A fast estimate tool for redeposition caused by sputtering during terrestrial testing

IEPC-2017-300

Presented at the 35^{th} International Electric Propulsion Conference,
Georgia Institute of Technology, Atlanta, Georgia, USA
October 8–12, 2017

Gunnar Bandelow,* Ralf Schneider†
Ernst-Moritz-Arndt-University, Greifswald, Germany
Julia Duras,‡ Norbert Koch§
Nuremberg Institute of Technology, Nuremberg, Germany
Stefan Kemnitz¶
University of Rostock, Rostock, Germany
and
Oleksandr Kalentev‖
Biomedizinische NMR Forschungs GmbH, Göttingen, Germany

Thrusters have to be qualified for satellite missions in large vacuum vessels to simulate the space environment. Results of these experiments are sensitive to possible modification of the beam properties due to the interaction of the energetic ions with the vessel walls. In the past a Monte Carlo code for simulating the interaction of ion thruster beams with vessel walls was developed to study these effects by Duras et al.[3]. The "SputterBackflow"-code (SPUBA) is now open-source and further polished to enable a user-friendly input of different geometries of the testchamber and thrusters, as well as ion distributions. Back-fluxes of different ion thrusters and test-setups are calculated to demonstrate the versatility of the code and its input structure.

I. Introduction

For satellite missions, ion thrusters have to be qualified in large vacuum vessels to simulate space environment. One caveat of results from these experiments is the possible modification of operational and performance characteristics of the thruster due to the interaction of energetic ions with the vessel walls. Impinging ions can produce sputtered impurities, which can stream back into the acceleration channel of the thruster and produce co-deposited layers. Over the long operation time of thousands of hours, such layers can modify the optimized geometry and therefore the thruster characteristics in terms of e.g. ion beam properties.

Beside the vessel wall material, the back-flux distribution towards the thruster exit is determined by the vessel geometry and the test setup. Particles, accelerated by the thruster, impinge on the vessel walls and sputter its surface. Due to micro roughness of the surface, the distribution function of the sputtered particles follows a cosine function[1,3]. Its orientation is mainly vertical to the surface, but nearly independent of the incident angle of the accelerated particles. Here, the vessel, including additional experimental setups as well as position and geometry of the thruster, is important. An example setup of a thruster cluster assembled in a vessel is shown in Figure 1. A typical strategy to overcome large back-fluxes is the implementation of

*PhD student, Insitute of Physics, gunnar.bandelow@uni-greifswald.de
†Full Professor, Institute of Physics, schneider@uni-greifswald.de
‡PhD student, Department of Applied Mathematics, Physics and Humanities, julia.duras@uni-greifswald.de
§Full Professor, Department of Applied Mathematics, Physics and Humanities, Norbert.Koch@th-nuernberg.de
¶PhD student, Institute of Computer Science, stefan.kemnitz@uni-greifswald.de
‖Researcher, oleksandr.kalentev@mpibpc.mpg.de

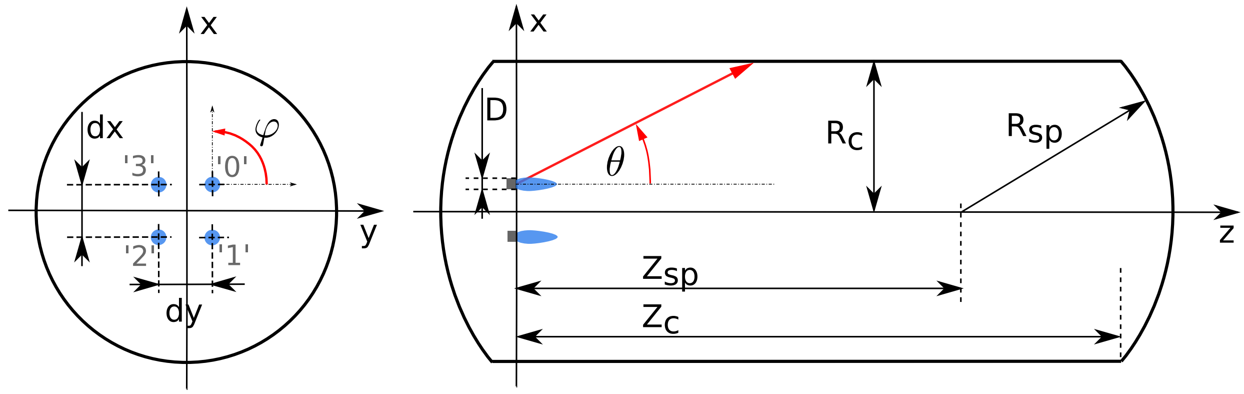

Figure 1. Sketch of a thruster cluster assembled in a vessel within a end-to-end test setup. 0 to 3 indicate the different thrusters.

tilted baffles away from the thruster. The cosine distribution at the surface results in a reduced amount of back-scattered particles towards the thruster channel[3].

