Ion Beam Simulator – IBSimu

Workshop at ICIS2025

Taneli Kalvas
taneli.kalvas@jyu.fi

University of Jyväskylä, Department of Physics

13 September 2025

ICIS2025, Oxford, UK

The contents

The point is to make the learning curve easier for beginners!

- Ion source plasma extraction – What is IBSimu and why?
- Methods behind it all.
- Installation.
- Example use cases, discussion, demonstrations.
- Similarities and differences to other beam-related codes.
- Questions from users.

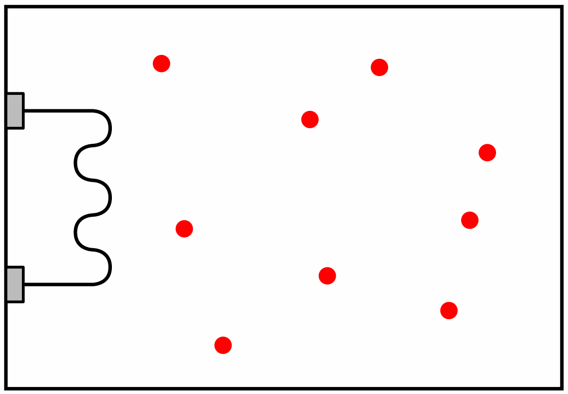
The schedule

Saturday 13th September 2025
9:00 -10:40 IBSimu workshop
10:40-11:10 Coffee&Tea
11:10-12:00 IBSimu workshop
12:00-13:00 Lunch
13:00-15:00 IBSimu workshop

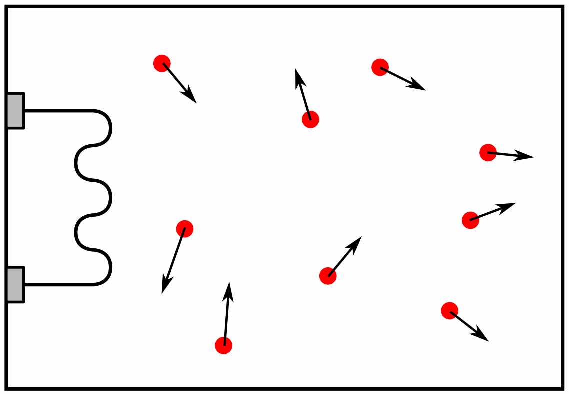
Ion source plasma extraction

What is IBSimu and why?

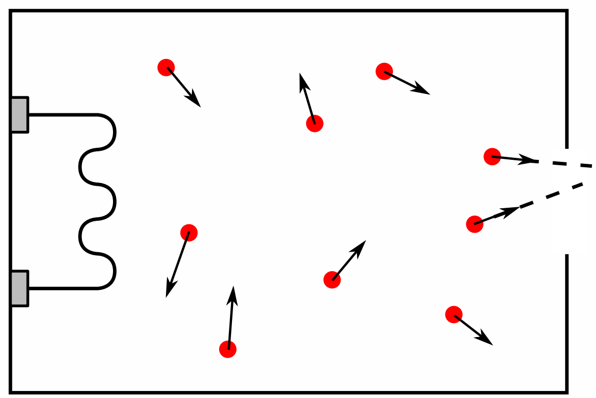
Basic beam extraction



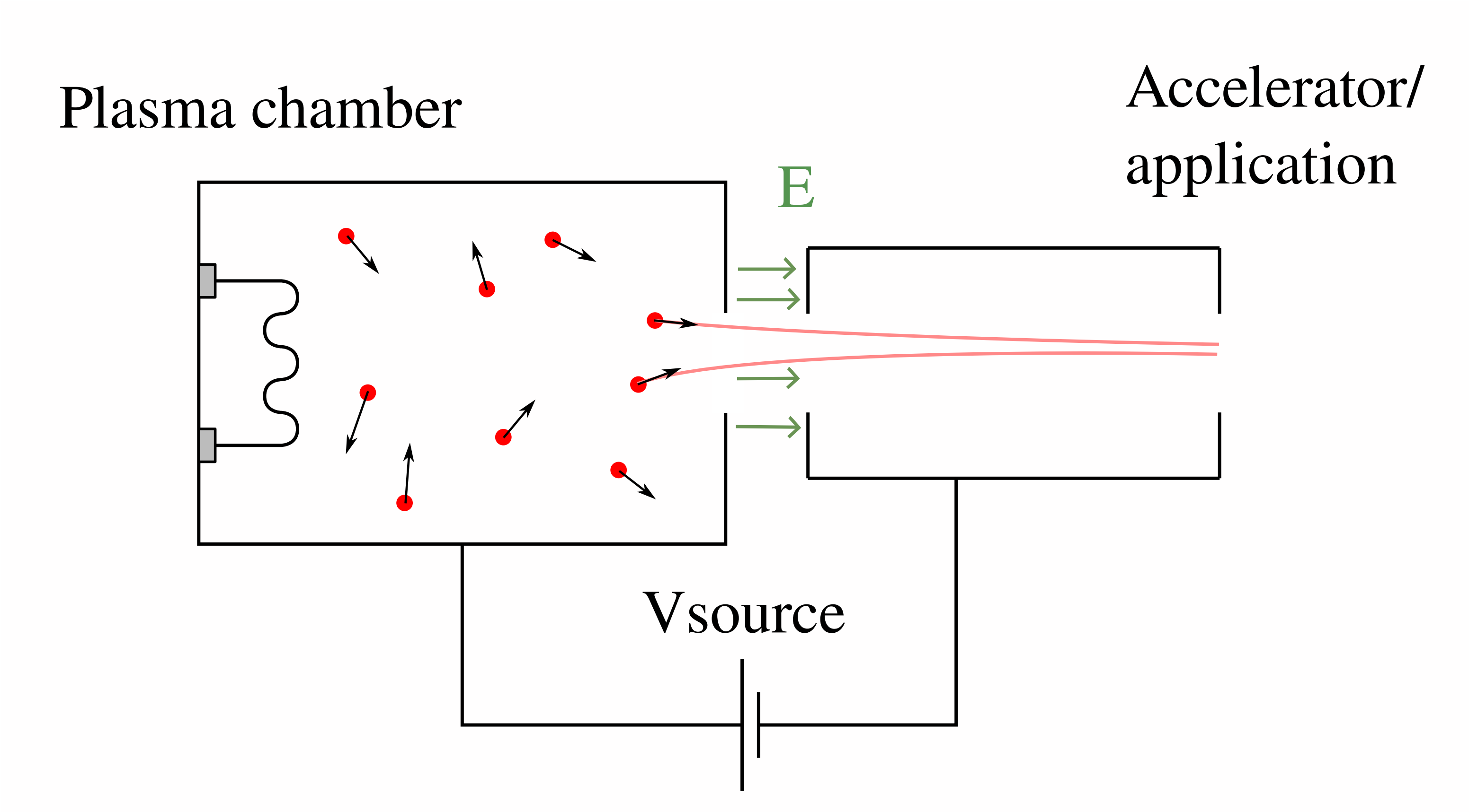
Basic beam extraction



Basic beam extraction



Basic beam extraction



Extraction complications

- Application requirements
  - Beam intensity, quality, species
- Plasma-beam transition physics
  - Plasma parameters: density, potential, temperature, ...
  - Beam intensity and geometry
- Electric and magnetic field
- Beam space charge
- Practical engineering constraints
  - Cooling, high voltage, aligning
  - Space for pumping, diagnostics, feedthroughs
  - Materials, cost

Extraction complications

- Application requirements
  - Beam intensity, quality
- Plasma-beam transition physics
  - Plasma parameters: density, potential, temperature, ...
  - Beam intensity and geometry
- Electric and magnetic field
- Beam space charge
- Practical engineering constraints
  - Cooling, high voltage, aligning
  - Space for pumping, diagnostics, feedthroughs
  - Materials, cost

Extraction complications

- Application requirements
  - Beam intensity, quality
- Plasma-beam transition physics
  - Plasma parameters: density, potential, temperature, . . .
  - Beam intensity and geometry
- Electric and magnetic field
- Beam space charge
- Practical engineering constraints
  - Cooling, high voltage, aligning
  - Space for pumping, diagnostics, feedthroughs
  - Materials, cost

Ion Beam Simulator

IBSimu is an ion optical code package made especially for the needs of ion source extraction design.

The code models:

- Systems of electrostatic and magnetic lenses
- High space charge beams (low energy)
- Positive and negative multispecies 3D plasma extraction

The code is made as a C++ library and is released freely under GNU Public Licence*

- Highly versatile and customizable
- Can be used for batch processing and automatic tuning of parameters

*) `http://ibsimu.sourceforge.net/`

Ion Beam Simulator

- T. Kalvas, et. al., ”IBSIMU: A three-dimensional simulation software for charged particle optics”, Rev. Sci. Instrum. 81, 02B703, (2010).
- T. Kalvas, “Development and use of computational tools for modelling negative hydrogen ion source extraction systems”, Department of Physics, University of Jyväskylä, Research Report No. 10/2013.

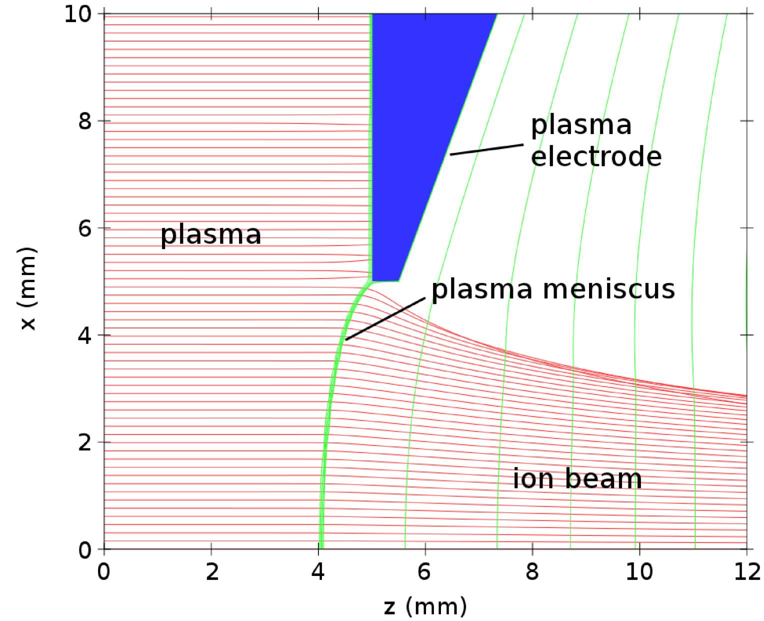
Used for

- CERN’s Linac4 RF H^- ion source
- Oak Ridge SNS RF H^- ion source
- ISIS RF H^- ion source @ RAL
- ITER H^-/D^- injectors
- HIISI 18 GHz RT-ECR ion source @ JYFL
- SMIS 37 GHz gasdynamic ion source @ IAP-RAS
- . . . and many other projects

Plasma-beam interface

Ions are extracted from a plasma ion source

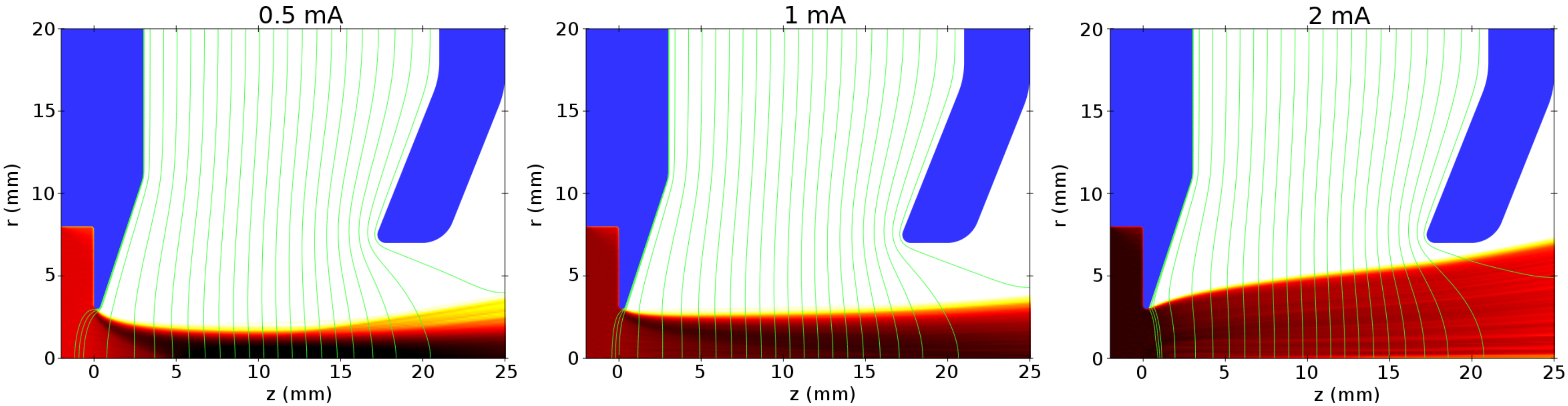
1. Full space charge compensation ($\rho_- = \rho_+$) in the plasma.
2. No compensation in extracted beam (single polarity).



The boundary is often thought as a sharp surface known as the *plasma meniscus* dividing the two regions.

- Works as a thought model.
- In reality compensation drops going from plasma to beam region in a transition layer with thickness $\sim \lambda_D$ known as the *plasma sheath*.
- E-field in extraction rises smoothly from zero.

Plasma sheath shape



- Plasma sheath is dependent on plasma parameters, including especially the beam current, and extraction field.
- Beam focusing is affected by sheath shape.

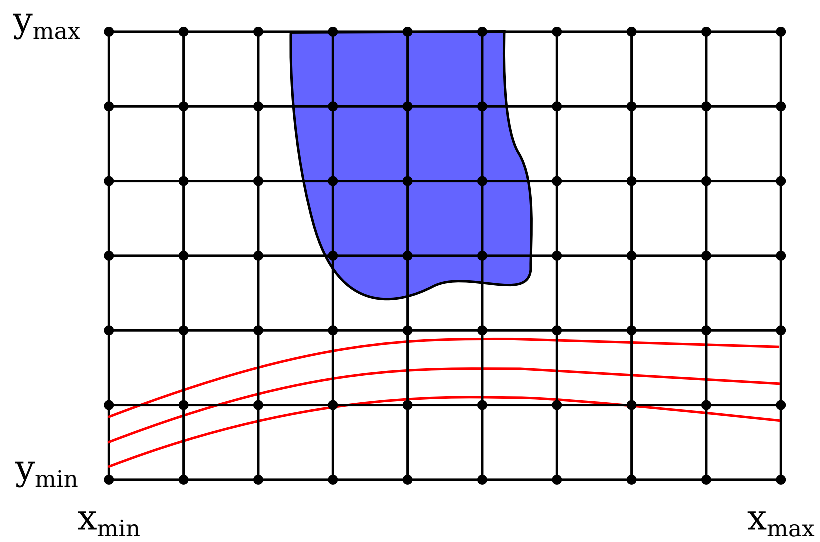
Solution of potential

Solution of potential from Poisson equation

$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0} \qquad (1)$$

and electric field

$$\vec{E} = -\nabla \phi. \qquad (2)$$



With plasma charge

$$\rho = \rho_{\text{beam}} + \rho_{\text{plasma}}, \qquad (3)$$

where

$$\rho_{\text{plasma}} = \rho_{\text{plasma}}(\phi, x). \qquad (4)$$

Particle trajectories

With known fields, the trajectories result from Lorentz force

$$\vec{F} = q\big(\vec{E} + \vec{v} \times \vec{B}\big). \tag{5}$$

The (classical) equations of motion

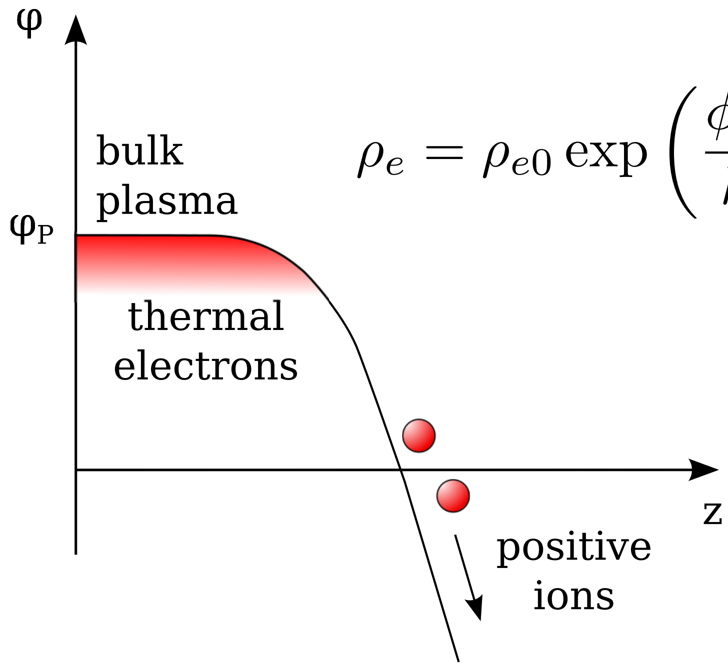
$$\frac{d\vec{x}}{dt} = \vec{v} \tag{6}$$

$$\frac{d\vec{v}}{dt} = \frac{q}{m}\big(\vec{E} + \vec{v} \times \vec{B}\big) \tag{7}$$

are usually solved either by some Runge-Kutta integrator or a Leapfrog method (Boris).

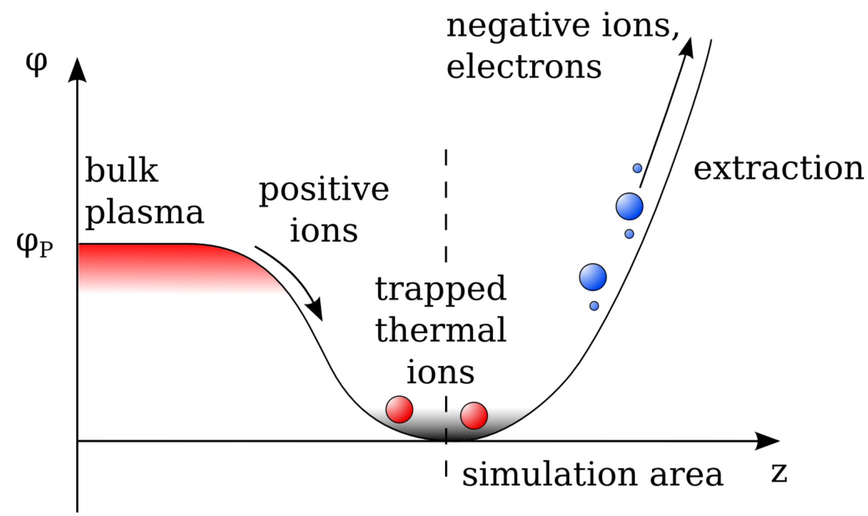
Plasma models

Positive ion extraction



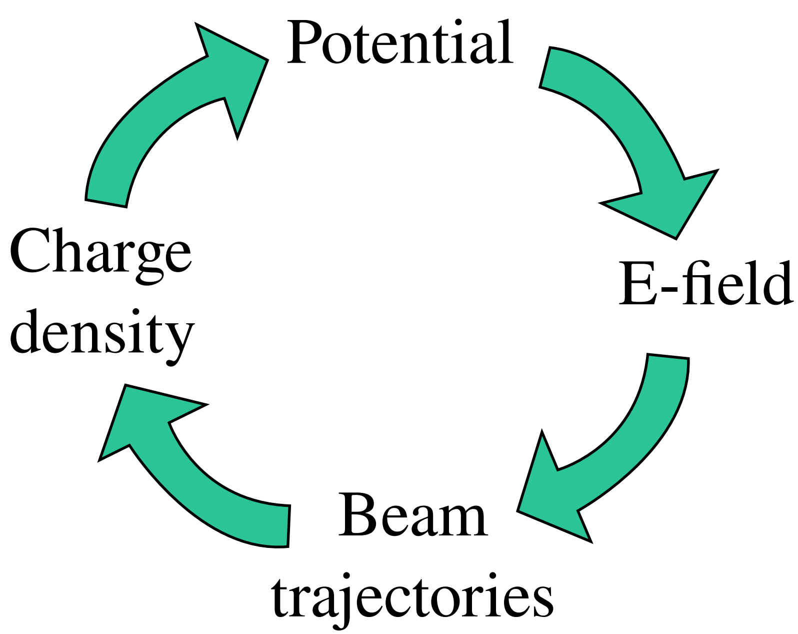
$$\rho_e = \rho_{e0} \exp\left(\frac{\phi - \phi_P}{kT_e/e}\right)$$

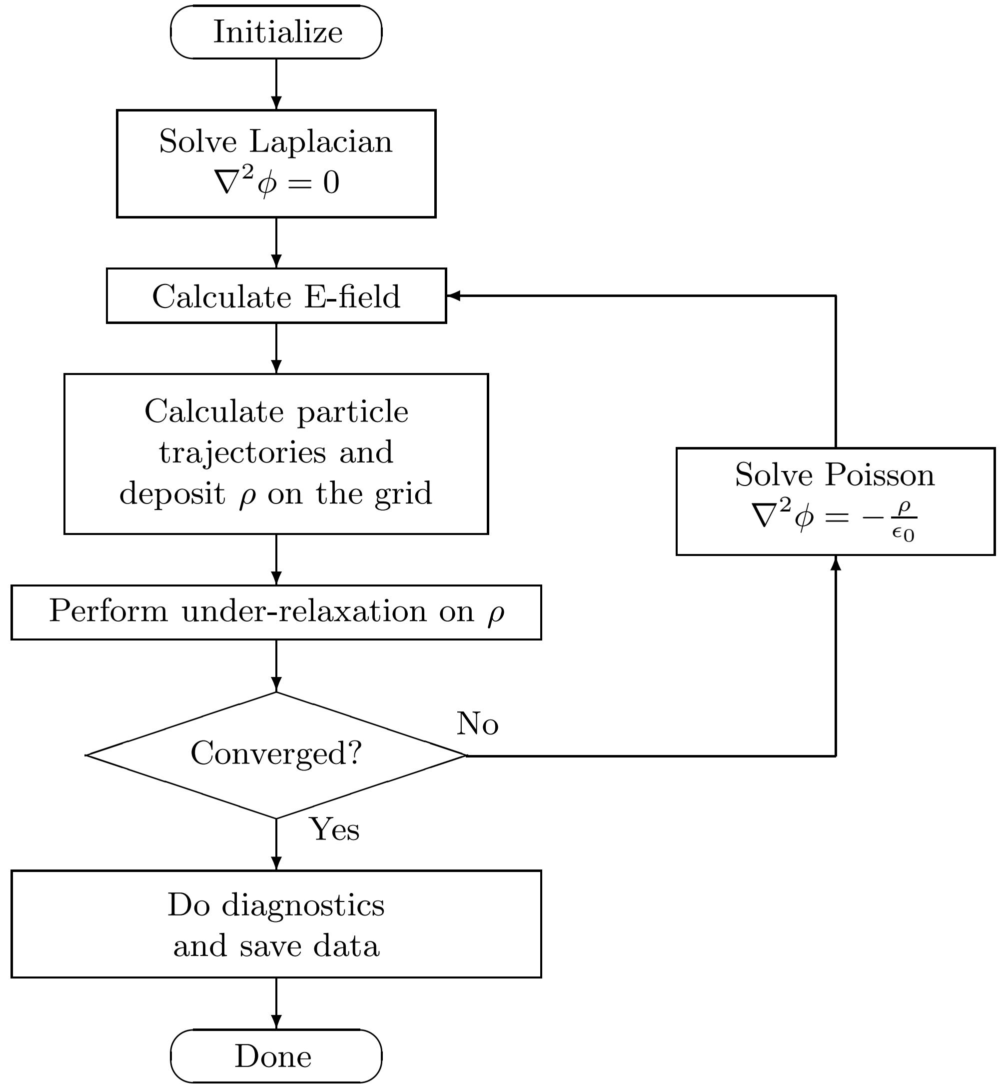
Negative ion extraction



Numerical solution

Coupled problem





Available codes of this type

- (N)IGUN — Plasma modelling for negative and positive ions, 2D only
- PBGUNS — Plasma modelling for negative and positive ions, 2D only
- KOBRA — More advanced 3D E-field solver, positive ion plasma modelling, PIC capability
- IBSIMU — Plasma modelling for negative and positive ions, 1D–3D E-field solvers

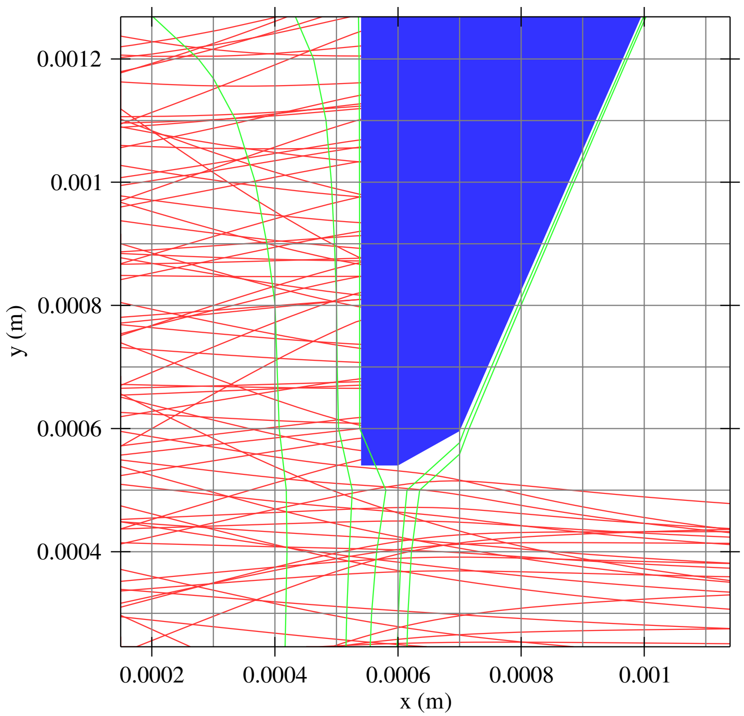
Components

What is IBSimu made of?

Ion optics with FDM

Calculation is based on evenly sized square cartesian grid(s):

- Solid mesh (node type): vacuum, solid, near solid, neumann boundary condition, ...
- Electric potential
- Electric field
- Magnetic field
- Space charge density
- Trajectory density



Electrostatic field solver

Poisson's equation

$$\nabla^2 \phi = -\frac{\rho}{\epsilon_0}$$

Finite Difference representation for vacuum node i:

$$\frac{\phi_{i-1} - 2\phi_i + \phi_{i+1}}{h^2} = -\frac{\rho_i}{\epsilon_0},$$

Neumann boundary node i:

$$\frac{-3\phi_i + 4\phi_{i+1} - \phi_{i+2}}{2h} = \frac{d\phi}{dx}$$

and Dirichlet (fixed) node i:

$$\phi_i = \phi_{\text{const}}$$

1D example

Solve a 1D system of length $L = 10$ cm, charge $\rho = 1 \cdot 10^{-6}$ C/m^3 and boundary conditions

$$\frac{\partial \phi}{\partial x}(x=0) = 0 \text{ V/m} \qquad \text{and} \qquad \phi(x=L) = 0 \text{ V}.$$

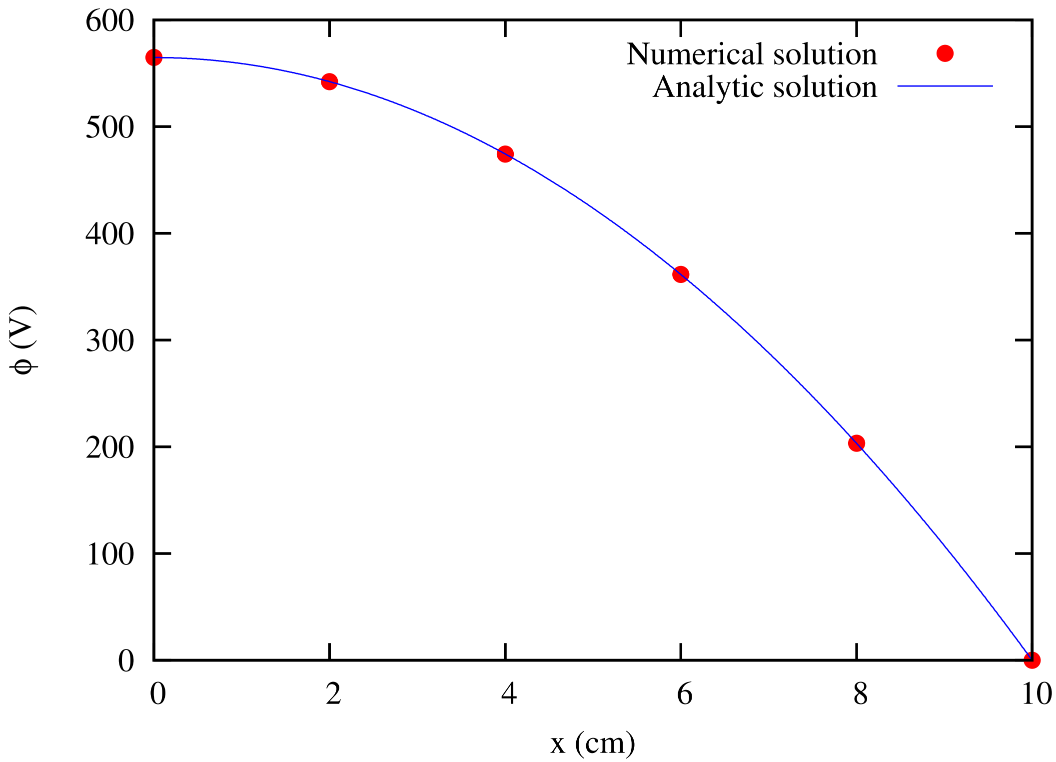
The system is discretized to $N = 6$ nodes. Problem in matrix form:

$$\begin{pmatrix} -3 & 4 & -1 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \phi_6 \end{pmatrix} = \begin{pmatrix} 2h \frac{\partial \phi}{\partial x}(0) \\ -h^2 \frac{\rho}{\epsilon_0} \\ -h^2 \frac{\rho}{\epsilon_0} \\ -h^2 \frac{\rho}{\epsilon_0} \\ -h^2 \frac{\rho}{\epsilon_0} \\ \phi(L) \end{pmatrix}$$

Solving the matrix equation we get ...

1D example

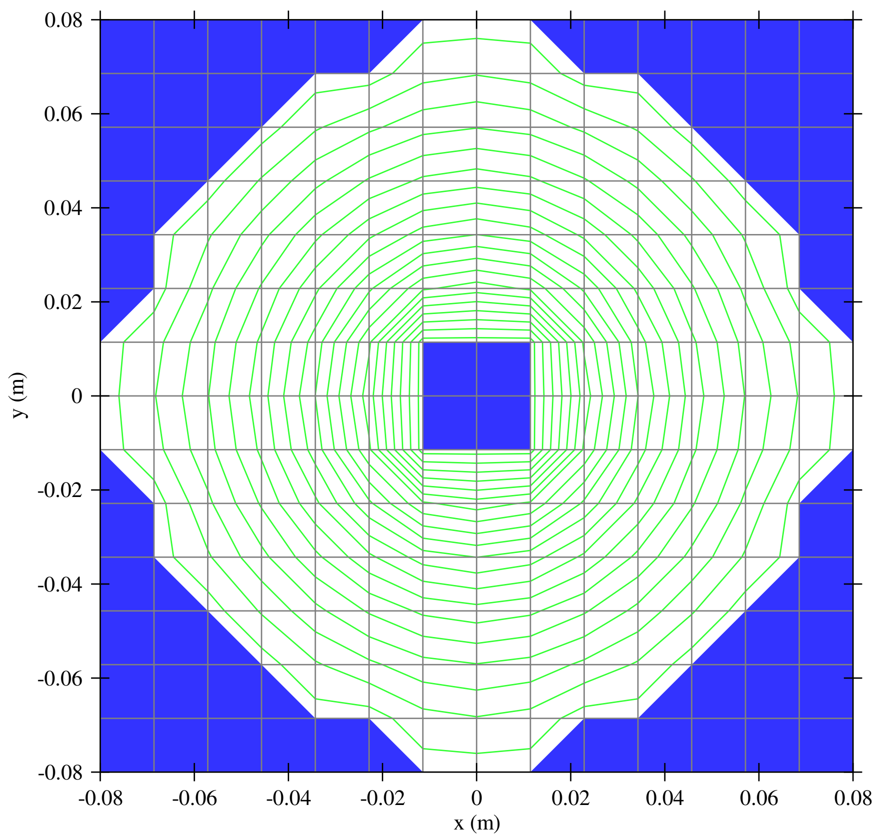
... perfect agreement with analytic result



but only because of flat charge distribution and boundaries defined exactly at node locations.

Jagged boundaries

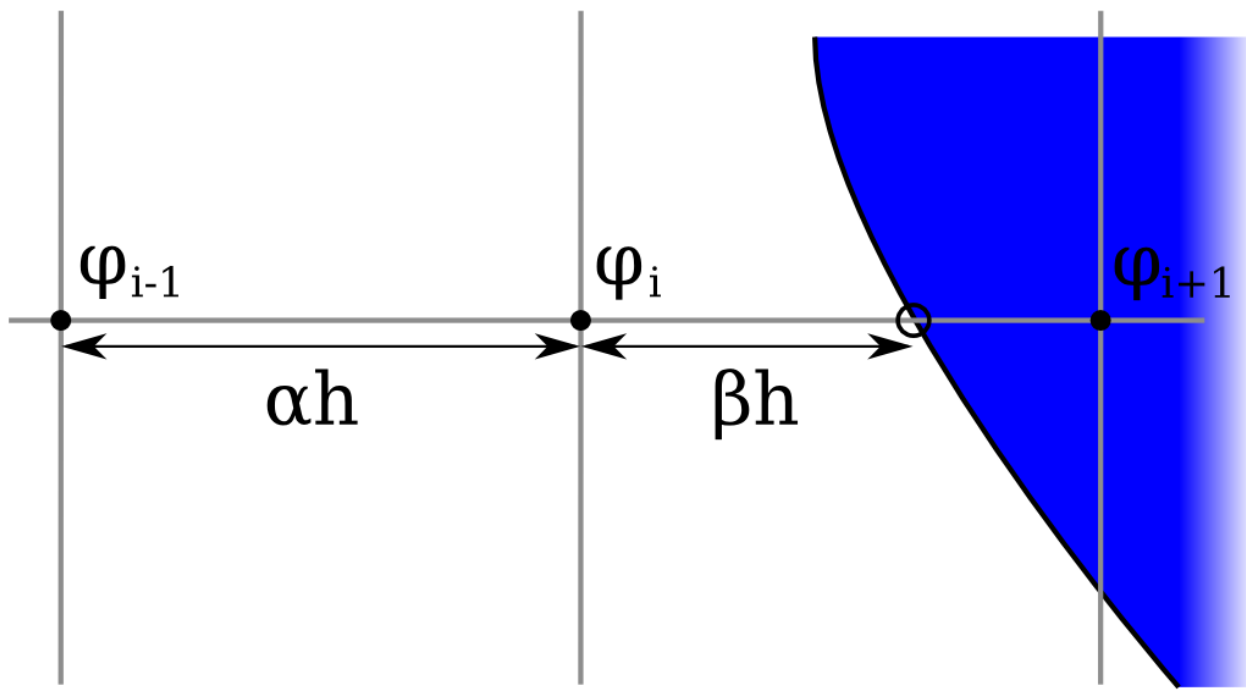
In higher dimensions basic FDM generally suffers from jagged boundaries (nodes don't coincide with surfaces).



Smooth boundaries

Derivatives in Poisson's equation featured with uneven distances

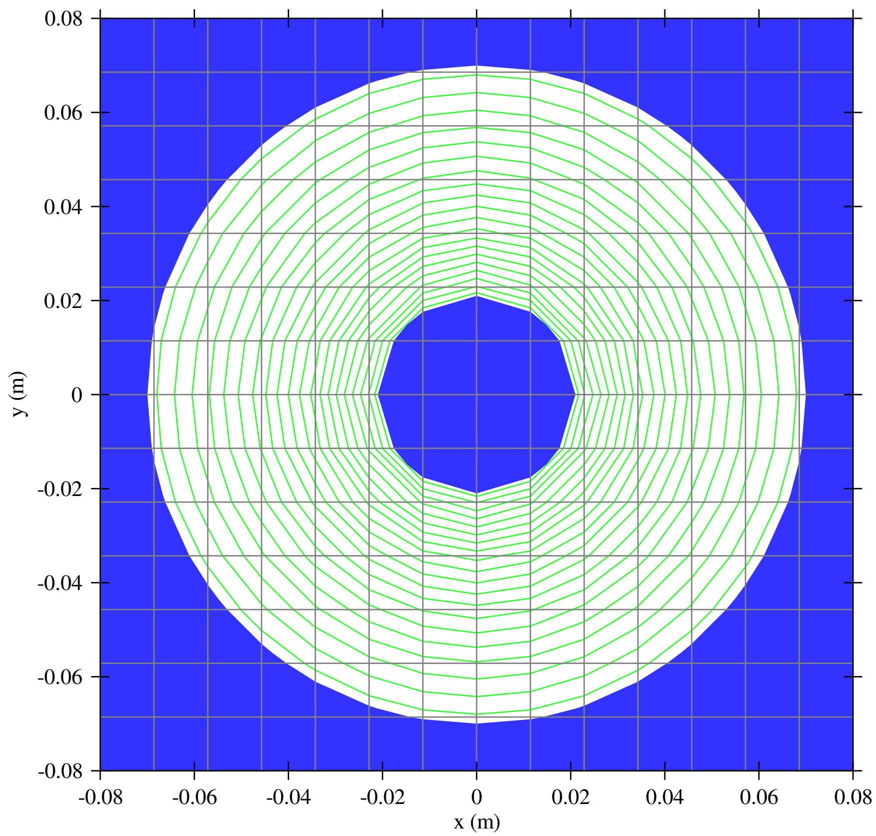
$$\frac{\beta\phi(x_0-\alpha h)-(\alpha+\beta)\phi(x_0)+\alpha\phi(x_0+\beta h)}{\frac{1}{2}(\alpha+\beta)\alpha\beta h^2}=-\frac{\rho(x_0)}{\epsilon_0}$$



Taneli Kalvas

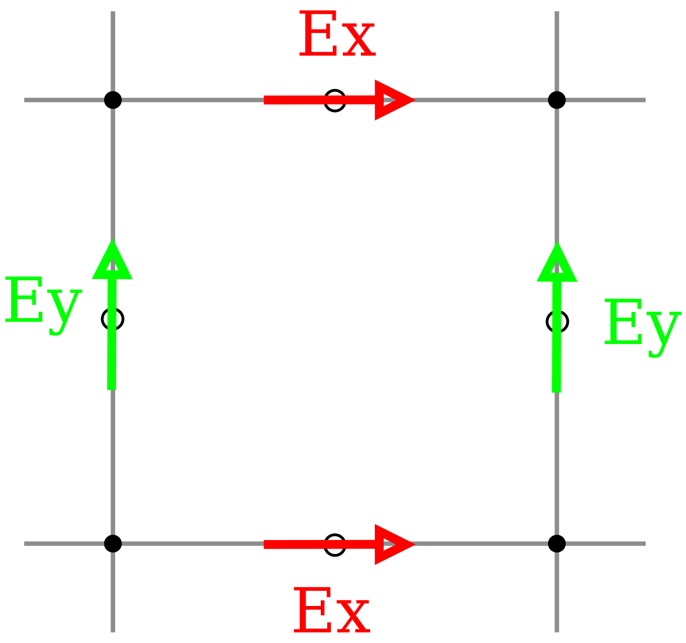
Smooth boundaries

A much better solution with smooth boundaries is achieved.



