2008-08-21

Testing Direct Simulation Monte Carlo Methods Against the Fluid Equations in the Inductively Coupled Plasma Mass Spectrometer

William R. Somers
Brigham Young University - Provo

TESTING DIRECT SIMULATION MONTE CARLO METHODS AGAINST THE FLUID EQUATIONS IN THE INDUCTIVELY COUPLED PLASMA MASS SPECTROMETER

by

William Somers

A thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Master of Science

Department of Physics and Astronomy

Brigham Young University

December 2008

BRIGHAM YOUNG UNIVERSITY

GRADUATE COMMITTEE APPROVAL

of a thesis submitted by

William Somers

This thesis has been read by each member of the following graduate committee and by majority vote has been found to be satisfactory.

Date Ross L. Spencer, Chair

Date Bryan G. Peterson

Date Paul B. Farnsworth

BRIGHAM YOUNG UNIVERSITY

As chair of the candidate's graduate committee, I have read the thesis of William Somers in its final form and have found that (1) its format, citations, and bibliographical style are consistent and acceptable and fulfill university and department style requirements; (2) its illustrative materials including figures, tables, and charts are in place; and (3) the final manuscript is satisfactory to the graduate committee and is ready for submission to the university library.

Date

Ross L. Spencer
Chair, Graduate Committee

Accepted for the Department

J. Ward Moody, Graduate Coordinator
Department of Physics and Astronomy

Accepted for the College

Thomas W. Sederberg, Associate Dean
College of Physical and Mathematical Sciences

ABSTRACT

TESTING DIRECT SIMULATION MONTE CARLO METHODS IN THE INDUCTIVELY COUPLED PLASMA MASS SPECTROMETER

William Somers

Department of Physics and Astronomy

Master of Science

A Direct Simulation Monte Carlo fluid dynamics code named FENIX has been employed to study gas flow-through properties of the inductively coupled plasma mass spectrometer (ICP-MS). Simulation data have been tested against the Navier-Stokes and heat equations in order to see if FENIX functions properly. The Navier-Stokes and heat equations have been constructed from simulation data and are compared term by term. This comparison shows that FENIX is able to correctly reproduce fluid dynamics throughout the ICP-MS simulation, with an exception immediately behind the ICP-MS sampler cone, where the continuum criterion for the Navier-Stokes equation is not met. Testing the data produced by Fenix also shows that this DSMC method correctly produces momentum and thermal boundary layer phenomenon as well. FENIX output data produce statistical fluctuations of about 2%. Limitations occur from fitting data near surfaces, incurring a relative error of about 5%,

and fitting data to take second derivatives where fluid velocity gradients are steep, introducing a relative error of about 10%.

ACKNOWLEDGMENTS

First and foremost, I would like to thank Ross Spencer, my graduate advisor, who has made this project possible. His direction and advice were indispensable. I would also like to thank Adam Payne, Jaron Kroegel, and Andrew Sampson, whose collaborative efforts coding FENIX kept our research moving forward. I am grateful to my family for their support, especially my wife Kathleen, whose patience and encouragement kept me going.

Appreciation is also in order for the funding received for this project, provided by the U.S. Department of Energy, Grant Number DE-FG02-03ER15410.

Contents

List of Figures

Chapter 1

Introduction

1.1 The Inductively Coupled Plasma Mass Spectrometer (ICP-MS)

The Inductively Coupled Plasma Mass Spectrometer (hereafter to be referred to as the ICP-MS) has been in use since 1983, when it was first introduced as a powerful instrument used for trace elemental analysis. A machine of considerable sensitivity, the ICP-MS is robust enough to accomplish ultra-trace analysis, detecting on a level of picograms per gram. In addition to sensitivity, the characteristic that distinguishes the ICP-MS from most other methods of elemental analysis is the ability to look at isotopic compositions.

1.1.1 Practical Applications and Uses of the ICP-MS

The ICP-MS is used by pharmaceutical companies to analyze and verify elemental composition of complex medical products; it is used in cancer clinics to measure protein biomarkers indicative of breast cancer; it is used in urology to detect illegal drugs or other toxins. The ICP-MS ability to detect trace elements makes it a popular method for detecting impurities in semiconductor products; it is useful in determining the efficacy of catalytic converters in automobiles, analyzing alloys, paper, glass, ceramics, dyes, paint, petroleum based products, plastic, rare earth element compounds, lab gloves, steel, silica, and sulfides. The ICP-MS is one of only two methods allowable by the Federal Government for testing arsenic levels in potable water. The U.S. Government also uses the ICP-MS to handle radio-nuclide analysis of nuclear waste. Pollution studies, landfill gas emissions, and forensics are also important applications of the ICP-MS.

The ICP-MS is used in a variety of fields to further academic research as well. Archaeologists use the ICP-MS to analyze ancient ceramics, metals, and other artifacts. Biologists use the ICP-MS to study blood, bones, red cells, tissues, and so forth. Fertilizers, leaves, and poisonous plants are also studied. Geologists analyze fossils and meteors and environmentalists analyze waterways looking for trace pollutants and mineral content.

1.1.2 ICP-MS Schematics and Details

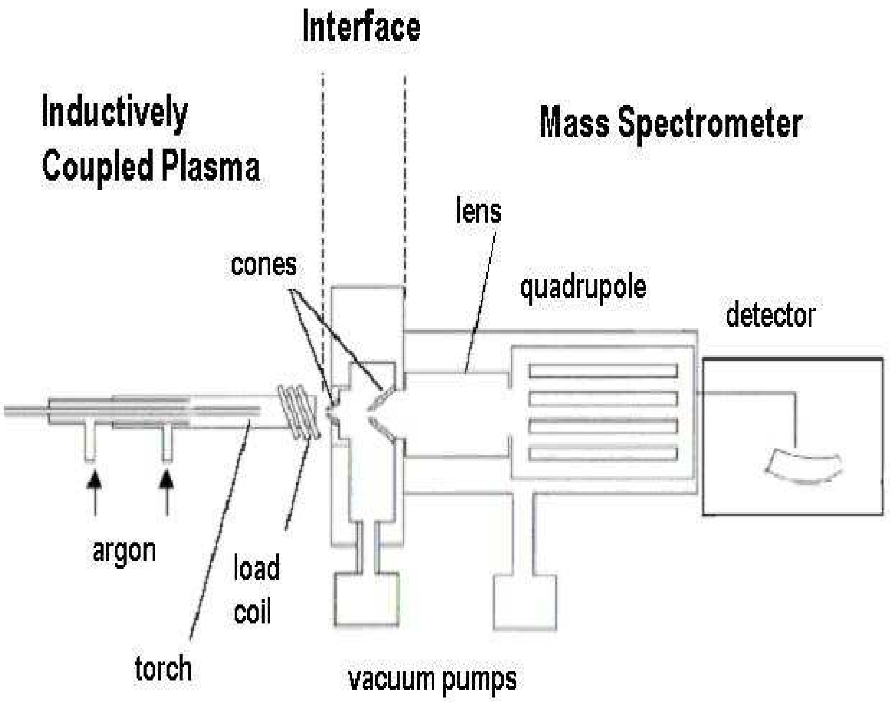

Figure 1.1 ICP-MS Schematic

Sample Introduction and Plasma Generation

A variety of methods may be used to introduce a sample into the ICP-MS. The most common is to convert the sample into an aerosol by means of a nebulizer, of which there are a variety of types. Other methods include laser ablation and electrothermal vaporization. The general function of each method is the same: convert the sample into small droplets or particles that can be channeled into the ICP-MS torch.

The analytes injected into the nebulizer may be all of one substance, or a mixture of substances, each with different concentrations. Included with the analyte(s) in question will be the sample matrix. These are other constituents which are injected into the ICP-MS along with the analyte. Matrix effects are being studied by Paul Farnsworth at BYU, [1] who asserts that small variations in the sample matrix may greatly alter the ionization properties of the plasma. For instance, including sodium with the analyte barium will alter the behavior of the barium [2].

The plasma is generated by means of a quartz tube and high power radio frequency coil, as shown in 1.1. A radio frequency generator is connected to an inductor coil which surrounds three concentric quartz tubes. The background gas (usually argon) is introduced into the outermost channel formed by the tubes. RF power, ranging from 800 to 1500 watts, is supplied to the quartz chamber, creating an alternating current corresponding to the generator frequency. Modern ICP-MS devices operate at 27 MHz. [3] The oscillating current produces a magnetic field within the coil, along with an associated electric field. To begin the formation of the plasma, as the gas passes through the field an initial high voltage spark strips some of the argon atoms of their electrons. These electrons are accelerated by the electric field, causing them to collide with other atoms, and more electrons are released. The argon atoms, ions and electrons leave the chamber heated, forming the plasma; common temperatures are between 4000K and 7000K. This heated gas is the inductively coupled plasma discharge.

The Vacuum Interface

The nebulized sample is introduced into the inner quartz tube, and is then carried into the plasma torch. (The intermediate quartz tube is pumped with argon, and this keeps the high temperature plasma from melting the other tubes.) Once in the plasma torch, analytes present in the sample break down into atoms and become ionized. These ions, along with the plasma discharge, impinge on the first of two conical nozzles with small circular apertures. The first nozzle, called the sampler cone, typically has an aperture diameter of 1 mm. Upstream from the sampler cone, the plasma discharge is open to atmospheric pressure. Differential pumping takes place on the downstream side of the sampler cone, and this region is pumped to a pressure of about 1 mbar. The pressure gradient across the nozzle allows the high density, high temperature plasma to expand into the downstream side of the sampler cone, cooling to lower temperature and reaching lower density.

Supersonic Expansion As the plasma passes through this nozzle, the pressure differential results in the acceleration, expansion and cooling of the gas. The expansion is supersonic, and forms a well known shock structure consisting of the barrel shock, the zone of silence, and the Mach disk. The zone of silence is a term that refers to the flow being supersonic, and is terminated by a stationary shock which allows the supersonic, low density gas to come up to the background pressure with slow flow.

The skimmer nozzle The expansion region in the ICP-MS is designed to produce what is called ”Ideal Skimming.” This is achieved when a second cone (the skimmer) is placed in the zone of silence. In the zone of silence, the streamlines of the fluid are nearly parallel, and collisions are hopefully negligible. If these conditions are met, the skimmer cone is able to extract an ion beam from the fluid, and this beam is analyzed by the mass spectrometer. The assumption that ideal skimming is achieved by placing the skimmer cone in the zone of silence has not been ratified; experimental studies have shown considerable evidence that there is a shock wave at the tip of the skimmer cone [4], and the BYU computational physics group has produced simulation data that verifies those findings. [5].

1.2 Previous Work

1.2.1 ICP-MS Studies

The ICP-MS has been studied since its creation. Physical structure and engineering are continually investigated, including nozzle sizes, nozzle apertures and the separation distance between the sampler cone and the skimmer cone. [6–8] Sample introduction and nebulizer properties have been studied, [9] and plasma properties in the nozzle have been investigated. [10, 11]. An effort to further document plasma behavior in the ICP-MS is currently being pursued by Paul Farnsworth at Brigham Young University. Laser-induced fluorescence has been used to achieve high resolution imaging of Barium ions near the sampling cone, giving a clearer picture of what the plasma is doing [2].

Ideal Nozzle Flow

There has been substantial progress made in understanding gas dynamics, especially as pertains to nozzle flow. Supersonic gas expansion through a nozzle has been well documented, and the ensuing shock structure is well known, and is available in the literature. [12]. These studies, though, assume negligible viscous and thermal conduction effects. Ideal models are used because calculating viscosity, thermal diffusion, and other gas properties are not straightforward when there are high temperatures present, or as in the ICP-MS, a small nozzle diameter is used. The effects of viscosity, thermal conduction, and thermal diffusion are significant in the ICP-MS, however, and can not be adequately represented with the ideal case. Such assumptions limit the view of complex gas behavior. There are several ways to include non-ideal effects; these methods are presented in Sec. 1.3.

1.3 Mathematical Models of Fluid Flow in the ICP-MS

There are three basic methods used to determine the characteristics in a given fluid flow: The Navier-Stokes fluid equations, the Boltzmann transport equation, and direct simulation computational methods. Each of these methods has the ability to provide information about the fluid flow in the ICP-MS, but each method has restrictions and limitations as well. Inspection of each method suggests that Direct Simulation Monte Carlo (DSMC) methods are best for studying the ICP-MS.

The Navier-Stokes (N-S) equation

The Navier-Stokes equation is Newton's second law applied to an element of fluid within the flow. The forces are pressure and viscous forces. For these equations to be valid, the physical properties of the fluid (velocity, temperature, etc) must not have gradients whose scale lengths are comparable to or smaller than the mean free path of a molecule. The ratio of the mean free path to the physical length scale is known as the Knudsen number (Kn). The Navier-Stokes equations are invalid when Kn is larger than about 0.1. The Knudsen number is calculated as follows:

$$\mathrm{Kn} = \frac{\lambda}{L} = \frac{1}{\sqrt{2}\, n\sigma\, L} \tag{1.1}$$

where

λ	=	particle mean free path
L	=	Physical scale length
σ	=	Collision cross sectional area
n	=	Number density

The Navier-Stokes equation holds in the region upstream of the sampler cone, in the sampler cone nozzle, and for a few millimeters downstream of the nozzle. About 5 millimeters past the sampler cone, however, the fluid density becomes low enough that collisions are rare and the Navier-Stokes equations fail. A calculation intending to model the full gas flow through the ICP-MS would have to discard the Navier-Stokes equation in the downstream region, and implement another method.

Kinetic Theory

The Boltzmann transport equation is a representation of the time evolution of the one-particle distribution function f in phase space. f tells how many particles in a given volume fall within a certain range of velocities and space - a phase-space density function. The Boltzmann transport equation can represent a collision-less gas or a set of interacting particles. It can include both relativistic and classical particles.

The Boltzmann transport equation contains all the information necessary to understand a fluid flow, and is valid when Kn is high, but its drawback is the difficulty of modeling it computationally. The collision term in the Boltzmann transport equation involves integrating over space and velocity, meaning a six-dimensional integral. Computationally, this requires that a fine grid be mapped over a 6-D simulation region, and a 6-D integral must be performed at every point on this grid. To describe the time-evolution of fluid properties, this integral must be performed every time step of the simulation . The computational cost of this method is excessive for high density gases or fluids. This limits the equation to very dilute gases with few particles, or gases in a very small volume.

Numerical and Computational Methods

The Navier-Stokes equation is invalid in portions of the ICP-MS, and the Boltzmann transport equation is too difficult to solve. A computational alternative to solving the fluid or transport equations is that of simulating particle collisions. Even when the Navier-Stokes equations break down or become insufficient particle collisions are still happening, and tracking them remains accurate and simple. By simulating particle collisions directly, a statistical solution to the Boltzmann transport equation may be given without having to solve the computationally expensive collision term in said equation.

Historically, the main limitation of simulation methods has been inadequate computational hardware, but with the availability of greater computer memory and faster processors, efficient simulations may be developed that directly simulate the interaction of gas particles. The most commonly used particle simulation method is known as Direct Simulation Monte Carlo (DSMC), and the data presented in this text was produced by this method.

1.3.1 DSMC Methods and Graeme Bird

Dr. Graeme Bird of The University of Sydney pioneered the use of Direct Simulation Monte Carlo (DSMC), a stochastic solution to the time evolution of interacting particles. DSMC is the most widely accepted computational method to model rarefied gas flow, and is used in many fields of research including chemistry, physics, aeronautics, and more. Bird's“Approach to Translational Equilibrium in a Rigid Sphere Gas” [13], published in 1963, was the first paper on the DSMC method. Bird's book *Molecular Gas Dynamics and the Direct Simulation of Gas Flows* [14] has been a central authority on the subject since it was published.

Bird has developed a DSMC Particle-In-Cell simulation routine that follows the principles of rarefied gas dynamics as outlined in his book. [14] Bird has done extensive work with his routine, building the foundation and basic algorithms of the DSMC method over a period of more than thirty years. He has made available DSV2, a 2-Dimensional DSMC program, as well as DSV3, its 3-Dimensional counterpart.

Simulations modeling all manner of situations have now been created using Particle-In-Cell DSMC methods, many of them based on Bird's programs: Space Shuttle plume impingement on the Space Station and Mir; aero-thermodynamic assessment of the Mars Global Surveyor performing an aero-braking maneuver at Mars; and the semiconductor industry in addressing problems that occur in rarefied environments (e.g., plasma etching and chemical vapor deposition) [15]. Bird's DSMC method is widely regarded as the acceptable way to model rarefied gases, and many groups use his programs to further their research efforts and solve difficult gas problems. We have used Bird's code, and found that it is not ideal for modeling the ICP-MS. The need to create complex geometry, implement boundary conditions, and have internal access to all of the functions of the code and the resultant data has led to the creation of FENIX, a DSMC code written under the direction of Ross Spencer at BYU to simulate the ICP-MS.

DSMC methods have been used to a limited degree to study specific aspects of the ICP-MS. Benson and Gimelshein used DSMC methods to study desolvation rates in the nebulizing chamber of the ICP-MS torch [9], and Macrossan and Lilley use DSMC methods to study argon shock structures [16]. FENIX is the first attempt to model the full flow of the ICP-MS.

1.4 Thesis Purpose

This paper will review data from three simulation regions of the ICP-MS, as obtained computationally using FENIX: upstream from the sampler cone, in the nozzle throat, and as far as 3 mm downstream from the cone. This project will demonstrate the level of capability of FENIX to correctly simulate this difficult gas dynamics problem. This paper will show that FENIX properly reproduces the Navier-Stokes and heat equations; thermal conduction, viscosity, and boundary layers will also be discussed. This calculation will give insight into nozzle flow that is difficult to characterize analytically.

Using data produced by FENIX, it will be shown that the integral of the inviscid Navier-Stokes equation, known as Bernoulli's equation, is satisfied over much of the nozzle. Bernoulli's equation fails in a thin layer about 0.14 mm wide next to the sampler cone in the nozzle. The failure is due to heat transfer from the gas to the cone across this thermal boundary layer. It will be further shown that the equations governing thermal and momentum transport are not followed closely immediately behind sampler cone, where the Knudsen number is high. FENIX will be shown to be accurate within 2% throughout much of the simulation, with error arising from statistical noise. Fitting data very close to surfaces adds an additional 5% error, and second derivative data fitting in areas where there are steep gradients adds a further 10% error. There is one region where all of the sources of error are present - right at the front corner of the sampler cone. In this area, error up to 17% is present.

Chapter 2

FENIX and the Data Analysis Method

2.1 The DSMC Method

2.2 Fenix

There are five essential processes to FENIX. They are: Building the simulation geometry, loading particles into the simulation, moving each particle each time step, colliding the particles, and sorting out all the information afterward to retrieve meaningful data.

Geometry Builder

Fenix is capable of simulating a variety of situations. According to the simulation geometry desired, the boundaries of the simulation are determined and the interior region is split into cells. There are actually two grids mapped onto the given geometry. There is a grid of collision cells, and this grid is used in FENIX to determine which particles will collide. Another grid system of data (or sampling) cells functions as the physical output grid. It is on this grid that fluid velocities, temperatures, densities, and so forth are cataloged. The collision cells are not uniform in size across the grid. Since the routine is stochastic, a minimum particle density is required per cell in order to produce statistically acceptable data. About 30 particles are required per cell in order for the collision routine to output data accurate within 2-3 percent. Nearest-neighbor collision routines require only seven particles per cell. [14] This minimum requirement demands that simulation regions with extremely low density have large collision cells. If a general idea of the density properties of a simulation to be run is not known a priori, then an iterative refinement method must be employed: the code is operated, and regions found to have excessively low density are assigned larger collision cell sizes the next time around. Regions with extremely high densities are given smaller cell sizes.

Another important function of the Geometry Builder section of FENIX is the "mask." While most cells in a simulation will represent empty space through which the particles flow, some may be located where surfaces are present. The ICP-MS sampler and skimmer cones, for instance, are part of the FENIX simulation. A fluid particle should not be permitted to travel through the metal of the nozzle, and so specific exclusion rules must be set for any grid cells near the nozzle. Furthermore, surfaces within the simulation may have particular temperature or chemical properties. Fig. 2.1 shows the mask configuration. Each set of colored cells is programmed with particular rules to reflect these surface properties. The blue cells along the left edge will share thermal energy with the impinging particles.

Reflecting particles off any surface is an involved process. A particle traveling toward a surface may be just outside it at the end of one time step, then well inside the surface as soon as the next time step is processed. The program looks for particles

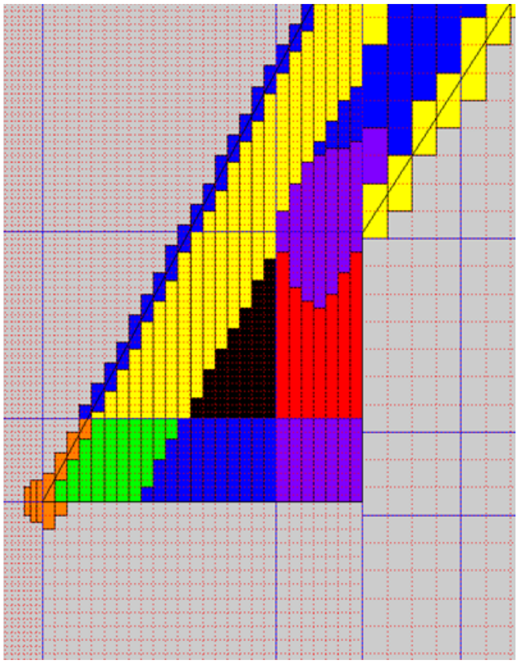

Figure 2.1 FENIX Mask

inside surfaces and reflects them back out according to their trajectories from the previous time step. This becomes difficult in areas where the geometry of a surface is smaller than the particles' mean free path, however. In this situation, some particles may travel all the way through the object and be in an "allowed" cell on the other side in just one time step. In order to prevent this, the mask extends beyond the actual surface wherever the objects are narrow - the tip of the sample cone is one such place. It can be seen in Fig.2.1 that the orange mask cells surround the tip of the sampler nozzle. These orange cells are programmed to look for particles whose trajectory will send them through the tip of the cone during the next time step. These particles are then reflected back away from the nozzle surface.

Particle Simulation Actions: Load, Move, and Sort

Load When the simulation begins its startup, simulation particles are loaded into the geometry. There are a number of ways that this may occur: one possibility is to have a complete vacuum at startup, and then read in particles along the back of the simulation region as time steps progress. This method is advantageous if one wants to analyze the shock waves that will be generated as the fluid particles rush through the vacuum and impinge on the sampler cone. If the desire is to analyze the simulation region at a steady state, it is computationally inefficient to use this method. An alternate method consists of loading the simulation region with a background gas with a profile similar to what you might expect to see after running the first method for a long time. As long as the boundary conditions have been properly specified, steady state is achieved with less computation. This method requires some pre-knowledge of the gas properties; a poor guess of the fluid's profile could increase computational time instead of saving it. The most efficient way to load the initial conditions into FENIX is by using data collected from a previous run. After an initial run using one of the previously mentioned startup methods has been done, continual use of FENIX is expedited by implementing a restart file from a previous run. The initial particle load consists of a Maxwellian distribution, according to equilibrium kinetic theory. The Maxwellian distribution is [17]:

$$f(v_x, v_y, v_z) = n\left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} \exp\left[\frac{-m(\,v_x^2 + v_y^2 + v_z^2)}{2kT}\right] \tag{2.1}$$

In this representation, n is the number density, k is the Boltzmann constant, m is the particle mass, v_x, v_y, and v_z are the components of the particle velocity, and T is the equilibrium temperature of the gas.

Move Once loaded into the simulation geometry, the particles are assumed to be ballistic. (Motion alteration due to electric fields has not yet been implemented in the code.) Simple ballistic motion is relatively easy to create; a time step is determined, and during each time step every particle is moved according to the following equations.

$$x_{(n+1)} = x_n + v_x\,\tau$$

$$y_{(n+1)} = y_n + v_y\,\tau$$

$$z_{(n+1)} = z_n + v_z\,\tau$$

where

x_n	$=$	x position at time step n
y_n	$=$	y position at time step n
z_n	$=$	z position at time step n
v_x	$=$	x-component of velocity
v_y	$=$	y-component of velocity
v_z	$=$	z-component of velocity
τ	$=$	time step

It is essential that the time step used in any DSMC code be tailored to the situation. A very small time step would produce more continuous data, but would require vast amounts of computation. A large time step increases computational ease, but may not be sensitive to changes within the fluid occurring over small regions. The fundamental requirement is that the time step must be less than the average collision time.

Collisions The stochastic nature of DSMC comes from the collision procedure, wherein the selection of particles that will collide each time step is generated randomly - hence the name Monte Carlo. In each cell, a designated number of candidate particle pairs are chosen for collision. The number of collision pairs selected from each cell is determined by kinetic theory. The particle cross section for argon is obtained by using the variable soft sphere model with the constants obtained from the experimental determination of argon viscosity by Macrossan and Lilley: [14, 16, 18, 19]:

$$\sigma = \frac{7.925 \times 10^{-18}}{v_r^{0.44}} \tag{2.2}$$

In the simulation routine, v_r is the relative velocity of a particle pair, and v_{rmax} is the upper bound of the set of relative velocities in each cell. A candidate pair is selected for collision according the the well-known acceptance-rejection algorithm:

$$\sigma\, v_r > R(\sigma\, v_r)_{max} \tag{2.3}$$

Here R is a uniformly distributed random number. Collisions are the manner through which the gas is able to communicate and propagate information, and therefore it is essential that the collision routine be as accurate as possible. One possible difficulty with this algorithm has already been mentioned in the geometry section: error occurs where there are not enough particles in a given cell to accurately employ a stochastic method. As mentioned, one solution to this trouble is to adapt the cell size to the local density function. Cells in low density areas are larger than those in high density region. A further refinement of the collision method is to require collisions to occur only between particles that are immediate neighbors in a cell. This nearest - neighbor selection criterion increases computational requirements, but also increases statistical accuracy and is currently employed in FENIX.

Sort The sorting process is the organization stage of a DSMC code. As each particle moves in a time step, they cross into new cells in the grid. Now each cell has a different number of particles in it than before, and therefore a different number of collision candidates. In order for the routine to continue, each particle must be sorted into its new cell. Furthermore, some particles will enter or leave the simulation region. Sorting allocates new particles into their respective cells, and deletes particles that have left the simulation. The sorting process also handles particles that have reflected off surfaces in the simulation.

Also essential to the sorting process is collecting data. As the code is based on statistics, information about the gas properties (velocity, temperature, density, etc...) is collected and averaged (in each cell) each time step. Data are also averaged over large blocks of time as the simulation approaches steady state. This data is written to files that will become the restart data files for future simulation runs.