In order to optimize the experimental setup a fast estimate of the expected back-flux is useful. The "SputterBackflow"-code (SPUBA) incorporates json-files as input and enables the user to estimate the back-flux for different experimental setups within minutes. The code can run on any desktop machine with minimal dependencies (CMake, Git, any C++ compiler).

II. Method

A simulation to describe erosion and redeposition processes can use different approximation levels of the underlying physics. The most crude approximation is given by a Monte-Carlo (MC) procedure simulating erosion and redeposition based on sputter yield tables. These yields are calculated from a binary-collision cascade (BCC) or a molecular dynamics (MD) model based on the impinging plasma fluxes. Such a model is particularly useful for designing a test setup due to its simplicity and flexibility in allowing to estimate the lifetime of ion thrusters within minutes. The "SputterBackflow"-code (SPUBA) will be presented in this work. It uses the primary distribution of emitted ions with respect to energies, angles and species by each of the implemented sources. Pseudo-particle distributions are sampled with Monte-Carlo algorithms. These pseudo-particles are followed interacting with the experimental environment, e.g. vessel walls, and can create sputtered particles, based on sputter rates. The back-flux of these eroded particles from the vessel walls towards the ion thruster acceleration channel is calculated. A more in-depth description can be found in Duras et al. [3].

III. Simulation Setup

In Figure 2 one can see the whole simulation cycle. Light blue marks the input files, that have to be adapted by the user. Starting point of the simulation is the creation of the pseudo-particles which interact with the chamber geometry and create sputtered particles. These are collected at the given thruster geometry. At the very end output is created for every single thruster.

For a simulation setup only three files are needed. The first file is the "input.json". Listing 1 shows an example for the structure of the input file. It contains the information about feeding gas, wall material and sputter yields. Any combination of sputter yields can be written into the file, within the simulation the parameters of "thruster_feedgas" and "wall_material" have to match a combination in the "yields" structure.

The file "source.json" has to contain at least one source, more are possible. Listing 2 shows an example

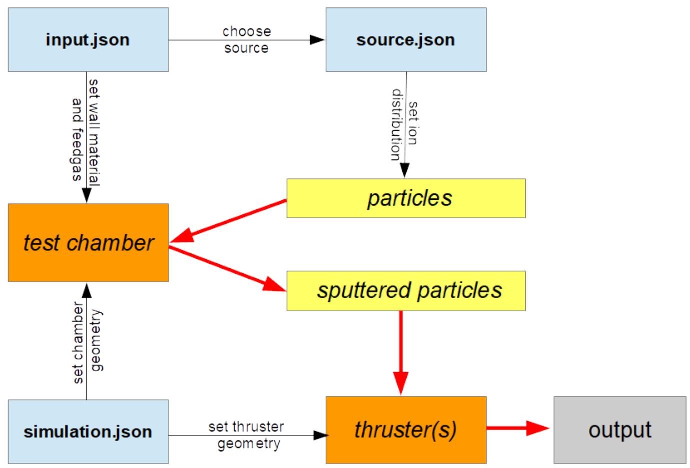

Figure 2. Flow of a simulation with three json-files, which define the whole process.

```
{
  "job_id": 42,
  "number_of_particles": 100000,
  "thruster_feedgas": "Xe",
  "wall_material": "Al",
  "baffles": true,
  "source_type": "thruster",
  "comment": "random string as comment",
  "yields": [
    { "wall": "Al",
      "feedgas": "Xe",
      "energy": [ "..." ],
      "yield":  [ "..." ]
    },
    { "wall": "C",
      "feedgas": "Xe",
      "energy": [ "..." ],
      "yield": [ "..." ]
    }
  ]
}
```

Listing 1: Example for input json-file

for the structure of the source file. Each source has to have a type and the number of angular bins plus the angular range of each bin. Furthermore, there have to be current density, mean energy and the fraction of double positively charged ions for each bin. To create a beam in phi-direction, an additional parameter "constant_phi" has to be set. If this parameter exists within a source, every particle will use the value of "constant_phi" instead of a 2π-sampling. The source used in the simulation is chosen via "input.json" and its position is defined due to the position of the first thruster in "simulation.json".