Electric field calculation

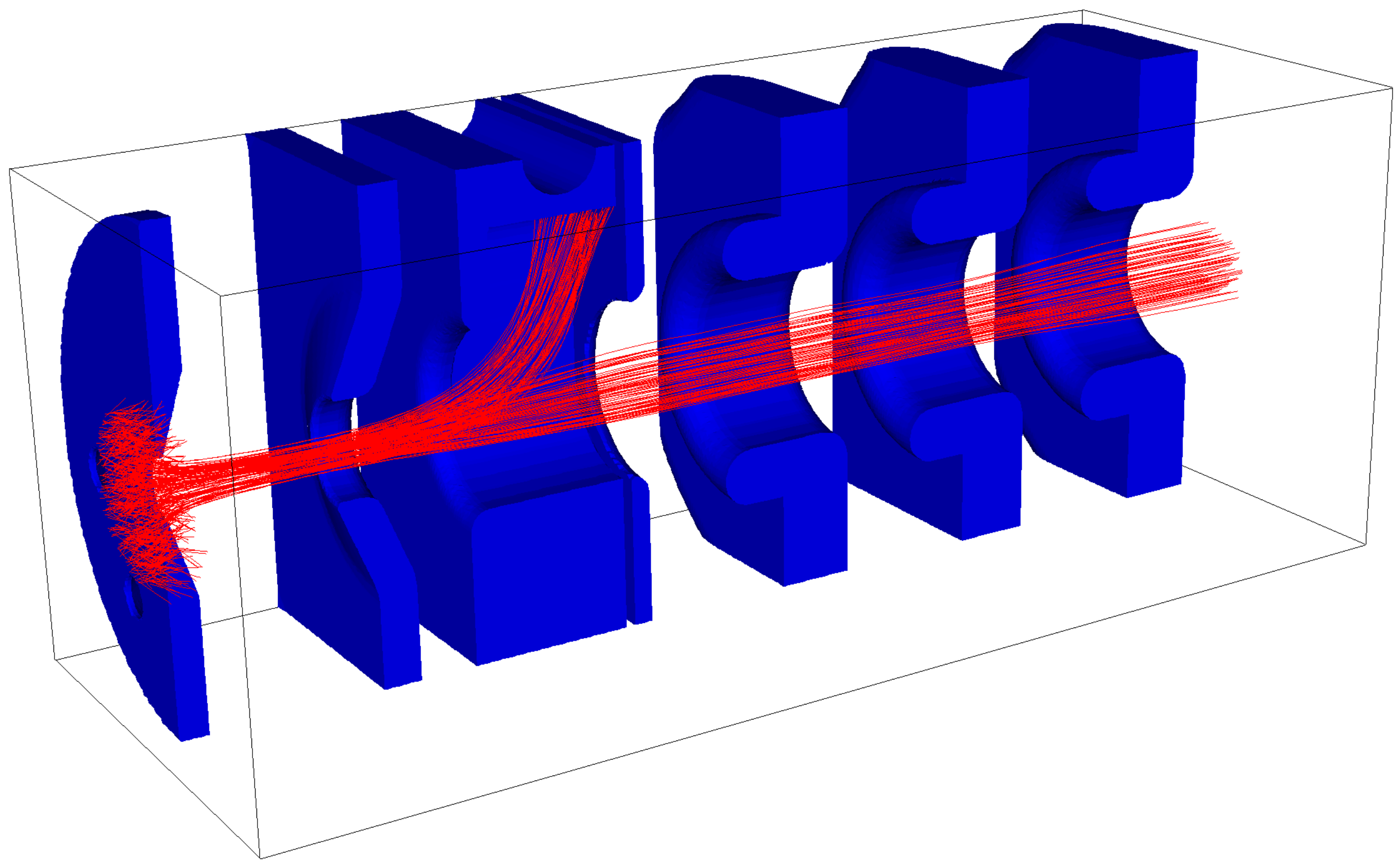
Electric field is calculated between the nodes simply by $E = \frac{V}{h}$.



Electric field nodes between potential nodes.

Problem geometries

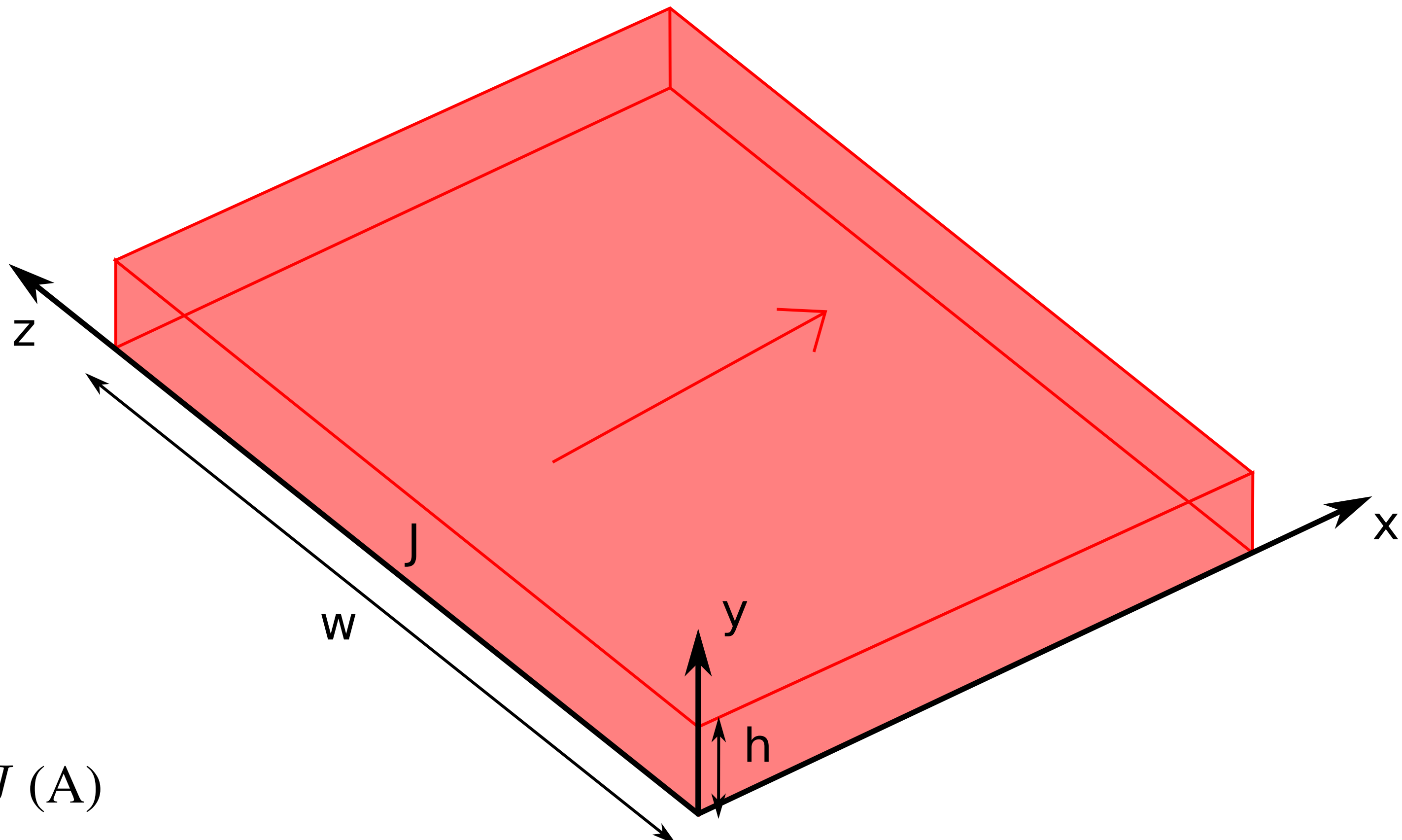
3D: $\vec{E}$, $\vec{B}$, $\vec{x}$, $\vec{v}$



Each simulated macro particle carries current I.

Problem geometries

Planar 2D: (E_x, E_y), B_z, (x, y), (v_x, v_y)



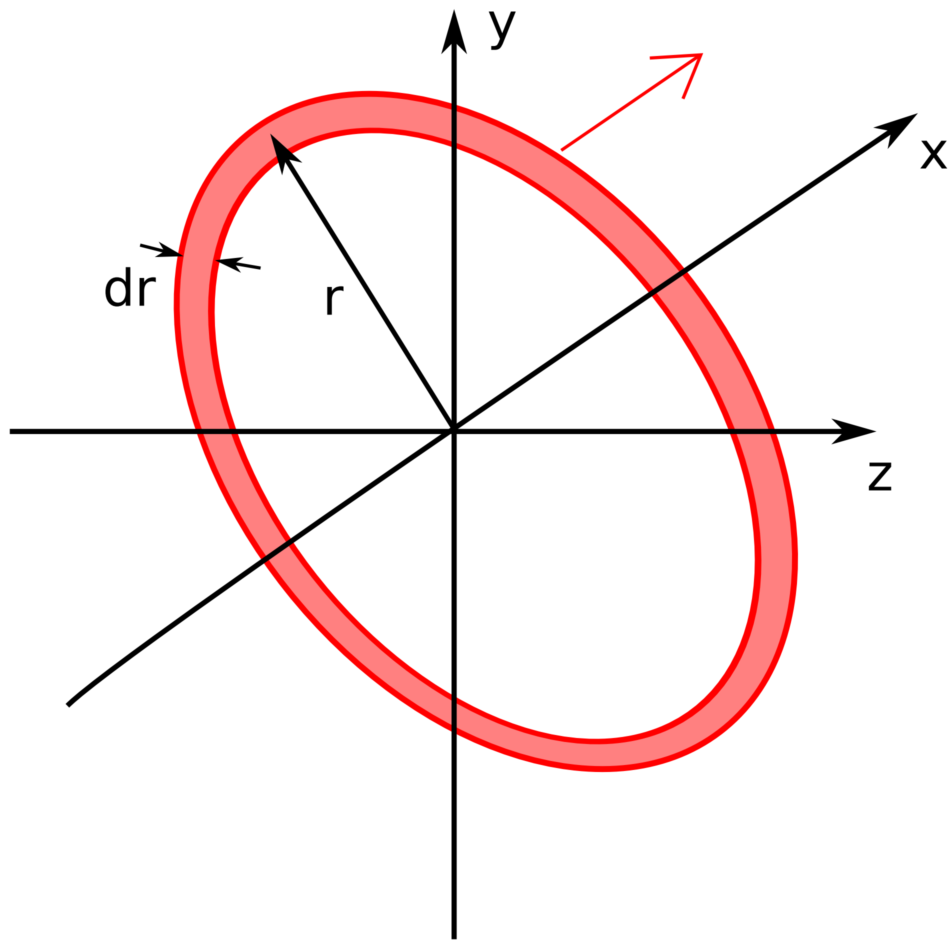
$I = whJ$ (A)

$\tilde{I} = hJ$ (A/m)

Infinite in z-direction

Problem geometries

Cylindrically symmetric: (E_x, E_r), (B_x, B_r, B_θ), (x, r), (v_x, v_r, ω)



$I = 2\pi r \mathrm{d}r J$ (A)

Trajectory calculation

Population of macro particles is calculated with following properties:

- Charge: q
- Mass: m
- Current carried: I or $\tilde{I}$
- Time, position and velocity coordinates:
  - 2D: (t, x, v_x, y, v_y)
  - Cylindrical symmetry: $(t, x, v_x, r, v_r, \omega)$, where $\omega = \frac{d\theta}{dt}$
  - 3D: $(t, x, v_x, y, v_y, z, v_z)$

Trajectory calculation

Calculation of trajectories done by integrating the equations of motion

$$\begin{aligned}
\frac{dx}{dt} &= v_x \\
\frac{dy}{dt} &= v_y \\
\frac{dz}{dt} &= v_z \\
\frac{dv_x}{dt} &= a_x = \frac{q}{m}(E_x + v_y B_z - v_z B_y) \\
\frac{dv_y}{dt} &= a_y = \frac{q}{m}(E_y + v_z B_x - v_x B_z) \\
\frac{dv_z}{dt} &= a_z = \frac{q}{m}(E_z + v_x B_y - v_y B_x)
\end{aligned}$$

Trajectory calculation

... and in cylindrical symmetry:

$$
\begin{aligned}
\frac{dx}{dt} &= v_x \\
\frac{dr}{dt} &= v_r \\
\frac{dv_x}{dt} &= a_x = \frac{q}{m}(E_x + v_r B_\theta - v_\theta B_r) \\
\frac{dv_r}{dt} &= a_r + r\omega^2 = \frac{q}{m}(E_y + v_\theta B_x - v_x B_\theta) + r\omega^2 \\
\frac{d\omega}{dt} &= \frac{1}{r}\left(a_\theta - v_r\omega\right) = \frac{1}{r}\left(\frac{q}{m}(v_x B_r - v_r B_x) - 2v_r\omega\right),
\end{aligned}
$$

where $v_\theta = r\frac{d\theta}{dt} = r\omega$

Trajectory calculation

For relativistic particles, the equation of motion in 3D is

$$\gamma \frac{d\vec{v}}{dt} + \gamma^3 \frac{\vec{v}}{c^2} \left(\vec{v} \cdot \frac{d\vec{v}}{dt} \right) = \frac{q}{m} (\vec{E} + \vec{v} \times \vec{B}),$$

where $\gamma = 1/\sqrt{1 - v^2/c^2}$ is the relativistic gamma factor. The particle acceleration $d\vec{v}/dt$ is solved from the equation above and it is used in the defining system of equations.

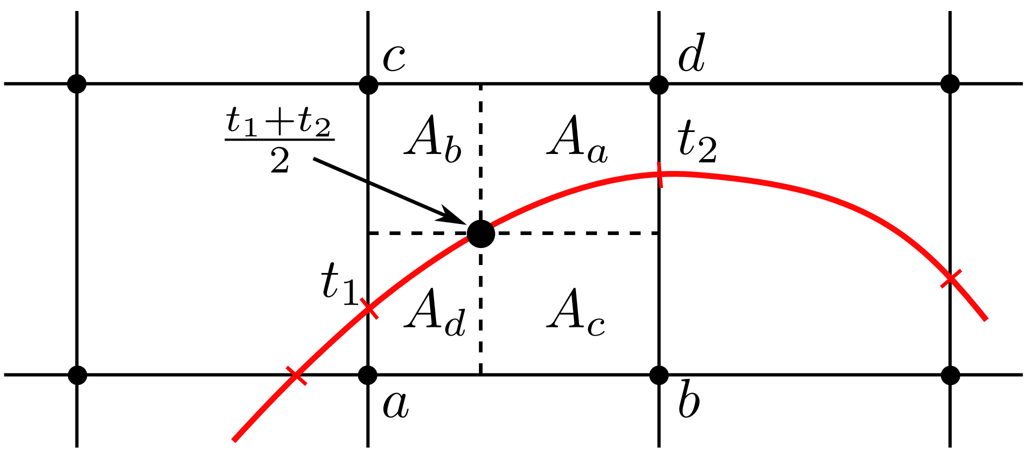
Space charge deposition

The space charge

$$Q = I\Delta t$$

is deposited inside each grid cell assuming that charge cloud is centered at the location, where the particle was at $t = \frac{1}{2}(t_1 + t_2)$. The nodes receive charge density $\rho_{i,j,k} = QS(\vec{r} - \vec{r}_{i,j,k})$ according to trilinear weighting function

$$S(x,y,z) = \begin{cases} \frac{1}{h^3}\left(1 - |x/h|\right)\left(1 - |y/h|\right)\left(1 - |z/h|\right) & |x| < h, |y| < h, |z| < h \\ 0 & \text{otherwise.} \end{cases}$$



Three modes of calculation

1. If fields ($\vec{E}$ and $\vec{B}$) are calculated and particles are tracked through them once (no iteration), one arrives at a solution, which is NOT dependent on space charge (zero current).

- Fast, < 10 min.

2. If the iteration cycle (with $\phi \rightarrow \vec{E} \rightarrow$ trajetories $\rightarrow \rho \rightarrow \phi$) is done until convergence, one solves a problem with *static fields* and *continuous beams*.

- IBSimu calculations mainly in this category
- Quite slow, ~ 1 h for 3D typically.

3. If beam or fields have dependencies on time, one has to use the particle-in-cell (PIC) method, which iterates the system chronologically through time in finite Δt time steps.

- Painfully slow, ~ 12 h for 3D typically.

Plasma flux

The plasma flux to a surface is

$$J = \frac{1}{4} q n \bar{v} = q n \sqrt{\frac{kT}{2\pi m}}$$

Extraction hole: ion beam samples plasma species with weight $\propto m^{-1/2}$.

Plasma flux sets the maximum current extractable

$$I = J A_{\text{meniscus}},$$

where the area of plasma meniscus $A_{\text{meniscus}} \neq A_{\text{aperture}}$ and therefore not quite constant. N-dimensional simulations needed for better estimates.

Thermal plasma sheath

Classic 1D plasma sheath theory: In an electron-ion plasma a positive plasma potential is formed due to higher mobility of electrons. Situation is described by Poisson equation

$$\frac{d^2U}{dx^2} = -\frac{\rho}{\epsilon_0} = -\frac{en_0}{\epsilon_0}\left[\sqrt{1-\frac{2eU}{m_i v_0^2}} - \exp\left(\frac{eU}{kT_e}\right)\right],$$

where the ions entering the sheath have an initial velocity

$$v_0 > v_{\mathrm{Bohm}} = \sqrt{\frac{kT_e}{m_i}}$$

or kinetic energy

$$E_0 > \frac{1}{2}m_i v_{\mathrm{Bohm}}^2 = \frac{1}{2}kT_e.$$

Model applies quite well for positive ion plasma extraction.

Positive ion plasma extraction model

Groundbreaking work by S. A. Self, *Exact Solution of the Collisionless Plasma-Sheath Equation*, Fluids **6**, 1762 (1963) and
J. H. Whealton, *Optics of single-stage accelerated ion beams extracted from a plasma*, Rev. Sci. Instrum. **48**, 829 (1977):

$$\frac{d^2U}{dx^2} = -\frac{\rho}{\epsilon_0} = -\frac{\rho_{\mathrm{rt}} + \rho_{\mathrm{e}}(U)}{\epsilon_0}$$

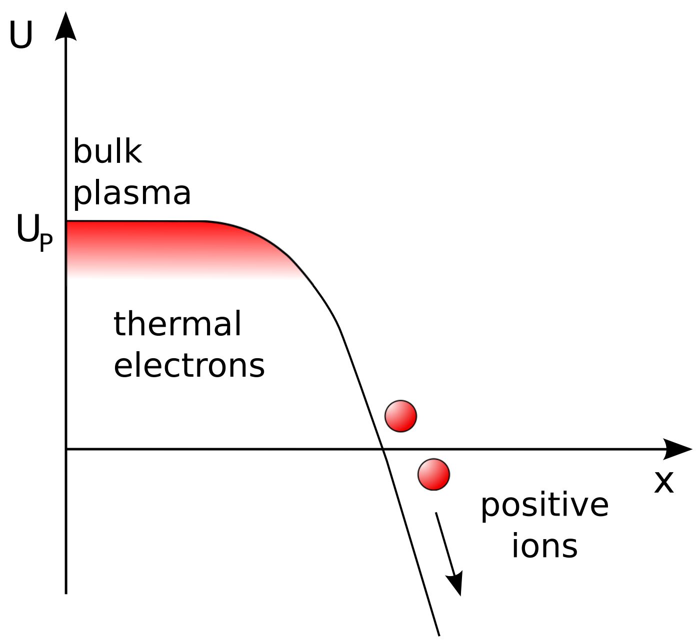
- Model has been used very successfully for describing positive ion extraction systems since.
- Assumptions: no ion collisions, no ion generation, electron density only a function of potential (no magnetic field).
- Take the model with a semiempirical approach and use it as a tool proving to yourself that it works for your case — don't take it for granted.

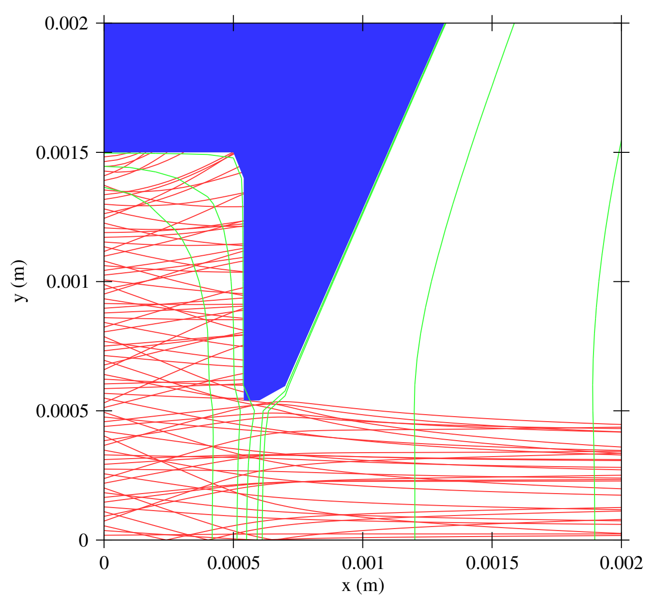
Positive ion plasma extraction model

Modelling of positive ion extraction

- Ray-traced positive ions entering sheath with initial velocity
- Nonlinear space charge term (analytic in Poisson's equation):

$$\rho_{\mathrm{e}} = \rho_{\mathrm{e}0} \exp\left(\frac{U - U_P}{kT_e/e}\right)$$





Difficulties in modelling extraction systems

1. Amount of parameters fed to the model is quite large

   - Extracted species: J_i, T_i, v_0
   - Positive ion plasma model: T_e, U_P
   - Negative ion plasma model: T_i, E_i/T_i, gas stripping loss of ions
   - All: space charge compensation degree and localization in LEBT

Methods: educated guessing (literature data), plasma measurements and matching to beam measurements (emittance scans).

2. Level of approximation in plasma model

   - Very difficult to estimate
   - Comparison to simulations containing more (and more accurate) physics

Installation, versions, new features, etc

Installation/compilation

- Where to get and what to get
  - Sourceforge/Files: Releases
  - Git: Development versions (recommended)
- Tools
  - Release: g++, POSIX compilation environment
  - Git package needs GNU toolchain autotools!
- Dependencies
  - GNU Scientific Library, zlib
  - GTK+-3.0 (cairo, libpng, FontConfig, Freetype2)
  - Conditional: GtkGLExt, UMFPACK, CSG
- Compilation, optimization

Versions

Stable releases

- Public versions started from 1.0.0, currently 1.0.6 (Jun 15 2015)

Development versions

- IBSimu does not have unstable releases, development versions are available through git, a revision control system.
- Version stamp is given according to the last stable release with additional "dev" tag, currently 1.0.6dev.
- Git tags each revision with a hexadecimal string, which is printed along with a date on salution string, for example

  `Ion Beam Simulator 1.0.6dev (d938f8b, Mon Jun 15 12:27:35 2015 +0300)`

- Every now and then a development version is packaged on `http://ibsimu.sourceforge.net/download.html` page.

Versions

Development versions

- Development versions have bug fixes, new features, more documentation, etc.
- More probable to have bugs, changes in API
- Use is recommended
- Different development branches may exist (there used to be a `new_solver` branch). Users will be informed if necessary.

Git

Optimization and non-standard pkgconfig directory, edit `.bashrc`:

```
export CFLAGS="-O2"
export LD_LIBRARY_PATH="/home/tvkalvas/lib"
export PKG_CONFIG_PATH="/home/tvkalvas/lib/pkgconfig"
```

Fetch IBSimu, configure and compile:

```
> git clone git://ibsimu.git.sourceforge.net/gitroot/ibsimu/ibsimu
> cd ibsimu
> ./reconf
> ./configure --prefix=/home/tvkalvas
> make -j4
> make install
```

Fetch IBSimu, configure and compile:

```
> git pull
```

It might not be necessary to run the reconf and configure scripts. The need depends on what was updated.

```
> ./reconf
> ./configure --prefix=/home/tvkalvas
> make -j4
> make install
```

Bug reports, requests, contributions

Found a bug or want something that does not exist?

- Use email list
- I will usually make small updates quickly upon request
- Larger scale work can be discussed

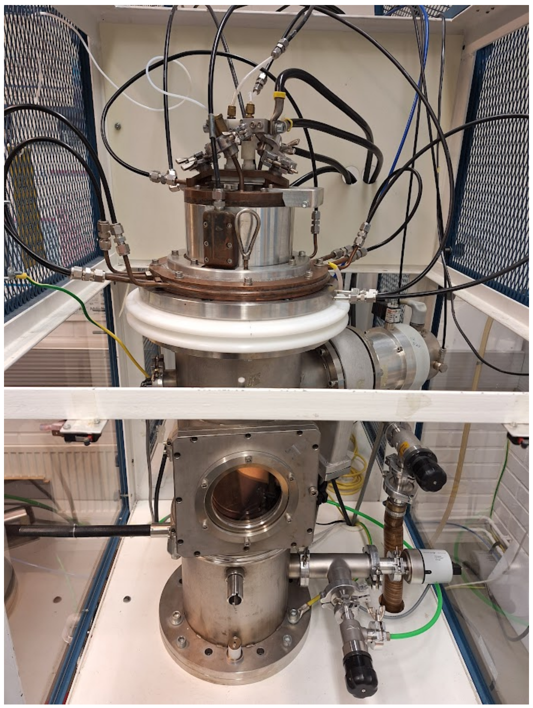
Contributions

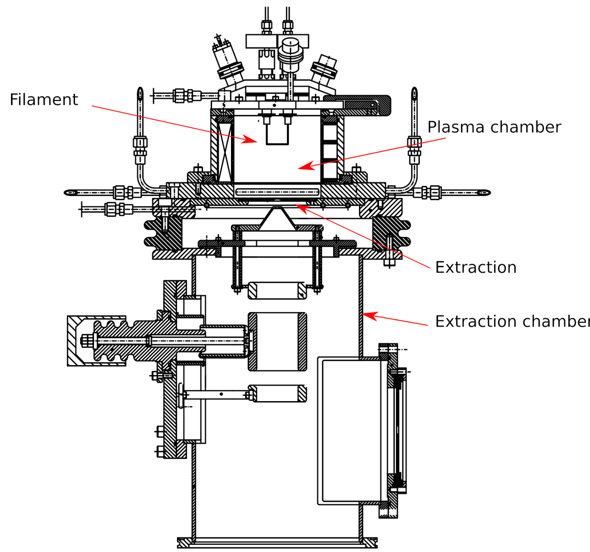
- If you are capable, you are welcome to contribute
- Please patch (even preliminary) work with `git diff`

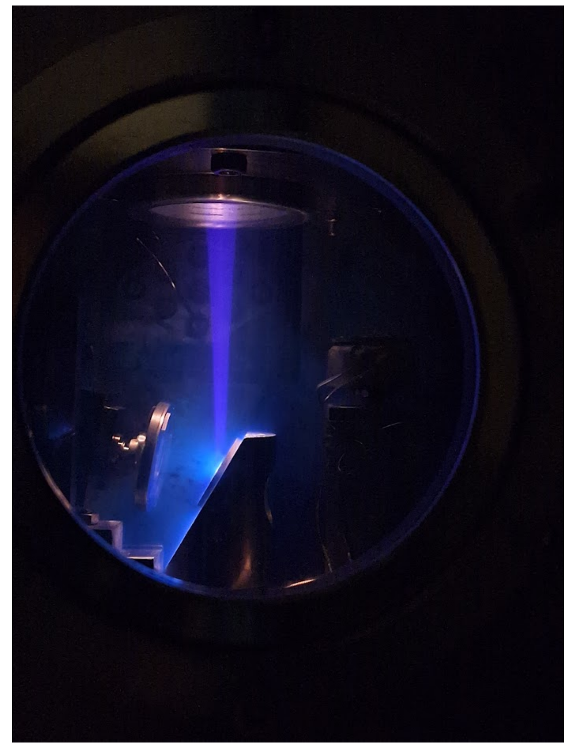
JYFL “Sputter ion source”

JYFL “Sputter ion source”

Filament driven ion source for production of Ar^+ beam at 10 keV, < 1 mA. Used for thin film production using sputtering.

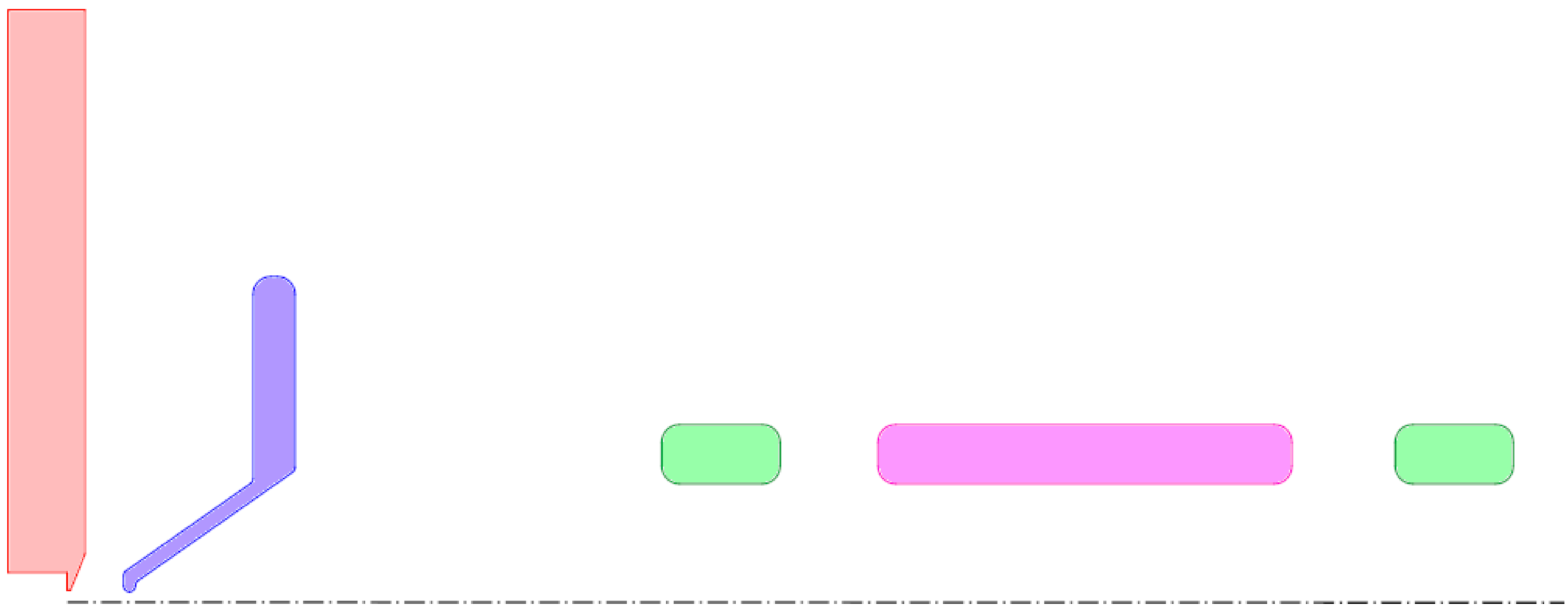






DXF file for geometry

LibreCAD: `sputter.dxf`



Electrodes in named layers: “plasma”, “puller”, “gnd” and “einzel”

Basic implementation of driving code

- Geometry: cylindrical symmetry, minimize simulation box
- Boundary conditions, plasma size
- Physical parameters: voltages, plasma and beam parameters
- Simulation parameters: number of iteration rounds, number of particles, under-relaxation, mesh size
- Diagnostics: check for convergence, plotter
- Reading the console output

See `sputter.cpp`

Basic positive ion extraction plasma model

Continuing with the Bohm sheath: Compensation by Boltzmann distributed electrons

$$\rho_e = \rho_{e0} \exp\left(\frac{\phi - \phi_S}{kT_e/e}\right)$$

and monoenergetic flux of ions accelerated by the pre-sheath to velocity

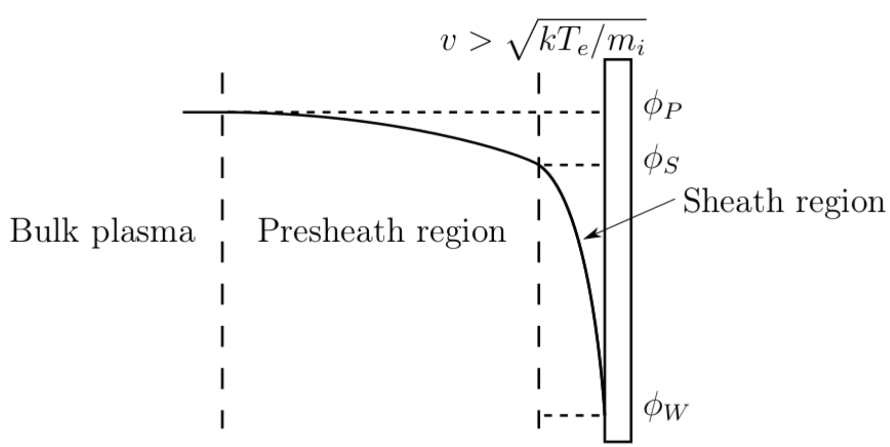
$$v \geq v_B = \sqrt{\frac{kT_e}{m_i}},$$

the Bohm velocity, which implies

$$\phi_P - \phi_S = \frac{kT_e}{2e}.$$

By equalizing particle fluxes one can solve the plasma potential

$$\phi_P - \phi_W = \frac{kT_e}{2e}\left(1 + \log\left(\frac{m_i}{2\pi m_e}\right)\right).$$



Numbers

In this case $T_e = 2$ eV, which leads to

$$\phi_P - \phi_S = \frac{kT_e}{2e} = 1 \text{ V}.$$

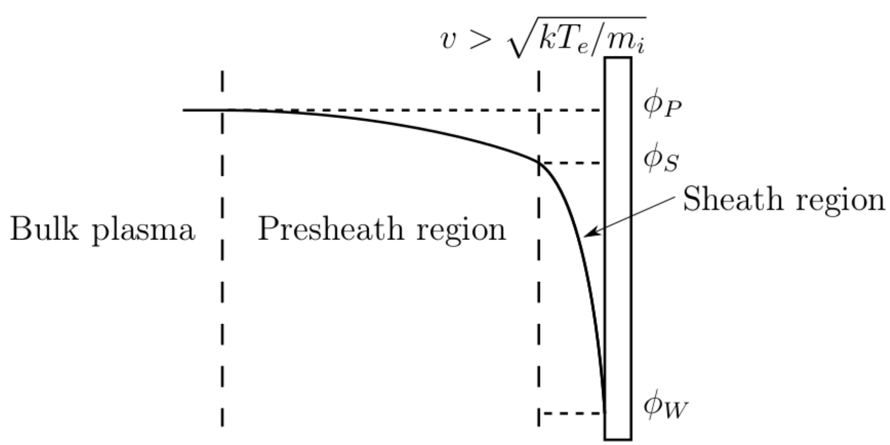
The plasma potential is thus

$$\phi_P - \phi_W = 1 \text{ V} \left(1 + \log\left(\frac{40 \cdot 1822}{2\pi}\right)\right) = 10.36 \text{ V}.$$

In IBSimu plasma potential is

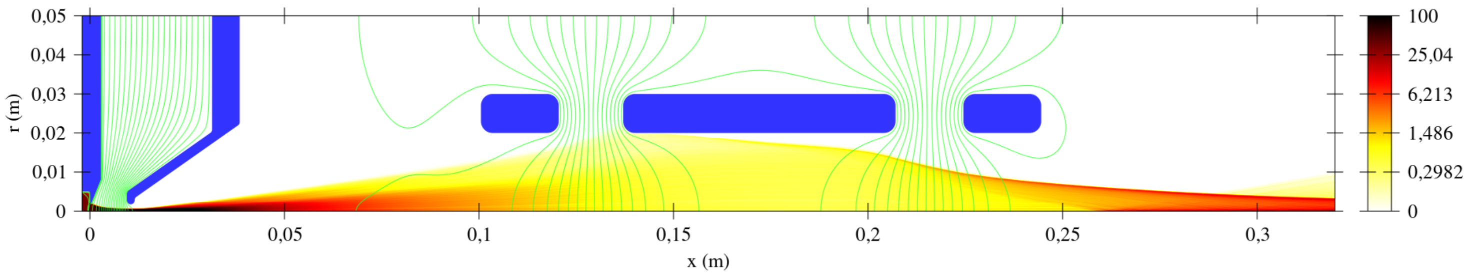
$$\phi_S - \phi_W = \phi_P - \phi_W - (\phi_P - \phi_S) = 9.36 \text{ V}.$$

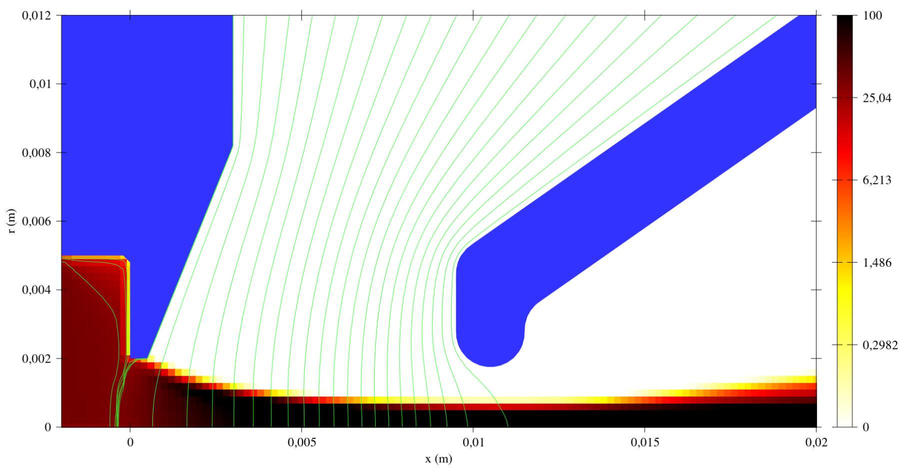
This is not enforced in IBSimu, but can be selected by the user.

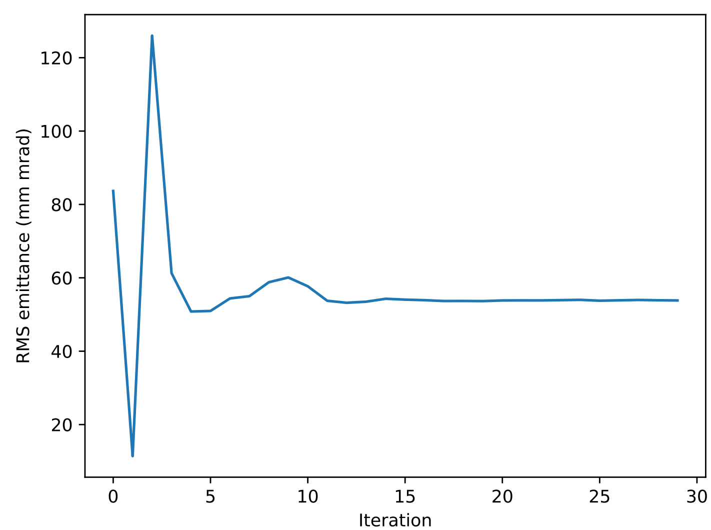


Quick calculation

Quick calculation on my laptop (30 iterations, <600 sec) shows decent results







Resolution

The previous simulation had $h = 0.2$ mm. With plasma electrode aperture $r = 2$ mm we have 10 mesh cells in the radius of the aperture. Emittance can be expected to be within 10 % of the “correct” value.

If we calculate the charge density n in the plasma flux

$$n = \frac{J}{qv_0},$$

we can calculate the Debye length.

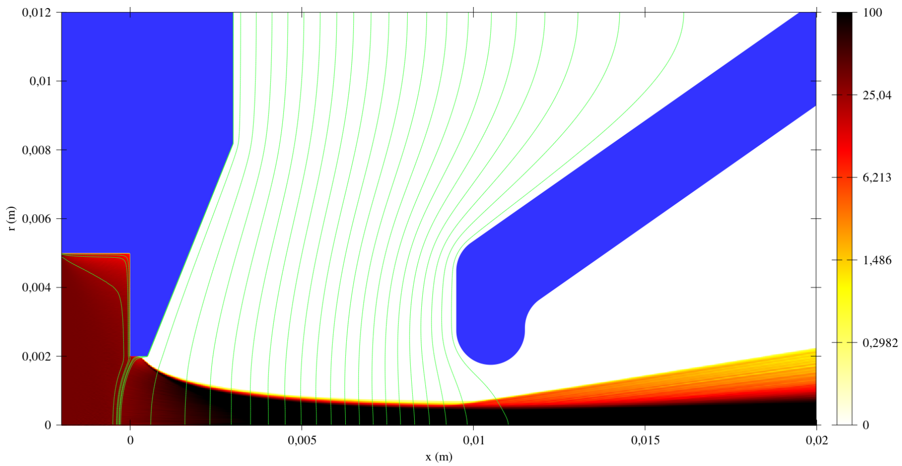
Common wisdom is that the sheath can only be properly resolved if the simulation mesh is of the order of Debye length

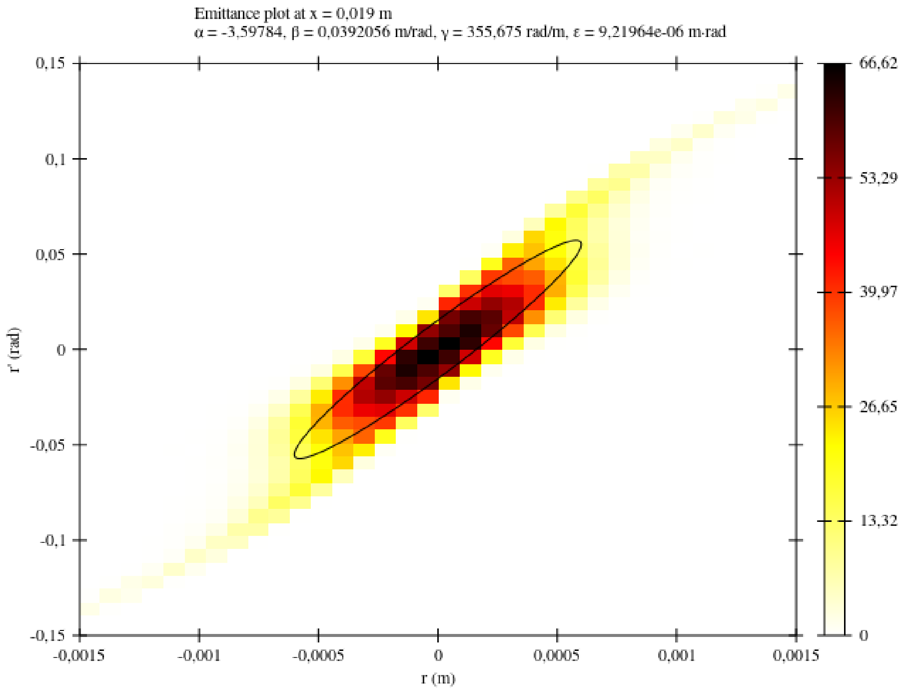
$$\lambda_D = \sqrt{\frac{\epsilon k_B T_e}{ne^2}} = 35\ \mu\text{m}.$$

However, in most IBSimu simulations the potential shape (far from electrodes) is well modelled even in cases with $h \sim 10\lambda_D$.

