Hybrid Electric Thruster Using

Gas Mixtures

Clyde Albert Jr. Barry Stoute

A DISSERTATION SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN EARTH AND SPACE SCIENCE AND ENGINEERING

YORK UNIVERSITY,

TORONTO, ONTARIO

December 2013

Abstract

Hybrid electric propulsion is a new technology that combines electrostatic and electromagnetic propulsions. The goal of this engineering study is to design a hybrid electric propulsion system that can provide a specific impulse that is higher than chemical thrusters, but uses less than 20 watts of power such that it can be used effectively in nanosatellites. This thesis describes RF electrothermal and electromagnetic thruster experiments that were undertaken to test the performance of gas mixtures, testing initially at lower temperatures a few hundred degrees above room temperature. A helium-nitrogen gas mixture is chosen in place of hydrazine because the mixture is non-toxic and does not require any heating equipment. In the study's electrothermal testing, the 50% nitrogen – 50% helium gas mixture is shown to have a specific impulse of 96.3 seconds. In the propulsion system's electromagnetic mode, 50% argon – 50% helium and 50% nitrogen – 50% helium gas mixtures have specific impulses of 603 and 801 seconds, respectively. In electrostatic propulsion mode, a 50% argon – 50% helium gas mixture has a specific impulse of 575 seconds, and 50% nitrogen – 50% helium gas mixture has a specific impulse of 805 seconds. The thesis describes subsequent experiments performed with the hybrid electric thruster operating in both electrostatic and electromagnetic propulsion modes over the course of two independent test cycles. The thesis provides a discussion on the expected performance of a space-based device utilizing the approaches studied in this investigation.

Acknowledgements

I would like to thank Dr. Brendan Quine for giving me the opportunity to conduct my research in the field of plasma propulsion under his expert guidance. I also would like to thank Stephen Brown for helping me throughout my plasma propulsion research, especially in the electronics that were used. I also would like to thank my doctoral committee members, Dr. James Laframboise and Dr. George Zhu, for guiding me throughout my doctorate program.

Finally, I would like to thank my family: Clyde, Jocelyn, Karen, and Shawn. They have supported me unfailingly through my doctoral degree and my academic career.

Table of Contents

List of Tables

List of Figures

Mathematical Symbols

The mathematical symbols are given in alphabetical order. The third column indicates the which section these symbols first appear. All units are in International System of Units.

Symbol	Definition	First Appearance
a	Acceleration	2.3
A_e	exhaust area	2.7.1
A_t	throat area	2.7.1
b	damping coefficient	3.2.3
B	magnetic field	2.5.5
$\overleftrightarrow{\mathscr{B}}$	Maxwell stress tensor	2.5.5
c_e	average thermal velocity of the electron	2.5.2
C_{cap}	capacitance	3.5.2
C_{tip}	tip constant	2.9
d_a	acceleration diameter	2.8.1
d_s	screen aperture diameter	2.8.1
D_0	dissociation energy (eV)	2.4
e^-	electron charge ($1.602 \cdot 10^{-19}$ C)	2.5.1
E	electric field	2.5.3
E_i	ionization energy (eV)	2.4

k_B	Boltzmann constant ($1.380 \cdot 10^{-23}$ J/K)	2.5.1
l_e	sheath length	2.8.1
l_{gap}	gap length between scr. and acc. electrode	2.8.1
L_{char}	characteristic length	3.7
L_{cone}	diffusion cone length	2.7.2
L_{ind}	inductance	3.5.2
m_b	mass of body	2.3
m_i	ion mass	2.5.2
m_p	mass of propellant	2.3
$\dot{m}_{ideal}$	ideal mass flow rate	2.7.3
M	total mass	2.3
Ma	Mach number	2.7.2
$\mathcal{M}$	Molecular Mass	2.7.2
n	number of particles per unit volume	2.4
n_e	number of electrons per unit volume	2.5.1
n_i	number of ions per unit volume	2.5.1
n_n	number of neutrals per unit volume	2.5.1
p	pressure	2.5.1
$\vec{p}$	momentum	F
p_a	ambient, or atmospheric, pressure	2.7.2
p_e	exhaust pressure	2.7.2

t_s	thickness of screen electrode	2.8.1
T_s	screen grid transparency	2.8.1
T	temperature	2.5.1
$T_e(K)$	electron temperature in kelvins	2.5.1
$T_e(eV)$	electron temperature in electron-volts	2.5.1
$\boldsymbol{v}$	velocity of particle or fluid	2.5.5
v_t	throat velocity	2.7.2
v_{ex}	exhaust velocity	2.7.2
$v_{ex,eff}$	effective exhaust velocity	2.7.2
$\boldsymbol{v}_i$	ion velocity	2.5.3
$\boldsymbol{v}_e$	electron velocity	2.5.3
$v_{PM}(M)$	Prandtl-Meyer expansion angle	2.7.2
$\mathbf{V}$	solution vector	2.6
$\cancel{V}$	Volume	2.11
V_-	negative voltage bias	2.10
V_+	positive voltage bias	2.10
V_b	beam or net voltage	2.8.1
V_{fl}	floating voltage	2.10
V_T	total voltage	2.8.1
V_{volt}	voltage	3.5.2
$\mathbf{X}_{conv}$	convective flux tensor	2.6

$\mathbf{X}_{\text{diss}}$	dissipative flux tensor	2.6
Z	ionization integer	2.5.2
α_{rise}	rise angle of cone	2.7.2
$\gamma = C_p/C_v$	heat capacity ratio, more commonly, ratio of specific heats of gas	2.5.5
Δv	delta V	2.3
Δx_{step}	spatial step in the x-direction	2.6
Δy_{step}	spatial step in the y-direction	2.6
ϵ_0	permittivity of free space $(8.854 \cdot 10^{-12} \text{ F} \cdot \text{m}^{-1})$	2.5.3
ϵ_p	dielectric constant of plasma	2.5.3
θ	angle (radians)	3.8
$\boldsymbol{\Theta}$	heat flux	2.5.5
κ	thermal conductivity	2.5.5
λ_D	Debye length	2.5.2
λ_{mfp}	Mean Free Path	3.7
Λ	plasma parameter	2.5.2
μ	viscosity	2.5.5
μ_0	permeability of free space $(1.256 \cdot 10^{-6} \text{ H/m})$	2.5.5
ρ_q	charge density	2.5.5
σ_V	Von Mises stress	3.2.2

$\sigma_{X,Y,Z}$	normal stresses on x, y, z direction	3.2.2
$\sigma_{cross\ section-ei}$	cross-section of electron-ion scattering	2.5.2
$\sigma_{p,DC}$	DC plasma conductivity	2.5.3
σ_p	plasma conductivity	2.5.3
ρ	density	2.5.5
$\dot{\rho}_e$	electron production rate	2.5.5
Ψ	rate of energy loss due to inelastic collisions	2.5.5
ω	input frequency	2.5.3
ω_n	natural frequency	3.2.3
CFD	Computational Fluid Dynamics	4.1
EM	Electromagnetic	1
ES	Electrostatic	1
ET	Electrothermal	1
JPL	Jet Propulsion Lab (NASA)	2.1
NASA	National Aeronautics and Space Agency	2.1

1 Introduction

Electric propulsion is a technology that uses plasma as the propellant. It typically has a higher specific impulse, or fuel efficiency, than conventional chemical rockets. Work on electric propulsion was initiated in 1906 by R. Goddard [Choueiri, 2004]. Significant experimental research on electric propulsion was not started until the mid-1960s. Electric propulsion currently has the potential to revolutionize space utilization. It can, for example, shorten interplanetary travel times between Earth and Mars, as with the VASIMR engine [Chang-Díaz *et al*, 1999; Arefiev and Breizman, 2004; Glover, 2011] and the NSTAR's ion engine [National Aeronautics and Space Administration, 1999; Polk *et al,* 2000].

Electric propulsion, or EP, is an effective choice for deep space missions and long term orbital station-keeping. The type of electric propulsion system required is dependent on its primary function. An electrothermal (ET) thruster is a type of electric thruster that uses a conventional rocket nozzle with an inert (or reactive) fluid as the propellant [Jahn, 1968]. This type of thruster has the lowest specific impulse of the three main EP types but yields a significant amount of thrust [Sutton & Biblarz, 2010]. An electrostatic (ES) thruster uses electrostatic grids to accelerate ions. The resulting propulsive force is comparatively small, typically millinewtons [Goebel & Katz, 2008]. This small force is enough to propel many spacecraft operating in low gravitational environments, making it ideal for deep-space missions because of its high specific impulse and efficiency [Martinez-Sanchez, 2004a]. An electromagnetic (EM) thruster is a thruster that the plasma (a hot, electrically neutral mixture of electrons, positive ions and neutral atoms or

molecules) is propelled from the Lorentz force (refer to section 2.5.5 for more details) ; this enhances specific impulse while yielding a relatively high level of thrust [Martinez-Sanchez, 2004a; Haag *et al.*, 2007]. These definitions, along with other definitions, are explained in the Glossary.

The doctoral research undertaken here investigates a type of thruster that is potentially versatile in both fuel efficiency and thrust, and capable of being operated in augmented-thrust mode (e.g. electromagnetic). The engine should fit within a nanosatellite and use less than 20 watts of power. The engine should be light-weight, less than two kilograms dry mass, and should have a higher specific impulse than conventional chemical rockets. The propellant it uses must not be cumbersome in storage, or toxic in use [Quine, 2010]. In large spacecraft, there are onboard chemical and ion thrusters; therefore, there is no need for an EM thruster. However, in a nanosatellite, the EM thruster is necessary since the chemical thruster would take up too much power and resources.

This research also investigates the use of gas mixtures in electric thrusters. The conventional rocket monopropellant hydrazine (N_2H_4) appears to be an excellent candidate for electrothermal propulsion systems. However, it is toxic to humans, and its use requires special equipment for safe storage and special conditions. Thus, it is impractical for some laboratory experimentations [Anonymous, Chronic Toxic Summary: Hydrazine, 2000; Sutton & Biblarz, 2010]. Instead, various helium-nitrogen gas mixture compositions are investigated as an alternative to hydrazine. Researchers often use inert helium in electrothermal thrusters because of its high specific impulse resulting from its

lower molecular mass. The major problem with helium is its viscosity. High fluid flow viscosity causes a loss in the resulting performance of the thruster [Bar-Meir, 2011]. Nitrogen, by contrast, has relatively low viscosity, which yields a higher efficiency in its engine performance [Bar-Meir, 2011]. A helium-nitrogen gas mixture is investigated in this study with the use of an aluminum-based convergent-divergent (C-D) exhaust nozzle for the ET unit. An electrothermal thruster built for this study is shown in Figure 1.1. The use of such a gas mixture is studied, with the expectation that it may yield a better efficiency in the thruster than using helium or nitrogen alone.

Figure 1.1: ET Thruster manufactured for this research

A helium-argon gas mixture is also investigated, in part because argon has one of the lowest ionization energy in the group of gases tested. The electrothermal thruster is capable of employing either gas mixture. As per figure 1.2, a glass exhaust nozzle was also used, and applied for running in ES and EM modes of propulsion.
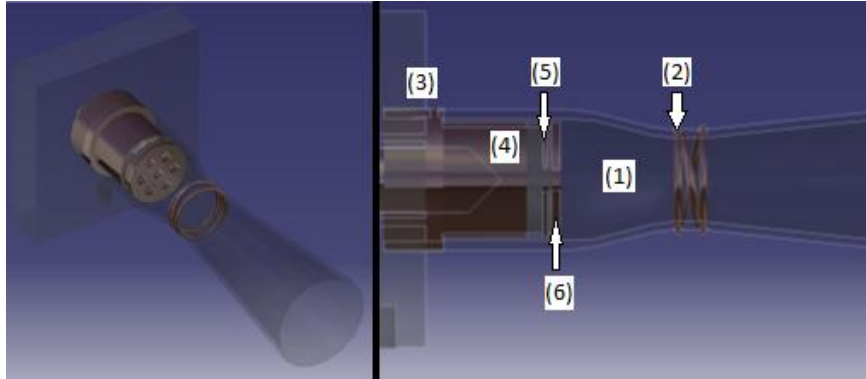
Figure 1.2a: Hybrid Electric Thruster Model of prototype built for this study (1) Glass C-D Nozzle (2) Reheating Element (3) Gas Inlet (4) Capacitive Electric Discharge for Plasma Production (5) Screen Electrode (6) Acceleration Electrode

Figure 1.2b: Hybrid Electric Thruster Ignition Test

The hybrid electric thruster developed for this study combines elements of the ES and EM thrusters and use a conventional glass nozzle to accelerate the plasma. A conventional glass C-D nozzle outfitted with a solenoid at the throat is used to reheat the gas and hence for ES and EM operation to maximize the heating of the gas (towards

producing a high-temperature plasma). A major challenge with plasma production is the high heat loss from the radio-frequency (RF) supply dissipating wasted energy. Therefore, connecting a solenoid from the RF supply to the nozzle is evaluated as a means to provide a higher performance for the thruster. The second component to be evaluated is a capacitive discharge chamber. This component acts both as a plasma production chamber for the electrostatic thruster and as an electromagnetic thruster. An electrostatic thruster component incorporates a screen and an acceleration electrodes for ES propulsion, and is designed for any low-thrust manoeuvres that might be required. The electromagnetic thruster component is designed for high-thrust manoeuvres, such as orbital station-keeping. This thruster is unique in its mass class because of its dual modes. Results for the hybrid electric thruster are presented for operations as one or more of three possible thruster types, illustrating a versatile new approach to electric propulsion design.

1.1 Thesis Organization

The thesis is organized into six chapters. The introductory chapter presents research objectives. The background chapter provides a brief overview of the history of electric propulsion and a literature review of the various types of electric propulsion. The background of plasma physics theory and electric propulsion engineering are also presented. Plasma physics theory covers atomic physics, plasma properties, collision theory, plasma in an electric field, and plasma generation. This chapter also describes the continuous fluid and plasma dynamics models needed in order to understand the flow of gas and plasma in the thruster. It concludes with a discussion of the function and the engineering of all three main types of electric propulsion systems.

The methodology chapter describes the design of the electrothermal thruster and hybrid electric thrusters, along with the procedures utilized in the simulation and experimental research undertaken here. This chapter also discusses the design criteria for this research. Solid mechanics models of the thrusters, both in the static and dynamic cases, are presented. Schematics of the electronics and the electromagnetic radiation encapsulation are presented. Simulations using the MacCormack method to demonstrate the fluid flow in the electrothermal thruster and the hybrid electric thruster are outlined. Experiments conducted using a thermal vacuum chamber in both medium and high vacuum conditions are described, along with the experiments using of a free-balance pendulum and a Langmuir probe. Results and conclusions are discussed thereafter in separate chapters.

2 Background

This chapter is divided into two sections: a discussion of the history and progress of electric propulsion research, and a summary of the theory used in the research undertaken here.

2.1 History

2.1.1 The Early Years of Electric Propulsion

The original concept of electric propulsion was proposed by Dr. Robert Goddard in 1906 [Choueiri, 2004]. He theorized that electrons could be used to propel a spacecraft. The primary challenge to his idea was that the mass of the electrons are too small to provide any substantial thrust. The use of electrons would also charge the spacecraft. When he published his article in September 1906, he failed to calculate the amount of energy required for the electrons to attain a velocity near the speed of light. Although Goddard never considered using ions (much heavier than electrons) as the main propellant of electrostatic propulsion, this original theory initiated research into electric propulsion. After Goddard's study of electric propulsion, numerous scientists, such as Oberth and Glushko, started researching the study of electric propulsion [Choueiri, 2004].

In 1918, Hermann Julius Oberth theorized the potential for electric engines to propel large spacecraft. He also envisioned the development of manned spaceflight. Oberth proposed a two-stage rocket for spaceflight that used an alcohol-oxygen liquid-propellant engine in its first stage and hydrogen-oxygen liquid-propellant engine in its second stage. This was discussed in his book *Wege zur Raumschiffahrt* (Ways to

Spaceflight), published in 1929. A chapter on electric propulsion, titled *Das elektrische Raumschiff* (The Electric Spaceship), explained the fuel savings of an electric engine used for deep space missions [Choueiri, 2004].

In 1929, Valentin Glushko tested the first electrothermal engine at the Laboratory Gas Dynamics located in Leningrad, USSR. Electrothermal propulsion creates thrust using a high-temperature gas (typically liquid in storage) as a propellant. The test was conducted on a standard test stand where the thrust or propulsive force was measured [Stuhlinger, 1964; Choueiri, 2004].

2.1.2 The Space Race

In 1958, the first ion engine model was demonstrated at Rocketdyne. In early 1959, another model was demonstrated at Electro-Optical-Systems. In 1960, the first full size electrostatic ion engine was tested at the National Aeronautics and Space Administration (NASA) Lewis (now Glenn) Research Center [Stuhlinger, 1964; Anonymous, NASA, 1999]. In early 1964, the first sub-orbital flight using an ion engine was performed by the United States. Later that same year, the first electric engine used in an interplanetary probe was developed by the Soviets [Stuhlinger, 1964; Anonymous, NASA, 1999].

During the mid-1960s, NASA Glenn Research Center developed the magnetoplasmadynamic (MPD) propulsion system. The MPD research was intermittent because of limited access to high-power electric supplies [Lapointe and Mikellides, 2001; Anonymous, NASA, 2004].

2.1.3 Deep Space 1

In October 1998, the first ion thruster designed for deep space missions came into operation on Deep Space 1, a spacecraft developed by Jet Propulsion Laboratory (JPL). The mission was to perform a flyby around the asteroid 9969 Braille. JPL and NASA worked on NSTAR (NASA Solar Electric Power Technology Application Readiness program) to develop the ion engine for Deep Space 1. The spacecraft completed its mission flying by a near-Earth asteroid in July 1999 and comet Borrelly in September 2001, reaching a record-breaking maximum change of speed at 4000 m/s without the use of gravitational assistance [Anonymous, NASA, 1999]. In June 2010, NASA's Dawn spacecraft broke Deep Space 1's speed record, reaching its maximum change of speed at 4300 m/s [Agle, 2010].

Figure 2.1: Deep Space 1

[Anonymous, NASA, 1999]

2.2 Progress of Electric Propulsion Research

Electric propulsion technology may be divided into three main types: electrothermal, electrostatic, and electromagnetic. Electrothermal propulsion uses fluid

(liquid or gas) as a heated propellant in a conventional nozzle [Sutton & Biblarz, 2010].

Electrostatic propulsion uses Coulomb-force electric fields to accelerate positive ions

[Serway & Jewett, 2004; Goebel & Katz, 2008]. Electromagnetic propulsion uses Lorentz

forces to propel the plasma [Jahn, 1968; Serway & Jewett, 2004]. Overall, electric

propulsion offers a promising method to improve the performance of space propulsion

systems, increase the fuel efficiency of these thrusters, and lower overall the cost of

spacecraft. Table 2.1 displays the performance of the different propulsion technologies, in

terms of specific impulses. Specific impulse is, in the vacuum of space, the change in

momentum per unit mass. A higher specific impulse indicates higher performance, as

explained in Section 2.3.

Table 2.1: Types of Electric Engines on a Standard Spacecraft Ordered by
Specific Impulse [Sutton & Biblarz, 2010]

Propulsion Type	Acceleration Type	Propellant	Specific Impulse (s)	Thrust (mN)
Monopropellant Rocket	Chemical	N_2H_4	200-250	$30\text{-}10^5$
Resistojet	Electrothermal	NH_3, N_2H_4, H_2	200-350	200-300
Arcjet	Electrothermal	NH_3, N_2H_4, H_2, N_2	400-1000	200-1000
Pulsed Plasma Thruster	Electromagnetic	Teflon	600-2000	0.05-10
Hall Thruster	Electrostatic	Xe, Ar	1500-2000	0.01-2000
Ion Thruster	Electrostatic	Xe, Kr, Ar, Bi	1500-8000	0.01-500
Magnetoplasmadynamic	Electromagnetic	Ar, Xe, H_2, Li	2000-5000	$10^{-3}\text{-}2000$

2.2.1 Electrothermal Propulsion

A high-temperature gas may be created in electrothermal engines using either a

direct current (DC) or an alternating current (AC) power supply. DC-based electrothermal

thrusters use a battery-like power source to heat the propulsion gas. An example of a DC-based thruster is the resistojet, which uses a resistance element (material which heats up readily when an electric current is passing through it) to heat up the gas. The active resistance element (fabricated from a high-temperature material, like tungsten) heats up the gas as it passes through; the gas maybe ionized from the heat [Jahn, 1968; Martinez-Sanchez, Space Propulsion, 2004; Sutton & Biblarz, 2010].
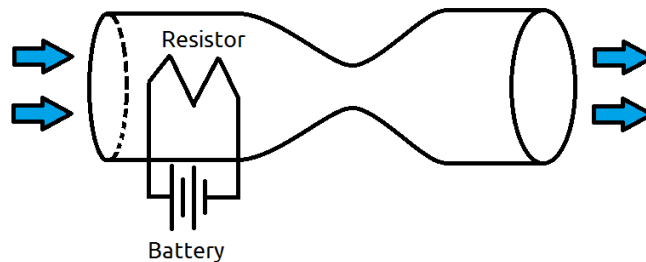
Figure 2.2: Resistojet Electrothermal Thruster

However, resistojets have a low efficiency because heating the gas for ionization requires a large power source [Jahn, 1968; Sutton & Biblarz, 2010]. The efficiency of a resistojet is further limited when the gas passes through the resistor and cools the resistor [Jahn, 1968].

An arcjet thruster is a DC-based electrothermal thruster that uses two electrodes and a high potential difference, typically 400 V at the minimum, to ionize the gas. The ionization is caused by an electrical arc created between the electrodes [Jahn, 1968; Martinez-Sanchez, 2004].
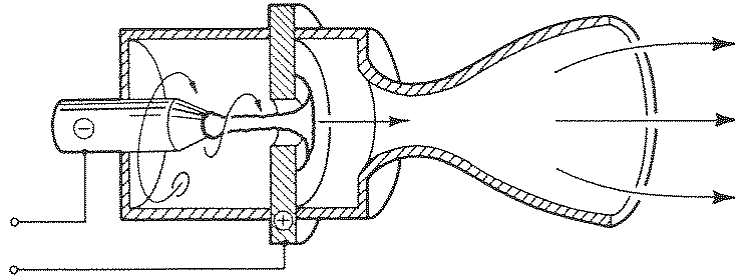
Figure 2.3: Arcjet Electrothermal Thruster. The lines represent the fluid path.

[Figure 6.19 in Jahn, 1968]

There is a considerable amount of research into arcjet thrusters. One of the reasons this technology is popular is its simplicity. The technology requires only a cathode tip, an anode nozzle, and a high-voltage power source [Jahn, 1968].

As an alternative to DC-based arcjet propulsion, AC-based electrothermal thrusters can use alternating currents to heat and potentially ionize the propellant gas. The amount of power required to sufficiently heat the gas depends inversely on the input frequency [Lieberman, 2003; Dinklage, 2005]. AC-based electrothermal thrusters are a better alternative to DC-based electrothermal thrusters because of their lower power requirement [Sutton & Biblarz, 2010].

AC electrothermal thrusters use an alternating current frequency with a range typically between 10 kHz and 10 GHz. The most common frequency used for producing plasma in the RF region is 13.56 MHz. AC electrothermal thrusters require shielding to carefully isolate the RF power plant from the rest of the system and controllers [Hrbud *et*

al., 2007; Takao *et al.*, 2007; Batishchev, 2009; Stein *et al.,* 2008; Takahashi *et al.*, 2009; Takahashi *et al.*, 2011].

The principal operation of the AC electrothermal thruster is similar to the standard electrothermal thruster. Figure 2.4 illustrates the capacitive plasma discharge.
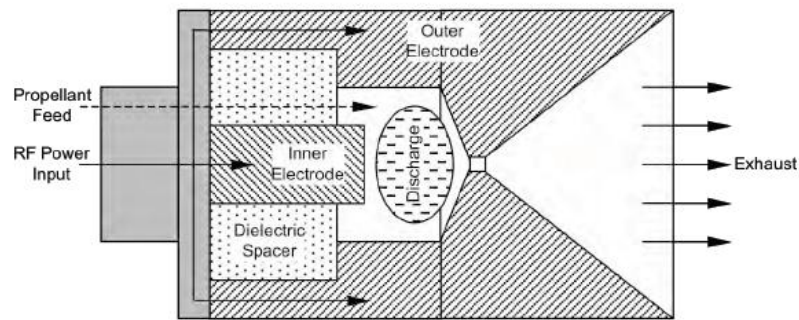
Figure 2.4: RF Electrothermal Thruster Schematic [Hrbud et al., 2009]

Purdue University has developed a miniature RF electrothermal thruster. Their work has shown that the temperature of the plasma increases as the input of either the frequency or power increase [Hrbud *et al.*, 2007]. At Kyoto University, a team has simulated and developed a microplasma thruster using a microwave antenna. They conclude that the performance increases as the mass flow rate increases [Takahashi *et al.*, 2009; Takahashi *et al.*, 2011].

Inductive plasma discharge is a method in which a solenoid is attached to a RF generator to produce plasma. The advantage of this method is that its lifespan of the device is longer than that of the capacitive discharge method [Jahn, 1968].
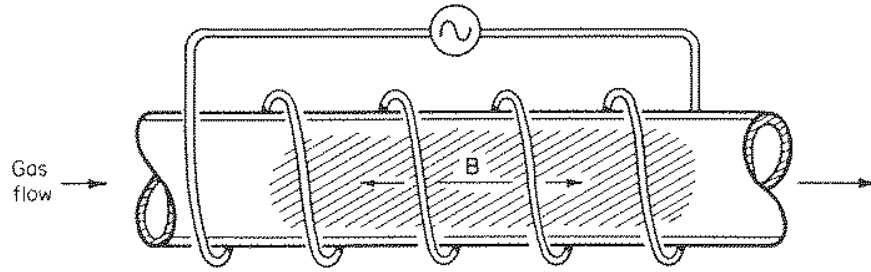
Figure 2.5: Inductive Plasma Discharge in a channel [Figure 6.31b in Jahn, 1968]

This solenoid configuration, shown in Figure 2.5, is commonly used for RF discharge. An example of such research, an inductively coupled plasma source was developed at Boise State University [Browning *et al.*, 2011].

Another type of antenna which is commonly used in AC heating is the helicon antenna. Helicon antenna is a specialized antenna shaped such that there is a high axial magnetic field; refer to *Mulchany et al.* for further information [Mulchany *et al.*, 2009].
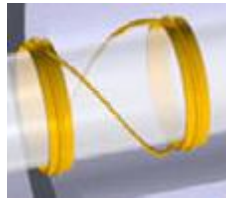
Figure 2.6: Example of a Helicon Antenna [Mulchany et al., 2009]

This configuration assists in the prevention of energy loss inside the plasma; thus, increasing ionization efficiency over its inductive counterpart. This method is very difficult to perfect, and design imperfections can cause the ionization efficiency to drop significantly. Douglas Palmer and Mitchell Walker of Georgia Institute of Technology

have conducted a performance analysis on the helicon thruster. Their research attempted to determine an optimal frequency for maximum performance, and they found that the lowest input frequency in their experiment produced the best results [Palmer *et al.*, 2008].

2.2.2 Electrostatic Propulsion

Electrostatic propulsion uses electric (electrostatic) fields to accelerate ions. There are two types of electrostatic propulsion systems: Hall and ion. According to *Goebel & Katz*, a Hall thruster is a

> […] type of electrostatic thruster utilizes a cross-field discharge described by the Hall effect to generate the plasma. An electric field established perpendicular to an applied magnetic field electrostatically accelerates ions to high exhaust velocities, while the transverse magnetic field inhibits electron motion that would tend to short out the electric field. Hall thruster efficiency and specific impulse is somewhat less than that achievable in ion thrusters, but the thrust at a given power is higher and the device is much simpler and requires fewer power supplies to operate [Goebel & Katz, 2008].

The PPS-1350 Hall thruster was developed by SNECMA for the SMART-1 spacecraft, which launched on November 13, 2004.
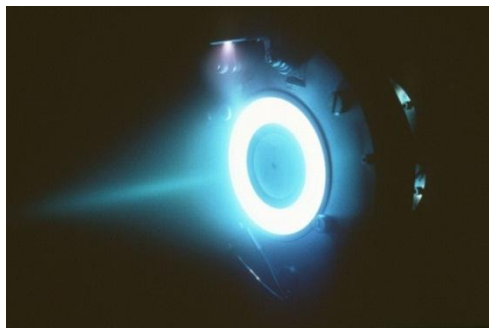
Figure 2.7: PPS-1350 Hall Thruster [Snecma, 2012]

The thruster provides a thrust of 68 mN with a specific impulse of 1640 s using 1.5 kW of power [Foing, 1999; Foing, 2003].

According to *Goebel & Katz*, an ion thruster uses two electrostatically charged grids with a large potential difference to accelerate the ions. The plasma generator in these thrusters can be either a DC or an AC power plant [Goebel & Katz, 2008]. A DC generator was used initially for ion propulsion during the early development between the 1960s and the late 1990s [Stuhlinger, 1964; Dryden 1967; NASA, 1999]. The DC generator consists of a hollow cathode that ejects electrons into a chamber. These electrons collide with the gas particles causing ionization. Around the plasma chamber, magnets are mounted to further ionize the gas particles by electron cyclotron resonance. This causes the electrons to spiral around the magnetic fields, increasing the chance of ionization [Goebel & Katz, 2008]. Over time, AC power sources have become more popular due to their higher longevity and lower power requirement.

Ion propulsion depends on the charge potential difference between the electrically charged grids. The performance of the electrostatic thruster depends on the mechanical and electrical configurations of the grids [Goebel & Katz, 2008]. This propulsion system type has been extensively researched with the potential for miniaturization without sacrificing performance [Wirz *et al.*, 2006; Yamamoto *et al.*, 2006; Nakayama *et al.*, 2007; Mueller *et al.*, 2008; Lubey *et al.*, 2011].

Researchers from NASA [Mueller *et al.,* 2008], California Institute of Technology [Wirz *et al.* 2006], and Pennsylvania State University [Lubey *et al.*, 2011] have worked

on how to maximize the performance of the ion thruster without the use of a large power source. NASA recently conducted a survey on the current state of miniaturized propulsion technology under research and development [Mueller *et al.,* 2008].

Figure 2.8a: Miniature Xenon Ion Thruster Relative Size to a Penny [Mueller et al., 2008]
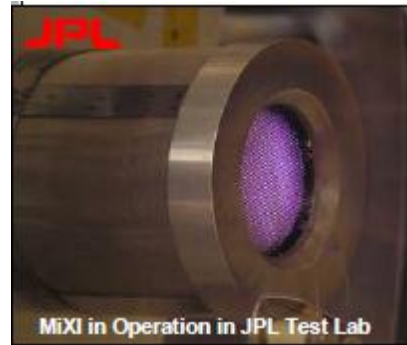
Figure 2.8b: MiXI in Operation at NASA's Jet Propulsion Laboratory Test Facility [Mueller et al., 2008]

At the Jet Propulsion Laboratory, researchers have developed a miniature xenon ion thruster (MiXI) which has a specific impulse between 2500 and 3000 seconds. The power required for the MiXI is between 13 W and 50 W [Mueller *et al.,* 2008]. A major challenge with the MiXI thruster is the cost of the propellant [Mueller *et al.*, 2008]. Xenon is an expensive gas, priced at approximately $120 per 100g, or $34,500 per cylinder tank [Brown, 2011; Chemicool.com, 2012].

Researchers from Pennsylvania State University have designed and built a microwave-powered ion thruster which consumes only 1 W of power, and uses argon and

xenon as its propellants [Lubey *et al.*, 2011]. The specific impulse of the ion thruster is approximately 5500 seconds [Lubey *et al.*, 2011].
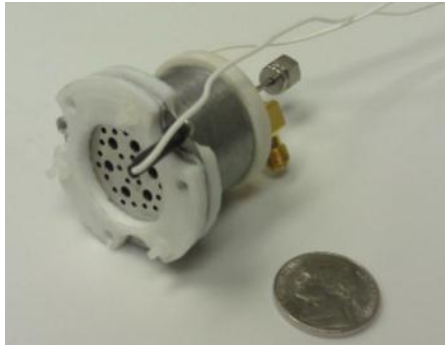
Figure 2.9a: Relative Size of This Miniature Microwave Ion Thruster [Lubey et al., 2011]
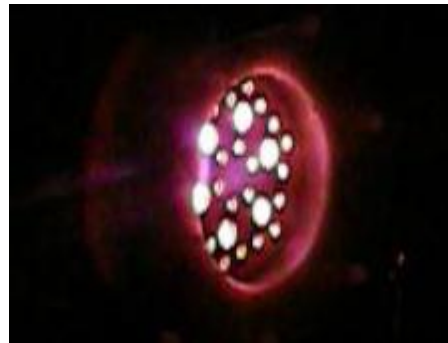
Figure 2.9b: Miniature Microwave Ion Thruster in Operation [Lubey et al., 2011]

They used both a Langmuir probe and a Faraday cup to measure the performance of the thruster. The Langmuir probe measures the electron temperature of a gas. It is a useful tool in electrostatic ion propulsion research for estimating thrust. A Faraday cup is a device which measures the number of charged particles in a plasma. The researchers have concluded by using both the Langmuir probe and the Faraday cup, the exhaust velocity was approximately 54 km·s^{-1}, indicating the performance of the thruster to be high at a specific impulse of 5500 seconds [Lubey *et al.*, 2011].

2.2.3 Electromagnetic Propulsion

There are three major electromagnetic thrusters which are being investigated at the moment: pulsed-plasma thruster, magnetoplasmadynamic thruster, and the variable specific impulse magnetoplasma rocket (VASIMR). Despite the difficulties in the

associated research, electromagnetic propulsion is theorized to be the most promising electric propulsion technology available [Anonymous, NASA, 2004]. It operates using predominately the Lorentz (EM) force for propulsion thrust [Jahn, 1968; Martinez-Sanchez, 2004a; Anonymous, NASA, 2004; Sutton & Biblarz, 2010].

A pulsed plasma thruster (PPT) is an electromagnetic thruster which typically uses capacitive discharge and a propellant, such as polytetrafluoroethyelene (PTFE) or inert gases. A high voltage source, such as a spark plug, ionizes the surface of the propellant. The capacitors then discharge an electric current between the electrodes, with the transient EM field then accelerating the resulting plasma. A PPT is typically a short-burst thruster with a thrust efficiency (see section 3.9 for further information) of less than 10% [Cassady *et al.*, 2000]. Despite its low efficiency, the specific impulse of the engine is around 3000 seconds, which is much higher than that of chemical rockets. Consequently, PPTs are designed for use during space manoeuvres rather than to provide the continuous thrust required for deep space applications [Jahn, 1968; Choueiri & Ziemer, 2001; Nawaz *et al.,* 2005; Cassibry, 2008]. The PPT is widely developed industrially, in part because of the simplicity of the relative technology.
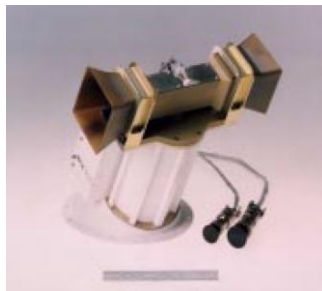
Figure 2.10: Pulsed Plasma Thruster [Cassady et al., 2000]

The magnetoplasmadynamic (MPD) thruster is another type of electromagnetic thruster. A MPD thruster produces and propels plasma with the use of a large direct current electric supply. The main difference between the MPD and the PPT is that the MPD power supply runs continuously. This method requires a large power source, in the order of megawatts, to have a sustainable propulsion at a relatively high thrust delivery (for an EP system). An alteriative implementation, similar to the PPT method, modifies the MPD thruster to a quasi-steady state propulsion by fast-switching capacitors [Guarducci *et al.*, 2011]. Quasi-steady state is a state which the thruster operates in pulses. The period of the pulses are typically 1 μs. One siginficant issue with MPD propulsion is the degradation of the materials of the surrounding structure because of the high input power and ionization levels. Superalloys and other specialized materials that can withstand the high input power may find an application in MPD technology [Jahn, 1968; Herdich *et al.*, 2006; Guarducci *et al.*, 2011].
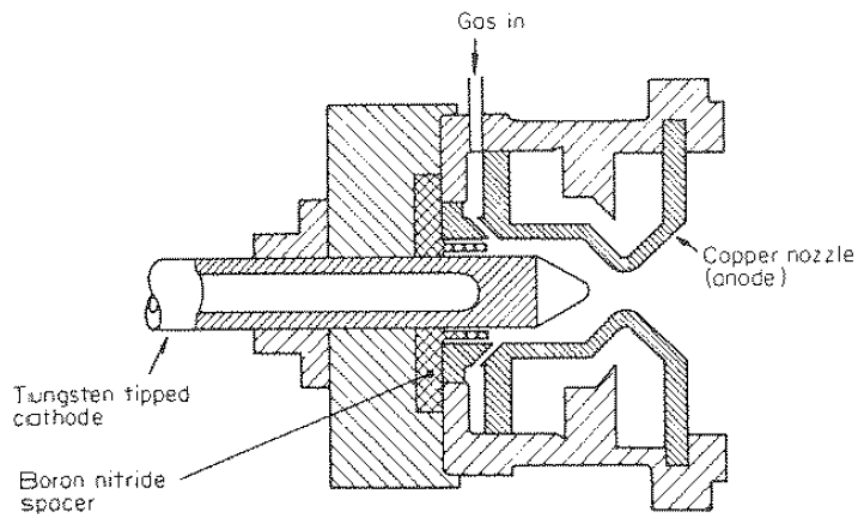
Figure 2.11: Magnetoplasmadynamic Thruster [Figure 8.15a in Jahn, 1968]

Some research on electromagnetic propulsion is focused on the use of a solenoid as an external magnet, to actively produce the magnetic field [Domonkos *et al.*, 1995].

Additionally, an external magnetic field, such as a solenoid, in an arcjet thruster may be applied [Miyasaka *et al.*, 2001]. With a magnetic field present, the arcjet becomes an applied-field electromagnetic thruster. Depending on the force produced by the electromagnetic (EM) effects, the EM force may be substantially larger than the electrothermal force, resulting in a de facto magnetoplasmadynamic thruster [Jahn, 1968; Miyasaka *et al.*, 2001; Hoyt, 2005; Kagaya & Tahara, 2005].

Former astronaut and founder of the Ad Astra Rocket Company in the United States, Franklin Chang-Diaz, developed the VASIMR. It uses radio frequencies to ionize the gas and increase the resulting plasma temperature to over a million kelvins. At the exhaust, the magnetic nozzle EM field accelerates the plasma. The thrust of the rocket is a mixture of electrothermal and electromagnetic effects [Chang-Díaz *et al.*, 1999; Arefiev & Breizman, 2004; Ahedo & Merino, 2011; Glover, 2011; Squire *et al.,* 2011].
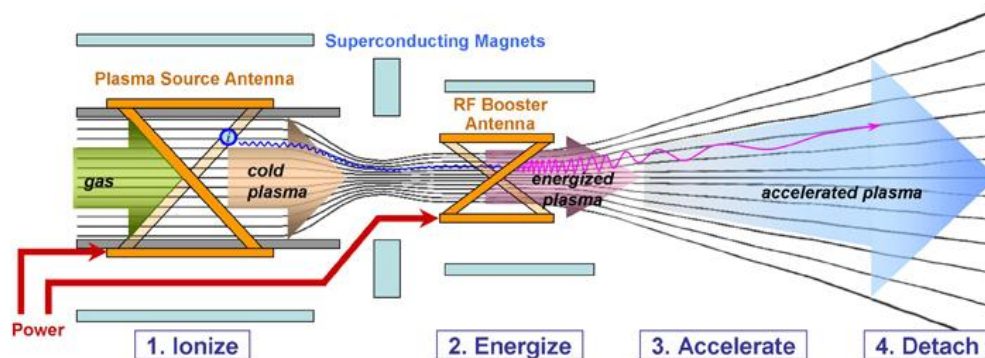
Figure 2.12: Operation of the VASIMR [Chang-Díaz et al., 1999]

21

This propulsion system is theorized to transport a spacecraft from Earth to Mars in 39 days. On September 30, 2009, the VASIMR prototype VX-200 became the most powerful electric engine to be demonstrated in operation to date. The thrust of the engine is approximately 0.5 N using a 200 kW power supply. The company is currently improving the performance of the engine [Chang-Díaz *et al*., 1999].
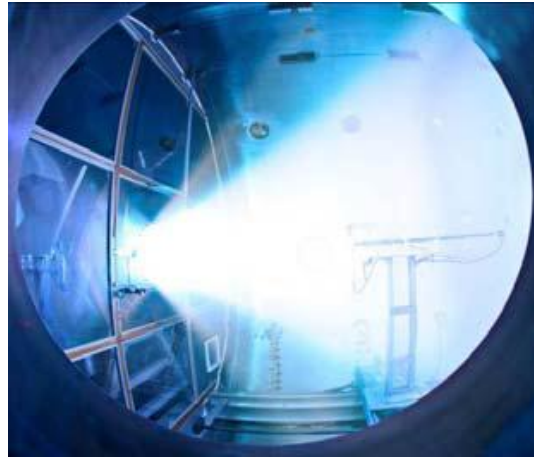
Figure 2.13: VX-200 in operation [Chang-Díaz et al., 1999]

2.3 Propulsion Basics

A primer on propulsion basics is necessary for understanding how electric thrusters work. The mass ejected from the thruster has an equal and opposite force effect to the spacecraft, according to Newton's 3rd law. The thrust, F_{thrust}, is described as the product of the exhaust velocity and the mass flow rate, [Goebel & Katz, 2008; Sutton & Biblarz, 2010],

$$F_{thrust} = -v_{ex}\frac{dm_p}{dt},$$

(2.1)

where v_{ex} is the exhaust velocity, and m_p is the mass of the propellant. The total mass of the spacecraft, M, is the sum of the propellant and the body, m_b, masses [Goebel & Katz, 2008; Sutton & Biblarz, 2010],

$$M = m_p + m_b. \tag{2.2}$$

When the propellant mass is depleted, the total mass of the spacecraft is reduced to its body mass with the associated change in velocity. The change in velocity, Δv, of the spacecraft is the product of the exhaust velocity of the propellant and the logarithmic full-depleted mass ratio [Goebel & Katz, 2008; Sutton & Biblarz, 2010],

$$\Delta v = v_{ex} \ln \left(\frac{m_b + m_p}{m_b} \right). \tag{2.3}$$

After the expenditure of the propellant, the depleted mass of the spacecraft is [Goebel & Katz, 2008; Sutton & Biblarz, 2010]

$$m_b = M \cdot \exp \left(-\frac{\Delta v}{v_{ex}} \right). \tag{2.4}$$

Specific impulse describes the fuel efficiency of a jet, rocket, or space engine. In space, it is the exhaust velocity of the propellant divided by the Earth's gravitational acceleration at sea level; or, more generally in the atmospheric or space flight, the amount of thrust divided by the reference weight flow rate of the propellant [Goebel & Katz, 2008; Sutton & Biblarz, 2010],

$$I_{sp} = \frac{F_{thrust}}{\dot{m} g_0} = \frac{v_{ex}}{g_0}. \tag{2.5}$$

Specific impulse indicates how efficiently the engine uses its fuel for propulsion. The higher the specific impulse (in units of seconds), the more fuel efficient the thruster is. A higher mass flow rate ($\dot{m} = \dot{m}_p$) or exhaust velocity increases the thrust [Goebel & Katz, 2008; Sutton & Biblarz, 2010].

If an engine is going to lift off from the Earth's surface to low-Earth orbit (LEO), the energy per kilogram required to overcome gravitational forces is [Quine, Seth, & Zhu, 2009]

$$E = g(h_{alt} - h) + \frac{1}{2}(v_{orb} - v_{ex})^2, \tag{2.6}$$

where g is the effective (mean) gravity, h_{alt} is the altitude of the orbit, h is the current position of the spacecraft relative to the surface, and v_{orb} is the orbital velocity [Quine, Seth, & Zhu, 2009]. Neglecting atmospheric effects, the thrust of the engine has to overcome the Earth's gravitational field,

$$F_{thrust} = M \cdot a = \dot{m}_p v_{ex} - Mg. \tag{2.7}$$

In order for the spacecraft to hover, the product of the specific impulse from the thruster and the mass flow rate of the propellant must equal to the total weight of the spacecraft,

$$M = I_{sp} \dot{m}_p. \tag{2.8}$$

The kinetic power, P_{ke}, is the rate of change of the kinetic energy from the propellant,

$$P_{ke} = \frac{1}{2} \dot{m}_p v_{ex}^2. \tag{2.9}$$

The kinetic power determines the thrust efficiency at a given input power.

For example, if a magnetoplasmadynamic thruster with a specific impulse of 3000 seconds is going to hover a spacecraft with a mass of 10 kg, the required mass flow rate is 3.3 g·s^{-1}. However, the energy required to hover the 10 kg spacecraft is approximately 1.4 MW. Currently, this is only attainable if the spacecraft has a large power plant or by use of conventional chemical rockets.

2.4 Ionization and Dissociation

Ionization energy, E_i, is the amount of energy required to remove an electron from its parent atom [Jahn, 1968]:

$$A + E_i \leftrightarrow A^+ + e^-. \tag{2.10}$$

Noble gases typically have higher ionization energies relative to those of other elements in the periodic table. For example, helium has the highest ionization energy at 24.6 eV, whereas monatomic hydrogen has an ionization energy at 13.6 eV. Bond dissociation is the process of the bond breaking apart in a molecule. The energy required to dissociate the bonds, D_0, varies among different molecules. Table 2.2 shows the ionization energies and dissociation energies of various gases [Jahn, 1968; Bose, 2004; Haynes 2012a; Haynes, 2012b].

Table 2.2: Gases and their Ionization Energy and Dissociation Energy [Haynes, 2012]

Element	Ionization Energy (eV/particle)	Bond Dissociation Energy (eV/particle)
Hydrogen	13.598	4.519
Helium	24.587	N/A
Nitrogen	14.536	9.764
Oxygen	13.618	5.121
Neon	21.567	N/A
Argon	15.762	N/A
Xenon	12.132	N/A

Ideally, xenon would be a good choice among the noble gases for plasma generation because of its relatively low ionization energy [Goebel & Katz, 2008].

Diatomic nitrogen has a dissociation energy that is lower than its ionization energy. Therefore, the molecule is typically dissociated first before being ionized [Jahn, 1968],

$$N_2 + D_0 \rightarrow 2N \tag{2.11}$$
$$2N + E_i \rightarrow 2N^+ - 2e^- \, .$$

Hence, the total energy needed to ionize nitrogen is 24.3 eV/particle, which is comparable to that of helium. Ionization can also occur from the diatomic phase of nitrogen. The ionization energy without dissociation for nitrogen is 15.52 eV [Haynes, 2012].
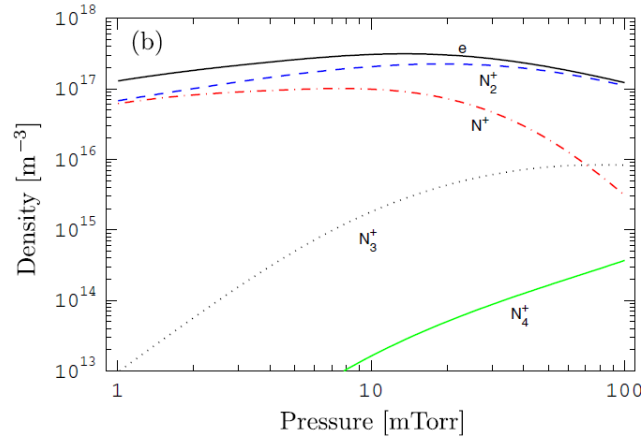
Figure 2.14: Density of each nitrogen atom and ion with respect to

the pressure [Thorsteinsson & Gudmundsson, 2009]

Figure 2.14 illustrates the global model of the density of each nitrogen species with respect to the pressure of the nitrogen. The significance of this diagram is to show what species of nitrogen exist in the plasma at various pressures [Thorsteinsson & Gudmundsson, 2009].

2.5 Plasma Physics

2.5.1 Plasma Properties

The fundamentals of plasma physics are necessary to design and build an electric thruster. These are developed from the ideal gas law, electricity and magnetism, and statistical chemistry. Given the values of the power, voltage input, input frequency, type of gas, pressure, mass flow rate, and the geometry of the generator, a detailed analysis of the plasma may be developed [Howard, 2002, Lieberman, 2003, Bellan, 2004; Dinklage *et al*., 2005, Goebel & Katz, 2008].

Let *n* be the particle density of an incoming gas:

$$n = \frac{p}{k_B T},$$ (2.12)

where p and T are the given pressure (Pa) and temperature (K) respectively. The Boltzmann constant, k_B, is defined as the constant that relates thermal energy with temperature as $1.381 \cdot 10^{-23}$ J·K^{-1} [Goebel & Katz, 2008].

The electron temperature, T_e, may be expressed in eV and in K. The conversion between eV and K is

$$T_e(K) = \frac{T_e(eV) \cdot e}{k_B},$$ (2.13)

where e is the absolute value of an electron charge as $1.602 \cdot 10^{-19}$ C [Lieberman, 2003].

The Saha function describes the ratio of the square of the number of electrons (n_e) to the number of neutral atoms or molecules (n_n). The equation was developed in order to explain the light spectrum from stars. The equation uses the temperature of the bulk plasma to solve the Saha function. However, this assumes the bulk plasma temperature is approximately equal to the electron temperature. In a cold plasma where ionization occurs using radio-frequencies, the electron temperature is ten to hundred times higher than the bulk plasma temperature. Therefore, the Saha function uses the electron temperature [Jahn, 1968; Howard, 2002; Dinklage et al., 2005; Takao, 2006],

$$S(T_e) = \frac{n_e^2}{n_n} = 2 \cdot \frac{q_i}{q_n} \left(\frac{2\pi m_e k_B T_e}{h^2} \right)^{\frac{3}{2}} e^{-\frac{eE_i}{k_B T_e}},$$ (2.14)

where q_i and q_n are the ground-state degeneracy of an ion and a neutral respectively, m_e is the mass of the electron at $9.109 \cdot 10^{-31}$ kg, and the Planck's constant, h, is given as

6.626·10^{-34} J·s. The ground state degeneracies are dependent on the element [Dinklage *et al.*, 2005]. For example, ground state degeneracy for an ion and a neutral in hydrogen are one and two, respectively [Howard, 2002; Martinez-Sanchez, 2004a]. The Saha equation was initially used in the simulations to find the temperature of the plasma. However, it has been removed. The full derivation of the Saha equation is in appendix A. The remenance of the equation is in the code, shown in appendix H.

2.5.2 Collision Theory

Collision theory is the study of particle collisions by a plasma (full discussion is in appendix B). The particle collisions determine the conductivity in the plasma. Collisions are described in terms of thermal speed; it is the speed at which the individual species are moving in a plasma. However, the overall plasma group velocity is the bulk velocity. This is analogous to a swarm of bees. The velocities of the individual bees may be very fast, yet the bulk velocity of the swarm could be very slow. The average thermal velocity, c_e, of an electron is [Howard, 2004; Dinklage *et al.*, 2005]

$$c_e = \sqrt{\frac{8}{\pi} \frac{k_B T_e}{m_e}}.$$
(2.15)

The Debye length (λ_D), or characteristic length, is the distance over a plasma in which the electric field of the charged particle is shielded from the thermal motion of the other charged particles [Howard, 2004; Dinklage *et al.*, 2005],

$$\lambda_D = \sqrt{\frac{\epsilon_0 k_B T_e}{n_e e^2}},$$
(2.16)

where ϵ_0 is the permittivity of free space, $8.854 \cdot 10^{-12}$ F·m^{-1}.

The plasma parameter, Λ, is a dimensionless quantity that characterizes the unmagnetized plasma,

$$\Lambda = 4\pi \cdot n_e \cdot \lambda_d^3 \; . \tag{2.17}$$

Strongly coupled plasma has the plasma parameter less than one. This means the potential energy of particles inside the plasma is more significant than the kinetic energy. Weakly coupled plasma has the plasma parameter more than one. In this case, the kinetic and thermal energy is significantly higher than the potential energy. Often, the plasma parameter is higher than one. The parameter is cubically proportional to the Debye length. The logarithmic of the plasma parameter, *ln(Λ)*, is called the Coulomb parameter; normally, this value is between 6 and 16 [Howard, 2002; Dinklage *et al.*, 2005].

The collision frequency, f_v, depends on the number of particles in the first species (electron, ion, or neutral atoms or molecules), the thermal energy of the first species, and the cross-sectional area of the collisions from the two species [Howard, 2002; Dinklage *et al.*, 2005]. The collision frequency of a typical electron-ion scattering is [Howard, 2002]

$$f_{v,ei} = \frac{n_i Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_e^2 c_e^3} \; . \tag{2.18}$$

The corresponding cross-sectional area for electron-ion scattering is

$$\sigma_{cross\ section-ei} = \frac{Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_e^2 c_e^4} \; . \tag{2.19}$$

Collisions also occur between the electrons and neutral atoms and molecules (or simply neutrals). The collision frequency between the two is the same as to the electron-ion collision frequency, without the chance of recombination or having any electric forces [Howard, 2002; Dinklage *et al.*, 2005]

$$f_{v,en} = n_n c_e \pi r_n^2 .$$ (2.20)

The radius of the electron is much smaller than the radius of the neutral atom or molecule; therefore, the radius of the electron may be neglected in the equation. The cross-sectional area, πr_n^2, is approximated to 10^{-19} m^2, depending on the gas. With the electron-neutral and electron-ion collision frequencies, the collision frequency of the electrons in a given plasma is [Howard, 2002; Dinklage *et al.*, 2005]

$$f_{v,e} = f_{v,ei} + f_{v,en} .$$ (2.21)

One of the objectives in the present research is to use gas mixtures in the thruster. In a plasma comprised of a gas mixture, the simplest model to use is to assume that the two species ionize individually without any collision [Jahn, 1968],

$$A + E_{i,A} \leftrightarrow A^+ + e^-,$$ (2.22a)

$$B + E_{i,B} \leftrightarrow B^+ + e^-,$$ (2.22b)

where A and B represent any two atomic species. The Saha functions (from equation 2.14) of the two species are [Jahn, 1968]

$$S_A(T_{e,A}) = 2 \cdot \frac{q_{i,A}}{q_{n,A}} \left(\frac{2\pi m_{e,A} k_B T_{e,A}}{h^2} \right)^{\frac{3}{2}} e^{-\frac{e E_{i,A}}{k_B T_{e,A}}} ,$$ (2.23a)

$$S_B(T_{e,B}) = 2 \cdot \frac{q_{i,B}}{q_{n,B}} \left(\frac{2\pi m_{e,B} k_B T_{e,B}}{h^2}\right)^{\frac{3}{2}} e^{-\frac{eE_{i,B}}{k_B T_{e,B}}}. \tag{2.23b}$$

The electron density in a mixture is comprised from each of the individual elements [Jahn, 1968]

$$n_e = n_{i,A} + n_{i,B} . \tag{2.24}$$

The ion densities depend on the Saha functions in equations 2.23a and 2.23b. Introducing the heavy particle density, it is the density of the ions and neutrals. The density of the heavy particles for each element is given as [Jahn, 1968]

$$n_{h,A} = n_{i,A} + n_{n,A} , \tag{2.25a}$$

$$n_{h,B} = n_{i,B} + n_{n,B}. \tag{2.25b}$$

Hence, there are five species in a two-element gas mixture: n_e, $n_{i,A}$, $n_{i,B}$, $n_{n,A}$, $n_{n,B}$. Expanded to three or more elements, the number of species in a gas mixture are

$$a_s = 2^{a_{el}} + 1, \tag{2.26}$$

where a_s and a_{el} are the number of species and the number of elements, respectively. In a low-temperature plasma, the ionization of the gas mixture is less than the ionization of the two elements mutually exclusive to each other. The physical reason for this phenomenon is that the electrons can be readily recombined from either of the ion elements of the gas mixture [Jahn, 1968; Dinklage *et al.*, 2005].

Collision theory in gas mixtures is similar to the collision theory in a pure gas. In a two-element gas mixture plasma, the electron-ion collision frequency is modified such that it takes into account elements A and B [Jahn, 1968; Howard, 2002],

$$f_{v,ei} = (n_{i,A} + n_{i,B}) \frac{Z^2 e^4 \ln(\Lambda)}{2\pi\epsilon_0^2 m_e^2 c_e^3}. \tag{2.27}$$

The electron-neutral particle collision frequency is also taken into account for multiple species [Jahn, 1968; Howard, 2002]:

$$f_{v,en} = c_e \left(n_{n,A} \pi r_{n,A}^2 + n_{n,B} \pi r_{n,B}^2 \right). \tag{2.28}$$

2.5.3 Plasma in an Electric Field

The dielectric constant (ϵ_p) in a plasma, which depends on the collision frequency, solves for the conductivity of the plasma. It also determines the polarization of the electric field in a plasma, which describes whether the Lorentz force of the plasma can be used. The dielectric constant of a plasma is [Lieberman, 2003]

$$\epsilon_p = \epsilon_0 \left(1 - \frac{\omega_{pe}^2}{\omega(\omega - if_{v,e})} \right), \tag{2.29a}$$

$$\omega_{pe} = \sqrt{\frac{e^2 n_e}{\epsilon_0 m_e}}, \tag{2.29b}$$

where ω_{pe} is the electron plasma frequency, and ω is the value of the input frequency from the RF power plant. The value, $f_{v,e}$, is the collision frequency of the electron-ion and electron-neutral interactions. The dielectric constant is typically complex and coupled with the electron-ion and electron-neutral collision frequencies [Howard, 2002; Lieberman, 2003; Bellan, 2004].

The current density, $\mathbf{J}$, is defined as [Howard, 2002; Bellan, 2004; Goebel & Katz, 2008],

$$\mathbf{J} = -n_i e \boldsymbol{v}_i + n_e e \boldsymbol{v}_e \,. \tag{2.30}$$

The current density is linearly proportional to the electric field, **E**, [Lieberman, 2003]

$$\mathbf{J} = i\omega(\epsilon_p - \epsilon_0)\mathbf{E} \,. \tag{2.31}$$

When there is an input frequency from a power generator, the direction of the electric field is opposite of the current [Lieberman, 2003]. Figure 2.15 illustrates the electric field in a plasma.
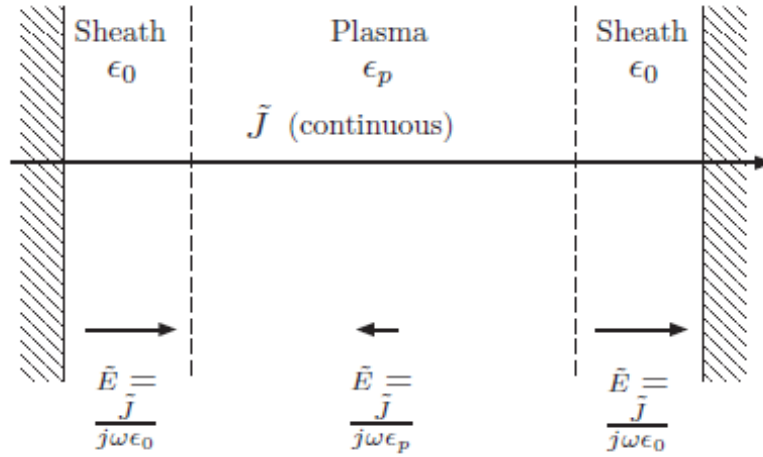
Figure 2.15: Illustration of a Negative Dielectric Constant Through the Plasma Medium

[Lieberman, 2003]

The electric field is much smaller inside the plasma than in the sheath between the plasma and the electrode. For further information, please see *Lieberman 2003*. The plasma conductivity is determined as the product of the frequency and the plasma dielectric constant [Lieberman, 2003],

$$\sigma_p = i\omega(\epsilon_p - \epsilon_0) \,. \tag{2.32}$$

34

Combining equation 2.29 with equation 2.32, the plasma conductivity is [Lieberman, 2003]

$$\sigma_p = \frac{e^2 n_e}{m_e \left(f_{v,e} + i\omega\right)}.$$ (2.33)

Similar to the plasma dielectric constant, the plasma conductivity is typically a complex number. Only in a pure DC discharge is the plasma conductivity a real number [Lieberman, 2003; Dinklage *et al.*, 2005],

$$\sigma_{p,DC} = \frac{e^2 n_e}{m_e f_{v,e}}.$$ (2.34)

2.5.4 Plasma Generation

Ohmic, or Joule, heating is the process in which the propellant inside a vessel is heated by an electrical current. There are two methods of Ohmic heating: capacitive and inductive [Howard, 2004]. Capacitive heating involves two electrodes with a potential difference. In a DC power supply, the potential difference must be in hundreds to tens of thousands of volts, depending on the gas pressure, in order to cause a plasma arc. An AC power supply does not require such a high potential difference in order to cause a plasma discharge. Inductive heating involves a solenoid carrying a high current to produce a magnetic field to cause a plasma discharge [Howard, 2002; Bellan, 2004; Goebel & Katz, 2008].

Ohmic heating is the current density and the electric field [Howard, 2002; Bellan, 2004; Goebel & Katz, 2008],

$$P_{abs} = \int_V \omega \cdot \left(\int_{\frac{1}{\omega}} \mathbf{J} \cdot \mathbf{E} dt \right) dV = \int_V \omega \cdot \left(\int_{\frac{1}{\omega}} \sigma_p \mathbf{E}^2 dt \right) dV, \tag{2.35}$$

where P_{abs} is the absorbing power. Ohmic heating is the dot product of the current density and the electric field, integrating over time multiplied by the angular frequency, being integrated over a specific volume. Using the method of contour integration, the absorption power for both capacitive and inductive heating can be simplified using the DC plasma conductivity and the phasor equivalent of the electric field [Lieberman, 2003; Howard, 2004; Goebel & Katz, 2008],

$$P_{abs} = \int_V \frac{1}{2} \left| \widetilde{\mathbf{E}} \right|^2 \sigma_{P,DC} \frac{f_{v,e}^2}{\omega^2 + f_{v,e}^2} \ dV. \tag{2.36}$$

2.5.5 Plasma as a Fluid

The purpose of modeling the plasma as a fluid is to simulate its flow in the electric engine. The behaviour of the plasma is described by the Navier-Stokes (NS) equations such that the fluid is considered electrically conductive. If the plasma is magnetized by a high current or if there is an external magnetic field, the equations are considered to be the magnetohydrodynamic (MHD) equations [Goebel & Katz, 2008; Bar-Meir, 2011]. The full discussion of the plasma modeled as a fluid (continuum mechanics) is discussed in appendix C.

The NS equations are written as [Goebel & Katz, 2008; Bar-Meir, 2011]

$$\frac{\partial \rho_e}{\partial t} + \boldsymbol{v} \cdot \nabla \rho_e = \dot{\rho}_e, \tag{2.37a}$$

$$\frac{\partial \rho}{\partial t} + \boldsymbol{v} \cdot \nabla \rho = \dot{\rho}, \tag{2.37b}$$

$$\frac{\partial \rho \boldsymbol{v}}{\partial t} + \boldsymbol{v} \cdot \nabla \rho \boldsymbol{v} = -\nabla p + \mu \nabla^2 \boldsymbol{v} - \nabla \left(\frac{2}{3} \mu \nabla \cdot \boldsymbol{v} \right) + \rho_q (\mathbf{E} + \boldsymbol{v} \times \mathbf{B}) - \rho f_{v,e}(\boldsymbol{v} - \boldsymbol{v_0}), \tag{2.37c}$$

$$\frac{\partial \mathcal{E}}{\partial t} + \boldsymbol{v} \cdot \nabla \mathcal{E} = \nabla \cdot \boldsymbol{\Theta} + \left(\mu \nabla \cdot \boldsymbol{v} - \left(\frac{2}{3} \mu \nabla \cdot \boldsymbol{v} \right) \right) \cdot \boldsymbol{v} + \rho_q \mathbf{E} \cdot \boldsymbol{v} + \mathbf{R}_{s_1} \cdot \boldsymbol{v} + Q_{s_1} - \Psi_{s_1}, \tag{2.37d}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}. \tag{2.37e}$$

The density continuity, equation 2.37b, prescribes that the total density is dependent on the rate of compression of the fluid, ($\dot{\rho} = -\rho \cdot (\nabla \cdot \boldsymbol{v})$) [Hutchinson & Friedberg, 2003; Goebel & Katz, 2008]. The electron continuity, equation 2.37a, describes the temporal and spatial changes in the electron density with respect to the production rate, $\dot{\rho}_e$. The production rate is the amount of electrons produced in the plasma during the discharge,

$$\dot{\rho}_e \cong \rho_n \rho_e \sigma_{impact,i} c_e . \tag{2.38}$$

The impact ionization cross-section area, $\sigma_{impact,i}$, is the area at which collisions cause ionization and excitation. This value depends on the atomic element [Goebel & Katz, 2008].

The plasma momentum, equation 2.37c, prescribes that the change in momentum in a plasma depends on the pressure gradient, viscous flow at subsonic speed, viscous flow at supersonic speed, Lorentz force density, and collision transfer momentum [Goebel & Katz, 2008]. The Lorentz force describes the force acting on the plasma due to electromagnetism [Martinez-Sanchez, 2004a],

$$\mathbf{F}_{Lorentz} = \iiint_V \left(\rho_q \mathbf{E} + \mathbf{J} \times \mathbf{B} \right) dV. \tag{2.39}$$

The electric field, **E**, is nonzero only whenever a capacitive discharge occurs [Martinez-Sanchez, 2004a].

The energy of the plasma, equation 2.37d, equals to the divergence of the heat flux, Ohmic heating, momentum change of a particle due to collisions, and the heat exchange between particles. The energy density, $\mathcal{E}$, is the summation of the internal energy, the kinetic energy, and the energy of the magnetic field [Sankaran *et al.,* 2002]

$$\mathcal{E} = \underbrace{\frac{p}{\gamma - 1}}_{Internal\ Energy} + \underbrace{\frac{1}{2}\rho \boldsymbol{v} \cdot \boldsymbol{v}}_{Kinetic\ Energy} + \underbrace{\frac{\mathbf{B} \cdot \mathbf{B}}{2\mu_0}}_{Magnetic\ Field\ Energy}. \tag{2.40}$$

Note that γ is the specific heat capacity ratio of the propellant gas. The energy of the magnetic field is assumed to be produced from either the plasma by an external magnetic field.

