

A Novel Approach for the Particle-in-Cell Modelling of Gridded Ion Engine Plume Neutralisation

J. I. Parke Sturrock¹, Z. Jelic¹, B. J. Evans¹ and R.J. Davis¹

¹Swansea University, Department of Aerospace Engineering, Bay Campus, Fabian Way, Swansea, SA1 8EN.

(Received: 02. Nov. 2021, Accepted: 24. Dec. 2021, Published online: 28. Dec. 2021)

ABSTRACT: The Particle-in-Cell modelling of gridded ion engine plume neutralisation has been simplified when compared to traditional methods. This results in significantly less computational resources being required. The NSTAR engine was modelled as a reference, where simulated specific impulse values were found to be 5% higher than the real engine. This method will be most suited to rapid prototyping and optimisation studies, where speed of simulations is an important factor.

(DOI: 10.31281/jtsp.v2i1.21)

868466@swansea.ac.uk

I. Introduction

Tsiolkovsky's Rocket Equation, Eq. (1), describes how a reaction engine operates.

$$\Delta V = V_e \ln \left(\frac{m_0}{m_f} \right) \quad (1)$$

The term ΔV is the desired change of velocity that a rocket or spacecraft requires to complete a given manoeuvre [1]. This is irrespective of the method of travel, due to the intrinsic relationship between velocity and orbital paths. The terms m_0 and m_f are the initial and final mass of the vehicle, respectively, where the difference in these values is the theoretical minimum amount of propellant required to complete the manoeuvre. The final parameter, V_e , is the reaction engine's exhaust velocity. This is often divided by standard gravity, g_0 (9.81 ms^{-2}), to give the value of specific impulse (I_{sp}). This is shown in Eq. (2). I_{sp} is the only controllable value, as it is governed by the propulsion system used, as well as which propellant it utilizes.

$$I_{sp} = \frac{V_e}{g_0} \quad (2)$$

Due to the exponential nature of Eq. (1), small increases in I_{sp} can lead to dramatic reductions in the amount of propellant required to obtain a given ΔV . SpaceX's Raptor engines have an I_{sp} of 374 s, which is a decent value for a hydrocarbon based fuel, in this case liquid methane [2]. Liquid

hydrogen (LH_2) engines have the highest I_{sp} of any widely used chemical propellant, with the European Space Agency's (ESA) VINCI engine having a value of 465 s [3]. LH_2 has a very low density (70 kgm^{-3}) and exists only at extreme cryogenic temperatures, below 20 K [4]. This leads to very large and bulky fuel tanks that require high levels of insulation. This increases the dry mass and drag of a rocket considerably, but the relatively small increase in I_{sp} (~25%) means LH_2 is the fuel of choice for a large number of launch vehicle upper stages. In order to increase propellant efficiency further, a new class of propulsion was created – electric propulsion (EP) [5].

Unlike chemical propulsion, EP devices do not rely on combustion to provide kinetic energy. Electrothermal devices are similar to chemical thrusters but use electrical heating to raise a propellant's potential energy rather than combustion. Electrostatic and electromagnetic thrusters use very different methods, however. Instead, they use electromagnetic forces to accelerate charged particles to far higher speeds than is possible with chemistry or electrothermal methods. This leads to orders of magnitude better propellant efficiency, resulting in smaller spacecraft that require less massive rockets for launch. The first contemporary ion engine, NASA's NSTAR thruster, has a specific impulse of 3200 s [6]. ESA has a design that has been experimentally ground tested, the Dual-Stage 4-

Grid (DS4G), with a potential specific impulse of 19 300 s, equivalent to an exit velocity of 189 km s^{-1} , more than 40 times that of VINCI [7].