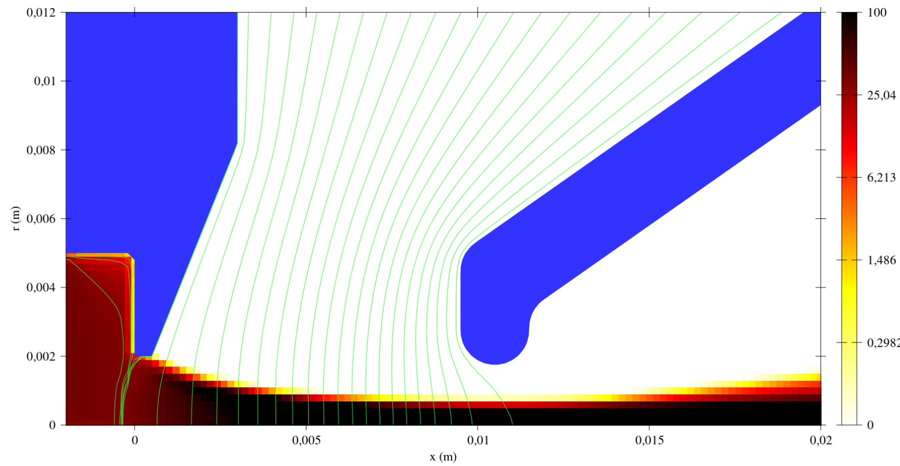
Zoomed in simulation

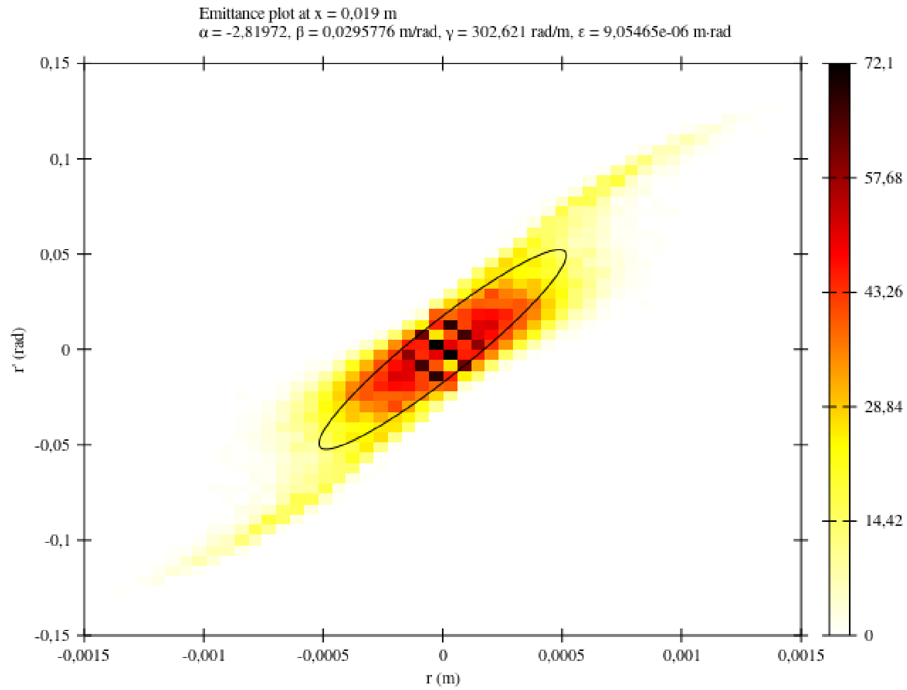
$h = 20$ µm





$h = 200$ µm

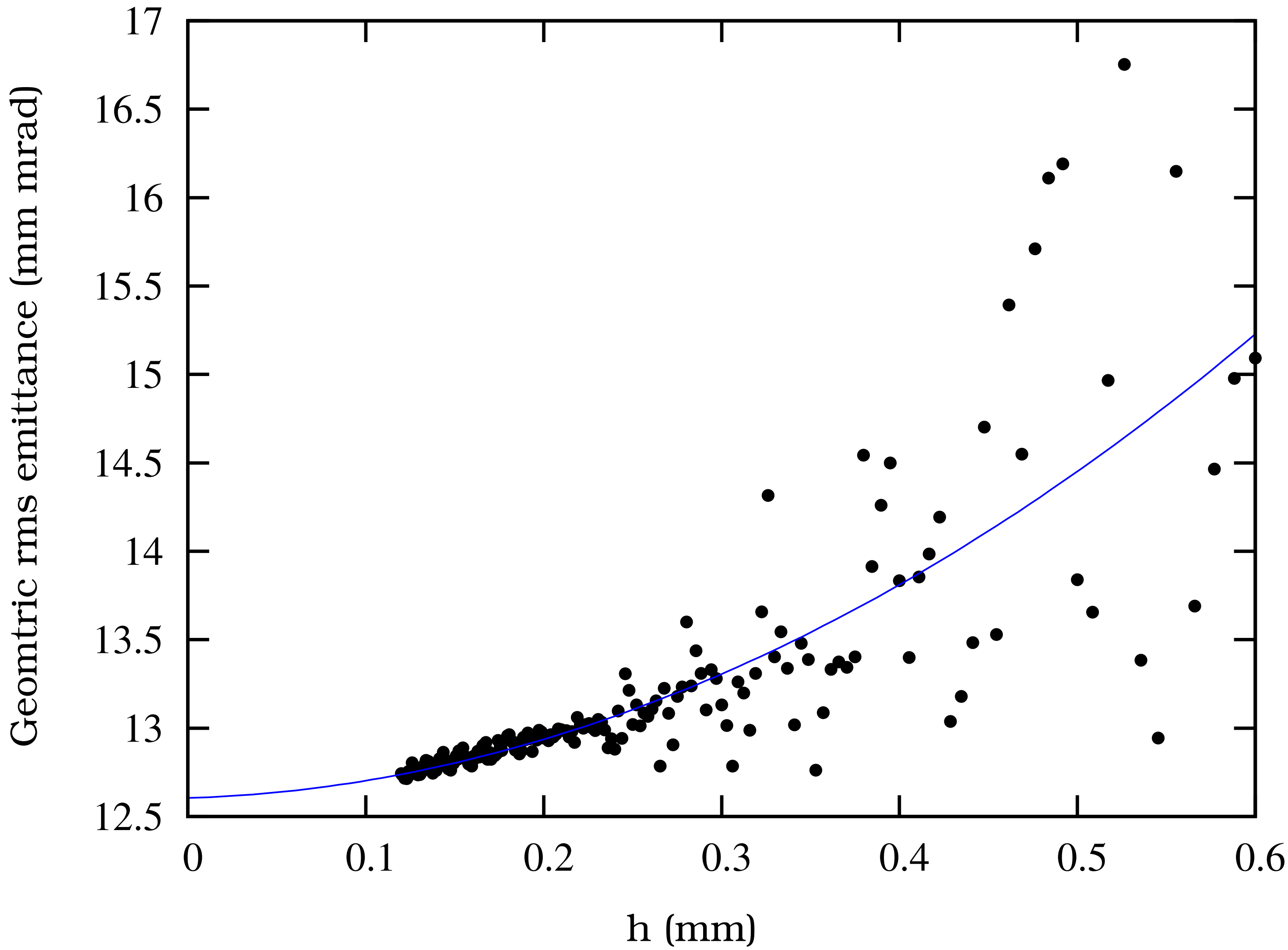




Taneli Kalvas

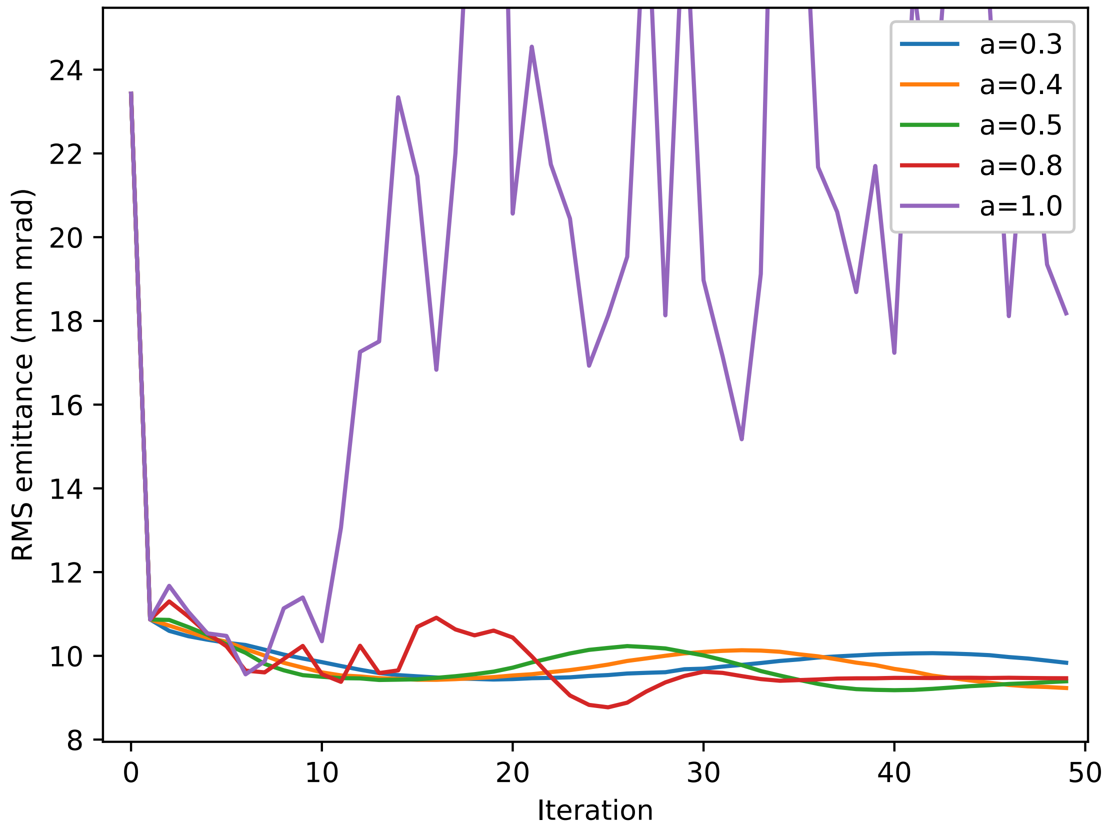
Effect of resolution generally

Different geometry, 3D case, $r_{\text{plasma}} = 3$ mm, $r_{\text{plasma}}/h \in [5, 25]$.



Convergence

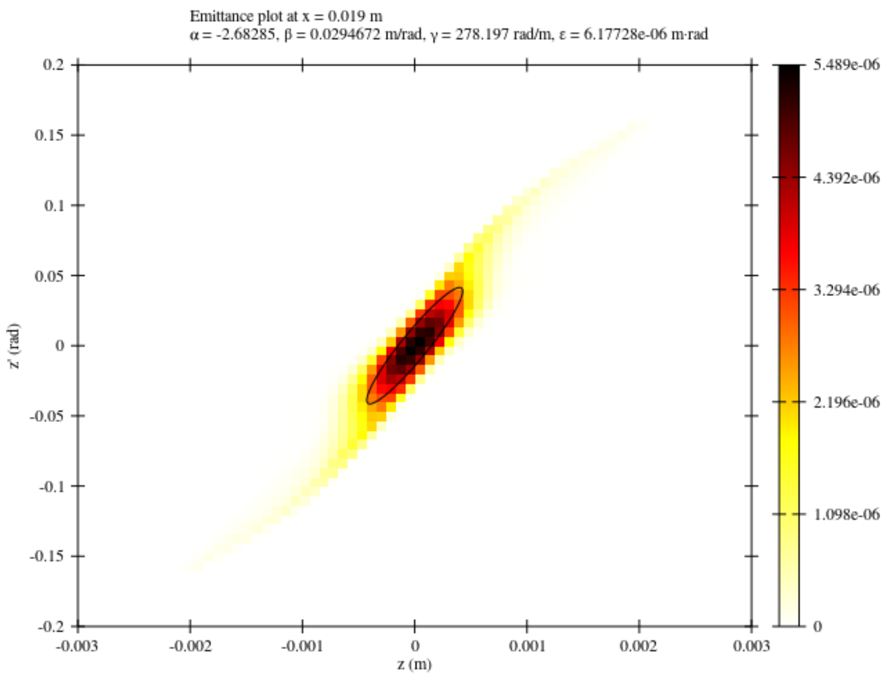
High resolution simulation did not converge with standard parameters. Needs under-relaxation.

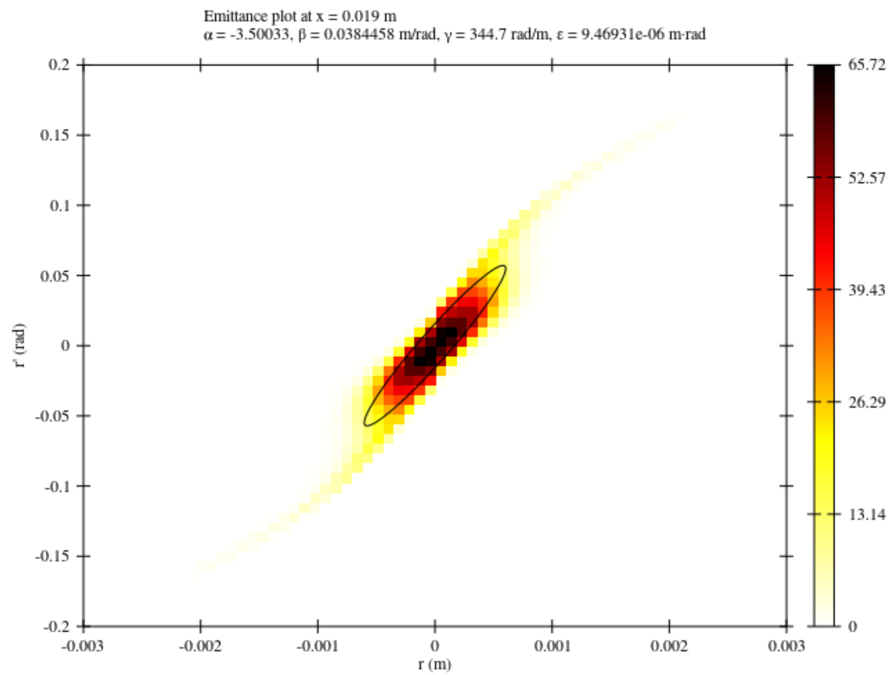


It is important to note that by default IBSimu particles are sampled from quasi-random distributions.

Emittances in cylindrical geometries

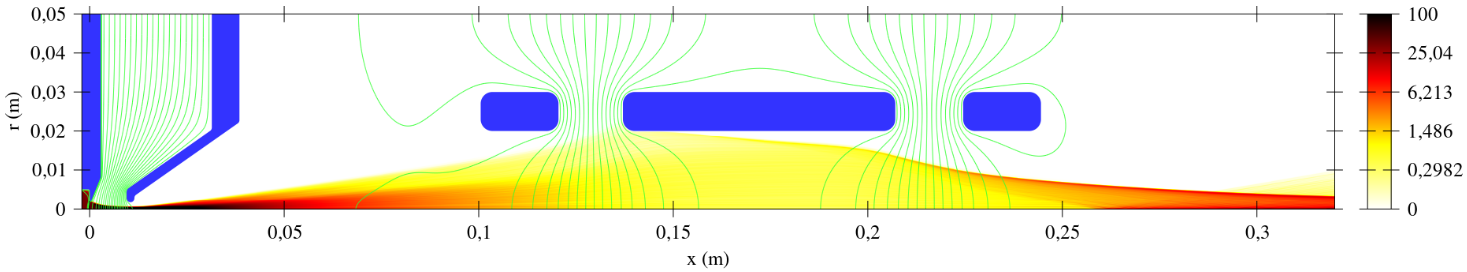
Emittance numbers are statistical rms values calculated using all particles.

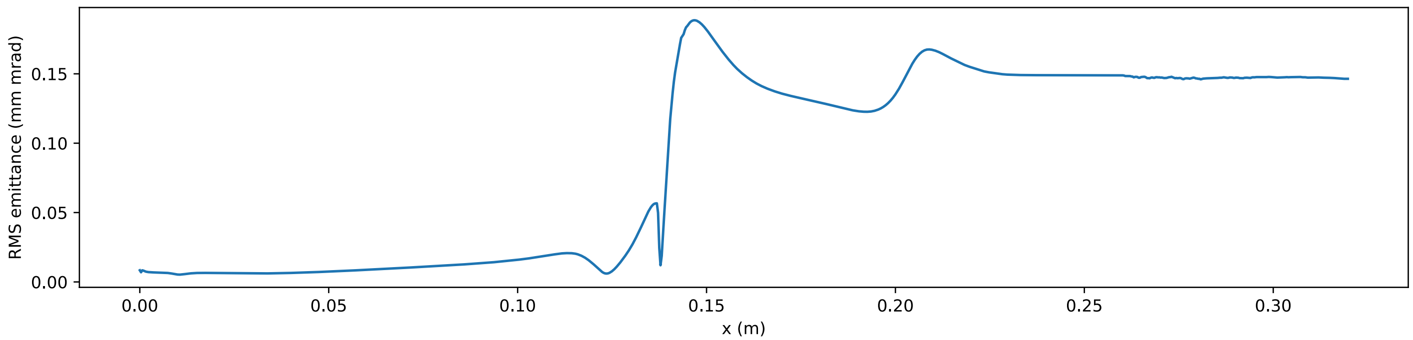


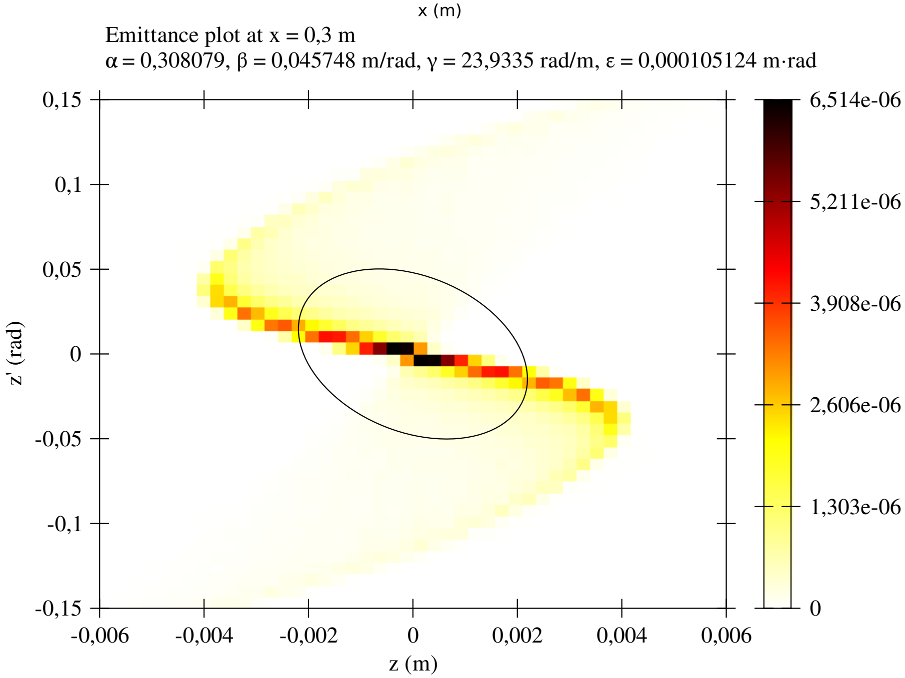


See `EmittanceConv` for conversion from cylindrical coordinates to Cartesian (z, z') emittance.

Normalized (r, r') rms emittance evolution



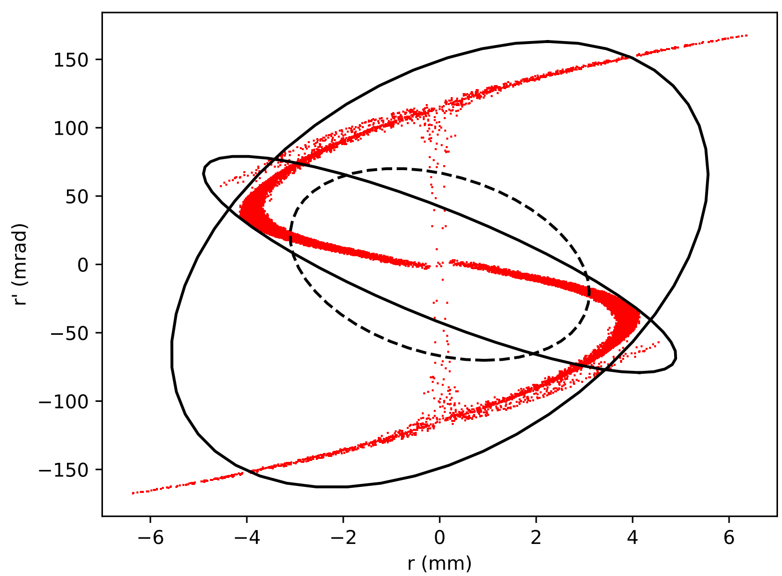


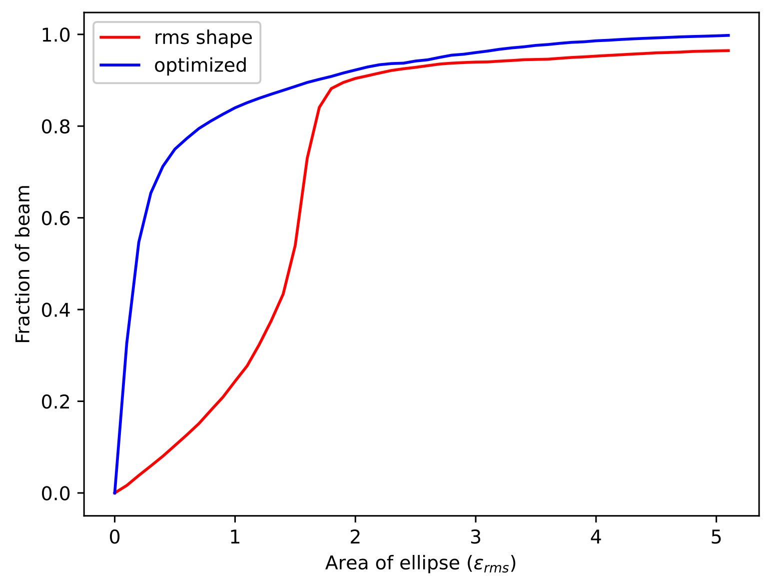


Relevance of pure rms emittance

Codes plot rms emittance because it is easy to calculate. In the real world it might or might not be relevant.

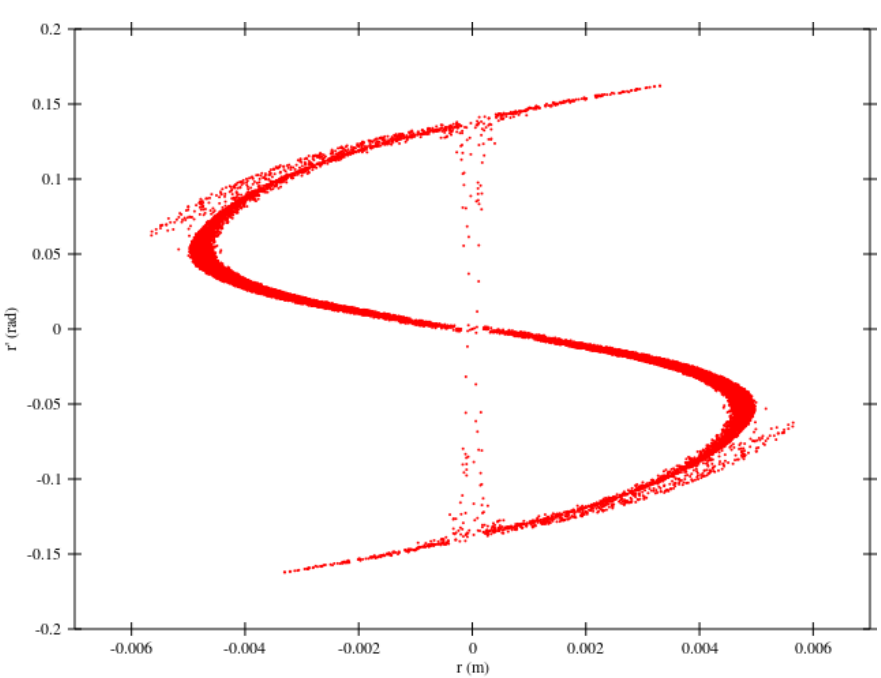
External tools, such as `emittance_fractions.py` exist to produce other type of numbers, e.g. 0.61 mA (84 %) within 207 mm mrad.

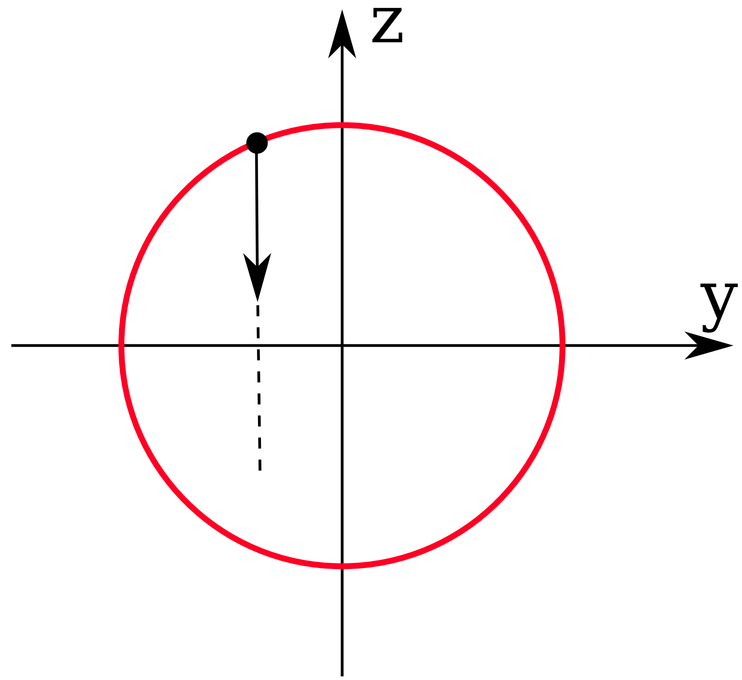




Peculiarities of cylindrical coordinate system

Particles have coordinates $(t, x, v_x, r, v_r, \omega)$. Rotation not visible in (r, r') plot. Phase space plot may easily by judged to have a simulation artefact, but it does not.

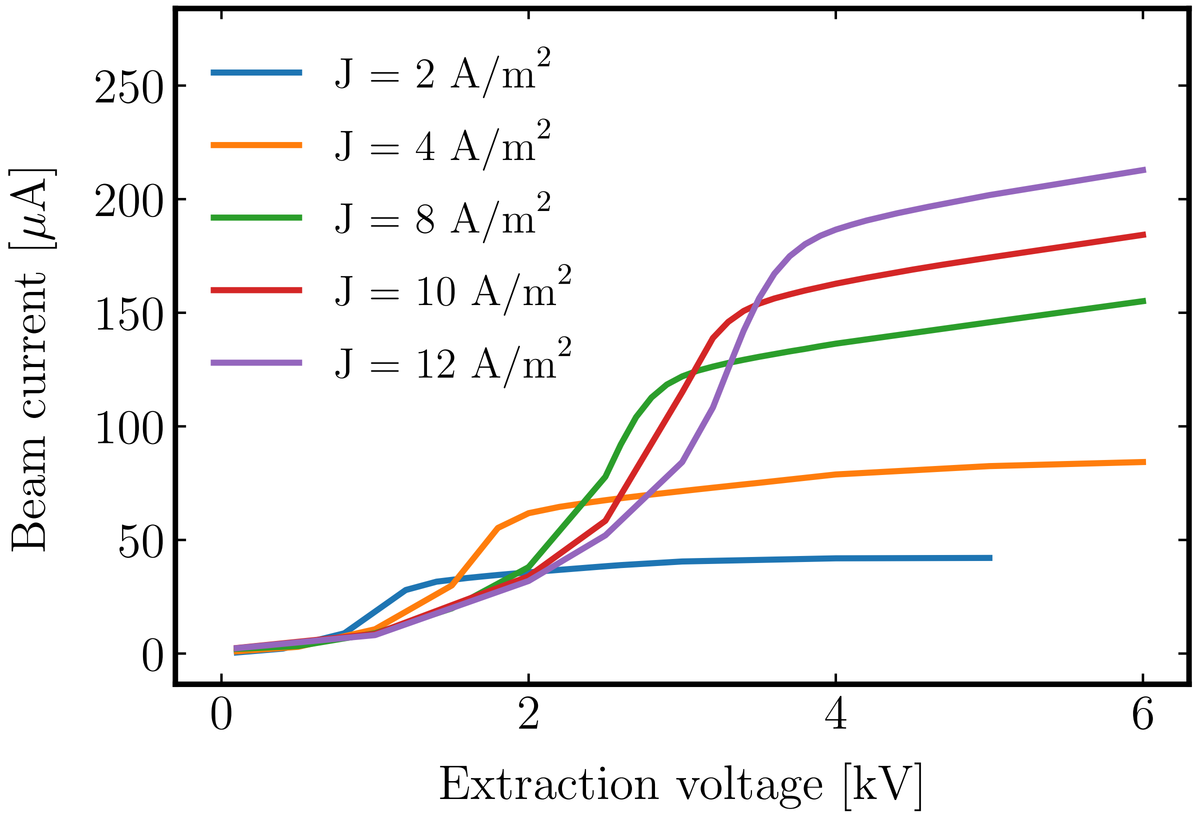




Having no rotation will cause issues! Avoid zero ion temperatures!

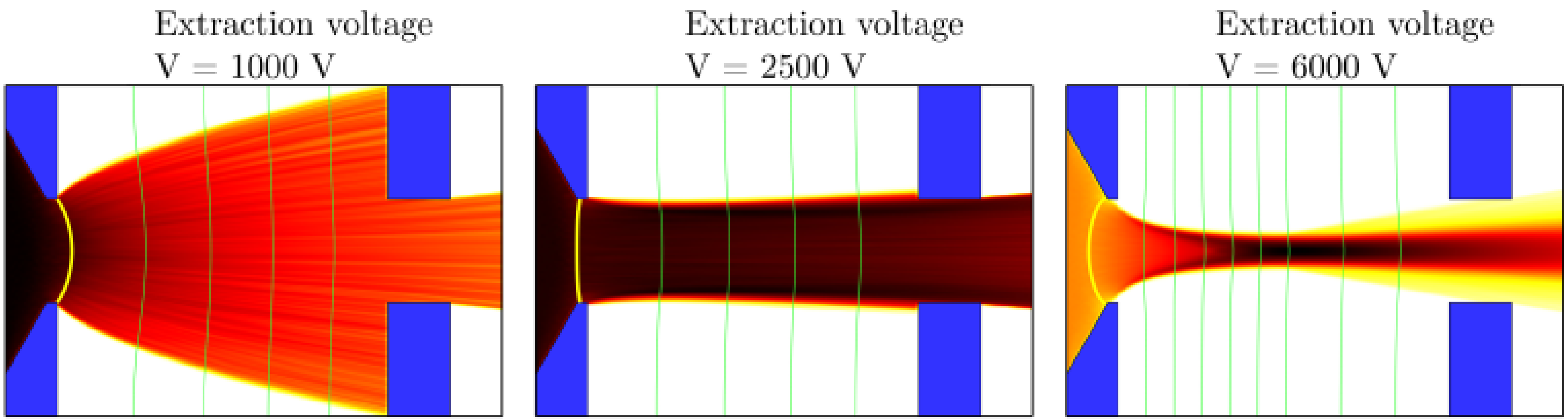
A closer look at the plasma model

Full operating regime can be modelled varying voltage and emitted current.



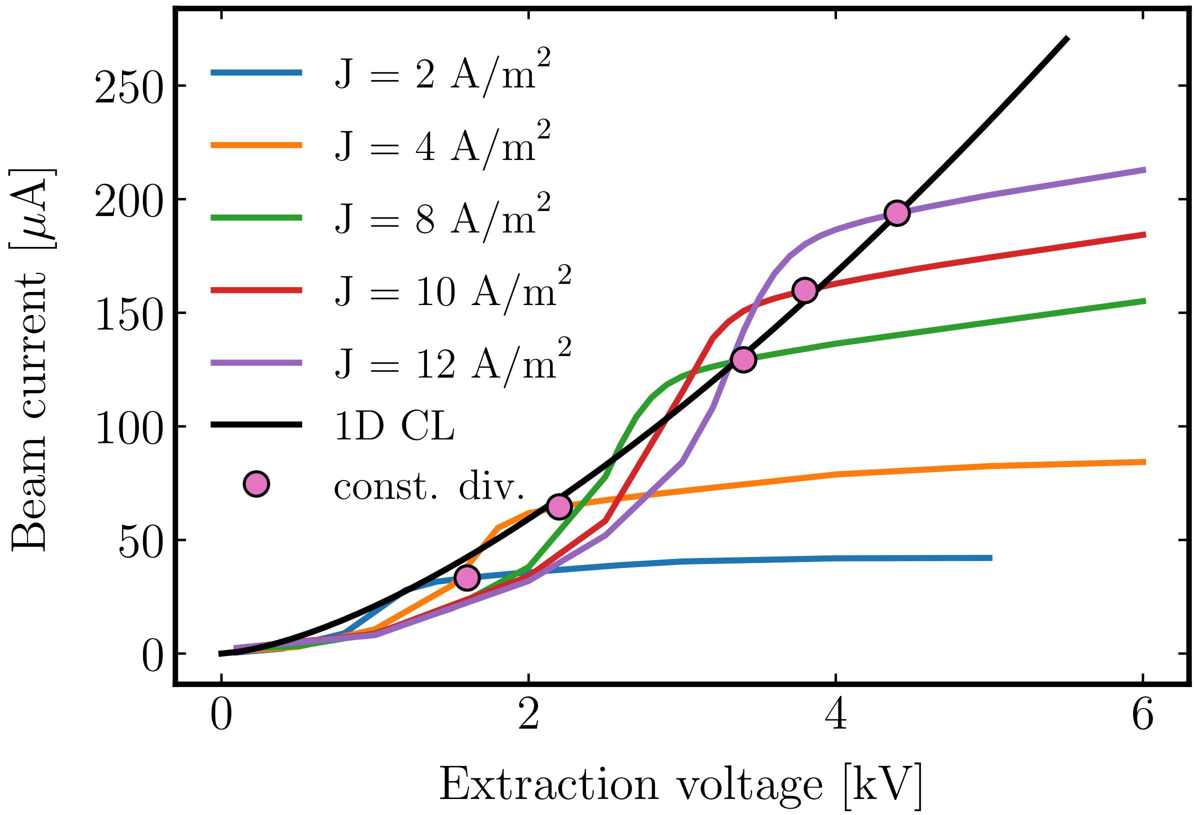
A closer look at the plasma model

What is happening?



A closer look at the plasma model

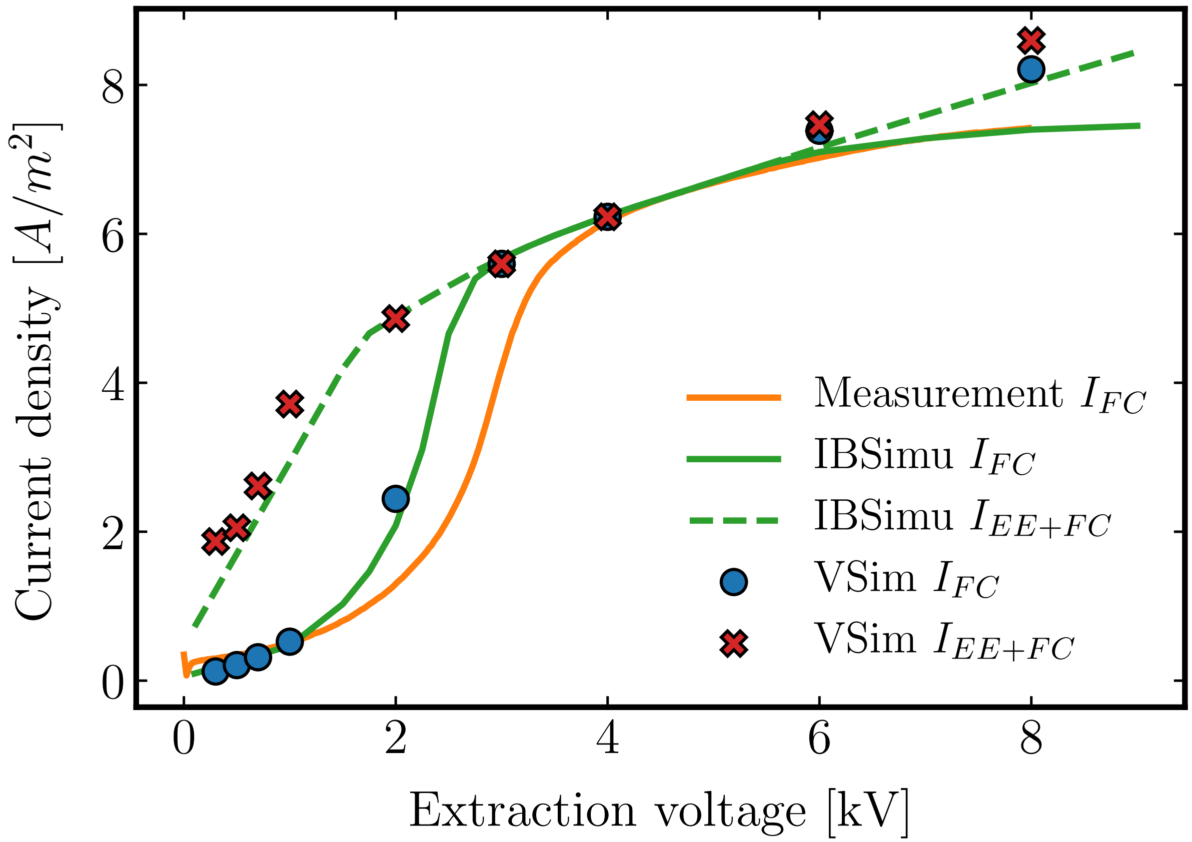
Full operating regime can be modelled varying voltage and emitted current.



From S. Kosonen, et al., *"Critical assessment of the applicability of the Child-Langmuir law to plasma ion source extraction systems"*, Plasma Sources Sci. Tech. 32, 075005 (2023).

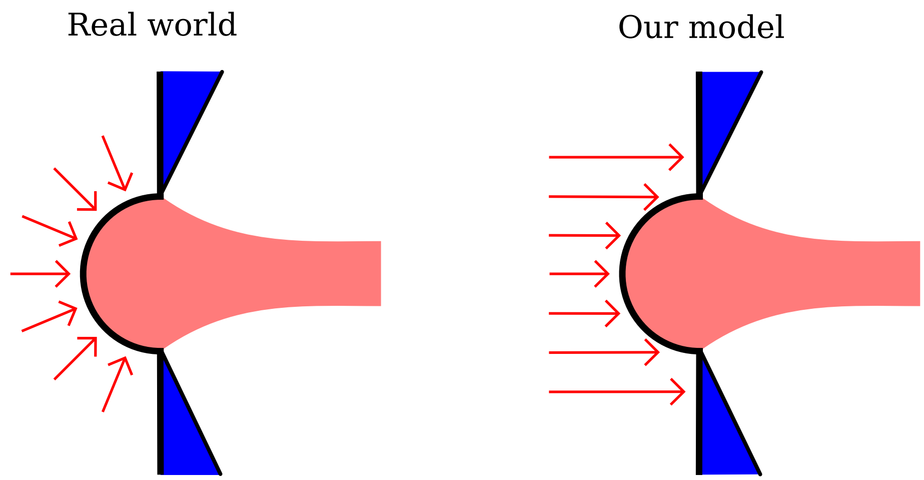
A closer look at the plasma model

Comparison to Vsim (PIC-code) and experiment

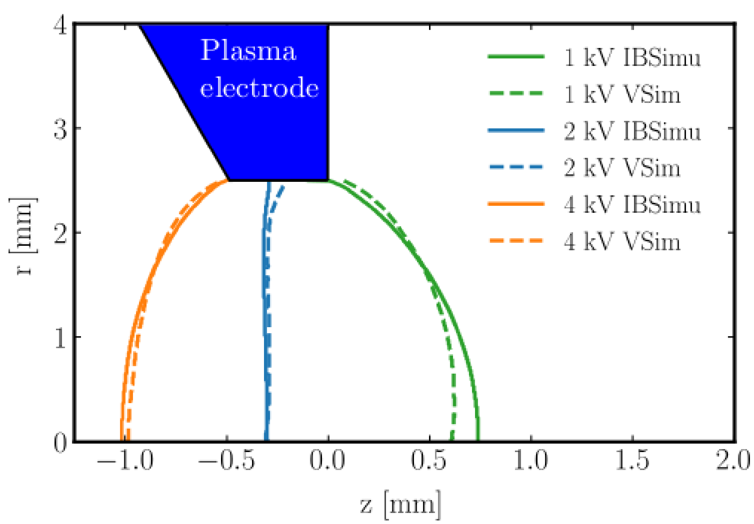


A closer look at the plasma model

Bohm sheath model is 1D, but now we use the same idea in higher dimensions with geometry. Our particles start at the sheath boundary, which is flat!



