CALCULATING THE SAMPLING CONE GAS FLOW IN AN INDUCTIVELY COUPLED PLASMA MASS SPECTROMETER

by

Daniel E. Wilcox

A senior thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Bachelor of Science

Department of Physics and Astronomy

Brigham Young University

March 2009

BRIGHAM YOUNG UNIVERSITY

DEPARTMENT APPROVAL

of a senior thesis submitted by

Daniel E. Wilcox

This thesis has been reviewed by the research advisor, research coordinator, and department chair and has been found to be satisfactory.

Date	Ross L. Spencer, Advisor
Date	Eric Hintz, Research Coordinator
Date	Ross L. Spencer, Chair

ABSTRACT

CALCULATING THE SAMPLING CONE GAS FLOW IN AN INDUCTIVELY COUPLED PLASMA MASS SPECTROMETER

Daniel E. Wilcox

Department of Physics and Astronomy

Bachelor of Science

Inductively coupled plasma mass spectrometers contain a sampling cone which accelerates an atmospheric-pressure gas to supersonic speeds. Calculating the flow properties as the gas passes through the cone is challenging because of the difficulty in specifying upstream boundary conditions and because the gas exhibits non-ideal effects as it passes through the cone. To calculate the flow, the Direct Simulation Monte Carlo algorithm was used on the BYU supercomputing cluster using about 200 processors and 600 million simulation particles with a three-week calculation time. Carefully crafted velocity and temperature boundary conditions were necessary. Evidence is presented that the calculated flowfield is a good solution to the problem.

ACKNOWLEDGMENTS

Special thanks to the Fulton Supercomputing Laboratory for providing parallel compute hardware and support. Also thanks to the Department of Energy grant given to Paul Farnsworth and Ross Spencer of Brigham Young University.

Contents

Table of Contents xi

List of Figures xiii

1 Introduction 1
- 1.1 Complete specification of the situation . . . 2
- 1.2 Assumptions to enable calculating the flow . . . 3

2 Methods 5
- 2.1 FENIX . . . 5
- 2.2 General information for this simulation . . . 6
- 2.3 Introduction to the boundary conditions . . . 7
- 2.4 Complete description of the upstream boundary conditions . . . 8
  - 2.4.1 Splash flow . . . 9
  - 2.4.2 Sink flow . . . 11
  - 2.4.3 Definition of the adjustment functions . . . 11
  - 2.4.4 Summary of the boundary conditions . . . 13

3 Results 15
- 3.1 General results . . . 15
- 3.2 DSMC conditions . . . 16
  - 3.2.1 Requirement 1 . . . 16
  - 3.2.2 Requirement 2 . . . 16
  - 3.2.3 Requirement 3 . . . 17
- 3.3 Reasonableness of the boundary conditions . . . 18
  - 3.3.1 Evidence 1 . . . 18
  - 3.3.2 Evidence 2 . . . 19
  - 3.3.3 Evidence 3 . . . 19
  - 3.3.4 Evidence 4 . . . 20

4 Conclusions 21
- 4.1 Significance . . . 21
  - 4.1.1 Downstream calculations . . . 21
  - 4.1.2 Trace element flow calculations . . . 22

4.2 Further Work 22

Bibliography **22**

A A Description of DSMC **25**
A.1 Validity 26
A.2 Collision Models 28
A.2.1 Hard Sphere 29
A.2.2 Variable Hard Sphere 29
A.2.3 Variable Soft Sphere 29
A.3 Boundary Conditions 30
A.3.1 'Adaptive' boundary conditions 31
A.3.2 Flux 32
A.4 Solid Interaction Models. 32
A.4.1 Thermalization 32
A.4.2 Linear Combinations. 33
A.4.3 Other Models. 33

B Figures **35**

List of Figures

B.1 Diagram of the sampling cone 36
B.2 Diagram of the sampling cone with parameter labels 37
B.3 Streamlines of the gas flow 38
B.4 Temperature in K 39
B.5 Pressure in 10^4 Pa 40
B.6 Number density in $10^{24}\,\mathrm{m}^{-3}$ 41
B.7 Flow speed in m/s 42
B.8 Flow speed in m/s (slow-moving gas only) 43
B.9 v_r in m/s (slow-moving gas only) 44
B.10 v_z in m/s (slow-moving gas only) 45
B.11 Mach number 46
B.12 Streamlines near the nozzle 47
B.13 Temperature in K near the nozzle 48
B.14 Pressure in 10^4 Pa near the nozzle 49
B.15 Number density in $10^{24}\,\mathrm{m}^{-3}$ near the nozzle 50
B.16 Flow speed in m/s near the nozzle 51
B.17 Mach number near the nozzle 52
B.18 Temperature on the left boundary ($z = z_0$) 53
B.19 Pressure on the left boundary ($z = z_0$) 54
B.20 Number density on the left boundary ($z = z_0$) 55
B.21 Speed on the left boundary ($z = z_0$) 56
B.22 Radial speed (v_r) on the left boundary ($z = z_0$) 57
B.23 Axial speed (v_z) on the left boundary ($z = z_0$) 58
B.24 Temperature on the top boundary ($r = r_0$) 59
B.25 Pressure on the top boundary ($r = r_0$) 60
B.26 Number density on the top boundary ($r = r_0$) 61
B.27 Speed on the top boundary ($r = r_0$) 62
B.28 Radial speed (v_r) on the top boundary ($r = r_0$) 63
B.29 Axial speed (v_z) on the top boundary ($r = r_0$) 64
B.30 Pressure at $z = 6.25914\,\mathrm{mm}$ (halfway through the nozzle) 65
B.31 Temperature at $z = 6.25914\,\mathrm{mm}$ halfway through the nozzle) 66
B.32 Number density at $z = 6.25914\,\mathrm{mm}$ (halfway through the nozzle) . . . 67
B.33 Flow speed at $z = 6.25914\,\mathrm{mm}$ (halfway through the nozzle) 68

B.34 Axial speed (v_z) at $z = 6.25914\,\text{mm}$ (halfway through the nozzle) . . 69
B.35 Mach number at $z = 6.25914\,\text{mm}$ (halfway through the nozzle) 70

Chapter 1

Introduction

Inductively coupled plasma mass spectrometers (ICPs) have a hollow 'sampling cone' with atmospheric pressure conditions outside the cone and low vacuum conditions inside (see Figure B.1). The cone has a nozzle at its apex through which gas accelerates to supersonic speeds and low number densities. Non-ideal effects are observed such as viscosity and heat conduction, which prevent the use of simple fluid-flow calculation methods. (DSMC) was chosen as the computational method to simulate the gas.

DSMC is a stochastic method for finding solutions to the Boltzmann equation, which describes the flow of gases. DSMC is a particle-based technique that is valid for all mean free path lengths significantly longer than the molecular diameter. It uses enormously fewer computational resources than Molecular Dynamics (MD), which attempts to simulate every atom. DSMC is more completely described in Appendix A. If the reader is not familiar with DSMC, it is strongly recommended that Appendix A be read before continuing.

The Navier-Stokes equations—which describe non-ideal fluid flow when the fluids can be considered continuous—could have been used to calculate the upstream flow properties, but they were not used for several reasons. First, writing software to solve

the Navier-Stokes equations is difficult because of the equations' non-linearity. While a commercial or open-source computational fluid dynamics software package could have been used, these packages are still very difficult to use.

Second, providing adequate boundary conditions for solving the Navier-Stokes equations is challenging because slightly inconsistent boundary conditions can make the equations yield no answer. It often takes a very long time to craft boundary conditions for these equations.

Finally, in the future we will perform calculations downstream of the sampling cone which will feature non-continuum effects, requiring a technique like DSMC. It was thought wise to use DSMC upstream as well, where the software could be tested by seeing if its results satisfy the Navier-Stokes equations.

1.1 Complete specification of the situation

Upstream of the sampling cone, an inductively coupled argon plasma is generated at atmospheric pressure. The plasma has a low ionization rate but very high temperature (4000-6000K). A fine mist of the desired sample is injected into the plasma, almost completely ionizing the sample. As the plasma flows toward the sampling cone, the majority splashes against the cone; the remainder is sucked through the nozzle and accelerated to supersonic speed. After flowing through the nozzle, part of the low-density supersonic flow is sent to the mass spectrometer.

The purpose of the calculation described in this thesis is to characterize the flow through the sampling cone before it goes on to the rest of the mass spectrometer.

1.2 Assumptions to enable calculating the flow

Due to the difficulty of calculating the flow, assumptions were made to simplify the calculation:

- The flow is rotationally symmetric, allowing us to reduce the problem to two dimensions (cylindrical coordinates)
- The inductively coupled plasma has a low ionization rate so plasma effects are negligible
- The gas is nearly pure argon so the sample ions do not affect gas flow
- The flow rate is a uniform $+20\,\mathrm{m/s}$ toward the sampling cone far upstream of the cone
- The upstream flow pressure is $10^5\,\mathrm{Pa}$, approximately atmospheric pressure
- The upstream velocity boundary conditions (outside the cone) do not need to be exact, because flow through the cone's nozzle mainly depends on upstream pressure and temperature
- The downstream boundary conditions (inside the cone) can be approximated by true vacuum since the flow through the sampling cone nozzle is supersonic, so it has no dependence on downstream conditions
- The sampling cone has a uniform temperature of $1500\,\mathrm{K}$

Chapter 2

Methods

Graeme Bird [2] provides a free DSMC program named DS2V, which provides a graphical user interface for setting up the problem of interest and calculating flows (http://www.aeromech.usyd.edu.au/dsmc_gab/). However, his software is not usable for our problem for two reasons. First, his software does not allow complex enough boundary conditions for our problem. Only the most simple boundary conditions are possible; very complex boundary conditions are required to avoid needing an unreasonably large computation region.

Second, Bird's software is not parallel and our problem is large enough that several hundred million particles are needed, requiring approximately 40GB of memory. Because so many particles are needed, computation time becomes intractable unless one or two hundred processors work on the simulation simultaneously.

2.1 FENIX

Due to the limitations of Bird's software, a new DSMC program named FENIX was developed. FENIX performs three-dimensional DSMC over a thin cylindrical wedge,

r	z
8.23134 mm	4.152 mm
6.0204 mm	0.520384 mm
6.5394 mm	0.520384 mm
6.5394 mm	0.887144 mm
8.891508 mm	4.152 mm

Table 2.1 Vertices of the polygon used to represent the sampling cone, measured from the lower-left corner of the simulation

making it essentially two-dimensional DMSC. It uses the variable soft sphere collision model. It allows arbitrary boundary conditions, including ones that 'adapt' to the flow. Surface interactions can be any combination of the specular reflection and thermalization models. Refer to Appendix A for the definition of these terms.

2.2 General information for this simulation

This simulation used 220 processors and 600 million simulated particles. All surface interactions with the sampling cone were modeled with full thermalization, at 1500 K.

For those familiar with the internals of FENIX, the number of real particles per unit simulated particle (N_{ef}) was 1.690×10^5. The extent of the simulation was 4.152 mm in r, 8.9268 mm in z, and 0.001 radians around the cylindrical axis. The polygon used to represent the sampling cone had vertices as specified in table 2.1.

The variable soft sphere (VSS) collision model dictates

$$\sigma(\theta) = \frac{\alpha A v_r^{-2\nu}}{4\pi} \left[\cos^2(\theta/2)\right]^{\alpha-1} \tag{2.1}$$

as its differential collision cross-section [5], where v_r is the relative speed of the colliding particles, θ the center-of-mass scattering angle, and α, ν, and A are the

VSS parameters. The chosen parameters were $\alpha = 1.66$, $\nu = 0.22$, and $A = 7.924 \times 10^{-18}\,\mathrm{m}^{2.44}/\mathrm{s}^{0.44}$, as described by Spencer [5].

2.3 Introduction to the boundary conditions

It is observed that the flow upstream from the sampling cone is a combination of splash flow and sink flow: some of the gas splashes against the sampling cone and the rest is sucked through the nozzle. We modeled this flow as the sum of an analytical splash flow and an analytical sink flow. Since solving the Euler or Navier-Stokes equations for the analytical form of the splash and sink flows is difficult—especially when including the boundary layers near the sampling cone surface—we developed a set of approximate formulas for the flow. These formulas have the following structure: there are relatively simple components that describe the splash and sink flows, multiplied by 'adjustment' formulas that describe the boundary layers and the upstream temperature profile. We are confident that the inaccuracies in our boundary conditions do not significantly affect the flow through the nozzle because we have performed this calculation using a wide range of boundary conditions which all yielded the same flowfield through the sampling nozzle.

There are two main purposes of having these complicated boundary conditions. First, we would like the calculated flow to be accurate for more than just the nozzle: at some point we will be including more effects in our calculations, and we need to know that the flow is fairly accurate for a few millimeters away from the nozzle. Second, comparing the nozzle flow with these complicated boundary conditions to nozzle flow with simple boundary conditions gives us confidence that the nozzle flow does not depend on boundary conditions.

Both the splash flow and the sink flow formulas we developed are approximate,

but they behave the way they intuitively should. The formulas were constrained to yield the correct flux through the boundaries, which allows a self-consistent steady state to occur.

The remaining boundary is downstream of the nozzle. The condition for this boundary was much simpler: particles that cross that boundary are simply allowed to leave.

2.4 Complete description of the upstream boundary conditions

Several features needed to be included in the sink and splash flow formulas:

- There is a temperature boundary layer close to the cone—the temperature goes down and the number density goes up as the gas approaches the sampling cone
- There is a velocity boundary layer close to the cone—the speed decreases and approaches zero as the gas approaches the sampling cone
- There is a constant-pressure flow far upstream of the sampling cone in the $+z$ direction at a uniform $20\,\mathrm{m/s}$
- A temperature profile is possible at this far upstream position that is only a function of the distance from the axis
- The upstream reference number density is $n_0 = 1.45 \times 10^{24}\,\mathrm{m^{-3}}$, and the upstream stagnation temperature is $T_0 = 5000\,\mathrm{K}$

We define several variables as depicted in Figure B.2. The origin from which these variables are measured is the theoretical apex of the sampling cone, as if the nozzle

weren't there. z_0 is the location of the z_{min} boundary of the simulation (it is negative). r_0 is the location of the r_{max} boundary of the simulation. At a given point, r is the distance from that point to the z axis, z is the Euclidean z-coordinate, θ is the angle about origin (the theoretical cone apex) measured from the $-z$ axis, $\mathbf{R}$ is the vector from the origin to that point, and R is the magnitude of that vector. α—a different α than the symbol used in the variable soft sphere collision model—is the angle of the sampling cone from its axis to its surface. $\theta_0 = \pi - \alpha$ is the maximum value of θ the gas can have (upstream of the nozzle). $z_c = r_0/\tan\alpha$ is the intersection of the cone with the r_{max} boundary of the simulation. ϕ is a flow angle; it only applies to the splash-flow formulas and represents the angle of the splash flow at a given point in space with respect to the $+z$ axis.

2.4.1 Splash flow

The basic splash flow formula is composed of two parts: a flow angle part and a flow speed part. The flow angle ϕ is given below:

$$\phi(\theta) = \alpha \left(\frac{\theta}{\theta_0}\right)^2 \tag{2.2}$$

The flow speed v is given by:

$$v^{(splash)}(r,\theta) = v_0 \left(1 - a\left(\frac{\theta}{\theta_0}\right)^2\right) H(\theta) \tag{2.3}$$

where $v_0 \approx 20\,\mathrm{m/s}$ such that $v_0 + v^{(sink)} = 20\,\mathrm{m/s}$ at $(r,z) = (0,z_0)$. A formula for $v^{(sink)}$—the portion of the upstream flow speed due to sink flow—will be provided later in this paper. Observe that ϕ approaches α as the flow approaches the sampling cone. a is an adjustable parameter to allow the flux of the splash flow through the simulation boundaries to be zero. A formula to find a will be derived a few paragraphs down. Note the presence of the function $H(\theta)$. It is an approximate function that

models how the speed decreases as the gas approaches the sampling cone—it models the velocity boundary layer near the sampling cone. It will be defined in Section 2.4.3.

Combining ϕ and $v^{(splash)}$ yields

$$v_z^{(splash)} = v_0 \left[1 - a\left(\frac{\theta}{\theta_0}\right)^2\right] H(\theta)\cos\phi \tag{2.4}$$

and

$$v_r^{(splash)} = v_0 \left[1 - a\left(\frac{\theta}{\theta_0}\right)^2\right] H(\theta)\sin\phi \tag{2.5}$$

To find a, observe that the flux through the z_{min} boundary due to splash flow is given by

$$\Gamma_b = 2\pi v_0 \left[\int_0^{r_0} n_0 M(r)\cos\phi r dr - a\int_0^{r_0} n_0 M(r)\cos\phi \left(\frac{\theta}{\theta_0}\right)^2 r dr\right] \tag{2.6}$$

where $n_0 M(r)$ is the number density on the z_{min} boundary as defined in Section 2.4.3. The above equation is only true since $H(\theta) \approx 1$ on the z_{min} boundary. The flux through the r_{max} boundary due to splash flow is given by

$$\Gamma_t = 2\pi v_0 r_0 \left[\int_{z_c}^{z_0} n_t(\theta) H(\theta)\sin\phi dz - a\int_{z_c}^{z_0} n_t(\theta) H(\theta)\sin\phi \left(\frac{\theta}{\theta_0}\right)^2 dz\right] \tag{2.7}$$

where $n_t(\theta)$ is an adjustment function that describes how the number density increases in the temperature boundary layer, and is defined in Section 2.4.3. Requiring the two fluxes to be the same—requiring that the splash flow not yield any flux through the nozzle—yields the following expression for a:

$$a = \frac{\displaystyle\int_0^{r_0} n_0 M(r)\cos\phi r dr + r_0\int_{z_c}^{z_0} n_t(\theta) H(\theta)\sin\phi dz}{\displaystyle\int_0^{r_0} n_0 M(r)\cos\phi \left(\frac{\theta}{\theta_0}\right)^2 r dr + r_0\int_{z_c}^{z_0} n_t(\theta) H(\theta)\sin\phi \left(\frac{\theta}{\theta_0}\right)^2 dz} \tag{2.8}$$

The value of this expression is determined by numeric integration. Determining a ensures that the splash flow does not contribute to any flow through the nozzle.