The heat flux is denoted as [Goebel & Katz, 2008]

$$\boldsymbol{\Theta} = -\kappa \nabla T. \tag{2.41}$$

The thermal conductivity, κ, of the plasma is strongly dependent on the electron temperature and the collision frequency [Goebel & Katz, 2008],

$$\kappa = 3.2 \frac{ne k_B T_e}{f_{v,e} m_e}. \tag{2.42}$$

$\mathbf{R}_{s_1}$ is the rate of momentum change per volume in the collisions. The subscripts s_1 and s_2 denote the primary and secondary species (electrons, ions and neutral atoms or molecules), respectively [Goebel & Katz, 2008],

$$\mathbf{R}_{s_1} = -\sum_{s_2} n_{s_1} m_{s_1} f_{v,s_1,s_2} (\boldsymbol{v}_{s_1} - \boldsymbol{v}_{s_2}). \tag{2.43}$$

Also, Ψ_{s_1} is the rate of energy loss per volume due to inelastic collisions such as ionization and excitation [Goebel & Katz, 2008],

$$\Psi_{s_1} = \frac{e \dot{\rho}_{s_1}}{m_{s_1}} E_i. \tag{2.44}$$

Q_{s_1} is the rate of heat exchange between particles per volume. The heat exchange consists of frictional, Q^R, and thermal, Q^T, energies, [Goebel & Katz, 2008]

$$Q_{s_1} = Q_{s_1}^R + Q_{s_1}^T, \tag{2.45a}$$

$$Q_{s_1}^R = -\sum_{s_2} \mathbf{R}_{s_1,s_2} \cdot \boldsymbol{v}_{s_1}, \tag{2.45b}$$

$$Q_{s_1}^T = \sum_{s_2} 2 n_{s_1} \frac{m_s}{m_a} f_{v,s_1,s_2} \frac{3}{2} \left(\frac{k_B T_{s_1}}{e} - \frac{k_B T_{s_2}}{e} \right). \tag{2.45c}$$

The right hand sides of the NS equations can be converted to fluxes in order to apply them to the computational fluid dynamic model. The Lorentz force density can be expressed as the sum of divergence of the Maxwell stress tensor and the temporal derivative of directional energy flux density. The Maxwell stress tensor is [Sankaran *et al.,* 2002]

$$\overleftrightarrow{\mathcal{B}} = \begin{bmatrix} \epsilon_0 E_1 E_1 + \dfrac{1}{\mu_0} B_1 B_1 - \dfrac{1}{2}\left(\epsilon_0 E^2 + \dfrac{1}{\mu_0} B^2\right) & \epsilon_0 E_1 E_2 + \dfrac{1}{\mu_0} B_1 B_2 & \epsilon_0 E_1 E_3 + \dfrac{1}{\mu_0} B_1 B_3 \\[2ex] \epsilon_0 E_2 E_1 + \dfrac{1}{\mu_0} B_2 B_1 & \epsilon_0 E_2 E_2 + \dfrac{1}{\mu_0} B_2 B_2 - \dfrac{1}{2}\left(\epsilon_0 E^2 + \dfrac{1}{\mu_0} B^2\right) & \epsilon_0 E_2 E_3 + \dfrac{1}{\mu_0} B_2 B_3 \\[2ex] \epsilon_0 E_3 E_1 + \dfrac{1}{\mu_0} B_3 B_1 & \epsilon_0 E_3 E_2 + \dfrac{1}{\mu_0} B_3 B_2 & \epsilon_0 E_3 E_3 + \dfrac{1}{\mu_0} B_3 B_3 - \dfrac{1}{2}\left(\epsilon_0 E^2 + \dfrac{1}{\mu_0} B^2\right) \end{bmatrix}. \tag{2.46}$$

The Lorentz force is the sum of the volumetric integration of the divergence of the Maxwell stress tensor and the temporal derivative of directional energy flux density [Sankaran *et al.,* 2002],

$$\mathbf{F}_{Lorentz} = \iiint_V \left(\nabla \cdot \overleftrightarrow{\mathcal{B}} - \epsilon_0 \frac{\partial}{\partial t} (\mathbf{E} \times \mathbf{B}) \right) dV . \tag{2.47}$$

In equation 2.37e, Faraday's law can be expressed as a flux. Using Ohm's law, the electric field yields two terms. The first term is the convective diffusion of the magnetic field. This is written as the antisymmetric tensor of the magnetic field and the velocity of the particle [Sankaran *et al.,* 2002],

$$\nabla \cdot (\boldsymbol{v}\mathbf{B} - \mathbf{B}\boldsymbol{v}) . \tag{2.48}$$

The second term is the resistive diffusion. This term is represented as the divergence of the resistive diffusion tensor, $\overleftrightarrow{\mathcal{E}}_{res}$, and it is the curl of the electric field [Sankaran *et al.,* 2002],

$$\nabla \cdot \overleftrightarrow{\mathcal{E}}_{res} = -\nabla \times \mathbf{E} . \tag{2.49}$$

The final term of flux conversion is the energy inside the plasma,

$$\nabla \cdot \mathbf{q} = \nabla \cdot \left(\underbrace{\left(\mu \nabla \boldsymbol{v} - \left(\frac{2}{3} \mu \nabla \cdot \boldsymbol{v} \right) \right) \cdot \boldsymbol{v}}_{Viscous\ Heating} + \underbrace{\frac{\mathbf{B} \times (\mathbf{E} + \boldsymbol{v} \times \mathbf{B})}{\mu_0}}_{Ohmic\ Heating} + \underbrace{\Theta}_{Conduction} \right). \tag{2.50}$$

The first term is the viscous heating, the second is Ohmic heating, and the third term is thermal conductivity [Sankaran *et al.,* 2002]. Hence, the NS equations in a flux model is now represented as

$$\frac{\partial}{\partial t} \begin{bmatrix} \rho_e \\ \rho \\ \rho \boldsymbol{v} \\ \mathcal{E} \\ \mathbf{B} \end{bmatrix} + \nabla \cdot \begin{bmatrix} \rho_e \boldsymbol{v} \\ \rho \boldsymbol{v} \\ \rho \boldsymbol{v} \boldsymbol{v} \\ \mathcal{E}\boldsymbol{v} - \overleftrightarrow{\mathcal{B}} \cdot \boldsymbol{v} \\ \boldsymbol{v}\mathbf{B} - \mathbf{B}\boldsymbol{v} \end{bmatrix} \tag{2.51}$$

$$= \nabla \cdot \begin{bmatrix} 0 \\ 0 \\ -\overleftrightarrow{\mathbf{p}} + \overleftrightarrow{\mathcal{B}} + \mu \nabla \boldsymbol{v} - \left(\frac{2}{3} \mu \nabla \cdot \boldsymbol{v} \right) \\ \mathbf{q} \\ \overleftrightarrow{\mathcal{E}}_{res} \end{bmatrix} + \begin{bmatrix} \dot{\rho}_e \\ 0 \\ -\epsilon_0 \frac{\partial}{\partial t}(\mathbf{E} \times \mathbf{B}) - \rho f_{v,e}(\boldsymbol{v} - \boldsymbol{v_0}) \\ 0 \\ \mathbf{0} \end{bmatrix}.$$

For the simulations, equation 2.51 is simplified to optimize the numerical calculations. The stress due to the walls of the nozzle, the force due to collisions, the temporal derivative of directional energy flux density, and external forces are ignored [Sankaran *et al.,* 2002],

$$\frac{\partial}{\partial t} \underbrace{\begin{bmatrix} \rho \\ \rho \boldsymbol{v} \\ \mathcal{E} \\ \mathbf{B} \end{bmatrix}}_{\mathbf{V}} + \nabla \cdot \underbrace{\begin{bmatrix} \rho \boldsymbol{v} \\ \rho \boldsymbol{v} \boldsymbol{v} \\ \mathcal{E}\boldsymbol{v} - \overleftrightarrow{\mathcal{B}} \cdot \boldsymbol{v} \\ \boldsymbol{v}\mathbf{B} - \mathbf{B}\boldsymbol{v} \end{bmatrix}}_{\mathbf{X}_{conv}} = \nabla \cdot \underbrace{\begin{bmatrix} 0 \\ \overleftrightarrow{\mathcal{B}} - \overleftrightarrow{\mathbf{p}} \\ \mathbf{q} \\ \overleftrightarrow{\mathcal{E}}_{res} \end{bmatrix}}_{\mathbf{X}_{diss}}. \tag{2.52}$$

The electron production equation is separated and entered in the plasma generation equations, which describe the temperatures and densities of the electrons, ions, and neutral particles,

$$\frac{d}{dt} n_e(t) = \dot{n}_e \quad , \tag{2.53a}$$

$$\frac{d}{dt}\left(\frac{3}{2} n_e(t) k_B T_e(t)\right) = Q_{abs} - \Psi \quad , \tag{2.53b}$$

$$\frac{d}{dt} n_n(t) = -\dot{n}_e \quad , \tag{2.53c}$$

$$\frac{d}{dt}\left(\frac{3}{2} n_n(t) k_B T_h(t)\right) = Q_{heat} \quad , \tag{2.53d}$$

where Q_{abs}, Q_{heat} and T_h are the absorption power density, the heat exchange between particles, and the temperature of the heavy particles, respectively. The temperature of the heavy particles is comprised of the temperatures of the ions and neutral particles. The absorption power density is the absorption power (equation 2.35) per unit volume [Goebel & Katz, 2008].

$$\dot{n}_e = n_n n_e \cdot (A T_e^4 + B T_e^3 + C T_e^2 + D T_e + E) \quad , \tag{2.54a}$$

$$Q_{abs} = \frac{P_{abs}}{V} \quad , \tag{2.54b}$$

$$\Psi = e n_e E_i \quad , \tag{2.54c}$$

$$Q_{heat} = 3 n_e \frac{m_e}{m_n} f_{v,e} \left(\frac{k_B T_e}{e} - \frac{k_B T_h}{e}\right) \quad . \tag{2.54d}$$

The production rate is simplified to a quartic formula for computational purposes. The values A, B, C, D, and E are element-specific; the values are in appendix D.

The electron density and temperature are entered in the heat equation to solve for the overall temperature of the plasma. The boundary conditions are the nozzle wall

temperature is kept constant, C_{temp}, and the aluminum tip is adiabatic [Bose, 2004; Wendt 2009],

$$\left.\frac{dT}{dr}\right|_{r=cathode\ wall} = 0$$
$$T(r)|_{r=anode\ wall} = C_{temp}$$
.
(2.55)

The heat equation is simplified to solve for the steady state of the plasma, and the equation varies along the radial direction such that [Bose, 2004],

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\kappa \cdot \frac{\partial T}{\partial r}\right) + Q_{abs} = 0 ,$$
(2.56)

where the value κ is the thermal conductivity of the plasma from equation 2.42. After solving for the plasma temperature, the value is entered in the plasma dynamics equation 2.52.

2.6 Computational Fluid Dynamics

The computational fluid dynamic model solves the fluid equations from Section 2.5.5. The meshing of the numerical model is transformed from a standard rectangular grid *(x, y)* to a nozzle-shaped grid (ξ, η) [Wendt, 2009],

$$\xi = x$$
$$\eta = y \cdot f^{-1}(x)$$
.
(2.57)

The function, *f(x)*, goes along the curve of the nozzle. This characterizes the nozzle-shaped grid in the physical plane as a rectilinear grid in the computational plane, as seen in figure 2.16. This method is called a boundary-fitted coordinate system [Wendt, 2009].
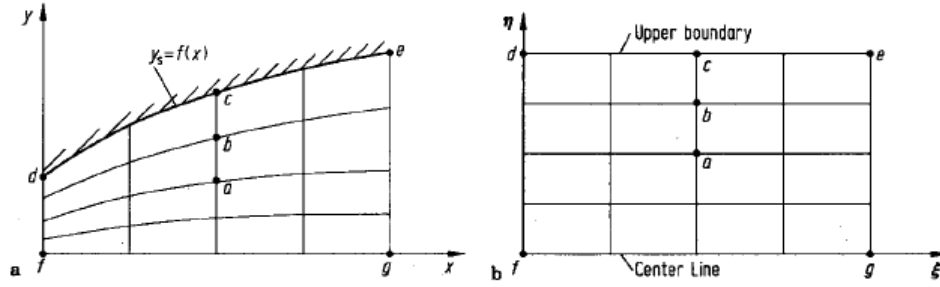
Figure 2.16: Boundary-Fitted Coordinate System [Figure 6.5 in Wendt, 2009]

Equation 2.52 can be written in the divergence form [Sankaran *et al.*, 2002],

$$\frac{\partial \mathbf{V}}{\partial t} + \nabla \cdot \mathbf{X}_{conv} = \nabla \cdot \mathbf{X}_{diss} \, , \tag{2.58}$$

where $\mathbf{V}$ is the solution vector. Also, $\mathbf{X}_{conv}$ and $\mathbf{X}_{diss}$ are the convective and dissipative flux tensors, respectively. These tensors are decoupled into two flux tensors, $\mathbf{X}$ and $\mathbf{Y}$, such that, [Wendt, 2009]

$$\frac{\partial \mathbf{V}}{\partial t} + \frac{\partial \mathbf{X}}{\partial \mathbf{x}} + \frac{\partial \mathbf{Y}}{\partial \mathbf{y}} = 0 \, . \tag{2.59}$$

The MacCormack method is a modified version of the Lax-Wendroff method that is simpler to implement in the present simulation. This method has a predictor and a corrector. The predictor step solves for the estimated value $v_{i,j}^{\overline{n+1}}$ by using the forward difference method [Wendt, 2009],

$$v_{i,j}^{\overline{n+1}} = v_{i,j}^{n} - \Delta t \cdot \left(\frac{x_{i+1,j}^{n} - x_{i,j}^{n}}{\Delta x_{step}} + \frac{y_{i,j+1}^{n} - y_{i,j}^{n}}{\Delta y_{step}} \right) . \tag{2.60}$$

Note that x_i and y_i are the steps in the ξ and η direction in the nozzle-shape grid. The corrector corrects the predicted value $v_{i,j}^{\overline{n+1}}$ by using the backward difference method [Wendt, 2009],

$$v_{i,j}^{n+1} = \frac{1}{2}\left(v_{i,j}^n + v_{i,j}^{\overline{n+1}} - \Delta t \cdot \left(\frac{x_{i,j}^n - x_{i-1,j}^n}{\Delta x_{step}} + \frac{y_{i,j}^n - y_{i,j-1}^n}{\Delta y_{step}}\right)\right). \tag{2.61}$$

Equations 2.60 and 2.61 employ the MacCormack's method. This ensures that the final value does not overshoot [Wendt, 2009].

2.7 Electrothermal Propulsion

2.7.1 Ideal Nozzle Conditions

The basis of the hybrid electric thruster is firstly the design of the electrothermal propulsion system. Electrothermal propulsion is the simplest form of electric propulsion since it uses thermodynamic expansion to accelerate the plasma [Sutton & Biblarz, 2010]. In rocketry, nozzle theory follows from basic thermodynamics. The basic thermodynamic principles are the temperature, pressure, volume, enthalpy, and entropy [Sutton & Biblarz, 2010],

$$\frac{T_2}{T_1} = \left(\frac{p_2}{p_1}\right)^{\frac{\gamma-1}{\gamma}} = \left(\frac{V_1}{V_2}\right)^{\gamma-1}. \tag{2.62}$$

Equation 2.62 shows the relations between temperature, pressure, and volume in an isentropic flow. In the theory of an ideal nozzle, several assumptions are made [Sutton & Biblarz, 2010]:

1) The propellant in the rocket is homogeneous: The density of the fluid is uniform throughout.

2) The propellant is in a gaseous state.

3) The propellant nominally obeys the ideal gas law.

4) The flow is adiabatic. This means that there is no significant change in temperature of the walls in comparison to that of the fluid.

5) Friction along the walls is negligible.

6) There are no flow discontinuities.

7) The flow is steady.

Within an isentropic and adiabatic flow, the relative entropy of the propellant is zero; thus no irreversible energy loss [Sutton & Biblarz, 2010].

2.7.2 Isentropic Nozzle

The isentropic nozzle employed for this study is a conical C-D nozzle design.
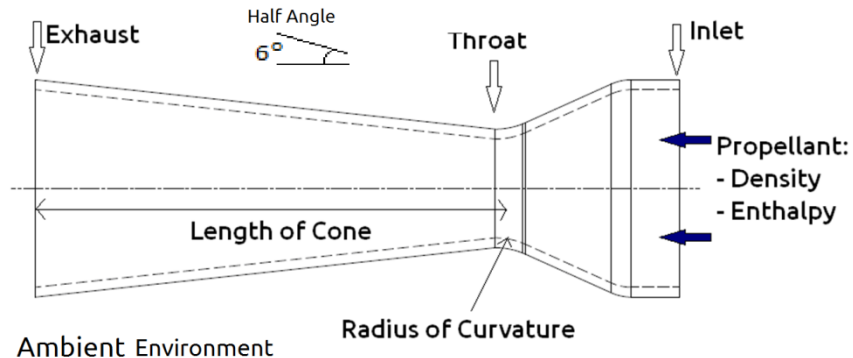
Figure 2.17: C-D Nozzle and its properties

A C-D nozzle has a converging section between the inlet and throat, and a diffuser between the throat and the exhaust. The throat is the point where the cross-sectional area of the nozzle is the smallest. At that point, the velocity of the propellant may be sonic if there is sufficient pressure coming from the inlet (choked flow condition). The ambient pressure ranges from standard (sea level) pressure on Earth to very low ambient pressure in space. For the usage of electric propulsion, the ambient environment is assumed to be in celestial space (pressure is less than 10^{-5} Torr) [Sutton & Biblarz, 2010].

In the sixth assumption discussed above for the performance of the nozzle, the pressure ratio of the inlet to atmosphere, or ambient environment, must be at least ten in order to avoid flow discontinuity (e.g., shock wave standing in the nozzle expansion) and to achieve supersonic exhaust speeds. Following *Sutton & Biblarz*, the exhaust Mach number, Ma_e, is dependent on the cross-sectional area ratio of the exhaust, A_e, and the throat, A_t, and the specific heat capacity ratio of the propellant gas, γ, [Sutton & Biblarz, 2010]

$$\frac{A_e}{A_t} = \frac{1}{Ma_e} \cdot \left(\frac{1 + \frac{\gamma - 1}{2} \cdot Ma_e{}^2}{\frac{\gamma + 1}{2}} \right)^{\frac{\gamma+1}{2(\gamma-1)}} . \tag{2.63}$$

Solving for the Mach number given the ratio of the exhaust to the throat area yields two possible solutions (subsonic and supersonic). The Mach number is greater than one (supersonic) for the present case.

The throat-inlet pressure ratio depends only on the specific heat ratio of the fluid when the flow is choked,

$$\frac{p_t}{p_i} = \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma}{\gamma-1}}.$$

(2.64)

Equation 2.64 assumes that the inlet flow speed is zero ($Ma_i = 0$). The expansion ratio at the nozzle exhaust exit plane depends on the specific heat ratio and the exit flow Mach number,

$$\frac{p_e}{p_i} = \left(\left(1 + \frac{\gamma-1}{2} \cdot Ma_e{}^2\right)^{\frac{\gamma}{\gamma-1}}\right)^{-1}.$$

(2.65)

The exhaust velocity of the thruster, v_{ex}, is the velocity of the propellant as it ejects from the nozzle. This velocity depends on the ratio of the exhaust pressure to the throat pressure. The equation includes the inlet temperature, T_i, of the propellant,

$$v_{ex} = \sqrt{T_i \cdot \frac{\mathcal{R}}{m} \cdot \frac{2\gamma}{\gamma+1} \cdot \left(1 - \left(\frac{p_e}{p_i}\right)^{\frac{(\gamma-1)}{\gamma}}\right)}.$$

(2.66)

The ideal mass flow rate determines the most efficient thrust possible in a thruster,

$$\dot{m}_{ideal} = A_t p_i \gamma \cdot \sqrt{\frac{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}} \cdot m}{\gamma \mathcal{R} T_i}}.$$

(2.67)

Therefore, the ideal vacuum thrust in an efficient thruster is

$$F_{thrust,ideal} = \dot{m}_{ideal} \cdot v_{ex}.$$

(2.68)

The calculation of the thrust in non-vacuum conditions is

$$F_{thrust} = \dot{m} \cdot v_{ex,eff} = \dot{m} \cdot v_{ex} + (p_e - p_a) \cdot A_e.$$

(2.69)

The effective exhaust velocity, $v_{ex,eff}$, takes into account the difference between the exhaust exit and atmospheric pressures,

$$v_{ex,eff} = v_{ex} + (p_e - p_a) \cdot \frac{A_e}{\dot{m}} \ . \tag{2.70}$$

The specific impulse of the thruster, as seen in equation 2.5, is the effective exhaust velocity divided by Earth's gravitational acceleration at sea level [Sutton & Biblarz, 2010].

The direction of the exhaust velocity is not completely in the axial component if using a conical nozzle. The radial direction of the exhaust velocity does not contribute to the thrust. Therefore, there is loss in performance because of the divergence of the conical nozzle. To achieve the best performance, the angle of rise, α_{rise} (i.e. the cone half-angle), has to be less than 15 degrees. Knowing the angle of rise, the radii of the throat, r_t, and exhaust, r_e, the length of the cone is [Komerath, 2004; Sutton & Biblarz, 2010]

$$L_{cone} = \left(r_t \left(\frac{r_e}{r_t} - 1 \right) + 1.5 \cdot r_t (\sec(\alpha_{rise}) - 1) \right) \cdot \cot(\alpha_{rise}). \tag{2.71}$$

The value, $1.5 \cdot r_t$, is the radius of curvature, shown in figure 2.18.
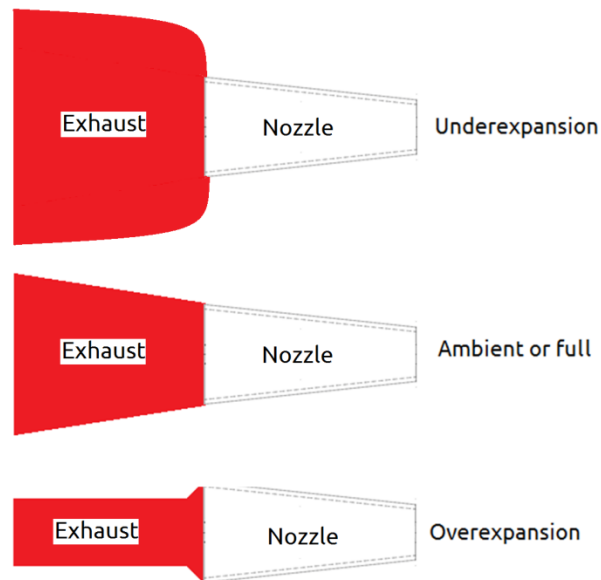
Figure 2.18: Exhaust Expansions

Figure 2.18 shows the different possible exhaust expansions in a nozzle. If the exhaust pressure is higher than the atmospheric pressure, then the nozzle is said to be underexpanded. This indicates that the inlet pressure is higher than required. If the exhaust pressure and atmospheric pressure are equal, then the nozzle is said to be fully expanded. This means the engine is operating under ideal conditions. If the atmospheric pressure is greater than the exhaust pressure, then the nozzle is said to be overexpanded [Sutton & Biblarz, 2010].

The exhaust flow of the engine forms a standing diamond pattern of shocks and expansion waves, which can be explained in part from the Prandtl-Meyer expansion equation which gives the flow deflection angle, v_{PM} [Darmofal, 2005]:

$$v_{PM}(M) = \sqrt{\frac{\gamma + 1}{\gamma - 1}} \cdot \tan^{-1} \sqrt{\frac{\gamma - 1}{\gamma + 1}(Ma^2 - 1)} - \tan^{-1}\sqrt{Ma^2 - 1}, \tag{2.72}$$

$$\theta_{PM} = v_{PM}(Ma_2) - v_{PM}(Ma_1), \tag{2.73}$$

where θ_{PM} is the net deflection angle of the local flow in passing through an expansion wave fan. The specific heat capacity ratio and local flow Mach number determine the value of the angle. The angles of the shocks that appear in the diamond pattern can be as seen from shock theory or numerical flow calculations [Martinez-Sanchez, Rocket Propulsion, 2004; Darmofal, 2005].

2.8 Electrostatic Propulsion

The primary limitation of electrothermal thrusters is that their specific impulse is dependent on the temperature of the gas or plasma. For small spacecraft where the maximum power allocated to the propulsion system is 20 W, the temperature of the heated gas would be around 100 to 200 degrees above room temperature. Electrostatic propulsion does not depend on the temperature of the plasma, but on the high voltage electrodes used to propel the ions [Goebel & Katz, 2008; Sutton & Biblarz, 2010]. Therefore, the hybrid electric thruster for this study utilizes the electrostatic thruster mode for sustained low-power low-thrust propulsion mission segments.

Electrostatic propulsion is involves accelerating ions between two electrodes that have an electric potential difference between them. The typical thrust range is between 0.01 mN and 500 mN, yet the specific impulse ranges in the thousands of seconds. In small spacecraft, the low thrust produced from the electrostatic thruster is sufficient when

integrated over time to perform large-scale orbital manoeuvres. A problem with electrostatic propulsion is the negative charge accumulation by the spacecraft from the positive ion exhaust. To rectify this problem, a neutralizer emits electrons to neutralize the ions [Martinez-Sanchez, 2004a; Goebel & Katz, 2008; Sutton & Biblarz, 2010].

2.8.1 Ion Optics

Ion optics involves focusing plasma beams through a set of charged electrodes. The first electrode, which is called the screen electrode, has a high positive potential. This extracts the ions from the plasma. The second electrode, which is called the acceleration electrode, has a low negative potential. The potential difference between the first and second electrodes accelerates the ions [Martinez-Sanchez, 2004a; Goebel & Katz, 2008; Sutton & Biblarz, 2010].
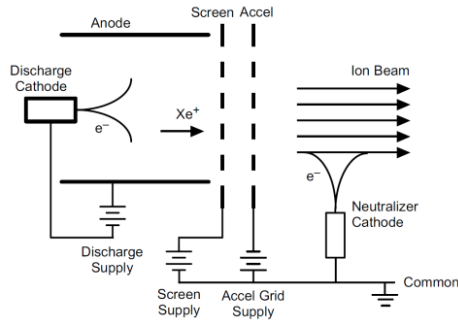
Figure 2.19: Electrostatic Thruster Configuration [Figure 5.3 from Goebel & Katz, 2008]

Figure 2.19 illustrates the configuration of the screen and electrodes, along with the plasma generation chamber and the electronics schematic. If the potential difference between the screen and acceleration electrodes is less than the electron temperature of the

plasma, then a Debye sheath is formed at the screen electrode. The Debye sheath is the boundary that contains high amount of ions and confines the escaping electrons. A small amount of electrons exist in the Debye sheath. However, if the potential difference is greater than the electron temperature, then a Child-Langmuir sheath is formed. In a Child-Langmuir sheath, the electrons are sufficiently repelled from the sheath that the boundary is saturated with ions. Therefore, almost no electrons exist in this sheath [Goebel & Katz, 2008]. Discussion on the plasma sheath is discussed in appendix E.
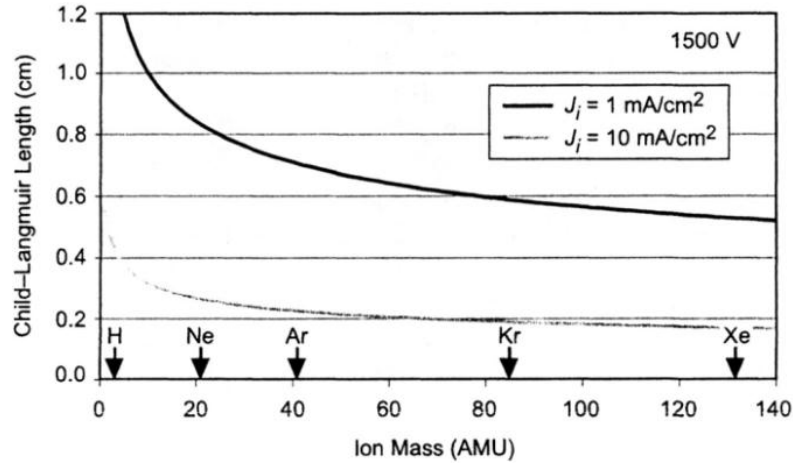
Figure 2.20: Child-Langmuir sheath length versus ion mass for two ion current densities at 1500-V acceleration voltage [Figure 5.1 from Goebel & Katz, 2008]

Figure 2.20 illustrates the Child-Langmuir sheath length versus the ion mass with relation to the current density. The graph shows that the heavier ions and the higher current density produce a shorter sheath length [Goebel & Katz, 2008].
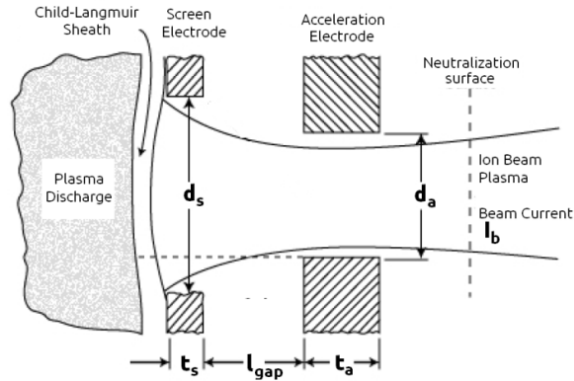
Figure 2.21: Schematic of the ion optics of one aperture in an electrostatic thruster

[Lecture 13-14, Figure 2 from Martinez-Sanchez, 2004a]

Figure 2.21 illustrates how the ions accelerate through the electrodes to give propulsion [Martinez-Sanchez, 2004a]. In front of the screen electrode, a Child-Langmuir sheath is formed with high amounts of ions [Martinez-Sanchez, 2004a].

Following *Goebel & Katz,* the maximum possible ion beam current, the current at which the ions accelerate from the electrodes, is limited to the perveance, $P_{per,max}$, at a given total voltage, V_T, [Goebel & Katz, 2008]

$$I_{b,max} = P_{per,\max} V_T^{\frac{3}{2}}.$$

(2.74)

The perveance is the amount of current an accelerator can extract and focus into a beam relative to the total voltage. The total voltage is the amount of voltage applied between the screen and the acceleration electrodes [Goebel & Katz, 2008],

$$V_T = |V_{screen}| + |V_{acc}|.$$

(2.75)

The maximum Child-Langmuir current density is derived as

$$J_{max} = \frac{4\epsilon_0}{9} \cdot \sqrt{\frac{2e}{m_i}} \cdot \frac{V_T^{\frac{3}{2}}}{l_e^2},$$

(2.76)

where m_i is the mass of the ions. The sheath thickness, l_e, is dependent on the gap and the dimensions of the screen aperture [Goebel & Katz, 2008],:

$$l_e = \sqrt{\left(l_{gap} + t_s\right)^2 + \frac{d_s^2}{4}}.$$

(2.77)

Given the Child-Langmuir equation, the maximum ion beam current is

$$I_{b,max} = \frac{4\epsilon_0}{9} \cdot \sqrt{\frac{2e}{m_i}} \cdot \frac{V_T^{\frac{3}{2}}}{l_e^2} \cdot \left(\frac{\pi d_s^2}{4}\right).$$

(2.78)

The exhaust velocity of the ion depends on the net voltage, V_b,

$$v_{ex,i} = \sqrt{\frac{2eV_b}{m_i}}.$$

(2.79)

The net, or beam, voltage is the absolute value of the difference of voltages of the screen electrode to the ground potential,

$$V_b = \left|V_{screen} - V_{ground}\right|.$$

(2.80)

The maximum thrust is given by

$$F_{beam,max} = N_{aperture} A_{active} J_{max} \cdot \frac{m_i v_{ex,i}}{e} \cdot \frac{I_b}{I_i} = N_{aperture} A_{active} \cdot \frac{8}{9} \epsilon_0 \cdot \frac{I_b}{I_i} \cdot \sqrt{\frac{V_b}{V_T}} \cdot \left(\frac{V_T}{l_e}\right)^2,$$

(2.81)

where A_{active} is the area of the acceleration aperture, and $N_{aperture}$ is the number of apertures in the electrode [Goebel & Katz, 2008]. The total ion current, I_i, is the positive current inside the plasma chamber. Equation 2.81 shows the maximum thrust is

dependent on that the maximum current density possible under ideal conditions. Therefore, the force of the ion thruster is calculated by the measured current density, J, [Wirz, 2006; Lubey, 2011],

$$F_{beam} = N_{aperture}A_{active} \cdot J \cdot \frac{m_i v_{ex}}{e} \cdot \frac{I_b}{I_i}. \tag{2.82}$$

The measurement of the current density is discussed in Section 2.10.

2.8.2 Electron Backstreaming

Electron backstreaming is defined as electrons from the neutralizer going back to the plasma generation chamber [Goebel & Katz, 2008].
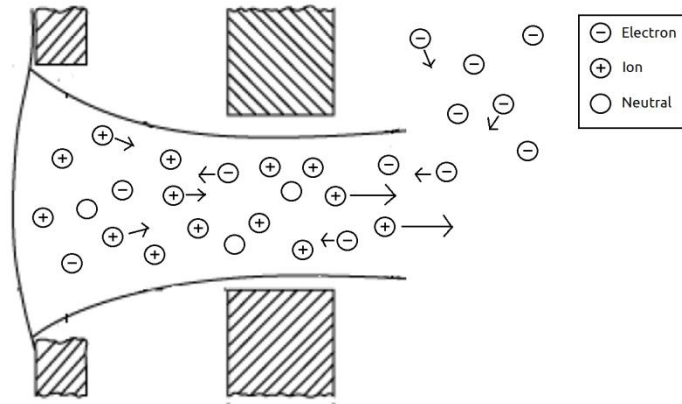
Figure 2.22: The Directions of the Ion and Electron flow

Electron backstreaming causes the ions and electrons recombining in the plasma chamber, hindering the performance in electrostatic thrusters. It occurs when the ratio of the net voltage (the potential between the screen electrode and the ground) to the total voltage (the total positive potential between the screen and acceleration electrodes) of the electrodes is near unity [Goebel & Katz, 2008]. The barrier from the accelerator electrode

56

is too weak to prevent the electrons streaming back to the generator. H.R. Kaufmann developed a theoretical maximum for the net-total voltage ratio to prevent backstreaming [Martinez-Sanchez, 2004a],

$$R_{max} = \left(\frac{V_b}{V_T}\right)_{max} = 1 - \frac{1}{5 \cdot \left(\frac{l_{gap}}{d_a} \cdot \exp\left(\frac{t_a}{d_a}\right)\right)}. \tag{2.83}$$

2.8.3 Neutralizing the Ions

The ions from the exhaust attach to any metallic surface on the spacecraft. Therefore, the spacecraft becomes charged, and many components may encounter electrical problems. In order to avoid this problem, a neutralizer is attached at the end of the engine, emitting electrons to neutralize the ions [Martinez-Sanchez, 2004a; Goebel & Katz, 2008].

There are two methods to emit electrons: heated tungsten filament and hollow cathode. The heated tungsten filament is a simple device to develop, but a large amount of energy is required to heat the filament [Martinez-Sanchez, 2004a; Goebel & Katz, 2008].

An alternative to a heated tungsten filament is a hollow cathode. A hollow cathode is comprised of a cathode insert, such as barium-oxide, inside a nozzle, where the heating coil is wrapped around the nozzle. There is research being dedicated to improving hollow cathodes for electrostatic thrusters. However, this is not a focus of the current research here since it is more suited for research by material engineering experts. Below is a table

of the different neutralizers and their current densities [Goebel & Katz, 2008]. Note that the neutralizer is illustrated in figure 3.20.

Table 2.3: List of Neutralizers and their Current Densities

Neutralizer	Current Density (A/cm^2)
Tungsten Filament (1950 C)	0.018
Tungsten Filament (2450 C)	2.110
Tungsten Filament (2950 C)	59.885
BaO Hollow Cathode (1000 C)	3.000
BaO Hollow Cathode (1300 C)	100.000
LaB_6 Hollow Cathode (1100 C)	0.010
LaB_6 Hollow Cathode (1350 C)	1.000

2.9 Electromagnetic Propulsion

The hybrid electric thruster also has an electromagnetic propulsion system mode to give higher levels of thrust when necessary. Electromagnetic propulsion uses magnetic fields to accelerate the plasma [Jahn, 1968; Martinez-Sanchez, 2004a]. The magnetic field can come from the plasma itself (self-generated or induced field) or it can come from a separate magnetic source (applied field). This section covers only the self-generated field. The hybrid electric thruster for this study does not use external magnetic sources to accelerate the plasma [Martinez-Sanchez, 2004a].

A self-generated field is defined as the magnetic field is produced by the plasma. This phenomenon is explained by the examination of Ampère's law [Serway & Jewett, 2004],

$$\mu_0 \mathbf{J} = \nabla \times \mathbf{B} \tag{2.84a}$$

$$\oint_C \mathbf{B} \cdot d\boldsymbol{\ell} = \mu_0 \iint_S \mathbf{J} \cdot d\mathbf{S}. \tag{2.84b}$$

The current in a coaxial thruster flows from the anode to the cathode. The direction of the resulting magnetic field is azimuthal. According to equation 2.84b, the surface integral of the current density of the total charge is the current going through the plasma. Thus, the magnetic field, which has only azimuthal direction, in the plasma is [Jahn, 1968; Martinez-Sanchez, 2004a]

$$B_\theta(r,z) = \frac{\mu_0 I}{2\pi r}\left(1 - \frac{z}{z_0}\right), \tag{2.85}$$

where I is the current, r is the radial position between the electrodes, z is the axial position relative to electrodes (see figure 2.23), and z_0 is the total length of the electrode. In figure 2.23, the current flows in the radial and axial direction [Martinez-Sanchez, 2004a].
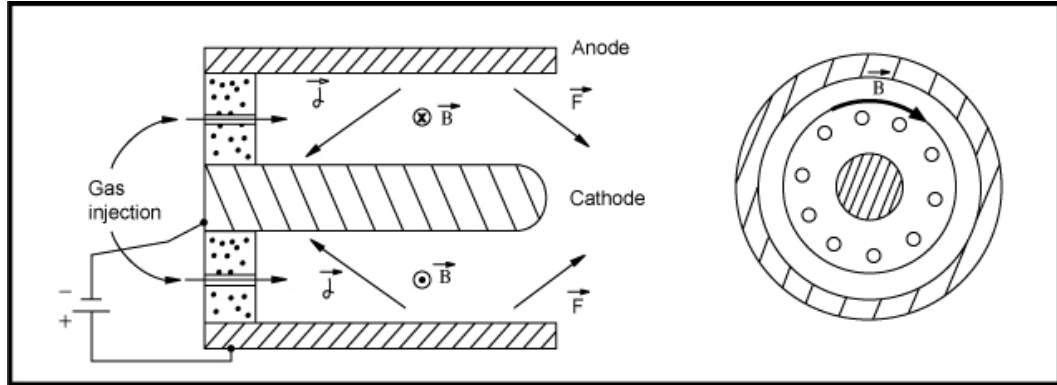
Figure 2.23: Magnetic Field, Current Density, and Force direction in the Plasma

[Lecture 22, Page 11 of Martinez-Sanchez, 2004a]

The plasma formed inside the thruster induces electron flow from the cathode to the anode because of the electric conductivity in the plasma.

Maecker's Law is a simple formula that is derived from the Lorentz force (magnetic force only) in equation 2.37 and the magnetic field in equation 2.85. The formula determines the force of the self-generated field electromagnetic coaxial thruster [Martinez-Sanchez, 2004a],

$$F_{Maecker} = \frac{\mu_0 I^2}{4\pi} \cdot \left(\ln\left(\frac{r_a}{r_c}\right) + C_{tip} \right),$$ (2.86)

where r_a and r_c are the radii of the anode and cathode of the coaxial thruster, respectively. The current I is the total current which comes across from the cathode to the anode through the plasma. The constant C_{tip} has a value between 0 and 0.75. Its value is determined by the examination of the integral form of Ampère's law, equation 2.84b [Jahn, 1964; Martinez-Sanchez, Space Propulsion, 2004a].
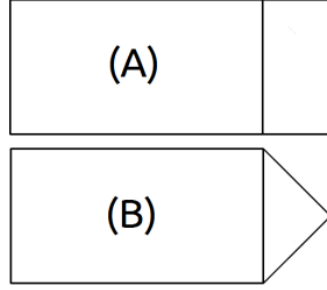
Figure 2.24: The Cathode Tip Configuration. Configuration A Yields a C_{tip} Value of Zero. Configuration B Yields a C_{tip} Value of 0.75.

Configuration B for the cathode tip can be easily manufactured using metals such as aluminum and steel. The configuration of the cathode tip is discussed in section 3.5.

In a quasi-steady state, the performance is measured in impulses. Impulse is the amount of force integrated over time, or the change in momentum. An impulse bit is the impulse per pulse. This is analyzed when the thruster is running in quasi-steady state,

$$I_{bit} = \int_0^t F_{Maecker}(t) \cdot dt = \int_0^t \left(\frac{\mu_0}{4\pi} \cdot \left(\ln\left(\frac{r_a}{r_c}\right) + C_{tip} \right) \cdot (I(t))^2 \right) dt, \tag{2.87}$$

where *I(t)* is the time-dependent current inside the plasma.

The exhaust velocity from the thruster reaches its peak velocity when the gas is fully ionized. A peak velocity is achieved when the kinetic energy is equal to the ionization energy of the particle. This velocity is called Alfvèn's critical velocity [Choueiri, 1998; Martinez-Sanchez, 2004],

$$v_{critical} = \sqrt{2 \cdot \frac{E_i}{m_i}}.$$ (2.88)

2.10 Langmuir Probe

A Langmuir probe measures the electron temperature, electron density, and current density of the plasma by measuring the current and voltage. In turn, these values determine the thrust of the ion thruster. The actual thrust coming out of the thruster is too small for noticeable movements in the thruster using an pendulum. A Langmuir probe is used in many ion propulsion experiments, and it is proven to be effective to indirectly measure the thrust, within 10 nN. Figure 2.25 shows the schematic of a triple Langmuir probe [Eckman *et al.*, 2001].
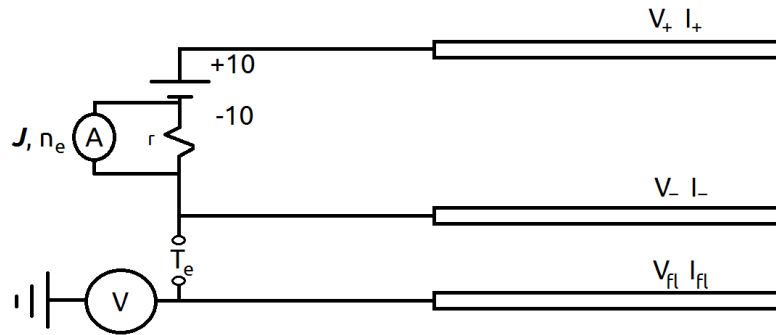
Figure 2.25: Triple Langmuir Probe Schematic

A triple Langmuir probe is a three-electrode probe that has all the electrodes having identical lengths and surface areas. This configuration is ideal for most electron temperature readings due to the simple setup. In a three-electrode probe, there are voltage biases on the two probes. The voltage biases are positive (V_+) and negative (V_-) of the given voltage from the power supply. The absolute values of the voltages must be higher

62

than the estimated electron temperature, typically above 10 V. The third electrode is floating (V_{fl}); once the ion beam passes through the electrodes, the floating electrode conducts, giving a voltage. Since the third electrode is floating, it filters out the RF interference from the plasma. The following equations use the thin-sheath current collecting model. This is assuming the thin-sheath is collisionless. The difference between the positive bias voltage and the floating voltage electrodes yields the electron temperature, in electron volts [Chen & Sekiguchi, 1965; Eckman *et al.*, 2001],

$$T_e(eV) = \frac{V_+ - V_{fl}}{\ln(2)} \; . \tag{2.89}$$

The Langmuir probe can also be used to measures the current of the plasma. An ammeter is placed between the positive and negative electrodes, parallel to a small resistance, *r*. The current, along with the dimensions of the electrodes and electron temperature, yields the current density and the ion density of the plasma [Eckman *et al.*, 2001],

$$J = \frac{I_{-+}}{A_{active}} \cdot \frac{1}{\exp\left(-\dfrac{V_{+-} - V_{+fl}}{T_e(eV)}\right) - 1} \; . \tag{2.90}$$

With the current density, the ion density is [Eckman *et al.*, 2001],

$$n_i = \frac{J \cdot \exp\left(\dfrac{1}{2}\right)}{e\sqrt{e \cdot \dfrac{T_e(eV)}{m_i}}} \; . \tag{2.91}$$

The thrust is measured by the results of the current density. It is then calculated using equation 2.82. Details of the construction of the triple Langmuir probe and how the thrust is calculated are discussed in Section 3.10.

2.11 Ingestion in Plasma Propulsion

When the thruster is running in a finite pressure environment, ingestion has to be taken into account. Ingestion is dependent on the ambient temperature and the mass of the ion [Goebel & Katz, 2008; Goebel, 2012],

$$\dot{V}_{ingested} = \frac{133.2}{4k_B T} \sqrt{\frac{8k_B T}{\pi m_i}} \cdot \frac{A_{active}}{4.479 \cdot 10^{23}} \cdot \tag{2.92}$$

The ingested flow rate is added to the total volumetric flow rate of the gas [Goebel & Katz, 2008; Goebel, 2012].

3 Methodology

A major constraint for this research is the development of a thruster which uses a maximum input power of 20 Watts. Sections 3.2 through 3.6 discuss the engineering design and development of the thruster. Section 3.7 discusses the simulation procedures. Sections 3.8 through 3.10 discuss the experimental procedures. The engineering section covers the nozzle design, ion optics, cathode-anode design, the RF electronics, and the DC discharge electronics. After the conditions are met, the systems are constructed with a bread-board electronic layout and encapsulated by a Faraday cage.

3.1 Design Criteria

Table 3.1: Design Criteria [Quine, 2010]

Criterion	Constraint
Maximum Power Usage	20 watts
Size of Spacecraft	1 kg – 10 kg
Survival Temperature Range	-40^0C $– +85^0$C
Vibration (Experimental Parameters)	15g Amplitude 20 Hz to 2000 Hz Random and Sinusoidal Frequency
Encapsulation Required	Yes
Minimum Specific Impulse (Deep Space)	500 seconds
Minimum Specific Impulse (Maneuvers)	100 seconds

Table 3.1 shows the target design criteria for the propulsion system. The maximum operating power of the propulsion system is 20 W; the ideal operating power of the propulsion system is 10 W. This system must produce enough thrust to have utility for manoeuvring a nanosatellite with a mass between 1 kg and 10 kg. A goal in the propulsion system is to have a minimum specific impulse of 500 seconds to ensure that the system is fuel efficient [Quine, 2010]. By comparison, miniature cold-gas thrusters and chemical thrusters have nominal specific impulses of approximately 65 seconds and 250 seconds, respectively [Sutton & Biblarz, 2010].

The operational temperature ranges between -40^0C and 85^0C, nominal for any nanosatellite. The typical structural vibration frequencies that must be avoided are between 60 Hz and 200 Hz [Quine, 2010]. The range of the vibration frequency shown in Table 3.1 is for mechanical vibration testing for failure analysis. Most standard electronics can withstand these temperature and vibration ranges. The mechanical parts, such as the nozzles and the electrodes, are simulated using CATIA$^{®}$ from Dassault Systems for the structural analyses [Anonymous, Dassault Systems, 2012].

3.2 Nozzle Design

The nozzle design is separated into two parts. The first part is the isentropic nozzle design. It takes into consideration the conical nozzle with the cone half-angle of 6 degrees. The second part is the structural analysis. This involves evaluating the mechanical properties of the nozzle. Two structural materials are used in the nozzle simulation analyses: aluminum and glass.

66

3.2.1 Nozzle Design – Fluid Dynamics

The exhaust-throat area ratio of the nozzle is important in the determination of the exhaust velocity and specific impulse. This is determined by equation 2.63 in section 2.7.1 and charted below for various values of γ [Sutton & Biblarz, 2010],
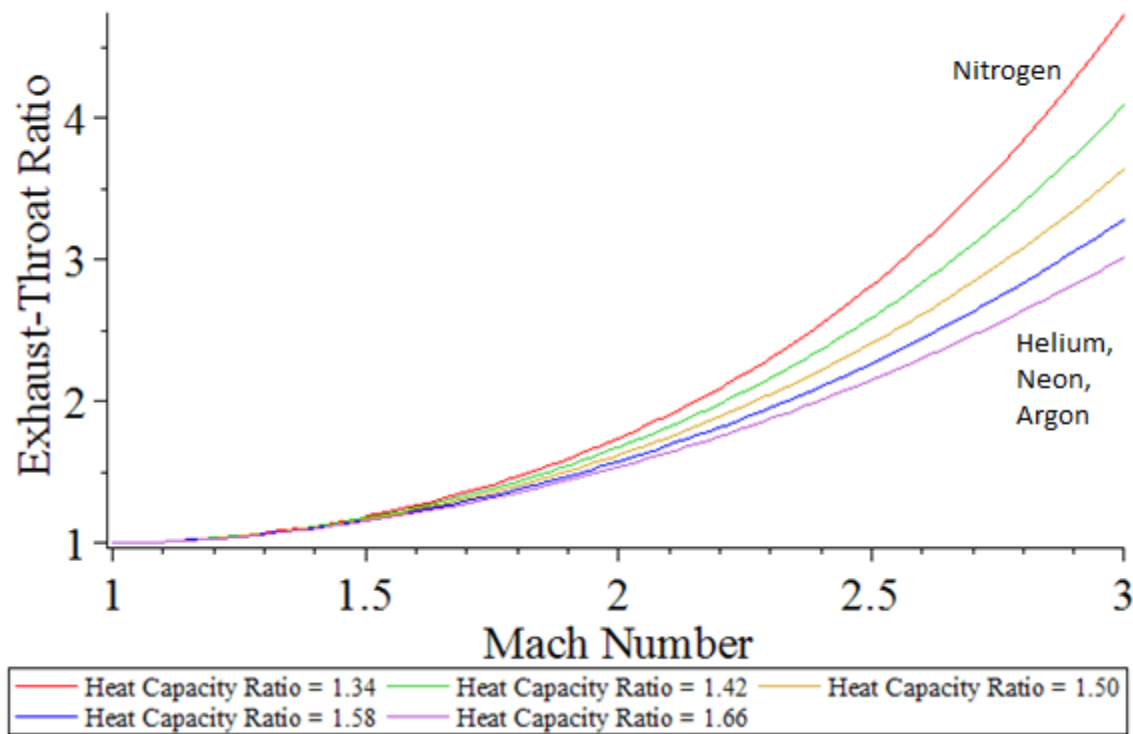
Figure 3.1: Exhaust-Throat Area Ratio vs. Exit Flow Mach Number

Figure 3.1 shows how the exhaust-throat area ratio depends on the heat capacity ratio. As the heat capacity ratio increases, the throat area required to obtain the same exit Mach number decreases. The heat capacity ratio, γ, is dependent on the type of gas used and the temperature of the gas. The Mach number increases significantly as the heat capacity ratio increases. Table 3.2 shows the specific impulse at different exhaust-throat area ratios,

assuming the use of nitrogen gas at an arbitrary inlet temperature of 500 K, which is typical for an electrothermal thruster.

Table 3.2: Exhaust-Throat Area and Specific Impulse Given a 500 K nitrogen gas

Exhaust-Throat Area Ratio	Specific Impulse (seconds)
4	82.723
8	88.800
16	92.904
32	95.795

The exhaust-throat area ratio chosen for the current study's nozzle is four, meeting thrust and I_{sp} requirements within the available volume of the nanosatellite.

As per equation 2.65, the exhaust-inlet pressure ratio is between 0.24 and 0.29 if the fluid just reaches sonic speed ($Ma_t = 1$). The ratio drops to 0.028 with the exit fluid speed just exceeding Mach 3. Figure 3.2 illustrates this behaviour.
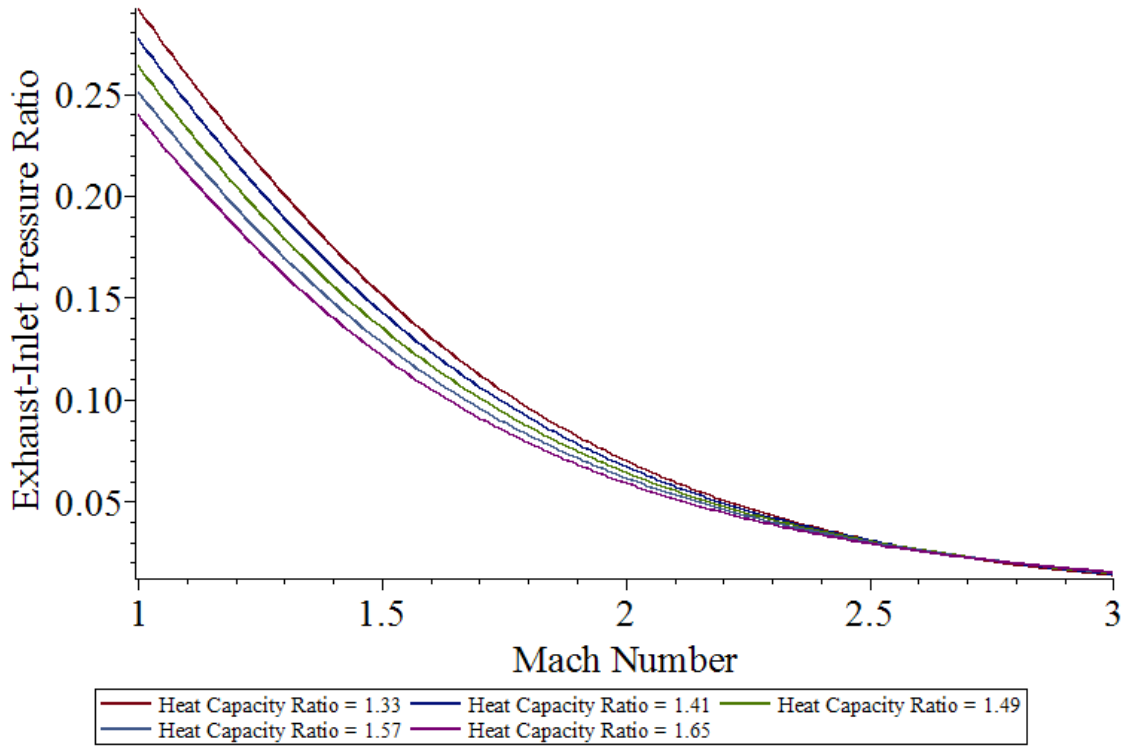
Figure 3.2: Exhaust-Inlet Pressure Ratio vs. Mach Number

The fluid speed depends on the inlet pressure of the gas. If the inlet pressure is too low in comparison to the exhaust pressure, the fluid does not go into supersonic speed. As the fluid moves faster than the speed of sound, the impact on the performance from the heat capacity ratio decreases.

The exhaust velocity depends on the heat capacity ratio of the fluid, and the heat capacity ratio depends on the type and temperature of the fluid as per equation 2.66. The specific heat ratio, γ, decreases as the temperature of the fluid increases. Figure 3.3 illustrates this behaviour.

69
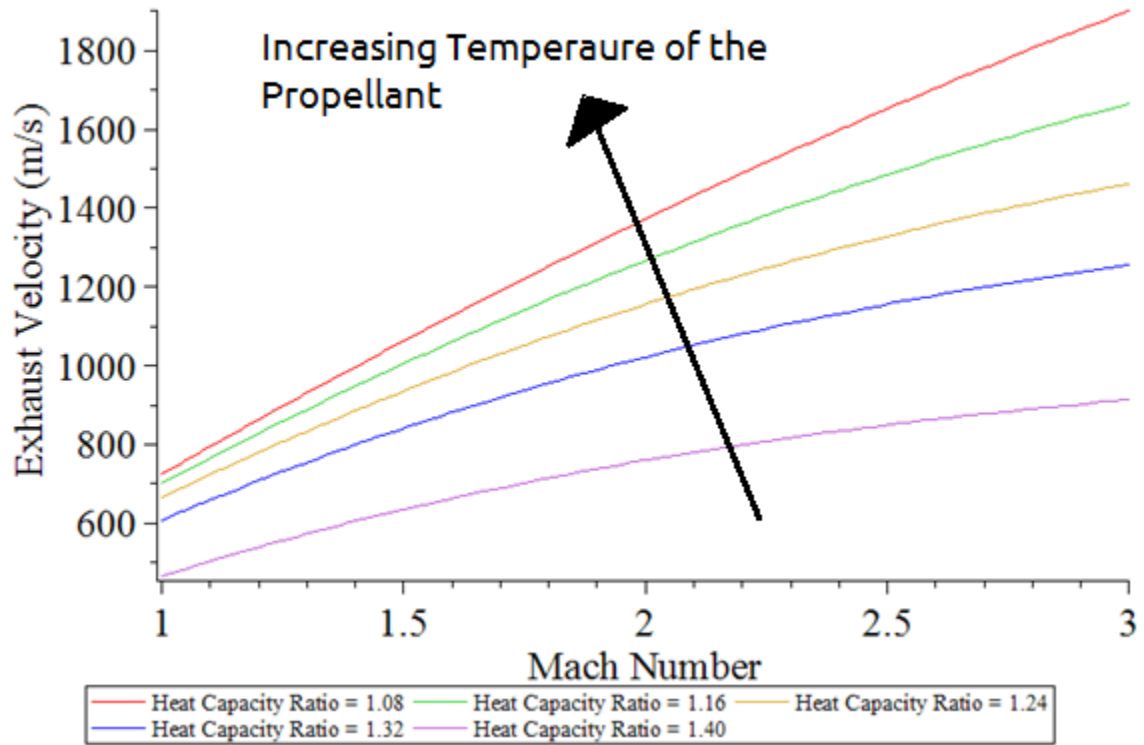
Figure 3.3: Exhaust Velocity vs. Mach Number

The exhaust velocity is dependent on the molecular mass, m, of the fluid. If a fluid with a lower molecular mass is used, the exhaust velocity increases as the square root of it, as shown in equation 2.66.

3.2.2 Nozzle Structural Analysis - Static

The static structural analysis focuses on the static stresses experienced on the structure from external forces. The general stress tensor in the right-hand Cartesian coordinates of any material is shown in equation 3.1 [Norton, 2000],

$$\sigma = \begin{bmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{bmatrix}.$$

$$(3.1)$$

The stress components along the diagonal, σ_{ii}, are the normal stresses. These stresses act normal to the surface of the material. The stress components not acting along the diagonal, $\tau_{ij}/i \neq j$, are the shear stresses. These components act tangential to the surface. The shear stresses are symmetrical, $\tau_{ij} = \tau_{ji}$. Figure 3.4 shows the stress tensor graphically.
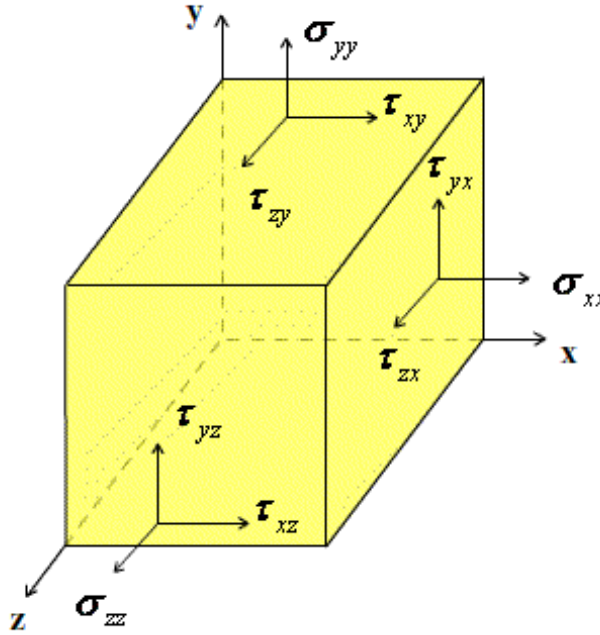
Figure 3.4: Stress Tensor on a Volumentric element[Mechanics of Fibre-Reinforced Composites, 2008]

The Von Mises stress, σ_V, is the equivalent stress which has the distortion energy equivalent to all the individual stresses combined. The Von Mises stress is a scalar quantity, defined as [Norton, 2000]:

71

$$\sigma_V = \sqrt{\frac{\left(\sigma_{xx} - \sigma_{yy}\right)^2 + \left(\sigma_{yy} - \sigma_{zz}\right)^2 + (\sigma_{zz} - \sigma_{xx})^2 + 6\left(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2\right)}{2}} \ . \tag{3.2}$$

One of the main requirements for the propulsion system is to withstand 15 g, or approximately 150 m/s², of acceleration [Quine, 2010]. During the research, two types of nozzles were manufactured, one made out of aluminum and one made out of glass. The aluminum nozzle is designed for the electrothermal thruster while the glass nozzle is designed for the hybrid electric thruster. Both of these nozzles are clamped to a rigid surface, such as an acrylic support. Two types of meshing are used in the structural loading simulations: coarse and fine of lengths 5 mm and 2.5 mm, respectively. The linear solid elements are used in finite element analysis. Figure set 3.5 shows the locations where vibration load is applied in simulation, to produce a bending of the nozzle structure. Note that the load is distributed across the nozzle; these are not point loads.
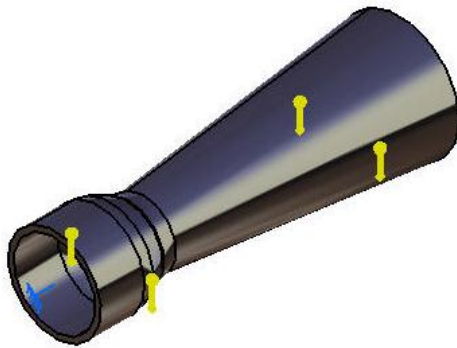
Figure 3.5a: Aluminum Nozzle with Loading Conditions
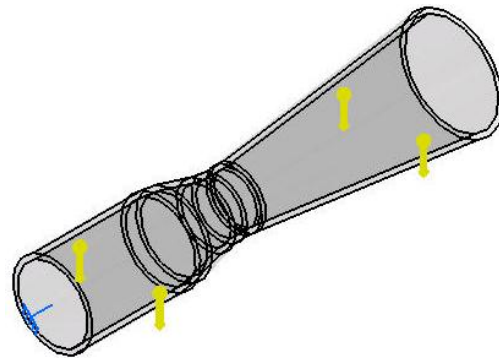
Figure 3.5b: Glass Nozzle with Loading Conditions

Table 3.3 shows the dimensions and material properties of the aluminum and glass nozzles; the nomenclature is defined in figure 2.17.

Table 3.3: Dimensions and Material Properties of the Aluminum and Glass Nozzles

Properties	Aluminum	Glass
Exhaust Diameter (mm)	20	
Throat Diameter (mm)	10	
Inlet Diameter (mm)	15	
Thickness (mm)	1	
Length of Cone (mm)	46	
Half-Angle (degrees)	6	
Total Length (mm)	58	86
Material Type	Aluminum 6061	Silica Glass
Young's Modulus (GPa)	68.9	68.0
Ultimate Strength (MPa)	117	33
Poisson's Ratio	0.33	0.19

The full mechanical drawings of the nozzles, along with the rest of the components, are in appendix F. The data obtained from the static (structure not undergoing any acceleration) simulation on the aluminum nozzle using coarse meshing is shown in figure 3.6.

Von Mises stress (nodal values)
kNm²
218
197
175
154
132
111
89
68
46
25
3
On Boundary
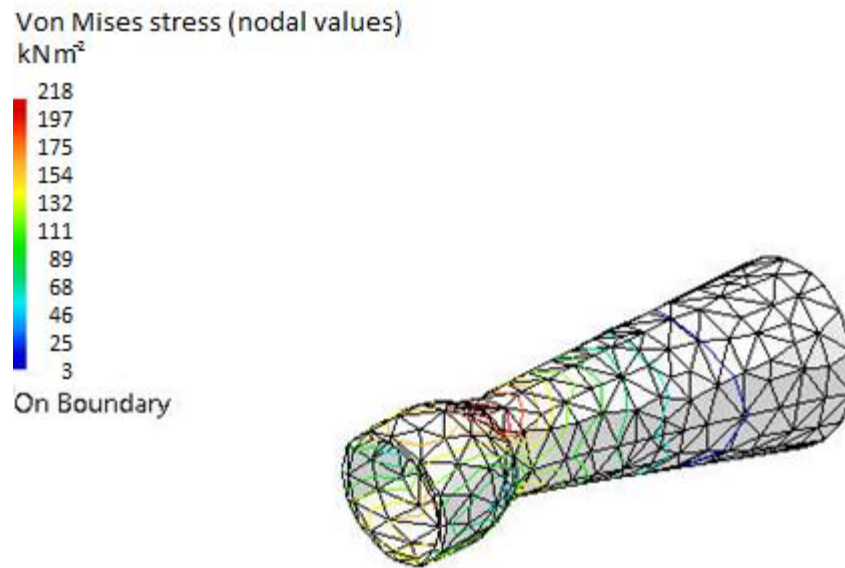
Figure 3.6: Results of the Static Loading Simulation on the Aluminum Nozzle using

Coarse Meshing.

The largest Von Mises stress in the aluminum nozzle under static conditions is 218 kPa

located at the throat portion of the nozzle. The ultimate tensile strength of aluminum is

between 40 and 50 MPa. The safety factor is the ratio of the ultimate strength to Von

Mises stress [Norton, 2000]. The safety factor in this model is 196. Therefore, it is safe to

assume that the aluminum nozzle can withstand 15 g acceleration.

Figure 3.7 shows the results of the static simulation on the aluminum nozzle using
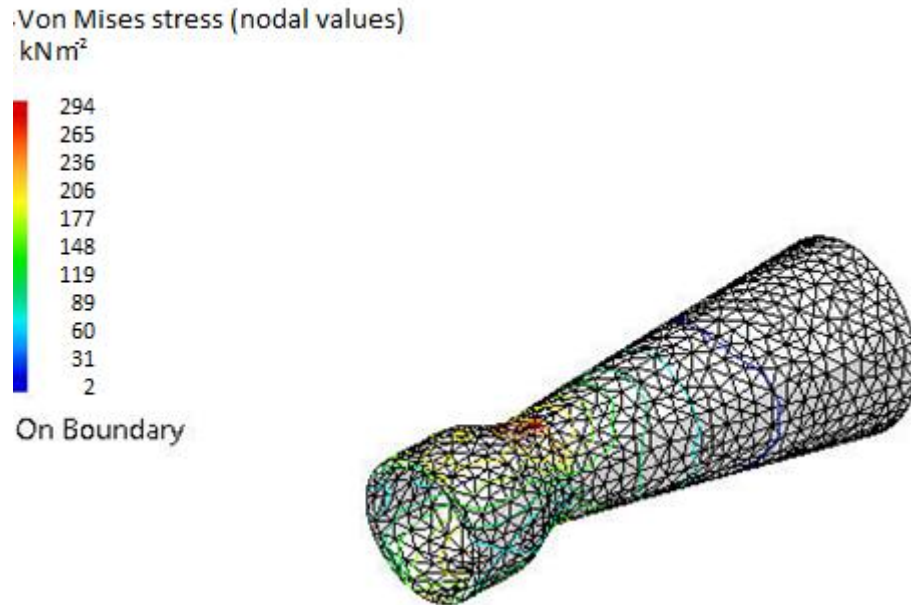
fine meshing.