2.3 The Navier-Stokes Equation

The Navier-Stokes equation describes the motion of viscous fluids and gases, and is valid wherever the Knudsen number is smaller than about 0.1. In the simulation, then, FENIX data should reproduce the Navier-Stokes equation wherever the Knudsen number is less than 0.1 Comparing the Navier-Stokes equations term by term using our simulation output data will test the ability of FENIX to model rarefied gas flow.

2.3.1 The Momentum Equations

The Navier-Stokes Equation is Newton's second law applied to a control element of fluid in a system and is expressed in vector form [20] as

$$\rho \frac{D\mathbf{u}}{D\,t} = -\nabla\mathbf{P} + \left[\nabla^{\mathbf{2}}\mu\ \mathbf{u} + \frac{\mathbf{1}}{\mathbf{3}}\nabla(\nabla\cdot\mu\ \mathbf{u})\right]\left(\frac{\partial\mathbf{u_i}}{\partial\mathbf{r_j}} + \frac{\partial\mathbf{u_j}}{\partial\mathbf{r_i}} - \frac{\mathbf{2}}{\mathbf{3}}\delta_{\mathbf{i}}\mathbf{j}\,\nabla\cdot(\mathbf{u})\right) \tag{2.4}$$

where

ρ	=	density
$\mathbf{u}$	=	velocity
$\mathbf{P}$	=	Pressure
μ	=	viscosity

The Navier-Stokes equation may be analyzed term by term; each term represents a particular physical trait of the fluid represented. The first term on the right side of Eq. 2.4 is the pressure part of the force due to collisions. The second (right-hand) term of Eq. 2.4 represents the viscous transfer of momentum due to collisions.

The Navier-Stokes equation is a non-linear partial differential equation, where the form of the solution is not known *a priori*. Often, the equation is simplified by assuming that the fluid is inviscid, has constant velocity, or is incompressible. It is simplified when analyzing a steady state fluid as well. In our case, however, only the steady state simplification applies.

Five or six mm downstream from the sampler cone in the ICP-MS, continuum fluid dynamics are not applicable, as the particle mean free path becomes very long and collisions become rare. In this regime, the Knudsen number (as previously discussed) is large, and the Navier-Stokes equation is invalid. However, the N-S equations may be employed in the nozzle itself, and upstream from it. Data from FENIX has been collected in these regions and analyzed to find if the code results correctly model the fluid equations.

2.3.2 Viscosity and Boundary Layers

As particles impinge on a surface, they impart momentum to the surface. This transfer of momentum is communicated via particle collisions, and extends into the body of the fluid for a finite distance. Beyond this point, the fluid body is unaware that the surface exists at all. This phenomenon is known as the momentum boundary layer, and its thickness, in thin-plane geometry, is characterized by the following: [21]

$$\delta = 5\sqrt{\frac{\mu\,L}{\rho\,v_s}} \tag{2.5}$$

In this expression, μ is the fluid viscosity, ρ is the density, v_s is the mainstream velocity, and L is the distance along the thin plate in the direction of the flow. In Eq. 2.5, the boundary layer is defined by the point where the velocity reaches 99% of its mainstream value, which accounts for the factor of 5 in the expression.

We shall investigate the computationally generated boundary layer thickness, and compare it to the expected theoretical results. Boundary layer thickness is significant, as a thick thermal boundary layer can alter the sampling and skimming process of the ICP-MS, as discussed by Douglas et al. [7] and Chambers et al [11] (noted previously in section 1.2.)

2.3.3 The Heat Equation

Fluids experience both external and internal exchange of heat. An external exchange means the fluid gives or receives energy from its surrounding environment. Internal heat exchange results from particle collisions, and it is this process that drives a fluid toward thermal equilibrium. The heat equation is [20]:

$$\begin{aligned}\frac{\partial T}{\partial t} + v\cdot\,\nabla T = &-(\gamma-1)\,T\;\nabla\cdot v + (\gamma-1)\,\frac{1}{nk_b}\nabla\cdot(k\nabla T)\\ &+\frac{(\gamma-1)}{nk_b}\;d_{i\,j}\;\frac{\partial u_i}{\partial x_j}\end{aligned} \tag{2.6}$$

where

$$d_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\nabla\cdot\mathbf{u}\right) \tag{2.7}$$

As with the Navier-Stokes equation, FENIX data will be tested against the heat equation to verify that heat transfer is performed properly by our DSMC code.

2.3.4 Bernoulli's Equation (Conservation of Energy)

A particularly simple method for partially testing the Navier-Stokes equations is Bernoulli's equation. Bernoulli's equation states that a quantity B is constant along streamlines for adiabatic, inviscid fluids. It is expected that the ideal assumptions will be valid down the central axis of the simulation, and will fail if analyzed in the boundary layer near the sampler cone, as heat conduction and viscous effects begin to make a noticeable effect in the fluid behavior. Bernoulli's equation is:

$$\mathrm{B} = \frac{1}{2}v^2 + \frac{\gamma}{\gamma - 1}\frac{kT}{m} = \mathrm{Constant} \tag{2.8}$$

Fenix correctly produces this phenomenon, as will be shown hereafter.

2.4 Data Analysis Methods

Raw data from FENIX is averaged in two ways: First, each grid point in the simulation represents a cell with multiple particles. The physical properties of those simulation particles are averaged to produce one data set per cell. The data sets produced from FENIX are as follows: density, temperature, pressure, Mach number, x-velocity, y-velocity, z-velocity, and velocity magnitude. Secondly, as the simulation reaches steady state, the data are successively averaged over multiple time steps. Data are output from FENIX according to the cell position to which they correspond. The grid point information delineating the cell locations is also made available for post-processing.

Matlab code has been written which uses this data to test FENIX against the fluid and heat equations. (See Appendix A.) As shown in Eqs. 2.4 and 2.6, the fluid and heat equations require that first and second order derivatives be taken of the data. Taking derivatives of data is not as straightforward as taking a derivative of an algebraic equation. The scatter and irregularities found in data sets create trouble if one simply tries to analyze them with finite difference methods. This problem is tackled by using a least squares fitting method.

2.4.1 A Taylor Expansion With a Least Squares Approximation

An appropriate way to find derivatives of data is to perform a least squares analysis, fitting the data to an adequate approximate function. In our two dimensional (r, z) grid, a rectangle of a predetermined size is chosen, and the least squares analysis is expanded about the middle of the rectangle, located at (r_0, z_0).

A second order function of the following form is assumed:

$$g(r, z, a, b, c, d, e, f) = a + b(r - r_o) + c(z - z_o) + d(r - r_o)^2 + e(z - z_o)^2 + f(r - r_o)(z - z_o) \tag{2.9}$$

This is the Taylor Expansion, with parameters $a - f$ as the characteristic coefficients, where r and z are the radial and coordinates of data cell centers. The expansion is second order because we need to take second derivatives in order to analyze the fluid equations. Fitting $g(r_i, z_i, a, b, c, d, e, f)$ to the data by least squares analysis requires that the square of the differences between the FENIX data values and the values of $g(r_i, z_i, a...f)$ be minimized.

Let F_i be the FENIX output data value of a given quantity (velocity, temperature, etc...), and let $g_i(r_i, z_i)$ be the Taylor expansion value of that same quantity. The index i represents each data cell in the analysis rectangle. S is defined as the sum of the squared residuals:

$$S = \sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \tag{2.10}$$

The minimization is done by taking the derivative of S with respect to each of the coefficients $a - f$, and setting them to zero:

$$\frac{\partial S}{\partial a} = 0 = \frac{\partial}{\partial a} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.11}$$

$$\frac{\partial S}{\partial b} = 0 = \frac{\partial}{\partial b} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.12}$$

$$\frac{\partial S}{\partial c} = 0 = \frac{\partial}{\partial c} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.13}$$

$$\frac{\partial S}{\partial d} = 0 = \frac{\partial}{\partial d} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.14}$$

$$\frac{\partial S}{\partial e} = 0 = \frac{\partial}{\partial e} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.15}$$

$$\frac{\partial S}{\partial f} = 0 = \frac{\partial}{\partial f} \left(\sum_{i=1}^{N} (F_i - g(r_i, z_i, a...f))^2 \right) \tag{2.16}$$

Consider the first of these equations, Eq. 2.11. Taking the derivative, we obtain:

$$\sum_{i=1}^{N} F_i = \sum_{i=1}^{N} \left(a + b\,\rho_i + c\,\zeta_i + d\,\rho_i^2 + e\,\zeta_i^2 + f\,\rho_i\,\zeta_i \right) \tag{2.17}$$

where $\rho_i = r_i - r_0$ and $\zeta_i = z_i - z_0$.

Inspection of this equation shows that it is one row of a set of linear equations. In matrix form this row is:

$$\begin{bmatrix} N & \sum_{i=1}^{N} \rho_i & \sum_{i=1}^{N} \zeta_i & \sum_{i=1}^{N} {\rho_i}^2 & \sum_{i=1}^{N} {\zeta_i}^2 & \sum_{i=1}^{N} \rho_i \, \zeta_i \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \sum_{i=1}^{N} F_i \tag{2.18}$$

The same may be done for each of the equations in the set. The resulting set of equations may all be combined into one, as follows:

$$\sum_{i=1}^{N} \begin{bmatrix} 1 & \rho_i & \zeta_i & {\rho_i}^2 & {\zeta_i}^2 & \rho_i \, \zeta \\ \rho_i & {\rho_i}^2 & \rho_i \, \zeta_i & {\rho_i}^3 & \rho_i \, {\zeta_i}^2 & \rho_i^2 \, \zeta_i \\ \zeta_i & \rho_i \, \zeta_i & {\zeta_i}^2 & {\rho_i}^2 \, \zeta_i & {\zeta_i}^3 & \rho_i \, {\zeta_i}^2 \\ {\rho_i}^2 & {\rho_i}^3 & {\rho_i}^2 \, \zeta_i & {\rho_i}^4 & \rho_i^2 \, \zeta_i^2 & \rho_i^3 \, \zeta_i \\ \zeta_i^2 & \rho_i \, \zeta_i^2 & \zeta_i^3 & \rho_i^2 \, \zeta_i^2 & \zeta_i^4 & \rho_i \, \zeta_i^3 \\ \rho_i \, \zeta_i & \rho_i^2 \, \zeta_i & \rho_i \, \zeta_i^2 & \rho_i^3 \, \zeta_i & \rho_i \, \zeta_i^3 & \rho_i^2 \, \zeta_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \sum_{i=1}^{N} \begin{bmatrix} F_i \\ \rho_i \, F_i \\ \zeta_i \, F_i \\ {\rho_i}^2 \, F_i \\ {\zeta_i}^2 \, F_i \\ \rho_i \, \zeta_i \, F_i \end{bmatrix} \tag{2.19}$$

The matrix is then inverted and the coefficients are found. This process is followed for each physical quantity, defining approximate functions for velocity, density, pressure, and temperature:

$$T = T_{00} + T_{01} \, \rho + T_{10} \, \zeta + T_{02} \, \rho^2 + T_{20} \, \zeta^2 + T_{11} \, \rho \, \zeta \tag{2.20}$$

$$P = P_{00} + P_{01} \, \rho + P_{10} \, \zeta + P_{02} \, \rho^2 + P_{20} \, \zeta^2 + P_{11} \, \rho \, \zeta \tag{2.21}$$

$$n = n_{00} + n_{01} \, \rho + n_{10} \, \zeta + n_{02} \, \rho^2 + n_{20} \, \zeta^2 + n_{11} \, \rho \, \zeta \tag{2.22}$$

$$vx = vx_{00} + vx_{01} \, \rho + vx_{10} \, \zeta + vx_{02} \, \rho^2 + vx_{20} \, \zeta^2 + vx_{11} \, \rho \, \zeta \tag{2.23}$$

$$vy = vy_{00} + vy_{01} \, \rho + vy_{10} \, \zeta + vy_{02} \, \rho^2 + vy_{20} \, \zeta^2 + vy_{11} \, \rho \, \zeta \tag{2.24}$$

$$vz = vz_{00} + vz_{01} \, \rho + vz_{10} \, \zeta + vz_{02} \, \rho^2 + vz_{20} \, \zeta^2 + vz_{11} \, \rho \, \zeta \tag{2.25}$$

In order to avoid the repetitious use of the constants $a-f$, they have been renamed in each equation (e.g. in the temperature equation, $a = T_{00}$, $b = T_{01}$ and so forth). It is then a simple matter to take derivatives of the functions in 2.20 - 2.25 both in the r and the z directions.

2.4.2 Application of Least Squares Fit to Data

The Taylor expansion integrated with the Least-Squares Approximation method does not magically circumvent the difficulty associated with having complex variations in (r, z). The choice of size of the analysis rectangle comes from a necessary trade-off between data smoothness and statistical accuracy. A very large analysis rectangle will smooth out all of the statistical noise, but the data over such a large region will not be fit well by a quadratic function of r and z. A small rectangle will give improved spatial resolution and show fine details, but if it is too small, statistical noise will dominate the data.

Numerical testing shows that a square analysis region of 8×8 grid points works about as well as seems possible, although it will be shown that in some cases, the analysis region size must be modified to accommodate steep derivatives. Using the FENIX data grid, the post-processing code defines such a region, which may be located anywhere in the simulation region. Using this method, velocity, temperature, density, and pressure functions may be created for any region of the ICP-MS.

The least squares fitting method used to reduce FENIX data is capable of smoothing the data enough to perform the second derivatives in the fluid and energy equations. It is limited to areas 7 grid points away from internal surfaces, however. This is due to a compromise between statistical and computational limitations. The fitting method uses rectangles of variable size to center the smoothing function. Smaller fitting rectangles are capable of approximating data closer to surfaces, but if the rectangles are too small, they won't contain enough information to give proper statistics. It is also possible to use a rectangle in which the observation point is at the edge rather than at the center to obtain values near the edge of the nozzle, but without centering the numerical method gives poor results. The results presented in Chapter 3 demonstrate this limitation.

2.4.3 Overview of Data Analysis Procedure

The following list gives a brief synopsis of the overall data reduction sequence:

1. Map Fenix data files into Matlab arrays
2. Define an analysis window in the area of interest within the simulation geometry
3. Looping through all the data within this window, build the coefficients in Eq. 2.10
4. Perform the Matrix inversion from Eq. 2.19, solving for the coefficients.
5. Use the coefficients to perform the derivatives needed to evaluate the terms in the fluid equations.
6. Move the analysis window to the next grid point along the desired path in the (r, z) plane
7. Repeat this process to find the variation of the terms in the fluid equations in r and z.

Chapter 3

Fluid Theory as Compared to Simulation Data

As the fluid properties in the ICP-MS are to be discussed, the temperature, density, and streamline profiles produced by FENIX are given here. In Fig. 3.1 the gas is shown to enter the simulation region at about 5400K. The cone is temperature is 1500K, and the gas cools down while it transfers heat to the cone in the thermal boundary layer. The cooling of the gas due to expansion can be seen in the nozzle and behind the cone.

The gas density, as seen in Figure 3.2 builds up against the cool cone in the upstream region because pressure is nearly constant for low-speed flow and the ideal gas law requires that a drop in temperature be accompanied by an increase in density. The density is higher in the boundary layer inside the nozzle as well because even though the gas is moving rapidly there, the radial profile of pressure is still nearly constant. But in the axial direction the pressure drops rapidly along the nozzle forcing the gas to expand creating very low density beyond the sampler cone.

The streamlines are shown in Fig. 3.3, and show the fluid flow through the nozzle.

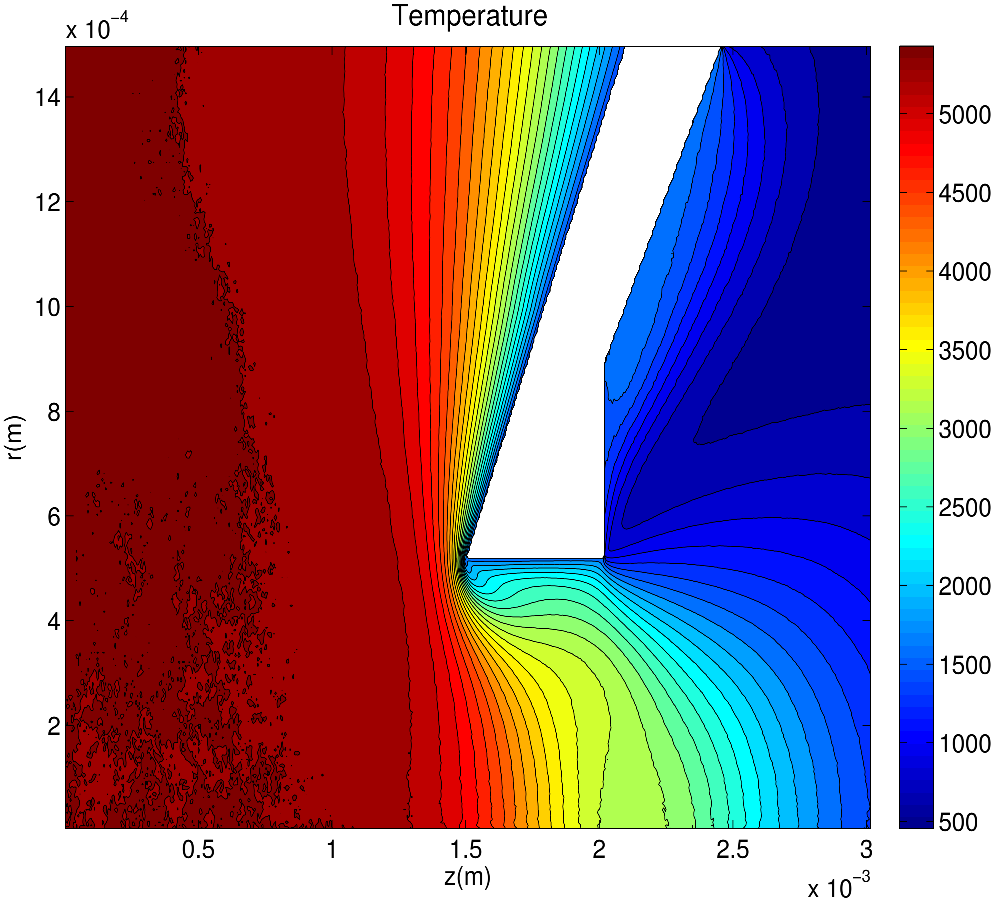

Figure 3.1 The simulated ICP-MS temperature profile presented here is produced by FENIX data.

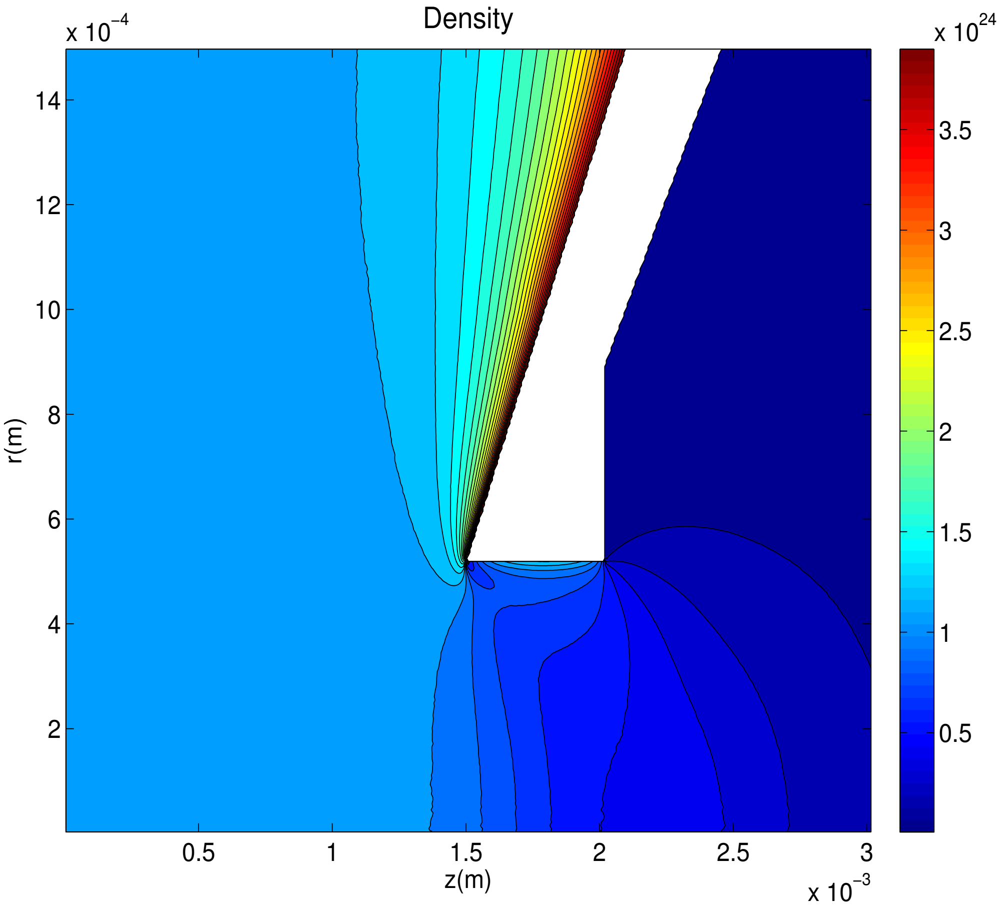

Figure 3.2 The simulated ICP-MS Density Profile presented here is produced by FENIX data

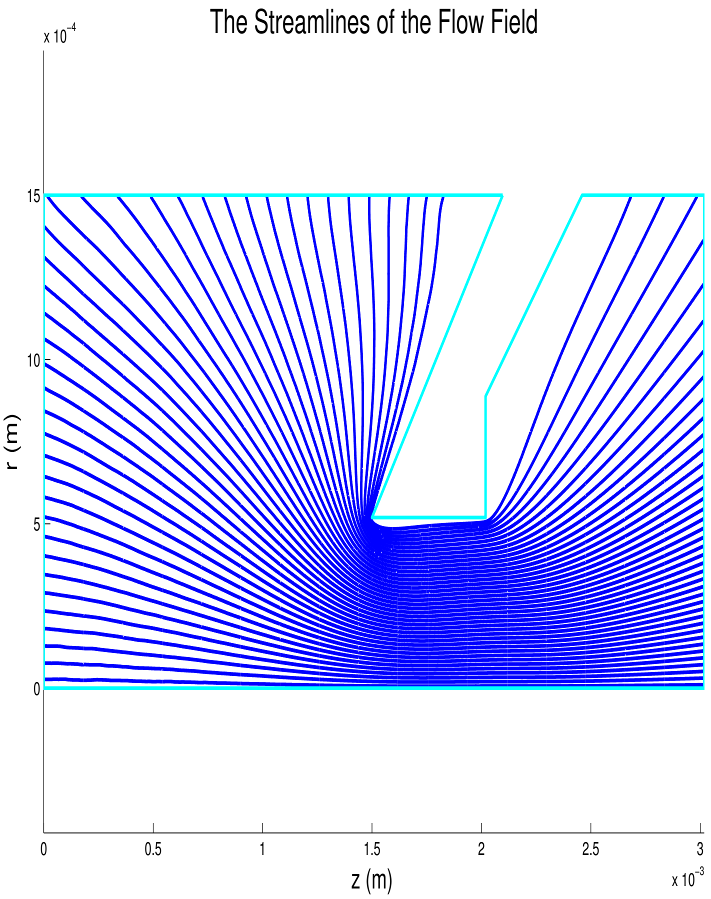

Figure 3.3 The simulated ICP-MS Fluid Streamlines presented here are produced from FENIX data.

3.1 Bernoulli's Equation

As a first example of testing FENIX, a simple equation will be analyzed: Bernoulli's equation. Bernoulli's equation is the integral of the Navier-Stokes equation along streamlines in the limit of zero viscosity and zero thermal conduction. Bernoulli's law states that the quantity B in Eq. 3.1 is constant along streamlines for adiabatic, inviscid fluids.

$$\mathrm{B} = \frac{1}{2}v^2 + \frac{\gamma}{\gamma - 1}\frac{kT}{m} = \mathrm{C} \tag{3.1}$$

B is very nearly the same constant for all streamlines in the ICP-MS simulation. This is due to the very small velocity and high temperature upstream. In the upstream region near $z = 0$, $v \approx 20\frac{m}{s}$, making $v^2 \ll \frac{\gamma}{\gamma-1}\frac{kT}{m}$. The upstream temperature is held at 5400 K, and as the temperature term is the dominant term, a constant temperature denotes a constant value of B for all streamlines.

An adiabatic gas or fluid is one for which no heat transfer occurs, and for which viscous effects are negligible. The gas may only compress or expand, undergoing adiabatic heating or cooling respectively. In order for non-ideal effects to be minimal, particle collisions should be very frequent, and the fluid should be unaware of any viscous layers created by external boundaries.

The ICP-MS sampler cone presents an external boundary, and any fluid passing along it will exchange heat with it. As seen in Fig. 3.1, there is a very strong temperature gradient near the nozzle. There is also a thermal boundary layer (inside the nozzle region) extending approximately 0.1 mm from the nozzle toward the axis of symmetry. The fluid is not adiabatic in any of these regions.

The viscous boundary layer is seen in Fig 3.4, which shows the z-component of velocity in the simulation region.

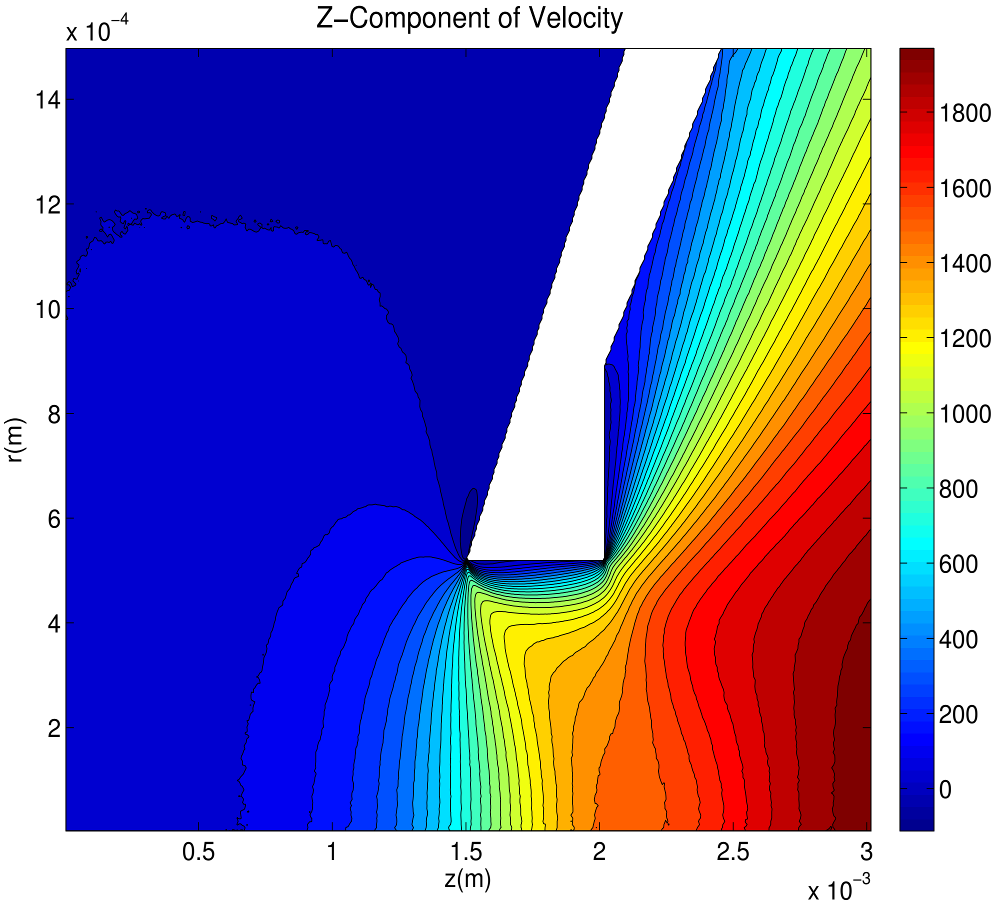

Figure 3.4 The simulated z-component of velocity presented here is produced by FENIX data.

