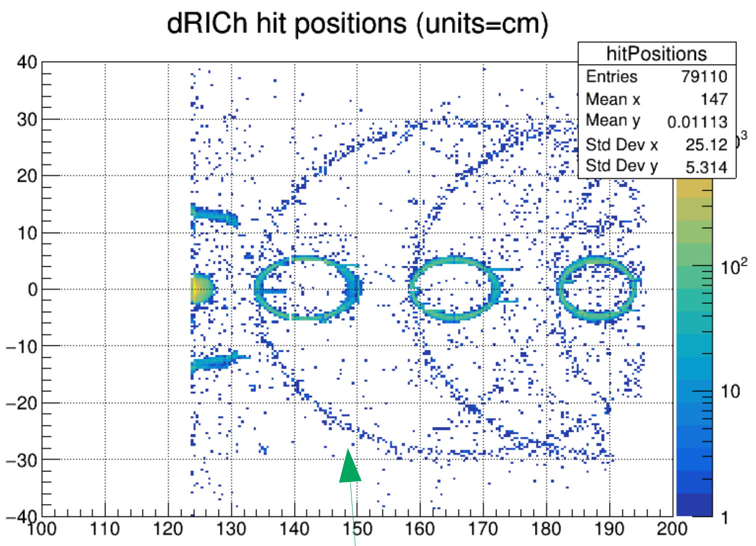
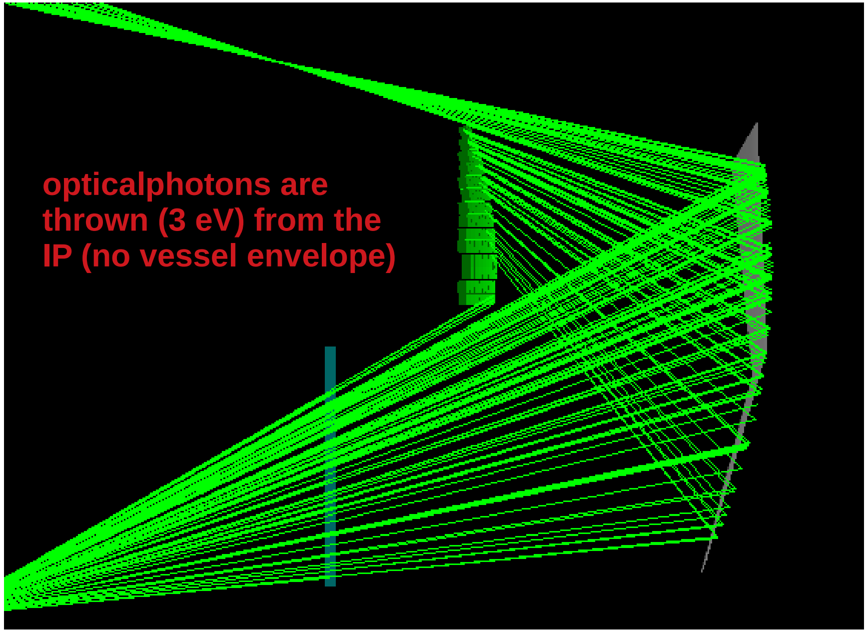
dRICh Optics Tuning for ATHENA Integration

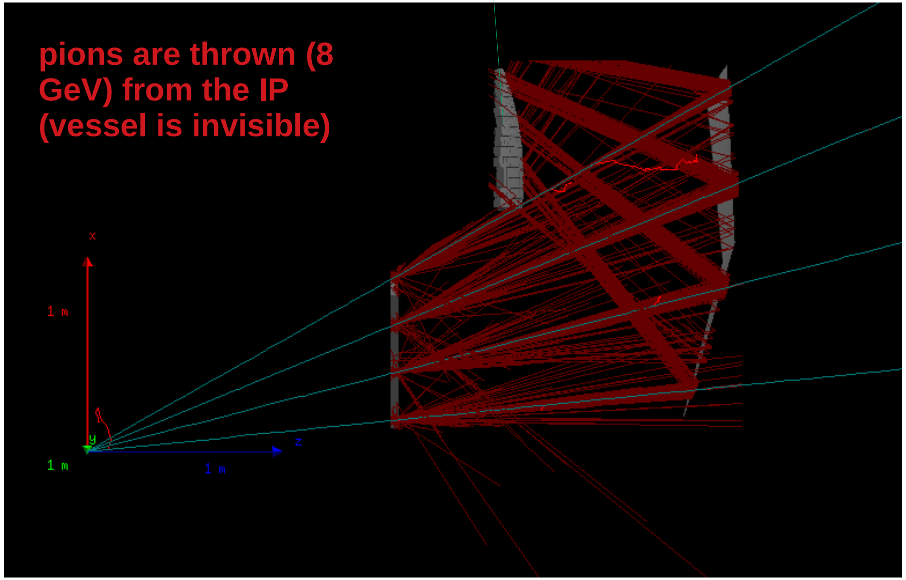
Christopher Dilks

Necessity to improve ATHENA dRICh optics

- Port of JLEIC → Fun4all → DD4hep → ATHENA integration
- ATHENA acceptance may differ from AI-optimized acceptance → degradation of quality
  - angular acceptance loss: low psuedorapidity trajectories reflected toward outer radial wall
  - aberration effects at high pseudorapidity (rings are smeared)
- GOAL: try to improve this for the given ATHENA acceptance
  - Need a “good enough” solution for short term; AI optimization can be used for long term

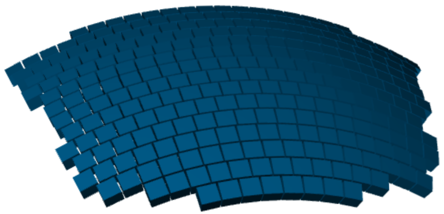






Available Optics Tuning Parameters

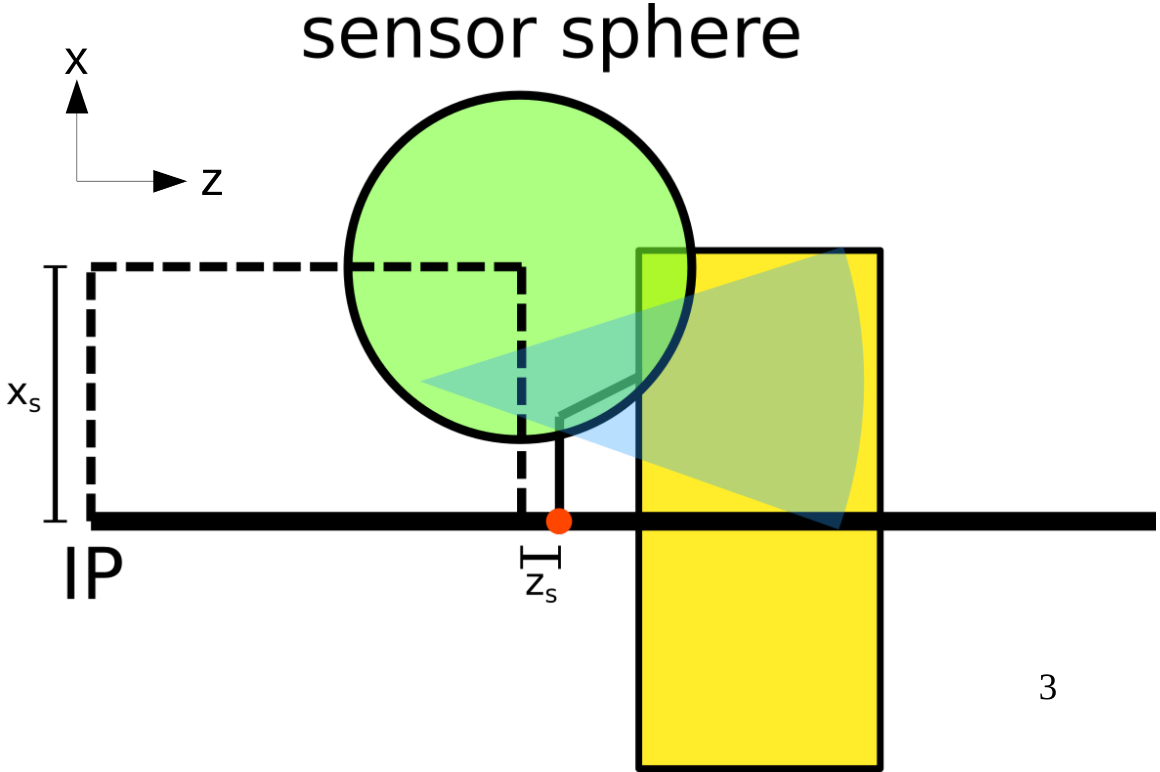
Sensor sphere: sensors are tiled on a sphere with specified radius and center coordinates (z_s,x_s), defined with signs specified with respect to vessel snout front (red point)



Cuts are used to take a subset of the sphere within the vessel

```
<sphere
  radius="160.0 * cm"
  centerx="DRICH_rmax2 + 5.0*cm"
  centerz="-80.0 * cm"
  debug="DRICH_debug_sensors"
  />
<sphericalpatch
  phiw="18*degree"
  rmin="DRICH_rmax1 + 0.0*cm"
  rmax="DRICH_rmax2 - 5.0*cm"
  zmin="DRICH_SnoutLength + 5.0*cm"
  />
```

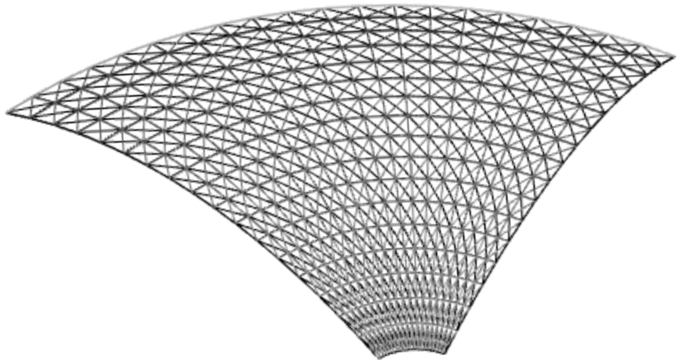
numbers are not optimal



Available Optics Tuning Parameters

Spherical Mirror: must specify a “backplane” position (maximum z-plane of the mirror surface)

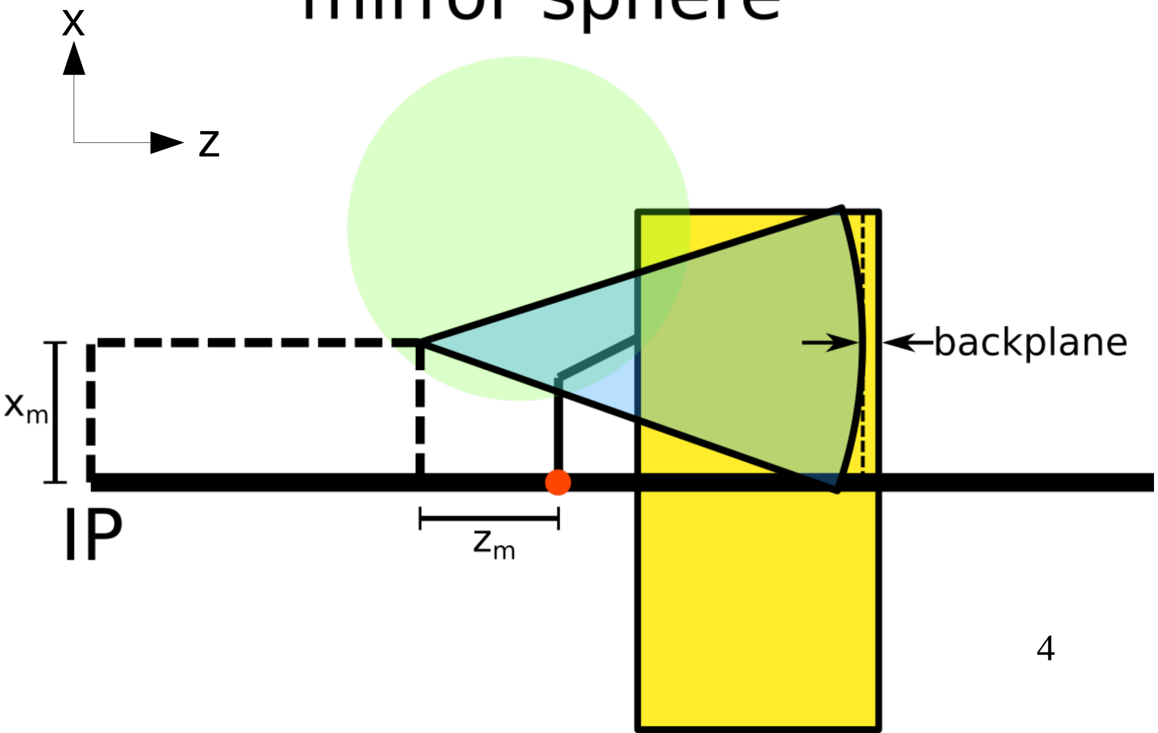
Mirror position (z_m,x_m) determined with the help of “tune” parameter; radius determined from z_m, given “backplane”