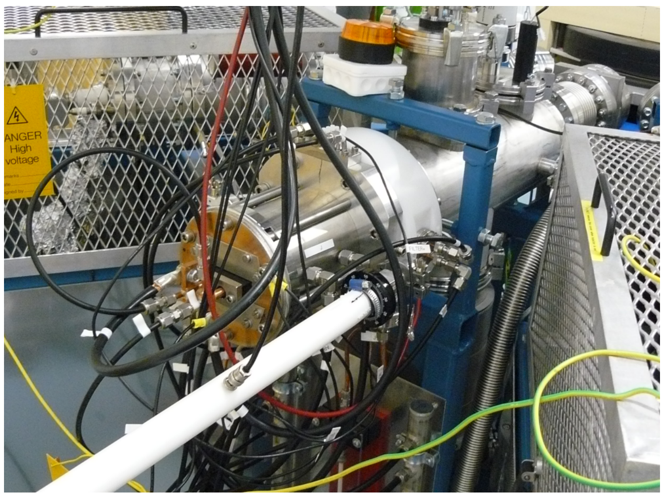
Error is emphasized at highly curved plasma meniscus shapes. Please note that this effect is mitigated by transverse ion temperature. Comparison to VSim, which includes pre-sheath:

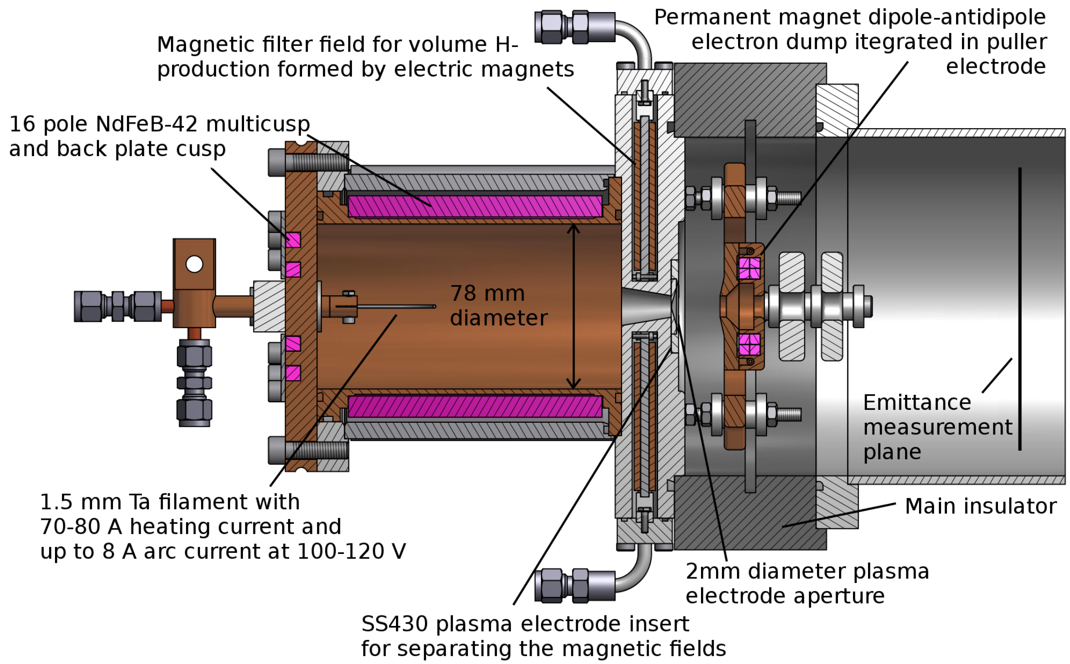


JYFL “Pellis”

JYFL "Pellis"

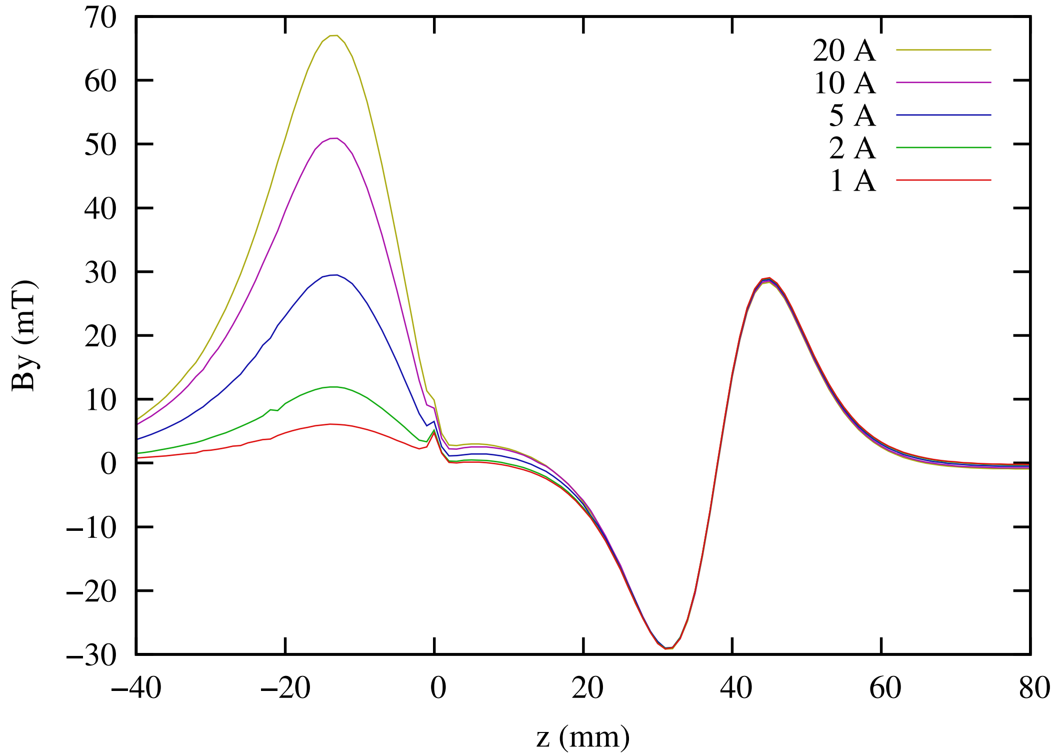
Pelletron light ion source, filament driven ion source for high brightness H^- beam at 10 keV, < 100 μA.

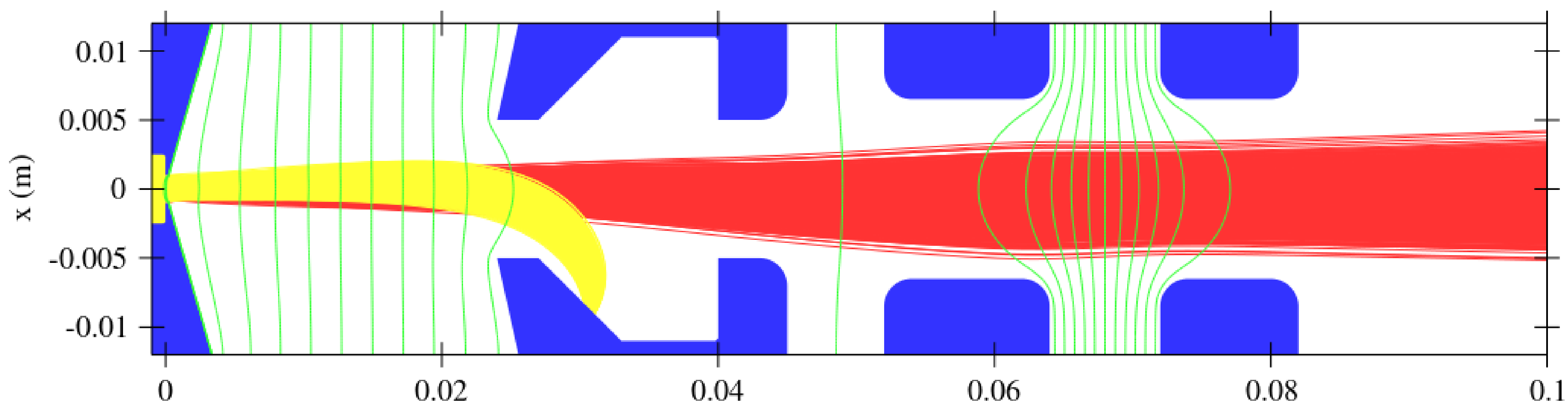




T. Kalvas, *"Development and use of computational tools for modelling negative hydrogen ion source extraction systems"*, Ph.D. Thesis, University of Jyväskylä, 2013.

Adjustable filter field

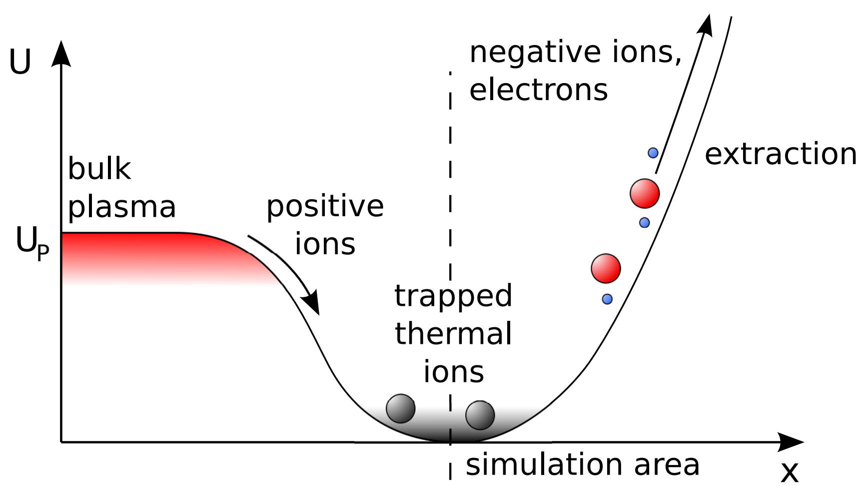




Negative ion plasma extraction model

Modelling of negative ion extraction

- Ray-traced negative ions and electrons
- Analytic thermal and fast positive charges
- $U = 0$ equipotential assumed at extraction
- Magnetic field suppression for electrons inside plasma



$$\rho_{\mathrm{th}} = \rho_{\mathrm{th0}} \exp\left(\frac{-eU}{kT_i}\right)$$

$$\rho_{\mathrm{f}} = \rho_{\mathrm{f0}} \left(1 - \mathrm{erf}\left(\frac{U}{U_p}\right)\right)$$

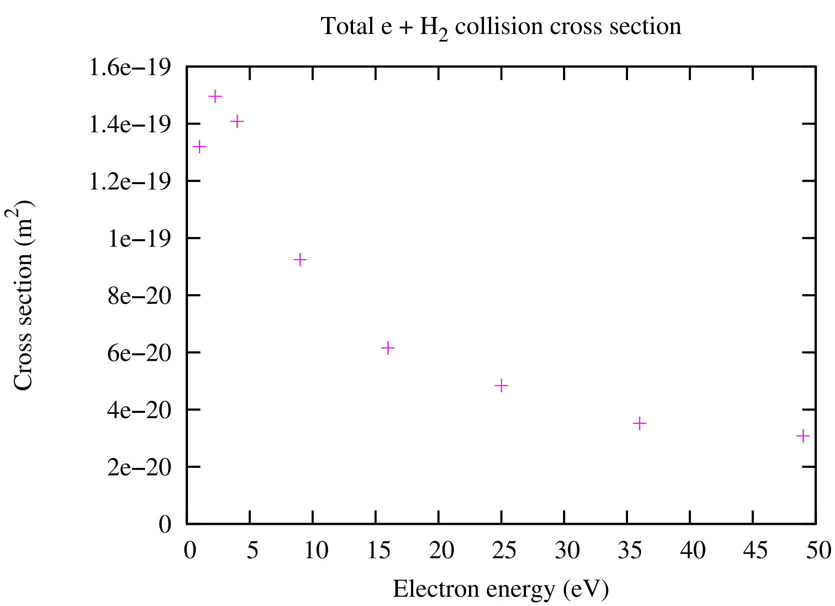
R. Becker, Rev. Sci. Instrum. 75, 1723 (2004) and

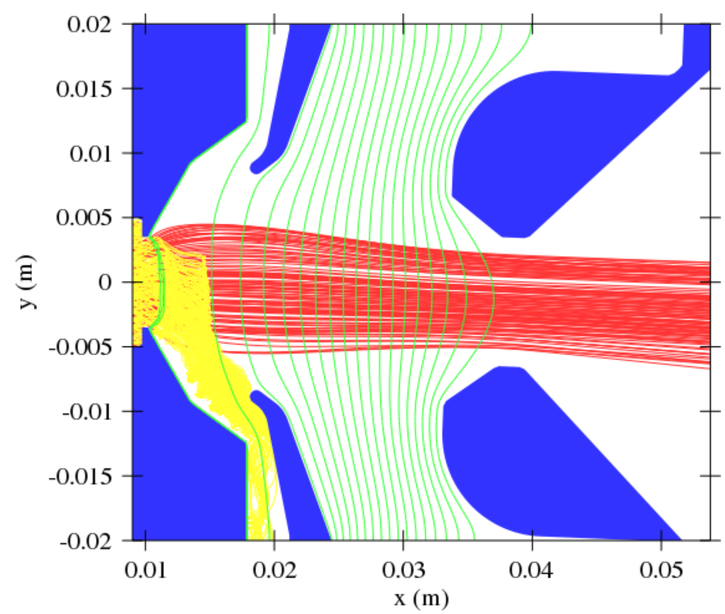
R. Becker, AIP Conf. Proc. 763, 194 (2005).

Negative ion plasma extraction model

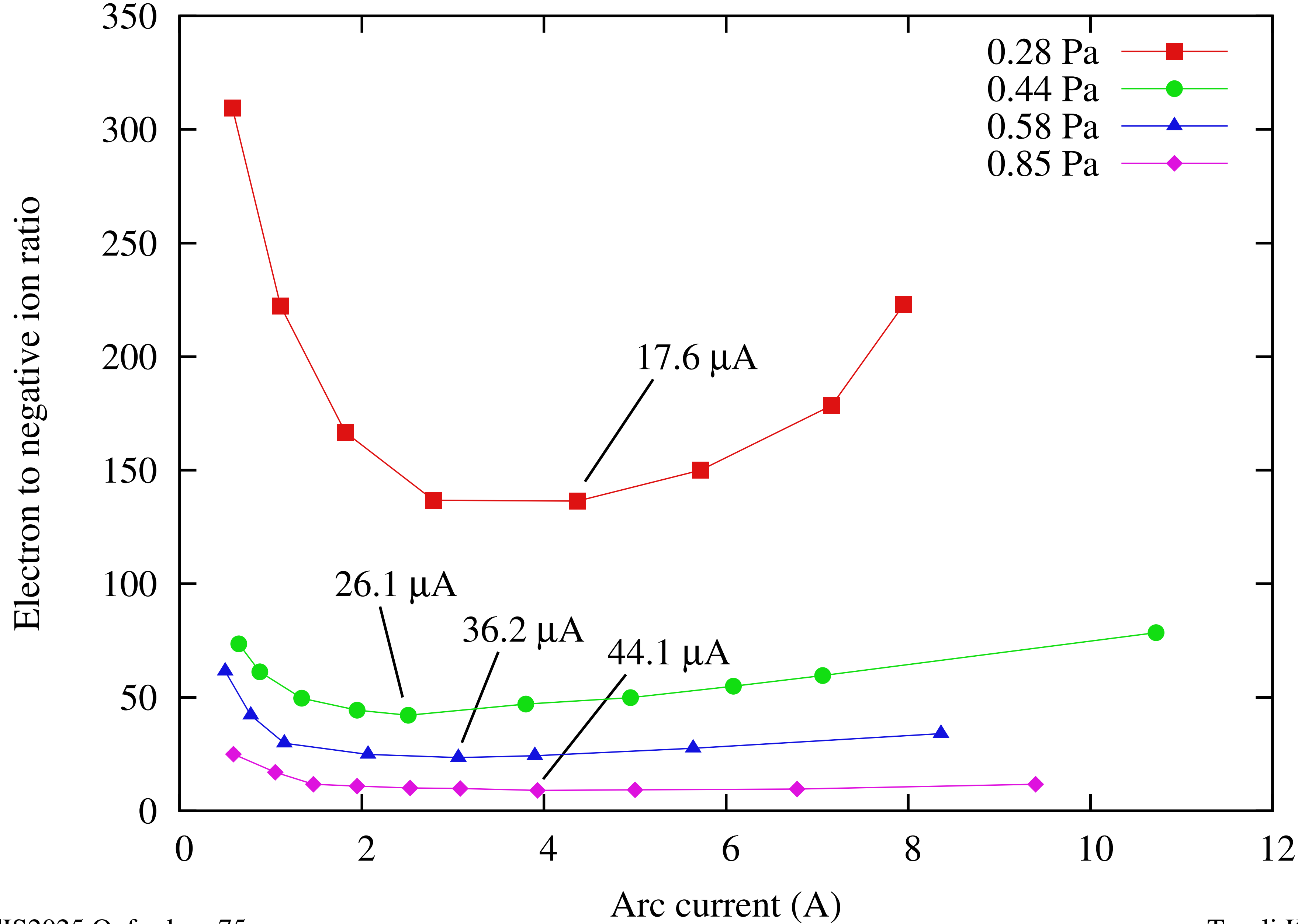
Magnetic field suppression for electrons inside plasma

- Electrons highly collisional until velocity large enough
- Ray-tracing simulations not capable of simulating diffusion
- Magnetic field does not dictate electron trajectories in plasma
  $\rightarrow$ B-field suppression is a sufficient approximation

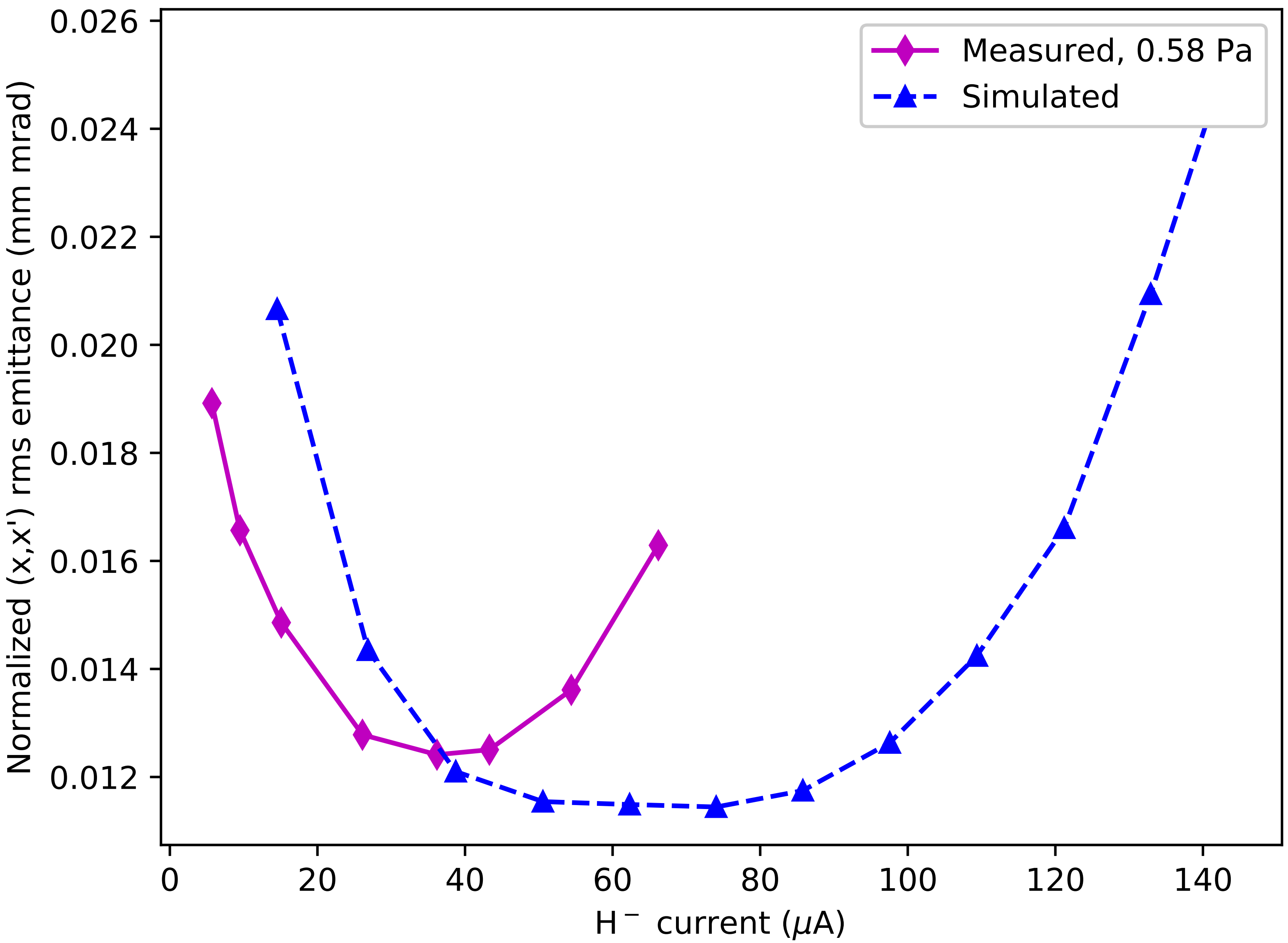




Effect of pressure (measurement)



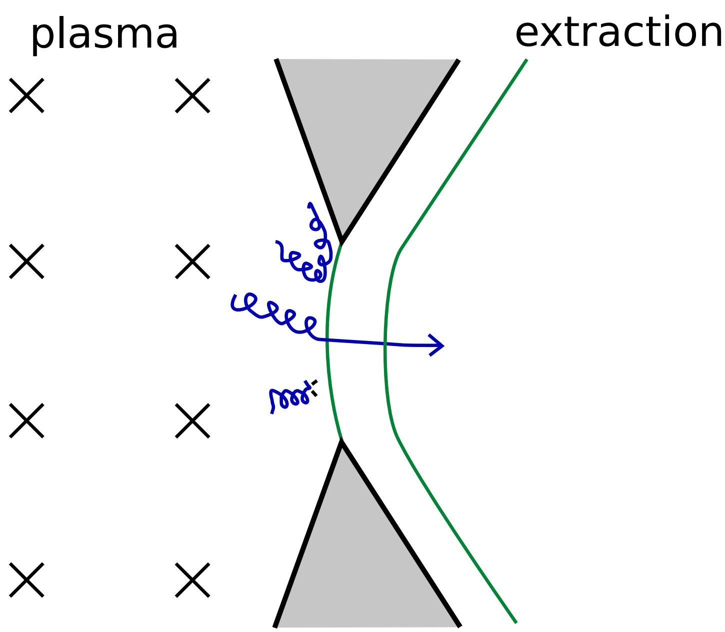
Emittance vs. H^- beam current



Plasma model failure?

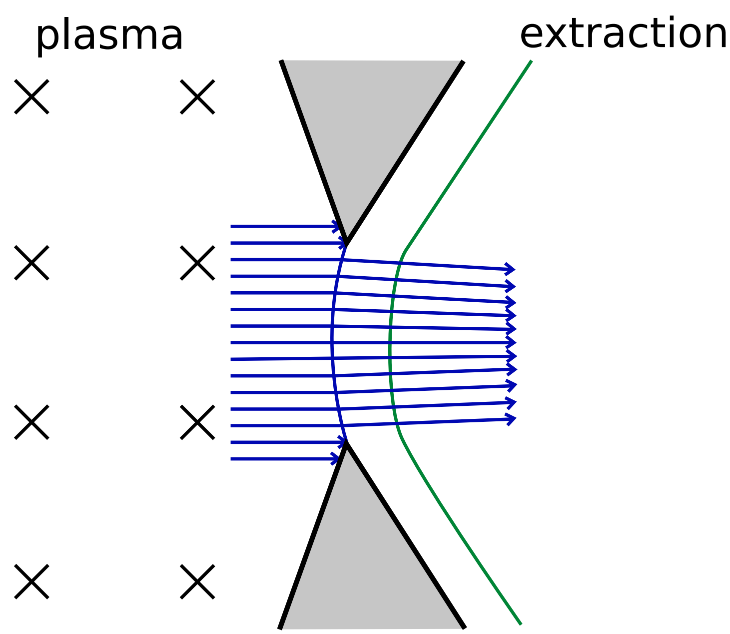
Modelling of ions ok

For electrons, physical processes are not properly modelled



Reality

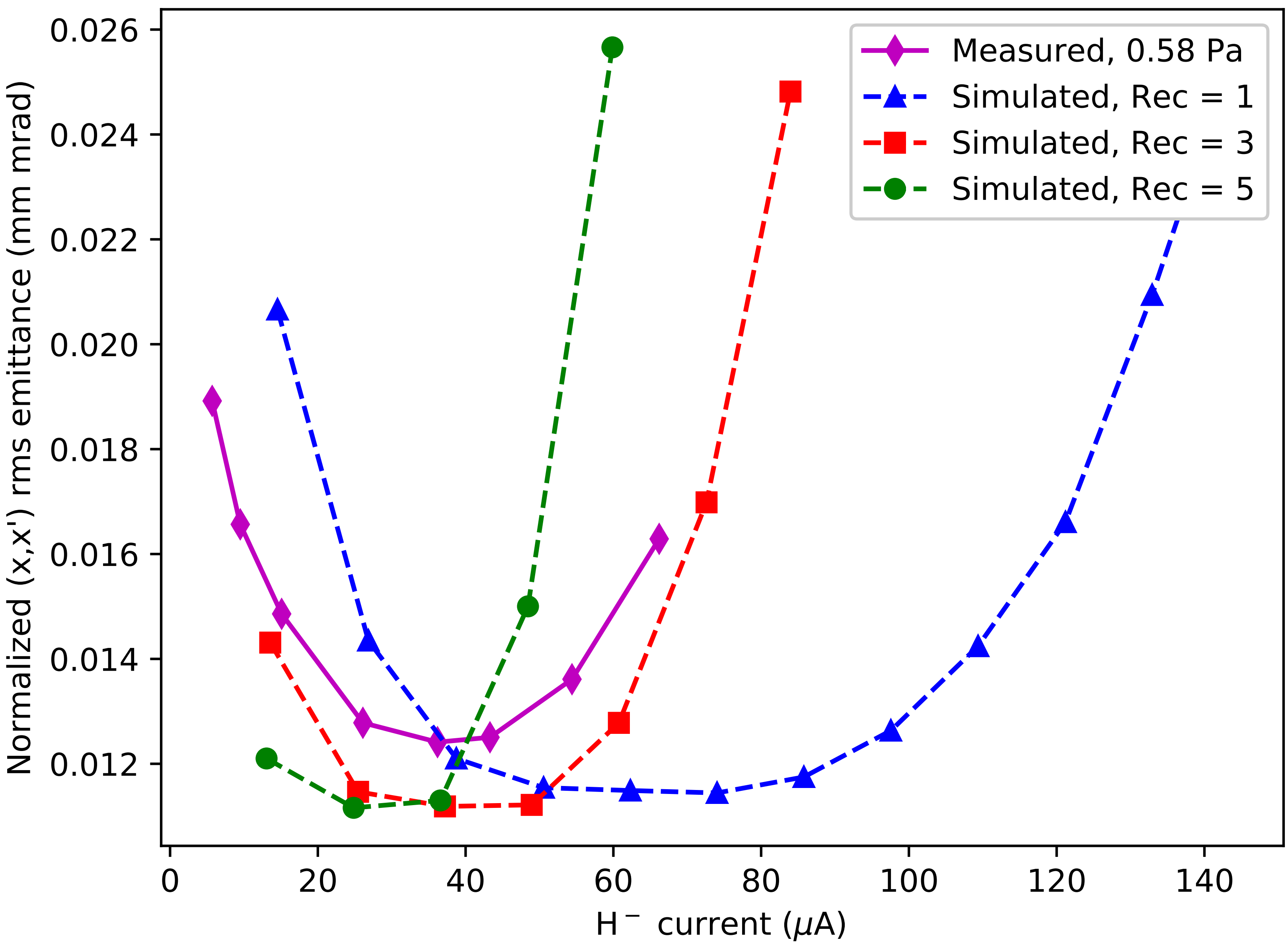
$$\rho_e \neq \frac{J_e}{v_e}$$



Model

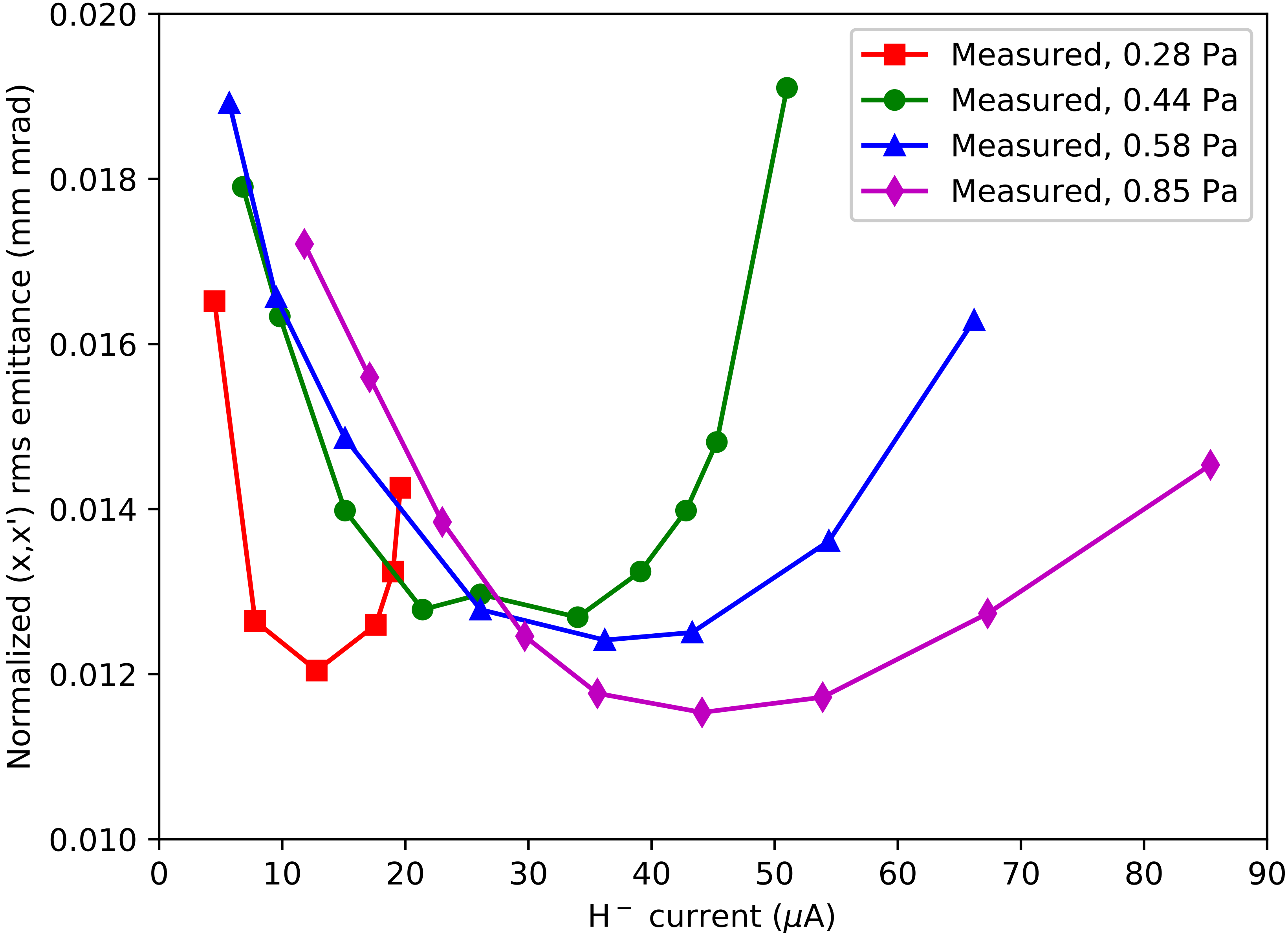
$$\rho_e = R_{\mathrm{ec}} \frac{J_e}{v_e} \quad \text{at} \quad \phi < 2\phi_P$$

Emittance vs. H^- beam current



 Taneli Kalvas

Emittance vs. H$^-$ beam current



H^- equivalent current

Meniscus dependent on charge density – not particle type.

$$\rho = \frac{J}{v}$$

and

$$v = \sqrt{\frac{2q\phi}{m}},$$

neglecting initial velocity at zero potential. Total charge

$$\rho_{\mathrm{tot}} = \rho_{H^-} + \rho_e = \frac{J_{H^-}}{v_{H^-}} + \frac{J_e}{v_e}.$$

Assuming

$$J_e = R_{ec} J_e^* = R_{ec} R_{ei} J_{H^-},$$

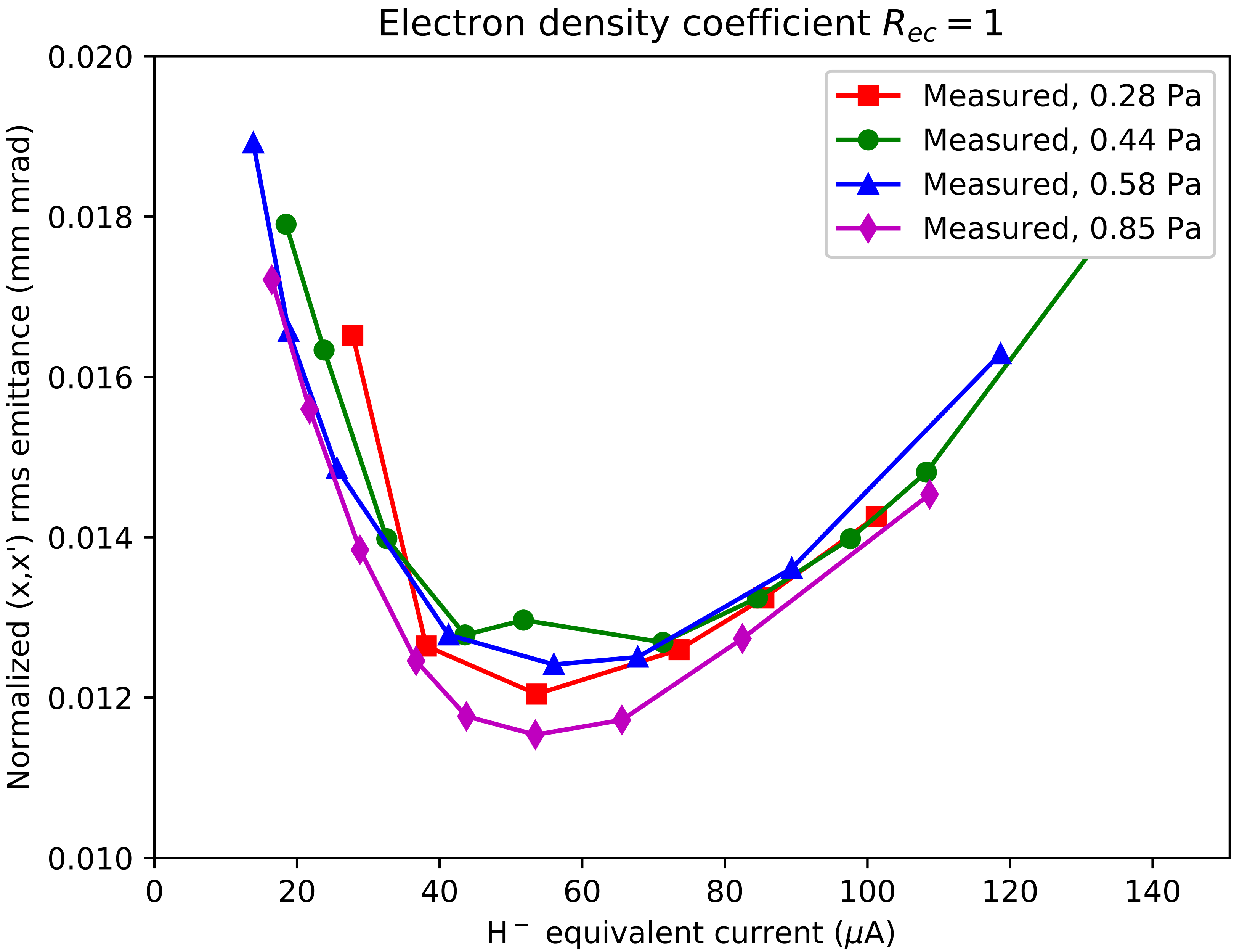
thus

$$\rho_{\mathrm{tot}} \propto J_{H^-}(1 + R_{ec} R_{ei} \sqrt{m_e/m_{H^-}})$$

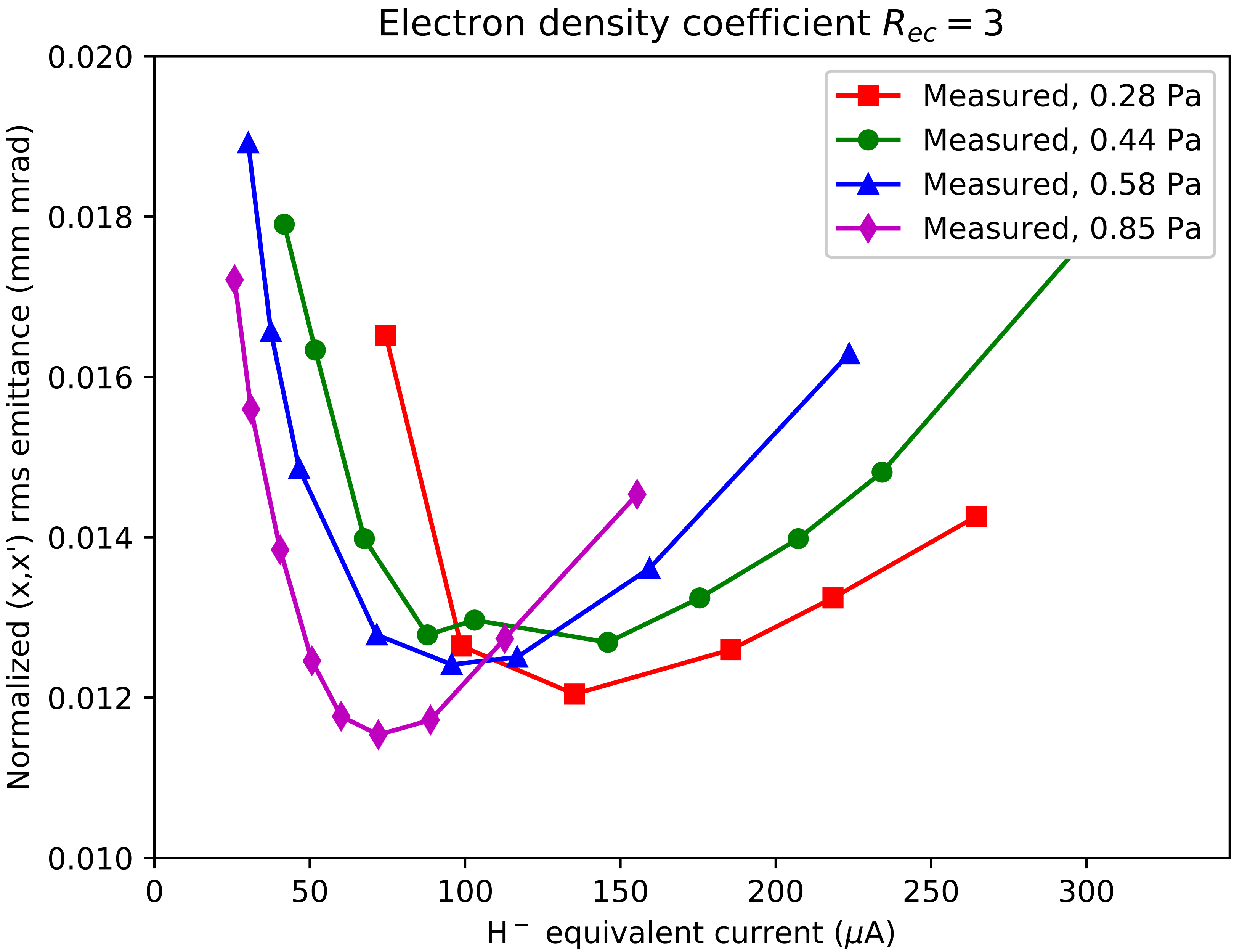
and

$$I_{\mathrm{eq}} = I_{H^-} + R_{ec} I_e^* \sqrt{m_e/m_{H^-}}.$$

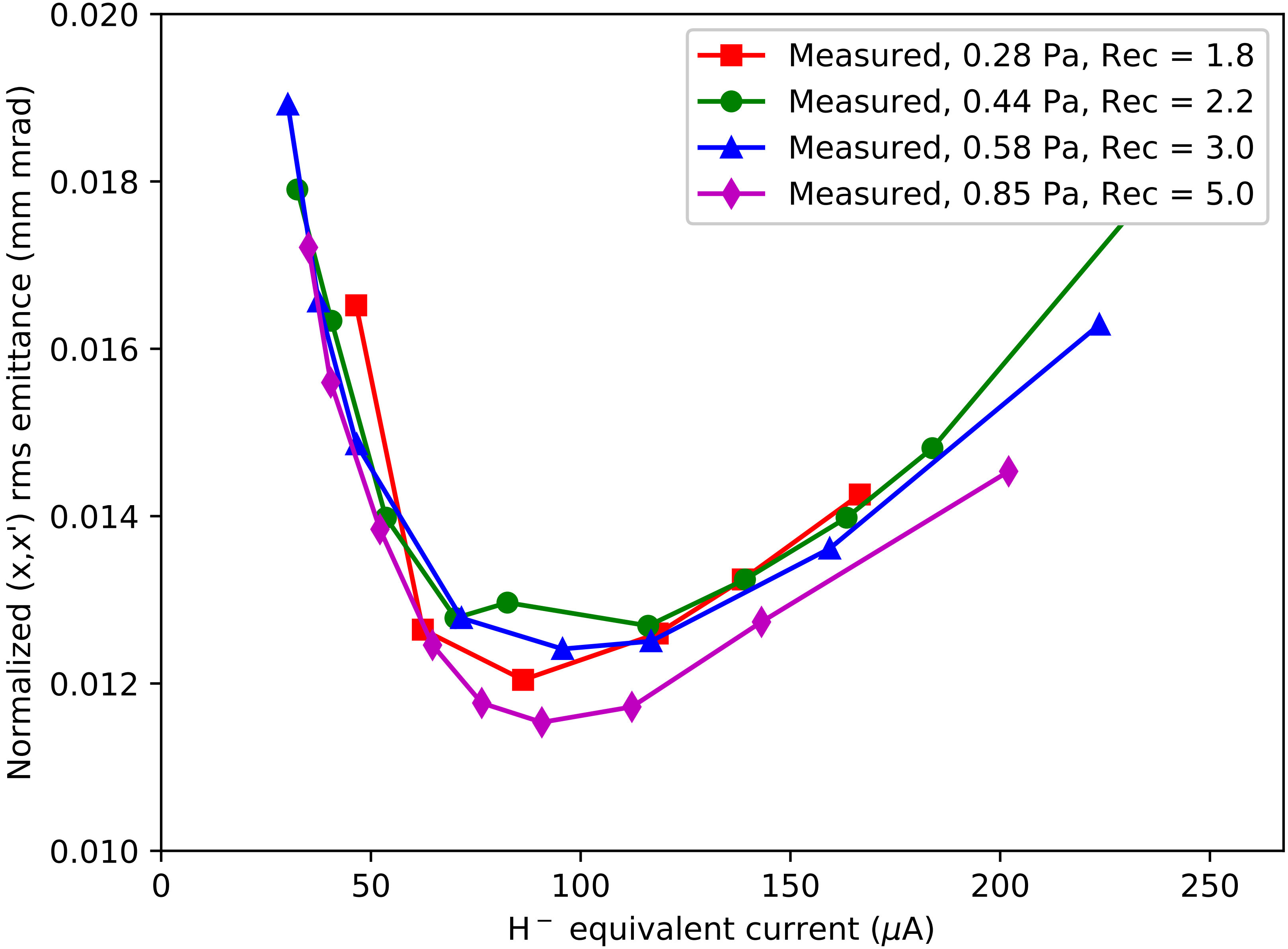
Emittance vs. H^- beam current



Emittance vs. H^- beam current



Emittance vs. H^- beam current



Total space charge effect

Previously we had equivalent current

$$I_{\mathrm{eq}} = I_{H^-} + R_{ec} I_e^* \sqrt{m_e/m_{H^-}}.$$

If the meniscus is only dependent on charge, we can also describe the system using a total space charge ratio

$$R_{\mathrm{tot}} = \frac{\rho_{\mathrm{tot}}}{\rho_{\mathrm{tot}}^*} = \frac{J_{H^-}(1 + R_{ec} R_{ei} \sqrt{m_e/m_{H^-}})}{J_{H^-}(1 + R_{ei} \sqrt{m_e/m_{H^-}})}.$$

For the 0.58 Pa case R_{ei} 20 and if $R_{ec} = 3$, we get $R_{\mathrm{tot}} = 1.6$.