The plasma discharge is, however, adiabatic along the axis of rotation in the simulation. In this region, the gas is unaware of any viscous or thermal boundaries, and of any cold surfaces. In this region, then, Bernoulli's equation should hold. The same data analysis method has been employed to examine Bernoulli's equation as the fluid and heat equations. There are no derivatives in Bernoulli's equation, though, keeping the data cleaner. In the post processing language described earlier, Bernoulli's equation becomes:

$$B = \frac{1}{2} v_{00}^2 + \frac{\gamma}{\gamma - 1} \frac{k_b T_{00}}{m}; \qquad (3.2)$$

where v_{00} represents the magnitude of the velocity in the data acquisition cell being processed, $\mathrm{k_b}$ is the Boltzmann constant, and m is the mass of argon. B is the constant quantity in Bernoulli's equation. In Fig. 3.5 it is shown that analyzing the data shows that Bernoulli's law is obeyed along the axis of the simulation region.

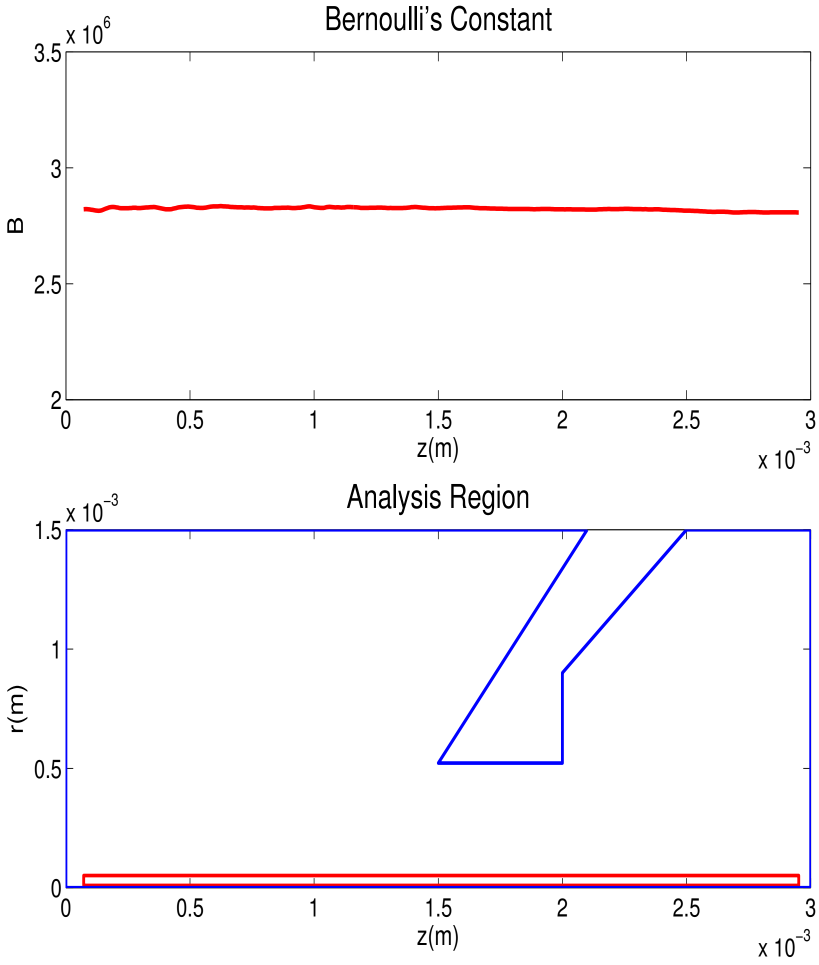

Figure 3.5 The value B in Bernoulli's Equation is evaluated along the central axis to determine if it remains constant.

In fact, the fluid's agreement to Bernoulli's equation (or deviance from it) gives us insight to the gas properties in the nozzle region. Scanning along the axis, and then moving progressively outward radially, one may see where the plasma discharge ceases to expand adiabatically through the nozzle.

On axis, B is a constant, the downstream value differing from the upstream by less than the statistical fluctuation level of about 2%. Approximately one third of the distance from the axis to the nozzle (0.21 mm), the deviation remains below 2%. By 0.39 mm, B is low by 10% because thermal conduction to the cool nozzle surface makes the flow be non-adiabatic.

It has been mentioned that the value of B is approximately constant for all streamlines. The question may arise, then, as to why an axial scan of B should decrease and later increase; B down the center of the nozzle should be conserved. This behavior is due to the fact that scans of B at higher radius cross into streamlines within the thermal boundary layer where heat is being conducted into the nozzle, then connect further downstream with the expanding streamlines from near the axis where B is constant. In Fig. 3.8, a closer look at streamlines shows why this is so: near the cone, streamlines are not parallel. B drops where the scan begins crossing into those streamlines of lower temperature that have already imparted heat to the the sampler cone. The streamlines have a somewhat circular motion through the nozzle, causing the higher radius scans of B to cross into lower temperature streamlines, and then back to higher temperature streamlines further downstream.

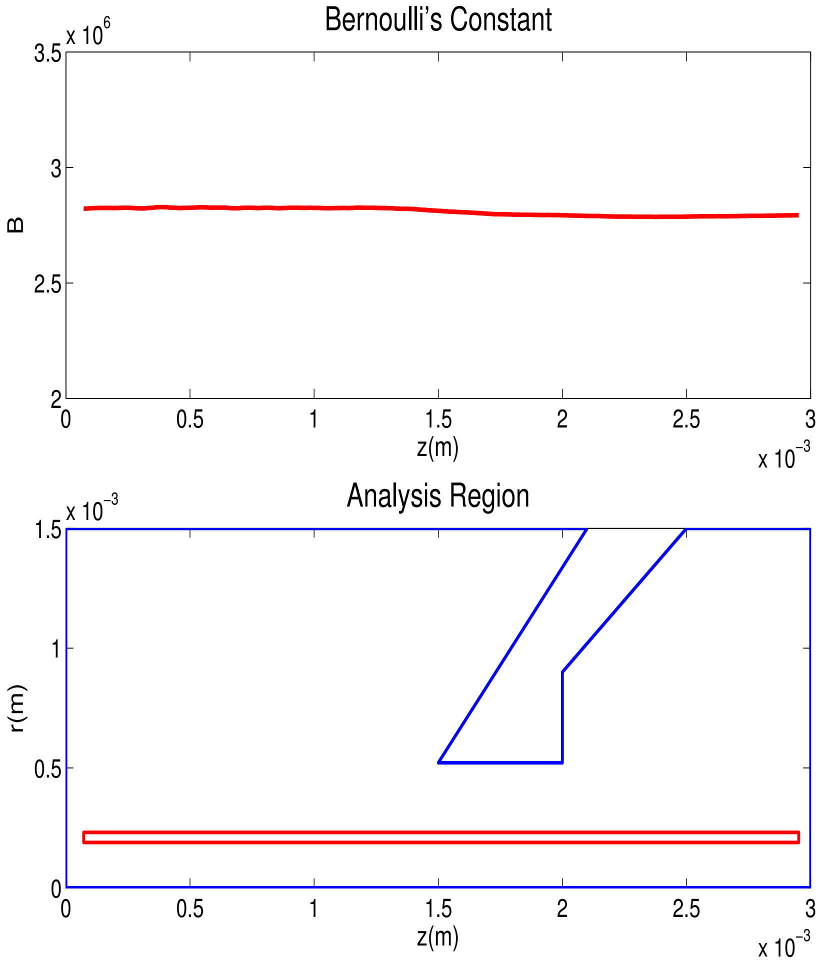

Figure 3.6 An axial scan of the value B from Bernoulli's equation is taken 0.18 mm from the central axis, showing that within 1% of its average upstream value, B remains constant.

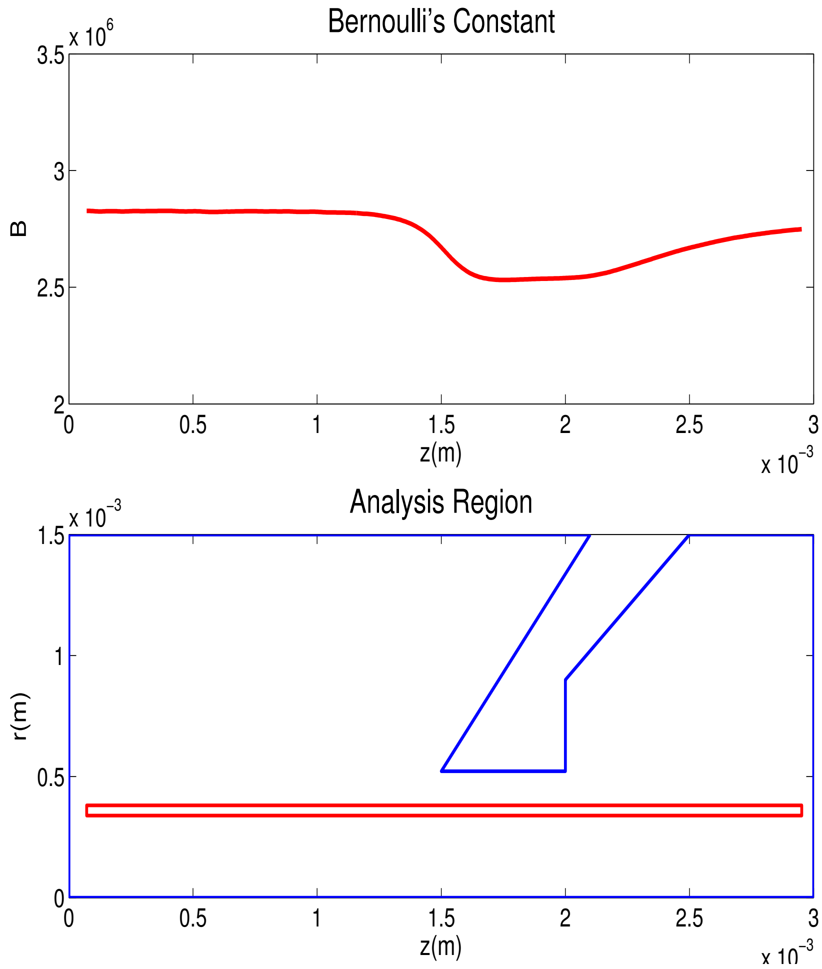

Figure 3.7 An axial scan of the value B from Bernoulli's equation is taken 0.36 mm from the central axis, showing that in the nozzle, B deviates about 10% from its upstream average value.

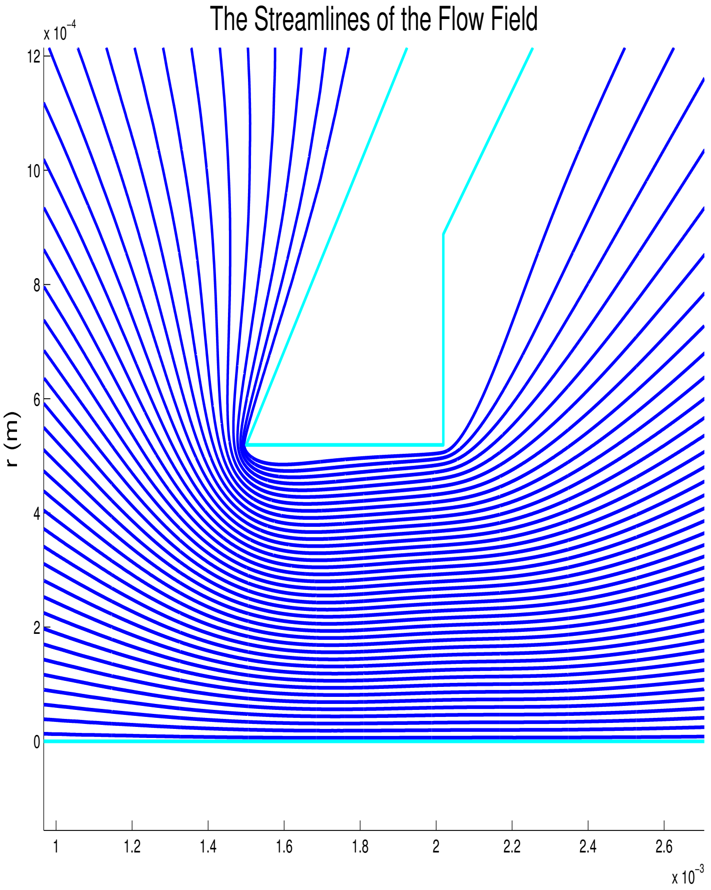

Figure 3.8 The simulated streamlines shown here are produced from FENIX data.

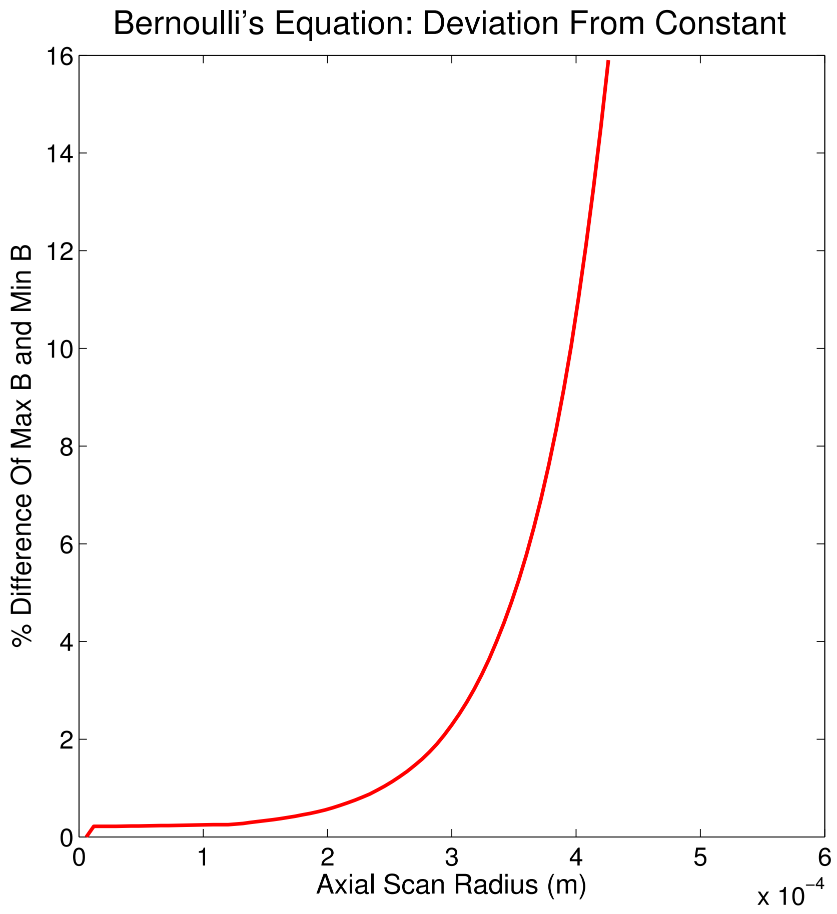

Figure 3.9 Axial scans of the value B from Bernoulli's equation were taken at different radii, and the percent difference between the upstream average value and the maximum value in the nozzle was taken. This figure is a radial scan of those percent differences.

3.2 The Navier-Stokes Fluid Equation

Recall that the Navier-Stokes equation is

$$\rho \frac{D\mathbf{u}}{D\,t} = -\nabla\mathbf{P} + \left[\nabla^{\mathbf{2}}\mu\ \mathbf{u} + \frac{\mathbf{1}}{\mathbf{3}}\nabla(\nabla\cdot\mu\ \mathbf{u})\right]$$

At steady state, any time derivatives are zero. Furthermore, the θ component is ignored, as the simulation assumes rotational symmetry. The material derivative on the left hand side of the Navier-Stokes equation, then, is expanded as:

$$\rho\left(u_r\frac{\partial u_r}{\partial r}+u_z\frac{\partial u_r}{\partial z}\right)\hat{r}+\rho\left(u_r\frac{\partial u_z}{\partial r}+u_z\frac{\partial u_z}{\partial z}\right)\hat{z}$$

These are the acceleration terms. The forces that produce them are on the right-hand side of the equation:

$$-\nabla\mathbf{P}+\left[\nabla^{\mathbf{2}}\mu\ \mathbf{u}+\frac{\mathbf{1}}{\mathbf{3}}\nabla(\nabla\cdot\mu\ \mathbf{u})\right]$$

In the regimes where the fluid equations are obeyed, the forces and the acceleration terms (the left and right hand sides of the equation) should be equal at every point in space. To analyze the ability of FENIX to properly simulate the fluid equations, portions of the simulation region are selected and the acceleration and force terms are plotted together. Analysis has been done upstream, downstream, and in the nozzle region of the simulation. The r-component of the Navier-Stokes equation will be presented first, followed by the z-component, with a discussion on boundary layer effects concluding Sec. 3.3

3.2.1 The r-Component of the Navier-Stokes Equation

The vector equation may be evaluated in component form, and it is in this manner that the data is best analyzed. The r component of the Navier-Stokes equation is given as:

$$
\begin{aligned}
\rho\left(u_r\frac{\partial u_r}{\partial r}+u_z\frac{\partial u_r}{\partial z}\right) &= \\
&-\frac{\partial P}{\partial r}+\mu\left[\frac{\partial^2 u_r}{\partial r^2}+\frac{1}{r}\frac{\partial u_r}{\partial r}+\frac{\partial^2 u_r}{\partial z^2}-\frac{u_r}{r^2}+\frac{1}{3}\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}r\,u_r+\frac{\partial u_z}{\partial z}\right)\right] \\
&+2\frac{\partial \mu}{\partial r}\frac{\partial u_r}{\partial r}+\frac{\partial \mu}{\partial z}\left(\frac{\partial u_r}{\partial z}+\frac{\partial u_z}{\partial r}\right)-\frac{2}{3}\frac{\partial \mu}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}r\,u_r+\frac{\partial u_z}{\partial z}\right) \qquad (3.3)
\end{aligned}
$$

The expression is complicated, but the Taylor expansion routine explained previously has already solved for a set of coefficients that return to us approximate functions for each physical parameter (density, pressure, viscosity, velocity, etc..):

$$f_{approx} = f_{00} + f_{10}\zeta + f_{01}\rho + f_{20}\zeta^2 + f_{02}\rho^2 + f_{11}\zeta\,\rho; \qquad (3.4)$$

and it is relatively simple to apply these approximation functions to the fluid equations. Using these forms, the left-hand and right-hand sides of the fluid equations can be calculated and compared.

The Upstream Region

Radial Analysis In the upstream region, particle velocities are small compared to the local speed of sound. This makes both velocities and pressure gradients small enough that statistical noise from DSMC dominates. The computatinonal cost of reducing this scatter is simply too great.

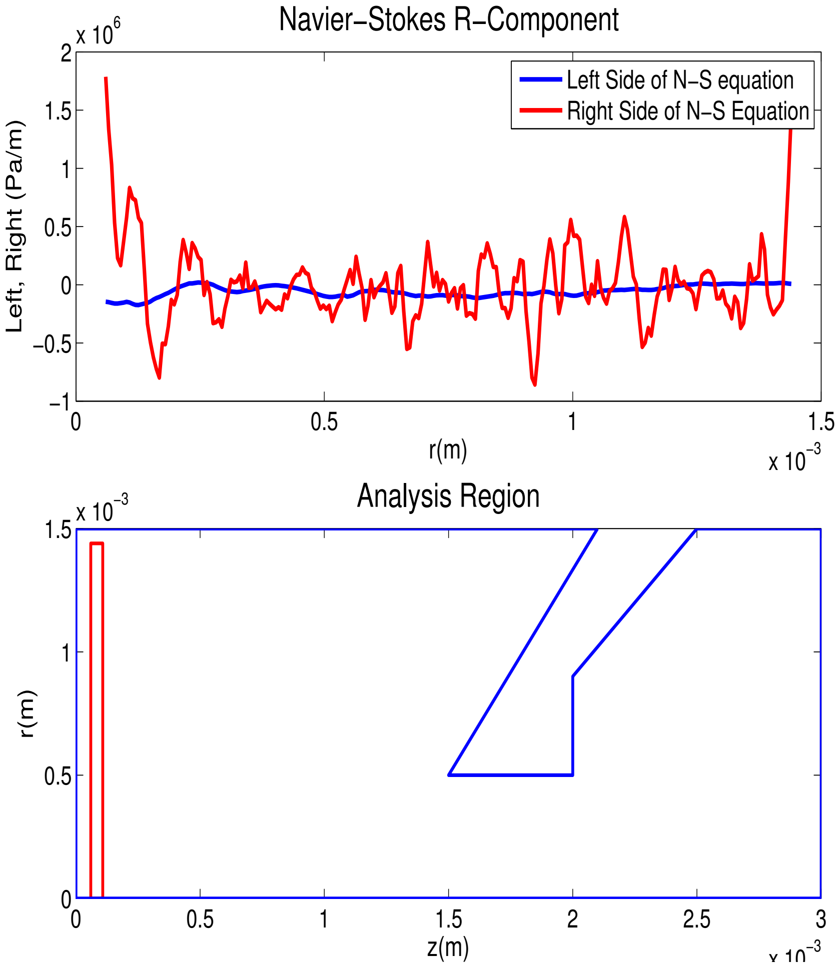

Figure 3.10 Radial scans of the left and right hand terms of the Navier-Stokes r-component 0.084 mm downstream from the front end of the simulation are compared.

Further downstream, as the analysis moves toward the nozzle, the flow velocity increases rapidly, as does the pressure gradient. The ability of the Navier-Stokes equations to test the simulation improves.

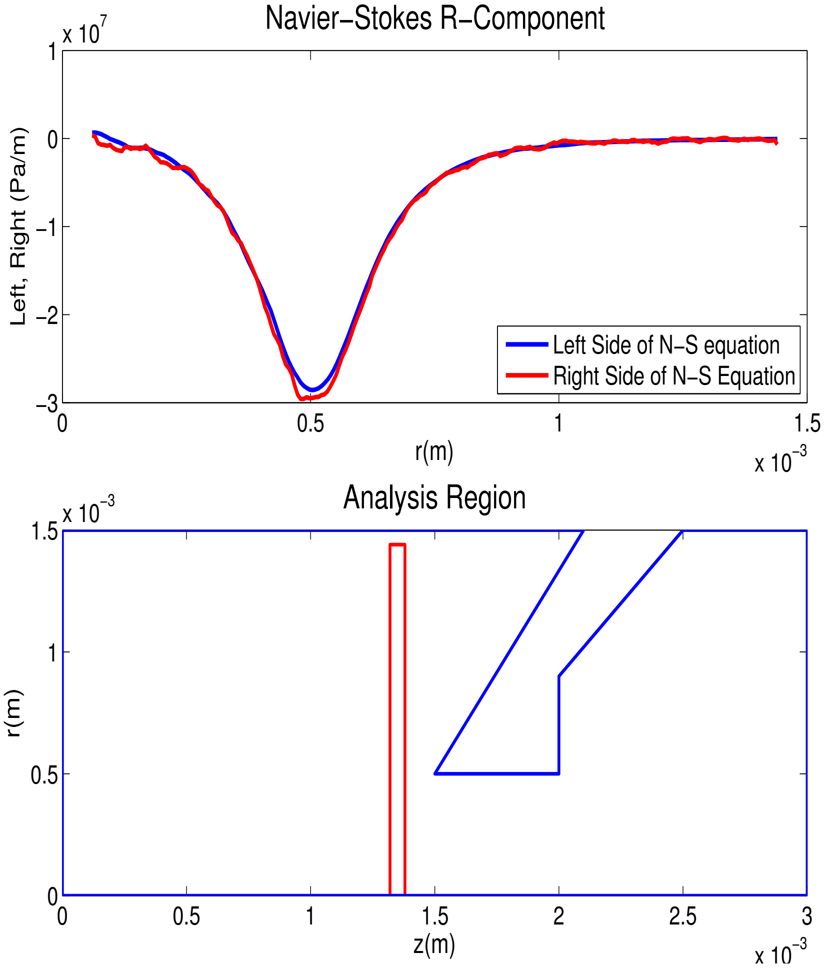

Figure 3.11 Radial scans of the left and right hand terms of the Navier-Stokes r-component 1.35 mm downstream from the front end of the simulation are compared.

From Figs. 3.11-3.12, it can be seen that the change in momentum is directed strongly in the negative R direction up to about about $r = 0.5$ mm. As the fluid nears the nozzle, it is pulled by the pressure gradient into the orifice. Some of the particles must turn quickly around the edge of the cone; this process happens over a relatively small length scale. The result of this interaction with the nozzle is high velocity gradients associated with an immediate change in direction. This presents a significant problem in testing the simulation in any region close to the nozzle, as seen in Fig. 3.12. The least squares fitting method uses a data field of 8 grid points, and if there are significant fluid property changes on a smaller scale than 8 grid points, the analysis will not notice them. Using a smaller analysis region implies that the statistics will be poorer, but the resolution will be better. For the region near the nozzle, then, a smaller data field (3 grid points) was used in the effort to improve resolution.