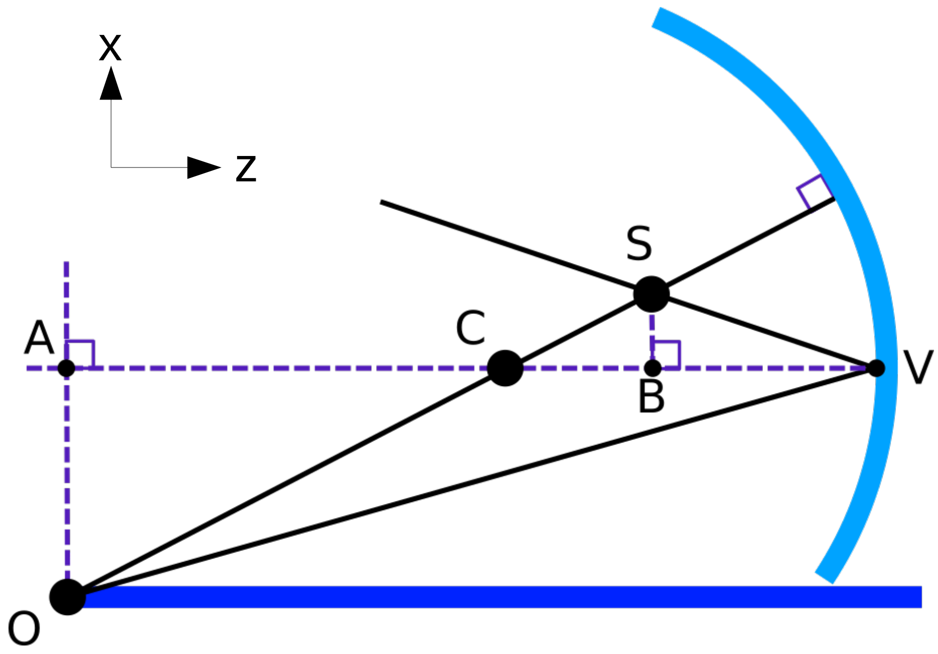
mirror sphere

```
<mirror
  material="Acrylic_DRICH"
  surface="MirrorSurface_DRICH"
  vis="DRICH_mirror_vis"
  backplane="DRICH_window_thickness + 5.0*cm"
  rmin="DRICH_rmin1 + DRICH_wall_thickness + 0.0*cm"
  rmax="DRICH_rmax2 - DRICH_wall_thickness - 2.0*cm"
  phiw="55*degree"
  thickness="0.2*cm"
  focus_retune="0.4"
  debug="DRICH_debug_mirror"
  />
```

numbers are not optimal



Spherical Mirror Optics



Goal: Focus an object at O (e.g., the IP, or photon vertex) at the position S

C = mirror center (radius = CV)
A,C,B,V are along the mirror's optical axis; at V the mirror tangent plane is parallel to the xy-plane

Given positions O and S, along with the length AV, one can solve the following equations for mirror center and radius (C and CV):

$$\begin{cases} \Delta ACO \sim \Delta BCS \\ \Delta AVO \sim \Delta BVS \end{cases}$$

Caveat: if O is far from the optical axis, spherical aberrations will cause the image to not appear at S

Alternatives to spherical mirrors:

- Parabolic/hyperbolic mirrors
  - Less aberrations, but introduces coma
  - Very difficult to manufacture to a high quality, and consequently expensive
- Segmented spherical mirrors with different radii
  - Could mitigate aberation effects, but can introduce ambiguity in reconstruction (which mirror was hit?)

3 attempts to improve optics:

1. Parameterize mirror such that Cherenkov photons focus at the center of the sensor sphere, and use ‘tune’ parameter to shift the focus closer to the sensors

2. Use ‘tune’ parameter to move the focal place, and scan the parameter space to try to find some sweet spot

3. Add a second, orthogonal tune parameter, for more focal point control

Note: this study is restricted to the sector centered on the +x axis, and to the xz plane, with the hope that spherical asymmetry will optimize the azimuthal space for “free”

Attempt #1 to Improve Optics

- Start by using spherical mirror equations to focus the IP at the center of the sensor sphere
  - For this attempt, the sensor sphere center and radius were also altered
  - Because of aberrations, the image will not appear in the sensor sphere center
- Then re-configure the mirror so that the image moves toward the centroid of the sensors themselves; this is done by the “tune” parameter, where:
  - tune = 0.0: ideal focus is at sensor sphere center
  - tune = 1.0: ideal focus is on the “centroid sensor”