Alternatively we can account the error to be due to ion charge, i.e.

$$J_{H^-}(1 + R_{ec} R_{ei} \sqrt{m_e/m_{H^-}}) = J_{H^-}(R_{ic} + R_{ei} \sqrt{m_e/m_{H^-}})$$

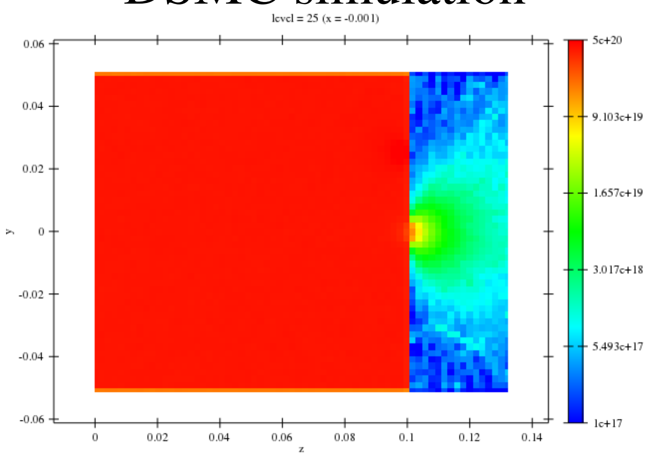
giving $R_{ic} = 1 + R_{ei} \sqrt{m_e/m_{H^-}}(R_{ec} - 1) = 1.93$.

Stripping loss?

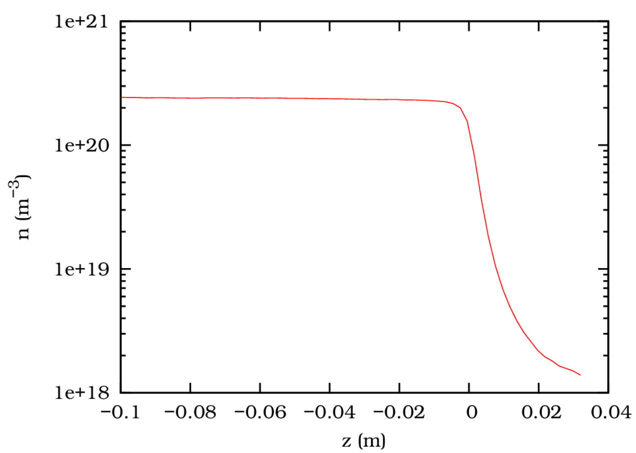
Can we account 1.93 for stripping?

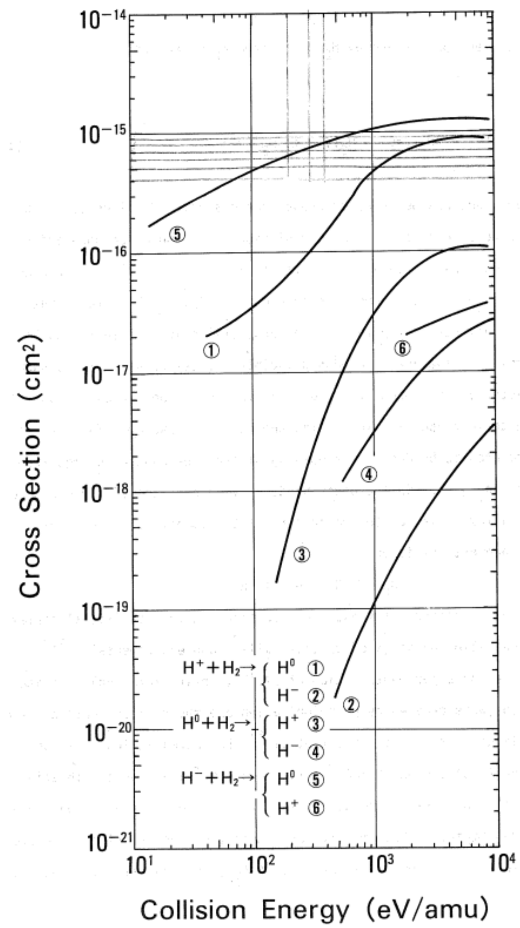
H. Tawara, "Atomic data involving hydrogens relevant to edge plasmas", IPPJ-AM-46, 1986

DSMC simulation



Particle density on axis





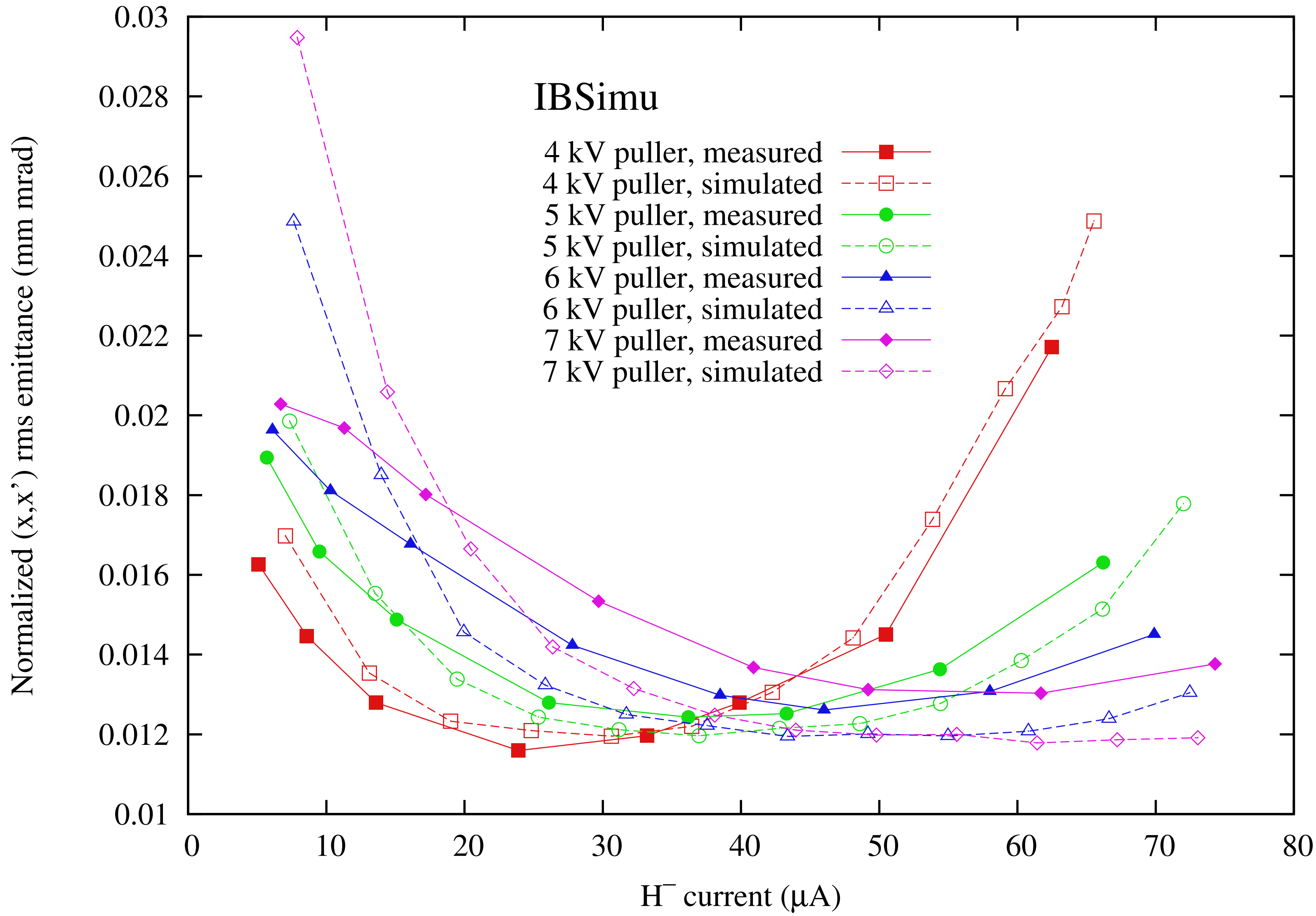
$P = 0.5 - 0$ Pa

$z = 10$ mm

$I/I* = \exp(n\sigma z) = 1.13$

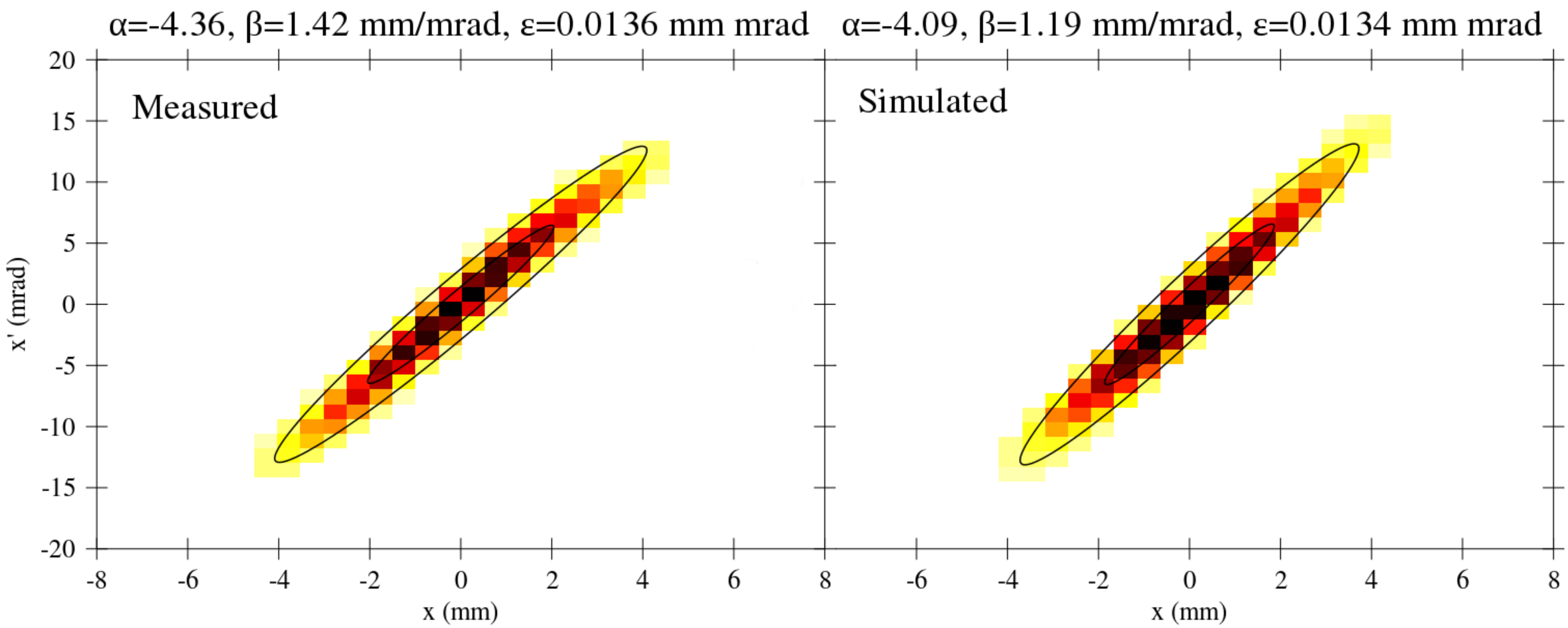
Emittance vs. beam current

$R_{ec} = 3$, $T_t = 0.75$ eV, $P = 0.58$ Pa



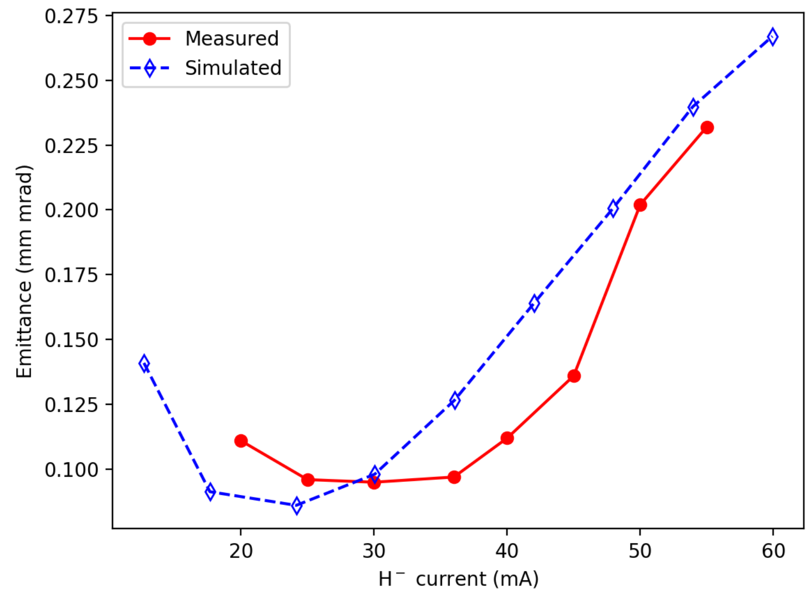
Phase space distribution

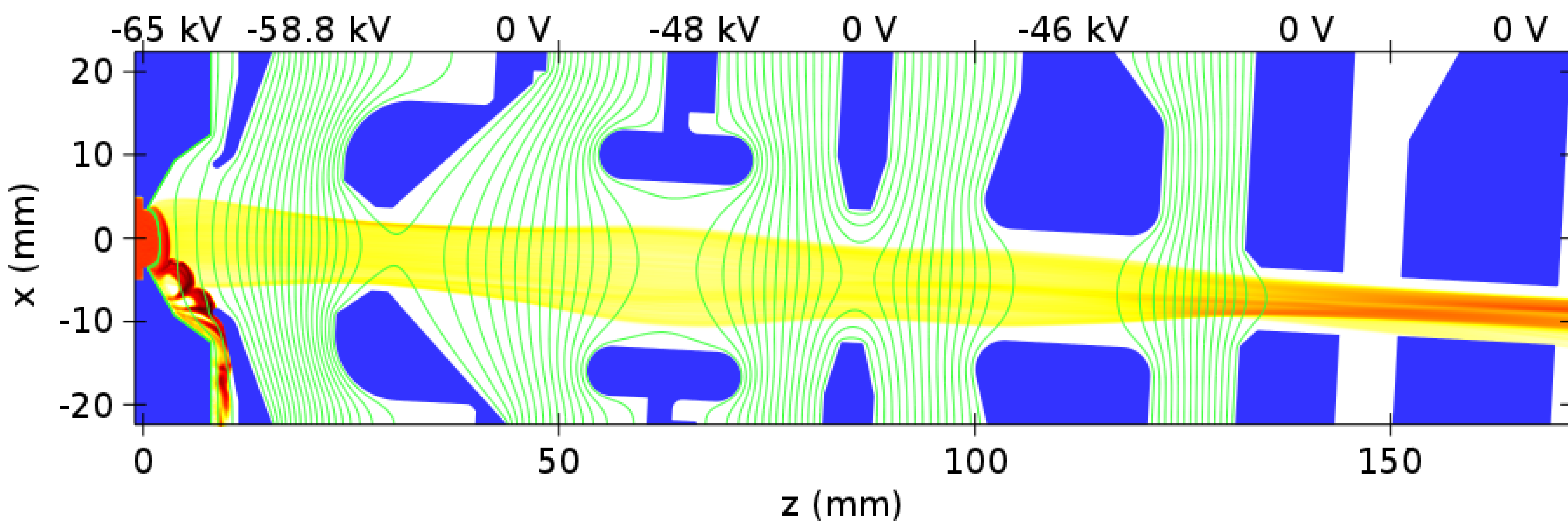
Measured and simulated (x, x') phase space patterns for 40 μA extracted H^- beam with 5 kV puller voltage.



Emittance comparison for SNS ion source

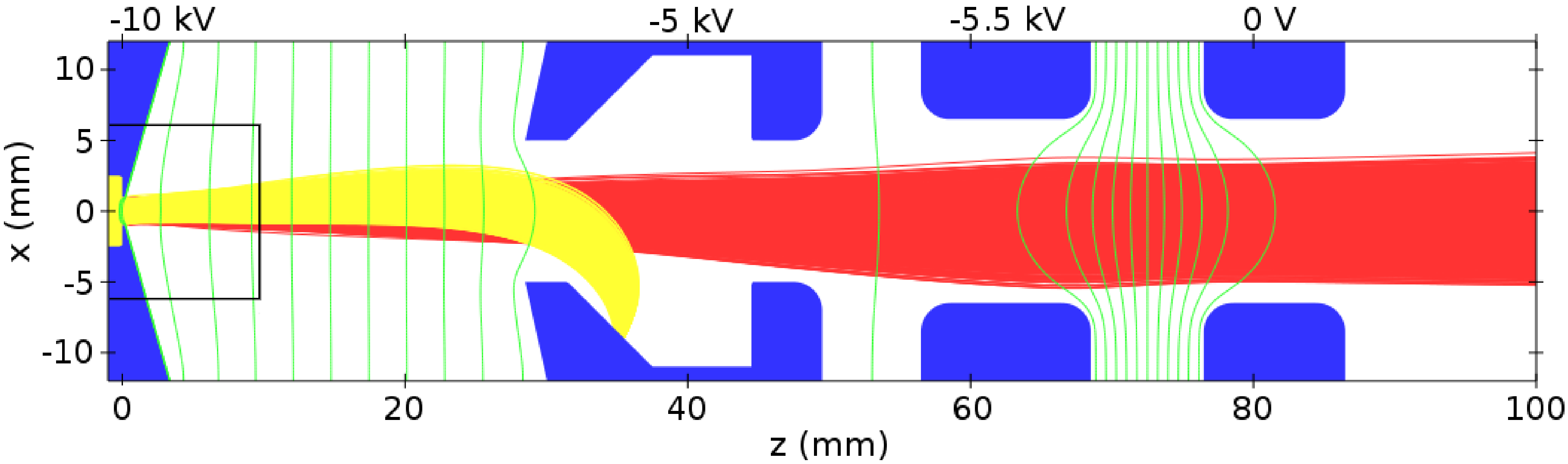
T. Kalvas, et al.,
Rev. Sci. Instrum. 83, 02A705 (2012).





Rough/Fine/Rough "trick"

A fine simulation region chosen, where higher resolution required.



1. Rough simulation calculated just like normally
2. Rough simulation provides boundary condition for fine simulation
3. Fine simulation calculated normally
4. Export potential and particles near the boundary
5. Calculate second rough simulation using fine potential and particles

Exporting particles

Kinetic energy depends on potential energy, $E = E_P + E_K$. Beam potential might be significant.

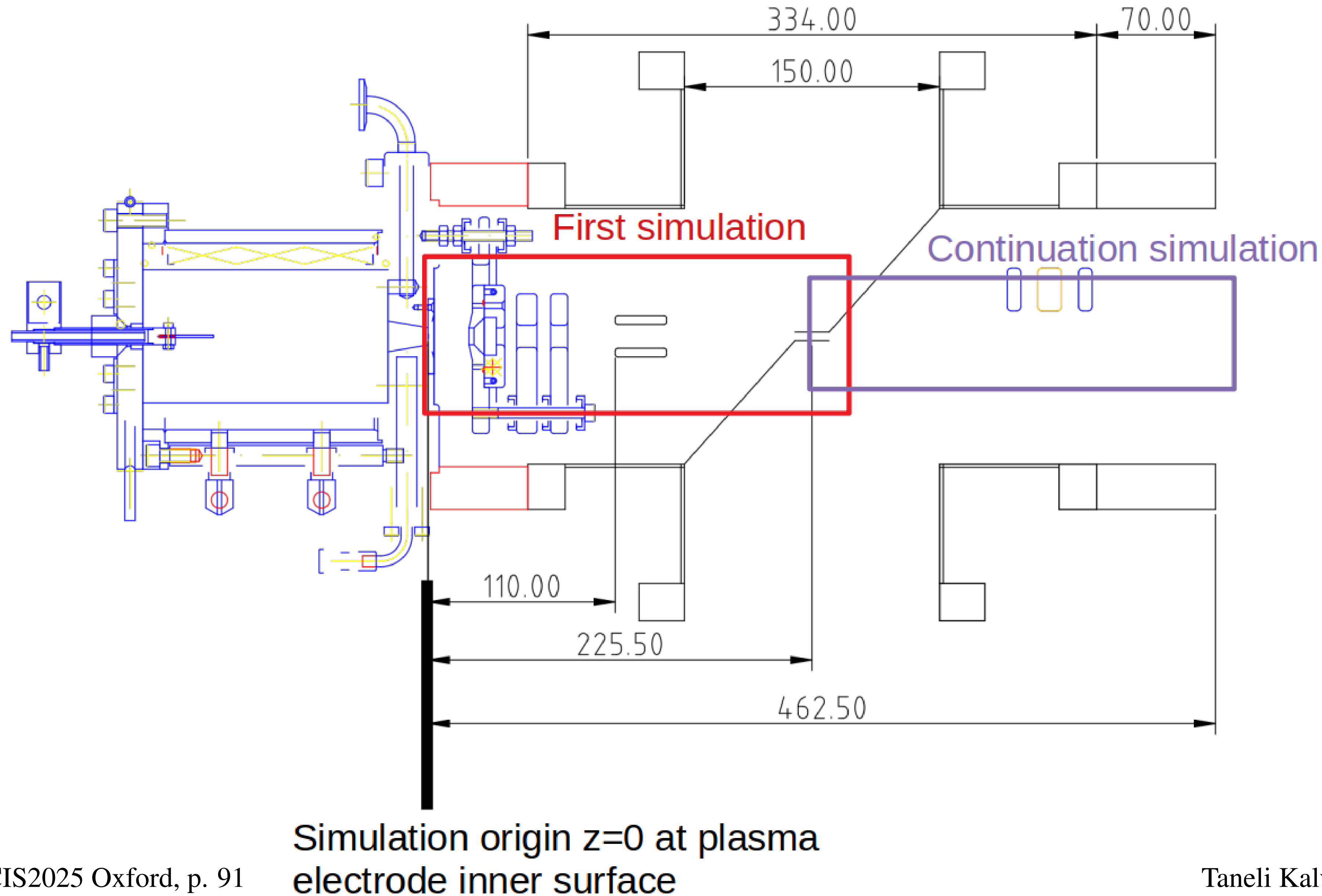
Exporting can be done

- By forcing export plane to constant potential (Dirichlet boundary).
- By exporting particle coordinates and potential distribution.
- By exporting phase space (x, x', y, y') and total energy E.

Note that Path manager export does not do any of these automatically.

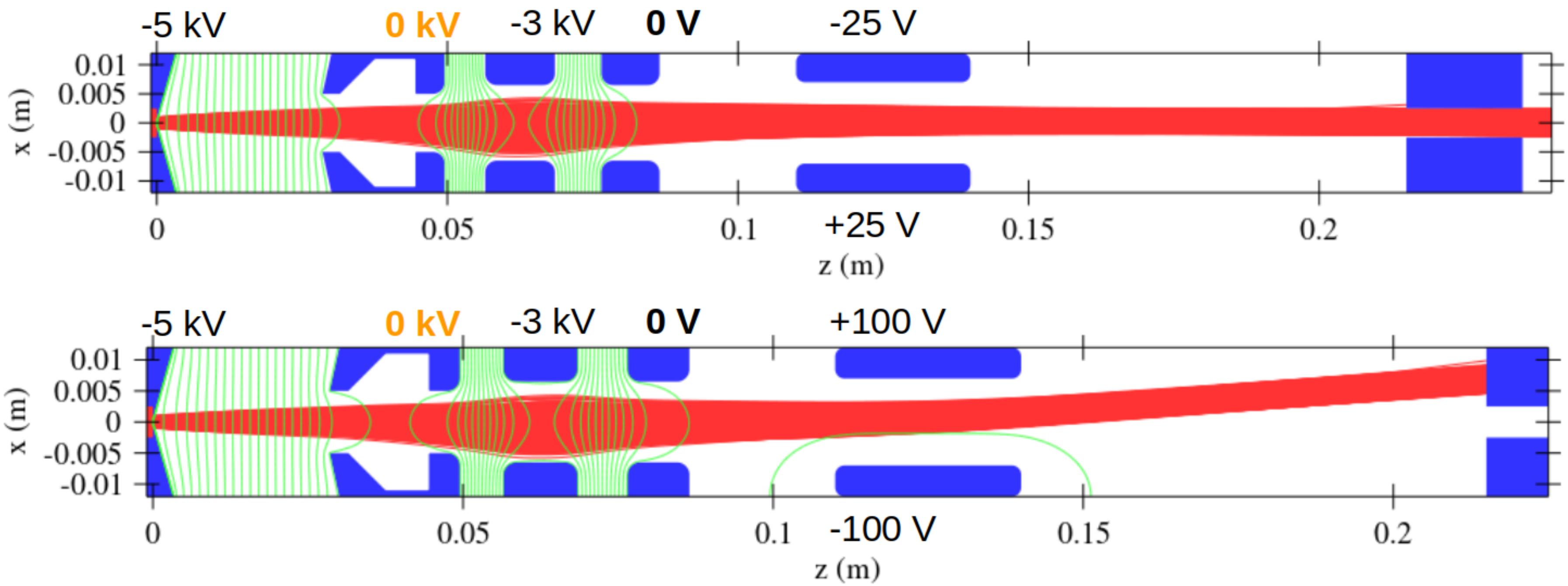
Continuation runs (CERN Alpha H^- source)

Similar considerations needed for continuation simulations



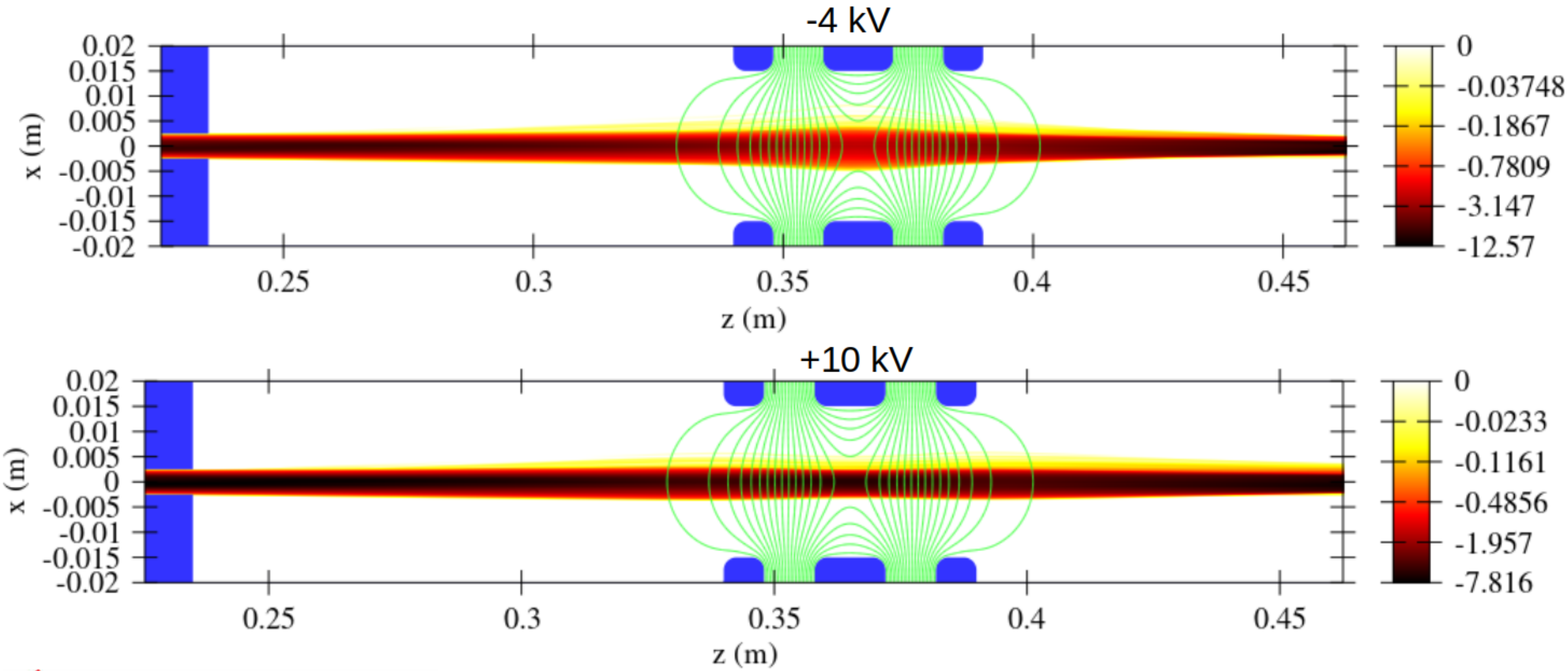
First simulation (CERN Alpha H^- source)

First part of simulation models beam formation, focusing electrodes and parallel plate sweeping.



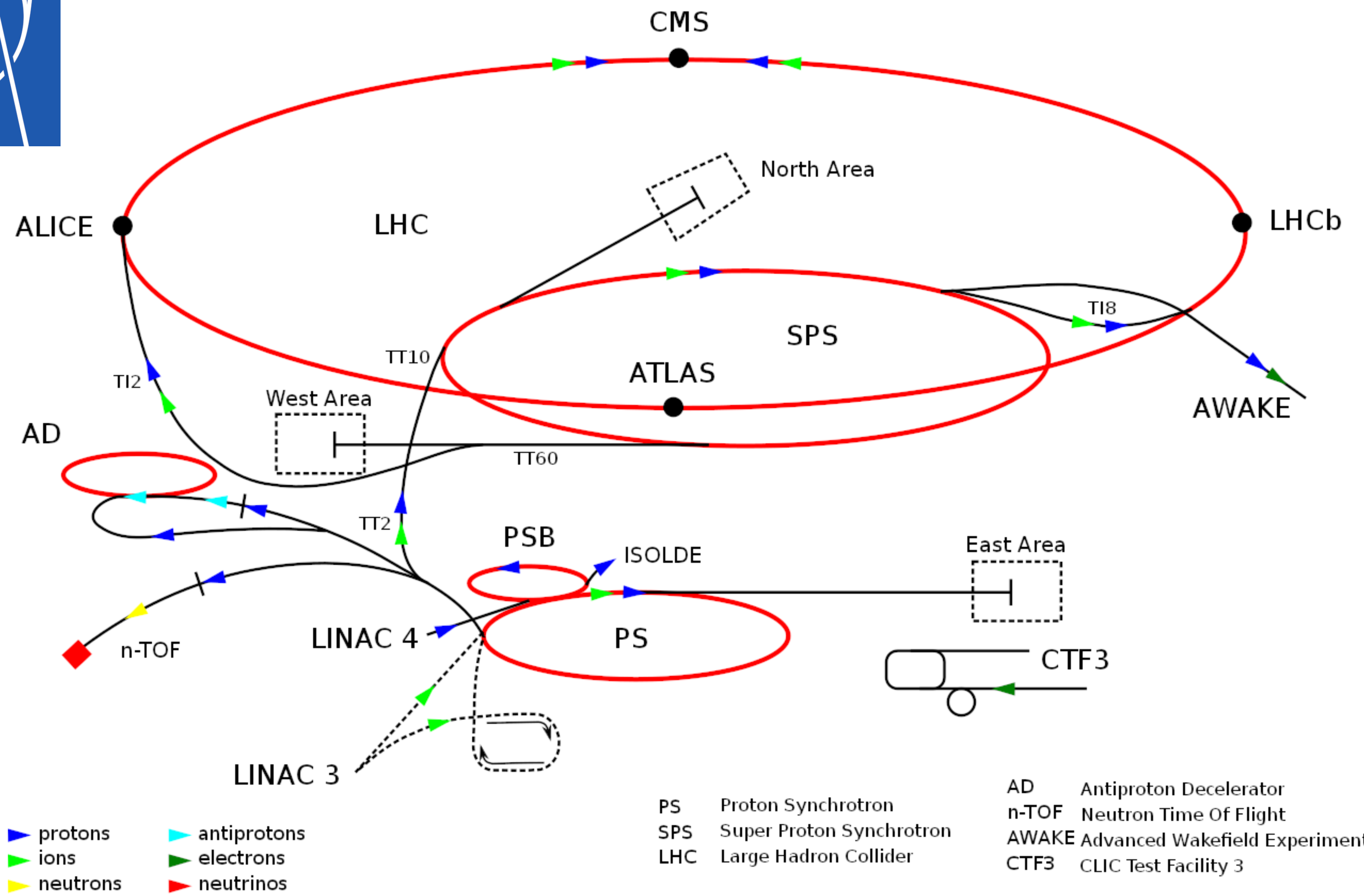
Continuation simulation (CERN Alpha H^- source)

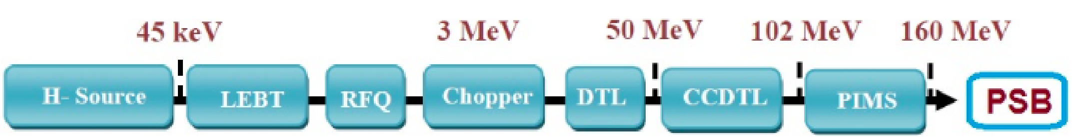
Continuation simulations models drift and an additional einzel lens.



