp}}{B} \frac{dB}{dz} \tag{A.67}$$

or to:

$$\frac{dW_{\perp}}{dz} = \frac{W_{\perp}}{B} \frac{dB}{dz} \tag{A.68}$$

with the particle energy conservation law $W \equiv W_{\perp} + W_{\parallel} = \text{const}$ taken into account. Equation (A.68) is integrated, and it yields

$$M \equiv \frac{W_{\perp}(z)}{B(z)} = \text{const.} \tag{A.69}$$

The quantity M is the magnetic moment of the particle, and it remains constant within the applied approximation of weak divergence of magnetic field lines, and it is known as the first adiabatic invariant [233]. It is easy to show the

existence of another adiabatic invariant, the so called third adiabatic invariant, which is the magnetic flux Φ_B through the surface encircled by the quasi-circular particle orbit [233]:

$$\Phi_B \equiv \pi r_L^2(z)B(z) = \pi \frac{m^2 \upsilon_\perp^2(z)}{q^2 B(z)} = 2\pi M \frac{m}{q^2} = \text{const.} \qquad \text{(A.70)}$$

Constancy of both the particle energy W and magnetic moment M determine the motion of the GC along the magnetic field [233]. If the considered charged particle with total energy $W = W_\perp + W_\parallel$ is moving with the velocity $\vec{\upsilon}_\parallel$ toward stronger magnetic field, its parallel kinetic energy $W_\parallel$ is falling off due to the force $\vec{F}_\parallel$ and eventually reaches zero at some location $z = z_r$ called the particle reflection point or the mirror point [233]:

$$W_\parallel(z_r) = W - MB(z_r) = 0 \,. \qquad \text{(A.71)}$$

The particle velocity $\vec{\upsilon}_\parallel$ changes its orientation and the GC of the particle starts moving in the opposite direction toward the region of a weaker magnetic field. The location of the reflection point z_r in a given magnetic field configuration depends on two constants [233] of motion specified as initial conditions: the kinetic energy W and the magnetic moment M of a moving particle.

From equation (A.71), we obtain the maximal magnetic field strength $B_{max} \equiv B(z_r)$ a particle with given W and M prescribed at some initial position $z = z_o$, can reach before being bounced back into the domain of a weaker magnetic field. From [233]

$$M = \frac{W_\perp(z)}{B(z)} = \frac{W}{B(z)} \sin^2\theta(z_o) \,, \qquad \text{(A.72)}$$

we get

$$B_{max} = \frac{B(z_o)}{\sin^2\theta(z_o)} \,, \qquad \text{(A.73)}$$

where the pitch angle θ is defined as:

$$\upsilon_\perp = \tan(\theta)\upsilon_\parallel \qquad \text{or} \qquad \upsilon_\perp = \sin(\theta)\upsilon \,. \qquad \text{(A.74)}$$

The same results for drift motions along the magnetic field are valid if the considered slightly divergent/convergent magnetic field topology based on Case A.1.2 is replaced by a similarly modified curved field from Case A.1.7. The

parallel motion of the GC takes place along a new curved coordinate s - line taken to follow a chosen $\vec{B}$ field line rather than along the z - axis [233].

The above expressions remain as they are only with the coordinate z formally replaced by the coordinate s. The curved magnetic field here introduces a transverse centrifugal drift $\vec{v}_C$ given by the equation (A.57). The described magnetic field topology and related charged particle motions are typically found in planetary magnetospheres and also in laboratory plasma trapping devices [233].

A.2 PROBEPIC Source Code

Since its first creation in 1996, several improvements on PROBEPIC algorithms have been already made in previous work. Those improvements were made throughout the system, which included adaptation of different velocity distributions (i.e., Maxwellian and non-Maxwellian) and temperature distributions (different ion and electron temperatures), improved code structure and script usage, handling complex geometries and materials (e.g., probe size/type, simulation domain, plasma model, etc.), use of maximal number of particles, code interface (input/output parameters), improved code execution (self-explanatory/user friendly parameter template) and simulation run time, and most importantly enhanced reliability.

There have been four versions of this simulation code: First version is created in 1996 by Keefer and Markusic. Version-2 through version-4 were created by the author in 2010 (OML/thick sheath theory), 2013 (SL/Thin sheath theory) and 2014 (modeling non-Maxwellian plasmas) respectively. Some of the sub-programs of PROBEPIC along with their functions are presented below.

