Pixel TPC simulation and reconstruction

Kees Ligtenberg, Peter Kluit, Jan Timmermans

LCTPC WP meeting

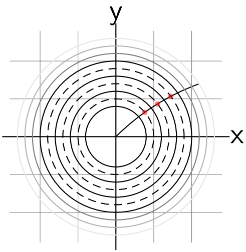
11 May 2017

Outline

1. Simulation
   - Pads
   - Pixels
2. Reconstruction
   - Pads
   - Pixels

Simulation of pads within ilcsoft

version 01-17-09, ILD_o1_v5



Volumes are organised as tube shaped layers, there are no pad columns

- Detector is described by DD4HEP geometry
- Pads have ideal 100% coverage
- Geant4 processes interactions of particle(s) from gun or event
- Single hit in TPC is deposited if energy is above threshold (32eV) in a single pad. Position of pad centre crossing is recorded
- Diffusion and hit resolution is simulated by smearing the hits by the expected resolution in $r\phi$ and z directions

Diffusion and hit resolution for pads

As found in code of TPCDigiProcessor

Diffusion and hit resolution are simulated by smearing the hits by the expected resolution

$$\sigma_{r\phi} = \sqrt{\sigma_{r\phi 0}^2 + \sigma_{\phi 0}^2 \sin^2(\phi_{\text{pad}}) + \frac{D_{r\phi}^2}{N_{\text{Eff}}} \sin(\theta_{\text{pad}}) \left(\frac{6\text{ mm}}{h_{\text{pad}}}\right) \left(\frac{4.0\text{ T}}{B}\right) L}$$

$$\sigma_z = \sqrt{\sigma_{z0}^2 + D_z^2 L}$$

,

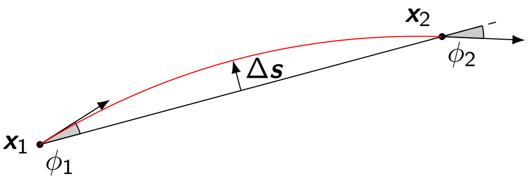
$$\sigma_{r\phi 0} = 0.05\text{ mm} \qquad \sigma_{z0} = 0.4\text{ mm} \qquad \sigma_{\phi 0} = 0.9\text{ mm}$$

$$D_{r\phi} = 0.025\text{ mm}/\sqrt{\text{cm}} \qquad D_z = 0.08\text{ mm}/\sqrt{\text{cm}} \qquad N_{\text{Eff}} = 22.$$

Simulation of pixels

- Pixels are described by the same cylindrical volumes in DD4HEP
- Pixels also have ideal 100% coverage
- Multiple hits per row can be deposited
- In order to simulate diffusion, hits are smeared transverse to track in x, y and z directions

- Interpolate the track with a parabola over a volume of 0.99 mm (18 pixel rows)



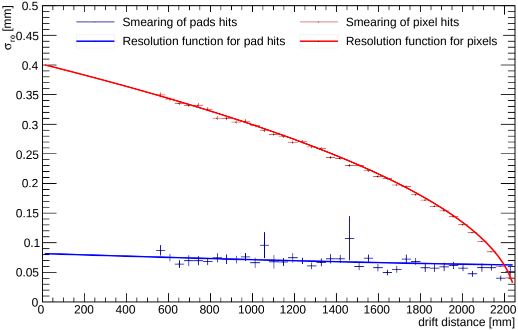
Diffusion and hit resolution for pixels

$$\sigma_{r\phi} = \sqrt{\sigma_{r\phi 0}^2 + D_{r\phi}^2 \left(\frac{4.0\ \mathrm{T}}{B}\right) L} \qquad \sigma_z = \sqrt{\sigma_{z0}^2 + D_z^2 L}$$