Figure 3.7: Results of the Static Simulation on the Aluminum Nozzle Using Fine Meshing.

In the fine mesh model, the highest Von Mises stress is 294 kPa at the throat, which converges to the results of the coarse mesh model. The safety factor in this model is 189. The fine mesh verifies that the simulation is accurate for the aluminum model.

The simulation result of the glass nozzle using coarse meshing is shown in figure 3.8. The largest Von Mises stress is 273 kPa located at the inlet and throat portion of the nozzle. The ultimate strength for glass is lower than that of aluminum; it is around 33 MPa.
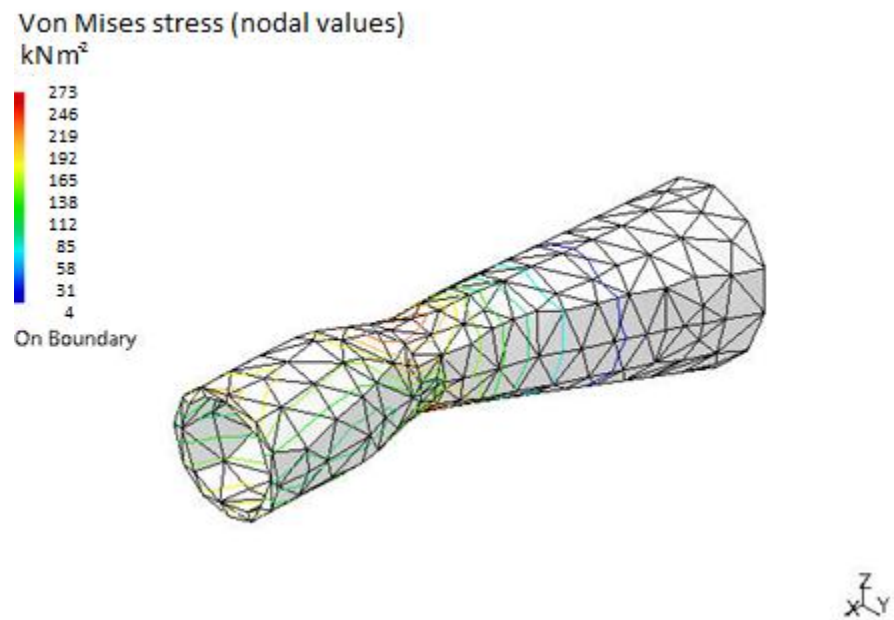
Figure 3.8: Results of the Static Loading Simulation on the Glass Nozzle Using Coarse Meshing.
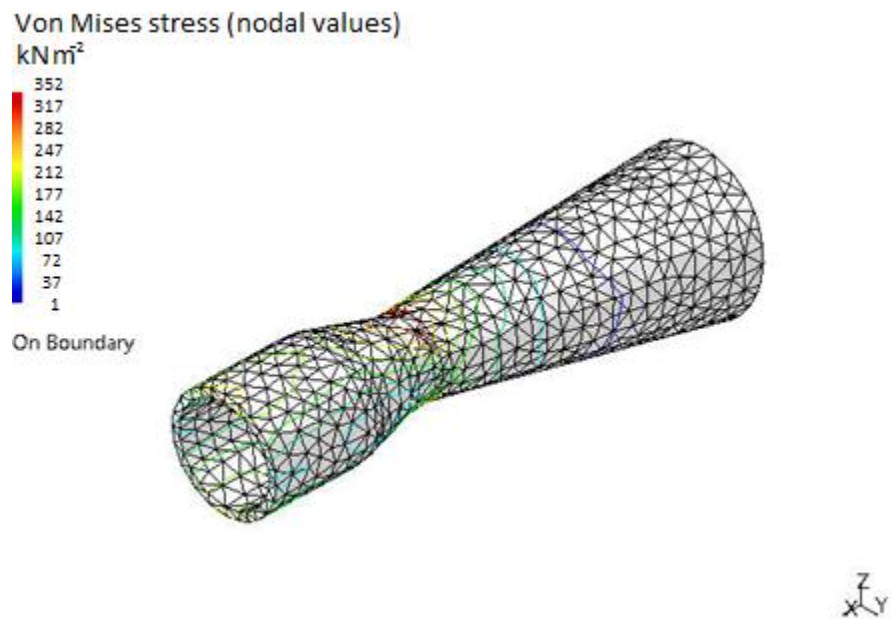
Figure 3.9: Results of the Static Loading Simulation on the Glass Nozzle Using Fine

Meshing.

76

Figure 3.9 shows the stress analysis results of the glass nozzle using fine meshing. In the fine meshing case, the largest Von Mises stress is 352 kPa. It is higher than the coarse mesh counterpart. However, the Von Mises stress in the static condition is still substantially lower than the ultimate strength of glass.

3.2.3 Nozzle Structural Analysis - Dynamic

A dynamic structural analysis was conducted to determine the natural frequencies of the structure. Finding the natural frequencies in a structure is important because the structure can fracture when the input frequency matches its natural frequency. A mechanical vibration model describes the nodal form of the structure. Equation 3.3 describes a single degree of freedom (SDOF) representation of a structure [Norton, 2000],

$$m\ddot{x} + b\dot{x} + kx = F , \qquad (3.3)$$

where m, b, k, and F are the mass, damping coefficient, stiffness, and input force respectively. One key element in mechanical vibration for the dynamic analysis wa the natural frequency, ω_n [Norton, 2000],

$$\omega_n = \sqrt{\frac{k}{m}} . \qquad (3.4)$$

A multiple degree of freedom (MDOF) equation model has the mass, damping, and stiffness values in $n \times n$ matrices and the force becomes an $n \times 1$ vector. The material damping is not required in order to find the natural frequency. Equation 3.5 shows the resulting matrix form [Norton, 2000],

$$\mathbf{M}\ddot{x} + \mathbf{K}x = \mathbf{F}. \tag{3.5}$$

Solving this system gives the values of the nodal positions, x. The natural frequencies are obtained by finding the eigenvalues of $\mathbf{K} - \omega^2\mathbf{M}$ [Norton, 2000; Damaren, 2010],

$$|\mathbf{K} - \omega^2\mathbf{M}| = \mathbf{0}. \tag{3.6}$$

Previously, figure set 3.5 showed the load conditions applied to the aluminum and glass nozzles in simulation. The same conditions are applied for the frequency analysis to find the natural frequencies of the structure. The simulations are performed using both coarse and fine meshing. Figure 3.10 shows the first modal frequency response meshing for the aluminum nozzle.
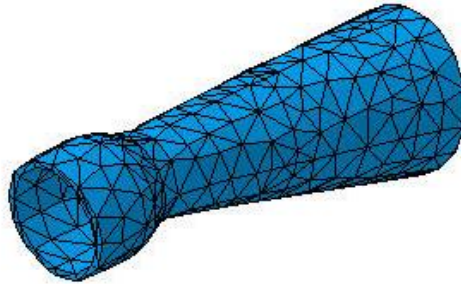
Figure 3.10: First Mode of the Aluminum Nozzle Using Coarse Meshing

The result shows that the first natural frequency of the nozzle is 1953 Hz, 2.4% below the upper limit of the experimental vibration frequency. The fine mesh model gives a fundamental frequency of 1824 Hz, 6.6% lower than its coarse mesh counterpart. This fine mesh analysis indicates that the results are not affected significantly by mesh size. Averaging the modal participation factor from all six degrees of freedom, it is above 80%. Therefore, displaying the first natural frequency in this model is valid. The total strain

78

energy, the potential energy from the static deflection, of the nozzle is 544 nJ, and the corresponding peak deflection from a vibration of 1824 Hz is 0.86 μm [Norton, 2000]. This analysis indicates that structural failure would occur at an acceleration load exceeding 500 g.

Figure 3.11 shows the first modal frequency response meshing for the glass nozzle.
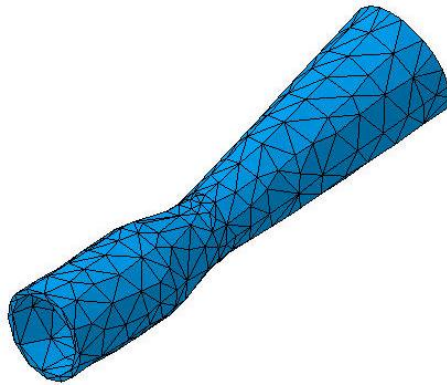
Figure 3.11: First Mode of the Glass Nozzle Using Coarse Meshing

The first natural frequency of the glass nozzle is 2075 Hz, 3.8% above the upper frequency limit constraint. The fine mesh model shows the fundamental frequency to be 1945 Hz, 6.3% lower than its coarse mesh counterpart. Averaging the modal participation factor from all six degrees of freedom, it is above 80%; thus, it is justify to use the first natural frequency. The strain energy in the glass nozzle is 386 nJ, and the corresponding deflection of this vibration is 0.75 μm. This analysis indicates that structural failure would occur at acceleration load exceeding 250 g.

3.3 RF Electronics Design

RF plasma generation is selected because of its low-power requirement for ionization of a propellant medium. The RF circuit used for the heating is a standard Clapp oscillator. To minimize parasitic inductance and capacitance in the circuit, the leads lengths are minimized; the circuit is placed on an aluminum block. The transistor used in the Clapp oscillator circuit is a BLF245 n-channel enhanced MOSFET [Anonymous, Phillips Semiconductors, 2003]. This high power MOSFET has the frequency output rating of 175 MHz, and its maximum voltage from the drain to source is 60 V. Figure 3.12 shows the Clapp oscillator circuit [Horowitz, 2001; Brown, 2010].
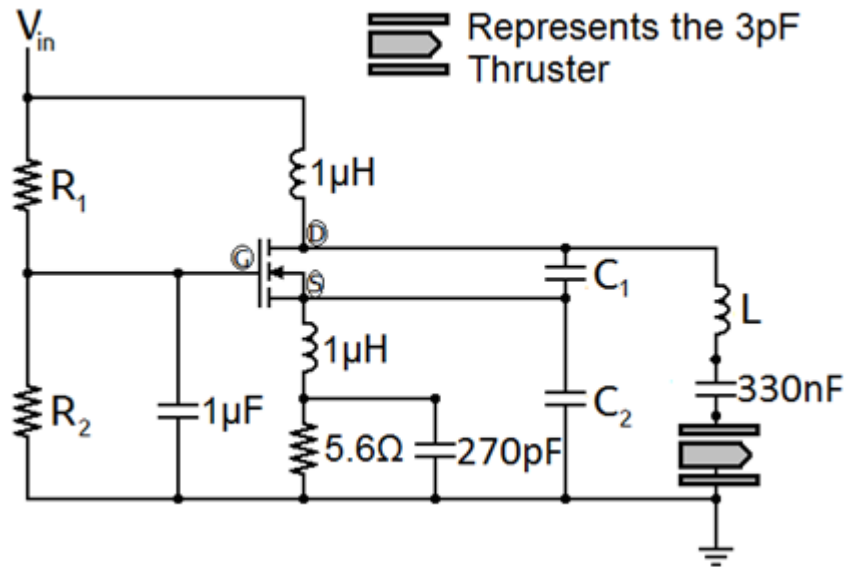
Figure 3.12: Clapp Oscillator used for Plasma Generation

There is a 270 pF capacitor that is in parallel with the 5.6 Ω resistor to by-pass the AC [Brown, 2010]. It prevents the loss of AC power across that resistor. The 330 nF capacitor

before the electrothermal thruster blocks the voltage surging which would go back to the circuit. The values R_1, R_2, C_1, C_2, L and V_{in} are the resistances, capacitances, the inductance, and the input DC voltage of the circuit. The optimal values for the resistors, capacitors and the inductor are shown in table 3.4.

Table 3.4: Values of the Resistors, Capacitors and the Inductor

Component	Value
R_1 Resistance	6.8 kΩ
R_2 Resistance	10 kΩ
C_1 Capacitance	20 pF
C_2 Capacitance	47 pF
L Inductance	330 nH

This circuit has output fundamental frequencies between 78 MHz and 112 MHz. The range of the DC voltage power supply used in the RF circuitry is between 10 V and 15 V; the corresponding current is between 0.25 A and 0.4 A. This particular voltage range fits the maximum power requirement of 20 W required in the development of the thruster.

3.4 Ion Thruster Design

For deep-space missions, the propulsion system must have a high specific impulse. Since the minimum design specific impulse for deep-space missions is set at 500 seconds, the electrostatic propulsion mode is ideal for this situation.

The neutralizer in the ion thruster has been omitted because the performance in testing is expected to be about 1% higher without the use of a neutralizer. The main concern in designing the neutralizer is that the electron beam must have the same current density as the ion beam [Martinez-Sanchez, 2004a]. In order to ensure this balance, the neutralizer would have the same design as the ion thruster, with the exception that the voltages of the electrodes would be reversed. This would be similar to an electron emitter from a cathode-ray tube television.

3.4.1 Ion Optics Design

The ion optics design for the electrostatic propulsion has a few constraints. There is a cost limitation in this research. The maximum allocated funding for this research was $10,000. This cost includes manufacturing from the machine shop, the materials, and additional human resources. The manufacturing and material limitations include the size of the holes in the screen and acceleration electrodes, and the durability of the material. The electrodes have a thickness of less than 1 mm; ideally, the higher the number of holes in the electrodes, the higher the thrust and specific impulse. The number of holes in the screen and acceleration electrodes is seven, as a reasonable compromise between performance and cost. The corresponding sizes of the holes in the screen and acceleration electrodes are 3 mm and 2.5 mm, respectively. The thicknesses of the electrodes are 0.5 mm and 1 mm, respectively. Finally, the separation between the electrodes is 0.5 mm.

The maximum current density is dependent on the total voltage of the electrodes and the sheath thickness. Figure 3.13 shows the variation of current density with respect

to the input total voltage and the ion velocity versus the beam voltage, for various molecular masses.

The total and net voltage constraints are based on the available equipment and the possibility of eliminating electron backstreaming. The maximum ratio of the total to net voltage is 0.814 according to electron backstreaming equation 2.83 [Martinez-Sanchez, 2004a].
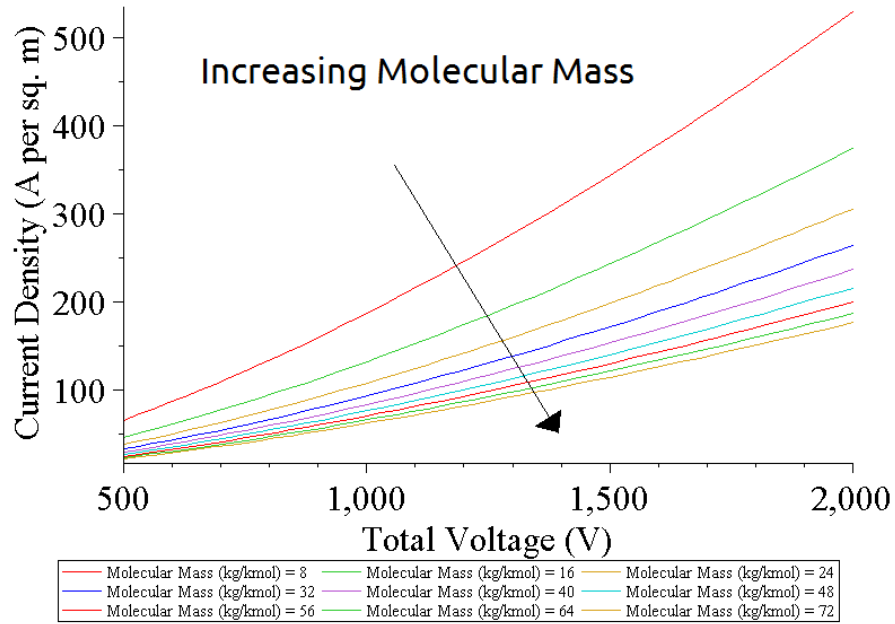
Figure 3.13a: Current Density vs. Total Voltage for Various Molecular Masses
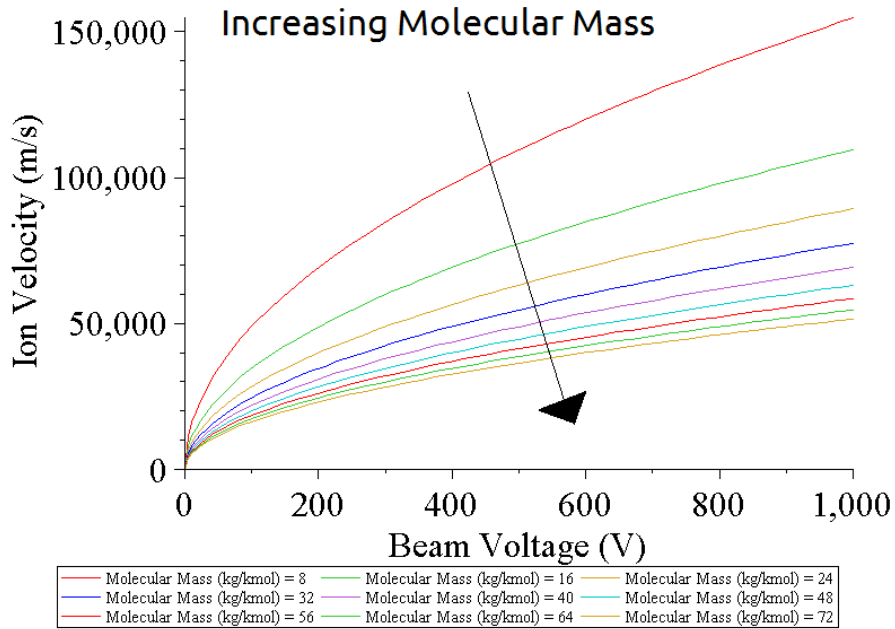
Figure 3.13b: Ion Velocity vs. Beam Voltage for Various Molecular Masses

The voltages of the screen and acceleration electrodes were designed to be 1000 V and -200 V, respectively. However, the power voltage supply from the laboratory offered a more limited range; the voltages of the screen and acceleration electrodes were set to 600 V and -60 V, respectively. The voltage of the screen electrode is varied between 200 V and 600 V to observe the effects of the beam voltage on the performance of the thruster. If a neutralizer were to be used, the charged grids would have to be 200 V and -1000 V respectively in order to match the current density of the ion beam.

Using equation 2.82, the corresponding force per aperture of the ion beam computed is 1.214 μN. With seven apertures, the total force of the ion beam is 8.5 μN. The ion beam velocity is approximately 82.7 km·s^{-1}. Thus, the theoretical specific impulse of the thruster using nitrogen is 8430 seconds, using a 0.14 μg·s^{-1} mass flow rate.

Inefficiencies in the system may include plasma generation, the efficiency of the power supplies, the manufacturing of the thruster, and the input mass flow rate.

3.4.2 Structural Analysis

For this analysis, the screen electrode is clamped at both outer circular edges while the acceleration electrode is clamped at one outer circular edge. The pressure on both electrodes is 500 Pa. The constraints of the screen and acceleration electrodes are shown in figure 3.14.
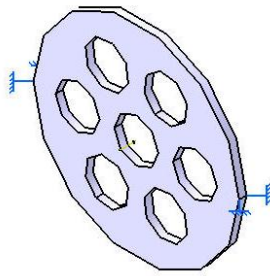
Figure 3.14a: Screen Electrode with Loading Conditions
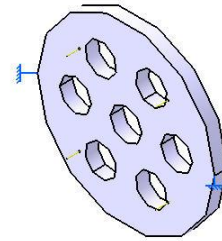
Figure 3.14b: Acceleration Electrode with Loading Conditions

The results of both analyses are shown in figure set 3.15.
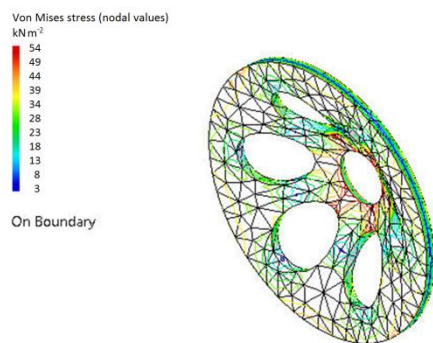
Figure 3.15a: Results of the Static Loading Simulation on the Screen Electrode.
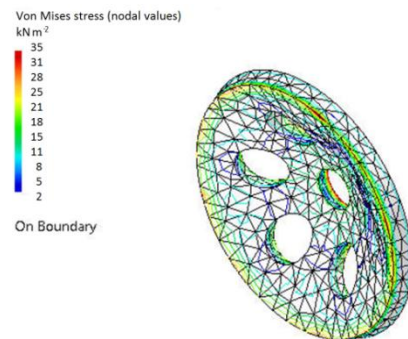
Figure 3.15b: Results of the Static Loading Simulation on the Acceleration Electrode.

Both of the results show that the location of highest stress is on the central apertures of the respective components. The Von Mises stresses for the screen and acceleration electrodes are 54.0 kPa and 34.7 kPa respectively. The ultimate tensile strength for brass is 550 MPa; therefore, the electrodes will not be damaged. The corresponding deflection displacements for the screen and acceleration electrodes are 11.00 μm and 3.65 μm, respectively. Both of these deflections occur at the centre of the electrodes but should not hinder the performance of the thruster. The natural frequencies for both the screen and acceleration electrodes are 42.7 kHz and 49.8 kHz, respectively. The corresponding strain energies are 33.8 pJ and 61.1 pJ, respectively. Since the natural frequencies are over 40 kHz above the upper frequency limit (~2 kHz), there would not be any risk of mechanical failure.

3.5 Electromagnetic Thruster Design

The electromagnetic (EM) thruster design is simpler than the previous designs since it involves only a cathode-anode mechanical design and a circuit design. A structural analysis of the EM components is not necessary because they are the least fragile components of the hybrid electric thruster.

3.5.1 Cathode-Anode Design

The radius of the anode is selected to be 7.3 mm because the anode needs to be inserted into the glass nozzle. The corresponding radius of the cathode is 4 mm with a tip. The tip increases the electromagnetic force, as per equation 2.86 [Jahn, 1968; Martinez-Sanchez, 2004a]. The constant C_{tip} is 0.75 because the cathode has a sharp end

(configuration B in figure 2.24). The relationship between the electromagnetic force and the induced current, *I*, is shown in figure 3.16. The force increases as the square of the induced current. Therefore for more thrust, it is recommended to have a high induced current. As most power supplies cannot provide a high current, an alternative approach is to use capacitors, and run the thruster in quasi-steady state. Quasi-steady state is a state where the thruster operates in pulses at 1 μs intervals.
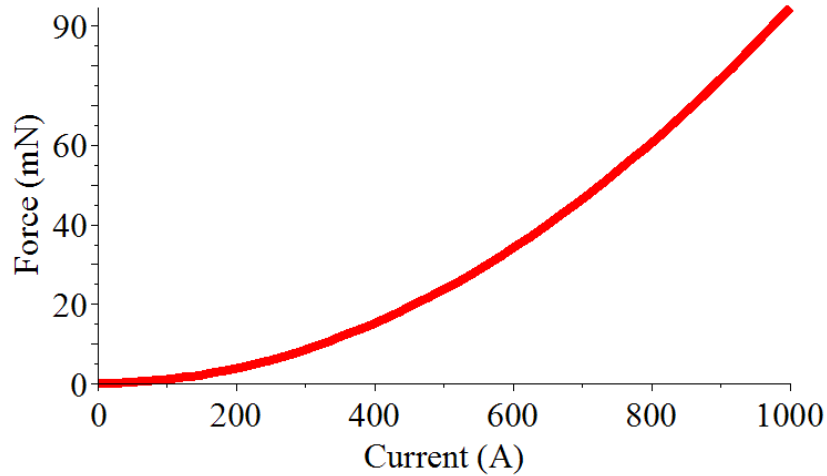
Figure 3.16: Electromagnetic Force vs. Induced Current

3.5.2 Circuit Design

The capacitors in the EM-mode thruster operate the thruster in quasi-steady state. The electronic configuration of the thruster is shown in figure 3.17 [Brown, 2012].
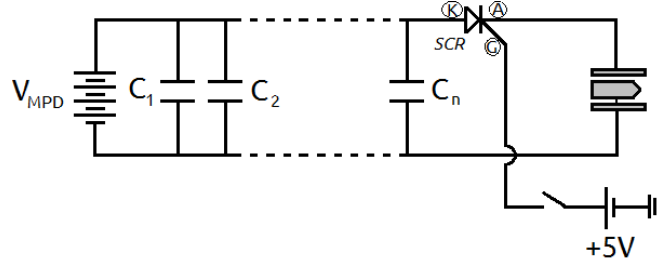
Figure 3.17: Electronic Configuration of the EM-mode thruster

The voltage supply for the EM, V_{MPD}, charges the capacitors, C_1, ..., C_n, and produces an arc in the thruster. The silicon-controlled rectifier (SCR) restricts the current when the gate (G) is off; when the gate is on, the current flows from the cathode (K) to the anode (A) of the SCR. Therefore, the capacitors discharge and deliver a high current to the EM thruster [Brown, 2012].

The model for the time-dependent of the total charge in the system is [Jahn, 1968]

$$L_{ind} \cdot \frac{d^2 Q_{charge}(t)}{dt^2} + R_{res} \cdot \frac{d Q_{charge}(t)}{dt} + \frac{1}{C_{cap}} \cdot Q_{charge}(t) = V_{MPD}, \qquad (3.7)$$

where $Q_{charge}(t)$ is the total charge in the capacitors at time t, L_{ind} is the internal inductance, R_{res} is the resistance, and C_{cap} is the capacitance. Normally, the internal resistance and inductance in capacitors are very low. There are 33 capacitors in the circuit; each of them has a capacitance of 100 nF and a rated voltage that of 1000 V. The minimum voltage to produce a plasma arc between two surfaces separated by 3.3 mm with the gas pressure of one Torr is 400 V. In the mathematical model, it is assume that the internal inductance and resistance are to be 100 nH and 1 Ω, respectively. The predicted result for the current versus time is shown in figure 3.18.
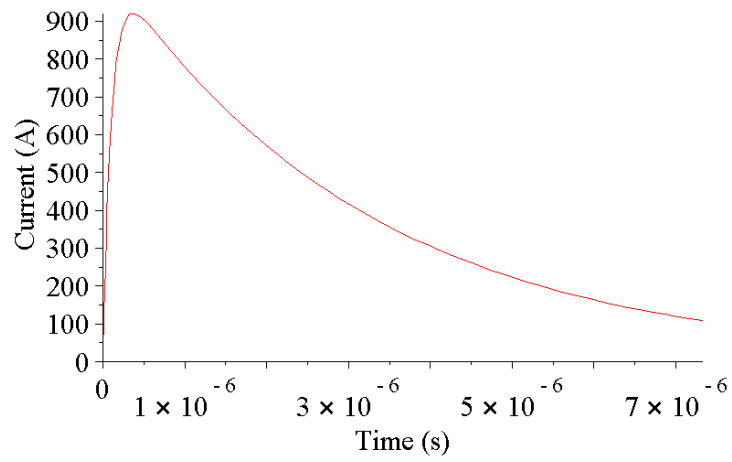
Figure 3.18: Current vs. Time Resulting From Equation 3.7

The maximum induced current from the capacitors is 920 A; this yields the maximum propulsive force of 80.15 mN. Integrating the force over time yields the overall impulse bit at 154.8 nN·s. The switch from figure 3.17 triggers the gate of the SCR.

3.6 Propulsion Schematics

3.6.1 Electrothermal Thruster Electronic Schematic
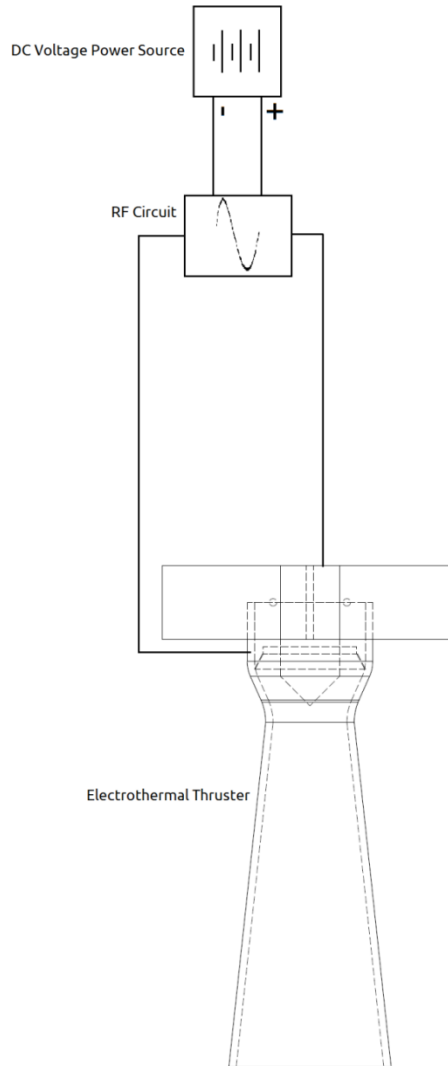
Figure 3.19: Electrothermal Thruster Electronic Schematic

The electronic schematic of the electrothermal thruster consists of a DC voltage power source, RF circuit and the electrothermal thruster itself. The RF circuit is used to heat and ionize the gas between the two electrodes. This system has been tested during

the research for a proof-of-concept design, and to experimentally investigate different gas mixtures. The results for the electrothermal thruster are discussed in chapter four. The mass of the thruster, the support, and the RF circuit is 0.11 kg.

3.6.2 Hybrid Electric Thruster Electronic Schematic

The electronics of the hybrid electric thruster in ion (ES) propulsion mode is comprised of the power source entering the RF circuit. The RF circuit ionizes the gas between the electrodes. Also, the RF circuit powers the neutralizer, when used. The power source charges the grids at the same time. There are DC-to-DC converters at the grids to increase the voltage from the power source to +1000 V and -200 V to the screen and acceleration grids, respectively.

The electronics of the MPD (EM) propulsion mode are comprised of the power source being converted to a voltage of 1000 V from the DC-to-DC converter. The power source charges the MPD capacitor banks, and the charged capacitor banks are isolated by the SCR. Once the SCR is activated, the capacitors are discharged and the current goes to the electrodes. In the RF circuit, there is a large capacitor which blocks the DC voltage of up to 1600 V; therefore, the RF circuit is protected. In the hybrid electric thruster, there is a 12-turn solenoid at the throat. The solenoid reheats the plasma to improve the overall performance. The mass of the thruster, the support, and the electronic components is 0.61 kg.

This propulsion system is a hybrid electric thruster utilizing RF ion (ES) and magnetoplasmadynamic (EM) propulsion modes. The operations of the two propulsion modes have been explained earlier in Sections 2.8 and 2.9.
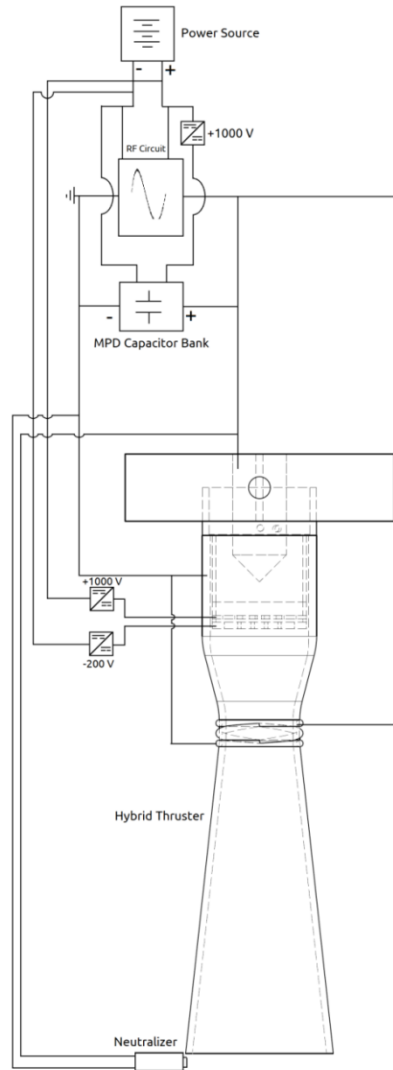
Figure 3.20: Hybrid Electric Thruster Electronic Schematic

3.6.3 Encapsulation

The propulsion systems emit high frequency radio waves when plasma is formed, thus having the potential to interfere with the other equipment in the spacecraft. In order to counter this problem, the RF generator and the capacitor banks are encapsulated with a Faraday cage to prevent RF wave propagation. The nozzle and tip of the thruster still emit RF radiation. However, the encapsulation should substantially reduce interference from the equipment.

3.7 Simulation

Flow simulations were conducted in order to calculate the fluid velocities in the thruster. The simulations are based on equation 2.52. [Sankaran *et al.,* 2002; Bar-Meir, 2011].

The plasma generation is calculated by using the plasma production equations 2.53a-d. Solving for the plasma generation equations in the time domain simplifies the calculations and demonstrates the rate at which the electrons are produced. This portion of the code was completely written by the author of this dissertation (Stoute). The graphs of the electron temperatures and densities are provided in appendix B. The program is given a set of initial values for electron temperature (eV), inlet pressure (Pa), voltage amplitude (V) and the input frequency (Hz). The initial electron temperature for all gases is set to 1 eV. The initial value of the pressure is set to 1 Torr (133 Pa). The voltage amplitude and input frequency depends on the propulsion system itself. For the RF electrothermal thruster and ion thruster modes, the voltage amplitude and frequency

are 15 V and 100 MHz, respectively. The voltage of the MPD thruster mode is set to 1000 V DC. In order to verify that the Navier-Stokes equations can be used in the simulations, the Knudsen number must be calculated. The Knudsen number is

$$Kn = \frac{\lambda_{mfp}}{L_{char}} = \frac{1}{n \cdot \sigma_{cross\ section}} \cdot \frac{1}{L_{char}} \approx \frac{k_B T}{10^{-19} p} \cdot \frac{1}{2 \cdot r_e}, \tag{3.8}$$

The characteristic length L_{char} is the diameter of the exhaust of the nozzle. The Knudsen number calculated under the inlet pressure and temperature conditions is 0.007. This means the fluid is in a continuum; therefore, the simulation can use the Navier-Stokes equations.

Figure 3.21 illustrates the algorithmic approach for modelling of the plasma production. Once the plasma temperature is calculated, equation 2.52 is calculated using the program which has been modified from the Navier-Stokes steady-state numerical computational fluid dynamics (CFD) model by Britton Olson of Stanford University [Olson, 2009]. The program, named the Olson-Stoute plasma propulsion simulator, solves the plasma fluid equations using the MacCormack method, described in Section 2.6 [Wendt, 2009]. The Olson-Stoute plasma propulsion simulator adds the full exhaust section of the nozzle, reformed the nozzle from a bell-shape to a conical shape, and adds the plasma fluid equations. Appendix H shows the full program.
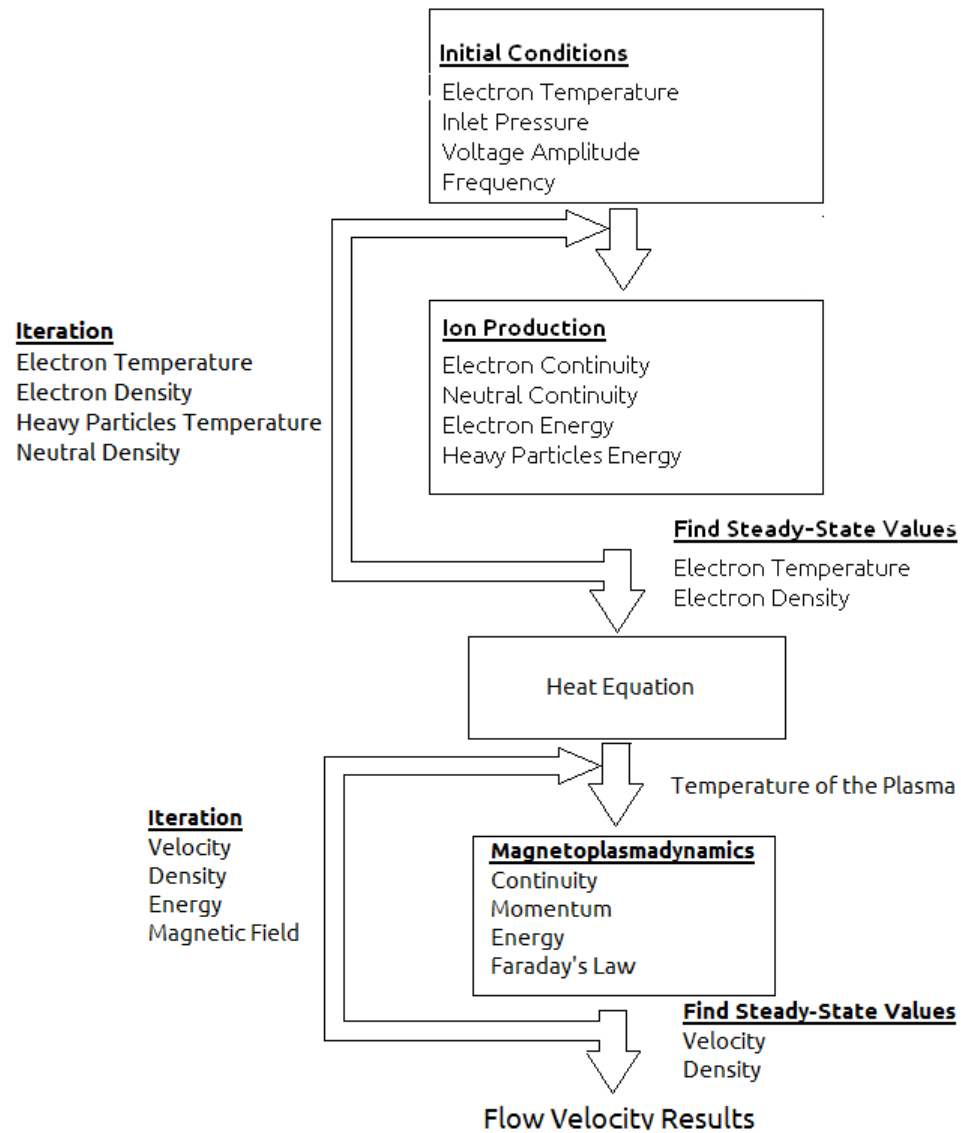
Figure 3.21: Flow Chart of Plasma Production and Fluid Dynamics in The Simulation

The program simulates the radio-frequency electrothermal thruster (RFET), and the hybrid electric thruster. The gases used in the RFET simulation are:

- Helium
- Neon
- Nitrogen
- Argon

- 10% Nitrogen – 90% Helium ($0.1N_2 - 0.9He$)
- 20% Nitrogen – 80% Helium ($0.2N_2 - 0.8He$)
- 50% Nitrogen – 50% Helium ($0.5N_2 - 0.5He$)
- 80% Nitrogen – 20% Helium ($0.8N_2 - 0.2He$)

Since the ET nozzle is made of aluminum, the wall temperature must be kept at or below 500 K. This temperature compensates for the thermal conductivity of the nozzle due to the plasma and the capacitive heating from the RF circuit. The input voltage and current of the RFET are 15 V and 0.4 A, respectively. The RF curcit has very little effect on the behaviour of the plasma other than heating. The gas mixtures used in the hybrid electric thruster simulation are:

- Helium
- Nitrogen
- Argon
- 50% Nitrogen – 50% Helium ($0.50N_2 - 0.50He$)
- 50% Argon – 50% Helium ($0.50Ar - 0.50He$)

The flow simulation maps the entire nozzle flow from the entrance to the exhaust. The heating element at the throat of the nozzle experiences the electromagnetic effects of the plasma. The nozzle of the hybrid electric thruster is made of glass, so only the plasma is affected by the electromagnetic field. The numerical CFD results are compared to the one-dimensional results calculated from this chapter to show the inaccuracies in the one-dimensional model. They are shown in section 4.1.

3.8 Electrothermal Thruster Experiment

The electrothermal thruster experiment is comprised of two setups: medium-vacuum chamber with a laser, and high-vacuum chamber with a light-emitting diode (LED). Initial experiments use the medium-vacuum chamber with a laser to measure the thrust. However, to improve the accuracy of the measurements, a LED and a photodiode are used in the high-vacuum chamber. The high-vacuum chamber is larger than the medium-vacuum chamber, so the LED and photodiode could be inside the chamber.

The medium vacuum chamber setup is shown in figure 3.22.
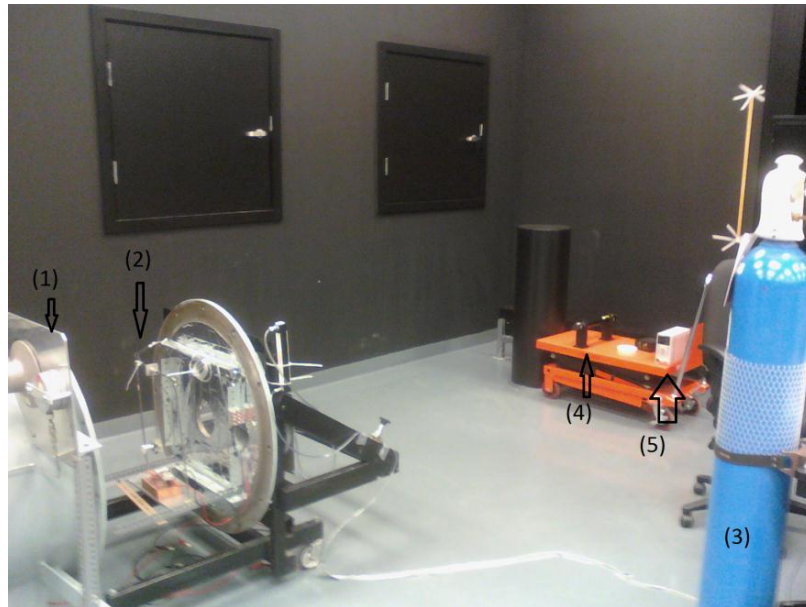
Figure 3.22: Experiment Setup for the Electrothermal Thruster Experiment. (1) Thermal Vacuum Chamber [TVAC] (2) Plasma Thruster and Its Structure (3) Compressed Nitrogen Gas (4) Green HeNe Laser (5) DC Power Supply for the RF Circuit

The pressure inside the thermal-vacuum chamber, or TVAC, for this experiment is 1 Torr (133 Pa). The reason for the relatively high pressure is due to the limitations of the TVAC chamber pumping performance. The thruster is attached to a simple pendulum without a counterweight. The laser, which acts as a guide to determine the thrust, is sensitive enough to detect the motion of the pendulum; the laser reflects off a mirror attached to the thruster. The reflected beam is then projected on the wall. A ruler is used to measure the beam deflection. A camera with a built-in UV filter is placed on the outside of the TVAC chamber, close to the glass opening in order to take photos of the thruster. Figure 3.23 illustrates this setup.
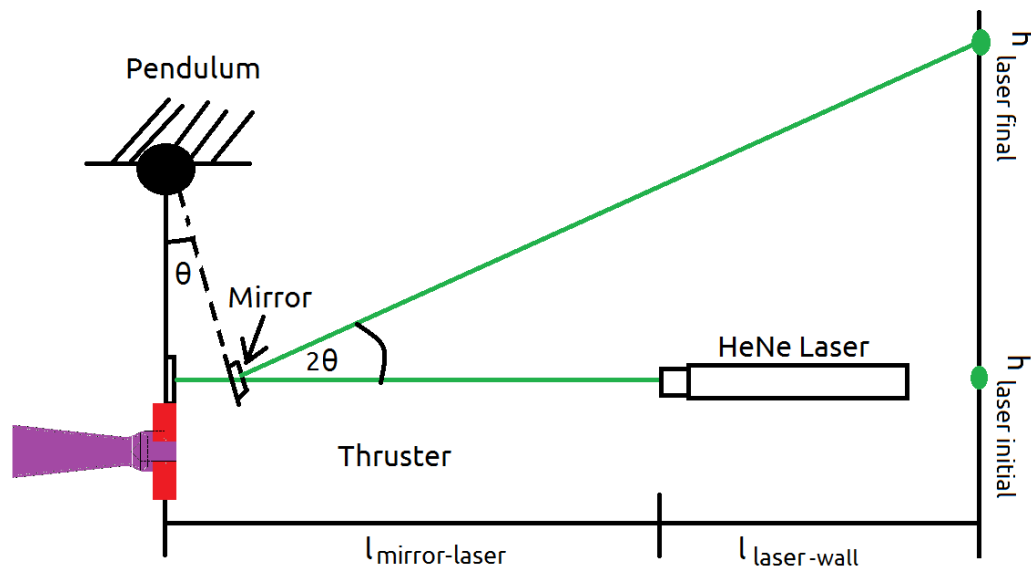
Figure 3.23: Pendulum-Laser Schematic

The angle of the thrust, θ, is calculated,

$$\theta = \frac{1}{2}\left(\arctan\left(\frac{h_{laser\,final}}{l_{mirror-laser} + l_{laser-wall}}\right) - \arctan\left(\frac{h_{laser\,initial}}{l_{mirror-laser} + l_{laser-wall}}\right)\right). \qquad (3.9)$$

Once the angle is calculated, the thrust is calculated as the product of the weight of the thruster, $W_{Thruster,}$ and the sine of the angle,

$$F_{Thrust} = W_{Thruster} \cdot \sin(\theta). \qquad (3.10)$$

The bar of the pendulum is very light in comparison to the thruster such that its mass is neglected.

The high vacuum chamber setup is shown in figure 3.24.
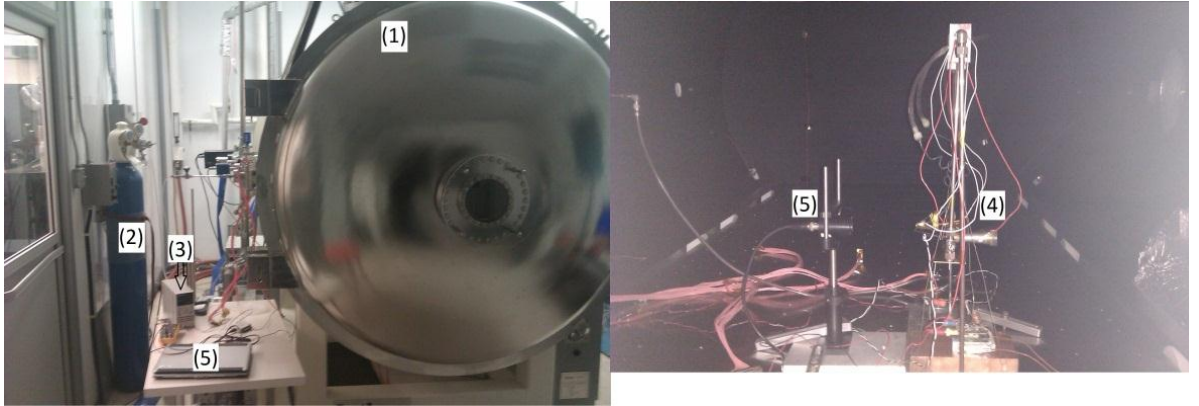
Figure 3.24: Experiment Setup for the Electrothermal Thruster Experiment. (1) Thermal Vacuum Chamber [TVaC] (2) Compressed Nitrogen Gas (3) DC Power Supply for the RF Circuit (4) Plasma Thruster and Its Structure (5) Photodiode and the Data Acquisition Hardware and Software

The ambient pressure inside the thermal-vacuum chamber, or TVaC, for this experiment is 10^{-5} Torr or 133×10^{-5} Pa with the exception of helium. The TVAC cannot remove

helium from its cryogenic pump. Thus, the ambient pressure is close to 10^{-1} Torr or 13.3 Pa.
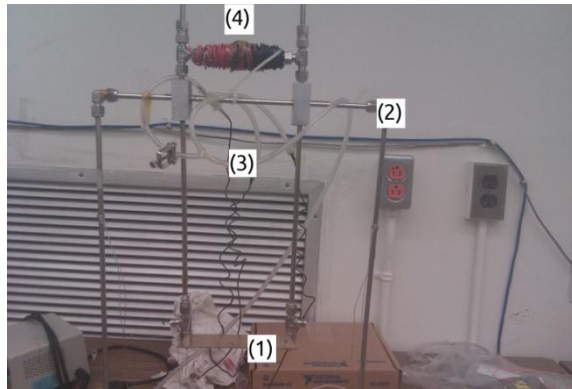
Figure 3.25: Cradle Pendulum. (1) The Cradle (2) The Support Structure (3) The Gas

Tubes (4) The Counterweight

Figure 3.25 shows the cradle system used in the electrothermal experiments. The thruster is mounted on the cradle, with a LED on top of the thruster. The pivots resting on the knife edges ensure little or no friction is introduced to the pendulum. The gas tubes are mounted such that there would not be any interference between them and the pendulum. At the top of the pendulum, there is a counterweight to magnify the moment of the pendulum. Figure 1.1 (repeated) shows how the gas is fed to the thruster. The tubes are installed away from the pendulum with significant slack in order to have little or no friction. The pendulum motion is tested by allowing it to oscillated and observe the time required for it to make a complete stop.

Figure 1.1: ET Thruster manufactured for this research

Figure 3.26 illustrates these points.
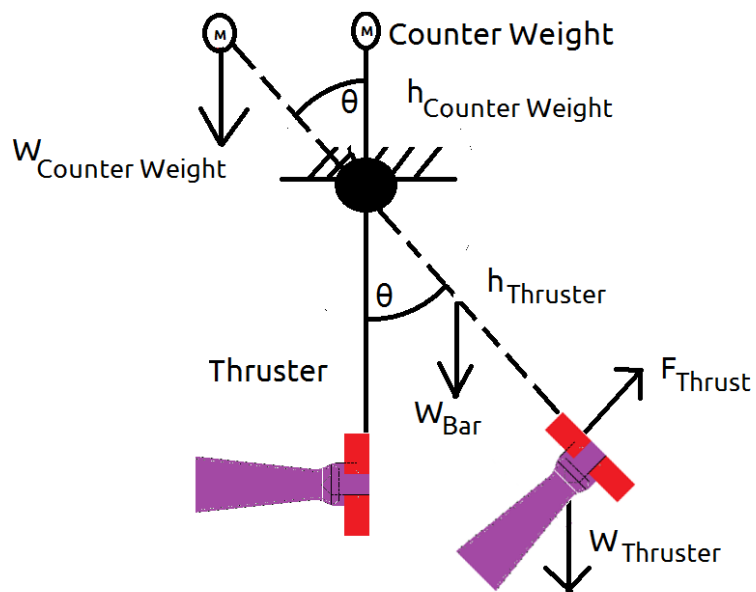
Figure 3.26: Pendulum-LED Schematic

The total thrust from the engine is

$$F_{Thrust} = \left(W_{Thruster} + W_{bar} - W_{Counterweight} \cdot \frac{h_{Counterweight}}{h_{Thruster}} \right) \cdot \sin(\theta), \qquad (3.11)$$

where $h_{Counterweight}$ and $h_{Thruster}$ are the center-of-mass distances of the counterweight and thruster, respectively. The weight of the bars, W_{bar}, is taken into account since it affects the measurement of the thrust. The counterweight, $W_{Counterweight}$, and its position determine the measured thrust.

Outside of the cradle, there is photodiode to read the input light intensity from the LED. The light spread of the LED is assumed to be a Gaussian distribution. The charge potential difference from the photodiode is recorded in the computer with a data acquisition (DAQ) board. A MATLAB program reads the voltage and converts it to force and specific impulse.
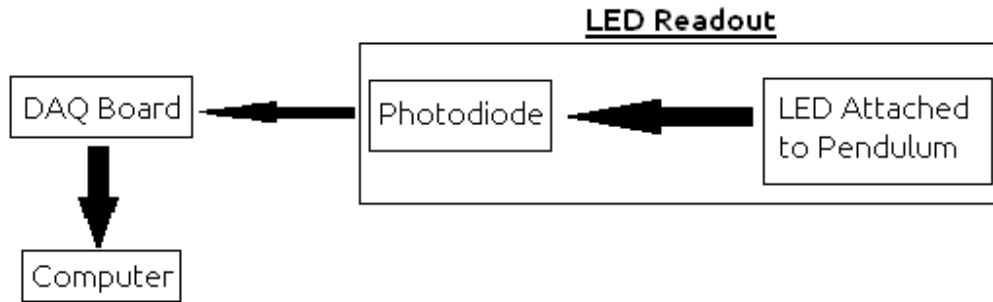
Figure 3.27: Schematic of the Readout Mechanism

Figure 3.27 shows the schematic of the readout mechanism. The LED readout needs to be calibrated for each experiment. To calibrate the readout, an accelerometer-based level is attached to the pendulum. The initial position of the pendulum is the zero, and at different

angles, the LED readout indicates a certain voltage. After taking several readings, a linear regression line is fitted into the curve, yielding a linear voltage equation given the angle,

$$\theta = aV + b \; , \tag{3.12}$$

where a and b are the slope and the y-intercept resulting from the linear regression.

The experiment determines the specific impulse of each gas, as well as of the nitrogen-helium gas mixture. The individual gases tested in this experiment are helium, nitrogen, neon and argon. The nitrogen-helium gas mixtures are tested using the following compositions: $0.1N_2 - 0.9He$, $0.2N_2 - 0.8He$, and $0.5N_2 - 0.5He$. The specific impulse is calculated from the mass flow rate of the system and the output thrust.

The performance is described by the modified specific impulse formula, shown in equation 3.12,

$$I_{sp} = \left(\frac{1}{g} \cdot \frac{F_{Thrust}}{\dot{m}} \right) \cdot (1 \pm C) \quad . \tag{3.13}$$

The constant C is the uncertainty constant in the experiment. The experimental errors include calibration error and the parasitic forces in the pendulum. The uncertainty was calculated from the data collected in the experiments. The uncertainty constant is 0.10 calculated from the data.

3.9 Hybrid Electric Thruster Experiment: Magnetoplasmadynamic Propulsion Mode

The first of the two tests in the hybrid electric thruster experiment is the EM or magnetoplasmadynamic (MPD) propulsion test. This test determines the performance and

efficiency of the so-called higher-thrust, short-term augmented-thrust mode of the thruster using the jet engine analogy. This test would run simultaneously with the ion propulsion. Because of the limited equipment available, this MPD test runs in quasi-steady state, without the ion propulsion system active.
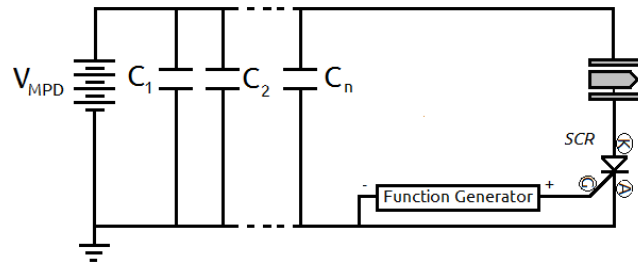
Figure 3.28: Schematic of MPD-mode test

Figure 3.28 illustrates the electronic circuit for the MPD test. It is similar to the MPD thruster circuit in figure 3.17, with the exception of the triggering. The function generator replaces the manual switch as the trigger, so that the thruster can run in quasi-steady state. It produces a +5 V pulse at 5.00 kHz.
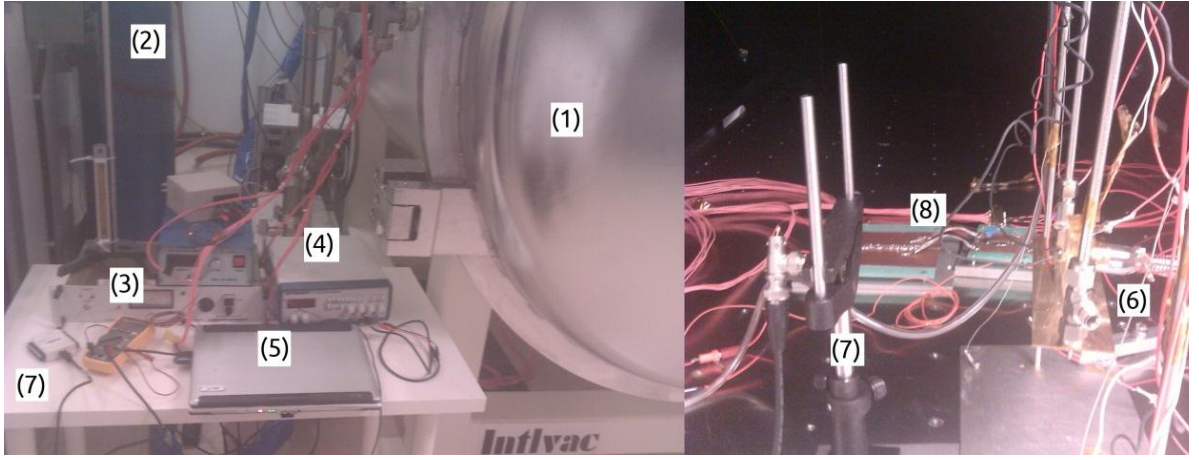
Figure 3.29: Experiment Setup for the MPD Thruster Experiment. (1) TVAC (2)

Compressed Gases (3) DC Power Supply for the MPD Circuit (4) Function Generator

(5) Computer for data acquisition (6) Hybrid electric Thruster and Its Structure (7)

Photodiode and the Data Acquisition Hardware (8) Capacitors for the MPD Thruster

Figure 3.29 shows the experimental setup for the MPD thruster. The compressed gases for this experiment are nitrogen, argon, and helium; the mixture compositions are $0.5N_2 - 0.5He$ and $0.5Ar - 0.5He$. The thruster is mounted on a pendulum and the experiment measures the propulsive force. In order to measure the thrust performance using helium gas, the experiment is performed under medium vacuum, between 10^{-1} Torr and 1.0 Torr.

The thrust efficiency is calculated using a ratio of the kinetic energy pulse to the capacitors' potential energy [Guarducci, 2011],

$$\eta_t = \frac{m_{bit} v^2}{C_{cap} V_{volt}^2}. \tag{3.14}$$

Low efficiencies are typical for a quasi-steady-state thruster.

3.10 Hybrid Electric Thruster Experiment: Ion Propulsion Mode

The second hybrid electric thruster experiment test is the ion, or electrostatic (ES), propulsion mode. This test measures the overall performance of the ion propulsion system only. Figure 3.31 shows the experiment setup for the ion propulsion run.
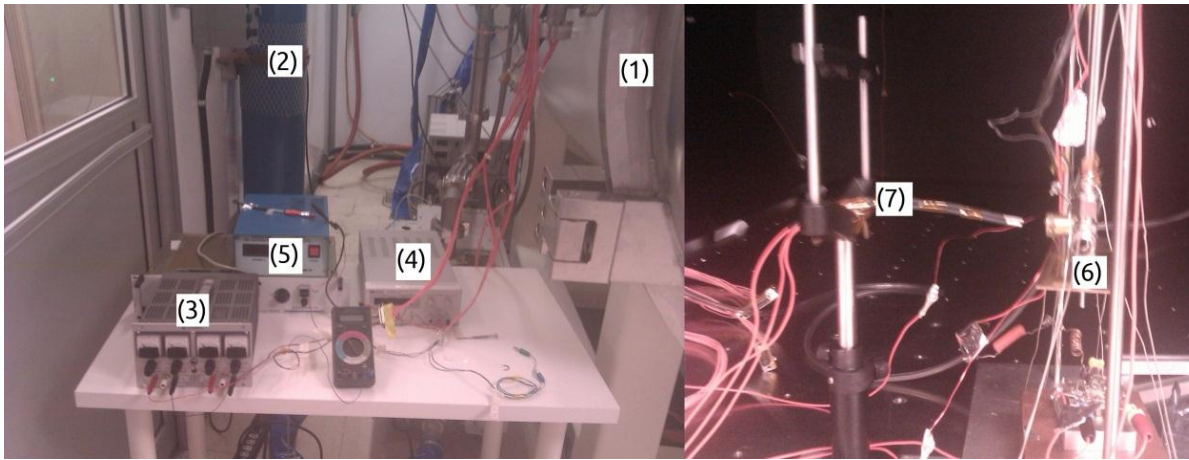
Figure 3.30: Experiment Setup for the Ion Drive Experiment. (1) TVAC (2) Compressed Gases (3) DC Power Supply for the RF Circuit and Langmuir Probe (4) DC Power Supply for the Screen Electrode (5) DC Power Supply for the Acceleration Electrode (6) Hybrid electric Thruster and Its Structure (7) Langmuir Probe

The additional two DC power supplies are for the screen and acceleration electrodes. These are high-voltage power supplies, yet the values of the current for both are within the milliamp range. The current is not important since the only purpose of the power supply is to charge the electrodes. The pendulum in this experiment is locked so that there is an accurate reading of the electron temperature.

The triple Langmuir probe is constructed using tin-coated copper wires, since the temperature of the plasma is not hot enough to melt the wires. The dimensions of the

cylindrical exposed tips are 1 cm in length by 1.5 mm in diameter, and the tips are facing towards the thruster, as seen in Figure 3.30. The wires are insulated and wrapped together such that none of the probes have contact with each other. The common sources of error and how they are rectified in the Langmuir probe are the following:

- Electron reflection: Low-energy electrons may reflect off the surface of the electrode. There is an assumption that all particles which strike the electrodes are annihilated [Laframboise, 1996].

- Contaminated electrodes: This problem causes resistance in the probe and limits the current in the probe. Cleaning the tips rectify the problem.

- RF Interference: In a triple Langmuir probe, the RF interference is filtered out since there is a floating potential. To ensure the best results, wires from the Langmuir probe are connected in a separate line in the TVAC.

The Langmuir probe construction is shown in Figure 3.31.
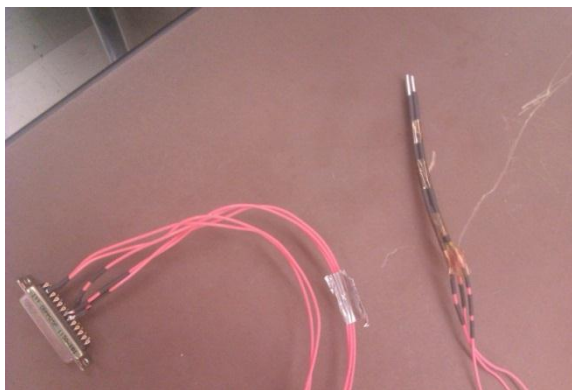
Figure 3.31: Triple Langmuir Probe Constructed in the Laboratory

Since this research focuses on the thruster design and performance in isolation, the ion thruster does not utilize a neutralizer. The performance of the thruster is not significantly affected, as observed by Yamamoto *et al.* (2006) and Koizumi & Kuninaka (2010); the performance difference is expected to be less than 1%. As mentioned in section 2.8.3, the neutralizer purpose is to prevent a negative charge build-up on the spacecraft. The test determines the overall performance of the hybrid electric thruster when it is in so-called lower-thrust, long-term sustained-thrust mode using the jet engine analogy.

The thrust is calculated by the current density measured in the Langmuir probe. The propulsive force of the ion thruster is calculated with the current density measure, as seen in equation 2.82. The ion beam current is the product of the current density measured and the area of the apertures in the acceleration electrode.

The power efficiency of the thruster is calculated as the ratio of the beam power to the total power of the system [Goebel & Katz, 2008],

$$\eta_p = \frac{I_b V_b}{I_b V_b + P_{input}}. \tag{3.15}$$

This determines how efficient the power is used in ion thruster. Also, the mass efficiency of the thruster is calculated as the ratio of the ionized mass flow rate to the total mass flow rate [Goebel & Katz, 2008],

$$\eta_m = \frac{\dot{m}_{ionized}}{\dot{m}}. \tag{3.16}$$

This determines the ionization efficiency of the plasma chamber. The overall thrust efficiency in a ion thruster is the product of the power efficiency and the mass efficiency,

$$\eta_t = \eta_p \cdot \eta_m. \tag{3.17}$$

4 Results and Analysis

This chapter focuses on the results of the simulations and experiments for the RF electrothermal thruster and the hybrid electric thruster. The results of the RF electrothermal thruster simulations and experiment were published in the 2012 Australian Space Science Conference proceedings (peer-reviewed) published in June 2013 [Stoute, C. A. B. *et al,* 2013].

4.1 Propulsion Simulations

Results from the RF electrothermal thruster simulations that are presented for the nozzle flow illustrate the fluid dynamics of the plasma, including the influence of gas properties and the Prandtl-Meyer expansion fans. Results from the hybrid electric thruster simulations illustrate the effects of the reheating of plasma. The simulation also shows what occurs when there is an external magnetic field present in the plasma flow. The results are presented in a RGB colour scheme and coded such that they show the speed of the plasma flow; the vector fields in the results show the direction of the flow. In order to distinguish the nozzles and the internal exhaust flow, the nozzles are outlined in white. The numerical results of the computational fluid dynamics model are compared to the results of the one-dimensional analyses in tables 4.1, 4.2, and 4.3.