NSTAR is a gridded ion engine. These thrusters first ionise a propellant, usually xenon, to create a plasma. These ionised particles are then accelerated through two charged grids by the force of electrostatic attraction. The first grid is charged so that it extracts the xenon ions out of the plasma but contains the electrons. The positive ions then pass through the second grid, which has the opposite charge. This accelerates the xenon ions out of the engine, imparting a thrust on the spacecraft by virtue of conservation of momentum. Once the ions have left the engine, they require neutralisation by electron bombardment. This stage is vital, else conservation of charge would be violated, causing the spacecraft to become electrically charged [8]. Fig. 1 shows a diagram of how the NSTAR engine operates.

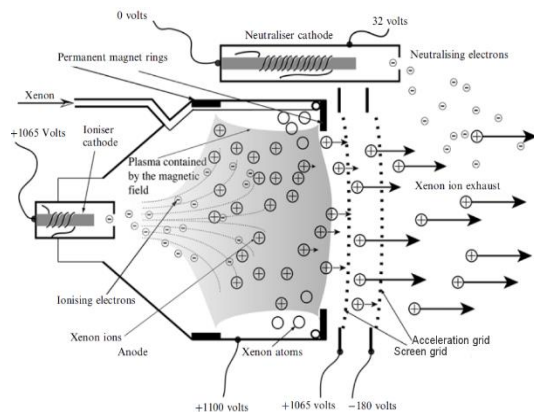


Figure 1: Diagram showing how the NSTAR engine operates [9]. This is a typical gridded ion engine.

Table 1 lists key parameters for this engine. All values are found in NASA's NSTAR Technology Validation Report for throttle level 15 (maximum) [6].

Engine Parameter	Value (Units)
Input Power	2567 (W)
Specific Impulse	3200 (s)
Total Impulse	2.73 (MNs)
Thrust	92.67 (mN)
Total Propellant Throughput	88 (kg)
Main Propellant Flow Rate (Xenon)	23.43 (sccm)
Cathode Flow Rate (Xenon)	3.70 (sccm)
Neutraliser Flow Rate (Xenon)	3.70 (sccm)

Table 1: NSTAR engine parameters

A common algorithm used for modelling plasma for spacecraft propulsion is the Particle-in-Cell method, first created in the 1950's [10]. The

stages of this algorithm will be discussed in the methodology section of this paper. Full-fidelity simulations performed on high-performance computers can take a month of continual simulation time, even in only two spatial dimensions [11].

In order to reduce code run times to a more manageable level, numerical acceleration techniques have been studied [12]. These include modifications such as artificially increasing the permittivity of vacuum, artificially increasing the mass of plasma ions, moving ions only after a certain multiple of time-steps (sub-cycling), as well as others. These can be combined to reduce run times by orders of magnitude.

The purpose of this paper is to introduce a novel numerical acceleration technique. This method is applicable to the plume neutralisation process of an ion engine. The studies performed were for gridded ion engines, however it is plausible that the method would be applicable to Hall Effect thruster PIC studies too, as the neutralisation process is similar. More detail is given near the end of the methodology section of this paper, but essentially this novel method approximates the neutralisation process. This involves only using one position coordinate check per each emitted xenon ion. This eliminates the need to introduce neutralising electrons emitted from the neutraliser, reducing particle loads substantially.

II. Methodology

The studies conducted for this paper use the Particle-in-Cell (PIC) method [13]. Fig. 2 illustrates how the essential parts of the algorithm function.

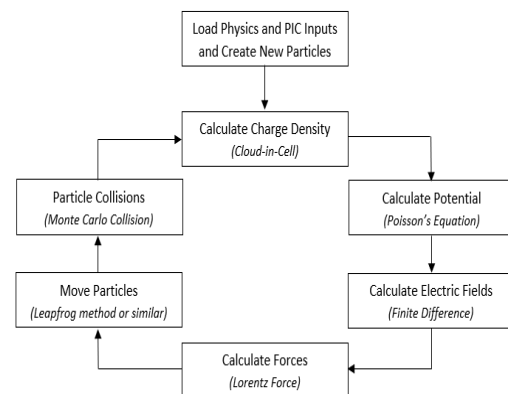


Figure 2: Flow chart explaining the core stages of the algorithm for the electrostatic Particle-in-Cell method. This was conducted in the studies for this paper.