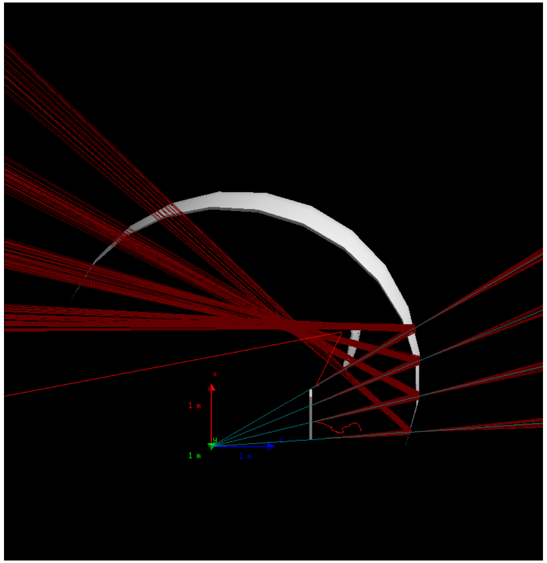
Throw 4 pions, equally spaced in θ

Gas volume radiates, aerogel radiation disabled

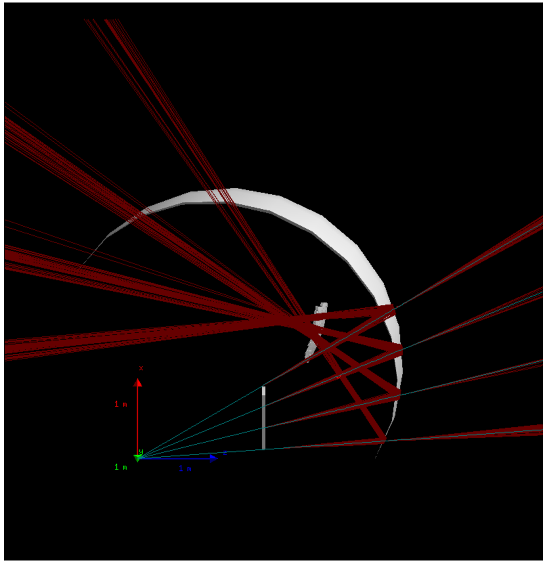
Sensors are non-interacting

Extended spherical mirror is shown

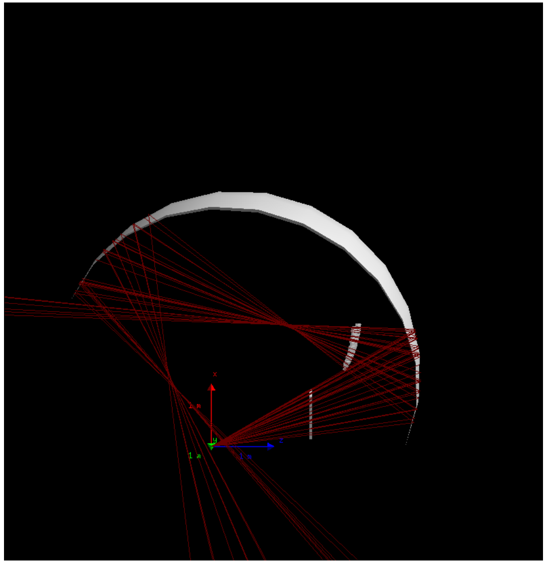
tune = 0.0



tune = 0.6

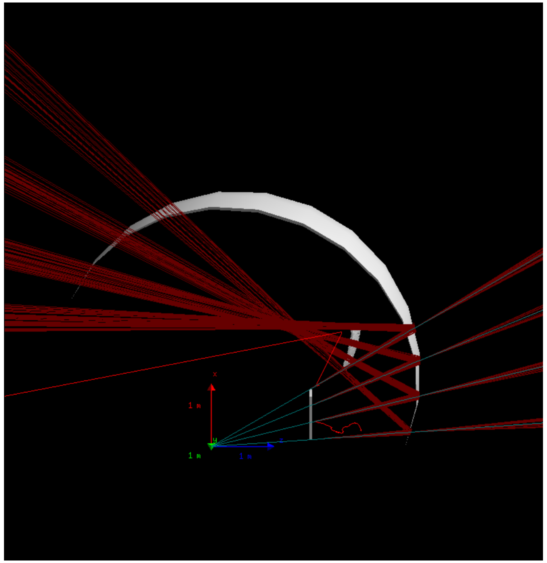


tune = 0.0



Throw 30 opticalphotons, within angular acceptance limits

All materials are “optical” vacuum, except for mirror

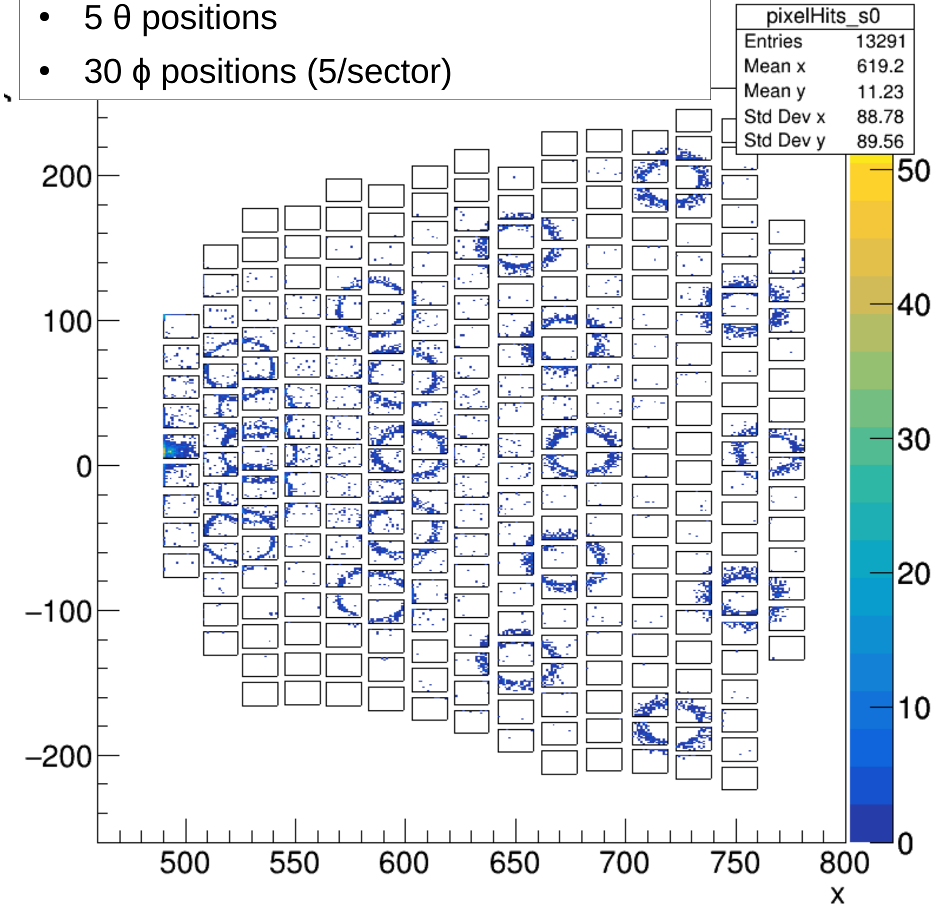


Throw 4 pions, equally spaced in θ

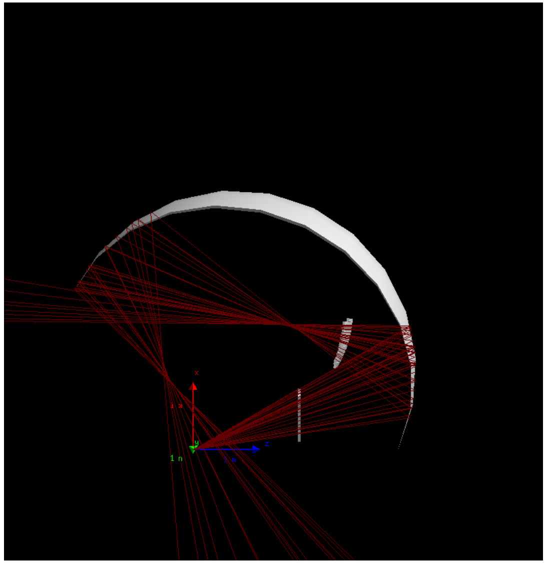
Gas volume radiates, aerogel radiation disabled

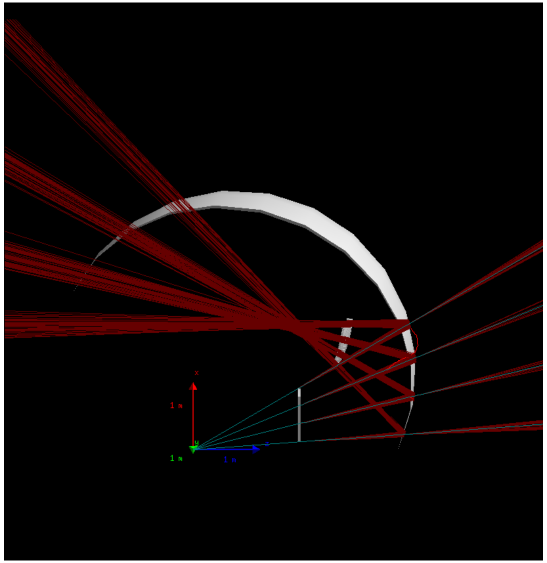
Hit readout from one sector, with 30 pions (8 GeV) thrown at each fixed θ,ϕ position (evenly distributed):

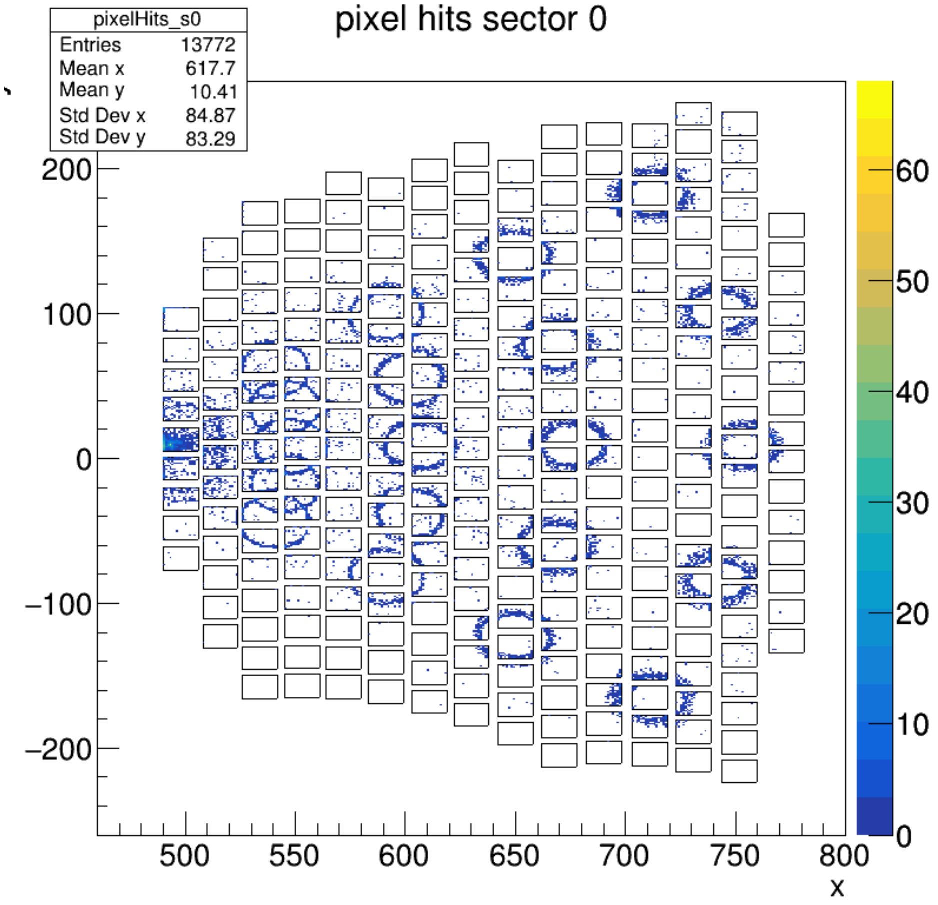
- 5 θ positions
- 30 ϕ positions (5/sector)



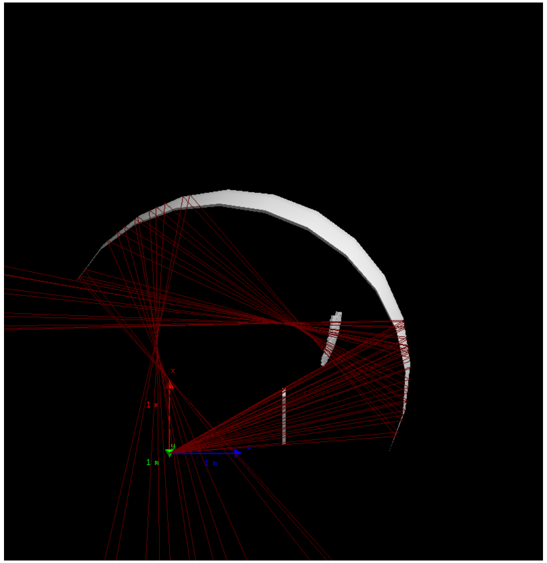
tune = 0.2

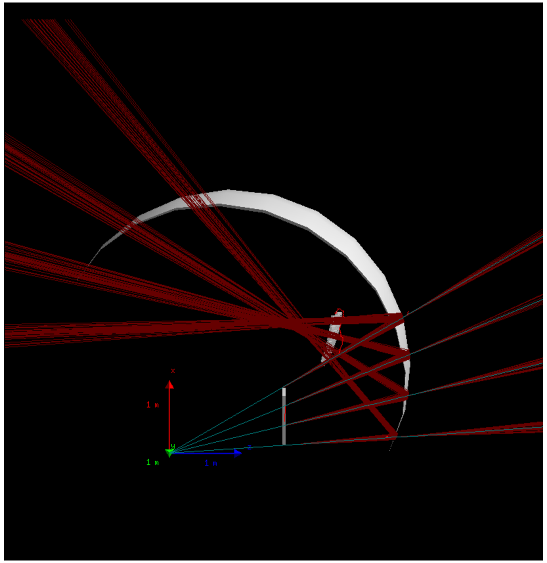


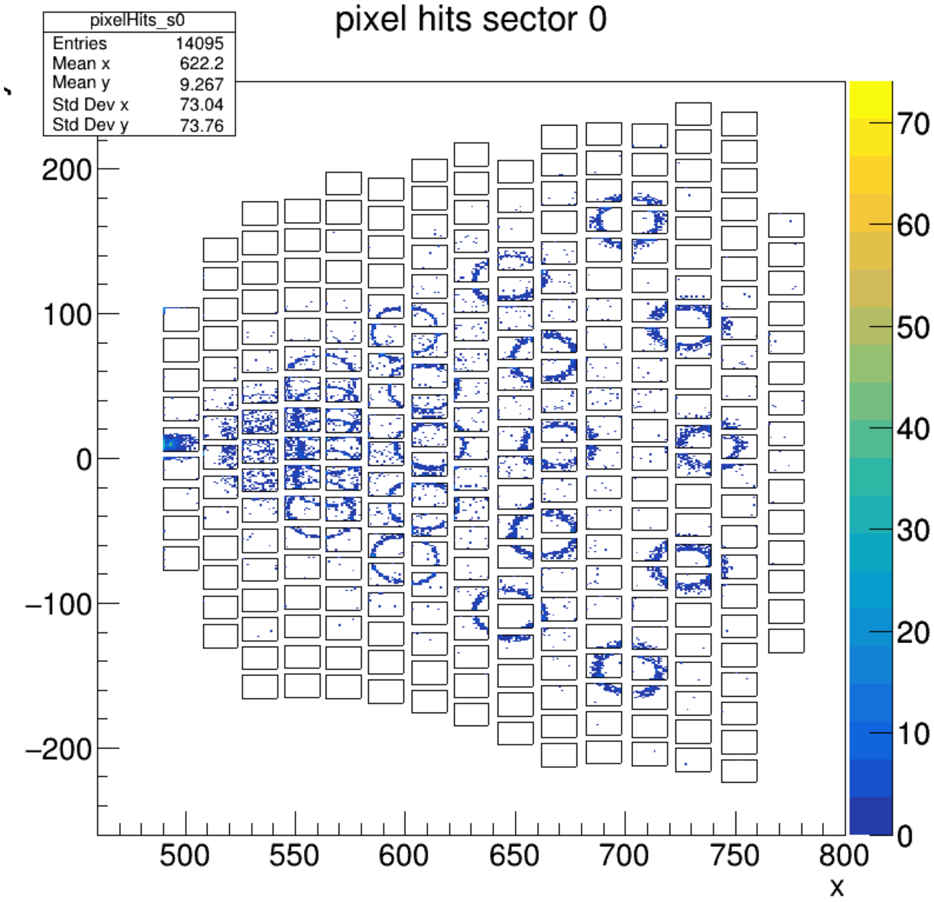



pixelHits_s0
Entries 13772
Mean x 617.7
Mean y 10.41
Std Dev x 84.87
Std Dev y 83.29
pixel hits sector 0
200
100
0
−100
−200
500
550
600
650
700
750
800
x
60
50
40
30
20
10
0

