Convergence of Stochastic Models for Electric Propulsion Plume Simulation

IEPC-2019-156

Presented at the 36th International Electric Propulsion Conference
University of Vienna, Austria
September 15-20, 2019

Samuel J. Araki[†], David L. Bilyeu[‡], Justin W. Koo[§] and Robert S. Martin
Air force Research Laboratory, Edwards Air Force Base, CA 93524, USA

Kenichi Kubota[¶], Kiyokazu Koga[||], Shinatora Cho[**] and Kiyoshi Kinefuchi[††]
Japan Aerospace Exploration Agency, Japan

Shinji Hatta[‡‡] and Hiroko O. Ueda[‡‡]
MUSCAT Space Engineering Co., Ltd, Kitakyushu, Fukuoka, Japan

Slow ions created by charge-exchange (CEX) collisions in an electric propulsion (EP) plume are the primary cause of sputtering on important spacecraft components. When using numerical tools to assess spacecraft integration, it is essential to perform a convergence study on the computational grid resolution and simulation particle count, which is often not performed rigorously. Using two different plume codes, TURF and MUSCAT, this paper studies the convergence behavior of the EP plume simulations with simple geometry and particle injection model. This is accomplished by utilizing two metrics, namely the earth mover's distance (EMD) and the relative difference in integrated quantity, computed for the ion flux distributions downstream of plume and on a solar array surface with no line of sight of the thruster exit. It is shown that the convergence of downstream ion flux distribution is achieved with a minimum resolution of the computational grid of four cells spanning along the thruster exit diameter. However, the ion flux distribution on the solar array does not converge even by further refining the grid resolution by two levels. Since the number of simulation particles are kept the same between different grid resolutions, the unconverging behavior is likely attributed to small number of particles per cell especially near the plume edge, leading to insufficient number of collisions for the DSMC method.

I. Introduction

In plumes of electric propulsion (EP) devices, several different species such as neutral atoms, ions, and electrons are present. Ions directly from the thruster (primary ions) as well as those created in the plume through charge-exchange (CEX) processes can contribute to sputtering of spacecraft components even in regions with no line of sight of the thruster channel. As a result, important components can be eroded and/or contaminated to reduce or lose their functionalities. Therefore, assessment of spacecraft integration is particularly important when designing spacecraft with EP devices, which is commonly accomplished

[*]Distribution A: Approved for Public Release; Distribution is Unlimited. PA No. 19459
[†]Computational Scientist, ERC Inc., In-Space Propulsion Branch (AFRL/RQRS), samuel.araki.ctr@us.af.mil
[‡]Modeling & Simulation Group Lead, In-Space Propulsion Branch (AFRL/RQRS)
[§]Technical Advisor, In-Space Propulsion Branch (AFRL/RQRS)
[¶]Researcher, Aeronautical Technology Directorate, Chofu, Tokyo, 182-8522, Japan
[||]Senior Engineer, Institute of Aerospace Technology, Tsukuba, Ibaraki, 305-8505, Japan
[**]Research Engineer, Research and Development Directorate, Sagamihara, Kanagawa, 252-5210, Japan
[††]Associate Senior Research Engineer, Space Technology Directorate I, Tsukuba, Ibaraki, 305-8505, Japan
[‡‡]Research Scientist, 1-1 Sensui-cho, Tobata-ku, Kitakyushu, Fukuoka, 804-9550, Japan

with numerical tools. The spacecraft module of Thermophysics Universal Research Framework (TURF)[1,2] developed at Air Force Research Laboratory (AFRL) can perform simulations for EP plumes. It uses hybrid particle/fluid approach where heavy species (singly and multiply charged ions and neutral atoms) are solved by Particle-in-Cell (PIC) method and electrons are solved via Boltzmann relation.[3] Both the spacecraft charging[4] and Hall-Effect Thruster (HET) device models are currently being incorporated into the module, which will allow for a strong coupling between these models. On the other hand, Multi-Utility Spacecraft Charging Analysis Tool (MUSCAT)[5] funded by Japan Aerospace Exploration Agency (JAXA) and currently maintained by MUSCAT Space Engineering Company (MUSE) has been developed primarily for spacecraft charging simulations. MUSCAT can simulate plasma via the full PIC method, however this method is not suitable for spacecraft integration study due to the large simulation domain and the small mesh size required by the full PIC method. Therefore, the same plume model as TURF's spacecraft module has been incorporated into MUSCAT, which makes a direct comparison of the two codes possible.

In simulating a large scale problem such as the EP plume, the hybrid PIC/fluid method can be expensive as it relies on both discrete particles and computational meshes. The method suffers from statistical noise due to the particles; noise is reduced by increasing the number of particles per cell and time-averaging the steady-state solution,[a] but only by the square-root of the particle count. In plume simulations, the CEX ions can be responsible for the majority of the sputtering, and care needs to be taken to ensure that they are resolved correctly. It is particularly difficult to increase the signal-to-noise ratio of the CEX ion flux distribution on spacecraft surfaces due to their small fraction compared to the primary ions and a large percentage leaving the domain without impacting any surface. In addition, trajectories of charged particles are affected by the electric field stored on the mesh, and therefore mesh resolution has to be increased in order to obtain converged solution. Nevertheless, finer mesh resolution reduces the particle count per cell, amplifying the statistical noise and contaminating the simulation results. In order to maintain the noise level, the number of particles has to be increased in proportion to the mesh resolution. Consequently, before reaching the particle count and mesh resolution providing the fully converged solution, the simulation often becomes too large to fit in the available computer memory. Furthermore, these simulations are commonly used for designing spacecraft, so a sufficiently rapid wall time induces another constraint for the simulation.

The objective of this research is to study the convergence behavior of stochastic particle simulation and identify sources of discrepancy between the EP plume models. Most of the calculations are carried out in TURF when comparing different implementations and models, while some comparisons are made against MUSCAT for the base simulations. In achieving the research goal, earth mover's distance (EMD) and the relative difference of total quantity are utilized to compare different discrete distributions. The EMD metric is superior to L2 norm in that it does not require the two distribution to be stored in the same mesh, which enables comparisons between results from the two codes. In this paper, a brief description of the two models is provided first. Then, the metrics are described in the following section. The third section provides the numerical results such as convergence behavior with mesh resolution and particle count and sensitivity with numerical implementations and models. Finally, this paper closes with summary and suggestions for future work.

II. Numerical Models

TURF is an in-house scientific computing framework that unifies operators by encapsulating them in TURF's general object, which enables modular code design such that different numerical methods or models can be plugged in and combined to build complex physics simulations. The spacecraft module is built upon TURF and contains capabilities for spacecraft simulations including Hall-thrusters, EP plumes, spacecraft integration, and spacecraft charging. On the other hand, MUSCAT, initially developed to analyze the charging status of satellites in different orbits, has been expanded for EP plume simulations. The two codes' most basic plume models are used for this study and are described herein.

The basic plume simulation uses the hybrid particle/fluid approach and assumes quasi-neutrality throughout the simulation domain. This assumption can be violated especially in a higher altitude where the Debye length becomes larger than the computational grid size. In contrast, quasi-neutrality holds for low-earth altitude orbits due to the high atmospheric plasma density. As illustrated in Fig. 1, the basic plume simulation contains four primary operations: (A) particle injection, (B) particle advancement, (C) potential solver, and (D) collisions. Standard methods are used for particle weighting and electric field calculation.[6,7] In order

[a]Time-averaging the steady-state solution is effectively increasing the number of sampled particles per cell

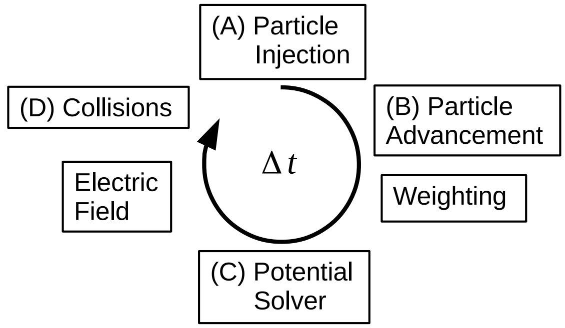


Figure 1. The PIC cycle. In the hybrid PIC/fluid approach for the EP plume simulation, ions are solved via the PIC method as illustrated above. Assuming quasi-neutrality, electron density is set equal to the ion density. Then, the potential is solved by inverting the Boltzmann relation, assuming Boltzmann electron and constant electron temperature.

to simplify the comparative study, the streaming Maxwellian source model is used for injection of ions and neutrals, and electric potential is solved by inverting Boltzmann relation. For assessment of spacecraft integration, the source model is likely be inappropriate for any thruster, and more sophisticated model should be used instead. Other models such as surface reflection and sputtering should also be included, but they are omitted for this study.