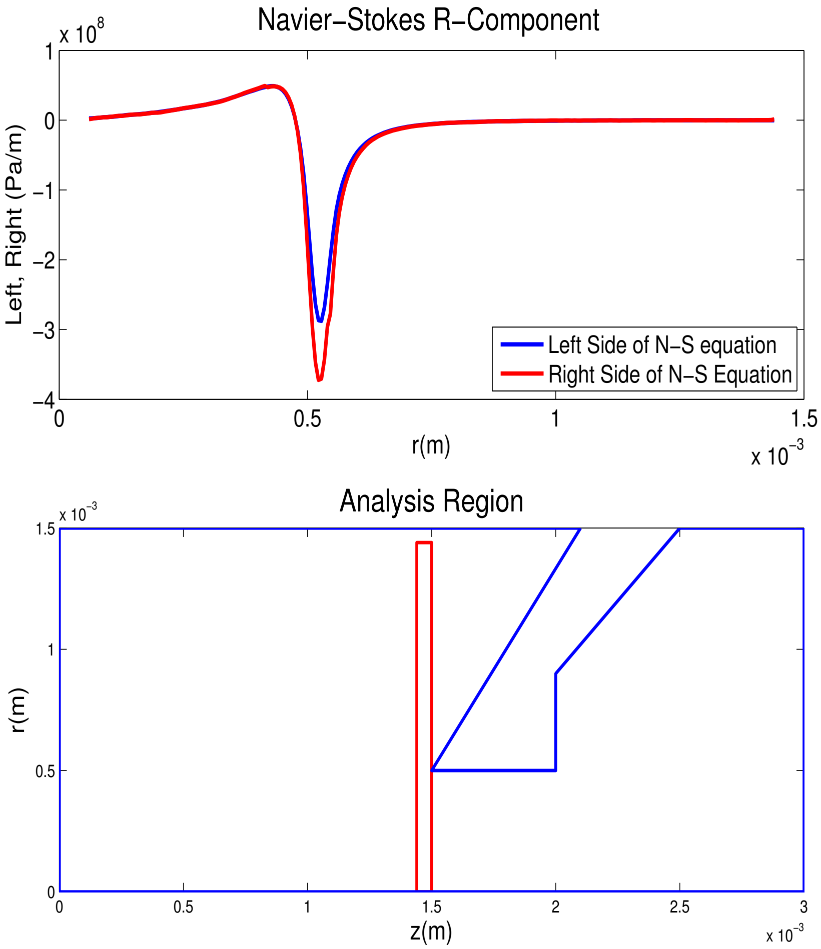

Figure 3.12 Radial scans of the left and right hand terms of the Navier-Stokes r-component 1.47 mm downstream from the front end of the simulation are compared.

Changing the analysis window did not correct the discrepancy. The Knudsen number was checked in this region and found to meet the Navier-Stokes requirement, and so we conclude that the velocity gradients around the tip of the nozzle are high enough to cause the least squares method of calculating second derivatives to fail.

As will be shown in Sec. 3.2.4, the left- and right-hand sides of the Navier-Stokes equation are of order 10^8. So the statistical noise in the upstream region has a relative error of about 1% while the error caused by the steep gradients near the nozzle tip, shown in Fig. 3.12 is of order 25%.

Axial Analysis In the previous section, the data analysis was performed by radially scanning the simulation data. It is useful to perform the analysis using axial scans as well. In the upstream region, there is almost no flow in the r direction, and fluctuations are large.

Scanning near the axis of symmetry is noisy. In this area, there are few particles per cell, resulting in poor statistics as seen in Fig. 3.13.

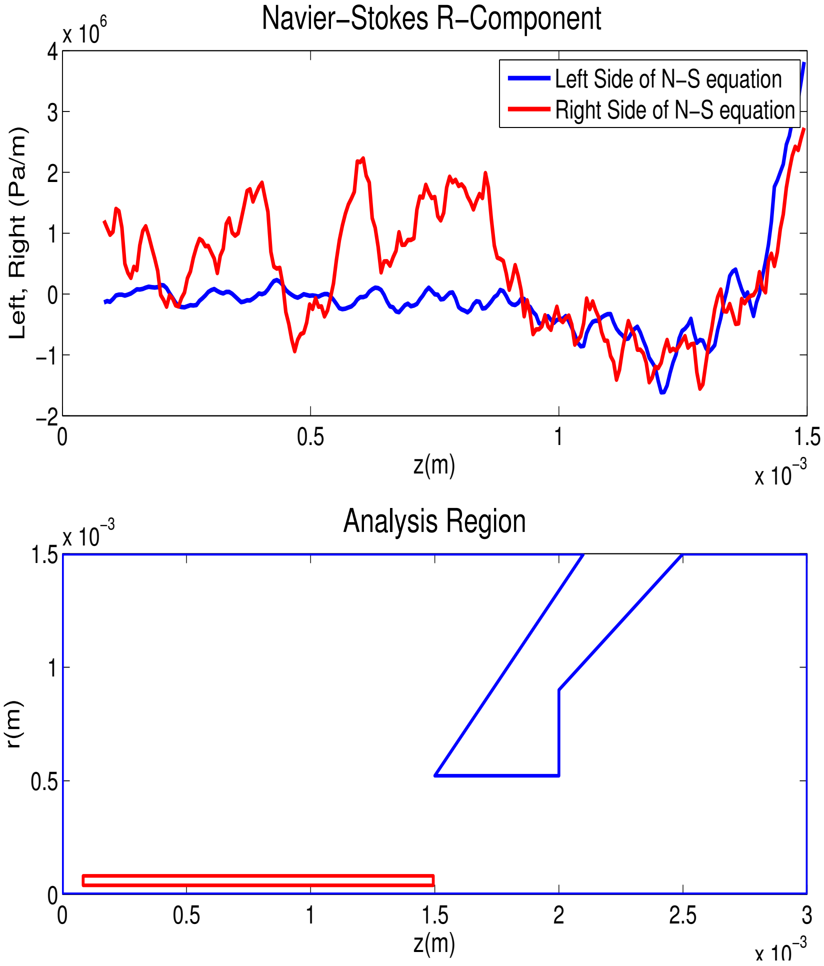

Figure 3.13 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.06 mm from the central axis of the simulation are compared.

Further from the axis, the statistics are better. Flow velocities in this region are high, and statistical fluctuations no longer dominate. It is seen in Fig. 3.14 that the r component of the fluid equations is obeyed in the upstream region.

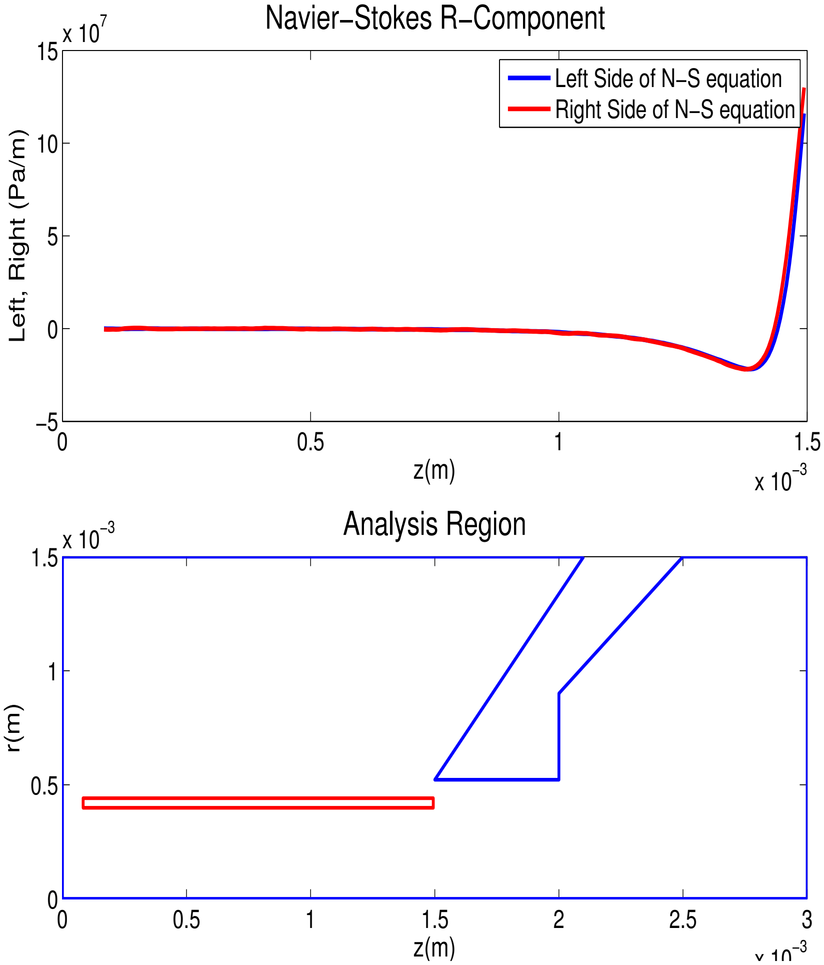

Figure 3.14 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.42 mm from the central axis of the simulation are compared.

As seen in Fig. 3.15, from about 1 mm downstream in the following figure, the change in momentum is in the decreasing r direction, as the fluid is pulled into the nozzle by the pressure gradient. The no slip condition though, requires that fluid does not travel parallel to the nozzle's surface at the fluid-surface interface. The r-component of momentum decreases as the fluid nears the surface of the cone.

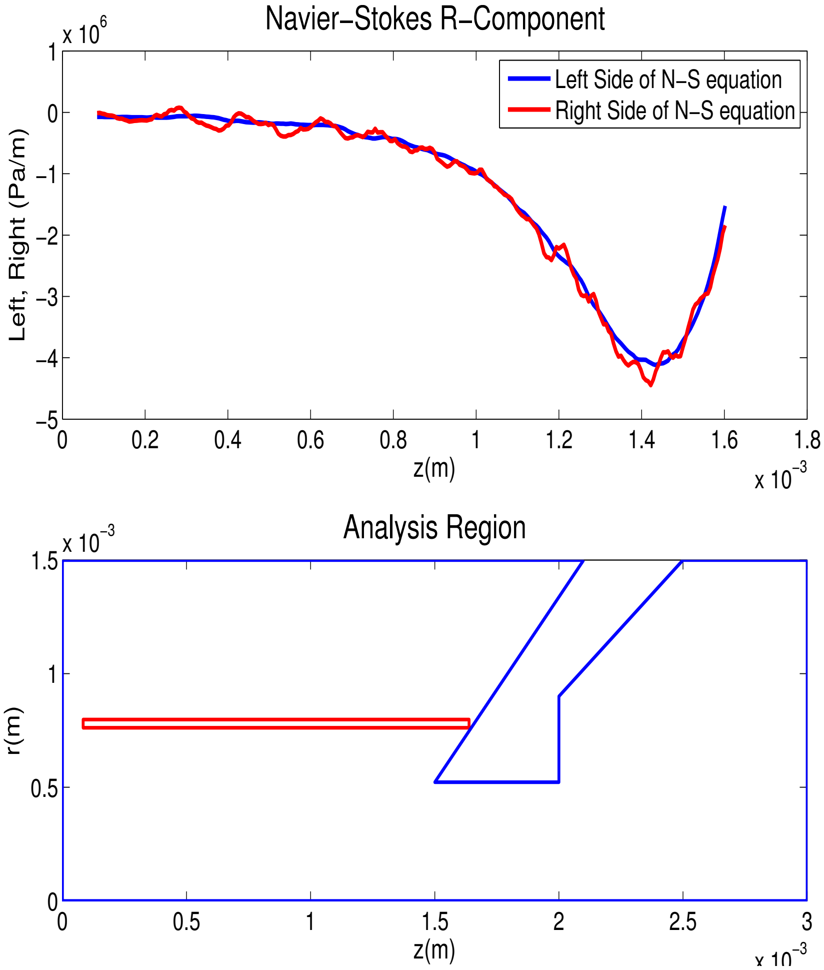

Figure 3.15 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.78 mm from the central axis of the simulation are compared.

As the radius increases, the fluid is less and less aware of the nozzle. There comes a point where the fluid is far away enough from the nozzle that it is not affected by it at all. In the large r portion of the upstream simulation, low particle velocities once again make the definition of flow somewhat inapplicable and the DSMC method gives results where statistical noise dominates, as seen in Fig. 3.16.

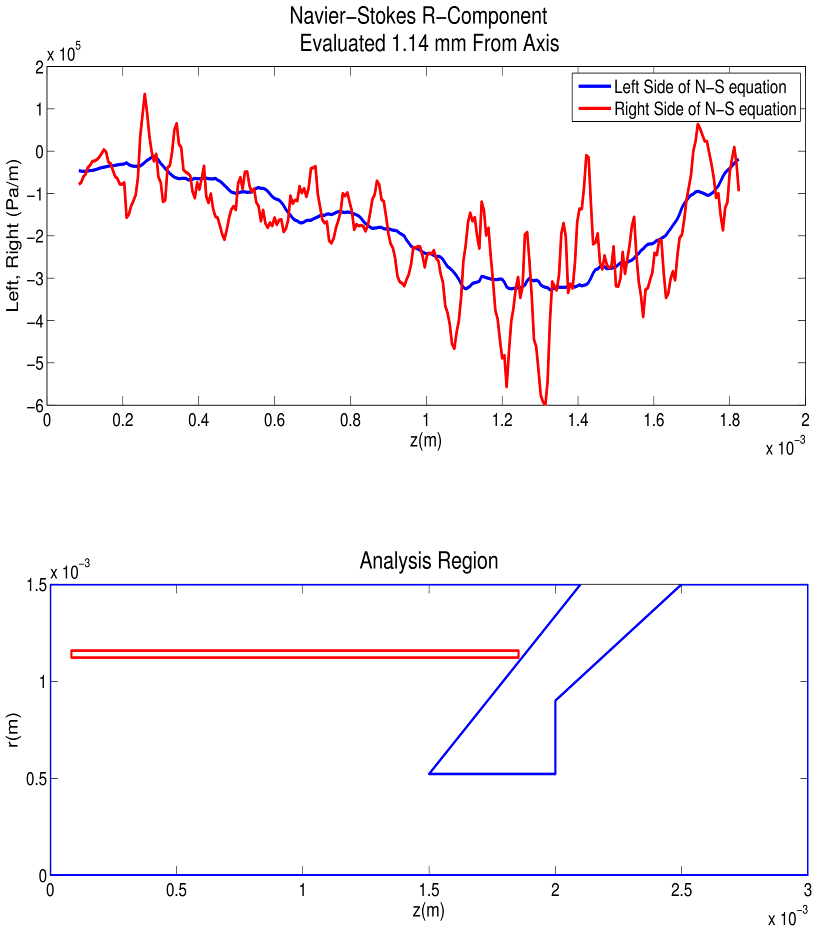

Figure 3.16 Axial scans of the left and right hand terms of the Navier-Stokes r-component 1.14 mm from the central axis of the simulation are compared.

FENIX is shown here to correctly reproduce the r component of the momentum equation in the upstream region of the ICP-MS simulation. Statistical noise is more dominant when scanning the r component of the momentum equation, because the force and acceleration terms are largest in the z direction. FENIX also is able to indicate locations where the data fitting methods used are insufficient. As shown in figure 3.12, this occurs where the velocity gradients become very large at the front tip of the sampler cone.

Note that the errors in these scans are also of order 1% away from the nozzle tip.

The Nozzle Region

Radial Analysis In the nozzle region, the fluid density is high (therefore collisions dominate the fluid behavior), the Knudsen number is low, and the streamlines are nearly parallel. This is the ideal location to apply the Navier-Stokes equations; the statistical fluctuation should be comparatively small here, since v_z is very large (near Mach 1). At the front end of the nozzle (the upstream end) there is one region previously discussed that will show some deviance from Navier-Stokes fluid theory, and this is right where the fluid rounds the tip of the nozzle. As previously mentioned, the near instantaneous change in direction leads to very large velocity gradients, and taking second derivatives of rapidly changing velocities produces poor results. It is expected that our testing will show good results everywhere except at the front and back end of the nozzle, as shown in Figs. 3.17 and 3.18.

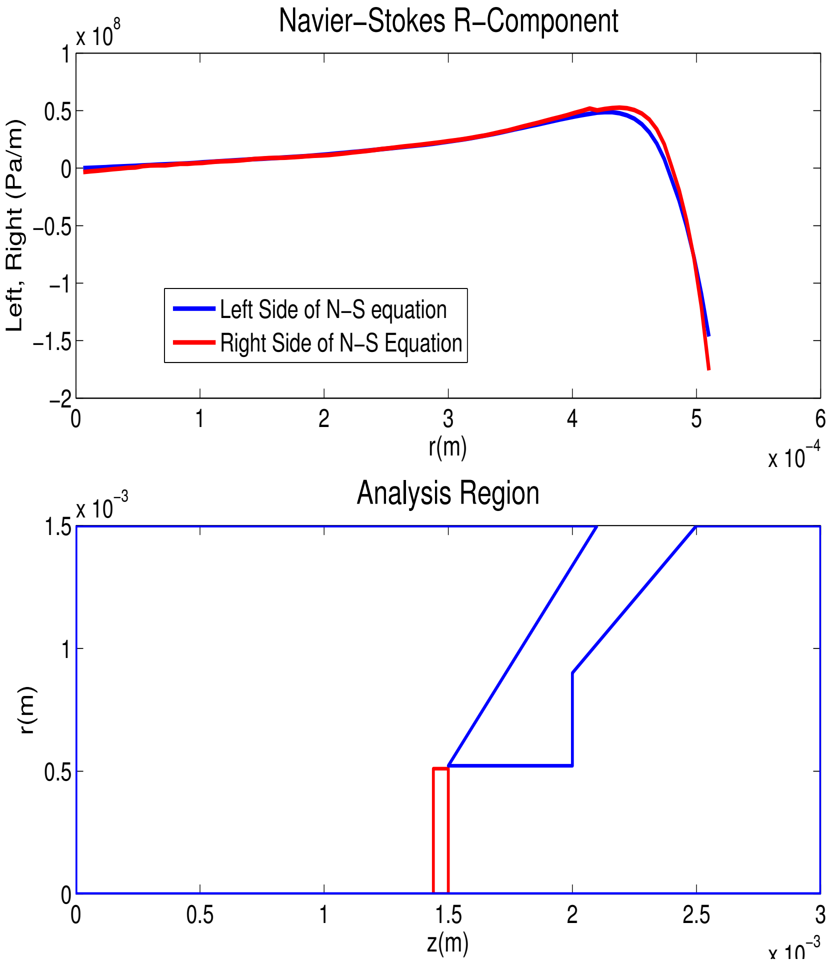

Figure 3.17 Radial scans of the left and right hand terms of the Navier-Stokes r-component 1.47 mm downstream from the front end of the simulation are compared.

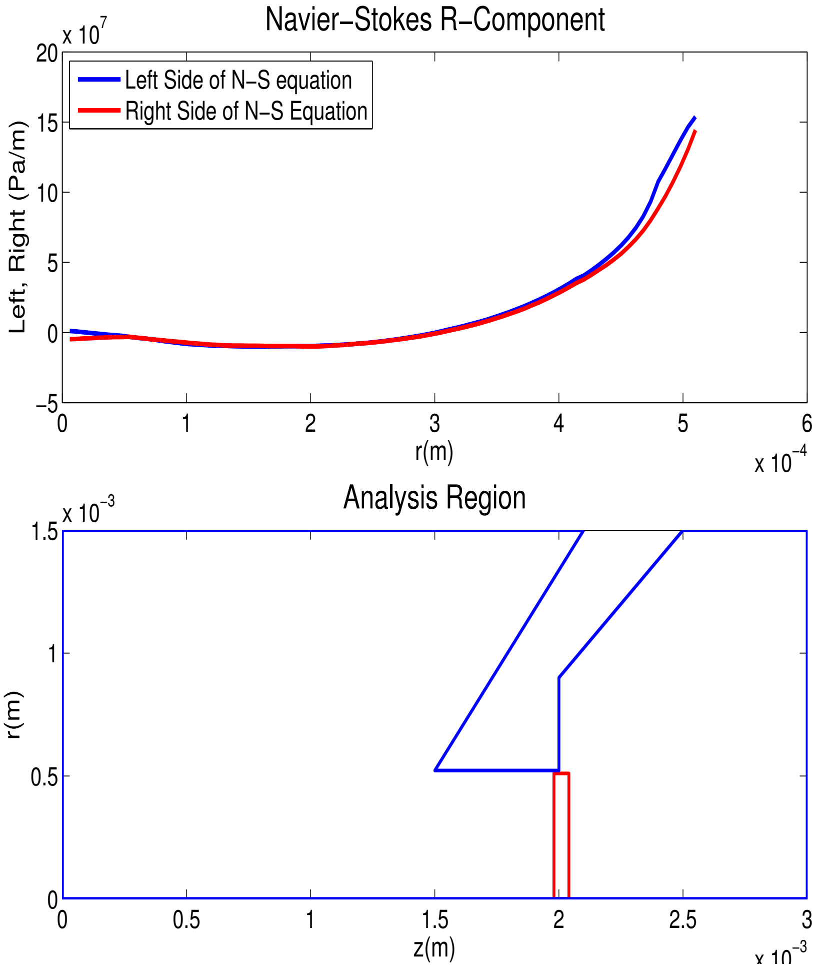

Figure 3.18 Radial scans of the left and right hand terms of the Navier-Stokes r-component 2.01 mm downstream from the front end of the simulation are compared.

The middle region of the nozzle is the ideal location for obedience to the Navier-Stokes equation, and is seen in Fig. 3.19 to be accurate within 2% in this area (comparing to the standard value of 10^8 used throughout for the Navier-Stokes equation.)

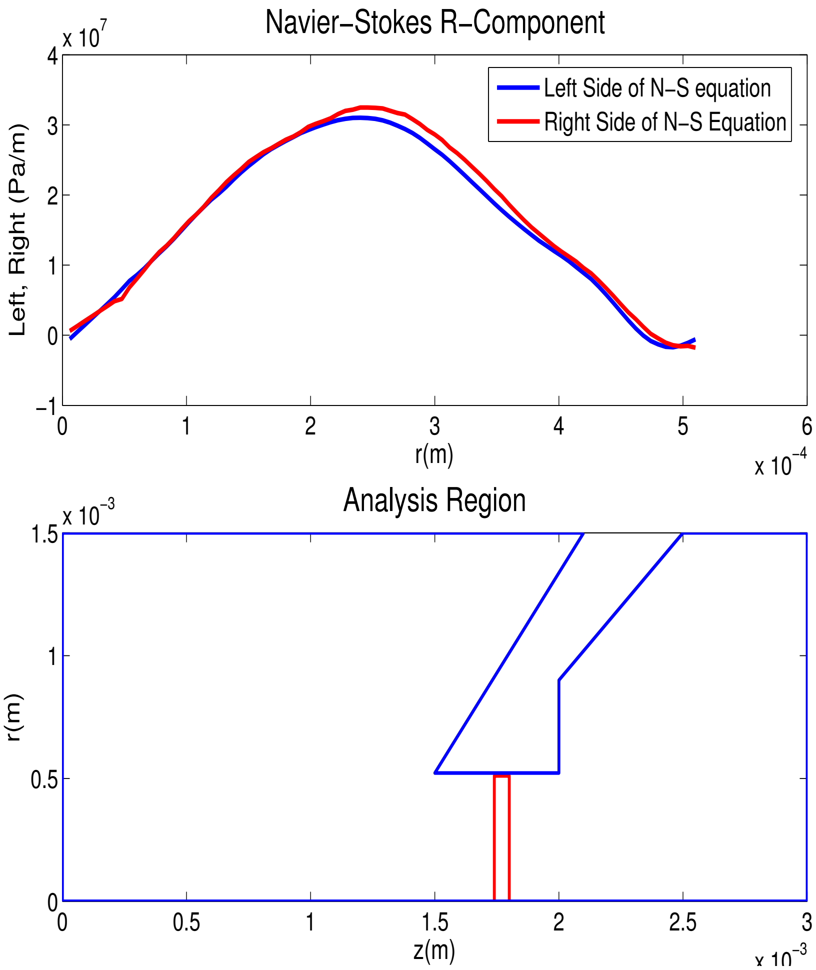

Figure 3.19 Radial scans of the left and right hand terms of the Navier-Stokes r-component 1.74 mm downstream from the front end of the simulation are compared.

Axial Analysis Axial data scans are taken as well. The statistics along the axis of symmetry are (as seen in Fig. 3.20) inadequate and statistical noise is highly noticeable.

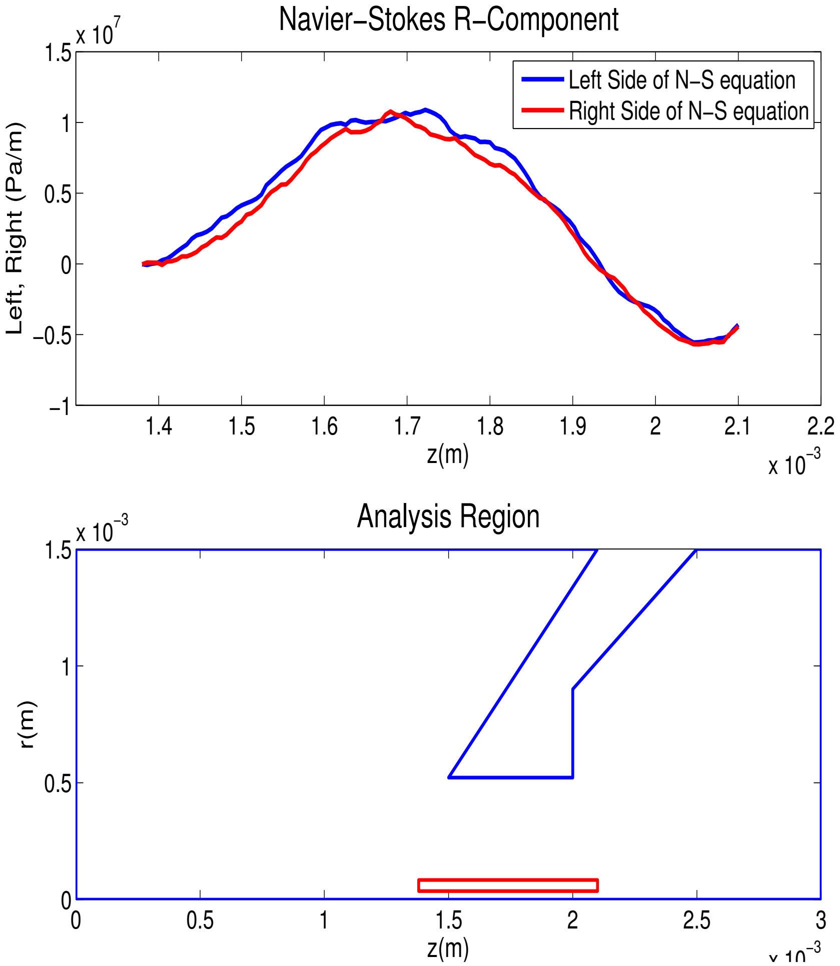

Figure 3.20 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.06 mm from the central axis of the simulation are compared.

