

3D Field Response Simulation

3D Field Response calculation

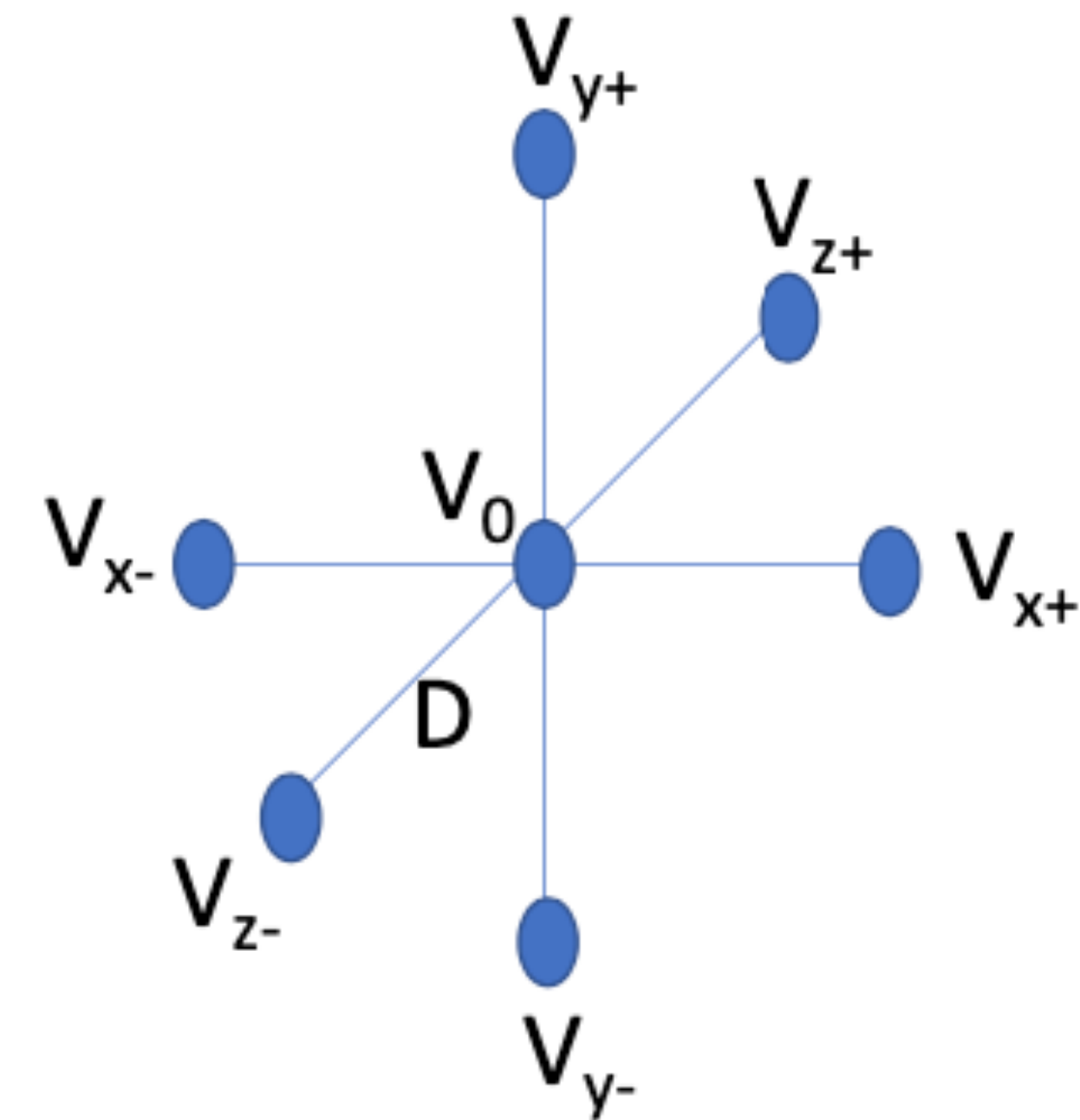
- Move from Wires to PCB planes in Liquid Argon detector design requires 3D field response calculation;
- Several packages, such as COMSOL, can provide the required simulation; however, the goal is to make faster and easier to use package;
- The idea is inspired by Fortran based simulation provided by Francesco for vertical drift detector;

Strategy

- Solve 1st Maxwell equation on a 3D lattice both for the electric field and the weighting potentials (collection/indiction) using Finite Difference Method

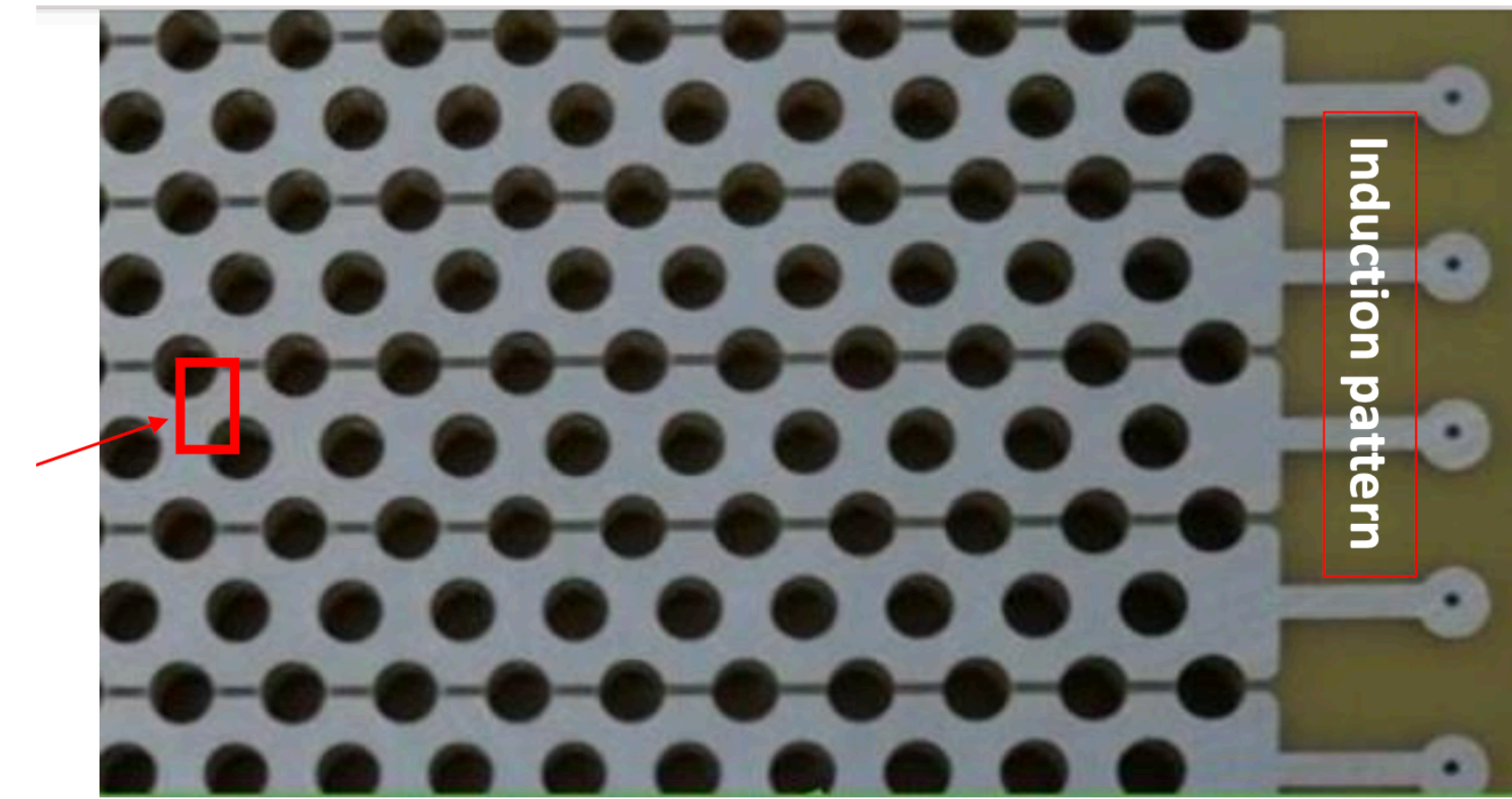
$$V_0 = \frac{(V_{x+} + V_{x-} + V_{y+} + V_{y-} + V_{z+} + V_{z-})}{6}$$