A. Streaming Maxwellian Particle Injection

Both TURF and MUSCAT use the acceptance-rejection procedure when injecting particles with the streaming Maxwellian source model. Let the stream velocity normal to the injection surface be v_n, the normal velocity component be v, and the thermal velocity be v_{th}. The distribution function of the normal velocity component is given as

$$f(s) = (s + s_n)\exp\left(-s^2\right) \tag{1}$$

where s_n and s are the normalized velocities given as $s_n = v_n/v_{th}$ and $s = v/v_{th}$, respectively. For the acceptance-rejection procedure, the maximum value of Eq. (1) is needed to bound $f(s)$.

$$f_{\max} = \frac{s_n + \left(s_n^2 + 2\right)^{1/2}}{2} \exp\left[-\frac{1}{2} - \frac{s_n}{2}\left\{s_n - (s_n^2+2)^{1/2}\right\}\right] \tag{2}$$

In the routine, the normalized velocity s is sampled by $s = \alpha(2U_1 - 1)$ where α is some cutoff value (i.e. 3 is used in TURF) and U_1 is a random number between 0 and 1. Then, the value s is accepted if satisfying $U_2 < f(s)/f_{\max}$ where U_2 is another random number.

B. Particle advancement

In the PIC model, the common choice of time-integrator is the Leapfrog method,[6] in which positions and velocities are updated at time points staggered by half the time-step, leading to the second-order accurate method.

$$\mathbf{v}_{k+1/2} = \mathbf{v}_{k-1/2} + \mathbf{a}_k\Delta t, \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{v}_{k+1/2}\Delta t \tag{3}$$

where the subscript k denotes the index for time-step, Δt is the simulation time-step, and $\mathbf{a}$, $\mathbf{v}$, and $\mathbf{x}$ are the acceleration, velocity, and position vectors, respectively. This scheme is used in MUSCAT. In contrast, the Euler method is used in TURF's plume simulation throughout this study.

$$\mathbf{v}_{k+1} = \mathbf{v}_k + \mathbf{a}_k\Delta t, \quad \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{v}_k\Delta t \tag{4}$$

The Euler method is only the first-order accurate method. The discrepancy in time-integrator methods can be the source of difference in the results from TURF and MUSCAT.

C. Inversion of Boltzmann Relation for Potential

Although both TURF and MUSCAT incorporate higher fidelity potential solvers, in this study, the electric potential is solved by inverting the Boltzmann relation while assuming Boltzmann electron fluid and constant electron temperature.

$$\phi - \phi^* = \frac{k_B T^*}{e} \ln\left(\frac{n_e}{n_e^*}\right) \tag{5}$$

where ϕ is the electric potential, k_B is the Boltzmann constant, T is the temperature, e is the elementary charge, n_e is the electron density, and the superscript $*$ denotes the reference value. Once the ion density is determined from ion particles, the electric potential is calculated by assuming quasi-neutrality and setting the electron density equal to the ion density.

D. Collisions

The collisional mechanisms included in the basic plume simulation are elastic collision between ions and neutrals, which includes momentum exchange (MEX) and CEX collisions. Both the MEX and CEX collisions are essentially the same process except that one or more electrons are exchanged between ions and neutrals for the CEX collisions. Nevertheless, the two processes are typically treated quite differently. The common way to treat the CEX collision is by only exchanging the charges between the collision pair, assuming that the collisional process only takes place in the long range such that no momentum exchange occurs. This is a marginally valid assumption, and the degree of validity is studied in Section IV.D. For the basic plume simulation, the variable hard-sphere (VHS) method is employed for the MEX collision, and no momentum-exchange is assumed for the CEX collision, as described in Ref. 8. Both TURF and MUSCAT use the Direct Simulation Monte Carlo (DSMC) method to approximate the two collisional processes. However, the DSMC schemes are slightly different; TURF uses Bird's no-time-counter (NTC) scheme,[9] and MUSCAT uses the modified Nanbu's scheme,[10,11] and both the schemes should produce the same results if implemented correctly.

In the NTC scheme, the maximum number of collisions is determined first.

$$N_{c,max} = N_i n_n \langle \sigma v_r \rangle_{max} \Delta t \tag{6}$$

where N_i is the number of ions in the cell, n_n is the neutral density, σ is the cross-section, and v_r is the relative speed between ion and neutral particles. The collision cross-section σ is generally a function of relative velocity, and the maximum number of collision in a cell is determined by using $\langle \sigma v_r \rangle_{max}$, the maximum value of σv. For each randomly selected N_c pairs of ions and neutrals, whether a collision actually happens is determined by the Monte Carlo Probability, P_c.

$$P_c = \frac{\langle \sigma v_r \rangle_{i,j}}{\langle \sigma v_r \rangle_{max}} \tag{7}$$

where i and j are the indices for ions and neutrals, respectively. Here, $\langle \sigma v_r \rangle_{ij} = \sigma(v_{r,ij}) v_{r,ij}$, and $v_{r,ij}$ is the relative speed between particles i and j given by $v_{r,ij} = |v_i - v_j|$. If $\langle \sigma v_r \rangle_{ij}$ is greater than $\langle \sigma v_r \rangle_{max}$, $\langle \sigma v_r \rangle_{max}$ is updated accordingly. A collision happens if $U_3 < P_c$ where U_3 is a random number between 0 and 1. Note that N_c is calculated first to bound the actual number of collisions, and P_c is utilized to reject oversampled collision pairs, which makes the collision algorithm efficient.

In the modified Nanbu's scheme, all the ions are considered for collisions, and the probability P_{ij} is used to determine if the ith ion experiences a collision with jth neutral. First, the index of neutral particle is chosen randomly as $j = \texttt{floor}(U_4 N_n)$ where U_4 is the random number and N_n is the number of neutrals. Then, using the same random number, a collision occurs when the inequality equation is satisfied.

$$U_4 > j/N_n - P_{ij}, \quad P_{ij} = \frac{n_n}{N_n} \Delta t \langle \sigma v_r \rangle_{i,j} \tag{8}$$

This method is slightly slower than the NTC method but is easily parallelizable.[12] In addition, the modified Nanbu's scheme results in less random drift from the true solution compared to the NTC method, since all the ions are considered for collisions as opposed to completely random sampling of ions in the NTC scheme.

E. Difference in Code Design

Among the primary operations of TURF and MUSCAT for the hybrid PIC/fluid simulation, there are discrepancies only in the particle advancement and collision schemes. However, there are fundamental differences in the code design; these include the locations of field data and representation of geometry. The field data are computed on the cell-centers and nodes in TURF and MUSCAT, respectively. For this reason, the data locations are offset by half the cell size when the computational grid matches exactly between the two codes. In order to match the data locations, the computational grid in TURF is shifted by half the cell size in the simulations presented herein. On the other hand, geometries are represented differently in TURF and MUSCAT. In TURF, the satellite geometry is represented by a triangulated surface mesh, which can cut through the volume grid. The surface mesh resolution can be kept the same between the simulations with different volume grid resolution. In contrast, MUSCAT does not use a surface mesh. Instead, the satellite geometry is represented by a combination of volume grid cells such that the geometry is pixelated. Consequently, the resolution of satellite geometry increases as the volume grid is refined. For the comparative study, the satellite geometry is set such that it is aligned with the computational grid, which removes the potential discrepancy between the two simulations.