A.2.1 PROBEPIC.cpp

"PROBEPIC.cpp" is the main program that drives all the sub-programs and functions, which are related to them.

```
#ifdef __linux__
 #include "cpgplot.h"
#else
 #include ".\\PGPLOT\\cpgplot.h"
#endif
#include <time.h>
#include <string.h>
#include "lprobe60.h"
#include "ngrid.hpp"
#include "init.hpp"
#include "ngrid.hpp"

// temporary shouldnt be global
FLOAT *random_num;
long rancount;

# include "random_number_v1.cpp"

/* forward declarations functions after main()*/
void processCommandLine(int argc,char **argv, FLOAT &Voltage, int &duration);
void *dwlmalloc(size_t n);
void output_data(PARTICLEDEF particle1[],PARTICLEDEF particle2[],long  numpart1,long  numpart2,long count,
PLASMA &pl, GEOMETRY &ge,INJECTIONBEAM &ib,BOUNDARY &bb);
```

```
void PGEX30(float xMax);
void PGEX31(float tMax);
void PGEX32(VGRID &grid,FLOAT radius);
unsigned long locate(GRIDDEF_T *xx, unsigned long n, float x);
void distrib(GRIDDEF_T *bins,unsigned long n,PARTICLEDEF *particle,unsigned long num_part, PLASMA &pl);

/*  This is the main program for a two dimensional PIC simulation of a Langmuir probe.  */
int main(int argc, char **argv) {
  int duration;
  long  seed,idx,iter,i6;
  long numelec=0;
  long numion=0;
FLOAT xPlot,yPlot,tMax;
  VGRID iongrid;
  VGRID grid;

  char string_t[30];
  char string_elec[30];

  // structs to be initalized
  SUPERGLOBAL sg;
  PLASMA pl;
  GEOMETRY ge;
  INJECTIONSETUP is;
  INJECTIONBEAM  ib;
  BOUNDARY bb;
  LOCATION lo;
  FIELDSOLVER fs;

// set superglobal members here
  sg.mgp1=74;
  sg.mgp2=74;
  sg.mgp=sg.mgp1*sg.mgp2;
  sg.mpart=800000;
  sg.V_TABLE_RES2=sg.mpart;

    char Inputfile[120]="Input.dat";//default input filename
    //if(--argc>0 && (*++argv)[0]=='-')
    //sscanf(++argv[2],"-%s\n",Inputfile);

    if(argc==4 && sscanf(argv[3],"%s\n",Inputfile)==0) {
     printf("invalid entry for inputfile.  Using Input.dat\n");
     char Inputfile[120]="Input.dat";//default input filename
    }else if (argc == 1) {
       printf("Enter the filename with input data: ");
       if ( scanf("%s",Inputfile)==0)
       printf("invalid entry.  Using Input.dat\n");
    }
    //Input file with most important parameters
printf("argc %d ; argv : %s %s %s %s\n", argc, argv[0], argv[1],argv[2], argv[3]);

   // set defaults for all structs. Reads from input file
  newparam(sg,pl,ge,is,ib,bb,lo,Inputfile);
  // set field solver structs
  initfieldparam(sg,fs);
```

```
 // init old style members
 // ngp_vert=ngp1;
 //ngp_horiz=ngp2;
  // large array initialization
 FLOAT *v_table_e=(FLOAT *)dwlmalloc((sg.mpart+1)*sizeof(FLOAT));
 FLOAT *v_table_i=(FLOAT *)dwlmalloc((sg.mpart+1)*sizeof(FLOAT));
 PARTICLEDEF *electron=(PARTICLEDEF *)dwlmalloc(sg.mpart * sizeof(PARTICLEDEF));
 PARTICLEDEF *ion=(PARTICLEDEF *)dwlmalloc(sg.mpart * sizeof(PARTICLEDEF));
// random num is currently global !!
 random_num=(FLOAT *)dwlmalloc((NPTS+1)*sizeof(FLOAT));

 /*   Initialize array of uniform random numbers  */
 seed=100;
 random_number(seed); // void random_number(long int seed)

 /*  Generate computational cells  */
 printf("main: Generate computational cells ...\n");
 gridgen(grid,sg,ge);
 gridgen(iongrid,sg,ge);

 printf("main:  DONE\n");

 /*  Generate look-up tables for ion and electron speeds  */
 printf("main: Generate look-up tables for %ld speeds...\n", sg.mpart);
 // choose the distribution: distrib=1 - Maxwellian; distrib=0 - Druyv

//set default

 make_velocity_table(pl,v_table_e,v_table_i,sg.mpart);
 printf("main: DONE\n");

 /*  Generate an initial uniform fill of the computational domain with electrons */
 printf("main: Generate an initial uniform fill of the computational domain with electrons...");
 rancount=0;
 numelec=initialize(electron,is.superei_init,ib.supere_init,v_table_e,pl.m_elec,pl.q_elec,ge);
 /*  Refresh array of uniform random numbers */
 seed=101;
 random_number(seed); // void random_number(long int seed)

 printf("main: Generate an initial uniform fill of the computational domain with ions...");
 rancount=0;
 numion=initialize(ion,is.superii_init,ib.superi_init,v_table_i,pl.m_ion,pl.q_ion,ge);
 printf("main: DONE\n");

 printf("main: initial number electrons ==>%ld   initial number ions ==>%ld\n",numelec,numion);

 //--------- start plot stuff----------------------------------------------------------------

 GRIDDEF_T bins[noBins];
 FLOAT min_val,max_val,delta,velocity,energy;
 int max_bin;
  //! minimum speed
 min_val = 0.0;
 //! get the maximum energy
```

```
// Salim: here energy instead of velocity
max_val = 0.0;
for(idx=0;idx<numelec;idx++) {
  energy= (pow(electron[idx].vx,2) +pow(electron[idx].vy,2) +pow(electron[idx].vz,2) )*pl.m_elec/2.0/pl.q_ion;
  if (energy > max_val) max_val = energy;
}
// set the lower limit for each bin
delta = (max_val-min_val)/noBins;     //! mean # of points in a bin
for(idx=0;idx<noBins;idx++) {
  bins[idx].r = min_val + idx*delta;    //! the lower limit of the bin
  bins[idx].v = 0.0;              //! shows how many data points fall in it
}
//! PUT v_table_e DATA IN BINS
distrib(bins,noBins,electron,numelec,pl);

//! get the bin with the maximum number of particles in it
max_bin = 0;
for(idx=0;idx<noBins;idx++) {
  if (bins[idx].v > bins[max_bin].v) max_bin = idx;
}
//---------------------- plot sequence ----------------------
//! Call PGOPEN to initiate PGPLOT and open the output device; PGOPEN
//! will prompt the user to supply the device name and type. Always
//! check the return code from PGOPEN.
  if(argc == 1) {

  if(cpgopen("/ps") < 0)
  exit(1);
} else { // if there are command line arguments, switch to console mode
// if(cpgbeg(0, "/NULL", 1, 1) != 1)
  if(cpgbeg(0, "/NULL", 1, 1) != 1)
  exit(1);
}
//if(cpgbeg(0, "/CGW", 1, 1) != 1)  exit(EXIT_FAILURE);
//!                        /GW:
//!                        /CGW:
//!                        /GWD:  crash
//!                        /CGWD: crash
//!if(PGBEG(0, "/WV", 1, 1) /= 1) STOP !UNIT, FILE, NXSUB, NYSUB
//!!CALL PGASK(1) ! FLAG=1(TRUE) : 'prompt state' of PGPLOT is set to ON
cpgbbuf(); //! Begin saving graphical output commands in an internal buffer; the
//! commands are held until a matching PGEBUF call
//cpgask(1); //! FLAG=1(FALSE) :prompt when plot befor plotting on another page

//check grid
PGEX32(grid,ge.radius);
cpgask(1); //! FLAG=0(FALSE) :do not prompt when plot befor plotting on another page

//cpgeras();
cpgsubp(2, 1);  //! two adjacent panels
//!
//! Print information about device.
//!CALL PGEX0
//!
//! Call the drawing subroutines
```

```
  ////!     Set manually the color table to default values
  //    cpgscr(0, 0.0, 0.0, 0.3); // changes color index 0 (background) to dark blue
  //    cpgscr(0, 0.0, 0.0, 0.0); // changes color index 0 (background) to black
  //cpgscr(5, 0.00, 1.00, 1.00); // cyan
  //cpgscr(1, 1.00, 1.00, 1.00); // white
  //cpgscr(3, 0.00, 1.00, 0.00); // green
  //cpgscr(6, 1.00, 0.00, 1.00); // magenta
  //cpgscr(14, 0.33, 0.33, 0.33); // dark gray
  //!---------------------------------
  //! select Left Panel
  cpgpanl(2, 1);
  //cpgeras(); //! Erase all graphics from the current page or panel.

  PGEX30(max_val);  //! draw axes

  //! draw graph of electron speed distribution
  cpgsci(6);   //! magenta
  for(idx=0;idx<noBins;idx++) {
    xPlot = bins[idx].r;
    yPlot = bins[idx].v/bins[max_bin].v;
    cpgpt(1,&xPlot,&yPlot,17); // graphic symbol #17
  }
  //! draw text
  cpgsci(1);  //! white
  sprintf(string_t,"t = %f sec",0.0);
  cpgptxt(2000000.0,0.65,0.0, 0.0,string_t); //! X, Y, ANGLE, FJUST, TEXT)

  sprintf(string_elec,"No. electrons: %ld", numelec);
  cpgptxt(2000000.0,0.50,0.0, 0.0,string_elec); //! X, Y, ANGLE, FJUST, TEXT)

  //! flush any pending commands to the      output device.
  cpgupdt();
  //!-------------------------------------------------------------------------------
  // for compaitbility need to get cmd-line args after plot input type
  processCommandLine(argc,argv,bb.V,duration);

/// This is added to provide a file with electron EEDF in simulation
  FILE *EEDF;
  char filename_eedf[200];
  sprintf(filename_eedf,"EEDF_%+06.2fV.dat.csv", bb.V);
  EEDF=fopen(filename_eedf,"a");

  for(idx=0;idx<noBins;idx++) {
    xPlot = bins[idx].r;
    yPlot = bins[idx].v/bins[max_bin].v;
    fprintf(EEDF,"%f  %f\n",xPlot, yPlot );
  }
fclose(EEDF);

  tMax=(FLOAT)duration;
  iter = duration/(ib.dti*1000000.0);
  printf("This simulation will perform %ld electron iterations\n",15*iter);
  printf("The output file is: current_q100.dat.csv ('append' mode)\n");
  printf("The program may be stopped at any time by pressing <CTRL>C\n");
```

```
printf("\nProgress:");

cpgask(1); //! FLAG=0(FALSE) :do not prompt when plot befor plotting on another page
// select Right Panel (ion/electron)
cpgpanl(1, 1);
//cpgeras(); // Erase all graphics from the current page or panel.
PGEX31(tMax);       // draw axes
cpgsci(6); // magenta for electrons
cpgptxt(tMax/3.0,73000.0,0.0, 0.0," electrons"); // X, Y, ANGLE, FJUST, TEXT)
cpgsci(3); // green for ions
cpgptxt(tMax/3.0,75000.0,0.0, 0.0," ions"); // X, Y, ANGLE, FJUST, TEXT)
//!----------------- end plot init stuff ----------------------------------------------

printf("main: LU decomposition of the difference matrix for the electric field solver...\n");
make_LU(grid,sg,ge,fs);

for (idx = 1; idx <= iter; idx++) {
 if (rancount > (NPTS - OUTPUT_FREQUENCY)) {
   seed += 100;
   random_number(seed);
   rancount=0;
 }
 /* electron iteration */
 reset_grid(grid);
 qweight(grid,ion,numion,ge);
 memcpy(&iongrid[0][0],&grid[0][0],(sizeof(GRIDDEF))*sg.mgp);
 for (i6=0;i6 < is.elec_iter;i6++) {
  if (i6)
    memcpy(&grid[0][0],&iongrid[0][0],(sizeof(GRIDDEF))*sg.mgp);

  qweight(grid,electron,numelec,ge);
  efield(grid,pl,ge,bb,fs);
  fweight(grid,electron,numelec,ge);
  numelec=inject(0, electron, numelec, v_table_e,sg,pl,ge,ib,is);
  mover(electron,numelec,ib.dte);
  numelec=boundary(0,electron,numelec,pl,ge,bb);
  //printf("main:eloop numelec=%ld\n",numelec);
 }
 /* ion iteration */
 if (idx%1==0){
 fweight(grid,ion,numion,ge);
 numion=inject(1, ion, numion, v_table_i,sg,pl,ge,ib,is);
 mover(ion,numion,ib.dti*1.0);
 numion=boundary(1, ion,numion,pl,ge,bb);
       }
 //!-----------------------------------------------------------------
 //! select Right Panel (ion/electron)
 cpgpanl(2, 1);
 //! restore world->screen transformation
 cpgswin(0.0,tMax,10000.0,80000.0); //! Lab-report.pdf (page 12)
 cpgslw(4);      //! line width
 cpgsci(6); //! magenta for electrons
 xPlot=idx*ib.dti*1e6;
 yPlot=(float)numelec;
 cpgpt(1,&xPlot,&yPlot,-2);      //! -1, -2  : a dot (diameter = current line width).
```

```
    cpgsci(3); //! green for ions
    yPlot=(float)numion;
    cpgpt(1,&xPlot,&yPlot,-2);      //! -1, -2  : a dot (diameter = current line width).
    cpgupdt(); //! display now !
    //!-----------------------------------------------------------------

    printf("\r%f %% electrons:%ld  ions:%ld",idx*100.0/iter,numelec,numion);
    fflush(stdout);
    if ( idx % OUTPUT_FREQUENCY == 1 ){
      printf("\n idx=%ld entering output _data\n",idx);
      output_data(electron,ion,numelec,numion,idx,pl,ge,ib,bb);
    }
  }
  // free up stuff */

  free(v_table_e);
  free(v_table_i);
  free(ion);
  free(electron);
  freefieldparam(fs);

  return EXIT_SUCCESS;
}
//************** utility functions ********************************************/
//******* nb forward declarations before main *********************************/

void processCommandLine(int argc,char **argv, FLOAT &Voltage, int &duration ){
  // set defaults
  Voltage=1.0f;
  duration=1;
  char Inputfile[120]="Input.dat";//default input filename

  if( argc !=1  && argc != 4) {
    printf("Usage: %s PROBEPOTENTIAL TIME Inputfile.dat \n", argv[0]);
    exit(1);
  }
  if(argc == 1) {
     printf("Enter V [V]: ");
     if (scanf("%f",&Voltage)!=1)
        printf("invalid entry.  Using 1 V.\n");

     printf("Enter the duration of the experiment [micro-sec]: ");
     if ( scanf("%d",&duration)!=1)
          printf("invalid entry.  Using 1 $\\mu$s\n");

  }else{
     char *endptr;

     Voltage = strtod(argv[1], &endptr);
     if(endptr == argv[1]) {
          printf("invalid entry for the potential.  Using 1 V.\n");
          Voltage = 1.0;
     }
     duration = strtol(argv[2], &endptr, 10);
     if(endptr == argv[2]) {
```

```
        printf("invalid entry for the simulation time.  Using 1 μs\n");
        duration=1.0;
    }
  }
 printf("V: %f, duration: %d μs, Inputfile: %s \n", Voltage, duration, Inputfile);
 return;
}
// Customized malloc - really needed ?
void *dwlmalloc(size_t n) {
  void*p = malloc(n);
  if (NULL == p) {
    fprintf(stderr,"dwlmalloc(): memory allocation failure\n");
    exit(1);
  }
  return p;
}
void output_data(PARTICLEDEF particle1[],PARTICLEDEF particle2[],long  numpart1,long  numpart2,long count,
                 PLASMA &pl, GEOMETRY &ge,INJECTIONBEAM &ib,BOUNDARY &bb
                ) {
  long  idx;
  FLOAT nde,ndi,tms;
  FLOAT avecurrent,avecurre,avecurri;
  char filename[200];
  FILE *currdata;

  sprintf(filename,"current_%+06.2fV.dat.csv", bb.V);
  currdata=fopen(filename,"a");

  avecurrent=bb.current/(OUTPUT_FREQUENCY*ib.dti);
  avecurre=bb.curre/(OUTPUT_FREQUENCY*ib.dti);
  avecurri=bb.curri/(OUTPUT_FREQUENCY*ib.dti);

  /*    COMPUTE NUMBER DENSITY     */

  nde=ndi=0.0;

  for (idx=0; idx < numpart1; idx++)
    if (particle1[idx].r <= ge.radius)
      nde += particle1[idx].m;
  nde /= ((pl.m_elec)*(pi*ge.radius*ge.radius*ge.length));

  for(idx=0;idx < numpart2;idx++)
    if(particle2[idx].r <= ge.radius)
      ndi += particle2[idx].m;
  ndi /= ((pl.m_ion)*(pi*ge.radius*ge.radius*ge.length));

  tms=count*ib.dti*1e6;
  if (count==1)
    fputs("   V, t[micro-sec],     ne,    ni,      nde,       ndi,       j,          je,          ji\n\n",currdata);

  fprintf(currdata,"%f, %f, %ld, %ld, %e, %e, %e, %e,
%le\n",bb.V,tms,numpart1,numpart2,nde,ndi,avecurrent,avecurre,avecurri);
  fclose(currdata);
  printf(" current: %f [mA]",avecurrent*1000.0);
```

```
  // if(tms > 2.2723e-02 ) {
  // //if (tms >=0.007625 ) {
  //   puts("\nexit for timing.  output_data.cpp");
  //   exit(0);
  // }

  bb.current=0.0;
  bb.curri=0.0;
  bb.curre=0.0;
}
void PGEX30(float xMax)  {// bins
//             int max_bin,i;
     cpgenv(0.0,xMax,0.0,1.0,0,-2);

// Draw first a grid at low brightness(14=dark gray)

     cpgsci(14); // Dark Gray
          cpgbox("G",0.0,0,"G",0.0,0); // automatic

// and then a frame and axes at full brightness.
     cpgsci(5);      // cyan
     cpgbox("ABCTSN",0.0,0,"ABCTSNV",0.0,0);// automatic

// Call PGLAB to label the graph in a different color (3=green).
     cpgsci(3);     // green
     cpgsci(1);     // white
     cpglab("velocity(W)","f(W) ","Velocity distribution during injection");
}; // PGEX30

void PGEX31(float tMax)   {
     cpgenv (0.0,tMax,10000.0,80000.0,0,-2);  // Lab-report.pdf (page 12)

// Draw first a grid at low brightness(14=dark gray)
     cpgsci(14); // Dark Gray
          cpgbox("G",0.0,0,"G",0.0,0); // automatic
// and then a frame and axes at full brightness.
     cpgsci(5);      // cyan
     cpgbox("ABCTSN",0.0,0,"ABCTSNV",0.0,0);// automatic

// Call PGLAB to label the graph in a different color (3=green).
     cpgsci(3);     // green
     cpgsci(1);     // white
     cpglab("time[micro-sec]","no. of particles ","Particle no. during injection");
          // http://www.starlink.rl.ac.uk/star/docs/sc15.htx/node10.html
};
//void PGEX32(GRIDDEF grid[ngp1+1][ngp2+1])    {
//void PGEX32(GRIDDEF grid[ngp1][ngp2],FLOAT radius)      {
void PGEX32(VGRID &grid,FLOAT radius)      {
          // draw the grid
          // (grid.x,grid.y) is the left-down corner of the cell
          // grid[][0],grid[][1],...,grid[][ngp2-1] advances on OX
          // grid[0][],grid[1][],...,grid[ngp1-1][] advances on OY

          static float xPlot[4],yPlot[4];
          float minX,maxX,minY,maxY;
```

```
    long ngp_vert,ngp_horiz;
      int nx,ny;

    ngp_vert=grid.size();
    ngp_horiz=grid[0].size();

      // determine the (x,y) limits
      minX=maxX=grid[0][0].x;
      minY=maxY=grid[0][0].y;
      for(nx =0;nx<ngp_horiz;nx++)
          for(ny=0;ny<ngp_vert;ny++) {
              if (grid[ny][nx].x > maxX) maxX = grid[ny][nx].x;
              if (grid[ny][nx].x < minX) minX = grid[ny][nx].x;
              if (grid[ny][nx].y > maxY) maxY = grid[ny][nx].y;
              if (grid[ny][nx].y < minY) minY = grid[ny][nx].y;
          };
      maxX =0.0;
      maxY =radius;

   cpgsci(14); // Dark Gray
   cpgenv(minX,maxX,minY,maxY,
            1, // the scales on Ox and Oy will be equal
              0); // draw box and label it with coordinates
       cpgbox("AP",0.0,0,"AP",0.0,0); // automatic

       // draw the cells trigonometric-wise
       cpgsci(1); // White
       for(nx =0;nx<ngp_horiz-1;nx++)
          for(ny=0;ny<ngp_vert-1;ny++) {
              xPlot[0]=grid[ny][nx].x;     yPlot[0]=grid[ny][nx].y;
              xPlot[1]=grid[ny][nx+1].x;   yPlot[1]=grid[ny][nx+1].y;
              xPlot[2]=grid[ny+1][nx+1].x; yPlot[2]=grid[ny+1][nx+1].y;
              xPlot[3]=grid[ny+1][nx].x;   yPlot[3]=grid[ny+1][nx].y;
              cpgline(4,xPlot,yPlot);
          }
     cpgsci(5); // cyan
          cpgbox("G",0.0,0,"G",0.0,0); // automatic
          cpgupdt(); // output the graph
//// Draw first a grid at low brightness(14=dark gray)
//     cpgsci(14); // Dark Gray
//       cpgbox("G",0.0,0,"G",0.0,0); // automatic
//// and then a frame and axes at full brightness.
//     cpgsci(5);  // cyan
//     cpgbox("ABCTSN",0.0,0,"ABCTSNV",0.0,0);     // automatic
//
//// Call PGLAB to label the graph in a different color (3=green).
//     cpgsci(3); // green
//     cpgsci(1); // white
//     cpglab("time[sec]","no. of particles ","Particle no. during injection");
}
unsigned long locate(GRIDDEF_T *xx, unsigned long n, float x) {
  unsigned long ju,jm,jl;
  int ascnd;

  jl=0;
```

```
  ju=n+1; // n intervals, (n+1) division points
  ascnd=(xx[n-1].r > xx[1].r);
  while (ju-jl > 1) {
    jm=(ju+jl) >> 1;
    if ((x > xx[jm].r) == ascnd)
      jl=jm;
    else
      ju=jm;
  }
  return jl;
}
void distrib(GRIDDEF_T *bins,unsigned long n,PARTICLEDEF *particle,unsigned long num_part,PLASMA &pl)
{
  unsigned long i,pos;
  float v;

  // Empty bins
  for(i=0;i<n;i++)
    bins[i].v = 0.0;

  // PUT DATA IN BINS
  //Salim: note that input changed into energy
  for(i=0;i<num_part;i++) {
    v= (pow(particle[i].vx,2) +pow(particle[i].vy,2) +pow(particle[i].vz,2) )*pl.m_elec/2.0/pl.q_ion;
    pos=locate(bins, n, v);
    bins[pos].v ++;
  }
}
```

