ASSESSMENT OF SPACECRAFT SYSTEMS INTEGRATION USING THE ELECTRIC PROPULSION INTERACTIONS CODE (EPIC)

Ioannis G. Mikellides*, Robert A. Kuharski†, Myron J. Mandell‡, Barbara M. Gardner#
Science Applications International Corporation
San Diego, CA

ABSTRACT

SAIC is currently developing the Electric Propulsion Interactions Code "EPIC", an interactive computer tool that allows the construction of a 3-D spacecraft model, and the assessment of interactions between its subsystems and the plume from an electric thruster. EPIC unites different computer tools to address the complexity associated with the interaction processes. This paper describes the overall architecture and capability of EPIC including the physics and algorithms that comprise its various components. Results from selected modeling efforts of different spacecraft-thruster systems are also presented.

INTRODUCTION

The substantial economical advantages over chemical systems have established clear paths of applicability of electric propulsion for both near and far-earth missions. A critical engineering issue in their integration on-board spacecraft is the potentially unfavorable interaction of their high-energy exhausts with the surrounding subsystems. Such interactions may affect mission lifetime and sometimes even threaten mission success. Their accurate assessment, within the design of particular spacecraft, is a complex task with no comprehensive, stand-alone tools or guidelines for the spacecraft designer in existence.

As part of an effort sponsored by the Space Environment and Effects (SEE) Program at the NASA Marshall Space Flight Center, SAIC is currently developing the Electric Propulsion Interactions Code "EPIC", an interactive computer tool that allows the construction of a 3-D spacecraft model, and the assessment of interactions between its surfaces and the plume from an electric thruster. To carry out the complete interactions assessment process in an expedited manner, EPIC unites different computer tools while allowing internal exchange of data under a common interface. For example, calculation of the plume from an ion or Hall thruster is performed by a 2-D plume code that computes the expansion of the main beam using a fluid algorithm while resonant charge-exchange is determined using a Particle-in-Cell (PIC) method. "Plume maps" generated using different simulation tools, such as Direct Simulation Monte Carlo (DSMC) or Magnetohydrodynamic (MHD) codes may also be easily incorporated into the 3-D tool. The main component of EPIC is a 3-D code that incorporates a variety of physical models for the assessment of interactions with the spacecraft including sputtering, surface heating, and induced torques.

EPIC CONCEPT AND ARCHITECTURE

The objective of EPIC is to design, develop and produce an integrated modeling package for calculating electric propulsion plume interactions with spacecraft systems. The suite of computer tools is designed to allow both expert and novice users to investigate EP plume-S/C interactions.

In compliance with the current program guidelines, the computer tool (built for the Win32 platform) will consist of the following main components:

- An enhanced version of Object Toolkit, the 3-D geometry-definition tool developed for the NASA Charging Analyzer Program (*Nascap-2k*),[1] to facilitate interactive generation of the spacecraft geometry and materials.

[*] Senior Staff Scientist, SAIC, Member AIAA
[†] Senior Staff Scientist, SAIC
[‡] Senior Staff Scientist, SAIC, Senior Member AIAA
[#] Program Manager, SAIC

- A 3-D interactions code built using the Module Integrator and Rule-based Intelligent Analytic Database (MIRIAD) architecture.
- A 2-D plume code that generates the plume map from an electrostatic thruster in the R-Z plane and,
- A fast, 1-D Hall effect thruster (HET) algorithm that provides the conditions near the exit (to guide the input into the 2-D plume code)

The bulk of EPIC's capability is the 3-D plume-surface interactions code. The code is built on the MIRIAD architecture, which has formed the core of several NASA and DoD programs, and has allowed spacecraft designers to quickly and affordably model the interactions between the spacecraft and space environment. MIRIAD's open architecture provides a framework for integrating a variety of physical models and their constituent data into a single executable application. The application then allows the user to define systems of interest and perform parametric studies with relative ease. Ideally, any variable in MIRIAD can be included in a parametric study.

In general, the user will provide the following input to EPIC,

- ➢ Satellite geometry and surface materials
- ➢ Thruster locations and plume parameters
- ➢ Case study parameters (e.g., sputter yield coefficients, orbit, hours of thruster operation etc.)

and will be able to conduct trade studies while generating then following output:

- ➢ Contour plots of the plume map in space (Fig 1, top), and of surface interactions on the 3-D spacecraft, such as surface erosion (Fig 1, bottom) and heating
- ➢ 1-D plots along surfaces (e.g. erosion depth on a solar array as a function of distance from the thruster)
- ➢ Integrated results over duration of mission (e.g. total induced torque in a given direction, total deposition of eroded material at a specific location on the S/C)
- ➢ List of results in text format for post-processing

The main EPIC user interface and the code that integrates the various components are programmed in C# (C-sharp). C# is a new language developed by Microsoft that provides an easy-to-use object orientated way to program the windows user interface, and makes it extremely easy to combine components written in different languages using either the new Common Language Runtime (CLR) or the Component Object Model (COM) that has been the backbone of windows programs for years. We have found that the combination of C# and the CLR foundation classes is a very powerful and easy way to get the desired functionality in the user interface.

The most natural and powerful way to communicate with the MIRIAD data server that coordinates EPIC calculations is through COM. It was therefore particularly important to be able to use COM components in EPIC. Any COM or dot net client can access EPIC's COM data server. For example EXCEL can be used to script calculations and manipulate data produced by EPIC. C# allows easy use of our COM-based 3d viewer (MxOpenGlArena) for viewing the system and displaying surface information and spatial data.

SPACECRAFT DEFINITION

Spacecraft surface definition for EPIC is performed using Object Toolkit. Object Toolkit (OTk) was written as the spacecraft definition module for the *Nascap-2K* plasma interactions code.[1] Written in pure Java, OTk is manifestly cross-platform. It is currently being generalized as a standalone object definition program for all SEE software tools requiring such capability.

OTk builds a spacecraft from its native object components, which include Brick, Panel, Boom, Dish, and Cylinder (Fig 2, top). In addition, OTk can import component parts built either with OTk or with industry standard Patran (interfaces to other standard formats are possible). OTk provides tools to align and position the components, and to "weld" them together for those applications that require surface elements to be compatible. Each surface element has attributes of material (a string) and conductor (an integer). Material properties (currently only properties appropriate to spacecraft charging) are editable within OTk. Materials and conductors are initially specified for the surface of a component, and may then be changed for individual surface elements. Mesh editing tools are provided as well. Fig 3 (right) shows an OTk object constructed to model the MESSENGER[2] spacecraft as part of an effort to study charging of surfaces near Mercury.

