A New Algorithm for the Neutral Gas in the Free-Molecule Regimes of Hall and Ion Thrusters

IEPC-2009-95

Ira Katz[1] and Ioannis G. Mikellides[2]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 91109, USA

Abstract: Hall and ion thrusters operate by ionizing a low density neutral propellant gas and accelerating the resultant ions. In the channel of Hall thrusters and in the discharge chambers of ion thrusters, the neutral gas Knudsen number is usually much greater than one; the mean free path for collisions between gas molecules is greater than typical channel or discharge chamber dimensions. While collisions between neutrals are not very important, ionizing collisions with electrons are of utmost importance and are included in all models of Hall and ion thrusters. The long mean free path for collisions between gas molecules has long been recognized by the developers of Hall and ion thruster codes, and the algorithms used to calculate neutral gas in general ignore velocity changes due to collisions with other neutral gas molecules. In this paper we present a new algorithm that exploits the fact that in ion and Hall thrusters very few collisions change the velocity vectors of neutral gases; the dominant collisions are ionization by electron impact or charge exchange. The algorithm assumes that, for neutrals emitted from a given surface, the particle velocity distribution function remains unchanged except for a scale factor that reflects the loss of neutrals to ionization. The sources of neutrals are gas inlets, such as the anode in Hall thrusters, and isotropic, thermally accommodated, gas molecules coming off thruster surfaces including recombined ions. The advantage of this algorithm over the conventional particle approach is the absence of statistical noise.

Nomenclature

b	=	boolean ray block function (b=1 ray unblocked, b=0 ray blocked)
$\bar{c}$	=	mean molecular speed
$f(\mathbf{x},\mathbf{v},t)$	=	molecular velocity distribution function
$f(\mathbf{x},\mathbf{v})$	=	normalized velocity distribution function
i	=	cell index
j	=	emitting surface index
J	=	collective emitting surface index
k	=	cell edge index
n	=	neutral particle density
$\mathbf{n}$	=	cell edge normal vector
T	=	emitted gas temperature
v	=	ionization frequency
Γ	=	neutral particle flux
Ω	=	solid angle

I.   Introduction

HALL and ion thrusters operate by ionizing a low density neutral propellant gas (or vapor) and accelerating the resultant ions. The physics of the ionization and ion acceleration processes are quire complex, and the electron and ion dynamics are controlled by electric and magnetic fields. The electric and magnetic fields are designed to make electron trajectories very long in order to maximize the probability of ionization, and the electric fields focus the ions to maximize thrust. The trajectories of neutral gas molecules are unaffected by these fields and the molecules move in straight lines until they are either ionized, hit a wall, or leave the thruster.

In the channel of Hall thrusters and in the discharge chambers of ion thrusters, the neutral gas Knudsen number is usually greater than one; that is the mean free path for collisions between gas molecules is greater than typical channel or discharge chamber dimensions. In this regime, the "free molecular flow" regime, continuum fluid models are not applicable. Computer models of Hall and ion thrusters typically simulate the neutral gas using particle in cell (PIC) algorithms combined with direct simulation Monte Carlo (DSMC) to account for ionization [1,2].

The PIC algorithm can lead to artificially large density fluctuations due to particle statistics [3]. While statistical fluctuations are known to cause heating in some plasma simulations, their impact on the simulations of Hall and ion thrusters is uncertain. Figure 1 shows the time dependent neutral gas density mid channel from an HPHall simulation of an 6 kW Hall thruster simulation [4]. The parameters for these calculations were not optimized to minimize statistical noise, nor were they modified to increase the noise to prove a point. Statistical noise can be reduced by increasing the number of macro particles, but additional particles slow the calculation. Careful tailoring of the weighting and emission algorithms also can reduce the PIC statistical nosie [2].

While the PIC algorithm is very popular for modeling collisionless plasmas, other techniques are common employed for modeling free molecular flow. Free molecular flow in the presence of walls has been traditionally modeled [5] using the same algorithms used for diffuse photon transport in thermal calculations [6]. In both cases, the incident molecules or photons are assumed to be fully accommodated on surfaces they hit,

Figure 1. Mid-channel neutral gas density for a 6 kW Hall thruster calculated using HPHall

and the fraction reemitted is assumed to be a Lambertian (cosine) distribution. The molecular flux incident on a given surface depends on the view factor between that surface and surfaces emitting molecules. Most spacecraft contamination codes also use an algorithm based upon this approach [7].