The method models the constituent particles of a plasma (xenon ions and electrons) and how they interact with each other. To keep the number of particles in the system realistic, each computational particle, or 'super particle' represents many thousands of real particles. Real and simulated particles behave the same, as their charge-to-mass ratio is equal. This is seen in Eq. (3), where Z is this ratio, q is the particle's charge and m is its mass.

$$Z = \frac{q}{m} \quad (3)$$

The plasma density close to NSTAR's ionising hollow cathode is approximately 10^{17} m^{-3} , which gives a plasma frequency of $1.783 \times 10^{10} \text{ Hz}$ [14]. This is calculated from Eq. (4).

$$f_{\text{plasma}} = \sqrt{\frac{n_p q_e^2}{m_e \epsilon_0}} \quad (4)$$

where n_p is the plasma density, q_e is the charge of an electron, m_e is the mass of an electron and ϵ_0 is the permittivity of free space. In order for the simulation to be stable, the time step size must be the inverse of this frequency, in this case 5.606×10^{-11} seconds. The discharge current of NSTAR's main hollow cathode is 13.13 A [14]. This is equal to 13.13 Cs^{-1} . As one coulomb of charge is equivalent to 6.242×10^{18} electrons, 8.195×10^{19} electrons are being emitted by the cathode per second. Therefore, 4.594×10^9 electrons are emitted by the cathode every time step. The number of electron super particles entering the domain every time step was set to 100, above the recommended minimum of 60 [15]. The specific weight of each computational particle is 4.594×10^7 .

As these particles are electrically charged, they create their own dynamic electric field which changes every time step. This is added to the externally applied field (from the charged grids). In order to calculate these self-fields, the charge density must be known. To achieve this, the simulation domain is divided into a regular grid pattern (creating square cells). Each particle's charge is then assigned to the four nearest grid points, weighted linearly based on the particle's proximity to each node. This is called the Cloud-in-Cell algorithm [16]. Potential can then be calculated at each grid point, using Poisson's Equation, Eq. (5).

$$\Delta\phi = -\frac{\rho}{\epsilon_0} \quad (5)$$

Δ is the Laplacian Operator, ϕ is the potential to be solved and ρ is the charge density. Electric field strength for each node is then the average potential of neighbouring nodes, divided by the length of each cell.

The electrostatic Lorentz Force drives the movement of particles (Eq. 6), where F is the force acting on the particle and E is the strength of an electric field that the particle of charge q experiences.

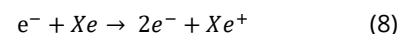
$$F = qE \quad (6)$$

Newton's second law is then used to calculate the acceleration, a . Combining these equations gives rise to Eq. (7). As can be seen, this further simplifies to the charge-to-mass ratio multiplied by the electric field strength.

$$a = \frac{qE}{m} = EZ \quad (7)$$

Multiplying the acceleration components by the time step size gives a new velocity, that leads to a new position. This completes the particle moving section.

Finally, particle collisions need to be considered. The two main algorithms are Monte Carlo Collision (MCC) and Direct Simulation Monte Carlo (DSMC) [17] [18]. The code for this paper uses MCC, which is much less computationally expensive than DSMC. Unlike DSMC, MCC allows the neutral propellant particles within the engine cavity to be unsimulated, which reduces particle counts massively when compared to DSMC. Any electrons within the simulation have a probability of causing an ionisation event. This event creates a xenon ion and one additional electron. This is described by the ionisation formula below.



where e^- is an electron, Xe is a neutral xenon propellant atom and Xe^+ is a xenon ion. Eq. (9) describes this 'collision probability' (P). n_n is the neutral propellant gas number density, σ is collisional cross-sectional area, g is the electron's velocity magnitude and Δt is the time since the last collision check. Once the probability has been determined, a random number is assigned. If this number is lower than the collision probability then the ionisation event occurs.

$$P = 1 - \exp(-n_n \sigma g \Delta t) \quad (9)$$

Collisional cross section is dependent on the velocity of the ionising electron, which has to be

determined experimentally. This data is available, however does not follow a particular function [19]. Average electron energies in the studies for this paper were found to be near 48 eV, corresponding to a collisional cross section of $4.75 \times 10^{-20} \text{ m}^2$, which was used for all calculations. This approximation saves large computational expense, as otherwise each electron would have to be cross referenced to a database at every time step. As there are typically hundreds of thousands of electrons, as well as time steps, this would lead to far longer code run times. Reducing run time is the focus of this paper, as the code is primarily intended for fast PIC simulations to evaluate multiple ion engine designs at an early stage of development.