A system (e.g. spacecraft) in EPIC's MIRIAD architecture consists of a combination of geometrical primitives or objects, defined by data structures called "entities", which describe each objects surfaces, material properties, etc. The advantage of this scheme is that objects can be individually transformed (i.e., moved, rotated or scaled) to accomplish, for example, rotation of solar arrays during an orbit. OTk will be modified to associate with each surface element an "EntityName" property. Spacecraft designers will be able to assign entity names to individual surfaces or to all the surfaces of a selected component. The user will be able to view the model color-coded by entity name. Additional surface properties can easily be added if needed.

OTk writes its output as an XML (eXtensible Markup Language) file, so that it can easily be processed by any application using standard XML libraries that are ubiquitous among programming languages and platforms. The use of XML makes it very easy to transform OTk's output to a structure more easily processed by the target application. The output specifies the node locations associated with element corners, and the quadrilateral and triangular surface elements that make up the surface. Associated with each surface element is a material name and a conductor number, and a set of properties is associated with each material name.

INTERACTIONS PHYSICS MODULE

The ability to assess a variety of spacecraft interactions with electric propulsion plumes, in three dimensions, is the bulk of EPIC's capability. The interactions models presently allowable by EPIC have been described in greater detail elsewhere,[3] and are summarized here for completeness:

- Ion flux to surfaces. The thruster ion flux at any point on a surface is computed using the incident ion density and velocity according to the angle between the surface normal vector and the ion velocity. Fluxes to points on surfaces account for the interference ("blocking") by other spacecraft surfaces. Specifically, if a straight line between the point in question and the thruster orifice intercepts any other surface the flux is zero.
- Sputtering, surface erosion/deposition. The sputtering at a point on a spacecraft surface caused by ion impact is calculated based on the material sputtering rate, which is a function of the sputter yield. The latter depends on the energy of the impacting ions and the angle between the flux vector and the surface-normal. In EPIC, the present sputter yields are polynomial fits to data and may therefore be modified by the user. Depending upon the duration of thruster operation the total surface erosion and re-deposition of sputtered particles onto other surfaces is determined by computing a net erosion/deposition rate. The net rate is determined by averaging the sputtering rates at each surface to produce a source term at the centroid of that surface. This source term is then used to determine a deposition rate for all other surfaces.
- Surface heating. The incident heat flux at each point on a spacecraft surface due to plume impact is calculated by summing the flux of energy across each surface element. The incident ion energy is assumed to be the sum of only the kinetic and ionization contributions. To account for the material-specific response to the incident energy flux, accommodation coefficients are incorporated in the model (similar to heat transfer coefficients in forced or free convection of fluids over surfaces).
- Induced moments on spacecraft. The model to compute spacecraft torques accounts for the contributions of thrust, produced directly by the propulsion system, and of particle impact produced by the impingement of the thruster plume on surfaces. The latter is computed using two limiting cases: *specular* (elastic) reflection from the surface (colliding particle is reflected with the same speed and incidence angle equals the reflection angle), and *fully diffused* reflection (also known as complete accommodation).[4]

ELECTRIC PROPULSION PLUME PHYSICS MODULE

Thruster plume definition in EPIC is performed using the Plume Wizard. The Plume Wizard will launch the 2-D plume-definition component as shown in Fig 4. The generation of the plume map in EPIC involves seven distinct steps:

- ➢ Thruster Type
- ➢ Engine Specification
- ➢ Grid Generation
- ➢ Neutral Gas
- ➢ Main Beam Expansion
- ➢ Neutral-Ion Interaction
- ➢ Export Plume Map

The 2-D component allows the designer to either import a plume map (produced by other codes) or generate it using the existing 2-D plume code.

__2-D Plume Code__. For low-density exhausts such as those associated with electrostatic thrusters, the 2-D plume maps are generated using a revised, finite element (FE) code (*Gilbert*). The maps consist of the spatial distributions of ion density and velocity. The plume model consists of two main components: a Lagrangian algorithm for determining the expansion of the main ion beam and a Particle-in-Cell solver for computing the dynamics of the charge-exchange plasma.

The main beam is assumed to be a collisionless, singly-ionized, quasi-neutral plasma expanding under the influence of the electric field. The latter is assumed to exist due to the presence of density gradients in the plasma. By comparison to heavy-particle motion, electrons reach dynamic equilibrium at much smaller characteristic times. The electron inertia term may therefore be neglected in the equation of motion. In the absence of electron-ion collisions and magnetic fields, conservation of momentum for the electrons is expressed by:

$$m_e \frac{D\vec{v}_e}{Dt} = e\nabla\phi - \frac{\nabla p}{n} = 0 \tag{1}$$

In equation (1), $\vec{v}_e$, ϕ and p are the electron velocity, electric potential and electron pressure, respectively. The electron mass is m_e, the charge is e and n is the plasma density ($n=n_e=n_i$). Assuming ideal gas behavior and isothermal electrons, integration of eqn (1) leads to the Boltzmann relation, which can be expressed in terms of the electric potential as follows:

$$\phi(n,T_e) = T_e \ln(n/n_\infty) \tag{2}$$

with T_e being the electron temperature and n_∞ is the reference plasma density at zero potential. Equation (2) is sometimes also known as the *barometric potential law*.

Ions are accelerated by the electric field, $\vec{E} = -\nabla\phi$ according to,

$$m_i \frac{D\vec{v}_i}{Dt} = -e\nabla\phi \tag{3}$$

Since the drift velocity of the ions, $\vec{v}_i$, is much greater than their thermal velocity, the high velocity ions are modeled as a fluid. The steady state, conservation equations of mass and momentum are solved in 2-D (R-Z) geometry. The numerical algorithm for computing the expansion of the main ion beam is based on a Lagrangian approach in which discrete mass elements, or otherwise "macroparticles" are released from the thruster exit and are "tracked" using fundamental trajectory kinematics.

Algorithms used for simulating electric propulsion plumes assume that various conditions at the thruster exit are given. For example, the flow of expelled neutrals must be known before performing any calculations to predict ion production rates in the plume due to charge-exchange. In electric propulsion systems such as HETs for example, it is difficult to predict conditions at the exit using empirical measurements exclusively. Furthermore, the effect of thruster geometry and operating conditions on the exhaust, and consequently the surrounding S/C surfaces, is difficult to assess by simple extrapolation of measured quantities associated with a given thruster.