A.2.2 grid.cpp

The computational grid is a structure, which associates the following properties to each grid-point: position, charge density, radial and axial electric field intensity, and electrostatic potential.

```
//#include "lprobe60.h"

void gridgen(GRIDDEF array[ngp1][ngp2],GEOMETRY  &ge) {
  long int count1,count2,cell1_count,cell6_count;
  float index1,index2;

  cell1_count=15,cell6_count=74;

  index1=-1.0;
  for (count1=0;count1 < cell1_count;count1++){ // rows 0...14
    ++index1;
    index2=-1.0;
    for (count2=0;count2 < ngp_horiz;count2++){
      ++index2;
      array[count1][count2].x=-ge.length+index2*ge.cell_width;
      array[count1][count2].y=index1*ge.cell_height1;
    }
```

```
  }
  index2=-1.0;
  for (count2=0;count2 < ngp_horiz;count2++){
    ++index2;
    array[15][count2].x=-ge.length+index2*ge.cell_width;
    array[15][count2].y= array[14][count2].y+ge.cell_height2;
  }
  index2=-1.0;
  for (count2=0;count2 < ngp_horiz;count2++){
    ++index2;
    array[16][count2].x=-ge.length+index2*ge.cell_width;
    array[16][count2].y=array[15][count2].y+ge.cell_height3;
  }
  index2=-1.0;
  for (count2=0;count2 < ngp_horiz;count2++){
    ++index2;
    array[17][count2].x=-ge.length+index2*ge.cell_width;
    array[17][count2].y=array[16][count2].y+ge.cell_height4;
  }
  index2=-1.0;
  for (count2=0;count2 < ngp_horiz;count2++){
    ++index2;
    array[18][count2].x=-ge.length+index2*ge.cell_width;
    array[18][count2].y=array[17][count2].y+ge.cell_height5;
  }
  //cell_height6 = (radius - array[18][0].y)/(ngp_vert-18);
  index1=0.0;
  for (count1=19;count1 < cell6_count;count1++){ // rows 19...73
    ++index1;
    index2=-1.0;
    for (count2=0;count2 < ngp_horiz;count2++) {
      ++index2;
      array[count1][count2].x=-ge.length+index2*ge.cell_width;
      array[count1][count2].y=array[18][count2].y+index1*ge.cell_height6;
    }
  }
  /*     Clean-up grid for next time step*/
  for (count1=0;count1 < ngp_vert;count1++){
    for (count2=0;count2 < ngp_horiz;count2++){
      array[count1][count2].q_dens=0.0;
          array[count1][count2].Ex=0.0;
      array[count1][count2].Ey=0.0;
      //printf("%ld %ld %f %f\n",count1,count2,array[count1][count2].x,array[count1][count2].y);
    }
    //putchar('\n');
  }
  //exit(0);
}
void reset_grid(GRIDDEF array[ngp1][ngp2]) {
  long int count1,count2;
  GRIDDEF*r;
  for (count1 = 0; count1 < ngp_vert; count1++) {
    r = array[count1];
    for (count2 = 0; count2 < ngp_horiz; count2++)
      r[count2].q_dens=0.0;
```

```
  }
}
/*      This function computes the elcetric field intensity at
        the gridpoints using the updated source term and the
        previously factored [A] matrix   */
void efield(GRIDDEF array[ngp1][ngp2],PLASMA &pl, GEOMETRY &ge, BOUNDARY &bb, FIELDSOLVER
&fs) {

  long  j,k,count1;
  FLOAT deltx,jmax,kmax;
  char trans[]="N";

  fs.nrhs=1;
  jmax=ngp_vert;
  kmax=ngp_horiz;
  deltx=ge.cell_width;

  /*    Zero out source term       */
  count1=-1;
  for(j=0;j<ngp_vert;j++){
    for(k=0;k<ngp_horiz;k++){
      count1++;
      fs.b[count1]=0.0;
    }
  }
  /*    Update source term  */
  count1=-1;

  for(j=0;j<ngp_vert;j++){
    for(k=0;k<ngp_horiz;k++){
      count1++;
      if(k==0 && j!=(jmax-1)){
          fs.b[count1]=-array[j][k].q_dens /EPS0 - pl.plasma_pot;
      }
      else if(j==(jmax-1) && k!=(kmax-1)){
          fs.b[count1]=-array[j][k].q_dens /EPS0 - pl.plasma_pot;
      }
      else if(j<=ge.jprobe && j>0 && k>=ge.kprobe){
          fs.b[count1]=-array[j][k].q_dens /EPS0 - bb.V;
      }
      else if(j<=ge.jprobe && j==0  && k>=ge.kprobe){
          fs.b[count1]=-array[j][k].q_dens /EPS0 - bb.V;
      }
      else if(j>ge.jprobe && k==(kmax-1)){
          fs.b[count1]=-array[j][k].q_dens /EPS0  - pl.plasma_pot;
      }
      else{
          fs.b[count1]=-array[j][k].q_dens /EPS0;
      }
    }
  }
  /*    Call Lapack to solve [a]x=b       */
  /*printf("%ld   %d\n",t_count2,ldab);*/

 sgbtrs_(trans,&fs.n1,&fs.lower_width,&fs.upper_width,&fs.nrhs,fs.abd,&fs.ldab,fs.piv,fs.b,&fs.ldb,&fs.info);
```

```
 if (fs.info != 0){
  printf("Band Solver Failed, info==>%d\n",fs.info);
  exit(1);
 }
 /*    Put phi into grid (note:this is a waste, but for now....)   */
 count1=-1;
 for(j=0;j<ngp_vert;j++){
  for(k=0;k<ngp_horiz;k++){
   count1++;
   array[j][k].phi=fs.b[count1];
  }
 }
 /*    Determine electric field at all gridpoints     */
 for(j=0;j<ngp_vert;j++){
  for(k=0;k<ngp_horiz;k++){
   if (k==0){
      array[j][k].Ex=1.0/(2.0*deltx)*(-3.0*array[j][0].phi+4.0*array[j][1].phi-array[j][2].phi);
      array[j][k].Ey=0.0;
   }
   else if (j==ge.jprobe &&  k >= ge.kprobe){
      array[j][k].Ey=1.0/(2.0*deltx)*(-3.0*array[j][k].phi+4.0*array[j][k+1].phi-array[j][k+2].phi);
      array[j][k].Ex=0.0;
   }
   else if (k==(ngp_horiz-1)){
      array[j][k].Ex=-1.0/(2.0*deltx)*(-3.0*array[j][ngp_horiz-1].phi+4.0*array[j][ngp_horiz-2].phi-array[j][ngp_horiz-
3].phi);
      array[j][k].Ey=0.0;
   }
   else if(j==0){
      array[j][k].Ex=-(array[j][k-1].phi-array[j][k+1].phi)/(2.0*deltx);
      array[j][k].Ey=0.0;
   }
   else if(j==(ngp_vert-1)){
      array[j][k].Ex=0.0;
      array[j][k].Ey=array[j-1][k].phi/ge.cell_height6;
   }
   else
      {
       array[j][k].Ex=(array[j][k-1].phi-array[j][k+1].phi)/(2.0*deltx);
       array[j][k].Ey=-((array[j+1][k].phi-array[j][k].phi)/(array[j+1][k].y-array[j][k].y))*(array[j][k].y-array[j-
1][k].y)/(array[j+1][k].y-array[j-1][k].y)
         -((array[j][k].phi-array[j-1][k].phi)/(array[j][k].y-array[j-1][k].y))*(array[j+1][k].y-array[j][k].y)/(array[j+1][k].y-
array[j-1][k].y);
      }
  }
 }
}
/* This function weights the field from the array */

void fweight(GRIDDEF array[ngp1][ngp2],PARTICLEDEF particle[],long  number,GEOMETRY ge) {
 long int count1;
 FLOAT dr,dr1,dr2,dz,dz1,dz2;
 PARTICLEDEF*p;
 GRIDDEF*a00,*a01,*a10,*a11;
```

```
 FLOAT den, n0, n1, n2, n3;

 for (count1=0;count1 < number;count1++) {
  p = particle+count1;
  if (ge.radius < p->r)
    p->Ey=p->Ex=0.0;
  else if ((p->celly+1 < ngp1) && (p->cellx+1 < ngp2)) {
    a00 = array[p->celly+0]+(p->cellx+0);
    a10 = array[p->celly+1]+(p->cellx+0);
    a01 = array[p->celly+0]+(p->cellx+1);
    a11 = array[p->celly+1]+(p->cellx+1);
    dr = a10->y*a10->y - a00->y*a00->y;
    dr1 = p->r-a00->y;
    dr2 = a10->y-p->r;
    dz = a01->x-a00->x;
    dz1 = p->x-a00->x;
    dz2 = a01->x-p->x;
    den = 2.0*dr*dz;
    n0 = dr2*dz2*(2.0*a10->y+3.0*a00->y-p->r);
    n1 = dr2*dz1*(2.0*a11->y+3.0*a00->y-p->r);
    n2 = dr1*dz2*(3.0*a10->y+2.0*a00->y-p->r);
    n3 = dr1*dz1*(3.0*a11->y+2.0*a01->y-p->r);
    p->Ey=(a00->Ey*n0+a01->Ey*n1+a11->Ey*n3+a10->Ey*n2)/den;
    p->Ex=(a00->Ex*n0+a01->Ex*n1+a11->Ex*n3+a10->Ex*n2)/den;
  } else
    printf("\nfweight: cellx=%lu celly=%lu \n",p->cellx,p->celly);
 }
}
 unsigned long locatex(GRIDDEF xx[ngp1][ngp2], unsigned long n, float x);
 unsigned long locatey(GRIDDEF xx[ngp1][ngp2], unsigned long n, float x);

void qweight(GRIDDEF array[ngp1][ngp2],PARTICLEDEF particle[],long int number,GEOMETRY &ge) {
 unsigned long positionx,positiony;
 long int count1;
 FLOAT dr,dr1,dr2,dz,dz1,dz2;
 FLOAT den, n0,n1,n2,n3;

 for(count1=0;count1 < number;count1++) {
  PARTICLEDEF*p = particle+count1;
  if (p->r <= ge.radius) {
   ////Lambert, April 2013 uniform grid allows simplification, which is made elsewhere in the code.
   positionx = (int)((p->x-(*array)->x)/ge.cell_width);// locatex(array,ngp2-1,p->x,&positionx);
   positiony=locatey(array,ngp1-1,p->r);
   if ((positionx+1) < ngp2 && (positiony+1) < ngp1) { // particle in domain cell ?

       GRIDDEF
         *a00 = array[positiony+0]+positionx+0,
         *a10 = array[positiony+1]+positionx+0,
         *a01 = array[positiony+0]+positionx+1,
         *a11 = array[positiony+1]+positionx+1;

       /*   SAVE CELL LOCATION FOR FORCE WEIGHT   */
       p->cellx=positionx;
       p->celly=positiony;
```

```
        dr = a10->y*a10->y-a00->y*a00->y;
        dr1 = p->r-a00->y;
        dr2 = a10->y-p->r;
        dz = a01->x-a00->x;
        dz1 = p->x-a00->x;
        dz2 = a01->x-p->x;

        den = (2.0*dr*dz);
        n0 = dr2*dz2*(2.0*a10->y+3.0*a00->y-p->r);
        n1 = dr2*dz1*(2.0*a11->y+3.0*a00->y-p->r);
        n2 = dr1*dz2*(3.0*a10->y+2.0*a00->y-p->r);
        n3 = dr1*dz1*(3.0*a11->y+2.0*a01->y-p->r);

        a00->q_dens += p->q*n0/den;
        a01->q_dens += p->q*n1/den;
        a10->q_dens += p->q*n2/den;
        a11->q_dens += p->q*n3/den;
     } else
        printf("\nqweight: cellx=%lu celly=%lu \n",positionx,positiony);
   }
  }
}
/* delta x must be constant because .[xz] doesn't appear.  lambert, 2013 */

//#include "lprobe60.h"
//#include "lapack.cpp"

/*      This function computes the LU decomposition of the band matrix generated
        by differencing Poisson's Equation on the grid    */
void make_LU(GRIDDEF array[ngp1][ngp2], SUPERGLOBAL &sg, GEOMETRY &ge,FIELDSOLVER &fs) {

  const char*name;
  int info;
  long  i,j,k,jmax,kmax,count,counter1,counter2,counter3,counter4,counter5;
  float ajm1ky, ajp0ky, ajp1ky;
  float cell_width;
  FILE *abddata,*abddata2;
  //FIELDSOLVER fs;

  // dereference
  cell_width=ge.cell_width;

  name = "abddata.dat";
  if (NULL == (abddata=fopen(name,"w+"))) {
    fprintf(stderr,"cannot open LU file %s\n",name);
    exit(1);
  }
  name = "abddata2.dat";
  if (NULL == (abddata2=fopen(name,"w+"))) {
    fprintf(stderr,"cannot open LU file %s\n",name);
    exit(1);
  }
  // initfieldparam(sg,fs);

  /*   MAKE BANDS   */
```

```
jmax=ngp_vert;
kmax=ngp_horiz;
counter1=fs.upper_width-1;
counter2=0;
counter3=-1;
counter4=0;
counter5=0;

/*  ****  */
/*    Initialize all non-relevant positions to zero  */
for(i=0;i < ngp;i++)
  fs.band1[i]=fs.band2[i]=fs.band3[i]=fs.band4[i]=fs.band5[i]=0.0;

for (j=0;j < jmax;j++){
  for(k=0;k < kmax;k++){
    ajm1ky = array[j-1][k].y;
    ajp0ky = array[j+0][k].y;
    ajp1ky = array[j+1][k].y;
    if(counter1<(ngp-1))
       counter1++;
    if(counter2<(ngp-1))
       counter2++;
    counter4++;counter5++;

    /*   y < r_probe      */
    if(j <= ge.jprobe){
       if (j==0 && k==1){
         counter4=0;
       }
       if(j==1  && k==0){
         counter5=0;
       }
       if( k < ge.kprobe){
         if(j==0){       /*     Lower boundary*/
           if(k==0) {
             fs.band1[counter1]=0.0;
             fs.band2[counter2]=0.0;
             fs.band4[counter4]=0.0;
             fs.band5[counter5]=0.0;
           }
           else {
             fs.band1[counter1]=pi*cell_width;
             fs.band2[counter2]=pi*ajp1ky*ajp1ky/(4.0*cell_width);
             fs.band4[counter4]=pi*ajp1ky*ajp1ky/(4.0*cell_width);
             fs.band5[counter5]=0.0;
           }
         }
         else{      /*     Rest to top of probe   */
           if(k==0) {
             fs.band1[counter1]=0.0;
             fs.band2[counter2]=0.0;
             fs.band4[counter4]=0.0;
             fs.band5[counter5]=0.0;
           }
```

```
            else{
              fs.band1[counter1]=2.0*pi*cell_width*(ajp0ky+(ajp1ky-ajp0ky)/2.0)/(ajp1ky-ajp0ky);
              fs.band2[counter2]=pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0)
                          -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width;
              fs.band4[counter4]=pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0)
                          -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width;
              fs.band5[counter5]=2.0*pi*cell_width*(ajp0ky-(ajp0ky-ajm1ky)/2.0)/(ajp0ky-ajm1ky);
            }
          }
        }
        else {      /*     on and inside probe    */
          fs.band1[counter1]=fs.band2[counter2]=fs.band4[counter4]=fs.band5[counter5]=0.0;
        }
      }
      else if (j>ge.jprobe && j<(jmax-1)){
        if((k > 0) && (k < (kmax-1))){
          fs.band1[counter1]=2.0*pi*cell_width*(ajp0ky+(ajp1ky-ajp0ky)/2.0)/(ajp1ky-ajp0ky);
          fs.band2[counter2]=pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-
ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width;
          fs.band4[counter4]=pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-
ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width;
          fs.band5[counter5]=2.0*pi*cell_width*(ajp0ky-(ajp0ky-ajm1ky)/2.0)/(ajp0ky-ajm1ky);
        }
        else if(k==0){
          fs.band1[counter1]=0.0;
          fs.band2[counter2]=0.0;
          fs.band4[counter4]=0.0;
          fs.band5[counter5]=0.0;
        }
        else  {
          fs.band1[counter1]=2.0*pi*( ajp0ky+(ajp1ky-ajp0ky)/2.0 )*cell_width/(ajp1ky-ajp0ky);
          fs.band2[counter2]=0.0;
          fs.band4[counter4]=pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0)
                      -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width;
          fs.band5[counter5]=2.0*pi*( array[j][ngp_horiz-1].y-(ajp0ky-ajm1ky)/2.0 )*cell_width/(ajp0ky-ajm1ky);
        }
      }
      else{
        /*fs.band1[counter1]=0.0;*/
        fs.band2[counter2]=0.0;
        fs.band4[counter4]=0.0;
        fs.band5[counter5]=0.0;
      }
    }
  }
  /*    Diagonal  */
  for (j=0;j < jmax;j++){
    /*   y < r_probe      */
    if(j <= ge.jprobe){
      for(k=0;k < kmax;k++){
        ajm1ky = array[j-1][k].y;
        ajp0ky = array[j+0][k].y;
        ajp1ky = array[j+1][k].y;
        counter3++;
        if(k >= 0 && (k < ge.kprobe)){
```

```
          if(j==0){
            if(k==0){
              fs.band3[counter3]=-1.0;
            }
            else{
              fs.band3[counter3]=-pi*(cell_width+(ajp1ky-ajp0ky)*(ajp1ky-ajp0ky)/(2.0*cell_width));
            }
          }
          else {
            if(k==0){
              fs.band3[counter3]=-1.0;
            }
            else{
              fs.band3[counter3]=-(2.0*pi*cell_width*(ajp0ky+(ajp1ky-ajp0ky)/2.0)/(ajp1ky-ajp0ky) +
pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-
ajm1ky)/2.0))/cell_width + pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-
ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width + 2.0*pi*cell_width*(ajp0ky-(ajp0ky-ajm1ky)/2.0)/(ajp0ky-
ajm1ky));
            }
          }
        }
        else{
          fs.band3[counter3]=-1.0;
        }
    }
   }
   /*   y > r_probe     */
   else if(j<(jmax-1)){
     for(k=0;k < kmax;k++){
        ajm1ky = array[j-1][k].y;
        ajp0ky = array[j+0][k].y;
        ajp1ky = array[j+1][k].y;
        counter3++;
        if(k > 0 && (k < (kmax-1)) && (j<(jmax-1))){
          fs.band3[counter3]=-(2.0*pi*cell_width*(ajp0ky+(ajp1ky-ajp0ky)/2.0)/(ajp1ky-ajp0ky) +
pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-
ajm1ky)/2.0))/cell_width + pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-
ajm1ky)/2.0)*(ajp0ky-(ajp0ky-ajm1ky)/2.0))/cell_width + 2.0*pi*cell_width*(ajp0ky-(ajp0ky-ajm1ky)/2.0)/(ajp0ky-
ajm1ky));
        }
        else if(k==0){
          fs.band3[counter3]=-1.0;
        }
        else{
          fs.band3[counter3]=-(2.0*pi*cell_width*(ajp0ky+(ajp1ky-ajp0ky)/2.0)/(ajp1ky-ajp0ky) +
pi*((ajp0ky+(ajp1ky-ajp0ky)/2.0)*(ajp0ky+(ajp1ky-ajp0ky)/2.0) -(ajp0ky-(ajp0ky-ajm1ky)/2.0)*(ajp0ky-(ajp0ky-
ajm1ky)/2.0))/cell_width + 2.0*pi*cell_width*(ajp0ky-(ajp0ky-ajm1ky)/2.0)/(ajp0ky-ajm1ky));
        }
     }
   }
   else{
     for(k=0;k < kmax;k++){
        counter3++;
        fs.band3[counter3]=-1.0;
     }
```

```
  }
 }
 /*for(i=0;i<ngp;i++){
  printf("%ld   %f   %f   %f   %f   %lf\n",i,fs.band1[i],fs.band2[i],fs.band3[i],fs.band4[i],fs.band5[i]);
  }*/

 /*    Form fs.band vector to be submitted to the fs.band solver   */
 count=-1;
 for(i=0;i<ngp;i++){
  for(j=1;j<=(2*fs.lower_width+fs.upper_width+1);j++){
   count++;
   if(j==fs.lower_width+1)                    fs.abd[count]=fs.band1[i];
   else if(j==(fs.lower_width+fs.upper_width))   fs.abd[count]=fs.band2[i];
   else if(j==(fs.lower_width+fs.upper_width+1))      fs.abd[count]=fs.band3[i];
   else if(j==(fs.lower_width+fs.upper_width+2))      fs.abd[count]=fs.band4[i];
   else if(j==(fs.lower_width+fs.upper_width+2+fs.lower_width-1))    fs.abd[count]=fs.band5[i];
   else     fs.abd[count]=0.0;
  }
 }
 printf("Going into LU factorization...\n");

 /*    Call LAPACK subroutine to perform the LU factorization   */

 sgbtrf_(&fs.m1,&fs.n1,&fs.lower_width,&fs.upper_width,fs.abd,&fs.ldab,fs.piv,&info);

 if (info==0){
  puts("LU decomposition successful...");
 }
 else{
  puts("LU decomposition failed...");
  exit(1);
 }
 fwrite(fs.abd,sizeof(float),count,abddata);
 fclose(abddata);

 fwrite(fs.piv,sizeof(int),ngp,abddata2);
 fclose(abddata2);

 //freefieldparam(fs);
}
unsigned long locatex(GRIDDEF xx[ngp1][ngp2], unsigned long n, float x) {
 unsigned long ju,jm,jl;
 int ascnd;

 jl=0;
 ju=n+1;// n intervals, (n+1) division points
 ascnd=(xx[0][n-1].x > xx[0][1].x);

 while (ju-jl > 1) {
  jm=(ju+jl) >> 1;
  if ((x > xx[0][jm].x) == ascnd)
   jl=jm;
  else
   ju=jm;
 }
```

```
  return jl;
}
inline unsigned long locatey(GRIDDEF xx[ngp1][ngp2], unsigned long n, float x) {
  unsigned long ju,jm,jl;

  jl=0;
  ju=n+1;// n intervals, (n+1) division points
  //ascnd=(xx[n-1][0].y > xx[1][0].y);  Lambert, 2013 April.  Always ascending
  while (ju-jl > 1) {
    jm=(ju+jl) >> 1;
    ( x > xx[jm][0].y ? jl : ju)=jm;
  }
  return jl;
}
```