This core algorithm has been implemented in the code for this paper. The plume neutralisation process is often performed by simulating electrons emitted from the neutralising hollow cathode [20]. DSMC is then performed to calculate whether a collision between an emitted xenon ion and a neutralising electron occurs, which results in both particles combining to become a neutral xenon atom. This extra particle load, along with the DSMC algorithm, leads to a significant amount of extra computational expense to model the neutralisation process.

The 3D model of the NSTAR assembly, complete with neutralising cathode, is seen in NASA's Technology Validation Report [6]. Based on measurements of this model, it is estimated that the bulk of neutralisation will occur downstream of the thruster exit, in a region approximately 30-40% the length of the engine past the exit. Converting xenon ions to neutral xenon atoms after a certain distance allows for the entire neutralisation process to be modelled in one simple step, bypassing the need for DSMC neutralisation, as well as the associated neutralising electrons. The term 'neutral plane' will now be introduced. This is a vertical line in the simulation domain with an x-value equal to a fraction of total engine length. Any xenon ions downstream of this line are converted to xenon atoms, simulating the neutralisation process. Fig. 3 shows the results of a study with the neutral plane value set to 1.3. This value is indicated by the dashed line. As NSTAR's length is 25cm, the neutral plane is set to an x value 30% larger than this – 32.5 cm [6]. Xenon ions are represented in red, neutral xenon atoms are coloured in magenta.

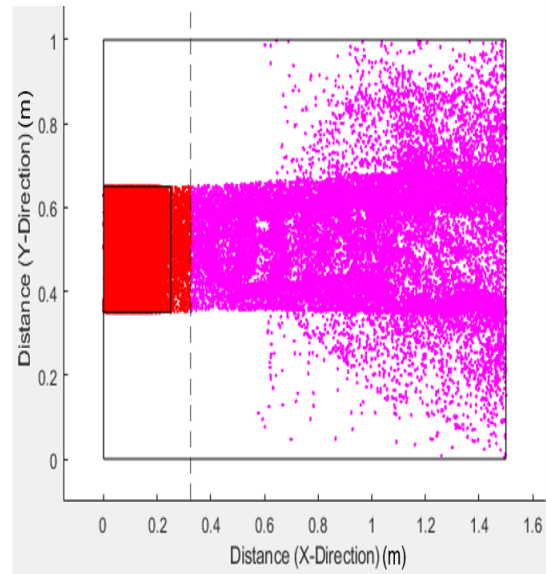


Figure 3: Graphic showing the concept of the neutral plane (shown by the dashed line). Any xenon ions (red) that travel downstream of this plane are converted to xenon atoms (magenta). NSTAR is represented by the black rectangle.

The simulations performed for this paper used Fortran 90. Post-processing of each test case was conducted using a custom MATLAB code and comparisons of multiple test cases used Excel.

III. RESULTS AND DISCUSSION

As the exact position of the neutralising hollow cathode's orifice is unknown, a range of studies were performed. The value of the neutral plane was set to 1.1 – 1.5, with increments of 0.2.