Notice that in Fig. 3.21 the noise is small because the scan is well away from the axis. Notice also that the scan is not near the corner of the nozzle. In this region the r-component of the Navier-Stokes equation is satisfied well, to an accuracy of 2%, as seen in previous comparisons.

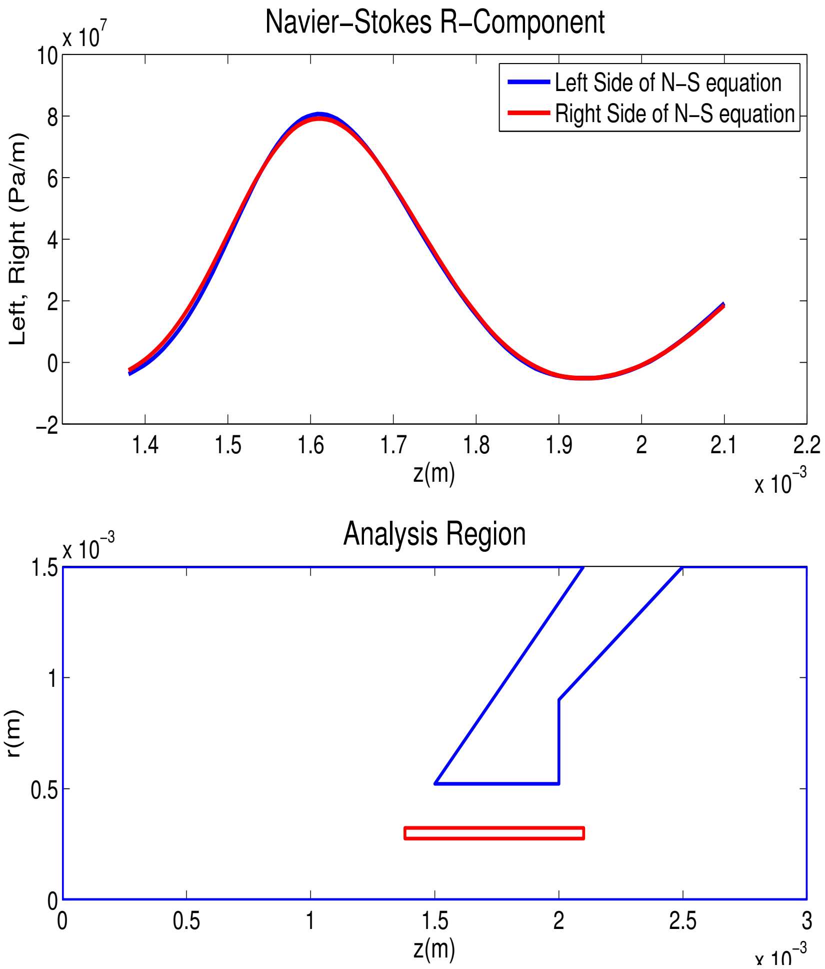

Figure 3.21 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.3 mm from the central axis of the simulation are compared.

Scanning with the typical 8 × 8 grid along the surface of the nozzle gives poor results. This is due to the difficulty of getting clean data along the surface of the nozzle. The analysis region smooths data over an 8 × 8 square, but along the surface of the nozzle, half of the analysis region is empty. This leads to poor extrapolation and fitting for the last 5-10 data points.

Using a thin least squares fitting rectangle that is 3 grid points wide in the r direction and 8 in the z direction allows axial scans to be taken close to the edge of the nozzle without having to extrapolate into the nozzle itself, giving better results at the nozzle-fluid interface. This scan (see Fig. 3.22) shows the effects of the corners of the nozzle where velocity gradients are high, again giving errors of about 20%. But away from the corners, the fluid equations are obeyed along the surface of the nozzle to an accuracy of about 2%, as seen before.

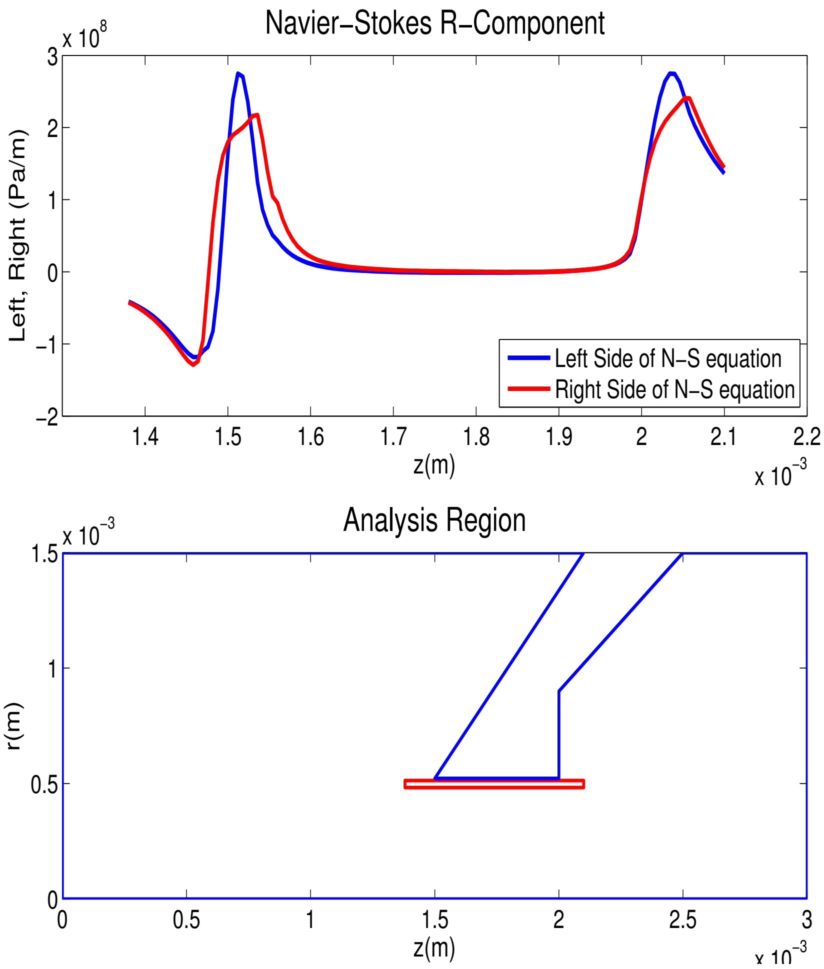

Figure 3.22 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.498 mm from the central axis of the simulation are compared.

3.2.2 The Downstream Region

The momentum equations are followed throughout the nozzle, and for some distance downstream, until the particles' mean free paths become very long and collisions no longer dominate the physics of the gas, at which point fluid dynamics are replaced by kinetic theory. The present simulation geometry extends three millimeters beyond the sampler cone exit, and fluid theory is valid this entire distance. In fact, with a well defined flow and the absence of any surfaces, the downstream expansion region is modeled very well by the fluid equations.

Figures 3.23 - 3.25 show the fluid equations' agreement as analyzed in radial strips. Fig. 3.23 shows a discrepancy at the back corner of the nozzle. As discussed before, this is where the fluid streamlines are bending rapidly around the nozzle. The velocity gradients are too high to correctly build a fitting function from the data.

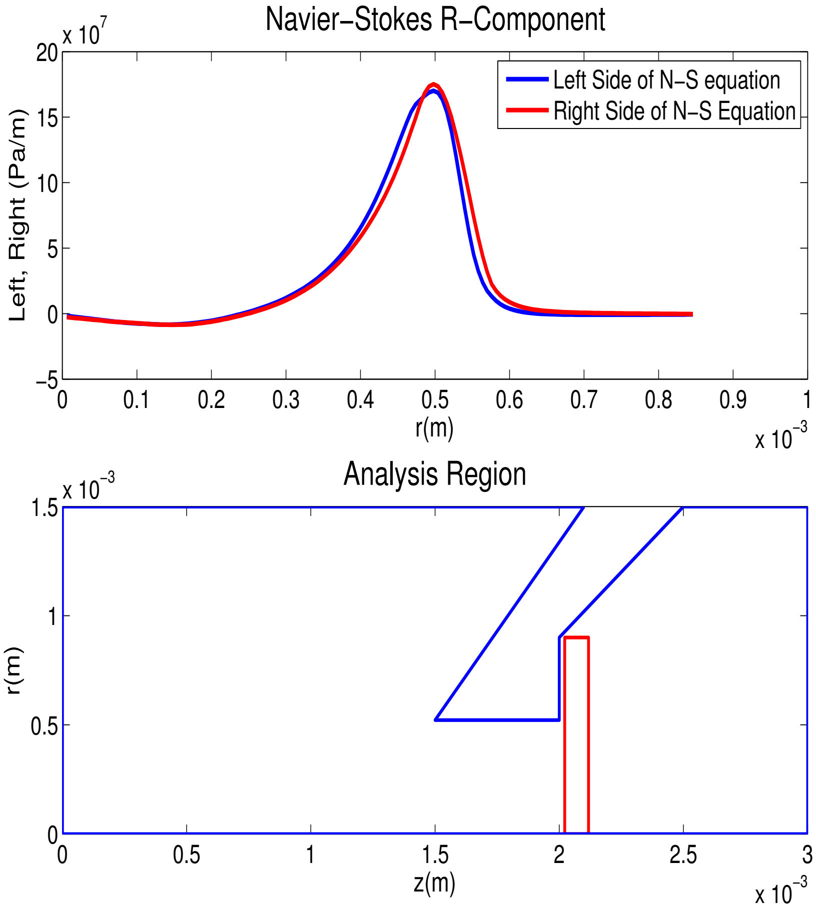

Figure 3.23 Radial scans of the left and right hand terms of the Navier-Stokes r-component 2.07 mm downstream from the front end of the simulation are compared.

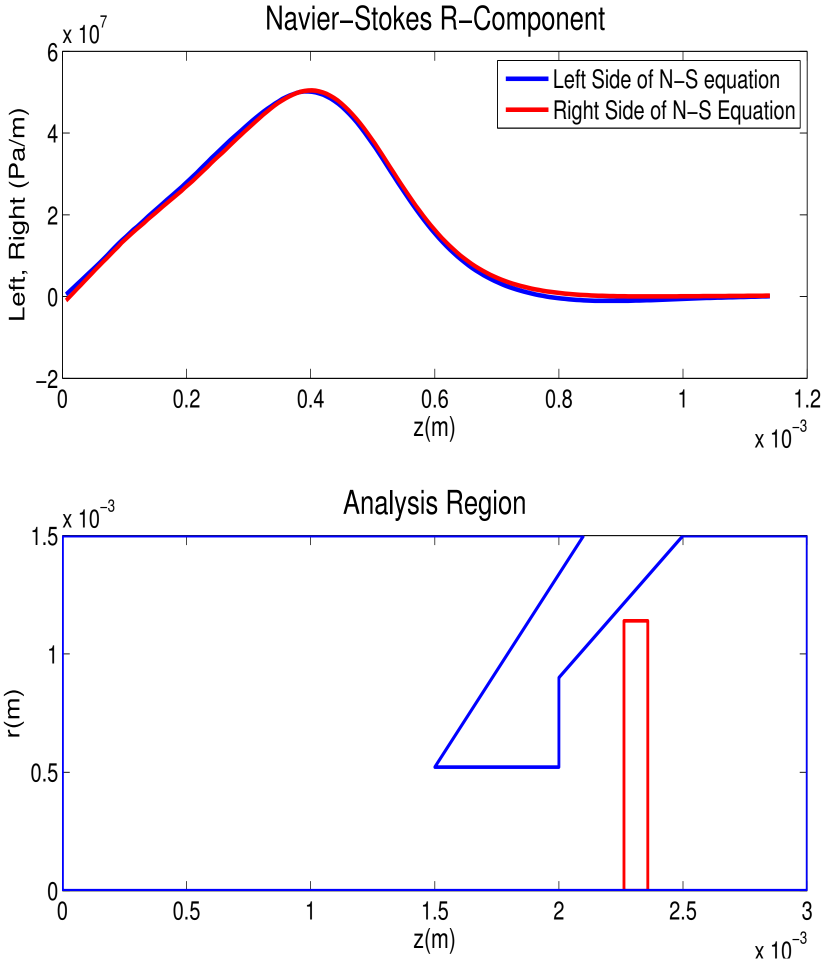

Figure 3.24 Radial scans of the left and right hand terms of the Navier-Stokes r-component 2.31 mm downstream from the front end of the simulation are compared.

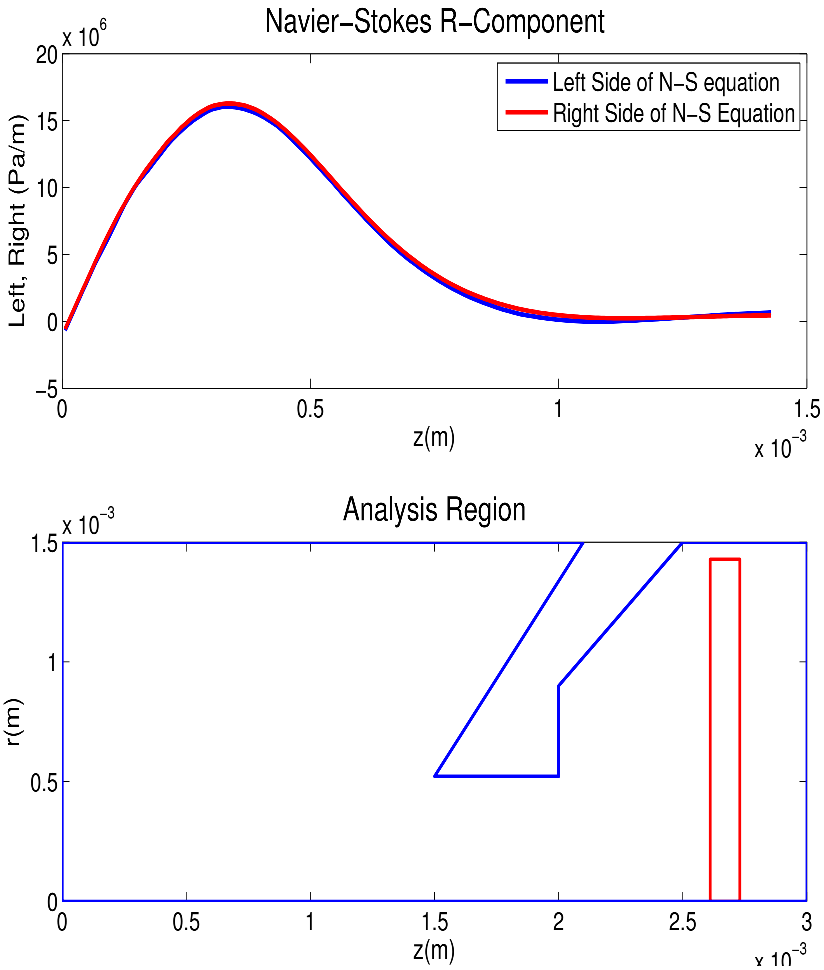

Figure 3.25 Radial scans of the left and right hand terms of the Navier-Stokes r-component 2.67 mm downstream from the front end of the simulation are compared.

From Figs. 3.23-3.25 the simulation is seen to correctly reproduce the Navier-Stokes equations in the downstream region to an accuracy of about 2% (comparing again to the standard value of 10^8.)

Axial Analysis Figures 3.26 - 3.29 show the fluid equations' agreement as analyzed in axial strips. Scans at values of r greater than 0.5 mm have a small pressure gradient term compared to scans directly behind the sampler cone aperture. The scale of terms is smaller, and it is easier to see in Figs. 3.26 - 3.29 a discrepancy in the Navier-Stokes equation.

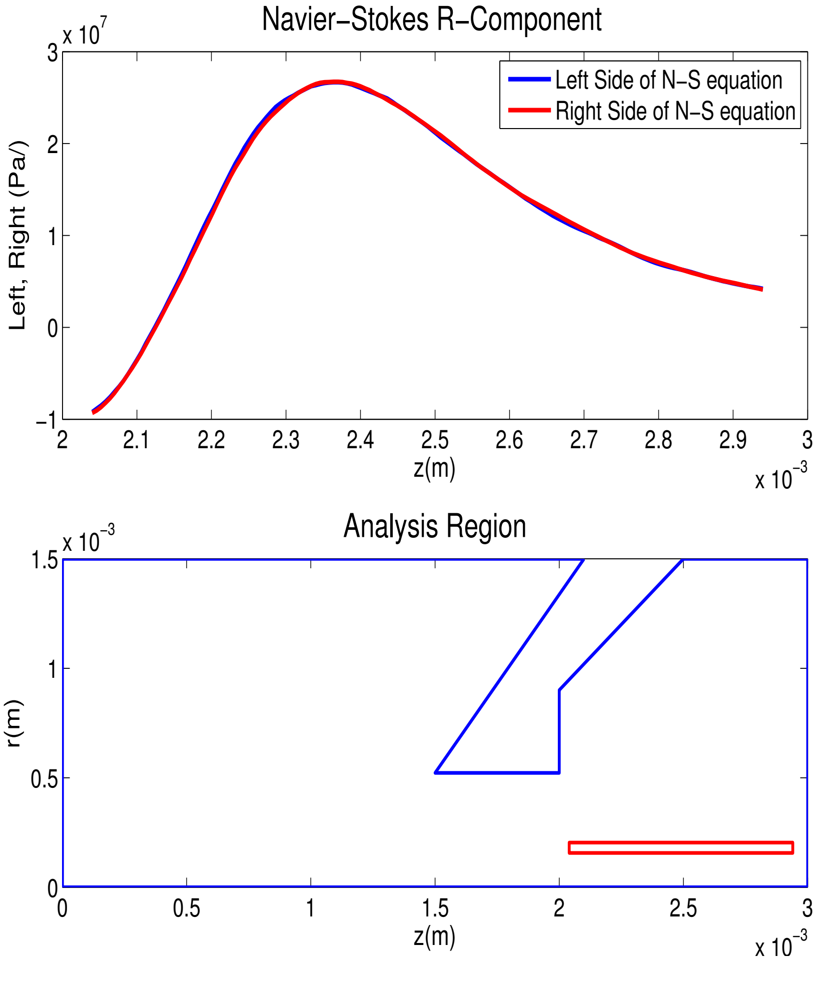

Figure 3.26 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.18 mm from the central axis of the simulation are compared.

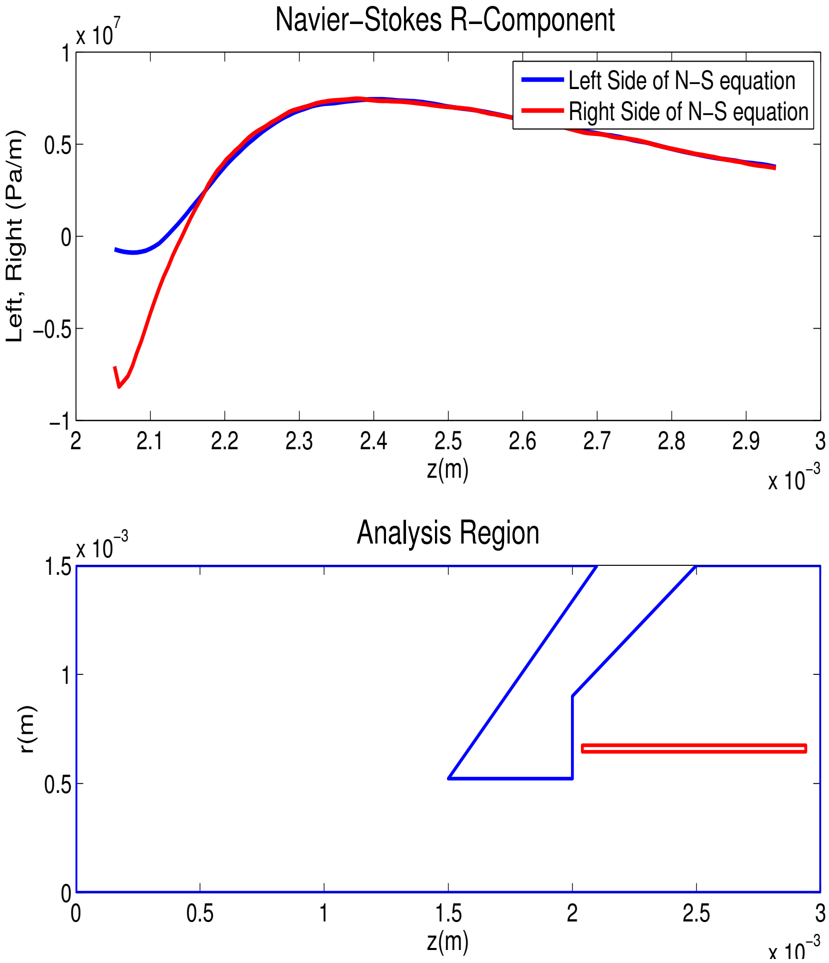

Figure 3.27 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.6 mm from the central axis of the simulation are compared.

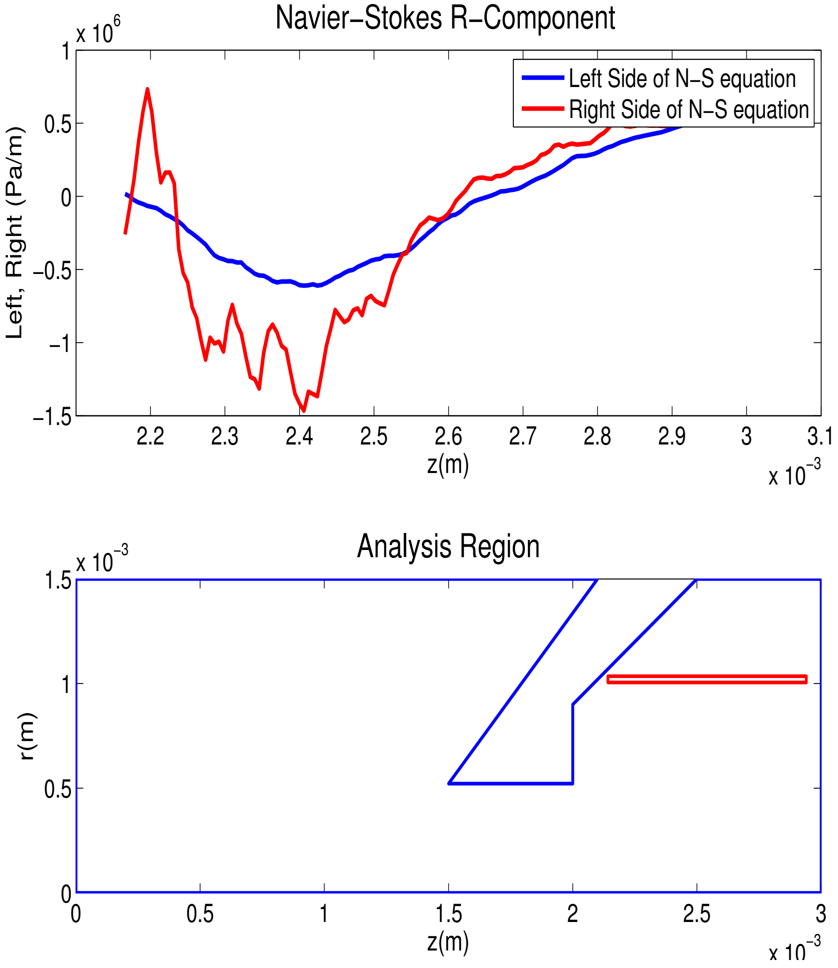

Figure 3.28 Axial scans of the left and right hand terms of the Navier-Stokes r-component 1.02 mm from the central axis of the simulation are compared.

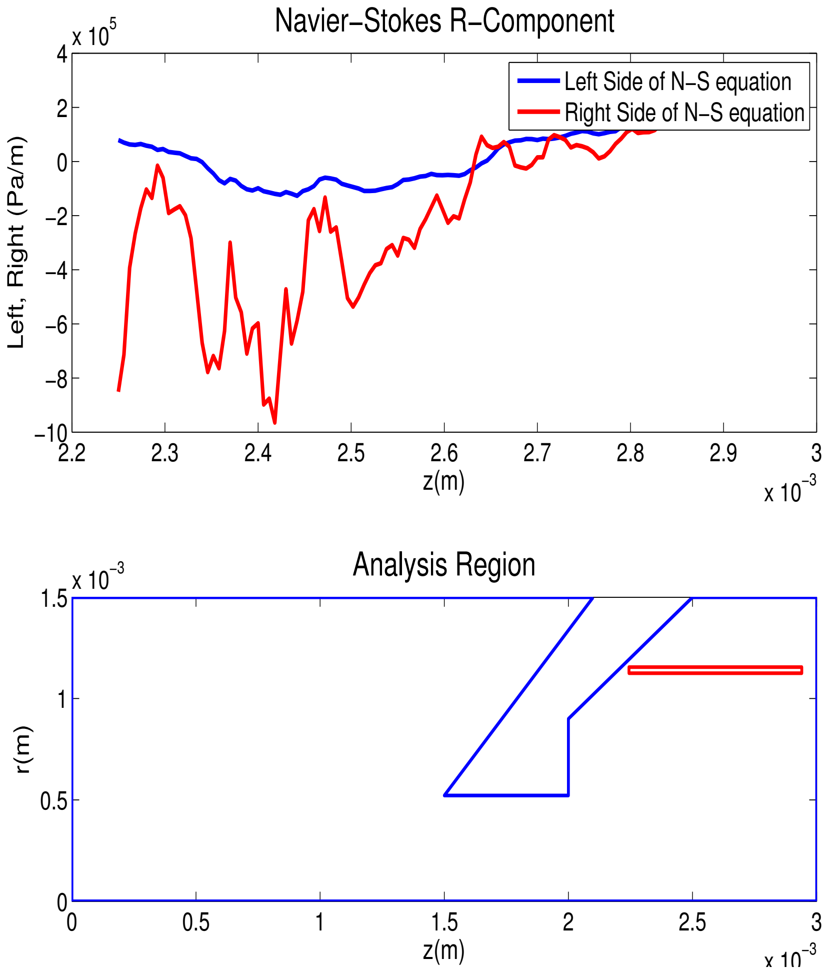

Figure 3.29 Axial scans of the left and right hand terms of the Navier-Stokes r-component 1.14 mm from the central axis of the simulation are compared.

The Navier-Stokes equation fails behind the sampler cone where r is large. In the expansion region, the fluid density decreases dramatically, and the mean free path increases. The Knudsen number was checked behind the nozzle to determine if the Navier-Stokes equation should be invalid. From Fig. 3.30, it is shown that Kn is greater than 0.1 in the region where the Navier-Stokes equation fails.