A.2.3 initialize.cpp

The sub-program "initialize.cpp" loads a uniform distribution of computational particles to begin the simulation. The velocities of each species are set by randomly picking values from the velocity tables created in PROBEPIC.

```
#include "init.hpp"
#include <time.h>

// temporary shouldnt be global
extern FLOAT *random_num;
extern long rancount;

/*This function sets initial plasma conditions and computes plasma
  parameters to be used throughout the rest of the simulation   */
int newparam(SUPERGLOBAL sg, PLASMA &pl, GEOMETRY &ge, INJECTIONSETUP &is,INJECTIONBEAM
&ib,BOUNDARY &bb ,LOCATION &lo, char * Inputfile) {

 FLOAT avveli,avvele,gammai,gammae;

 FLOAT numiperstep_back,numiperstep_front,
  numiperstep_side,numeperstep_back,numeperstep_front,numeperstep_side,fluxi_back,fluxi_front,
  fluxi_side,fluxe_back,fluxe_front,fluxe_side,afront,aback,aside;

//Logfile with system configuration
char Logfile[70]="out.log";
FILE *logfile;

 FILE *in;

 /** _____ Reading Input File _____ **/
    char *line;
    line = (char *) calloc(300,sizeof(char));
    in = fopen(Inputfile,"r");
    int scan=0;
```

```
printf("___________________________________________________ \n");
printf("    Start reading input data from: %s\n\n",Inputfile);

if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=scan=fscanf(in, "%s %d\n",line,&pl.distribution_type);
if (pl.distribution_type==1)
  printf("  EEDF : Maxwellian distribution\n");
else if (pl.distribution_type==0)
  printf("  EEDF : Druyvesteyn distribution\n");
else if (pl.distribution_type==2)
  printf("  EEDF : Bi-MAxwellian distribution\n");
else {
  printf("  EEDF undefined. Choosing defaut= Maxwellian\n");
  pl.distribution_type=1;
}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
printf("\n  ** Plasma parameters\n");
scan=fscanf(in, "%s %f\n",line,&pl.T );
printf("  Plasma temperature of electrons: %2.2f eV \n",pl.T);

if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.T_i);
printf("  Plasma temperature of ions:      %2.2f eV \n",pl.T_i );

if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.n);
printf("  Plasma density:                  %3.1e m^-3 \n",pl.n );

if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.m_ion );
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.q_ion );

if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.m_elec );
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
scan=fscanf(in, "%s %f\n",line,&pl.q_elec );

printf("  Ion mass      %2.3e kg and charge %2.3e \n",pl.m_ion, pl.q_ion );
```

```
   printf("  Electron mass %2.3e kg and charge %2.3e \n",pl.m_elec, pl.q_elec );

   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   printf("\n  ** Geometry\n");
   scan=fscanf(in, "%s %f\n",line,&ge.r_probe );
   printf("  Probe radius    %f\n",ge.r_probe );
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   scan=fscanf(in, "%s %f\n",line,&ge.l_probe);
   printf("  Probe length    %f\n",ge.l_probe );

   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}

   scan=fscanf(in, "%s %f\n",line,&pl.T_h);

   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   if( fgets (line,299,in) == 0 ) {printf("Could not read file \"%s\"\n",Inputfile);fclose(in);return 1;}
   scan=fscanf(in, "%s %f\n",line,&pl.E_t);
   if (pl.distribution_type==2)
   {
   printf("\n  **Additional parameters for Bi-Maxwellian distribution\n");
   printf("  Temperature of warm electrons: %2.2f eV\n",pl.T_h);
   printf("  Energy E_t :                   %2.2f eV\n", pl.E_t);
   }
   printf("\n  >>Finished reading input file\n");
   printf("_____________________________________________________ \n");
   fclose(in);
   free(line);
/** _______________________________ Finished reading _______________________**/

/*   NUMBER OF PARTICLES TO INJECT AT EACH TIME STEP     */
/*
superii_init=40000;
superii_back=2;
superii_front=2;
superii_side=4;
superei_init=40000;
superei_back=5;
superei_front=5;
superei_side=8; */

/********************** PLASMA *******************************************************/
// (all fields)
//pl.T=2.85;
```

```
// pl.T_i=0.1*pl.T; // different temperature for ions
// pl.n=7.0e16;
// pl.m_ion=6.62e-26; //mass of argon atom
// pl.q_ion=1.6e-19;
// pl.m_elec=9.11e-31;
// pl.q_elec=-1.6e-19;
pl.plasma_pot=0.0;

//parameters for two temperature distribution
//pl.T_h=2.9; // [eV] high energy electrons' temperature
//pl.E_t=3.0; // [eV] energy of distribution change from one to another
pl.normalize=sqrt(M_2_PI) * (sqrt(M_PI_2)   //normalization factor for the
  *erff( sqrt(pl.E_t / pl.T) )            // two-temperature maxwell distribution
  - sqrt(2*pl.E_t / pl.T) * exp(-pl.E_t / pl.T))
+ pow(pl.T_h/pl.T,1.5)/sqrt(M_PI)*exp(-pl.E_t/pl.T)*
   (2*sqrt(pl.E_t/pl.T_h) + exp(pl.E_t/pl.T_h)*sqrt(M_PI)*(1-erff(sqrt(pl.E_t/pl.T_h))  )  ) ;

/********************** GEOMETRY ****************************************************/
//ge.r_probe=0.00004445;
//ge.l_probe=0.002;
ge.l_insul=1.5*ge.l_probe;
//The probe radius is equal to (two) cell_height1. 2.0 could be changed
float coefficient=ge.r_probe/0.00003125/2.0;
 printf("init.cpp: coefficient %f", coefficient);
ge.radius=0.00395705*coefficient;
ge.length=4.0*ge.l_probe;

ge.cell_height1=0.00003125*coefficient;
ge.cell_height2=0.00003713*coefficient;
ge.cell_height3=0.00004301*coefficient;
ge.cell_height4=0.00004889*coefficient;
ge.cell_height5=0.00005477*coefficient;
ge.cell_height6=0.00006065*coefficient;

ge.cell_width=0.0000574;
//!*******************************
ge.cell_width = ge.length/(sg.mgp2-1); // ngp2=ngp_horiz
//!*******************************
ge.r_insul=ge.r_probe + ge.cell_height1;

/*  Parameteters needed for field solver    */
ge.jprobe=(long)(ge.r_probe/ge.cell_height1);
ge.kprobe=((long)(ge.length-ge.l_probe-ge.l_insul)/ge.cell_width);

/********************** INJECTION SETUP ********************************************************/
// all of INJECTIONSETUP
//is.injsuperi;
//is.injsupere;
is.superii_init=20000;//40000
is.superii_back=1;//2
is.superii_front=1;//2
is.superii_side=2;//4

is.superei_init=100000;//40000
is.superei_back=5; //5
```

```
is.superei_front=5;//5
is.superei_side=8;//8
// is.rancount;
is.elec_iter=15;

/********************** INJECTION BEAM ***************************************************/
/* injection Beam */
ib.height=2.0 * ge.radius;
ib.width=2.0 * ge.radius;
ib.h=6.9*sqrt((pl.T*11588.7)/(pl.n*1.0e-6)) * 0.01;
ib.wpdt=0.009;
ib.wp=2.0*pi*9000.0*sqrt(pl.n*1.0e-6); // plasma frequency
ib.dt=ib.wpdt/ib.wp;
ib.dte=ib.dt;
ib.dti=15.0*ib.dte;

ib.sigmae=sqrt((boltz*pl.T*11588.7)/pl.m_elec);
ib.sigmai=sqrt((boltz*pl.T_i*11588.7)/pl.m_ion);
/* Boundary Fluxing Setup */

afront=4.0* ge.radius * ge.radius;
aside=2.0 * ge.radius * ge.length;
aback=4.0 * ge.radius * ge.radius - pi*ge.r_insul*ge.r_insul;

avveli=sqrt((8.0*boltz*pl.T_i*11589.7)/(pi*pl.m_ion));
avvele=sqrt((8.0*boltz*pl.T*11589.7)/(pi*pl.m_elec));
gammai=pl.n*avveli/4.0;
gammae=pl.n*avvele/4.0;
fluxi_back=gammai*aback;
fluxe_back=gammae*aback;
fluxi_front=gammai*afront;
fluxe_front=gammae*afront;
fluxi_side=gammai*aside;
fluxe_side=gammae*aside;

numiperstep_back=fluxi_back*ib.dti;
numeperstep_back=fluxe_back*ib.dte;
numiperstep_front=fluxi_front*ib.dti;
numeperstep_front=fluxe_front*ib.dte;
numiperstep_side=fluxi_side*ib.dti;
numeperstep_side=fluxe_side*ib.dte;

ib.superi_back=numiperstep_back/((FLOAT)is.superii_back);
ib.supere_back=numeperstep_back/((FLOAT)is.superei_back);
ib.superi_front=numiperstep_front/((FLOAT)is.superii_front);
ib.supere_front=numeperstep_front/((FLOAT)is.superei_front);
ib.superi_side=numiperstep_side/((FLOAT)is.superii_side);
ib.supere_side=numeperstep_side/((FLOAT)is.superei_side);

/* Initial Loading Injection Setup */
bb.VOLUME=ge.length*ib.height*ib.width - (pi* ge.r_probe * ge.r_probe * ge.l_probe) - (pi* ge.r_insul * ge.r_insul *
ge.l_insul);

ib.superi_init=pl.n * ((pi* ge.radius * ge.radius * ge.length)-
            (pi* ge.r_probe * ge.r_probe * ge.l_probe) - (pi* ge.r_insul * ge.r_insul * ge.l_insul)) /
```

```
((FLOAT)is.superii_init);

  ib.supere_init=pl.n * ((pi* ge.radius * ge.radius * ge.length)-
                (pi* ge.r_probe * ge.r_probe * ge.l_probe) - (pi* ge.r_insul * ge.r_insul * ge.l_insul)) /
((FLOAT)is.superei_init);
  // un-used
  //injsuperi=superii_init;
  //injsupere=superei_init;

  bb.V=0.0;
  bb.current=0.0;
  bb.curri=0.0;
  bb.curre=0.0;

  // LOCATION UNUSED AT THE MO
  lo.x=521288629;
  lo.y=362436069;
  lo.z=16163801;
  lo.c=1;
  lo.n2=1131199299;

       sprintf(Logfile,"./PROBEPIC_Log_Te%1.1f_n%2.1e_Distr%d.log\0",pl.T,pl.n,pl.distribution_type);

printf("\n  Writing logfile...");

  /** ____________________________________ write logfile ____________________________________ **/
    logfile = fopen(Logfile,"a");

fprintf(logfile,"**********************************************************************************************************
************************\n\n");

//get time
time_t rawtime;  struct tm * timeinfo;  time ( &rawtime );  timeinfo = localtime ( &rawtime );

    fprintf(logfile," ** Logfile for PROBEPIC plasma simulation. Thick sheath case **\n");
    fprintf (logfile, "      Current local time and date: %s\n\n", asctime (timeinfo) );

     //Description:

      if (pl.distribution_type==1)
      fprintf(logfile," EEDF : Maxwellian distribution\n");
  else if (pl.distribution_type==0)
      fprintf(logfile," EEDF : Druyvesteyn distribution\n");
  else if (pl.distribution_type==2)
      fprintf(logfile," EEDF : Bi-MAxwellian distribution\n");
  else {
      fprintf(logfile," EEDF undefined!! Choosing defaut= Maxwellian\n");
  }
    fprintf(logfile,"\n ** Plasma parameters **\n");
    fprintf(logfile," Plasma temperature of electrons: %2.2f eV \n",pl.T);
    fprintf(logfile," Plasma temperature of ions:      %2.2f eV \n",pl.T_i );
    fprintf(logfile," Plasma density:                  %2.2e m^-3 \n",pl.n );
       if (pl.distribution_type==2)
    {
    fprintf(logfile,"\n** Additional parameters for Bi-Maxwellian distribution\n");
```

```
    fprintf(logfile," Temperature of warm electrons: %2.2f eV\n",pl.T_h);
    fprintf(logfile," Energy E_t : %2.2f eV\n", pl.E_t);
    }
    fprintf(logfile,"\n** Particle types\n");