2.4.2 Sink flow

The basic sink flow formula is simple:

$$\mathbf{v}^{(sink)} = -\frac{S}{R^2} H(\theta)\hat{\mathbf{R}} \tag{2.9}$$

which is similar to the formula for ideal incompressible spherical sink flow. S is a constant that reflects how much flow goes through the nozzle. S can be found by noting that the particle flux through the nozzle is given by

$$\frac{dN}{dt} = G\overline{n}_0 C_s D^2 \tag{2.10}$$

according to ideal duct theory [4], but where $G = 0.38$ is a modified duct flow constant found by Spencer et al [5]. $\overline{n}_0 = n_0 M(0.0)$ is the upstream stagnation number density, $C_s = \sqrt{5k_B T_0/3m}$ is the speed of sound, and $D = 2\times 0.519\,\text{mm}$ is the nozzle diameter. $M(r)$ is a z_{min} number density profile defined in Section 2.4.3. Since the splash flow contributes no flux through the nozzle, the particle flux through the nozzle is also given by

$$\frac{dN}{dt} = 2\pi S \int_0^{\theta_0} n_0 M(z_0 \sin\theta) H(\theta) \sin\theta d\theta \tag{2.11}$$

so long as θ_0 is not much greater than $\pi/2$. Equating the two formulas for dN/dt leads to the following formula for S:

$$S = \frac{G\overline{n}_0 C_s D^2}{2\pi \int_0^{\theta_0} n_0 M(z_0 \sin\theta) H(\theta) \sin\theta d\theta} \tag{2.12}$$

which can be numerically solved once $H(\theta)$ and $M(r)$ are specified.

2.4.3 Definition of the adjustment functions

The velocity boundary layer adjustment function $H(\theta)$ was defined as

$$H(\theta) = \text{erf}\left(\beta_1 k(\theta_0 - \theta)\right) \tag{2.13}$$

where $k = \sqrt{v_0 * 10^{-3}/\nu}$ and $\beta_1 = 0.6$ is a dimensionless parameter for adjusting the thickness of the velocity boundary layer. ν is a different ν than the symbol used in the collision model; it is the kinematic viscosity and is given by Macrossan and Lilley [8] as

$$\nu = \frac{3.758 \times 10^{-7}}{m n_0} T_0^{0.72} \tag{2.14}$$

where m is the atomic mass of argon and n_0 and T_0 are in SI units.

This definition of k yields a circular dependency: k depends on v_0, S depends on k, and v_0 depends on S. An iterative scheme was used to find (v_0, k, S) such that k and S are correct relative to v_0, and $v_0 + v^{(sink)} = 20\,\mathrm{m/s}$ at $(r, z) = (0, z_0)$.

To model the temperature boundary layer, a function $T(\theta)$ was defined that models how the temperature decreases as the gas approaches the sampling cone:

$$T(\theta) = T_s + (T_0 - T_s)\,\mathrm{erf}\,(\beta_2 k(\theta_0 - \theta)) \tag{2.15}$$

where $T_s = 1500\,\mathrm{K}$ is the temperature of the sampling cone and $\beta_2 = 1.2$ is a second dimensionless parameter for adjusting the thickness of the temperature boundary layer.

Due to the constant-pressure boundary condition, the temperature boundary layer also yields a number density boundary layer: $n_t(\theta)$ is given by:

$$n_t(\theta) = n_0 \frac{T_0}{T(\theta)} \tag{2.16}$$

Finally, the temperature profile at z_{min} is represented as a number density profile $M(r)$ which goes to 1 at r_0. As such, it is a fractional number density profile. In other words, the number density at z_{min} is given by $n_0 M(r)$ and the temperature at z_{min} is given by $P/(n_0 M(r) k_B)$ where $P = 10^5\,\mathrm{Pa}$ is the upstream pressure. For this research, $M(r)$ was defined to be 1 for all r.

2.4.4 Summary of the boundary conditions

The total velocity boundary condition is given by the sum of the sink and splash velocity terms: $(v_r, v_z) = (-\|\mathbf{v}^{(sink)}\| \sin\theta + v_r^{(splash)}, \|\mathbf{v}^{(sink)}\| \cos\theta + v_z^{(splash)})$.

The number density boundary condition at z_{min} is $n_0 M(r)$, and at r_{max} is $n_0 n_t(\theta)$.

The temperature boundary condition is $P/(nk_B)$ where n is given above by the number density boundary condition, k_B is the Boltzmann constant, and $P = 10^5\,\mathrm{Pa}$.

Chapter 3

Results

3.1 General results

Figures B.3–B.17 summarize the flowfield.

The gas is at nearly constant pressure over the entire upstream region (Figure B.5), except very close to the nozzle. This is the expected gas behavior. The streamlines (Figure B.3) reveal a separatrix, showing some of the gas going into the nozzle and the rest splashing against the sampling cone, as expected. The temperature (Figure B.4) and velocity (Figure B.8) boundary layers are clearly visible, and exhibit little or no pinching at the r_{max} boundary of the simulation. This required adjustment of β_1 and β_2; the final values are as reported in Section 2.4.3.

Figures B.12–B.17 show what goes on inside the nozzle. There is a very large acceleration through the nozzle, tempered by velocity boundary layers near the nozzle surface. There are prominent temperature boundary layers near the nozzle surface as well. The flow becomes supersonic (mach number equal to one) about halfway through the nozzle: this is the expected result [6].

3.2 DSMC conditions

DSMC has three main requirements for it to be valid (see Appendix A); each requirement is discussed below.

3.2.1 Requirement 1

The first requirement is that the mean molecular spacing must be much larger than the molecular diameter. The highest mean molecular spacing happens near the sampling cone, where $T = 1500\,\mathrm{K}$. There, $n = 4.833 \times 10^{24}\,\mathrm{m}^{-3}$ so the mean molecular spacing is $5.914 \times 10^{-9}\,\mathrm{m}$.

The average collision cross-section is $\langle\sigma\rangle = 0.749A/v_{th}^{2\nu}$ where A and ν are the VSS paramters, and $v_{th} = \sqrt{k_B T/m}$. At 1500 K, $\langle\sigma\rangle = 3.67 \times 10^{-19}\,\mathrm{m}^2$ which corresponds to an effective diameter of $3.42 \times 10^{-10}\,\mathrm{m}$, which is much smaller than the mean molecular spacing.

3.2.2 Requirement 2

The second requirement is that the mean distance between collision partners must be much less than the mean free path. The mean distance between collision partners is the mean distance between a simulated particle and its nearest neighbor, since collisions only happen between nearest neighbors. This condition is most difficult to satisfy on the axis. The mean free path is given by $l = (\sqrt{2}\langle\sigma\rangle n)^{-1}$, which is equal to $1.73 \times 10^{-6}\,\mathrm{m}$ on-axis since $n = 1.45 \times 10^{24}$ and $\langle\sigma\rangle = 2.816 \times 10^{-19}\,\mathrm{m}^2$ at 5000 K.

The mean collision separation is given by Equation A.10 to be $n_s^{-1/2}/2$, where n_s is the two-dimensional number density of the simulated particles. $n_s = 2\pi r w n/N_{ef} \approx 8.577 \times 10^{15}\,\mathrm{m}^{-3} \cdot r$ where $w = 0.001$ radians is the cylindrical wedge width, so the radial distance at which the mean collision separation is one-third the mean free

path is $8.8 \times 10^{-5}\,\mathrm{m}$, which is about one sixth of the nozzle radius. While this may seem significant, it is not for two reasons. First, the cross-sectional area in the x-y plane of the invalid region is about one thirty-fifth of the cross-sectional area of the sampling nozzle. Second, physically speaking the gradients in r of the number density, temperature, pressure, and velocity must go to zero as r approaches zero. Longer separations between collision partners just tends to force this condition a little more strongly.

In the nozzle, $T \approx 3000\,\mathrm{K}$ and $n \approx 6 \times 10^{23}\,\mathrm{m}^{-3}$, so $\langle\sigma\rangle = 3.15 \times 10^{-19}\,\mathrm{m}^2$ and the mean free path is about $3.74 \times 10^{-6}\,\mathrm{m}$. On the other hand, the mean simulated particle density decreases to $n_s \approx 3.549 \times 10^{15}\,\mathrm{m}^{-3} \cdot r$, so the radial distance at which the mean collision separation is one-third the mean free path is $4.53 \times 10^{-5}\,\mathrm{m}$—slightly better than the upstream case and less than a tenth of the nozzle radius.

3.2.3 Requirement 3

The third requirement is that there must be enough particles per collision cell to obtain good statistics. The collision cells that border the axis have the smallest volume and therefore the fewest particles. On average, they have about 3.8 particles, and all other collision cells have more than twice as many on average. In the nozzle the number density is lower, about $6 \times 10^{23}\,\mathrm{m}^{-3}$, making the first collision cell have about 1.6 particles, and the second collision cell about 3.6 particles. However, the first collision cell only extends to $r = 2.076 \times 10^{-5}\,\mathrm{m}$, so the invalidity very close to the axis is essentially negligible.

3.3 Reasonableness of the boundary conditions

There are four different evidences that the boundary conditions we provided are reasonable.

3.3.1 Evidence 1

The first evidence is that the total flux through the upstream simulation boundaries equals the flux through the nozzle. In general, the particle flux through a surface S is given by

$$\int_S n\mathbf{v} \cdot \mathbf{dA} \tag{3.1}$$

where $\mathbf{v}$ is the flow velocity and n is the number density. Therefore, the particle flux through the nozzle at a given position z_f is given by

$$\Phi_n = \int_0^{r_n} n(r, z_f) v_z(r, z_f) 2\pi r dr \tag{3.2}$$

where $r_n = 0.519\,\text{mm}$ is the radius of the nozzle. Evaluating this integral numerically via the midpoint rule at $z_f = 6.25914\,\text{mm}$ yields $\Phi_n = 7.7459 \times 10^{20}\,\text{s}^{-1}$. Refer to Figures B.32 and B.34 for the relevant information.

The flux through the z_{min} boundary is given by a similar integral:

$$\Phi_z = \int_0^{r_0} n(r, z_0) v_z(r, z_0) 2\pi r dr \tag{3.3}$$

which yields $\Phi_z = 1.5101 \times 10^{21}\,\text{s}^{-1}$ when evaluated by the midpoint rule. Refer to Figures B.20 and B.23.

The flux through the r_{max} boundary is given by a slightly different integral:

$$\Phi_r = \int_{z_0}^{z_c} n(r_0, z) v_r(r_0, z) 2\pi r_0 dz \tag{3.4}$$

which yields $\Phi_r = 7.2495 \times 10^{20}\,\text{s}^{-1}$ when evaluated by the midpoint rule. Refer to Figures B.26 and B.28.

The flux through the upstream boundaries is therefore given by $\Phi_z - \Phi_r$. Evaluating the supposed equality, $\Phi_z - \Phi_r = \Phi_n$ to within 0.352% of the total flux because the difference between the two sides is $1.0601 \times 10^{19}\,\mathrm{s}^{-1}$. The fact that the flux in through the boundaries is nearly identical to the flux through the nozzle is evidence that the simulation has reached a steady-state solution.

3.3.2 Evidence 2

The second evidence that the boundary conditions are reasonable is that the flux through the nozzle is the correct amount. In the boundary-condition specification section, a 'constant' S was introduced, under the assumption that it is fairly independent of upstream boundary conditions. This constant S was solved for by equating two formulas for flux through the nozzle:

$$\frac{dN}{dt} = G\overline{n}_0 C_s D^2 = 2\pi S \int_0^{\theta_0} n_0 M(z_0 \sin\theta) H(\theta) \sin\theta d\theta \tag{3.5}$$

This yields $S = 6.912 \times 10^{-5}\,\mathrm{m}^2/\mathrm{s}$, with the numerical result of these formulas for flux through the nozzle yielding 7.8208×10^{20}. This result can be checked since there is yet another way to calculate the flux through the nozzle once the simulation has run: computing Φ_n as defined in Equation 3.2. The fact that $\Phi_n = 7.8208 \times 10^{20}$ to within 1% implies that it is reasonable to call S a constant, and to base the sink-flow portion of the boundary conditions on S, especially when combined with the other simulations that yielded identical flows through the nozzle independent of the upstream velocity boundary conditions.

3.3.3 Evidence 3

The third evidence that the boundary conditions are reasonable is that the boundary conditions are obeyed. As mentioned in Appendix A, DSMC finds a solution even

if the boundary conditions are inconsistent. In a sense, it modifies the boundary conditions to find a solution.

Observe Figures B.18 through B.23. They plot the requested z_{min} boundary condition against the computed z_{min} boundary characteristics. Figures B.24 through B.29 plot the same quantities along the r_{max} boundary. The fact that the requested boundary conditions agree extremely well with the computed boundary characteristics implies that the provided boundary conditions were consistent, or nearly so.

3.3.4 Evidence 4

The final evidence that the computed solution is acceptable is that the boundary conditions make sense. Upstream, the velocity is a nearly steady 20 m/s, as expected. Figures B.4 and B.8 reveal the computed temperature and velocity boundary layers near the sampling cone. Notice their contours go smoothly into the boundary without bending. This is evidence that the fudge factors and the boundary layer terms in the boundary conditions adequately approximate the boundary layers. While the velocity boundary layer contours do not go into the boundary as smoothly as the temperature boundary layer contours, this is acceptable because the discrepancies are on the order of a few m/s, while the flow through the nozzle is approximately 1500 m/s.

In fact, it is remarkable that DSMC is providing any sort of velocity resolution at all at the 1 m/s scale. The thermal velocities of the particles are $\sqrt{k_b T/m} \approx 1000$ m/s, requiring about 10^6 samples in a given location to provide velocity resolution at the 1 m/s scale for that location.

Chapter 4

Conclusions

4.1 Significance

These results do not directly affect the design of an ICP very much. However, these results enable other very important calculations: downstream calculations and trace element flow calculations.

4.1.1 Downstream calculations

By carefully observing the flow through the nozzle, very good upstream boundary conditions can be generated for downstream simulations. In other words, there is more gas-flow apparatus that needs simulating which is downstream of the sampling nozzle, and the upstream simulation allows a near-exact upstream boundary condition for the downstream simulation. These downstream simulations can more directly affect the design of an ICP.

To use the upstream simulation as an upstream boundary condition for a downstream simulation, the code is modified to store positions and velocities of particles as they cross a plane in the nozzle. These particles are then fed into the sampling

nozzle boundary of the downstream simulation.

4.1.2 Trace element flow calculations

Although the sample ions do not affect the argon flow much, the argon flowfield affects the sample ion flow and the sample ion diffusion rate. Applying elementary kinetic theory to the sample ion flow neglecting ion-ion interactions yields a fairly simple partial differential equation for ion flow which is valid for most of the upstream region (everywhere the continuum fluid equations are valid).

It is also possible to use DSMC directly on the particle ions to compute the ion flow. This requires the collision model to be modified so the ions stochastically collide with the pre-computed argon flowfield instead of with each other.

The trace element flow calculations are important for the design of an ICP since the nozzle size and shape affects how the trace elements flow, and a major design goal of any mass spectrometer is to accelerate as much of the sample as possible and get it to the parts of the machine that actually measure mass.

4.2 Further Work

There are several computations that need to succeed this one. The downstream region is the obvious example, using the results of the nozzle flow from this computation for the upstream boundary condition. Also, trace-element flow calculations need to be done. Finally, work can be done to include more effects in the simulation. For example, as the argon comes out of the inductively coupled plasma source, the argon spins slightly about the axis. Further, it may be possible to include some plasma effects in the simulation, especially near the nozzle where they are more likely to affect the outcome.

Bibliography

[1] Garcia AL, Bell JB, Crutchfield WY, and Alder BJ. Adaptive mesh and algorithm refinement using direct simulation monte carlo. *Journal of Computational Physics*, 154:134–155, 1999.

[2] G. A. Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Clarendon Press, 1994.

[3] G. A. Bird. Sophisticated dsmc, September 2007. Notes prepared for a short course at the DSMC07 meeting at Santa Fe http://www.aeromech.usyd.edu.au/dsmc_gab/Resources/DSMC07notes.pdf (accessed 25 March 2009); electronic copy included in this thesis' data CD.

[4] D. J. Douglas and J. B. French. Gas dynamics of the inductively coupled plasma mass spectrometry interface. *J. Anal. Atom. Spectrom.*, 3:743–747, 1988.

[5] R. L. Spencer et al. Modeling the gas flow upstream and in the sampling nozzle of the inductively coupled plasma mass spectrometer via the direct simulation monte carlo algorithm. *Spectrochim. Acta Part B*, 2009.

[6] H. W. Liepmann and A. E. Puckett. *Introduction to Aerodynamics of a Compressible Fluid*. John Wiley and Sons, 1947.

