Tracking Updates

Norman Graf
SLAC
September 5, 2013

Magnetic Field in slic

- Currently using ideal dipole field
  - constant B_Y
- Have 2D map of quarter of field (+,+)
  - Only provides B_Y
  - Extend by reflection to (+,-),(-,-),(-,+)
- Expanded into full 3D magnetic field map
  - (B_X,B_Y,B_Z) @ (x,y,z)
- Released in hps-detectors/fieldmap
  - HPS_b18d36_unfolded.dat
  - $(0,B_Y,0)$ @ (x,y,z)

Full 3D Field Map

- Input (B_X,B_Y,B_Z) @ (x,y,z) on a regular Cartesian Grid
- Use linear interpolation to determine field at an arbitrary point within a box represented by surrounding grid points.
- Both memory and compute intensive, but simple and straightforward.
  - Would prefer to have polynomial fit to the field.
- Classes written for use in slic (C++) and lcsim (Java).
- GeomConverter code available to incorporate into lcdd and lcsim.
- Field map is defined in magnet coordinates.
  - Able to translate field in compact description

Field Map usage in compact.xml

```
<field
        type="FieldMap3D“
         name=“HPSDipoleFieldMap3D"
         filename="HPS_b18d36_unfolded.dat"
         offsetX="1.0"
         offsetY="2.0"
         offsetZ="3.0“
/>
```

Work done in collaboration with Jeremy McCormick.

Magnetic Field in org.lcsim

- Work ongoing to fully incorporate 3D magnetic field map in org.lcsim reconstruction.
- Magnetic field map handling (reading, interpolating, caching, etc.) implemented.
- A fourth-order Runge-Kutta stepper with adaptive step size has been implemented.
  - Tested RK propagation against helical propagation in constant field.
  - Tested in piece-wise constant test fields where analytic solution can be calculated.
  - Work ongoing to compare with G4 stepper.

trf SurfZPlane

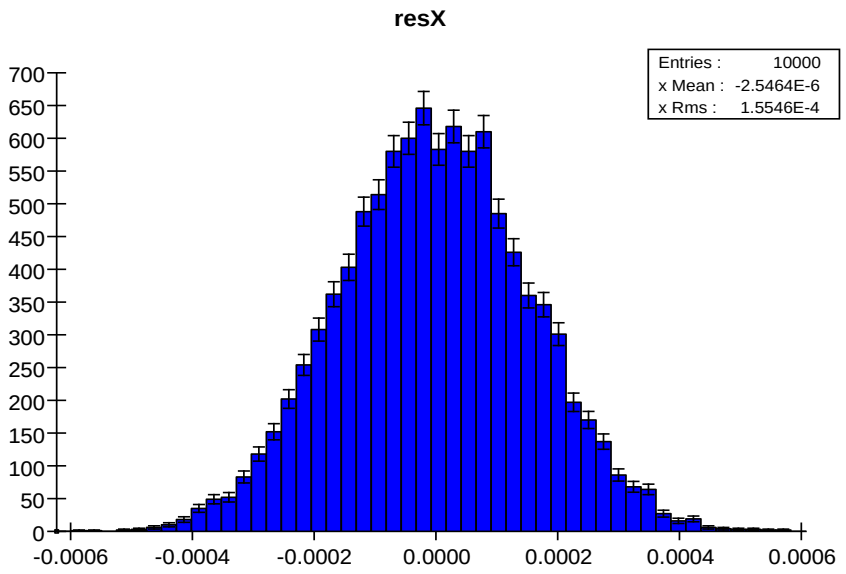
- Defined as a plane perpendicular to the z axis
  - BSurfZPlane adds boundaries to the surface.
- Track vector on this Surface defined as:

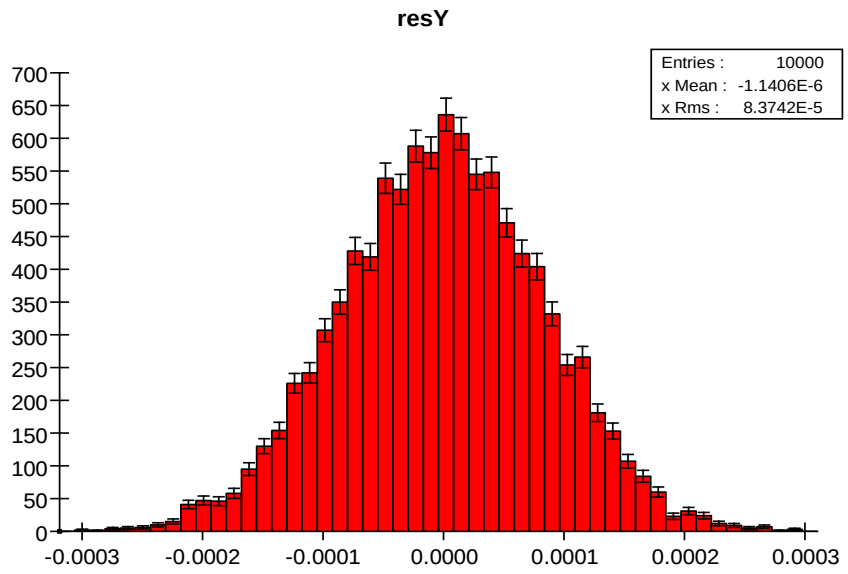
$$\begin{pmatrix} x \\ y \\ dx/dz \\ dy/dz \\ q/p \end{pmatrix}_{@z}$$