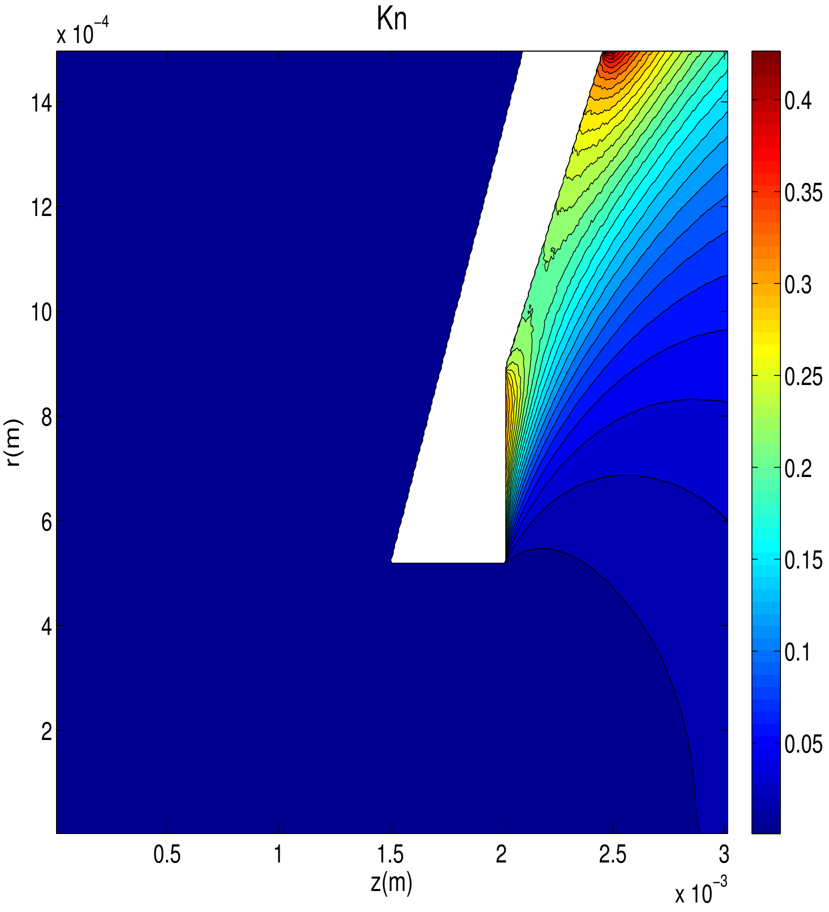

Figure 3.30 The Knudsen Number in the ICP-MS

3.2.3 Summarizing data from the r-component of the Navier-Stokes equation

This section verifies that FENIX correctly reproduces the Navier-Stokes fluid equations as long as Kn is small enough. In the downstream region behind the nozzle, particle density is very low and velocity is high (see Figs. 3.2,3.4. Particle collisions are rare, and the mean free path is long. These conditions give a high value of Kn (see Fig. 3.30), and FENIX has shown that the Navier-Stokes equations break down in this region downstream of the nozzle as expected. For the r-component of the Navier-Stokes equation, FENIX data shows statistical fluctuations are on the order of 2×10^6, whereas, the data sets are generally on the order of 10^8. Typical error is around 2%, but there are two areas in the simulation where the error is as high as 20%. One is near the nozzle tips where the data analysis method has trouble computing second derivatives where the gradients are large and the second is near the nozzle surface in the downstream region where the density is so low that the Knudsen number is unacceptably large.

3.2.4 The z-Component of the Navier-Stokes equation

The Z-component of the momentum equation is given as follows:

$$
\begin{aligned}
\rho\,(v_x\,\frac{\partial v_z}{\partial r} + v_z\,\frac{\partial v_z}{\partial z}) = \\
\mu\,(\frac{\partial {v_x}^2}{\partial r^2} + \frac{4}{3}\,\frac{\partial^2 v_z}{\partial z^2}) + v_z(\frac{\partial^2 \mu}{\partial r^2} + \frac{4}{3}\,\frac{\partial^2 \mu}{\partial z^2}) + \frac{1}{3}\,(v_x\,\frac{\partial^2 \mu}{\partial x \partial y} \\
+ \mu\,\frac{\partial^2 v_x}{\partial x \partial y}) + 2\,\frac{\partial \mu}{\partial x}\,\frac{\partial v_z}{\partial r} + \frac{8}{3}\,\frac{\partial \mu}{\partial z}\,\frac{\partial v_z}{\partial z} + \frac{1}{3}\,(\frac{\partial \mu}{\partial r}\,\frac{\partial v_x}{\partial z} \\
+ \frac{\partial v_x}{\partial r}\,\frac{\partial \mu}{\partial z}) + \frac{1}{r_0}\,(v_z\,\frac{\partial \mu}{\partial r} + \mu\,\frac{\partial v_z}{\partial r} + \frac{1}{3}\,(v_x\,\frac{\partial \mu}{\partial z} + \mu\,\frac{\partial v_x}{\partial z}))
\end{aligned}
\tag{3.5}
$$

The z- component of the momentum equation has also been fully analyzed throughout the ICP-MS simulation region, as was the r component. Data has been reviewed in the upstream, nozzle, and downstream regions, in both axial and radial analysis arrays. Because the r- component has been presented in complete detail, with particular notes made on any anomalies or discrepancies, it is not necessary to present the z component analysis in full. The z component of the fluid equation has been found to be valid in the same regimes as the r component. The high Kn region behind the sampler cone causes the Navier-Stokes equations to fail. Also, the fitting method fails due to large velocity gradients at the same locations throughout the simulation for both components and the relative errors follow the same pattern as in the radial scans.

The two most significant scans of the z-component are the main flow down the center of the nozzle, and the flow through the boundary layer.

The full flow through the nozzle is seen in Fig. 3.31.

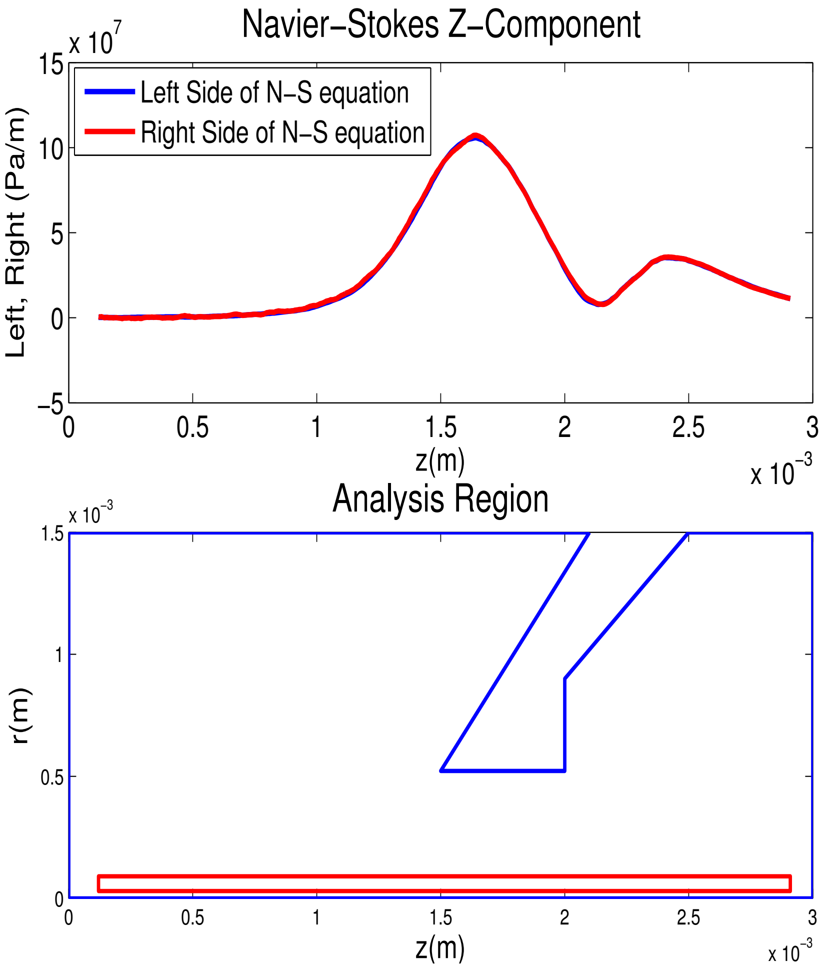

Figure 3.31 Axial scans of the left and right hand terms of the Navier-Stokes z-component 0.06 mm from the central axis of the simulation are compared.

The z component in the nozzle boundary layer is seen in Fig. 3.32.

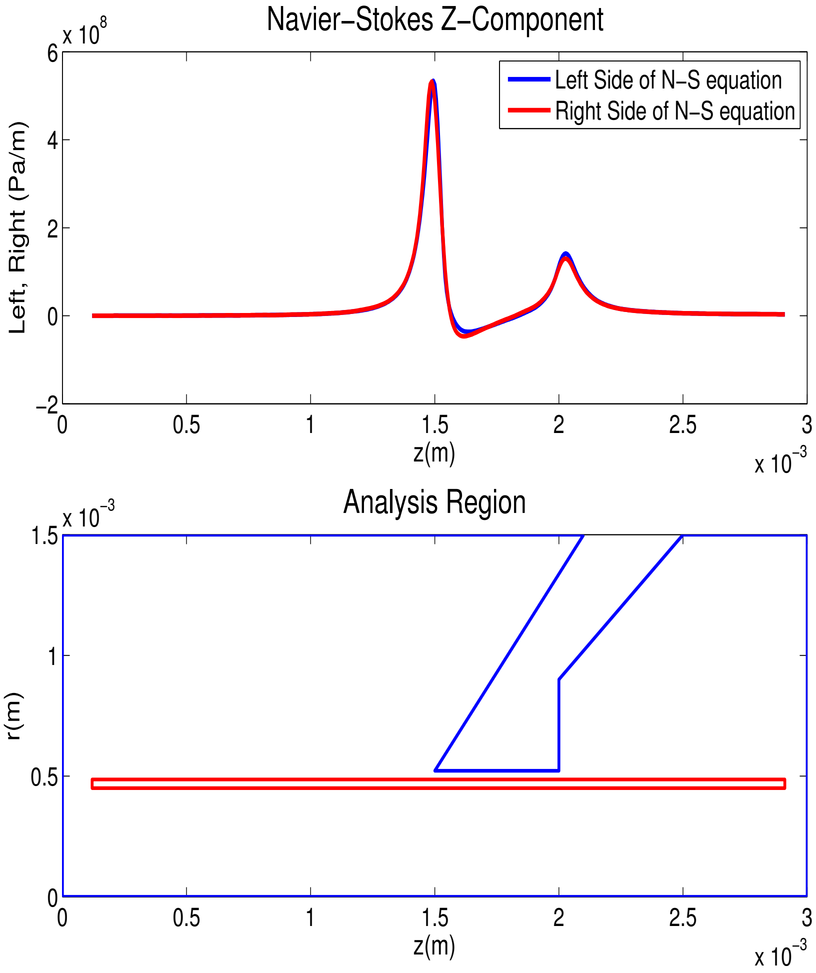

Figure 3.32 Axial scans of the left and right hand terms of the Navier-Stokes r-component 0.468 mm from the central axis of the simulation are compared.

3.3 Boundary Layers

The momentum boundary layer is often called the viscous layer. The layer develops as the particles exchange momentum with the fixed sampler cone, and then re-enter the main flow. These particles now have decreased momentum, and particles colliding with them also exchange momentum. The decrease in particle momentum is smaller this time, and smaller yet for each subsequent collision. This effect causes a layer of decreased momentum in the flow, but its effects are only noticed by the particles in the main flow out to a distance known as the boundary layer width, δ. Particles beyond the boundary layer do not lose momentum, and are unaware of the existence of the nozzle. This phenomenon can be seen in Fig. 3.4. The boundary layer width δ is defined by the point where the stream velocity within the boundary layer reaches 99% of the mainstream fluid velocity [21]:

$$\delta = 5\sqrt{\frac{\mu\, L}{\rho\, v_s}} \tag{3.6}$$

The 99% condition is the reason for the factor of 5.

In the ICP-MS, viscosity is not a constant, but a function strongly dependent on temperature. There is a strong temperature gradient in the boundary layer, meaning that viscosity changes rapidly, and an exact value for boundary layer width is not analytically available. An approximate value can be calculated by using average density and temperature values in the boundary layer, and an average stream velocity on the axis of symmetry. The actual values used are as follows:

$\mu = 1.1 \times 10^{-4}$ kg/m/s

$\rho = 0.05$ kg/m^3

$v_s = 1200$ m/s

$L = 5 \times 10^{-4}$ m

(L is axial length along the sampler cone in the direction of the flow.) This calculation results in a boundary layer thickness of $delta = 0.15$ mm.

The boundary layer width can be estimated from the simulation data by determining where the boundary layer fluid velocity reaches 99% of the mainstream velocity. The mainstream reference velocity v_s is not constant, but can be reasonably determined as an average z velocity in the middle of the nozzle. This calculation, using a mainstream velocity of 1200 m/s shows the boundary layer stream velocity reaches 99% of the mainstream velocity about 0.165 mm from the cone surface, which is reasonably close to the calculation from Eq. 3.6.

One other interesting aspect of the boundary layer phenomena in FENIX is the boundary layer's shape. A typical boundary layer formed by flow past a flat plate will grow indefinitely in width, but the boundary layer inside the nozzle does not do this. This is an effect caused by the acceleration of the gas through the nozzle. Because the stream velocity appears in the denominator of Eq. (3.6) the boundary layer becomes thinner as the velocity increases, producing a symmetric boundary layer that is thin both at the entrance and at the exit. Producing boundary layer phenomenon through direct simulation is an important accomplishment. By this means, FENIX is shown to simulate molecule interactions with thermalized surfaces, and to process momentum transfer from particles to surfaces as well as particle-particle.

3.4 The Heat Equation

In Chapter 2, the heat equation was shown to be:

$$\begin{aligned}\frac{\partial T}{\partial t} + v \cdot \nabla T + (\gamma - 1)\, T\ \nabla \cdot v =& (\gamma - 1)\frac{1}{nk_b}\nabla \cdot (k \nabla T) \\ & + \frac{(\gamma - 1)}{nk_b}\, d_{i\,j}\, \frac{\partial u_i}{\partial x_j}\end{aligned} \tag{3.7}$$

The heat transport equation has produced the most interesting results of any of the topics presented in this work. While FENIX has been shown to obey properly the momentum equations in all regions where they are predicted to be valid, the heat equation is not so well behaved in our simulation.

3.4.1 The Upstream Region

Radial analysis arrays show in Fig. 3.33 that noise dominates the upstream region. Note that if we use for a comparison value 4×10^9 (see Fig.3.38) then the error in this case is less than 1%.

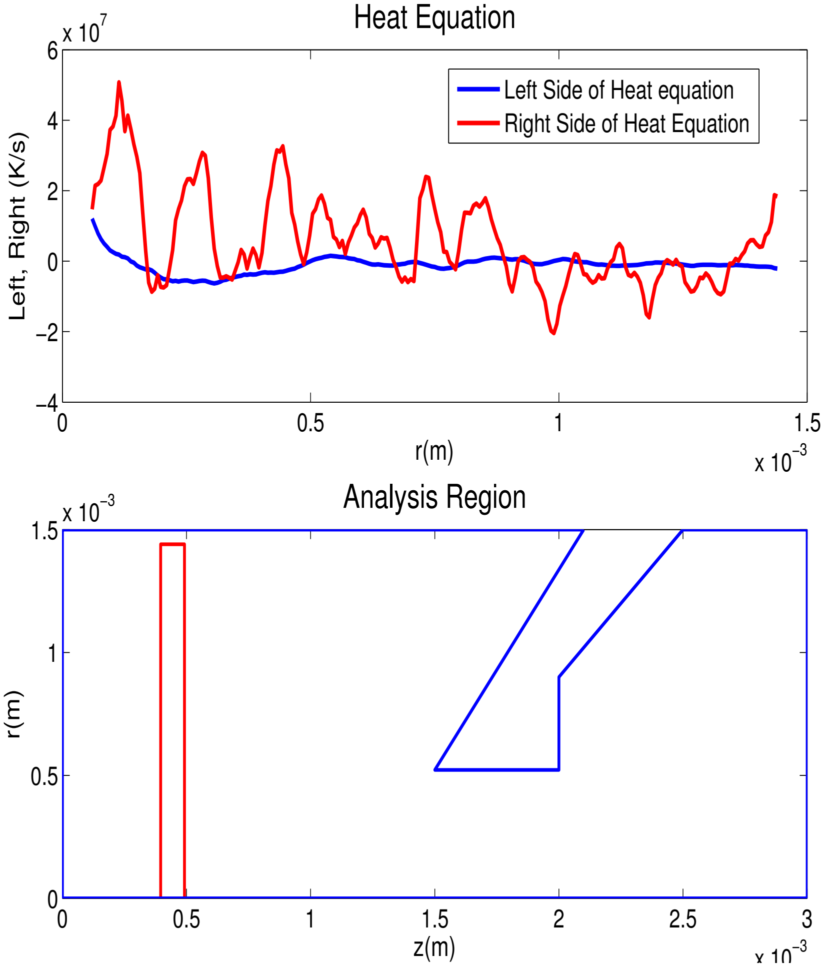

Figure 3.33 Radial scans of the left and right hand terms of the heat equation 0.44 mm downstream from the front end of the simulation are compared.

At 1.284 mm downstream the energy terms become larger, and the analysis begins to improve, as shown in Fig3.34, giving an error of about 1% again.

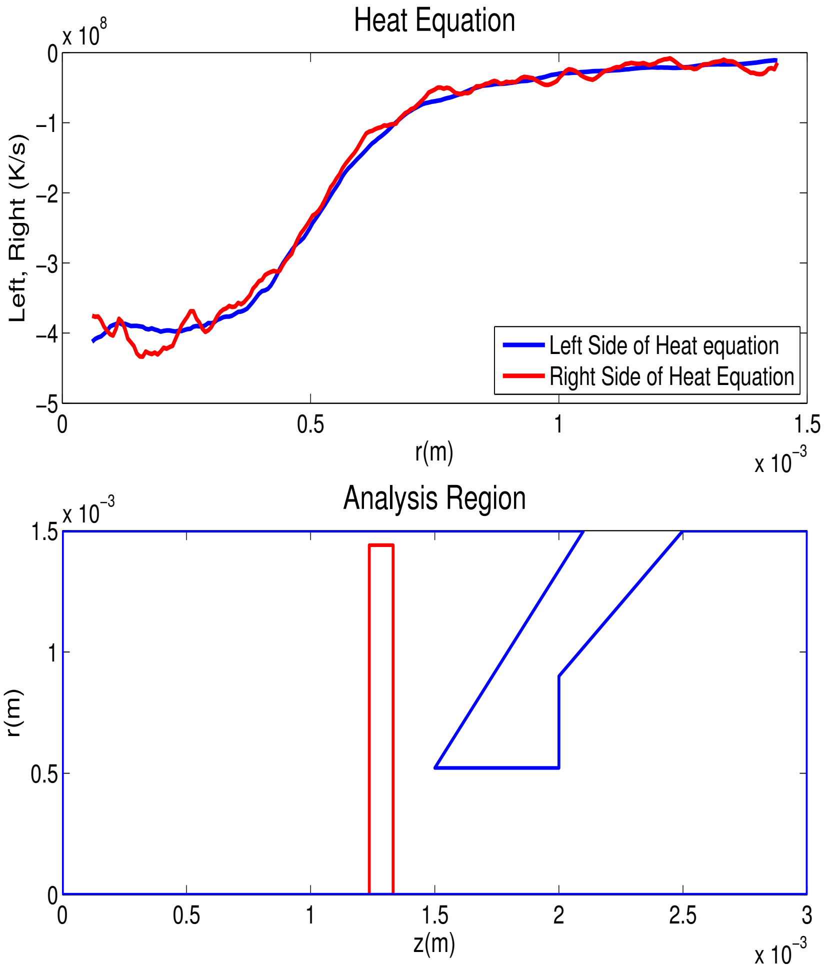

Figure 3.34 Radial scans of the left and right hand terms of the heat equation 1.28 mm downstream from the front end of the simulation are compared.

At the front tip of the sampler cone, the least squares fitting method had trouble with the second derivatives in the Navier-Stokes equation because of the high velocity gradients. The heat equation also has second derivatives; it is seen in Fig. 3.35 that the error in the heat equation terms reaches about 10%.

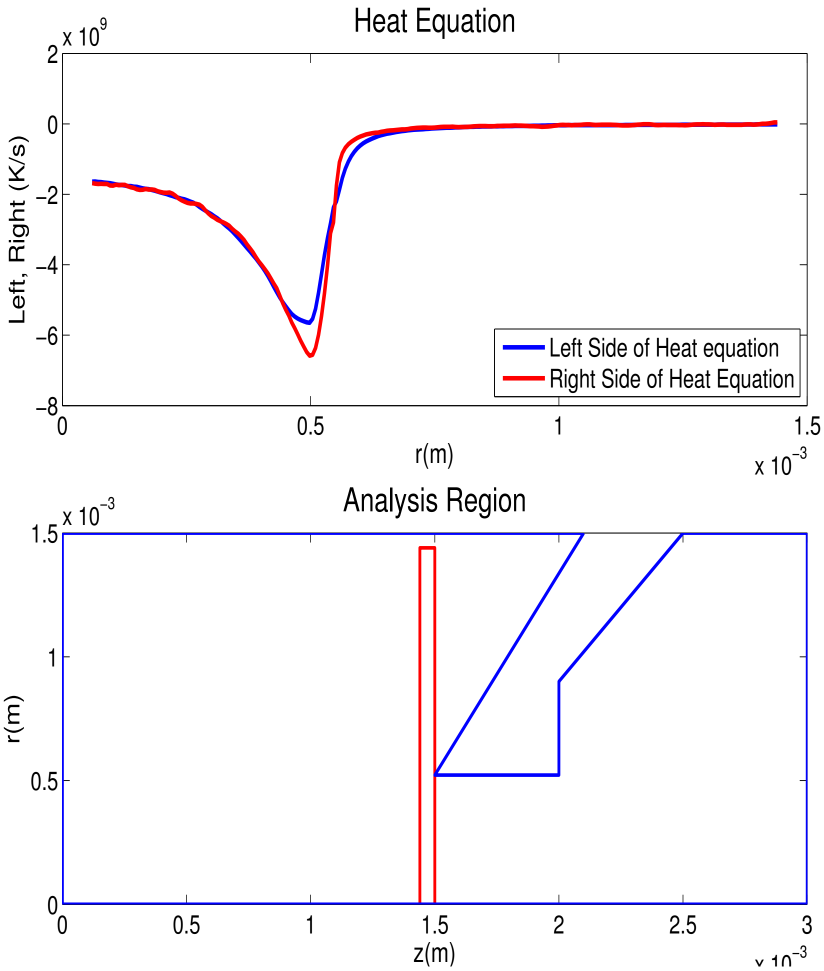

Figure 3.35 Radial scans of the left and right hand terms of the heat equation 1.47 mm downstream from the front end of the simulation are compared.

Axial scans in Figs. 3.36 and 3.37 also show that the heat equation is obeyed in the upstream region, although in areas where the radial distance from the axis is greater than about 0.6 mm, statistical noise dominates because velocity gradients are very small.

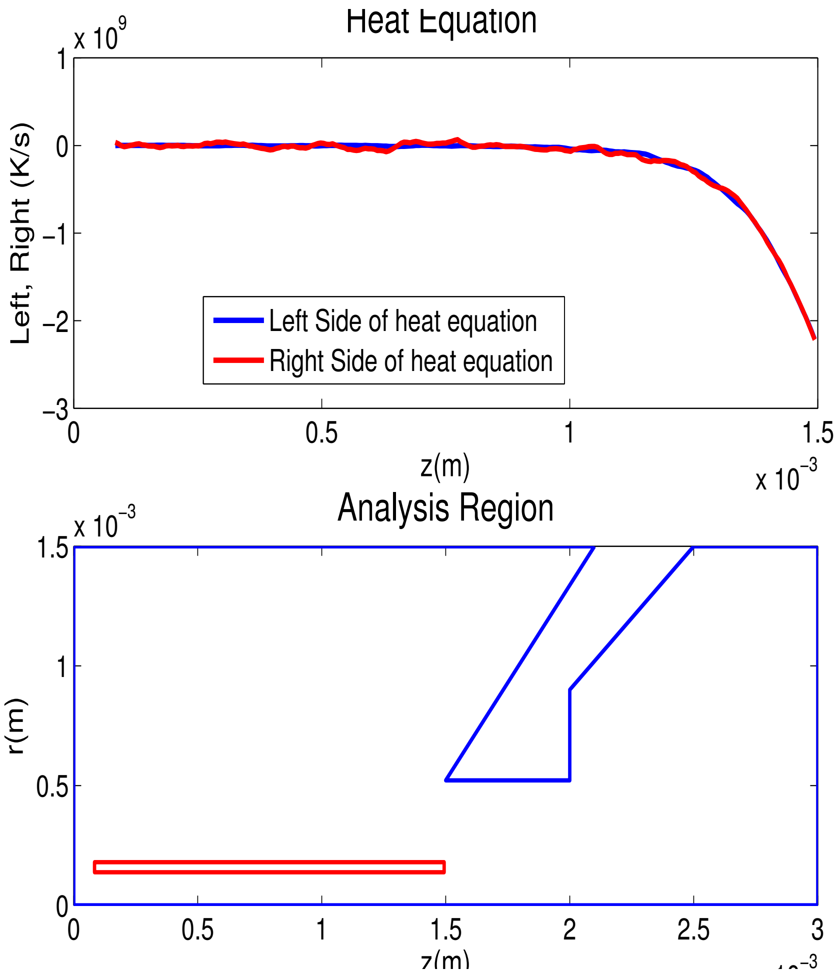

Figure 3.36 Axial scans of the left and right hand terms of the heat equation 0.18 mm from the central axis of the simulation are compared.