- To minimize computing time:
 - Use python matrix calculation
 - Use python GPU-based tensors
 - Find minimal symmetry volume for 3D drift and switch to 2D calculation at the boundaries of 3D weighting field
- No edge effects included (eg: field cage, PCB is considered as infinite)

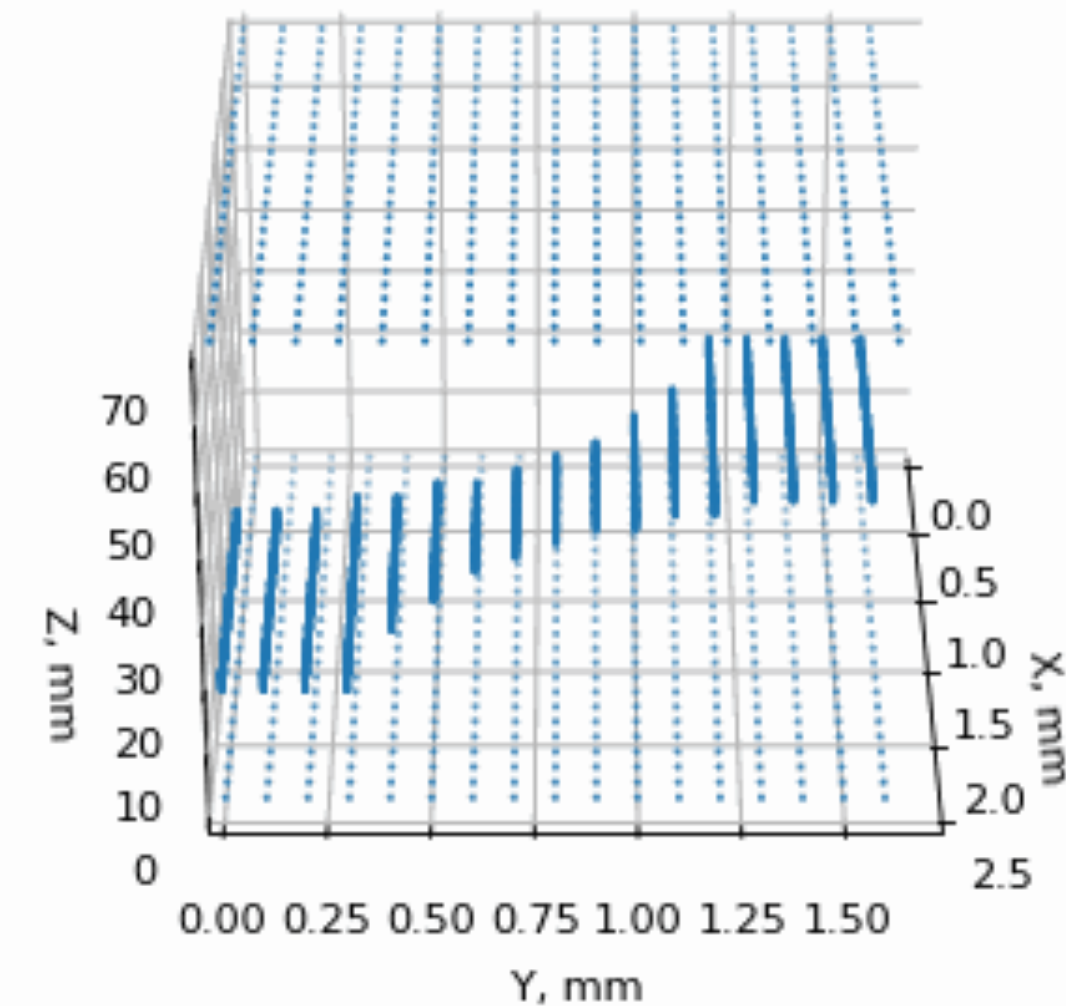


3D Drift field

- Minimal symmetry for 3D drift calculation:
 - Quarter of pcb strip (Area 2.5mm x 1.67mm) with hole diameter ~2.5mm;
- Boundary Conditions:
 - Plane at 2000mm : -10000V;
 - Plane at 0mm : 0V;
 - Induction plane at 20mm : 0V;
 - Collection plane at 17mm : 2000V;