tune = 0.4



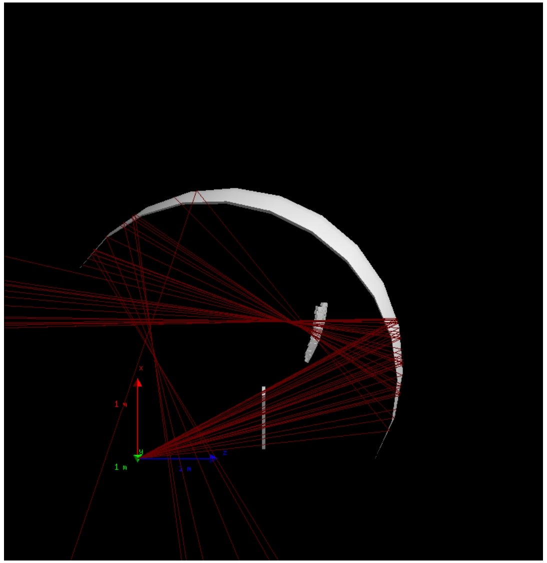


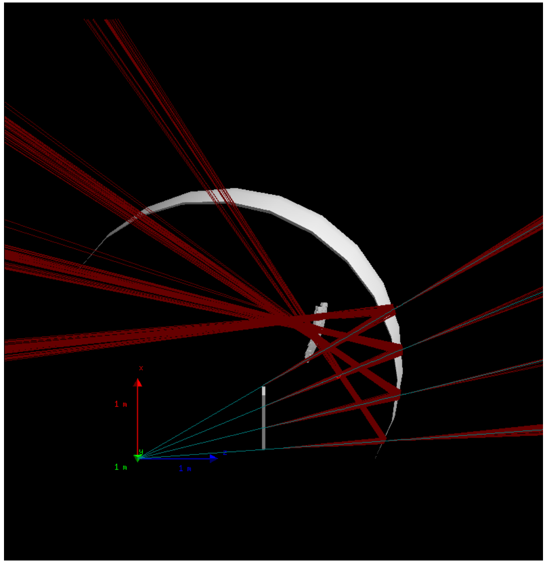

pixelHits_s0
Entries 14095
Mean x 622.2
Mean y 9.267
Std Dev x 73.04
Std Dev y 73.76
pixel hits sector 0
200
100
0
−100
−200
500
550
600
650
700
750
800
x
70
60
50
40
30
20
10
0

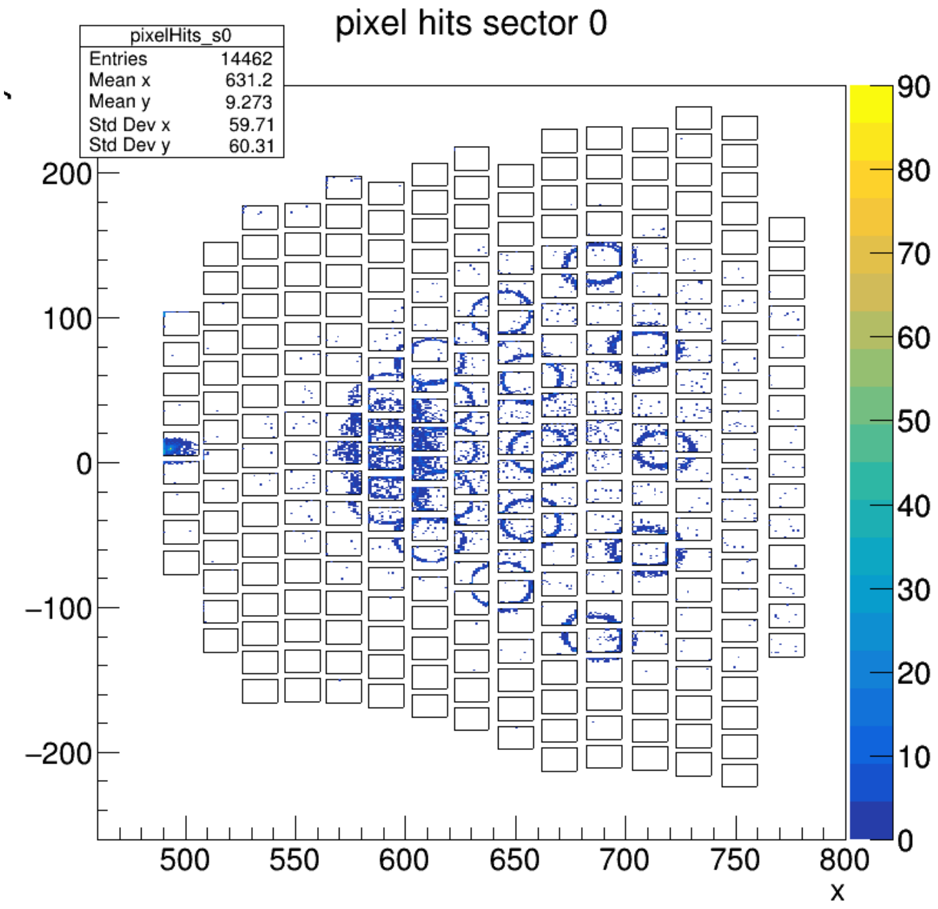
tune = 0.6

Higher tune effects:

- Focal point closer to sensors
- High-θ resolution a bit better
- Low-θ resolution significantly worse (aberrations/coma)







Attempt #2 to Improve Optics

- Brute force "optimization": scan the parameter space and try to find a "sweet spot"
  - Parameters: sensor sphere center (centerZ,centerX), radius, and aforementioned "tune"
- Iterative procedure:
  - Start with a large lattice spacing
  - Pick the best-looking configuration
  - Re-run with a smaller lattice in the neighborhood of that configuration
  - Re-iterate as necessary
- AI optimization is a *<u>necessity</u>* in the long term
  - For the short term, we only have time and resources to produce something reasonable

```
# iteration 1: (units=cm)                          # snoutLength=50, vesselLength=180, tankRadius=200
it['retune'] = list(np.linspace( 0.0, 0.6, 4 ))
it['centerZ'] = list(np.linspace( -90, 30, 5 ))  # defined w.r.t. snout front-plane
it['centerX'] = list(np.linspace( tankRadius-20, tankRadius+20, 3 ))
it['radius'] = list(np.linspace( snoutLength - centerZ + 10, vesselLength - centerZ - 10, 4 ))
```

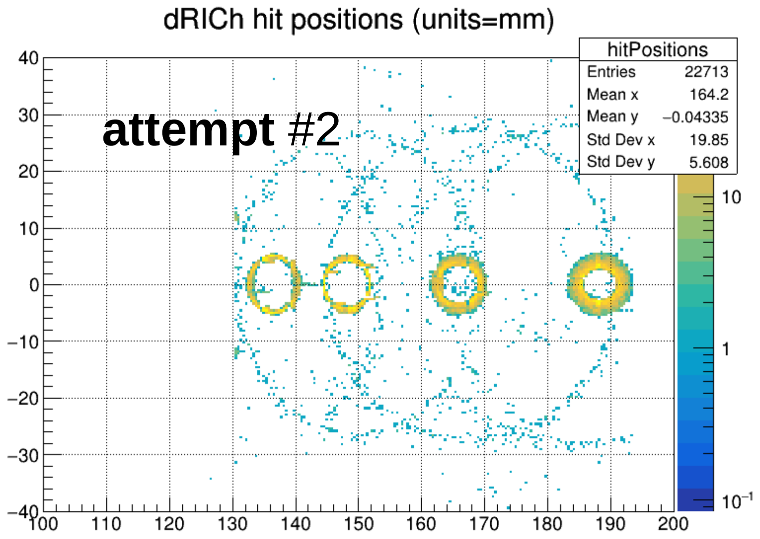
Best parameters (chosen by eye!!): retune=0.2, centerZ=–30, centerX=220, radius=126.7

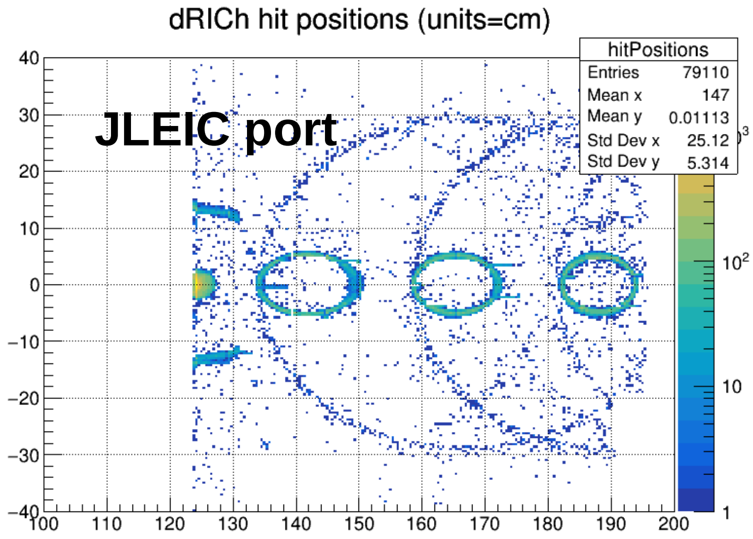
```
# iteration 2:
it['retune'] = list(np.linspace( 0.1, 0.3, 3 ))
it['centerZ'] = list(np.linspace( -50, 0, 6 ))
it['centerX'] = list(np.linspace( tankRadius+10, tankRadius+30, 3 ))
it['radius'] = list(np.linspace( snoutLength - centerZ + 40, vesselLength - centerZ - 40, 4 ))
```

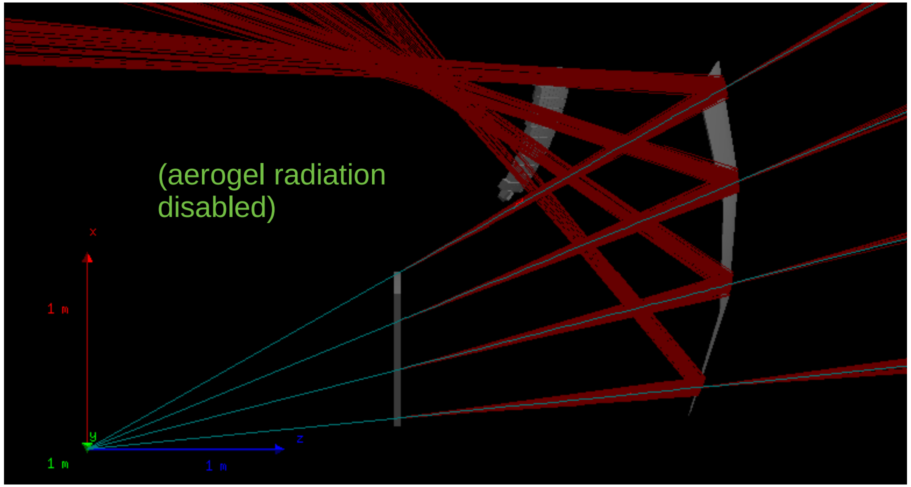

Best parameters (chosen by eye!!): retune=0.1, centerZ=–40, centerX=210, radius=130.0