Each study was performed for one million time steps, with a quasi-steady state being reached after around 400 000 time steps. Total run times for the codes were between 2:55 h and 3:05 h. The number and type of each particle was recorded. Additionally, the values of specific impulse and divergence angles were also calculated. Specific impulse is a measure of the axial exit velocity of each xenon atom. Divergence angles are the mean average of all particles' angles with respect to the engine. If it is zero, then it means the particle's y component of position is between the top and bottom of the engine. Experimentally, the divergence angle of a gridded ion thruster is defined as the profile of 95% of the emitted beam current [14]. This effectively removes outliers, so this has been implemented in post-processing too. Mass flow rates were not recorded for this study, as

changing the value of the neutral plane has negligible influence on the electric fields within the engine, which drive the ionisation process and therefore mass flow rates. Fig. 4 shows the particle count of the 1.3 neutral plane test case.

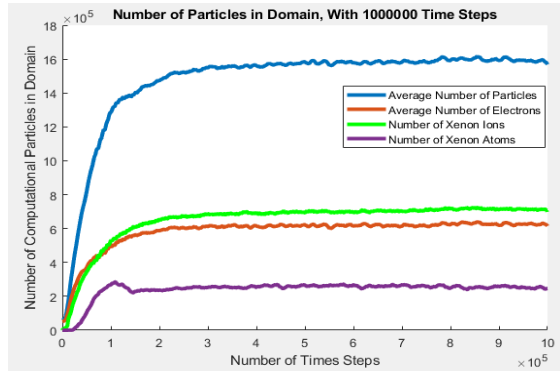


Figure 4: Graph showing how the numbers of each particle species, as well as total number of particles, varies with time. Quasi-steady state reached after 400 000 time steps. Neutral plane is again 1.3 for this example.

The number of electrons fluctuates frequently, so a moving mean was conducted to smooth out variance. The window for the moving mean was set to 10 000 time steps. This was not necessary for the xenon species.

Total particle count is quasi-steady state at around 1.6 million total particles, which was the same for all test cases. The number of xenon ions tends to be slightly larger than the number of electrons present. This is more pronounced the larger the value of neutral plane. Inside the real engine, quasi-neutral plasma (equal number of xenon ions and electrons) exists. Within the simulated engine there is a slight excess of ions, however the source of this could not be determined. Despite this, the noticeable difference of ions to electrons can mostly be attributed to the fact that any ion that has left the engine but has yet to be neutralised will be counted. Electrons are prevented from leaving the engine cavity due to electrostatic repulsion of the screen grid, so every electron counted here is within the engine plenum.

Fig. 5 shows a comparison of mean specific impulse as a function of the value of the neutral plane. Due to variation in the number of each particle species, specific impulse was calculated only for the steady state region (400 000+ time steps). The blue line shows this mean value for each test case, whereas the orange line shows the real engine's specific impulse value (3200 s).

Increasing the value of the neutral plane causes specific impulse to converge to around 3350 s.

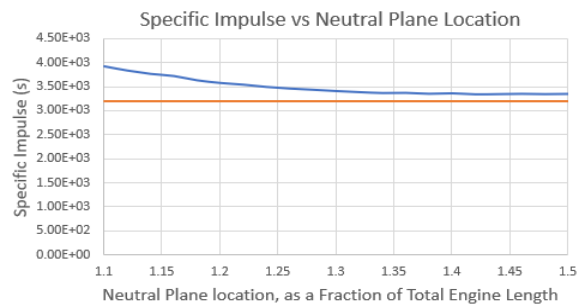


Figure 5: Graph showing how changing the value of neutral plane affects the mean specific impulse value of the simulated NSTAR engine. Blue line shows the simulated values for each neutral plane value. The orange line shows the real NSTAR engine's specific

Due to the exact position of NSTAR's neutralising hollow cathode being unknown, it is not possible to select an exact neutral plane value that is known to be representative. Based off the 3D model measurement estimates, the value of a neutral plane would likely be between 1.3 and 1.4. This gives a specific impulse range of between 3405.6 s and 3358.5 s, corresponding to an overestimate of between 6.4% and just under 5.0%, respectively. The overestimation of specific impulse could potentially be caused by the unknown issue of the number of xenon ions within the engine being slightly higher than the number of electrons. A slight excess of xenon ions likely causes the emitted ions to be travelling slightly faster than they should do, as the excess positive charge will cause them to be accelerated more due to electrostatic repulsion. Specific impulse is overestimated rather significantly when the neutral plane is set to 1.2 or less. This is likely due to the fact that the particles do not experience enough change in velocity in the y direction before being neutralised (after which acceleration is zero). Therefore, the atoms' velocity is disproportionately biased in the x direction, which directly increases specific impulse.