[7] Charles R. Lilley and Michael N. Macrossan. Methods for implementing the stream boundary condition in dsmc computations. *Int. J. Numer. Meth. Fluids*, 42:1363–1371, 2003.

[8] M. N. Macrossan and C.R. Lilley. Viscosity of argon at temperatures $>$ 2000 K from measured shock thickness. *Phys. Fluids*, 15:3452–3457, 2003.

[9] E.S. Oran, C.K. Oh, and B.Z. Cybyk. Direct simulation monte carlo: Recent advances and applications. *Annu. Rev. Fluid Mech.*, 30:403–41, 1998.

[10] M. Berberan Santos. On the distribution of the nearest neighbor. *Am. J. Phys.*, 54, 1986.

Appendix A

A Description of DSMC

DSMC [2] is a method invented by Graeme Bird to compute rarefied gas flows. It is a time-stepping particle-based method that models collisions stochastically, and decouples particle movement from collisions. This method does not attempt to simulate every gas particle, but it simulates 'representative' gas particles. Each simulated particle 'represents' some number $N_{ef} \sim 10^5$–10^9 of physical particles. DSMC is accurate for all mean free paths $\lambda \gg d$ where d is the molecular diameter [9], provided enough particles are used to make the mean separation between collision partners much less than the mean free path [3]. Further, DSMC automatically includes viscous and heat conduction effects.

DSMC has two phases, which are cycled through alternately. In the first phase, particles are not allowed to collide, but are moved a displacement $\tau\mathbf{v}$, where τ is a time step somewhat less than the mean time between collisions and $\mathbf{v}$ is the given particle's velocity. In the second phase, particles do not move, but are allowed to 'collide' with their nearest neighbors in a fashion that conserves energy and linear momentum, thus giving random new velocities to the particles. Collisions between particles of high relative velocities are favored, since in a real gas collisions between

particles of high relative velocity are more likely than collisions between particles of low relative velocity.

Particles are sorted into 'collision cells,' which are small spatial regions much smaller in extent than a mean free path. Collisions are only allowed to happen between particles in the same collision cell. Collisions are two-body; three-body collisions are not allowed. Each time step, a certain number of 'candidate collisions' are evaluated in each collision cell. The number of candidate collisions in a particular cell is a function of the size of the particles, τ, the number of particles in the cell, the volume of the cell, and how many real particles each representative particle represents. For each candidate collision, a random particle in the cell is selected, and the nearest neighbor of that particle within the collision cell is termed its 'collision partner.' Each candidate collision is stochastically accepted or rejected, with the probability of acceptance dependent on relative velocity, the size of the particles, and the collision model (see below). If a candidate collision is accepted, new velocities are assigned to the particles, in a manner that assigns random velocities but conserves energy and linear momentum.

A.1 Validity

DSMC requires several assumptions to be valid for it to be applicable [9]:

- The mean molecular spacing must be much larger than the molecular diameter
- The mean simulated collision separation must be much less than the mean free path [3]
- There must be enough particles per collision cell to obtain good statistics

The number of particles per collision cell doesn't need to be very high for a tolerable level of statistics; Bird claims that five to ten simulated molecules per collision cell is adequate when using nearest-neighbor collision models [3].

Notice that the mean molecular spacing is related to but not the same as the mean simulated collision separation. The mean molecular spacing is simply $n^{-1/3}$, where n is the number density. However, the mean simulated collision separation is talking about representative particles, and is somewhat lower than the mean spacing between representative particles since only collisions between nearest neighbors are allowed. The distribution of the mean simulated collision separation is derived below.

According to Santos [10], the distribution of the distance to a particle's nearest neighbor in a three-dimensional gas is

$$f_3(r) = 4\pi r^2 n \exp\left(-\frac{4}{3}\pi r^3 n\right) \tag{A.1}$$

which has a mean value of

$$\frac{1}{9}\frac{3^{5/6}\pi^{2/3}}{2^{1/3}\Gamma(2/3)}n^{-1/3} \tag{A.2}$$

which is approximately $0.554n^{-1/3}$.

Referring to the derivation given by Santos, if N evenly-distributed particles are in a large box of size V the probability of having K particles within a distance r of some point inside the box is

$$p(K) = \binom{N}{K}\left(\frac{4}{3}\pi r^3\frac{1}{V}\right)^K\left(1-\frac{4}{3}\pi r^3\frac{1}{V}\right)^{N-K} \tag{A.3}$$

as given by the binomial distribution. The probability that no particles occur ($K = 0$) in this region of radius r is the same as one minus the probability that a particle does occur in that region:

$$1 - \int_0^r f_3(r')dr' = \left(1-\frac{4}{3}\pi r^3\frac{1}{V}\right)^N \tag{A.4}$$

Taking the limit $N \to \infty$ while fixing $n = N/V$ yields

$$1 - \int_0^r f_3(r')dr' = \exp\left(-\frac{4}{3}\pi r^3 n\right) \tag{A.5}$$

Differentiating both sides with respect to r yields $f_3(r)$ as given above. Extending this technique to two-dimensional gases (N evenly-distributed particles over an area A) yields

$$p(K) = \binom{N}{K}\left(\frac{\pi r^2}{A}\right)^K\left(1 - \frac{\pi r^2}{A}\right)^{N-K} \tag{A.6}$$

so

$$1 - \int_0^r f_2(r')dr' = \left(1 - \frac{\pi r^2}{A}\right)^N \tag{A.7}$$

Taking the limit $N \to \infty$ while fixing the two-dimensional number density $n_2 = N/A$ yields

$$1 - \int_0^r f_2(r')dr' = \exp\left(-\pi r^2 n\right) \tag{A.8}$$

Differentiating both sides yields

$$f_2(r) = 2\pi r n_2 \exp\left(-\pi r^2 n_2\right) \tag{A.9}$$

which has mean value

$$\frac{1}{2}n_2^{-1/2} \tag{A.10}$$

A.2 Collision Models

There are several different collision techniques (models) discussed here: hard sphere, variable hard sphere, and variable soft sphere. They differ in the the probability that a particular candidate collision is accepted and in the manner of assigning new velocities to collision partners. In all cases, linear momentum and energy are conserved by computing the center of mass velocity and the relative velocity of the collision partners. The center of mass velocity and the magnitude of the relative velocity are conserved, but the direction of the relative velocity is randomly reassigned [2].

A.2.1 Hard Sphere

The hard sphere collision model dictates the cross section of candidate collisions to not depend on the relative speed of the particles. In other words, the molecules are all modeled by billiard balls with fixed mass and diameter. Once a collision is accepted, the direction of relative velocity is randomly reassigned according to an isotropic spherical distribution. This is the simplest model, but it usually cannot properly account for the temperature dependence of viscosity, nor can it properly describe diffusion [2].

A.2.2 Variable Hard Sphere

The variable hard sphere model allows for viscosity dependence on temperature by altering the cross-section of a candidate collision to depend on the relative speed of the particles. In this model, the molecules are modeled by billiard balls, but their size is re-determined for each possible candidate collision according to the relative speed of the molecules in question. However, the post-collision relative velocity direction is still isotropically distributed. This model can allow viscosity to vary in a more realistic fashion, but no improvements are made to alter the diffusion coefficients [2].

A.2.3 Variable Soft Sphere

The variable soft sphere model extends the variable hard sphere model by allowing the post-collision velocity direction distribution to depend on the relative speed of the particles. More precisely, the relative velocity direction change is pulled from a distribution that is not isotropic, but depends on the original relative velocity direction and speed. In this way, one can imagine that the dynamically growing and shrinking billiard balls of the Variable Hard Sphere model are replaced by dynamically

growing and shrinking soft foam balls. This model allows the diffusion coefficients to be corrected in addition to viscosity [2]. Because of the additional accuracy, this is the model we have chosen to use.

A.3 Boundary Conditions

Boundary conditions in DSMC are the mechanism for allowing particles to enter in at the sides of the simulation. After all, particles are constantly leaving the simulation due to the dynamic nature of gases, so a mechanism must be in place to allow particles to enter the simulation in a way consistent with the problem of interest.

Boundary conditions can be implemented in DSMC by surrounding the simulation region by 'ghost cells' full of particles. At each time step, during the moving phase, particles in the ghost cells are allowed to migrate into the simulation region. Collisions do not occur in ghost cells. At each time step, after the moving phase, all of the particles in the ghost cells are removed, and are replaced with new particles according to the desired distribution [1] [7]. Thus, DSMC allows the user to specify arbitrary gas conditions.

It is entirely possible to specify inconsistent boundary conditions. For example, one can specify constant flow outward from the simulation region, at atmospheric pressure and temperature. However, this condition cannot be satisfied over any significant period of time. Once the pressure in the simulation reaches somewhat less than the boundary pressure, the flow into and the flow out from the simulation will equalize. This will result in a compromise—DSMC will find a solution to the fluid flow that approximates your boundary conditions, but will not actually follow the boundary conditions, since the specified boundary conditions are inconsistent. In this manner, DSMC may be termed 'forgiving:' it finds a solution to the fluid flow

even if the specified boundary conditions are incorrect or inconsistent.

We observe that when DSMC disobeys the specified boundary conditions, velocity is the most affected quantity, but pressure and number density are also slightly affected. Temperature tends to be the least disobeyed.

In order to create particles in the ghost cells, the user must specify number density, temperature, and velocity (all three components of it) at every boundary position. Velocities of the particles in the ghost cells are distributed according to the 'drifting' local Maxwell-Boltzmann distribution. That is, at a particular location each component v_i of the velocities of the particles is independently distributed:

$$f(v_i) = \sqrt{\frac{m}{2\pi k_B T}} \exp\left(-\frac{1}{2}\frac{m}{k_B T}(v_i - v_{i0})^2\right) \tag{A.11}$$

where k_B is the Boltzmann constant, T is the local specified temperature, $i \in \{x, y, z\}$, and v_{i0} is the local specified flow velocity in the i direction. This makes the following joint distribution of particles' positions $\mathbf{r}$ and velocities $\mathbf{v}$:

$$f(\mathbf{r}, \mathbf{v}) = n(\mathbf{r})\sqrt{\frac{m}{2\pi k_B T(\mathbf{r})}} \exp\left(-\frac{1}{2}\frac{m}{k_B T(\mathbf{r})}\|\mathbf{v} - \mathbf{v}_0(\mathbf{r})\|^2\right) \tag{A.12}$$

where $n(\mathbf{r}$ is the specified number density as a function of position, $T(\mathbf{r})$ is the specified temperature as a function of position, and $v_0(\mathbf{r})$ is the specified flow velocity as a function of position.

A.3.1 'Adaptive' boundary conditions

Sometimes, a DSMC user knows certain aspects of a flow but not others. It is sometimes possible in DSMC to 'adapt' some of the boundary conditions to the flow as it is happening. For example, maybe the pressure, temperature, and number density are known, but not the flow velocity as a function of position. In some cases like this, it may be possible to have the ghost cells be populated according to the known pressure, temperature, and number density distribution, but use the flow velocity from

the simulation region to specify that aspect of the gas distribution in the ghost cells. To avoid instability, adaptive boundary conditions need to be averaged over a few thousand time steps before updating. The larger the simulation region, the longer the averaging period between adaptive updates needs to be.

A.3.2 Flux

When designing boundary conditions, it can be useful to keep in mind that the total flux into and out of the simulation must be zero. This can help prevent instability in adaptive boundary conditions, and can help ensure that the specified velocities are obeyed, and that the specified pressures and number densities are obeyed.

A.4 Solid Interaction Models.

It is usually of interest in a fluid flow to have solid objects in the flow. There are three models of solid interaction with fluid particles discussed here.

Specular Reflection

The first solid-interaction model is specular reflection. This model treats objects as particle mirrors. If a particle hits an object, its velocity is just reflected off the object like a mirror reflects light. This model does not allow an object to have a temperature, and corresponds to a frictionless surface into which no heat can flow (thermal insulation).

A.4.1 Thermalization

The second solid-interaction model is thermalization. This model allows objects to have a temperature, and transmit temperature information to particles. This model

treats an object surface as if it were a planar boundary with a reservoir of gas with a perfect Maxwellian distribution of particles, at the specified object temperature. Given this definition, when a particle hits an object, its velocity is randomly reassigned according to a distribution that is related to the Maxwell distribution. The distribution of the particle's new velocity is the distribution of velocities of particles that cross an imaginary plane surrounded by a Maxwellian distribution of particles—in one direction only. The parallel components of the distribution are identical to that of a Maxwell distribution. The perpendicular component of the distribution is proportional to the radial component of a two-dimensional Maxwell distribution.

A.4.2 Linear Combinations.

It is possible to define an interaction model as a linear combination of the specular and thermalization models. A combination model seems especially appropriate in a rarefied gas situation, where a particle bouncing off an object may thermalize somewhat, but may also specularly reflect somewhat. In this case, some parameter p ranging from 0 to 1 could indicate how much of each model is being used.

A.4.3 Other Models.

Other models can also be envisioned. For example, several types of diffuse reflection may be appropriate, depending on the roughness of the surface and if the surface is more rough in some directions than others. An absorption model could also be necessary, should the material be slightly porous or have an affinity for the gas. Since these other models are not needed for our problem, they are not further discussed here.