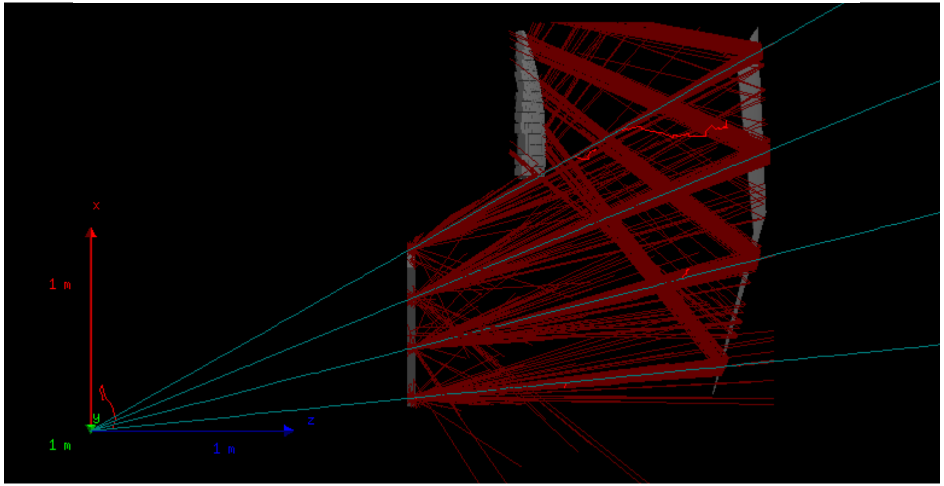
Attempt #2 to Improve Optics

Result: maximized angular acceptance, but damaged resolution at high pseudorapidity









Attempt #3 to Improve Optics

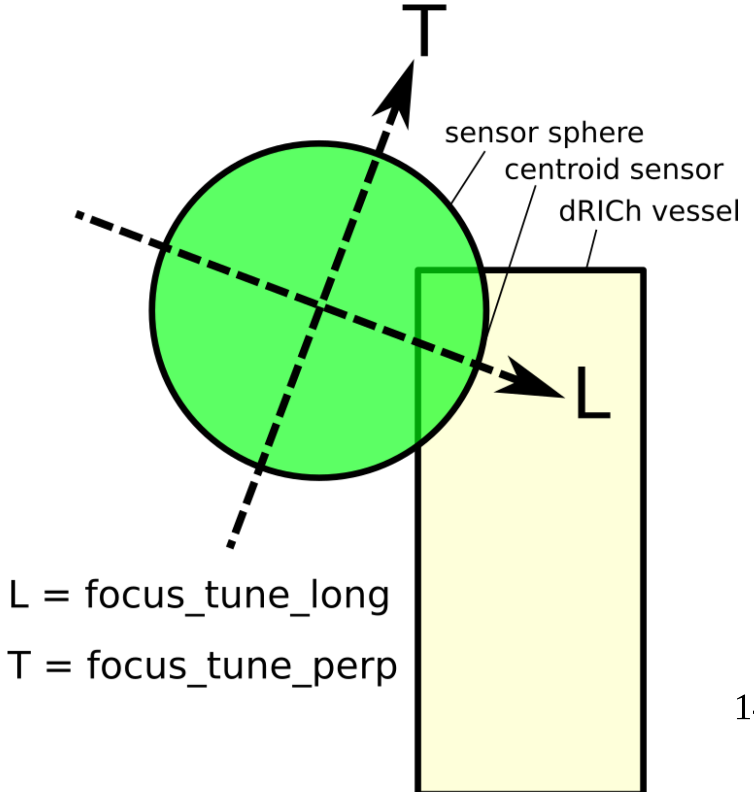
- Add another “tune” parameter ‘focus_tune_perp’, to allow movement *transverse* to the first tune parameter’s direction, now called ‘focus_tune_long’
- Then started with *original* JLEIC sensor sphere dimensions and positioning, and “nudged” things around until there was some improvement

Mirror

- backplane = 5 cm from back wall
- focus_tune_long = 0.5
- focus_tune_perp = 0.2

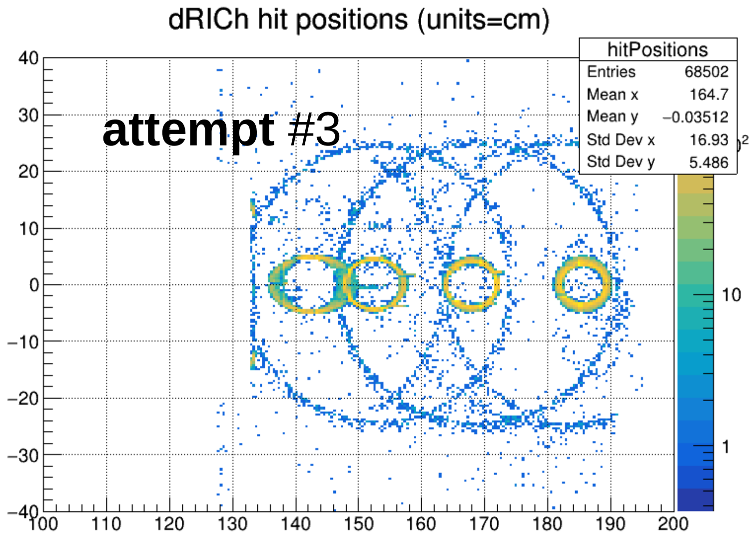
Sensor Sphere

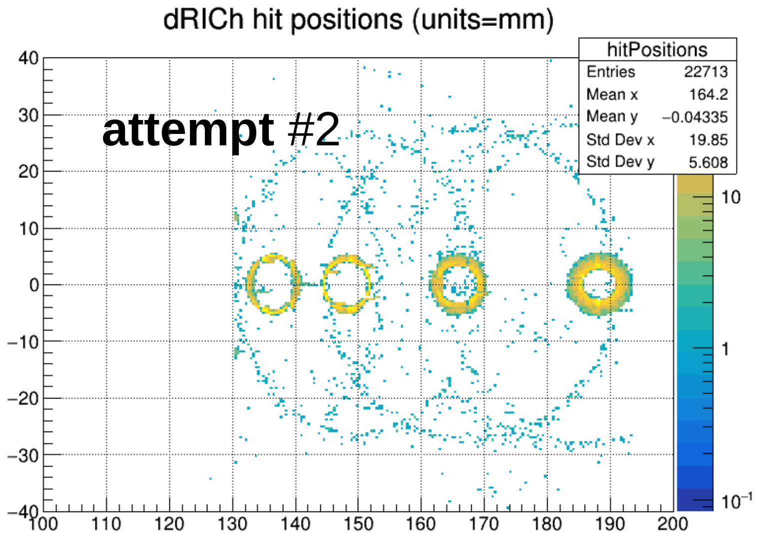
- centerz = –87 cm (w.r.t. front wall)
- centerx = vessel radius – 35 cm (=165 cm)
- radius = 160 cm