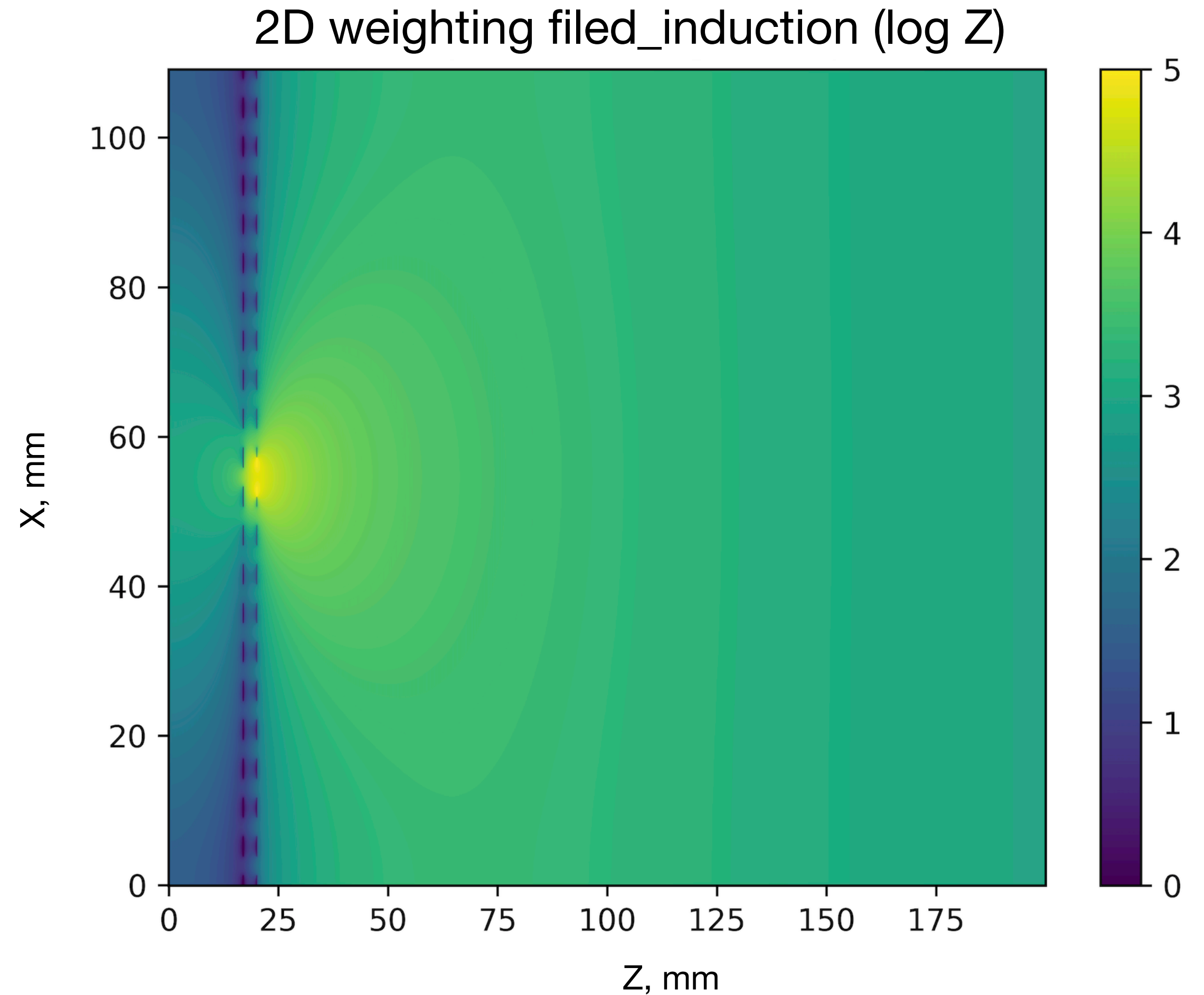
store/boundary/drift3d.npz



Boundary Conditions Example

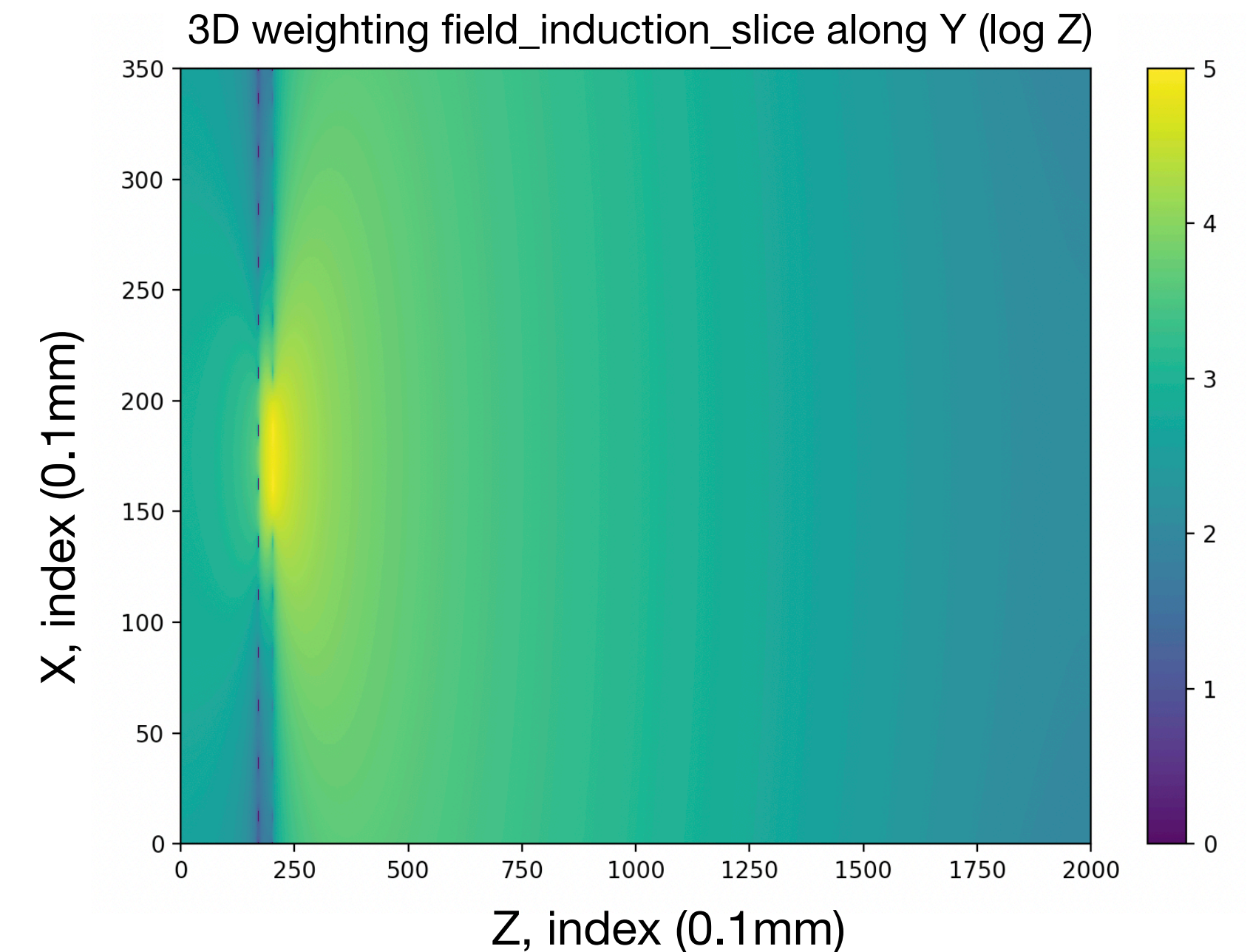