    fprintf(logfile," Ion mass      %3.3e kg and charge %3.3e \n",pl.m_ion, pl.q_ion );
    fprintf(logfile," Electron mass %3.3e kg and charge %3.3e \n",pl.m_elec, pl.q_elec );

    fprintf(logfile,"\n** Geometry\n");
    fprintf(logfile," Probe radius %f\n",ge.r_probe );
    fprintf(logfile," Probe length %f\n",ge.l_probe );
    fprintf(logfile," Aspect ratio: length/r %f \n",ge.l_probe/ge.r_probe );

    fprintf(logfile,"\n** Simulation domain\n");
    fprintf(logfile," Radius of the cell %f\n", ge.radius);
    fprintf(logfile," Simulation cell radius divided by probe radius %f\n", ge.radius/ge.r_probe);
    //checking cell dimensions
    if (ge.r_probe>ge.radius||ge.l_probe>ge.radius)
    fprintf(logfile,"WARNING!: Increase the radius of the cell. It is too small or the probe is too big!!\n");
    //checking cell dimensions
    fprintf(logfile," Length of the cell %f\n",ge.length);
    fprintf(logfile," Simulation cell length divided by probe length %f\n", ge.length/ge.l_probe);
    if (ge.r_probe>ge.length||ge.l_probe>ge.length)
    fprintf(logfile,"WARNING!: Increase the length of the cell. It is too small or the probe is too big!!\n");

    fprintf(logfile, "\n" );
    fprintf(logfile," Initial number of superions %ld and superelectrons %ld \n",is.superii_init,is.superei_init);
    fprintf(logfile, " Average velocity of ions %2.3f ; of electrons %2.3e\n", avveli, avvele);
    fprintf(logfile, " Plasma frequency %f ; Debey length %f\n", ib.wp, ib.h);
       fprintf(logfile,"\n** Rescaled parameters:\n");
    fprintf(logfile, " Rescaled probe size: r_probe/Debey_length = %f\n", ge.r_probe / ib.h);
    fprintf(logfile," Rescaled simulation domain radius: radius/Debey_length %f\n",ge.radius / ib.h);
    fprintf(logfile," Rescaled cell height: grid height/Debey_length %f\n", ge.cell_height1 / ib.h);

    printf("  Logfile written.\n");
    fclose(logfile);
  /** ________________________________  <-- ________________________________________ **/
}
void initfieldparam(SUPERGLOBAL &sg, FIELDSOLVER &fs ){

 /*   FIELD SOLVER GLOBAL VARIABLES   */
  fs.test1=0;
  fs.test2=0;
  fs.test3=0;
  fs.ldab=3*sg.mgp2+1;
  fs.n1=sg.mgp;
  fs.m1=sg.mgp;
  fs.ldb=sg.mgp;
  fs.upper_width=sg.mgp2;
  fs.lower_width=sg.mgp2;
  fs.info=0;
  fs.nrhs=0;
  fs.piv=(int *)malloc(sg.mgp*sizeof(int));

  fs.band1=(FLOAT*)malloc(sg.mgp*sizeof(FLOAT));
  fs.band2=(FLOAT*)malloc(sg.mgp*sizeof(FLOAT));
```

```
  fs.band3=(FLOAT*)malloc(sg.mgp*sizeof(FLOAT));
  fs.band4=(FLOAT*)malloc(sg.mgp*sizeof(FLOAT));
  fs.band5=(FLOAT*)malloc(sg.mgp*sizeof(FLOAT));

  /*  Large array initialization  */
  fs.abd=(FLOAT *)malloc((2*fs.lower_width+fs.upper_width+2)*sg.mgp*sizeof(FLOAT));
  fs.b=(FLOAT *)malloc(sg.mgp*sizeof(FLOAT));
  fs.phi=(FLOAT *)malloc(sg.mgp*sizeof(FLOAT));
}
void freefieldparam(FIELDSOLVER &fs){

  free(fs.band1);
  free(fs.band2);
  free(fs.band3);
  free(fs.band4);
  free(fs.band5);

  /*  Large array initialization  */
  free(fs.piv);
  free(fs.abd);
  free(fs.b);
  free(fs.phi);

}
long initialize(PARTICLEDEF particle[],long  init_num,FLOAT super_init,FLOAT *v_table,FLOAT m,FLOAT
q,GEOMETRY &ge ) {
  long idx,rannum;
  FLOAT theta,phi,v,r;

  for(idx=0;idx < init_num;idx++){

    theta=pi*random_num[rancount++];
    phi=2.0*pi*random_num[rancount++];
    rannum=((long)V_TABLE_RES*random_num[rancount++]);

    v=v_table[rannum];
    rancount++; // nb only hear for compaitibility
    r=ge.radius * sqrt(random_num[rancount++]);

    particle[idx].x=-ge.length * random_num[rancount++];

    particle[idx].y = r*sin(theta);
    particle[idx].z = r*cos(theta);
    particle[idx].r = sqrt(particle[idx].y*particle[idx].y + particle[idx].z *particle[idx].z);
    particle[idx].vx = v * sin(theta) *sin(phi);
    particle[idx].vy = v * cos(theta);
    particle[idx].vz = v * sin(theta) * cos(phi);
    particle[idx].Ex = 0.0;
    particle[idx].Ey = 0.0;
    particle[idx].q = q * super_init;
    particle[idx].m = m * super_init;
  }
  return init_num;
}
inline float erff(float x) {
```

```
 float erf = (float)gsl_sf_erf((double) x);
 return erf;
};
void make_velocity_table(PLASMA pl,FLOAT *v_table_e, FLOAT *v_table_i,long sz) {

 int converge;
 long  i,num_div,temp;
 float  accuracy,prob_delta,p_calc,p_exact,delta_v,v,error;

 /*  Initialize parameters  */

 num_div=sz;
 prob_delta=1.0/((float)num_div);
 accuracy=0.1*prob_delta;           /* 10 percent of spacing  */
 p_exact = 0.0;
 v = 0.0;

 /*  Electrons   */
 for (i=1; i < num_div; i++) {
  temp = (long) v;    /* The next two lines set the numbers past the decimal point to zero.   */
  v = (float) temp;   /*  There is probably a more compact way to do this.                    */
  p_exact+=prob_delta;
  if (p_exact > 1.0) p_exact = 1.0; // due to arithmetic round up, p_exact may exceed 1.0
  delta_v = 1.0;
  converge = 0;
  do {
            if (pl.distribution_type==1) { //Maxwellian
            p_calc=sqrt(M_2_PI) * pow(pl.m_elec / (boltz * pl.T * 11588.7),1.5)
       * (sqrt(M_PI_2) *  pow(boltz * pl.T *11588.7 / pl.m_elec,1.5)
          *erff(sqrt(pl.m_elec / (2.0 * boltz * pl.T * 11588.7)) * v)
          - boltz * pl.T * 11588.7 * v / pl.m_elec * exp(-pl.m_elec * v * v
                                        / (2.0 * boltz * pl.T * 11588.7)));
                                              }
            else if (pl.distribution_type==0) //Druyvesteyn
            { float en = pl.m_elec * v * v/2.0;
            float kT=boltz * pl.T * 11588.7;
            p_calc=1-0.8160489390982629*gsl_sf_gamma_inc(0.75, (0.243*en*en)/kT/kT);
              //printf("probability1 %f or exact %f\n",p_calc, p_exact);
                  }
  else if (pl.distribution_type==2) //Bi-Maxwellian
                  {    float en = pl.m_elec * v * v/2.0;
                  //rescale temperatures
                  float kT1=boltz * pl.T * 11588.7;float kT2=boltz * pl.T_h * 11588.7;
                  //rescale energy of switching between two temperatures
                  float et=pl.E_t*pl.q_ion;
            if ( en < et )
                  {//low energy part of distribution
                  p_calc = sqrt(M_2_PI) * (sqrt(M_PI_2)   //normalization factor for the distribution
              *erff( sqrt(en/kT1) )
                              - sqrt(2*en/kT1) * exp(-en / kT1));
                  p_calc/=pl.normalize;
                  //printf("%lu. probability1 %.20f or error %f delta %f v %f \n", i,p_calc, error,en/et, v);
                  }
            else if ( en > et || en == et )
                  {   //high energy part of distribution
```

```
                    p_calc= sqrt(M_2_PI) * (sqrt(M_PI_2)   //normalization factor for the distribution
                                *erff( sqrt(pl.E_t / pl.T) )
                            - sqrt(2*pl.E_t / pl.T) * exp(-pl.E_t / pl.T))+
                            pow(pl.T_h/pl.T,1.5)/sqrt(M_PI)*exp(-en/kT2-pl.E_t/pl.T)*
                            (-2*exp(pl.E_t/pl.T_h)*sqrt(en/kT2)+exp(en/kT2)*
            (2*sqrt(pl.E_t/pl.T_h) + exp(pl.E_t/pl.T_h)*sqrt(M_PI)*(erff(sqrt(en/kT2))-erff(sqrt(pl.E_t/pl.T_h)) ) ) );
                    //normalize the distribution
                    p_calc/=pl.normalize;
     //  printf("%lu. probability2 %.20f or error %f norm %f v %f \n", i,p_calc,
error,pl.normalize/1.1827728497625067, v);
                    }
            }
                //NOTE: here another distributions could be implemented
                //using the pattern "else if (distribution==n){p_calc = another distribution}

    error = p_exact - p_calc;

    if (fabs(error) < accuracy)
        converge = 1;
    else if (error > 0)
        v+=delta_v;
    else {
        v-=delta_v;
        delta_v*=1.0e-1;
    }
   } while(!converge);
   v_table_e[i-1]=v;
  }
  v_table_e[num_div-1]= v_table_e[num_div-2] + v_table_e[num_div-2] - v_table_e[num_div-3];    /*  Avoid
singularity v = infinity when p_exact = 1  */

  printf("Done creating electron velocity table...\n");

  /*  Reset parameters  */
  p_exact = 0.0;
  v = 0.0;

accuracy=prob_delta;
  /*  Ions  */
  for (i=1; i < num_div; i++) {

    temp = (long) v;    /* The next two lines set the numbers past the decimal point to zero.  */
    v = (float) temp;   /*  There is probably a more compact way to do this.                */
    p_exact+=prob_delta;
    if (p_exact > 1.0) p_exact = 1.0; // due to arithmetic round up, p_exact may exceed 1.0
    delta_v = 1.0;
    converge = 0;
    do {
                if (pl.distribution_type==1)
                { //Maxwell distribution
                p_calc=sqrt(M_2_PI) * pow((pl.m_ion ) / (boltz * pl.T_i * 11588.7),1.5)
                * (sqrt(M_PI_2) *  pow(boltz * pl.T_i *11588.7 / (pl.m_ion ),1.5)
            *erff(sqrt((pl.m_ion ) / (2.0 * boltz * pl.T_i * 11588.7))  * v)
            - boltz * pl.T_i * 11588.7 * v / (pl.m_ion ) * exp(-(pl.m_ion ) * v * v
                                        / (2.0 * boltz * pl.T_i * 11588.7))); }
```

```
            else if (pl.distribution_type==0)
            { //Druyvesteyn distribution
            float en = pl.m_ion * v * v/2.0; //rescale energy
            float kT=boltz * pl.T_i * 11588.7;//rescale temperature to eV
            p_calc=1-0.8160489390982629*gsl_sf_gamma_inc(0.75, (0.243*en*en)/kT/kT);
             //printf("probability2 %f or exact %f\n",p_calc, p_exact);
                  }
            else if (pl.distribution_type==2) {
                  //Maxwellian for ions
            p_calc=sqrt(M_2_PI) * pow((pl.m_ion ) / (boltz * pl.T_i * 11588.7),1.5)
            * (sqrt(M_PI_2) *  pow(boltz * pl.T_i *11588.7 / (pl.m_ion ),1.5)
            *erff(sqrt((pl.m_ion ) / (2.0 * boltz * pl.T_i * 11588.7))  * v)
            - boltz * pl.T_i * 11588.7 * v / (pl.m_ion ) * exp(-(pl.m_ion ) * v * v
             / (2.0 * boltz * pl.T_i * 11588.7))); }

    error = p_exact - p_calc;

    if (fabs(error) < accuracy)
        converge = 1;
    else if (error > 0)
        v+=delta_v;
    else {
        v-=delta_v;
        delta_v*=1.0e-1;
    }
   } while(!converge);
   v_table_i[i-1]=v;
  }
  v_table_i[num_div-1]= v_table_i[num_div-2] + v_table_i[num_div-2] - v_table_i[num_div-3];     /*  Avoid singularity
v = infinity when p_exact = 1  */
  printf("Done creating ion velocity table...\n");