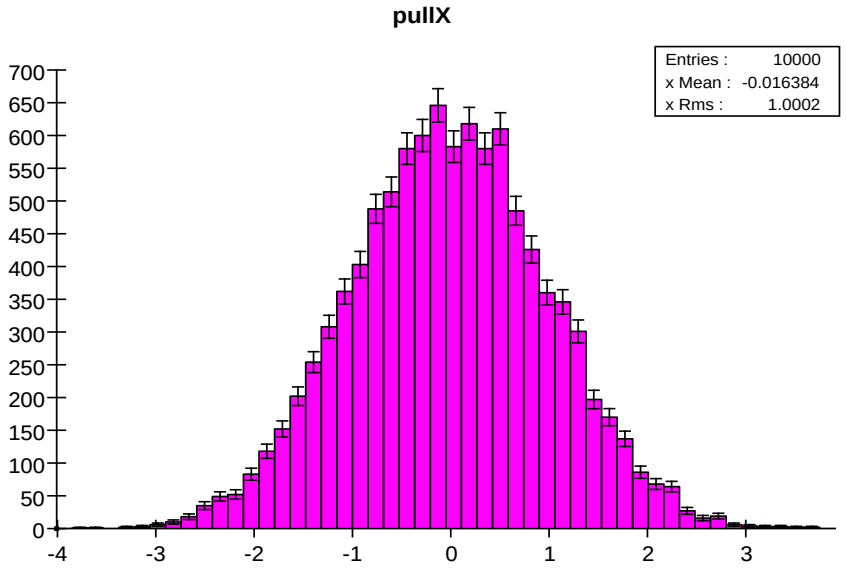
Standalone Fitting Tests

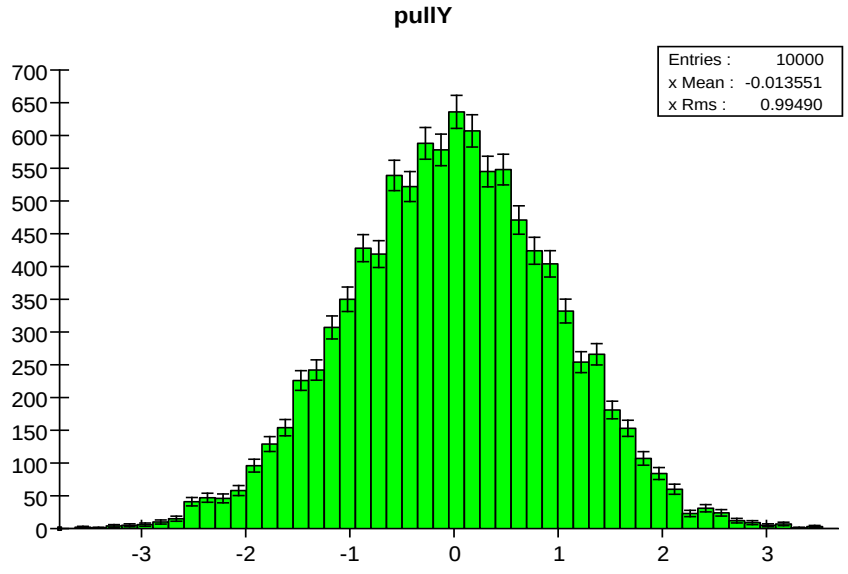
- Using simple five-station tracker with ten planes.
- Using constant B_Y magnetic field
- Generate tracks with flat track vector distributions.
- Propagate tracks to each plane, smear intercept to produce one-dimensional u and v hits (strips).
- Fit list of hits using Kalman Filter to get optimal fit at last plane
- Refit from last to first plane.
- Propagate fit to origin.
- Compare track parameters at origin.

Positions

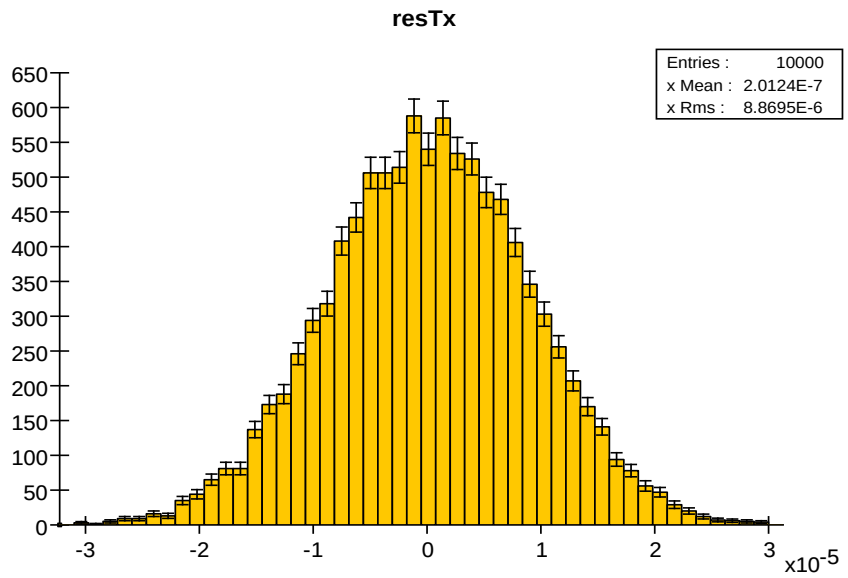

resX
Entries : 10000
x Mean : -2.5464E-6
x Rms : 1.5546E-4
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-0.0006
-0.0004
-0.0002
0.0000
0.0002
0.0004
0.0006