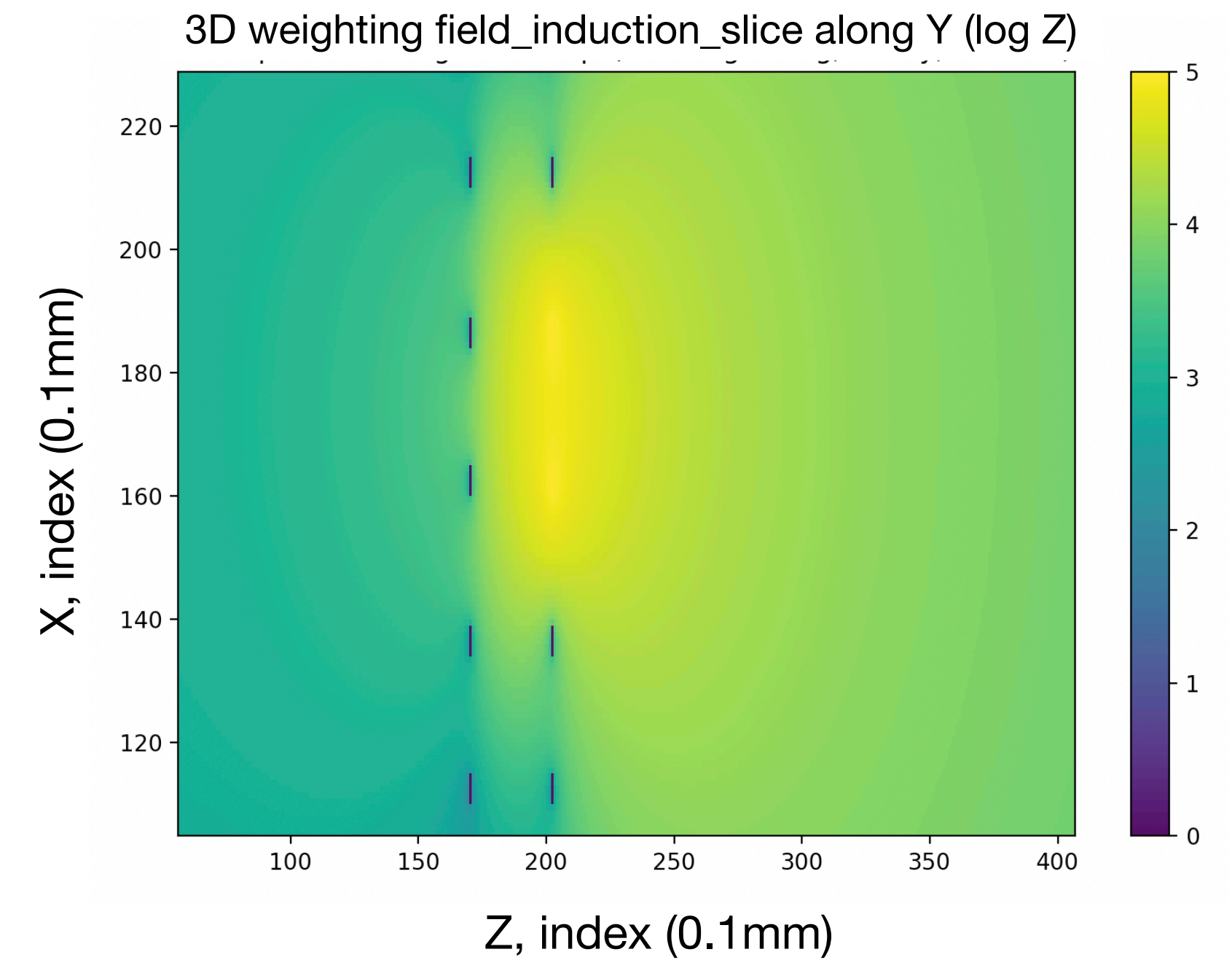
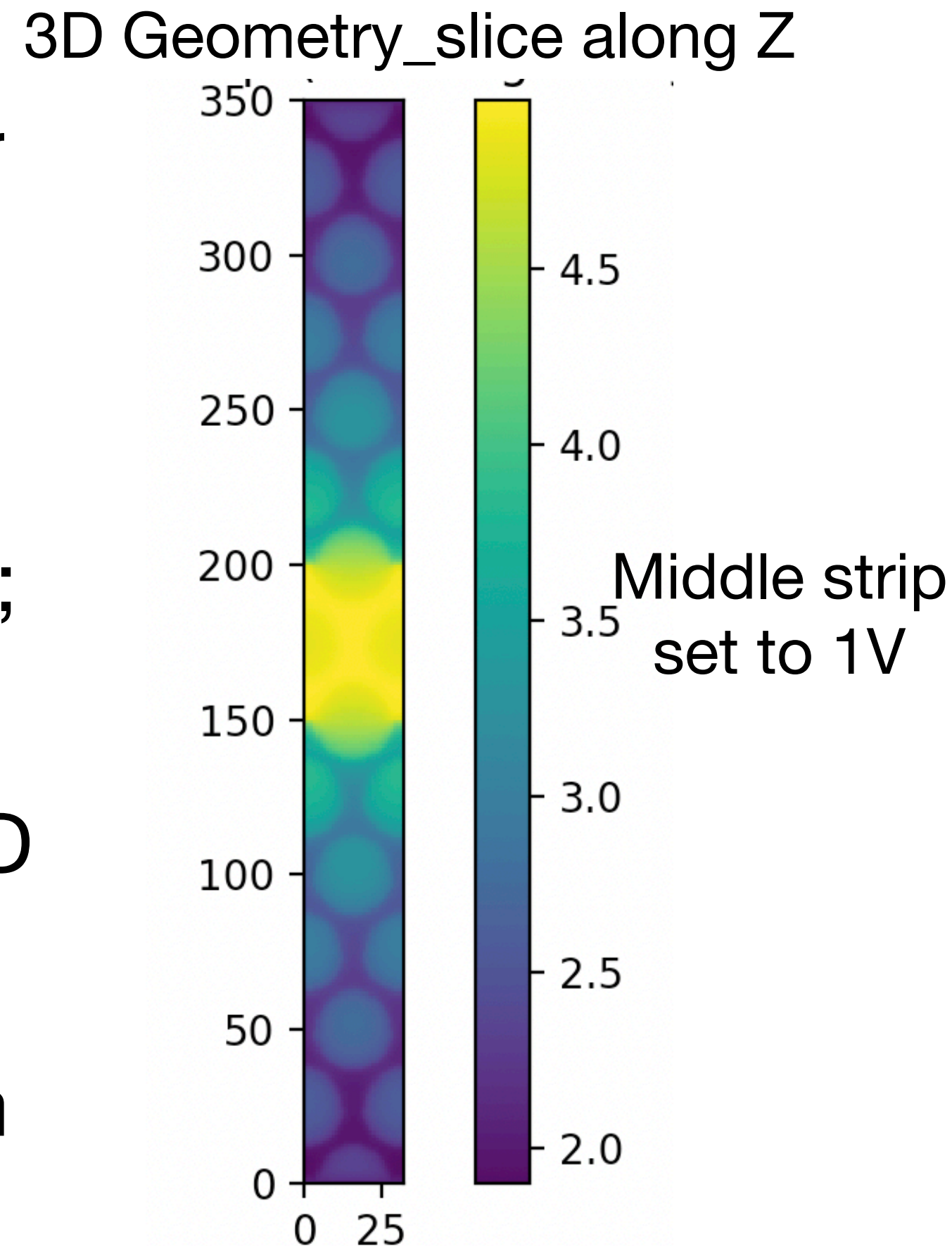
2D weighting field