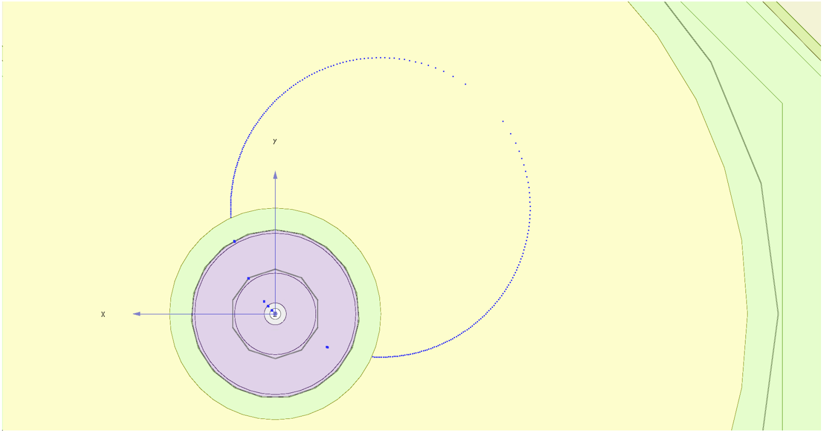
$$\sigma_{r\phi 0} = 0.055/\sqrt{12}\ \mathrm{mm} \quad \sigma_{z0} = 0.07\ \mathrm{mm} \quad \sigma_{\phi 0} = 0.9\ \mathrm{mm}$$

$$D_{r\phi} = 0.025\ \mathrm{mm}/\sqrt{\mathrm{cm}} \quad D_z = 0.08\ \mathrm{mm}/\sqrt{\mathrm{cm}}$$

Diffusion coefficient D_z from pads still needs to be adapted for pixels

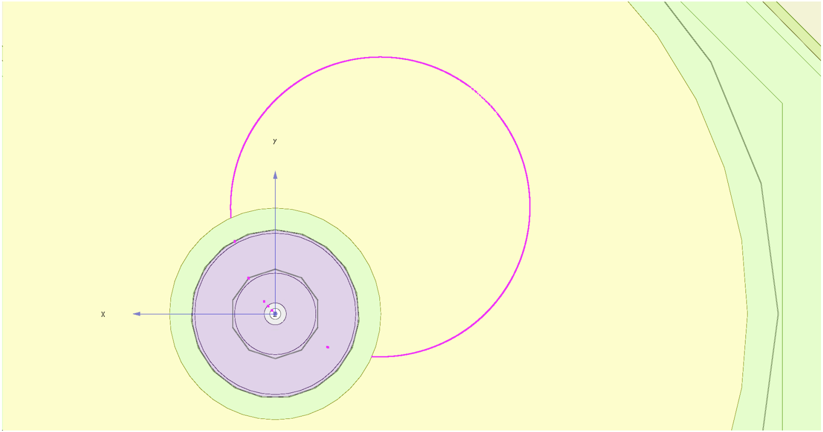


Pad simulation of a 700 MeV muon



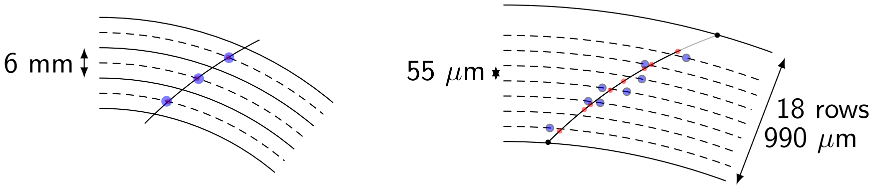
Simulated pad hits are only at layer centre crossing

Pixel simulation of a 700 MeV muon



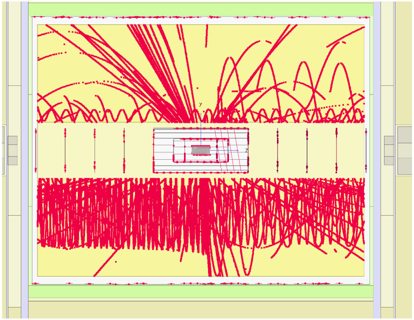
Interpolated pixel hits are placed everywhere along the track

Simulation of pad hits compared to pixel hits



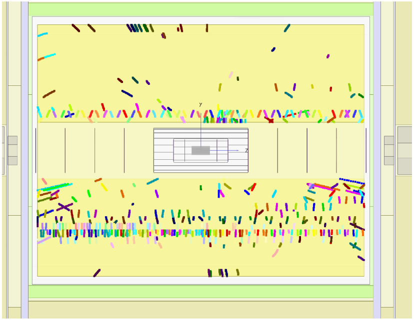
Pad hits	Pixel hits
6 mm $\times$ 1 mm	55μm $\times$ 55μm
Exactly one hit per layer	Multiple or no hits per layer
22 electrons per hit	1 electron per hit
Only diffusion in $r\phi$ and z	Diffusion in x, y and z
$\sim$200 hits per track	$\sim$10 000 hits per track