Appendix B

Figures

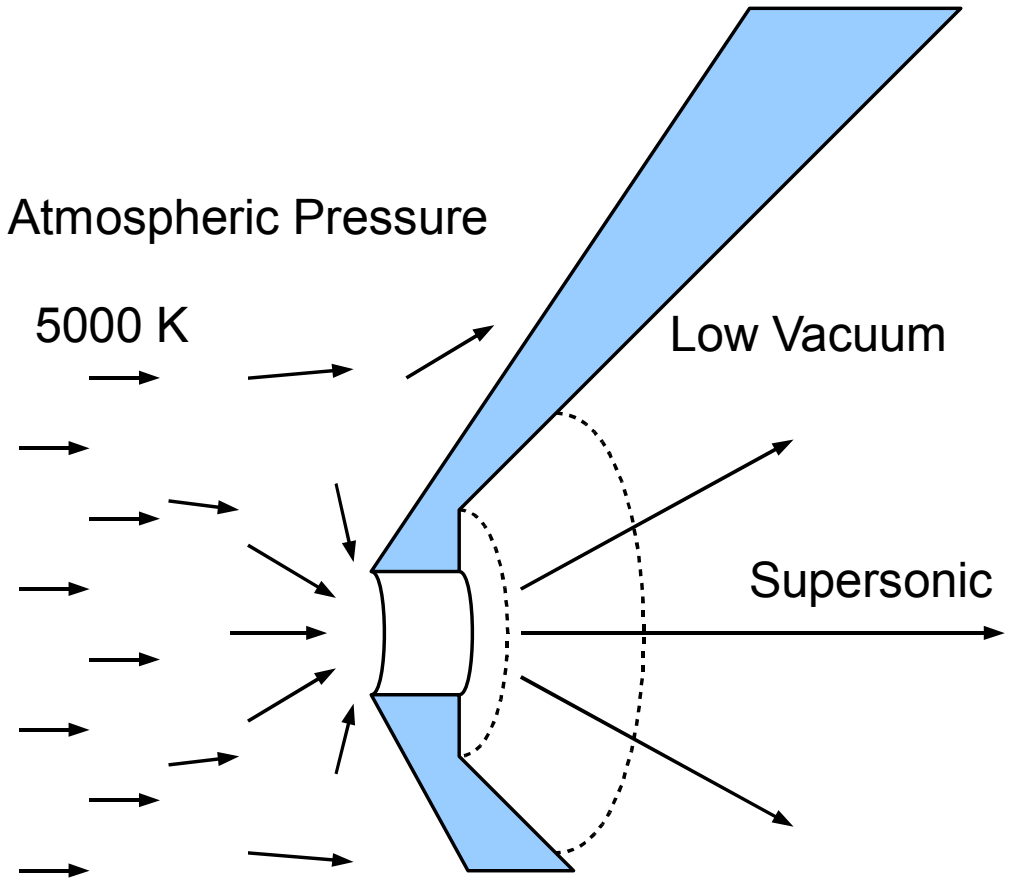

Figure B.1 Diagram of the sampling cone

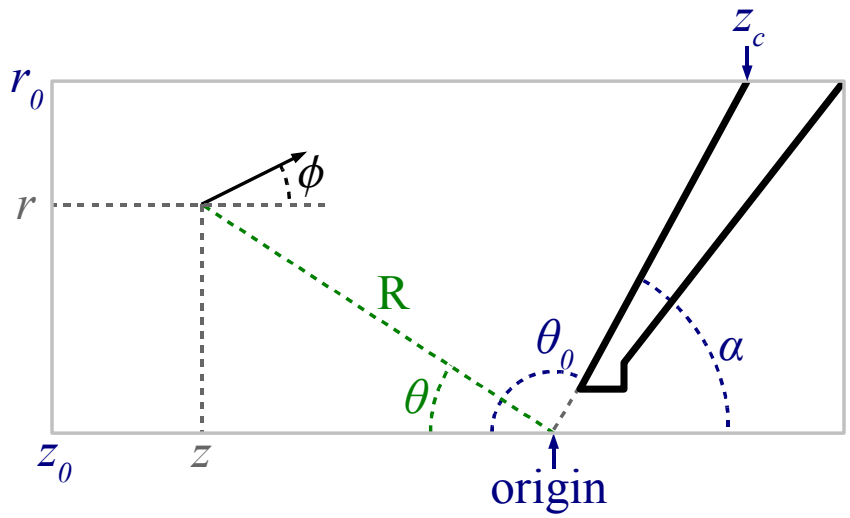

Figure B.2 Diagram of the sampling cone with parameter labels

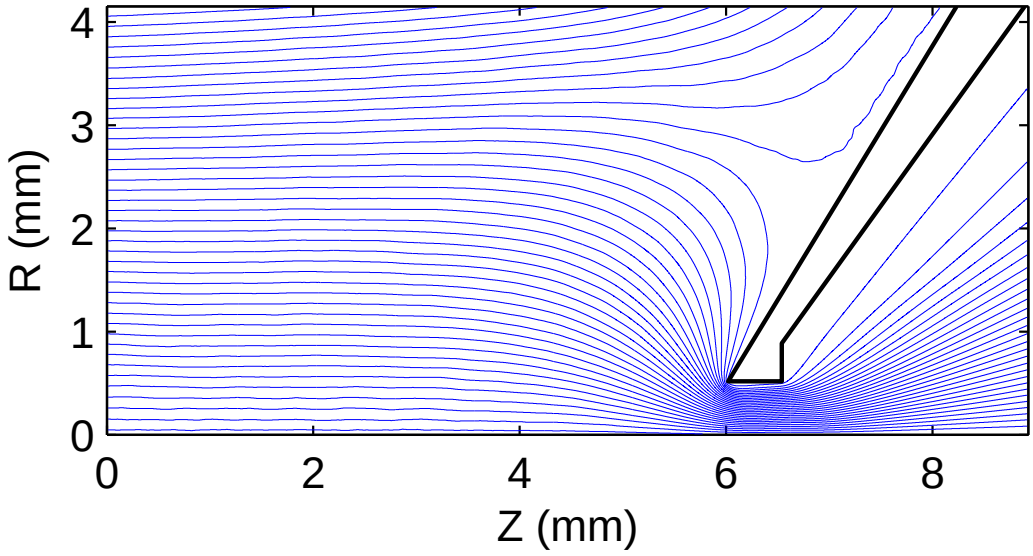

Figure B.3 Streamlines of the gas flow

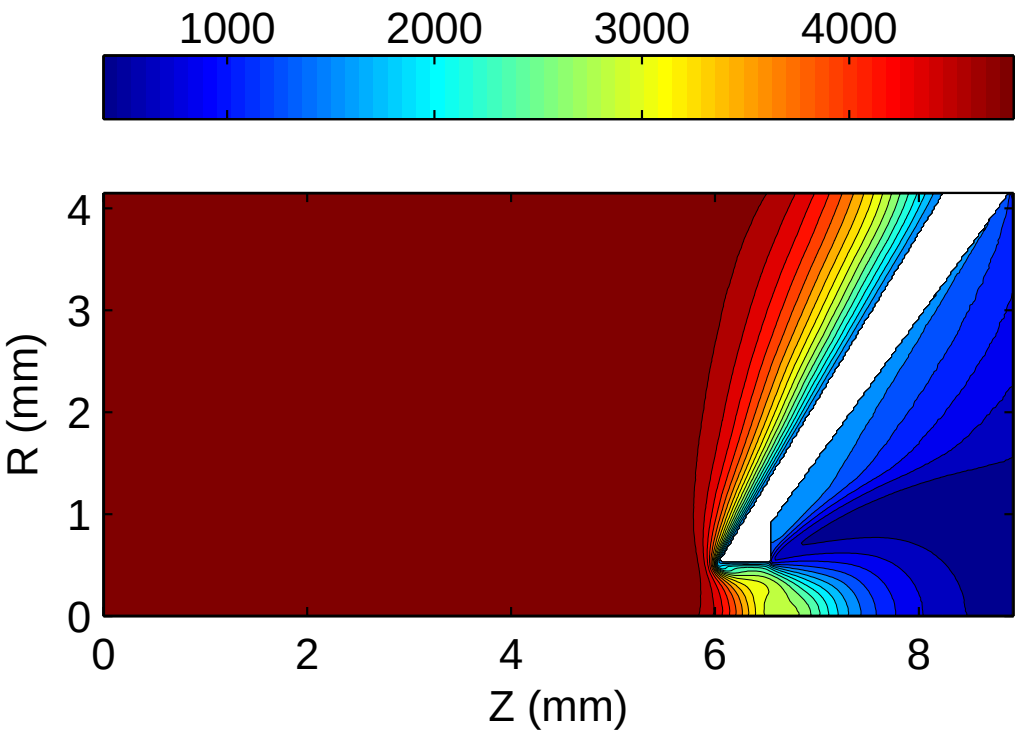

Figure B.4 Temperature in K

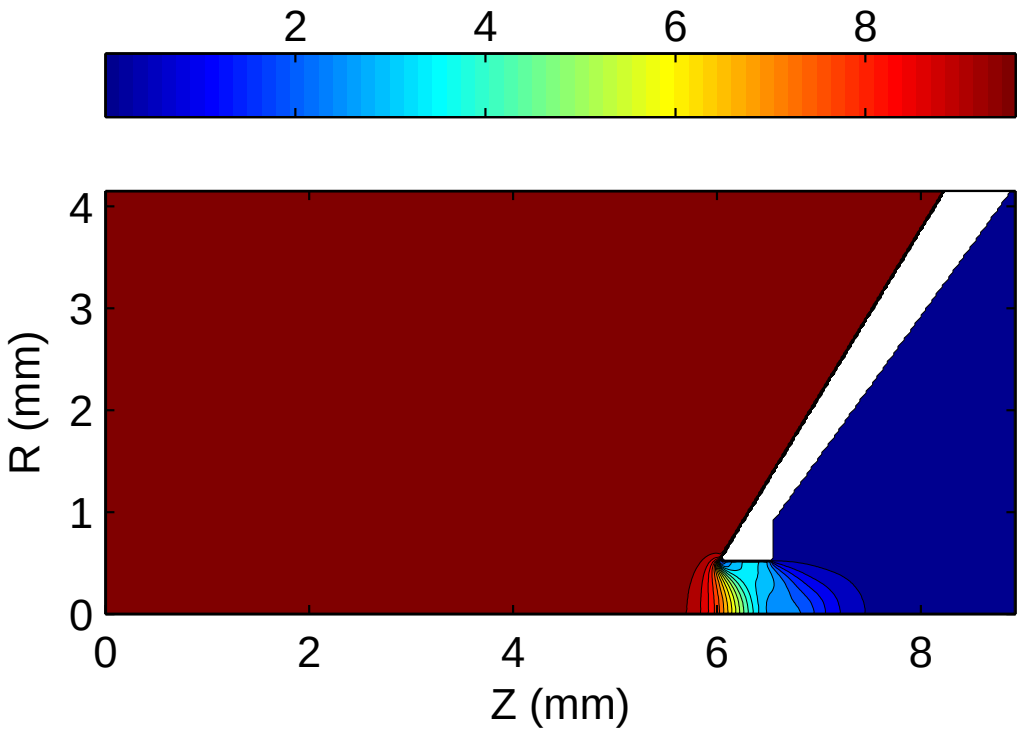

Figure B.5 Pressure in 10^4 Pa

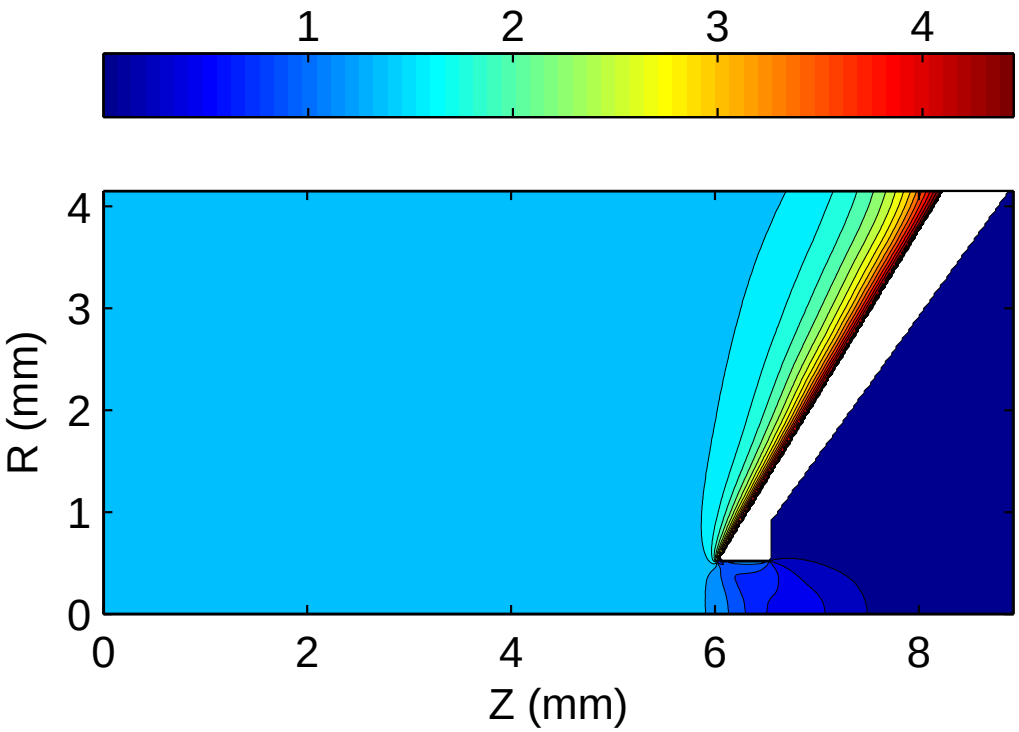

Figure B.6 Number density in $10^{24}\,\mathrm{m}^{-3}$

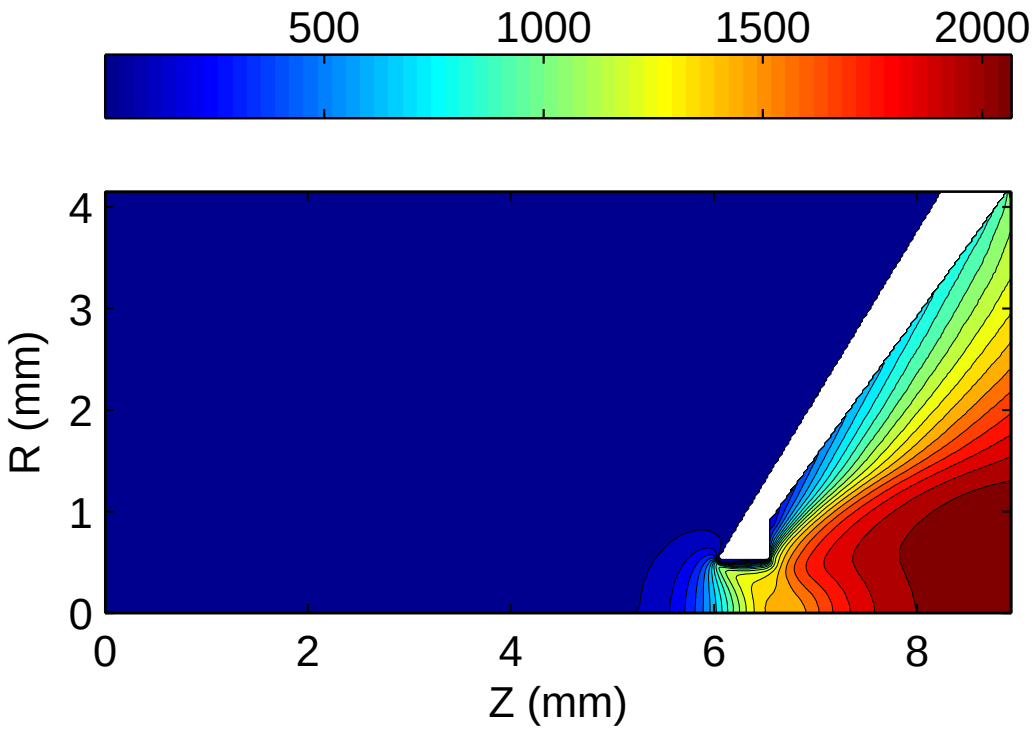

Figure B.7 Flow speed in m/s

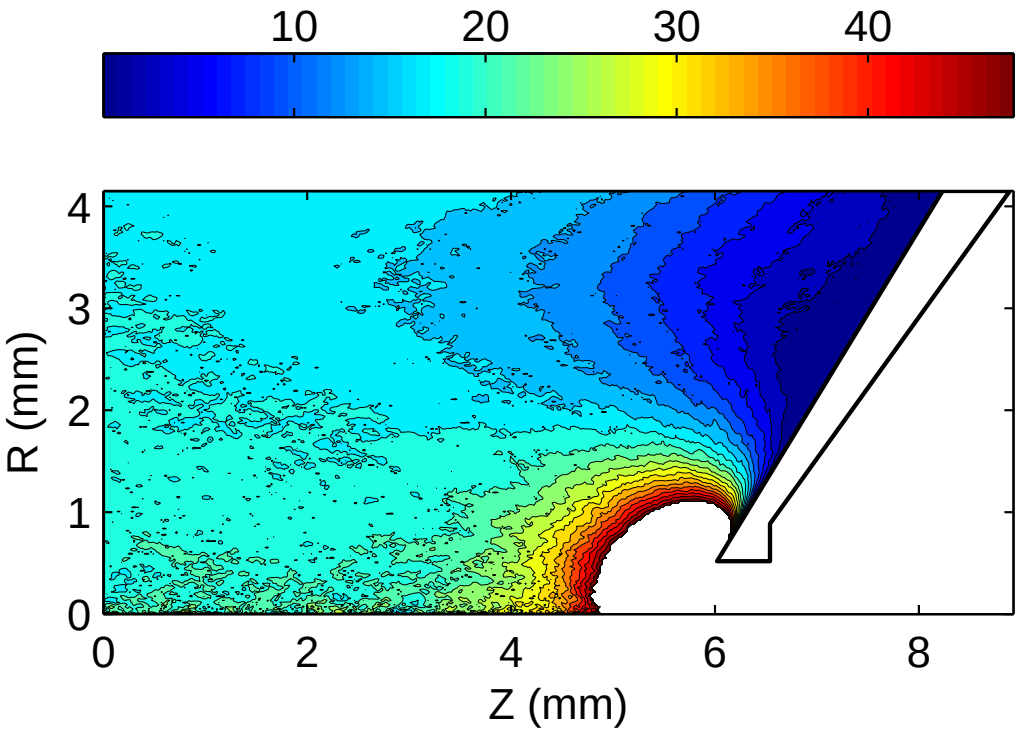

Figure B.8 Flow speed in m/s (slow-moving gas only)

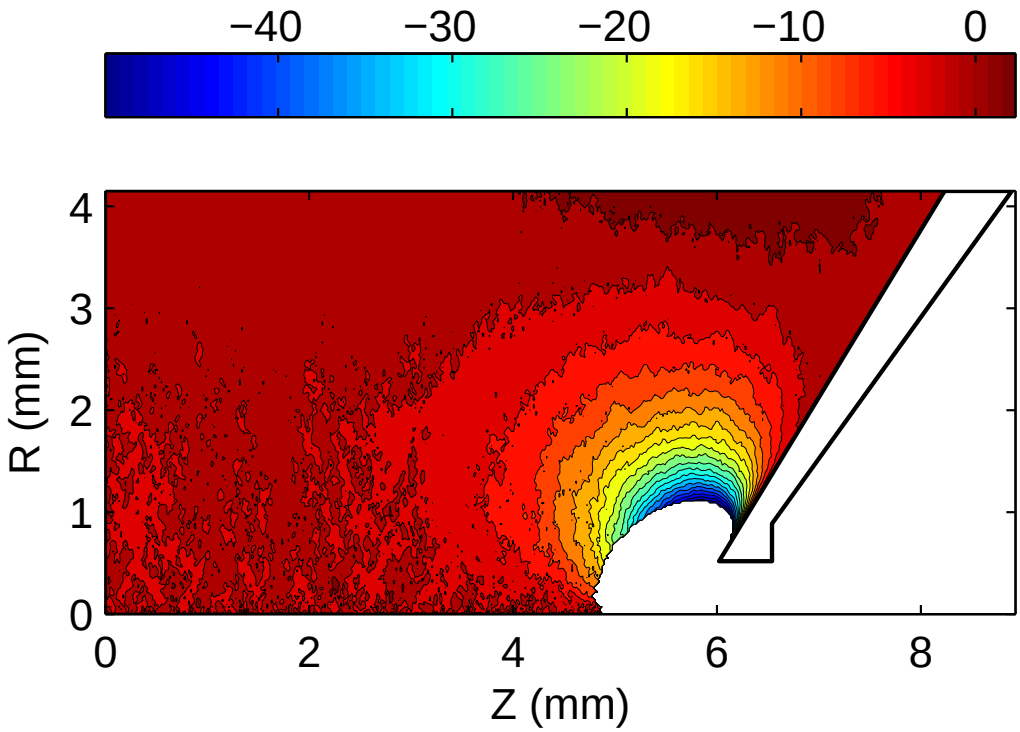

Figure B.9 v_r in m/s (slow-moving gas only)