Linac4 H^- ion source

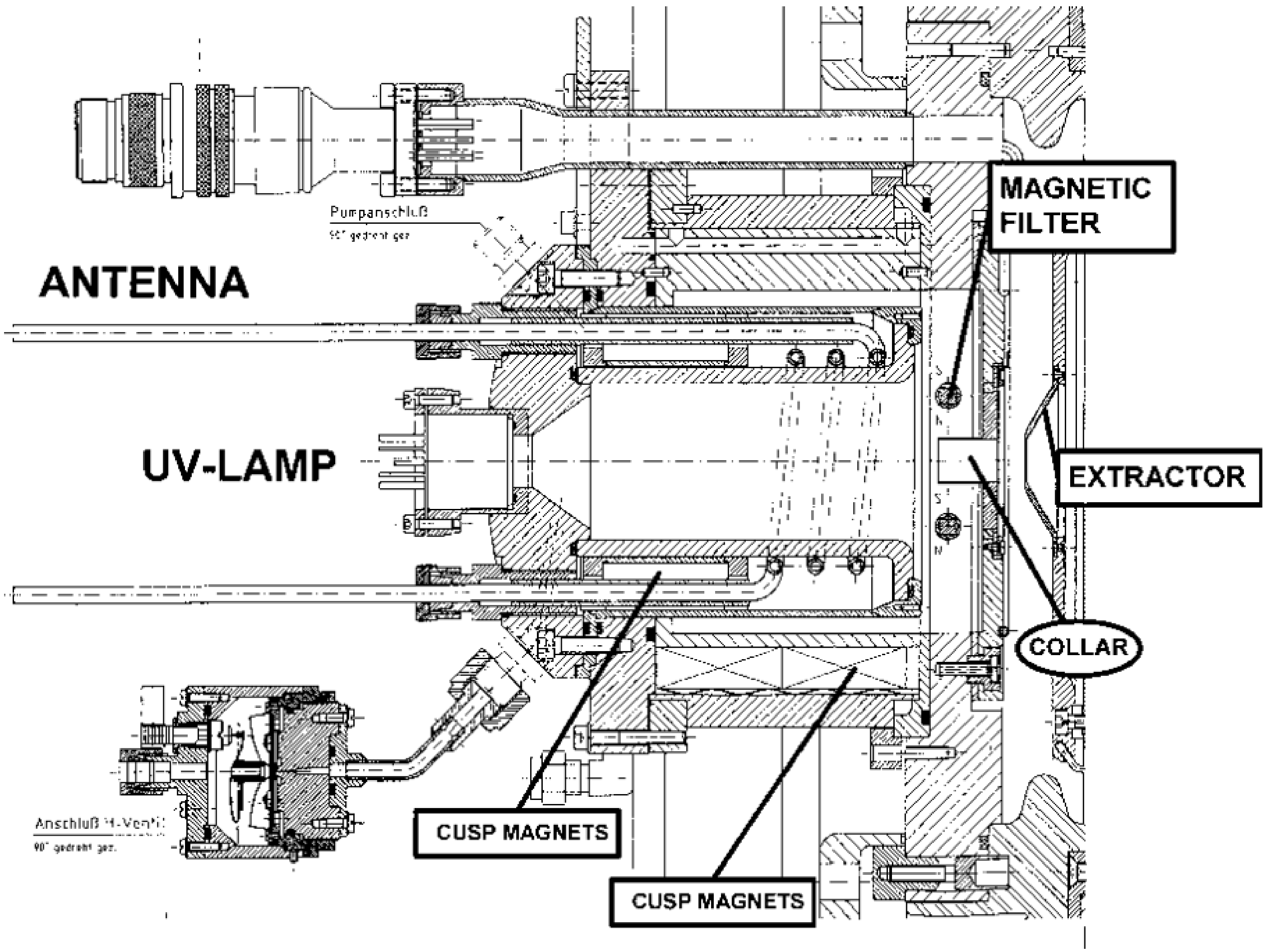
CERNs Linac4 injector





DESY source

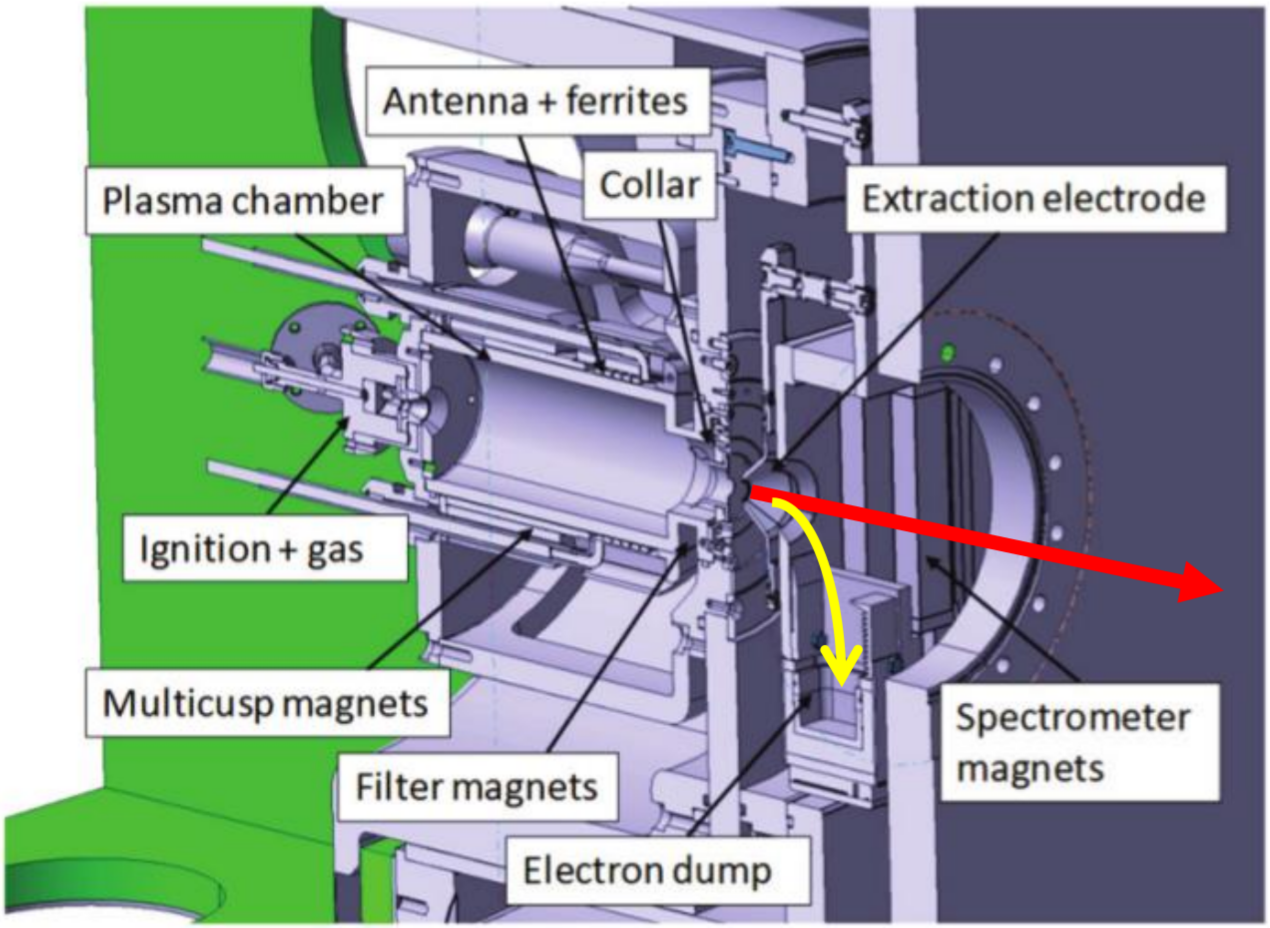
RF source from DESY at 35 kV producing 40 mA H^- in 100 μs pulses at 1 Hz.

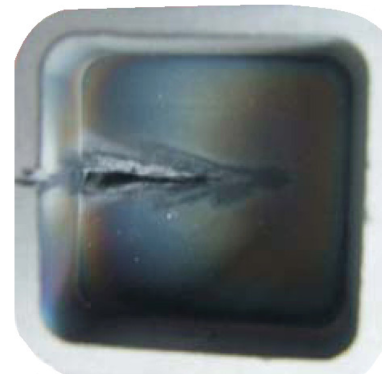


J. Peters, International Linac Conference, 2000.

DESY source at CERN

At CERN the source was run at 45 kV producing 23 mA H^- in 400 us pulses at 2 Hz and >1 A of electrons.

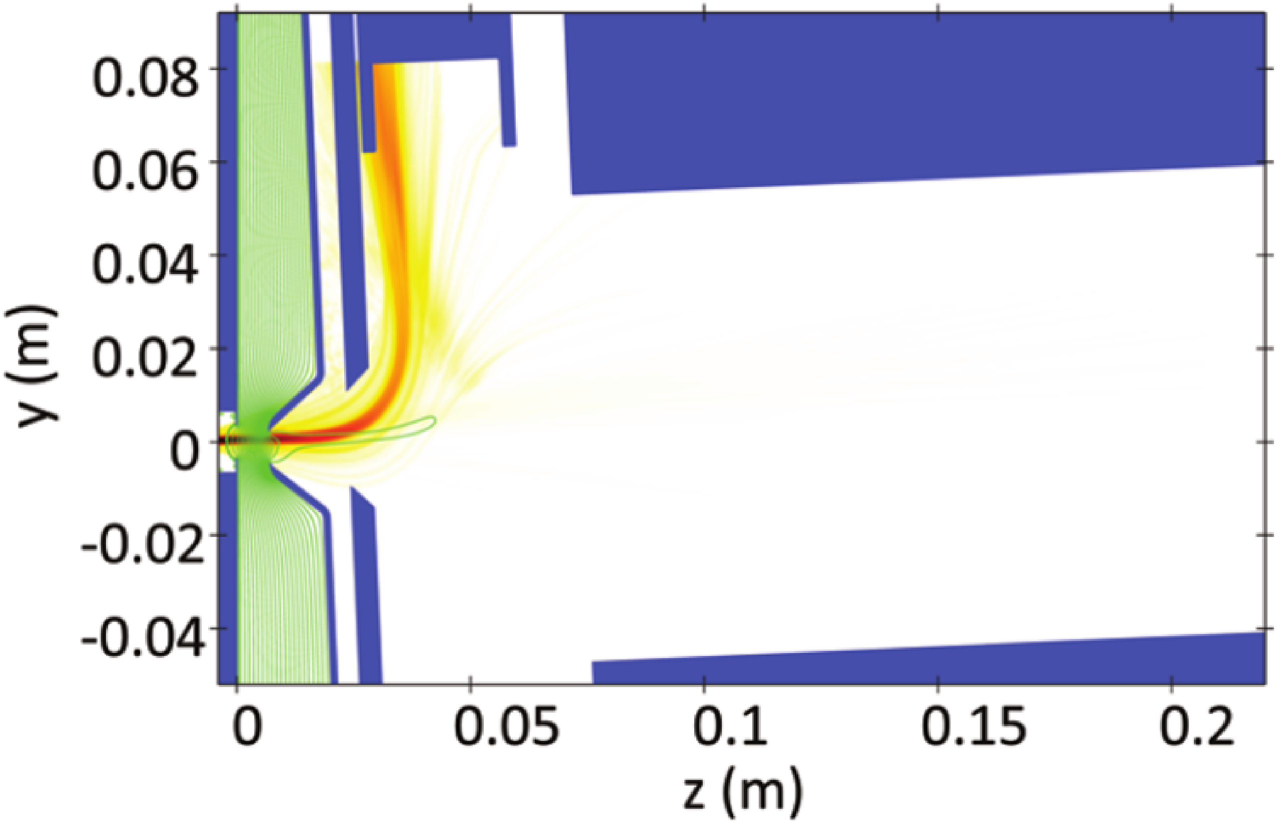


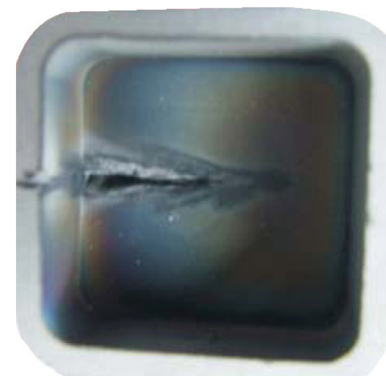


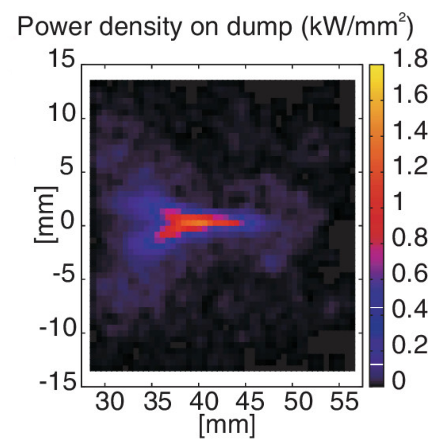
Ø. Midttun, T. Kalvas, M. Kronberger, et al., Rev. Sci. Instrum 83, 02B710 (2012).

DESY source at CERN

At CERN the source was run at 45 kV producing 23 mA H^- in 400 us pulses at 2 Hz and >1 A of electrons.

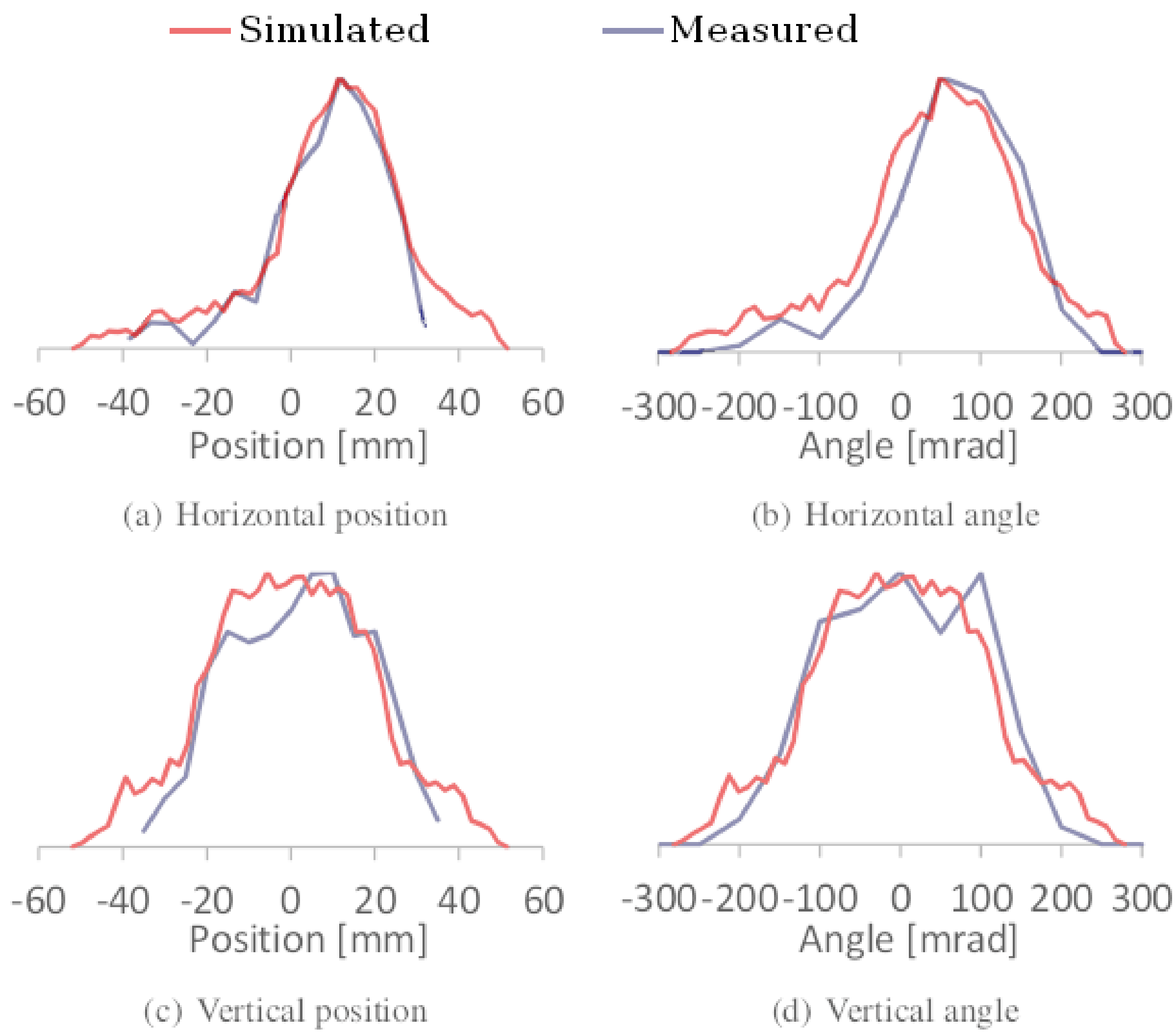






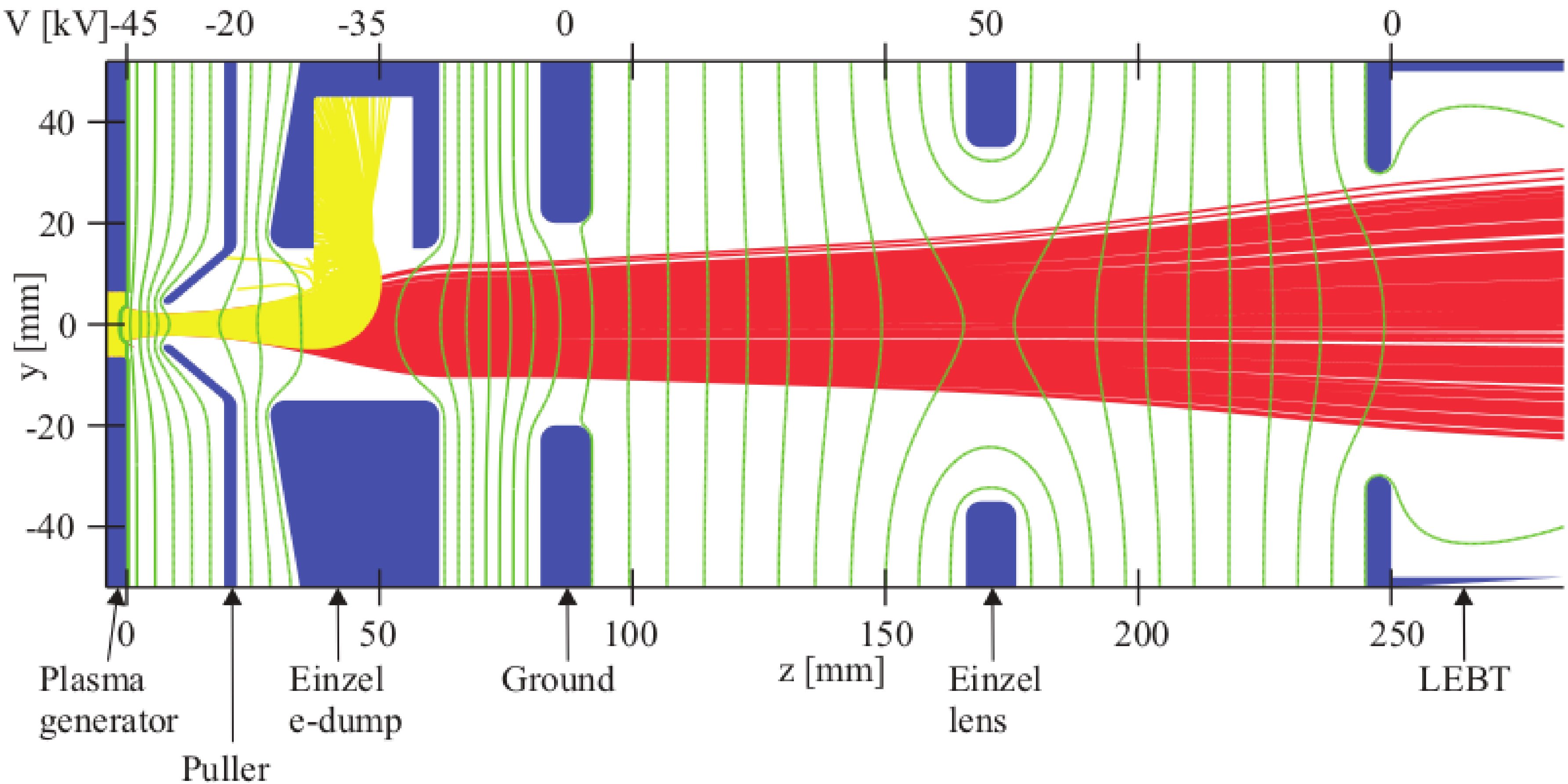
Ø. Midttun, T. Kalvas, M. Kronberger, et al., Rev. Sci. Instrum 83, 02B710 (2012).

DESY source at CERN



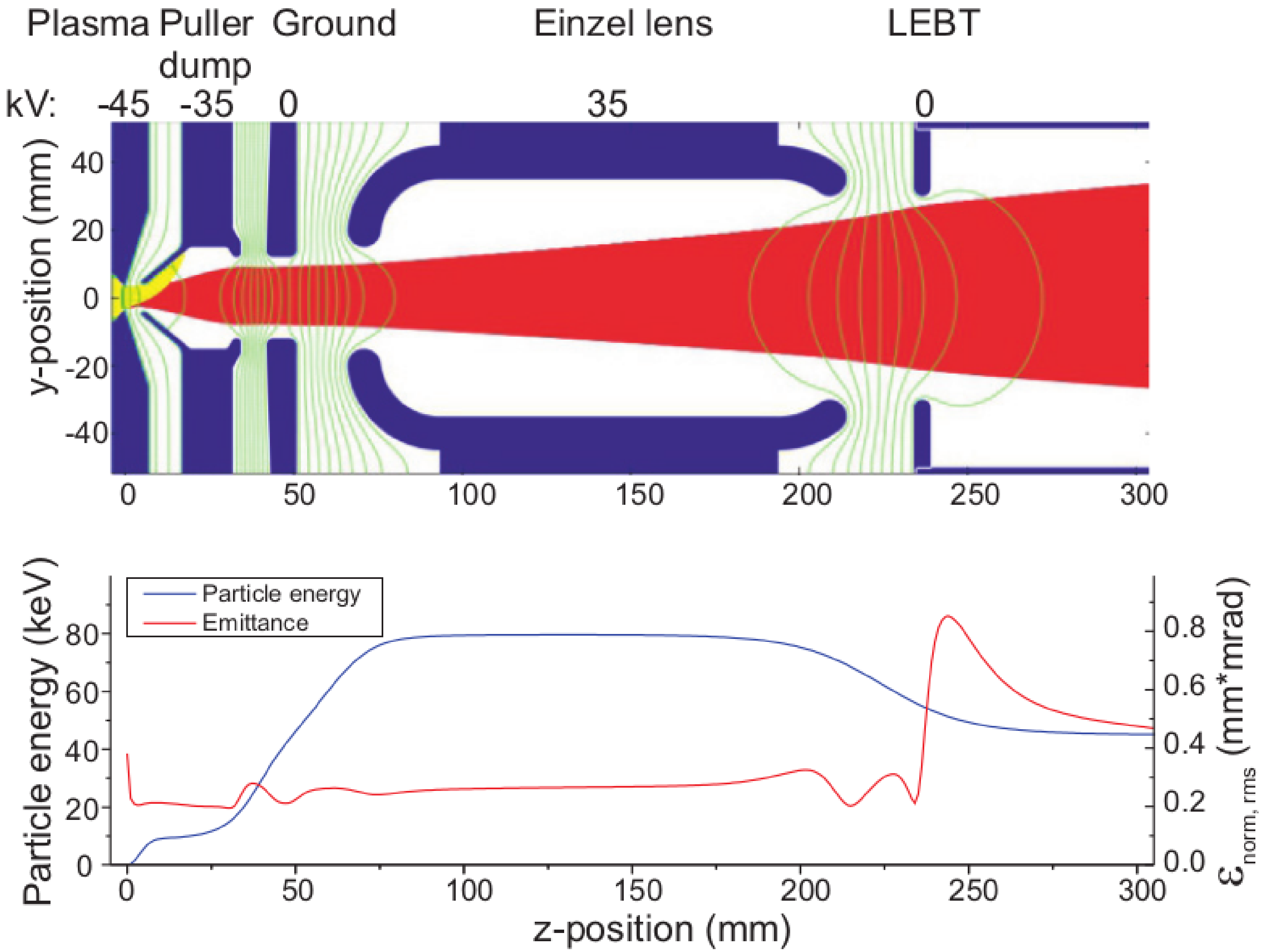
Ø. Midttun, "Improved beam extraction for a negative hydrogen ion source for the LHC injector chain upgrade, Linac4", Ph. D. thesis, University of Oslo 2014.

Linac4 IS01 – The einzel dump



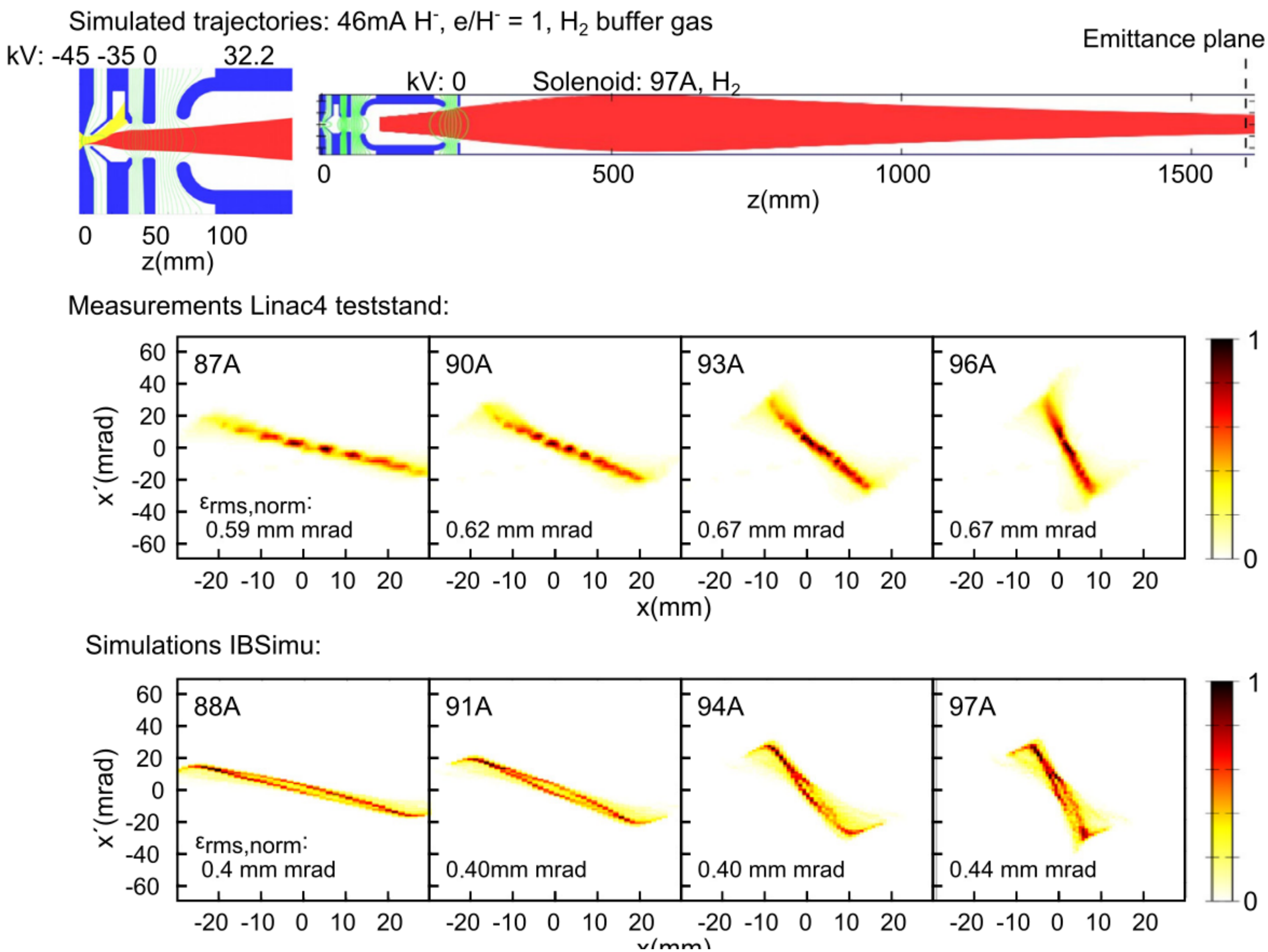
Ø. Midttun, T. Kalvas, M. Kronberger, et al., AIP Conf. Proc. 1515, 481 (2013).

Linac4 IS02 – Cesiated ion source



D. Fink, AIP Conf. Proc. 1655, 030006 (2015).

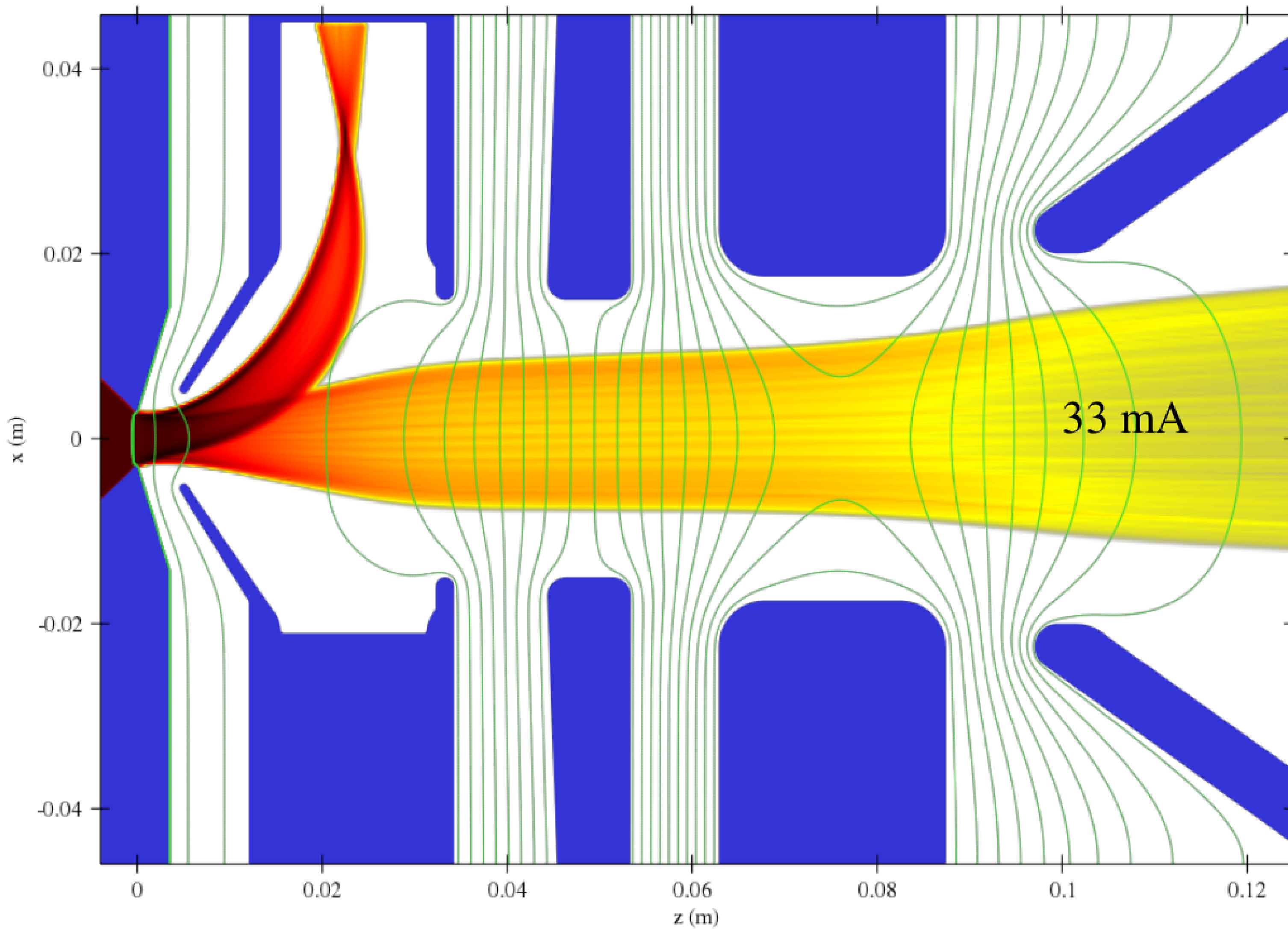
Linac4 IS02 – Cesiated ion source



D. Fink, T. Kalvas, et al., Nucl. Instrum. Meth. A 904, 179 (2018).

Linac4 IS03b

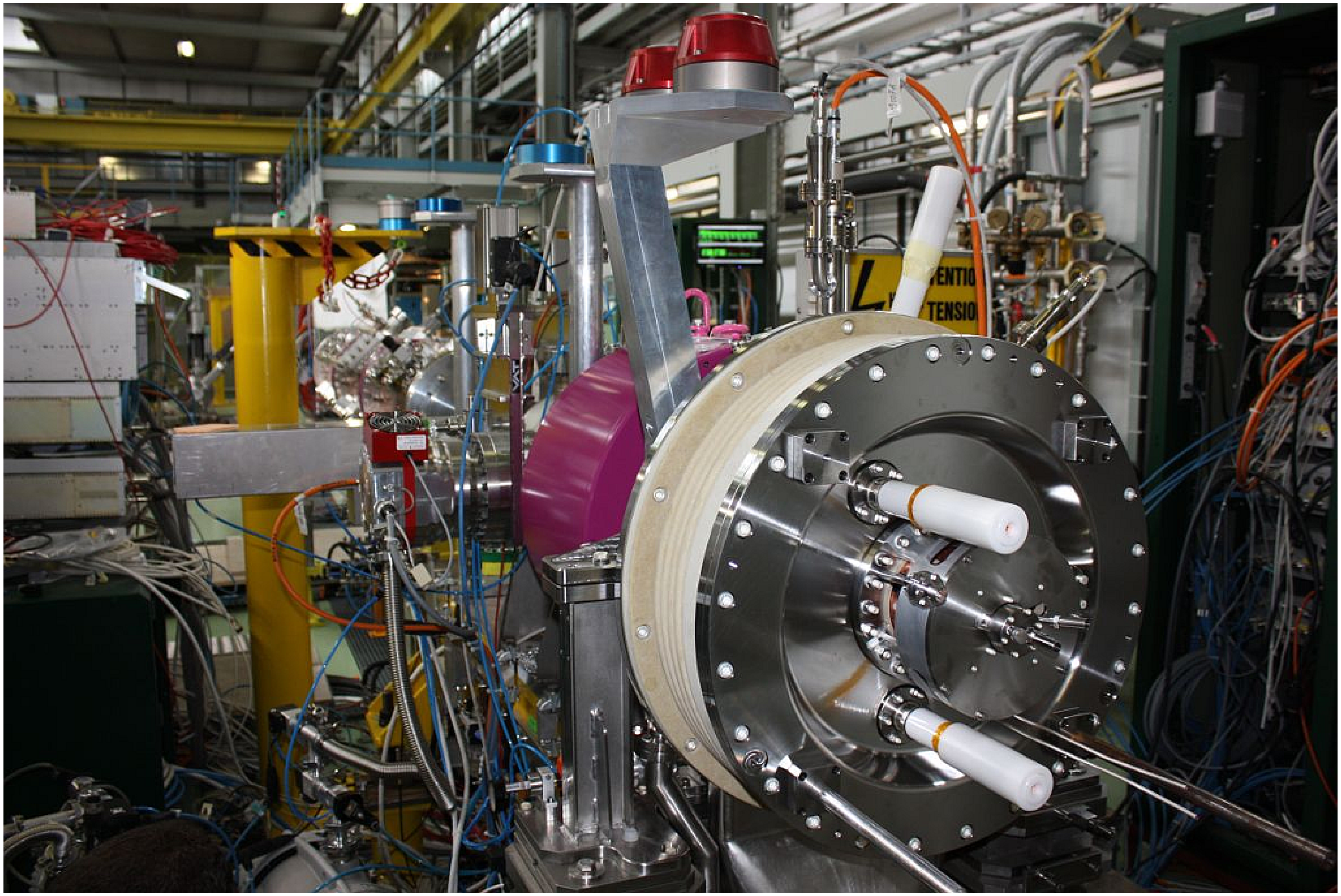
IS03a, IS03b optimized the filter field, fixed the electron dump leakage and reduced the emittance growth in the extraction einzel lens.



 Taneli Kalvas

Linac4 ion source and beam accelerator

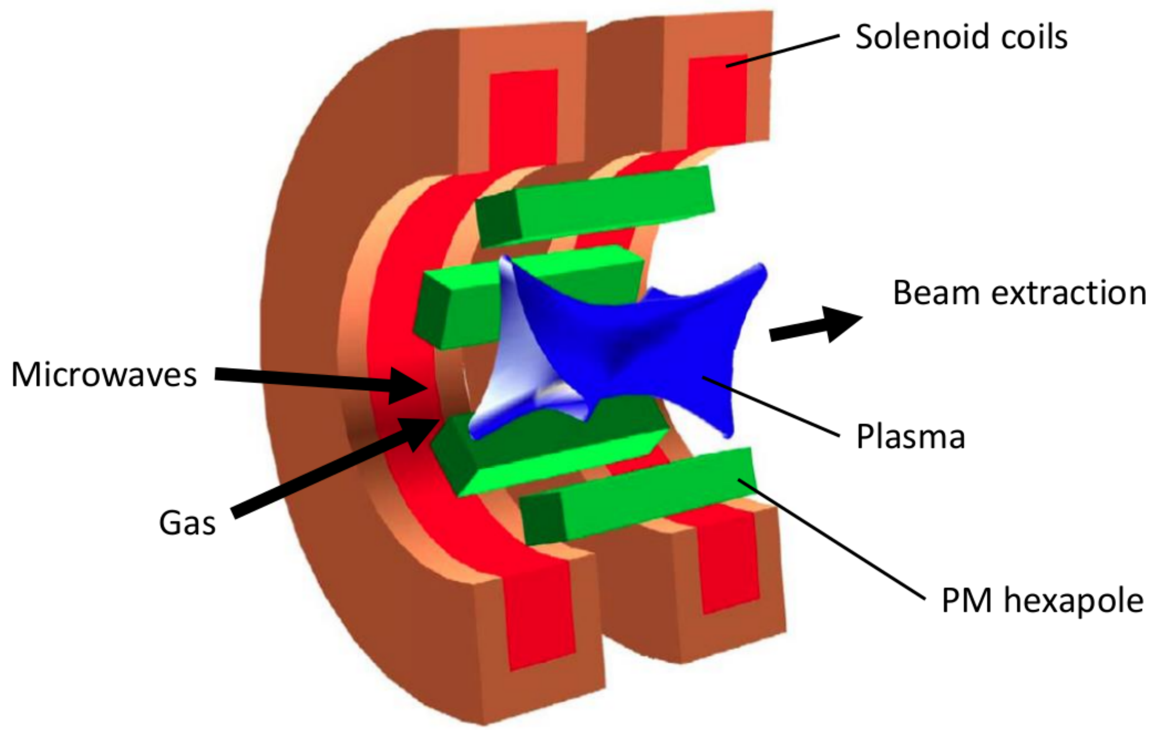
Linac4 at test bench, operational since 2017



CUBE PM-ECR ion source

ECRIS

Conventional Electron Cyclotron Resonance (ECR) ion source



Resonant condition:

$$\omega_e = \frac{eB}{m} = \omega_{\mathrm{RF}} \tag{8}$$

Field scaling:

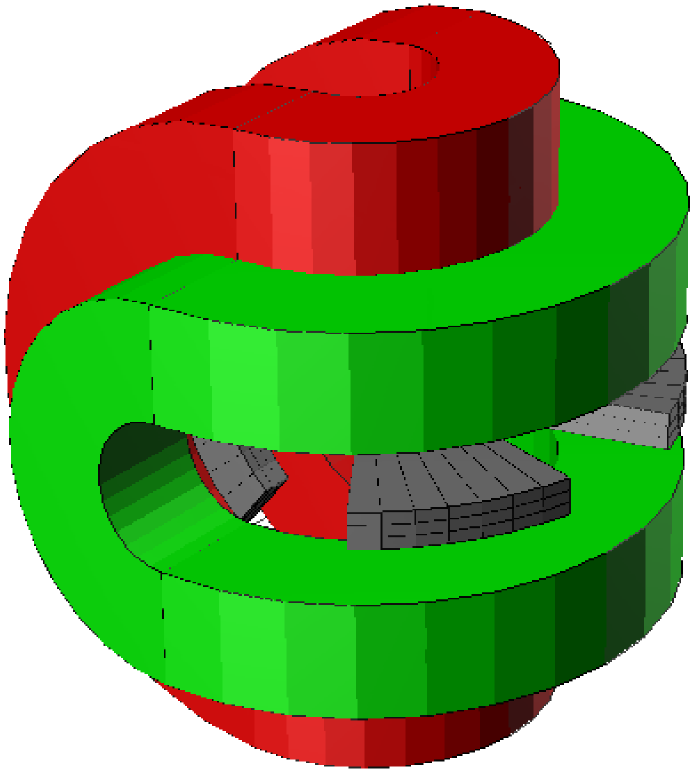
$$B_{\mathrm{inj}} = 4B_{\mathrm{ECR}} \tag{9}$$

$$B_{\mathrm{rad}} = 2B_{\mathrm{ECR}} \tag{10}$$

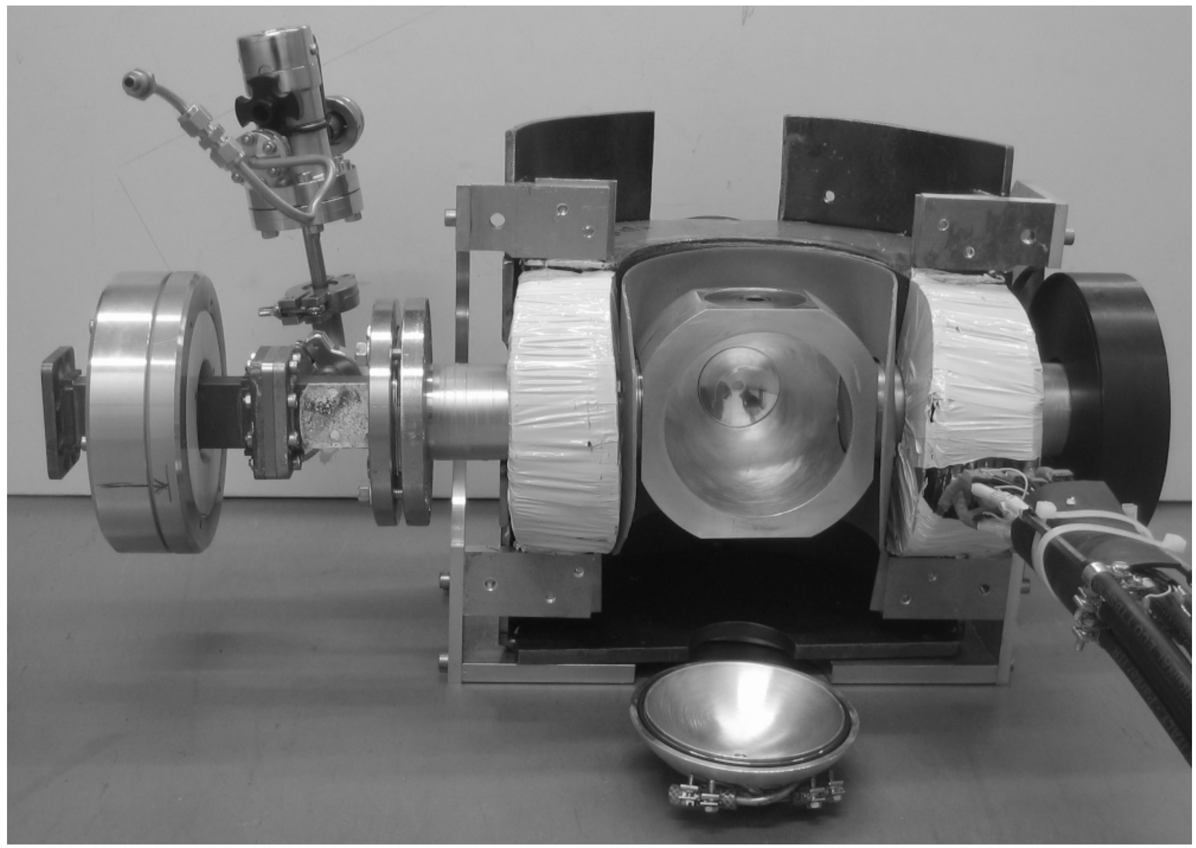
$$B_{\mathrm{ext}} \approx 0.9B_{\mathrm{rad}} \tag{11}$$

$$B_{\mathrm{min}} \approx 0.4B_{\mathrm{rad}} \tag{12}$$

ARC-ECRIS



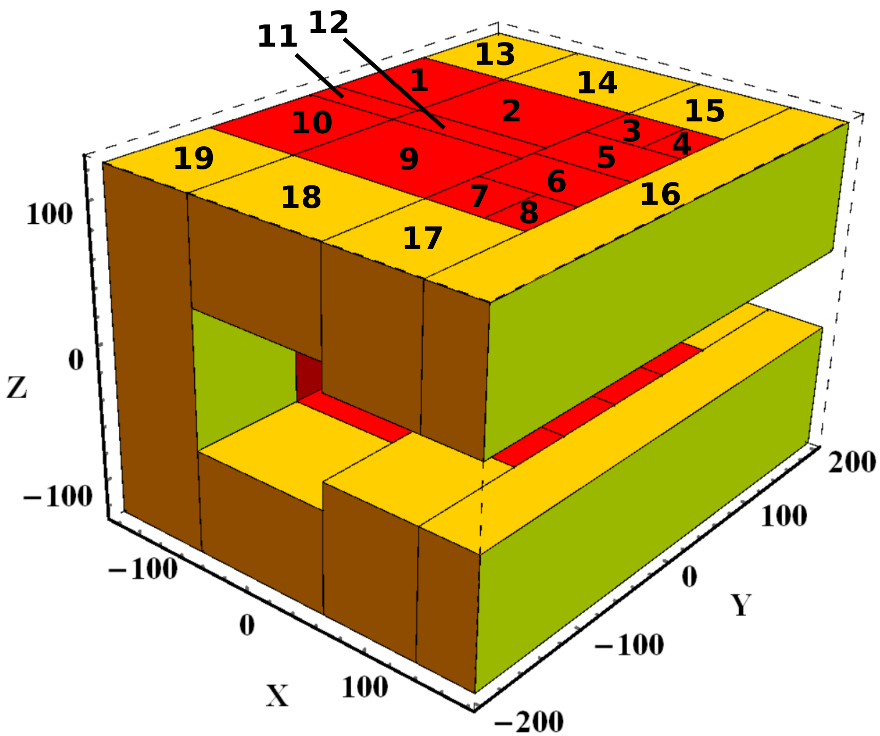
First ARC-ECRIS prototype

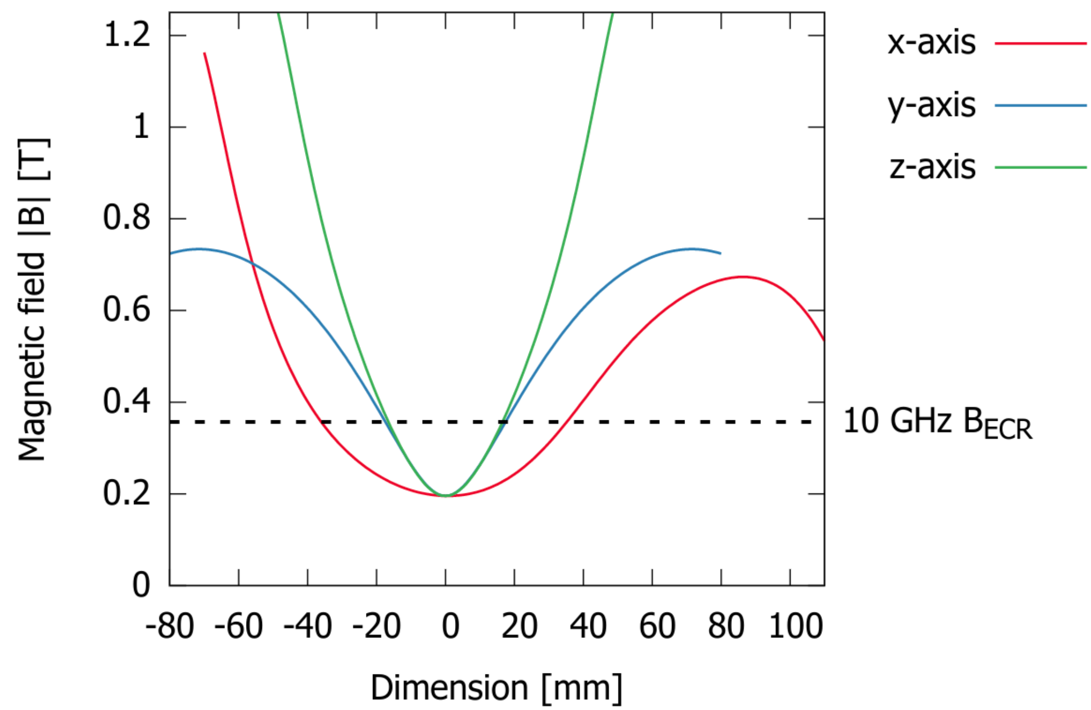


H. Koivisto, P. Suominen, et al., Rev. Sci. Instrum. 83, 02A312 (2012).

CUBE-ECRIS

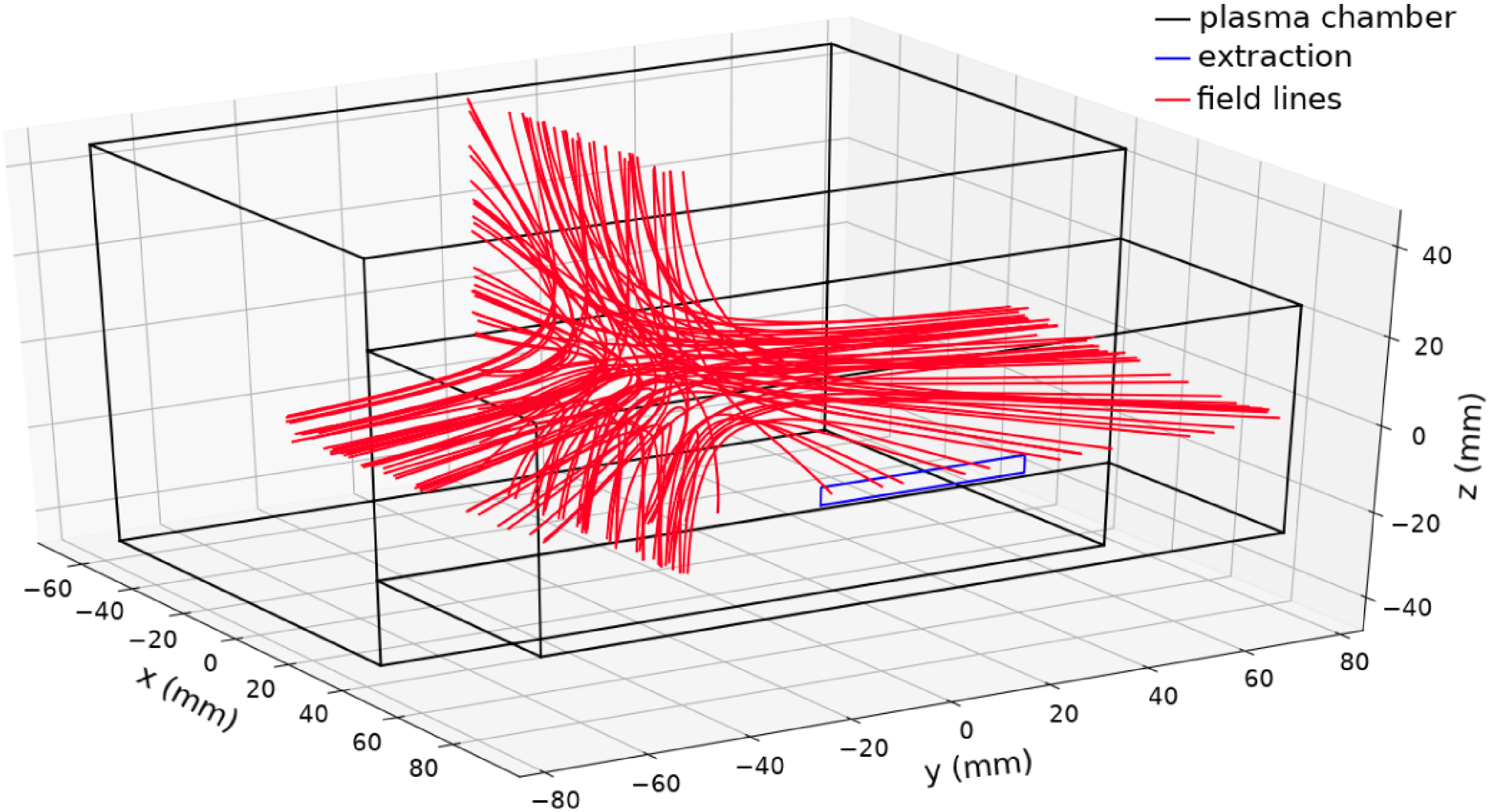
All permanent magnet minimum-B quadrupole field topology



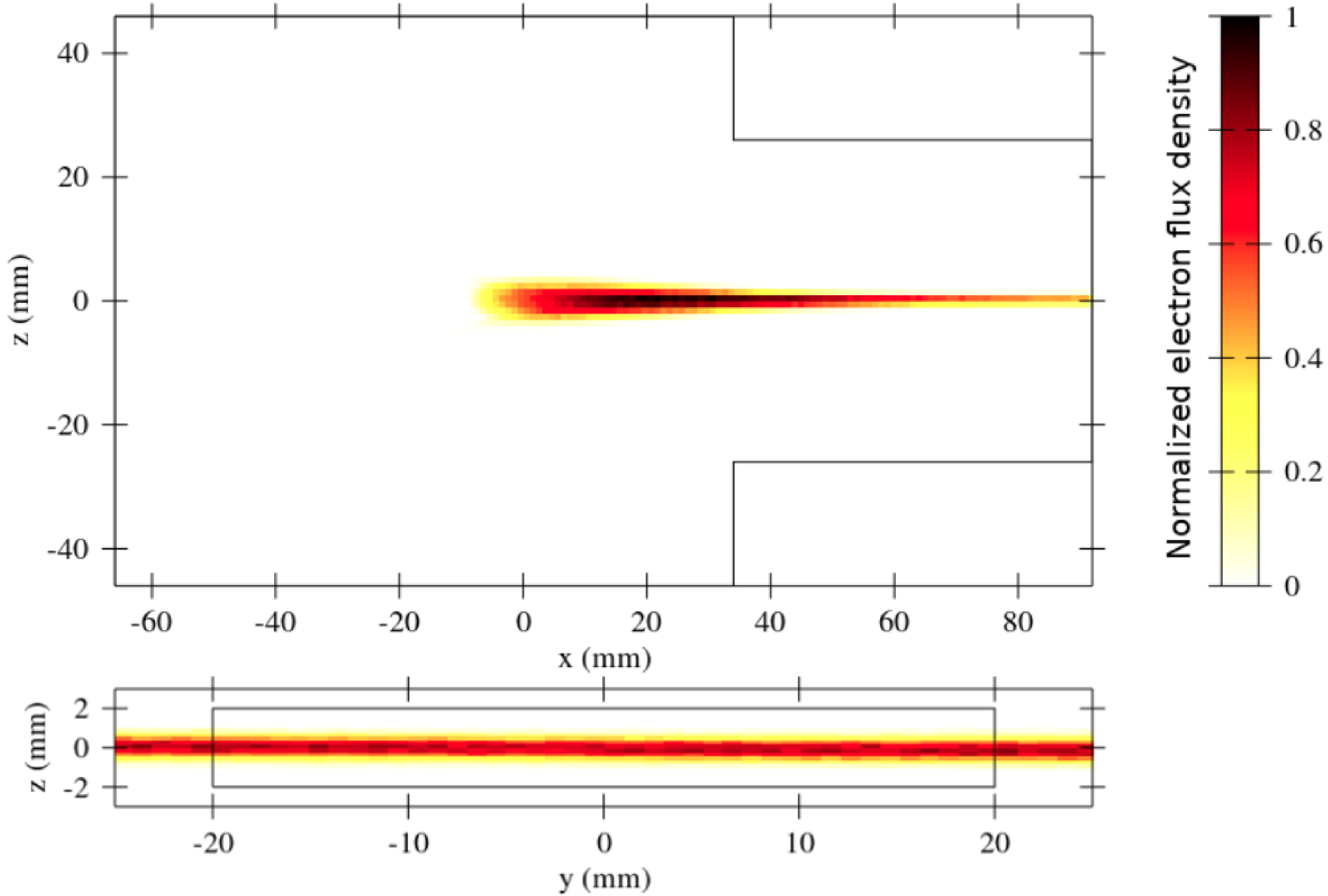


T. Kalvas et al., Journal of instrumentation 15, P06016 (2020).

Loss lines from plasma



Loss lines from plasma



Plasma model

Previously we had

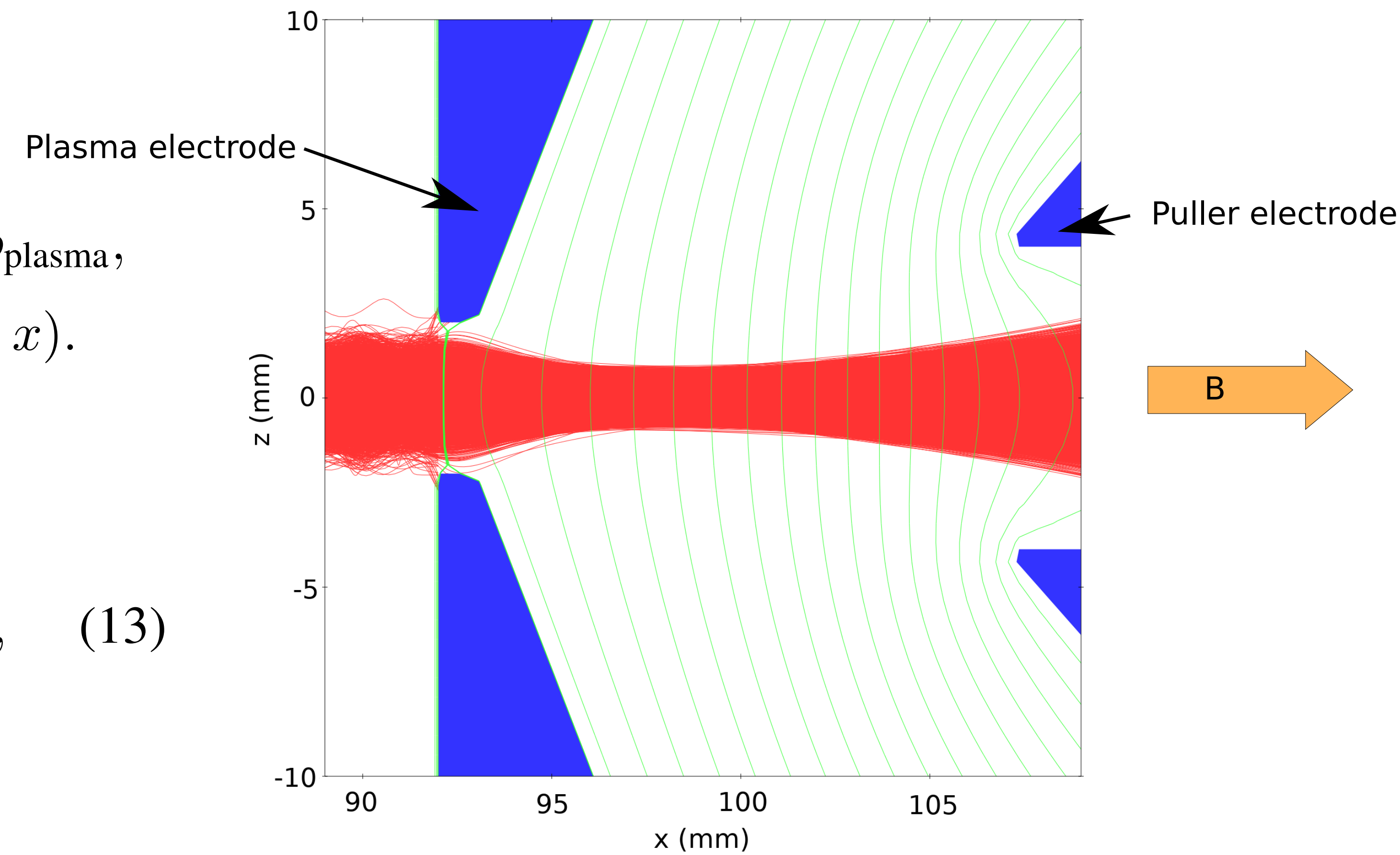
$$\rho = \rho_{\text{beam}} + \rho_{\text{plasma}},$$

$$\rho_{\text{plasma}} = \rho_{\text{plasma}}(\phi, x).$$

Now

$$\rho = S(U) \cdot \rho_{\text{beam}}, \qquad (13)$$

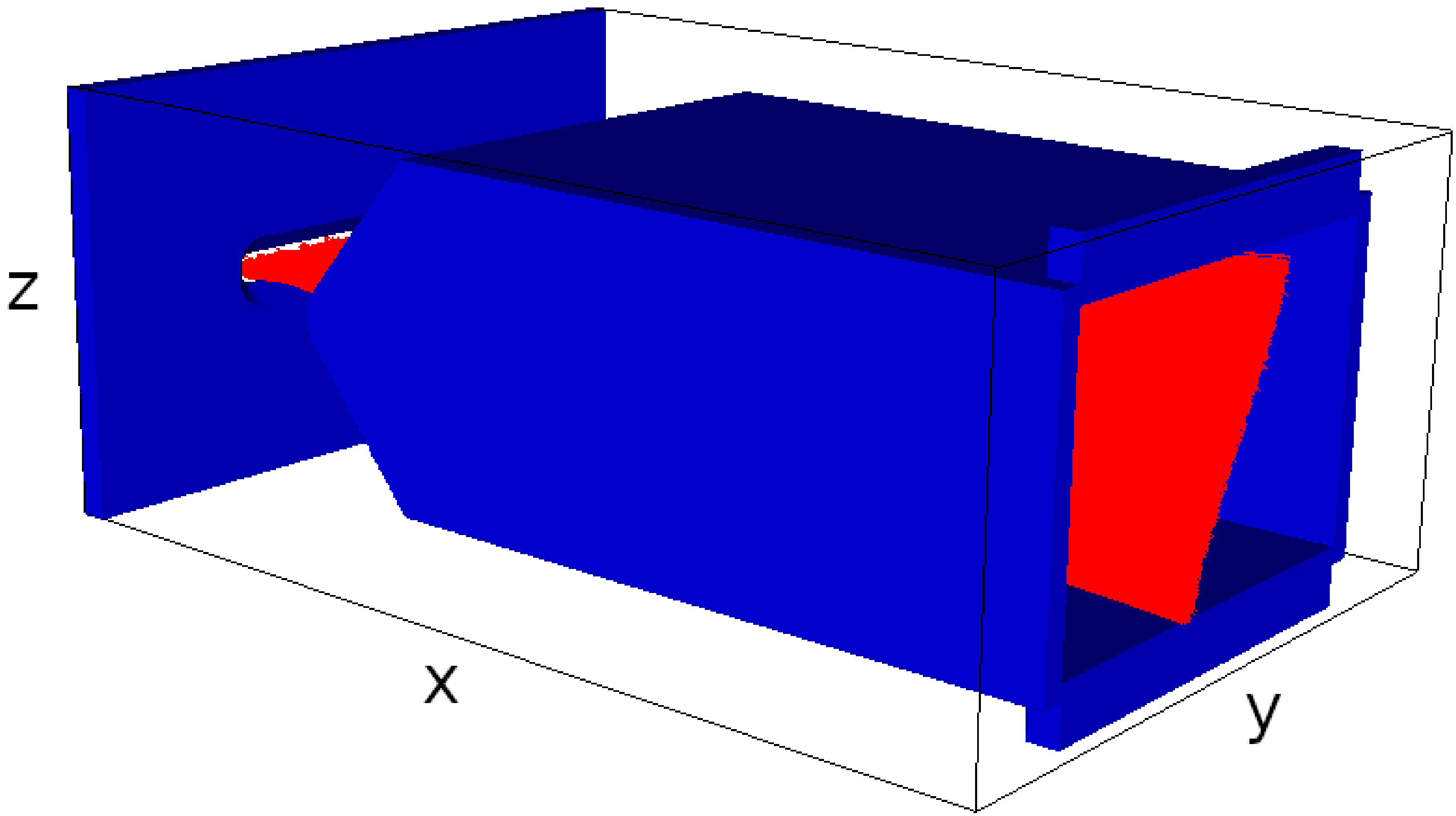
where



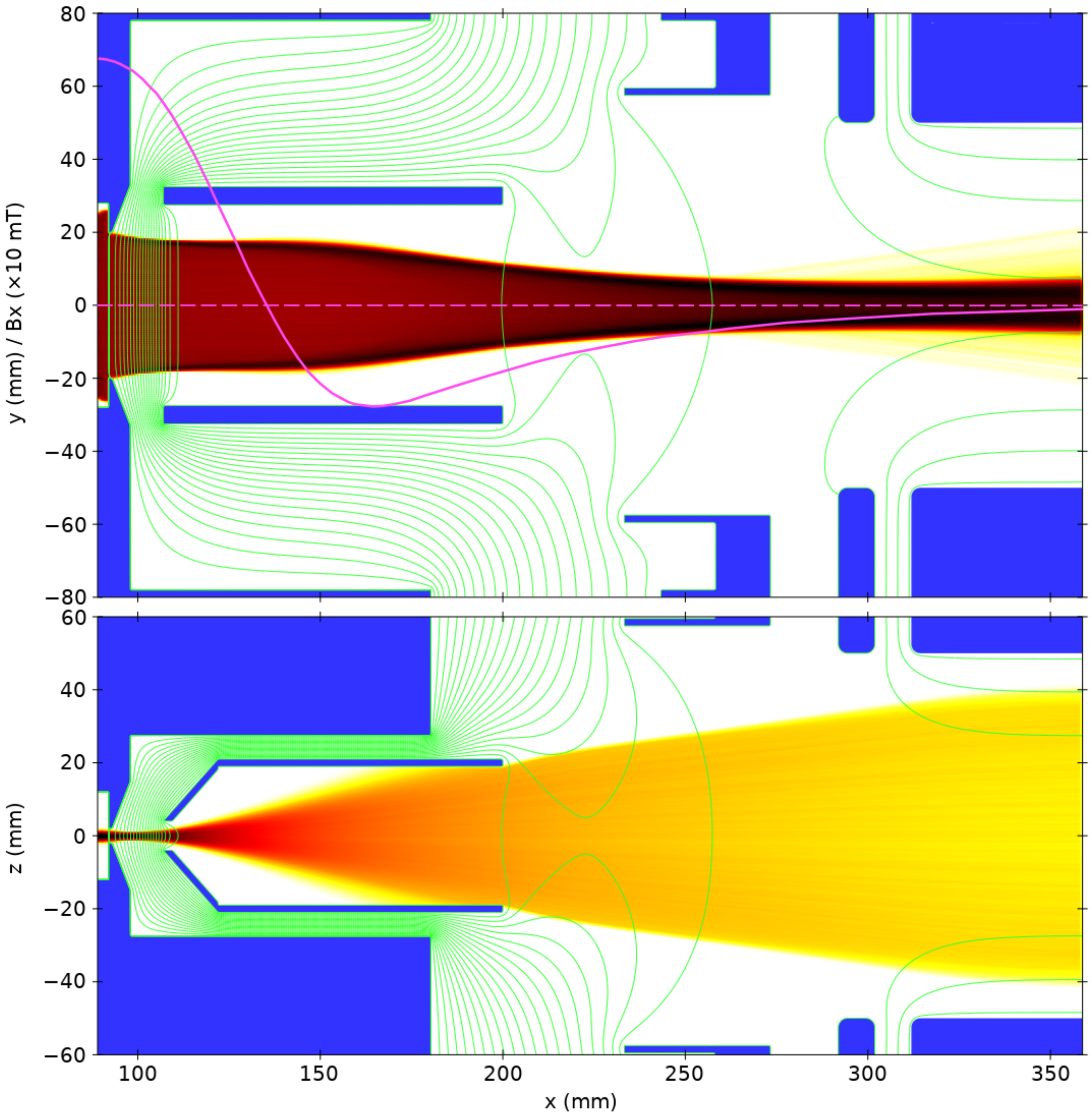
$$S(U) = \frac{1}{2}\left(1 + \tanh\left(\frac{U - U_M}{T_M}\right)\right). \qquad (14)$$

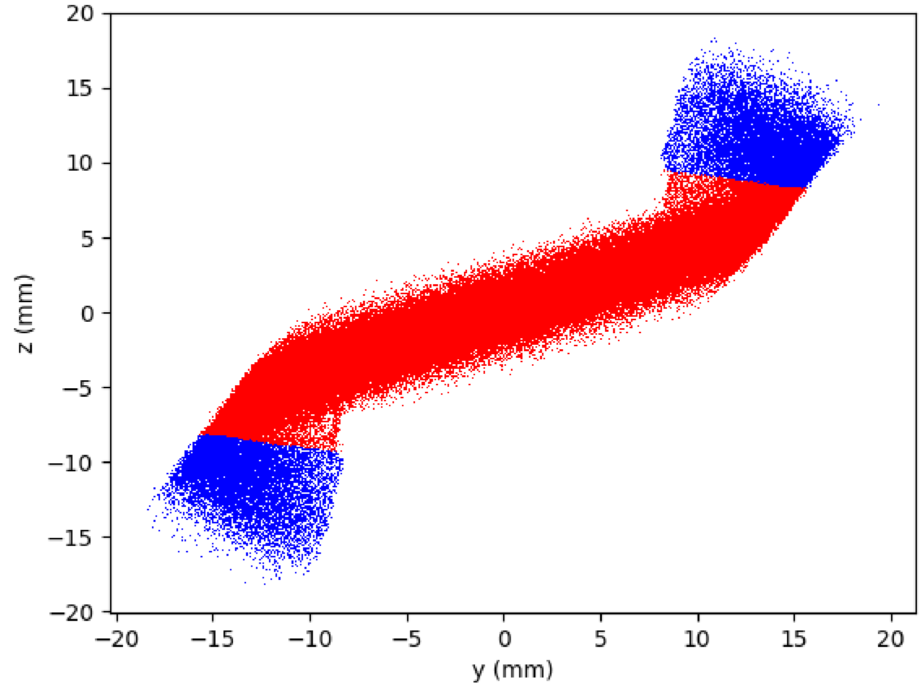
P. Veltri et al., Rev. Sci. Instrum. 85 02A711 (2014).

Beam rotation

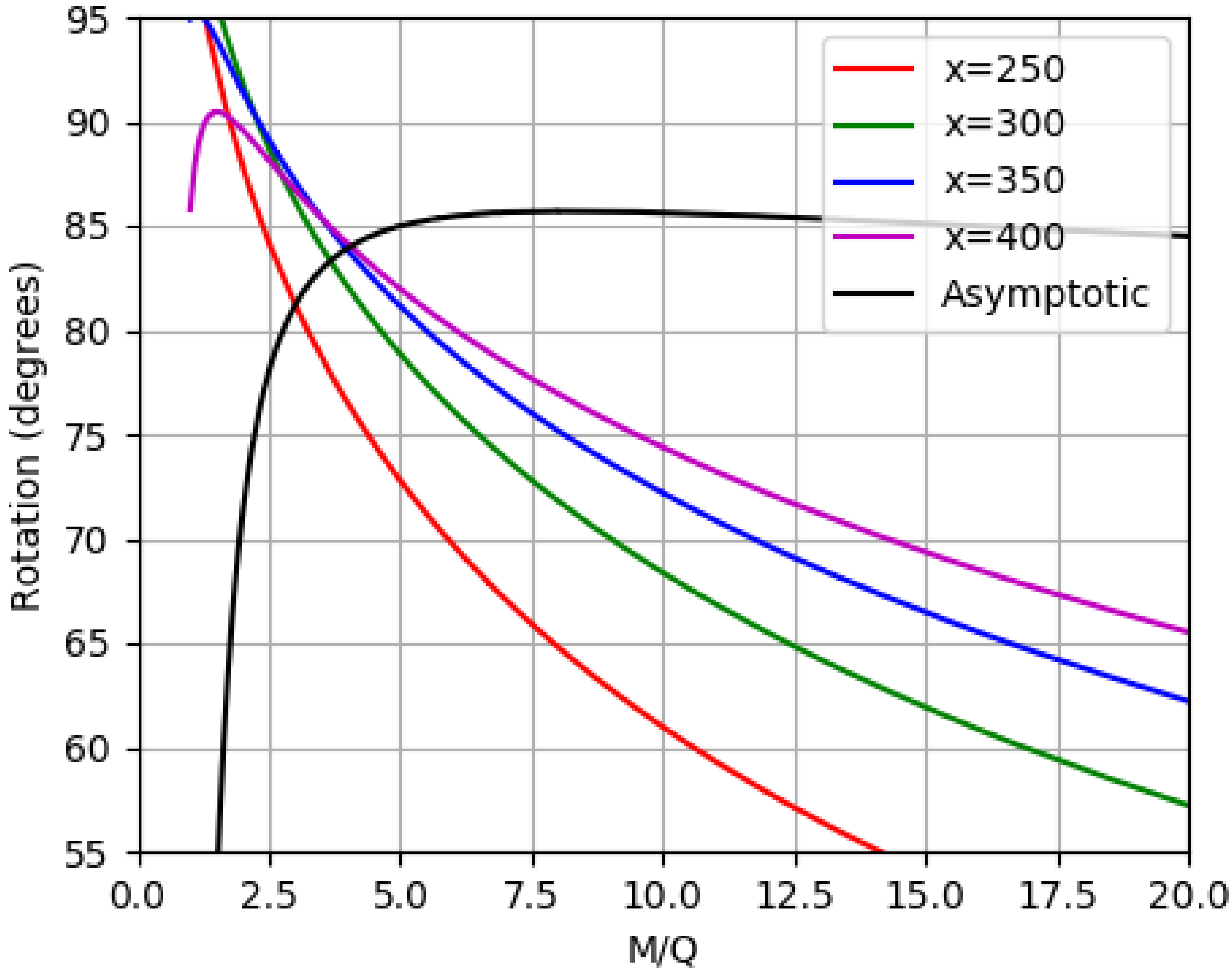


Beam rotation

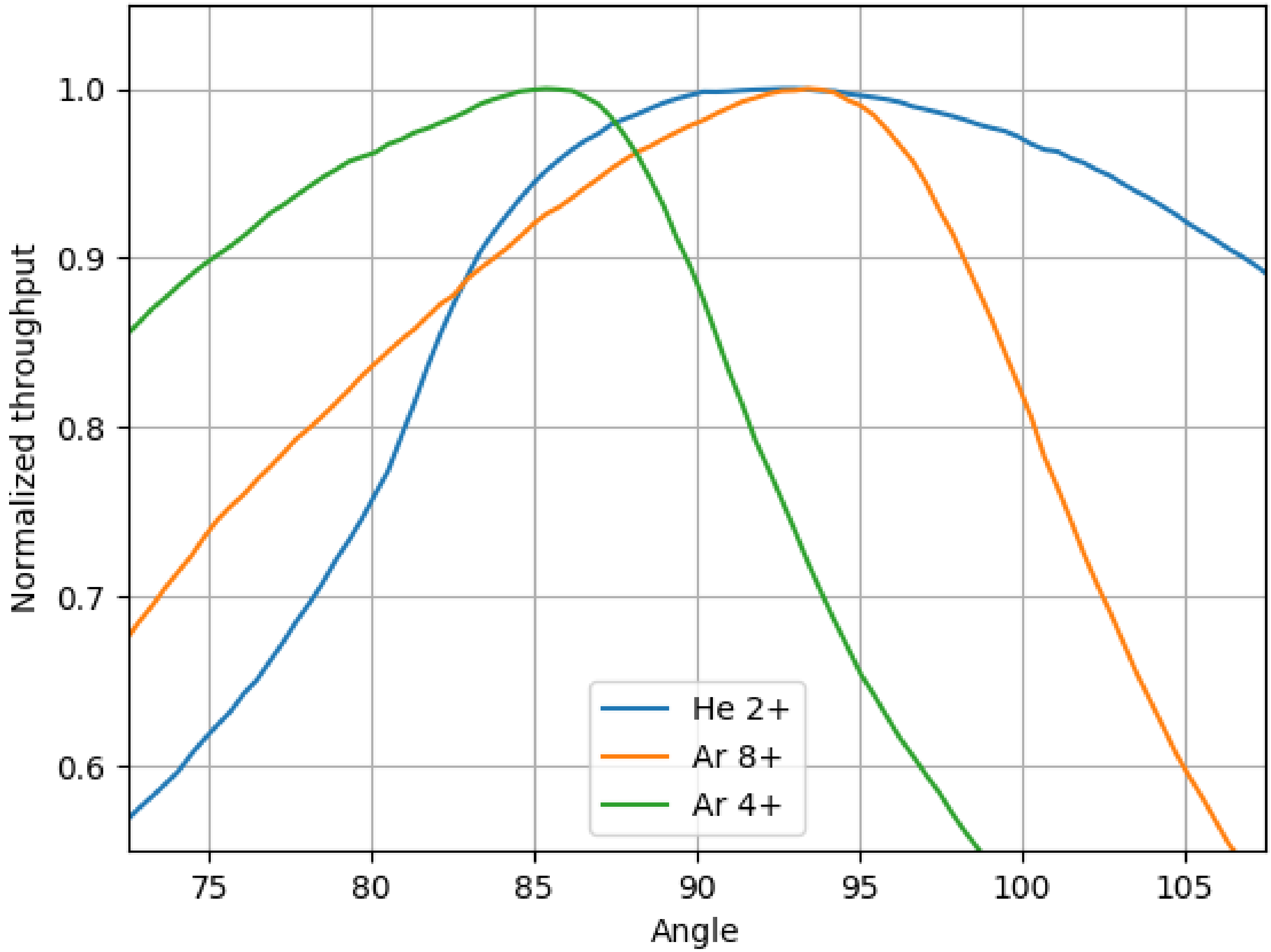




Beam rotation



Beam rotation



ToF-ERDA detector modelling

Time-of-Flight Elastic Recoil Detection Analysis

Incident beam from < 10 MeV to $N \cdot 10$ MeV

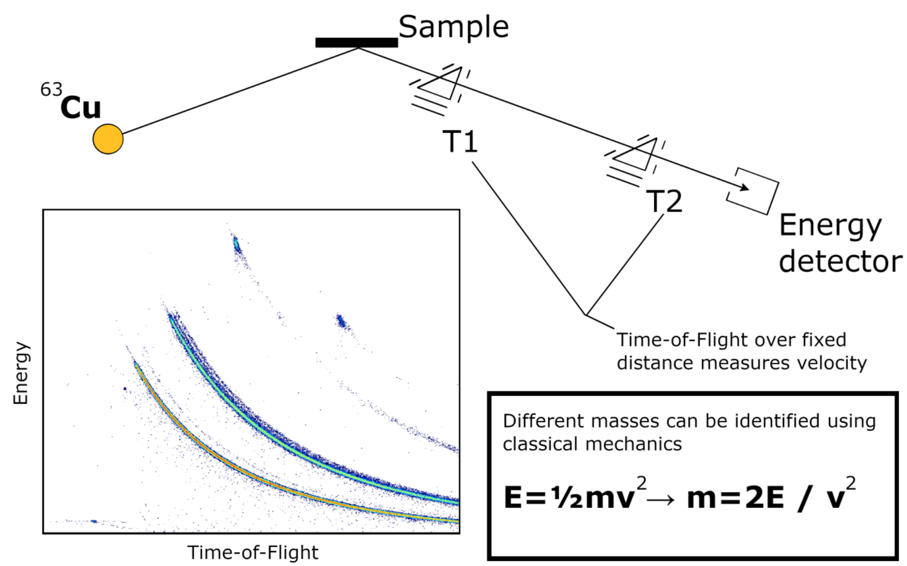
Sample atoms recoil to detector

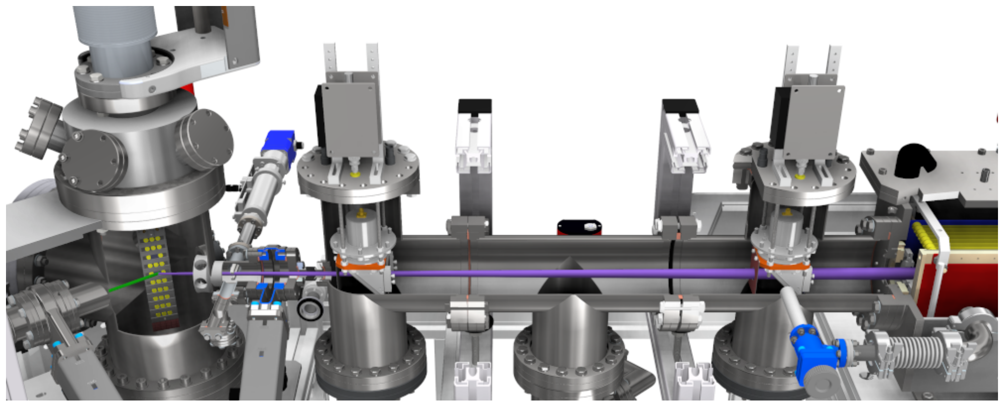
Energy and velocity signals

Superior to RBS with light elements in heavy substrate

Elements from H to U

1 nm depth resolution

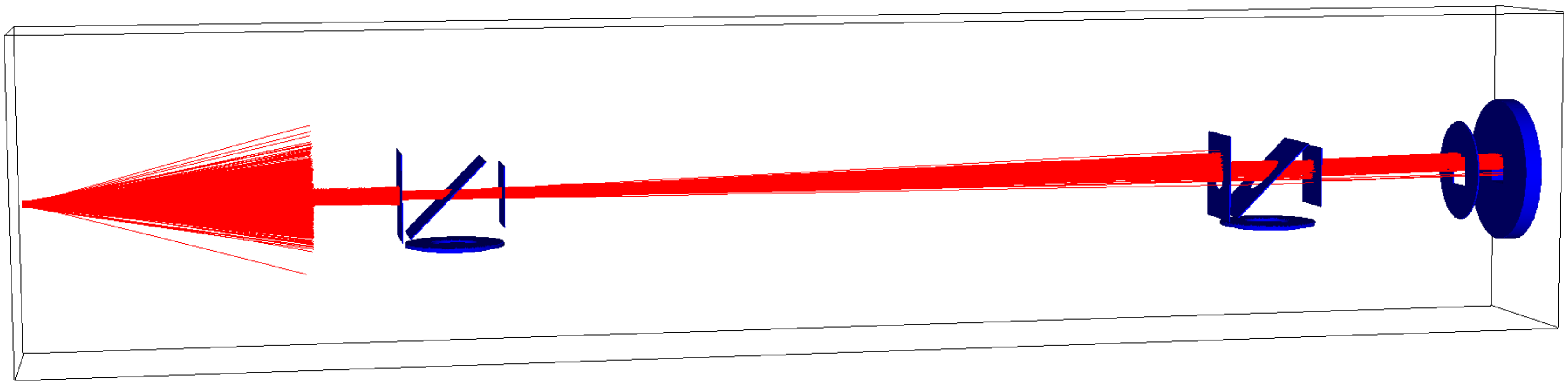




 Taneli Kalvas

Time-of-Flight Elastic Recoil Detection Analysis

IBSimu used to model the ToF gates



Features:

- Collimation on geometry
- Fields from gates
- Monte Carlo scattering implemented on foils (callback routines)

 Taneli Kalvas

Other codes

What IBSimu is not!