- Defined 2D geometry for weighting field calculation for 11 strips :
 - 1 middle strip at 1V ;
 - 10 strips at 0V on each side
- Calculate weighting field using 2D FDM with 0.1mm grid:
 - Grid : 1092x2000



3D weighting field

- Defined 3D geometry for weighting field calculation for 7 strips:
 - 1 middle strip at 1V;
 - 3 strips at 0V on each side;
- 2D solution applied as a boundary condition for the 3D problem at ± 3 strips
- Solve using FDM with 0.1mm grid spacing:
 - Grid 350x34x2000



FDM Calculation

- Total of 3 FDM simulations:

- 3D drift field
- 2D weighting field
- 3D weighting field (with 2D case as boundary condition)

- FDM need on two inputs:

- Number of steps to cover all volume
- Number of epochs to do the solution refinement

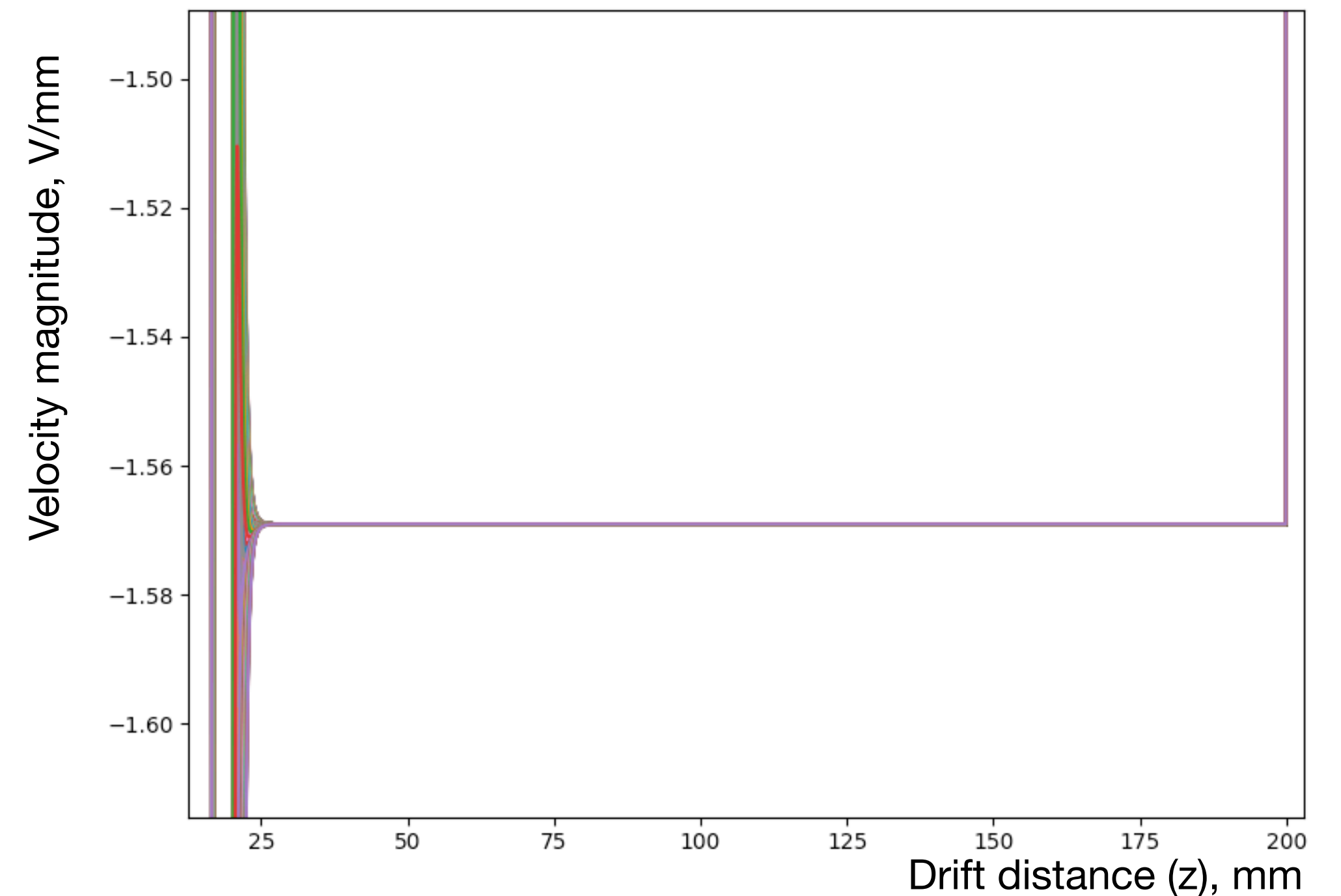
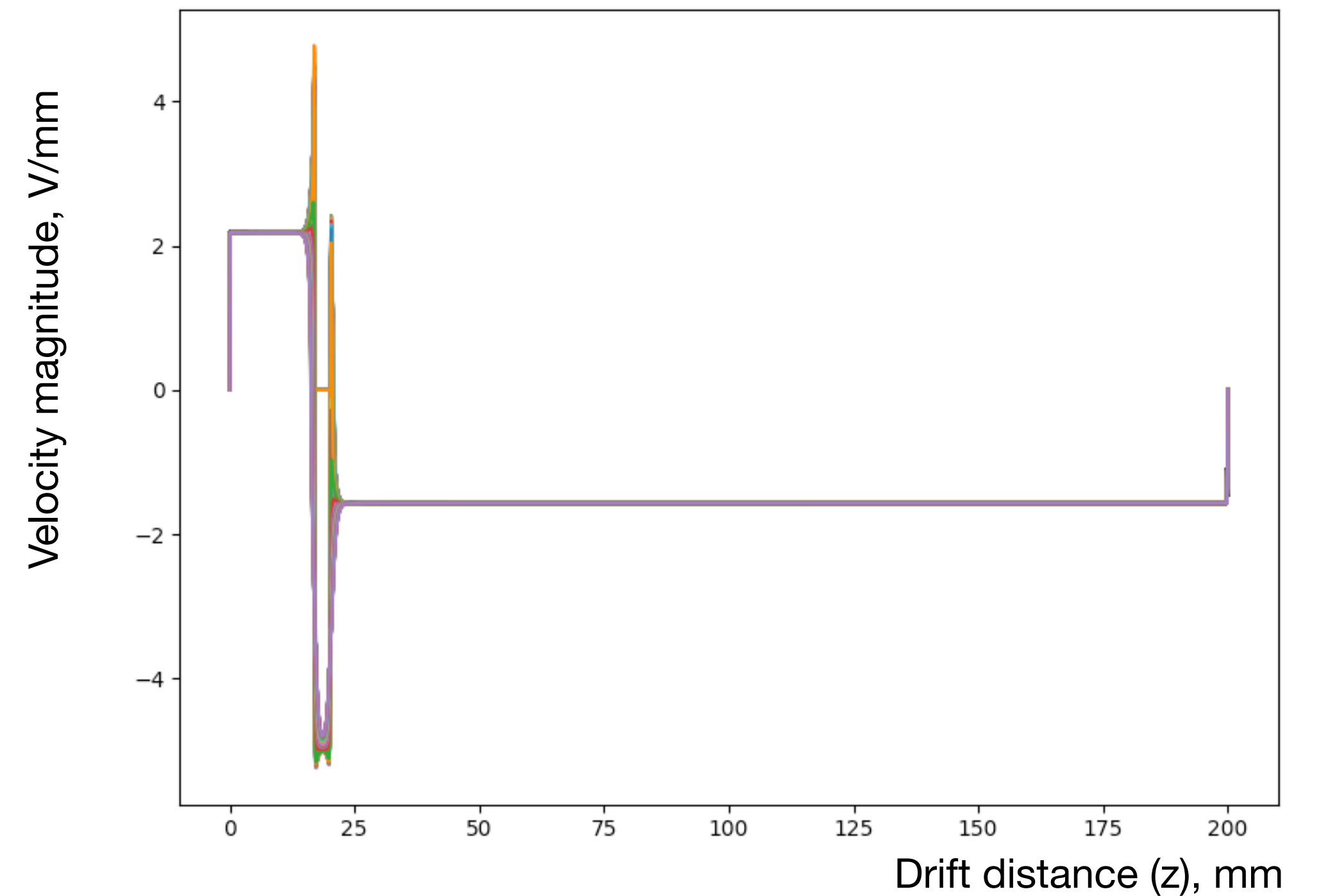
Grid Points (# of epochs/steps)	GPU (PyTorch tensors)
3D drift: 25x17x2000 (8ep, 1000000 steps)	~40min
2D weighting: 1092x2000 (4ep, 1250000 steps)	~44min
3D weighting: 350x34x2000 (2ep, 1000000 steps)	~212min

Velocity in 3D drift volume

- Check Velocity magnitude through simulated 3D drift volume, fix (x,y) coordinates and follow vertical path(z axis) :
 - X axis on the plots correspond to vertical(Z axis) in 3D drift volume;
 - Different color lines are velocity through with different fixed (x,y) coordinates
- Uniform 1.57 V/mm velocity through the drift volume