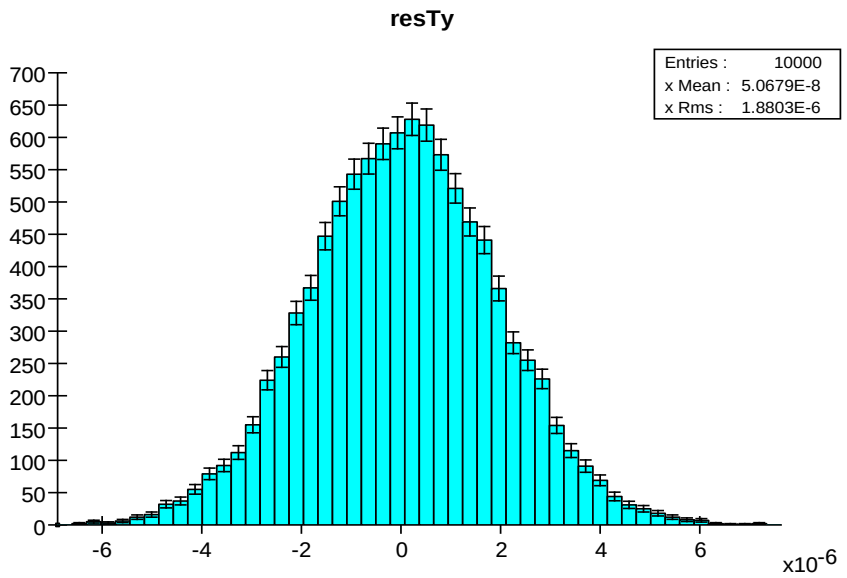

resY
Entries : 10000
x Mean : -1.1406E-6
x Rms : 8.3742E-5
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-0.0003
-0.0002
-0.0001
0.0000
0.0001
0.0002
0.0003

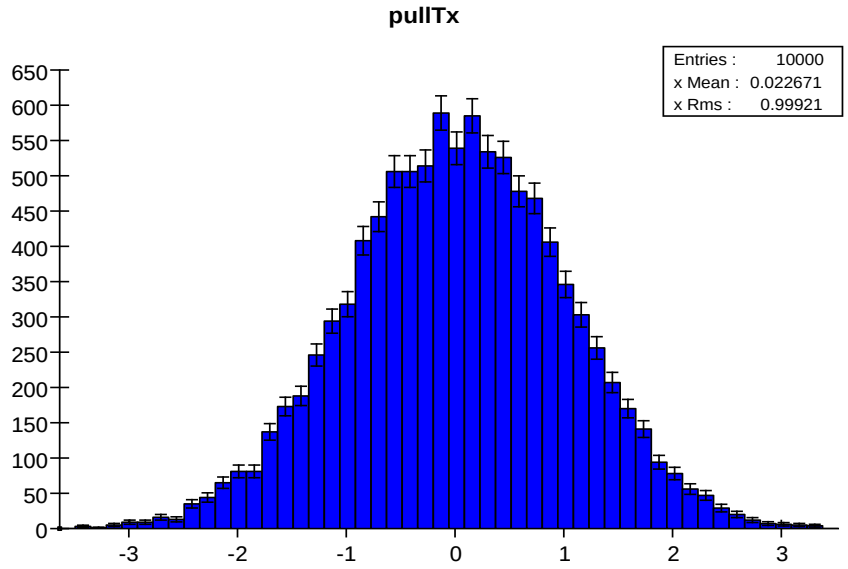

pullX
Entries : 10000
x Mean : -0.016384
x Rms : 1.0002
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-4
-3
-2
-1
0
1
2
3

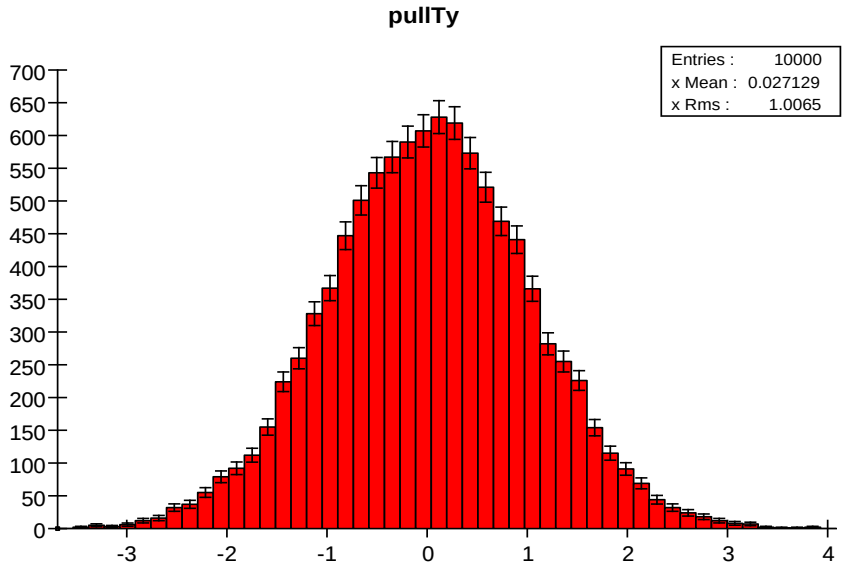

pullY
Entries : 10000
x Mean : -0.013551
x Rms : 0.99490
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-3
-2
-1
0
1
2
3