4.1.1 RF Electrothermal Thruster

Figure 4.1 shows the results for the RF electrothermal thruster using different gases. Helium has the highest exhaust speed at 1959 m·s^{-1}. The exhaust speed for neon is less than half of the one from helium, at 867 m·s^{-1}. Argon and nitrogen show similar

performance results at 706 and 784 m·s^{-1}, respectively. The performance difference between argon and nitrogen is 10%.
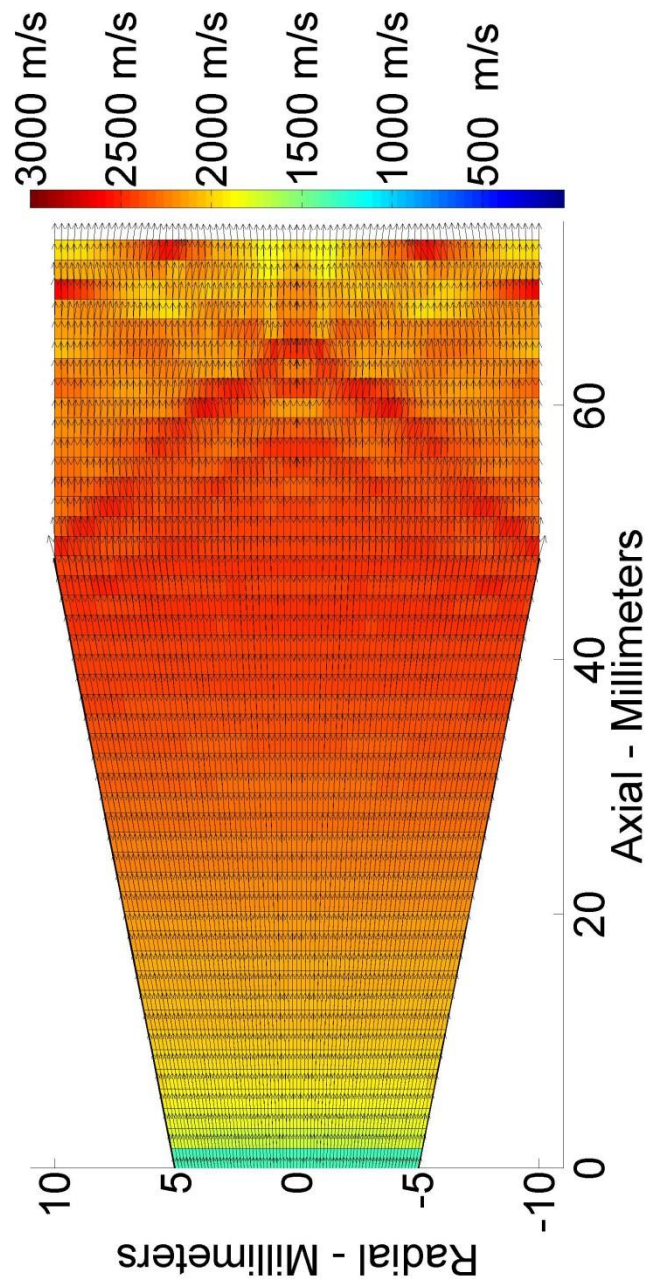
Figure 4.1a: Helium. Peak Exhaust Velocity: 1959 m·s^{-1}.
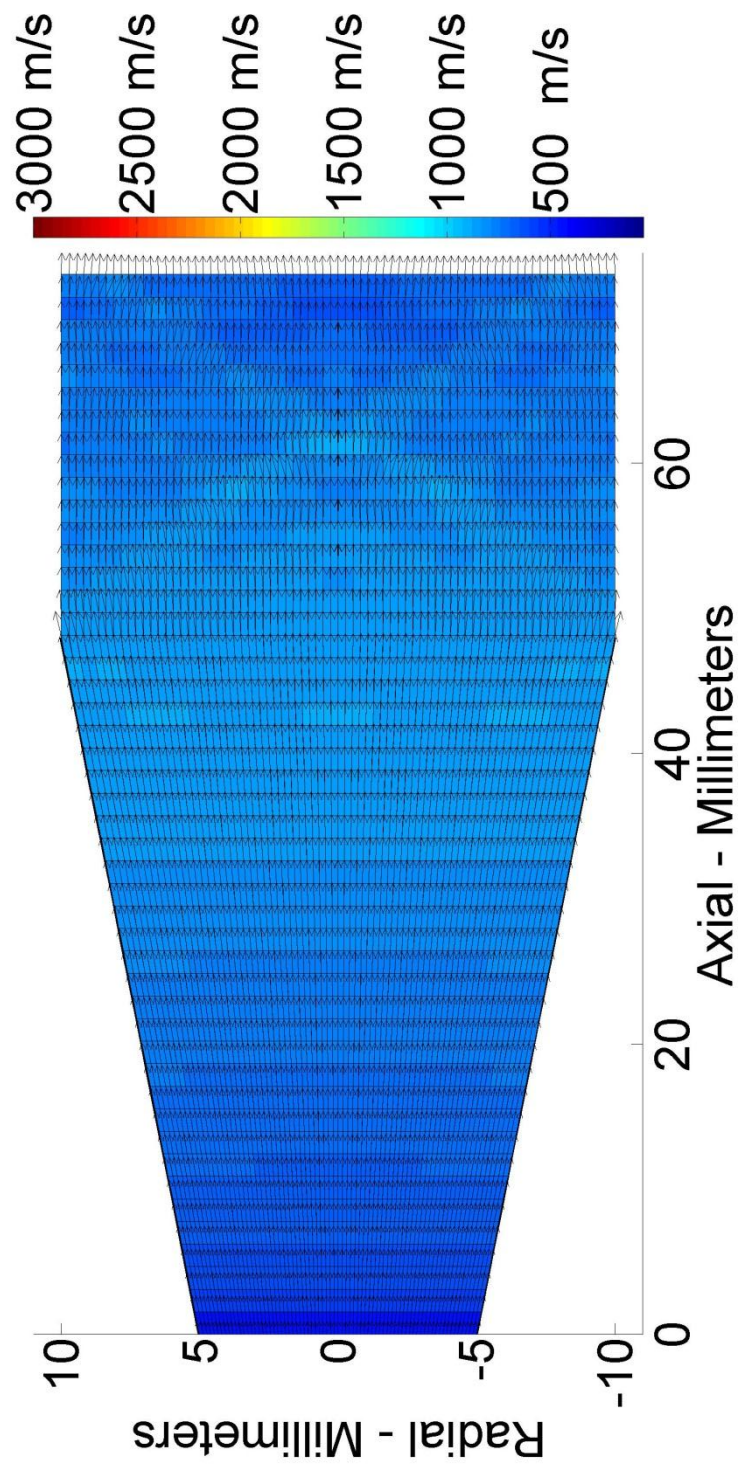
Figure 4.1b: Nitrogen. Peak Exhaust Velocity : 784 m·s⁻¹ .
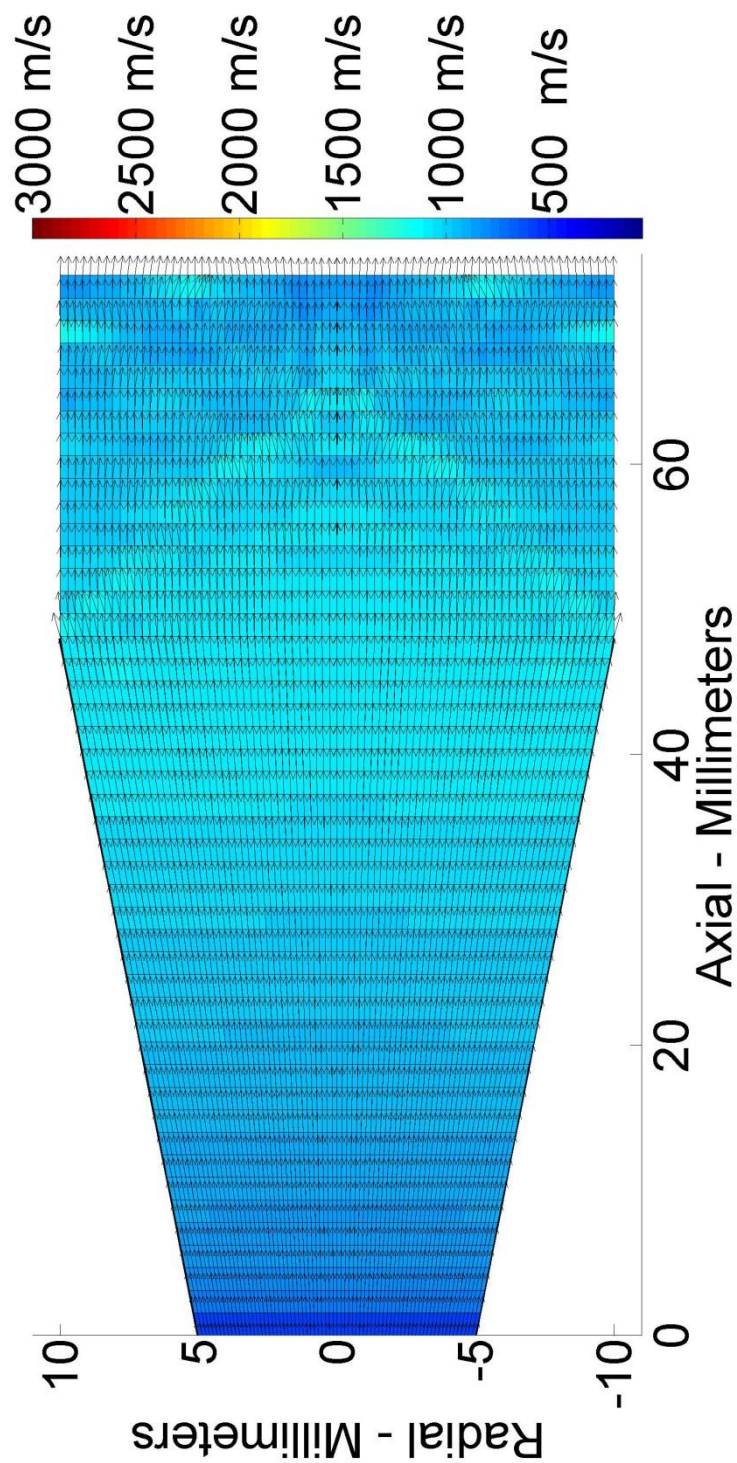
Figure 4.1c: Neon. Peak Exhaust Velocity: 867 m·s⁻¹.
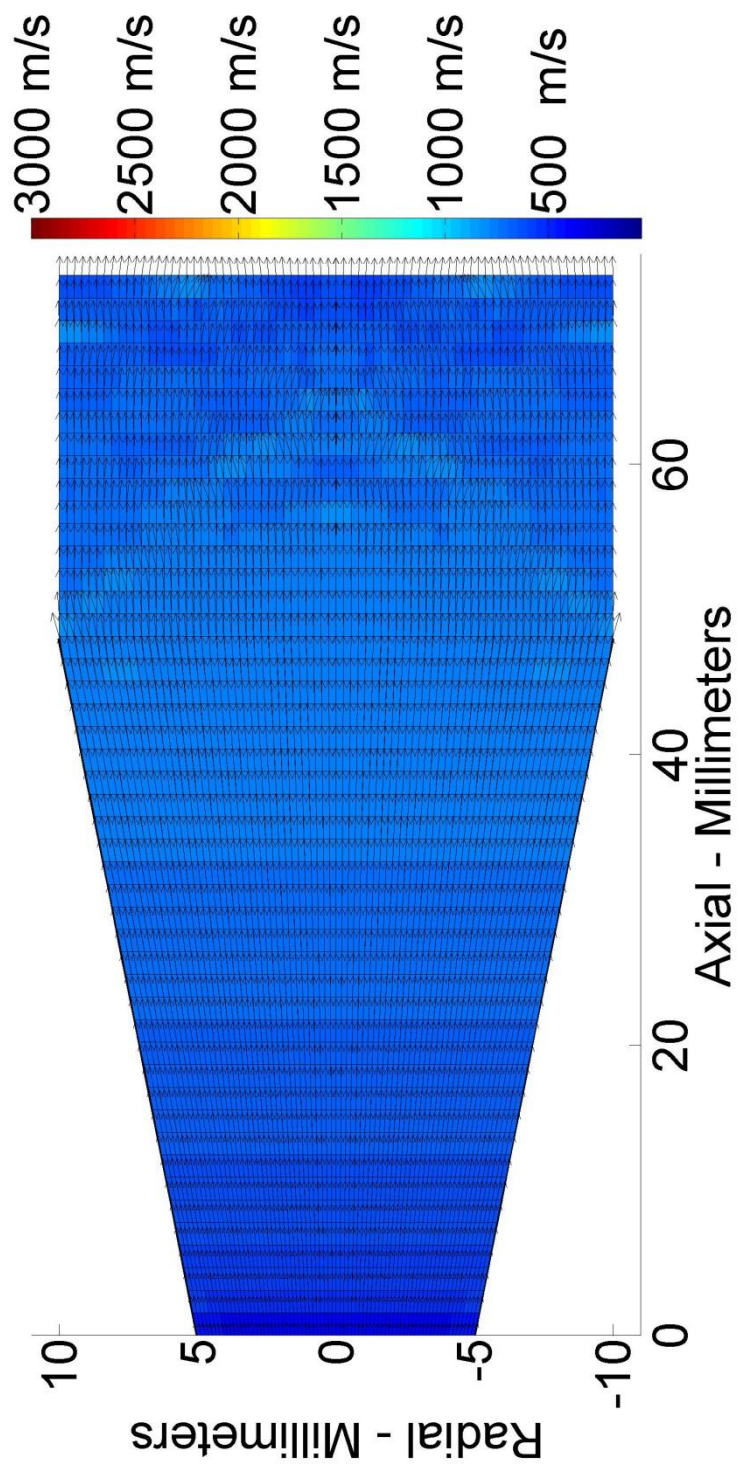
Figure 4.1d: Argon. Peak Exhaust Velocity : 706 m·s⁻¹.

113

The exhaust profiles in all four simulations are the result of an overexpanded nozzle. The ambient pressure in the space environment is less than 10^{-5} Torr. Because of the finite precision of the machine (software model limit), the ambient pressure in the simulation cannot go below 10^{-3} Torr. The ratio of the inlet pressure to the ambient pressure is normalized to 10:1 in order to match the experimental results. As mentioned in Section 3.7, the viscous forces are omitted to simplify the calculations. Figure 4.2 compares the temperature and pressure distribution of the exhaust of helium in the simulation and in the experiment. The temperature plot in figure 4.2a illustrates how the temperature of the plasma cools as the volume expands. The light emitted in figure 4.2c is the photons released from the ionized gas. Refer to *Jahn* for more details on this phenomenon [Jahn, 1968].
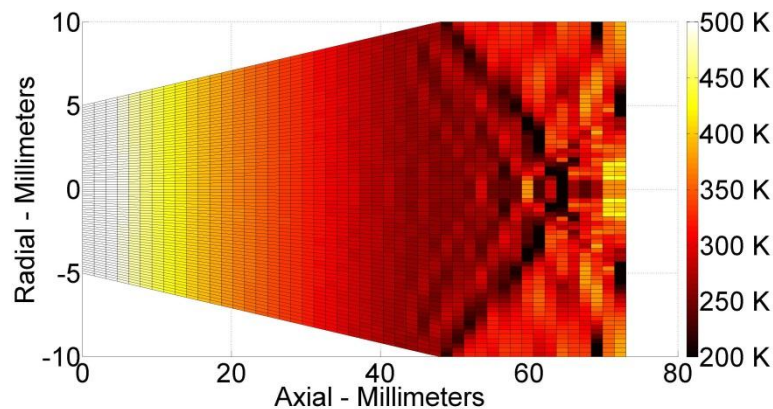
Figure 4.2a: Temperature Model of the Helium Thruster
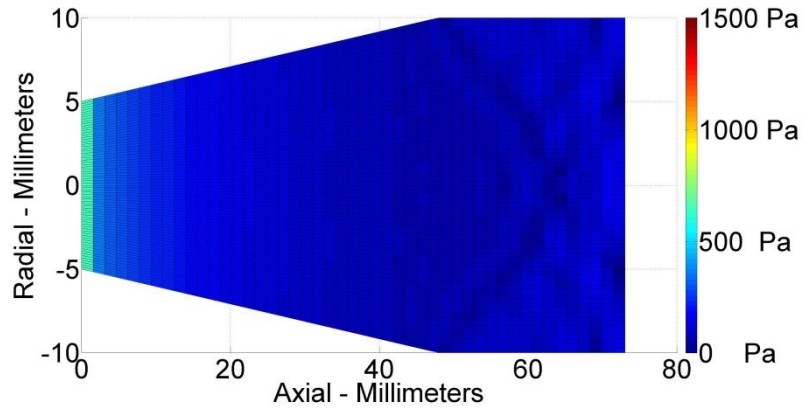
Figure 4.2b: Pressure Model of the Helium Thruster
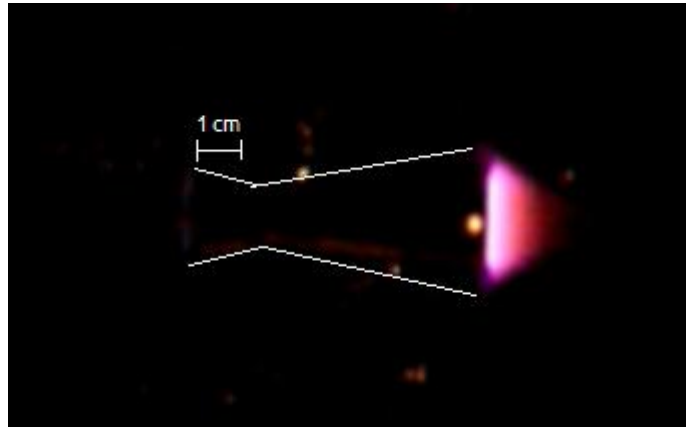
Figure 4.2c: Helium Experiment

Figure 4.1b shows the exiting flow angle in nitrogen is larger than in the other test gases. This is because it has a higher specific heat capacity at constant pressure, 29.185 kJ·kmol^{-1}·K^{-1}; in comparison to 20.195 kJ·kmol^{-1}·K^{-1} for helium, neon, and argon. The Prandtl-Meyer expansion fan (equation 2.72-73) and the exhaust velocity (equation 2.66) explain the diamond pattern, as discussed in Section 2.7.2. The angle of the fan increases as the heat capacity ratio, γ, lowers from 1.66 for the inert gases to 1.4 for the nitrogen gas.

Table 4.1: Specific Impulse and Force Performance for Different Elements

Element	CFD I_{sp} (seconds)	One-Dimensional I_{sp} (seconds)	Error (%)	Force (mN)
Helium	199	208	-4.5	3.54
Neon	88	92	-4.6	3.54
Nitrogen	80	83	-3.8	3.65
Argon	72	66	+8.3	3.54

Table 4.1 shows the specific impulses from the simulations, both in the CFD model and the one-dimensional model. The error compares the CFD model to the one-dimensional model assuming that the one-dimensional model is precise. Also, Table 4.1 shows the propulsive force calculated from the CFD model. Helium, neon and argon have the error of 4.5%, 4.6%, and 8.3%, respectively. Nitrogen has the lowest error, at 3.8%. Also, nitrogen has the highest propulsive force in the CFD model at 3.65 mN.
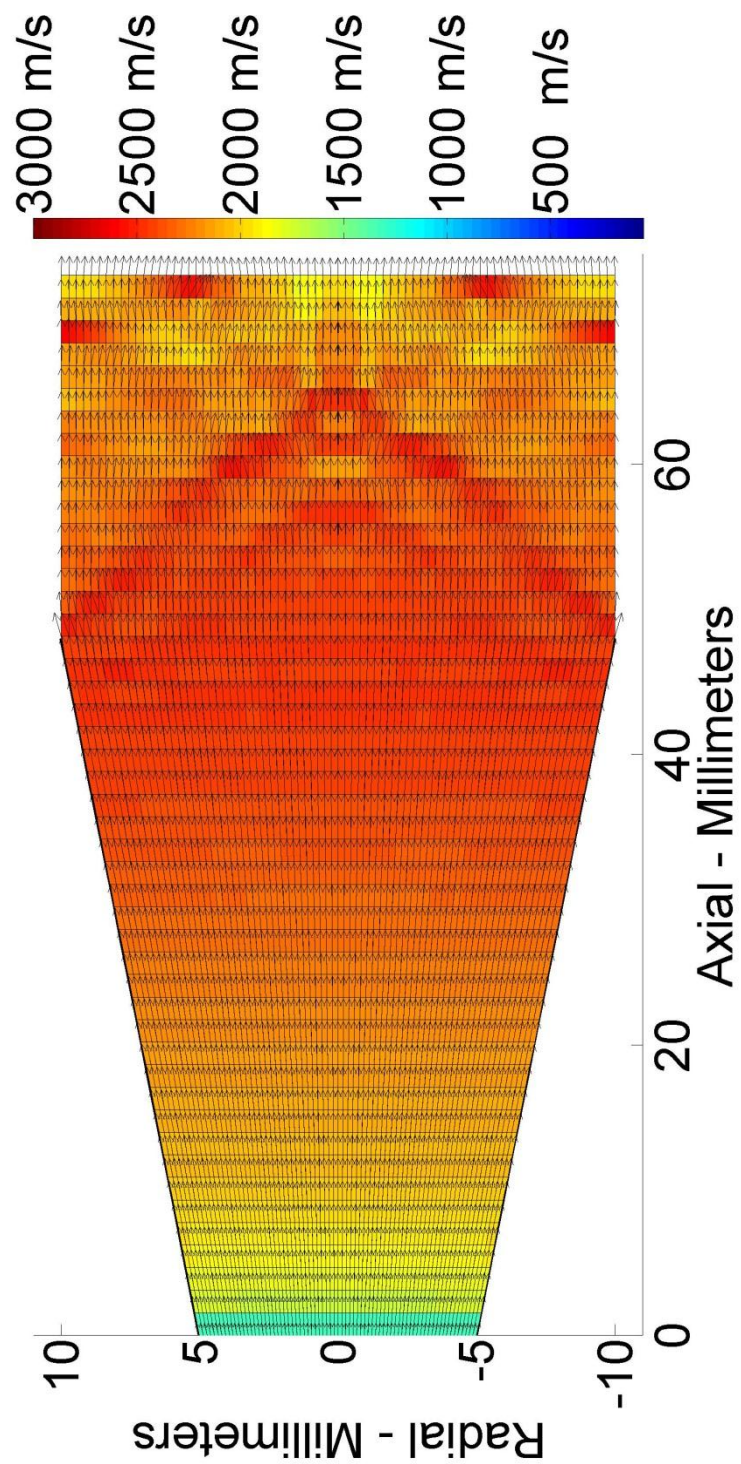
Figure 4.3a: He with Pressure Ratio 10:1
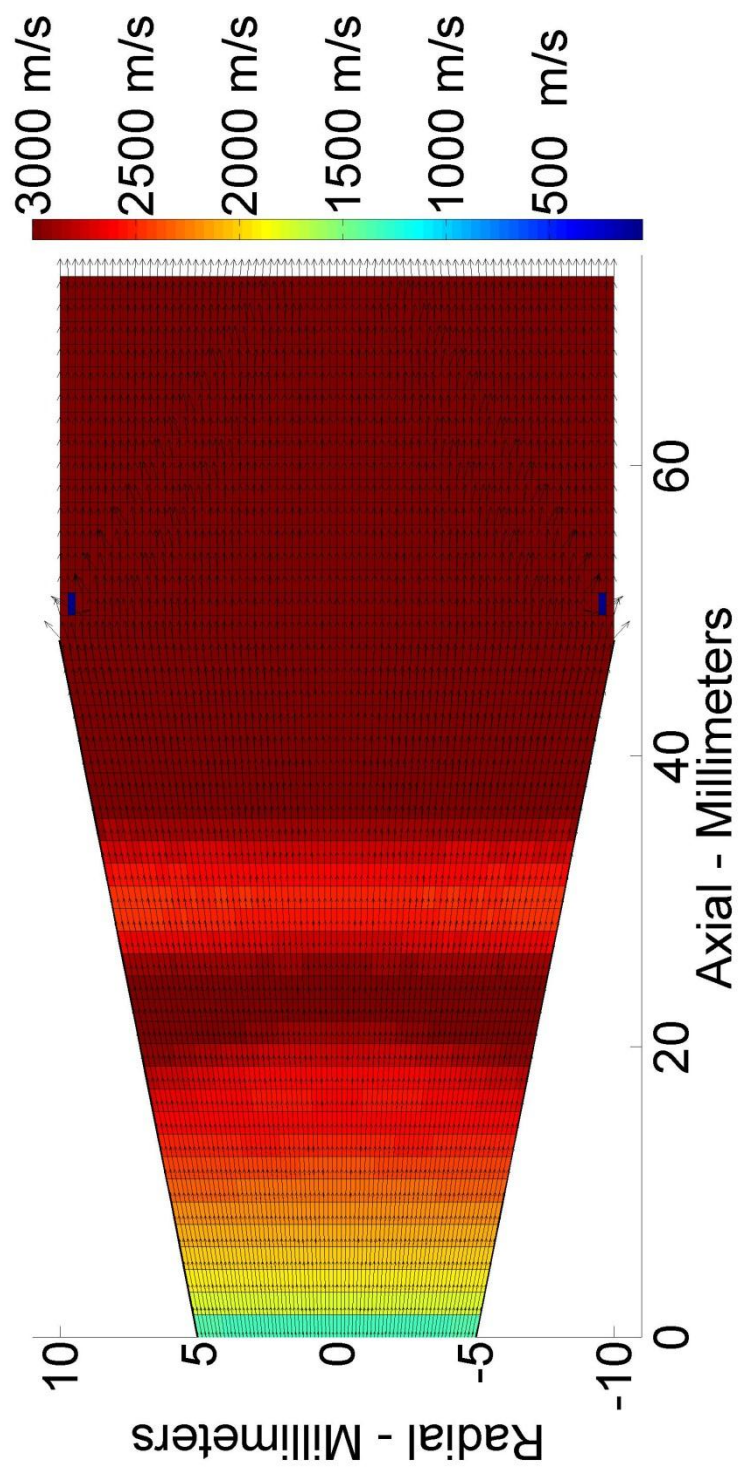
Figure 4.3b: He with Pressure Ratio 50:1
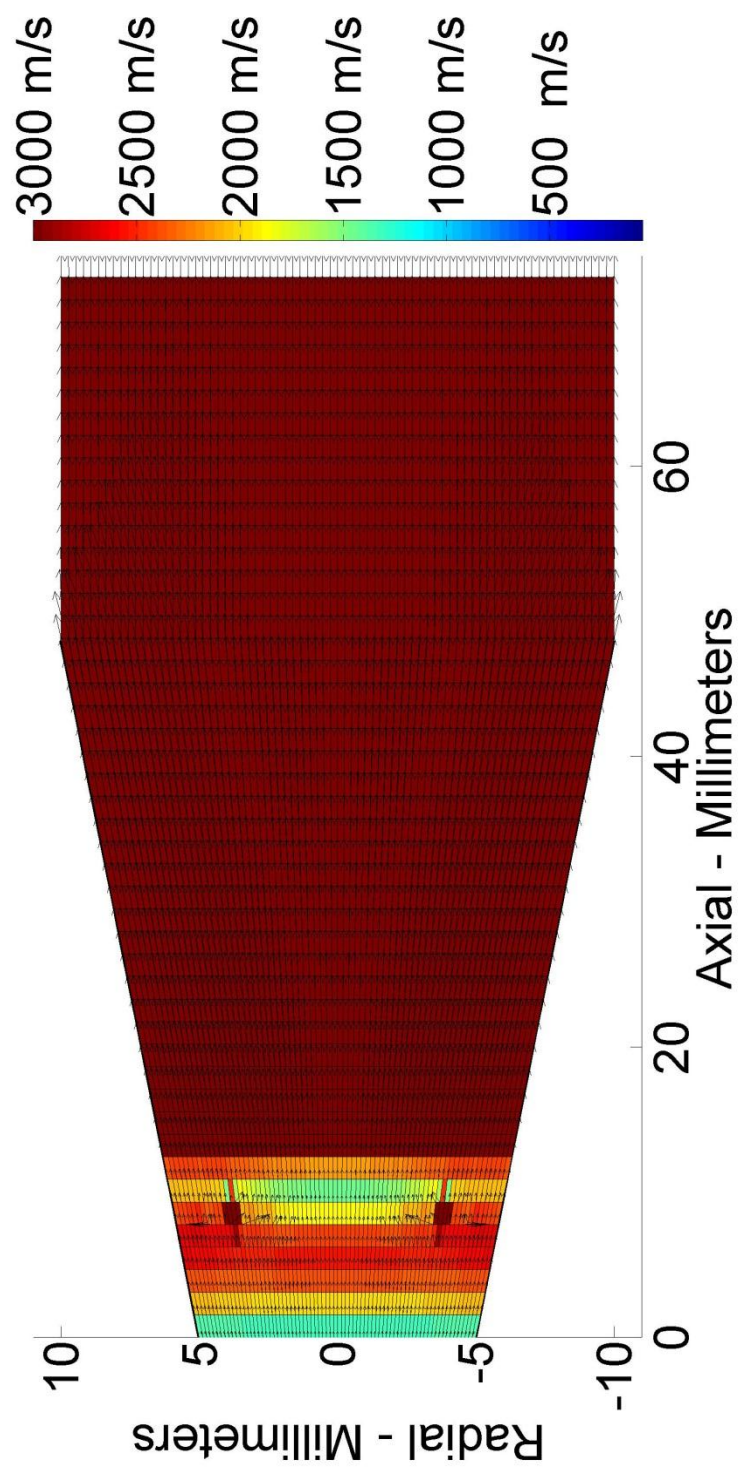
Figure 4.3c: He with Pressure Ratio 100:1
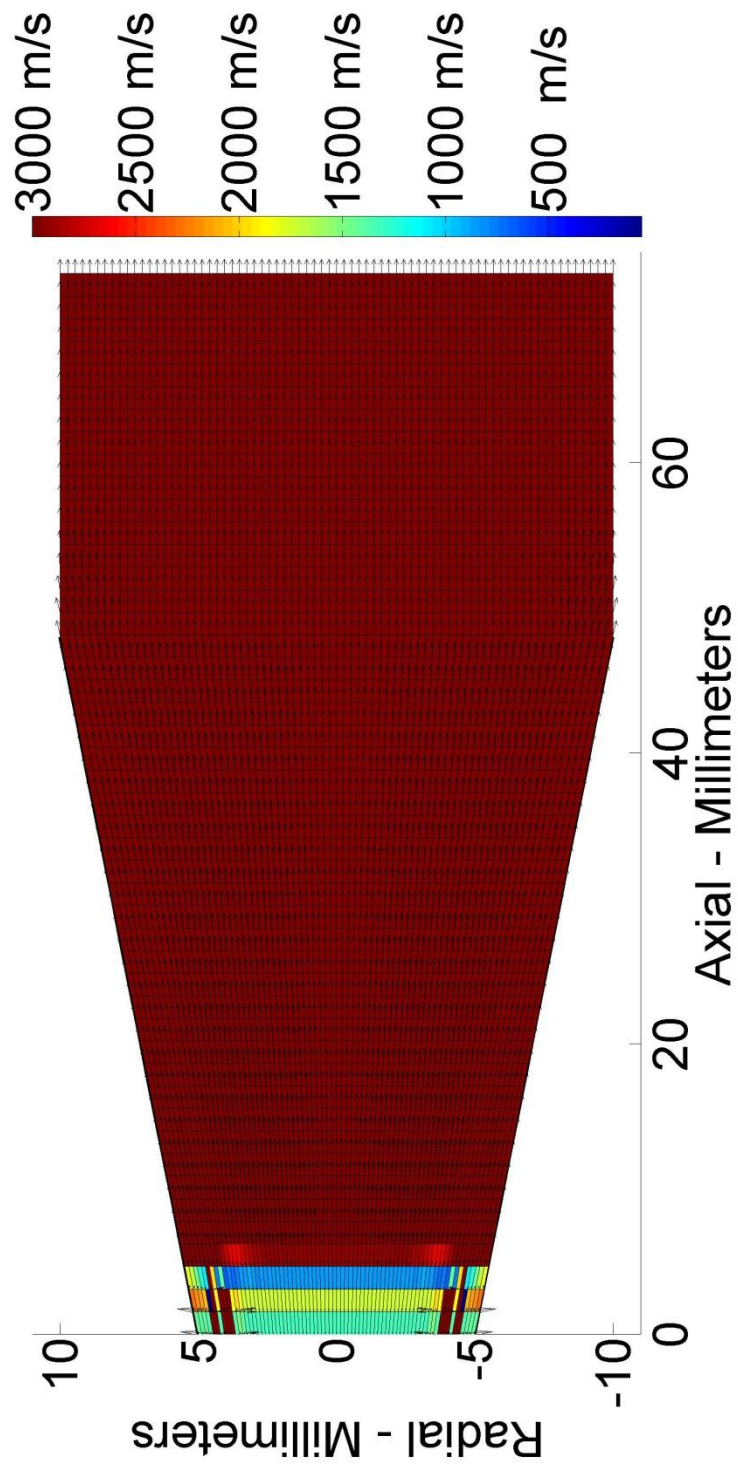
Figure 4.3d: He with Pressure Ratio 500:1

Figure 4.3 shows the helium exhaust at various inlet-ambient pressure ratios. The nozzle is overexpanded. The simulations also show that the velocities increase significantly as the inlet pressure increases.

Figure 4.4 shows the local speeds of the nitrogen-helium gas mixtures. As expected, the higher helium content yields higher exhaust velocity. All of the exhaust fluid flows outward, thus all the simulations show overexpanded nozzles. The nitrogen content determines the exiting flow angle of the exhaust.

Table 4.2 Specific Impulse and Force Performance for Different Gas Mixtures

Gas Mixture	CFD I_{sp} (seconds)	One-Dimensional I_{sp} (seconds)	Error (%)	Force (mN)
10% N_2 – 90% He	155	167	-7.7	3.53
20% N_2 – 80% He	131	145	-10.7	3.58
50% N_2 – 50% He	95	101	-6.3	3.57
80% N_2 – 20% He	89	88	+1.1	3.63

Table 4.2 shows the specific impulses and forces with different gas mixtures; the error compares the specific impulses of the two models. Regarding Table 4.2, the CFD models in all of the mixtures show lower specific impulses than their one-dimensional results, except for the $0.8N_2 – 0.2He$. Also, the $0.8N_2 – 0.2He$ has the highest force in the CFD model at 3.63 mN.
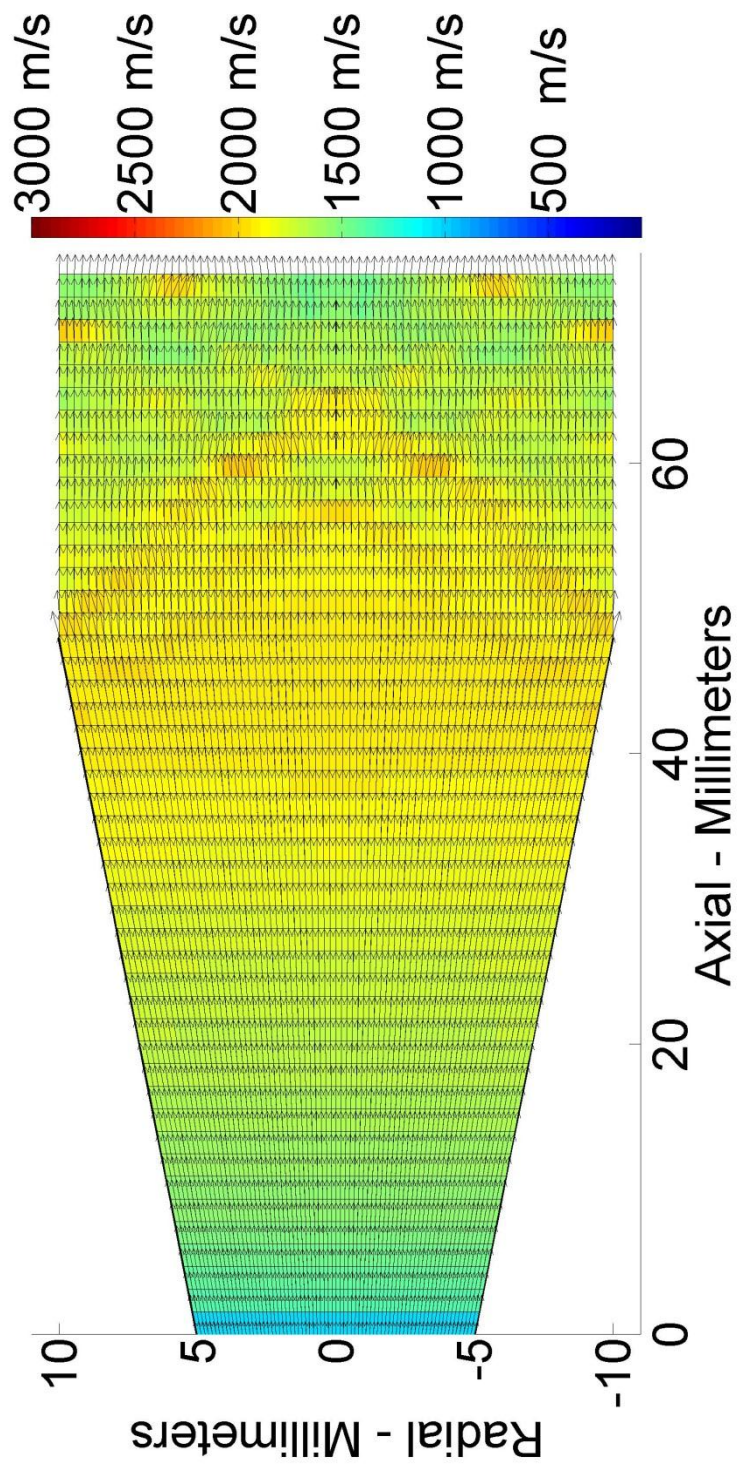
Figure 4.4a: $0.1N_2 - 0.9He$ Peak Exhaust Velocity: $1520\ m \cdot s^{-1}$.
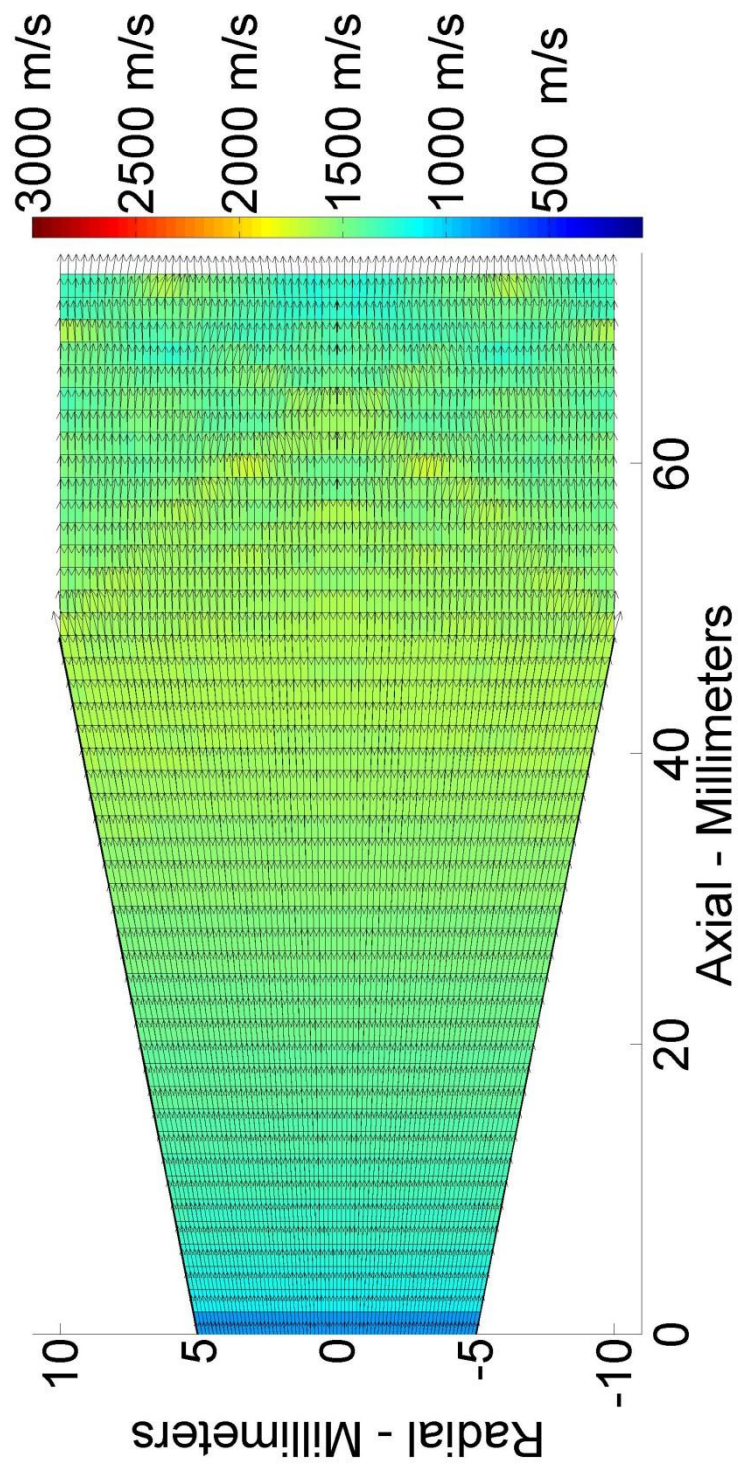
Figure 4.4b: $0.2N_2 - 0.8He$. Peak Exhaust Velocity: $1285 \ m \cdot s^{-1}$.
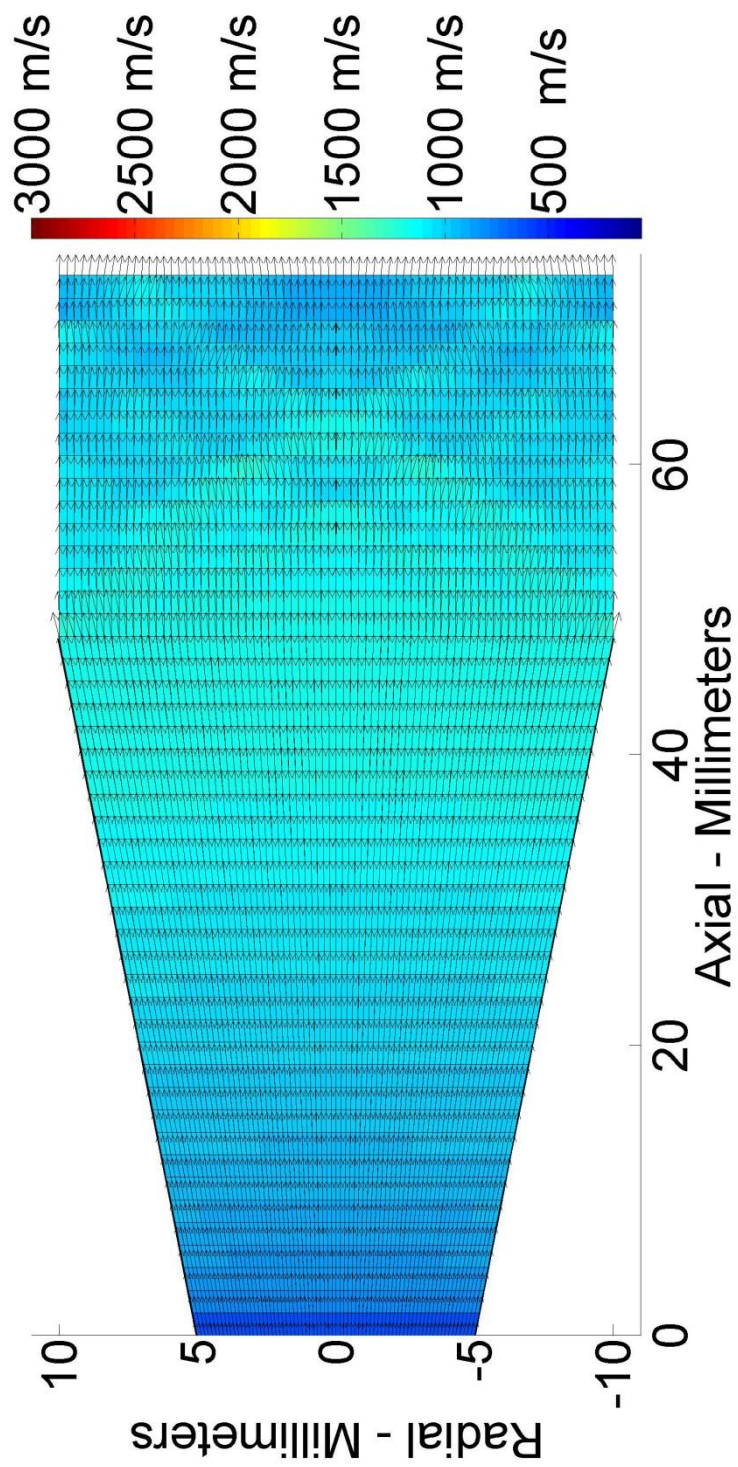
Figure 4.4c: $0.5N_2 - 0.5He$. Peak Exhaust Velocity: $932\ m \cdot s^{-1}$.
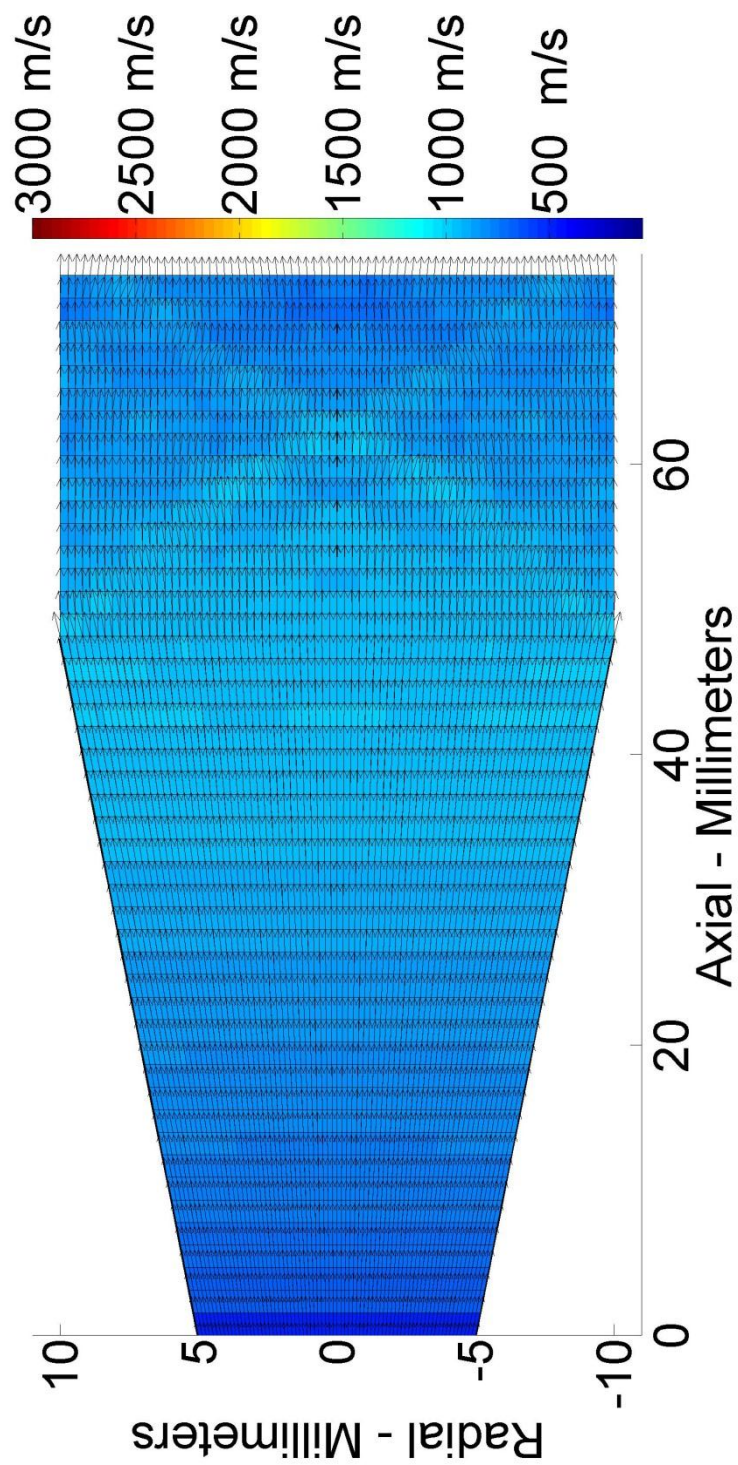
Figure 4.4d: $0.8N_2 – 0.2He$. Peak Exhaust Velocity: $873\ m \cdot s^{-1}$.

One effect that the model does not show is electron ionization. Since helium has a higher ionization energy (24.6 eV) than monatomic and diatomic nitrogen (14.3 eV and 15.52 eV), the ion production in helium is far less than in nitrogen. The ion current, equation 2.53a calculated using ions, for helium and nitrogen are 57.2 mA and 176 mA, respectively.

4.1.2 Hybrid Electric Thruster – Glass Nozzle

Figure 4.5 shows the results of the hybrid electric thruster simulation test using various gases. The simulation shows the velocity vector field when the MPD thruster mode is activated. The exhaust velocities of the electromagnetic thruster are assumed to be at the Alfvèn critical speed (equation 2.88). The results show helium to have the largest exhaust velocity at 33.4 km/s; 50% nitrogen – 50% helium at 16.0 km/s; 50% argon – 50% helium at 12.1 km/s; nitrogen at 10.9 km/s; and argon at 8.6 km/s.

Figures 4.5a, c-f show the velocities throughout the nozzle. Figure 4.5b shows the the inlet section from 4.5a of the plasma flow using helium as the propellant. There are standing shock waves appearing as the supersonic plasma enters 5 mm into the inlet; the entrained shock waves continue throughout the inlet. After the plasma exits the throat of the nozzle, it exhibits conventional diffusing at supersonic speeds. The exhausts from all five gases show that the nozzles are overexpanded. This shows that the exhaust pressure is higher than the atmospheric pressure, as seen previously in Section 4.1.1, figure set 4.1.
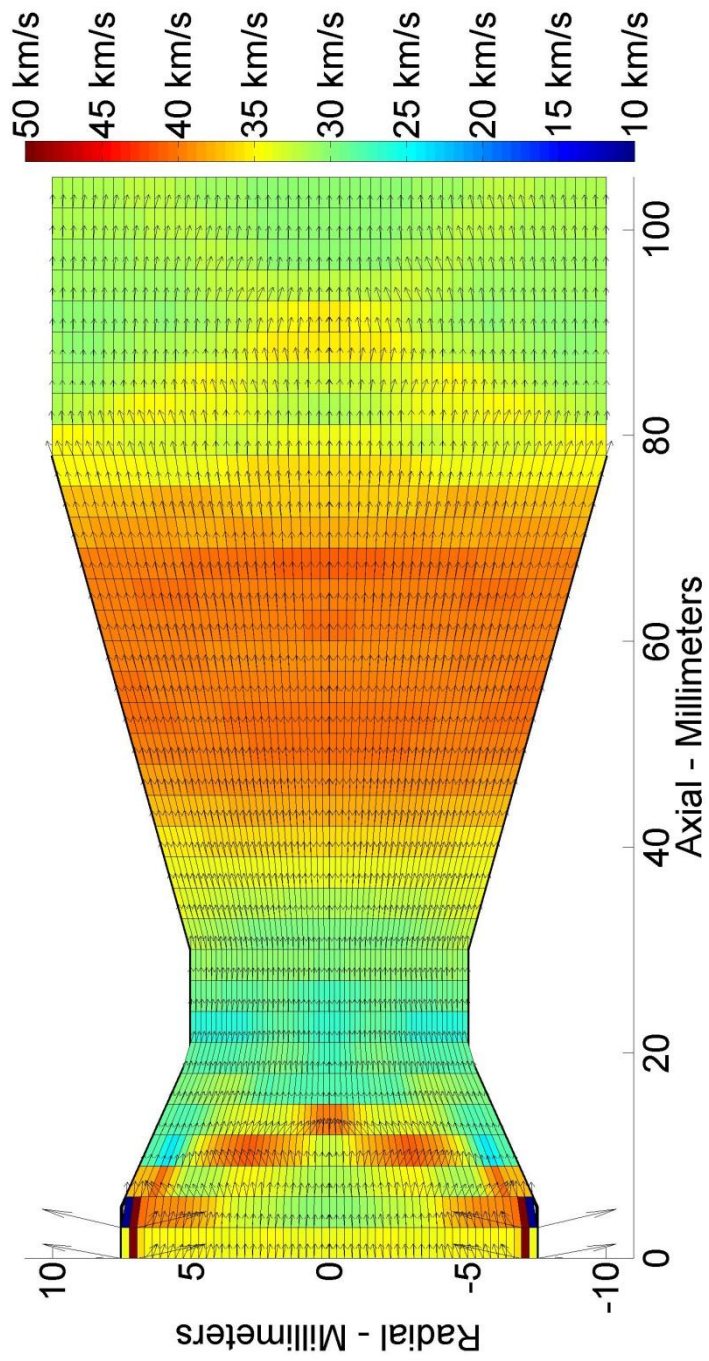
Figure 4.5a: Helium. Exhaust Speed: 33.4 km·s⁻¹ .
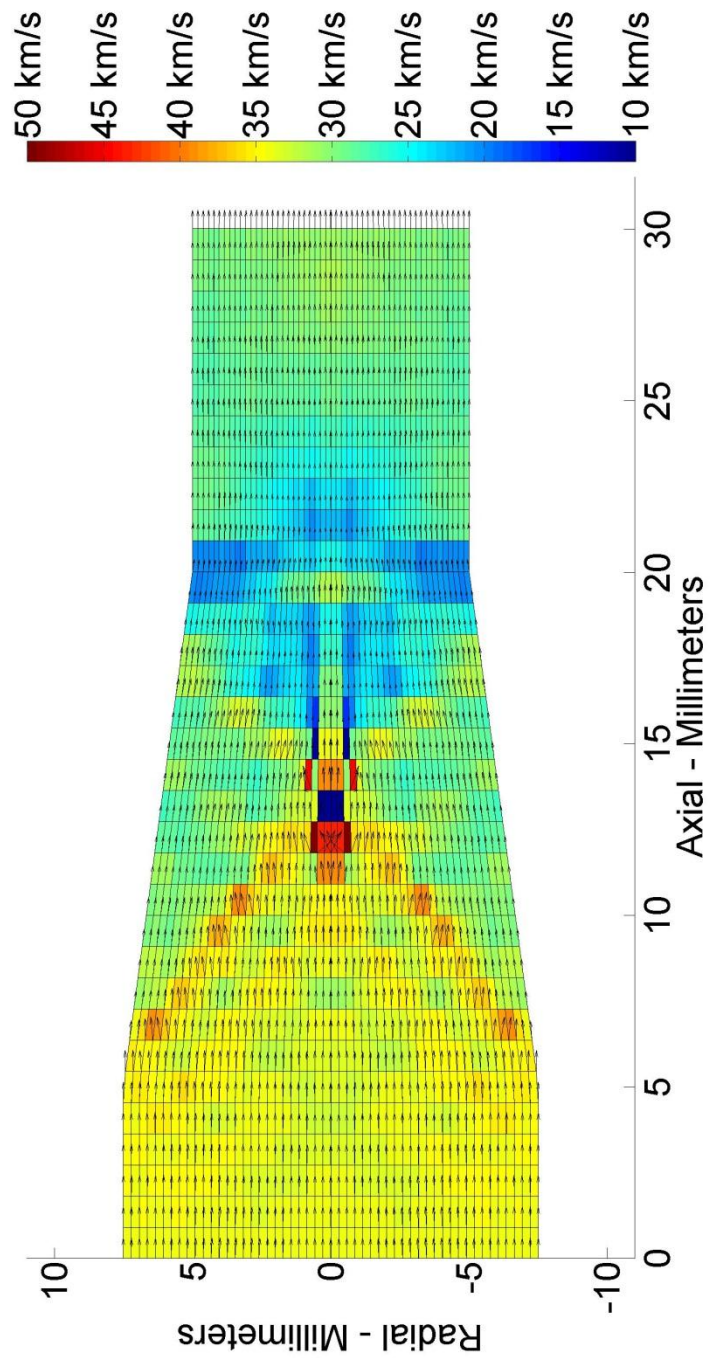
Figure 4.5b: Inlet region of the nozzle from figure 4.5a of the Glass Nozzle using Helium.
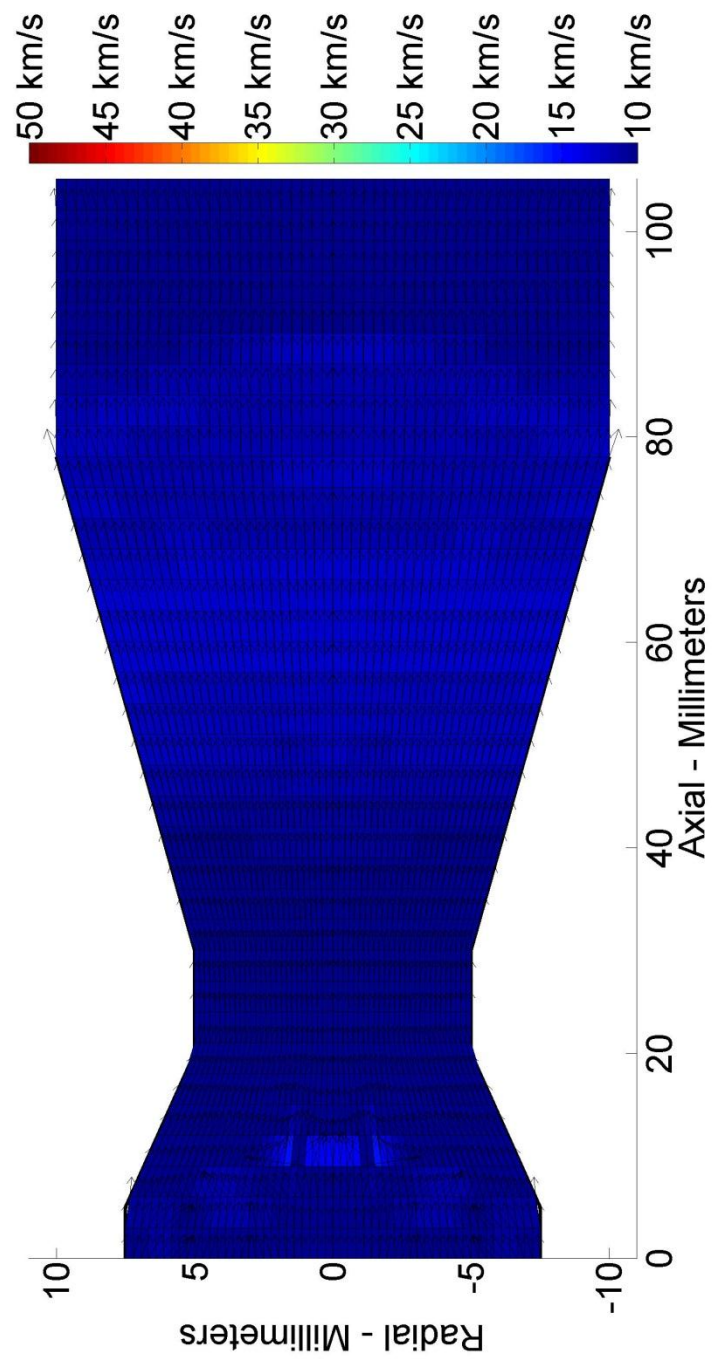
Figure 4.5c: Nitrogen. Exhaust Speed: 10.9 km·s⁻¹ .
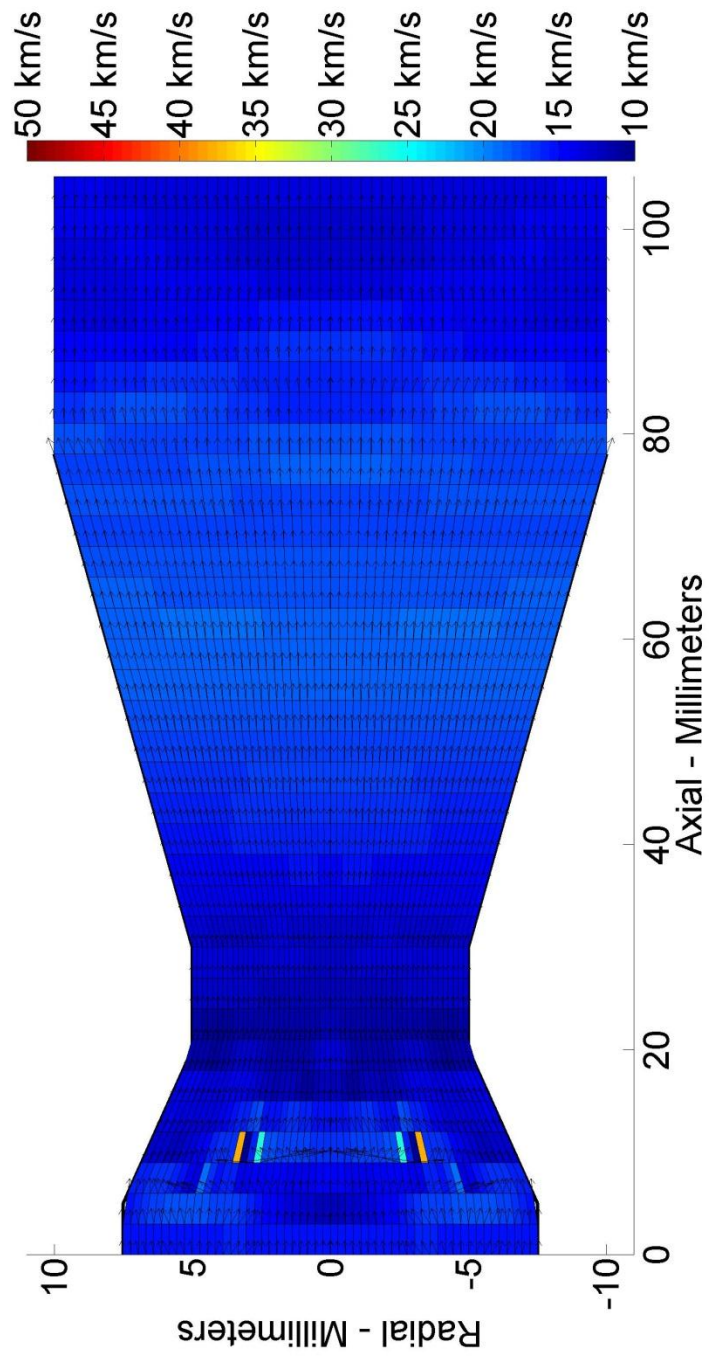
Figure 4.5d: 0.5N₂ – 0.5He. Exhaust Speed:16.0 km·s⁻¹ .
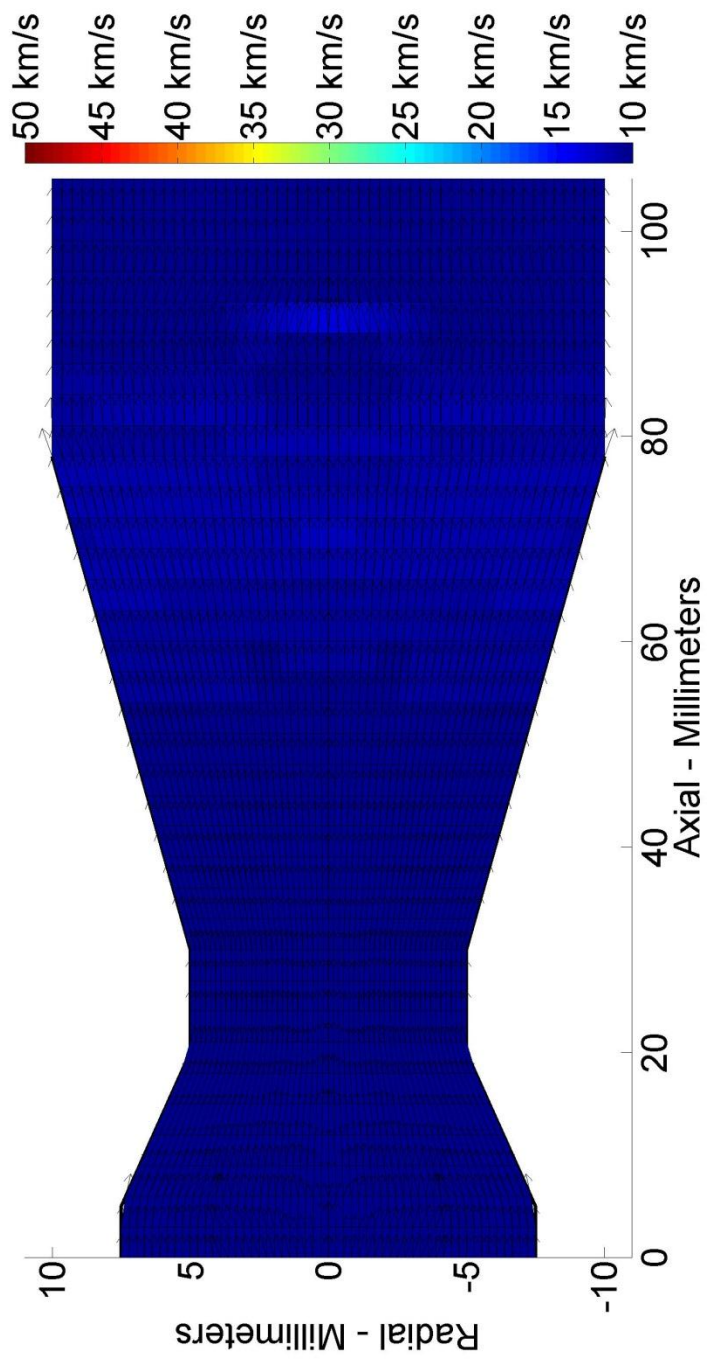
Figure 4.5e: Argon. Exhaust Speed: 8.6 km·s⁻¹ .
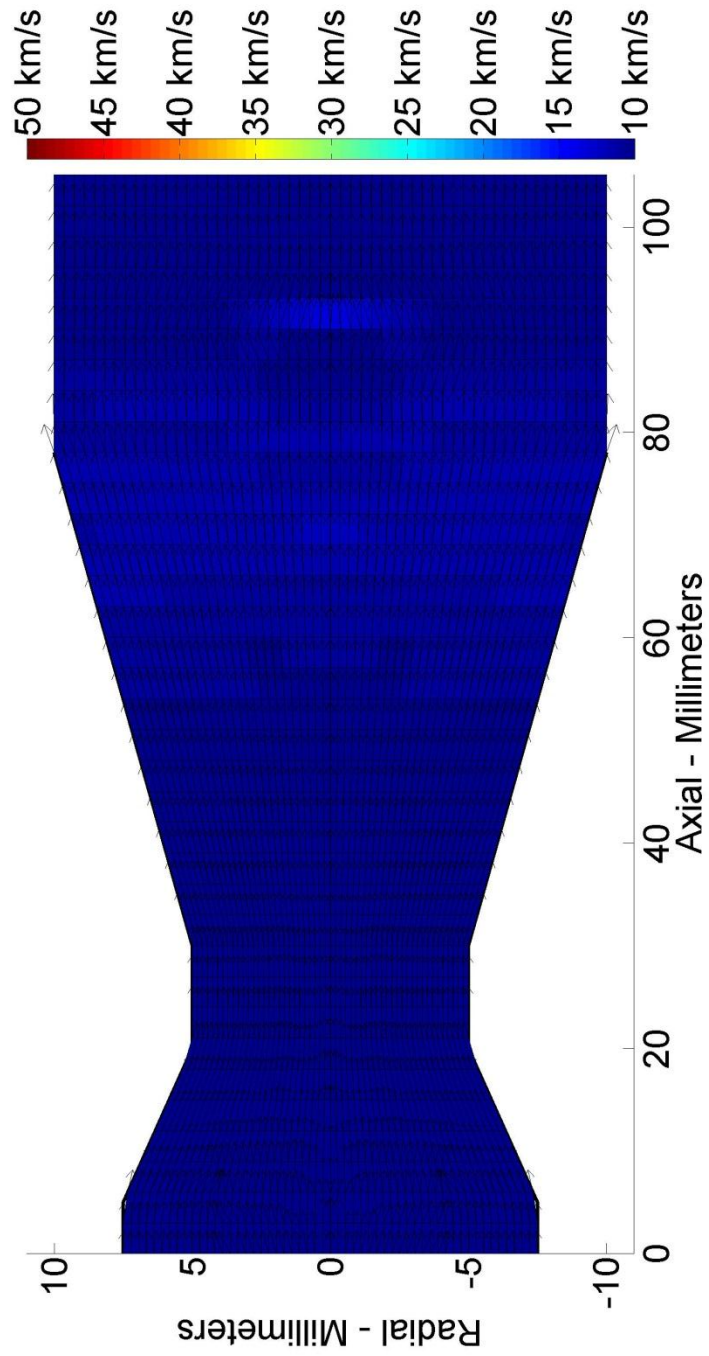
Figure 4.5f: 0.50Ar − 0.50He. Exhaust Speed:12.1 km·s⁻¹ .