The view factor approach assumes that the transit time between surfaces is negligible compared with other timescales in the problem. However, in Hall and ion thrusters, the neutral gas molecules move more slowly than either ions or electrons. The algorithm presented below uses view factors to generate approximate molecular velocity distribution functions on a grid throughout the thruster volume, and uses a first-order upwind algorithm to move molecules between grid cells.

II.   The Algorithm

The basic concept of the algorithm is to solve for the neutral gas density by integrating forward in time an approximate Boltzmann equation. The simplified Boltzmann equation for the neutral gas including ionization but ignoring all other collisions is

The 31st International Electric Propulsion Conference, University of Michigan, USA
September 20 – 24, 2009

$$f \equiv f(\mathbf{x}, \mathbf{v}, t)$$

$$n \equiv n(\mathbf{x}, t) = \int f \, d\mathbf{v}$$

$$v \equiv v^{ionize}(\mathbf{x}, t) \tag{1}$$

$$\frac{\partial f}{\partial t} + \mathbf{v} \bullet \nabla f = -v \, f$$

The right-hand term in the last part of Equation 1 expresses the loss of gas molecules with a given velocity to ionization. Notice that velocity changes due to charge exchange collisions have been ignored.

The computational space is divided into 2-D into convex polygon cells, i, with gas sources at surface edges, j, as shown in Figure 2.

Since the simplified Boltzman equation is linear, the neutral gas velocity distribution function,

$$f_i \equiv f(\mathbf{x}_i, \mathbf{v}, t) \tag{2}$$

at the center of cell i may be written as the sum of the contributions to the distribution function by molecules emitted from surfaces, j,

$$f_i = \sum_j f_i^{\,j} \tag{3}$$

For computational efficiency, several adjacent edge cells are coalesced together treated as a single edge cell.

$$J = \{j, \ldots\}$$

$$n_i^J = \sum_{j \in J} n_i^{\,j} \tag{4}$$

The major approximation in the new algorithm is to assume that ionization changes the velocity only through a linear scale factor, The validity of this approximation is discussed later in this paper.

surface j

$$f_i$$
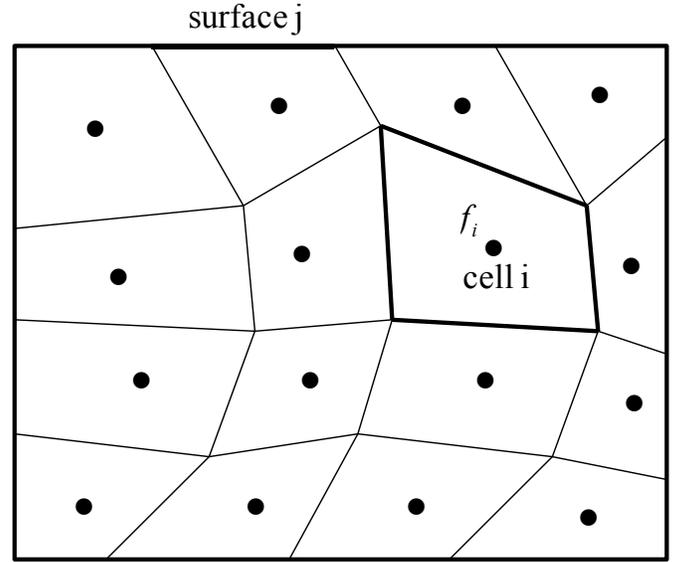
cell i

Figure 2. Computational space divided into polygons, i, and gas emitting surfaces, j

$$f_i^J(\mathbf{v}) \approx n_i^J \, \mathrm{f}_i^J(\mathbf{v}), \tag{5}$$

where n_i^J is the density of gas in cell i emitted from surface J, and $\mathrm{f}_i^J(\mathbf{v})$ is the normalized velocity distribution function at the center of cell i emitted from surface j in the absence of ionization.