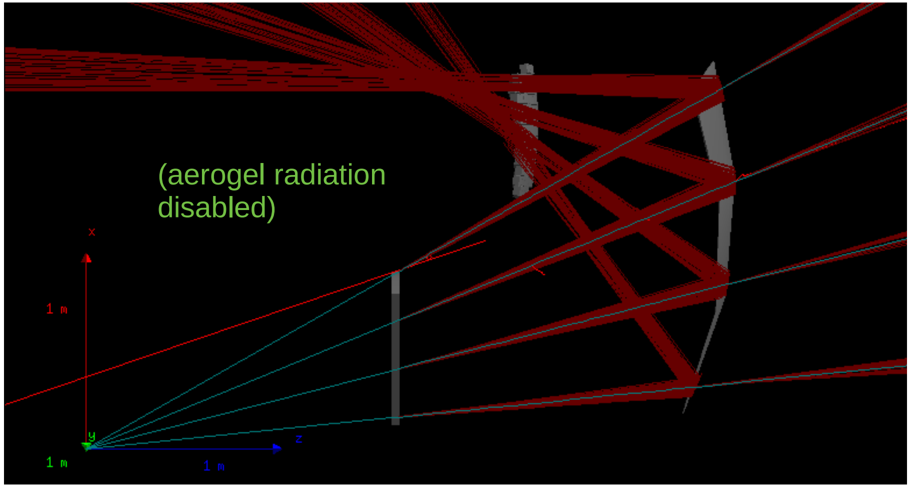


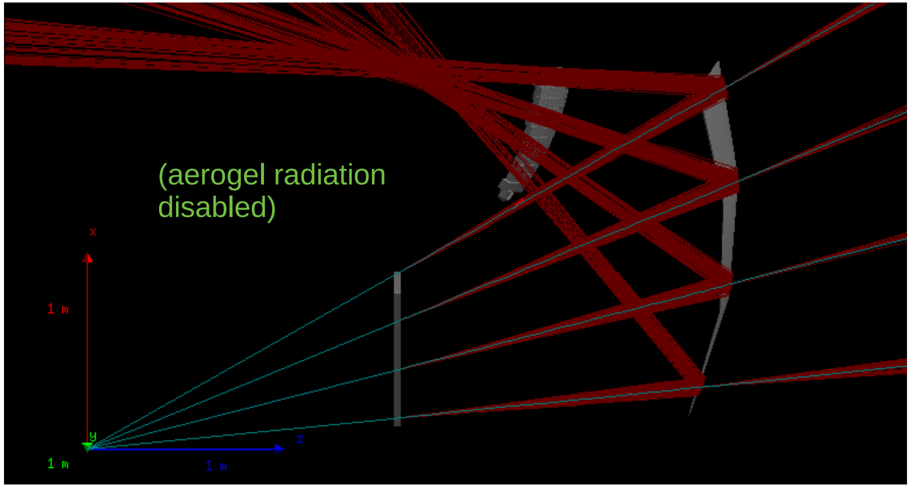
Attempt #3 to Improve Optics

Result: comparing to attempt #2 → better resolution at low pseudorapidity



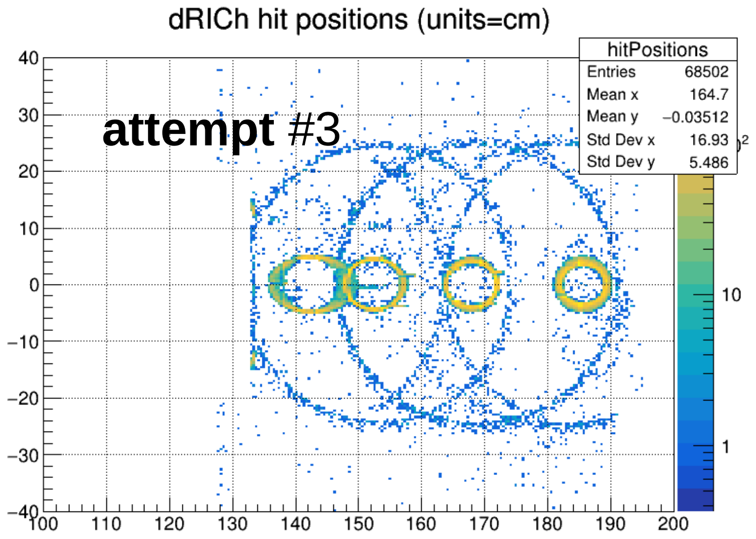


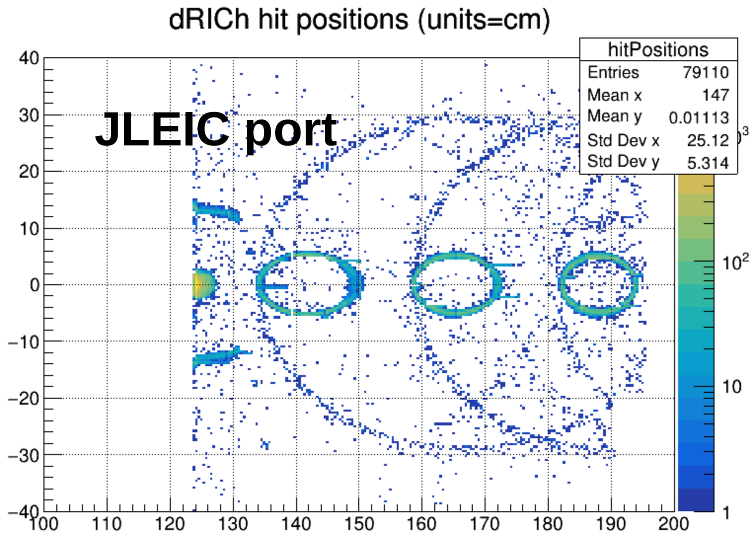


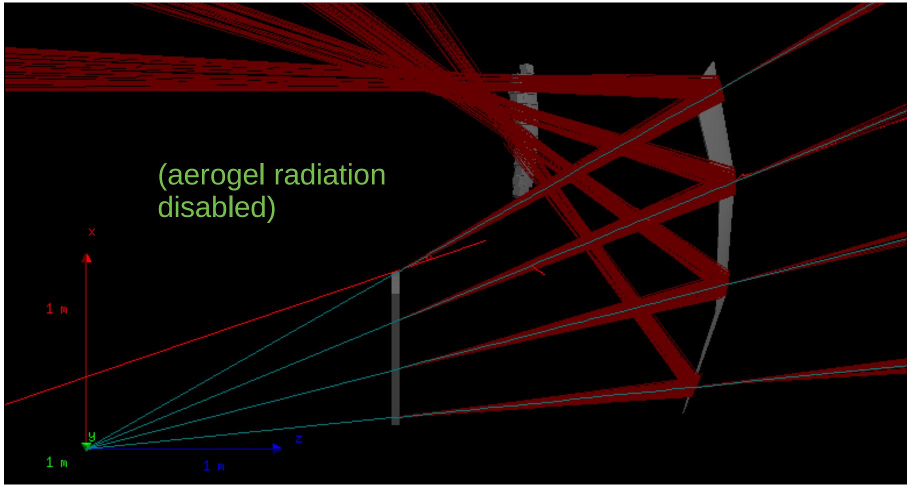


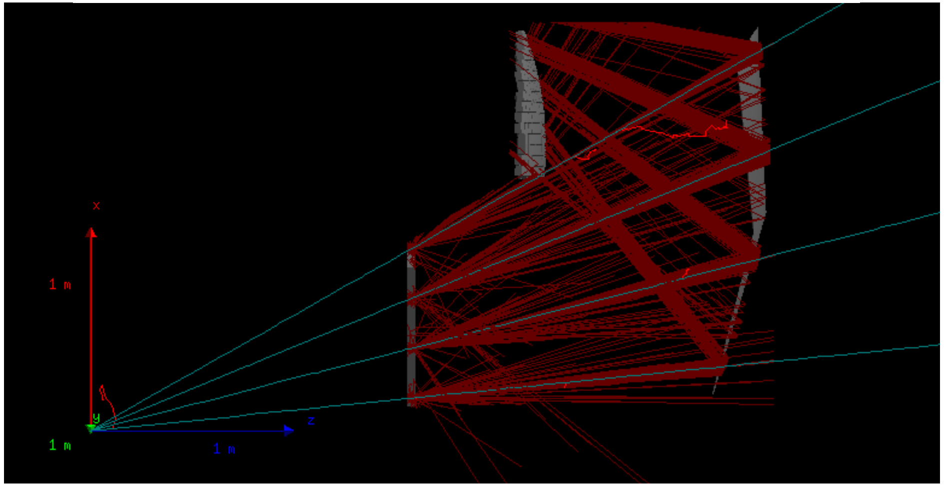
Attempt #3 to Improve Optics

Result: all 4 rings visible, some increased coma at high pseudorapidity



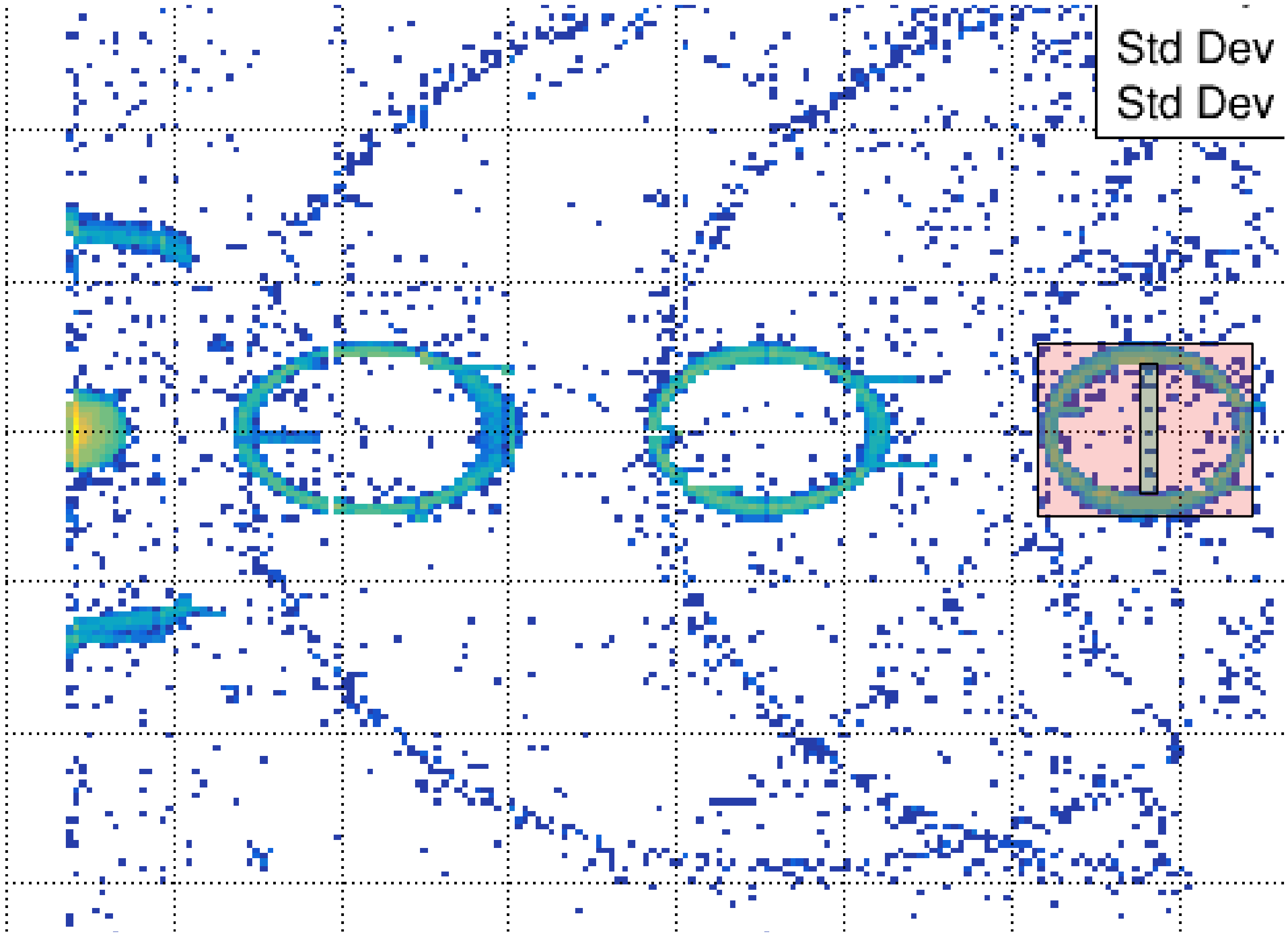




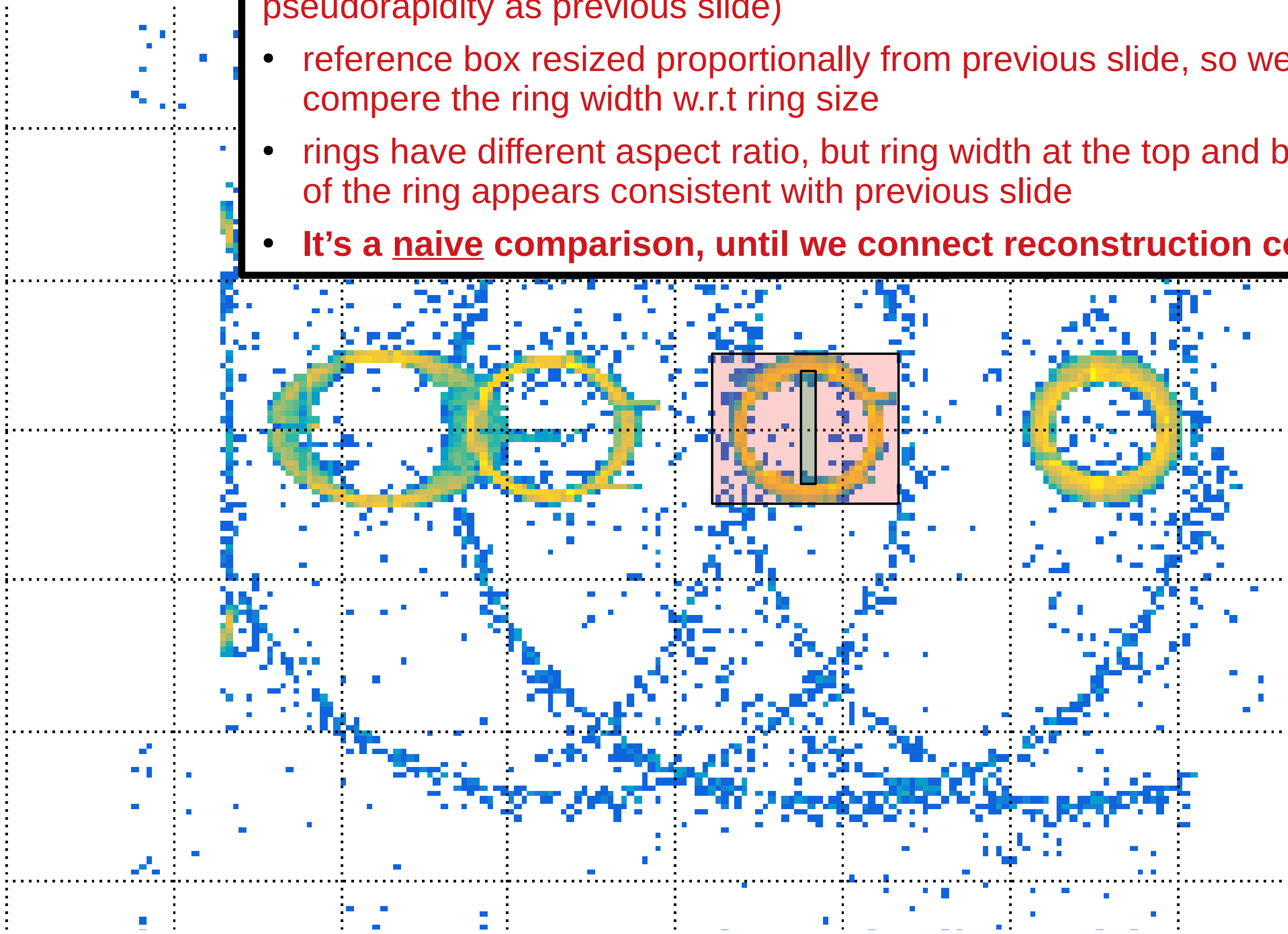


Compare JLEIC port with …

Reference box drawn on the 3rd gas ring



… with Attempt #3



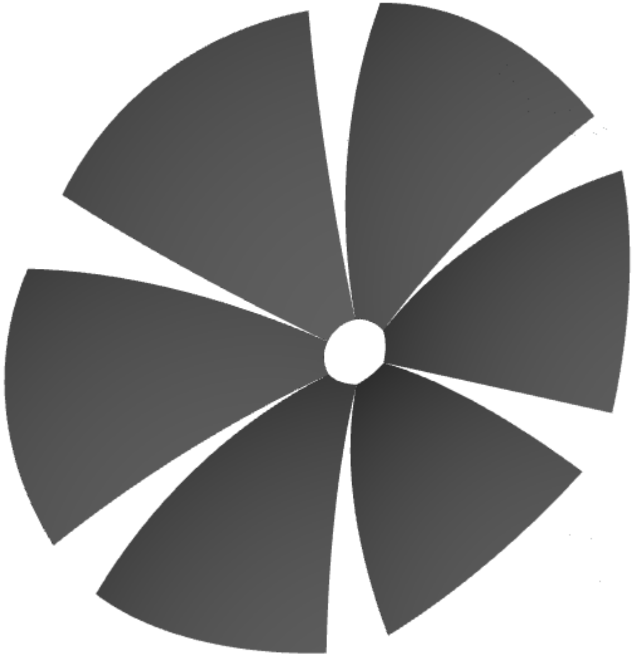
Reference box drawn on the 3rd gas ring (so generated pion has same pseudorapidity as previous slide)