Table 4.3: Specific Impulse and Force Performance for Different Gas Mixtures

Gas Mixture	CFD I_{sp} (seconds)	One Dimensional I_{sp} (seconds)	Error (%)	Force (mN)
Helium	3410	3501	-2.6	9.32
50% N_2 - 50% He	1628	1566	+4.0	7.59
50% Ar - 50% He	1232	1354	-9.9	7.86
Nitrogen	1116	1024	+8.9	6.04
Argon	876	889	-1.5	6.08

Table 4.3 shows the specific impulses from the one-dimensional model and the CFD model, and the propulsive force from the CFD model. The results show that the error magnitudes between the five pairs of results for the specific impulse are on average 4.7%. The 50% Ar - 50% He exhibits the highest computational error, at 9.9%.

The purpose of modelling the nozzle is to demonstrate the effects of reheating and magnetic field on the hybrid electric thruster. The nozzle throat is wrapped with a solenoid, as stated in chapter three, such that it reheats the gas at the throat in order to improve the performance of the thruster. Figure 4.6 shows the performance of the hybrid electric thruster under different external magnetic fields at the throat, showing only the exhausts to better visualize the events at the exhaust. These results show that there is an increase in the exhaust velocity as the magnetic field increased. The specific impulse increases as the square of the magnetic field intensity.
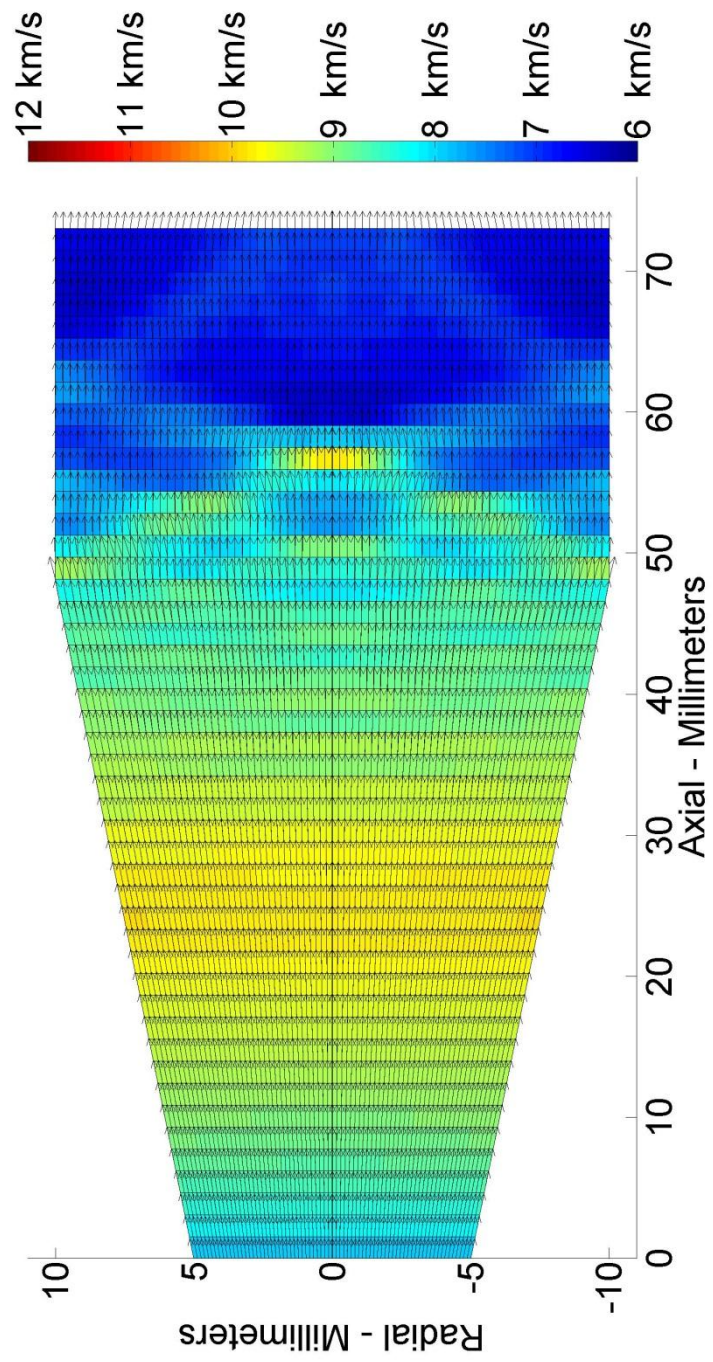
Figure 4.6a: 0 T External Magnetic Field
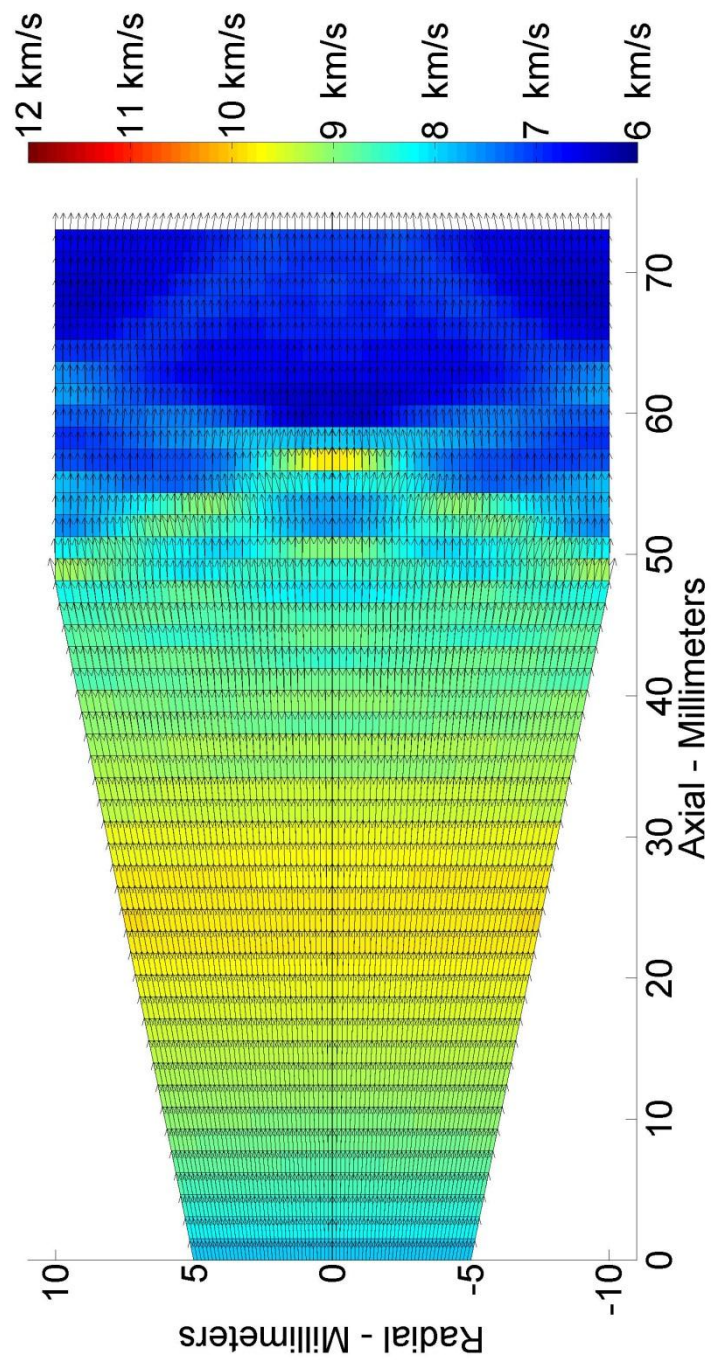
Figure 4.6b: 1.3 mT External Magnetic Field

135
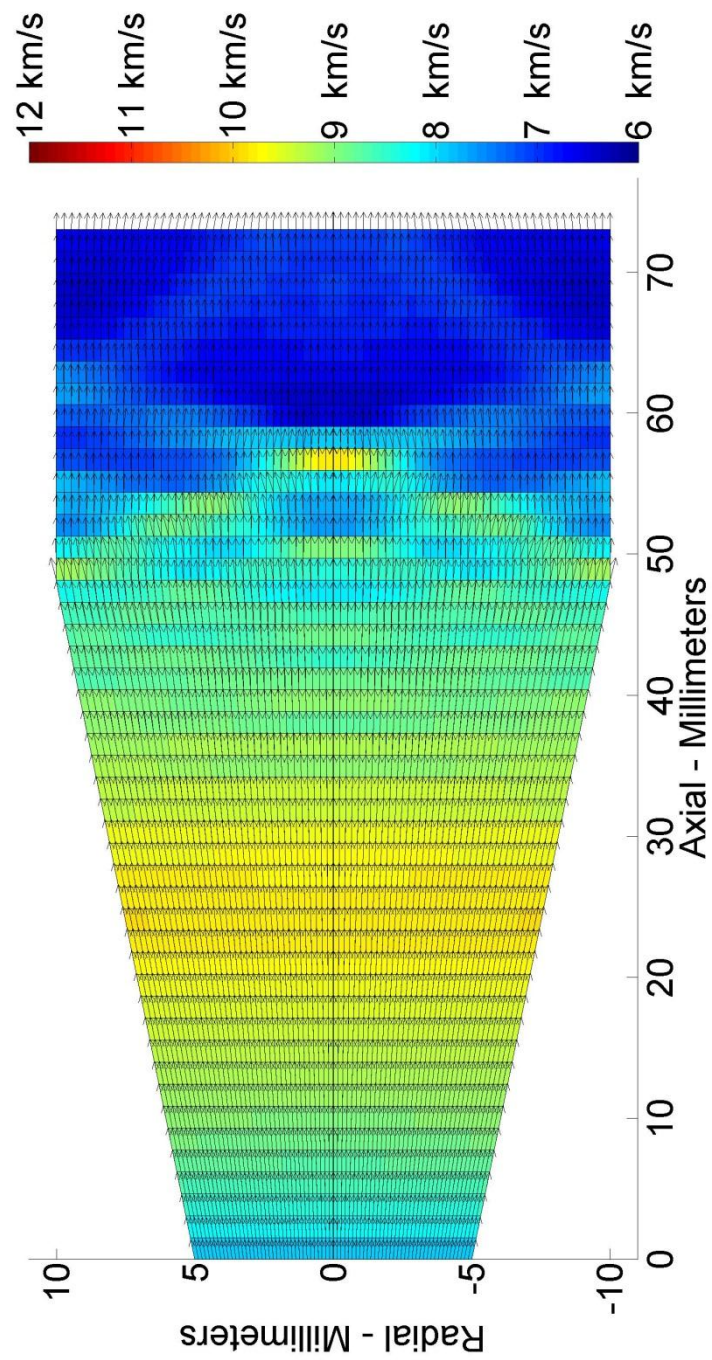
Figure 4.6c: 12.5 mT External Magnetic Field
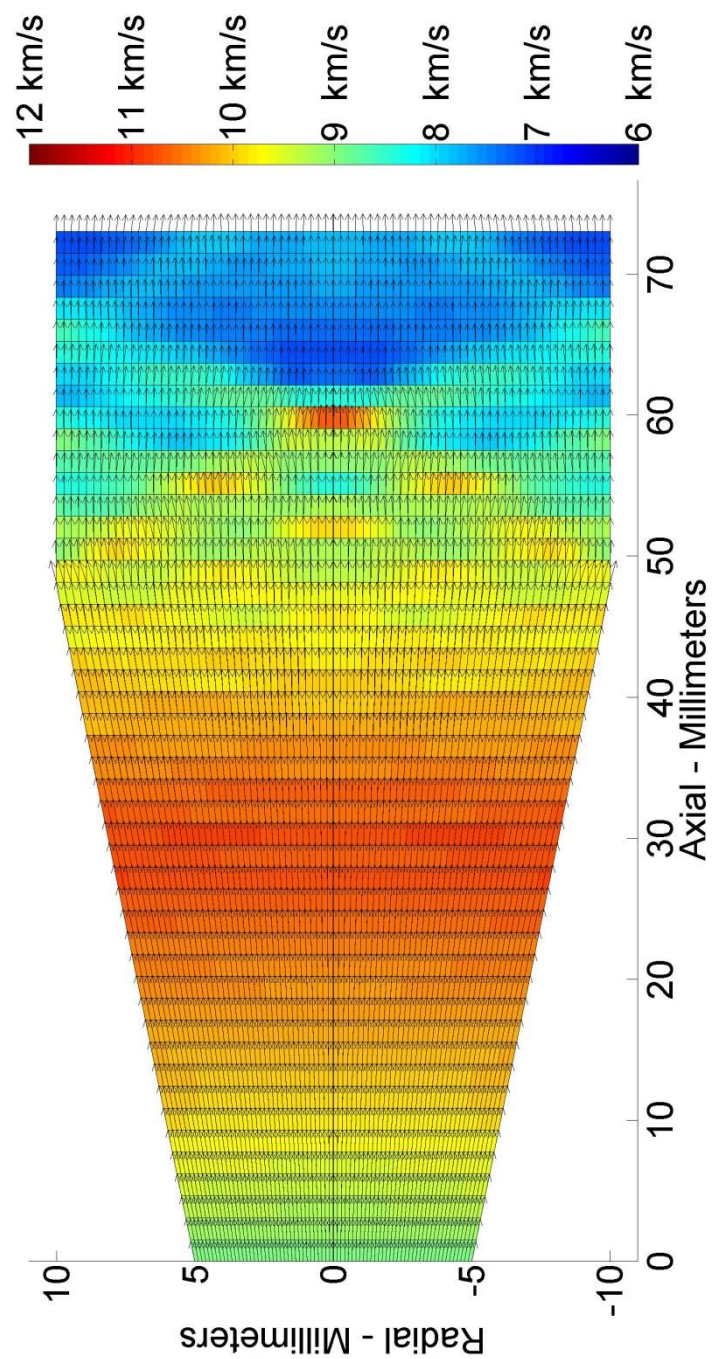
Figure 4.6d: 50.0 mT External Magnetic Field

137

Table 4.4: Specific Impulse of Argon at Different Magnetic Field Strengths

Magnetic Field (mT)	CFD I_{sp} (seconds)	One Dimensional I_{sp} (seconds)	Error (%)	Expected Enhancement (%)
0	876	889	-1.5	N/A
1.3	932	892	+4.5	6.4
12.5	954	920	+3.7	8.9
50.0	1029	1016	+1.3	17.5

Table 4.4 shows the computation errors between the one-dimensional and CFD model of the reheating of the plasma; the average error is approximately 2.8%. When the magnetic field is above 12.5 mT, there is a significant increase in the performance. The performance increase between 0 and 1.3 mT is 6.4%.

4.2 RF Electrothermal Thruster Experiment

The experimental results of the electrothermal thruster show the performance of the individual gases, as well as the nitrogen-helium gas mixtures. The pure gases are shown as a proof-of-concept reference for the thruster.

4.2.1 RF Electrothermal Thruster – Pure Gases
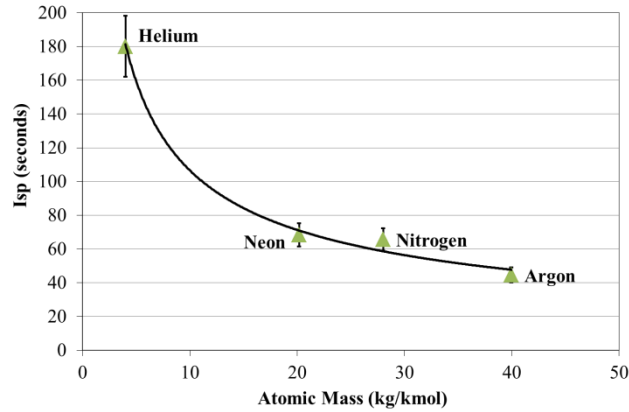
Figure 4.7a: RF Plasma Graph for Individual Gases in the Small TVAC in Medium

Vacuum
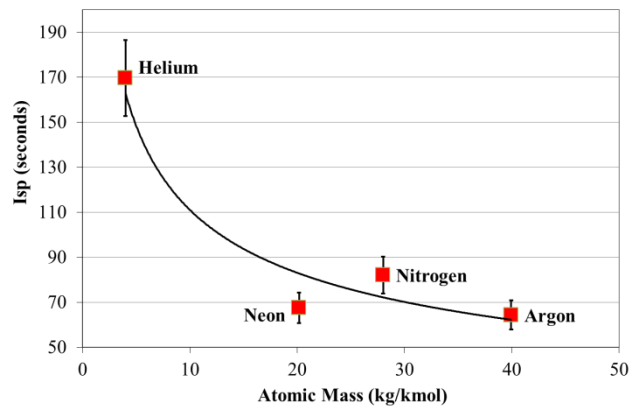
Figure 4.7b: RF Plasma Graph for Individual Gases in the Large TVAC in High Vacuum

Figure 4.7 shows the specific impulses of each gas tested. The trend lines follow the exhaust velocity equation 2.66 divided by gravity. Those from the medium-vacuum test are similar to the those from high-vacuum test. The reason for this is that the medium-vacuum test used a laser, a mirror, and ruler scale to find the propulsive force and, consequently, the specific impulse. A factor which contributes to inaccuracies in the

139

measurements is the position of the laser relative to the ruler on the wall of the room and to the small TVAC. However, in the large TVAC, the accuracy improves by approximately 8%. The inaccuracies are corrected by using a LED and a photodiode. The LED/photodiode is required to be calibrated before each experiment run. This method has proven to be more accurate than the laser method, since the photodiode is reading the position of the LED, rather than the wall. In addition, the trendlines agree with the uncertainty from the small TVAC results. The Table 4.5 shows the force and mass flow rate for each gas.

Table 4.5: Force and Mass Flow Rate of Gases at Nozzle Exit

Gas	Force (mN)	Mass Flow Rate (mg/s)
Helium	6.59	3.95
Neon	3.26	4.81
Nitrogen	4.05	5.04
Argon	1.83	2.87

Using the results from the large TVAC, helium shows the highest specific impulse at 170 seconds. Nitrogen has a specific impulse of 82 seconds. Neon is shown to have a specific impulse of 69 seconds. Argon has the lowest specific impulse at approximately 65 seconds. In addition, comparing the specific impulses from the simulation to the experimental results for nitrogen, the difference is less than 2.5%.

Figure 4.8 shows the exhausts of the thruster using the different gases under medium vacuum. With a pressure inlet-to-atmosphere ratio of 10:1, the exhaust pressure is lower than the atmospheric pressure (see equation 2.69); therefore, the exhausts shown

are experiencing overexpansion. The white lines represent the outline of the nozzle. The gases shown in the figure are helium, neon, and nitrogen-helium mixtures. In figure 4.8c, the helium in the nitrogen-helium plasma enhances the image of the exhaust. Pure nitrogen plasma is not readily visible with a camera due to a built-in UV filter in the camera. The exhausts of the all the thrusters indicate overexpansion. This means, as stated earlier, that the exhaust pressure is too low in comparison to the ambient pressure in the chamber.
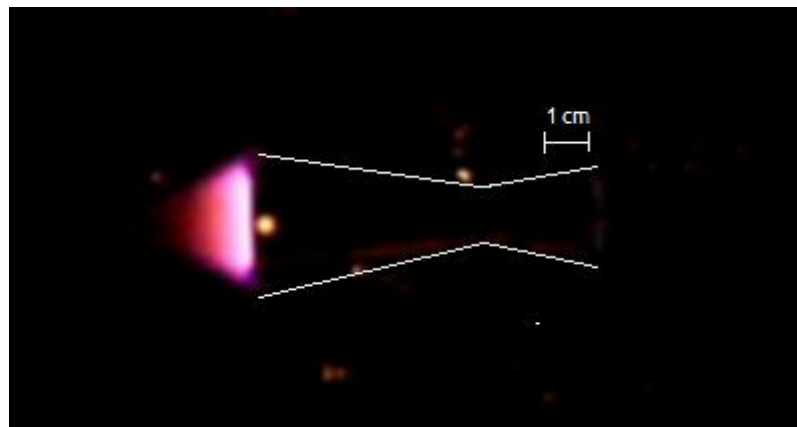
Figure 4.8a: Helium Thruster
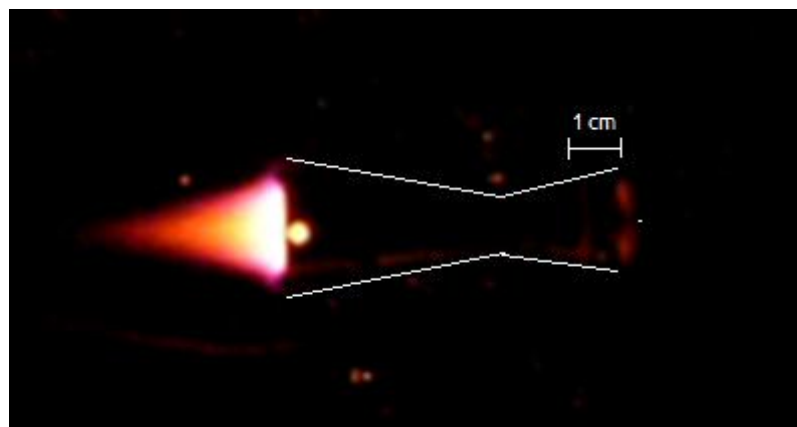
Figure 4.8b: Neon Thruster
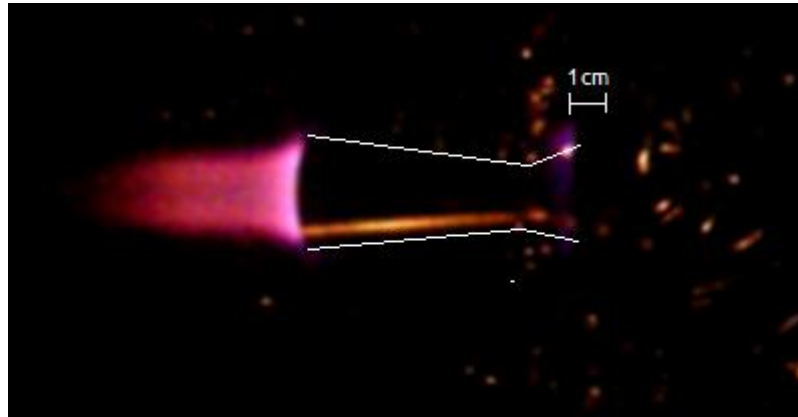
Figure 4.8c: Nitrogen-Helium Thruster

4.2.2 RF Electrothermal Thruster – Nitrogen-Helium Gas Mixture
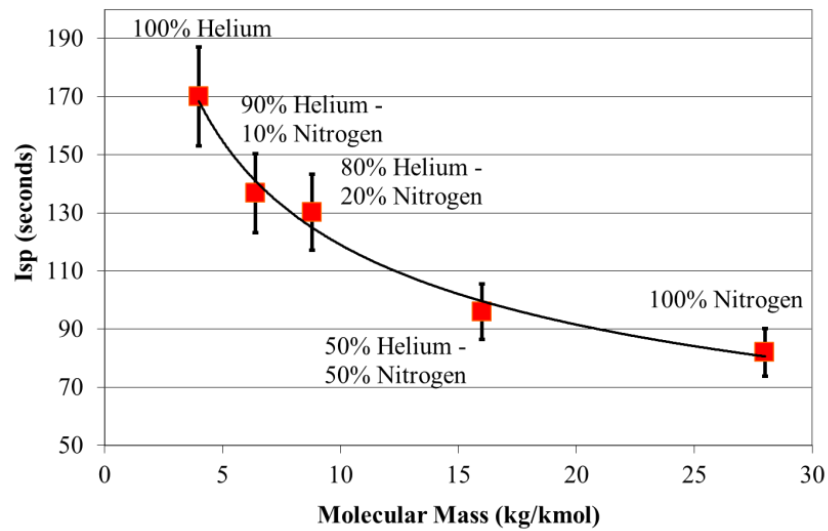
Figure 4.9: Specific Impulse of Nitrogen-Helium Gas Mixture

Figure 4.9 shows the specific impulses for various average molecular masses. The trendline in that figure follows to the exhaust velocity equation 2.66. The graph illustrates that the specific impulse from each propellant is related to the gas mixture ratio. Table 4.6 shows the corresponding force and mass flow rate of the gas mixtures.

Table 4.6: Force and Mass Flow Rate of Nitrogen-Helium Gas Mixtures

Gas Mixture	Force (mN)	Mass Flow Rate (mg/s)
Helium	6.59	3.95
10% N_2 – 90% He	4.89	3.56
20% N_2 – 80% He	4.77	3.74
50% N_2 – 50% He	4.76	5.06
Nitrogen	4.05	5.04
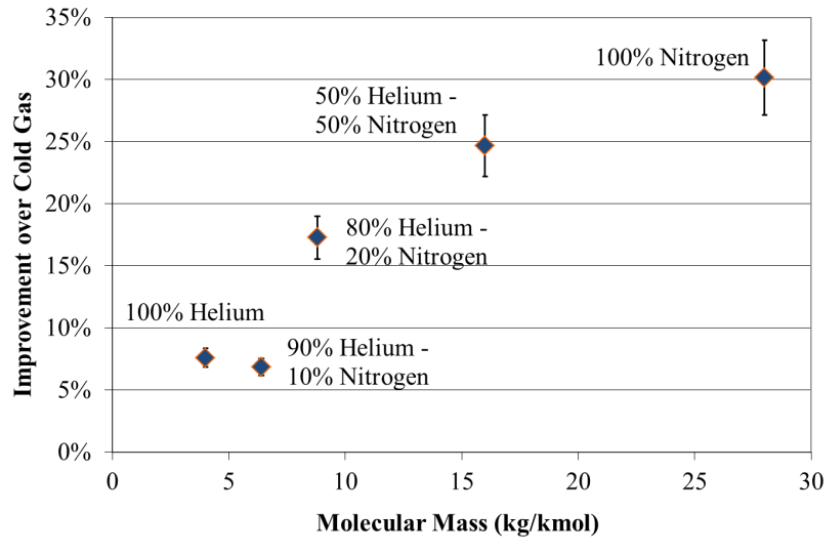
Figure 4.10: Difference in Specific Impulse as a Percentage Compared to Cold Gas.

Figure 4.10 shows the improvement over cold gas for different gas mixtures. The experiment shows that the highest gain in specific impulse is the one with pure nitrogen. This is possibly due to the dissociation of the diatomic structure before ionization. Even though the ionization energy of diatomic nitrogen is 15.52eV, not all the ionized nitrogen would be dissociated. Therefore, both monatomic and diatomic nitrogen ions exist; the

143

free electrons from the ionized nitrogen may help ionize the helium by electron bombardment.

4.3 Hybrid Electric Thruster Experiment

4.3.1 Magnetoplasmadynamic Propulsion for an Augmented-Thrust Mode
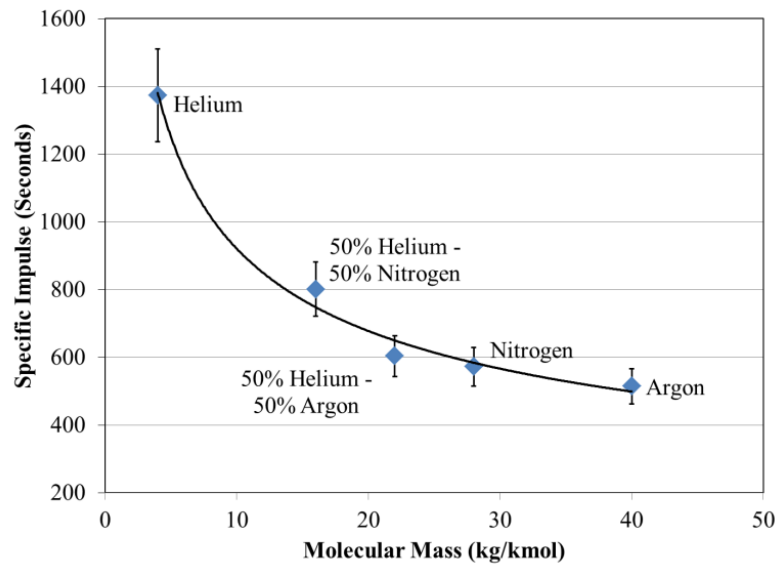
Figure 4.11: Specific Impulse of Argon-Helium and Nitrogen-Helium Gas Mixtures in

MPD Thruster Mode. Mass flow rates - Helium: 21mg/s; 0.5N₂-0.5He: 46mg/s;

0.5Ar-0.5He: 55mg/s; Nitrogen: 40mg/s; Argon: 57mg/s

Figure 4.11 shows the specific impulses of the MPD thruster using argon-helium and nitrogen-helium mixtures with the trendline that follows equation 2.66. The specific impulses of all the gases are above 500 seconds. This shows that the MPD portion of the propulsion system works well. Figure 4.11 illustrates that helium performed the best while argon performed the worst.

The 50% argon – 50% helium mixture has results comparable with argon. This means that there is marginal increase in the specific impulse when mixing with helium. The 50% nitrogen – 50% helium mixture shows a major improvement over the 50% argon – 50% helium mixture.

Table 4.7: Thrust Efficiency (Equation 3.14)

Gas Mixture	Thrust Efficiency
Helium	15.3%
50% N_2 - 50% He	19.8%
50% Ar - 50% He	15.4%
Nitrogen	11.3%
Argon	11.6%

Table 4.7 shows the thrust efficiency (equation 3.14) of the gas mixtures. The thruster efficiency table shows that nitrogen and argon have the lowest efficiencies. The pure helium and the 50% mixture show similar thrust efficiency. The helium – nitrogen mixture, notably, has the highest efficiency. The low efficiencies show that the amount of energy put in does not convert completely into usable thrust, possibly because of the energy transfer loss from the capacitors to the plasma for acceleration.

Table 4.8: Impulse Bit (Equation 2.87)

Gas Mixture	Impulse Bit (μN·s)
Helium	2.84
50% N_2 - 50% He	6.29
50% Ar - 50% He	6.51
Nitrogen	5.05
Argon	5.75

Table 4.8 shows the impulse bit of the gas mixtures. The impulse bit is the force integrated over 200 μs, as seen in equation 2.87. The impulse bit is significant in determining the change in momentum over a period of time. The argon-helium mixture shows the highest impulse bit. For spacecraft, the argon-helium mixture shows the highest change in momentum. In augmented-thrust mode, this mixture helps accelerate the spacecraft to the target velocity.

Table 4.9: Ionization Fraction

Gas Mixture	Alfvèn's Critical Specific Impulse (s)	Specific Impulse from Experiment (s)	Ionization Fraction (%)
Helium	3410	1374	40%
50% N_2 - 50% He	1628	801	49%
50% Ar - 50% He	1232	603	49%
Nitrogen	1116	572	51%
Argon	876	514	59%

Table 4.9 shows how much the gas was ionized by comparing the experimental specific impulses to the theoretical maximum specific impulse from Alfvèn's critical velocity (equation 2.88). The ionization fraction shows how much of the gas has been ionized for electromagnetic propulsion. The results show that helium has the lowest ionization fraction at 40% while argon has the highest at 59%.

4.3.2 Ion Propulsion for Sustain-Thrust Mode
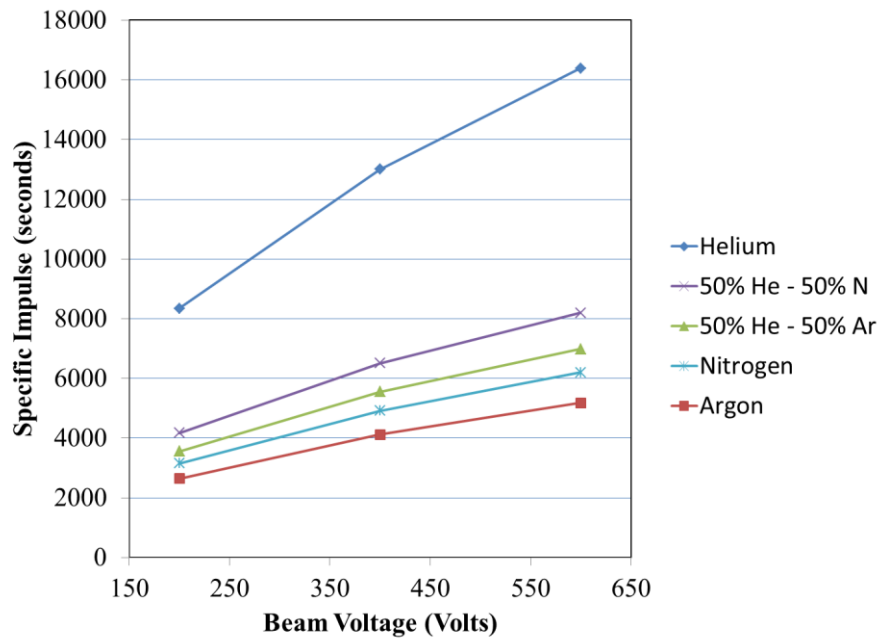
Figure 4.12: Ideal Specific Impulse of Nitrogen-Helium and Argon-Helium Gas Mixtures at Different Net Voltages

Figure 4.12 shows the ideal specific impulse from equation 2.79 of the ES (ion) thruster with the nitrogen-helium and argon-helium gas mixtures. In an ideal chemical or ET thruster, helium is deemed to have the best specific impulse, much higher than the rest of the gases, due to its low molecular mass. As the beam voltage increases for ES propulsion, the specific impulse of each gas mixture increases almost proportionally. A

key issue that arises with helium is its high ionization energy and the associated higher power requirement, which makes it impractical to use it as a main ES propellant for this application.
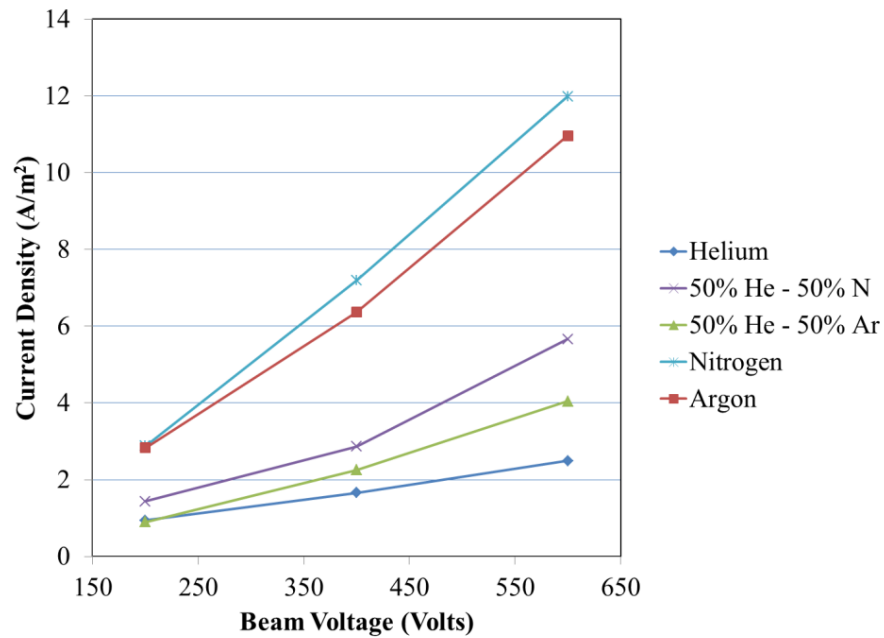
Figure 4.13a: Current Density of Nitrogen-Helium and Argon-Helium Gas Mixtures at Different Net Voltages
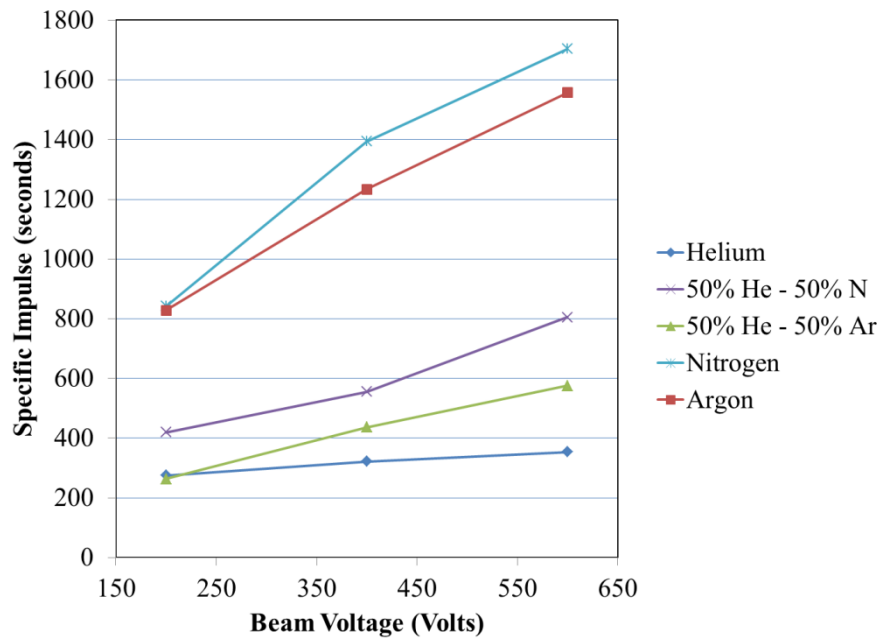
Figure 4.13b: Actual Specific Impulse of Nitrogen-Helium and Argon-Helium Gas

Mixtures at Different Net Voltages

Figure set 4.13 shows the current density and the specific impulse as measured in the laboratory.
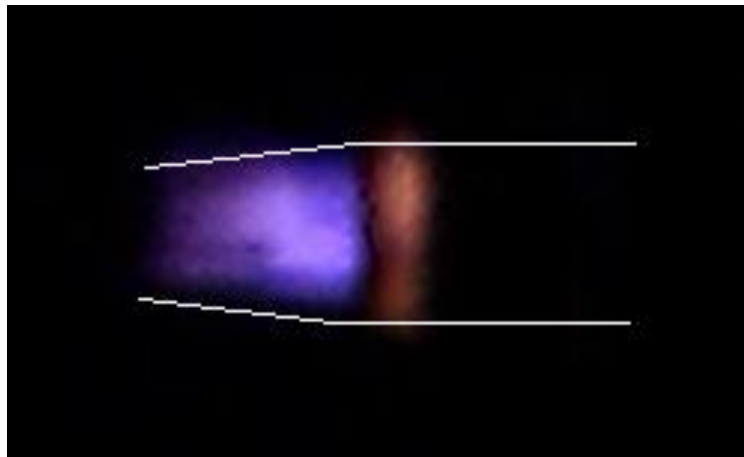
Figure 4.14: Ion Thruster in Operation. The white outline represents the nozzle of the

hybrid thruster.

Figure 4.14 shows the ion thruster in operation. The current density determines the performance of the thruster, as calculated from equation 2.82. The mass flow rate includes the input and ingested mass flow (section 2.11). Figure 4.13b shows that pure nitrogen has the highest specific impulse, at 1703 seconds. Argon also has a high specific impulse, at 1558 seconds. Pure helium has the worst performance in the ion thruster category.

Table 4.10: Power Efficiency at different beam voltages

Gas Mixture	200 V	400 V	600 V
Helium	24.1%	52.8%	71.6%
50% He - 50% N_2	15.7%	42.8%	68.9%
50% He - 50% Ar	14.1%	45.1%	68.9%
Nitrogen	18.3%	52.8%	73.7%
Argon	27.2%	62.7%	75.5%

Table 4.10 shows the power efficiencies at different beam voltages. Note that the power efficiencies do not include the theoretical neutralizer power input, which is approximately 1% of the total performance. The efficiency increases as the beam voltage increases. A significant result is that the mixtures have the lowest efficiencies. At higher beam voltages, the power efficiencies of all the gases are within reasonable values.

Table 4.11: Mass Efficiency at Different Beam Voltages

Gas Mixture	200 V	400 V	600 V	600 V (Normalized to 25 $\mu g \cdot s^{-1}$)
Helium	3.3%	2.5%	2.2%	10.0%
50% He - 50% N_2	10.1%	8.5%	9.8%	22.8%
50% He - 50% Ar	7.4%	7.9%	8.2%	23.2%
Nitrogen	26.7%	28.4%	27.5%	44.1%
Argon	31.4%	30.0%	30.1%	48.2%

Table 4.11 shows the mass efficiencies of the thruster. The normalized column represents the mass efficiency when the mass flow rate is at 25 $\mu g \cdot s^{-1}$ without any ingestion. The efficiencies across the different beam voltages are similar to each other. The mass efficiencies in pure nitrogen and argon are the highest. It was also observed that the mass efficiencies at all three beam voltages are similar to each other. Thus, the mass efficiency is almost independent of the beam voltage.

5 Discussion

Both the RF electrothermal thruster and the hybrid electric thruster have shown promising experimental data results for both manoeuvres and deep-space missions. Table 5.1 shows the delta-V and the kinetic power, both explained in section 2.3, of the spacecraft using different gases and thruster types. The calculations assume that the dry mass of the spacecraft is 10 kg and the total mass of the propellant on board is 5 kg, with methods adopted from Goebel & Katz and Sutton & Biblarz.

Table 5.1: Delta V and Kinetic Power

Type of Engine	Gas Mixture	Mass Flow Rate in Operation	Delta V (m/s)	Kinetic Power (watts)	Thrust Efficiency (%)
RF Electrothermal Thruster	Argon	5 mg/s	256	0.998	13.3
	Nitrogen	4.7 mg/s	326	1.527	20.4
	50% N_2 - 50% He	4.8 mg/s	382	2.141	28.5
Hybrid electric Thruster - Ion Thruster	Argon	25 μg/s	6195	2.918	33.3
	Nitrogen	25 μg/s	6775	3.490	35.5
	50% Ar - 50% He	25 μg/s	2288	0.398	15.7
	50% N_2 - 50% He	25 μg/s	3200	0.779	11.2
Hybrid electric Thruster - MPD Thruster	Argon	500 μg/s	2045	6.40	11.6
	50% Ar - 50% He	500 μg/s	2400	8.80	15.4
	50% N_2 - 50% He	500 μg/s	3182	15.4	19.8
	Helium	200 μg/s	5465	18.4	15.3

In a situation where neither the ion thruster nor the MPD thruster is operational and the only operational electronics in the propulsion system is the RF power plant, the RF electrothermal thruster would provide a sufficient delta-V for the spacecraft. The MPD thruster using helium gas mixtures provides a better delta-V and kinetic power than using

argon or nitrogen. However, the MPD thruster has a high mass flow rate, resulting in a low specific impulse, as seen in equation 2.5, and depleting the fuel supply quickly. Because the thruster requires high power for continuous operation, the MPD thruster should only be used for a quick change in vehicle velocity, as designed.

Table 5.2: Specific Impulses According to Author, with other data from Takao (2007) and Takahashi (2011)

	Gas & Specific Impulse	
Author	Helium	Argon
(Stoute, 2013) 100MHz – 7.5W	170	70
(Takahashi, 2011) 6GHz – 3W	150	N/A
(Takahashi, 2011) 6GHz – 6W	250	N/A
(Takao, 2007) 6GHz – 6W	N/A	79

Table 5.2 compares the results of the electrothermal thrusters from other work. Takahashi and Takao both used 6GHz as their input frequency, and their input power was between 3 and 6 W. In the helium test, Takahashi 3 W thruster results are comparable to the RF electrothermal thruster results with a difference of approximately 10%. Conversely, Takahashi 6 W thruster has a specific impulse of 250 s, a 47% difference from the RF electrothermal thruster. There is a 13% difference in specific impulses between Takao and the RF electrothermal thruster using argon [Takao, 2007; Takahashi, 2011].

The hybrid electric thruster results show that a combination of electrostatic and electromagnetic propulsion is possible. The electromagnetic propulsion provides the high propulsive force for a short period of time for the augmented-thrust phase, while the electrostatic propulsion provides the high specific impulse for deep space missions (sustained-thrust period).

6 Conclusion

In combining the findings from the electrothermal, electromagnetic, and electrostatic propulsion results, the optimal gas mixture is determined experimentally as 50% nitrogen – 50% helium, as seen in chapter four. This result can be explained theoretically because of its high Reynolds number, a satisfactory specific impulse, and a high thrust efficiency. These conclusions are analyzed under a power-limited, mass-unlimited spacecraft. Under a mass-limited, power-limited spacecraft, it is recommended to use pure helium. For a nanosatellite, power is a limiting factor since the maximum power available is 20 W. The recommendations for operating an electric thruster using helium and nitrogen also include the following suggestions:

- Partition the gas mixtures in separate tanks: The nature of the gases mixed in the same tank may not be as expected when the hybrid electric thruster is in operation.
- Directly inject the gases from the two separate tanks into the thruster: This ensures the proper mixture of nitrogen and helium. The injected gases can be independent or in tandem.

In full operation, the thruster operates in low Earth orbit, and the voltages of the screen and acceleration electrodes are +1000 V and -200 V respectively. The size of the thruster is 20 mm in diameter at the exhaust, 10 mm in diameter at the throat. The length of the nozzle is 86 mm. The nozzle is made of glass while the plasma chamber and electrodes are made of copper. The full mechanical details of the thruster are in sections 3.2.2, 3.2.3, and 3.4.2.

The extrapolated performance results at these voltage levels are:

- The specific impulses of the hybrid electric thruster in sustained-thrust (ES) and augmented-thrust (EM) modes are approximately 1400 and 800 seconds, respectively.

- The corresponding propulsive force outputs from both ES and EM modes are 34.3 μN and 3.92 mN, respectively.

- The approximate power inputs required to give this performance are approximately 10 W in sustained-thrust mode and 20 W in augmented-thrust mode. The energy outputs per kilogram for both modes are respectively 94.3 MJ and 0.18 MJ in deep space. The delta-Vs for a 10 kg spacecraft with 5 kg of fuel are 5568 m/s and 3200 m/s. This is sufficient for the spacecraft to leave Low Earth Orbit.

6.1 Future Work

The hybrid electric thruster has been shown to have performance characteristics that would make it a suitable thruster for small spacecraft. The different operational modes of the engine offer versatility in performance. Optimisation of the design of the thruster may improve its performance further. A list of future optimisation work may include:

- Plasma Generation: The hybrid electric thruster could be refitted using alloys to reduce material fatigue and to improve the performance of the thruster. Additionally, the chamber could utilise permanent magnets to improve ionization for the ion thruster mode using electron cyclotron resonance to increase the degree of ionization by electron bombardment.

- Nozzle Design: The nozzle could be refitted with a range of exhaust-throat area ratios to determine the optimal design. As a cost of increasing that ratio, the nozzle might experience structural problems due to the force of the fluid. The nozzle could also be redesigned with a variable-geometry inlet.

- Electrostatic grids: The acceleration grids in the electrostatic thruster mode could be improved by using a different type of mesh grid. The mesh grid could be varied in geometry, porosity, and material choice.

- Addition of gas species: There is a possibility that the addition of another inert gas, such as argon, may further increase the performance of the thruster.

- Exposing the nozzle to solar radiation to reheat the plasma: Doing this may improve the performance of the thruster. The disadvantage of this is that the method only works when there is sufficient solar radiation. For deep-space missions to Mars and beyond, the solar radiation may be too weak for reheating.
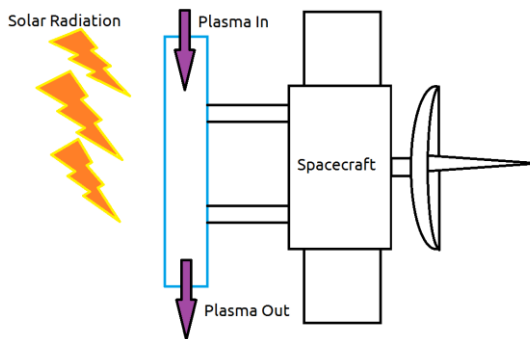
Figure 6.1: Reheating from Solar Radiation

Glossary

The following are selected definitions in the field of electric propulsion in alphabetical order. The definitions are direct quotes from the books used in this study.

Arcjet

An arcjet is also an electrothermal thruster that heats the propellant by passing it though a high current arc in line with the nozzle feed system. While there is an electric discharge involved in the propellant path, plasma effects are insignificant in the exhaust velocity because the propellant is weakly ionized. The I_{sp} is limited by the thermal heating to less than about 700 s for easily stored propellants [Goebel & Katz, 2008].

Electromagnetic Thruster

The acceleration of a body of ionized gas by the interaction of currents driven through the gas with magnetic fields established either by those currents or by external means [Jahn, 1968].

Electrostatic Thruster

Electrostatic thrusters accelerate heavy charged atoms (ions) by means of a purely electrostatic field. Magnetic fields are used only for auxiliary purposes in the ionization chamber [Martinez-Sanchez, 2004a].

Electrothermal Thruster

Electrothermal propulsion comprises all techniques whereby a propellant gas is heated electrically and then expanded through the nozzle to convert its thermal energy to a jet of direct kinetic energy [Jahn 1968].

Hall Thruster

This type of electrostatic thruster utilizes a cross-field discharge described by the Hall effect to generate the plasma. An electric field established perpendicular to an applied magnetic field electrostatically accelerates ions to high exhaust velocities, while the transverse magnetic field inhibits electron motion that would tend to short out the electric field. Hall thruster efficiency and specific impulse is somewhat less than that achievable in ion thrusters, but the thrust at a given power is higher and the device is much simpler and requires fewer power supplies to operate [Goebel & Katz, 2008].

Ion Thruster

Ion thrusters employ a variety of plasma generation techniques to ionize a large fraction of the propellant. These thrusters then utilize biased grids to electrostatically extract ions from the plasma and accelerate them to high velocity at voltages up to and exceeding 10 kV. Ion thrusters feature the highest efficiency (from 60% to >80%) and very high specific impulse (from 2000 to over 10,000 s) compared to other thruster types [Goebel & Katz, 2008].

Magnetoplasmadynamic Thruster

Magnetoplasmadynamic (MPD) thrusters are electromagnetic devices that use a very high current arc to ionize a significant fraction of the propellant, and then electromagnetic forces (Lorentz $\mathbf{J} \times \mathbf{B}$ forces) in the plasma discharge to accelerate the charged propellant. Since both the current and the magnetic field are usually generated by the plasma discharge, MPD thrusters tend to operate at very high powers in order to generate

sufficient force for high specific impulse operation, and thereby also generate high thrust compared to the other technologies described above [Goebel & Katz, 2008].

Pulsed Plasma Thruster

A pulsed plasma thruster (PPT) is an electromagnetic thruster that utilizes a pulsed discharge to ionize a fraction of a solid propellant ablated into a plasma arc, and electromagnetic effects in the pulse to accelerate the ions to high exit velocity. The pulse repetition rate is used to determine the thrust level [Goebel & Katz, 2008].

Resistojet

Resistojets are electrothermal devices in which the propellant is heated by passing through a resistively heated chamber or over a resistively heated element before entering a downstream nozzle. The increase in exhaust velocity is due to the thermal heating of the propellant, which limits the I_{sp} to low levels (<500 s) [Goebel & Katz, 2008].

References

Agle, D. C., NASA's Dawn Spacecraft Fires Past Record for Speed Change, Jet Propulsion Laboratory, Report: 2010-192, National Aeronautics and Space Administration, 7 June 2010.

Ahedo, E., and Merino, M., Two-dimensional supersonic plasma acceleration in a magnetic nozzle, Physics of Plasma, American Institute of Physics, **17**, 2010.

Ahedo, E., and Merino, M., On plasma detachment in propulsive magnetic nozzles, Physics of Plasma, American Institute of Physics, **18**, 2011.

Almeida, D. P., Fontes, A. C., and Godinho, C. F. L., Electron-impact ionization cross section of neon (σ_{n+}, n=1-5), Journal of Physics B: Atomic, Molecular and Optical Physics, **28** (1995), 3335-3345.

Andreussi, T., and Pegoraro, F., Magnetized plasma flows and magnetoplasmadynamic thrusters, Physics of Plasma, American Institute of Physics, **17**, 2010**,** (063507) 1-11.

Anonymous, Chronic Toxic Summary: Hydrazine, Determination of Noncancer Chronic Reference Exposure Levels, Office of Environmental Health Hazard Assessment, California, December 2000.

Anonymous, Dassault Systems and 3DEXPERIENCE Company, CATIA (V5R21), 21 March 2012.

Anonymous, Mechanics of Fibre-Reinforced Composites, University of Cambridge, January 2008.

Anonymous, National Aeronautics and Space Administration, Innovative Engines Glenn Ion Propulsion Research Tames the Challenges of 21st Century Space Travel, Glenn Research Center, Cleveland, Ohio, March 1999.

Anonymous, National Aeronautics and Space Administration, Magnetoplasmadynamic Thrusters, Glenn Research Center, Cleveland, Ohio, November 2004.

Anonymous, PPS-1350 Press Release, Snecma S. A., 6 December 2012.

Anonymous, Xenon, Chemicool Periodic Table, Chemicool.com, 18 October 2012.

Arefiev, A. V., and Breizman, B., N., Theoretical components of the VASIMR plasma propulsion concept, Physics of Plasma, American Institute of Physics, **11**, 2004.

Baity, F. W., Barber, G. C., Carter, M. D., Goulding, R. H., Sparks, D. O., Chang-Diaz, F. R., McCaskill, C. E., and Squire, J. P., Design of RF Systems for the RTD Mission VASIMR, Oak Ridge National Laboratory, Oak Ridge, Tennessee, American Institute of Physics, Conference Proceedings, **485**, 417 (1999).

Baker, J., and Morgan, O. T., Effect of Paramagnetism and Diamagnetism on Theoretical Rocket Performance, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **20** (6) November 2004**,** 1012-1017.

Bar-Meir, G., Basics of Fluid Mechanics, Free Software Foundation, Massachusetts, 2011.

Bartosiewicz, Y., Proulx, P., and Mercadier, Y., A self-consistent two-temperature model for the computation of supersonic argon plasma jets, Journal of Physics D: Applied Physics, Institute of Physics Publishing, **35**, 2002**,** 2139-2148.

Batishchev, O. V.,, Minihelicon Plasma Thruster, IEEE Transactions on Plasma Science, IEEE, **37** (8) 2009.

Bellan, P. M., Fundamentals of Plasma Physics, California Institute of Technology, Pasedena, California, 2004.

Biloiu, I. A., and Scime, E. E., Ion acceleration in Ar–Xe and Ar–He plasmas. I. Electron energy distribution functions and ion composition, Physics of Plasma, American Institute of Physics, **17,** 2010.

Bose, T. K., High Temperature Gas Dynamics, Springer, Berlin Heidelberg, 2004.

Breizman, B. N., Tushentsov, M. R., and Arefiev, A. V., Magnetic nozzle and plasma detachment model for a steady-state flow, Physics of Plasma, American Institute of Physics, **15**, 2008.

Brown, S., Centre for Research in Earth and Space Science, York University, Private Communication, 2010-2012.

Browning, J., Lee, C., Plumlee, D., Shawver, S., Loo, S. M., Yates, M., McCrink, M., Taff, T., A Miniature Inductively Coupled Plasma Source for Ion Thrusters, IEEE Transactions on Plasma Science, IEEE, **39** (11) November 2011**,** 3187-3195.

Burkhart, J. A., Magnetoplasmadynamic Arc Thrusters, U. S. Patent Office, 29 May 1973, U. S. Patent Number 3735591.

Cassady, R. J., Hoskins, W. A., Campbell, M., and Rayburn, C., A Micro Pulsed Plasma Thruster (PPT) for the "Dawgstar" Spacecraft, Institute of Electrical and Electronics Engineers, 2000.

Cassibry, J. T., Comparison of Directly and Inductively Coupled Pulsed Electromagnetic Thrusters, IEEE Transactions on Plasma Science, IEEE, **36** (5) October 2008**,** 2180-2188.

Chang Díaz, F.R., Squire, J.P., Ilin, A.V., McCaskill, G.E., Nguyen, T.X., Winter, D.S., Petro, A.J., Goebel, G.W., Cassady, L.D., Stokke, K.A., Dexter, C.E., Graves, T.P., Amador Jr., L., and George, J.A., The Development of the VASIMR Engine, International Conference on Electromagnetics in Advanced Applications, Torino, Italy, 13 Sep 1999.

Charles, C., Plasmas for spacecraft propulsion, Journal of Physics D: Applied Physics, Institute of Physics Publishing, **42**, 2009.

Chen, S.-L., Sekiguchi, T., Instantaneous Direct-Display System of Plasma Parameters by Means of Triple Probe, Journal of Applied Physics, American Institute of Physics, **36** (8) August 1965**,** 2363-2375.

Chianese, S. G., and Micci, M. M., Microwave Electrothermal Thruster Chamber Temperature Measurements and Performance Calculations, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **22** (1) January 2006**,** 31-37.

Choueiri, E., A Critical History of Electric Propulsion: The First 50 Years (1906–1956), Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **20** (2) March 2004**,** 193-203.

163

Choueiri, E., and Ziemer, J. K., Scaling laws for electromagnetic pulsed plasma thrusters, Plasma Sources and Technology, Institute of Physics Publishing, **15**, 2001**,** 395-405.

Choueiri, E., Scaling of Thrust in Self-Field Magnetoplasmadynamic Thrusters, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **20** (5), September-October 1998, 744-753.

Chu, K.-Y., Kubota, K., Funaki, I., and Okuno, Y., Numerical Investigations of a Repetitively Pulsed MPD Thruster, IEEJ Transactions on Electrical and Electronic Engineering, **7**, 2012**,** 234-239.

Coletti, M., Simple Thrust Formula for an MPD Thruster with Applied-Magnetic Field from Magnetic Stress Tensor, American Institute of Aeronautics and Astronautics, 2007, 5284.

Coletti, M., Guarducci, F., Gabriel, S. B., A micro PPT for Cubesat application: Design and preliminary experimental results, Acta Astronautica, Elsevier, **69**, 2011**,** 200-208.

Conrads, H., and Schmidt, M., Plasma generation and plasma sources, Plasma Sources and Technology, Institute of Physics Publishing, **9**, 2000**,** 441-454.

Damaren, C. J., Spacecraft Dynamics and Control II Lecture Notes, University of Toronto, Ontario, Winter 2010.

Darmofal, D., Aerodynamics Lecture Notes, Massachusettes Institute of Technology, Massachusettes, Fall 2005.

Demetriou, M. A., and Gatsonis, N. A., Control of Unsteady Electromagnetic acceleration in Parallel Plate Channels, Worchester Polytechnic Institute, n.d.

Dinklage, A., Klinger, T., Marx, G., Schweikhard, L., Plasma Physics, Springer, Berlin Heidelberg, 2005.

Domonkos, M. T., Gallimore, A. D., Myers, R. M., Thompson, E., Preliminary Pulsed MPD Thruster Performance, 31st AIAA Joint Propulsion Conference, San Diego, California, June 1995.

Dryden, H. L., Masek,T. D., , Electron Bombardment Ion Engine, U. S. Patent Office, 10 Oct 1967, U. S. Patent Number 3345820.

Eckman, R., Lawrence, B., Gatsonis, N. A., and Pencil, E. J., Triple Langmuir Probe Measurements in the Plume of a Pulsed Plasma Thruster, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **17** (4) July 2001**, 762-771.

Edamitsu, T., and Tahara, H., Performance Measurement and Flowfield Calculation of an Electrothermal Pulsed Plasma Thruster with a Propellant Feeding Mechanism, 29th International Electric Propulsion Conference, Princeton University, 31 Oct 2005.

Edamitsu, T., Hirofumi, A., Matsumoto, A., and Tahara, H., Research and Development of a Pulsed Plasma Thruster in Osaka University, 29th International Electric Propulsion Conference, Princeton University, 31 Oct 2005.

Eom, G. S., Bae, I. D., Cho, G., Hwang, Y. S., and Choe, W., Helicon plasma generation at very high radio frequency, Plasma Sources and Technology, Institute of Physics Publishing, **15**, 2001**,** 417-422.

Fitzpatrick, R., Plasma Physics, University of Texas at Austin, 31 March 2011.

Foing, B. H., Racca, G. D., Marini, A., Heather, D. J., Koschny, D., Grande, M., Huovelin, J., Keller, H. U., Nathues, A., Josset, J. L., Malkki, A., Schmidt, W., Noci, G., Birkl, R., Iess, L., Sodnik, Z., McManamon, P., SMART-1 Mission to the moon: Technology and Science Goals, Advance Space Research, Elsevier, **31** (11) 2003, 2323-2333.

Foing, B. H., Racca, G. D., SMART-1 Team, The ESA SMART-1 mission to the Moon with solar electric propulsion, Advance Space Research, Elsevier, **23** (11) 1999, 1865-1870.

Funaki, I., Nishiyama, K., Kuninaka, H., Toki, K., Shimizu, Y., and Toki, H., 20mN-class Microwave Discharge Ion Thruster, 27th International Electric Propulsion Conference, Pasadena, CA, 15 Oct 2001.

Gilland, J. H., System-Level Models of Self- and Applied-Field MPD Thrusters, American Institute of Aeronautics and Astronautics, Joint Propulsion Conference, Fort Lauderdale, FL, 11-14 July 2004.

Glover, T., VASIMR VX-200 Performance and VASIMR VX-200 Performance and Near-term SEP Capability for Unmanned Mars Flight, Ad Astra Rocket Company, 19 Jan 2011.

Goebel, D. M., Analytical Discharge Model for RF Ion Thrusters, IEEE Transactions on Plasma Science, IEEE, **36** (5) October 2008**,** 2111-2121.

Goebel, D. M., Jet Propulsion Laboratory, Private Communication, 24 Oct 2012.

Goebel, D. M., and Katz, I., Fundamentals of Electric Propulsion: Ion and Hall Thrusters, Wiley, New Jersey, 2008.

Goedheer, W. J., Lecture notes on radio-frequency discharges, dc potentials, ion and electron energy distributions, Plasma Sources and Technology, Institute of Physics Publishing, **9**, 2000**,** 507-516.

Guarducci, F., Paccani, G., and Lehnert, J., Quasi steady MPD performance analysis, Acta Astronautica, Elsevier, **68**, 2011**,** 904-914.

Haag, D., Fertig, M., Herdrich, G., Kurtz, H., Winter, M., and Auweter-Kurtz, M., Experimental Investigations of Magnetoplasmadynamic Thrusters with Coaxial Applied Magnetic Field, 30th International Electric Propulsion Conference, Florence, Italy, 17 Sep 2007.

Hammond, E. P., Mahesh, K., Moin, P., A Numerical Method to Simulate Radio-Frequency Plasma Discharges, Journal of Computational Physics, Elsevier, **176**, 2002**,** 402-429.

Haynes, W. M., CRC Handbook of Chemistry and Physics, CRC Press/Taylor and Francis, Boca Raton, FL., 92nd Edition (Internet Version 2012), Bond Dissociation Energies in Diatomic Molecules.

Haynes, W. M., CRC Handbook of Chemistry and Physics, CRC Press/Taylor and Francis, Boca Raton, FL., 92nd Edition (Internet Version 2012), Ionization Energies of Atoms and Atomic Ions.

Herdrich, G., Auweter-Kurtz, M., Fertig, M., Nawaz, A., and Petkow, D., MHD flow control for plasma technology applications, Vacuum, Elsevier, **80**, 2006**,** 1167-1173.

Horowitz, P., and Hill, W., The Art of Electronics, The Press Syndicate of the University of Cambridge, Cambridge, 2001.

Howard, J., Introduction to Plasma Physics, Australian National University, 2002.

Hoyt, R. P., Magnetic Nozzle Design for High-Power MPD Thrusters, 29th International Electric Propulsion Conference, Princeton University, 31 Oct 2005.

Hrbud, I., Kemp, G. E., Yan, A. H., and Gedrimas, J. G., Review of RF Plasma Thruster Development, 30th International Electric Propulsion Conference, Florence, Italy, 17 Sep 2007.

Hussaini, M. M., and Korte, J. J., Investigation of Low-Reynolds-Number Rocket Nozzle Design Using PNS-Based Optimization Procedure, NASA Technical Memorandum 110295, 1 Nov 1996.

Hutchinson, I., and Freidberg, J., Introduction to Plasma Physics, Massachusetts Institute of Technology, Massachusetts, Fall 2003.

Inutake, M., Ando, A., Hattori, K., Tobari, H., Makita, T., Shibata, M., Kasashima, Y. and Komagome, T., Generation of supersonic plasma flows using an applied-field

MPD arcjet and ICRF heating, Plasma Physics and Controlled Fusion, IOP Publishing, **49**, 2007**.**

Ito, T., Gascon, N., Crawford, W. S., and Cappelli, M. A., Ultra-Low Power Stationary Plasma Thruster, 29th International Electric Propulsion Conference, Princeton University, 31 Oct 2005.

Itikawa, Y., Cross Sections for Electron Collisions with Nitrogen Molecules, Journal of Physical and Chemical Reference Data, **35** (1) 2006.

Ivanov, M. S., Markelov, G. N., Ketsdever, A. D., and Wadsworth, D. C., Numerical Study of Cold Gas Micronozzle Flows, American Institute of Aeronautics and Astronautics, 1999, 1-15.

Jahn, R. G., Physics of Electric Propulsion, McGraw-Hill, New York, 1968.

Janson, S. W., Helvajian, H., Hansen, W. W., and Lt. Lodmell, J., Microthrusters for Nanosatellites, The Second International Conference on Integrated Micro Nanotechnology for Space Applications, Pasadena, CA, 11 Apr 1999.