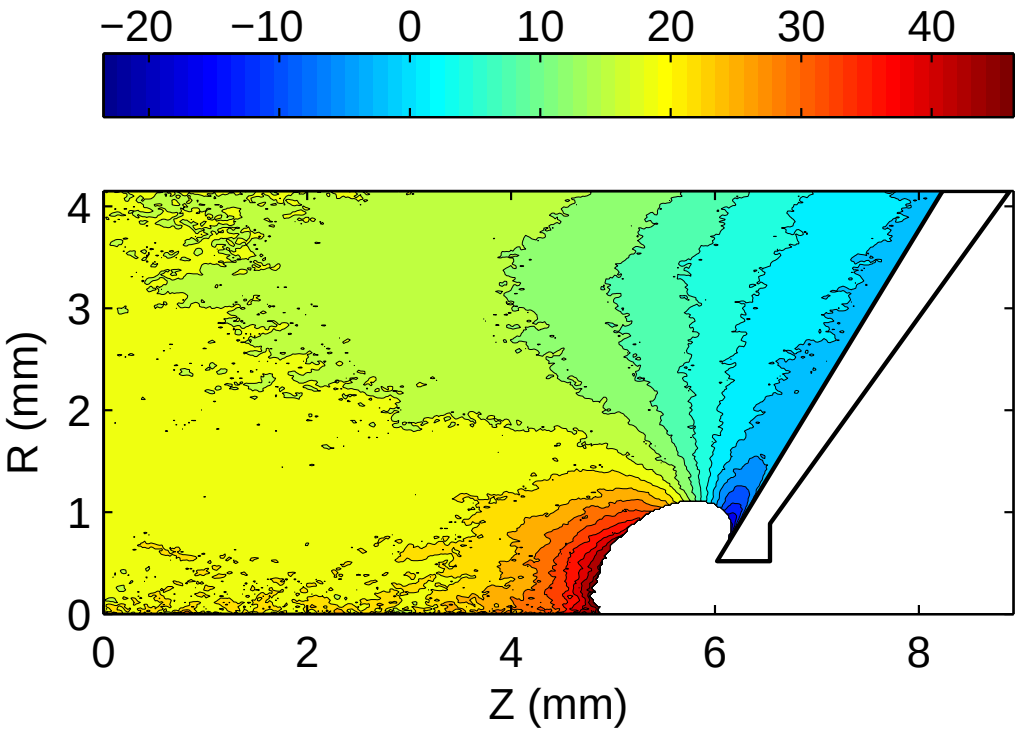

Figure B.10 v_z in m/s (slow-moving gas only)

