Low orbit plasma-satellite interaction modelling

Richard Marchand
University of Alberta, Canada
E-mail: Richard Marchand@ualberta.ca

Abstract. A model is presented for describing the interaction of satellites with space plasmas. The model relies on a combination of two computational approaches; a Particle In Cell (PIC) code to describe electrons and ions fully kinetically, and a particle in cell capable of calculating detailed particle distribution functions with minimal statistical noise. Both models use unstructured tetrahedral meshes to discretise simulation domains and spacecraft geometry with realistic shapes. The PIC code solves for the self-consistent electric fields given calculated volume charges and charge deposition on the various satellite components. The test-particle code is used to calculate particle distribution functions from fields obtained from PIC simulations, with minimal statistical errors. For simplicity, magnetic fields are assumed to be negligible.

1. Introduction: Why spacecraft-plasma modelling?

In situ measurements of space place plasmas provide a powerful means of characterising and understanding the complex dynamics of these systems. Such measurements are increasingly available, with the many science satellites that have been launched in recent years, and the one scheduled for deployment in the coming years. When interpreting *in situ* measurements, however, it is important to account for the fact that the spacecraft, and often the measuring instruments themselves, are part of the plasma environment. As a result, measurement do not represent the state of the plasma that would exist in the absence of the satellite, and a proper interpretation may require a careful analysis to correct for, or at least assess the perturbation effects. Several models were developed over the years to describe the interaction of satellites with plasmas. A first class of models is focused on spacecraft charging, and their use was primarily for assessing the likelihood of arcing and failure of certain components. Another class of models is aimed at simulating electromagnetic fields near spacecrafts and their instruments for the purpose of characterising their interaction with surrounding plasma and assessing perturbation effects. If needed, these models can be used to correct for perturbation effects and hence, enhance the interpretation of the measurements. The model presented here is in that latter category. It is used to calculate the electrostatic sheath surrounding a satellite and its instruments. These fields in turn can be used to calculate the effect of the sheath on measurements such as particle distributions and plasma flows. For

simplicity, we limit our attention to plasma conditions that are characteristic of low orbit satellites, for which photoionisation and secondary electron effects are negligible. Also, for simplicity, the model is electrostatic. It doesn't account for magnetic perturbations and the background magnetic field is ignored.

2. Numerical model

The simulation tool presented here is based on a particle in cell (PIC) code to compute the electric fields self-consistently for given satellite geometry and set of plasma parameters. This model uses an unstructured tetrahedral mesh capable of representing satellites and simulation domains with arbitrary boundaries and boundary conditions. The mesh is also adaptive, as its resolution can be varied in space so as to better resolve smaller features on the satellite, or regions where plasma parameters vary more abruptly. Adaptivity, however, is static in time. The PIC approach used here follows standard similar approaches, whereby simulation particles represent macro-particles, or groups of particles. This is done so that, for a given density and temperature, physical quantities such as the plasma frequency, the Debye length and the thermal velocity are preserved. This is accomplished by tracking particles with physical charges and masses, and by attributing each species a statistical weight corresponding to the effective number of actual (physical) particles represented by every simulation particle.

2.1. Initial particle distributions

An arbitrary number of electron and ion species can be included in the model. Each species is characterised by a density, a temperature and a drift velocity. Ion species are also characterised by a charge and mass. The code is run with realistic electron and ion charge and masses. In the initial state of a simulation, all particle species are distributed uniformly through the simulation domain. This is accomplished by distributing a given number of particles for each species in each tetrahedral mesh element. The number of particles in a given element k is given by

$$\tilde{n}_{s,k} = N_s int\{\frac{v_k}{V}\} + \delta, \tag{1}$$

where $int(x)$ is the largest integer not exceeding x, v_k is the volume of the element and V is the total simulation domain, and δ is either 0 or 1 with the probability of 1 being$N_s v_k/V - int(N_s v_k/V)$. The algorithm used to distribute a point randomly in a tetrahedron is given in Appendix A. After initialisation, each particle species is distributed approximately uniformly throughout the volume and, provided that the input parameters be set appropriately, the plasma will be neutral. The statistical weight attributed to particle species s is given by $w_s = n_s V/N_s$, where n_s is the physical number of particles per unit volume for species s. Velocity distributions correspond to drifting Maxwellians corresponding to the densities, temperatures and drift velocities specified for each species. Particle pushing is done with a standard leap frog algorithm.