$$\int \mathrm{f}_i^J(\mathbf{v}) \, d\mathbf{v} = 1 \tag{6}$$

The normalized velocity distribution functions are calculated only once, from the geometrical view factor from the cell i to the surface j. The cell-centered density is the sum of the gas densities from each surface,

$$n_i = \sum_j n_i^J \tag{7}$$

The gas density is found by time marching a separate simplified Boltzmann equation for the gas from each surface, solving a first-order upwind mass continuity equation that includes ionization.

$$\frac{\partial f_i^J}{\partial t} + \mathbf{v}^J \bullet \nabla f_i^J = -\nu\, f_i^J \tag{8}$$

Gas emanates from a surface with a positive normal velocity, v, and a thermal spread perpendicular to the surface. At large distances from the surface, the velocity spread perpendicular to the surface is reduced due to geometrical selection. Figure 3 shows how the velocity distribution function narrows (gray region) as the distance from an emitting surface increases. This happens because the emitting surface view factor decreases with distance.

Since there is a range of velocities at any point in space, in the continuity equation there can be fluxes of the same species, J, both leaving and entering a cell surface at the same time.

The outward going flux across cell edge, k, can be defined as

$${}^{k}\Gamma_i^J = \int f_i^J(\mathbf{v})\ {}^{k}\mathbf{n} \bullet \mathbf{v}\ d\mathbf{v} \tag{9}$$

Figure 3. Schematic of the change in the angular part of the distribution function, f^J, at different location relative to the surface. Gray indicates the directions of molecules emitted from surface J; white indicates that no molecules from surface J are going in that direction.

where ${}^{k}\mathbf{n}$ is the outward pointing normal from the cell. Using the assumption that the normalized velocity distribution function remains unchanged, the flux can be written in terms of an effective velocity

$${}^{k}\Gamma_i^J = n_i^J \int \mathrm{f}_i^J(\mathbf{v})\ {}^{k}\mathbf{n} \bullet \mathbf{v}\ d\mathbf{v}$$
$${}^{k}\Gamma_i^J = n_i^J\ {}^{k}\mathbf{v}_i^J \tag{10}$$

The Boltzmann equation integrated over a cell volume yields a mass continuity equation

$$\frac{\partial n_i^J}{\partial t} V_i - n_i^J \sum_k \mathbf{v}_i^{J\,k} A + \sum_k n_{i'}^J\ \mathbf{v}_{i'}^{J\,k} A = -\nu_i\, n_i^J\, V_i \tag{11}$$

where V_i is the cell volume and ${}^{k}A$ is the edge area. The first summation is over the fluxes leaving cell i; the second summation is over fluxes enter from neighboring cells. Note that the distribution function no longer appears explicitly in the Boltzmann equation, only the velocity averaged dot products with the surface normals appear. This equation has the form of a fluid equation with one major difference. There are two velocities associated with each edge; the first is for gas leaving the cell, the second for gas entering the cell.

The computational grid, including a coalesced gas emitting surface, J, is shown in Figure 4. Also shown in the figure are the two velocities associated with a cell edge; one brings gas in, the other transports gas out. The code stores for each cell

Figure 4. Computational grid with cell centered partial densities, n_i^J, and gas emitting combined surfaces, J. Notice the two velocities associated with a single edge.

The 31st International Electric Propulsion Conference, University of Michigan, USA
September 20 – 24, 2009

edge two velocities, $^+\mathbf{v}^J$ and $^-\mathbf{v}^J$, for gas emitted from each surface, J. The plus and minus signs refer respectively to counter-clockwise and clockwise with respect to the cell edge vector.

The calculation of these velocities in cylindrical geometry is as follows. First a surface of revolution is generated from each edge J as shown in Figure 5. The surface is then divided into triangles. An example of this for a vertical surface is shown in Figure 6. Note that by symmetry, only half a circle is needed. Figure 6 shows uniform angular spacing. An improved algorithm varies the angular spacing so that the smallest triangles are nearest the R-Z plane. This provides the highest resolution for points closest to the surface.