Therefore, specifically for Hall thruster plumes EPIC incorporates a one-dimensional computer model of the acceleration region (Fig 4). The physics include collisions of electrons with neutrals, ions, and walls. The electron energy treatment includes ohmic heating, thermal diffusion and convection, and losses due to ionization, collisions with walls and atomic excitation.[5] In a Hall thruster both the current density and ion velocity vector vary along the exit so an ion flux and density profiles are implemented at the exit boundary.[6]

The neutral gas density has two components in space: un-ionized (beam) particles from the thruster and un-ionized particles from the hollow cathode. The beam of neutrals from the thruster is computed using an annular anode gas flow model with isotropic emission from a ring. The profile of neutrals from the thruster is computed using two disk emissions defined by the solid angles subtended by each disk, and subtracting the smaller from the larger.[4] The flow rate of neutrals from the thruster exit is estimated using the known anode flow rate and propellant utilization. The hollow cathode is offset by a distance r_{HC} from the thruster. Its axial location is assumed to be at z=0. The constant-temperature neutrals are emitted isotropically from the neutralizer, and their speed is estimated based on the same assumptions used for the thruster neutrals.

Fast ions from the main beam undergo charge-exchange (CEX) with neutral particles, resulting in slow-moving ions and fast-moving neutrals,

$${}_{fast}Xe^+ + {}_{slow}Xe \rightarrow {}_{slow}Xe^+ + {}_{fast}Xe \tag{4}$$

In EPIC's 2-D plume module, charge-exchange is computed using a two-dimensional PIC code. The rate of CEX-ion production rate $\dot{n}_{CEX}$ is determined by,

$$\dot{n}_{CEX} = n\vec{v}_i\sigma_{CEX}n_o \qquad (5)$$

In eqn (5) n_o is the particle density of neutrals and σ_{CEX} is the cross section associated with the charge exchange collision. In contrast to the approach for the calculation of the main beam ions, where direct use of equation (2) provides the potential, the PIC algorithm solves the 2-D Poisson's equation on a finite element grid and iterates until steady state CEX densities and potentials are self-consistent. The plasma density in Poisson's equation is the sum of the main-beam and CEX densities. The first (computed by the Lagrangian method), and the prescribed neutral gas profile (above), are used as input for the calculation. Based on the fact that the density of CEX ions is more than two orders of magnitude less than of the main-beam ions, no correction is made to the latter as a result of the first.

EPIC APPLICATIONS

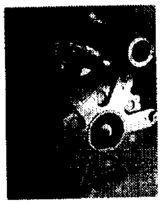
The various physics modules of EPIC have been applied to a variety of problems in the past involving both the study of electric propulsion plumes alone, and their interactions with the spacecraft.

In a collaborative effort with the Jet Propulsion Laboratory to examine, among others, how the Langmuir probe operation on board the DS-1 spacecraft affected ion flux measurements, simulations of the plume from the NSTAR (NASA Solar Electric Power Technology Application Readiness) ion thruster showed good agreement with flight data.[7] Specifically, using a prescribed Gaussian profile for the main beam, the measured charge-exchange plume fluxes for a variety of thruster operating conditions were compared with simulations using the PIC component of the 2-D plume code (Fig 5). It was confirmed that the thruster charge-exchange plasma has much lower density in space than in chamber tests. The good comparisons with the flight data validated the ability of the 2-D code to predict the ion thruster generated plasma environment.

More recently, comparisons of ion flux measurements (courtesy of the NASA Glenn Research Center) of the main beam from an SPT100, acquired onboard two Russian geosynchronous communication satellites (Express A#2 and A#3) yielded good agreement with values computed by the 2-D plume code, as shown in Fig 6. The plume maps were then incorporated in the 3-D spacecraft interactions code to predict, among others, the induced torques on the spacecraft (Fig 7).[4] The effort was part of an ongoing program to develop a Direct-Drive Hall-Effect Thruster System,[8] and produced the first comparisons of Hall thruster plume simulations with flight data, and computed torques with measurements from the onboard Attitude control System (ACS).

Finally, as part of a collaborative effort with AFRL (Edwards AFB) to investigate plume interactions of a low power (200W) Hall thruster with the TechSat21 spacecraft, simulations using the 2-D plume code produced good comparisons with measurements conducted at AFRL[9] and MIT[10] labs. Figure 8 shows comparisons between the EPIC plume model and data for vacuum chamber pressures of 2e-5 Torr (top) and 6e-6 Torr (bottom). The measurements at AFRL were conducted using a double cylindrical Langmuir probe that was placed 60cm downstream of the thruster exit. A hyperbolic tangent function was fitted to the data to provide the electron temperature and number density. The latter was then scaled to 1-m using a ~1/r^2 saling. The current density was computed using the measured number density and a average ion energy E/q of 200 V. A similar setup was used successfully in the past to measure the ion flux downstream of a Hall thruster.[11] At MIT, the data was acquired using a Faraday cup. The overall experimental setup is described in reference 10. For the plume calculations, all quantities implemented as input were deduced from information provided by AFRL and Busek,[12] part of which is provided in Table 1.

Table 1. Specifications used in the simulations of the plasma plume from the 200W Busek Hall Thruster.

	Parameter	Value
	Inner & Outer Radii (cm)	0.6, 1.5
	HC to Centerline (cm)	6
	Anode Flow rate (mg/s)	0.74
	HC Flow Rate (mg/s)	0.08
	Thrust (mN)	11.4
	(Anode) Isp (sec)	1570
	Propellant Utilization	0.9

As already mentioned in a previous section, the model assumes a fixed electron temperature throughout the plume. Measurements in the plume of various, higher-power (0.7-1.6 kW) HETs reveal electron temperatures that range between 6-11eV in the near-field region (a few centimeters downstream of the exit) falling to less than 2-3eV far from the thruster exit.[11,13,14] In the AFRL double Langmuir probe measurements the temperature varied from 0.6 to 1.6 eV, 60 cm from the thruster exit. Equation (2) suggests that the ratio of plasma potential to electron temperature goes to zero far away from the exit, as the plasma density approaches n_∞. Both the simulations and simple scaling (~$1/r^2$) suggest that the plasma density drops by more than two orders of magnitude within a few tens of centimeters away from the thruster exit. Therefore, in view of both the magnitude and (more importantly) the gradients of the plasma potential at distances greater than 20-50 cm, a high-temperature choice (~8eV) introduces a relatively small error on the ion trajectories. In contrast, close to the exit where the electron temperature is indeed high as confirmed by measurements, a low-value choice (~2eV) would have a more erroneous effect on the expanding plasma because the largest electric forces are concentrated close to the thruster exit. Consequently, in view of the dominant role that the near-field forces have on the evolution of the plume the high-temperature choice appears more reasonable. Indeed, a more accurate calculation would incorporate the effects of non-uniform temperature using energy conservation considerations with a proper model of thermal conductivity for the electrons. Nevertheless, for comparison purposes Fig 8 depicts results for Te=8eV and 2eV.