Kagaya, Y., and Tahara, H., Swirl Acceleration in a Quasi-Steady MPD Thruster by Applied Magnetic Nozzle, 29th International Electric Propulsion Conference, Princeton University, 31 Oct 2005.

Katz, I., Anderson, J. R., Polk, J. E., and Brophy, J. R., One-Dimensional Hollow Cathode Model, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **19** (4) July 2003**,** 595-600.

Ketsdever, A. D., Clabough, M. T., Gimelshein, S. F., Alexeenko, A., Experimental and Numerical Determination of Micropropulsion Device Efficiencies at Low

Reynolds Numbers, , American Institute of Aeronautics and Astronautics, **43** (3) March 2005**,** 633-641.

Kim, Y.-K., Irikura, K. K., Rudd, M. E., Ali, M. A., and Stone, P. M., Electron-Impact Cross Sections for Ionization and Excitation, National Institute of Standards and Technology, August 2004.

Komerath, N., Rocket Propulsion, Georgia Institute of Technology, Georgia, 2004.

Koizumi, H., and Kuninaka, H., Miniature Microwave Discharge Ion Thruster Driven by 1 Watt Microwave Power, Journal of Power and Propulsion, **26** (3) May-June 2010, 601-604.

Krulle, G., Auweter-Kurtz, M., and Sasoh, A., Technology and Application Aspects of Applied Field Magnetoplasmadynamic Propulsion, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **14** (5) September1998**,** 754-763.

Lankford, D. W., A Study of Electron Collision Frequency in Air Mixtures and Turbulent Boundary, Technical Report, Air Force Systems Command, Kirtland Air Force Base, New Mexico, October 1972.

Laframboise, J. G., Plasma Physics/Special Topics Lecture Notes, York University, Toronto, 1996.

LaPointe, M. R., and Mikellides, P. G., High Power MPD Thruster Development at the NASA Glenn Research Center, 37th Joint Propulsion Conference, Salt Lake City, Utah, 8 Jul 2001.

Lieberman, M., Principles of Plasma Discharge, UC Berkley, Berkley, 2003.

Lingwei, Z., Yu, L., Zheng, W., and Junxue, R., Numerical Simulation of Ion Extraction Through Ion Thruster Optics, Plasma Science and Technology, Institute of Physics Publishing, **21** (1) Feb 2010**,** 103-108.

Liskovskiy, V. A., and Yegroenkov, V. D., Rf breakdown of low-pressure gas and a novel method for determination of electron-drift velocities in gases, Journal of Physics D: Applied Physics, Institute of Physics Publishing, **31**, 1998**,** 3349-3357.

Longmeir, B. W., and Hershkowitz, N., "Electrodeless" Plasma Cathode for Neutralization of Ion Thrusters, 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Tucson, Arizona, 10 Jul 2005.

Lubey. D. P., Bilen, S. G., Micci, M. M., and Tauney, P.-Y., Design of the Miniature Microwave-Frequency Ion Thruster, 32nd International Electric Propulsion Conference, Wiesbaden, Germany, 11 Sep 2011.

Luna, H., Michael, M., Shah, M. B., Johnson, R. E., Latimer, C. J., and McConkey, J. W., Dissociation of N_2 in Capture and Ionization Collisions with Fast H+ and N+ Ions and Modeling of Positive Ion Formation in the Titan Atmosphere, Journal of Geophysical Research, American Geophysical Union, **108** (E4, 5003), 2003.

Martinez-Sanchez, M., Rocket Propulsion, Massachusetts Institute of Technology, Massachusetts, 2004, In Text: 2004b.

Martinez-Sanchez, M., Space Propulsion, Massachusetts Institute of Technology, Massachusetts, 2004, In Text: 2004a.

Martinez-Sanchez, M., Massachusetts Institute of Technology, Private Communication, 2010-2012.

171

Miller, J., MiXI Thruster Tungsten and Thoriated Tungsten Testing, American Institute of Aeronautics and Astronautics, n.d.

Miyasaka, T., Katsurayama, H., and Fujiwara, T., Effect of applied magnetic field on Arcjet Thruster, 27th International Electric Propulsion Conference, Pasadena, CA, 15 Oct 2001.

Mueller, J., Ziemer, J., Hofer, R., Wirz, R., and O'Donnell, T., A Survey of Micro-Thrust Propulsion Options for Microspacecraft and Formation Flying Missions, Cube Sat - 5th Annual Developers Workshop, 2008.

Mulcahy, J. M., Browne, D. J., Stanton K. T., Chang-Diaz, F. R., Cassady, L. D., Berisford, D. F., and Bengston, R. D. , Heat flux estimation of a plasma rocket helicon source by solution of the inverse heat conduction problem, International Journal of Heat and Mass Transfer, Elsevier, **52**, 2009**,** 2343-2357.

Murawski, K., Numerical solutions of magnetohydrodynamic equations, Technical Sciences, Bullentin of The Polish Academy of Sciences, **59** (1) 2011.

Murawski, K., and Tanaka, T., Modern Numerical Schemes for Solving Magnetohydrodynamic Equations, Astrophysics and Space Science. Kluwer Academic Publishers, **254**, 1997**,** 187-210.

Myers, R. M., Kelly, A. J., and Jahn, R. G., Energy Deposition in Low-Power Coaxial Plasma Thrusters, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **7** (5) 1990**,** 732-739.

Nakayama, Y., Funaki, I., and Kuninaka, H., Sub-Milli-Newton Class Miniature Microwave Ion Thruster, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **23** (2) March 2007**,** 495-499.

Nawaz, A., Auweter-Kurtz, M., Herdrich, G., and Kurtz, H., Experimental Setup of a Pulsed MPD Thruster at IRS, European Conference for Aerospace Science, 4 Jul 2005.

Norton, R. L., Machine Design - An Integrated Approach, Prentice-Hall, New Jersey, 2000.

Ogawa, H., and Boyce, R. R., Physical Insight into Scramjet Inlet Behaviour via Multi-Objective Design Optimisation, American Institute of Aeronautics and Astronautics, **50** (8) 2012, 1773-1783.

Olson, B. J., Navier-Stokes CFD solver, Department of Aeronautics and Astronautics, Stanford Univeristy, 2009.

Palaszweki, B., Electric Propulsion for Future Space Missions Part 1 (Presentation), NASA Glenn Research Center at Lewis Field, n.d.

Palmer, D. D., Walker, M. L. R., Manete, M., Carlsson, J., Bramanti, C., and Pavarin, D., Performance Analysis of a Medium-Power Helicon Thruster, American Institute of Aeronautics and Astronautics, Joint Propulsion Conference & Exhibit, 21-23 July 2008.

Phelps, A. B., Cross Sections and Swarm Coefficients for Nitrogen Ions and Neutrals in N_2 and Argon Ions and Neutrals in Ar for Energies from 0.1 eV to 10 keV, J. Phys. Chem. Ref. Data 20, American Institute of Physics, **557** (1991), 557-574.

Anonymous, Phillips Semiconductors Author, BLF245 VHF Power MOS Transistor Manual, Phillips, 2003.

Polk, J. E., Kakuda, R. Y., Anderson, J. R., Brophy, J. R., Rawlin, V. K., Sovey, J., Hamley, J., In-Flight Performance of the NSTAR Ion Propulsion System on the Deep Space One Mission, IEEE, **4** 2000, 123-148.

Power, J. L., Microwave Electrothermal Propulsion for Space, IEEE Transactions on Microwave Theory and Techniques, IEEE, **40** (6) June 1992, 1179-1191.

Quine, B. M., Seth, R. K., and Zhu, Z. H., A free-standing space elevator structure: A practical alternative to the space tether, Acta Astronautica, **65** (2009), 365-375.

Quine, B., Earth and Space Science and Engineering, Private Communication, 15 May 2010.

Radmilovic-Radjenovic, M. and Lee, J. K., Modeling of breakdown behavior in radio-frequency argon discharges with improved secondary emission model, Physics of Plasmas, American Institute of Physics, **12**, 2005.

Sankaran, K., Choueiri, E. Y., and Jardin, S. C., Comparison of Simulated Magnetoplasmadynamic Thruster Flowfields to Experimental Measurements, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **21** (1) 2005, 129-138.

Sankaran, K., Martinelli, L., Jardin, S. C., and Choueiri, E. Y., A flux-limited numerical method for solving the MHD equations to simulate propulsive plasma flows, International Journal for Numerical Methods in Engineering, **53**, 2002**,** 1415-1432.

Segal, C., The Scramjet Engine Processes and Characteristics, Cambridge University Press, New York, 2009.

Serway, R. A., Jewett, J. W., Physics for Scientists and Engineers, Thomson Brooks/Cole, 2004.

Shamrai, K. P., Virko, V. F., and Kirichenko, G. S., Modeling Helicon Antenna Coupling to a Short Dense-Plasma Column, n.d.

Smart, M. K., Scramjet Inlets, North Atlantic Treaty Organization, RTO-EN-AVT-185, September 2010.

Sou, H., Takao, Y., Noutsuka, T., Mori, Y., Uemura, K., and Nakashima, H., A study of plasma propulsion system with RF heating, Vacuum, Elsevier, **59**, 2000**,** 73-79.

Squire, J. P., Olsen, C. S., Chang Díaz, F. R., Cassady, L. D., Longmier, B. W., Ballenger, M. G., Carter, M. D., Glover, T. W., McCaskill, G. E., and Bering , III, E. A., VASIMR® VX-200 Operation at 200 kW and Plume Measurements: Future Plans and an ISS EP Test Platform, The 32nd International Electric Propulsion Conference, Wiesbaden, Germany, 11 Sep 2011.

Stein, W. B., Alexeenko, A. A., and Hrbud, I., Performance Modeling of an RF Coaxial Plasma Thruster, 43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Cincinnati, OH, 8 Jul 2007.

Stein, W. B., Alexeenko, A. A., and Hrbud, I., RFCCD Microthruster Performance via Numerical Simulation, 46th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 7 Jan 2008.

Stein, W. B., Alexeenko, A. A., and Hrbud, I., Performance Modeling of a Coaxial Radio-Frequency Gas-Discharge Microthruster, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **24** (5) Sep-Oct 2008**,** 1007-1017.

Stoute, C. A. B., Quine, B., Brown, S., Khazarie, S., RF Electrothermal Propulsion Testing Using N2-He Gas Mixture in Low-Temperature Plasma, 12th Annual Australian Space Science Conference, National Space Society of Australia Ltd., Melbourne, 24 – 26 September 2012. Published in conference's proceedings, June 2013, 978-0-9775740-6-3.

Stuhlinger, E., Ion Propulsion for Space Flight, McGraw-Hill, New York, 1964.

Sutton, G. P., and Biblarz, O., Rocket Propulsion Elements, Wiley, New Jersey, 2010.

Takahashi, T., Takao, Y., Eriguchi K., and Ono, K., Numerical and experimental study of microwave-excited microplasma and micronozzle flow for a microplasma thruster, Physics of Plasmas, American Institute of Physics, **16,** 083505, 2009.

Takahashi, T., Takao, Y., Eriguchi K., and Ono, K., Microwave-excited microplasma thruster with helium and hydrogen propellants, Physics of Plasmas, American Institute of Physics, **18**, 063505, 2011.

Takao, Y., and Ono, K., A miniature electrothermal thruster using microwave-excited plasmas: a numerical design consideration, Plasma Sources and Technology, Institute of Physics Publishing, **15**, 2006**,** 211-227.

Takao, Y., Eriguchi, K., and Ono, K., A miniature electrothermal thruster using microwave-excited microplasmas: Thrust measurement and its comparison with numerical analysis, Journal of Applied Physics, American Institute of Physics, **101**, 2007.

Thorsteinsson, E. G., and Gudmundsson, A global (volume averaged) model of a nitrogen discharge: I. Steady state, Plasma Source Science and Technology, IOP Publishing, **18** (2009), 1-15.

Toki, K., Shinohara, S., Tanikawa, T., and Shamrai, K., Small helicon plasma source for electric propulsion, Thin Solid Films, Elsevier, 19 Sep 2005, 597-600.

Toki, K., Shinohara, S., Tanikawa, T., Funaki, I., and Shamrai, K., Preliminary Investigation of Helicon Plasma Source for Electric Propulsion Applications, International Electric Propulsion Conference, 03-0168.

Van Noord, J. L., NEXT Ion Thruster Thermal Model, 43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, 8 Jul 2007.

Vieira, R., Pham-Huu, C., Keller N., and Ledoux, M. J., New carbon nanofiber/graphite felt composite for use as a catalyst support for hydrazine catalytic decomposition, ChemCool, Royal Society of Chemistry, 3 Apr 2002.

Wendt, J. F., Computational Fluid Dynamics, Springer, Berlin Heidelberg, 2009.

Whalen, M. V., Low Reynolds Number Nozzle Flow Study, NASA Technical Memorandum 100130, July 1987.

Wirz, R., Computational Modeling of a Miniature Ion Thruster Discharge, 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Tucson, Arizona, 10 Jul 2005.

Wirz, R., Sullivan, R., Przybylowski, J., and Silva, M., Discharge Hollow Cathode and Extraction Grid Analysis for the MiXI Ion Thruster, 42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Sacremento, California, 1 Jul 2006.

Yamamoto, N., Hirokazu, M., Kataharada, H., Nakashima, H., and Takao, Y., Antenna Configuration Effects on Thrust Performance of Miniature Microwave Discharge Ion Engine, Journal of Power and Propulsion, American Institute of Aeronautics and Astronautics, **22** (4) July 2006, 925-928.

Yamamoto, N., Kondo, S., Chikaoka, T., Nakashima, H., and Masui, H., Effects of magnetic field configuration on thrust performance in a miniature microwave discharge ion thruster, Journal of Applied Physics, American Institute of Physics, **102**, 2007**,** (122304) 1-6.

Yang, J., Xu, Y., Meng, Z., and Yang, T., Effect of applied magnetic field on a microwave plasma thruster, Physics of Plasma, American Institute of Physics, **15**, 2008.

Ziemer, J. K., and Choueiri, E. Y., Dimensionless Performance Model for Gas-Fed Pulsed

Plasma Thrusters, American Institute of Aeronautics and Astronautics, AIAA-98-

3661, 13-15 July 1998.

Appendix A: Saha Equation Derivation

We derive the Saha equation derivation (section 2.5.1) following *Laframboise 1996* lecture notes. Inside a cube of size L with periodic boundary conditions on the walls, neutral atoms with a density n_n are place with a corresponding temperature *T*. Some of them ionize at this temperature, producing equal amount of ions (n_i) and electrons (n_e), assuming that neither neutrals nor ions possess internal degree of freedom [Laframboise 1996].

In local thermodynamic equilibrium, the (un-normalized) probability of finding an individual neutral in any non-degenerate state of the total energy, $\mathcal{E}$, is [Laframboise 1996]

$$e^{-\frac{\mathcal{E}}{k_B T}}.$$

(A. 1)

The total probabilities of the ionized and neutral configurations must be solved. If the atom is not ionized, it is considered to be a quantum-mechanical free particle. If the atom is ionized, the ion and electron are treated as free particles [Laframboise 1996].

If a neutral atom has a mass *m* and momentum $\mathbf{p}$, then $\mathcal{E} = p^2 \cdot (2m)^{-1}$ and the probability of finding it with this momentum is

$$e^{-\frac{\mathbf{p}^2}{2m k_B T}}.$$

(A. 2)

And the probability of finding the ion and electron with these momenta is

$$e^{-\frac{eE_i}{k_B T}} \cdot e^{-\frac{\mathbf{p}_i^2}{2m_i k_B T}} \cdot e^{-\frac{\mathbf{p}_e^2}{2m_e k_B T}}.$$

(A. 3)

The atoms are considered to be confined within the cubic box which has the length L under periodic boundary conditions for each atom's wave function on all box surfaces. The possible values of the momentum vector are given by $\mathbf{p} = h\mathbf{k}/2\pi$ with the wave vector $\mathbf{k} = 2\pi \cdot L^{-1}(N_x, N_y, N_z)$, where N_x, N_y, and N_z are integers. To each distinct triad of these integers corresponds a single wave function (ignoring spins) [Laframboise 1996].

If L is large, then these states are closely spaced in momentum and for the total number of states in the momentum interval $d p_x d p_y d p_z$

$$d^3 \mathbf{N} = dN_x dN_y dN_z \rightarrow \frac{L^3 d p_x d p_y d p_z}{h^3} \tag{A.4}$$

The total probability for the non-ionized state is then [Laframboise 1996]

$$P_n = \sum_{N_x,N_y,N_z} e^{-\frac{\mathcal{E}_{Nx,Ny,Nz}}{k_B T}} \rightarrow \iiint_{-\infty}^{\infty} e^{-\frac{\mathcal{E}_{Nx,Ny,Nz}}{k_B T}} d^3 \mathbf{N}, \tag{A.5}$$

$$= \frac{L^3}{h^3} \iiint e^{-\frac{\mathbf{p}^2}{2mk_B T}} d^3 \mathbf{p},$$

$$= \left(\frac{2\pi m k_B T}{h^2}\right)^{\frac{3}{2}} L^3.$$

The total probability for the ionized state is [Laframboise 1996]

$$P_{ie} = \sum_{N_{x_i},N_{y_i},N_{z_i}} \sum_{N_{xe},N_{ye},N_{ze}} e^{-\frac{\mathcal{E}_{N_{x_i},[\dots],N_{ze}}}{k_B T}} \rightarrow \iiint_{-\infty}^{\infty} \iiint_{-\infty}^{\infty} e^{-\frac{\mathcal{E}_{Nx,Ny,Nz}}{k_B T}} d^3 \mathbf{N}_i d^3 \mathbf{N}_e, \tag{A.6}$$

$$= e^{-\frac{eE_i}{k_B T}} \left(\frac{L^3}{h^3} \iiint e^{-\frac{\mathbf{p}_i^2}{2m_i k_B T}} d^3 \mathbf{p}_i\right)\left(\frac{L^3}{h^3} \iiint e^{-\frac{\mathbf{p}_e^2}{2m_e k_B T}} d^3 \mathbf{p}_e\right),$$

$$= e^{-\frac{eE_i}{k_B T}} L^6 \left(\frac{2\pi m_i k_B T}{h^2}\right)^{\frac{3}{2}} \left(\frac{2\pi m_e k_B T}{h^2}\right)^{\frac{3}{2}}.$$

The relative probabilities of finding a single atom ionized or non-ionized are then P_{ie} and P_n. The ratio P_{ie}/P_n then also equals the relative proportions of ion-electron pairs and atoms in or box. Also $P_{ie} = P_i P_e$, and $P \propto nL^3$ for e, i, and n. Thus the probabilities yield [Laframboise 1996]

$$\frac{P_{ie}}{P_n} = \frac{P_i P_e}{P_n} = \frac{(n_i L^3)(n_e L^3)}{n_n L^3}. \tag{A.7}$$

Therefore the Saha's equation in its simplest form is

$$\frac{n_i n_e}{n_n} = \left(\frac{2\pi m_e k_B T}{h^2}\right)^{\frac{3}{2}} e^{-\frac{eE_i}{k_B T}}. \tag{A.7}$$

Equation A.7, the simple Saha's equation, is derived under local thermodynamic equilibrium, such as a thermal plasma. According to J. Meichner in *Dinklage et al.*, in a low-temperature plasma where the temperature of the electron is much higher than the heavy particles, the Saha's equation is derived under partial thermodynamic equilibrium. Under a partial thermodynamic equilibrium, the elastic collisions between the electrons and the heavy particles, and the heavy particle confine time have small kinetic energy transfers. The temperature of the heavy particles are in room temperature under low-pressure because of the low collision frequencies between the electrons and the heavy particles [Dinklage *et al.*, 2005].

Appendix B: Classical Coulomb Collision Theory

Expanding from section 2.5.2, Coulomb collision theory is based on Coulomb forces between two charged particles. The collision between the two particles in a plasma is an elastic process. The Coulomb force between the two test charges e_q and e_Q is [Howard, 2002]

$$F_{Coulomb} = \frac{e_q e_Q}{4\pi\epsilon_0 r^2}. \tag{B.1}$$

The Debye sheath in a plasma limits the Coulomb force such that an effective cross-section can be solved (see section 2.5.2 for details on cross-section). To produce an effective momentum transfer, which occurs when the angle between the colliding particles is 90^0, requires a densely populated plasma. Thus, calculating the mean free path would determine the statistical probability of collision occurring in the plasma. The mean free path, $\lambda_{mfp,s}$, is the reciprocal of the product of the cross-sectional area, $\sigma_{cross\ section-s_1,s_2}$, and the particle density, n_{s_1}, of the specific species [Howard, 2002],

$$\lambda_{mfp,s} = \frac{1}{n_{s_1}\sigma_{cross\ section-s_1,s_2}}. \tag{B.2}$$

where s_1 and s_2 are the species (electron, ion, or neutral) colliding with each other [Howard, 2002].

Thermal speed is the speed at which the individual species is moving in a plasma. The average thermal velocity of a species is [Howard, 2002]

$$c_s = \sqrt{\frac{8}{\pi}\frac{k_B T_s}{m_s}}. \tag{B.3}$$

The mass m_s is specific for each species; whether it is an electron, ion or neutral [Howard, 2002].

The plasma frequency is formed from microscopic deviations from the plasma in a quasi-neutral state. The plasma frequency is derived from the motion of the electron inside a static electric field. Thus, within the static and planar electric field, the plasma frequency for a species is [Howard, 2002]

$$\omega_{p,s} = \sqrt{\frac{ne^2}{\epsilon_0 m_s}}. \tag{B.4}$$

The Debye length, or characteristic length, is the distance over a plasma in which the electric field of the charge particle is shielded from the thermal motions of the other charged particles [Howard, 2002],

$$\lambda_D = \sqrt{\frac{\epsilon_0 k_B T_e}{n_e e^2}}. \tag{B.5}$$
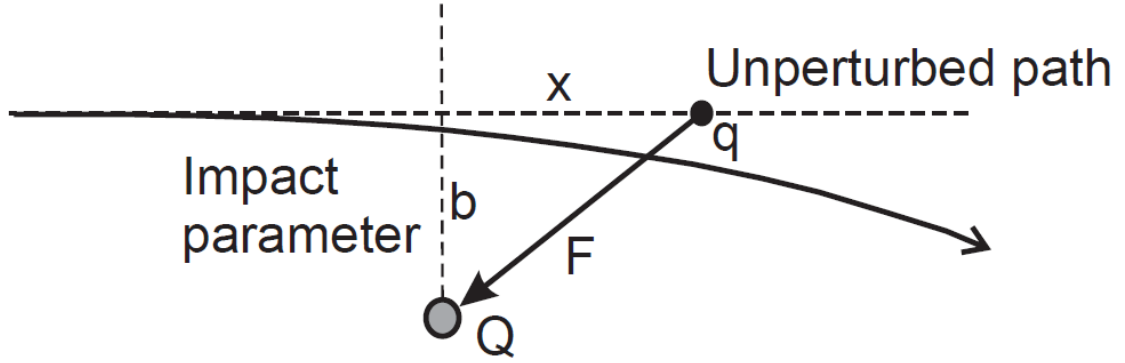
Figure B.1: Illustration of two charged particles and their paths [Figure 2.9 of Howard,

2002]

In a plasma, as two particles approach each other, their paths perturbed due to
their charge. The impact parameter, b, is the distance between the unpertured paths of the
two particles. The perpendicular force is proportional to the impact parameter [Howard,
2002],

$$F_\perp = \frac{e_q e_Q}{4\pi\epsilon_0 r^2}\frac{b}{r}. \tag{B.6a}$$

Note that r is the radius between the two particles, which can be represented as

$$\sqrt{x^2 + b^2}. \tag{B.6b}$$

Taking the impulse of the perpindular force over infinite distance and time yields,

$$\int_{-\infty}^{\infty} F_\perp dt = \frac{e_q e_Q b}{4\pi\epsilon_0} \int_{-\infty}^{\infty} \frac{dx}{(x^2 + b^2)^{\frac{3}{2}}}. \tag{B.7}$$

185

Integrating both yields the impulse,

$$\delta m v_\perp = \frac{e_q e_Q b}{4\pi\epsilon_0}\left(\left(\frac{x}{b^2\sqrt{x^2+b^2}}\right)\Bigg|_\infty - \left(\frac{x}{b^2\sqrt{x^2+b^2}}\right)\Bigg|_{-\infty}\right). \tag{B.7}$$

Taking the limit of x as it approaches to positive infinity yields the change in velocity of the particle as

$$\delta v_\perp = \frac{2e_q e_Q}{4\pi\epsilon_0 m v b}. \tag{B.8}$$

Taking the statistical average over the distribution of the collsions (see *Howard 2002* for more information), the closest approach for two test particles in a collision is

$$b_{min} = \frac{2e_q e_Q}{4\pi\epsilon_0 m v^2}. \tag{B.9}$$

The test particle does not experience an electric charge outside its Debye length. Therefore, b_{max} is λ_d. The plasma parameter, Λ, is the ratio of the Debye length and the minimum impact parameter [Howard, 2002],

$$\Lambda = 4\pi \cdot n_e \cdot \lambda_d^3 = \frac{\lambda_d}{b_{min}}. \tag{B.10}$$

The Coulomb parameter $\ln(\Lambda)$ is the logarithmic of the plasma parameter.

The collision frequency is described as [Howard, 2002]

$$f_{v,s_1 s_2} = n_{s_1} c_{s_1} \sigma_{cross\ section-s_1,s_2}. \tag{B.11}$$

The frequency depends on the number of particles in the first species, the thermal energy of the first species, and the cross-section area between the two species. The cross-sectional area, $\sigma_{cross\ section-s_1,s_2.}$, is the value at which scattering occurs between the two species. The effective cross section occurs when the two species collide at 90^0. The collision frequency of a typical electron-ion scattering is

$$f_{v,ei} = \frac{n_i Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_e^2 c_e^3}. \tag{B.12}$$

Since $f_{v,ei} = n_i \sigma_{cross\ section\text{-}ei} c_e$, then the cross section for electron-ion scattering is

$$\sigma_{cross\ section-ei} = \frac{Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_e^2 c_e^4} \tag{B.13}$$

The electron-ion collision frequency is the same as the electron-electron collision frequency, assuming that the plasma is quasi-neutral,

$$f_{v,ei} = \frac{n_e Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_e^2 c_e^3}. \tag{B.14}$$

For an ion-ion and ion-electron scattering, the collision frequency is linearly proportional to the electron-ion collision frequency [Howard, 2002]

$$f_{v,ii} = f_{v,ei} \sqrt{\frac{m_e}{m_i}} = \frac{n_i Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_i^{\frac{1}{2}} m_e^{\frac{3}{2}} c_e^3}, \tag{B.15a}$$

$$f_{v,ie} = f_{v,ei} \frac{m_e}{m_i} = \frac{n_i Z^2 e^4 \ln(\Lambda)}{2\pi \epsilon_0^2 m_i m_e c_e^3}. \tag{B.15b}$$

Under cold plasma, the gas is not fully ionized. In most cold plasma, only a fraction of the gas is ionized; the rest of the gas contains neutral atoms. The collision frequency between two neutrals in a homogeneous gas is [Howard, 2002]

$$f_{v,nn} = n_n c_n \cdot 2\pi r_n^2 = 2 n_n \pi r_n^2 \sqrt{\frac{8}{\pi} \frac{k_B T_n}{m_n}}. \qquad (\text{B.16})$$

Typically, the neutral-neutral collision frequency is very low due to the nature of its temperature. The neutral-ion collision frequency is similar to the neutral-neutral frequency since both species are nearly the same mass and temperature [Howard, 2002].

Electron-neutral collisions occur frequently in a partially ionized plasma. The collision frequency between the two is similar to the electron-ion collision frequency, without the chance of recombination or having any electric forces. Thus, this can be treated as the neutral-neutral collision of two different species,

$$f_{v,en} = n_n c_e \pi r_n^2 . \qquad (\text{B.17})$$

The radius of the electron is much smaller than the radius of the neutral; therefore, the radius is neglected in the equation. The cross section area can, πr_n^2, can be approximated to a degree of 10^{-19} m^2 [Howard, 2002].

Appendix C: Continuum Theory of Plasma

In this appendix we derive the continuum theory of plasmas.. However, in this appendix, the theory goes more into detail. This model considers the plasma as three interpenetrating fluids (electrons, ions, neutrals). Collision frequencies are introduced in the model to represent the collision effects in the plasma. The model is analogous to the Navier-Stokes equations in section 2.5.5.

The first equation in the continuum theory is mass continuity in the plasma,

$$\frac{\partial \rho_s}{\partial t} + \nabla \cdot (\rho_s \boldsymbol{v}_s) = 0. \tag{C.1}$$

This indicates that there is no change in the mass of the particle, or the number of total particles (electron, ion, neutral) in the fluid. The equation $\nabla(\rho_s \boldsymbol{v}_s)$ can be represented as,

$$\nabla \cdot (\rho_s \boldsymbol{v}_s) = \underbrace{\boldsymbol{v} \cdot \nabla \rho_s}_{convective\ derivative} + \underbrace{\rho_s (\nabla \cdot \boldsymbol{v})}_{rate\ of\ compression}. \tag{C.2}$$

The convective derivative represents the change of the density due to motion. The rate of compression represents the change of the volume due to the motion of the density [Laframboise, 1996; Howard, 2002; Hutchinson & Friedberg, 2003].

The second equation is the momentum continuity in the plasma,

$$\frac{\partial \rho_s \boldsymbol{v}_s}{\partial t} + \boldsymbol{v}_s \cdot \nabla \rho_s \boldsymbol{v}_s = -\nabla p + \mu \nabla^2 \boldsymbol{v}_s - \nabla \left(\frac{2}{3} \mu \nabla \cdot \boldsymbol{v}_s\right) + \rho_q (\mathbf{E} + \boldsymbol{v}_s \times \mathbf{B}) - \rho_s f_{v,s_1 s_2} (\boldsymbol{v}_{s_1} - \boldsymbol{v}_{s_2}). \tag{C.3}$$

The variables on the left hand side of the equation are the material derivative of the moving particle. The right hand side of the equations shows the pressure gradient of the given particle, the shear stress at subsonic speed, the shear stress at supersonic speed, the

Lorentz force, and the collisions between any of the two species. The collisions between the two particles are the frictional effects due to the collisions [Laframboise, 1996; Howard, 2002].

The final equation is the energy continuity,

$$\nabla p_s = v_{iso,s}^2 \nabla \rho_s, \tag{C.4}$$

where $v_{iso,s}$ is the isothermal sound speed. The energy continuity shows the gradient of the pressure is proportional to the gradient of the density of the fluid [Laframboise, 1996; Howard, 2002].

In a partially ionized plasma, the current density in a plasma is the difference in bulk velocities between the electrons and ions [Howard, 2002],

$$\mathbf{J} = -n_i e \mathbf{v}_i + n_e e \mathbf{v}_e = e(\Gamma_e - \Gamma_i), \tag{C.5}$$

where Γ_e and Γ_i are the electron and ion flux quantities, respectively. The overall particle flux describes the net velocity of particles in a species [Howard, 2002],

$$\Gamma_s = \mu_{m,s} n_s \mathbf{E} - D_s \nabla n_s. \tag{C.6}$$

The first quantity of the particle flux describes the mobility of the particles in an electric field. The mobility is the ratio of the specific charge to the collision frequency inside the plasma [Howard, 2002],

$$\mu_{m,s} = \frac{e}{m_s v_{s_1,s_2}}. \tag{C.7}$$

The second quantity in equation G.6 is Fick's Law of Diffusion. Fick's Law describes the flux of particles from a highly dense region to a lower one assuming there are either uncharged particles or no significant electric field present [Howard, 2002],

$$\Gamma_{Fick's,s} = -D_s \nabla n_s. \tag{C.8}$$

Note that D_s is the diffusion coefficient of the given species. The diffusion coefficient describes the step length of the particles in a medium at a given collision frequency [Howard, 2002],

$$D_s = \frac{k_B T_s}{m_s \nu_s} \sim \lambda_{mfp,s}^2 f_{\nu,s_1 s_2}. \tag{C.8}$$

Ambipolar diffusion, D_a, is the diffusion effect of both the electrons and ions within an electric field. The ambipolar diffusion contributes to the loss energy due to recombination [Howard, 2002],

$$D_a = \frac{\mu_{m,e} D_i - \mu_{m,i} D_e}{\mu_{m,i} + \mu_{m,e}}. \tag{C.9}$$

This can occur in a magnetic field, but the assumption in this manner is the magnetic field is negligible in the calculation. Qualitatively, the particle diffusion describes the divergence of Fick's first law of diffusion [Howard, 2002].

Appendix D: Constants for the Production Rate Quartic Equation

$$\dot{n}_e = n_n n_e \cdot (A T_e^4 + B T_e^3 - C T_e^2 + D T_e - E)$$

Gas Type	A	B	C	D	E
Helium[1]	$+2 \cdot 10^{-34}$	$-9 \cdot 10^{-30}$	$+2 \cdot 10^{-25}$	$-2 \cdot 10^{-21}$	$+5 \cdot 10^{-18}$
Neon[2]	$+3 \cdot 10^{-34}$	$-2 \cdot 10^{-29}$	$+4 \cdot 10^{-25}$	$-3 \cdot 10^{-21}$	$+1 \cdot 10^{-17}$
Nitrogen[3]	$+1 \cdot 10^{-33}$	$-6 \cdot 10^{-29}$	$+9 \cdot 10^{-25}$	$-6 \cdot 10^{-21}$	$+2 \cdot 10^{-17}$
Argon[4]	$+1 \cdot 10^{-33}$	$-6 \cdot 10^{-29}$	$+9 \cdot 10^{-25}$	$-7 \cdot 10^{-21}$	$+2 \cdot 10^{-17}$
Xenon[5]	$+1 \cdot 10^{-33}$	$+4 \cdot 10^{-30}$	$-1 \cdot 10^{-24}$	$+1 \cdot 10^{-20}$	$-5 \cdot 10^{-17}$

Table D.1: Constants for the Production Rate Quartic Equation. Note that these are all approximations within 15% of the values. (Cited in Section 2.5.5)

1. Kim *et al.*, 2004

2. Almeieda *et al.*, 1995

3. Itikawa, 2005

4. Phelps, 1990

5. Goebel & Katz, 2008

Appendix E: Sheath and Probes

This appendix was cited in section 2.8.1 of the dissertation. There is a region between the electrodes and the plasma when the gas undergoes a capacitive discharge. The sheath is a region where there is high amount of ions due to the electrons charging the walls of the electrodes. The region confines the escaping electrons electrostatically. Since the thermal velocity of the electrons is faster than the thermal velocity of ions, the electrons quickly stick to the walls, thus yielding a negative potential. Note that in the region, the quasi-neutrality assumption cannot hold [Howard, 2002; Goebel & Katz, 2008].

Considering a plasma which is in contact with this electrode wall, the maximum potential of the plasma shown in figure E.1 is at the centre; where the warm ions are. The cold ions, which are near the walls, have a lower potential.
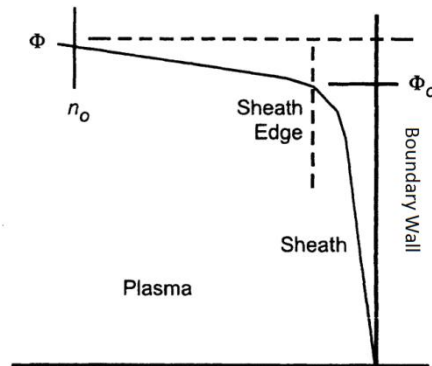
Figure E.1: Plasma in contact with a boundary wall [Figure 3.3 of Goebel & Katz, 2008]

The ionic current density entering the sheath edge can be found from the substance density at the sheath edge and its ion velocity. At the sheath edge, the plasma density decreases due to high amount of positive ions along the wall. This can be explained with the Boltzmann behaviour [Howard, 2002; Goebel & Katz, 2008]

$$n_e = n_0 \cdot e^{\left(\frac{e\phi}{k_B T_E}\right)}, \tag{E.1}$$

where n_0 is the particle density in the potential free region, and ϕ is the potential across

the plasma. At the sheath's edge, the ion's Bohm velocity is [Howard, 2002]

$$v_{B,i,0} = \sqrt{\frac{k_B T_e}{m_i}}. \tag{E.2}$$

Where m_i is the mass of the positive ion. Substituting ϕ as the electron temperature of the

plasma in equation E.1 and combining the results with E.2, the current density of the ions

entering the sheath is [Lieberman, 2003; Goebel & Katz, 2008]

$$J_{Sheath} = \frac{1}{2} n_e \sqrt{\frac{k_B T_e}{m_i}}. \tag{E.3}$$

The Child-Langmuir sheath is the simplest case of a sheath in a plasma. This case

occurs when the potential across the sheath is significantly large such that the electrons

repel from the sheath. Equation E.4 states that the current density which can pass through

a planar sheath is limited by the space-charge effects [Goebel & Katz, 2008]

$$J_{max} = \frac{4\epsilon_0}{9} \cdot \sqrt{\frac{2e}{m_i}} \cdot \frac{V^{\frac{3}{2}}}{l_e}, \tag{E.4}$$

where V is the potential difference between the electrodes. The corresponding potential

along the plasma wall is plasma potential along the wall is [Howard, 2002]

$$\phi_w = \phi_0 \ln\left(\frac{2\pi m_e}{m_i}\right), \tag{E.4}$$

where ϕ_0 is the potential along the wall of the electrode (see figure E.1) [Howard, 2002].

Studying the behaviour of the sheath within an electrode, there can be an analysis of the behaviour of the Langmuir probe. The initial assumption is the probe has a negative bias voltage such that all the ions are collected at the electrode and the electrons are repelled.
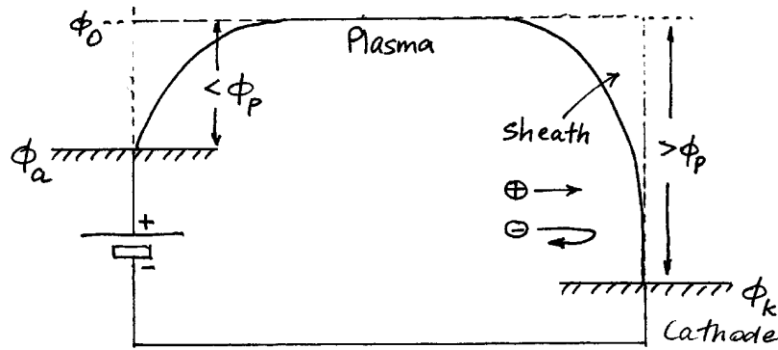
Figure E.2: Schematic Drawing Showing the Potential Drops Around the Plasma Circuit

[Figure 3.12 of Howard, 2002]

The ion current density which is collected is J_i and the ion current is

$$I = n_0 v_{B,i,0} eA, \tag{E.5}$$

where A is the area collected. There is a potential drop φ_{00} - φ_0 in the pre-sheath to accelerate the ion to Bohm speed

$$-e\phi_S = -e(\phi_0 - \phi_{00}) = \frac{1}{2} m_i v_{B,i,0}^2 = \frac{1}{2} k_B T_e \tag{E.6}$$

where ϕ_s is the sheath edge potential, and ϕ_{00} is the potential along the wall. The electron density at the edge is described as the Boltzmann behaviour in equation E.1. The ion saturation current is

$$I_{si} = n_0 v_{B,i,0} eA = \frac{1}{2} n_{00} eA \left(\frac{k_B T_e}{m_i}\right)^{\frac{1}{2}} \tag{E.7}$$

195

Thus, the probe measures the particle density of the plasma given the electron temperature [Howard, 2002].

Consider a probe potential $V_{pp} = V - V_p$ where V_p is the plasma potential $V_p = \varphi_w$. The current of the electrons at the probe is

$$I = I_{se}e^{\frac{eV_{pp}}{k_B T_e}}, \tag{E.8a}$$

$$I_{se} = \frac{Aen_0\langle v \rangle}{4}. \tag{E.8b}$$

The value I_{se} is the electron saturation current. For a positive probe bias, all the electrons are collected and $I = I_{se}$. Note that

$$\frac{I_{se}}{I_{si}} \approx \sqrt{\frac{m_i}{m_e}} \gg 1 \tag{E.7}$$

The current − voltage characteristic in a Langmuir probe is shown in figure E.3 [Howard, 2002].
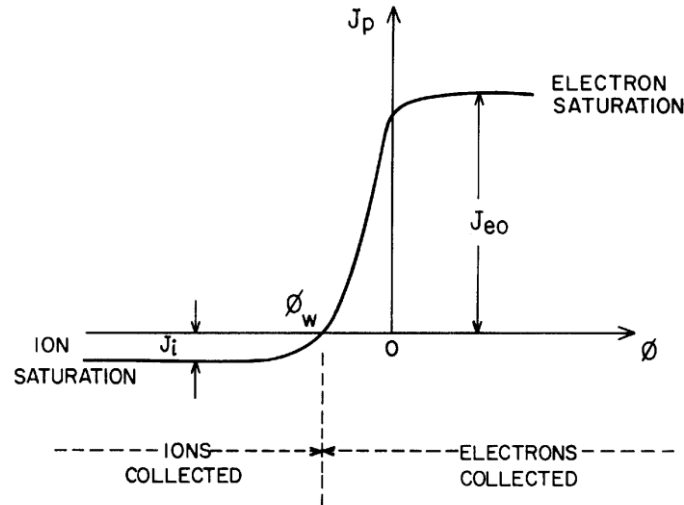
Figure E.3: The Langmuir probe I − V characteristic showing the electron and ion contributions and the plasma.[Figure 3.13 of Howard, 2002]

Appendix F: Mechanical Drawings

The following are the mechanical drawings of the electrothermal thruster and the hybrid electric thruster (cited in Section 3.2.2). Note that the drawings are not to scale because they must fit within the given margins. Section F.1 contains the drawings of the electrothermal thruster. Section F.2 contains the drawings of the hybrid electric thruster. The properties of the materials used in the electrothermal thruster are described in table F.1, and the hybrid electric thruster are described in table F.2.

Component	Material	Young's Modulus (GPa)	Ultimate Strength (MPa)	Poisson's Ratio
Support	Acrylic	3.3	40	0.37
Nozzle	Aluminum 6061	68.9	117	0.33
Tip	Aluminum 6061	68.9	117	0.33

Table F.1: Material Properties for the Electrothermal Thruster

Component	Material	Young's Modulus (GPa)	Ultimate Strength (MPa)	Poisson's Ratio
Support	Acrylic	3.3	40	0.37
Nozzle	Silica Glass	68.0	33	0.19
Anode	Annealed Copper	110.0	210	0.34
Tip	Aluminum 6061	68.9	117	0.33
Ring	Acrylic	3.3	40	0.37
Electrode	Annealed Copper	110.0	210	0.34
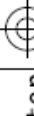
Table F.2: Material Properties for the Hybrid Electric Thruster

F.1 Electrothermal Thruster Mechanical Drawings

Isometric view
Scale: 1:1

Exploded View

York University

Thruster

DESIGNED BY:
Barry Stoute
DATE: Feb 18 2011
CHECKED BY:
Brendan Quine
DATE: XXX
SIZE Letter
SCALE 1:1
WEIGHT (kg) <1
DRAWING NUMBER
SHEET 1/1

This drawing is our property; it can't be reproduced or communicated without our written agreement.

Ø 22

Ø 10

Right view
Scale: 2:1

4

3.5

Ø 17

R 5

Front view
Scale: 2:1

12 ±1

Ø 12   R 7.5

46 ±1

1

12° +1° 0°

Aluminum Nozzle

Note: 1mm Thicness throughout the Nozzle. Indicated on the Working Sheet. All diameters indicated are the outer diameters.

York University

DESIGNED BY: Barry Stoute			I	–
DATE: Feb 18 2011			H	–
CHECKED BY: Brendan Quine			G	–
DATE: TBD			F	–
			E	–
SIZE Letter		DRAWING NUMBER	D	–
		Nozzle	C	–
SCALE 2:1	WEIGHT (kg) 0.01	SHEET 1/1	B	–
			A	–

Front view
Scale: 4:1

15

4

45°

$\varnothing 8 \left(\begin{array}{c} +0.024 \\ +0.015 \end{array} \right)$

Aluminum Tip

York University

DESIGNED BY:
Barry Stoute
DATE:
Feb 18 2011
CHECKED BY:
Brendan Quine
DATE:
XXX

SIZE
Letter

WEIGHT (kg)
0.00

DRAWING NUMBER

Tip

SCALE
4:1

SHEET
1/1

I | —
H | —
G | —
F | —
E | —
D | —
C | —
B | —
A | —

200

Front view
Scale: 2:1

Right view
Scale: 2:1

40

40

Ø 15 Ø 17

Ø 1

$\varnothing 8 \left(\begin{array}{c} +0.015 \\ 0 \end{array} \right)$

10

5

I	—	
H	—	
G	—	
F	—	
E	—	
D	—	
C	—	
B	—	
A	—	

201

F.2 Hybrid Electric Thruster Mechanical Drawings

Front view
Scale: 2:1

Left view
Scale: 2:1

17

7

1

10

1

0.5

R 0.5 +0.1 / 0

R 0.5 +0.2 / +0.1

18° (+2° / -2°)

Ø 1

Ø 1

Ø 14 Ø 13

Ø 15

Ø 17

DESIGNED BY:
Barry Stoute
DATE:
March 30 2012
CHECKED BY:
TBD
DATE:
TBD

Copper - Anode

All dimensions are in mm. The most important dimension is the 17mm diameter. This has to be a fit for the acrylic block.
The material can be copper or any solderable material.

York University

DRAWING NUMBER

Anode

SIZE
Letter

SCALE
3:2

WEIGHT (kg)
0.01

SHEET
1/1

I —
H —
G —
F —
E —
D —
C —
B —
A —

203

See Note Below
On Thickness

Front view
Scale: 3:1

R 7

R 6.5

Left view
Scale: 3:1

Acrylic Ring

4 Rings:
0.5mm Thick
1.0mm Thick
2.0mm Thick
5.0mm Thick

DESIGNED BY:
Barry Stoute
DATE:
March 29 2012
CHECKED BY:
TBD
DATE:
TBD

York University

DRAWING NUMBER

SIZE
Letter

WEIGHT (kg)
0.01

Ring

SCALE
3:1

SHEET
1/1

This drawing is our property; it can't be reproduced or communicated without our written agreement.

	I	–
	H	–
	G	–
	F	–
	E	–
	D	–
	C	–
	B	–
	A	–

See Notes Below
On Thickness

Front view
Scale: 4:1

$\varnothing\ 14\ ^{+0.02}_{\ 0}$

Left view
Scale: 4:1

DESIGNED BY:			Copper Electrode	I	—
Barry Stoute				H	—
DATE:				G	—
March 29 2012		4 Thicknesses: 0.5mm, 1mm, 1.5mm, 2mm.	F	—
CHECKED BY:		The material can be copper or any solderable		
TBD		material.	E	—
DATE:				
TBD		York University	D	—
SIZE	DRAWING NUMBER			C	—
Letter	Electrode			B	—
SCALE	WEIGHT (kg)	SHEET			
4:1	0.00	1/1		A	—

York University

Electrode

SHEET 1/1

Screen Electrode

60 Degree Apart from each other

⌀ 3

4

3.46

3.46

4

2

2
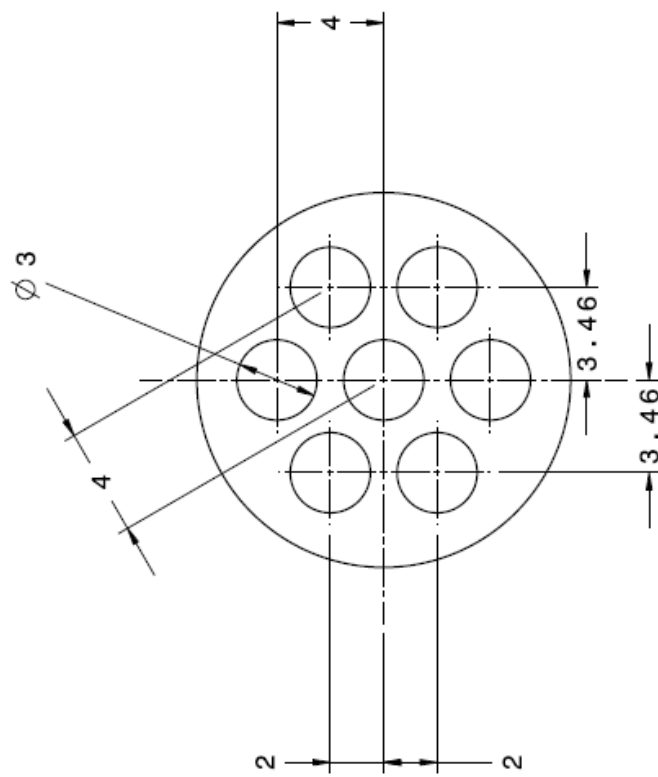
Left view
Scale: 5:1

Acceleration Electrode

60 Degrees Apart from each other

USE THIS DRAWING FIRST!

⌀ 2.5

4

4

3.46

3.46

2

2

Left view
Scale: 5:1
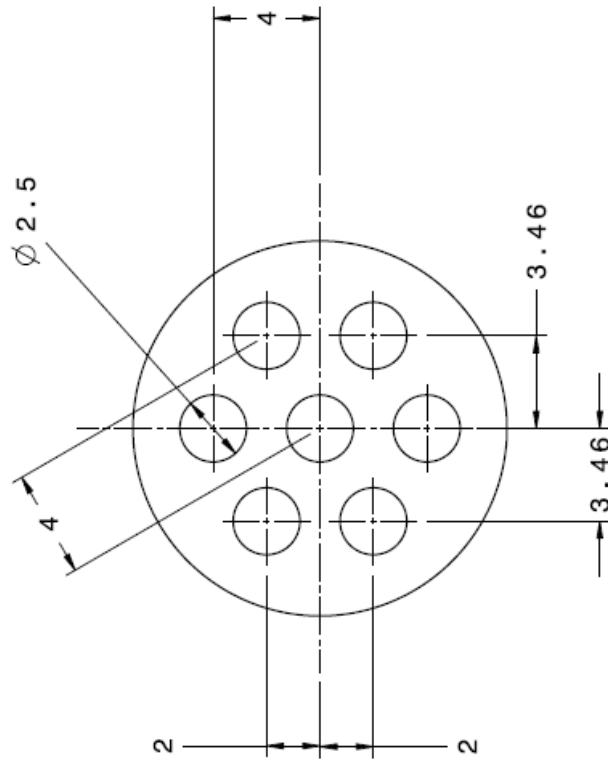
Appendix G: Glow Discharge Graphs
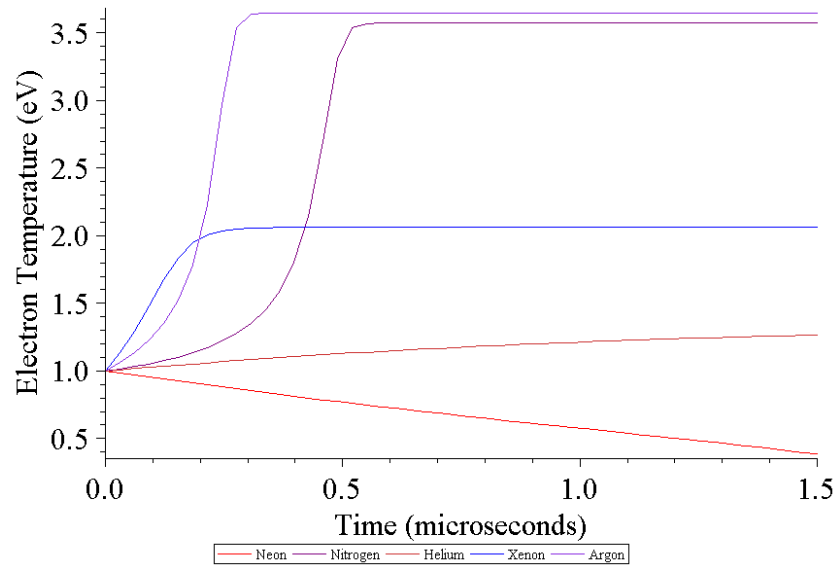
Figure G.1: Electron Temperature vs. Time (Cited in Section 3.7)
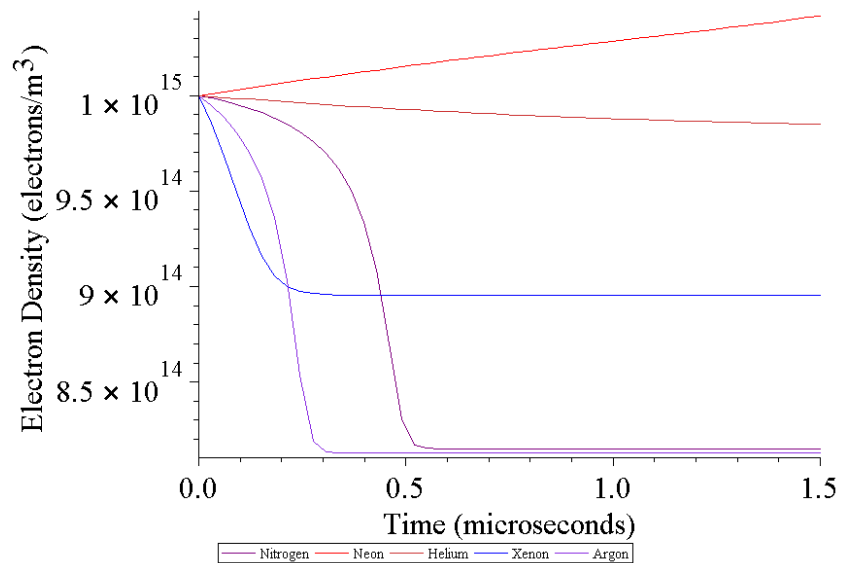
Figure G.2: Electron Density vs. Time (Cited in Section 3.7)

Appendix H: Programming Code Used in Simulations

This section of the dissertation has been cited in section 3.7.

H.1 Glow Discharge

This program is written completely by the author of this dissertation, C. A. B. Stoute, using Maple software. The calculated results from the program have been applied to the computational fluid dynamic program.

> $restart; with(plots) : with(linalg) : with(ArrayTools) : Setup(mathematicalnotation = true) :$
> $with(VectorCalculus) : with(DETools) : with(PDETools) :$

<u>Setting the Variables</u>

> $NE := ne0 \cdot ne(t); NN := nn0 \cdot nn(t); TE := Te(t) \cdot Te0; TH := Th(t) \cdot Th0; U := u(t) \cdot u0; E0$
> $:= -\dfrac{\Phi}{radius}; EF := E0 \cdot \cos(\omega \cdot t)$

$$NE := ne0\, ne(t)$$

$$NN := nn0\, nn(t)$$

$$TE := Te(t)\, Te0$$

$$TH := Th(t)\, Th0$$

$$U := u(t)\, u0$$

$$E0 := -\frac{\Phi}{radius}$$

$$EF := -\frac{\Phi \cos(\omega\, t)}{radius}$$

209

Plasma Generation Equations

> $kiz := \left(S1 \cdot \text{sqrt}\left(\dfrac{32}{\pi} \cdot \left(\dfrac{me+mn}{me \cdot mn} \right) \right) \cdot (kB \cdot TE)^{\frac{3}{2}} \cdot \left(\dfrac{Ui}{2 \cdot kB \cdot \text{TE}} + 1 \right) \cdot \exp\left(-\dfrac{Ui}{kB \cdot TE} \right) \right);$
>
> $S1 := 4\text{E}-3;$
> $g0 := 1;$
> $g1 := 0.5;$
>
> $kre := \left(\left(\dfrac{1}{kiz} \cdot \dfrac{2 \cdot g1}{g0} \cdot \left(\dfrac{2 \cdot \pi \cdot me \cdot kB \cdot TE}{h^2} \right)^{\frac{3}{2}} \cdot \exp\left(-\dfrac{Ui}{kB \cdot TE} \right) \right)^{-1} \right);$
>
> \#nedot:= kiz $\cdot$ NN $\cdot$ NE $- kre \cdot (NE)^3;$

$$kiz := 4\,S1\,\sqrt{2}\ \sqrt{\dfrac{me+mn}{\pi\,me\,mn}}\ (kB\,Te(t)\,Te0)^{3/2} \left(\dfrac{1}{2}\ \dfrac{Ui}{kB\,Te(t)\,Te0} + 1 \right) e^{-\frac{Ui}{kB\,Te(t)\,Te0}}$$

$$S1 := 0.004$$

$$g0 := 1$$

$$g1 := 0.5$$

$$kre := \dfrac{0.008000000000\,\sqrt{\dfrac{me+mn}{\pi\,me\,mn}}\ (kB\,Te(t)\,Te0)^{3/2} \left(\dfrac{1}{2}\ \dfrac{Ui}{kB\,Te(t)\,Te0} + 1 \right)}{\left(\dfrac{\pi\,me\,kB\,Te(t)\,Te0}{h^2} \right)^{3/2}}$$

> $vei := \dfrac{1}{128} \cdot \dfrac{\text{NE} \cdot e^4 \cdot \ln\left(4 \cdot \pi \cdot NE \left(\dfrac{\varepsilon0 \cdot kB \cdot TE}{NE \cdot e^2} \right)^{3/2} \right) \cdot \pi}{\varepsilon0^2 \cdot kB^2 \cdot TE^2};$
>
> $ven := 6.\,10^{-20} \cdot NE \cdot \sqrt{2} \cdot \sqrt{\dfrac{kB \cdot TE}{\pi \cdot me}}\ ;$
>
> $ve := vei + ven;\quad U\_i := E\_i \cdot e;\quad Ui := U\_i;$

$$vei := \dfrac{1}{128}\ \dfrac{ne0\,ne(t)\,e^4 \ln\left(4\,\pi\,ne0\,ne(t) \left(\dfrac{\varepsilon0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2} \right)^{3/2} \right) \pi}{\varepsilon0^2\,kB^2\,Te(t)^2\,Te0^2}$$

$$ven := 6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\ \sqrt{\dfrac{kB\,Te(t)\,Te0}{\pi\,me}}$$

210

$$ve := \frac{1}{128} \frac{ne0\,ne(t)\,e^4 \ln\left(4\,\pi\,ne0\,ne(t)\left(\frac{\varepsilon0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2}\right)^{3/2}\right)\pi}{\varepsilon0^2\,kB^2\,Te(t)^2\,Te0^2}$$

$$+ 6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\,\sqrt{\frac{kB\,Te(t)\,Te0}{\pi\,me}}$$

$$U\_i := E\_i\,e$$

$$Ui := E\_i\,e$$

> $$\kappa e := \left(\frac{3.2\cdot e\cdot kB\cdot TE\cdot(NE+NN)}{ve\cdot me}\right);\ \kappa h := \left(\frac{3.2\cdot e\cdot kB\cdot TH\cdot(NE+NN)}{vi\cdot mi}\right);$$

$$\kappa e := (3.2\,e\,kB\,Te(t)\,Te0\,(ne0\,ne(t)+nn0\,nn(t)))\Big/$$

$$\left(\left(\frac{1}{128}\frac{ne0\,ne(t)\,e^4\ln\left(4\,\pi\,ne0\,ne(t)\left(\frac{\varepsilon0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2}\right)^{3/2}\right)\pi}{\varepsilon0^2\,kB^2\,Te(t)^2\,Te0^2}\right.\right.$$

$$\left.\left.+ 6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\,\sqrt{\frac{kB\,Te(t)\,Te0}{\pi\,me}}\right)me\right)$$

$$\kappa h := \frac{3.2\,e\,kB\,Th(t)\,Th0\,(ne0\,ne(t)+nn0\,nn(t))}{vi\,mi}$$

> $$Qabs := \frac{NE\cdot e^2}{me\cdot ve}\cdot\frac{(ve)^2}{\omega^2+(ve)^2}\cdot\frac{E0^2}{2};$$

$$Qabs := \frac{1}{2}\left(ne0\,ne(t)\,e^2\left(\frac{1}{128}\frac{ne0\,ne(t)\,e^4\ln\left(4\,\pi\,ne0\,ne(t)\left(\dfrac{\varepsilon 0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2}\right)^{3/2}\right)\pi}{\varepsilon 0^2\,kB^2\,Te(t)^2\,Te0^2}\right.\right.$$

$$\left.\left. +\,6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\,\sqrt{\frac{kB\,Te(t)\,Te0}{\pi\,me}}\right)\Phi^2\right)\left/\left(me\left(\omega^2\right.\right.\right.$$

$$\left.+\left(\frac{1}{128}\frac{ne0\,ne(t)\,e^4\ln\left(4\,\pi\,ne0\,ne(t)\left(\dfrac{\varepsilon 0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2}\right)^{3/2}\right)\pi}{\varepsilon 0^2\,kB^2\,Te(t)^2\,Te0^2}\right.\right.$$

$$\left.\left.\left.\left. +\,6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\,\sqrt{\frac{kB\,Te(t)\,Te0}{\pi\,me}}\right)^2\right)radius^2\right)\right.$$

> $Qelas := 3\cdot\dfrac{me}{mn}\cdot NE\cdot ve\cdot kB\cdot(TE - TH);$

$$Qelas := \frac{1}{mn}\left(3\,me\,ne0\,ne(t)\left(\frac{1}{128}\frac{ne0\,ne(t)\,e^4\ln\left(4\,\pi\,ne0\,ne(t)\left(\dfrac{\varepsilon 0\,kB\,Te(t)\,Te0}{ne0\,ne(t)\,e^2}\right)^{3/2}\right)\pi}{\varepsilon 0^2\,kB^2\,Te(t)^2\,Te0^2}\right.\right.$$

$$\left.\left. +\,6.000000000\,10^{-20}\,ne0\,ne(t)\,\sqrt{2}\,\sqrt{\frac{kB\,Te(t)\,Te0}{\pi\,me}}\right)kB\,(Te(t)\,Te0 - Th(t)\,Th0)\right)$$

> $UN := 0\cdot10^3 : UI := UN : Ren := -NE\cdot me\cdot ven\cdot(U - UN) : Rei := -NE\cdot me\cdot vei\cdot(U - UI) :$
$Rin := -NE\cdot mn\cdot vin\cdot(U - UN) : Rne := -Ren : Rie := -Rei : Rni := -Rin : Qfrici := -Rin$
$\cdot UI - Rie\cdot UI - Rni\cdot UN - Rne\cdot UN : Qfrici\_on := 1 :$

> $H1 := \text{diff}(NE, t)\cdot f0 = nedot;$

$$H1 := ne0\left(\frac{d}{dt}\,ne(t)\right)f0 = nedot$$

> $H2 := diff\left(\dfrac{3}{2}\cdot NE\cdot kB\cdot TE, t\right)\cdot f0 = Qabs - e\cdot nedot\cdot E\_i$

$$H2 := \left(\frac{3}{2}\, ne0 \left(\frac{d}{dt}\, ne(t) \right) kB\, Te(t)\, Te0 + \frac{3}{2}\, ne0\, ne(t)\, kB \left(\frac{d}{dt}\, Te(t) \right) Te0 \right) f0$$

$$= \frac{1}{2} \left(ne0\, ne(t)\, e^2 \left(\frac{1}{128}\, \frac{ne0\, ne(t)\, e^4 \ln\!\left(4\,\pi\, ne0\, ne(t) \left(\frac{\varepsilon0\, kB\, Te(t)\, Te0}{ne0\, ne(t)\, e^2} \right)^{3/2} \right) \pi}{\varepsilon0^2\, kB^2\, Te(t)^2\, Te0^2} \right. \right.$$

$$\left. + 6.000000000\,10^{-20}\, ne0\, ne(t)\, \sqrt{2}\, \sqrt{\frac{kB\, Te(t)\, Te0}{\pi\, me}} \right) \Phi^2 \Bigg) \Bigg/ \left(me \left(\omega^2 \right. \right.$$

$$+ \left(\frac{1}{128}\, \frac{ne0\, ne(t)\, e^4 \ln\!\left(4\,\pi\, ne0\, ne(t) \left(\frac{\varepsilon0\, kB\, Te(t)\, Te0}{ne0\, ne(t)\, e^2} \right)^{3/2} \right) \pi}{\varepsilon0^2\, kB^2\, Te(t)^2\, Te0^2} \right.$$

$$\left. \left. \left. + 6.000000000\,10^{-20}\, ne0\, ne(t)\, \sqrt{2}\, \sqrt{\frac{kB\, Te(t)\, Te0}{\pi\, me}} \right)^2 \right) radius^2 \right) - e\, nedot\, E\_i$$

> $H3 := diff(NN, t) \cdot f0 = -nedot;$

$$H3 := nn0 \left(\frac{d}{dt}\, nn(t) \right) f0 = -nedot$$

> $H4 := diff\left(\frac{3}{2} \cdot (NE + NN) \cdot kB \cdot TH,\, t \right) \cdot f0 = 0 \cdot Qelas$

$$H4 := \left(\left(\frac{3}{2}\, ne0 \left(\frac{d}{dt}\, ne(t) \right) + \frac{3}{2}\, nn0 \left(\frac{d}{dt}\, nn(t) \right) \right) kB\, Th(t)\, Th0 + \left(\frac{3}{2}\, ne0\, ne(t) \right. \right.$$

$$\left. + \frac{3}{2}\, nn0\, nn(t) \right) kB \left(\frac{d}{dt}\, Th(t) \right) Th0 \Bigg) f0 = 0$$

Initial Values

> $ne0 := 10^{15};\, nn0 := 10^{22};\, Te0 := 11620;\, Th0 := 293;\, f0 := 100 \cdot 10^6;\, u0 := 10^3;$
>   *# Parameters which works*

$$ne0 := 1000000000000000$$

213

$$nn0 := 10000000000000000000000$$

$$Te0 := 11620$$

$$Th0 := 293$$

$$f0 := 100000000$$

$$u0 := 1000$$

> $\omega := f0 \cdot 2 \cdot \pi;\ \Phi := -15;\ radius := \dfrac{3.5}{1000}$

$$\omega := 200000000\,\pi$$

$$\Phi := -15$$

$$radius := 0.003500000000$$

> $mn := m \cdot mp + m \cdot me;\ \#\ mass\ of\ the\ atom$
$mi := m \cdot mp + m \cdot me - me;\ \#\ mass\ of\ an\ ion$
$q := -e\ \#\ Charge\ of\ an\ electron$

$$mn := m\,mp + m\,me$$

$$mi := m\,mp + m\,me - me$$

$$q := -e$$

Standard Constants Used

> $\varepsilon 0 := 8.85\mathrm{E}{-}12 : \mu 0 := 4 \cdot \pi \cdot 10^{-7} : e := 1.602\mathrm{E}{-}19 : me := 9.109\mathrm{E}{-}31 : mp := 1.6726$
$\cdot 10^{-27} : mp\_me := \dfrac{mp}{me} : h := 6.6262 \cdot 10^{-34} : h\_bar := evalf\left(\dfrac{h}{2 \cdot \mathrm{pi}}\right) : c0 := 2.9979$
$\cdot 10^{8} : a0 := 5.2918 \cdot 10^{-11} : A\_atomic := evalf\left(\pi \cdot a0^{2}\right) : p0 := 101.33\mathrm{E}3 : kB$
$:= 1.38\mathrm{E}{-}23 : Torr := \dfrac{1}{0.0075} : atm := 101.34\mathrm{E}3 : SB\sigma := 5.67\mathrm{E}{-}8 : b$
$:= 2.897\mathrm{E}{-}3 : g0 := 9.8066 : R := 8.314 : R1000 := R \cdot 1000 :$