For each triangle, a ray is drawn from the center of the triangle m to the point i. If the ray intersects any thruster surface prior to reaching the cell center the ray is considered blocked.
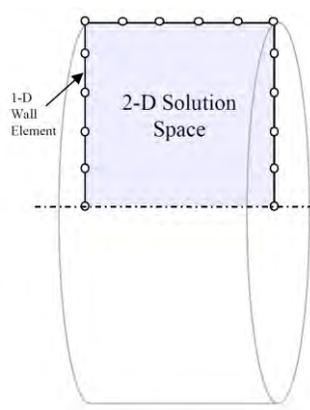
Figure 5. Edges on a 2-D grid represent a surface of revolution
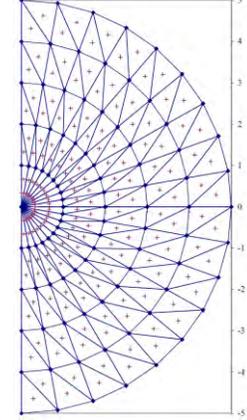
Figure 6. Triangulated surface of revolution.

$$r_{m,i} \equiv \text{ ray from triangle } m \text{ to point } i$$

$$b_{m,i} = \begin{cases} 1, & r_{m,i} \text{ unblocked} \\ 0, & r_{m,i} \text{ blocked} \end{cases} \tag{12}$$

The normalized density from J to a point i is found by summing the view factors of the unblocked rays.

$$\mathrm{n}_i^J = \frac{1}{4\pi} \sum_{m \in J} b_{m,i}\, \Omega_{m,i} \tag{13}$$

The view factor for the triangle is calculated using the algorithm of Oosterom and Strackee[8]. The positive and negative fluxes across an edge with normal $^k\mathbf{n}$ are

$$^+\Gamma_i^J = \frac{\bar{c}_J}{4\pi} \sum_{m \in J} b_{m,i}\, \Omega_{m,i} \frac{\max(\mathbf{r}_{m,i} \bullet {}^k\mathbf{n}, 0)}{|\mathbf{r}_{m,i}|}$$

$$^-\Gamma_i^J = -\frac{\bar{c}_J}{4\pi} \sum_{m \in J} b_{m,i}\, \Omega_{m,i} \frac{\min(\mathbf{r}_{m,j} \bullet {}^k\mathbf{n}, 0)}{|\mathbf{r}_{m,i}|} \tag{14}$$

where the average molecular speed, $\bar{c}_J$, is defined as,

$$\bar{c}_J \equiv \sqrt{\frac{8\, k\, T_J}{\pi\, m}} \tag{15}$$

and T_J is the temperature of surface J.

To avoid numerical issues due to shadowing, the fluxes for an edge are calculated twice, using each vertex of the edge as the field point, and the average of the two fluxes is used to find the velocities.

$$\left\langle {}^{+}\Gamma_i^J \right\rangle = \frac{1}{2}\left({}^{+}\Gamma_i^J\Big|_{\text{vertex 1}} + {}^{+}\Gamma_i^J\Big|_{\text{vertex 2}} \right)$$

$$\left\langle {}^{-}\Gamma_i^J \right\rangle = \frac{1}{2}\left({}^{-}\Gamma_i^J\Big|_{\text{vertex 1}} + {}^{-}\Gamma_i^J\Big|_{\text{vertex 2}} \right)$$

(14)

The two velocity vectors associated with the edge for gas emanating from J, ${}^{+}\mathbf{v}^J$ and ${}^{-}\mathbf{v}^J$, are calculated using the edge averaged fluxes and the cell centered normalized density. Since the fluxes and the density are calculated at different locations, the velocities are limited to be less than or equal to the thermal speed

$$ {}^{+}\mathbf{v}^J = \min\left(\frac{\left\langle {}^{+}\Gamma_i^J \right\rangle}{n_i^J}, \bar{c}_J \right) $$

(16)

$$ {}^{-}\mathbf{v}^J = \max\left(\frac{\left\langle {}^{-}\Gamma_i^J \right\rangle}{n_i^J}, \bar{c}_J \right) $$