F. Upgrades to TURF to Better Match with MUSCAT

There are several changes made to TURF since Ref. 3 in order to better match the results from MUSCAT. These include the first-order sampling of fields, electric field calculation near walls, and the collision method. Although the charge density was determined by weighting ions in the first-order manner, the outputted fields were sampled using the zeroth-order method, while MUSCAT sampled using the first-order method[b]. In order to make the comparison of results easier, the first-order sampling was implemented in TURF. Another change made in TURF was on the electric field calculation near geometry boundaries. Previously, TURF imposed the Dirichlet boundary condition on the volume cells that intersected any geometry boundary.[3] This created a numerical sheath near the geometry, artificially increasing or decreasing the ion flux onto the geometry surfaces. A portion of a volume cell intersecting a geometry is in a free space and can contain particles. By computing the density according to the particles instead of simply imposing the Dirichlet boundary condition, the effect of numerical sheath was reduced. Nevertheless, the numerical artifact near walls was still present. In an effort to further remove the numerical sheath, the near-wall treatment was improved by extrapolating the potential from the free space into the cells inside the geometry as described in Ref. 4. Lastly, TURF used the Monte Carlo collision (MCC) method instead of DSMC, assuming that the ion density is much smaller than the neutral density. In the near-plume region, the ion density is typically one order of magnitude lower, but it turned out that the difference in densities is too small for the assumption to be valid. The MCC method did not take into account of the depletion of neutral population, which directly affected the CEX ion generation rate and thus the ion flux to the spacecraft surfaces. Furthermore, in the MCC method, a neutral particle is sampled from a fluid so the neutral particle velocity can only be approximated by assuming a known distribution, and the pre-collision neutral velocity significantly affects the destination of CEX ions. For this reason, the collision method in TURF is switched to the DSMC method, which is also used in MUSCAT. The effects of near-wall electric field calculation and collision method are studied in Section IV.D.

III. Earth Mover's Distance

The EMD is a measure of distance between two probability distributions, which can be used to quantify the sensitivity of results with different inputs, implementations, and models. This metric is also called the Wasserstein metric in mathematics. Given two discrete distributions f and g with n and m bins, respectively, computing the EMD involves solving the linear programming problem to minimize the overall cost.[13–15]

$$\text{minimize } \sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij}w_{ij} \tag{9}$$

such that:

[b]The zeroth- and first-order sampling distribute particle quantities to a single cell-centered point and multiple surrounding cell-centered points, respectively.

$$w_{ij} \geq 0, \quad \sum_{i=1}^{n} w_{ij} = g_j, \quad \sum_{j=1}^{m} w_{ij} = f_i \tag{10}$$

where subscripts i and j denote indices of bins on distributions f and g, respectively, d_{ij} is the distance between bins i and j, and w_{ij} is the particles (or weights) to be moved from bins i to j. The solution of Eq. (9) contains n by m matrix of w_{ij} values. It should be noted that the location and size of bins on f and g are not required to match. The linear programming problem is solved by utilizing the Python optimal transport library, POT,[16] which is based on the algorithm described in Ref. 15. The method used in POT requires that the distributions f and g to be normalized by the total ($\sum_i^n f_i$ and $\sum_j^m g_j$, respectively). Also, the distance d_{ij} is normalized by the maximum value ($\max_{i,j} d_{ij}$). With the normalization applied to d_{ij}, f_i, and g_j and the solution of w_{ij} from Eq. (9), the EMD simply becomes the sum of costs between all the pairs of bins.

$$W_1 = \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} w_{ij} \tag{11}$$

When POT solves the linear programming problem to determine the EMD using the normalized quantities, it is essentially solving the "balanced" optimal transport problem, in which the two distributions integrate to the same value. This metric provides insight into how close the features of the distributions look. However, in many physical problems, it is not guaranteed that integrations of the distributions match. There are numerous studies toward "unbalanced" or "unnormalized" optimal transport,[17,18] but utilizing these advanced methods is beyond the scope of this paper. Therefore, another metric that quantifies the difference in the integrated quantities is defined by taking the relative difference in the total of the two distributions.

$$D_m = \frac{h_{max} - h_{min}}{h_{min}}, \quad h_{min} = \min\left(\sum_{i=1}^{n} f_i, \sum_{j=1}^{m} g_j\right), \quad h_{max} = \max\left(\sum_{i=1}^{n} f_i, \sum_{j=1}^{m} g_j\right), \tag{12}$$

IV. EP Plume Simulations

A. Simulation Setup

In the EP plume simulation for this study, the satellite geometry consists of the body, thruster, and solar array, and the ion current is collected by the downstream plate (CP1) and the solar array (CP2), as shown in Fig. 2. TURF uses the triangulated mesh shown in Fig. 2, while MUSCAT uses the geometry represented by a combination of volume grid cells. The currents on the collection plates, CP1 and CP2, are mainly contributed by the primary and CEX ions, respectively. These distributions of the collected current from simulations of different inputs, implementations, and models are compared using the two metrics. The solar

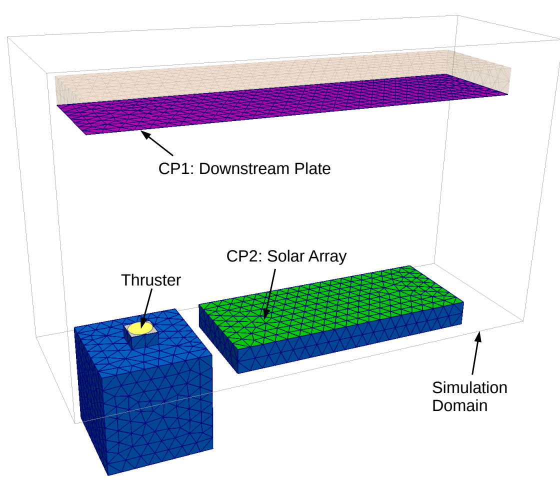


Figure 2. Satellite geometry for the convergence study. Two collection plates, CP1 and CP2, are located downstream and slightly behind the thruster exit plane, collecting mainly the primary and CEX ions, respectively. Dimensions of satellite geometry are: body ($0.4 \times 0.4 \times 0.4$), thruster ($0.1 \times 0.1 \times 0.05$), and solar array ($1.0 \times 0.4 \times 0.1$) in the unit of meters.

Table 1. Inputs for Streaming Maxwellian Source.

Inputs	Species	
	Ion	Neutral
Normal Velocity, m/s	$20,000$	0
Tangential Velocity, m/s	$2,500$	0
Temperature, K	$10,000$	500
Mass Flow Rate, kg/s	5×10^{-6}	5×10^{-7}
Particle Weight	1×10^{10}	5×10^{10}

Table 2. Grid Resolution.

Resolution	dx
0	0.2
1	0.1
2	0.05
3	0.025
4	0.0125
5	0.00625

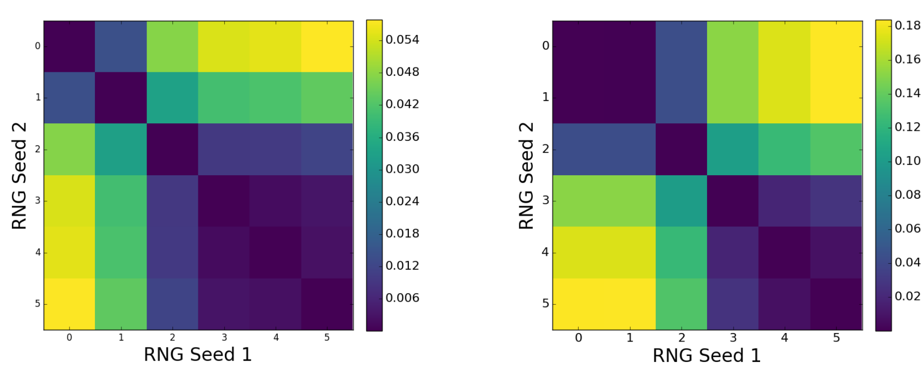
array, CP2, is located slightly behind the thruster exit plane, and therefore only a small fraction of the CEX ions impact the surface when the stream velocity of ions is in the direction normal to the thruster exit plane, leading to such a large signal-to-noise ratio as to prevent any comparative study between different conditions. For this reason, a tangential component of the stream velocity toward the solar array is added to enhance the CEX ion impact on CP2. Table 1 shows the inputs for the streaming Maxwillian source used to inject ion and neutral particles. Note that all simulation results presented are from TURF unless noted otherwise.