- reference box resized proportionally from previous slide, so we can compere the ring width w.r.t ring size
- rings have different aspect ratio, but ring width at the top and bottom of the ring appears consistent with previous slide
- **It's a __naive__ comparison, until we connect reconstruction code**

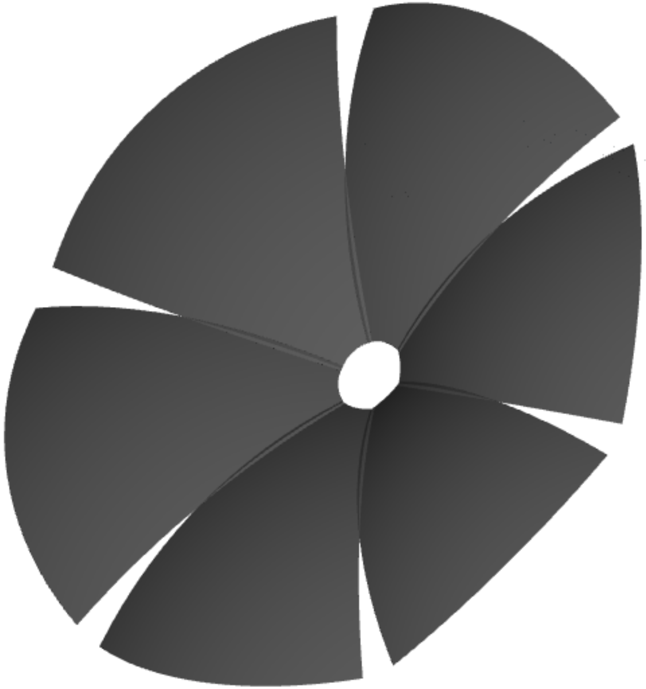
Azimuthal Acceptance

- spherical mirrors are cut from a sphere, given angular ranges Δϕ andΔθ, then rotated/translated to a specific position
- Resulting effective azimuthal coverage is not optimal

Δϕ=53°
– widest coverage with no geometry overlaps
– this value depends on optics parameters
– approximately 90% of 2π acceptance



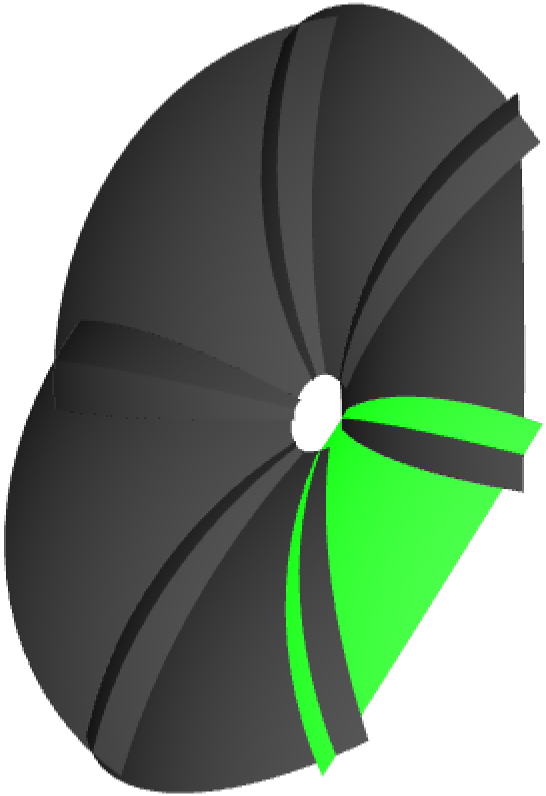
Δϕ=60°
→ overlaps at low θ
→ some small gaps at high θ



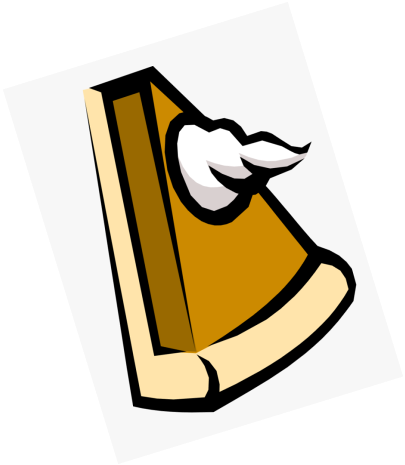
Azimuthal Acceptance

Increase azimuthal coverage with boolean solids

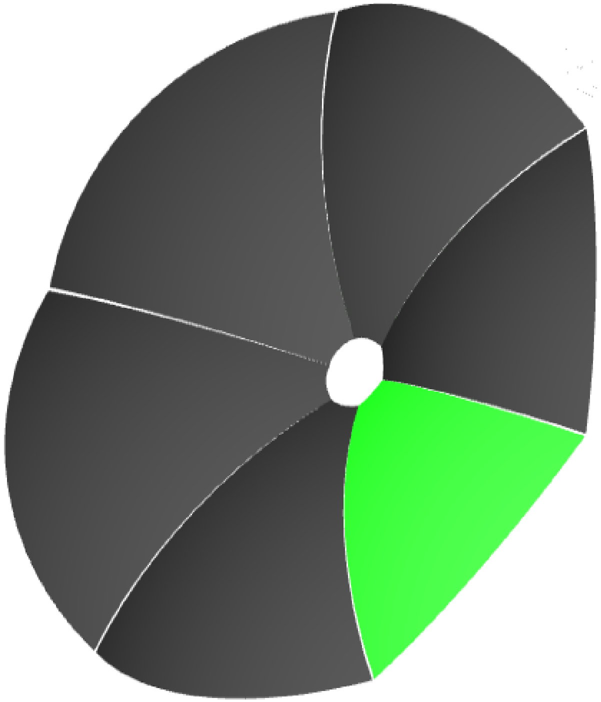
- let $\Delta\phi$ as high as it needs to be so there are no gaps
- intersect each mirror with an azimuthal “pie slice” volume
- azimuthal gaps are filled (allowing a small gap for realism)
- caveats
  - boolean solids + optical photons = performance degradation, but should be ok if boolean ops are minimal
  - what will happen to rings that span 2 sectors? fiducial cuts?