Slopes

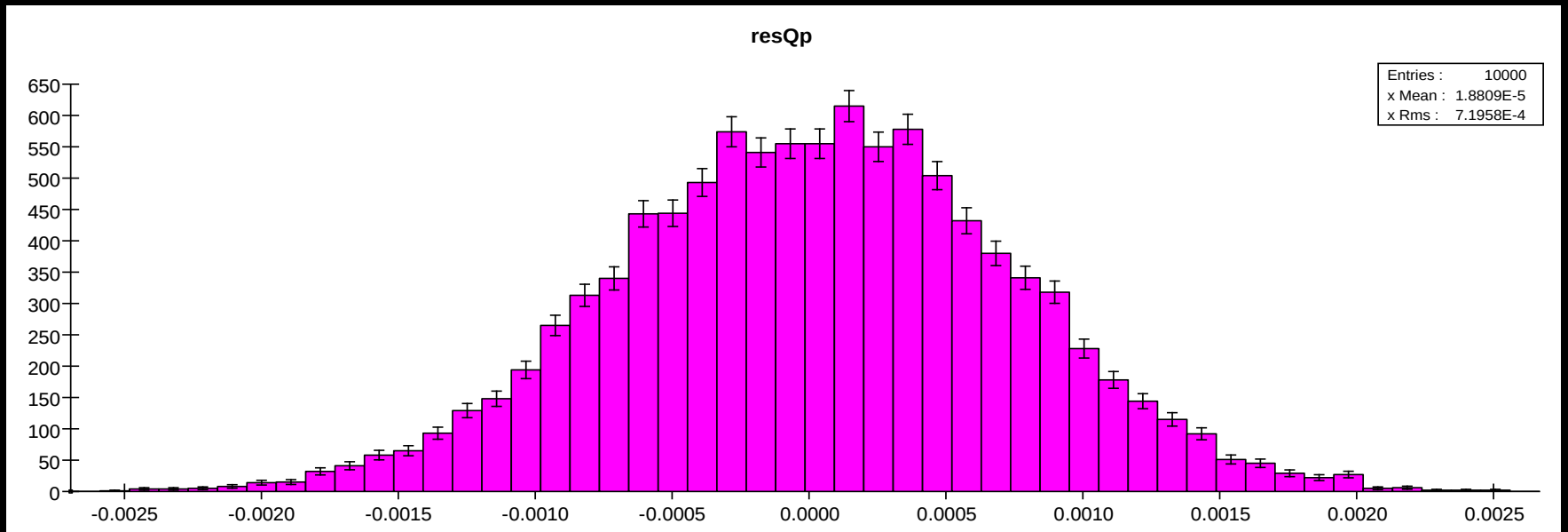

resTx
Entries : 10000
x Mean : 2.0124E-7
x Rms : 8.8695E-6
x10-5

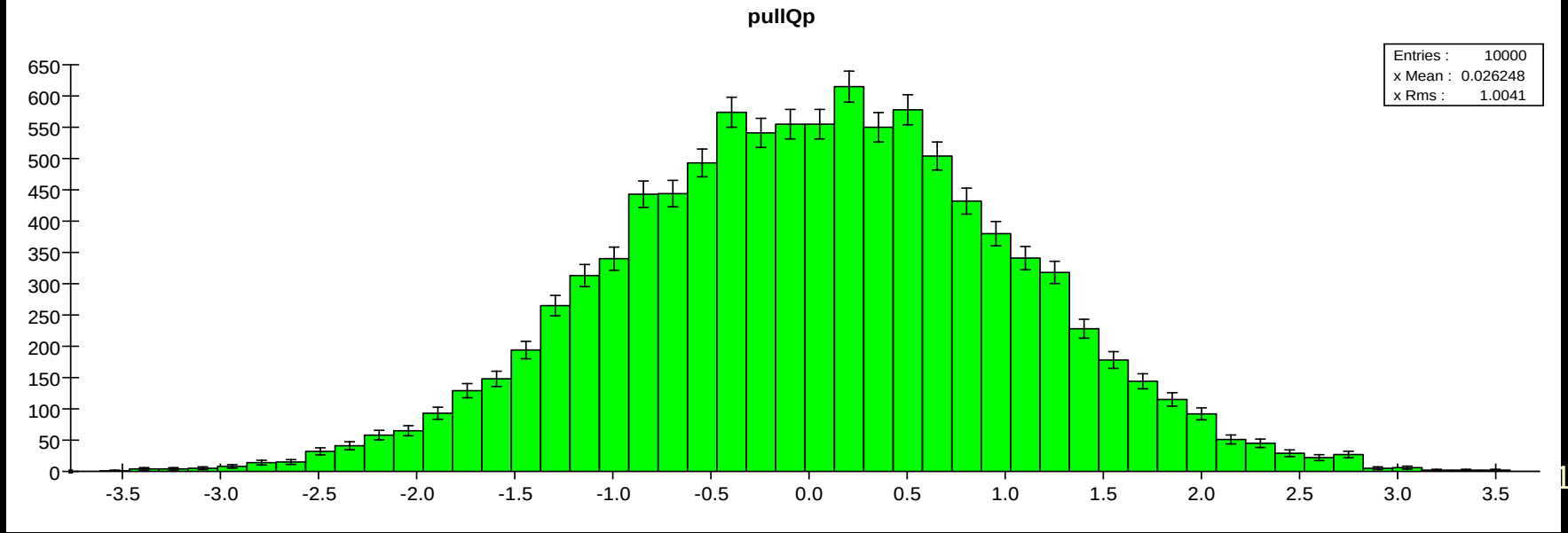

resTy
Entries : 10000
x Mean : 5.0679E-8
x Rms : 1.8803E-6
x10-6


pullTx
Entries : 10000
x Mean : 0.022671
x Rms : 0.99921


pullTy
Entries : 10000
x Mean : 0.027129
x Rms : 1.0065

Momentum (Q/p)


resQp
Entries : 10000
x Mean : 1.8809E-5
x Rms : 7.1958E-4
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-0.0025
-0.0020
-0.0015
-0.0010
-0.0005
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025


pullQp
Entries : 10000
x Mean : 0.026248
x Rms : 1.0041
650
600
550
500
450
400
350
300
250
200
150
100
50
0
-3.5
-3.0
-2.5
-2.0
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5

Fitting Snapshot

- Machinery seems to be working in that pulls are consistent with unit normal:

$$\sigma_X = 1.0002$$

$$\sigma_Y = 0.9949$$

$$\sigma_{X'} = 0.9992$$

$$\sigma_{Y'} = 1.0065$$

$$\sigma_{q/p} = 1.0041$$

Track Interaction

- trf package includes Interactor API to model the material effects encountered by charged particles traversing the detector.
  - Multiple Coulomb Scattering implemented as simple gaussian.
  - Bethe-Bloch used for energy loss.
  - Both model material as generic X/X_0
  - OK for collider detector physics where most particles can safely be assumed to be pions.

Next Steps

- Generate tracking detector from compact detector
  - Developing “tilted” ZPlane
  - Deriving mathematics of intersection and Jacobian
- Modify interaction processes
  - refine for low-energy electrons
  - use specific material properties
- Add field map to HPS detector simulation
- Analyze slic output along lines of standalone code
- Analyze the test data