The simulation domain spans from $(-0.3, -0.3, -0.1)$ to $(1.5, 0.3, 1.0)$, and the resolution of volume grid cell is varied in this study. Six different levels of volume grid resolution are used as summarized in Table 2. While the TURF surface mesh remains the same between simulations, the resolution of current collection surface mesh changes according to the volume grid resolution in MUSCAT.

B. Simulation without Collisions

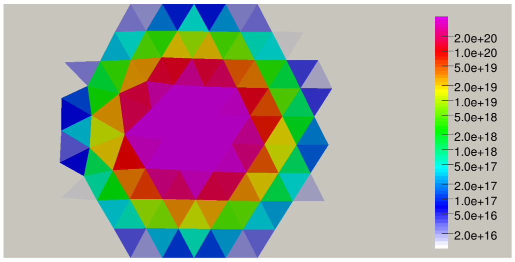
The convergence study of the EP simulation is first performed without any collisional effect. Without the CEX collisions, ions rarely impact the solar array, so only the ion flux on the downstream plate is studied. There is the noise associated with the stochastic nature of particle simulations, and the degree of noise affecting W_1 and D_m is quantified by running the same simulation with different sequence of random numbers obtained with different random number seeds. Figure 3 shows (a) the EMD and (b) the relative difference in total flux between simulations with different grid resolution. As seen in the diagonal entries (upper-left to lower-right) of the Fig. 3, both W_1 and D_m are small, indicating that the statistical noise is not significantly affecting the two metrics. The off-diagonal entries of Fig. 3 represent the difference in the distribution between the volume grid resolution. Only after Resolution 3, both W_1 and D_m remain small. In other words, convergence is nearly achieved, and the distribution remains almost the same when further refining the volume grid resolution.

Figure 4 shows the ion flux distribution when simulated with different volume grid resolution. Just from a visual inspection, it is clearly seen that converged results are obtained from Resolution 3, 4, and 5. Note that for Resolution 4, four grid cells span across the diameter of the thruster exit area, indicating roughly

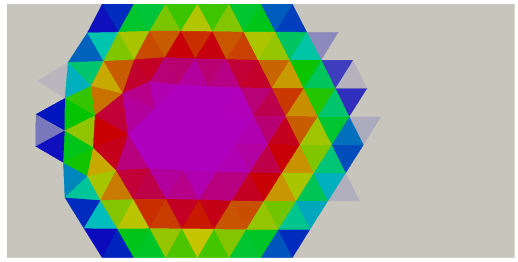


(a) W_1 (b) D_m

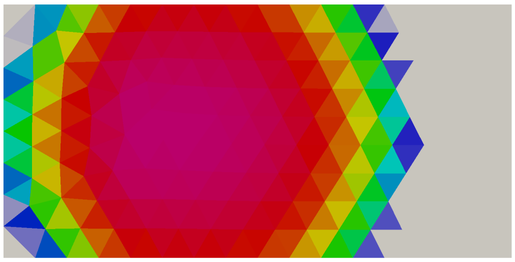
Figure 3. Comparison of ion flux distributions on the downstream plate (CP1). Two set of simulations with different random number seeds are performed on six different volume grid resolutions (0 to 5). Plotted are the metrics for determining the degree of discrepancy between simulations.



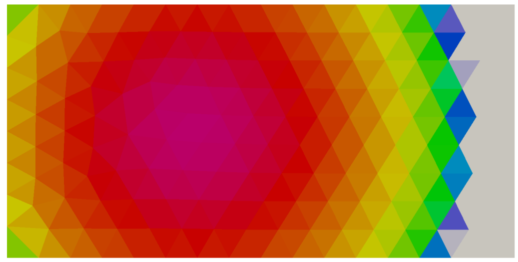
(a) TURF: Resolution 0.



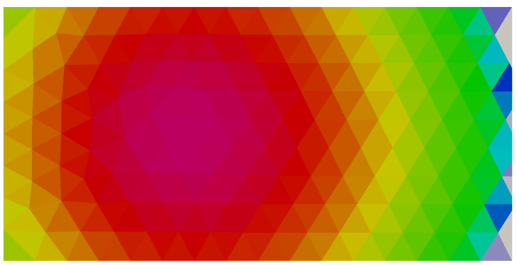
(b) TURF: Resolution 1.



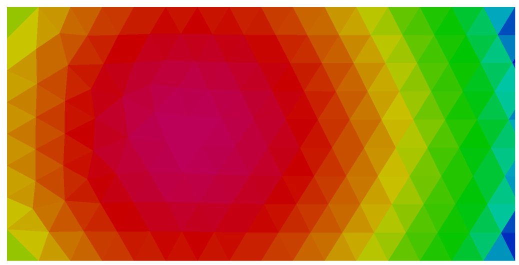
(c) TURF: Resolution 2.



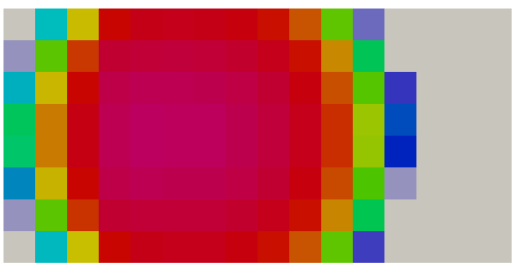
(d) TURF: Resolution 3.



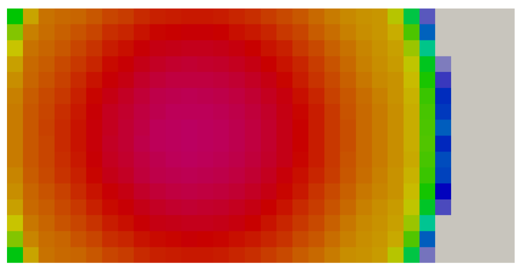
(e) TURF: Resolution 4.



(f) TURF: Resolution 5.



(g) MUSCAT: Resolution 2.



(h) MUSCAT: Resolution 3.

Figure 4. Ion flux distribution on the downstream plate (CP1) obtained from simulations from TURF and MUSCAT with different volume grid resolution (see Table 2).

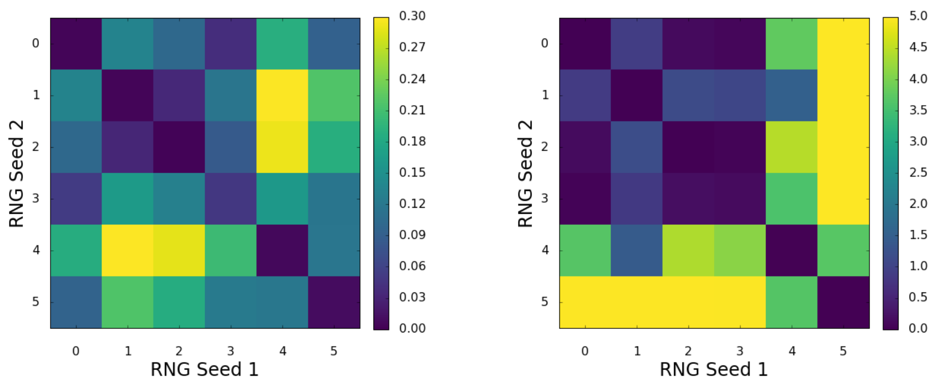
the minimum volume grid resolution to obtain fairly converged plume profile. It is also seen that results from MUSCAT agree very well with the TURF's results for the corresponding grid resolutions, confirming that the Maxwellian source, the particle advancement, and the electric field calculations are performed consistently between TURF and MUSCAT.