}
/*   This function moves the particles   */
void mover(PARTICLEDEF particle[],long  num_part,FLOAT timestep) {

  long  idx;
  FLOAT accelx,accelr, sintheta, costheta, art, qom;

  for (idx=0; idx < num_part; idx++) {
   qom = particle[idx].q / particle[idx].m;
   accelx= qom * particle[idx].Ex;
   accelr= qom * particle[idx].Ey;
   sincosf(atan2f(particle[idx].y,particle[idx].z), &sintheta, &costheta);
   particle[idx].vx += accelx * timestep;
   art = accelr * timestep;
   particle[idx].vy += art * sintheta;
   particle[idx].vz += art * costheta;
   particle[idx].x += particle[idx].vx * timestep;
   particle[idx].y += particle[idx].vy * timestep;
   particle[idx].z += particle[idx].vz * timestep;
   particle[idx].r = hypotf(particle[idx].y, particle[idx].z);
  }
}
/*   This function sorts out which particles have crossed a boundary,i.e.
     the probe or extent of computational domain.  Particles crossing the
```

```
    probe boundary are counted as current   */
// iflag 0 -elections 1 ions
long boundary (int iflag, PARTICLEDEF particle[],long num_part,PLASMA &pl, GEOMETRY &ge,BOUNDARY
&bb) {
  long  idx;
  long  i2;
  long  numtemp;
  FLOAT r_part;

  numtemp=num_part;
  i2=-1;

  for (idx=0;idx < numtemp;idx++){
   i2++;
   r_part=particle[i2].r;

   /*   OUT OF COMPUTATIONAL DOMAIN        */
   if (particle[i2].x >0.0 || particle[i2].x < -ge.length ||
         particle[i2].y >ge.radius || particle[i2].y < -ge.radius ||
         particle[i2].z >ge.radius || particle[i2].z < -ge.radius) {
       --num_part;
        particle[i2].x=particle[num_part].x;
        particle[i2].y=particle[num_part].y;
        particle[i2].z=particle[num_part].z;
          particle[i2].r=particle[num_part].r;
          particle[i2].vx=particle[num_part].vx;
          particle[i2].vy=particle[num_part].vy;
          particle[i2].vz=particle[num_part].vz;
          particle[i2].q=particle[num_part].q;
          particle[i2].m=particle[num_part].m;

          --i2;
   /*   CONTACT WITH PROBE      */
   } else if (r_part <= ge.r_probe &&
             particle[i2].x >=-(ge.l_probe+ge.l_insul) &&
             particle[i2].x <=-ge.l_insul) {
       --num_part;

       particle[i2].x=particle[num_part].x;
       particle[i2].y=particle[num_part].y;
       particle[i2].z=particle[num_part].z;
       particle[i2].r=particle[num_part].r;
       particle[i2].vx=particle[num_part].vx;
       particle[i2].vy=particle[num_part].vy;
       particle[i2].vz=particle[num_part].vz;
       particle[i2].m=particle[num_part].m;
       particle[i2].q=particle[num_part].q;

       bb.current += particle[i2].q;

       (iflag==1 ? bb.curri :bb.curre) +=particle[i2].q;
       --i2;
    } // end else
  }// end for
```

```
  return num_part;
}
long inject(int iflag,  // [I] 0-elections, 1 -ions
          PARTICLEDEF particle[],
          long  num_part,
          float *v_table,
            SUPERGLOBAL &sg,
            PLASMA &pl,
            GEOMETRY ge,
            INJECTIONBEAM &ib,
          INJECTIONSETUP &is
            ) {
  long  i,rannum;
  float theta,phi,v;
  float m;
  float q;

  float super_back;
  float super_front;
  float super_side;
  long  superinj_back;
  long  superinj_front;
  long  superinj_side;

  // electons
  if(iflag==0){
    m=pl.m_elec;
    q=pl.q_elec;

    super_back=ib.supere_back;
    super_front=ib.supere_front;
    super_side=ib.supere_side;

    superinj_back=is.superei_back;
    superinj_front=is.superei_front;
    superinj_side=is.superei_side;
    // ion
  }else{
    m=pl.m_ion;
    q=pl.q_ion;

    super_back=ib.superi_back;
    super_front=ib.superi_front;
    super_side=ib.superi_side;

    superinj_back=is.superii_back;
    superinj_front=is.superii_front;
    superinj_side=is.superii_side;
  }
  // NPTS; no. of elements of array 'random_num'
  if (NPTS<rancount+5*superinj_front) rancount=0; //! make sure that 'racount' stays in [0..NPTS]

  for(i=0;i < superinj_front;i++){
    /*     FRONT      */
    theta=acos(sqrt(1 - random_num[rancount]));
```

```
  rancount ++;
  phi=2.0*pi*random_num[rancount];
  rancount ++;
  rannum=sg.mpart*random_num[rancount];
  rancount ++;
  v=v_table[rannum];
  particle[num_part].x=-ge.length;
  particle[num_part].y=ib.height * random_num[rancount] - ib.height / 2.0;
  rancount ++;
  particle[num_part].z=ib.width * random_num[rancount] - ib.width / 2.0;
  rancount ++;
  particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                          * particle[num_part].z);
  particle[num_part].vx=v * cos(theta);
  particle[num_part].vy=v*sin(theta)*cos(phi);
  particle[num_part].vz=v*sin(theta)*sin(phi);
  particle[num_part].q= q * super_front;
  particle[num_part].m= m * super_front;
  particle[num_part].Ex=0.0;
  particle[num_part].Ey=0.0;
  num_part++;
  if (num_part >= sg.mpart) {
    printf("*** Too many particles injected: increase 'sg.mpart' in lprobe60.h ****");
    exit(1);
  };
}
// NPTS; no. of elements of array 'random_num'
if (NPTS<rancount+3) rancount=0; //! make sure that 'racount' stays in [0..NPTS]

for(i=0;i < superinj_back;i++){
  /*    BACK    */
  theta=acos(sqrt(1 - random_num[rancount]));
  rancount ++;
  phi=2.0*pi*random_num[rancount];
  rancount ++;
  rannum=sg.mpart*random_num[rancount];
  rancount ++;
  v=v_table[rannum];

  do{
    // NPTS; no. of elements of array 'random_num'
    if (NPTS<rancount+2) rancount=0; //! make sure that 'racount' stays in [0..NPTS]

    particle[num_part].x=0.0;
    particle[num_part].y=ib.height * random_num[rancount] - ib.height / 2.0;
    rancount ++;
    particle[num_part].z=ib.width * random_num[rancount] - ib.width / 2.0;
    rancount ++;
    particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                            * particle[num_part].z);
    particle[num_part].vx=-v * cos(theta);
    particle[num_part].vy=v*sin(theta)*cos(phi);
    particle[num_part].vz=v*sin(theta)*sin(phi);
    particle[num_part].Ex=0.0;
    particle[num_part].Ey=0.0;
```

```
   particle[num_part].q= q * super_back;
   particle[num_part].m= m * super_back;
  }while( particle[num_part].r <= ge.r_insul);
  num_part++;
  if (num_part >= sg.mpart) {
   printf("*** Too many particles injected: increase 'sg.mpart' in lprobe60.h ****");
   exit(1);
  };
 }
 // NPTS; no. of elements of array 'random_num'
 if (NPTS<rancount+20*superinj_side) rancount=0; // make sure that 'racount' stays in [0..NPTS]
 for(i=0;i < superinj_side;i++){
  /*     BOTTOM  SIDE            */
  theta=acos(sqrt(1 - random_num[rancount]));
  rancount ++;
  phi=2.0*pi*random_num[rancount];
  rancount ++;
  rannum=sg.mpart*random_num[rancount];
  rancount ++;
  v=v_table[rannum];
  particle[num_part].x=-ge.length * random_num[rancount];
  rancount ++;
  particle[num_part].y=-ge.radius;
  particle[num_part].z=ib.width * random_num[rancount] - ib.width / 2.0;
  rancount ++;
  particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                          * particle[num_part].z);
  particle[num_part].vx=v * sin(theta) *sin(phi);
  particle[num_part].vy=v * cos(theta);
  particle[num_part].vz=v*sin(theta)*cos(phi);
  particle[num_part].Ex=0.0;
  particle[num_part].Ey=0.0;
  particle[num_part].q= q * super_side;
  particle[num_part].m= m * super_side;
  num_part++;
  if (num_part >= sg.mpart) {
   printf("*** Too many particles injected: increase 'sg.mpart' in lprobe60.h ****");
   exit(1);
  };
  /*     TOP    Side    */
  theta=acos(sqrt(1 - random_num[rancount]));
  rancount ++;
  phi=2.0*pi*random_num[rancount];
  rancount ++;
  rannum=sg.mpart*random_num[rancount];
  rancount ++;
  v=v_table[rannum];
  particle[num_part].x=-ge.length * random_num[rancount];
  rancount ++;
  particle[num_part].y=ge.radius;
  particle[num_part].z=ib.width * random_num[rancount] - ib.width / 2.0;
  rancount ++;
  particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                          * particle[num_part].z);
  particle[num_part].vx=v * sin(theta) * sin(phi);
```

```
particle[num_part].vy=-v * cos(theta);
particle[num_part].vz=v * sin(theta) * cos(phi);
particle[num_part].Ex=0.0;
particle[num_part].Ey=0.0;
particle[num_part].q= q * super_side;
particle[num_part].m= m * super_side;
(num_part)++;
if (num_part >= sg.mpart) {
  printf("*** Too many particles injected: increase 'sg.mpart' in lprobe60.h ****");
  exit(1);
};
/*    FRONT SIDE              */
theta=acos(sqrt(1 - random_num[rancount]));
rancount ++;
phi=2.0*pi*random_num[rancount];
rancount ++;
rannum=sg.mpart*random_num[rancount];
rancount ++;
v=v_table[rannum];
particle[num_part].x=-ge.length * random_num[rancount];
rancount ++;
particle[num_part].y=ib.height * random_num[rancount] - ib.height / 2.0;
rancount ++;
particle[num_part].z=ge.radius;
particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                    * particle[num_part].z);
particle[num_part].vx=v * sin(theta) * cos(phi);
particle[num_part].vy=v * sin(theta) * sin(phi);
particle[num_part].vz=-v * cos(theta);
particle[num_part].Ex=0.0;
particle[num_part].Ey=0.0;
particle[num_part].q= q * super_side;
particle[num_part].m= m * super_side;
(num_part)++;
if (num_part >= sg.mpart) {
  printf("*** Too many particles injected: increase 'sg.mpart' in lprobe60.h ****");
  exit(1);
};
/*   BACK SIDE   */
theta=acos(sqrt(1 - random_num[rancount]));
rancount ++;
phi=2.0*pi*random_num[rancount];
rancount ++;
rannum=sg.mpart*random_num[rancount];
rancount ++;
v=v_table[rannum];
particle[num_part].x=-ge.length * random_num[rancount];
rancount ++;
particle[num_part].y=ib.height * random_num[rancount] - ib.height / 2.0;
rancount ++;
particle[num_part].z=-ge.radius;
particle[num_part].r=sqrt(particle[num_part].y*particle[num_part].y + particle[num_part].z
                    * particle[num_part].z);
particle[num_part].vx=v * sin(theta) * cos(phi);
particle[num_part].vy=v * sin(theta) * sin(phi);
```

```
    particle[num_part].vz=v * cos(theta);
    particle[num_part].Ex=0.0;
    particle[num_part].Ey=0.0;
    particle[num_part].q= q * super_side;
    particle[num_part].m= m * super_side;
    (num_part)++;
    if (num_part >= sg.mpart) {
      printf("*** Too many particles injected: increase 'numpart' in lprobe60.h ****");
      exit(1);
    };
  }
  return num_part;
}
```