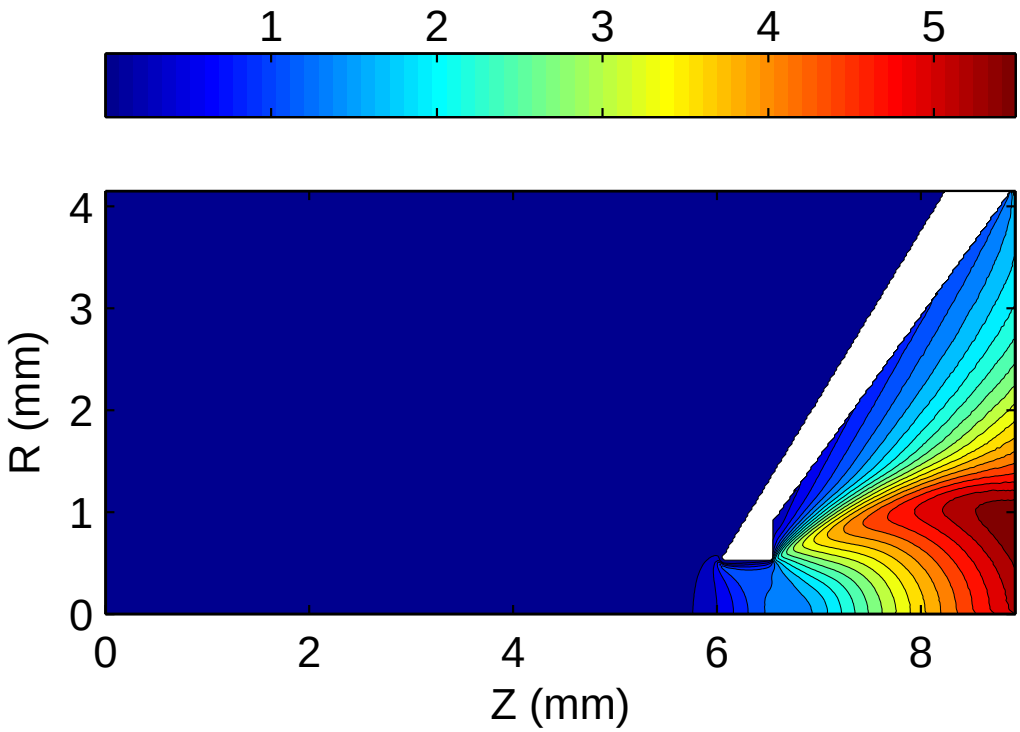

Figure B.11 Mach number

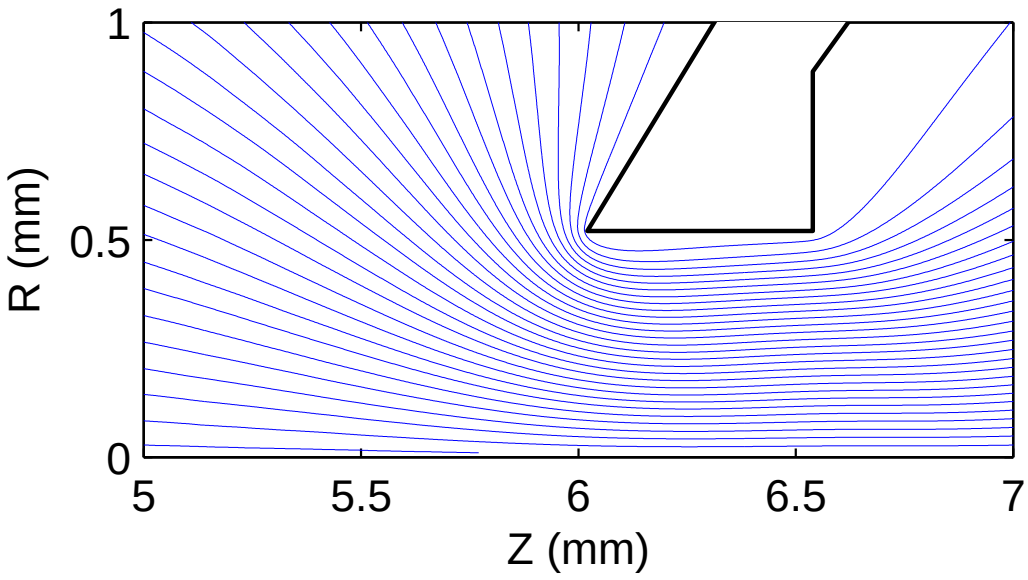

Figure B.12 Streamlines near the nozzle

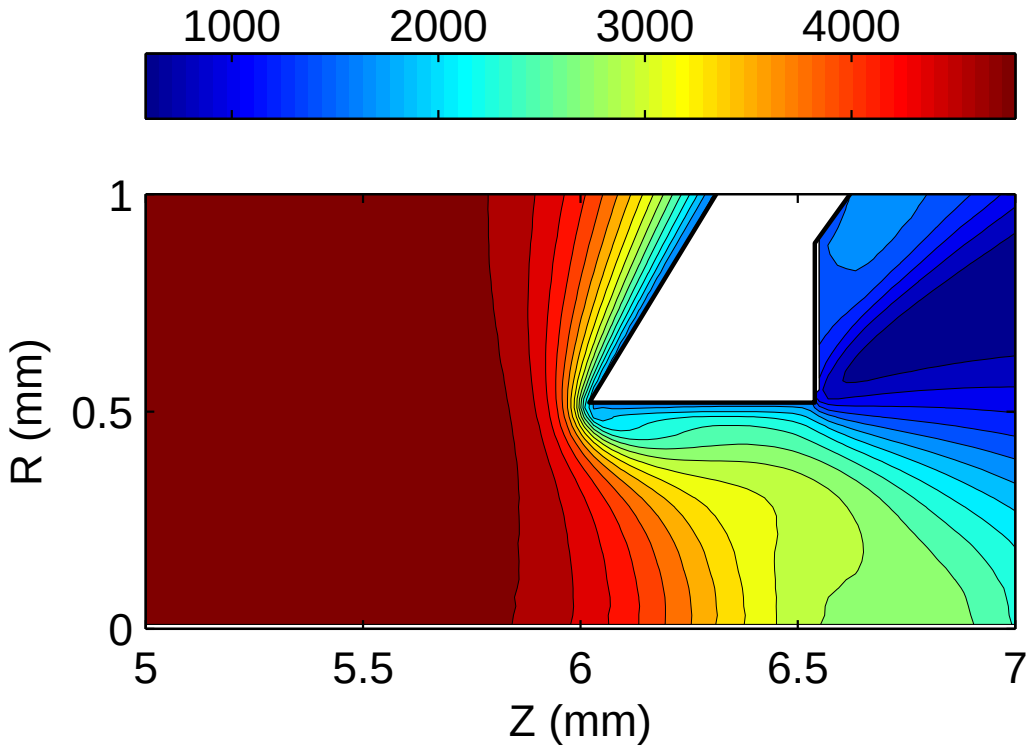

Figure B.13 Temperature in K near the nozzle

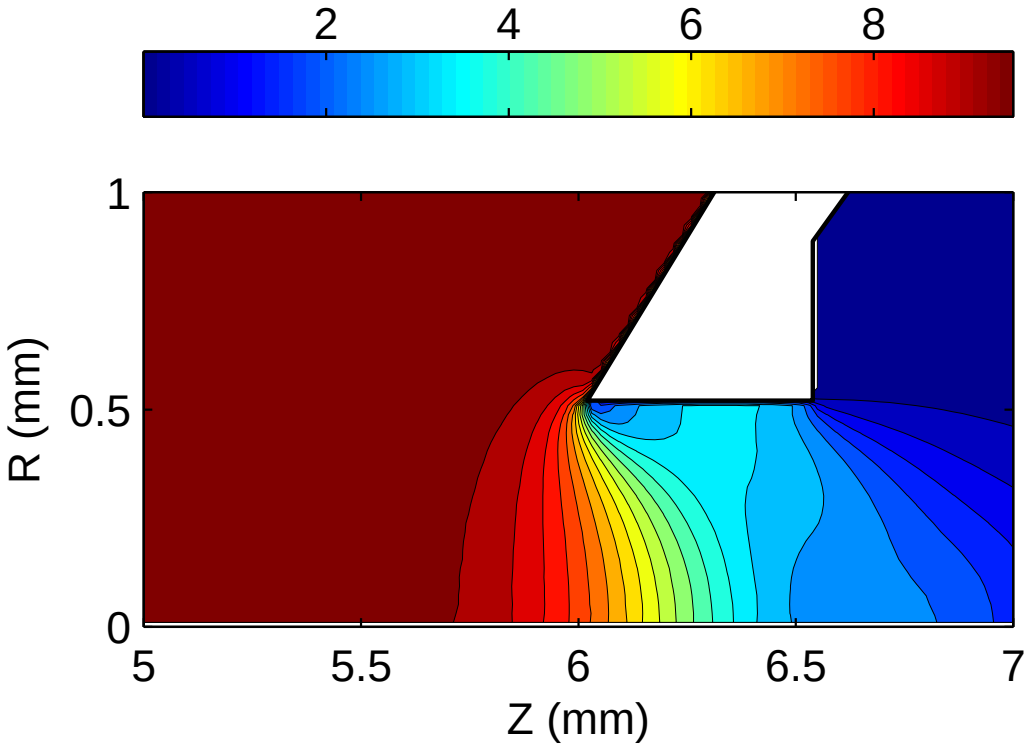

Figure B.14 Pressure in 10^4 Pa near the nozzle

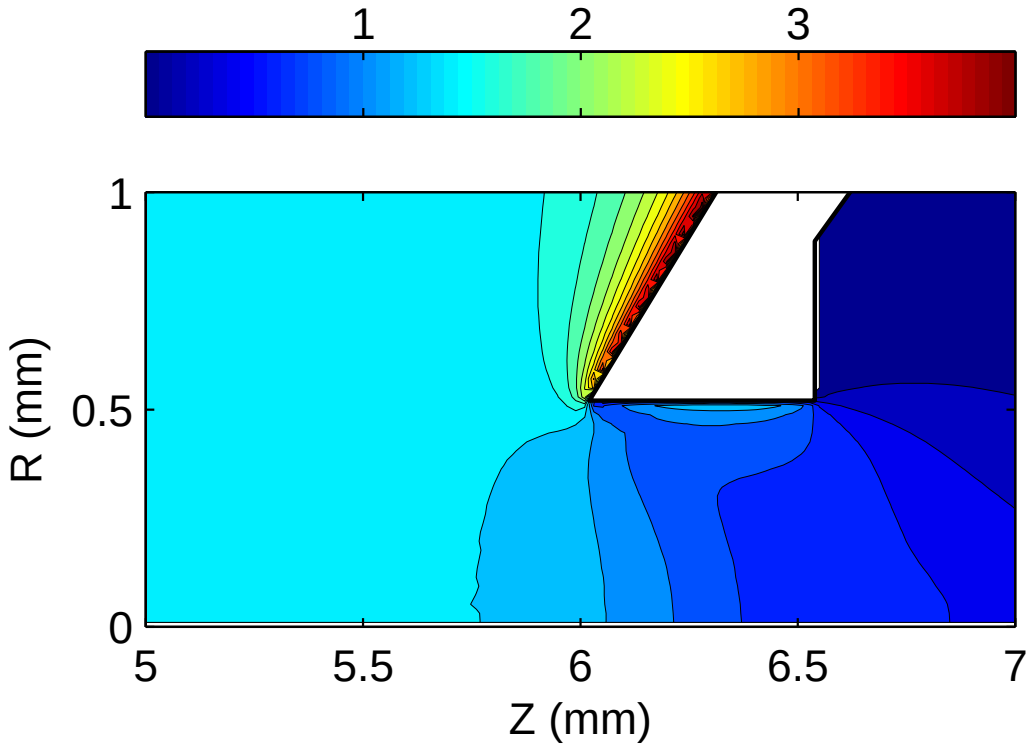

Figure B.15 Number density in $10^{24}\,\mathrm{m}^{-3}$ near the nozzle

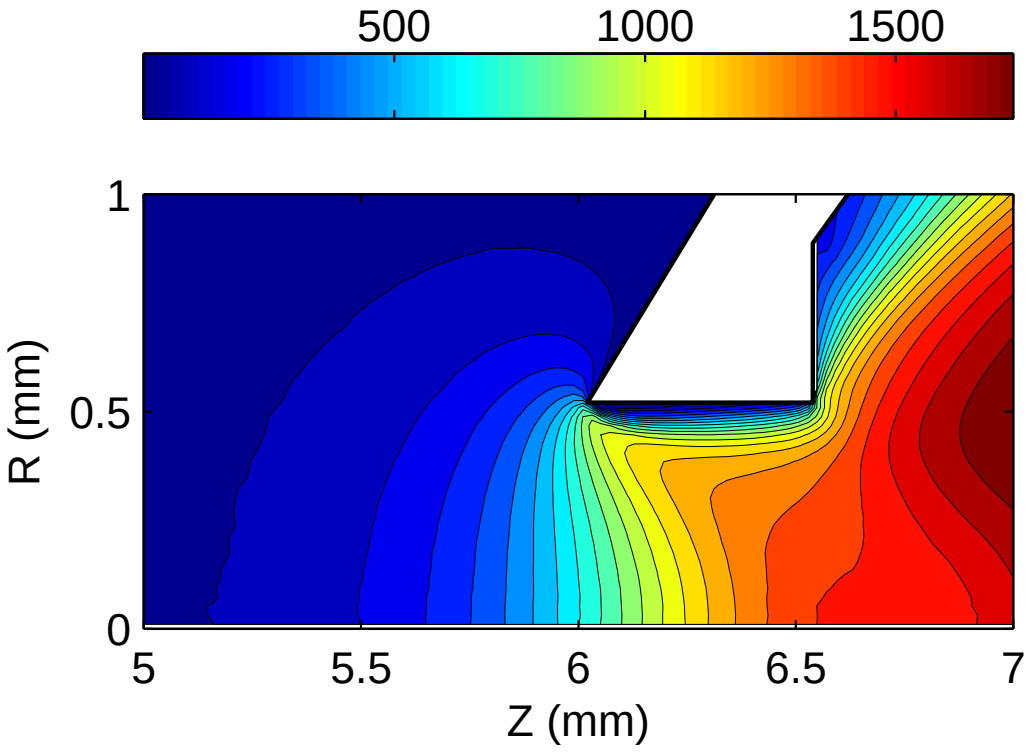

Figure B.16 Flow speed in m/s near the nozzle

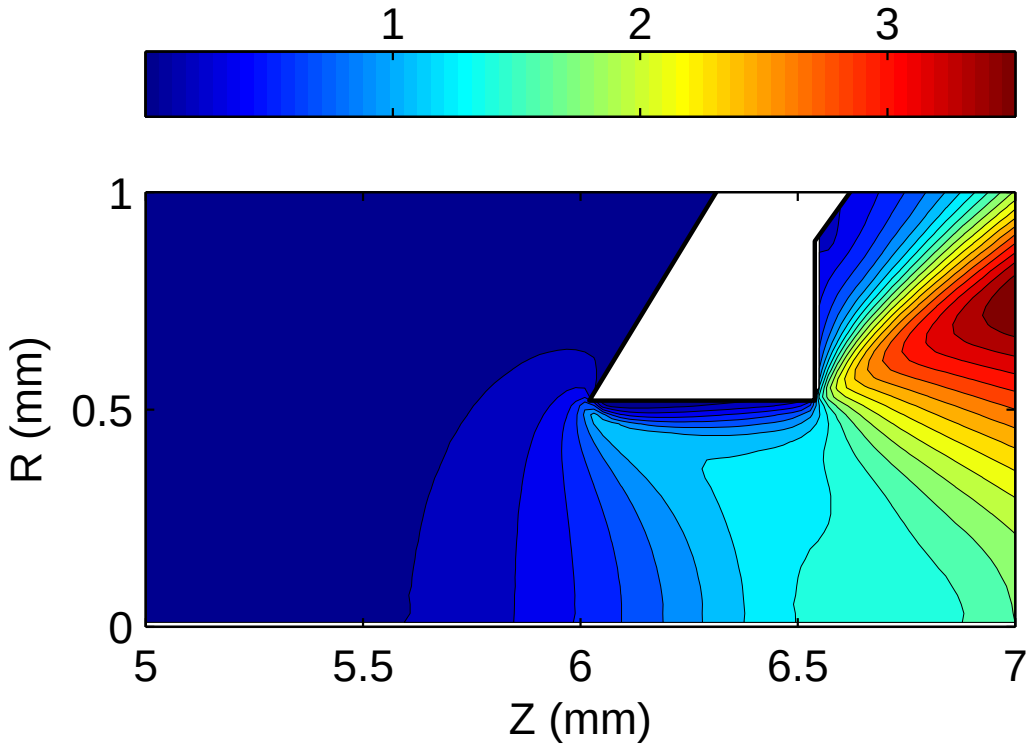

Figure B.17 Mach number near the nozzle

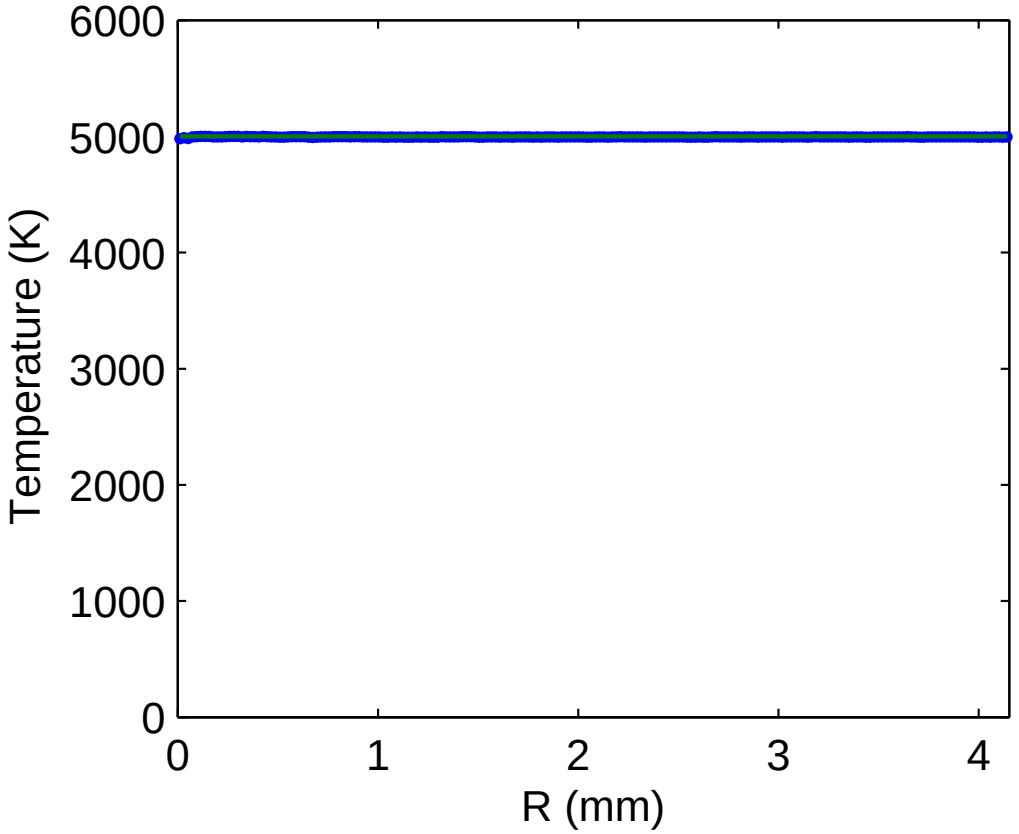

Figure B.18 Temperature on the left boundary ($z = z_0$); solid line the requested boundary condition

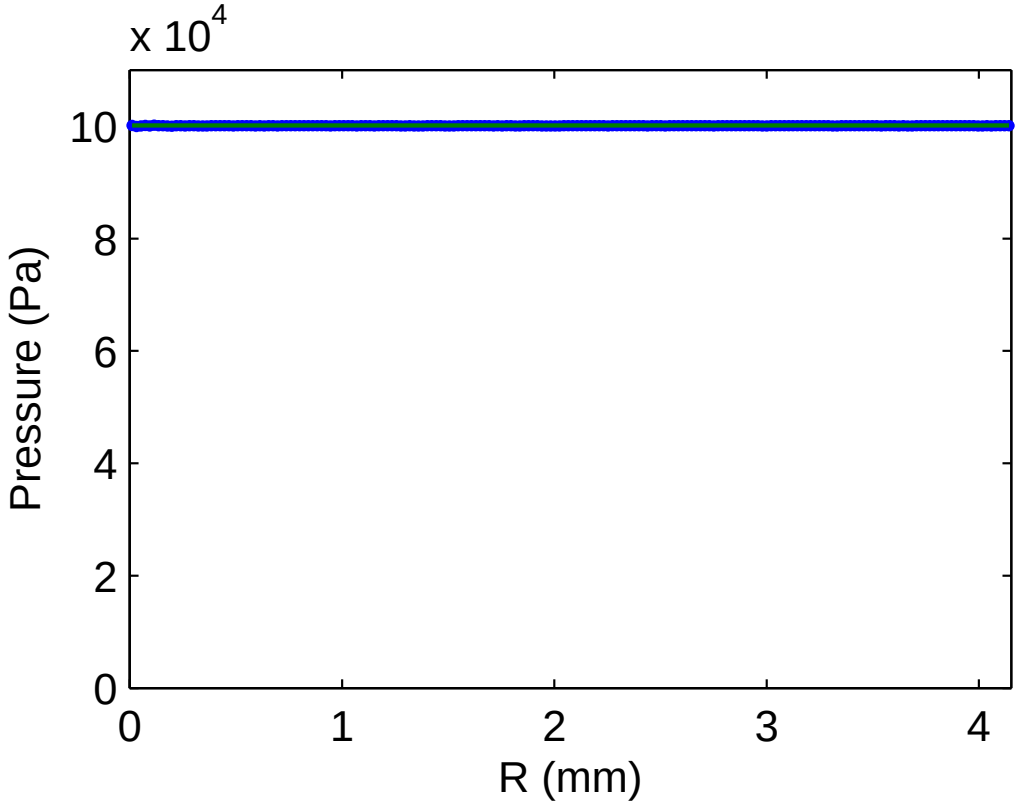

Figure B.19 Pressure on the left boundary ($z = z_0$); solid line the requested boundary condition

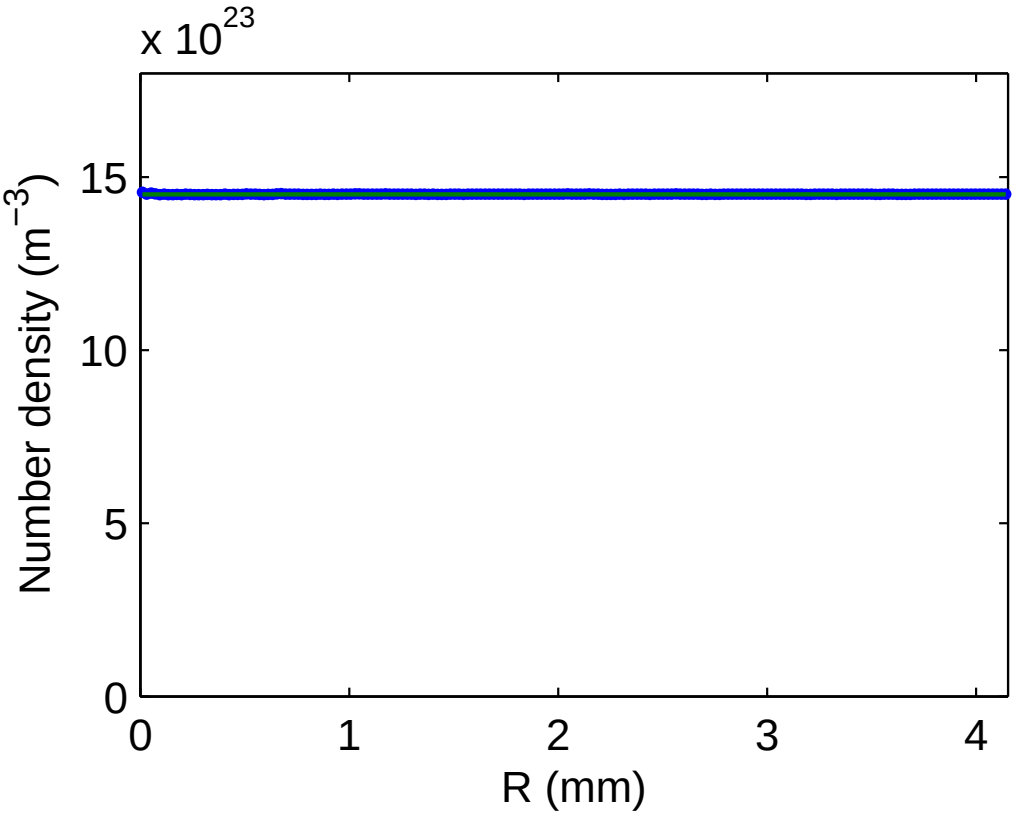

Figure B.20 Number density on the left boundary ($z = z_0$); solid line the requested boundary condition

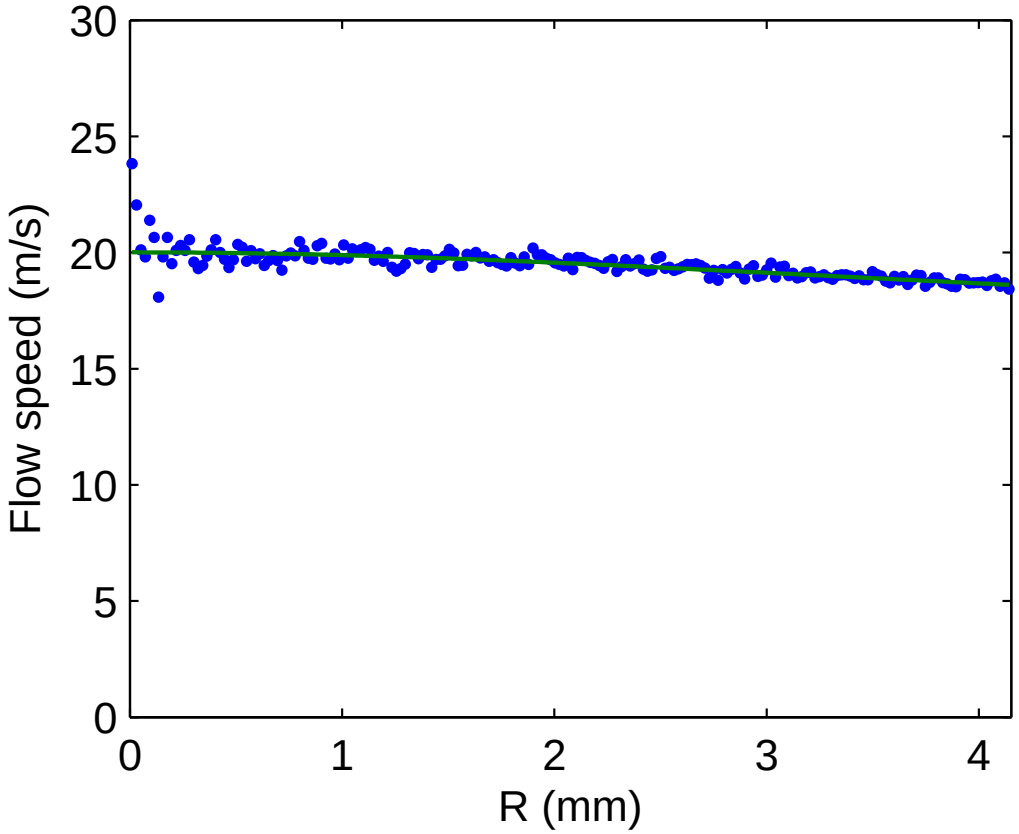

Figure B.21 Speed on the left boundary ($z = z_0$); solid line the requested boundary condition

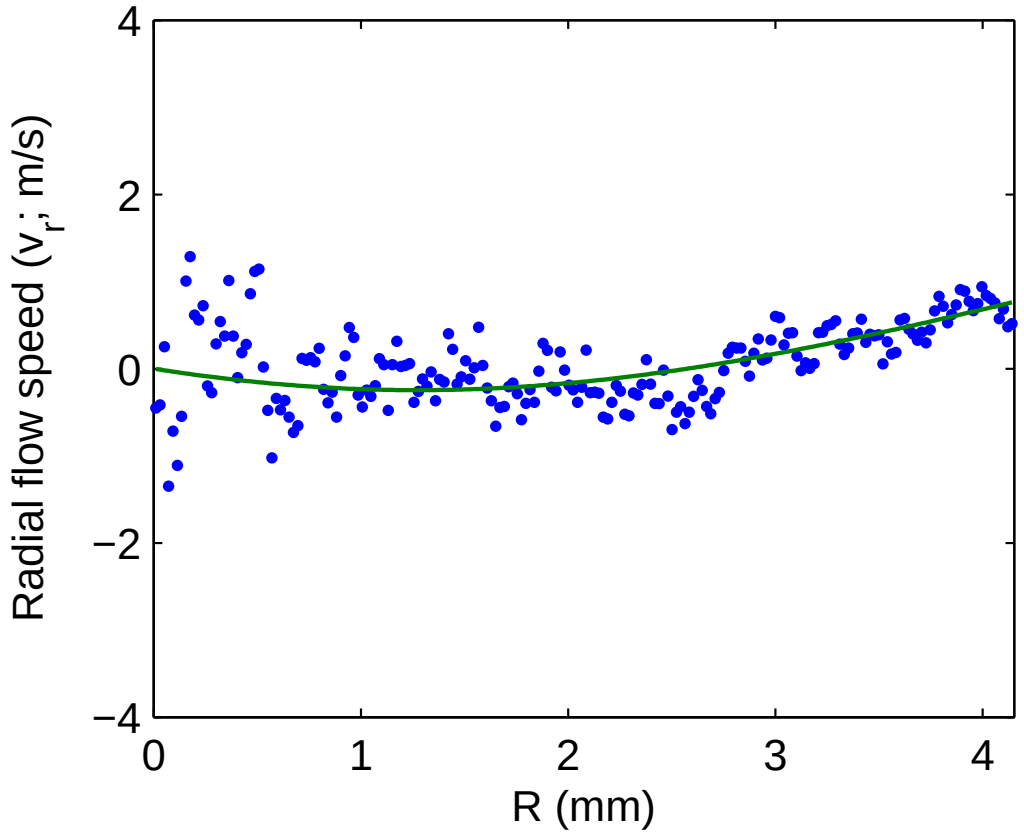

Figure B.22 Radial speed (v_r) on the left boundary ($z = z_0$); solid line the requested boundary condition

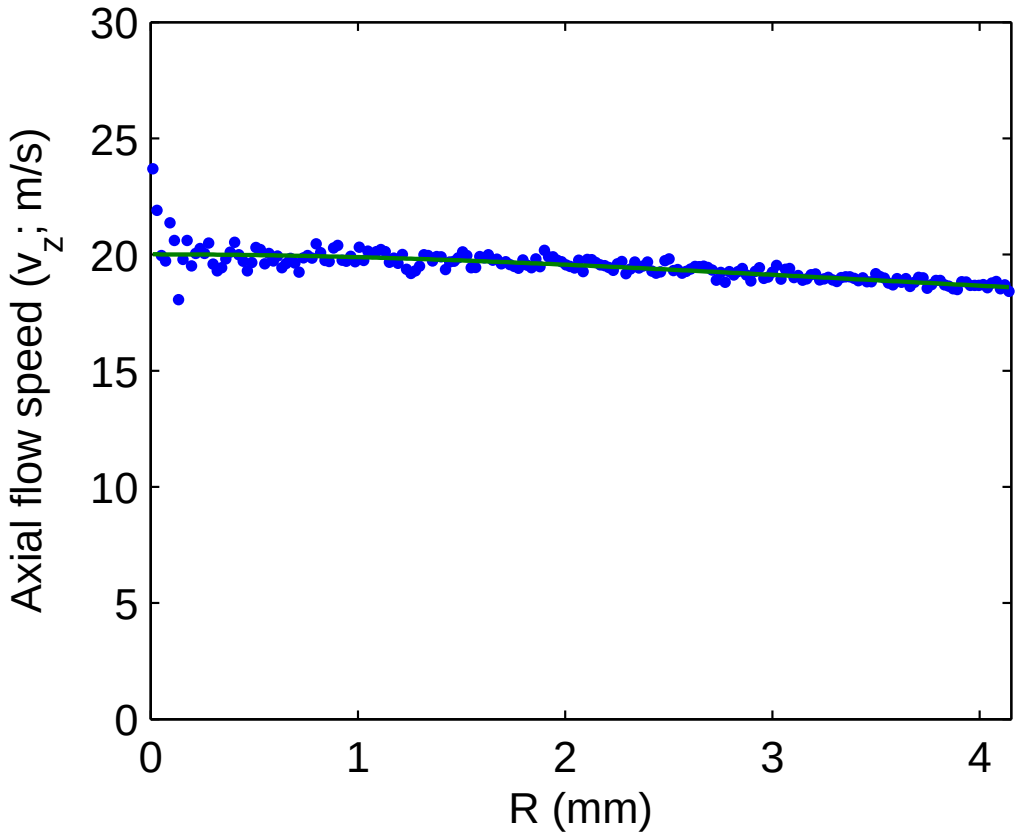

Figure B.23 Axial speed (v_z) on the left boundary ($z = z_0$); solid line the requested boundary condition