CONCLUSIONS

The near and long-term benefits of electric propulsion are now widely recognized and accepted by both government and private sectors. Solar EP systems have been used successfully in numerous missions, with many more readied for near-term applications. Through the new Nuclear Systems Initiative NASA also focuses in nuclear electric propulsion (NEP) as the "way to defeat distance and time in exploring the outer planets." Integration of EP systems on spacecraft presents however serious barriers to implementation. In the absence of accurate assessments, interactions between the propulsion system and the surrounding spacecraft components may threaten mission success. Quantifying such interactions within the design of specific spacecraft is a complex task with no comprehensive tools for the spacecraft engineer in existence. In response to the growing need by the electric propulsion community for a stand-alone design tool that addresses these issues, SAIC is developing the Electric Propulsion Interactions Code, EPIC. EPIC is a windows-based interactive, computer tool that allows the spacecraft designer to conduct the complete interactions process including geometry definition, EP plume generation, and interactions evaluation in 3-D. Most of the main EPIC user interface is now operational while work continues on the integration and communication with the Object Toolkit and the 2-D plume tool. It is anticipated that a fully functional beta version of the tool will exist by the end of July, 2002.

ACKNOWLEDGMENTS

This work is supported by the Space Environments and Effects (SEE) Program at the NASA Marshall Space Flight Center under contract no. NAS 8-02-028.

REFERENCES

[1] Mandell, M.J., *et al.*, "*Nascap-2k*, A Spacecraft Charging Analyis Code for the 21st Century," AIAA Paper 2001-0957, AIAA Aerospace Sciences Meeting & Exhibit, 39th, Reno, NV, Jan. 2001.

[2] Katz, I., Mandell, M.J., Gardner, B.M., and Mauer, R., "MESSENGER Spacecraft Charging Analysis," Presented a the 7th Spacecraft Charging Conference in Noordvijk, The Netherlands, 2001.

[3] Mikellides, I.G., *et al.*, "A Hall-Effect Thruster Plume and Spacecraft Interactions Modeling Package," IEPC Paper 01-251, Pasadena, California, October 2001.

[4] Mikellides, I.G., *et al.*, "Plume Modeling of Stationary Plasma Thrusters and Interactions with the Express-A Spacecraft," *Journal of Spacecraft and Rockets*, Log. No. A11411, Nov. 13 2001, IEPC Preprint 01-251.

[5] Mikellides, I.G., Katz, I., Mandell, M.J., "Modeling the Exit Plane of a Hall Effect Thruster," AIAA Paper 01-3505, Salt Lake City, UT, July 2001.

[6] Katz, I. *et al.*, "A Hall Effect Thruster Plume Model Including Large-Angle Elastic Scattering," AIAA Paper 01-3355, Salt Lake City, UT, July 2001.

[7] Davis, V.A., *et al.*, "Ion Engine Generated Charge Exchange Environment: Comparison Between NSTAR Flight Data and Numerical Simulations," AIAA Paper 00-3529, July 2000.

[8] Jongeward, G.A., *et al.*, T. "High Voltage Solar Arrays for a Direct Drive Hall-Effect Propulsion System," IEPC Paper 01-327, Pasadena, California, October 2001.

[9] Hargus, W., and Reed, G., "The Air Force Clustered Hall Thruster Program," AIAA Paper 2002-3678, July 2002.

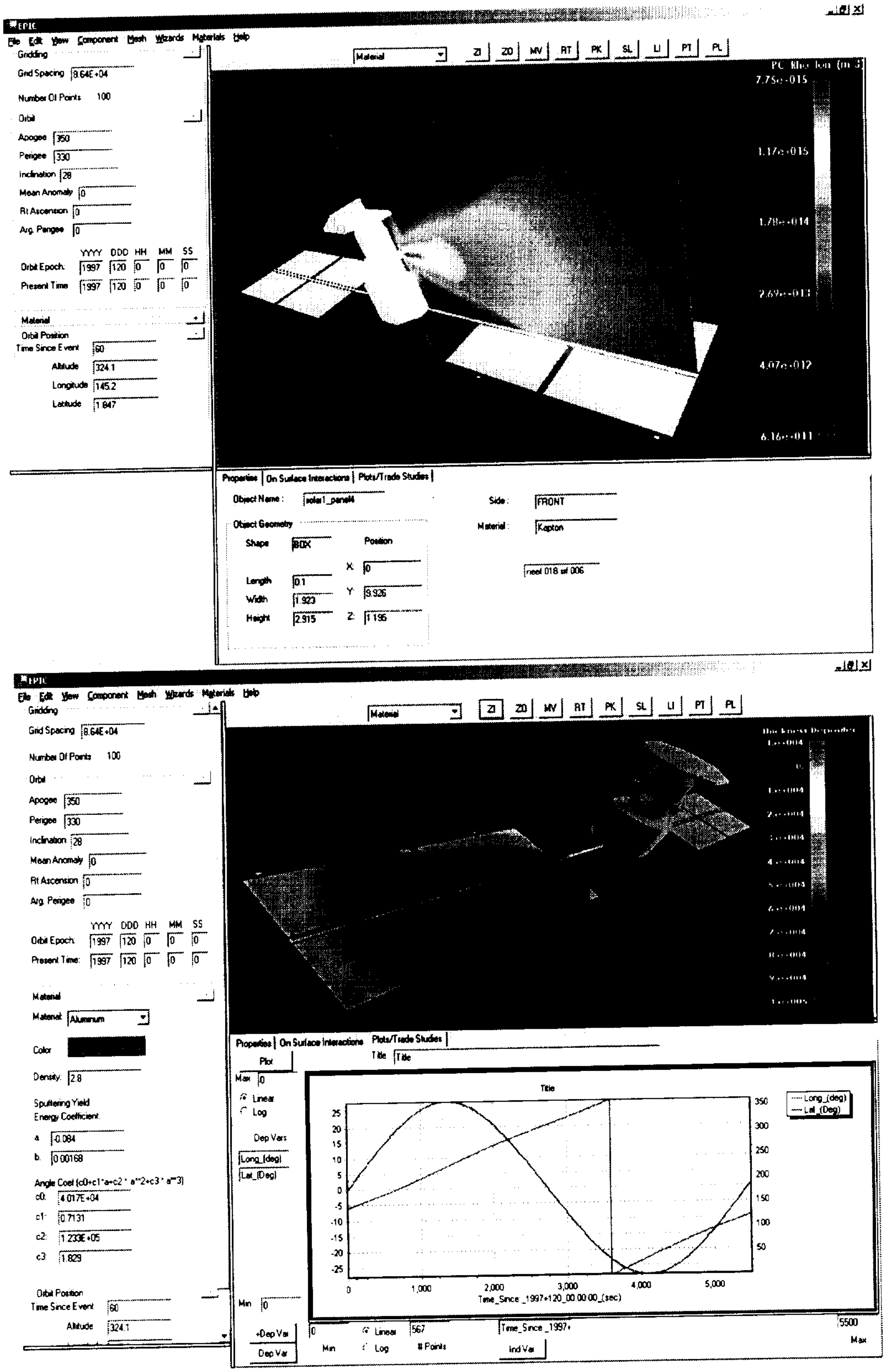
[10] Pacros, A., "Instruments Design and Testing for a Hall Thruster Plume Experiment on the Space Shuttle," M.S. Thesis, MIT, Massachusetts, June 2002.