A.2.4 init.cpp

```
#ifndef INIT_H
 #define INIT_H

#include <gsl/gsl_sf_erf.h>
#include <gsl/gsl_sf_gamma.h>
#include "lprobe60.h"

void sincosf(float x, float *sin, float *cos);

int newparam(SUPERGLOBAL sg, PLASMA &pl, GEOMETRY &ge, INJECTIONSETUP
&is,INJECTIONBEAM &ib,BOUNDARY &bb ,LOCATION &lo,char * Inputfile);
void initfieldparam(SUPERGLOBAL &sg, FIELDSOLVER &fs );
void freefieldparam(FIELDSOLVER &fs);
long initialize(PARTICLEDEF particle[],long  init_num,FLOAT super_init,FLOAT *v_table,FLOAT
m,FLOAT q,GEOMETRY &ge );
void make_velocity_table(PLASMA pl,FLOAT *v_table_e, FLOAT *v_table_i,long sz);
void mover(PARTICLEDEF particle[],long  num_part,FLOAT timestep);
long boundary (int iflag, PARTICLEDEF particle[],long num_part,PLASMA &pl, GEOMETRY
&ge,BOUNDARY &bb);
long inject(int iflag, PARTICLEDEF particle[], long  num_part, float *v_table, SUPERGLOBAL &sg,
PLASMA &pl, GEOMETRY ge, INJECTIONBEAM &ib, INJECTIONSETUP &is);

#endif
```

A.2.5 lprobe.h

This is PROBEPIC's header file that includes some of the codes overall design philosophy. Structures were used in the program for clarity. For example, particles (electrons, ions, etc.) are of the type PARTICLEDEF—which has the associated properties of position, velocity, charge, etc.

```
#ifndef LPROBE60_H
 #define LPROBE60_H

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <gsl/gsl_sf_erf.h>
#include <vector>

#define FLOAT float

#define EPS0 8.85e-12
#define EPSAlO3 10.44e-12
#define pi M_PI

//#define ngp1 74
//#define ngp2 74
//#define ngp (ngp1*ngp2)

#define numpart 800000
#define boltz 1.38e-23
#define RAD 0.00378864
#define NPTS 1000005 // ! # of data points
#define V_TABLE_RES numpart

#define noBins 100
#define OUTPUT_FREQUENCY 100L

// some thing to do with fortran C++ wrappers
// won't compile on gcc-.4.7.3 unless present
//void *__gxx_personality_v0;

typedef unsigned long int unlong;

/*  SUPERGLOBAL   */
typedef struct{
  long  mgp1;
  long  mgp2;
  long  mgp;
  long  mpart;
  long  V_TABLE_RES2;
} SUPERGLOBAL;

typedef struct {
  float x,y,q_dens,Ey,Ex,phi;
} GRIDDEF;

typedef struct {
  float x,y,z,r,vx,vy,vz,Ex,Ey,q,m;
  unsigned long cellx,celly;
} PARTICLEDEF;

//--- BINS -------------
typedef struct {
  float r,v;
```

```
}GRIDDEF_T;

typedef std::vector<std::vector<GRIDDEF> > VGRID;

//long int ngp_vert,ngp_horiz;

/* SYSTEM GEOMETRY */
typedef struct {
 //  long  ngp_vert;
 //long  ngp_horiz;
 long jprobe;
 long kprobe;
 long jinsul;
 long kinsul;

 float cell_height1;
 float cell_height2;
 float cell_height3;
 float cell_height4;
 float cell_height5;
 float cell_height6;
 float cell_height;
 float cell_width;
 float r_probe;
 float r_insul;
 float l_probe;
 float l_insul;
 float radius;
 float length;
} GEOMETRY;

/*  PLASMA PARAMETER  VARIABLES   */
typedef struct{
 int distribution_type;
 float T;
 float T_i; //ion temperature
 float n;
 float m_ion;
 float q_ion;
 float m_elec;
 float q_elec;
 float m_part;
 float plasma_pot;
    //parameters for two temperature distribution
 float T_h; // [eV] high energy electrons' temperature
 float E_t; // [eV] energy of distribution change from one to another
 float normalize; //normalization factor for the distribution
} PLASMA;

/*  INJECTION SETUP  VARIABLES    */
typedef struct{
 long  injsuperi;
 long  injsupere;
 long  superii_init;
 long  superii_back;
```

```
  long  superii_front;
  long  superii_side;
  long  superei_init;
  long  superei_back;
  long  superei_front;
  long  superei_side;
  long  elec_iter;
} INJECTIONSETUP;

typedef struct{
  float supere;
  float superi;
  float sigma;
  float sigmae;
  float sigmai;
  float wpdt;
  float h;
  float wp;
  float dt;

  float superi_init;
  float superi_back;
  float superi_front;
  float superi_side;
  float supere_init;
  float supere_back;
  float supere_front;
  float supere_side;

  float height;
  float width;
  float dti;
  float dte;

  //float *random_num;
  //float *v_table_i;
  //float *v_table_e;
} INJECTIONBEAM;

/*  GRAPHICS  VARIABLES  */
typedef struct{
  long view;
  long page;
} GRAPHICS;

/*  BOUNDARY  VARIABLES  */
typedef struct{
  float V;
  float current;
  float VOLUME;
  float curri;
  float curre;
} BOUNDARY;

/*  FIELD SOLVER  VARIABLES  */
```

```
typedef struct {
  int test1;
  int test2;
  int test3;
  int ldab;
  int n1;
  int m1;
  int ldb;
  int upper_width;
  int lower_width;
  int info;
  int nrhs;
  int *piv;

  float *band1;
  float *band2;
  float *band3;
  float *band4;
  float *band5;
  float *abd;
  float *b;
  float *phi;

} FIELDSOLVER;

typedef struct{
  unsigned long x;
  unsigned long y;
  unsigned long z;
  unsigned long c;
  unsigned long n2;
} LOCATION;

#endif
```