III.   Results

The algorithm presented above has been implemented in a new, 2-D, R-Z, Hall effect thruster mode named "Hall 2De"[9]. Several tests were performed to verify that view factors were calculated accurately and that the shadowing was done correctly. One of the first tests was a direct comparison an HPHall simulation of the BPT-4000 thruster. Comparison of the neutral gas density along the centerline calculated by the two codes is shown in Figure 7. Note that the HPHall result is averaged over thousands of cycles, and the new algorithm is an instantaneous snapshot. The two results are in excellent agreement until ionization has burned up over ninety percent of the gas. About a centimeter past the channel exit plane, the gas density in both calculations is less than 1/1000 of the density near the anode.
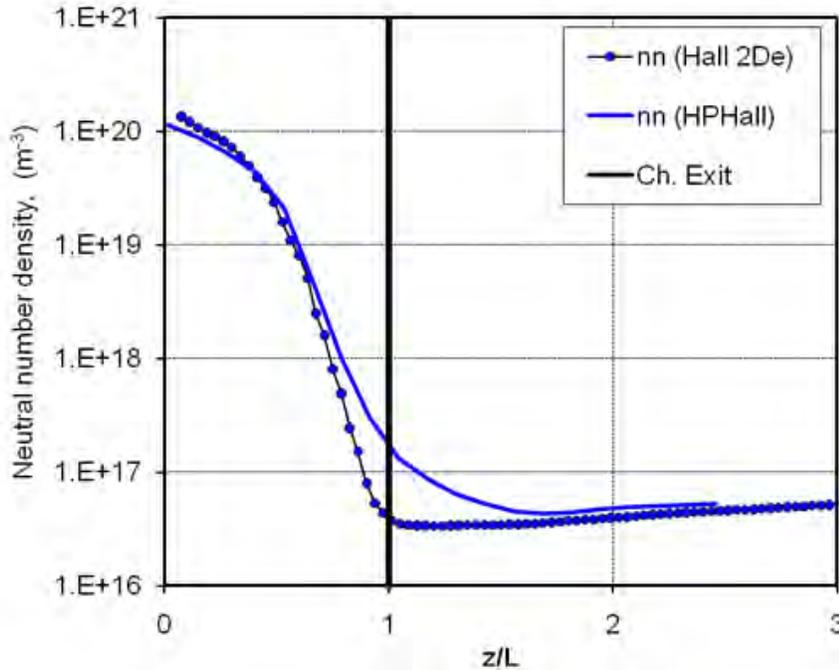
Figure 7. Comparison between the neutral gas densities at the center of the thruster channel as calculated by HPHall2 and the new Hall code (Hall 2De).

Another direct comparison was made between the two codes for calculations of a 6 kW Hall thruster. Neutral gas density contour plots are shown in Figure 8. Hall 2De uses a coarser mesh that follows magnetic field lines; HPHall uses a finer, almost rectilinear mesh. The gas densities are similar, with much of the differences attributable to the d
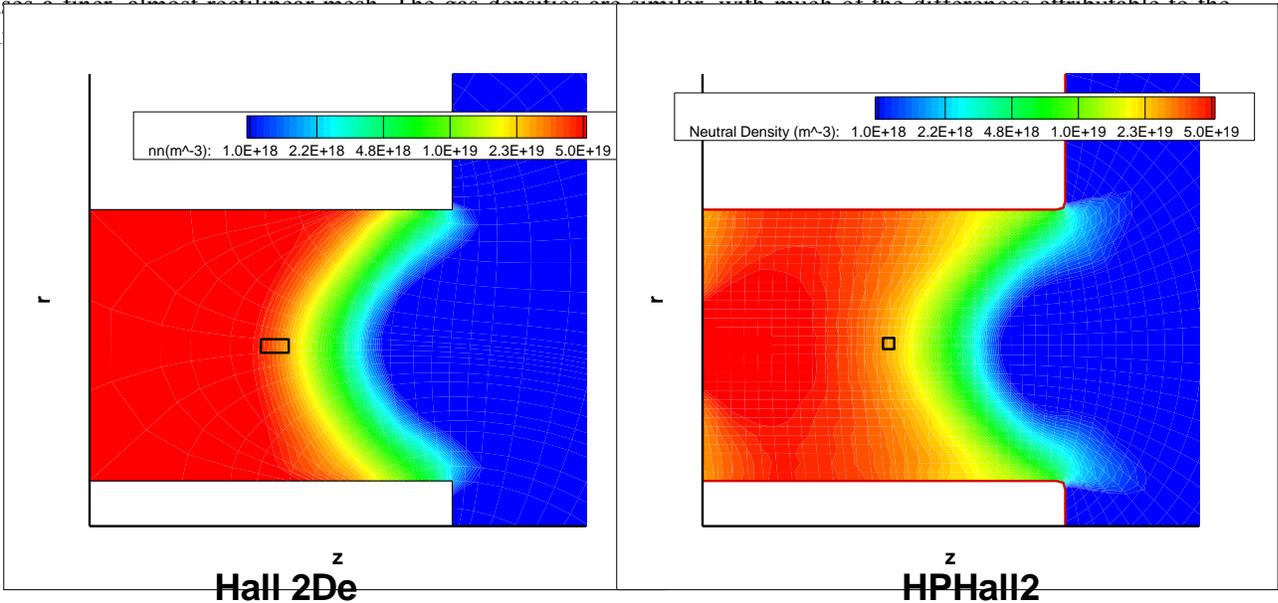
Figure 8. Channel and near plume neutral gas densities calculated by the new algorithm in Hall @De compared with time average results from HPHall