C. Full Simulation: Convergence

Elastic collisions (MEX and CEX) are included after the convergence is shown to be achieved for the case without collisions. Just as the previous study, the simulations with six different volume grid resolutions are performed for two different random number seeds to quantify the degree of statistical noise. For the downstream plate, similar results as the case without collisions are obtained such that the distribution starts to converge after Resolution 3. When the collisional effect is incorporated, a small fraction of ions reach the region that the main beam never reaches, which is attributed to the CEX ions. Nevertheless, the distribution is dominated by the primary ions, driving the two metrics, W_1 and D_m.

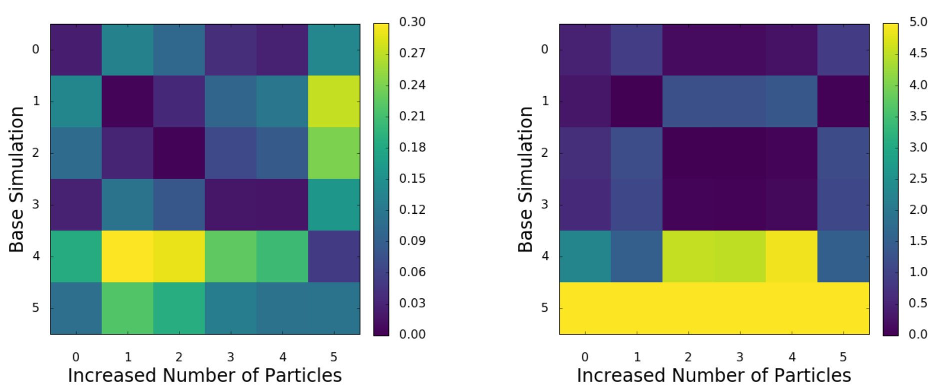
On the solar array, the ion flux distribution does not seem to converge with the grid resolution. Figure 5 shows W_1 and D_m indicating the closeness of distributions with regards to the feature and the magnitude, respectively. As shown in Fig. 5(a), all the off-diagonal entries are significantly larger than the diagonal entries, implying that results are notably different between volume grid resolutions and that the simulation has not reached convergence. The diagonal entries indicate how much W_1 is affected by the statistical noise, and since the off-diagonal entries are larger, the statistical noise is not the primary cause of unconverged solution. When observing the D_m metric in Fig. 5(b), it is clearly seen that the metric is much larger for Resolution 4 and 5, which means that total ion flux on the solar array changes significantly for these resolutions.

The cause of unconverging behavior with grid resolutions is further investigated by comparing the simulation results obtained with different number of particles. For the large particle count simulation, the particles weights for ions and neutrals are reduced to 5×10^8 and 5×10^9, respectively. Figure 6 shows the two metrics between the base simulations and the simulations with increased number of particles. As seen in Figs. 6(a) and 6(b), the diagonal entries for Resolutions 4 and 5 are no longer small for both the metrics, implying that the ion flux solution changes drastically when the number of particles is increased. The ion flux distribution on the solar array is primarily contributed by the CEX ions created near the thruster and



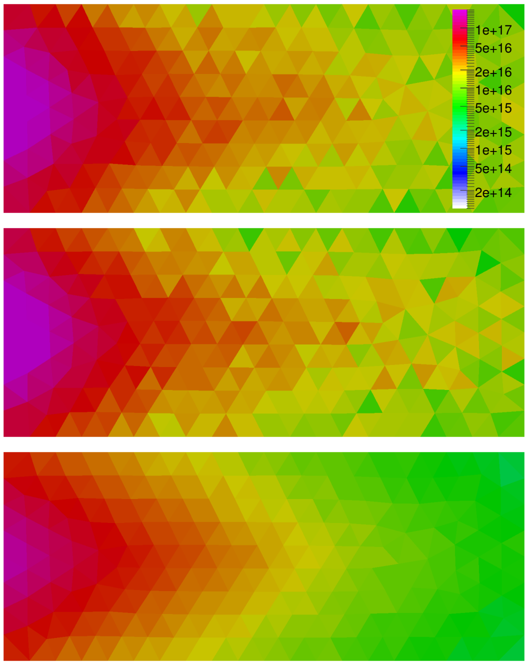
(a) W_1 (b) D_m

Figure 5. Comparison of ion flux distributions on the solar array (CP2). Two set of simulations with different random number seeds are performed on six different volume grid resolutions (0 to 5). Plotted are the metrics for determining the degree of discrepancy between simulations.

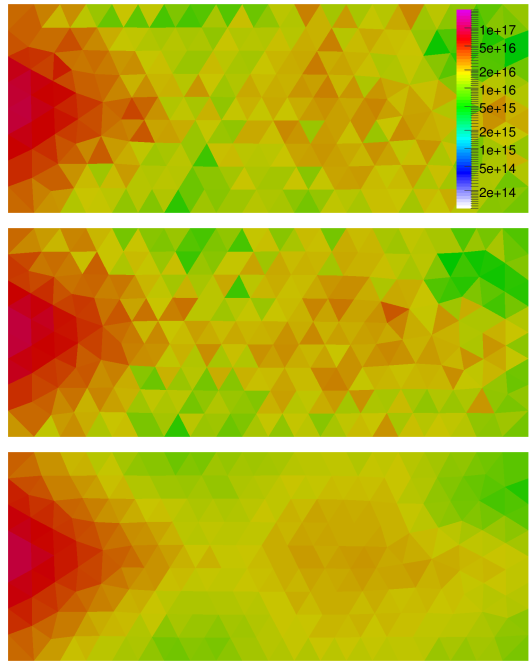


(a) W_1 (b) D_m

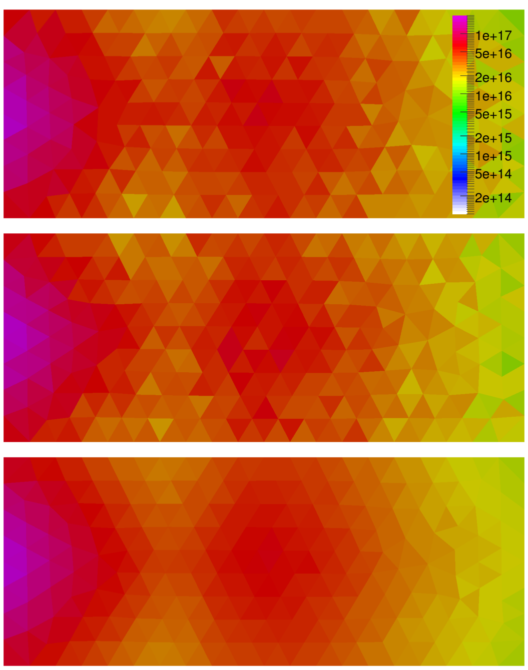
Figure 6. Comparison of ion flux distributions on the solar array (CP2). Two set of simulations with different number of simulation particles are performed on six different volume grid resolutions (0 to 5). Base simulation uses the particle weights given in Table 1, and the total simulation particles are increased by setting the particle weights to be 5×10^8 and 5×10^9 for ions and neutrals, respectively. Plotted are the metrics for determining the degree of discrepancy between simulations.



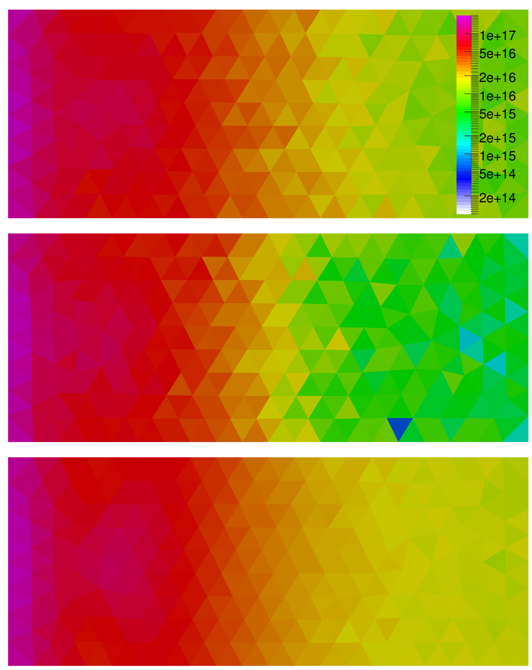
(a) TURF: Resolution 0.