[11] Haas, J.M., and Gallimore, A.D., "An Investigation of Internal Ion Number Density and Electron Temperature Profiles in a Laboratory-Model Hall Thruster," AIAA Paper 00-3422, July 2000.

[12] Hruby, V., Pote, B., Freeman, C. and Connolly, W., "Low Power, Hall Thruster Propulsion System," IEPC Paper 99-092, Kitakyushu, Japan, Oct. 1999.

[13] Kim, S. W., Foster, J. E., and Gallimore, A. D., "Very-Near-Field Plume Study of a 1.35 kW SPT-100," AIAA Paper 96-2972, July 1996.

[14] Fife, J.M., "Hybrid PIC-Modeling and Electrostatic Probe Survey of Hall Thrusters," Ph.D. Dissertation, MIT, Massachusetts, Sept. 1998.



Gridding

Grid Spacing 8.64E+04

Number Of Points 100

Orbit

Apogee 350

Perigee 330

Inclination 28

Mean Anomaly 0

Rt Ascension 0

Arg. Perigee 0

	YYYY	DDD	HH	MM	SS
Orbit Epoch:	1997	120	0	0	0
Present Time	1997	120	0	0	0

Material

Orbit Position

Time Since Event 60

Altitude 324.1

Longitude 145.2

Latitude 1.847

Object Name : solar1_panel4

Side : FRONT

Material : Kapton

Object Geometry

Shape BOX

Position

X: 0

Length 0.1

Y: 9.926

Width 1.923

Height 2.915

Z: 1.195

neel 018 sf 006

Gridding

Grid Spacing 8.64E+04

Number Of Points 100

Orbit

Apogee 350

Perigee 330

Inclination 28

Mean Anomaly 0

Rt Ascension 0

Arg. Perigee 0

	YYYY	DDD	HH	MM	SS
Orbit Epoch:	1997	120	0	0	0
Present Time:	1997	120	0	0	0

Material

Material: Aluminum

Color

Density: 2.8

Sputtering Yield
Energy Coefficient:

a. -0.084

b. 0.00168

Angle Coef (c0+c1*a+c2 * a**2+c3 * a**3)

c0: 4.017E+04

c1: 0.7131

c2: 1.233E+05

c3: 1.829

Orbit Position

Time Since Event 60

Altitude 324.1

Title Title

Max 0

● Linear ○ Log

Dep Vars: Long_(deg), Lat_(Deg)

Min 0

Min 0 ● Linear ○ Log # Points 567 Time_Since_1997+ Max 5500

Fig 1. Screen shots of the Electric Propulsion Interactions Code (EPIC).