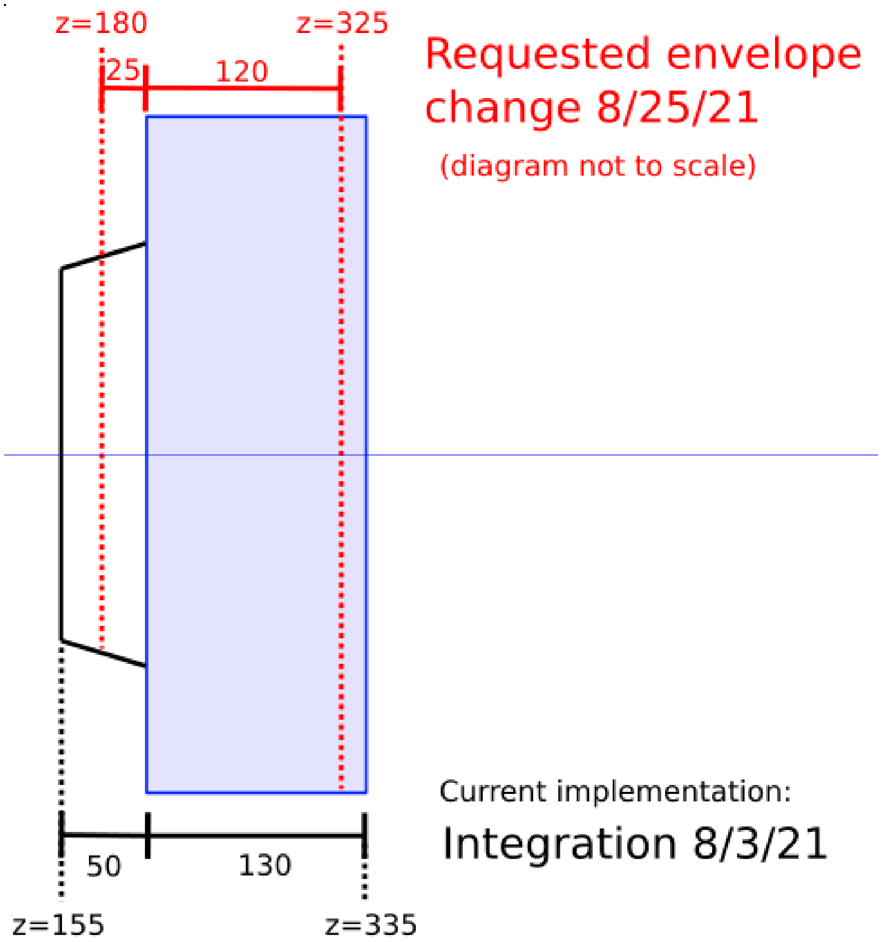
$\cap$

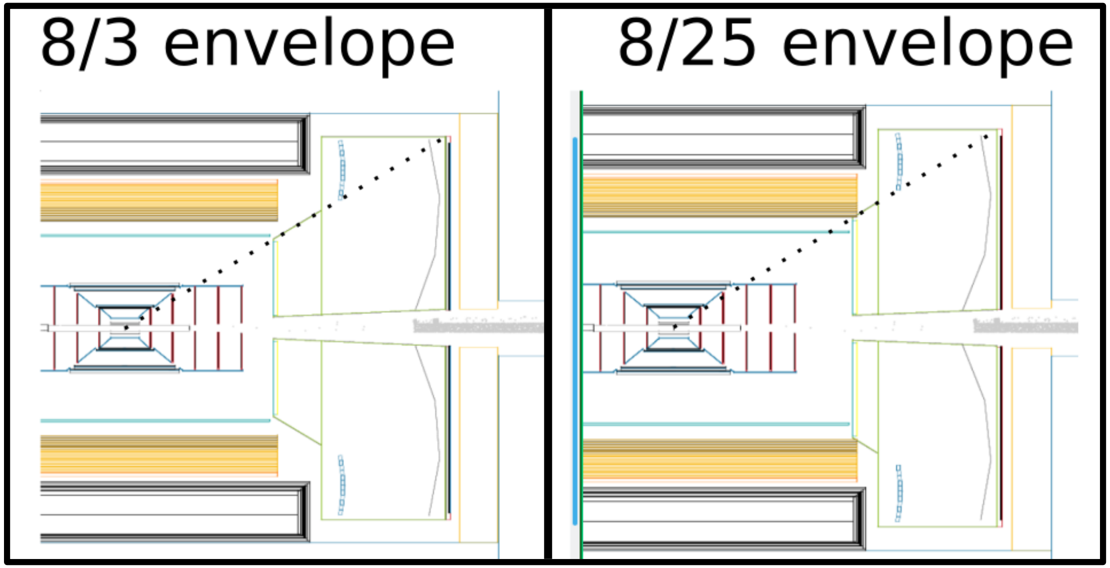


$=$



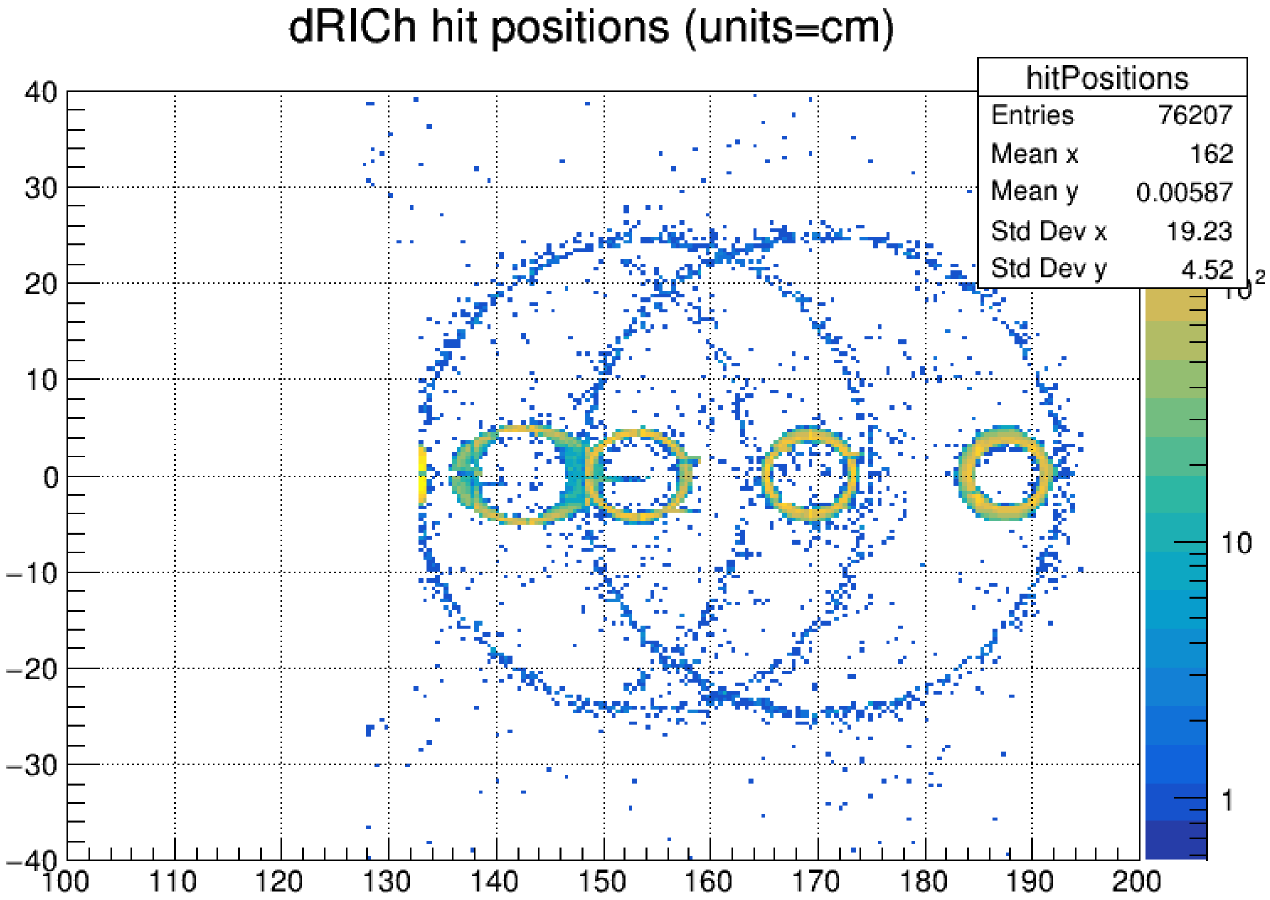
Envelope Changes





8/3 Envelope Optics Test

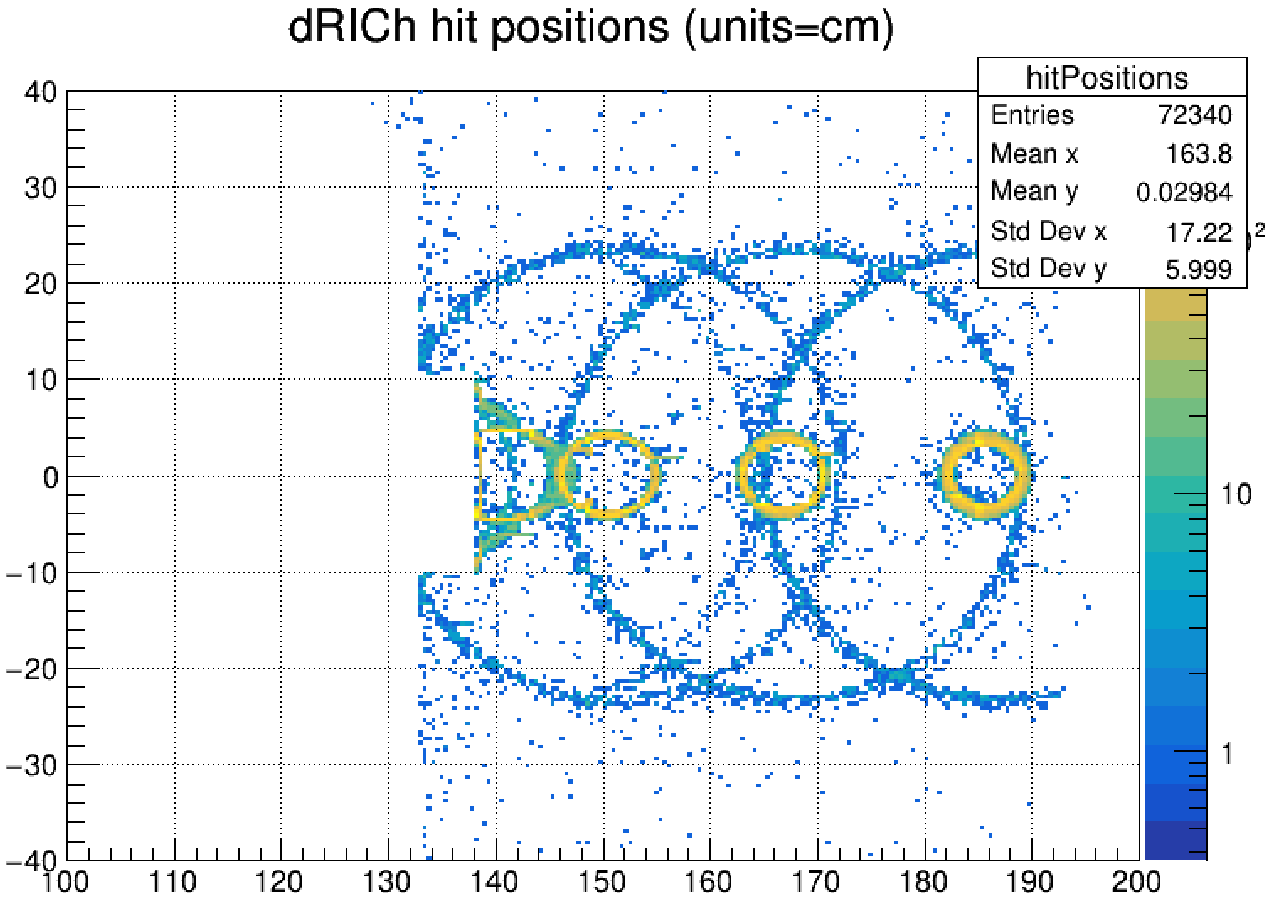
compare to new envelope next slide



1,000 pions (8 GeV) at each of four directions

8/25 Envelope Optics Test

Some re-tuning needed, but not much



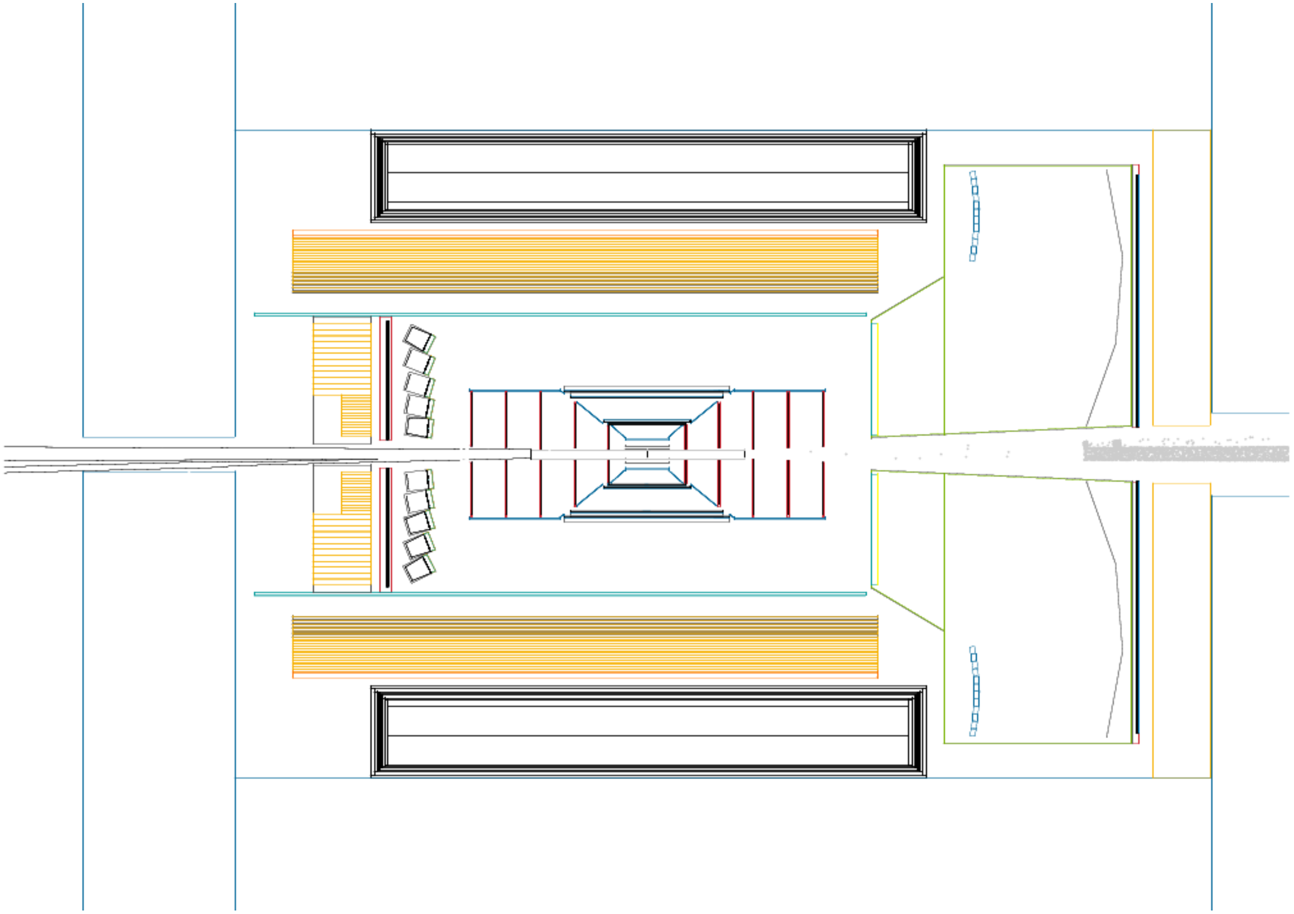
1,000 pions (8 GeV) at each of four directions

Outlook

- **Additional tasks:**
  - is the gas absorption length correct (1m)?
  - cross checks with other implementations
  - would we be better off with a non-spherical sensor placement, to compensate for aberrations / comas?

- **Highest Priority:**
  - pick a “good enough” optics configuration for now, we must move on
  - reconstruction
  - decide what performance plots we want

Envelope Changes



Envelope Changes