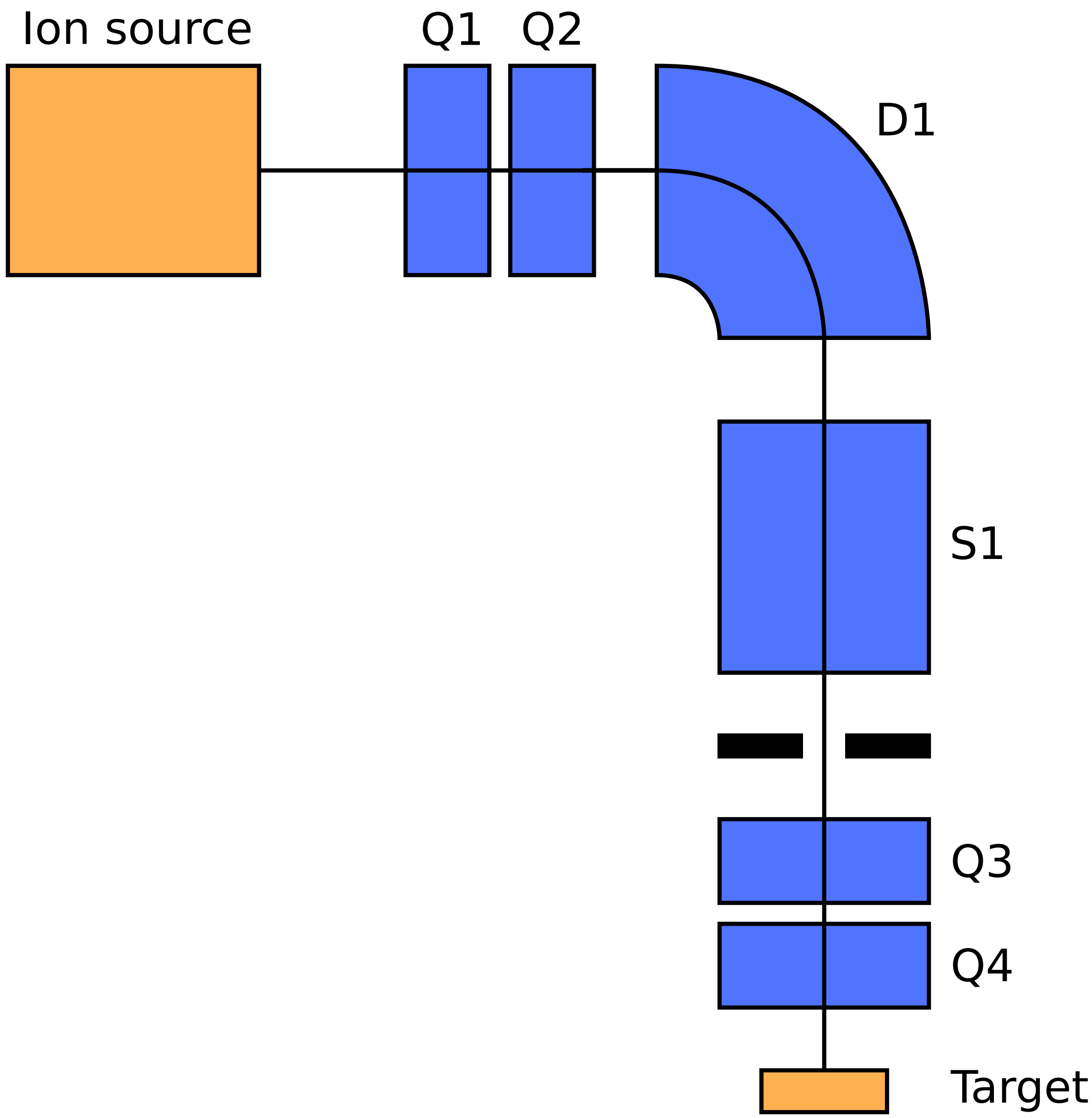
Tools of the trade

Use something which is suitable for your problem!

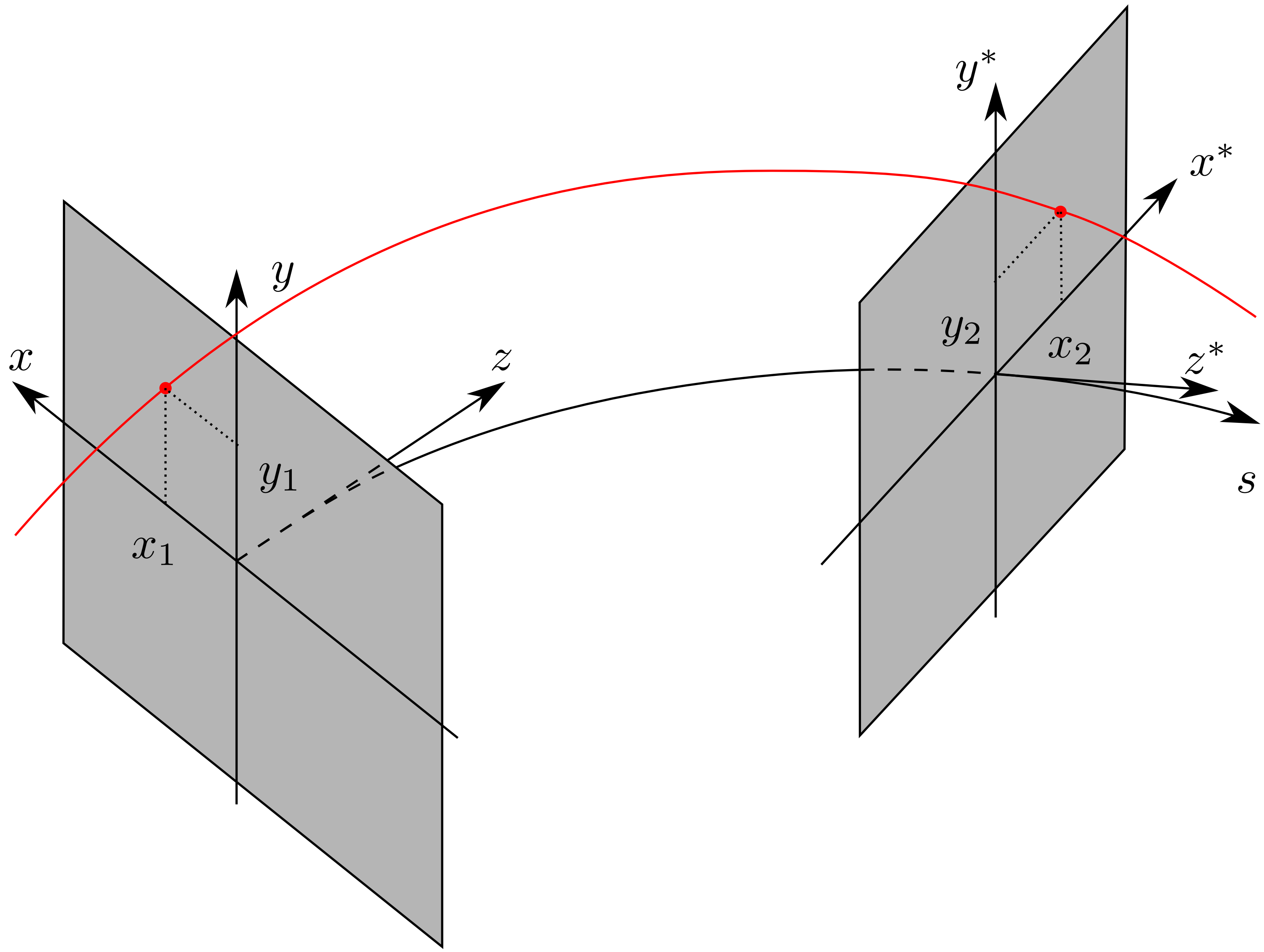
- Ion-optical software based on N^{th}-order approximation of trajectories.
- Electromagnetic field programs: POISSON SUPERFISH, FEMM, RADIA-3D, VECTOR FIELDS (OPERA), COMSOL MULTIPHYSICS, LORENTZ, etc. Some with and some without particle tracking capability.
- Specialized ion source extraction software.
- Many other specialized programs for modelling beam space charge compensation, bunching, cyclotron injection, collisional ion source plasmas, linear accelerators, etc. with PIC/MCC/hybrid methods.

Science is most often about doing something that is far from routine $\Rightarrow$ Often custom tools are needed.

Example ion optical system



Curvilinear coordinate system



The s-axis is defined by the reference particle and y is “up”.

Curvilinear coordinate system

The coordinates of each particle consists of the longitudinal coordinate s, transverse position (x, y) and velocity components (v_x, v_y, v_z).

Also tangents of the trajectory angles, often defined using components of the momentum vector

$$x' = \frac{p_x}{p_z} = \tan(\alpha) \tag{15}$$

$$y' = \frac{p_y}{p_z} = \tan(\beta) \tag{16}$$

are used. Please note that this is not the only convention. Also

$$x' = \frac{p_x}{p_0} \tag{17}$$

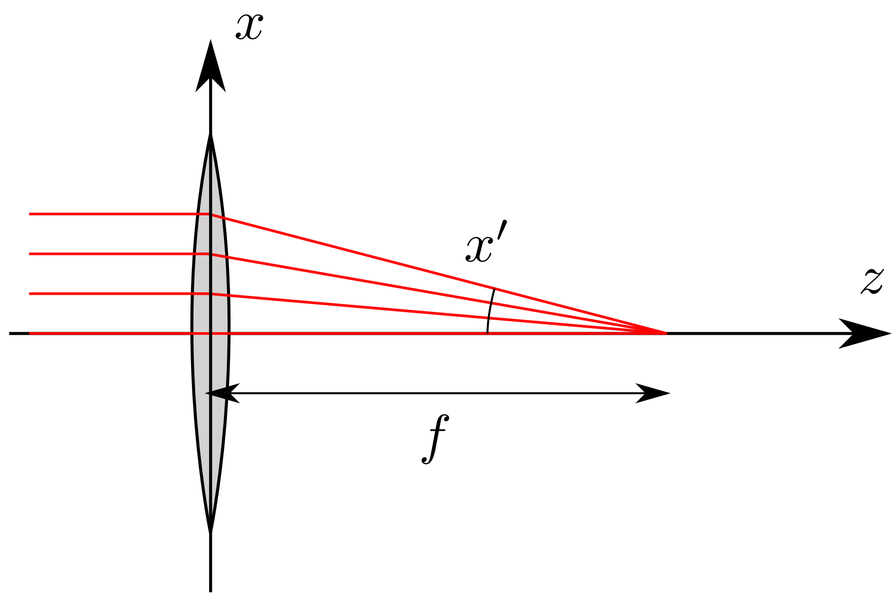
is used, where p_0 is the momentum of the reference particle. While x' and y' are not really angles, typically units of rad or mrad are used.

Light optics: ideal thin lens

Why "ion optics"? Ion propagation analogous to propagation of photons in *light optics*, i.e. systems of lenses.

Thin optical lens:

$$\Delta x' = -\frac{x}{f} \tag{18}$$



Matrix representation

Each ray of light is described with position x and direction x' conveniently given as components of a vector $\vec{X}$.

As a ray of light passes through elements/matter/anything the results can be described with a function: $\vec{X}_2 = f(\vec{X}_1) = \big(f_1(x, x'), f_2(x, x')\big)$.

According to Taylor, the functions can be represented by an infinite series

$$f_i = a_{i,0} + a_{i,1}x + a_{i,2}x' + a_{i,3}x^2 + a_{i,4}x'^2 + a_{i,5}xx' + \cdots . \tag{19}$$

If coordinate system is chosen wisely, the constant term $a_{i,0} = 0$ and by approximating higher order terms to zero one arrives at a linear model

$$(x_2, x_2') = (a_{1,1}x_1 + a_{1,2}x_1', a_{2,1}x_1 + a_{2,2}x_1') \tag{20}$$

conveniently given by matrix multiplication

$$\vec{X}_2 = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} \vec{X}_1. \tag{21}$$

Matrix representation of thin lens

In a thin lens, position of the light ray can not change:

$$x_2 = x_1. \tag{22}$$

The angle changes due to focusing action as given before: $\Delta x' = -x/f$, now written as

$$x_2' = x_1' - x_1/f. \tag{23}$$

These equations can be written as a matrix:

$$\begin{pmatrix} x_2 \\ x_2' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -1/f & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}. \tag{24}$$

Taneli Kalvas

Matrix representation of drift space

In empty space of length L, the position of the light ray changes as

$$x_2 = x_1 + x'_1 L. \tag{25}$$

The angle does not change.

$$x'_2 = x'_1. \tag{26}$$

Equations as a matrix:

$$\begin{pmatrix} x_2 \\ x'_2 \end{pmatrix} = \begin{pmatrix} 1 & L \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x'_1 \end{pmatrix}. \tag{27}$$

Transport of light ray using matrices

A light ray passing through an optical system with drift length L_1, a thin lens with focal length f and a drift length L_2 is described by

$$\begin{pmatrix} x_4 \\ x_4' \end{pmatrix} = \begin{pmatrix} 1 & L_2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -1/f & 1 \end{pmatrix} \begin{pmatrix} 1 & L_1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}. \tag{28}$$

According to matrix algebra we can evaluate the matrix multiplications giving

$$\begin{pmatrix} x_4 \\ x_4' \end{pmatrix} = \begin{pmatrix} 1 - L_2/f & L_1 + L_2 - L_1 L_2/f) \\ -1/f & 1 - L_1/f \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}. \tag{29}$$

What does this mean?

Meaning of matrix elements

Earlier we presented the matrix transport equation as

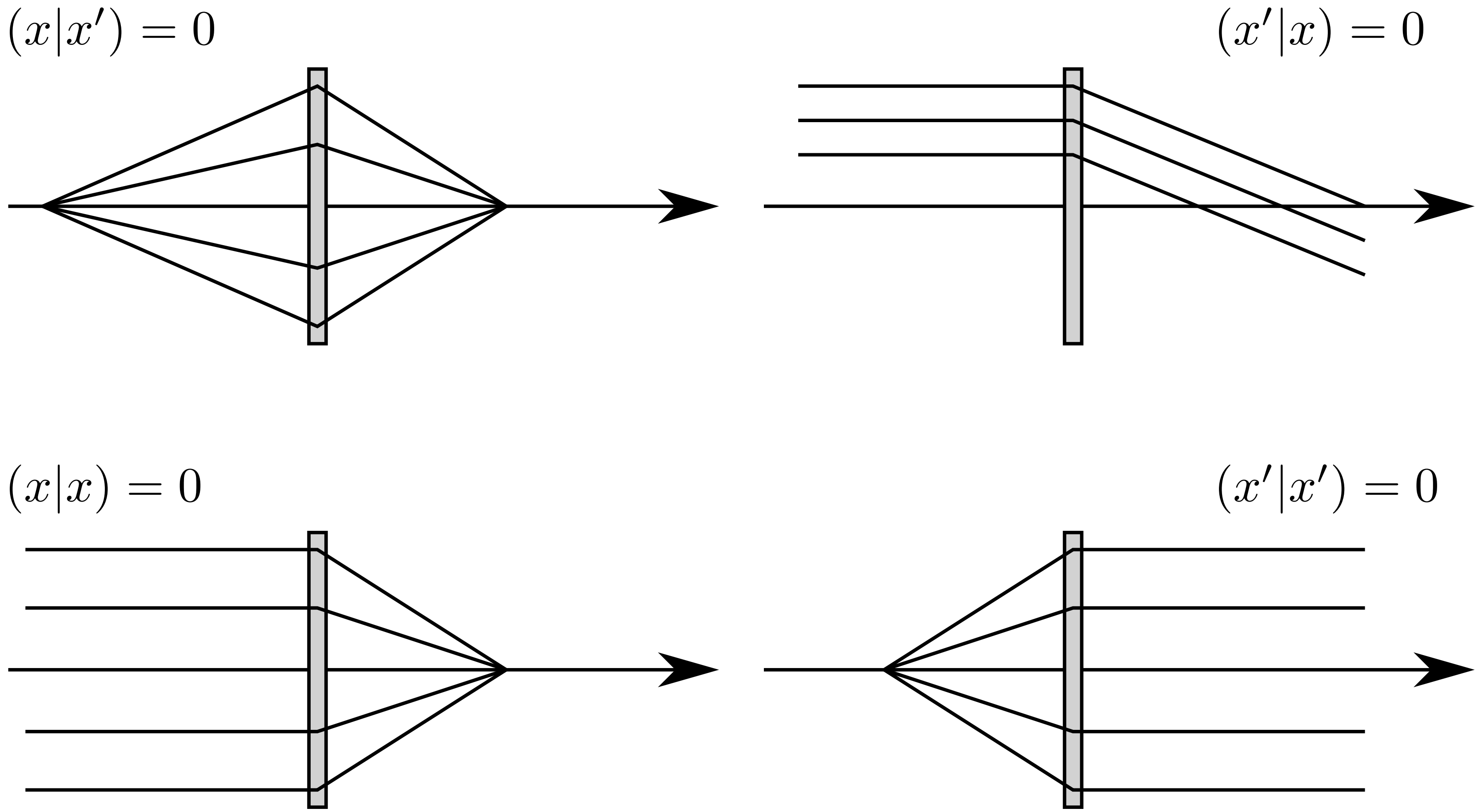
$$\begin{pmatrix} x_2 \\ x_2' \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}. \tag{30}$$

What are these $a_{i,j}$ *transfer coefficients*? Often used nomenclature uses $(x|x')$ to mark the effect of x' to x. The whole transport equation then becomes

$$\begin{pmatrix} x_2 \\ x_2' \end{pmatrix} = \begin{pmatrix} (x|x) & (x|x') \\ (x'|x) & (x'|x') \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}. \tag{31}$$

Meaning of matrix elements

Useful special cases, where one of the elements is zero



Back to the lens system

Requiring point-to-point focusing, $(x|x') = 0$ on matrix Eq. (29):

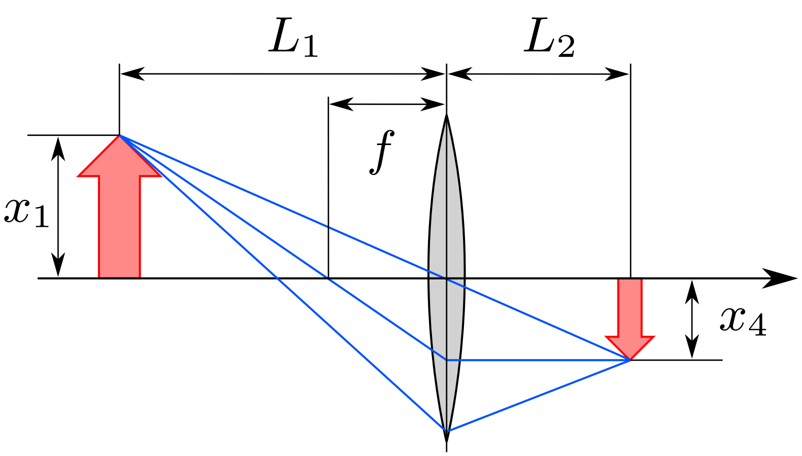
$$\begin{pmatrix} x_4 \\ x_4' \end{pmatrix} = \begin{pmatrix} 1 - L_2/f & L_1 + L_2 - L_1 L_2/f) \\ -1/f & 1 - L_1/f \end{pmatrix} \begin{pmatrix} x_1 \\ x_1' \end{pmatrix}.$$

is

$$L_1 + L_2 - L_1 L_2/f = 0, \tag{32}$$

which can also be written more familiarly as

$$\frac{1}{L_1} + \frac{1}{L_2} = \frac{1}{f}, \tag{33}$$



i.e. *the lens equation*. In that case, the first line of the matrix equation

$$x_4 = 1 - \frac{L_2}{f} x_1 \tag{34}$$

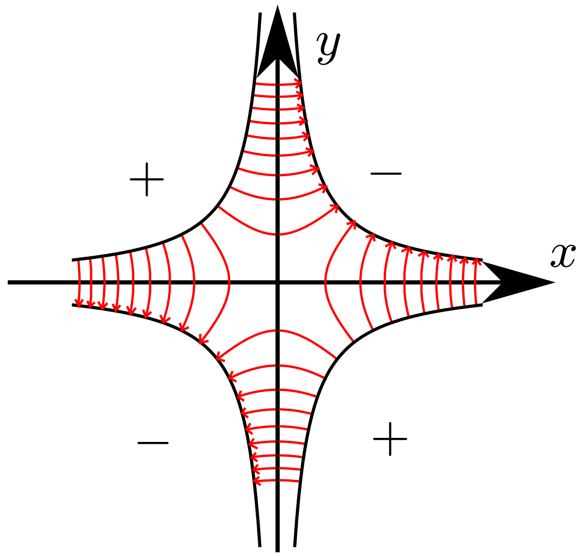
gives the magnification of the lens system.

Ion optical elements

- Consists of electric and magnetic fields
- Particle trajectories from Lorentz force: $\vec{F} = q(\vec{E} + \vec{v} \times \vec{B})$
- Fields usually from a mathematically simple set and assumed to be confined within a finite length L from s_1 to s_2.
- Perform the needed operations: bending, focusing, accerating, separating, ...
- Majority of ion optical elements are so-called transverse field or *multipole* elements with $B_z = E_z = 0$. These you know as the *dipole*, *quadrupole*, *hexapole*, ...
- These perform simple operations and the matrices can be produced (to some degree) with pen and paper.

Taneli Kalvas

Quadrupole focusing



From ideal quadrupole fields:

$$B_x = B_0 \frac{y}{r_0} \tag{35}$$

$$B_y = B_0 \frac{x}{r_0} \tag{36}$$

First approximation: velocity in z-direction

$$\vec{F} = q\vec{v} \times B = qv_z(B_x\hat{y} - B_y\hat{x}) = qv_z \frac{B_0}{r_0}(y\hat{y} - x\hat{y}) \tag{37}$$

Linear dependency on position just like a thin lens!

Focusing in x, defocusing in y (or other way round if $B_0 < 0$)!

Quadrupole transfer matrix

Starting from the equations of motion

$$m\frac{d^2x}{dt^2} = -qv_z\frac{B_0}{r_0}x \qquad \text{and} \qquad m\frac{d^2y}{dt^2} = qv_z\frac{B_0}{r_0}y, \tag{38}$$

changing variables with the chain rule

$$\frac{d^2x}{dt^2} = \frac{d^2x}{dz^2}\frac{d^2z}{dt^2} = \frac{d^2x}{dz^2}\left(\frac{dz}{dt}\right)^2 = x''v_z^2 \tag{39}$$

one arrives at

$$x'' = -k^2x \qquad \text{and} \qquad y'' = k^2y, \tag{40}$$

where

$$k^2 = \frac{qB_0}{r_0mv_z} \tag{41}$$

For positive k^2 the solution is

$$x(z) = c_1\cos(kz) + s_1\sin(kz) \tag{42}$$

$$y(z) = c_2\cosh(kz) + s_2\sinh(kz) \tag{43}$$

Quadrupole transfer matrix

Further, by using the initial conditions for the trajectory (x, x', y and y') one can solve the unknown coefficients and arrives at

$$\begin{pmatrix} x_2 \\ x'_2 \end{pmatrix} = \begin{pmatrix} \cos(kL) & \frac{1}{k}\sin(kL) \\ -k\sin(kL) & \cos(kL) \end{pmatrix} \begin{pmatrix} x_1 \\ x'_1 \end{pmatrix} \tag{44}$$

$$\begin{pmatrix} y_2 \\ y'_2 \end{pmatrix} = \begin{pmatrix} \cosh(kL) & \frac{1}{k}\sinh(kL) \\ k\sinh(kL) & \cosh(kL) \end{pmatrix} \begin{pmatrix} y_1 \\ y'_1 \end{pmatrix}, \tag{45}$$

where L is the length of the quadrupole in z-direction.

Taneli Kalvas

Quadrupole focusing

Example on focusing in both directions with a Quadrupole doublet.

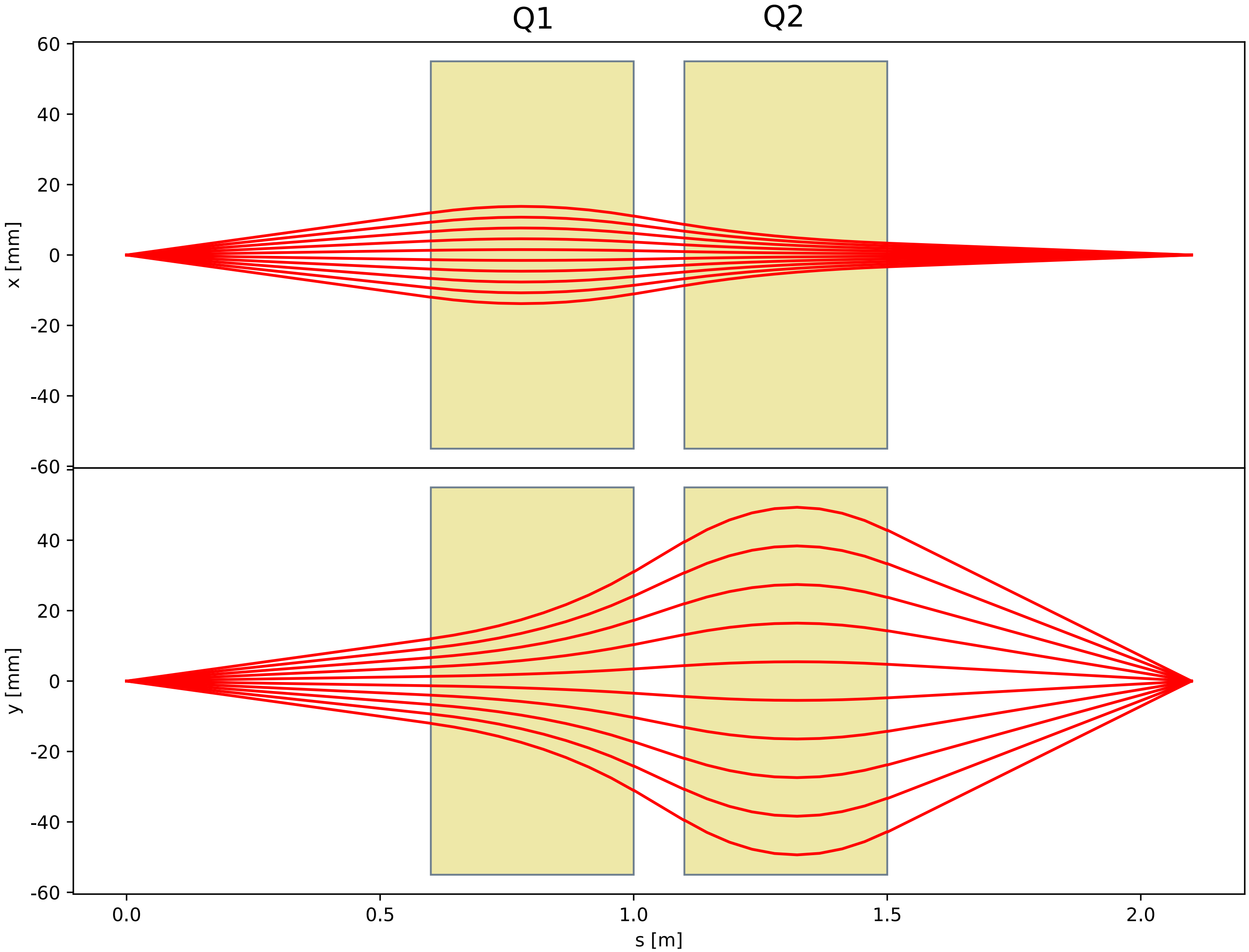
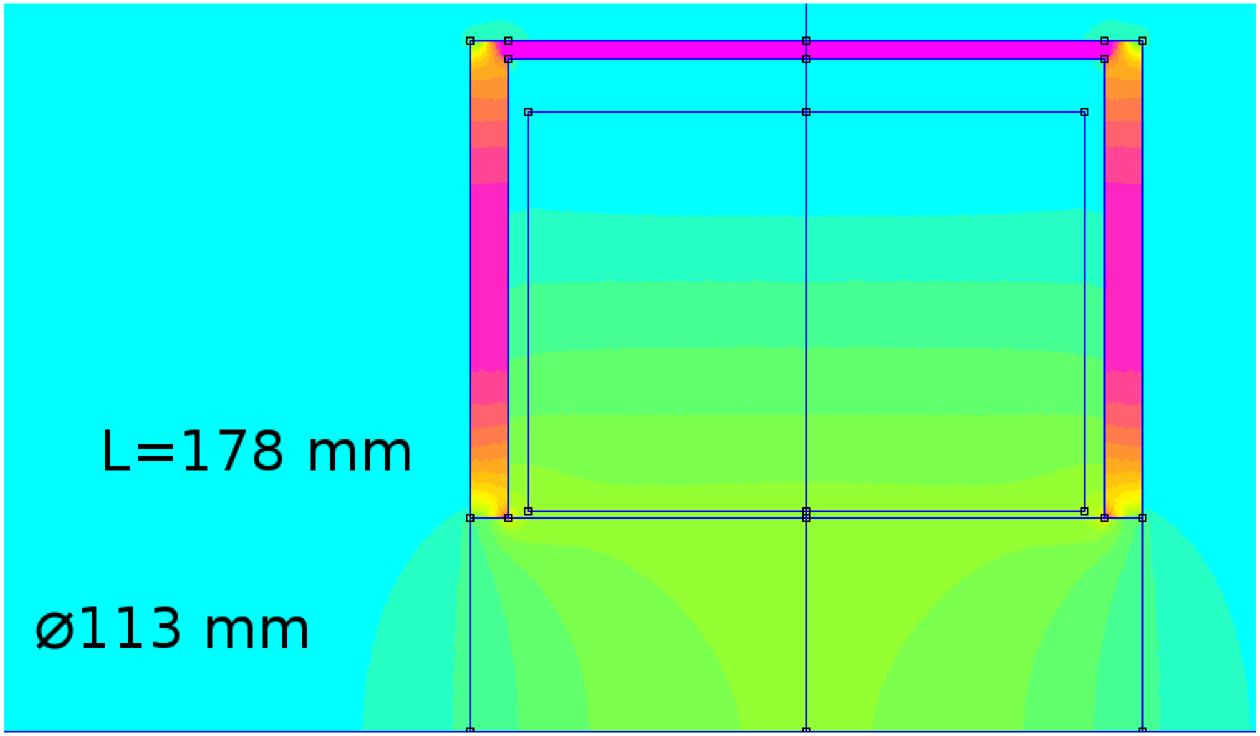


Image from Python Ion Optics Library, PIOL

Solenoid fitting example

JYFL injection line solenoid

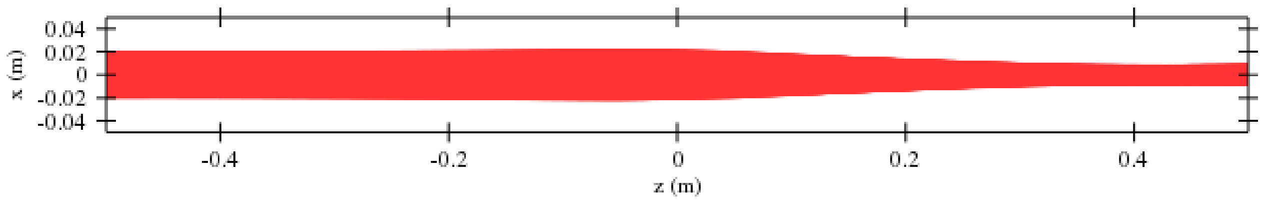
- Modelled with FEMM for 100 A induction current



Grid data output (using MATLAB script) within $z \in [0, 350]$, $r \in [0, 50]$
Data mirrored to fill $z \in [-350, 350]$.

Solenoid fitting example

IBSimu simulation to track particles through the magnetic field



Linear fitting to $(x_0, x'_0, y_0, y'_0) \rightarrow (x_1, x'_1, y_1, y'_1)$ produces a matrix

$$R = \begin{pmatrix} -0.16945 & 0.272915 & -0.169915 & 0.268352 \\ -1.75785 & -0.172501 & -1.72776 & -0.165878 \\ 0.169905 & -0.268383 & -0.169429 & 0.272935 \\ 1.72775 & 0.165847 & -1.75778 & -0.172471 \end{pmatrix}$$

This is a transfer matrix for drift + solenoid + drift.

Solenoid fitting example

Goal: coefficients L and B_0 for linear solenoid model $R_{\mathrm{sol}} =$

$$\begin{pmatrix} \cos(\phi)\cos(\phi) & \sin(\phi)\cos(\phi)/K & \sin(\phi)\cos(\phi) & \sin(\phi)\sin(\phi)/K \\ -\sin(\phi)\cos(\phi)K & \cos(\phi)\cos(\phi) & -\sin(\phi)\sin(\phi)K & \sin(\phi)\cos(\phi) \\ -\sin(\phi)\cos(\phi) & -\sin(\phi)\sin(\phi)/K & \cos(\phi)\cos(\phi) & \sin(\phi)\cos(\phi)/K \\ \sin(\phi)\sin(\phi)K & -\sin(\phi)\cos(\phi) & -\sin(\phi)\cos(\phi)K & \cos(\phi)\cos(\phi) \end{pmatrix},$$

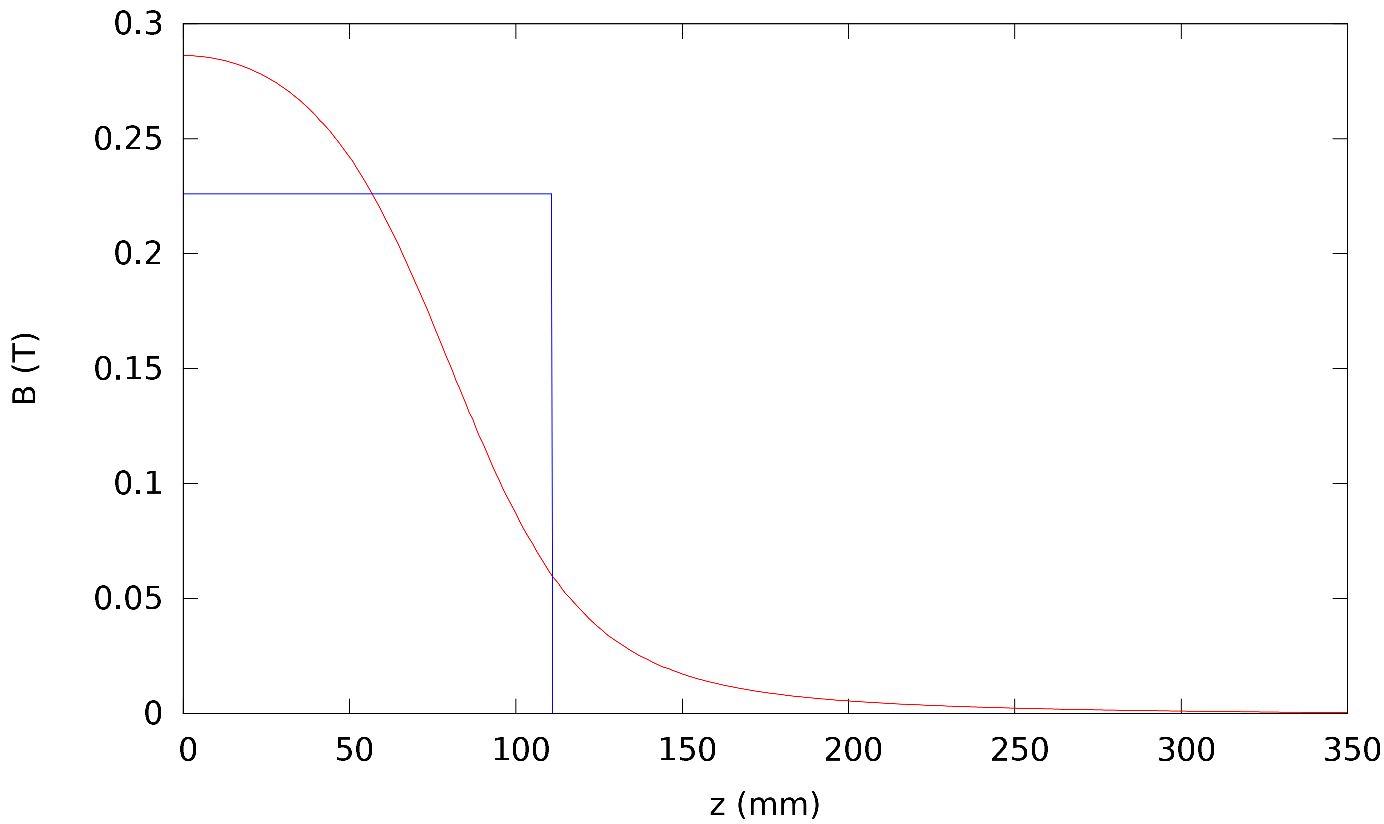
where $\phi = \frac{1}{2}B_0L/B_r$ and $K = \phi/L$.

Fitting of drift$(0.5\text{ m} - 0.5L)$ + sol(B_0, L) + drift$(0.5\text{ m} - 0.5L)$ produced coefficients

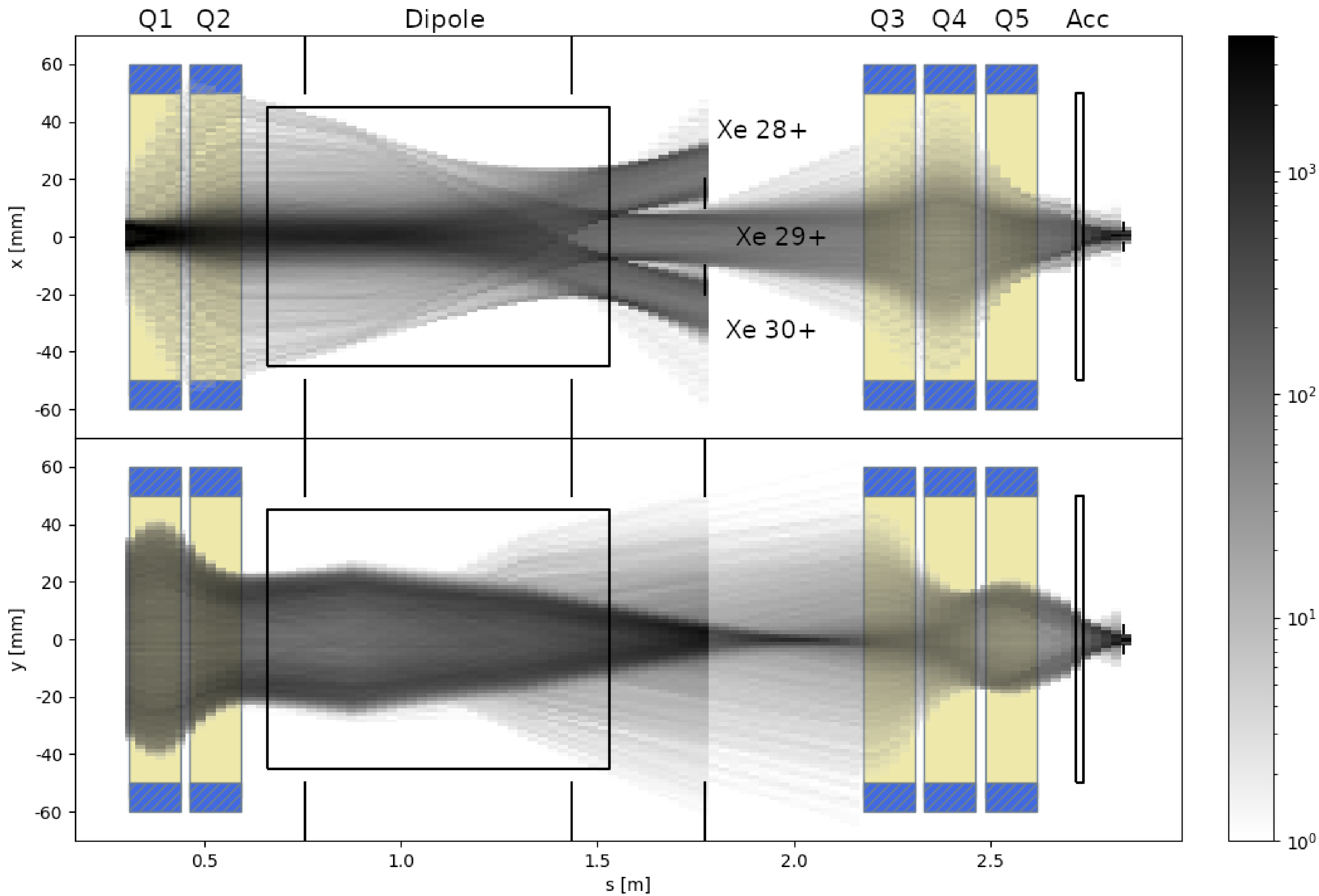
$B_0 = 0.226$ T
$L = 0.222$ m

Solenoid fitting example

Resulting linear solenoid model vs. real field on axis



Transfer matrix example: IBSimu continuation



Traditional transfer matrix optics

- Matrices based on analytic formulation, numerical integration of fields or fitting experimental/simulation data.
- The whole system can be described with one matrix:

$$R_{\text{system}} = R_N \cdots R_2 \cdot R_1$$

- Can also transport elliptical envelopes in addition to trajectories:

$$\sigma_1 = R\sigma_0 R^T, \text{ where}$$
$$\sigma = \epsilon \begin{pmatrix} \beta & -\alpha \\ \alpha & \gamma \end{pmatrix}$$

- Advantage: calculation is fast (automatic optimization, etc)
- May include additional space charge induced divergence growth for beam envelopes and/or rms emittance growth modelling for particle distributions.

Codes of this type

- TRANSPORT — One of the classics, up to 2nd or 3rd order calculation, no space charge
- COSY INFINITY, GICOSY — Up to infinite order, no space charge
- GIOS — Up to 3rd order, space charge of KV-beam
- DIMAD — Up to 3rd order, space charge of KV-beam
- TRACE-3D — Mainly linear with space charge of KV-beam
- PATH MANAGER (TRAVEL) — Up to 2nd order, more advanced space charge modelling for particle distributions (mesh or Coulomb)
- PIOL — Python driven Ion Optics Library, infinite order, space charge, Monte Carlo capabilities. Customizable, open source.

Some of the codes are more suitable for low energies, choose carefully!