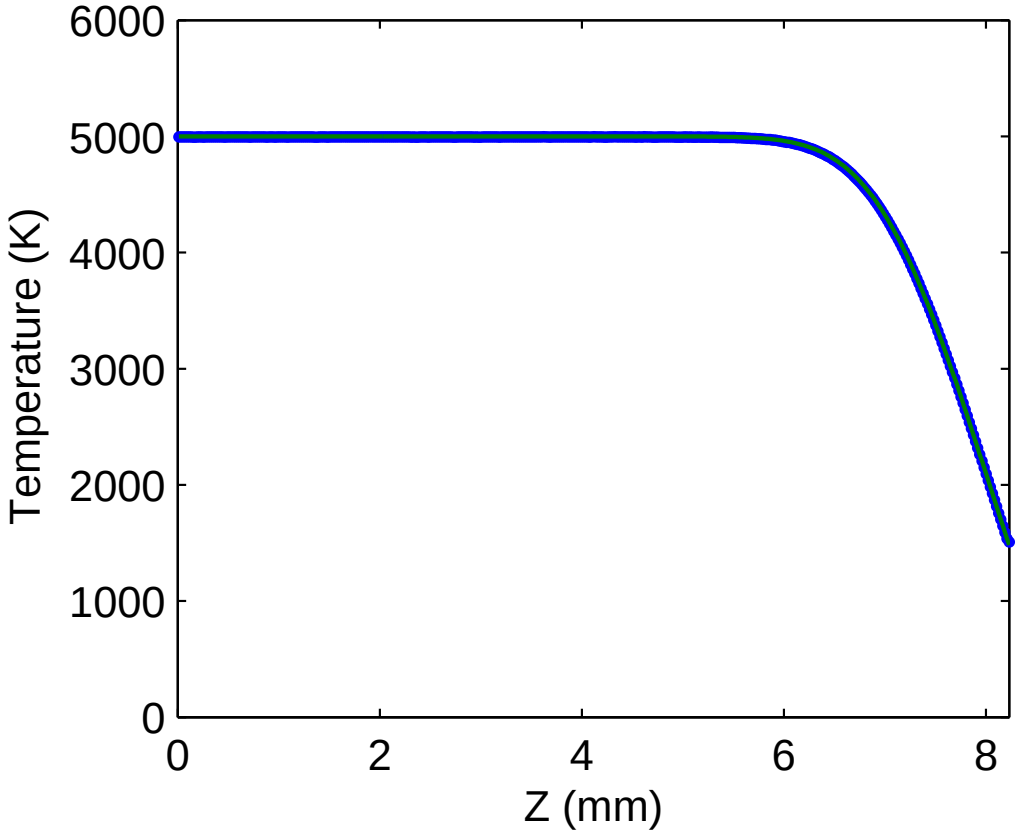

Figure B.24 Temperature on the top boundary ($r = r_0$); solid line the requested boundary condition

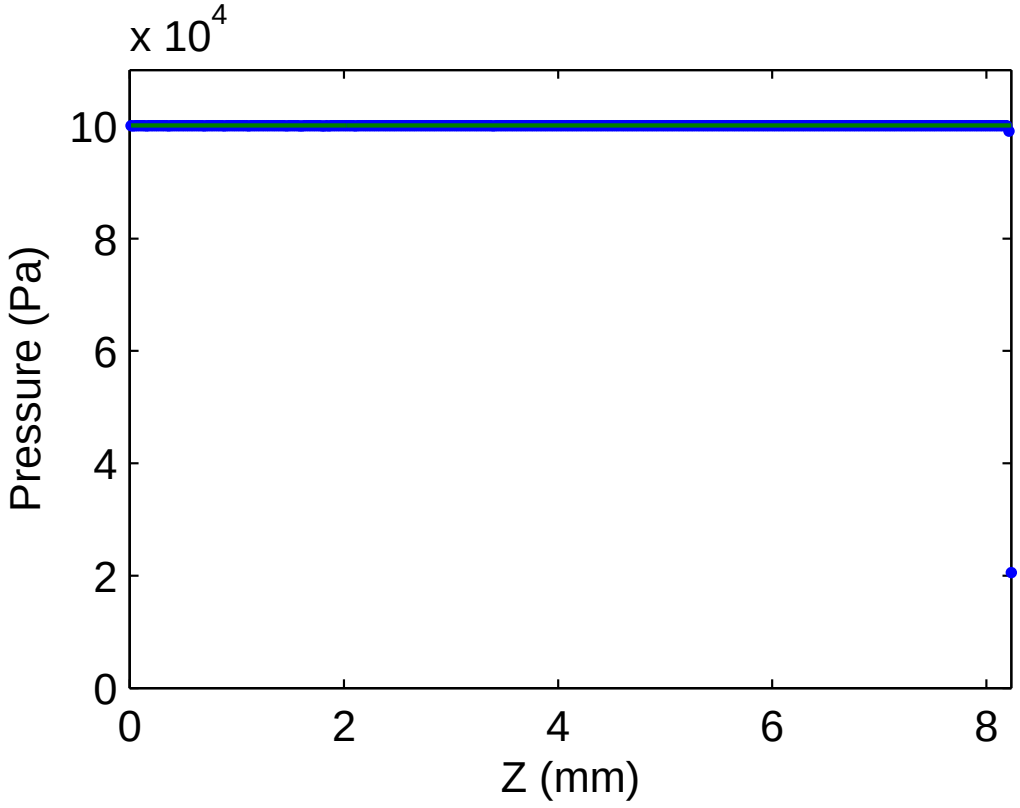

Figure B.25 Pressure on the top boundary ($r = r_0$); solid line the requested boundary condition

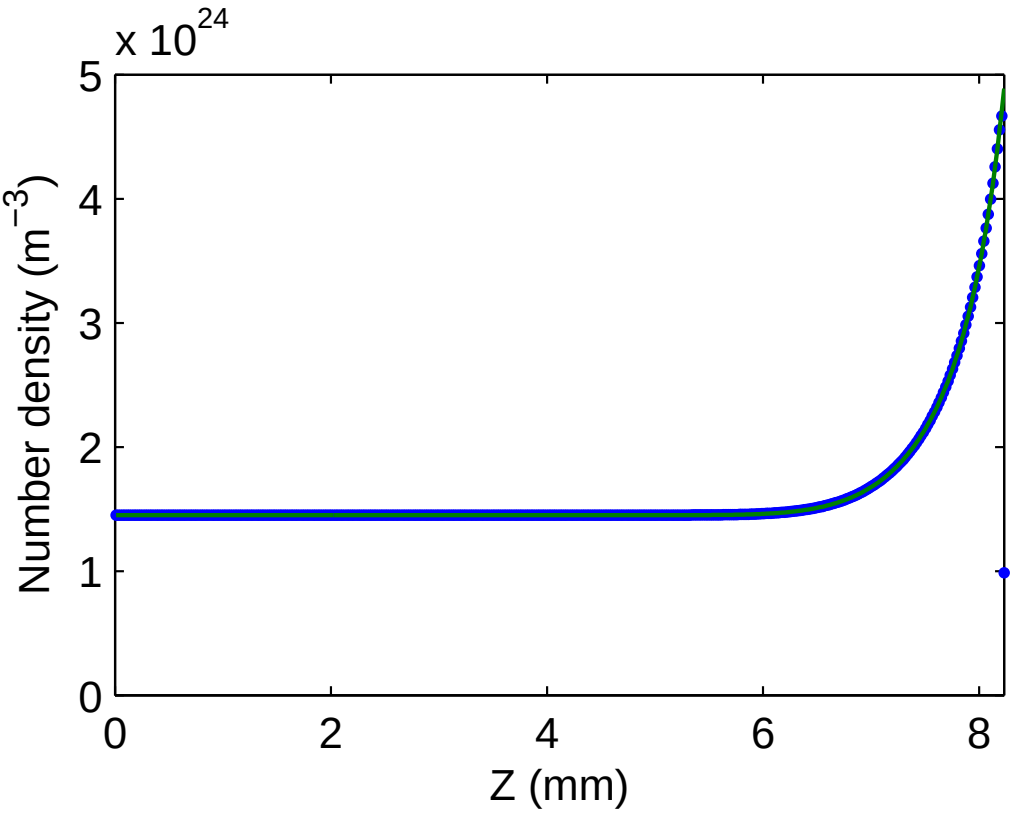

Figure B.26 Number density on the top boundary ($r = r_0$); solid line the requested boundary condition

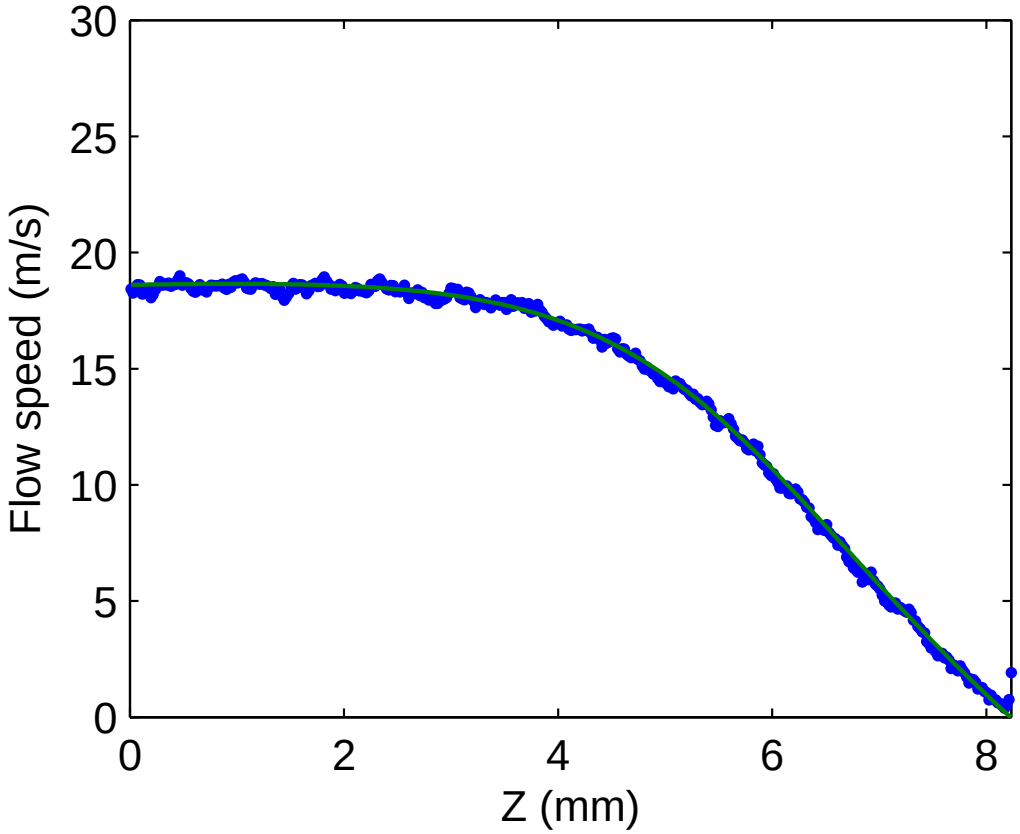

Figure B.27 Speed on the top boundary ($r = r_0$); solid line the requested boundary condition

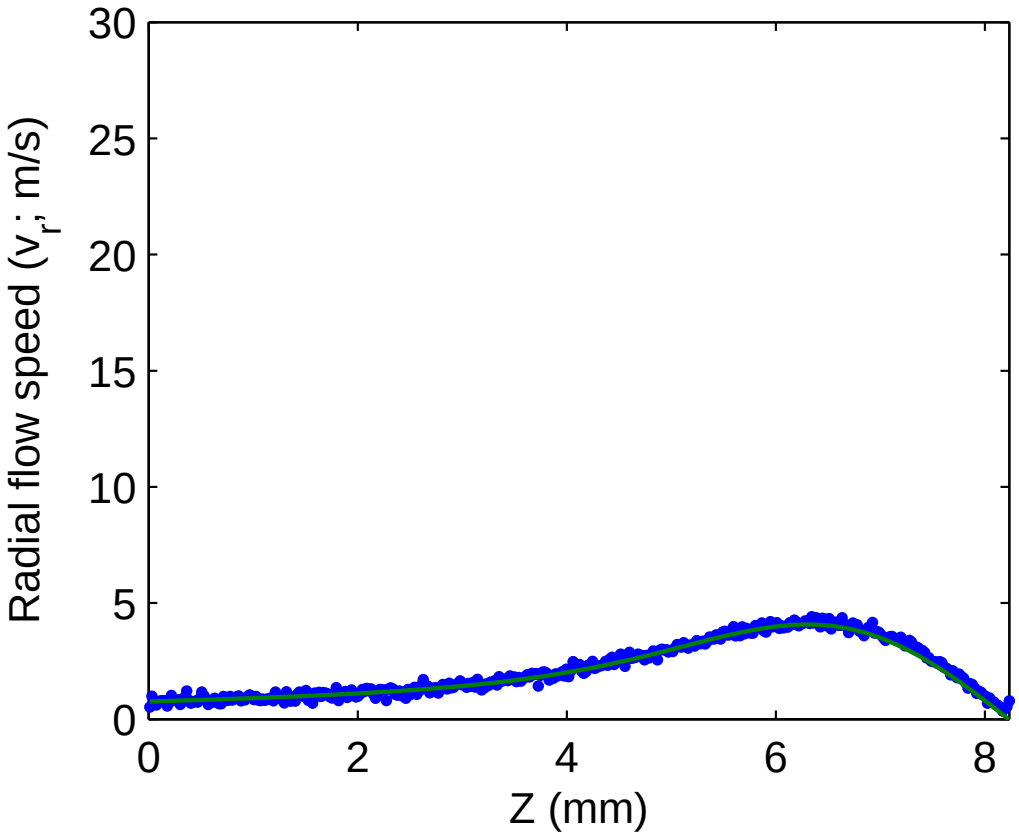

Figure B.28 Radial speed (v_r) on the top boundary ($r = r_0$); solid line the requested boundary condition

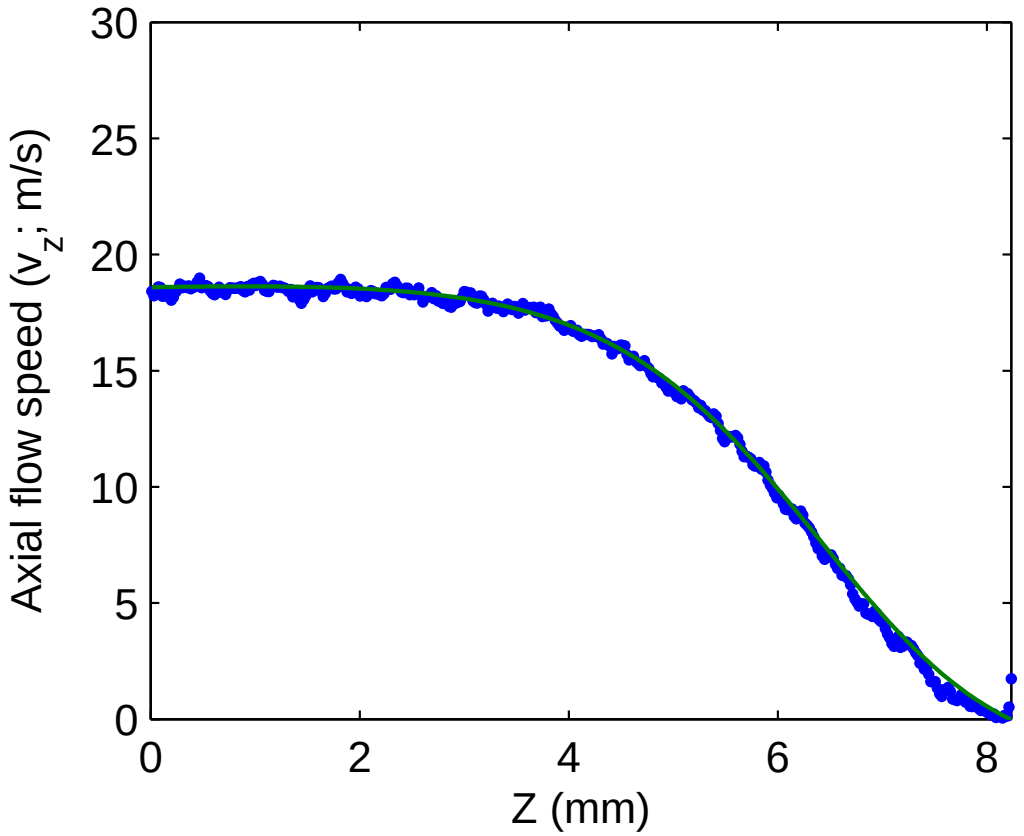

Figure B.29 Axial speed (v_z) on the top boundary ($r = r_0$); solid line the requested boundary condition