Fig. 6 shows a comparison of the mean divergence angles of the same simulations. As a large proportion of non-outlying particles are neither above the top of the engine, nor below the bottom, their values of divergence angle are zero. This leads to a relatively low divergence. The actual beam divergence of the real NSTAR engine is around 5 degrees [6]. While not a smooth function, a clear link can be made between how increasing the value of the neutral plane

exponentially increases divergence angle. Neutral plane values of 1.10, 1.12 and 1.14 created zero divergence at all (due to the removal of outliers).

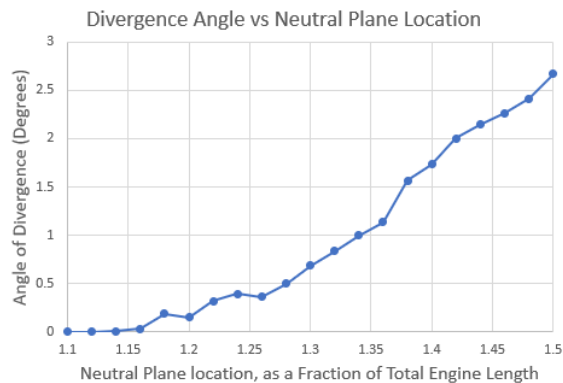


Figure 6: Graph showing how changing the value of neutral plane affects the angle of beam divergence.

The simulated divergence angle is not representative of the real engine. With a divergence of 5 degrees, this would imply that the real engine's emitted beam current has a higher component of radial velocity, which is both unavoidable and undesirable. If the simulated engine had a divergence value closer to the actual engine then this would reduce axial velocity slightly, bringing the specific impulse closer to the real value. It is important to note that the neutral propellant being emitted is non-collisional. If it was, the divergence angle would likely be considerably higher (possibly even higher than 5 degrees). Collisional checks are generally computationally expensive, so they were omitted from the code, which focuses on increasing the speed of the PIC method.

IV. Conclusion

This paper has demonstrated a novel approach to circumventing a computationally expensive part of the PIC algorithm for gridded ion engine plume neutralisation simulations. Specific impulse values are within 5% of the real NSTAR engine. Code execution times are around three hours. This technique will be particularly valid for rapid prototyping, for example evaluating many potential propellants or optimising grid voltage settings for new thruster designs.

A logical next stage of evolution would be to expand the neutral plane to a neutral 'zone' – a region where a particle is given a varying neutralisation probability, based on its distance

from the thruster exit. This would work in a similar way to Monte Carlo Collisions. This has not yet been attempted as more information on the neutralising cathode would be required.

V. Notes

The codes for this paper are under development for the lead author's ongoing Doctor of Philosophy in Aerospace Engineering. Source codes of the PIC studies and post-processors will be available to the public in 2024, upon completion of studies.