(b) TURF: Resolution 1.

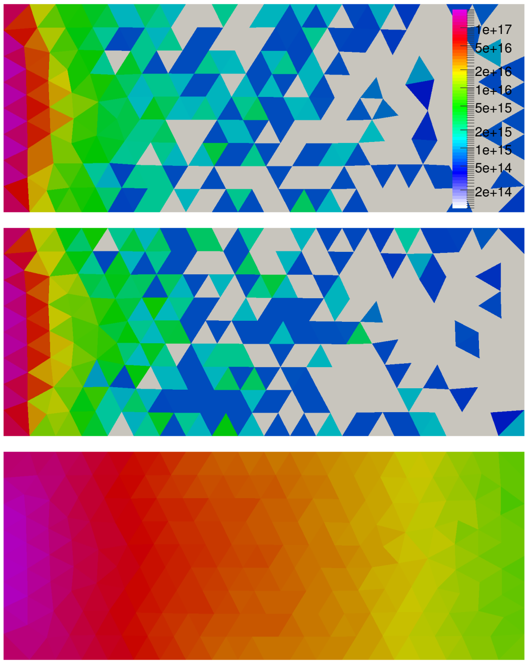


(c) TURF: Resolution 2.

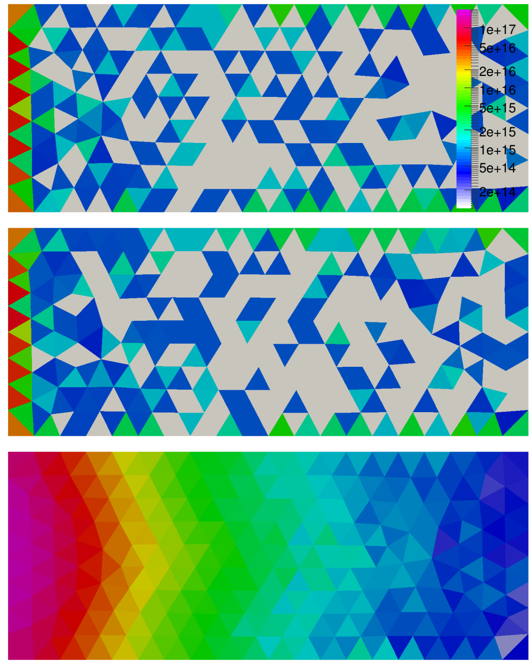


(d) TURF: Resolution 3.

Figure 7. Ion flux distribution on the solar array (CP2) obtained from TURF's simulations with different volume grid resolution (see Table 2). For each resolution, the top two figures are obtained from the base simulation with different random number seed, and the bottom figure is obtained from the simulation with increased number of simulation particles.

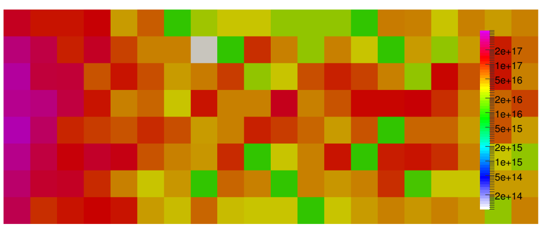


(a) TURF: Resolution 4.

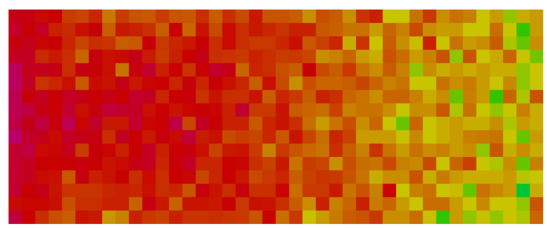


(b) TURF: Resolution 5.

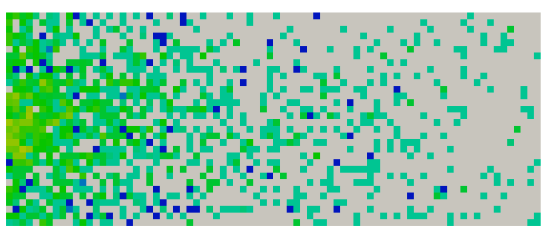
Figure 8. Ion flux distribution on the solar array (CP2) obtained from TURF's simulations with different volume grid resolution (see Table 2). For each resolution, the top two figures are obtained from the base simulation with different random number seed, and the bottom figure is obtained from the simulation with increased number of simulation particles.



(a) MUSCAT: Resolution 2.



(b) MUSCAT: Resolution 3.



(c) MUSCAT: Resolution 4.

Figure 9. Ion flux distribution on the solar array (CP2) obtained from MUSCAT.

accelerated by the potential gradient at high angles with respect to the thruster normal. The accuracy of the DSMC method drops when the number of collisions per cell is small. As the volume grid resolution is increased while maintaining the overall particle count, the number of collisions per cell becomes smaller, and this becomes even more severe near the plume edge, where the primary ions rarely reach. Therefore, increasing the number of simulation particles can potentially mitigate the unconverging behavior of the EP plume simulation. Figures 7 and 8 show the ion flux distributions on the solar array (CP2) for the six different volume grid resolutions for the base simulations and the simulations with increased number of particles. The differences in the distributions for Resolution 4 and 5 are clearly shown.

Figure 9 shows the ion flux distribution on the solar array (CP2) obtained from MUSCAT. While the magnitude is off by some factor for all the resolutions, the distributions are qualitatively similar between the two codes. MUSCAT predicts that the distribution changes significantly from Resolution 3 to 4, which is consistent with the results from TURF. For the discrepancy in magnitude, it is potentially attributed to the inconsistency with the collision rate calculation, but further investigation on the source of discrepancy is necessary.

D. Full Simulation: Difference in Implementation and Model

In the course of comparative study between TURF and MUSCAT, it is apparent that a slight modification of a numerical implementation or model can significantly change the ion flux distribution when it is primarily comprised of the CEX ions. The CEX collisions are rare events in that the number of CEX ions is significantly less than the number of primary ions. For this reason, the statistical noise becomes as large to mask important features in the ion flux distribution. In addition to the statistical noise, many other factors can contribute to the alternation of ion trajectories as described in Section II.F. The various improvements made to arrive at the current plume model have motivated an investigation into the sensitivity of the solution to the numerical implementations and models, even though the convergence has not been obtained for the ion flux distribution on the solar array (CP2). Seven different implementations and models are investigated, which are listed in Table 3 along with the detailed description of the cases. Figure 10 shows the two metrics (W_1 and D_m) as a function of the grid resolution, and Fig. 11 shows the ion flux distribution for Resolution 2. The distinctively different trend is seen in Case (c) in that both the metrics are relatively low up to Resolution 2 and start to increase with the refined resolution. As the mesh is refined, the number of particles per cell decreases, which amplifies the statistical noise and causes the solution to random walk. In the future, the time-averaged electric field will be used in the EP plume simulation with TURF. For the other cases, it is not clear from the metrics whether the ion flux distributions are significantly different. For Resolution 2, the EMD metric increases in the order of (c), (g), (e), (f), (d), (a), and (b), and this is visually in the order deviating from the base simulation result when closely looking at Fig. 11. Although the classical scattering

Table 3. Different implementations and models used in the sensitivity study. The ones used in the base simulation are given in parenthesis.