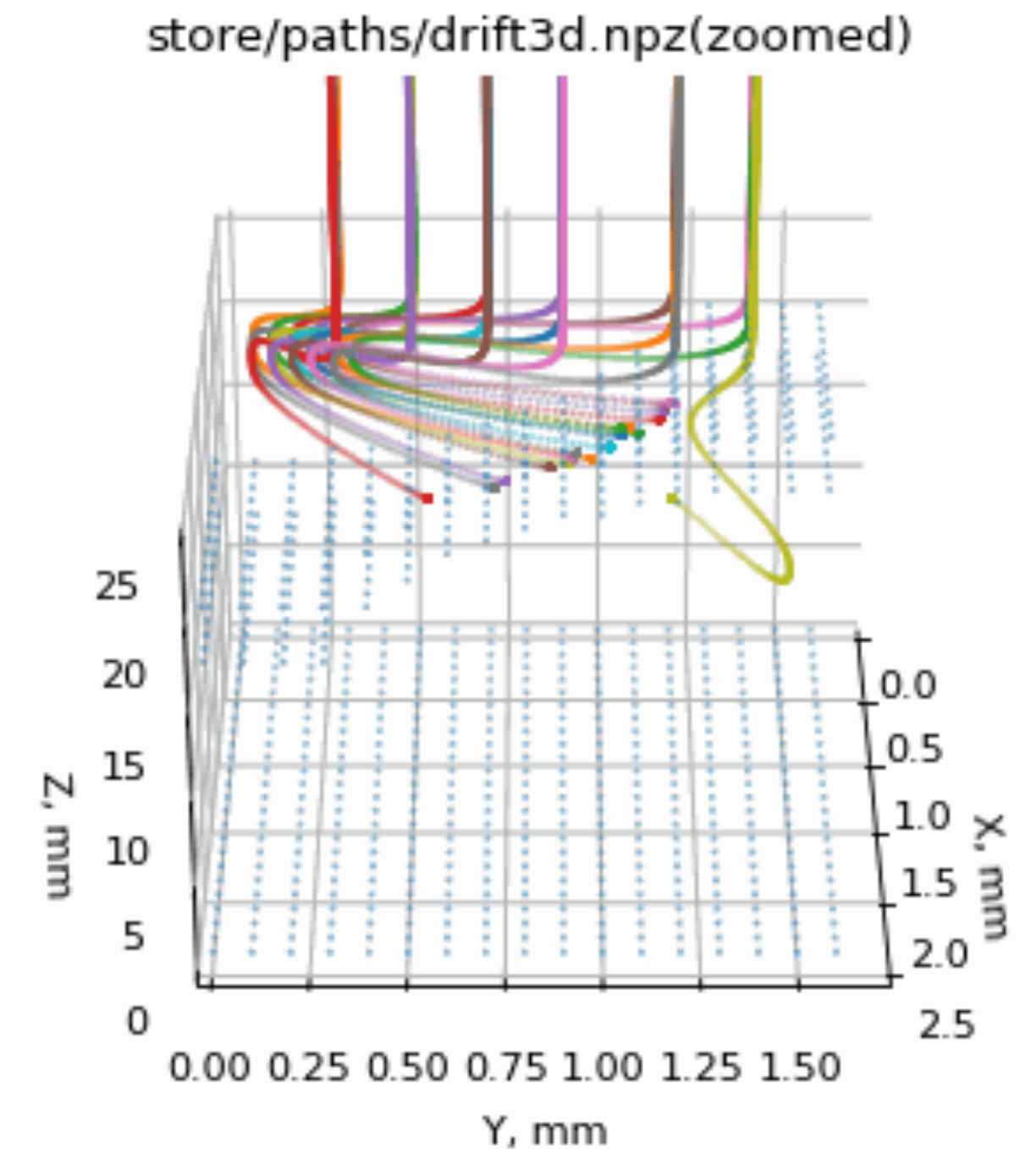
- Boundaries (zero field and velocity):

- 200mm(anode)
- 17mm(collection)
- 20mm(induction)
- 0mm(ground)