Time dependent densities for the two codes are shown in Figure 9. The densities are from the cells outlined in black in Figure 8. At these locations ionization has reduced the neutral gas density by about half. Since the codes compute somewhat different plasma densities, the average values of the gas utilized at the particular locations are different. However, it is obvious that the new algorithm accomplishes its goal of greatly reducing numerical fluctuations.
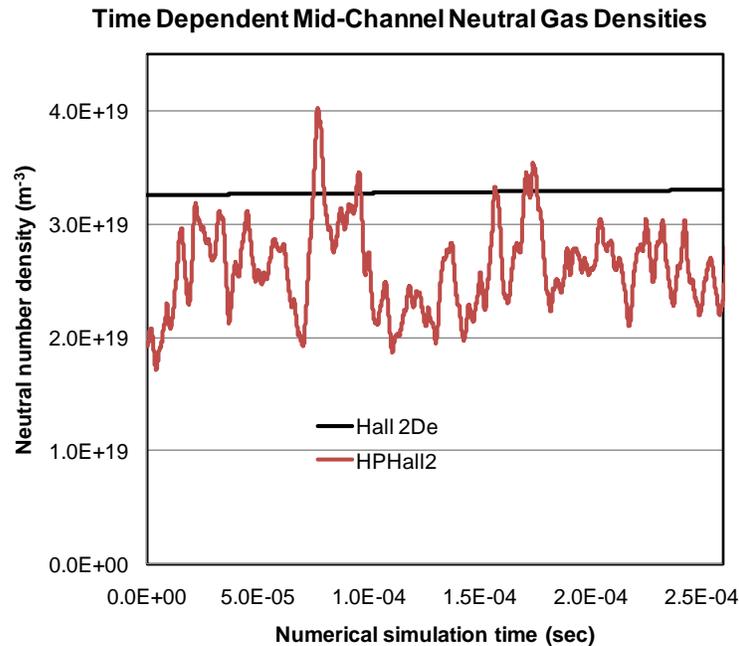
Figure 8. Time dependent mid-channel neutral gas densities calculated by the new neutral gas algorithm in Hall 2De compared with PIC – Monte Carlo calculations from HPHall

The 31st International Electric Propulsion Conference, University of Michigan, USA
September 20 – 24, 2009

IV.   Discussion

A numerical algorithm has to be efficient, stable, and, most importantly, accurate. The accuracy of the PIC algorithm is well established. The new algorithm above has demonstrated efficiency and stability. In this section is presented a preliminary analysis of the aspects affecting the algorithm's accuracy.

The principle assumption in the model is that the neutral gas molecules move in straight lines at constant velocity. In the simulations above, the peak neutral density was less than 10^{20} #/m^3. Assuming a molecular radius of 2.18 Å, the neutral-neutral collision mean free path is over 6 centimeters, or more than twice the channel width. This high density region is limited to just near the anode; everywhere else the mean free path is much longer.

The second assumption is that the velocity distribution function doesn't change in time. While related to the absence of momentum changing collisions, to be strictly valid, this assumption requires that the attenuation of density due to ionization be uniform for all angles. Just as the accuracy of PIC improves with the number of particles, the accuracy of the algorithm above can be improved by increasing the number of the gas emitting surfaces, J. In the calculation shown in Figure 8, there were only three combined surfaces in the channel (back wall, inner and outer walls). The anode and each small wall element individually emit gas, but there are only three velocity fields that transport the gas. Increasing the number of velocity fields, J, and decreasing the size of each, decreases the angular spread in the distribution functions, thereby reducing errors due to changes in time. If the angular distribution function were a delta function, the distribution function would not change in time since the particle move in straight lines.