Case	Implementation/Model
(a)	Node-centered electric field (Cell-centered)[a]
(b)	Original treatment of near-wall region (Improved treatment to suppress numerical sheath)[b]
(c)	Time-averaged electric field (Instantaneous)[c]
(d)	Using target neutral positions as initial positions of CEX ions (Incident ion positions)[d]
(e)	Collision model: Monte Carlo collision method[e] (DSMC)
(f)	Collision model: Monte Carlo collision method with depletion of neutrals[f] (DSMC)
(g)	Collision model: Classical scattering with ab-initio potential (VHS and switching velocity vector)[g]

[a] The computational grid is shifted by half a cell so the cell-centers and nodes are located exactly on the same positions as the cell-centered and node-centered cases, respectively. The main difference is that the electric field is computed based on the charge density and potential calculated on positions offset by half the grid size. To compute the electric field on the cell or node (i,j) in 2D, the cell-centered case uses the potentials stored in cells (i,j), $(i-1,j)$, $(i+1,j)$, $(i,j-1)$, and $(i,j+1)$, while the node-centered case uses cells $(i-1,j-1)$, $(i,j-1)$, $(i,j-1)$, and (i,j) (see Ref. 3).

[b] Details are described in Section II.F.

[c] The simulation stores the instantaneous and time-averaged values for the charge density. The electric field can be computed based on either the value.

[d] In the DSMC method, a collision partner is randomly selected from the list of particles in the given cell. The initial position of an CEX ion can be assigned with either the position of the primary ion or the neutral atom. The base model uses the primary ion position, given that it would not be physical as the CEX ion can be created slightly off from the plume when using the neutral atom position.

[e] Standard MCC method as described in Section II.F.

[f] After the MCC calculation, particle weights of all the neutral atoms in the given cell are reduced according to the number of newly created physical CEX ions.

[g] The models used in the base simulation are the VHS method for the MEX collision and the exchange of velocity vector for the CEX collision. A higher-fidelity model for the elastic collision is the classical scattering with the ab-initio interatomic potential (see Ref. 19).

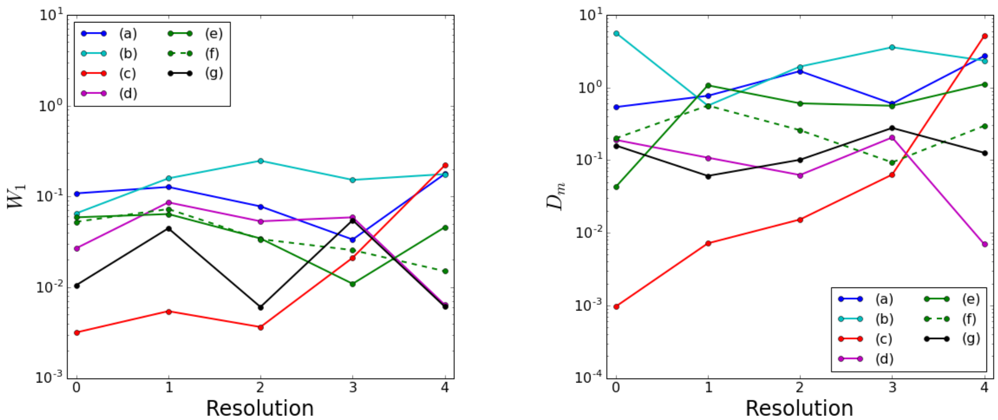
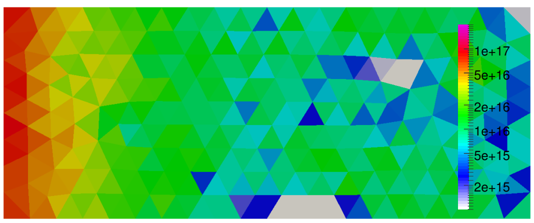
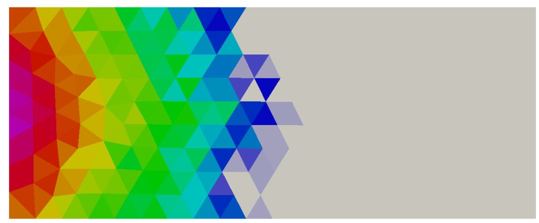


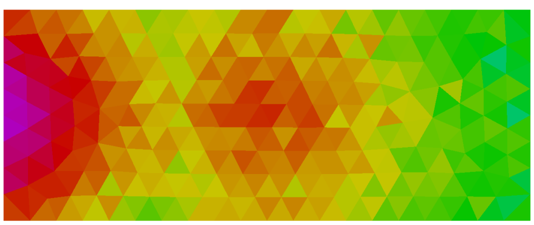
Figure 10. Metrics for degree of discrepancy, W_1 and D_m, as a function of the volume grid resolution. Seven different implementations/models are compared against the base simulation.



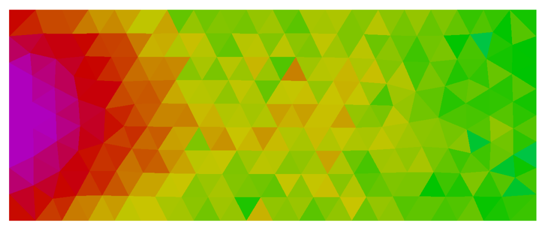
(a)



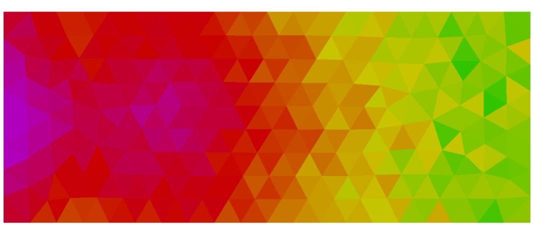
(b)



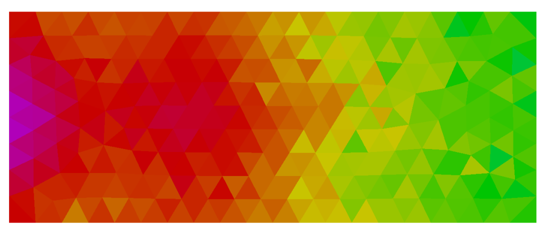
(c)



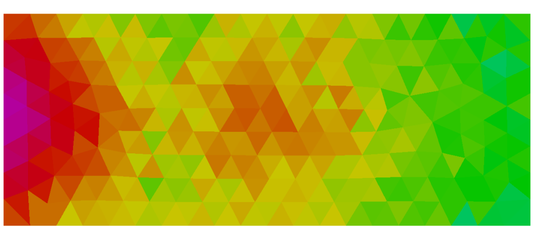
(d)



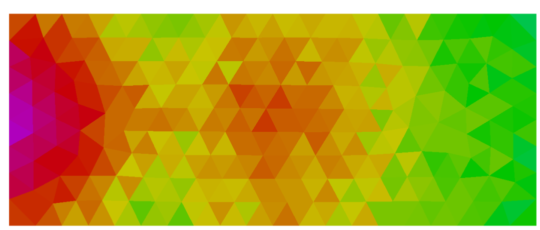
(e)



(f)



(g)



(h) Base Simulation

Figure 11. Ion flux distribution on the solar array (CP2) for Resolution 2 obtained with different implementations and models listed in Table 3. The letters in the subcaption correspond to the listed case number. The distribution from the base simulation is also shown in the bottom right figure.

model (Case (g)) produces the least distinctive distribution from the base simulation when excluding Case (c), D_m is approximately 0.1, indicating 10% difference in the total ion flux. For Cases (d), (e), and (f), the distributions are moderately different, which is attributed to the initial velocities or positions of the CEX ions when they are created. As expected, the MCC method over-predicts the ion flux magnitude, while the magnitude becomes comparable to the base DSMC model when neutral density is reduced based on the CEX ion generation rate. For Cases (a) and (b), the distributions are completely different in feature and magnitude. For Case (a), even though the electric field is computed at the same positions, the computed value is significantly different due to the mismatch of the electric potential data positions. The node-centered case roughly misses the peak of the injection distribution, under-predicting the electric field near the plume center. For Case (b), the numerical sheath near the spacecraft bus pulls the CEX ions, turning them toward the solar array and enhancing the ion-flux closer to the spacecraft bus.