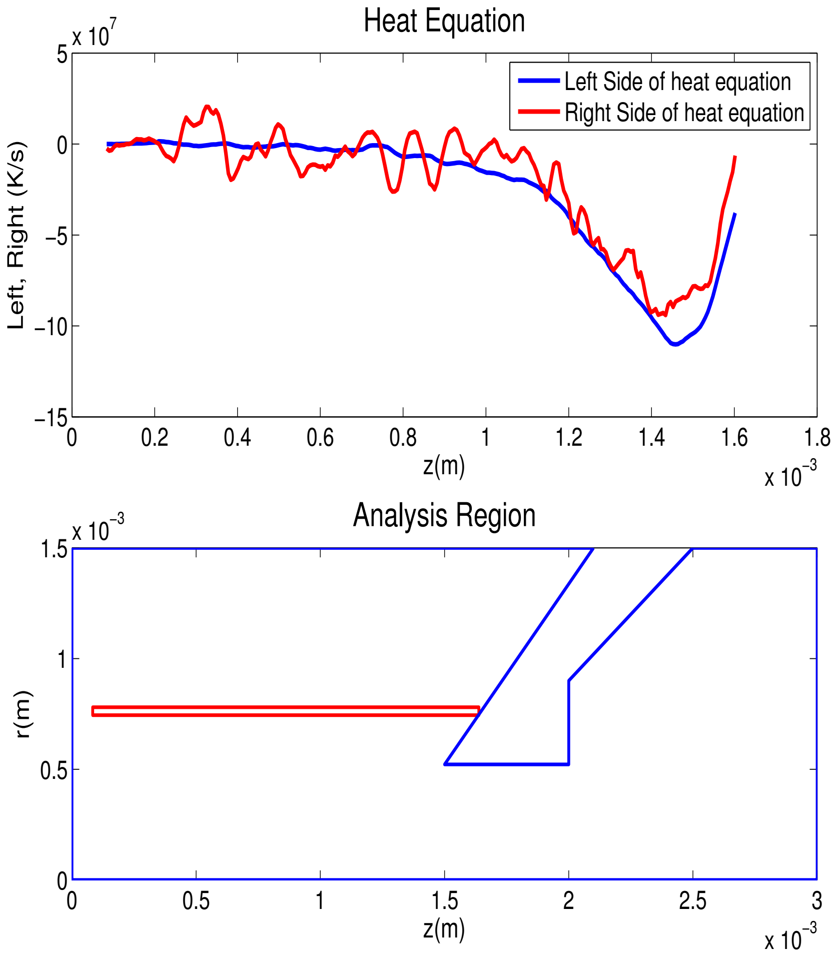

Figure 3.37 Axial scans of the left and right hand terms of the heat equation 0.78 mm from the central axis of the simulation are compared.

3.4.2 The Nozzle Region

Figs. 3.38 and 3.39 show that the heat equation is well behaved in the center of the nozzle region, until the analysis reaches the thermal boundary layer where the relative error is 10%.

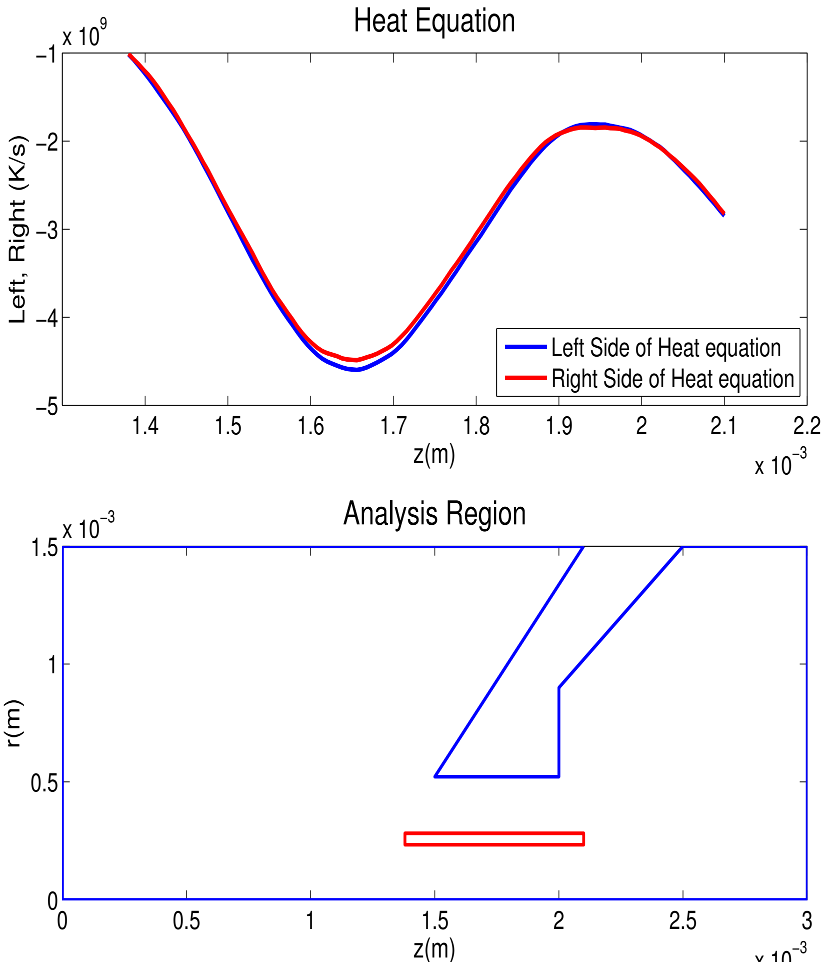

Figure 3.38 Axial scans of the left and right hand terms of the heat equation 0.258 mm from the central axis of the simulation are compared.

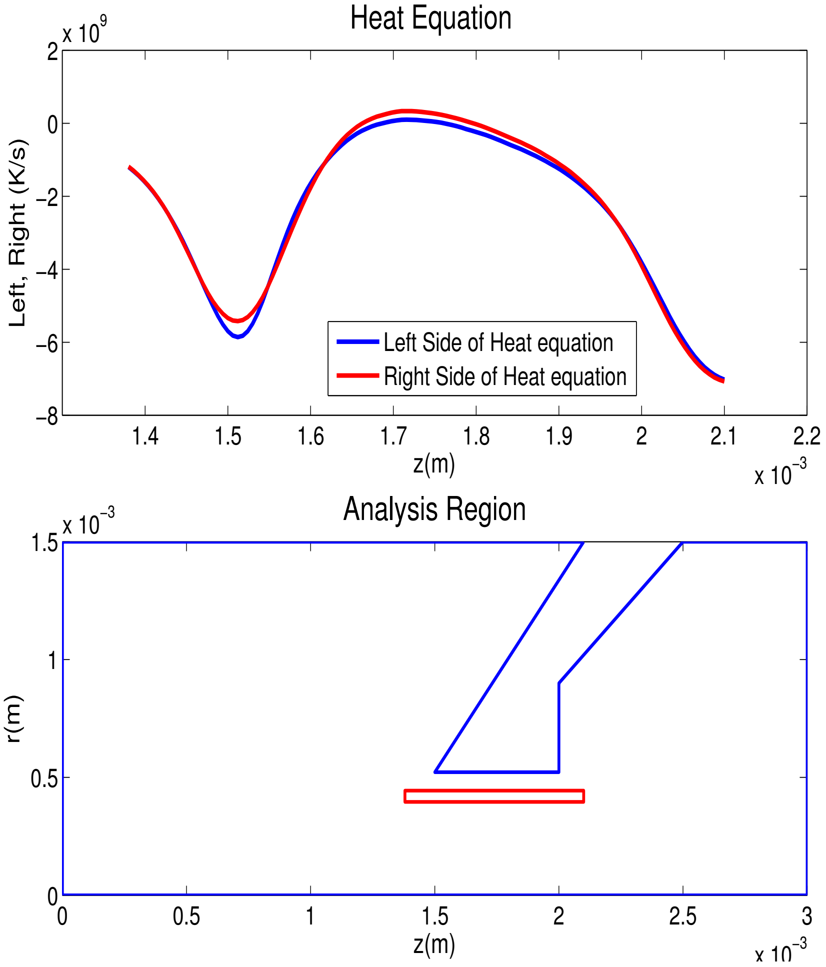

Figure 3.39 Axial scans of the left and right hand terms of the heat equation 0.242 mm from the central axis of the simulation are compared.

In the thermal boundary layer close to the nozzle surface, the heat equation performs somewhat poorly, especially at the front tip of the sampler cone where the maximum relative error reaches 20%, as shown in Fig. 3.40.

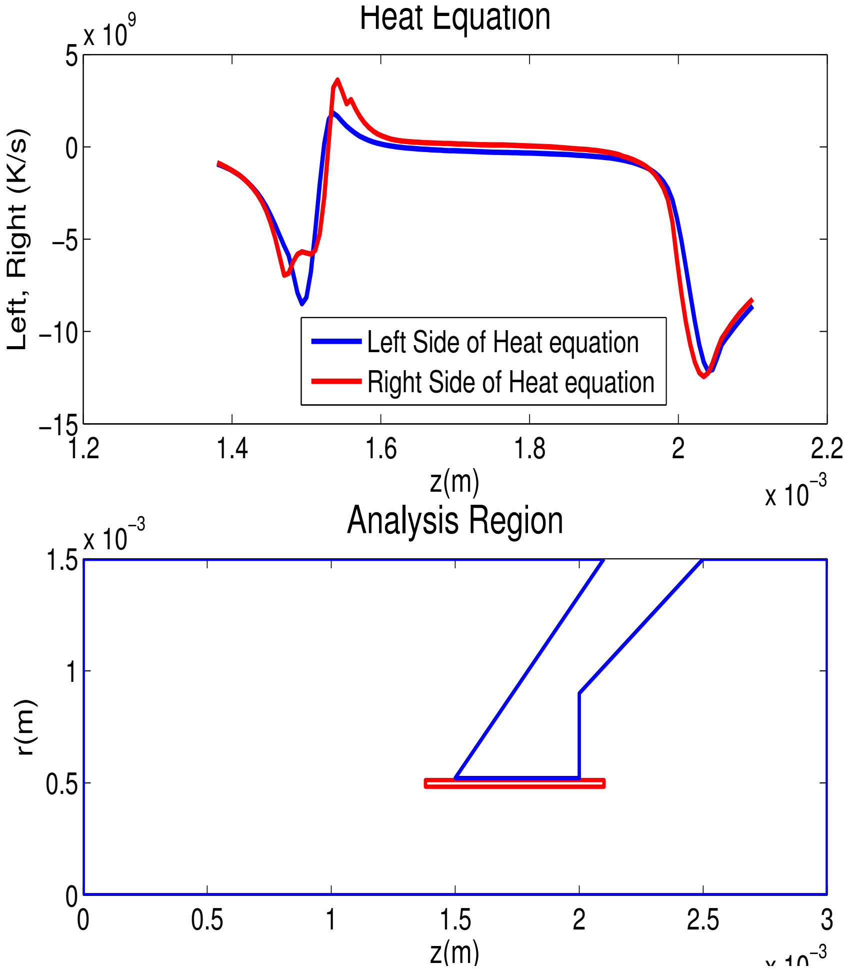

Figure 3.40 Axial scans of the left and right hand terms of the heat equation 0.498 mm from the central axis of the simulation are compared.

This effect is also easily seen in radial analysis arrays. It is observed in Fig. 3.41 that the heat equation is not correct where the analysis cells enter the thermal boundary layer. The last 5-10 grid points do not give accurate data, because the data smoothing function is inadequate near the cone's surface. The least squares approximation grid is cell-centered, and does not perform well on the edge of a boundary. But there is a 10% deviation in the energy transport terms as far as 20 grid points away from the nozzle.

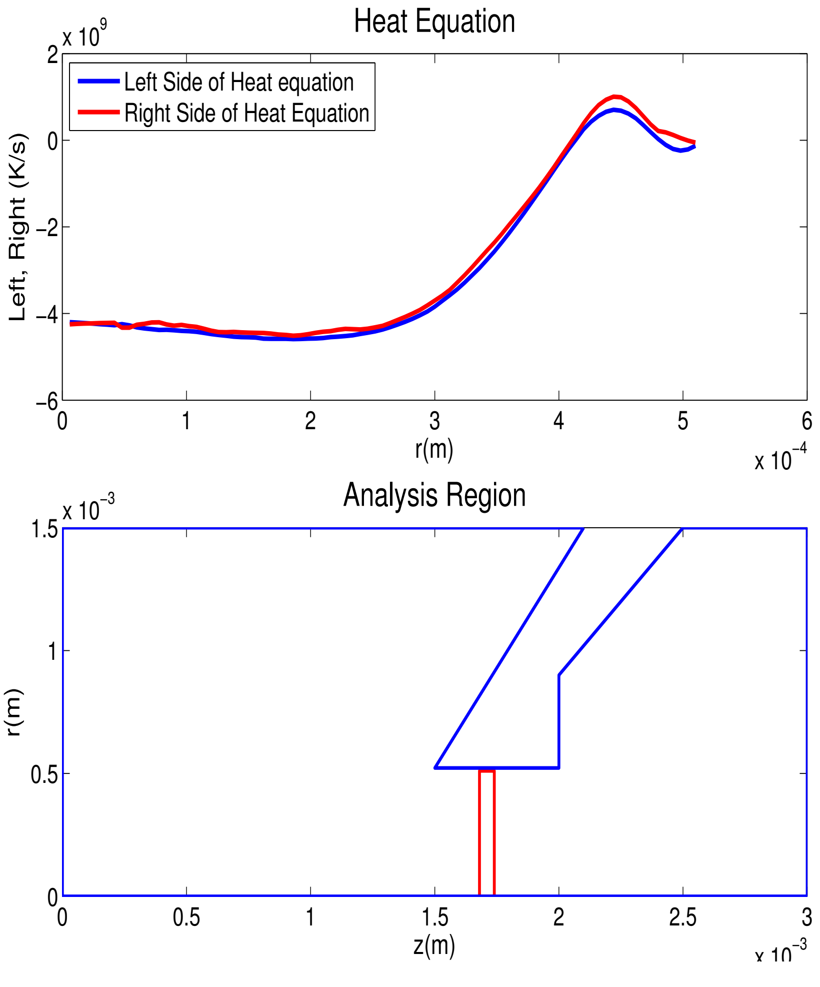

Figure 3.41 Radial scans of the left and right hand terms of the heat equation 1.71 mm downstream from the front end of the simulation are compared.

3.4.3 The Downstream Region

The downstream region has the same trouble as the other areas. The heat equation is not obeyed near the surface of the nozzle.

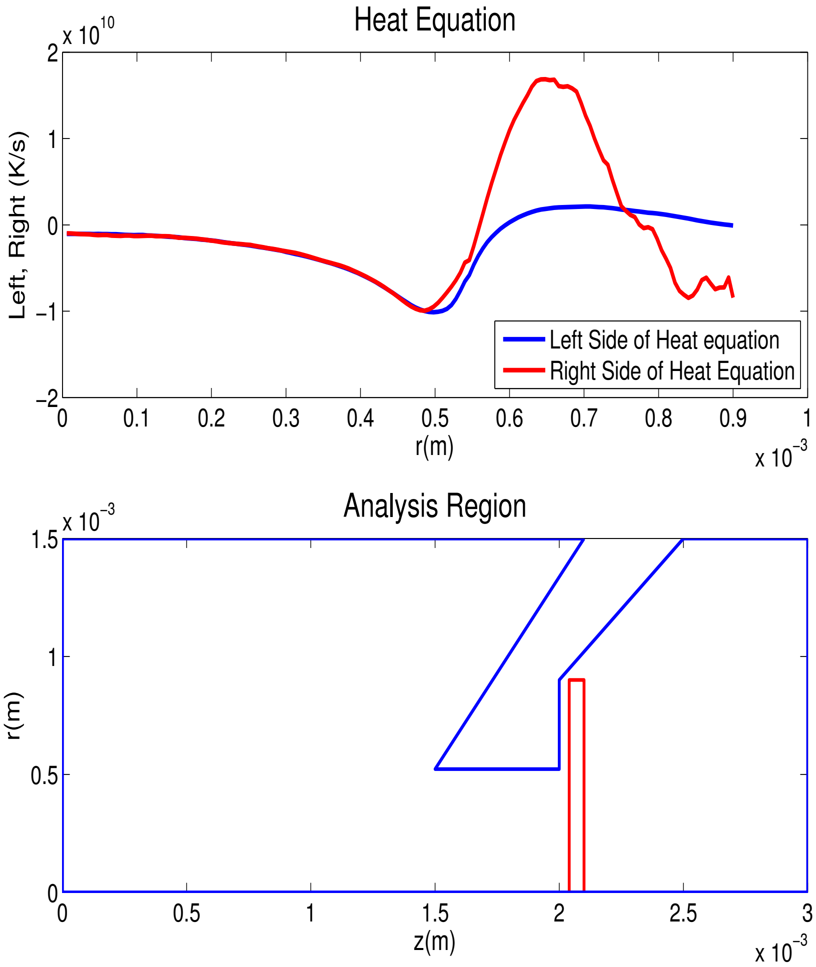

Figure 3.42 Radial scans of the left and right hand terms of the heat equation 2.07 mm downstream from the front end of the simulation are compared.

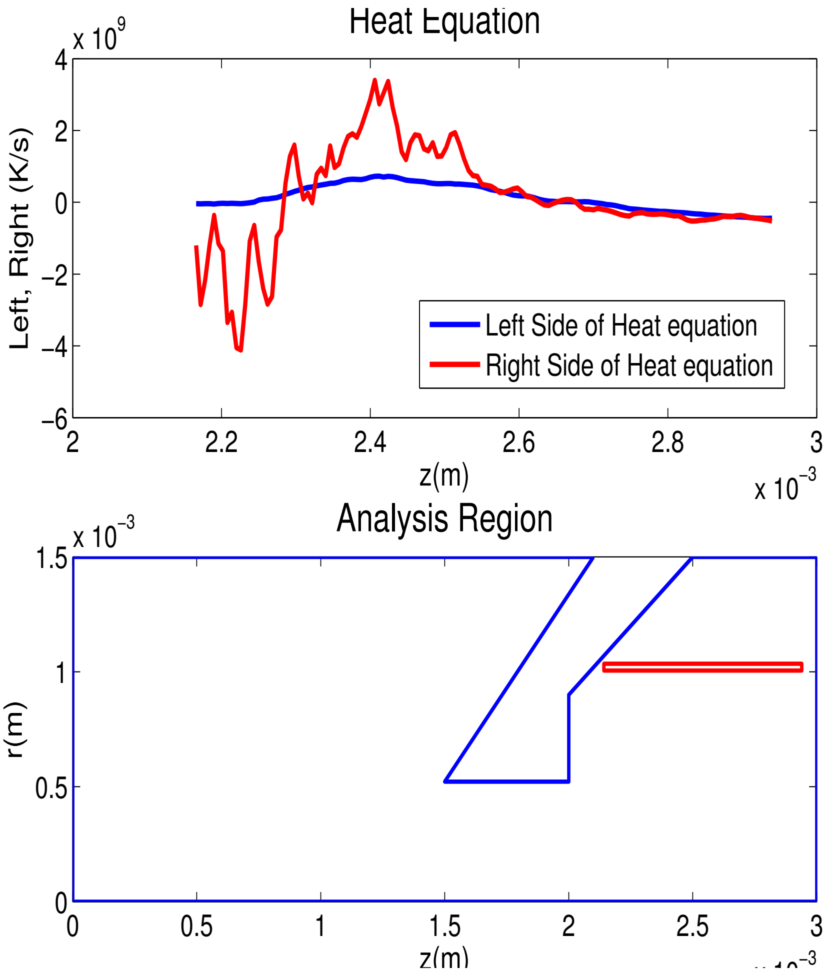

Figure 3.43 Axial scans of the left and right hand terms of the heat equation 1.02 mm from the central axis of the simulation are compared.

Figs. 3.26-3.29 show that the Navier-Stokes equation is not obeyed in the region behind the sampler cone, and Figs. 3.42 and 3.43 show that the heat equation is not followed either. Fig. 3.30 indicates a high value of Kn in this area, and the Navier-Stokes equation is expected to fail here. The heat equation shows sensitivity to high Kn values as well.

Simulation testing in the nozzle orifice shows that the heat equation performs poorly in the thermal boundary layer. It is anticipated that the analysis will perform poorly when applied within 7 grid points from the sampler cone, but the heat equation deviates considerably in areas further away from the nozzle (see Fig. 3.39). Further research will be needed to clarify the results of analyzing the heat equation.

Chapter 4

Conclusion

4.1 The Momentum Equations

A comparison has been made between the results of FENIX, a Direct-Simulation Monte Carlo code and the fluid equations. It has been shown that FENIX correctly reproduces the Navier-Stokes fluid equations when the Knudsen number is small enough, and is able to show where the Navier-Stokes equation breaks down at high values of Kn. The Navier-Stokes equation fails directly behind the sampler cone where particle density is very low. In this area, adherence to the fluid equations is not expected; the Knudsen number exceeds the continuum limit.

Analysis of FENIX data near the front tip of the nozzle indicated that the terms in the fluid equations did not agree. In this area, the least squares fitting method cannot adequately perform second derivatives due to the high velocity gradients present from the quickly changing direction of the streamlines. The sampling cells in the simulation were also not small enough to properly resolve these rapid changes. FENIX does not give clean data in the far upstream region either, because both sides of the Navier-Stokes equation are very small and dominated by statistical noise. DSMC methods perform poorly where particle velocities are very low. Downstream, the flow is well defined and FENIX is shown to properly reproduce the fluid equations.

4.2 The Heat Equation

FENIX properly obeys the heat equation in some areas of the ICP-MS simulation. The heat equation is not followed in the thermal boundary layer of the sampler nozzle (see Fig. 3.39, or in the downstream region directly behind the nozzle (see Fig. 3.43. Further research is necessary to establish a full understanding of why the heat equation fails in these parts of our simulation. In the downstream region behind the nozzle, Kn is high and it is probable that the heat equation is sensitive to Kn.

4.3 Boundary Layer Theory and Bernoulli's Equation

Boundary layer phenomena are produced by FENIX, and have been shown to be consistent with expected results. The momentum and thermal boundary layers have been reviewed, and were found to follow well the existing models. While there is no established method to determine an exact boundary layer width when the viscosity and mainstream velocity vary, using average values gives order of magnitude agreement between FENIX results and an analytic expression. The data from FENIX also obey Bernoulli's law in regions where transport is not important. This investigation also gave insight into the thermal boundary layer width, as the fluid is no longer adiabatic once it is aware of the boundary layer.

4.4 FENIX

The FENIX simulation of fluid flow in the inductively coupled plasma mass spectrometer has been tested against the Navier-Stokes and heat equations. Term by term comparison in the Navier-Stokes and heat equations shows that FENIX is able to correctly reproduce fluid dynamics to within about 2% except when steep gradients are encountered.