Solving for the ODEs

> $IC := ne(0) = 1, Te(0) = 1, nn(0) = 1, Th(0) = 1$

$$IC := ne(0) = 1,\ Te(0) = 1,\ nn(0) = 1,\ Th(0) = 1$$

> $timer := 1.5 \cdot 10^{-6}$

> $timesteps := f0 \cdot timer$

Nedot is a Quadric Function

> $HHe := subs\Big(\Big[CpMol = 20.786, \rho = 0.1782, m = 4, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{2372.3}{96.485}, nedot$
> $= NE \cdot NN \cdot \big((\text{2E-34} \cdot TE^4 - \text{9E-30} \cdot TE^3 + \text{2E}{-}25 \cdot TE^2 - \text{2E}{-}21 \cdot TE + \text{5E}{-}18)\big)\Big], [H1,$
> $H2, H3, H4]\Big)[\,] :$

> $HNe := subs\Big(\Big[CpMol = 20.786, \rho = 0.9002, m = 20.179, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{2080.7}{96.485},$
> $nedot = NE \cdot NN \cdot \big((\text{3E}{-}34 \cdot TE^4 - \text{2E}{-}29 \cdot TE^3 + \text{4E}{-}25 \cdot TE^2 - \text{3E}{-}21 \cdot TE$
> $+ \text{1E}{-}17)\big)\Big], [H1, H2, H3, H4])[\,] :$

> $HN2 := subs\Big(\Big[CpMol = 29.12, \rho = 1.251, m = 28, kk = \dfrac{CpMol}{CpMol - R}, E\_i = 15.52, nedot = NE$
> $\cdot NN \cdot \big((\text{1E}{-}33 \cdot TE^4 - \text{6E}{-}29 \cdot TE^3 + \text{9E}{-}25 \cdot TE^2 - \text{6E}{-}21 \cdot TE + \text{2E}{-}17)\big)\Big], [H1, H2,$
> $H3, H4]\Big)[\,] :$

> $HAr := subs\Big(\Big[CpMol = 20.786, \rho = 1.784, m = 39.986, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{1520.6}{96.485},$
> $nedot = NE \cdot NN \cdot \big((\text{1E}{-}33 \cdot TE^4 - \text{6E}{-}29 \cdot TE^3 + \text{9E}{-}25 \cdot TE^2 - \text{7E}{-}21 \cdot TE$
> $+ \text{2E}{-}17)\big)\Big], [H1, H2, H3, H4])[\,] :$

> $HXe := subs\Big(\Big[CpMol = 20.786, \rho = 5.894, m = 131.293, kk = \dfrac{CpMol}{CpMol - R}, E\_i = 12.127,$
> $nedot = NE \cdot NN \cdot \big((\text{1E}{-}33 \cdot TE^4 + \text{4E}{-}30 \cdot TE^3 - \text{1E}{-}24 \cdot TE^2 + \text{1E}{-}20 \cdot TE$
> $- \text{5E}{-}17)\big)\Big], [H1, H2, H3, H4])[\,] :$

Nedot is independent and Motion of Electron

> $\#HHe := subs\left(\left[CpMol = 20.786, \rho = 0.1782, m = 4, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{2372.3}{96.485}\right], [H1, H2, H3, H4]\right)[\ ]:$

> $\#HNe := subs\left(\left[CpMol = 20.786, \rho = 0.9002, m = 20.179, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{2080.7}{96.485}\right], [H1, H2, H3, H4]\right)[\ ]:$

> $\#HN2 := subs\left(\left[CpMol = 29.12, \rho = 1.251, m = 28, kk = \dfrac{CpMol}{CpMol - R}, E\_i = 15.52\right], [H1, H2, H3, H4]\right)[\ ]:$

> $\#HAr := subs\left(\left[CpMol = 20.786, \rho = 1.784, m = 39.986, kk = \dfrac{CpMol}{CpMol - R}, E\_i = \dfrac{1520.6}{96.485}\right], [H1, H2, H3, H4]\right)[\ ]:$

> $\#HXe := subs\left(\left[CpMol = 20.786, \rho = 5.894, m = 131.293, kk = \dfrac{CpMol}{CpMol - R}, E\_i = 12.127\right], [H1, H2, H3, H4]\right)[\ ]:$

Solving

>
$HEHe := dsolve([HHe, IC], type = numeric);$
$HENe := dsolve([HNe, IC], type = numeric);$
$HEN2 := dsolve([HN2, IC], type = numeric);$
$HEAr := dsolve([HAr, IC], type = numeric);$
$HEXe := dsolve([HXe, IC], type = numeric);$

$$HEHe := \mathbf{proc}(x\_rkf45)\ ...\ \mathbf{end\ proc}$$

$$HENe := \mathbf{proc}(x\_rkf45)\ ...\ \mathbf{end\ proc}$$

216

$$HEN2 := \textbf{proc}(x\_rkf45) \ ... \ \textbf{end proc}$$

$$HEAr := \textbf{proc}(x\_rkf45) \ ... \ \textbf{end proc}$$

$$HEXe := \textbf{proc}(x\_rkf45) \ ... \ \textbf{end proc}$$

> $t\_ := evalf\left(\dfrac{rhs(HENe(timesteps)[1])}{f0} \right)$

$$t\_ := 0.000001500000000$$

Electron Temperature Results

> $p1 := odeplot\big(HEHe, \big[10^6 \cdot f0^{-1} \cdot t, Te(t), color = orange\big], t = 0 \,..\, timesteps, legend$
$\qquad = \text{"Helium"}\big) :$
$p2 := odeplot\big(HENe, \big[10^6 \cdot f0^{-1} \cdot t, Te(t), color = red\big], t = 0 \,..\, timesteps, legend = \text{"Neon"}\big) :$
$p3 := odeplot\big(HEN2, \big[10^6 \cdot f0^{-1} \cdot t, Te(t), color = \text{"Purple"}\big], t = 0 \,..\, timesteps, legend$
$\qquad = \text{"Nitrogen"}\big) :$
$p4 := odeplot\big(HEAr, \big[10^6 \cdot f0^{-1} \cdot t, Te(t), color = \text{"BlueViolet"}\big], t = 0 \,..\, timesteps, legend$
$\qquad = \text{"Argon"}\big) :$
$p5 := odeplot\big(HEXe, \big[10^6 \cdot f0^{-1} \cdot t, Te(t), color = \text{"Blue"}\big], t = 0 \,..\, timesteps, legend = \text{"Xenon"}\big) :$

> $plots[display](\{p1, p2, p3, p4, p5\}, labels = [\text{"Time (microseconds)"},$
$\qquad \text{"Electron Temperature (eV)"}], labeldirections = [horizontal, vertical], titlefont = [TIMES,$
$\qquad ROMAN, 24], font = [TIMES, ROMAN, 24]);$
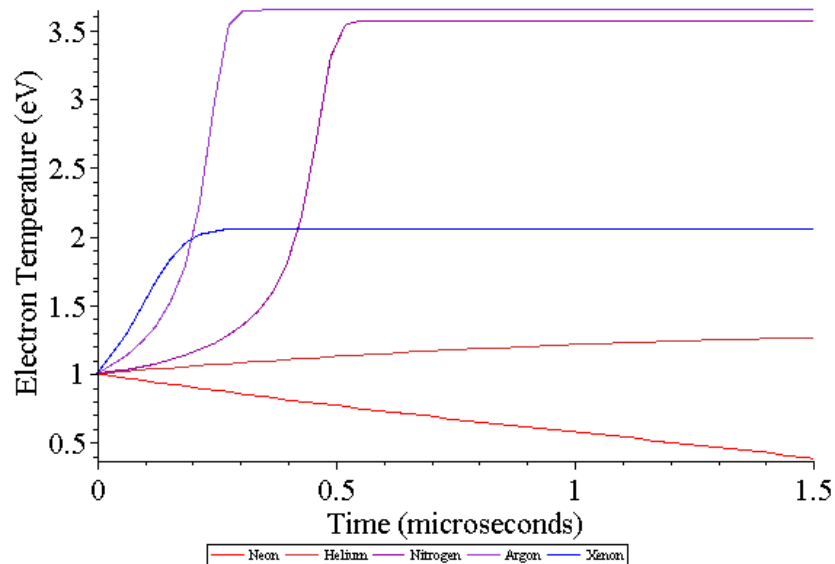
Electron Density Results

>

$p1 := odeplot\big(HEHe, \big[10^6 \cdot f0^{-1} \cdot t, nn0 \cdot nn(t), color = orange\big], t = 0 ..timesteps, legend$
$\quad = "Helium"\big) :$

$p2 := odeplot\big(HENe, \big[10^6 \cdot f0^{-1} \cdot t, ne0 \cdot ne(t), color = red\big], t = 0 ..timesteps, legend$
$\quad = "Neon"\big) :$

$p3 := odeplot\big(HEN2, \big[10^6 \cdot f0^{-1} \cdot t, nn0 \cdot nn(t), color = "Purple"\big], t = 0 ..timesteps, legend$
$\quad = "Nitrogen"\big) :$

$p4 := odeplot\big(HEAr, \big[10^6 \cdot f0^{-1} \cdot t, ne0 \cdot ne(t), color = "BlueViolet"\big], t = 0 ..timesteps, legend$
$\quad = "Argon"\big) :$

$p5 := odeplot\big(HEXe, \big[10^6 \cdot f0^{-1} \cdot t, ne0 \cdot ne(t), color = "Blue"\big], t = 0 ..timesteps, legend$
$\quad = "Xenon"\big) :$

> $plots[display](\{p1, p2, p3, p4, p5\}, labels = ["Time (microseconds)",$
$\quad "Electron Density (electrons/m^3)"], labeldirections = [horizontal, vertical], titlefont$
$\quad = [TIMES, ROMAN, 24], font = [TIMES, ROMAN, 24]);$
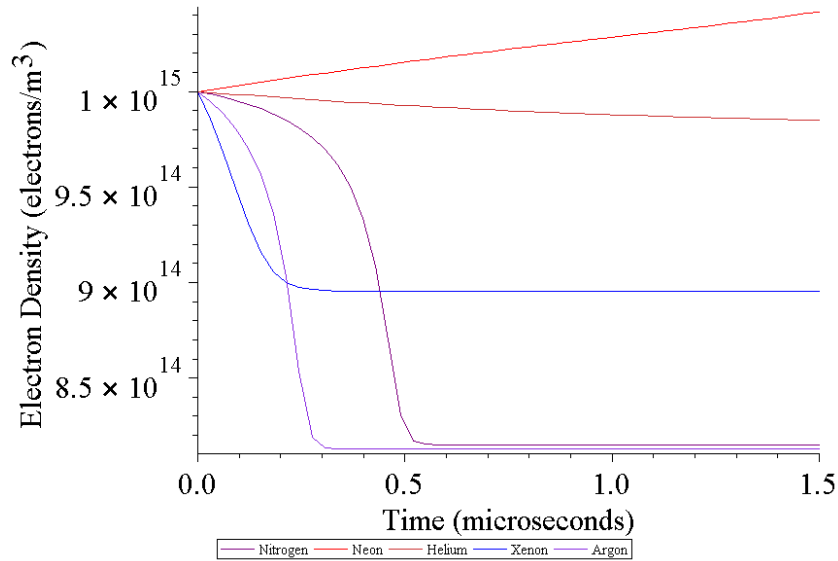
> $IpHe = 1.05 \cdot 10^{15} \cdot 10^{22} \cdot \big(2\text{E-}34 \cdot 0.33^4 - 9\text{E-}30 \cdot 0.33^3 + 2\text{E-}25 \cdot 0.33^2 - 2\text{E-}21 \cdot 0.33$
$\quad + 5\text{E-}18\big) \cdot e \cdot 1.864 \cdot 10^{-7} \cdot \pi$

$nidotHe = 0.000001567510268\,\pi$

218

> $IpN2 = 8.1 \cdot 10^{14} \cdot 10^{22} \cdot \left(1\text{E}-33 \cdot 3.55^4 - 6\text{E}-29 \cdot 3.55^3 + 9\text{E}-25 \cdot 3.55^2 - 7\text{E}-21 \cdot 3.55 + 2\text{E}-17 \right) \cdot e \cdot 1.864 \cdot 10^{-7} \cdot \pi$

$$nidotHe = 0.000004831519477\,\pi$$

>

H.2 MATLAB Code for Simulations

The MATLAB code for the computational fluid dynamic model of the thruster have been modified from the original author Britton Jeffery Olson. Olson have developed three files: nozzle.m, noz_mesh.m, noz_cfd.m, and solver.m. Three of the four files have been modified by Stoute:

- The nozzle.m has not been modified.

- The noz_mesh.m has been modified to include an exhaust section of the nozzle. Also, the nozzle has been changed from a bell-shaped nozzle to a conical nozzle, a full converging-diverging nozzle, and an inlet.

- The noz_cfd.m has been modified to include the magnetoplasmadynamic (MPD) equations (see section 2.5.5). The program has been adjusted to the RFET and to the Scramjet models.

- The solver.m has been modified to include the MPD equations embedded to find the solution. There is also a minimum timestep in the solver to prevent the program from crashing. Finally, the boundary conditions have been extended to include the ambient environment.

In addition to the modification to the original files, the author added files to compliment the CFD simulations. These include RFET.m and scramjet.m.

```
RFET.m

function [rho] = rfet(T_c, T_amb, P_c, P_amb, Cp, W, mu_f0, kth0, h_exh,
h_th, h_tip, V_dc, I_dc, I_ext, freq, activate_rail, Gas_Name)

%    Modified Navier-Stokes CFD solver using Magnetoplasmadynamic
%    equations for electrothermal thruster

%    Author Modified the Program:
%    C. A. Barry Stoute
%    Ph.D. Candidate
%    York University
%    Department of Earth and Space Science and Engineering

%    Original Author of Program:
%    Britton Jeffrey Olson
%    Ph.D. Candidate
%    Stanford Univeristy
%    Department of Aero/Astro

%%%%%%%    Introduction and Background from Original Author    %%%%%%%%
%    This program gives the ideal nozzle geometry using the method of
%    characteristics for a Quasi-2D Diverging Nozzle.  Assume gas is
%    exhausting from a combustion chamber that has no mass flow rate in.
%    Using 2D nozzle flow relations, an optimal throat area is found that
%    will produce the max amount of thrust for the given ambient pressure
%     and combustion chamber parameters.  This Area is automatically set
and
%    fed into the method of chactertistics portion of that code.  The
%    method of characteristics also uses the exit Mach number that
%    corresponds to the ideal exit area.

%%%%%%%    Introduction and Background from Modified Author    %%%%%%%%

%     Note, DO NOT call the returning Magnetic Field vector B, it will
crash
%    the simulation. Suggested name: Bmag

%%%%%%%    Directions for running the program    %%%%%%%%
%
%    Put in the desire values in the following order

%    T_c = Temperature in the combustion chamber (K)
%    T_amb = Ambient temperature (K)
%    P_c = Pressure in the combustion chamber (Pa)
%    P_amb = Ambient pressure (Pa)
%    Cp = Specific Heat (Pressure)
%    W = Molecular Mass
%    width = The diameter of the exhaust
%    h_th = The diameter of the throat
%    h_tip = The diameter of the tip (Cathode)
%    V_dc = The nominal input of the DC voltage to the circuit
```

```
%   I_dc = The nominal output of the DC current from the circuit
%   I_ext = External current supplying to the solenoid
%   freq = The desire frequency

%   Figure(1): Static thrust as function of exit area
%    Figure(2): Nozzle design and plots of Mach number & Pressure vs
Length
%   Figure(3): CFD simulation of designed nozzle (Mach Number)
%   Figure(4): CFD simulation of designed nozzle (Magnetic Field)

% Note, 1st row is the x direction (axial), 2nd row is the y direction
% radial, 3rd row is the z direction

% From nozzle.m

% clear all;
% clc;

% T_c = Temperature in the combustion chamber (K)
% P_c = Pressure in the combustion chamber (Pa)
% P_amb = Ambient pressure (Pa)
% T_amb = Ambient temperature (K)
% Cp = Specific Heat of Argon (Pressure)

%global P_amb
global gamma x y cfl Vol T_c nnoz R maxspeed minspeed boxex total_time
P_amb dx dy

rt = 0.5*h_th;
re = 0.5*h_exh;
width = rt;



%   Problem parameters
Torr = 133;
P_c = P_c*Torr;
P_amb = P_amb*Torr;

gamma = Cp/(Cp-8.314);    % Ratio of Specific Heats Cp/Cv (Gamma)
gammak = gamma;

mu0 = 4*pi*10^(-7);
e0 = 8.85E-12;

alpha_noz = 6;

noz_length        =        (rt*(re/rt-1)+1.5*rt*(sec(alpha_noz*pi/180)-
1))/tan(alpha_noz*pi/180);

% Speed Range
```

```matlab
maxspeed = 3000;
minspeed = 500;

%   Mesh Parameters

conical = 1;
boxex = 0;
d_element = 1;
total_time = 0;

%   CFD Portion parameters
cfl = 0.8;    %  Courant-Friedricks-Lewy stability factor (<1) (Default:
0.8)
tt = 400/cfl;    %  Number of time steps to take
init = 1;     %   Initialize  or  use  previous  run's  data?  [1-init  0-
previous]
% visit = 0;   %  Output viz file? [1-yes 0-no]


E = [0; -V_dc/(0.5*h_th-0.5*h_tip)];
Bb = [0*mu0*I_ext*50/noz_length; 0*mu0*I_dc/(2*pi*re)];
Bb0 = mu0*I_ext*50/noz_length;
EB = 0.5*(e0*E'*E+mu0^(-1)*Bb'*Bb);

Max_stress       =       (1/(4*pi))*[e0*E(1)*E(1)+mu0^(-1)*Bb(1)*Bb(1)-EB
e0*E(1)*E(2)+mu0^(-1)*Bb(1)*Bb(2);   e0*E(2)*E(1)+mu0^(-1)*Bb(2)*Bb(1)-EB
e0*E(2)*E(2)+mu0^(-1)*Bb(2)*Bb(2)-EB];
Res_dif = [E(1) 0; 0 E(2)];
visc_tens = [mu_f0 0; 0 mu_f0]; % Viscosity Tensor
kth = [kth0 0; 0 kth0]; % Thermal Conductivity Tensor

nozzle
noz_mesh
noz_cfd

median_speed = median(Speeds,1);
median_density = median(rho,1);
median_mag = median(Bmag,1);

for j = 1:size(u,2)
    for i = 1:size(u,1)
        if isnan(S_dir(i,j))
            S_dir(i,j) = 0;
            B_dir(i,j) = 0;
        end
        if Speeds(i,j) > median_speed(j)*2
            Speeds(i,j) = median_speed(j);
        end
        if rho(i,j) < 0
            rho(i,j) = 0;
            Speeds(i,j) = 0;
        elseif rho(i,j) > median_density(j)*3
```

```matlab
            rho(i,j) = median_density(j);
        else
        end
        if Bmag(i,j) > median_mag(j)*2
            Bmag(i,j) = median_mag(j);
        elseif isnan(Bmag(i,j))
            Bmag(i,j) = 0;
        end
        u(i,j) = Speeds(i,j)*cos(S_dir(i,j));
        v(i,j) = Speeds(i,j)*sin(S_dir(i,j));
        Bx(i,j) = Bmag(i,j)*cos(B_dir(i,j));
        By(i,j) = Bmag(i,j)*sin(B_dir(i,j));
    end
end

Isp = mean(Speeds(end,:))/9.81

%%    Plot the Velocity Vector field
% figure(12);clf;
%
% colormap(jet);
%
%
%   quiverc(x.*1000,y.*1000,u(1:size(p,1)-1,1:size(p,2)-1),v(1:size(p,1)-
1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
%               quiverc(x.*1000,-y.*1000,u(1:size(p,1)-1,1:size(p,2)-1),-
v(1:size(p,1)-1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
% axis([0 noz(end,1)*1020 -re*1100 re*1100]);
%
% h = colorbar;
% set(gca,'CLim', [minspeed maxspeed]);
%
%
% set(gcf, 'Position', [1 1080 1920 1080]);
% set(gcf,'PaperPositionMode','auto');
%
% ylabela = get(h,'YTickLabel');
% meterspersecond = repmat(' m/s',length(ylabela),1);
% ylabela = [ylabela meterspersecond];
% set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 48);
% set(h,'YTickLabel',ylabela, 'FontSize', 48);
%
% xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
% scrsz = get(0,'ScreenSize');
% print('-f12','-djpeg',['Velocity Vector Field of ' Gas_Name]);

%%    Plot the Pressure
% figure(13);clf;
%
```

```matlab
% surf(x.*1000,y.*1000,p(1:size(p,1)-1,1:size(p,2)-1));
% view(0,90);
%
% hold on;
%
% surf(x.*1000,-y.*1000,p(1:size(p,1)-1,1:size(p,2)-1));
% view(0,90);
%
% colormap(jet);
% set(gcf, 'Position', [1 1080 1920 1080]);
% set(gcf,'PaperPositionMode','auto');
% h = colorbar;
% set(gca, 'CLim', [0 600]);
% ylabela = get(h,'YTickLabel');
% meterspersecond = repmat(' Pa',length(ylabela),1);
% ylabela = [ylabela meterspersecond];
% set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 48);
% set(h,'YTickLabel',ylabela, 'FontSize', 48);
%
% xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
%
% print('-f13','-djpeg',['Pressure of ' Gas_Name]);


%%   Plot the Force
% figure(14);clf;
%
% F_u = u.^2.*rho.*((0.0025^2)*pi*0.25);
% F_v = v.^2.*rho.*((0.0025^2)*pi*0.25);
%
%                 quiverc(x.*1000,y.*1000,F_u(1:size(p,1)-1,1:size(p,2)-
1),F_v(1:size(p,1)-1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
%             quiverc(x.*1000,-y.*1000,F_u(1:size(p,1)-1,1:size(p,2)-1),-
F_v(1:size(p,1)-1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
% axis([0 noz(end,1)*1020 -re*1100 re*1100]);
%
% h = colorbar;
% % set(gca,'CLim', [minspeed maxspeed]);
%
%
% set(gcf, 'Position', [1 1080 1920 1080]);
% set(gcf,'PaperPositionMode','auto');
%
% ylabela = get(h,'YTickLabel');
% meterspersecond = repmat(' N',length(ylabela),1);
% ylabela = [ylabela meterspersecond];
% set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 48);
% set(h,'YTickLabel',ylabela, 'FontSize', 48);
%
% xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
```

225

```
% scrsz = get(0,'ScreenSize');
% print('-f14','-djpeg',['Force Vectors of ' Gas_Name]);

%%   Plot the Temperature
% figure(15);clf;
%
% surf(x.*1000,y.*1000,T(1:size(p,1)-1,1:size(p,2)-1));
% view(0,90);
%
% hold on;
%
% surf(x.*1000,-y.*1000,T(1:size(p,1)-1,1:size(p,2)-1));
% view(0,90);
%
% colormap(hot);
% set(gcf, 'Position', [1 1080 1920 1080]);
% set(gcf,'PaperPositionMode','auto');
% h = colorbar;
% set(gca, 'CLim', [200 500]);
% ylabela = get(h,'YTickLabel');
% meterspersecond = repmat(' K',length(ylabela),1);
% ylabela = [ylabela meterspersecond];
% set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 48);
% set(h,'YTickLabel',ylabela, 'FontSize', 48);
%
% xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
%
% print('-f15','-djpeg',['Temperature Distribution of ' Gas_Name]);

%%
clear global
```

NOZZLE.m

```
%   Method of Characteristics
num = 15;        % Number of Characteristic lines
theta_i = .03;   % Initial step in theta
plotter = 0;     % Set to '1' to plot nozzle

dh = h_th/100;
max_iter = 10000;
R = 8314/W;

%

%   Part A

%find where P becomes u
h(1) = h_th;
```

```matlab
A_star = h_th*width;
M =1;
dM1 = .1;
for i=1: max_iter
    h(i) = h(1) + (i-1)*dh;
    Ae(i) = h(i)*width;
    A_Asq = (Ae(i)/A_star)^2;
    A_ratio(i)=sqrt(A_Asq);

    %Newton Rhapson on Eq. 5.20 - Anderson text
    res = 1;
    if i > 1
        M = Ma(i-1);
    end

     while res > .001
        M2 = M + dM1;
        funa1    =    -A_Asq    +    (1/M^2)*((2/(gamma+1))*(1+(gamma-
1)*M^2/2))^((gamma+1)/(gamma-1));
        funa2    =    -A_Asq    +    (1/M2^2)*((2/(gamma+1))*(1+(gamma-
1)*M2^2/2))^((gamma+1)/(gamma-1));
        dv_dm = (funa2-funa1)/dM1;

        M = M - funa1/dv_dm;
        res = abs(funa1);

    end
    Ma(i) = M;

    % Find Pressure
    P(i) = P_c*(1+(gamma-1)*Ma(i)^2/2)^(-gamma/(gamma-1));

    % Find thrust for each point
    Te(i) = T_c/(1+(gamma-1)*Ma(i)^2/2);
    Tt(i) = T_c/(1+(gamma-1)/2);
    Ve(i) = Ma(i)*sqrt(Te(i)*gamma*R);
    Vt(i) = sqrt(Tt(i)*gamma*R);
    rhot(i) = P(i)/(R*Te(i));
    mdot(i) = rhot(i)*Ve(i)*Ae(i);
    TT(i) = mdot(i)*Ve(i) + (P(i) - P_amb)*Ae(i);

    if P(i) < P_amb
        %break
        %Calculate the pressure if shock wave exists at the exit plane
        P_exit = P(i)*(1+(gamma*2/(gamma+1))*(Ma(i)^2-1));

         if P_exit <= P_amb
             P(i) = P_exit;
             break
         else
         end
```

```matlab
        else
        end

    end

%  figure(2)
%  plot(Ae,TT)
%  title('Thrust curve')
%  xlabel('Exit Area (m^2)')
%  ylabel('Thrust (N)')

%    Part B
%    Determine the nominal exit area of the nozzle
%    to maximize thrust

[a,b]=max(TT);
%    Over or Underexpand the nozzle
%  b = b;
%  A_max = Ae(b);
%  Max_thrust = TT(b);
%  hold on;
%  plot(A_max,Max_thrust,'r*')
%  legend('Thrust Curve','Max Thrust')

%    Part C
%    Method of Characteristics

M_e = Ma(b);          %Mach number at ideal exit



%Find theta_max by using equation 11.33
%theta_max    =    (180/pi)*(sqrt((gamma+1)/(gamma-1))*atan((sqrt((gamma-
1)*(M_e^2-1)/(gamma+1))))-atan(sqrt(M_e^2-1)))/2;
theta_max = 15;

%  D_theta for each char line
del_theta = (theta_max - theta_i)/(num-1);

% Find

for i=1:num
    %    Initialize mach numeber

    for j=1:num
        if i==1
            %Theta for each line (first lines)
            theta(i,j) = theta_i + del_theta*(j-1);
            nu(i,j) = theta(i,j);
            K_m(i,j) = theta(i,j) + nu(i,j);
            K_p(i,j) = theta(i,j) - nu(i,j);
```

228

```matlab
        elseif i > 1

            K_p(i,j) = -K_m(1,i);

            % Find Thetas
            if j >= i
                theta(i,j) = del_theta*(j-i);
            else
                %theta(i,j) = theta(j,i-1);
                theta(i,j) = theta(j,i);

            end
            nu(i,j) = theta(i,j) - K_p(i,j);
            K_m(i,j) = theta(i,j) + nu(i,j);
        end

    % Prandtl-Meyer function (using Newton Rhapson)
    dM = .1; % Leave at about .1
    if j == 1
        M_ex(i,j) = 1.00;
    else
        M_ex(i,j) = M_ex(i,j-1);
    end
    M = M_ex(i,j);

    res = 1;
    while res > .01
        M2 = M + dM;
        funv1        =        (-nu(i,j)*(pi/180)+(sqrt((gamma+1)/(gamma-
1))*atan((sqrt((gamma-1)*(M^2-1)/(gamma+1))))-atan(sqrt(M^2-1))));
        funv2        =        (-nu(i,j)*(pi/180)+(sqrt((gamma+1)/(gamma-
1))*atan((sqrt((gamma-1)*(M2^2-1)/(gamma+1))))-atan(sqrt(M2^2-1))));
        dv_dm = (funv2-funv1)/dM;

        M = M - funv1/dv_dm;
        res = abs(funv1);

    end
    M_ex(i,j) = M;

    % Find the angle mu
    mu(i,j) = (180/pi)*asin(1/M_ex(i,j));

    end

    % Add last point to char line
    theta(i,num+1) = theta(i,num);
    nu(i,num+1) = nu(i,num);
    K_m(i,num+1) = K_m(i,num);
    K_p(i,num+1) = K_p(i,num);
end
```

```matlab
char = zeros(num,num+1,2);
for i=1:num

    for j=1:num+1

% Draw points of intersection
        %    Point 1 of all char lines
        if j == 1
            char(i,j,1) = 0;
            char(i,j,2) = h_th/2;
        end

        %    Where first line hits the symmetry line
        if i == 1 & j==2
            char(i,j,1)   =   (-h_th/2)/tan((pi/180)*(theta(1,j-1)-mu(1,j-
1)));
            char(i,j,2) = 0;
        end

        %    Where all other lines hit the symmetry line
        if j == i+1 & j>2
              char(i,j,1)   =  -char(i-1,j,2)/tan((pi/180)*(.5*theta(i,j-
2)-.5*(mu(i,j-2)+mu(i,j-1)))) + char(i-1,j,1);
              char(i,j,2) = 0;
              test(i,j) = (theta(i,j-2)-.5*(mu(i,j-2)+mu(i,j-1)));
              testpty(i,j) = char(i-1,j,2);
              testptx(i,j) = char(i-1,j,1);
        end

        %    All other data points for char 1 calculated
        if i ==1 & j>2 & j ~= i+1
            C_p       =        tan((pi/180)*(.5*(theta(i,j-2)+theta(i,j-
1))+.5*(mu(i,j-2)+mu(i,j-1))));
            C_m = tan((pi/180)*(.5*(theta(j-1,1)+theta(i,j-1))-.5*(mu(j-
1,1)+mu(i,j-1))));
              A = [1,-C_m;1,-C_p];
              B = [char(1,1,2) - char(1,1,1)*C_m;
                  char(1,j-1,2) - char(1,j-1,1)*C_p];
              iterm(1,:)=inv(A)*B;
              char(i,j,1) = iterm(1,2);
              char(i,j,2) = iterm(1,1);
        end

        %    All other points for all char lines calculated
        if i > 1 & j~=i+1 & j>2
            C_p       =        tan((pi/180)*(.5*(theta(i,j-2)+theta(i,j-
1))+.5*(mu(i,j-2)+mu(i,j-1))));
            C_m     =     tan((pi/180)*(.5*(theta(i-1,j-1)+theta(i,j-1))-
.5*(mu(i-1,j-1)+mu(i,j-1))));
              A = [1,-C_m;1,-C_p];
```

```matlab
            B = [char(i-1,j,2) - char(i-1,j,1)*C_m; char(i,j-1,2) -
char(i,j-1,1)*C_p];

            iterm(1,:) = inv(A)*B;
            char(i,j,1) = iterm(1,2);
            char(i,j,2) = iterm(1,1);
        end
    end
end


%  Fill in similar points (where char lines share points)
for i = 2:num
    for j=2:num
        char(j,i,1) = char(i-1,j+1,1);
        char(j,i,2) = char(i-1,j+1,2);
    end
end

% ******Make the nozzle shape and extend the char lines to wall******

%   Initial start point of the nozzle (at throat)
noz(1,1) = 0;
noz(1,2) = h_th/2;

%   Find all the points of the nozzle
%** This is where the shape of the nozzle is determine (Conical, Bell,
%etc...)
for i = 2 : num
    %   Find different slopes and points to intersect
    m1 = tan((pi/180)*(theta(i-1,num)+mu(i-1,num)));
    if i ==2
        m2 = (pi/180)*theta_max;
    else
        m2 = ((pi/180)*(theta(i-1,num+1)));
    end
    m3 = ((pi/180)*(theta(i-1,num)));
    m4 = tan((m2+m3)/2);

    A = [1,-m4; 1,-m1];
    B = [noz(i-1,2) - noz(i-1,1)*m4; char(i-1,num+1,2) - char(i-
1,num+1,1)*m1];

    iterm(1,:) = inv(A)*B;
    noz(i,1) = iterm(1,2);
    noz(i,2) = iterm(1,1);

    %   Extend char lines to wall
    char(i-1,num+2,1)= noz(i,1);
    char(i-1,num+2,2)= noz(i,2);
end
```

```matlab
%Last line
m1 = tan((pi/180)*(theta(num,num)+ mu(num,num)));
m2 = ((pi/180)*(theta(num-1,num)));
m3 = ((pi/180)*(theta(num,num+1)));
m4 = tan((m2+m3)/2);


A = [1,-m4; 1,-m1];
B   =   [noz(num,2)   -   noz(num,1)*m4;   char(num,num+1,2)   -
char(num,num+1,1)*m1];

iterm(1,:) = inv(A)*B;
noz(num+1,1) = iterm(1,2);
noz(num+1,2) = iterm(1,1);



%   Extend char lines to wall
char(num,num+2,1)= noz(num+1,1);
char(num,num+2,2)= noz(num+1,2);


if plotter ==1

%   Plot the nozzle shape
% figure(1);clf;
% subplot(2,1,1);
% plot(noz(:,1),noz(:,2),'k','LineWidth',3)
% hold on;
% [a,b] = max(noz);
% plot(a(1),A_max/width/2,'g*')
%
% %   Plot for loop for char lines
% for i = 1 : num
%     figure(1)
%     hold on;
%     plot(char(i,:,1),char(i,:,2))
%     hold on;
%     plot(char(i,:,1),-char(i,:,2))
% end
%
% %   Plot the nozzle shape (bottom side)
% figure(1)
% subplot(2,1,1)
% hold on;
% plot(noz(:,1),-noz(:,2),'k','LineWidth',3)
% hold on;
% plot(a(1),-A_max/width/2,'g*')
% title('Max Thrust (minimum length) Nozzle Design')
% xlabel('Nozzle length (m)')
% ylabel('Nozzle height (m)')
% legend('Nozzle shape','Area_e_x_i_t(predicted)','Char. Lines')
else
end
```

```matlab
%   Find  % errors in A/A* and Mexit
% error_Area = 100*(width*2*noz(num,2) - A_max)/(A_max);
% error_Mach = 100*(M_e - M_ex(num,num))/M_e;



%   Plot Mach Number and pressure through nozzle using the quasi-1D
%   area relations.  (Isentropic expansion through nozzle)
Mnoz(1) = 1.0;  %   Choked Flow
M = Mnoz(1);
for i=1: size(noz,1)
    Ae(i) = 2*noz(i,2)*width;
    A_Asq = (Ae(i)/A_star)^2;
    A_ratio(i)=sqrt(A_Asq);

    %Newton Rhapson on Eq. 5.20 - Anderson text
    res = 1;
    if i > 1
        M = Mnoz(i-1);


     while res > .001
        M2 = M + dM1;
        funa1    =    -A_Asq    +    (1/M^2)*((2/(gamma+1))*(1+(gamma-
1)*M^2/2))^((gamma+1)/(gamma-1));
        funa2    =    -A_Asq    +    (1/M2^2)*((2/(gamma+1))*(1+(gamma-
1)*M2^2/2))^((gamma+1)/(gamma-1));
        dv_dm = (funa2-funa1)/dM1;

        M = M - funa1/dv_dm;
        res = abs(funa1);

    end
    Mnoz(i) = M;
    end
    % Find Pressure
    Pnoz(i) = P_c*(1+(gamma-1)*Mnoz(i)^2/2)^(-gamma/(gamma-1));
end

% figure(1);
% subplot(2,1,2)
% plot(noz(:,1),Mnoz,'r*')
% hold on;
% plot(noz(:,1),Pnoz/P_amb,'b*')
% hold on;
% plot(noz(size(noz,1),1),M_e,'go')
% hold on;
% plot(noz(size(noz,1),1),1,'go')
% xlabel('Nozzle length (m)')
% ylabel('Mach number and P/P_c')
%                                                        legend('Mach
Number','P/P_a_m_b','M_e_x_i_t(predicted)','P_a_m_b/P_a_m_b')
```

NOZ_MESH.m (For RFET Only)

```matlab
%   ****** See nozzle.m for instructions ******
%  Program to extrapolate the data points from nozzle design and make a
%  uniform grid spacing in the x-direction
%  Change nothing... simply run this script


%  Find the minimum spacing given by the method of characteristics
%  and set as the dx value

left_edge = 0;
top_edge = 0;
right_edge = 0;

if conical == 1

    noz(1,1) = 0;
    noz(1,2) = rt;
    for i = 2:num+1
        noz(i,1) = noz(1,1) + i*noz_length/(num+1);
        noz(i,2) = rt + (re-rt)*(i)/(num+1);
    end
else
end

nacelle = noz.*1000;
noz_edge = size(noz,1);

if boxex == 1; % Include Exhaust Box?

    exbox(1,1) = noz(end,1);
    exbox(1,2) = noz(end,2);

    left_edge = size(exbox,1);

    for i = left_edge:left_edge+num+1
        exbox(i,1)    =    exbox(left_edge,1)    +    1.25*h_exh*(i-
left_edge)/(num+1);
        exbox(i,2) = exbox(left_edge,2);
    end

    top_edge = size(exbox,1);
    noz = vertcat(noz,exbox);

else
end
```

234

```matlab
dx = 0;
for i=1: size(noz,1)-1
    len = noz(i+1,1) - noz(i,1);
    if (len < dx || i == 1)
        if len > 0
            dx = len;
        end
    end
end

%  Explicitly give the dx value here
dx = max(noz(:,1))/ceil(max(noz(:,1))/dx);
dx = dx/d_element;

nnoz = ceil(max(noz(1:noz_edge,1))/dx);
n = ceil(max(noz(:,1))/dx);

% len = max(noz(:,1));
% n = 50;    %  Note # of points is actually n+1
% dx = len/n

%  Pick m points in y as some factor of x points
yfactor = 0.8;
m = ceil(yfactor*n);

%  Make uniform x-distribution of points
xmax = 0;
i = 1;
while xmax < max(noz(:,1))
    xmax = dx*(i-1);
    x(i,1:m) = xmax;

    if (xmax - noz(noz_edge,1) <= dx)
        noz_edge_x = i;
    end

    i = i+1;

end

%  Make the y-points and extrapolate linearly from closest points to fit
%  the nozzle geometry

%  Initialize and assign last value
y(1:size(x,1),1:size(x,2)) = 0; % y = zeros(size(x,1),size(x,2))
y(1,size(y,2)) = noz(1,size(noz,2)); % First row - Last Column of y is
equal to the first row - last column on noz (i.e. the y direction)
y(size(y,1),size(y,2)) = noz(size(noz,1),size(noz,2)); % Last row - Last
Column of y is equal to the last row - last column of noz
```

```matlab
% Barry's Modification: Changed the y meshing such that it takes care of
% the nozzle only.

for i = 1 : size(x,1)-1  % iteration of i from 1 to the size of the rows
of x minus 1 % Changed from minus 1 to minus 2

    j = 1;
    while x(i,1) >= noz(j,1)    % finding the smallest box of x
        x1 = noz(j,1);
        x2 = noz(j+1,1);
        y1 = noz(j,2);
        y2 = noz(j+1,2);

        j = j + 1;
    end

    slope = (y2 - y1)/(x2 - x1);  % finding the slope
    y(i,size(y,2)) = y1 + slope*(x(i,1)-x1);  % ith row - last column is
the y = mx + b such that b is y1 and x is the ith row of 1st column

    %Fill in mesh
    dy = y(i,size(y,2))/(size(y,2)-1);    % dy is derived from the ith
row and last column divide by the number of column in y - 1
    dy = dy/d_element;
    for k = 1: size(y,2)
        y(i,k) = dy*(k-1);  % this progress of dy over the columns
    end

end

% Comment left off

% Fill in mesh
dy = y(size(y,1),size(y,2))/(size(y,2)-1);  % filling bottom mesh
for k = 1: size(y,2)
    y(size(y,1),k) = dy*(k-1);
end

NOZ_CFD.m

%   ****** See nozzle.m for instructions ******
%   With the given grid from the nozzle code solve the flow
%   using MacCormack's finite volume method

%   Make some variables global  %% Barry's Mod: Brought globals at top



% global gamma R x y cfl Vol T_c nnoz
```

```matlab
%    Initialize the domain here
if (init == 1)
    n = size(x,1);
    m = size(x,2);

    p(1:n+1,1:m+1)  =  P_c*(1+(gamma-1)/2)^(-gamma/(gamma-1));  %  change
n+1 to nnoz+1 to represent the edge of the nozzle
%       p(nnoz+2:n+1,1:m+1) = P_amb;%*(1+(gamma-1)/2)^(-gamma/(gamma-1));
% add 26+2:n+1 to represent the open air

    T(1:n+1,1:m+1)  =  T_c*(1+(gamma-1)/2);  %  change  n+1  to  nnoz+1  to
represent the edge of the nozzle
%       T(nnoz+2:n+1,1:m+1) = T_amb; % add 26+2:n+1 to represent the open
air

    u(1:n+1,1:m+1) = 1.25*M_e*sqrt(gamma*R*T_c);   % change n+1 to nnoz+1
to represent the edge of the nozzle DEFAULT: 1.25*M_e*sqrt(gamma*R*T_c)
    v(1:n+1,1:m+1) = 0;

    rho = p./(R.*T);

    Bx(1:nnoz+1,1:m+1) = Bb(1);
    By(1:nnoz+1,1:m+1) = Bb(2);
    Bx(nnoz+2:n+1,1:m+1) = 0;
    By(nnoz+1:n+1,1:m+1) = 0;

    e   =  p/(gammak-1)   +   (1/2)*rho.*(u.*u  +  v.*v)  +  (1/2)*(10^(-
7))*(Bx.*Bx + By.*By)/(mu0);


    %   Cast into conservation form
    Q(:,:,1) = rho;      %  Conservation of Mass
    Q(:,:,2) = rho.*u;   %  Conservation of X-Momentum
    Q(:,:,3) = rho.*v;   %  Conservation of Y-Momentum
    Q(:,:,4) = e;        %  Conservation of Energy
    Q(:,:,5) = Bx;       %  Faraday's Law in X-Direction
    Q(:,:,6) = By;       %  Faraday's Law in Y-Direction

    %   Get and store the volumes and surface flux terms
    Vol(1:size(x,1)+1,1:size(x,2)+1) = 1;
    for i=1 : n-1
        for j=1 : m-1
        side1 = ( x(i,j)-x(i+1,j)  )*y(i+1,j+1) + ( x(i+1,j)-x(i+1,j+1)
)*y(i,j)...
            + ( x(i+1,j+1)-x(i,j)  )*y(i+1,j);
        side2 = ( x(i,j)-x(i+1,j+1)  )*y(i,j+1) + ( x(i+1,j+1)-x(i,j+1)
)*y(i,j)...
            + ( x(i,j+1)-x(i,j)  )*y(i+1,j+1);
        Vol(i+1,j+1) = (1/2)*( abs(side1) + abs(side2) );
        end
    end
```

```matlab
        end


%  Main iteration loop for integration in time
for k = 1: tt
    Q = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, tt,
freq);  %  Call the solver to advance one time step
%
%
%
%     k;                  %  Leave here for iteration counter
%
%     rho = Q(:,:,1);
%     u = Q(:,:,2)./rho;
%     v = Q(:,:,3)./rho;
%     e = Q(:,:,4);
%     Bx = Q(:,:,5);  % MPD
%     By = Q(:,:,6);  % MPD
%
%     Speeds = sqrt(u.^2 + v.^2);
%     S_dir = wrapToPi(atan(v./u));
%     p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
%
%     figure(5);clf;
%     set(gcf,'renderer','ZBuffer');
%                surf(x.*1000,y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%        hold on;surf(x.*1000,-y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%     colormap(jet);colorbar;
%           xlabel('(S)  Axial  -  Millimeters');ylabel('(S)  Radial  -
Millimeters');title(k);
end

rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);  % MPD
By = Q(:,:,6);  % MPD

p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));
Mach = sqrt(u.^2 + v.^2)./ss;
Bmag = sqrt(Bx.^2 + By.^2) + eps;
B_dir = wrapToPi(atan(By./Bx));
Speeds = sqrt(u.^2 + v.^2) + eps;
S_dir = wrapToPi(atan(v./u));
m_dot = Speeds.*rho*(0.0025^2)*pi*0.25;
```

238

```
Force = m_dot.*Speeds;


SOLVER.m (For RFET only)

%    ****** See nozzle.m for instructions ******
%    solver.m

%    Solver for the macormack method
function [Q] = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth,
tt, freq)
global gamma R

[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, tt, freq); % MPD are Bx By

%    Take one MacCormack step
[Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, tt, freq);


end



%%%%%%%    flowvars    %%%%%%%%
function  [u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,   E,   Bb,   EB,
Max_stress, Res_dif, visc_tens, kth, tt, freq) % MPD are Bx By
global gamma R viscosity Csuth Tsuth

%    Calculate the actual flow variables at each time step
rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);  % MPD
By = Q(:,:,6);  % MPD

Esin = E;%*sin(freq*2*pi*tt*10^(-7));


p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));

mu0 = 4*pi*10^(-7);
e0 = 8.85E-12;

mu_f = viscosity*((Tsuth+Csuth)./(T+Csuth)).*(T/Tsuth).^(3/2);

% visc_tens = [mu_f 0; 0 mu_f];

Hall    =    mu0^(-1)*[Bb(2)*(Bb(2).*u-Bb(1).*v);    -(Bb(1)*(Bb(2).*u-
Bb(1).*v))];
```

```matlab
MPD_e       =       [(Max_stress(1,1)-visc_tens(1,1)).*u+(Max_stress(1,2)-
visc_tens(1,2)).*v-Hall(1);(Max_stress(2,1)-
visc_tens(2,1)).*u+(Max_stress(2,2)-visc_tens(2,2)).*v-Hall(2)];


dx = 0.001; dy = 0.001;



ux = diff(u)./dx; ux = vertcat(ux, rand(1,size(ux,2)));
uy = diff(u)./dy; uy = vertcat(uy, rand(1,size(uy,2)));
vx = diff(v)./dx; vx = vertcat(vx, rand(1,size(vx,2)));
vy = diff(v)./dy; vy = vertcat(vy, rand(1,size(vy,2)));


stress_xx = -2/3*mu_f.*(ux+vy) + 2*mu_f.*(ux);
stress_yy = -2/3*mu_f.*(ux+vy) + 2*mu_f.*(vy);
stress_xy = mu_f.*(uy+vx);
stress_yx = stress_xy;

% Continuity
F(:,:,1) = rho.*u;
G(:,:,1) = rho.*v;

% X Momentum
F(:,:,2) = rho.*u.*u + p;% - 1*stress_xx - Max_stress(1,1);
G(:,:,2) = rho.*u.*v;% - 1*stress_xy - Max_stress(2,1);

% Y Momentum
F(:,:,3) = rho.*u.*v;% - 1*stress_yx - Max_stress(1,2);
G(:,:,3) = rho.*v.*v + p;% - 1*stress_yy - Max_stress(2,2);

% Energy
F(:,:,4) = (e+p).*u - (MPD_e(1));% - 0*u.*stress_xx - 0*v.*stress_xy;
G(:,:,4) = (e+p).*v - (MPD_e(2));% - 0*u.*stress_yy - 0*v.*stress_yx;

% X Faraday
F(:,:,5) = -Esin(1);      % MPD
G(:,:,5) = Bb(1).*v-Bb(2).*u;     % MPD

% Y Faraday
F(:,:,6) = -Bb(2).*u-Bb(1).*v;     % MPD
G(:,:,6) = -Esin(2);      % MPD



end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%    mac    %%%%%%%%

function [Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth,
tt, freq)
global gamma R %E Bb EB Max_stress Res_dif visc_tens kth
```

```matlab
Q0 = Q;

%   Forward flux
[Qflux,dt] = flux_mc(Q,-1, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, tt, freq);
Qbar = Q - dt*Qflux;
Q = Qbar;

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, tt,
freq);

%   Backward flux
[Qflux,dt] = flux_mc(Q,0, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, tt, freq);
Q = (1/2)*(Q0 + Qbar - dt*Qflux );

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, tt,
freq);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%    flux_mc    %%%%%%%%

function [Qflux,dt] = flux_mc(Q, dd, E, Bb, EB, Max_stress, Res_dif,
visc_tens, kth, tt, freq)
global x y Vol cfl total_time

[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, tt, freq);
nx = size(x,1);
ny = size(x,2);

a(1:nx+1,1:ny+1) = 0;
b(1:nx+1,1:ny+1) = 0;
c(1:nx+1,1:ny+1) = 0;
Qflux(1:nx+1,1:ny+1,1:6) = 0;

%   Get the fluxes
for i = 2: size(x,1)
    for j = 2: size(x,2)

    ii = i-1;
    jj = j-1;

    %   Right face
```

```matlab
        sfpx = y(ii+1,jj+1)-y(ii+1,jj);
        sfpy = -( x(ii+1,jj+1)-x(ii+1,jj) );

        %   Left face
        sfmx = -( y(ii,jj+1) - y(ii,jj) );
        sfmy = ( x(ii,jj+1)-x(ii,jj) );

        %   Top face
        sgpx = -( y(ii+1,jj+1) - y(ii,jj+1) );
        sgpy = x(ii+1,jj+1) - x(ii,jj+1);

        %   Bottom face
        sgmx = ( y(ii+1,jj)-y(ii,jj) );
        sgmy = -( x(ii+1,jj) - x(ii,jj) );

        %   Get the flux
        Qflux(i,j,:) = ( F(i+1+dd,j,:)*sfpx + G(i+1+dd,j,:)*sfpy + ...
            F(i+dd,j,:)*sfmx + G(i+dd,j,:)*sfmy + F(i,j+1+dd,:)*sgpx ...
            + G(i,j+1+dd,:)*sgpy + F(i,j+dd,:)*sgmx + G(i,j+dd,:)*sgmy );

        %   Normalize by Volume
        Qflux(i,j,:) = Qflux(i,j,:)./Vol(i,j);

        %   CFL terms
        a(i,j) = abs(u(i,j)*sfpx + v(i,j)*sfpy);
        b(i,j) = abs(u(i,j)*sgpx + v(i,j)*sgpy);
        c(i,j) = ss(i,j)*sqrt(abs( sfpx^2 + sfpy^2) ...
            + abs( sgpx^2 + sgpy^2) );

    end
end

invdt = max(max((a+b+c)./Vol));
dt = cfl/invdt;

if dt < 10^(-12/cfl)
    dt = 0;
end

% dt

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%    boundary    %%%%%%%
```

```matlab
function [Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, tt, freq)
global x y gamma R P_amb T_amb T_c boxex nnoz vel_mpd
[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, tt, freq);

% Barry's Modification: This is nozzle with exhaust
if boxex == 1


    % see Red Book of Vector Calculus & Partial Differential Equations
for
    % Gamma boundaries

    %   Problem boundary conditions here
    nx = size(x,1);
    ny = size(x,2);

    zx = nnoz; % Edge of Nozzle
    zy = ny; % Edge of Nozzle

    %   Top Wall
    p(:,ny+1) = p(:,ny);
%     p(zx+1:end,ny+1) = P_amb;

    v(:,ny+1) = 0;
    u(:,ny+1) = 0;

    rho(:,ny+1) = rho(:,ny);


    %   Symmetry line
    p(:,1) = p(:,2);

    v(:,1) = -v(:,2);
    u(:,1) = u(:,2);

    rho(:,1) = rho(:,2);
    Bx(:,1) = Bx(:,2);      % MPD
    By(:,1) = -By(:,2);     % MPD

    %   Inflow-shouldn't change from initialization
    u(1,:) = sqrt(gamma*R*T_c);
    v(1,:) = v(2,:);

    Bx(1,:) = Bx(2,:);  % MPD
    By(1,:) = By(2,:);  % MPD

    %   Out flow - set to upstream cells
    u(nx+1,:) = u(nx,:);
    v(nx+1,:) = v(nx,:);
```

243

```matlab
    p(nx+1,:) = p(nx,:);
%      p(zx+1,:) = p(zx,:);
%      p(nx+1,:) = P_amb;

    rho(nx+1,:) = rho(nx,:);
%      Bx(nx+1,:) = Bx(nx,:);   % MPD
%      By(nx+1,:) = By(nx,:);   % MPD

    %    EOS
    mu0 = 4*pi*10^(-7);
    e  =  p/(gamma-1)  +  (1/2)*rho.*(u.*u  +  v.*v)  +  0.5*(Bx.*Bx  +
By.*By)/mu0;

    Q(:,:,1) = rho;
    Q(:,:,2) = rho.*u;
    Q(:,:,3) = rho.*v;
    Q(:,:,4) = e;
    Q(:,:,5) = Bx;   % MPD
    Q(:,:,6) = By;   % MPD
  % Barry's Modification: This is the nozzle from the original program
else
    %    Problem boundary conditions here
    nx = size(x,1);
    ny = size(x,2);

    %    Top Wall
    p(:,ny+1) = p(:,ny);
    v(:,ny+1) = 0;
    u(:,ny+1) = 0;
    rho(:,ny+1) = rho(:,ny);
    Bx(:,ny+1) = Bx(:,ny); % MPD
    By(:,ny+1) = By(:,ny); % MPD


    %    Symmetry line
    p(:,1) = p(:,2);
    v(:,1) = -v(:,2);
    u(:,1) = u(:,2);
    rho(:,1) = rho(:,2);
    Bx(:,1) = Bx(:,2);      % MPD
    By(:,1) = -By(:,2);     % MPD

    %    Inflow-shouldn't change from initialization
    u(1,:) = sqrt(gamma*R*T_c);
    v(1,:) = v(2,:);
    Bx(1,:) = Bx(2,:);   % MPD
    By(1,:) = By(2,:);   % MPD

    %    Out flow - set to upstream cells
    u(nx+1,:) = u(nx,:);
    v(nx+1,:) = v(nx,:);
```

```matlab
    p(nx+1,:) = p(nx,:);   %P_amb;
    rho(nx+1,:) = rho(nx,:);
    Bx(nx+1,:) = Bx(nx,:);   % MPD
    By(nx+1,:) = By(nx,:);   % MPD


    %   EOS
    mu0 = 4*pi*10^(-7);
    e  =  p/(gamma-1)  +  (1/2)*rho.*(u.*u  +  v.*v)  +  0.5*(Bx.*Bx  +
By.*By)/mu0;

    Q(:,:,1) = rho;
    Q(:,:,2) = rho.*u;
    Q(:,:,3) = rho.*v;
    Q(:,:,4) = e;
    Q(:,:,5) = Bx;   % MPD
    Q(:,:,6) = By;   % MPD


end


end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


SCRAMJET.m



function [Speeds] = scramjet(T_c, T_amb, P_c, P_amb, Cp, W, mu_f0,
h_exh, h_th, h_tip, V_dc, I_dc, I_ext, freq, is_mpd, is_es, Gas_Name,
He_Content, E_i)


%   Modified Navier-Stokes CFD solver using Magnetoplasmadynamic
%   equations for electrothermal thruster

%   Author Modified the Program:
%   C. A. Barry Stoute
%   Ph.D. Candidate
%   York University
%   Department of Earth and Space Science and Engineering

%   Original Author of Program:
%   Britton Jeffrey Olson
%   Ph.D. Candidate
%   Stanford Univeristy
%   Department of Aero/Astro

%%%%%%%   Introduction and Background from Original Author   %%%%%%%
%   This program gives the ideal nozzle geometry using the method of
%   characteristics for a Quasi-2D Diverging Nozzle.  Assume gas is
%   exhausting from a combustion chamber that has no mass flow rate in.
%   Using 2D nozzle flow relations, an optimal throat area is found that
%   will produce the max amount of thrust for the given ambient pressure
```

```matlab
%   and combustion chamber parameters.  This Area is automatically set
and
%   fed into the method of charactertistics portion of that code.  The
%   method of characteristics also uses the exit Mach number that
%   corresponds to the ideal exit area.

%%%%%%%    Introduction and Background from Modified Author   %%%%%%%%

%   Note, DO NOT call the returning Magnetic Field vector B, it will
crash
%   the simulation. Suggested name: Bmag

%%%%%%%    Directions for running the program     %%%%%%%%
%
%   Put in the desire values in the following order

%   T_c = Temperature in the combustion chamber (K)
%   T_amb = Ambient temperature (K)
%   P_c = Pressure in the combustion chamber (Pa)
%   P_amb = Ambient pressure (Pa)
%   Cp = Specific Heat (Pressure)
%   W = Molecular Mass
%   width = The width of the nozzle
%   h_th = The diameter of the throat
%   h_exh = The diameter of the exhaust
%   h_tip = The diameter of the tip (Cathode)
%   V_dc = The nominal input of the DC voltage to the circuit
%   I_dc = The nominal output of the DC current from the circuit
%   I_ext = External current supplying to the solenoid
%   freq = The desire frequency

%   Figure(1): Static thrust as function of exit area
%    Figure(2): Nozzle design and plots of Mach number & Pressure vs
Length
%   Figure(3): CFD simulation of designed nozzle (Mach Number)
%   Figure(4): CFD simulation of designed nozzle (Magnetic Field)

% Note, 1st row is the x direction (axial), 2nd row is the y direction
% radial, 3rd row is the z direction

% From nozzle.m

% clear all;
% clc;

% T_c = Temperature in the combustion chamber (K)
% P_c = Pressure in the combustion chamber (Pa)
% P_amb = Ambient pressure (Pa)
% T_amb = Ambient temperature (K)
% Cp = Specific Heat of Argon (Pressure)
```

```matlab
global gamma x y cfl Vol T_c nin ns na ng nout R maxs mins boxex P_amb
omega k Bb0 v0 sigma_dc impulse_time mi rt
% Thruster's Physical Properties
rt = 0.5*h_th;
re = 0.5*h_exh;
width = h_th;


inlet_length = 5/1000; %25mm
inlet_radius = 7.5/1000;
entrance_height = inlet_radius-rt;
entrance_length = 15/1000; %15mm
mid_length = 10/1000; % 3mm


Lchar = 4*pi*0.005^2/(2*pi*0.005);


ra = inlet_radius;
rc = 0.5*h_tip;
rg = ra-rc;
cmpd = 2*pi*(ra+rc)/2;


%   Problem parameters
Torr = 133;
P_c = P_c*Torr;
P_amb = P_amb*Torr;


gamma = Cp/(Cp-8.314);    % Ratio of Specific Heats Cp/Cv (Gamma)
gammak = gamma;


impulse_time = 0;


mu0 = 4*pi*10^(-7);
e0 = 8.85E-12;


dh = h_th/100;
max_iter = 10000;
R = 8314/W;
omega = freq*2*pi*10^6;


kB = 1.38*10^(-23);
epsilon0 = 8.85*10^(-12);
me = 9.109*10^(-31);
mp = 1.672600000*10^(-27);
mi = mp*W;
h_planck = 6.626200000*10^(-34);
e_charge = 1.602*10^(-19);
%%
% Meshing
conical = 1;
boxex = 1;
d_element = 1;


%   CFD Portion parameters
```

```matlab
cfl = 0.07;   %  Courant-Friedricks-Lewy stability factor (<1) (Default:
0.8)
tt = 500;    %  Number of time steps to take (Default: 500)
init = 1;      %   Initialize  or  use  previous  run's  data?  [1-init  0-
previous]
CFD_weight = 12;% The program crashes when the velocity gets too high.
The mach number is calculated under heated gas. However, the speed of
sound in plasma is completely different. See thermal velocity of a
plasma. To ensure proper velocities, this weight function is
implemented.



%%   Electromagnetic Propulsion

Fmaeckers = I_dc^(2)*mu0*log(ra/rc+0.75)*(4*pi)^(-1);
m_dot = 5*10^(-6);

%                           alpha_ionization                            =
(mi/(e_charge*E_i))*(4/(9*sqrt(3)))*((cmpd/rg)*(mu0*I_dc^2)/m_dot)^2;


vel_mpd = sqrt(2*E_i*e_charge/mi)


alpha_noz = 6;


noz_length       =       (rt*(re/rt-1)+1.5*rt*(sec(alpha_noz*pi/180)-
1))/tan(alpha_noz*pi/180);

%% Min and Max Limits of the Colormap
v0 = vel_mpd;
maxspeed = 50;
minspeed = 10;



E = [0; 0];
E0 = sqrt(E(1)^2+E(2)^2);
Bb = [0.5*mu0*12*I_ext/0.003; 0];
Bb0 = sqrt(Bb(1)^2+Bb(2)^2)


EB = 0.5*(e0*(E(1)^2+E(2)^2)+mu0^(-1)*(Bb(1)^2+Bb(2)^2));

Max_stress       =        (1/(4*pi))*[e0*E(1)*E(1)+mu0^(-1)*Bb(1)*Bb(1)-EB
e0*E(1)*E(2)+mu0^(-1)*Bb(1)*Bb(2);   e0*E(2)*E(1)+mu0^(-1)*Bb(2)*Bb(1)-EB
e0*E(2)*E(2)+mu0^(-1)*Bb(2)*Bb(2)-EB];


Res_dif = [E(1) 0; 0 E(2)];


visc_tens = [mu_f0 0; 0 mu_f0]; % Viscosity Tensor


%% Finding the Plasma Conductivity and the Thermal Energy
```

```matlab
Te = 1.7;
T_e = Te*e_charge/kB;
n0_N = (1-He_Content)*P_c/(kB*T_amb);
n0_He = He_Content*P_c/(kB*T_amb);
n0 = n0_N+n0_He;


SahaN = 2.410E21 * T_e^(3/2)*exp(-e_charge*15.8/(kB*T_e));
SahaHe = 2.410E21 * T_e^(3/2)*exp(-e_charge*24.59/(kB*T_e));


ni_N = n0_N/(1+sqrt(1+n0_N/SahaN));
nn_N = ni_N*ni_N/SahaN;


ni_He = n0_He/(1+sqrt(1+n0_He/SahaHe));
nn_He = ni_He*ni_He/SahaHe;


ne = ni_N + ni_He;
ni = ne;
nn = nn_He+nn_N;


ce = sqrt(8*kB*T_e/(pi*me));
lambda_d = sqrt(Te*epsilon0/(e_charge*ne));
big_lambda = (4*pi*ne*lambda_d^3);


nu_ei = ni*e_charge^4*log(big_lambda)/(2*pi*epsilon0^2*me^2*ce^3);
nu_en = nn*ce*2*10^(-19);
nu_e = nu_ei+nu_en;


sigma_dc = ne*e_charge^2/(me*nu_e);


k = 3.2*e_charge*kB*T_e*n0/(nu_e*me);
kth = [k 0; 0 k];
%%

Pabs                                                               =
0.5*(100E6*2*pi*rt/(2*sqrt(2))*Bb0)^2*sigma_dc*(nu_e^2)/((100E6*2*pi)^2+
nu_e^2);
Maggy = Bb0^2*rt^2*pi*0.003/(2*mu0);



%%
nozzle
noz_mesh
noz_cfd

median_speed = median(Speeds,1);
median_density = median(rho,1);
median_mag = median(Bmag,1);

for j = 1:size(u,2)
    for i = 1:size(u,1)
        if isnan(S_dir(i,j))
```