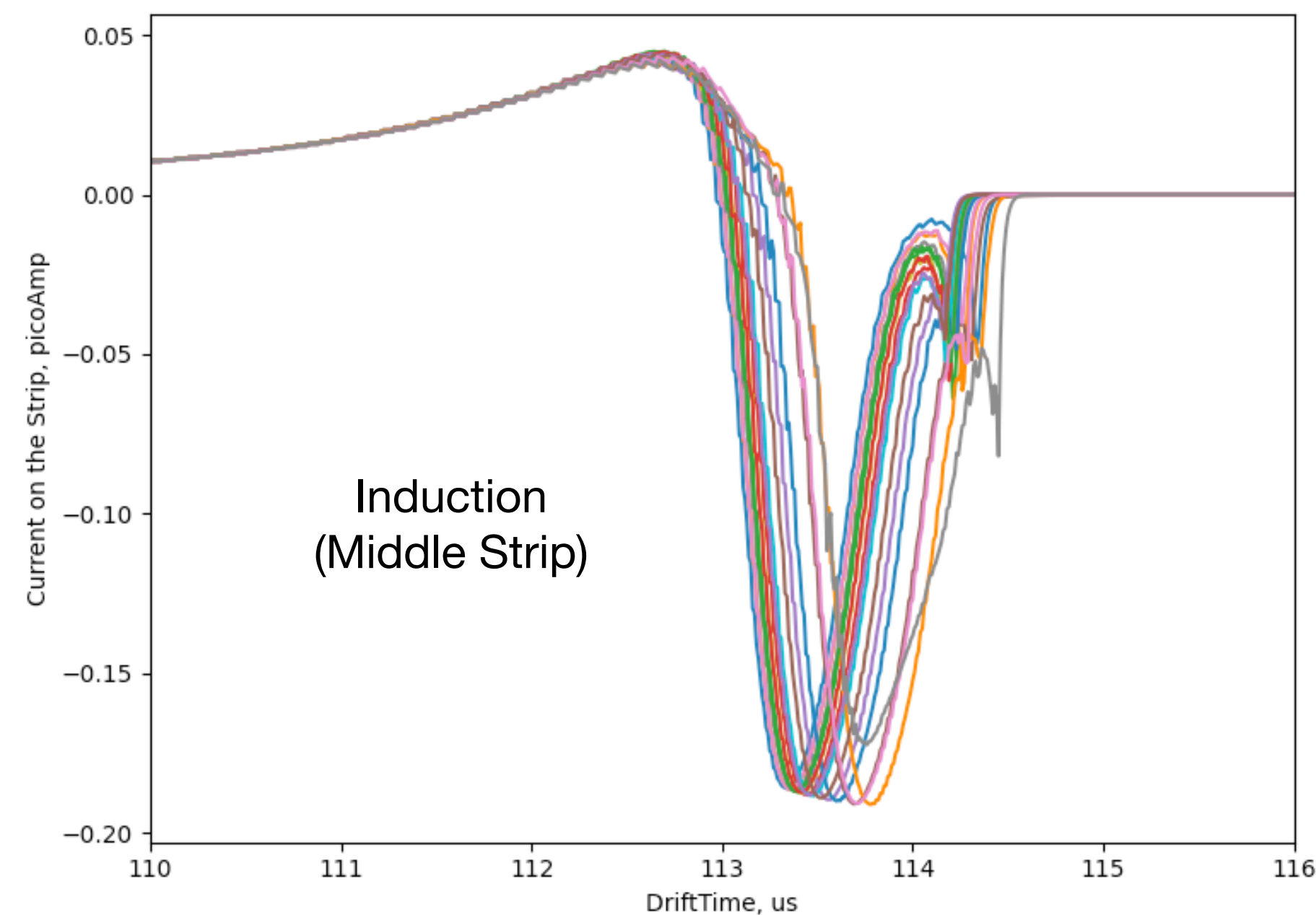
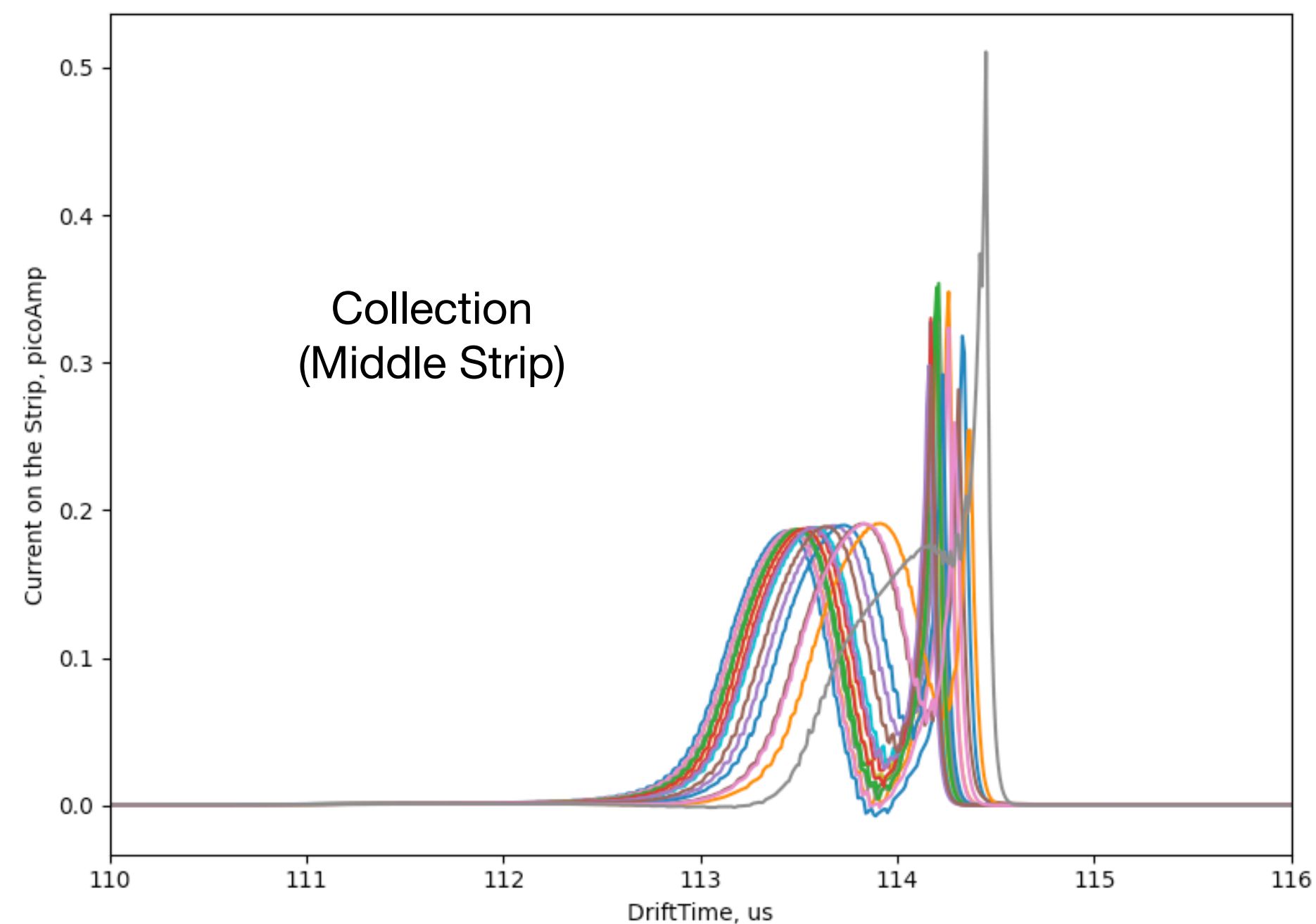
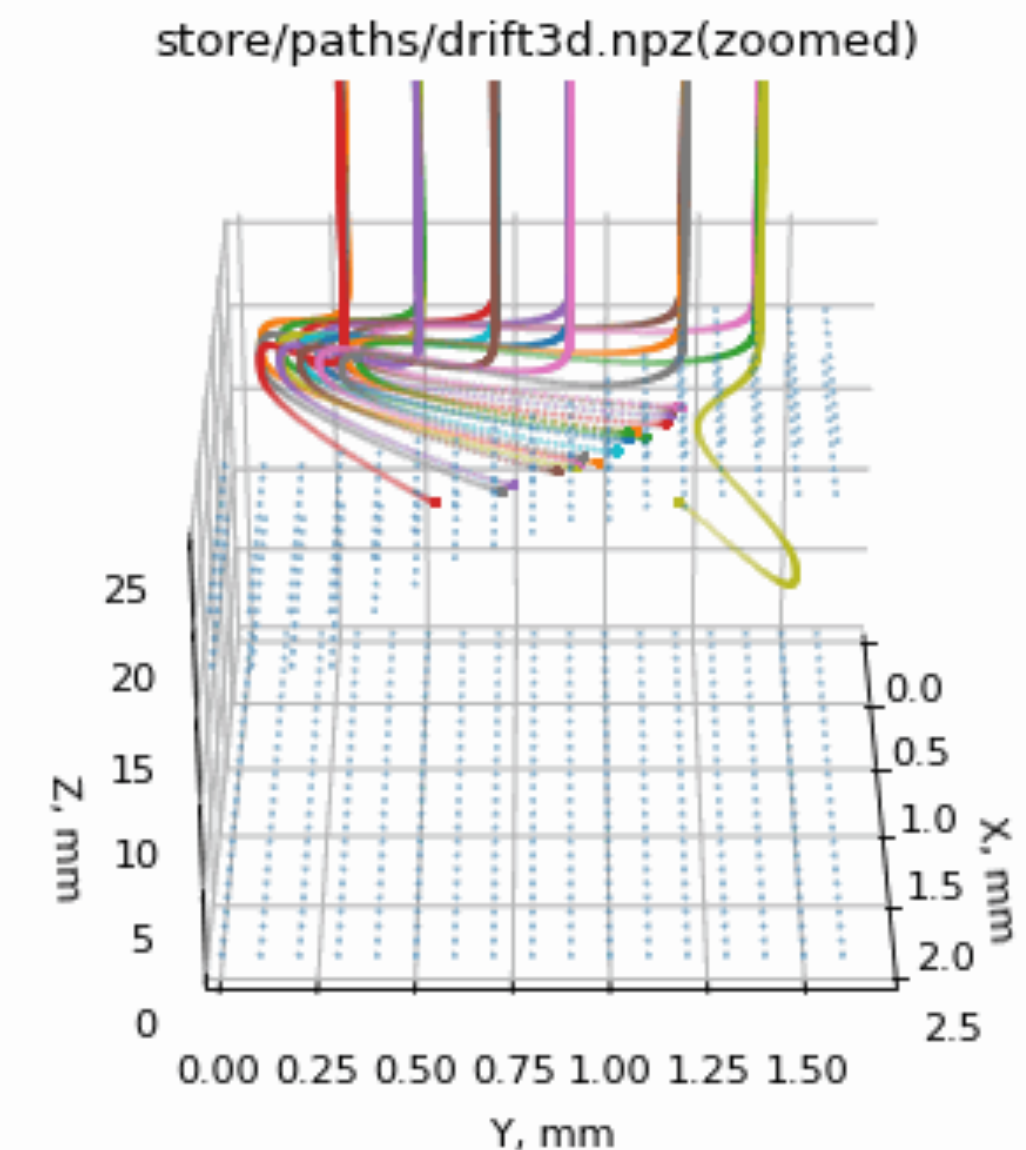
Drift Paths

- Drift paths are calculated using Runge-Kutta implementation in scipy;
- Choose paths to check behavior near the volume boundary and near PCB;
- Starting less than 0.2mm from the boundary makes electron to go out of boundaries.