Appendix A

Post-Processing Matlab Code

```
% fenix fluid analysis post processor
% to be used after fenixpix has loaded the data arrays

close all;

clear btemp bpress bdens bmach bvelx bvely bvelz bvmag bmu



    j0=294;

    i0=1;

    block=10;

    jmin=j0-block;
    jmax=j0+block;
    imin=i0-block;
    imax=i0+block;



    counteri=i0;
    counterj=j0;
    %mu=1.3E-4;
for m=1:87;


    % % a=input(’ Enter the center analysis point, [j,i] (z,r) - ’);
    % %
    % % j0=a(1);i0=a(2);
    % %
    % % a=input(’ Enter [jmin,jmax] - ’);
    % % jmin=a(1);jmax=a(2);
    % %
    % % a=input(’ Enter [imin,imax] - ’);
    % % imin=a(1);imax=a(2);
    %this j0 - value chooses to look in the middle of the  nozzle


    j0=294;
    jmin=j0-block;
    jmax=j0+block;
    imin=i0-block;
    % if imin is less than 1, change it to 1
    imin=max(1,imin);
    imax=i0+block;
    % if imax is greater than 87, change it to 87
    imax=min(imax,87);

    % define the center of the small analysis region in its own
    % units

    jmid=block+1;
    imid=i0-imin+1;



% choose the physical quantity from fenixpix to fit to
    f=TEMP;
    mu=3.76e-7*TEMP.^.72;
    g=PRES;
    h=DENS;
    s=MACH;
    velx=VX;
    vely=VY;
    velz=VZ;
    vmag=VMAG;


    % compute the sums
    S00=0.;S01=0.;S02=0.;S10=0.;S20=0.;S11=0.;
    S10=0.;S11=0.;S12=0.;S20=0.;S30=0.;S21=0.;
    S01=0.;S02=0.;S03=0.;S11=0.;S21=0.;S12=0.;
    S20=0.;S21=0.;S22=0.;S30=0.;S40=0.;S31=0.;
    S02=0.;S03=0.;S04=0.;S12=0.;S22=0.;S13=0.;
    S11=0.;S12=0.;S13=0.;S21=0.;S31=0.;S22=0.;
    D00temp=0.;D10temp=0.;D01temp=0.;D20temp=0.;D02temp=0.;D11temp=0.;
    D00mu=0.;D10mu=0.;D01mu=0.;D20mu=0.;D02mu=0.;D11mu=0.;
    D00press=0.;D10press=0.;D01press=0.;D20press=0.;D02press=0.;D11press=0.;
    D00dens=0.;D10dens=0.;D01dens=0.;D20dens=0.;D02dens=0.;D11dens=0.;
    D00mach=0.;D10mach=0.;D01mach=0.;D20mach=0.;D02mach=0.;D11mach=0.;
    D00velx=0.;D10velx=0.;D01velx=0.;D20velx=0.;D02velx=0.;D11velx=0.;
    D00vely=0.;D10vely=0.;D01vely=0.;D20vely=0.;D02vely=0.;D11vely=0.;
    D00velz=0.;D10velz=0.;D01velz=0.;D20velz=0.;D02velz=0.;D11velz=0.;
    D00vmag=0.;D10vmag=0.;D01vmag=0.;D20vmag=0.;D02vmag=0.;D11vmag=0.;




    zeta=Z-Z(i0,j0);zeta=zeta/dzblk;
    rho=R-R(i0,j0);rho=rho/drblk;

    for j=jmin:jmax
        for i=imin:imax

            S00=S00+1.;
            S10=S10+zeta(i,j);
            S01=S01+rho(i,j);
            S20=S20+zeta(i,j)^2;
            S02=S02+rho(i,j)^2;
            S11=S11+zeta(i,j)*rho(i,j);
```

```
S30=S30+zeta(i,j)^3;
S12=S12+zeta(i,j)*rho(i,j)^2;
S21=S21+zeta(i,j)^2*rho(i,j);
S03=S03+rho(i,j)^3;
S40=S40+zeta(i,j)^4;
S31=S31+zeta(i,j)^3*rho(i,j);
S22=S22+zeta(i,j)^2*rho(i,j)^2;
S13=S13+zeta(i,j)*rho(i,j)^3;
S04=S04+rho(i,j)^4;

D00temp=D00temp+f(i,j);
D10temp=D10temp+f(i,j)*zeta(i,j);
D01temp=D01temp+f(i,j)*rho(i,j);
D20temp=D20temp+f(i,j)*zeta(i,j)^2;
D11temp=D11temp+f(i,j)*zeta(i,j)*rho(i,j);
D02temp=D02temp+f(i,j)*rho(i,j)^2;


D00mu=D00mu+mu(i,j);
D10mu=D10mu+mu(i,j)*zeta(i,j);
D01mu=D01mu+mu(i,j)*rho(i,j);
D20mu=D20mu+mu(i,j)*zeta(i,j)^2;
D11mu=D11mu+mu(i,j)*zeta(i,j)*rho(i,j);
D02mu=D02mu+mu(i,j)*rho(i,j)^2;

D00press=D00press+g(i,j);
D10press=D10press+g(i,j)*zeta(i,j);
D01press=D01press+g(i,j)*rho(i,j);
D20press=D20press+g(i,j)*zeta(i,j)^2;
D11press=D11press+g(i,j)*zeta(i,j)*rho(i,j);
D02press=D02press+g(i,j)*rho(i,j)^2;

D00dens=D00dens+h(i,j);
D10dens=D10dens+h(i,j)*zeta(i,j);
D01dens=D01dens+h(i,j)*rho(i,j);
D20dens=D20dens+h(i,j)*zeta(i,j)^2;
D11dens=D11dens+h(i,j)*zeta(i,j)*rho(i,j);
D02dens=D02dens+h(i,j)*rho(i,j)^2;

D00mach=D00mach+s(i,j);
D10mach=D10mach+s(i,j)*zeta(i,j);
D01mach=D01mach+s(i,j)*rho(i,j);
D20mach=D20mach+s(i,j)*zeta(i,j)^2;
D11mach=D11mach+s(i,j)*zeta(i,j)*rho(i,j);
D02mach=D02mach+s(i,j)*rho(i,j)^2;


D00velx=D00velx+velx(i,j);
D10velx=D10velx+velx(i,j)*zeta(i,j);
D01velx=D01velx+velx(i,j)*rho(i,j);
D20velx=D20velx+velx(i,j)*zeta(i,j)^2;
D11velx=D11velx+velx(i,j)*zeta(i,j)*rho(i,j);
D02velx=D02velx+velx(i,j)*rho(i,j)^2;

D00vely=D00vely+vely(i,j);
D10vely=D10vely+vely(i,j)*zeta(i,j);
D01vely=D01vely+vely(i,j)*rho(i,j);
D20vely=D20vely+vely(i,j)*zeta(i,j)^2;
D11vely=D11vely+vely(i,j)*zeta(i,j)*rho(i,j);
D02vely=D02vely+vely(i,j)*rho(i,j)^2;


D00velz=D00velz+velz(i,j);
D10velz=D10velz+velz(i,j)*zeta(i,j);
D01velz=D01velz+velz(i,j)*rho(i,j);
```

```
        D20velz=D20velz+velz(i,j)*zeta(i,j)^2;
        D11velz=D11velz+velz(i,j)*zeta(i,j)*rho(i,j);
        D02velz=D02velz+velz(i,j)*rho(i,j)^2;

        D00vmag=D00vmag+vmag(i,j);
        D10vmag=D10vmag+vmag(i,j)*zeta(i,j);
        D01vmag=D01vmag+vmag(i,j)*rho(i,j);
        D20vmag=D20vmag+vmag(i,j)*zeta(i,j)^2;
        D11vmag=D11vmag+vmag(i,j)*zeta(i,j)*rho(i,j);
        D02vmag=D02vmag+vmag(i,j)*rho(i,j)^2;
    end
end

% load the matrix A and the right-hand side vector b

clear A;clear b;

  j=1;
Atemp(j,1)=S00;Atemp(j,2)=S10;Atemp(j,3)=S01;Atemp(j,4)=S20;
Atemp(j,5)=S02;Atemp(j,6)=S11;btemp(j)=D00temp;   j=2;
Atemp(j,1)=S10;Atemp(j,2)=S20;Atemp(j,3)=S11;Atemp(j,4)=S30;
Atemp(j,5)=S12;Atemp(j,6)=S21;btemp(j)=D10temp;j=3;
Atemp(j,1)=S01;Atemp(j,2)=S11;Atemp(j,3)=S02;Atemp(j,4)=S21;
Atemp(j,5)=S03;Atemp(j,6)=S12;btemp(j)=D01temp;   j=4;
Atemp(j,1)=S20;Atemp(j,2)=S30;Atemp(j,3)=S21;Atemp(j,4)=S40;
Atemp(j,5)=S22;Atemp(j,6)=S31;btemp(j)=D20temp;   j=5;
Atemp(j,1)=S02;Atemp(j,2)=S12;Atemp(j,3)=S03;Atemp(j,4)=S22;
Atemp(j,5)=S04;Atemp(j,6)=S13;btemp(j)=D02temp;   j=6;
Atemp(j,1)=S11;Atemp(j,2)=S21;Atemp(j,3)=S12;Atemp(j,4)=S31;
Atemp(j,5)=S13;Atemp(j,6)=S22;btemp(j)=D11temp;      j=1;
Amu(j,1)=S00;Amu(j,2)=S10;Amu(j,3)=S01;Amu(j,4)=S20;Amu(j,5)=S02;
Amu(j,6)=S11;bmu(j)=D00mu;   j=2;
Amu(j,1)=S10;Amu(j,2)=S20;Amu(j,3)=S11;Amu(j,4)=S30;Amu(j,5)=S12;
Amu(j,6)=S21;bmu(j)=D10mu;   j=3;
Amu(j,1)=S01;Amu(j,2)=S11;Amu(j,3)=S02;Amu(j,4)=S21;Amu(j,5)=S03;
Amu(j,6)=S12;bmu(j)=D01mu;   j=4;
Amu(j,1)=S20;Amu(j,2)=S30;Amu(j,3)=S21;Amu(j,4)=S40;Amu(j,5)=S22;
Amu(j,6)=S31;bmu(j)=D20mu;   j=5;
Amu(j,1)=S02;Amu(j,2)=S12;Amu(j,3)=S03;Amu(j,4)=S22;Amu(j,5)=S04;
Amu(j,6)=S13;bmu(j)=D02mu;   j=6;
Amu(j,1)=S11;Amu(j,2)=S21;Amu(j,3)=S12;Amu(j,4)=S31;Amu(j,5)=S13;
Amu(j,6)=S22;bmu(j)=D11mu;    j=1;
Apress(j,1)=S00;Apress(j,2)=S10;Apress(j,3)=S01;Apress(j,4)=S20;
Apress(j,5)=S02;Apress(j,6)=S11;bpress(j)=D00press;   j=2;
Apress(j,1)=S10;Apress(j,2)=S20;Apress(j,3)=S11;Apress(j,4)=S30;
Apress(j,5)=S12;Apress(j,6)=S21;bpress(j)=D10press;   j=3;
Apress(j,1)=S01;Apress(j,2)=S11;Apress(j,3)=S02;Apress(j,4)=S21;
Apress(j,5)=S03;Apress(j,6)=S12;bpress(j)=D01press;   j=4;
Apress(j,1)=S20;Apress(j,2)=S30;Apress(j,3)=S21;Apress(j,4)=S40;
Apress(j,5)=S22;Apress(j,6)=S31;bpress(j)=D20press;   j=5;
Apress(j,1)=S02;Apress(j,2)=S12;Apress(j,3)=S03;Apress(j,4)=S22;
Apress(j,5)=S04;Apress(j,6)=S13;bpress(j)=D02press;   j=6;
Apress(j,1)=S11;Apress(j,2)=S21;Apress(j,3)=S12;Apress(j,4)=S31;
Apress(j,5)=S13;Apress(j,6)=S22;bpress(j)=D11press;    j=1;
Adens(j,1)=S00;Adens(j,2)=S10;Adens(j,3)=S01;Adens(j,4)=S20;
Adens(j,5)=S02;Adens(j,6)=S11;bdens(j)=D00dens;   j=2;
Adens(j,1)=S10;Adens(j,2)=S20;Adens(j,3)=S11;Adens(j,4)=S30;
Adens(j,5)=S12;Adens(j,6)=S21;bdens(j)=D10dens;   j=3;
Adens(j,1)=S01;Adens(j,2)=S11;Adens(j,3)=S02;Adens(j,4)=S21;
Adens(j,5)=S03;Adens(j,6)=S12;bdens(j)=D01dens;   j=4;
Adens(j,1)=S20;Adens(j,2)=S30;Adens(j,3)=S21;Adens(j,4)=S40;
Adens(j,5)=S22;Adens(j,6)=S31;bdens(j)=D20dens;   j=5;
Adens(j,1)=S02;Adens(j,2)=S12;Adens(j,3)=S03;Adens(j,4)=S22;
Adens(j,5)=S04;Adens(j,6)=S13;bdens(j)=D02dens;   j=6;
Adens(j,1)=S11;Adens(j,2)=S21;Adens(j,3)=S12;Adens(j,4)=S31;
Adens(j,5)=S13;Adens(j,6)=S22;bdens(j)=D11dens;     j=1;
Amach(j,1)=S00;Amach(j,2)=S10;Amach(j,3)=S01;Amach(j,4)=S20;
Amach(j,5)=S02;Amach(j,6)=S11;bmach(j)=D00mach;   j=2;
Amach(j,1)=S10;Amach(j,2)=S20;Amach(j,3)=S11;Amach(j,4)=S30;
Amach(j,5)=S12;Amach(j,6)=S21;bmach(j)=D10mach;   j=3;
Amach(j,1)=S01;Amach(j,2)=S11;Amach(j,3)=S02;Amach(j,4)=S21;
Amach(j,5)=S03;Amach(j,6)=S12;bmach(j)=D01mach;   j=4;
Amach(j,1)=S20;Amach(j,2)=S30;Amach(j,3)=S21;Amach(j,4)=S40;
Amach(j,5)=S22;Amach(j,6)=S31;bmach(j)=D20mach;   j=5;
Amach(j,1)=S02;Amach(j,2)=S12;Amach(j,3)=S03;Amach(j,4)=S22;
Amach(j,5)=S04;Amach(j,6)=S13;bmach(j)=D02mach;   j=6;
Amach(j,1)=S11;Amach(j,2)=S21;Amach(j,3)=S12;Amach(j,4)=S31;
Amach(j,5)=S13;Amach(j,6)=S22;bmach(j)=D11mach;     j=1;
Avelx(j,1)=S00;Avelx(j,2)=S10;Avelx(j,3)=S01;Avelx(j,4)=S20;
Avelx(j,5)=S02;Avelx(j,6)=S11;bvelx(j)=D00velx;   j=2;
Avelx(j,1)=S10;Avelx(j,2)=S20;Avelx(j,3)=S11;Avelx(j,4)=S30;
Avelx(j,5)=S12;Avelx(j,6)=S21;bvelx(j)=D10velx;   j=3;
Avelx(j,1)=S01;Avelx(j,2)=S11;Avelx(j,3)=S02;Avelx(j,4)=S21;
Avelx(j,5)=S03;Avelx(j,6)=S12;bvelx(j)=D01velx;   j=4;
Avelx(j,1)=S20;Avelx(j,2)=S30;Avelx(j,3)=S21;Avelx(j,4)=S40;
Avelx(j,5)=S22;Avelx(j,6)=S31;bvelx(j)=D20velx;  j=5;
Avelx(j,1)=S02;Avelx(j,2)=S12;Avelx(j,3)=S03;Avelx(j,4)=S22;
Avelx(j,5)=S04;Avelx(j,6)=S13;bvelx(j)=D02velx;   j=6;
Avelx(j,1)=S11;Avelx(j,2)=S21;Avelx(j,3)=S12;Avelx(j,4)=S31;
Avelx(j,5)=S13;Avelx(j,6)=S22;bvelx(j)=D11velx;    j=1;
Avely(j,1)=S00;Avely(j,2)=S10;Avely(j,3)=S01;Avely(j,4)=S20;
Avely(j,5)=S02;Avely(j,6)=S11;bvely(j)=D00vely;   j=2;
Avely(j,1)=S10;Avely(j,2)=S20;Avely(j,3)=S11;Avely(j,4)=S30;
Avely(j,5)=S12;Avely(j,6)=S21;bvely(j)=D10vely;   j=3;
Avely(j,1)=S01;Avely(j,2)=S11;Avely(j,3)=S02;Avely(j,4)=S21;
Avely(j,5)=S03;Avely(j,6)=S12;bvely(j)=D01vely;   j=4;
Avely(j,1)=S20;Avely(j,2)=S30;Avely(j,3)=S21;Avely(j,4)=S40;
Avely(j,5)=S22;Avely(j,6)=S31;bvely(j)=D20vely;   j=5;
Avely(j,1)=S02;Avely(j,2)=S12;Avely(j,3)=S03;Avely(j,4)=S22;
Avely(j,5)=S04;Avely(j,6)=S13;bvely(j)=D02vely;   j=6;
Avely(j,1)=S11;Avely(j,2)=S21;Avely(j,3)=S12;Avely(j,4)=S31;
Avely(j,5)=S13;Avely(j,6)=S22;bvely(j)=D11vely;     j=1;
Avelz(j,1)=S00;Avelz(j,2)=S10;Avelz(j,3)=S01;Avelz(j,4)=S20;
Avelz(j,5)=S02;Avelz(j,6)=S11;bvelz(j)=D00velz;   j=2;
Avelz(j,1)=S10;Avelz(j,2)=S20;Avelz(j,3)=S11;Avelz(j,4)=S30;
Avelz(j,5)=S12;Avelz(j,6)=S21;bvelz(j)=D10velz;   j=3;
Avelz(j,1)=S01;Avelz(j,2)=S11;Avelz(j,3)=S02;Avelz(j,4)=S21;
Avelz(j,5)=S03;Avelz(j,6)=S12;bvelz(j)=D01velz;   j=4;
Avelz(j,1)=S20;Avelz(j,2)=S30;Avelz(j,3)=S21;Avelz(j,4)=S40;
Avelz(j,5)=S22;Avelz(j,6)=S31;bvelz(j)=D20velz;   j=5;
Avelz(j,1)=S02;Avelz(j,2)=S12;Avelz(j,3)=S03;Avelz(j,4)=S22;
Avelz(j,5)=S04;Avelz(j,6)=S13;bvelz(j)=D02velz;   j=6;
Avelz(j,1)=S11;Avelz(j,2)=S21;Avelz(j,3)=S12;Avelz(j,4)=S31;
Avelz(j,5)=S13;Avelz(j,6)=S22;bvelz(j)=D11velz;    j=1;
Avmag(j,1)=S00;Avmag(j,2)=S10;Avmag(j,3)=S01;Avmag(j,4)=S20;
Avmag(j,5)=S02;Avmag(j,6)=S11;bvmag(j)=D00vmag;   j=2;
Avmag(j,1)=S10;Avmag(j,2)=S20;Avmag(j,3)=S11;Avmag(j,4)=S30;
Avmag(j,5)=S12;Avmag(j,6)=S21;bvmag(j)=D10vmag;   j=3;
Avmag(j,1)=S01;Avmag(j,2)=S11;Avmag(j,3)=S02;Avmag(j,4)=S21;
Avmag(j,5)=S03;Avmag(j,6)=S12;bvmag(j)=D01vmag;  j=4;
Avmag(j,1)=S20;Avmag(j,2)=S30;Avmag(j,3)=S21;Avmag(j,4)=S40;
Avmag(j,5)=S22;Avmag(j,6)=S31;bvmag(j)=D20vmag;  j=5;
Avmag(j,1)=S02;Avmag(j,2)=S12;Avmag(j,3)=S03;Avmag(j,4)=S22;
Avmag(j,5)=S04;Avmag(j,6)=S13;bvmag(j)=D02vmag;   j=6;
Avmag(j,1)=S11;Avmag(j,2)=S21;Avmag(j,3)=S12;Avmag(j,4)=S31;
Avmag(j,5)=S13;Avmag(j,6)=S22;bvmag(j)=D11vmag;



btemp=btemp';
```

```
bmu=bmu';
bpress=bpress';
bdens=bdens';
bmach=bmach';
bvelx=bvelx';
bvely=bvely';
bvelz=bvelz';
bvmag=bvmag';




% solve for the coefficients
ctemp=Atemp\btemp;
cmu=Amu\bmu;
cpress=Apress\bpress;
cdens=Adens\bdens;
cmach=Amach\bmach;
cvelx=Avelx\bvelx;
cvely=Avely\bvely;
cvelz=Avelz\bvelz;
cvmag=Avmag\bvmag;


f00=ctemp(1);f10=ctemp(2);f01=ctemp(3);f20=ctemp(4);f02=ctemp(5);f11=ctemp(6);
mu00=cmu(1);mu10=cmu(2);mu01=cmu(3);mu20=cmu(4);mu02=cmu(5);mu11=cmu(6);
g00=cpress(1);g10=cpress(2);g01=cpress(3);g20=cpress(4);g02=cpress(5);g11=cpress(6);
h00=cdens(1);h10=cdens(2);h01=cdens(3);h20=cdens(4);h02=cdens(5);h11=cdens(6);
s00=cmach(1);s10=cmach(2);s01=cmach(3);s20=cmach(4);s02=cmach(5);s11=cmach(6);
velx00=cvelx(1);velx10=cvelx(2);velx01=cvelx(3);velx20=cvelx(4);velx02=cvelx(5);velx11=cvelx(6);
vely00=cvely(1);vely10=cvely(2);vely01=cvely(3);vely20=cvely(4);vely02=cvely(5);vely11=cvely(6);
velz00=cvelz(1);velz10=cvelz(2);velz01=cvelz(3);velz20=cvelz(4);velz02=cvelz(5);velz11=cvelz(6);
vmag00=cvmag(1);vmag10=cvmag(2);vmag01=cvmag(3);vmag20=cvmag(4);vmag02=cvmag(5);vmag11=cvmag(6);


% now build the approximation to test it

tempapprox=f00+f10*zeta+f01*rho+f20*zeta.^2+f02*rho.^2+f11*zeta.*rho;

muapprox=mu00+mu10*zeta+mu01*rho+mu20*zeta.^2+mu02*rho.^2+mu11*zeta.*rho;

pressapprox=g00+g10*zeta+g01*rho+g20*zeta.^2+g02*rho.^2+g11*zeta.*rho;

densapprox=h00+h10*zeta+h01*rho+h20*zeta.^2+h02*rho.^2+h11*zeta.*rho;

machapprox=s00+s10*zeta+s01*rho+s20*zeta.^2+s02*rho.^2+s11*zeta.*rho;

velxapprox=velx00+velx10*zeta+velx01*rho+velx20*zeta.^2+velx02*rho.^2+velx11*zeta.*rho;

velyapprox=vely00+vely10*zeta+vely01*rho+vely20*zeta.^2+vely02*rho.^2+vely11*zeta.*rho;

velzapprox=velz00+velz10*zeta+velz01*rho+velz20*zeta.^2+velz02*rho.^2+velz11*zeta.*rho;

vmagapprox=vmag00+vmag10*zeta+vmag01*rho+vmag20*zeta.^2+vmag02*rho.^2+vmag11*zeta.*rho;


%This is the routine that calculates the actual derivatives and
%Looks at the fluid equations

igrid=imin:imax;
jgrid=jmin:jmax;
```

```
    tempgrid=tempapprox(igrid,jgrid);
    mugrid=muapprox(igrid,jgrid);
    pressgrid=pressapprox(igrid,jgrid);
    densgrid=densapprox(igrid,jgrid);
    velxgrid=velxapprox(igrid,jgrid);
    velygrid=velyapprox(igrid,jgrid);
    velzgrid=velzapprox(igrid,jgrid);




    %hstrip is in number of particles.  I want kg/m^3 for N-S eqns.  (SI units)
    densSI=densgrid*40*(1.660d-27);





    z0=j0;
    r0=i0;



    diffvrr=velx01/drblk;
    diffvzz=velz01/drblk;
    diffvrz=velx10/dzblk;
    diffpr=g01/drblk;

    diffvrrarray(m)=diffvrr;
    diffvzzarray(m)=diffvzz;
    diffvrzarray(m)=diffvrz;
    diffprarray(m)=diffpr;



%    viscous term stuff: %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%     mu1=3.76e-7*tempgrid(imid,jmid).^.72  %  this is old.  I now have
%     mugrid;


     mu=mugrid(imid,jmid);


    visctrm1 = 2*velx20/dzblk^2 + 8/3*velx02/drblk^2 + 1/3*velz11/drblk/dzblk + ...
               4/3*velx01/r0/drblk -4/3*velx00/r0^2;
    visctrm2=2*(mu01*vely01)/drblk/drblk;
    visctrm3=(mu10*vely10*velz01)/drblk/drblk;
    visctrm4=-(2/3*mu01*(vely00/r0+vely01+velz10))/drblk/drblk;

    visctrmtot=visctrm1+visctrm2+visctrm3+visctrm4;



    visctrmarray1(m)=visctrm1;
    visctrmarray2(m)=visctrm2;
    visctrmarray3(m)=visctrm3;
    visctrmarray4(m)=visctrm4;
    visctrmarraytot(m)=visctrmtot;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%   end viscous term stuff  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
leftside=densSI(imid,jmid)*(velxgrid(imid,jmid)*diffvrr+velzgrid(imid,jmid)*diffvrz);
leftsidearray(m)=leftside;

%pause


rightside=-diffpr;
rightsidearray(m)=rightside+mu*(visctrm1)+(visctrm2+visctrm3+visctrm4);



ibot(m)=imin;itop(m)=imax;



metersi(m)=counteri*drblk;
counteri=counteri+1;

metersj(m)=counterj*drblk;
counterj=counterj+1;





%zero out the matrices



Atemp=zeros;%(size(Atemp));
Amu=zeros;
Apress=zeros;%(size(Atemp));
Adens=zeros;%(size(Atemp));
Amach=zeros;%(size(Atemp));
Avelx=zeros;%(size(Atemp));
Atvely=zeros;%(size(Atemp));
Avelz=zeros;%(size(Atemp));
Avmag=zeros;%(size(Atemp));


btemp=zeros;%(size(btemp));
bmu=zeros;
bpress=zeros;%(size(btemp));
bdens=zeros;%(size(btemp));
bmach=zeros;%(size(btemp));
bvelx=zeros;%(size(btemp));
bvely=zeros;%(size(btemp));
bvelz=zeros;%(size(btemp));
bvmag=zeros;%(size(btemp));



ctemp=zeros;%(size(ctemp));
cmu=zeros;
cpress=zeros;%(size(ctemp));
cdens=zeros;%(size(ctemp));
cmach=zeros;%(size(ctemp));
cvelx=zeros;%(size(ctemp));
cvely=zeros;%(size(ctemp));
cvelz=zeros;%(size(ctemp));
cvmag=zeros;%(size(ctemp));
```

```
    %move up in the nozzle

    i0=i0+1;



end



    plot(metersi,leftsidearray)
    hold on
    plot(metersi,rightsidearray,'r-')

    figure
    plot(metersi,mu*visctrmarray1)
    figure
    plot(metersi,mu*visctrmarray2)
    figure
    plot(metersi,mu*visctrmarray3)
    figure
    plot(metersi,mu*visctrmarray4)
```

Bibliography

[1] P. Farnsworth and A. C. Lazar, “Matrix effect studies in the inductively coupled plasma with monodisperse droplets - part 1: the influence of matrix on the vertical analyte emission profile,” Appl. Spectrosc. **53,** 457–464 (1999).

[2] P. Farnsworth, A. Mills, and J. Macedone, “High resolution imaging of barium ions and atoms near the sampling cone of an inductively coupled plasma mass spectrometer,” Spectrochimica Acta Part B: Atomic Spectroscopy **61,** 1039–1049 (2006).

[3] R. Thomas, “A beginner’s guide to ICP-MS,” http://matematicas.udea.edu.co/~carlopez/beginer_guide_icpms.pdf (Accessed Sept. 29, 2007).

[4] J. E. Patterson, P. Farnsworth, and B. S. Duersch, “Optically determined velocity distributions of metastable argon in the second stage of an inductively coupled plasma mass spectrometer,” Spectrochimica Acta Part B: Atomic Spectroscopy **54,** 537–544 (1999).

[5] A. Payne, “Analyzing the Skimming Process of an ICP-MS Using a Monte Carlo Simulation,” Senior Thesis .

[6] G. Horlick and M. Vaughn, "Effect of sampler and skimmer orifice size on analyte and analyte oxide signals in indictively coupled plasma-mass spectrometry," Spectrochimica Acta Part B: Atomic Spectroscopy **45,** 1301–1311 (1990).

[7] D. Douglas and J. French, "Gas Dynamics of the inductively coupled plasma mass spectrometry interface," J. Anal. At. Spectrosc. **3,** 743–747 (1988).

[8] G. Horlick and J. Lam, "A Comparison of argon and mixed gas plasmas for inductively coupled plasma-mass spectrometry," Spectrochimica Acta Part B: Atomic Spectroscopy **45,** 1313–1325 (1990).

[9] C. Benson, S. F. Gimelshein, D. Levin, and A. Montaser, "Simulation of droplet heating and desolvation in an inductively coupled plasma - Part I," Spectrochimica Acta Part B: Atomic Spectroscopy **56,** 1453–1471 (2001).

[10] M. Fraser and D. Beauchemin, "Effect of concomitant elements on the distribution of ions in inductively coupled plasma-mass spectroscopy. Part 1. Elemental ions," Spectrochimica Acta Part B: Atomic Spectroscopy **55,** 1705–1731 (2000).

[11] D. Chambers and G. Hieftje, "Fundamental studies of the sampling process in an inductively coupled plasma mass spectrometer - III: Monitoring the ion beam," Spectrochimica Acta Part B: Atomic Spectroscopy **46,** 785–804 (1991).

[12] D. Miller, "Free Jet Expansion," Am. J. Phys. **38,** 978–983 (1970).

[13] G. Bird, "Average energy flow of optical pulses dispersive media," Phys. Rev. Lett. **84,** 2370–2373 (2000).

[14] G. Bird, in *Molecular Gas Dynamics*, 4th ed., Molecu, ed., (American Institute of Physics, Woodbury, NY, 1990).

[15] B. Higgins, "University of Missouri Research Reactor Center ICP-MS Introduction Page," http://web.missouri.edu/~umcreactorweb/pages/ac_icpms1.shtml (Accessed Sept. 29, 2007).

[16] M. N. Macrossan and C. R. Lilley, "Viscosity of argon at temperatures $>$ 200 K from measured shock thickness," Phys. Fluids **15,** 3452–3457 (2003).

[17] C. Kittel and H. Kroemer, *Thermal Physics*, 2 ed. (W.H. Freeman, New York, 1980).

[18] H. M. K. Koura and T. Shimada, "A test of equivalence of the variable-hard-sphere and inverse-power-law models in the direct-simulation Monte Carlo method," Phys. Fluids A **3,** 2459–2465 (1991).

[19] K. Koura and H. Matsumoto, "Variable soft sphere molecular model for air species," Phys. Fluids A **4,** 1083–1085 (1992).

[20] B. Sanderson, *The Physics of Plasmas*, 4th ed. (Cambridge University Press, New York, 2003).

[21] H. Schlichting, *Boundary Layer Theory* (McGraw-Hill, New York, 1951).

Index