VI. References

- [1] D. Pettit, The Tyranny of the Rocket Equation [Webpage]. NASA, 2012. Available at: www.nasa.gov/mission_pages/station/expeditions/expedition30/tryanny.html
- [2] N. Brugge. "Raptor Engine, Rocket Propulsion", b14643 Database. [Webpage] Available at: http://www.b14643.de/Spacerockets_2/United_States_1/SpaceX_ITS/Propulsion/engines.html
- [3] D. Greuel, K. Schafer, S. Schlechtrie, "Test Facilities for VINCI", CEAS Space J, Volume 5, Pages 39-48, 2013. <https://10.1007/s12567-013-0043-8>
- [4] F. Gregory, "Safety Standard for Hydrogen and Hydrogen Systems", NASA Office of Safety and Mission Assurance, 1997.
- [5] R. J. Cybulski et al, "Results from SERT I Ion Rocket Flight Test", NASA Technical Note D-2718, Lewis Research Centre, 1965.
- [6] J. Brophy et al, "Deep Space 1 Technology Validation Report - Ion Propulsion System (NSTAR)", NASA Jet Propulsion Laboratory, March 2001.
- [7] C. Bramanti et al. "The Innovative Dual-Stage 4-Grid Ion Thruster Concept – Theory and Experimental Results", International Astronautical Federation, 2006. <https://10.2514/6.IAC-06-C4.4.07>
- [8] J. Wang, J. Polk, D. Brinza, "Ion Thruster Modeling: Particle Simulations and Experimental Validations", AIP Conference Proc.; Journal Volume: 663; Journal Issue: 1, Conference 23, 2003. <https://10.1063/1.1581590>
- [9] K. Mani, F. Toppoto, A. Cervone, "Dual Chemical Electric Propulsion System Design for Interplanetary CubeSats", 6th Space

- Propulsion Conference, 2018. UUID: e4f297a2-36b1-4d9e-95b2-0befc0d07d7c
- [10] M.W. Evans, FH. Harlow, "The Particle-in-Cell Method for Hydrodynamic Calculations", Los Alamos Scientific Laboratory Report, 1957. Report Number: LA-2139.
- [11] T. Lafleur, P. Chabert, "The Role of Instability-Enhanced Friction on 'Anomalous' Electron and Ion Transport in Hall-Effect Thrusters", Plasma Sources Sci. Technol., Volume 27, Number 1, December 2017. <https://10.1088/1361-6595/aa9efe>
- [12] T. Yuan, J. Ren, J. Zhou, Z. Zhang, Y. Wang, H. Tang, "The Effects of Numerical Acceleration Techniques on PIC-MCC Simulations of Ion Thrusters", AIP Adv., Volume 10, Issue 4, April 2020. <https://10.1063/1.5113561>
- [13] C. K. Birdsall, AB. Langdon, "Plasma Physics via Computer Simulation", Taylor and Francis Publishers, 2004.
- [14] D.M. Goebel, I. Katz, "Fundamentals of Electric Propulsion: Ion and Hall Thrusters", Wiley Publishers, 2008.
- [15] V. Croes et al, "The Effect of Alternative Propellants on the Electron Drift Instability in Hall-Effect Thrusters: Insight from 2D Particle-In-Cell Simulations", Phys. Plasmas, Volume 25, Issue 6, April 2018. <https://10.1063/1.5033492>
- [16] J. Li, Y. Zhang, J. Wu, Y. Cheng, X. Du, "Particle Simulation Model for Self-Field Magnetoplasmadynamic Thruster", Energies, Volume 12, Issue 8, April 2019. <https://10.3390/en12081579>
- [17] C.K. Birdsall, "Particle-in-Cell Charged-Particle Simulations, Plus Monte Carlo Collisions with Neutral Atoms, PIC-MCC", IEEE Trans. Plasma Sci., Vol. 19, Issue 2, 65-85, 1991. <https://10.1109/27.106800>
- [18] C.D. Munz et al, "Coupled Particle-In-Cell and Direct Simulation Monte Carlo Method for Simulating Reactive Plasma Flows", Comptes Rendus Mécanique, Vo. 342, Issue 10, Pages 662-670, 2014. <https://10.1016/j.crme.2014.07.005>
- [19] J.A. Syage, "Electron-Impact Cross Sections for Multiple Ionization of Kr and Xe" Phys. Rev. A, Volume 46, Number 9, November 1992. <https://10.1103/PhysRevA.46.5666>
- [20] G. Cai, H. Zheng, L. Liu, X. Ren, B. He, "Three-Dimensional Particle Simulation of Ion Thruster Plume Impingement", Acta

Astronautica, Volume 151, Pages 645-654, 2018.

<https://10.1016/j.actaastro.2018.07.007>



Open Access. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit: <http://creativecommons.org/licenses/by/4.0/>.