249

```matlab
                S_dir(i,j) = 0;
                B_dir(i,j) = 0;
            end
            if Speeds(i,j) > median_speed(j)*5
                Speeds(i,j) = median_speed(j);
            end
            if rho(i,j) < 0
                rho(i,j) = 0;
                Speeds(i,j) = 0;
            elseif rho(i,j) > median_density(j)*5
                rho(i,j) = median_density(j);
            else
            end
            if Bmag(i,j) > median_mag(j)*2
                Bmag(i,j) = median_mag(j);
            elseif isnan(Bmag(i,j))
                Bmag(i,j) = 0;
            end
            u(i,j) = Speeds(i,j)*cos(S_dir(i,j));
            v(i,j) = Speeds(i,j)*sin(S_dir(i,j));
            Bx(i,j) = Bmag(i,j)*cos(B_dir(i,j));
            By(i,j) = Bmag(i,j)*sin(B_dir(i,j));
        end
end

vex = mean(Speeds(noz_edge,:))
% Ibit = vex*m_dot*impulse_time
% mbit = impulse_time*m_dot
% Isp = vex/9.81
% mRe = mu0*Lchar*sigma_dc*vex

Impulse = impulse_time*m_dot*Speeds;
Iu = rho.*u;
Iv = rho.*v;



%%   Plot the Velocity vector field
figure(12);clf;

maxs = maxspeed;
mins = minspeed;

colormap(jet);

quiverc(x.*1000,y.*1000,u(1:size(p,1)-1,1:size(p,2)-
1)./1000,v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0
%     plot(heater(:,1),heater(:,2),'Color','r','LineWidth',3);
% end
```

```matlab
quiverc(x.*1000,-y.*1000,u(1:size(p,1)-1,1:size(p,2)-1)./1000,-
v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0
%     plot(heater(:,1),-heater(:,2),'Color','r','LineWidth',3);
% end
axis([0 noz(end,1)*1050 -re*1100 re*1100]);

h = colorbar;
set(gca,'CLim', [mins maxs]);


set(gcf, 'Position', [1 1080 1920 1080]);
set(gcf,'PaperPositionMode','auto');


ylabela = get(h,'YTickLabel');
meterspersecond = repmat(' km/s',length(ylabela),1);
ylabela = [ylabela meterspersecond];
set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 36);
set(h,'YTickLabel',ylabela, 'FontSize', 36);


xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
scrsz = get(0,'ScreenSize');
print('-f12','-djpeg',['Velocity Vector Field of ' Gas_Name]);

%%   Plot the Impulse Density
% figure(13);clf;
%
% maxmag = Ibit;
% minmag = 0;
%
% maxs = maxmag;
% mins = minmag;
%
% colormap(jet);
%
%                   quiverc(x.*1000,y.*1000,Iu(1:size(p,1)-1,1:size(p,2)-
1),Iv(1:size(p,1)-1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0
%     plot(heater(:,1),heater(:,2),'Color','r','LineWidth',3);
% end
%                quiverc(x.*1000,-y.*1000,Iu(1:size(p,1)-1,1:size(p,2)-1),-
Iv(1:size(p,1)-1,1:size(p,2)-1));%view(0,90);
% hold on;
% plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0
%     plot(heater(:,1),-heater(:,2),'Color','r','LineWidth',3);
% end
% axis([0 noz(end,1)*1050 -re*1100 re*1100]);
```

```matlab
%
% h = colorbar;
% set(gca,'CLim', [mins maxs]);
%
%
% set(gcf, 'Position', [1 1080 1920 1080]);
% set(gcf,'PaperPositionMode','auto');
%
% ylabela = get(h,'YTickLabel');
% meterspersecond = repmat(' Ns/m^3',length(ylabela),1);
% ylabela = [ylabela meterspersecond];
% set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 44);
% set(h,'YTickLabel',ylabela, 'FontSize', 44);
%
% xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
% scrsz = get(0,'ScreenSize');
% print('-f13','-djpeg',['Impulse Density of ' Gas_Name]);
%
% clear global
% clear variables


NOZ_MESH.m (for SCRAMJET ONLY)

%   ****** See nozzle.m for instructions ******
%  Program to extrapolate the data points from nozzle design and make a
%  uniform grid spacing in the x-direction
%  Change nothing... simply run this script

%% BARRY'S ADDON: Adjusting Nozzle to incorporate inlet

cnoz = noz;    % Renaming the original nozzle

clear noz

% Adding Inlet to Nozzle
noz(1,1) = 0;
noz(1,2) = inlet_radius;
noz(2,1) = inlet_length;
noz(2,2) = noz(1,2);

inlet_i = size(noz,1);

noz(3,1) = noz(2,1) + entrance_length;
noz(3,2) = rt;

entrance_i = size(noz,1);

noz(4,1) = noz(3,1) + mid_length;
noz(4,2) = noz(3,2);
```

```matlab
mid_i = size(noz,1);



%% Barry's Addon: Replacing Bell Nozzle with Conical Nozzle

if conical == 1

    for i = 5:num+1
        noz(i,1) = noz(4,1) + i*noz_length/(num+1);
        noz(i,2) = rt + (re-rt)*i/(num+1);
    end
elseif conical == 0
else
    % Shifting the nozzle
    noz = vertcat(noz,cnoz);
     for i = 4:size(noz,1)
         noz(i,1) = noz(i,1)+noz(4,1);
     end

    % Shifting char lines
%     for i = 1:size(char,1)
%         for j = 1:size(char,2)
%             char(i,j,1) = char(i,j,1) + noz(4,1);
%         end
%     end
end

noz_edge = size(noz,1);

nacelle = noz*1000;

heater = noz(3:4,1:2)*1000;

if boxex == 1;

    exbox(1,1) = noz(end,1);
    exbox(1,2) = noz(end,2);

    left_edge = size(exbox,1);

    for i = left_edge:left_edge+num+1
        exbox(i,1)    =    exbox(left_edge,1)    +    1.25*h_exh*(i-
left_edge)/(num+1);
        exbox(i,2) = exbox(left_edge,2);
    end

    top_edge = size(exbox,1);
    noz = vertcat(noz,exbox);

else
```

253

```matlab
    end


    %%

    %  Find the minimum spacing given by the method of characteristics
    %  and set as the dx value
    dx = 0;
    for i=1: size(noz,1)-1
        len = noz(i+1,1) - noz(i,1);
        if (len < dx || i == 1)
            if len > 0
                dx = len;
            end
        end
    end
    %%
    %  Explicitly give the dx value here
    dx = max(noz(:,1))/ceil(max(noz(:,1))/dx);

    if dx < 0.003
        dx = 0.003;
    else
    end

    dx = dx/d_element;


    %dx = 0.001;
    nshift = 2;
    ninlet = ceil(max(noz(1:inlet_i,1))/dx) - nshift;
    nentrance = ceil(max(noz(1:entrance_i,1))/dx) - nshift;
    nmid = ceil(max(noz(1:mid_i,1))/dx) - nshift;
    nnoz = ceil(max(noz(1:noz_edge,1))/dx);
    n = max(noz(:,1))/dx;

    % len = max(noz(:,1));
    % n = 50;    %  Note # of points is actually n+1
    % dx = len/n;

    %  Pick m points in y as some factor of x points
    yfactor = .8;
    m = ceil(yfactor*n);

    %  Make uniform x-distribution of points
    xmax = 0;
    i = 1;
    while xmax < max(noz(:,1))
        xmax = dx*(i-1);
        x(i,1:m) = xmax;
        i = i+1;
    end
```

```
%xmax = 0.03;

%  Make the y-points and extrapolate linearly from closest points to fit
%  the nozzle geometry

%  Initialize and assign last value
y(1:size(x,1),1:size(x,2)) = 0;
y(1,size(y,2)) = noz(1,size(noz,2));
y(size(y,1),size(y,2)) = noz(size(noz,1),size(noz,2));
for i = 1 : size(x,1)-1

    j = 1;
    while x(i,1) >= noz(j,1)
        x1 = noz(j,1);
        x2 = noz(j+1,1);
        y1 = noz(j,2);
        y2 = noz(j+1,2);
        j = j + 1;
    end

    slope = (y2 - y1)/(x2 - x1);
    y(i,size(y,2)) = y1 + slope*(x(i,1)-x1);

    %Fill in mesh
    dy = y(i,size(y,2))/(size(y,2)-1);
    for k = 1: size(y,2)
        y(i,k) = dy*(k-1);
    end

end

%Fill in mesh
dy = y(size(y,1),size(y,2))/(size(y,2)-1);
for k = 1: size(y,2)
    y(size(y,1),k) = dy*(k-1);
end

NOZ_CFD.m (For SCRAMJET ONLY)

%   ****** See nozzle.m for instructions ******
%   With the given grid from the nozzle code solve the flow
%   using MacCormack's finite volume method




%   Initialize the domain here
n = size(x,1);
m = size(x,2);

p(1:n+1,1:m+1) = P_c*(1+(gamma-1)/2)^(-gamma/(gamma-1));
```

```matlab
T(1:n+1,1:m+1) = T_c*(1+(gamma-1)/2);


rho = p./(R.*T);


u(1:n+1,1:m+1) = (v0+e_charge*Bb0*rt/mi);
u(1:nmid+1,1:m+1) = 0.8*(v0+e_charge*Bb0*rt/mi);
u(1:nentrance+1,1:m+1) = (v0);
v(1:n+1,1:m+1) = 0;


Bx(1:n+1,1:m+1) = 0;      % MPD
By(1:n+1,1:m+1) = 0;      % MPD
Bx(nentrance-1:nmid+1,1:m+1) = Bb(1);     % MPD
By(nentrance-1:nmid+1,1:m+1) = Bb(2);     %MPD


e  =  p/(gammak-1)  +  (1/2)*rho.*(u.*u  +  v.*v)  +  (1/2)*(Bx.*Bx  +
By.*By)/(mu0);


% St = f0*Lchar/v0; % St is scalar
% Eu = (Pthroat-P_amb)./(rho0.*v0^2); % Eu is a matrix
% Mst = Max_stress0./(rho0.*v0^2); % Mst is a matrix
% Mv = v0/(f0*Lchar); % Mv is scalar
% Ea = Lchar.*e0*f0./(rho0*v0^3);
% Eb = ee0./(rho0.*v0^2);

%   Cast into conservation form
Q(:,:,1) = rho;      %  Conservation of Mass
Q(:,:,2) = rho.*u;   %  Conservation of X-Momentum
Q(:,:,3) = rho.*v;   %  Conservation of Y-Momentum
Q(:,:,4) = e;        %  Conservation of Energy
Q(:,:,5) = Bx;       %  Faraday's Law in X-Direction     % MPD
Q(:,:,6) = By;       %  Faraday's Law in Y-Direction     % MPD

%   Get and store the volumes and surface flux terms
Vol(1:size(x,1)+1,1:size(x,2)+1) = 1;
for i=1 : n-1
    for j=1 : m-1
        side1 = ( x(i,j)-x(i+1,j) )*y(i+1,j+1) + ( x(i+1,j)-x(i+1,j+1)
)*y(i,j)...
               + ( x(i+1,j+1)-x(i,j) )*y(i+1,j);
        side2 = ( x(i,j)-x(i+1,j+1) )*y(i,j+1) + ( x(i+1,j+1)-x(i,j+1)
)*y(i,j)...
               + ( x(i,j+1)-x(i,j) )*y(i+1,j+1);
        Vol(i+1,j+1) = (1/2)*( abs(side1) + abs(side2) );
    end
end


%   Main iteration loop for integration in time
for k = 1: tt
```

```matlab
    Q = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v0, Vol, I_ext);  %  Call the solver to advance one time step
%     rho = Q(:,:,1);
%     u = CFD_weight.*Q(:,:,2)./rho;
%     v = CFD_weight.*Q(:,:,3)./rho;
%
%     Speeds = sqrt(u.^2 + v.^2);
%
%     figure(5);clf;
%     set(gcf,'renderer','ZBuffer');
%              surf(x.*1000,y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%        hold  on;surf(x.*1000,-y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%     colormap(jet);colorbar;
%           xlabel('(S)  Axial  -  Millimeters');ylabel('(S)  Radial  -
Millimeters');
%     title(k);
end

%  Get the flow variables back from conserved variables

rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);  % MPD
By = Q(:,:,6);  % MPD

p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));
Mach = sqrt(u.^2 + v.^2)./ss;

Bmag = sqrt(Bx.^2 + By.^2);
B_dir = wrapToPi(atan(By./Bx));
Speeds = sqrt(u.^2 + v.^2);
S_dir = wrapToPi(atan(v./u));

SOLVER.m  (FOR SCRAMJET ONLY)

%  ****** See nozzle.m for instructions ******
%  solver.m

%  Solver for the macormack method
function [Q] = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth,
x, y, v0, Vol, I_ext)
global gamma R

[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, x, y, v0, Vol, I_ext); % MPD are Bx By
```

```matlab
%   Take one MacCormack step
[Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y, v0,
Vol, I_ext);


end



%%%%%%%    flowvars    %%%%%%%
function  [u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,
Max_stress, Res_dif, visc_tens, kth, x, y, v0, Vol, I_ext) % MPD are Bx
By
global gamma R omega cfl k sigma_dc mi ninlet nentrance nmid nnoz

nx = size(x,1);
ny = size(x,2);

ix = ninlet;
ex = nentrance;
mx = nmid;
zx = nnoz;

iy = ny;
ey = ny;
my = ny;
zy = ny;




%   Calculate the actual flow variables at each time step using
%%  STATIC MAGNETIC FIELD
% rho = Q(:,:,1);
% u = Q(:,:,2)./rho;
% v = Q(:,:,3)./rho;
% e = Q(:,:,4);
% Bx = Q(:,:,5);   % MPD
% By = Q(:,:,6);   % MPD
%
% p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
% T = p./(R*rho);
% ss = sqrt(abs(gamma*R*T));
%
% mu0 = 4*pi*10^(-7);
% e0 = 8.85E-12;
%
% Max_stress_xx = mu0^(-1)*(Bb(1)*Bb(1) - 0.5*(Bb(1)^2+Bb(2)^2));
% Max_stress_xy = mu0^(-1)*Bb(1)*Bb(2);
% Max_stress_yx = mu0^(-1)*Bb(1)*Bb(1);
% Max_stress_yy = mu0^(-1)*(Bb(2)*Bb(2) - 0.5*(Bb(1)^2+Bb(2)^2));
%
% MPD_e_x = Max_stress_xx.*u + Max_stress_xy.*v;% + kth(1,1).*T;
% MPD_e_y = Max_stress_yx.*u + Max_stress_yy.*v;% + kth(2,2).*T;
```

258

```matlab
%
% F(:,:,1) = rho.*u;
% F(:,:,2) = rho.*u.*u + p - Max_stress(1,1);
% F(:,:,3) = rho.*u.*v - Max_stress(1,2);
% F(:,:,4) = (e+p).*u-(MPD_e_x);
% F(:,:,5) = -E(2);      % MPD
% F(:,:,6) = (Bb(1).*u-Bb(1).*v);    % MPD
%
% G(:,:,1) = rho.*v;
% G(:,:,2) = rho.*u.*v - Max_stress(2,1);
% G(:,:,3) = rho.*v.*v + p - Max_stress(2,2);
% G(:,:,4) = (e+p).*v-(MPD_e_y);
% G(:,:,5) = (Bb(1).*v-Bb(2).*u);    % MPD
% G(:,:,6) = -E(1);     % MPD
%

%%  DYNAMIC MAGNETIC FIELD
rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);   % MPD
By = Q(:,:,6);   % MPD

p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));

mu0 = 4*pi*10^(-7);
e0 = 8.85E-12;
e_charge = 1.602E-19;

Max_stress_xx = mu0^(-1)*(Bx.*Bx - 0.5*(Bx.^2+By.^2));
Max_stress_xy = mu0^(-1)*Bx.*By;
Max_stress_yx = mu0^(-1)*By.*Bx;
Max_stress_yy = mu0^(-1)*(By.*By - 0.5*(Bx.^2+By.^2));

MPD_e_x = Max_stress_xx.*u + Max_stress_xy.*v;
MPD_e_y = Max_stress_yx.*u + Max_stress_yy.*v;


% Continuity
F(:,:,1) = rho.*u;
G(:,:,1) = rho.*v;

% X Momentum
F(:,:,2) = rho.*u.*u + p - Max_stress_xx;
G(:,:,2) = rho.*u.*v - Max_stress_xy;

% Y Momentum
F(:,:,3) = rho.*u.*v - Max_stress_yx;
G(:,:,3) = rho.*v.*v + p - Max_stress_yy;
```

```matlab
% Energy
F(:,:,4) = (e+p).*u - (MPD_e_x);
G(:,:,4) = (e+p).*v - (MPD_e_y);


% X Faraday
F(:,:,5) = -E(2);       % MPD
G(:,:,5) = Bx.*v-By.*u;     % MPD


% Y Faraday
F(:,:,6) = By.*u-Bx.*v;     % MPD
G(:,:,6) = -E(1);       % MPD


end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



%%%%%%%    mac    %%%%%%%%

function [Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x,
y, v0, Vol, I_ext)
global gamma R

Q0 = Q;

%   Forward flux
[Qflux,dt] = flux_mc(Q,-1, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v0, Vol, I_ext);
Qbar = Q - dt*Qflux;
Q = Qbar;

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v0, Vol, I_ext);

%   Backward flux
[Qflux,dt] = flux_mc(Q,0, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v0, Vol, I_ext);
Q = (1/2)*(Q0 + Qbar - dt*Qflux );

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v0, Vol, I_ext);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%% Barry's Note: Finite Difference Method used in finding the Flux of Q
%%%%%%%    flux_mc    %%%%%%%
function [Qflux,dt] = flux_mc(Q, dd, E, Bb, EB, Max_stress, Res_dif, 
visc_tens, kth, x, y, v0, Vol, I_ext)
global cfl impulse_time

[u,v,rho,p,e,Bx,By,T,ss,F,G] = flowvars(Q, E, Bb, EB, Max_stress, 
Res_dif, visc_tens, kth, x, y, v0, Vol, I_ext);
nx = size(x,1);
ny = size(x,2);

a(1:nx+1,1:ny+1) = 0;
b(1:nx+1,1:ny+1) = 0;
c(1:nx+1,1:ny+1) = 0;
Qflux(1:nx+1,1:ny+1,1:6) = 0;
%   Get the fluxes
for i = 2: size(x,1)
    for j = 2: size(x,2)

    ii = i-1;
    jj = j-1;

    %   Right face
    sfpx = y(ii+1,jj+1)-y(ii+1,jj);
    sfpy = -( x(ii+1,jj+1)-x(ii+1,jj) );

    %   Left face
    sfmx = -( y(ii,jj+1) - y(ii,jj) );
    sfmy = ( x(ii,jj+1)-x(ii,jj) );

    %   Top face
    sgpx = -( y(ii+1,jj+1) - y(ii,jj+1) );
    sgpy = x(ii+1,jj+1) - x(ii,jj+1);

    %   Bottom face
    sgmx = ( y(ii+1,jj)-y(ii,jj) );
    sgmy = -( x(ii+1,jj) - x(ii,jj) );

    %   Get the flux
    Qflux(i,j,:) = ( F(i+1+dd,j,:)*sfpx + G(i+1+dd,j,:)*sfpy + ...
        F(i+dd,j,:)*sfmx + G(i+dd,j,:)*sfmy + F(i,j+1+dd,:)*sgpx ...
        + G(i,j+1+dd,:)*sgpy + F(i,j+dd,:)*sgmx + G(i,j+dd,:)*sgmy );

    %   Normalize by Volume
    Qflux(i,j,:) = Qflux(i,j,:)./Vol(i,j);

    %   CFL terms
    a(i,j) = abs(u(i,j)*sfpx + v(i,j)*sfpy);
    b(i,j) = abs(u(i,j)*sgpx + v(i,j)*sgpy);
    c(i,j) = ss(i,j)*sqrt(abs( sfpx^2 + sfpy^2) ...
        + abs( sgpx^2 + sgpy^2) );
```

```matlab
        end
end

invdt = max(max((a+b+c)./Vol));
dt = cfl/invdt;
if dt < 10^(-12)
    dt = 0;
end
% dt

impulse_time = impulse_time + dt;



end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



%%%%%%%    boundary    %%%%%%%

function [Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v0, Vol, I_ext)
global gamma R nentrance nmid ninlet nnoz mi
[u,v,rho,p,e,Bx,By,T,ss,F,G] = flowvars(Q, E, Bb, EB, Max_stress,
Res_dif, visc_tens, kth, x, y, v0, Vol, I_ext);

%   Problem boundary conditions here
nx = size(x,1);
ny = size(x,2);

ix = ninlet;
ex = nentrance;
mx = nmid;
zx = nnoz;

mu0 = 4*pi*10^(-7);
e_charge = 1.602E-19;

iy = ny;
ey = ny;
my = ny;
zy = ny;

    %   Top Wall
    p(:,ny+1) = p(:,ny);

    v(:,ny+1) = 0;
    u(:,ny+1) = 0;
    rho(:,ny+1) = rho(:,ny);
```

```matlab
        Bx(:,iy+1) = 0; % MPD
        By(:,iy+1) = 0; % MPD
        Bx(ex:mx,iy+1) = Bx(ex:mx,iy); % MPD
        By(ex:mx,iy+1) = By(ex:mx,iy); % MPD


        %   Symmetry line
        p(:,1) = p(:,2);
        v(:,1) = -v(:,2);
        u(:,1) = u(:,2);
        rho(:,1) = rho(:,2);
        Bx(:,1) = Bx(:,2);     % MPD
        By(:,1) = By(:,2);     % MPD


        %   Inflow-shouldn't change from initialization
        u(1,:) = v0/CFD_weight;
        if I_ext > 0
            Bb0 = 0.5*mu0*12*I_ext/0.003;
            u(mx,:) = u(ex,:) + (e_charge*Bb0*0.005/mi);
        end
        v(1,:) = v(2,:);
%       Bx(1,:) = Bx(2,:);  % MPD
%       By(1,:) = By(2,:);  % MPD


        %   Out flow - set to upstream cells
        u(nx+1,:) = u(nx,:);
        v(nx+1,:) = v(nx,:);
        p(nx+1,:) = p(nx,:);
        rho(nx+1,:) = rho(nx,:);
        Bx(nx+1,:) = Bx(nx,:);  % MPD
        By(nx+1,:) = By(nx,:);  % MPD


        %   EOS
        mu0 = 4*pi*10^(-7);
        e =  abs(p)/(gamma-1)  +  (1/2)*rho.*(abs(u.*u)  +  abs(v.*v))  +
0.5*(abs(Bx.*Bx) + abs(By.*By))/mu0;

        Q(:,:,1) = rho;
        Q(:,:,2) = rho.*u;
        Q(:,:,3) = rho.*v;
        Q(:,:,4) = e;
        Q(:,:,5) = Bx;  % MPD
        Q(:,:,6) = By;  % MPD
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


SCRAMJET.m (COMPONENTS)


function [Speeds] = scramjet(T_c, T_amb, P_c, P_amb, Cp, W, mu_f0,
h_exh, h_th, h_tip, V_dc, I_dc, I_ext, freq, is_mpd, is_es, Gas_Name,
He_Content, E_i)
```

```
%   Modified Navier-Stokes CFD solver using Magnetoplasmadynamic
%   equations for electrothermal thruster


%   Author Modified the Program:
%   C. A. Barry Stoute
%   Ph.D. Candidate
%   York University
%   Department of Earth and Space Science and Engineering


%   Original Author of Program:
%   Britton Jeffrey Olson
%   Ph.D. Candidate
%   Stanford Univeristy
%   Department of Aero/Astro


%%%%%%%    Introduction and Background from Original Author    %%%%%%%
%   This program gives the ideal nozzle geometry using the method of
%   characteristics for a Quasi-2D Diverging Nozzle.  Assume gas is
%   exhausting from a combustion chamber that has no mass flow rate in.
%   Using 2D nozzle flow relations, an optimal throat area is found that
%   will produce the max amount of thrust for the given ambient pressure
%    and combustion chamber parameters.  This Area is automatically set
and
%   fed into the method of charactertistics portion of that code.  The
%   method of characteristics also uses the exit Mach number that
%   corresponds to the ideal exit area.


%%%%%%%    Introduction and Background from Modified Author    %%%%%%%


%    Note, DO NOT call the returning Magnetic Field vector B, it will
crash
%   the simulation. Suggested name: Bmag


%%%%%%%    Directions for running the program      %%%%%%%
%
%   Put in the desire values in the following order


%   T_c = Temperature in the combustion chamber (K)
%   T_amb = Ambient temperature (K)
%   P_c = Pressure in the combustion chamber (Pa)
%   P_amb = Ambient pressure (Pa)
%   Cp = Specific Heat (Pressure)
%   W = Molecular Mass
%   width = The width of the nozzle
%   h_th = The diameter of the throat
%   h_exh = The diameter of the exhaust
%   h_tip = The diameter of the tip (Cathode)
%   V_dc = The nominal input of the DC voltage to the circuit
%   I_dc = The nominal output of the DC current from the circuit
%   I_ext = External current supplying to the solenoid
%   freq = The desire frequency
```

```
%   Figure(1): Static thrust as function of exit area
%    Figure(2): Nozzle design and plots of Mach number & Pressure vs
Length
%   Figure(3): CFD simulation of designed nozzle (Mach Number)
%   Figure(4): CFD simulation of designed nozzle (Magnetic Field)

% Note, 1st row is the x direction (axial), 2nd row is the y direction
% radial, 3rd row is the z direction

% From nozzle.m

% clear all;
% clc;

% T_c = Temperature in the combustion chamber (K)
% P_c = Pressure in the combustion chamber (Pa)
% P_amb = Ambient pressure (Pa)
% T_amb = Ambient temperature (K)
% Cp = Specific Heat of Argon (Pressure)

global gamma x y cfl Vol T_c nin ns na ng nout R maxs mins boxex P_amb
omega k Bb0 v1 sigma_dc impulse_time mi rt nentrance nmid
%% Thruster's Physical Properties
rt = 0.5*h_th;
re = 0.5*h_exh;
width = h_th;

inlet_length = 5/1000; %25mm
inlet_radius = 7.5/1000;
entrance_height = inlet_radius-rt;
entrance_length = 15/1000; %15mm
mid_length = 10/1000; % 3mm


Lchar = 4*pi*0.005^2/(2*pi*0.005);


ra = inlet_radius;
rc = 0.5*h_tip;
rg = ra-rc;
cmpd = 2*pi*(ra+rc)/2;

%%   Problem parameters
Torr = 133;
P_c = P_c*Torr;
P_amb = P_amb*Torr;

gamma = Cp/(Cp-8.314);    % Ratio of Specific Heats Cp/Cv (Gamma)
gammak = gamma;

impulse_time = 0;
```

265

```matlab
mu0 = 4*pi*10^(-7);
e0 = 8.85E-12;

dh = h_th/100;
max_iter = 10000;
R = 8314/W;
omega = freq*2*pi*10^6;

kB = 1.38*10^(-23);
epsilon0 = 8.85*10^(-12);
me = 9.109*10^(-31);
mp = 1.672600000*10^(-27);
mi = mp*W;
h_planck = 6.626200000*10^(-34);
e_charge = 1.602*10^(-19);

% Meshing
conical = 1;
boxex = 1;
d_element = 1;

%%   CFD Portion parameters
cfl = 0.065;     %  Courant-Friedricks-Lewy stability factor (<1)
(Default: 0.8)
tt = 400/cfl;    % Number of time steps to take (Default: 500)
init = 1;    %  Initialize or use previous run's data? [1-init 0-
previous]


%%   Electromagnetic Propulsion

Fmaeckers = I_dc^(2)*mu0*log(ra/rc+0.75)*(4*pi)^(-1);
m_dot = 5*10^(-6);

alpha_ionization                                        =
(mi/(e_charge*E_i))*(4/(9*sqrt(3)))*((cmpd/rg)*(mu0*I_dc^2)/m_dot)^2;

if alpha_ionization > 1
    vel_mpd = sqrt(2*E_i*e_charge/mi);
else
    vel_mpd = alpha_ionization*sqrt(2*E_i*e_charge/mi);
end

alpha_noz = 6;

noz_length        =        (rt*(re/rt-1)+1.5*rt*(sec(alpha_noz*pi/180)-
1))/tan(alpha_noz*pi/180);

%%   Magnetic Acceleration

E = [0; 0];
```

```matlab
E0 = sqrt(E(1)^2+E(2)^2);
Bb = [mu0*12*I_ext/0.003; 0];
Bb0 = sqrt(Bb(1)^2+Bb(2)^2);
EB = 0.5*(e0*(E(1)^2+E(2)^2)+mu0^(-1)*(Bb(1)^2+Bb(2)^2));
Max_stress    =    (1/(4*pi))*[e0*E(1)*E(1)+mu0^(-1)*Bb(1)*Bb(1)-EB
e0*E(1)*E(2)+mu0^(-1)*Bb(1)*Bb(2);   e0*E(2)*E(1)+mu0^(-1)*Bb(2)*Bb(1)-EB
e0*E(2)*E(2)+mu0^(-1)*Bb(2)*Bb(2)-EB];
Res_dif = [E(1) 0; 0 E(2)];
visc_tens = [mu_f0 0; 0 mu_f0];

%% Min and Max Limits of the Colormap
v0 = vel_mpd;
maxspeed = 40; %12 for Reheat, 40 for different mixtures
minspeed = 0; % 6 for Reheat, 0 for different mixtures


%%


Te = 1.7;
T_e = Te*e_charge/kB;
n0_N = (1-He_Content)*P_c/(kB*T_amb);
n0_He = He_Content*P_c/(kB*T_amb);
n0 = n0_N+n0_He;


SahaN = 2.410E21 * T_e^(3/2)*exp(-e_charge*15.8/(kB*T_e));
SahaHe = 2.410E21 * T_e^(3/2)*exp(-e_charge*24.59/(kB*T_e));


ni_N = n0_N/(1+sqrt(1+n0_N/SahaN));
nn_N = ni_N*ni_N/SahaN;


ni_He = n0_He/(1+sqrt(1+n0_He/SahaHe));
nn_He = ni_He*ni_He/SahaHe;


ne = ni_N + ni_He;
ni = ne;
nn = nn_He+nn_N;


ce = sqrt(8*kB*T_e/(pi*me));
lambda_d = sqrt(Te*epsilon0/(e_charge*ne));
big_lambda = (4*pi*ne*lambda_d^3);


nu_ei = ni*e_charge^4*log(big_lambda)/(2*pi*epsilon0^2*me^2*ce^3);
nu_en = nn*ce*2*10^(-19);
nu_e = nu_ei+nu_en;


sigma_dc = ne*e_charge^2/(me*nu_e);


k = 3.2*e_charge*kB*T_e*n0/(nu_e*me);
kth = [k 0; 0 k];
%%
```

```
%                                   Pabs                                  =
0.5*(100E6*2*pi*rt/(2*sqrt(2))*Bb0)^2*sigma_dc*(nu_e^2)/((100E6*2*pi)^2+
nu_e^2);
% Maggy = Bb0^2*rt^2*pi*0.003/(2*mu0);


%%
nozzle

noz_mesh_inlet
noz_cfd_inlet
%
% noz_mesh_exhaust
% noz_cfd_exhaust


NOZ_MESH_INLET.m

%   ****** See nozzle.m for instructions ******
%  Program to extrapolate the data points from nozzle design and make a
%  uniform grid spacing in the x-direction
%  Change nothing... simply run this script

%% BARRY'S ADDON: Adjusting Nozzle to incorporate inlet

cnoz = noz;    % Renaming the original nozzle

clear noz

noz(1,1) = 0;
noz(1,2) = inlet_radius;
noz(2,1) = inlet_length;
noz(2,2) = noz(1,2);
for i = 3:num+1
    noz(i,1) = noz(2,1) + i*(entrance_length)/(num+1);
    noz(i,2) = ra + (rt-ra)*(i)/(num+1);
end

inlet_edge = size(noz,1);

noz(num+2,1) = noz(num+1,1)+mid_length;
noz(num+2,2) = noz(num+1,2);

noz_edge = size(noz,1);

heater = noz(inlet_edge:noz_edge,:)*1000;

nacelle = noz*1000;

%%
```

```matlab
% Find the minimum spacing given by the method of characteristics
% and set as the dx value
dx = 0;

for i=1: size(noz,1)-1
    len = noz(i+1,1) - noz(i,1);
    if (len < dx || i == 1)
        if len > 0
            dx = len;
        end
    end
end

% Explicitly give the dx value here
dx = max(noz(:,1))/ceil(max(noz(:,1))/dx);
dx = dx/d_element;

nshift = 2;

nentrance = ceil(max(noz(1:inlet_edge,1))/dx) - nshift;
nmid = ceil(max(noz(1:noz_edge,1))/dx) - nshift;
n = ceil(max(noz(:,1))/dx);

% len = max(noz(:,1));
% n = 50;   %  Note # of points is actually n+1
% dx = len/n

% Pick m points in y as some factor of x points
yfactor = 0.8;
m = ceil(yfactor*n);

% Make uniform x-distribution of points
xmax = 0;
i = 1;
while xmax < max(noz(:,1))
    xmax = dx*(i-1);
    x(i,1:m) = xmax;

    if (xmax - noz(noz_edge,1) <= dx)
        noz_edge_x = i;
    end

    i = i+1;

end

% Make the y-points and extrapolate linearly from closest points to fit
% the nozzle geometry

% Initialize and assign last value
```

```matlab
y(1:size(x,1),1:size(x,2)) = 0; % y = zeros(size(x,1),size(x,2))
y(1,size(y,2)) = noz(1,size(noz,2)); % First row - Last Column of y is
equal to the first row - last column on noz (i.e. the y direction)
y(size(y,1),size(y,2)) = noz(size(noz,1),size(noz,2)); % Last row - Last
Column of y is equal to the last row - last column of noz

% Barry's Modification: Changed the y meshing such that it takes care of
% the nozzle only.

for i = 1 : size(x,1)-1  % iteration of i from 1 to the size of the rows
of x minus 1 % Changed from minus 1 to minus 2

    j = 1;
    while x(i,1) >= noz(j,1)    % finding the smallest box of x
        x1 = noz(j,1);
        x2 = noz(j+1,1);
        y1 = noz(j,2);
        y2 = noz(j+1,2);

        j = j + 1;
    end

    slope = (y2 - y1)/(x2 - x1);  % finding the slope
    y(i,size(y,2)) = y1 + slope*(x(i,1)-x1);  % ith row - last column is
the y = mx + b such that b is y1 and x is the ith row of 1st column

    %Fill in mesh
    dy = y(i,size(y,2))/(size(y,2)-1);     % dy is derived from the ith
row and last column divide by the number of column in y - 1
    dy = dy/d_element;
    for k = 1: size(y,2)
        y(i,k) = dy*(k-1);  % this progress of dy over the columns
    end

end

% Comment left off

% Fill in mesh
dy = y(size(y,1),size(y,2))/(size(y,2)-1);  % filling bottom mesh
for k = 1: size(y,2)
    y(size(y,1),k) = dy*(k-1);
end
```

NOZ_CFD_INLET.m

```matlab
%   ****** See nozzle.m for instructions ******
%   With the given grid from the nozzle code solve the flow
%   using MacCormack's finite volume method
```

```matlab
%   Initialize the domain here
n = size(x,1);
m = size(x,2);


p(1:n+1,1:m+1) = P_c*(1+(gamma-1)/2)^(-gamma/(gamma-1));
T(1:n+1,1:m+1) = T_c*(1+(gamma-1)/2);


rho = p./(R.*T);


v1 = v0;


u(1:n+1,1:m+1) = v1;
v(1:n+1,1:m+1) = 0;


Bx(1:n+1,1:m+1) = 0;     % MPD
By(1:n+1,1:m+1) = 0;     % MPD


e  =  p/(gammak-1)  +  (1/2)*rho.*(u.*u  +  v.*v)  +  (1/2)*(Bx.*Bx  +
By.*By)/(mu0);



%   Cast into conservation form
Q(:,:,1) = rho;      %  Conservation of Mass
Q(:,:,2) = rho.*u;   %  Conservation of X-Momentum
Q(:,:,3) = rho.*v;   %  Conservation of Y-Momentum
Q(:,:,4) = e;        %  Conservation of Energy
Q(:,:,5) = Bx;       %  Faraday's Law in X-Direction    % MPD
Q(:,:,6) = By;       %  Faraday's Law in Y-Direction    % MPD

%   Get and store the volumes and surface flux terms
Vol(1:size(x,1)+1,1:size(x,2)+1) = 1;
for i=1 : n-1
    for j=1 : m-1
        side1 = ( x(i,j)-x(i+1,j) )*y(i+1,j+1) + ( x(i+1,j)-x(i+1,j+1)
)*y(i,j)...
            + ( x(i+1,j+1)-x(i,j) )*y(i+1,j);
        side2 = ( x(i,j)-x(i+1,j+1) )*y(i,j+1) + ( x(i+1,j+1)-x(i,j+1)
)*y(i,j)...
            + ( x(i,j+1)-x(i,j) )*y(i+1,j+1);
        Vol(i+1,j+1) = (1/2)*( abs(side1) + abs(side2) );
    end
end



%   Main iteration loop for integration in time
for k = 1: tt
    Q = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v1, Vol, I_ext);  % Call the solver to advance one time step
%     rho = Q(:,:,1);
%     u = CFD_weight.*Q(:,:,2)./rho;
```

```matlab
%      v = CFD_weight.*Q(:,:,3)./rho;
%
%      Speeds = sqrt(u.^2 + v.^2);
%
%      figure(5);clf;
%      set(gcf,'renderer','ZBuffer');
%                   surf(x.*1000,y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%        hold  on;surf(x.*1000,-y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%      colormap(jet);colorbar;
%             xlabel('(S)  Axial  -  Millimeters');ylabel('(S)  Radial  -
Millimeters');
%      title(k);
end

%   Get the flow variables back from conserved variables

rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);   % MPD
By = Q(:,:,6);   % MPD

p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));
Mach = sqrt(u.^2 + v.^2)./ss;

Bmag = sqrt(Bx.^2 + By.^2);
B_dir = wrapToPi(atan(By./Bx));
Speeds = sqrt(u.^2 + v.^2);
S_dir = wrapToPi(atan(v./u));

%%
median_speed = median(Speeds,1);
median_density = median(rho,1);
median_mag = median(Bmag,1);

for j = 1:size(u,2)
    for i = 1:size(u,1)
        if isnan(S_dir(i,j))
            S_dir(i,j) = 0;
            B_dir(i,j) = 0;
        end
        if Speeds(i,j) > median_speed(j)*3
            Speeds(i,j) = median_speed(j);
        end
        if rho(i,j) < 0
            rho(i,j) = 0;
            Speeds(i,j) = 0;
```

```matlab
        elseif rho(i,j) > median_density(j)*3
            rho(i,j) = median_density(j);
        else
        end
        if Bmag(i,j) > median_mag(j)*2
            Bmag(i,j) = median_mag(j);
        elseif isnan(Bmag(i,j))
            Bmag(i,j) = 0;
        end
        u(i,j) = Speeds(i,j)*cos(S_dir(i,j));
        v(i,j) = Speeds(i,j)*sin(S_dir(i,j));
        Bx(i,j) = Bmag(i,j)*cos(B_dir(i,j));
        By(i,j) = Bmag(i,j)*sin(B_dir(i,j));
    end
end

vthroat_avg = mean(Speeds(end,:));
vhtroat = Speeds(end,:);
vthroat_ratio = vthroat_avg/v0
pcompression = mean(p(1,:))/mean(p(end,:))

%%   Plot the Velocity vector field
figure(12);clf;

maxs = maxspeed;
mins = minspeed;

colormap(jet);

quiverc(x.*1000,y.*1000,u(1:size(p,1)-1,1:size(p,2)-
1)./1000,v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0 && I_ext < 1
%     plot(heater(:,1),heater(:,2),'Color','b','LineWidth',3);
% elseif I_ext > 1 && I_ext < 10
%     plot(heater(:,1),heater(:,2),'Color','g','LineWidth',3);
% elseif I_ext > 10 && I_ext < 100
%     plot(heater(:,1),heater(:,2),'Color','r','LineWidth',3);
% end
quiverc(x.*1000,-y.*1000,u(1:size(p,1)-1,1:size(p,2)-1)./1000,-
v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
% if I_ext > 0 && I_ext < 1
%     plot(heater(:,1),-heater(:,2),'Color','b','LineWidth',3);
% elseif I_ext > 1 && I_ext < 10
%     plot(heater(:,1),-heater(:,2),'Color','g','LineWidth',3);
% elseif I_ext > 10 && I_ext < 100
%     plot(heater(:,1),-heater(:,2),'Color','r','LineWidth',3);
% end
axis([0 noz(end,1)*1050 -re*1100 re*1100]);
```

```matlab
h = colorbar;
set(gca,'CLim', [mins maxs]);


set(gcf, 'Position', [1 1080 1920 1080]);
set(gcf,'PaperPositionMode','auto');

ylabela = get(h,'YTickLabel');
meterspersecond = repmat(' km/s',length(ylabela),1);
ylabela = [ylabela meterspersecond];
set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 36);
set(h,'YTickLabel',ylabela, 'FontSize', 36);

xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
scrsz = get(0,'ScreenSize');
print('-f12','-djpeg',['Inlet Velocity Vector Field of ' Gas_Name]);

NOZ_MESH_EXHAUST.m

%   ****** See nozzle.m for instructions ******
%  Program to extrapolate the data points from nozzle design and make a
%  uniform grid spacing in the x-direction
%  Change nothing... simply run this script


%  Find the minimum spacing given by the method of characteristics
%  and set as the dx value

left_edge = 0;
top_edge = 0;
right_edge = 0;

clear noz
clear nacelle
clear x
clear y

if conical == 1

    noz(1,1) = 0;
    noz(1,2) = rt;

    mid_edge = 1;

    for i = 2:num+1
        noz(i,1) = noz(1,1) + i*noz_length/(num+1);
        noz(i,2) = rt + (re-rt)*(i)/(num+1);
    end
else
```

274

```matlab
    end

nacelle = noz.*1000;
noz_edge = size(noz,1);

if boxex == 1;

    exbox(1,1) = noz(end,1);
    exbox(1,2) = noz(end,2);

    left_edge = size(exbox,1);

    for i = left_edge:left_edge+num+1
        exbox(i,1)    =    exbox(left_edge,1)    +    1.25*h_exh*(i-
left_edge)/(num+1);
        exbox(i,2) = exbox(left_edge,2);
    end

    top_edge = size(exbox,1);
    noz = vertcat(noz,exbox);

else
end

dx = 0;
for i=1: size(noz,1)-1
    len = noz(i+1,1) - noz(i,1);
    if (len < dx || i == 1)
        if len > 0
            dx = len;
        end
    end
end

%  Explicitly give the dx value here
dx = max(noz(:,1))/ceil(max(noz(:,1))/dx);
dx = dx/d_element;

nshift = 2;

nmid = 1;
nnoz = ceil(max(noz(1:noz_edge,1))/dx);
n = ceil(max(noz(:,1))/dx);

% len = max(noz(:,1));
% n = 50;    %  Note # of points is actually n+1
% dx = len/n

%  Pick m points in y as some factor of x points
yfactor = 0.8;
m = ceil(yfactor*n);
```

275

```matlab
%  Make uniform x-distribution of points
xmax = 0;
i = 1;
while xmax < max(noz(:,1))
    xmax = dx*(i-1);
    x(i,1:m) = xmax;

    if (xmax - noz(noz_edge,1) <= dx)
        noz_edge_x = i;
    end

    i = i+1;

end

%  Make the y-points and extrapolate linearly from closest points to fit
%  the nozzle geometry

%  Initialize and assign last value
y(1:size(x,1),1:size(x,2)) = 0; % y = zeros(size(x,1),size(x,2))
y(1,size(y,2)) = noz(1,size(noz,2)); % First row - Last Column of y is
equal to the first row - last column on noz (i.e. the y direction)
y(size(y,1),size(y,2)) = noz(size(noz,1),size(noz,2)); % Last row - Last
Column of y is equal to the last row - last column of noz

% Barry's Modification: Changed the y meshing such that it takes care of
% the nozzle only.

for i = 1 : size(x,1)-1  % iteration of i from 1 to the size of the rows
of x minus 1 % Changed from minus 1 to minus 2

    j = 1;
    while x(i,1) >= noz(j,1)    % finding the smallest box of x
        x1 = noz(j,1);
        x2 = noz(j+1,1);
        y1 = noz(j,2);
        y2 = noz(j+1,2);

        j = j + 1;
    end

    slope = (y2 - y1)/(x2 - x1);  % finding the slope
    y(i,size(y,2)) = y1 + slope*(x(i,1)-x1);  % ith row - last column is
the y = mx + b such that b is y1 and x is the ith row of 1st column

    %Fill in mesh
    dy = y(i,size(y,2))/(size(y,2)-1);    % dy is derived from the ith
row and last column divide by the number of column in y - 1
    dy = dy/d_element;
    for k = 1: size(y,2)
```

```matlab
        y(i,k) = dy*(k-1);   % this progress of dy over the columns
    end

end

% Comment left off

% Fill in mesh
dy = y(size(y,1),size(y,2))/(size(y,2)-1);   % filling bottom mesh
for k = 1: size(y,2)
    y(size(y,1),k) = dy*(k-1);
end
```

NOZ_CFD_EXHAUST.m

```matlab
%   ****** See nozzle.m for instructions ******
%   With the given grid from the nozzle code solve the flow
%   using MacCormack's finite volume method



%   Initialize the domain here
n = size(x,1);
m = size(x,2);


p(1:n+1,1:m+1) = P_c*(1+(gamma-1)/2)^(-gamma/(gamma-1));
T(1:n+1,1:m+1) = T_c*(1+(gamma-1)/2);

rho = p./(R.*T);

v1 = vthroat_avg;

u(1:n+1,1:m+1) = v1;
v(1:n+1,1:m+1) = 0;

Bx(1:n+1,1:m+1) = 0;     % MPD
By(1:n+1,1:m+1) = 0;     % MPD
Bx(nentrance-1:nmid+1,1:m+1) = Bb(1);%/CFD_weight;     % MPD
By(nentrance-1:nmid+1,1:m+1) = Bb(2);%/CFD_weight;     %MPD

e   =   p/(gammak-1)   +   (1/2)*rho.*(u.*u   +   v.*v)   +   (1/2)*(Bx.*Bx   +
By.*By)/(mu0);

clear Q
clear Vol

%   Cast into conservation form
Q(:,:,1) = rho;      %  Conservation of Mass
Q(:,:,2) = rho.*u;   %  Conservation of X-Momentum
```

```matlab
Q(:,:,3) = rho.*v;   %  Conservation of Y-Momentum
Q(:,:,4) = e;        %  Conservation of Energy
Q(:,:,5) = Bx;       %  Faraday's Law in X-Direction    % MPD
Q(:,:,6) = By;       %  Faraday's Law in Y-Direction    % MPD

%   Get and store the volumes and surface flux terms
Vol(1:size(x,1)+1,1:size(x,2)+1) = 1;
for i=1 : n-1
    for j=1 : m-1
        side1 = ( x(i,j)-x(i+1,j) )*y(i+1,j+1) + ( x(i+1,j)-x(i+1,j+1)
)*y(i,j)...
                + ( x(i+1,j+1)-x(i,j) )*y(i+1,j);
        side2 = ( x(i,j)-x(i+1,j+1) )*y(i,j+1) + ( x(i+1,j+1)-x(i,j+1)
)*y(i,j)...
                + ( x(i,j+1)-x(i,j) )*y(i+1,j+1);
        Vol(i+1,j+1) = (1/2)*( abs(side1) + abs(side2) );
    end
end


%   Main iteration loop for integration in time
for k = 1: tt
    Q = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v1, Vol, I_ext);  %  Call the solver to advance one time step
%     rho = Q(:,:,1);
%     u = CFD_weight.*Q(:,:,2)./rho;
%     v = CFD_weight.*Q(:,:,3)./rho;
%
%     Speeds = sqrt(u.^2 + v.^2);
%
%     figure(5);clf;
%     set(gcf,'renderer','ZBuffer');
%               surf(x.*1000,y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%        hold  on;surf(x.*1000,-y.*1000,Speeds(1:size(p,1)-1,1:size(p,2)-
1));view(0,90);
%     colormap(jet);colorbar;
%           xlabel('(S)  Axial  -  Millimeters');ylabel('(S)  Radial  -
Millimeters');
%     title(k);
end


%   Get the flow variables back from conserved variables

rho = Q(:,:,1);
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);  % MPD
By = Q(:,:,6);  % MPD


p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
```

```
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));
Mach = sqrt(u.^2 + v.^2)./ss;

Bmag = sqrt(Bx.^2 + By.^2);
B_dir = wrapToPi(atan(By./Bx));
Speeds = sqrt(u.^2 + v.^2);
S_dir = wrapToPi(atan(v./u));

clear Q

%%
median_speed = median(Speeds,1);
median_density = median(rho,1);
median_mag = median(Bmag,1);

for j = 1:size(u,2)
    for i = 1:size(u,1)
        if isnan(S_dir(i,j))
            S_dir(i,j) = 0;
            B_dir(i,j) = 0;
        end
        if Speeds(i,j) > median_speed(j)*3
            Speeds(i,j) = median_speed(j);
        end
        if rho(i,j) < 0
            rho(i,j) = 0;
            Speeds(i,j) = 0;
        elseif rho(i,j) > median_density(j)*3
            rho(i,j) = median_density(j);
        else
        end
        if Bmag(i,j) > median_mag(j)*2
            Bmag(i,j) = median_mag(j);
        elseif isnan(Bmag(i,j))
            Bmag(i,j) = 0;
        end
        u(i,j) = Speeds(i,j)*cos(S_dir(i,j));
        v(i,j) = Speeds(i,j)*sin(S_dir(i,j));
        Bx(i,j) = Bmag(i,j)*cos(B_dir(i,j));
        By(i,j) = Bmag(i,j)*sin(B_dir(i,j));
    end
end
vex = mean(Speeds(noz_edge,:));
Ibit = vex*m_dot*impulse_time;
mbit = impulse_time*m_dot;
Isp = vex/9.81
mRe = mu0*Lchar*sigma_dc*vex;

Impulse = impulse_time*m_dot*Speeds;
Iu = rho.*u;
Iv = rho.*v;
```

```matlab
%%  Plot the Velocity vector field
figure(12);clf;

maxs = maxspeed;
mins = minspeed;

colormap(jet);

quiverc(x.*1000,y.*1000,u(1:size(p,1)-1,1:size(p,2)-
1)./1000,v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),nacelle(:,2),'Color','w','LineWidth',3);
quiverc(x.*1000,-y.*1000,u(1:size(p,1)-1,1:size(p,2)-1)./1000,-
v(1:size(p,1)-1,1:size(p,2)-1)./1000);%view(0,90);
hold on;
plot(nacelle(:,1),-nacelle(:,2),'Color','w','LineWidth',3);
axis([0 noz(end,1)*1050 -re*1100 re*1100]);

h = colorbar;
set(gca,'CLim', [mins maxs]);



set(gcf, 'Position', [1 1080 1920 1080]);
set(gcf,'PaperPositionMode','auto');


ylabela = get(h,'YTickLabel');
meterspersecond = repmat(' km/s',length(ylabela),1);
ylabela = [ylabela meterspersecond];
set(gca, 'Position', [0.0975 0.1335 0.7389 0.7915], 'FontSize', 36);
set(h,'YTickLabel',ylabela, 'FontSize', 36);

xlabel('Axial - Millimeters');ylabel('Radial - Millimeters');
scrsz = get(0,'ScreenSize');
print('-f12','-djpeg',['Exhaust Velocity Vector Field of ' Gas_Name]);


SOLVER.m (FOR SCRAMJET EXPONENTS ONLY)

%   ****** See nozzle.m for instructions ******
%   solver.m

%   Solver for the macormack method
function [Q] = solver(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth,
x, y, v1, Vol, I_ext)
global gamma R

[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, x, y, v1, Vol, I_ext); % MPD are Bx By
```

```matlab
%   Take one MacCormack step
[Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y, v1,
Vol, I_ext);


end



%%%%%%%    flowvars    %%%%%%%%
function  [u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,
Max_stress, Res_dif, visc_tens, kth, x, y, v1, Vol, I_ext) % MPD are Bx
By
global gamma R omega cfl k sigma_dc mi ninlet nentrance nmid nnoz
%   Calculate the actual flow variables at each time step using
%%  STATIC MAGNETIC FIELD
% rho = Q(:,:,1);
% u = Q(:,:,2)./rho;
% v = Q(:,:,3)./rho;
% e = Q(:,:,4);
% Bx = Q(:,:,5);  % MPD
% By = Q(:,:,6);  % MPD
%
% p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
% T = p./(R*rho);
% ss = sqrt(abs(gamma*R*T));
%
% mu0 = 4*pi*10^(-7);
% e0 = 8.85E-12;
%
% Max_stress_xx = mu0^(-1)*(Bb(1)*Bb(1) - 0.5*(Bb(1)^2+Bb(2)^2));
% Max_stress_xy = mu0^(-1)*Bb(1)*Bb(2);
% Max_stress_yx = mu0^(-1)*Bb(1)*Bb(1);
% Max_stress_yy = mu0^(-1)*(Bb(2)*Bb(2) - 0.5*(Bb(1)^2+Bb(2)^2));
%
% MPD_e_x = Max_stress_xx.*u + Max_stress_xy.*v;% + kth(1,1).*T;
% MPD_e_y = Max_stress_yx.*u + Max_stress_yy.*v;% + kth(2,2).*T;
%
% F(:,:,1) = rho.*u;
% F(:,:,2) = rho.*u.*u + p - Max_stress(1,1);
% F(:,:,3) = rho.*u.*v - Max_stress(1,2);
% F(:,:,4) = (e+p).*u-(MPD_e_x);
% F(:,:,5) = -E(2);     % MPD
% F(:,:,6) = (Bb(1).*u-Bb(1).*v);     % MPD
%
% G(:,:,1) = rho.*v;
% G(:,:,2) = rho.*u.*v - Max_stress(2,1);
% G(:,:,3) = rho.*v.*v + p - Max_stress(2,2);
% G(:,:,4) = (e+p).*v-(MPD_e_y);
% G(:,:,5) = (Bb(1).*v-Bb(2).*u);     % MPD
% G(:,:,6) = -E(1);     % MPD


%%  DYNAMIC MAGNETIC FIELD
rho = Q(:,:,1);
```

```matlab
u = Q(:,:,2)./rho;
v = Q(:,:,3)./rho;
e = Q(:,:,4);
Bx = Q(:,:,5);   % MPD
By = Q(:,:,6);   % MPD


p = (gamma-1)*(e-(1/2)*rho.*(u.*u+v.*v));
T = p./(R*rho);
ss = sqrt(abs(gamma*R*T));


mu0 = 4*pi*10^(-7);


Max_stress_xx = mu0^(-1)*(Bx.*Bx - 0.5*(Bx.^2+By.^2));
Max_stress_xy = mu0^(-1)*Bx.*By;
Max_stress_yx = mu0^(-1)*By.*Bx;
Max_stress_yy = mu0^(-1)*(By.*By - 0.5*(Bx.^2+By.^2));


MPD_e_x = Max_stress_xx.*u + Max_stress_xy.*v;
MPD_e_y = Max_stress_yx.*u + Max_stress_yy.*v;



% Continuity
F(:,:,1) = rho.*u;
G(:,:,1) = rho.*v;

% X Momentum
F(:,:,2) = rho.*u.*u + p - Max_stress_xx;
G(:,:,2) = rho.*u.*v - Max_stress_xy;

% Y Momentum
F(:,:,3) = rho.*u.*v - Max_stress_yx;
G(:,:,3) = rho.*v.*v + p - Max_stress_yy;

% Energy
F(:,:,4) = (e+p).*u - (MPD_e_x);
G(:,:,4) = (e+p).*v - (MPD_e_y);

% X Faraday
F(:,:,5) = -E(2);       % MPD
G(:,:,5) = Bx.*v-By.*u;      % MPD

% Y Faraday
F(:,:,6) = By.*u-Bx.*v;     % MPD
G(:,:,6) = -E(1);       % MPD

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%    mac    %%%%%%%
```

```matlab
function [Q] = mac(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x,
y, v1, Vol, I_ext)
global gamma R

Q0 = Q;

%   Forward flux
[Qflux,dt] = flux_mc(Q,-1, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v1, Vol, I_ext);
Qbar = Q - dt*Qflux;
Q = Qbar;

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v1, Vol, I_ext);

%   Backward flux
[Qflux,dt] = flux_mc(Q,0, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v1, Vol, I_ext);
Q = (1/2)*(Q0 + Qbar - dt*Qflux );

[Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens, kth, x, y,
v1, Vol, I_ext);

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%% Barry's Note: Finite Difference Method used in finding the Flux of Q
%%%%%%%    flux_mc    %%%%%%%
function [Qflux,dt] = flux_mc(Q, dd, E, Bb, EB, Max_stress, Res_dif,
visc_tens, kth, x, y, v1, Vol, I_ext)
global cfl impulse_time

[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, x, y, v1, Vol, I_ext);
nx = size(x,1);
ny = size(x,2);

a(1:nx+1,1:ny+1) = 0;
b(1:nx+1,1:ny+1) = 0;
c(1:nx+1,1:ny+1) = 0;
Qflux(1:nx+1,1:ny+1,1:6) = 0;
%   Get the fluxes
for i = 2: size(x,1)
    for j = 2: size(x,2)

    ii = i-1;
```

```matlab
        jj = j-1;

        %    Right face
        sfpx = y(ii+1,jj+1)-y(ii+1,jj);
        sfpy = -( x(ii+1,jj+1)-x(ii+1,jj) );

        %    Left face
        sfmx = -( y(ii,jj+1) - y(ii,jj) );
        sfmy = ( x(ii,jj+1)-x(ii,jj) );

        %    Top face
        sgpx = -( y(ii+1,jj+1) - y(ii,jj+1) );
        sgpy = x(ii+1,jj+1) - x(ii,jj+1);

        %    Bottom face
        sgmx = ( y(ii+1,jj)-y(ii,jj) );
        sgmy = -( x(ii+1,jj) - x(ii,jj) );

        %    Get the flux
        Qflux(i,j,:) = ( F(i+1+dd,j,:)*sfpx + G(i+1+dd,j,:)*sfpy + ...
            F(i+dd,j,:)*sfmx + G(i+dd,j,:)*sfmy + F(i,j+1+dd,:)*sgpx ...
            + G(i,j+1+dd,:)*sgpy + F(i,j+dd,:)*sgmx + G(i,j+dd,:)*sgmy );

        %    Normalize by Volume
        Qflux(i,j,:) = Qflux(i,j,:)./Vol(i,j);

        %    CFL terms
        a(i,j) = abs(u(i,j)*sfpx + v(i,j)*sfpy);
        b(i,j) = abs(u(i,j)*sgpx + v(i,j)*sgpy);
        c(i,j) = ss(i,j)*sqrt(abs( sfpx^2 + sfpy^2) ...
            + abs( sgpx^2 + sgpy^2) );

    end
end

invdt = max(max((a+b+c)./Vol));
dt = cfl/invdt;
if dt < 10^(-15) || dt > 1
    dt = 0;
end
% dt

impulse_time = impulse_time + dt;



end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%    boundary    %%%%%%%%

function [Q] = boundary(Q, E, Bb, EB, Max_stress, Res_dif, visc_tens,
kth, x, y, v1, Vol, I_ext)
global gamma R nentrance nmid mi
[u,v,rho,p,e,Bx,By,T,ss,F,G]  =  flowvars(Q,  E,  Bb,  EB,  Max_stress,
Res_dif, visc_tens, kth, x, y, v1, Vol, I_ext);

%   Problem boundary conditions here
nx = size(x,1);
ny = size(x,2);


ex = nentrance;
mx = nmid;


mu0 = 4*pi*10^(-7);
e_charge = 1.602E-19;


%   Top Wall
p(:,ny+1) = p(:,ny);
v(:,ny+1) = 0;
u(:,ny+1) = 0;
rho(:,ny+1) = rho(:,ny);
Bx(:,ny+1) =  Bx(:,ny); % MPD
By(:,ny+1) =  By(:,ny); % MPD


%   Symmetry line
p(:,1) = p(:,2);
v(:,1) = -v(:,2);
u(:,1) = u(:,2);
rho(:,1) = rho(:,2);
Bx(:,1) = Bx(:,2);     % MPD
By(:,1) = By(:,2);     % MPD


%   Inflow-shouldn't change from initialization
% u(1,:) = sqrt(gamma*R*T_c);
u(1,:) = v1;
if I_ext > 0
    Bb0 = 0.5*mu0*12*I_ext/0.003;
    u(mx,:) = u(1,:) + (e_charge*Bb0*0.005/mi);
end


v(1,:) = v(2,:);
Bx(1,:) = Bx(2,:);   % MPD
By(1,:) = By(2,:);   % MPD


%   Out flow - set to upstream cells
u(nx+1,:) = u(nx,:);
v(nx+1,:) = v(nx,:);
p(nx+1,:) = p(nx,:);
rho(nx+1,:) = rho(nx,:);
Bx(nx+1,:) = 0;   % MPD
```

285

```matlab
By(nx+1,:) = 0;  % MPD


%   EOS
mu0 = 4*pi*10^(-7);
e = p/(gamma-1) + (1/2)*rho.*(u.*u + v.*v) + 0.5*(Bx.*Bx + By.*By)/mu0;


Q(:,:,1) = rho;
Q(:,:,2) = rho.*u;
Q(:,:,3) = rho.*v;
Q(:,:,4) = e;
Q(:,:,5) = Bx;  % MPD
Q(:,:,6) = By;  % MPD
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


QUIVERC.m


function hh = quiverc(varargin)
% Modified version of Quiver to plots velocity vectors as arrows
% with components (u,v) at the points (x,y) using the current colormap


% Bertrand Dano 3-3-03
% Copyright 1984-2002 The MathWorks, Inc.


%QUIVERC Quiver color plot.
%    QUIVERC(X,Y,U,V) plots velocity vectors as arrows with components
(u,v)
%    at the points (x,y).  The matrices X,Y,U,V must all be the same size
%    and contain corresponding position and velocity components (X and Y
%       can  also  be  vectors  to  specify  a  uniform  grid).   QUIVER
automatically
%    scales the arrows to fit within the grid.
%
%    QUIVERC(U,V) plots velocity vectors at equally spaced points in
%    the x-y plane.
%
%    QUIVERC(U,V,S) or QUIVER(X,Y,U,V,S) automatically scales the
%    arrows to fit within the grid and then stretches them by S.  Use
%    S=0 to plot the arrows without the automatic scaling.
%
%    QUIVERC(...,LINESPEC) uses the plot linestyle specified for
%    the velocity vectors.  Any marker in LINESPEC is drawn at the base
%    instead of an arrow on the tip.  Use a marker of '.' to specify
%    no marker at all.  See PLOT for other possibilities.
%
%    QUIVERC(...,'filled') fills any markers specified.
%
%    H = QUIVERC(...) returns a vector of line handles.
%
%    Example:
%        [x,y] = meshgrid(-2:.2:2,-1:.15:1);
```

```matlab
%       z = x .* exp(-x.^2 - y.^2); [px,py] = gradient(z,.2,.15);
%       contour(x,y,z), hold on
%       quiverc(x,y,px,py), hold off, axis image
%
%    See also FEATHER, QUIVER3, PLOT.
%    Clay M. Thompson 3-3-94
%    Copyright 1984-2002 The MathWorks, Inc.
%    $Revision: 5.21 $  $Date: 2002/06/05 20:05:16 $
%-----------------------------------------------------------

global maxs mins



set(gca, 'color', 'blue');
% Arrow head parameters
alpha = 0.33; % Size of arrow head relative to the length of the vector
beta = 0.23;  % Width of the base of the arrow head relative to the
length
autoscale = 1; % Autoscale if ~= 0 then scale by this.
plotarrows = 1; % Plot arrows
sym = '';

filled = 0;
ls = '-';
ms = '';
col = '';
lw=1;

nin = nargin;
% Parse the string inputs
while isstr(varargin{nin}),
  vv = varargin{nin};
  if ~isempty(vv) & strcmp(lower(vv(1)),'f')
    filled = 1;
    nin = nin-1;
  else
    [l,c,m,msg] = colstyle(vv);
    if ~isempty(msg),
      error(sprintf('Unknown option "%s".',vv));
    end
    if ~isempty(l), ls = l; end
    if ~isempty(c), col = c; end
    if ~isempty(m), ms = m; plotarrows = 0; end
    if isequal(m,'.'), ms = ''; end % Don't plot '.'
    nin = nin-1;
  end
end


error(nargchk(2,5,nin));
```

```matlab
% Check numeric input arguments
if nin<4, % quiver(u,v) or quiver(u,v,s)
  [msg,x,y,u,v] = xyzchk(varargin{1:2});
else
  [msg,x,y,u,v] = xyzchk(varargin{1:4});
end
if ~isempty(msg), error(msg); end

if nin==3 | nin==5, % quiver(u,v,s) or quiver(x,y,u,v,s)
  autoscale = varargin{nin};
end

% Scalar expand u,v
if prod(size(u))==1, u = u(ones(size(x))); end
if prod(size(v))==1, v = v(ones(size(u))); end

if autoscale,
  % Base autoscale value on average spacing in the x and y
  % directions.  Estimate number of points in each direction as
  % either the size of the input arrays or the effective square
  % spacing if x and y are vectors.
  if  min(size(x))==1,  n=sqrt(prod(size(x)));  m=n;  else  [m,n]=size(x);
end
  delx = diff([min(x(:)) max(x(:))])/n;
  dely = diff([min(y(:)) max(y(:))])/m;
  len = sqrt((u.^2 + v.^2)/(delx.^2 + dely.^2));
  autoscale = autoscale*0.9 / max(len(:));
  u = u*autoscale; v = v*autoscale;
end

minrangespeed = mins*autoscale;

%---------------------------------------------
% Define colormap
vr=sqrt(u.^2+v.^2);
vrangle = wrapToPi(atan(v./u));

for i = 1:size(u,1)
    for j = 1:size(u,2)
        if isnan(vrangle(i,j))
            vrangle(i,j) = 0;
        end
        if vr(i,j) < minrangespeed
            vr(i,j) = minrangespeed;
            u(i,j) = minrangespeed*cos(vrangle(i,j));
            v(i,j) = minrangespeed*sin(vrangle(i,j));
        end

    end
end

% vrn=round(vr/max(vr(:))*64);
```

```matlab
maxrangespeed = maxs*autoscale;
vrn=round(vr/maxrangespeed*64);
for i = 1:size(vrn,1)
    for j = 1:size(vrn,2)
        if vrn(i,j) > 64
            vrn(i,j) = 64;
        end
    end
end
CC=colormap;
ax = newplot;
next = lower(get(ax,'NextPlot'));
hold_state = ishold;

%---------------------------------------------
% Make velocity vectors and plot them

x = x(:).';y = y(:).';
u = u(:).';v = v(:).';
vrn=vrn(:).';
uu = [x;x+u;repmat(NaN,size(u))];
vv = [y;y+v;repmat(NaN,size(u))];
vrn1= [vrn;repmat(NaN,size(u));repmat(NaN,size(u))];

uui=uu(:);  vvi=vv(:);  vrn1=vrn1(:); imax=size(uui);
hold on

 for i=  1:3:imax-1
    ii=int8(round(vrn1(i)));
    if ii==0; ii=1; end
    c1= CC(ii,1);    c2= CC(ii,2);    c3= CC(ii,3);
    plot(uui(i:i+1),vvi(i:i+1),'linewidth',lw,'color',[c1 c2 c3]);
end

%---------------------------------------------
% Make arrow heads and plot them
if plotarrows,

  hu = [x+u-alpha*(u+beta*(v+eps));x+u; ...
        x+u-alpha*(u-beta*(v+eps));repmat(NaN,size(u))];
  hv = [y+v-alpha*(v-beta*(u+eps));y+v; ...
        y+v-alpha*(v+beta*(u+eps));repmat(NaN,size(v))];
  vrn2= [vrn;vrn;vrn;vrn];

 uui=hu(:);  vvi=hv(:);  vrn2=vrn2(:); imax=size(uui);

 for i=  1:imax-1
    ii=int8(round(vrn2(i)));
    if ii==0; ii=1; end
    c1= CC(ii,1);    c2= CC(ii,2);    c3= CC(ii,3);
    plot(uui(i:i+1),vvi(i:i+1),'linewidth',lw,'color',[c1 c2 c3]);
 end
```

```matlab
else
  h2 = [];
end
%---------------------------------------------

if ~isempty(ms), % Plot marker on base
  hu = x; hv = y;
  hold on
  h3 = plot(hu(:),hv(:),[col ms]);
  if filled, set(h3,'markerfacecolor',get(h1,'color')); end
else
  h3 = [];
end

if ~hold_state, hold off, view(2); set(ax,'NextPlot',next); end

if nargout>0, hh = [h1;h2;h3]; end
set(gca, 'color', [0 0 0],'Xcolor','k','Ycolor','k');
set(gcf, 'color', [1 1 1]);
set(gcf, 'InvertHardCopy', 'off');
```