Track finding for pads using Clupatra



Tracker hits

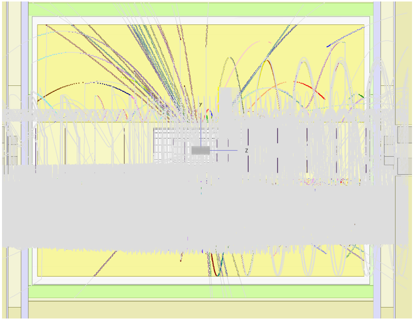
Track finding for pads using Clupatra



Seeds

1. Seed finding
   - Uses nearest neighbour clustering by distance in a pad row range of 15 rows

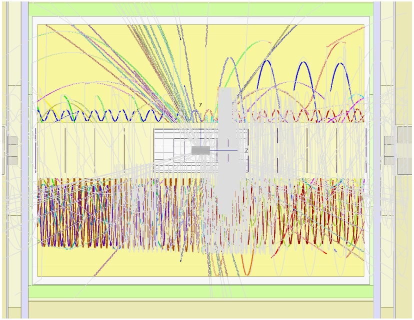
Track finding for pads using Clupatra



Seed fit and extended

1. Seed finding
   - Uses nearest neighbour clustering by distance in a pad row range of 15 rows
2. Fit track to seeds
   - use first, middle and last hit to initialise track parameters
3. Extend track inwards (and outwards)
   - Uses Kalman filter (Kaltest) in MarlinTrk

Track finding for pads using Clupatra



Track fit

1. Seed finding
   - Uses nearest neighbour clustering by distance in a pad row range of 15 rows
2. Fit track to seeds
   - use first, middle and last hit to initialise track parameters
3. Extend track inwards (and outwards)
   - Uses Kalman filter (Kaltest) in MarlinTrk
4. Merge split segments

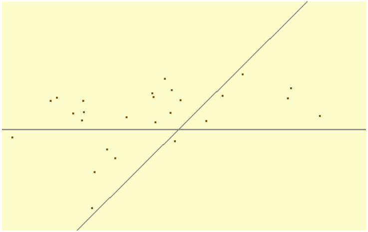
Fit tracks by Extended Kalman filter

Fit track by an Extended Kalman Filter: a recursive fitting algorithm working in steps:

- Predict state at next site using propagator $\boldsymbol{a}_k^{k-1} = \boldsymbol{f}_k(\boldsymbol{a}_k)$
  - $\boldsymbol{a}_k$ contains track parameters $(d_\rho, \phi_0, \kappa, d_z, \tan\lambda)$
- Update with measurement $\boldsymbol{m}_k$ using state-to-measurement projector $\boldsymbol{h}_k(\boldsymbol{a}_k^{k-1})$
  - Add hit and update if $\chi^2 < \chi^2_{\text{threshold}}(=35)$
  - $\boldsymbol{m}_k$ are coordinates of a cylindrical surface $(r\phi, z)$

Issues when applying pad-track-reconstruction to pixel-hits

- Seed finding: CPU time of nearest neighbour clustering scales as $\mathcal{O}(N^2)$
  Unsuitable for many thousands of pixel hits
- Track fit: initialise Kalman filter with first, middle and last hit
  3 hits do not fix the track tight enough, first hits can pull the track fit in the wrong direction

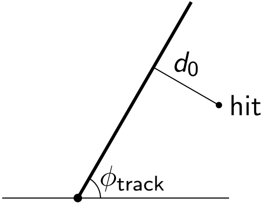


- Hits restricted to a cylindrical surface
  For pixel another representation is more suitable

Track finding for pixel TPC

- Perform clustering by ϕ (Hough-transform like)
  - Fill histogram of hits by ϕ in pad row range of 750 pixel rows
  - Maximum bin is cluster with track candidate if more than 200 hits
  - construct a straight line from the detector center to the average position
  - Cut hits on distance from this line (10mm in $r\phi$ and 3mm rz)
  - initialise track fit with this line