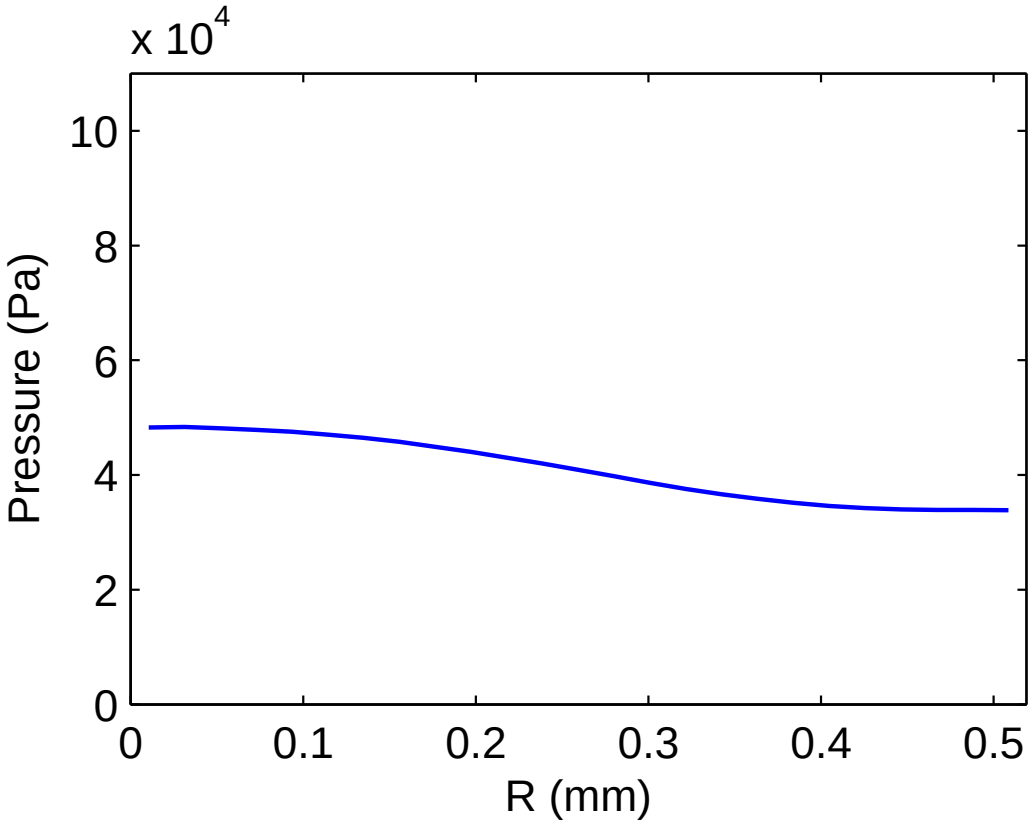

Figure B.30 Pressure at $z = 6.25914\,\text{mm}$ (halfway through the nozzle)

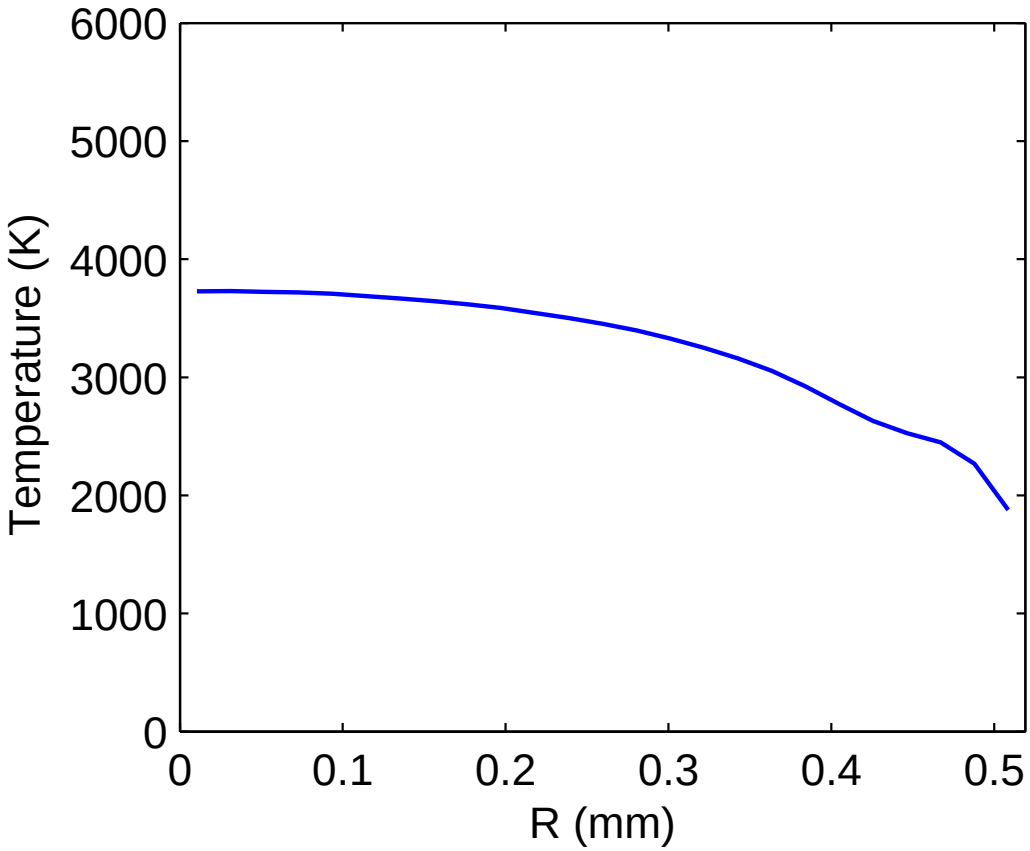

Figure B.31 Temperature at $z = 6.25914\,\mathrm{mm}$ halfway through the nozzle)

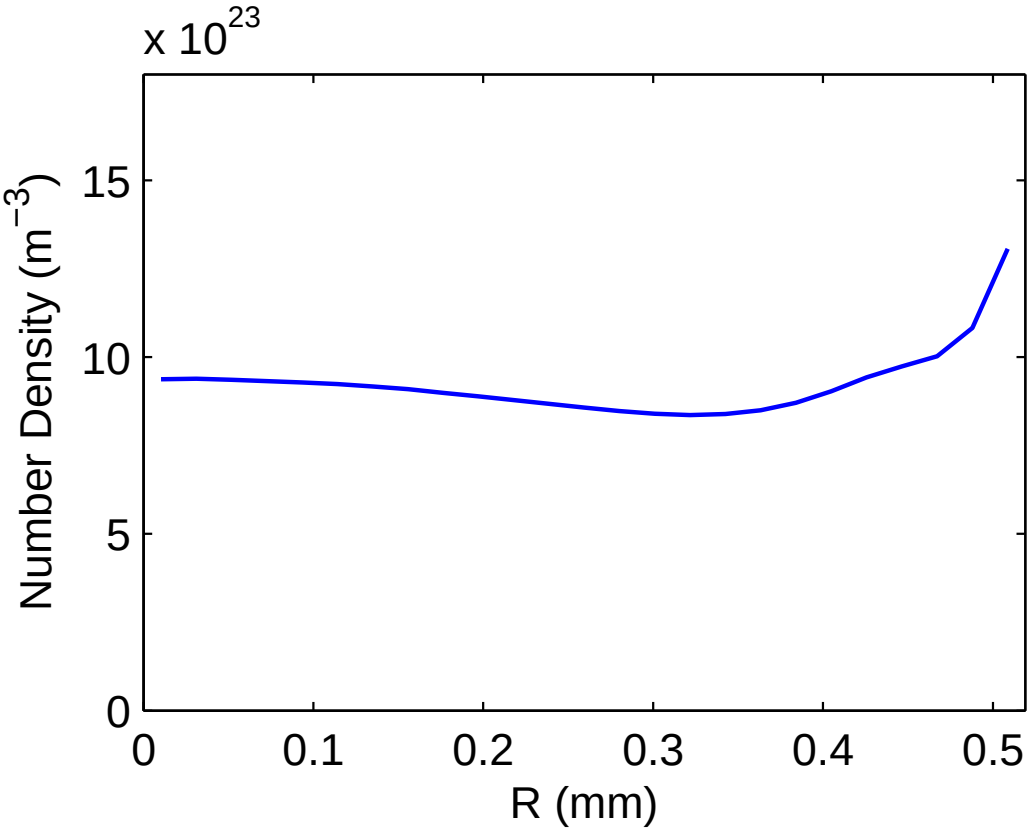

Figure B.32 Number density at $z = 6.25914\,\text{mm}$ (halfway through the nozzle)

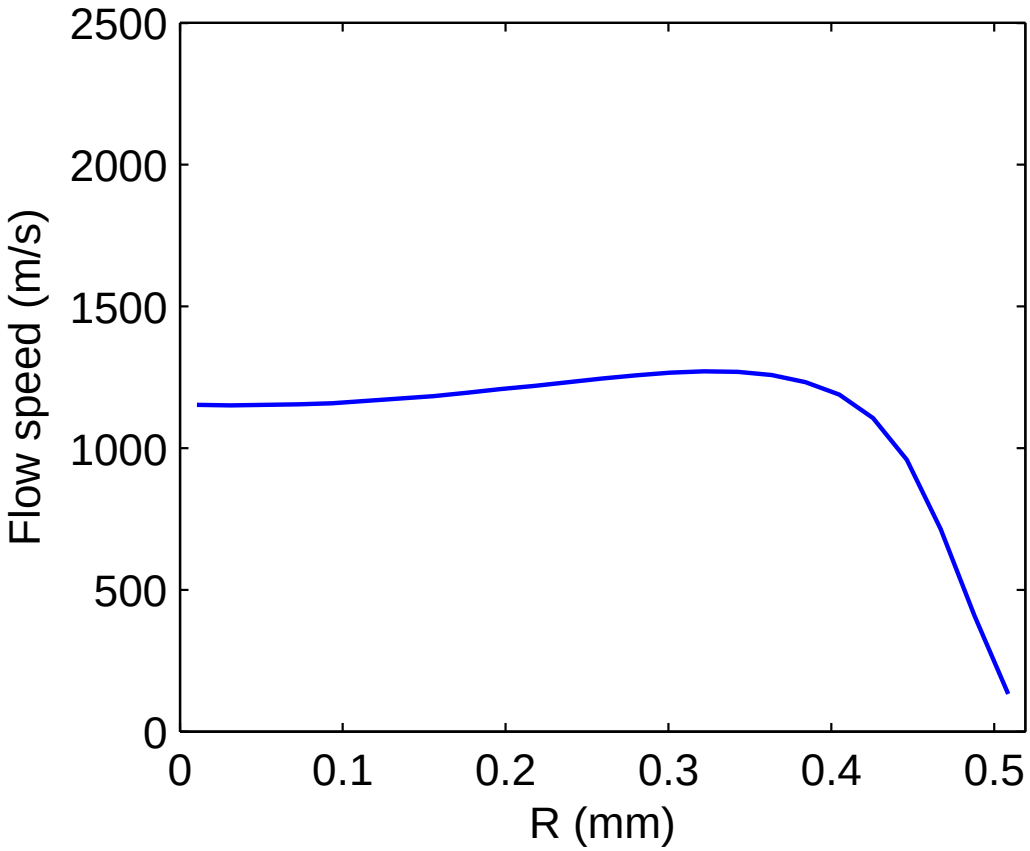

Figure B.33 Flow speed at $z = 6.25914\,\mathrm{mm}$ (halfway through the nozzle)

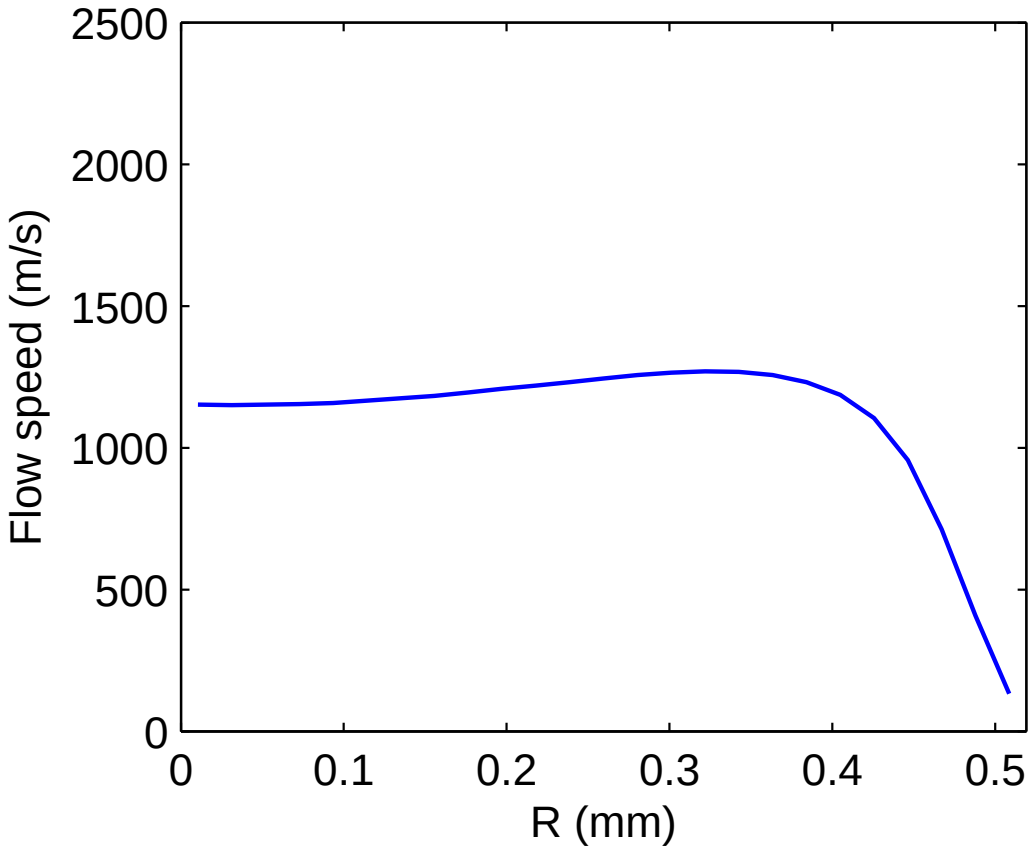

Figure B.34 Axial speed (v_z) at $z = 6.25914\,\text{mm}$ (halfway through the nozzle)

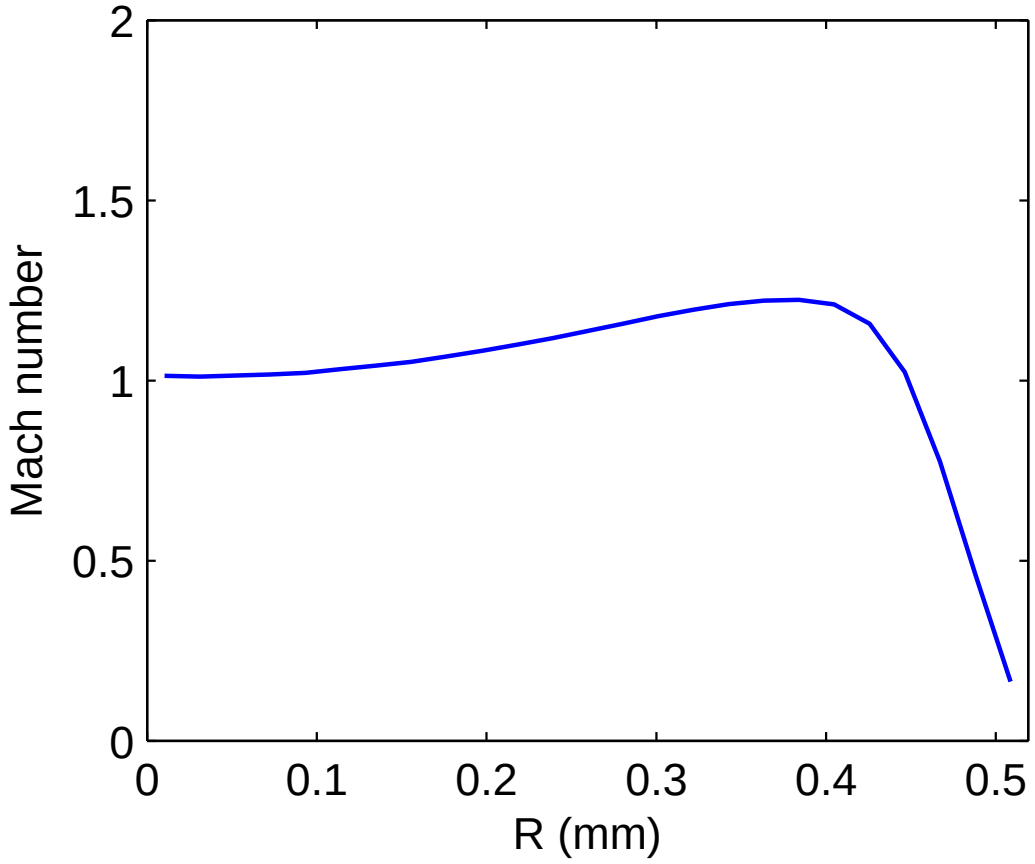

Figure B.35 Mach number at $z = 6.25914\,\mathrm{mm}$ (halfway through the nozzle)