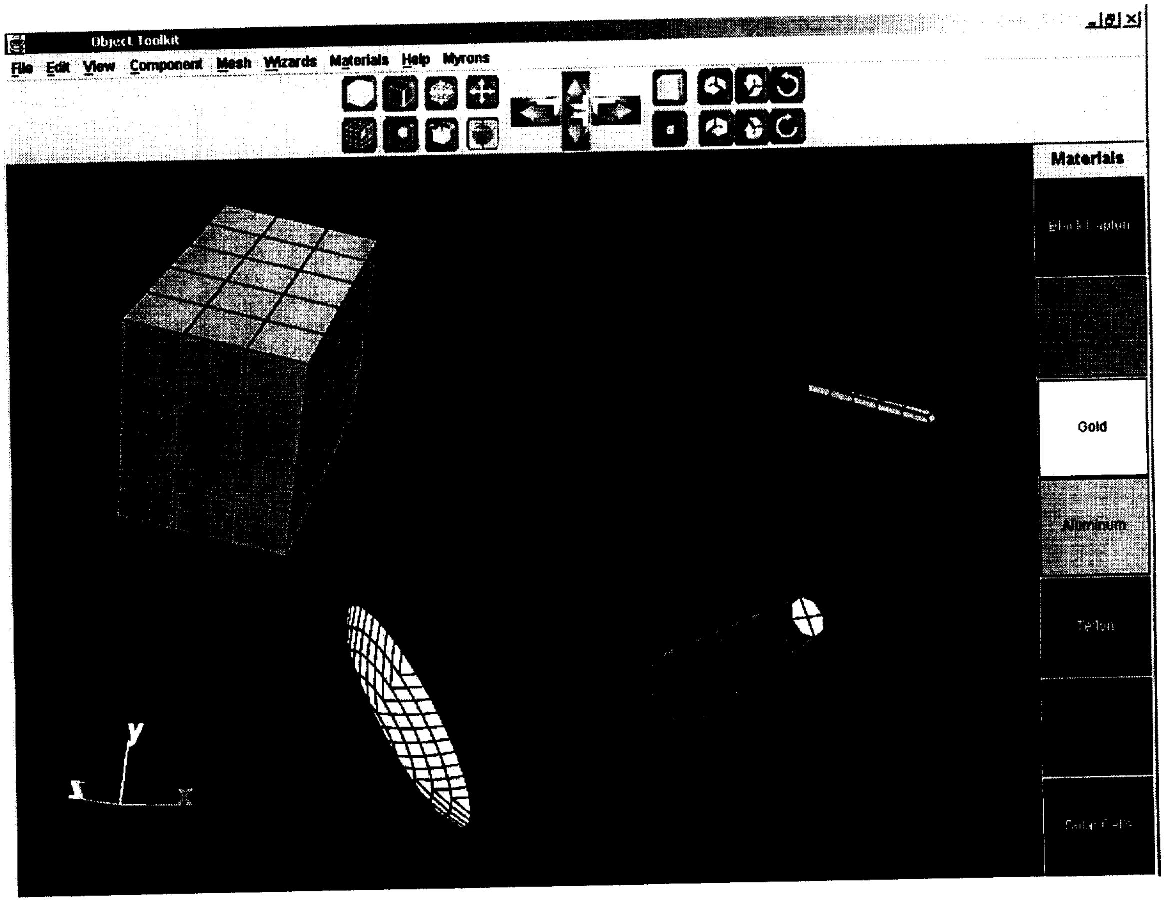


Fig 2. Basic object components in the Object Toolkit.

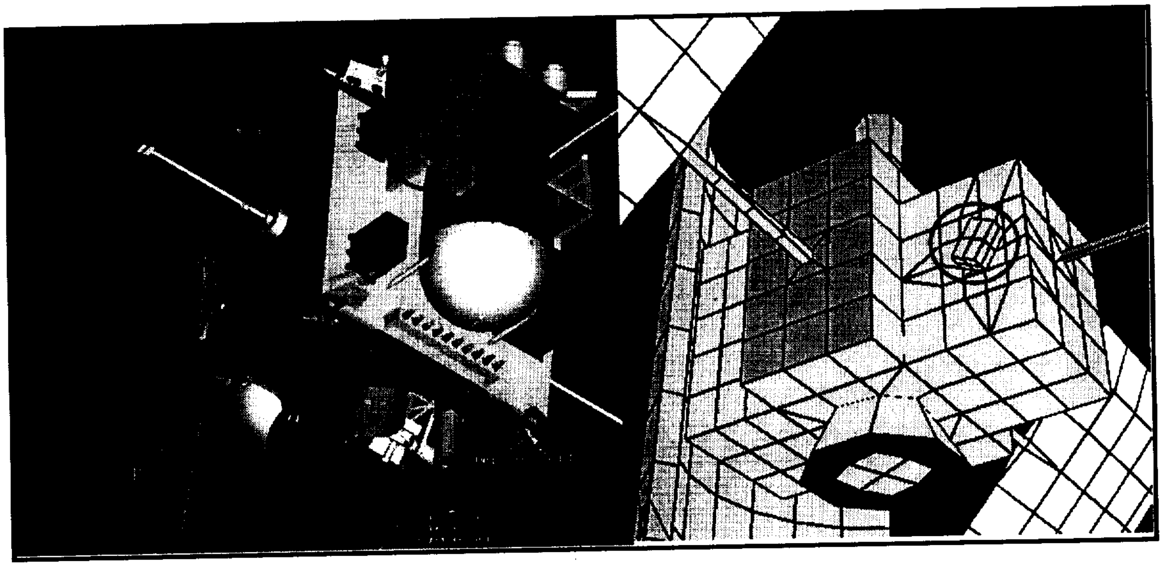
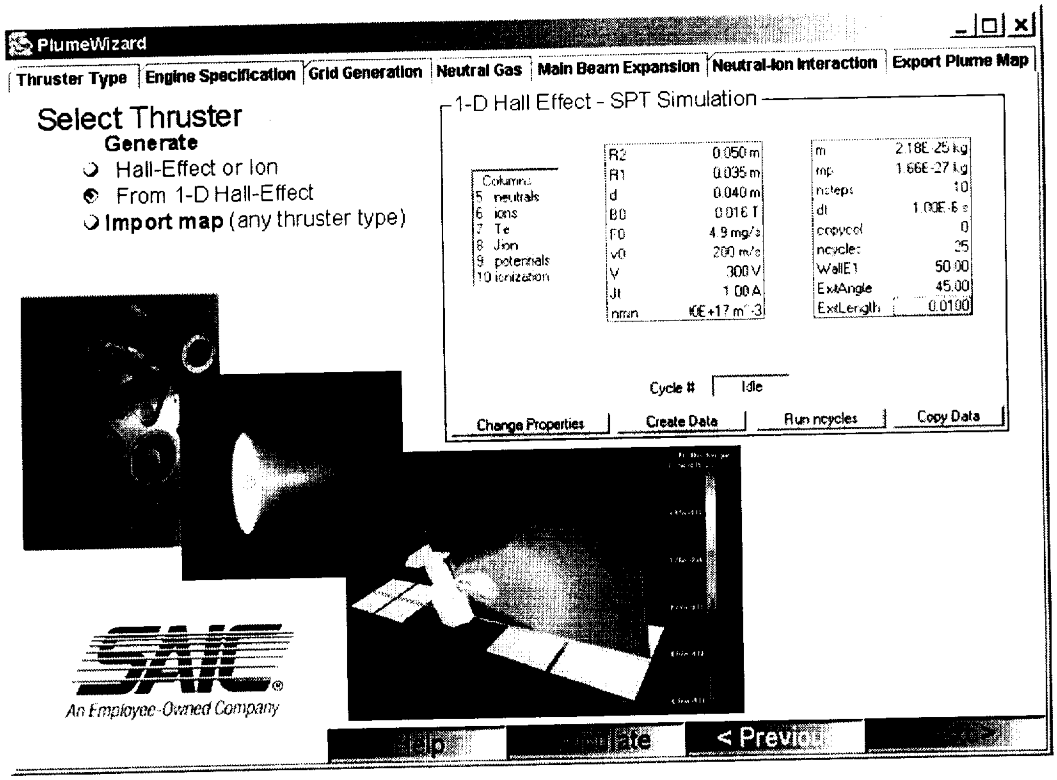


Fig 3. Left: CAD model of the MESSENGER bus. Right: Object Toolkit model. Circled is the FIPS ion detector.