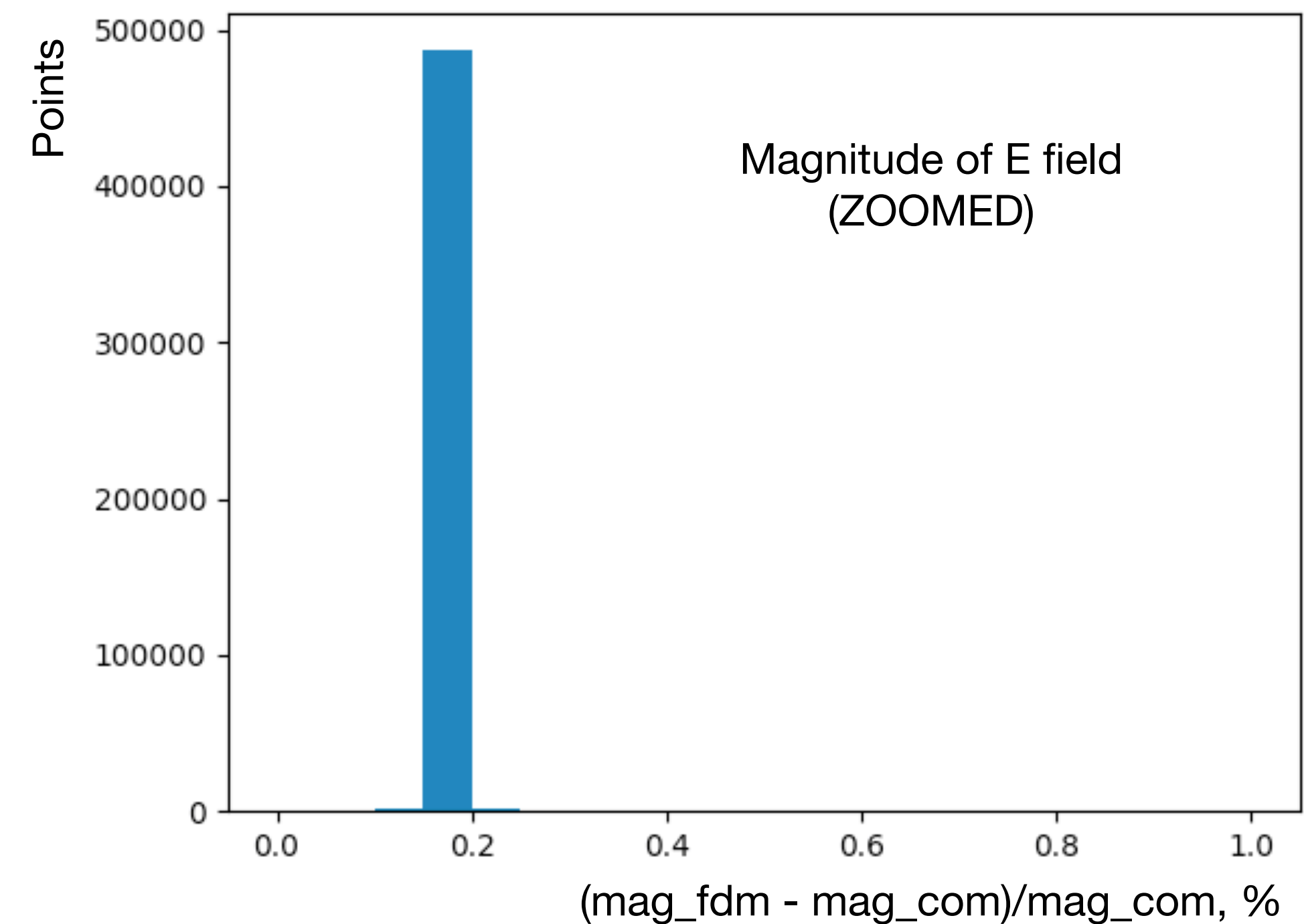
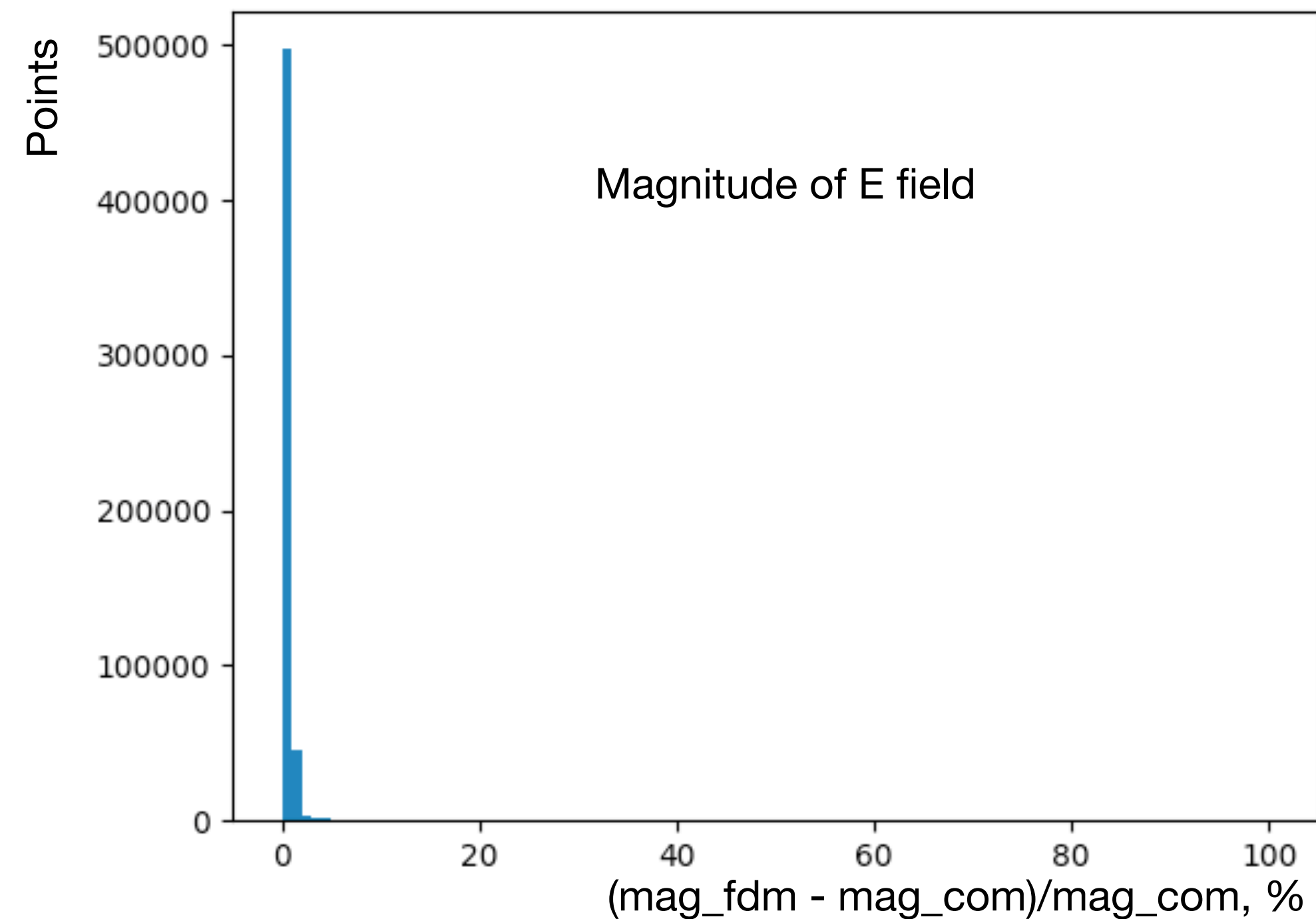
Induced current/Charge

- Total charge simulated $\sim 1e$
- Check total charge induced on collection plane:
 - IntegralCharge= 0.9999227581305705 IntegralCharge= 0.9999227582489765
 - IntegralCharge= 0.9999227582414939 IntegralCharge= 0.9999227581663778
 - IntegralCharge= 0.999922758247561 IntegralCharge= 0.999922758247545
 - IntegralCharge= 0.9999227646976012 IntegralCharge= 0.9999227646992445
 - IntegralCharge= 0.9999227646989095 IntegralCharge= 0.9999227646994582
 - IntegralCharge= 0.9999227646999055 IntegralCharge= 0.9999227646764435
 - IntegralCharge= 0.9999227729712071 IntegralCharge= 0.9999227735210839
 - IntegralCharge= 0.9999227737381025 IntegralCharge= 0.9999227737382792
 - IntegralCharge= 0.9999227737096039 IntegralCharge= 0.9999227737369193



Validation with COMSOL

- 3D drift field COMSOL simulation provided by Yichen
- Compare drift Efield
- Initial validation excludes 0.1mm regions from all boundaries (including pcb);
- Just 0.2% difference!



Next Steps

- Validation with current vertical drift simulation
- Implementing convolution function with detector response
- Validation with the data from vertical drift?