Aforesaid file "simulation.json" contains all information about two main objects called "TESTCHAMBER" and "THRUSTERS". Each of these are composed of a "name" and a "geometry" structure. The "geometry" structure itself can be assembled from several simple geometric objects (CIRCLE, TUBE, CYLINDER, ...), which should form a closed volume.

```json
{
  "sources": [
    {
      "type": "beam",
      "number_of_angle_bins": 19,
      "delta_angle_theta": 5,
      "current_density": [ "..." ],
      "energy": [ "..." ],
      "fraction": [ "..." ],
      "constant_phi": 30
    },
    {
      "type": "thruster",
      "number_of_angle_bins": 45,
      "delta_angle_theta": 2,
      "current_density": [ "..." ],
      "energy": [ "..." ],
      "fraction": [        "..." ]
    }
  ]
}
```

Listing 2: Example for source json-file

```json
{
  "simulation": {
    "name": "2 thruster test",
    "TESTCHAMBER": {
      "geometry": [ "..." ]
    },
    "THRUSTERS" : [
      {
        "name" : "thruster_01",
        "geometry": [ "..." ]
      },
      {
        "name" : "thruster_02",
        "geometry": [ "..." ]
      }
    ]
  }
}
```

Listing 3: Example for simulation json-file

IV. Results

In the following the results of an example run will be shown to demonstrate the SPUBA package. The shape of the modelled test chamber resembles a cylinder ($Z_c = 1.6\,\mathrm{m}$, $R_c = 1.2\,\mathrm{m}$) with a spherical cap at the end and has a radius of $R_{sp} = 1.825\,\mathrm{m}$. A source/thruster is located at position $(0.3\,m, 0.3\,m, 0)$.

The ion current density and energy distribution of this source are similar to those published for SPT-100[2] and the fraction of double positively charged ions is arbitrary created. Figure 3 is the direct rendering of the chosen source.

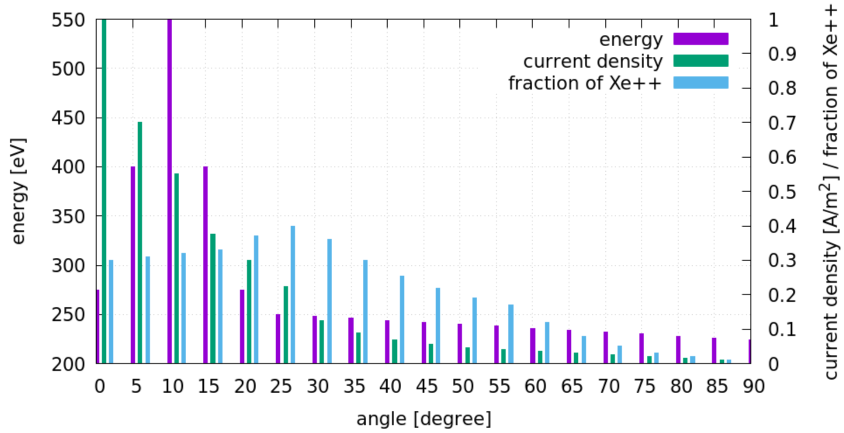

Figure 3. Example of an ion distribution input into the simulation. The violet bars are the different energies of the angle bins in electronvolts (left axis), while the current density in ampere per square meter (green) and the fraction of double positively charged ions (light blue) are assigned to the right axis.

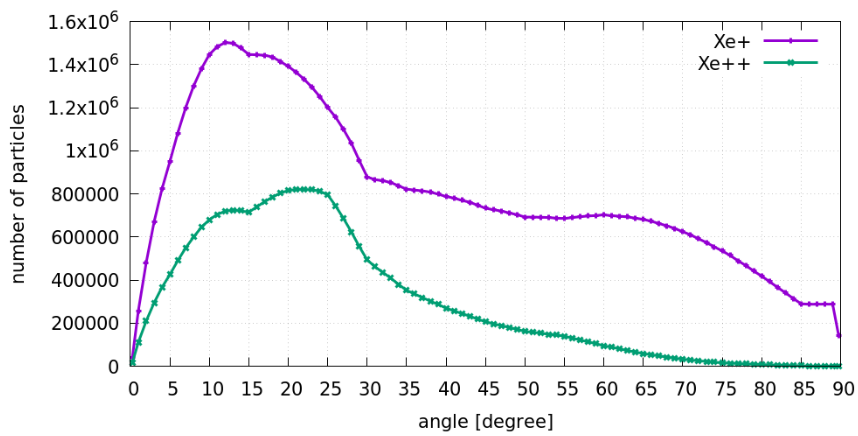

Figure 4. Distribution of simulated pseudo particles based on the source input (s.Fig. 3)

Figure 4 shows the pseudo-particles distribution, that is created via an Monte Carlo algorithm. Each pseudo-particle can create sputtered particles based on the given yields.