Finally, the algorithm assumes that the speed distribution doesn't change. For molecules re-emitted from walls with a Maxwellian distribution, ionization changes the speed distribution function. The ionization collision loss term in Equation 1 is independent of velocity. However, the transit time depends inversely on the speed of a particle; slower particles take more time and are more likely to be ionized, faster particles are less likely to be ionized. As a result, in 1-D the speed distribution function changes with the degree of ionization. The first molecules ionized are the slower moving ones. The constant speed assumption ignores this and initially predicts that more gas remains. When over ninety percent of the gas has been ionized, the remaining Maxwellian gas molecules have high velocities and move very quickly through the ionizing region. In this case the constant speed assumption predicts a higher degree of ionization then than the exact solution. A

Figure 9. Effect of constant speed distribution assumption on the density of a Maxwellian gas moving through a uniform ionizing medium.

simple improvement to the algorithm is to have multiple velocity bins. Having even two velocity bins reduces the maximum error by about a factor of four. In the Hall thruster example above, the ionization region is very intense and very short. The effect of the constant speed assumption moves ion generation less than one grid cell, and, since the algorithm is strictly mass conserving, should have no impact on the total ion currents.

Another source of error is numerical diffusion associated with the first-order, upwind, algorithm used to time integrate the mass continuity equation.

V.   Conclusion

A new algorithm for calculating neutral gas densities in computer simulations of ion and Hall thrusters has been developed an applied in a new Hall thruster model. This new algorithm takes advantage of the fact that almost all neutral gas molecules in an ion or Hall thruster have straight line, constant velocity, trajectories until they are either ionized, hit a wall, or leave the system. The algorithm is related to the view factor approach used in spacecraft

contamination modeling. Compared with the PIC method commonly used in Hall thruster codes, the new algorithm achieves the goal of quiet and smooth results. The first order upwind differencing algorithm introduces some spatial diffusion. Like most numerical algorithms, higher accuracy can be achieved with finer resolution. A preliminary analysis also shows that the accuracy of this algorithm can be improved by finer spatial resolution of the emitting surfaces (this results in finer angular resolution of the velocity distribution function), and by dividing the Maxwellian speed distribution into two or more bins.

Acknowledgments

References

[1] Fife, J. M., "Hybrid-PIC Modeling and Electrostatic Probe Survey of Hall Thrusters," Ph.D. Thesis, Aeronautics and Astronautics, Massachusetts Institute of Technology, 1998.

[2] Parra, F. I., Ahedo, E., Fife, J. M., and Martinez-Sanchez, M., "A Two-Dimensional Hybrid Model of the Hall Thruster Discharge," Journal of Applied Physics 100, 023304 (2006).

[3] Birdsall, C. K., and Langdon, A. B., *Plasma Physics Via Computer Simulation*, McGraw-Hill Book Company, New York, 1985

[4] Hofer, R.R., et al.," Efficacy of Electron Mobility Models in Hybrid-PIC Hall Thruster Simulations", AIAA Paper 2008-4924, July 2008

[5] Clausing P., "The Flow of Highly Rarefied gases through Tubes of Arbitrary Length", Ann. Physik, **12**, 961, 1932

[6] Holman, J. P., *Heat Transfer*, 7th ed., McGraw-Hill, Inc., New York, 1990

[7] Brent, D.A., Cottrell, F.D., Henderson, K.A., and Dahbura, R.S., "Modeling of Spacecraft Using a Modified Version of MOLFLUX and Comparison with a Continuous Flux Model, SPIE Vol. 2864, 1999

[8] Van Oosterom, A., and Strackee, J., "The Solid Angle of a Plane Triangle", IEEE Transactions of Biomedical Engineering Vol. BME-30, No. 2, February 1983

[9] Mikellides, I. G., Katz, I., Hofer, R. H., and Goebel, D.M., "Hall-Effect Thruster Simulations with 2-D Electron Transport and Hydrodynamic Ions", IEPC-2009-114, Ann Arbor, Michigan, September, 2009