Track fitting for pixel hits



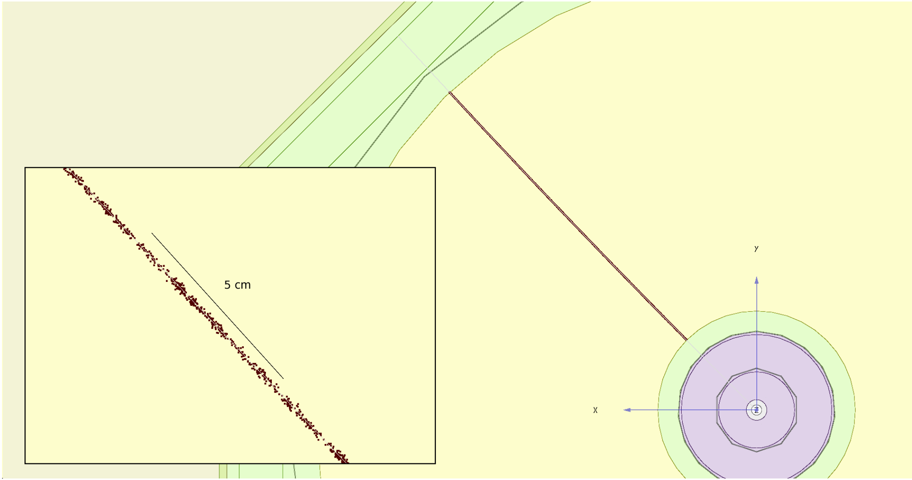
Define alternative measure with $\boldsymbol{m}_k$ as a function of $\boldsymbol{a}_k^{k-1}$

$$\boldsymbol{m}_k(\boldsymbol{a}_k^{k-1}) = \begin{pmatrix} d_0 \\ z \end{pmatrix} = \begin{pmatrix} \Delta x \sin(\phi_{\text{track}}) - \Delta y \cos(\phi_{\text{track}}) \\ z_{\text{hit}} + \tan\lambda(\Delta x \cos(\phi_{\text{track}}) + \Delta y \sin(\phi_{\text{track}})), \end{pmatrix}$$