2.2. Partition into satellite components and mutual capacitances

The model can account for a satellite with different electrical components that can be left floating or that can be electrically biased with respect to one another. Each component is represented in terms of surface elements or, in practice, in terms of sets of contiguous triangles on the satellite surface. In solving for Poisson's equation, it is necessary to know the mutual capacitance matrix C for these various elements. This matrix is such that, in the absence of volume charge density, the charge q_i on component i is related to the potential V_j of other components j through

$$q_i = \sum_{j=1}^{N} C_{ij} V_j, \tag{2}$$

where N is the number of electrical components on the spacecraft. The matrix elements C_{ij} are calculated as follows:

(i) For each surface component j , we solve the Laplace equation $\nabla^2\phi = 0$ with boundary condition $\phi = 1V$ on component j and $\phi = 0$ on every other boundary element (including the outer simulation domain boundary).

(ii) The corresponding charge on each surface element is then obtained from

$$q_j = C_{ij} = \frac{1}{\epsilon_0} \int da\vec{E} \cdot \hat{n}, \tag{3}$$

where ϵ_0 is the permittivity of free space and $\hat{n}$ is the unit vector perpendicular to the surface, pointing outward. Owing to the linear superposition of electrostatic fields, the charge q_i on element i corresponding to an arbitrary distribution of voltages V_j must then be given by

$$q_i = \sum_{i=1}^{N} C_{ij} V_j \tag{4}$$

where C_{ij} are the elements of the mutual capacitance matrix. Note that in Eq. 3 the surface of the element under consideration is assumed to be equipotential and integration is only carried on the outside of the satellite. In reality, components are separated by dielectric materials and electric fields between them exist inside the spacecraft. A more correct expression for the charge q_i would therefore require an integration over the entire close surface enclosing the element. In practice, however, the fields inside the satellite cannot be calculated without detailed knowledge of the satellite design and fabrication. Ideally, our model should be based on measured values of C_{ij}, but such measured values are almost never available. In this model, we therefore use Eq. 3 as an approximation for the elements of the mutual capacitance matrix.

Note that equation 4 can be inverted to yield

$$V_i = \sum_{i=1}^{N} C_{ij}^{-1} q_j \tag{5}$$

with C_{ij}^{-1} being the elements of the inverse of the mutual capacitance matrix.

As particles are pushed in time, a record is kept of the net charge deposited on every satellite component. At the beginning of a simulation, the charge of every element is set to zero. At each timestep, the charge q_i deposited on element i is incremented by the sum of the charges of all the particles that are incident on that element times their statistical weight.

2.3. Volume and surface charge densities

With the simulation domain being discretised in terms of tetrahedral cells, charge density is defined at the cell vertices. Given a distribution of N particles of charges q_i at positions $\vec{r}_i$, the charge density at vertex $\vec{r}_k$ is calculated as follows:

$$\rho_k = \frac{1}{\mathcal{V}_k} \sum_{i=1}^{N} q_i N_i(\vec{r}), \tag{6}$$

where N_k is the linear interpolation function equal to unity at vertex k. For each element t (tetrahedron) having $\vec{r}_k$ as a vertex, N_k is linear in t and it vanishes at the three vertices opposite k, and N_k vanishes everywere else. In Eq. 6, $\mathcal{V}_k$ is a quarter of the sum of the volumes of the elements having k as one of their vertices. This is effectively the volume of the Voronoi cell associated with k.

2.4. Boundary conditions and Poisson's equation

Poisson's equation is solved at each timestep for the given charges q_i collected by the various satellite elements, and for the computed volume charge density $\rho(\vec{r})$. For simplicity the electrostatic potential is assumed to be zero on the outer boundary. In the absence of a background magnetic and electric field and this assumption should be valid provided that the outer boundary be sufficiently far from the spacecraft. The next step is to specify the potential on all satellite components in order for Poisson's equation to be well defined in terms of a Dirichlet boundary value problem. This is inferred from charges q_i collected by the various components and from the volume charge density. When used as input in a test-particle code, the electric fields obtained from the PIC model can be used to calculate particle distribution functions with minimal statistical noise (Marchand, Commun. Comput. Phys. doi: 10.4208/cicp.201009.280110a, pp. 471-483, 2010).