Select Thruster

Generate

○ Hall-Effect or Ion

● From 1-D Hall-Effect

○ **Import map** (any thruster type)

1-D Hall Effect - SPT Simulation

Columns
5 neutrals
6 ions
7 Te
8 Jion
9 potentials
10 ionization

R2	0.050 m	m	2.18E-25 kg
R1	0.035 m	mp	1.66E-27 kg
d	0.040 m	nsteps	10
B0	0.016 T	dt	1.00E-6 s
F0	4.9 mg/s	copycol	0
v0	200 m/s	ncycles	25
V	300 V	WallE1	50.00
Jt	1.00 A	ExtAngle	45.00
nmin	1E+17 m^-3	ExtLength	0.0100

Cycle # Idle

Fig 4. EPIC's plume definition component showing option to execute 1-D Hall thruster simulation.

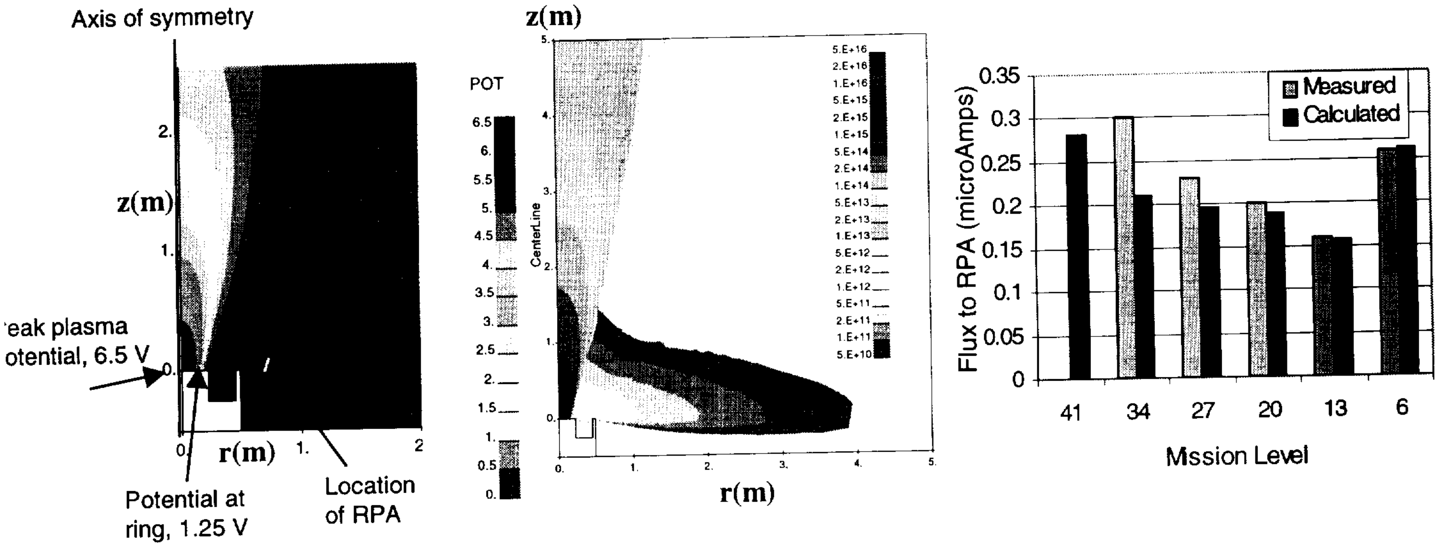


Fig 5. Simulations of DS-1 ion thruster plume showing plasma potential distribution (left), ion density in m^{-3} (middle) and comparisons with measured ion current by onboard RPA (right).

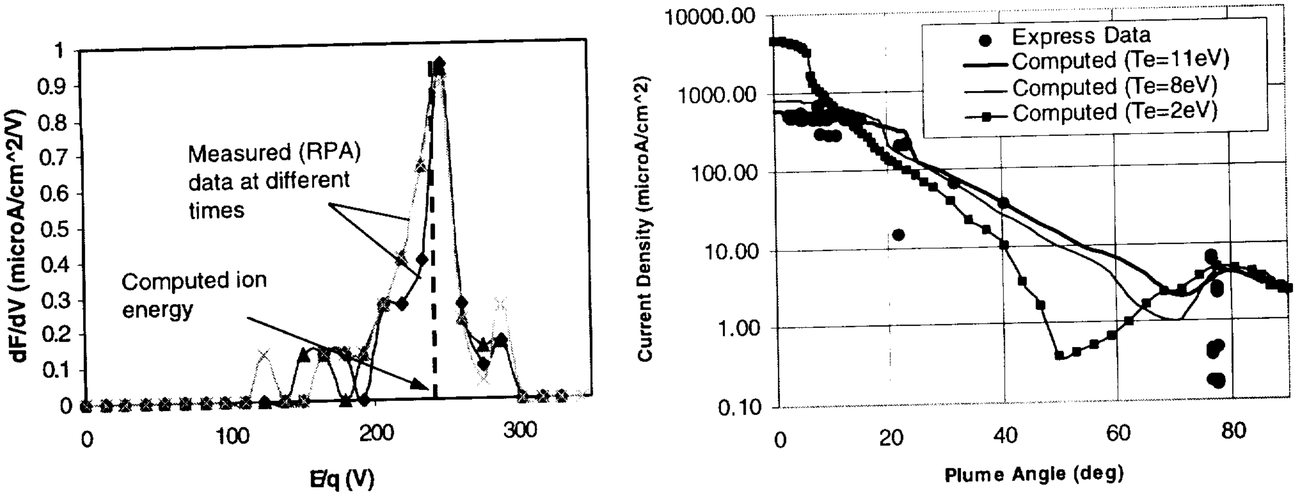


Fig 6. Comparisons between flight measurements and computed main beam ion energy (left), and current density 1m from an SPT-100 onboard the EXPRESS-A spacecraft.