A.2.6 random_number_v1.cpp

The generation of pseudo-random numbers is an important element of any PIC or PIC/Monte Carlo simulation. The quality of the random numbers plays a major role in the statistical accuracy of the simulation, which is determined by testing the generator on multiple platforms.

```
  //#define EXTERN extern
  //#include "lprobe60.h"

  /* Common Block Declarations */
  struct klotz0_1_ {
     FLOAT buff[607];
     long int ptr;
  };
```

```
#define klotz0_1 (*(struct klotz0_1_ *) &klotz0_)
#define min(a,b) (a<b)?a:b

struct klotz1_1_ {
   FLOAT xbuff[1024];
   long int first, xptr;
};
//#define klotz1_1 (*(struct klotz1_1_ *) &klotz1_)

/* Initialized data */
struct KLOTZ {
   long int fill_1[1214];
   long int e_2;
   } klotz0_ = { {0}, 0 };

//struct {
//    FLOAT fill_1[1024];
//    long int e_2[2];
//    FLOAT e_3;
//    } klotz1_ = { {0}, 0, 0, 0. };

int zufall_(long int n, FLOAT *a)
//long int n;
//FLOAT *a;
{
   long int buffsz = 607;

   long int left, aptr, bptr, aptr0, i, k, q;
   FLOAT t;
   long int nn, vl, qq, k273, k607, kptr;

/* portable lagged Fibonacci series uniform random number */
/* generator with "lags" -273 und -607: */
/* W.P. Petersen, IPS, ETH Zuerich, 19 Mar. 92 */

   aptr = 0;
   nn = n;

L1:

   if (nn <= 0) {
        return 0;
   }
/* factor nn = q*607 + r */

   q = (nn - 1) / 607;
   left = buffsz - klotz0_1.ptr;

   if (q <= 1) {

/* only one or fewer full segments */

        if (nn < left) {
          kptr = klotz0_1.ptr;
            for (i = 0; i < nn; ++i) {
```

```
            a[i + aptr] = klotz0_1.buff[kptr + i];
          }
          klotz0_1.ptr += nn;
          return 0;
        } else {
         kptr = klotz0_1.ptr;
/*pragma _CRI ivdep*/
          for (i = 0; i < left; ++i) {
            a[i + aptr] = klotz0_1.buff[kptr + i];
          }
          klotz0_1.ptr = 0;
          aptr += left;
          nn -= left;
/*  buff -> buff case */
          vl = 273;
          k273 = 334;
          k607 = 0;
          for (k = 0; k < 3; ++k) {
/*pragma _CRI ivdep*/
            for (i = 0; i < vl; ++i) {
              t = klotz0_1.buff[k273+i]+klotz0_1.buff[k607+i];
              klotz0_1.buff[k607+i] = t - (FLOAT) ((long int) t);
            }
            k607 += vl;
            k273 += vl;
            vl = 167;
            if (k == 0) {
               k273 = 0;
            }
          }
          goto L1;
        }
   } else {
/* more than 1 full segment */

      kptr = klotz0_1.ptr;
/*pragma _CRI ivdep*/
        for (i = 0; i < left; ++i) {
          a[i + aptr] = klotz0_1.buff[kptr + i];
        }
        nn -= left;
        klotz0_1.ptr = 0;
        aptr += left;

/* buff -> a(aptr0) */

        vl = 273;
        k273 = 334;
        k607 = 0;
        for (k = 0; k < 3; ++k) {
          if (k == 0) {
/*pragma _CRI ivdep*/
            for (i = 0; i < vl; ++i) {
              t = klotz0_1.buff[k273+i]+klotz0_1.buff[k607+i];
              a[aptr + i] = t - (FLOAT) ((long int) t);
```

```
            }
            k273 = aptr;
            k607 += vl;
            aptr += vl;
            vl = 167;
          } else {
/*pragma _CRI ivdep*/
            for (i = 0; i < vl; ++i) {
               t = a[k273 + i] + klotz0_1.buff[k607 + i];
               a[aptr + i] = t - (FLOAT) ((long int) t);
            }
            k607 += vl;
            k273 += vl;
            aptr += vl;
          }
        }
        nn += -607;

/* a(aptr-607) -> a(aptr) for last of the q-1 segments */

        aptr0 = aptr - 607;
        vl = 607;

        for (qq = 0; qq < q-2; ++qq) {
           k273 = aptr0 + 334;
/*pragma _CRI ivdep*/
           for (i = 0; i < vl; ++i) {
             t = a[k273 + i] + a[aptr0 + i];
             a[aptr + i] = t - (FLOAT) ((long int) t);
           }
           nn += -607;
           aptr += vl;
           aptr0 += vl;
        }

/* a(aptr0) -> buff, last segment before residual */

        vl = 273;
        k273 = aptr0 + 334;
        k607 = aptr0;
        bptr = 0;
        for (k = 0; k < 3; ++k) {
           if (k == 0) {
/*pragma _CRI ivdep*/
             for (i = 0; i < vl; ++i) {
                t = a[k273 + i] + a[k607 + i];
                klotz0_1.buff[bptr + i] = t - (FLOAT) ((long int) t);
             }
             k273 = 0;
             k607 += vl;
             bptr += vl;
             vl = 167;
           } else {
/*pragma _CRI ivdep*/
             for (i = 0; i < vl; ++i) {
```

```
                t = klotz0_1.buff[k273 + i] + a[k607 + i];
                klotz0_1.buff[bptr + i] = t - (FLOAT) ((long int) t);
            }
            k607 += vl;
            k273 += vl;
            bptr += vl;
          }
        }
        goto L1;
    }
} /* zufall_ */
long int zufalli_(long int  seed)
//long int seed;
{
    /* Initialized data */

    long int kl = 9373;
    long int ij = 1802;

    /* Local variables */
    long int i, j, k, l, m;
    FLOAT s, t;
    long int ii, jj;

/*  generates initial seed buffer by linear congruential */
/*  method. Taken from Marsaglia, FSU report FSU-SCRI-87-50 */
/*  variable seed should be 0 < seed <31328 */

    if (seed != 0) {
        ij = seed;
    }
    i = ij / 177 % 177 + 2;
    j = ij % 177 + 2;
    k = kl / 169 % 178 + 1;
    l = kl % 169;
    for (ii = 0; ii < 607; ++ii) {
        s = 0.;
        t = .5;
        for (jj = 1; jj <= 24; ++jj) {
            m = i * j % 179 * k % 179;
            i = j;
            j = k;
            k = m;
            l = (l * 53 + 1) % 169;
            if (l * m % 64 >= 32) {
              s += t;
            }
            t *= (FLOAT).5;
        }
        klotz0_1.buff[ii] = s;
    }
    return 0;
} /* zufalli_ */
long int zufallsv_(FLOAT *svblk)
//FLOAT *svblk;
```

```
{
    long int i;

/*  saves common blocks klotz0, containing seeds and */
/*  polong inter to position in seed block. IMPORTANT: svblk must be */
/*  dimensioned at least 608 in driver. The entire contents */
/*  of klotz0 (polong inter in buff, and buff) must be saved. */

    /* Function Body */
    svblk[0] = (FLOAT) klotz0_1.ptr;
/*pragma _CRI ivdep*/
    for (i = 0; i < 607; ++i) {
        svblk[i + 1] = klotz0_1.buff[i];
    }
    return 0;
} /* zufallsv_ */
long int zufallrs_(FLOAT *svblk)
//FLOAT *svblk;
{
    long int i;

/*  restores common block klotz0, containing seeds and pointer */
/*  to position in seed block. IMPORTANT: svblk must be */
/*  dimensioned at least 608 in driver. The entire contents */
/*  of klotz0 must be restored. */

    klotz0_1.ptr = (long int) svblk[0];
/*pragma _CRI ivdep*/
    for (i = 0; i < 607; ++i) {
        klotz0_1.buff[i] = svblk[i + 1];
    }
    return 0;
} /* zufallrs_ */
void random_number(long int seed)
{
//    extern long int fischet_(), zufalli_(), normalt_(), zufallt_();

    zufalli_(seed);
    zufall_(NPTS,random_num);
}
```

A.2.7 output_data.cpp

PROBEPIC is capable of producing both graphical and text output. The graphical output is very useful for debugging purposes, while the text output provides the simulation results.

```
//#include "lprobe60.h"

void output_data(
            PARTICLEDEF particle1[],PARTICLEDEF particle2[],long  numpart1,long  numpart2,long count,
```

```
PLASMA &pl, GEOMETRY &ge,INJECTIONBEAM &ib,BOUNDARY &bb) {
  long  idx;
  FLOAT nde,ndi,tms;
  FLOAT avecurrent,avecurre,avecurri;
  char filename[200];
  FILE *currdata;

  sprintf(filename,"current_%+06.2fV.dat.csv", bb.V);
  currdata=fopen(filename,"a");

  avecurrent=bb.current/(OUTPUT_FREQUENCY*ib.dti);
  avecurre=bb.curre/(OUTPUT_FREQUENCY*ib.dti);
  avecurri=bb.curri/(OUTPUT_FREQUENCY*ib.dti);

  /*    COMPUTE NUMBER DENSITY     */

  nde=ndi=0.0;

  for (idx=0; idx < numpart1; idx++)
    if (particle1[idx].r <= ge.radius)
      nde += particle1[idx].m;
  nde /= ((pl.m_elec)*(pi*ge.radius*ge.radius*ge.length));

  for(idx=0;idx < numpart2;idx++)
    if(particle2[idx].r <= ge.radius)
      ndi += particle2[idx].m;
  ndi /= ((pl.m_ion)*(pi*ge.radius*ge.radius*ge.length));

  tms=count*ib.dti*1e6;

  if (count==1)
    fputs("   V, t[micro-sec],     ne,    ni,      nde,       ndi,       j,          je,          ji\n\n",currdata);

  fprintf(currdata,"%f, %f, %ld, %ld, %e, %e, %e, %e,
%le\n",bb.V,tms,numpart1,numpart2,nde,ndi,avecurrent,avecurre,avecurri);
  fclose(currdata);
  printf(" current: %f [mA]",avecurrent*1000.0);

  if(tms > 2.2723e-02 ) {
  //if (tms >=0.007625 ) {
    puts("\nexit for timing.  output_data.cpp");
    exit(0);
  }
  //t_count=0;

  bb.current=0.0;
  bb.curri=0.0;
  bb.curre=0.0;
}
```

Vita

Densu Aktas was born in 1987 in Istanbul. She finished her high school in 2001 and entered the university to study Electronic Engineering degree at Marmara University in Istanbul. She graduated from the university in 2005. During her studying engineering, she won several scholarships from honor organizations both in her country and abroad. Then she got her second degree in Electromechanical Engineering Technology at MTSU in 2007 to combine her electrical and mechanical knowledge for technical applications. Because of her professors' recommendations, she entered the graduate school at UT Space Institute in 2007 to pursue Masters Degree. She completed the degree requirements for both Aviation Systems and Mechanical Engineering. In 2010 she was accepted to PhD program to pursue her degree in Mechanical Engineering at UT Space Institute, specializing in electric propulsion. She performed her graduate research in plasma diagnostics and simulations.