The Sputtered particle flux that enters the thruster (at $z = 0.09\,\mathrm{m}$) and hits the channel wall is depicted in Figure 5. Due to the position of the thruster in the vessel mainly a half-circle of angles is exposed to the redeposition flux. The flux itself decreases with the length of the thruster due to the thruster exit acting as a filter (shadowing effect).

The bottom of the thruster (Fig. 6 connects to the channel at $z = 0$, thus the redeposition flux is dominant in the angle range between 120 and 330, as well. Such data, as shwon in Figure 5 and 6, is available for every other thruster present at start of the simulation in the simulation json-file .

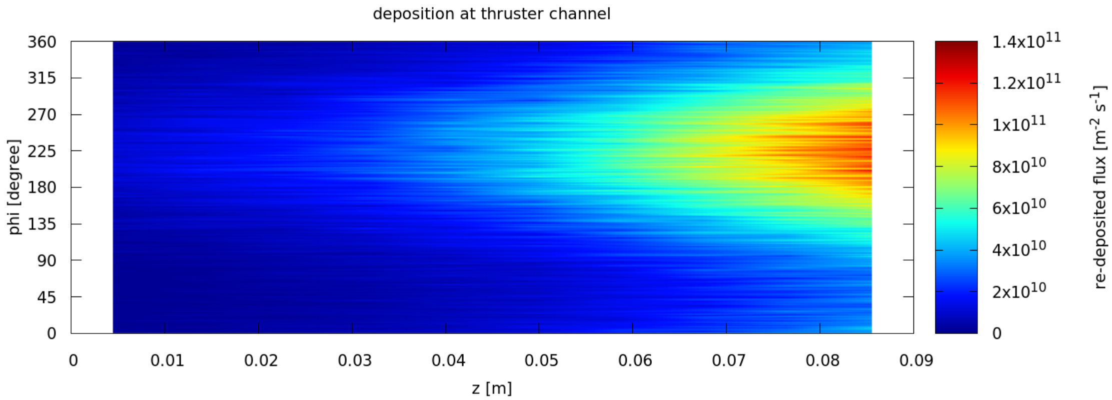

Figure 5. Flux of re-deposited particles at the thruster channel wall.

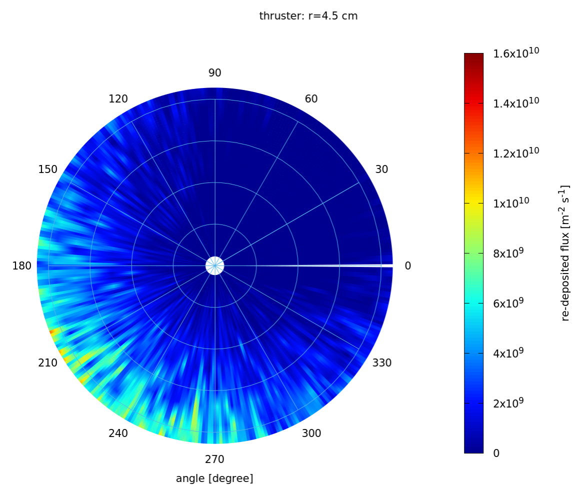

Figure 6. Flux of re-deposited particles at the thruster bottom (anode).

V. Conclusions

A fast estimate tool was developed to predict redeposition caused by sputtering through ions of thrusters in a testchamber. The setup of the simulation is user-friendly done via json-files and the code package itself is available at https://github.com/compusHGW/spuba.git. Results for different geometries and thruster configurations can be achieved on a short timescale.

VI. Acknowledgement

This work was supported by the German Space Agency DLR through projects 50RS1510 and 50RS1508. We also gratefully acknowledge the support of the Bavarian State Ministry of Education and Cultural Affairs, Science and the Arts in frame of the project "ITSim – Skalierung von Ionenantrieben mittels numerischer Simulation".

References

[1] Eckstein, W., "Computer Simulation of Ion-Solid Interaction", *10th vol. of Springer Series in Material Science*, Springer-Verlag, New York, 1991.

[2] King, L. “Transport-property and mass spectral measurements in the plasma exhaust plume of a Hall-effect space propulsion system,” Ph.D. Dissertation, University of Michigan, 1998.

[3] Duras, J., Kalentev, O., Schneider, R., Matyash, K., Lskow, K., Geiser, J. ”Monte-Carlo re-deposition model during terrestrial measurement of ion thrusters”, *Acta Polytechnica Journal of advanced engineering.* Vol. 55, No. 1, 2015, pp. 7-13.