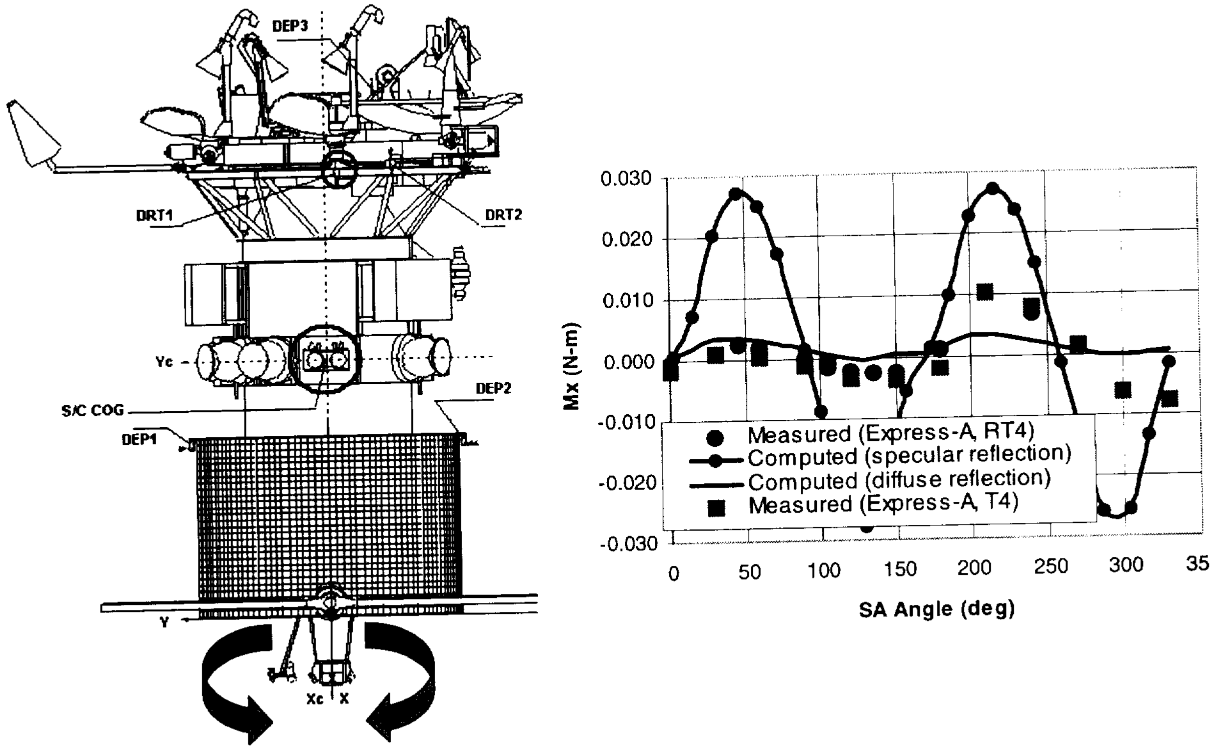


Fig 7. Comparison between computed and measured torque around the x-axis (right) induced by firings of the RT4 SPT100 on board the EXRESS-A spacecraft (left). The measurements were taken during a rotation of the solar array.

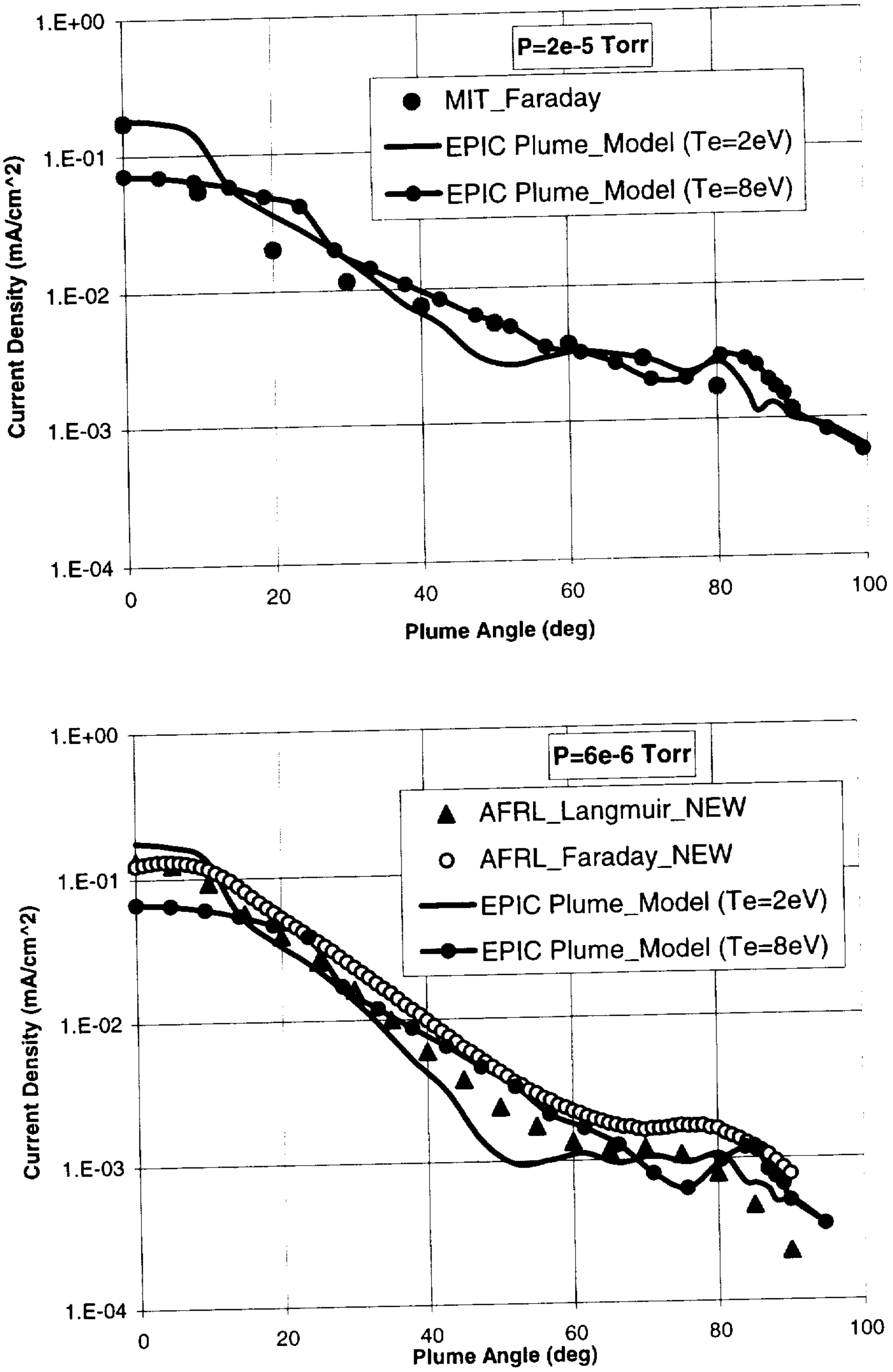


Fig 8. Comparisons between current density measurements and results from EPIC's 2-D plume model, 1-m downstream of the 200W Busek Hall thruster. Top: Vacuum chamber pressure=2e-5 Torr. Bottom: Vacuum chamber pressure=6e-6 Torr.