For cases (a)–(d), it is expected that the solution converges with refined resolution of the volume grid and increased number of simulation particles. Nevertheless, it is often not possible to have a uniform mesh resolving the plume region, as the whole simulation domain typically spans tens of the thruster size. In keeping the particle count per cell, the number of simulation particles has to be increased according to the volume grid resolution. However, these would rapidly use up the available computer memory. In resolving this issue, one can use an unstructured mesh or a Cartesian mesh with adaptive mesh refinement (AMR), which can be the path forward for TURF, while MUSCAT already has the AMR capability. Together with the finer mesh resolution near the thruster exit, the number of particles for the collision calculation in the region can be increased while keeping the particle count far from the thruster relatively low by three different methods. The simplest method would be to delete particles outside of the important region as done in Ref. 20. The downside of this method is that it requires a priori knowledge of the origins of the CEX ions impacting the spacecraft. Another method is using the merge algorithm[21] to reduce the particle count far from the thruster. Lastly, the number of collisions can be artificially increased by the fractional collision algorithm developed recently.[22] In essence, this method splits the incident ions during the collision stage and collides these particles against the neutral atoms of different weights. As a result, the ratio of the CEX ions to the primary ions can be increased, which reduces the statistical noise of the CEX ion flux on the important spacecraft components while keeping the number of other species the same.

V. Summary and Future Work

This paper studied the convergence behavior of the EP plume simulation as the computational grid resolution was refined while keeping the particle count the same. A simplified plume simulation was set up in two different codes, TURF with the spacecraft module and MUSCAT, which used the stream Maxwellian injection of ions and neutrals and measured the ion flux distributions on the downstream plate and the solar array. The earth mover's distance and the relative difference in the integrated flux were used to quantify the discrepancy.

For the simulations with and without collisions, convergence of the distributions on the downstream plate was achieved. In attaining the converged solution, a grid resolution corresponding to at least four cells spanning along the thruster diameter were required. The results obtained from TURF and MUSCAT were consistent. When including the elastic collisions, it was not possible to obtain converged ion flux distribution on the solar array, which was contributed primarily by the CEX ions. This was likely due to the smaller number of particles per cell, leading to insufficient number of collisions required for accurate DSMC calculations especially near the plume edge. It was also shown that the ion flux distribution on the solar array was very sensitive to different implementations and models. Although the solutions are expected to converge when increasing the number of simulation particles, it is probably not possible to attain enough particle count without special methods. The proposed solution is to use an unstructured mesh or a Cartesian mesh with adaptive mesh refinement to increase the mesh resolution near the thruster. Together with that, the number of collisions can be increased by (1) only keeping the primary ions and neutral atoms near the thruster, (2) using the merge algorithm developed by Martin and Cambier,[21] or (3) employing the fractional collision algorithm recently developed by Araki and Martin.[22]

References

[1]Martin, R. S. and Koo, J. W., *Thermophysics Universal Research Framework - Infrastructure Release*, Air Force Research Laboratory (AFRL/RQRS), April 2015.

[2]Araki, S. J., Martin, R. S., Bilyeu, D., Koo, J. W., Tran, J., and Kawashima, K. A., *Tutorial for Thermophysics Universal Research Framework –TURF-IR 2017a–*, Air Force Research Laboratory (AFRL/RQRS), May 2017.

[3]Araki, S. J., Martin, R. S., Bilyeu, D. L., and Koo, J. W., "SM/MURF: Current Capabilities and Verification as a Replacement of AFRL Plume Simulation Tool COLISEUM," *52nd AIAA/SAE/ASEE Joint Propulsion Conference*, Salt Lake City, Utah, July 2016, AIAA 2016-4939.

[4]Araki, S. J. and Barrie, A. C., "Electric Propulsion Plume Simulation Coupled with Spacecraft Charging," *54th AIAA/SAE/ASEE Joint Propulsion Conference*, Cincinnati, Ohio, July 2018, AIAA 2018-4906.

[5]Muranaka, T., Hosoda, S., Kim, J.-H., Hatta, S., Ikeda, K., Hamanaga, T., Cho, M., Usui, H., Ueda, H. O., Koga, K., and Goka, T., "Development of Multi-Utility Spacecraft Charging Analysis Tool (MUSCAT)," *IEEE Transactions on Plasma Science*, Vol. 36, No. 5, 2008, pp. 2336–2349.

[6]Birdsall, C. K. and Langdon, A. B., *Plasma Physics via Computer Simulation*, IOP Publishing Ltd., 1991.

[7]Birdsall, C. K., "Particle-in-Cell Charged-Particle Simulations, Plus Monte Carlo Collisions With Neutral Atoms, PIC-MCC," *IEEE Transactions on Plasma Science*, Vol. 19, No. 2, 1991, pp. 65–85.

[8]Allison, D., Baldwin, J., and Sharfe, M., "Development of BEPPA: An Object-Oriented Parallel Code for Full 3-D Spacecraft Plume Analysis and Satellite Design," *44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, Hartford, Connecticut, July 2008, AIAA 2008-4999.

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

[10]Nanbu, K., "Stochastic Solution Method of the Boltzmann Equation II. Simple Gas, Gas Mixture, Diatomic Gas, Reactive Gas, and Plasma," *Reports of the Institute of Fluid Science*, Vol. 8, Tohoku University, March 1996, pp. 77–125.

[11]Nanbu, K., "Probability Theory of Electron-Molecule, Ion-Molecule, Molecule-Molecule, and Coulomb Collisions for Particle Modeling of Materials Processing Plasmas and Gases," *IEEE Transactions on Plasma Science*, Vol. 28, No. 3, 2000, pp. 971–990.

[12]Boyd, I. D. and Stark, J. P. W., "On the Use of the Modified Nanbu Direction Simulation Scheme," *Journal of Computational Physics*, Vol. 80, 1989, pp. 374–386.

[13]Rubner, Y., Tomasi, C., and Guibas, L. J., "A Metric for Distributions with Applications to Image Databases," *6th International Conference on Computer Vision*, IEEE, Bombay, India, 1998, pp. 59–66.

[14]Rubner, Y., Tomasi, C., and Guibas, L. J., "The Earth Mover's Distance As a Metric for Image Retrieval," .

[15]Bonneel, N., van de Panne, M., Paris, S., and Heidrich, W., "Displacement Interpolation Using Lagrangian Mass Transport," *Proceedings of the 2011 SIGGRAPH Asia Conference*, SA '11, ACM, New York, NY, USA, 2011, pp. 1–12.

[16]Flamary, R. and Courty, N., "POT Python Optimal Transport library," 2017.

[17]Benamou, J.-D., "Numerical Resolution of an Unbalanced Mass Transport Problem," *ESAIM: Mathematical Modelling and Numerical Analysis*, Vol. 84, No. 3, 2003, pp. 851–868.

[18]Gangbo, W., Li, W., Osher, S., and Puthawala, M., "Unnormalized Optimal Transport," February 2019, arXiv:1902.092267v1.

[19]Araki, S. J. and Wirz, R. E., "Ion-Neutral Collision Modeling Using Classical Scattering With Spin-Orbit Free Interaction Potential," *IEEE Transactions on Plasma Science*, Vol. 41, No. 3, 2013, pp. 470–480.

[20]Araki, S. J., "Sensitivity of Ion Flux on Spacecraft to Neutral Density in Hall Thruster Plume," *35th International Electric Propulsion Conference*, Georgia Institute of Technology, USA, 2017, pp. 1–12, IEPC-2017-539.

[21]Martin, R. S. and Cambier, J.-L., "Octree Particle Management for DSMC and PIC Simulations," *Journal of Computational Physics*, Vol. 327, 2016, pp. 943–966.

[22]Araki, S. J. and Martin, R. S., "Interspecies Fractional Collisions," *Vacuum*, Under Review.