The distance to the track d_0 better represents the measurement

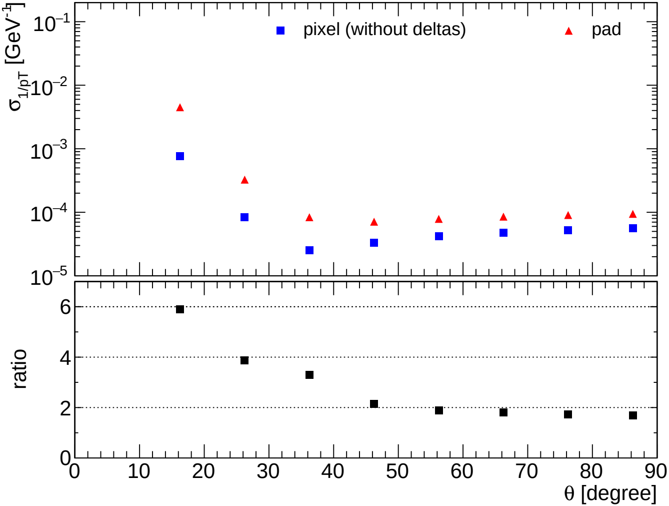
Fit of straight track

50 GeV muon



Momentum resolution from track fit

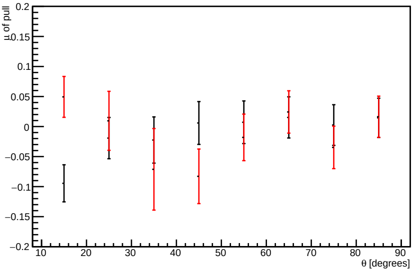
50 GeV muon

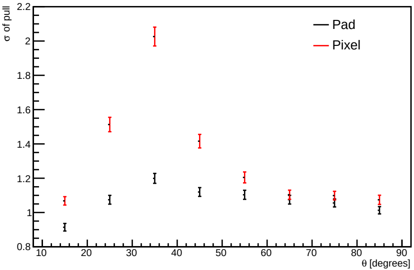


Pixels hits simulated with delta's, but rejected before reconstruction

Pull of $1/p_T$

from 8×1000 tracks of 50 GeV muons

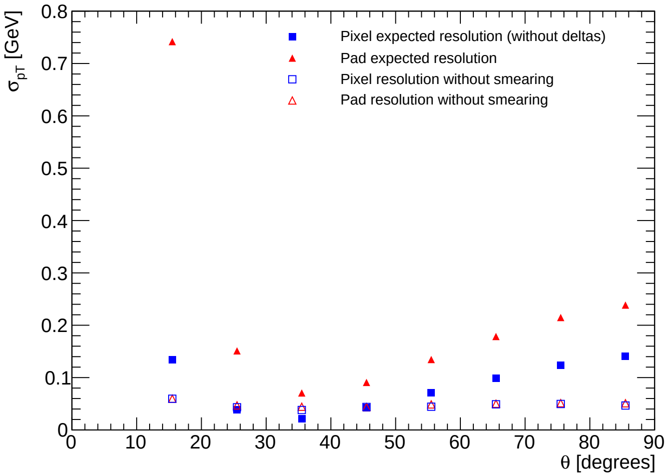




Mean μ does not indicate any biases

σ of pull is too large at angles with the best momentum resolution

Distortion of σ of pull



p_T difference between input and fit to unsmeared hits is $\sim$ 40 MeV

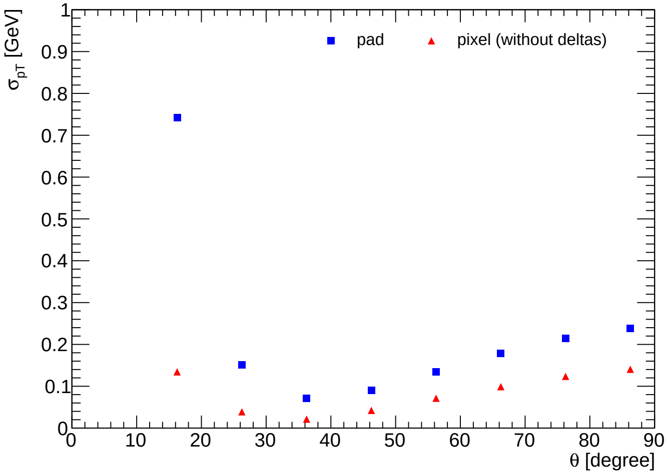
σ of pull is increased by precision settings or a bug in the code

Conclusion

- A muon track was successfully simulated and reconstructed with a pixel readout
- First estimates of the pixel readout performance show a factor $\sim 2-6$ improvement over to the pad readout
- Next steps:
  - Fix pull of track fit
  - Do delta rejection using an algorithm
  - Continue studies of performance of pixel readout
  - Investigate dE/dx performance
  - Implement an endplate layout with more realistic coverage ($\sim$ 80%)
  - Simulate and reconstruct physics events with a pixel readout

Momentum resolution from track fit covariance matrix

50 GeV muon



Extended Kalman filter

Recursive fitting algorithm to find state vector $\boldsymbol{a}_k$ and covariance $\boldsymbol{C}_k$ at site k from a series of measurements $\boldsymbol{m}_k$ by procedure:

- Predict
  - $\boldsymbol{a}_k^{k-1} = \boldsymbol{f}_{k-1}(\boldsymbol{a}_{k-1})$, where $\boldsymbol{f}_k(\boldsymbol{a}_k)$ is the state-propagator
  - $\boldsymbol{C}_k^{k-1} = \boldsymbol{F}_{k-1}\boldsymbol{C}_{k-1}\boldsymbol{F}_{k-1}^T + \boldsymbol{Q}_{k-1}$, where $\boldsymbol{F}_{k-1} = \frac{\partial \boldsymbol{f}_{k-1}}{\partial \boldsymbol{a}_{k-1}}$, and $\boldsymbol{Q}_k$ the covariance of the process noise
- Update
  - $\boldsymbol{a}_k = \boldsymbol{a}_k^{k-1} + \boldsymbol{K}_k \left(\boldsymbol{m}_k - \boldsymbol{h}_k(\boldsymbol{a}_k^{k-1})\right)$, where $\boldsymbol{h}_k(\boldsymbol{a}_k)$ the projector, $\boldsymbol{K}_k = \boldsymbol{C}_k^{k-1}\boldsymbol{H}_k^T(\boldsymbol{V}_k + \boldsymbol{H}_k\boldsymbol{C}_k^{k-1}\boldsymbol{H}_k^T)^{-1}$, $\boldsymbol{H}_k = \frac{\partial \boldsymbol{h}_k}{\partial \boldsymbol{a}_k^{k-1}}$, and $\boldsymbol{V}_k$ the coveriance of the measurement noise
  - $\boldsymbol{C}_k = \left((\boldsymbol{C}_k^{k-1})^{-1} + \boldsymbol{H}_k^T\boldsymbol{G}_k\boldsymbol{H}_k\right)^{-1}$, where $\boldsymbol{G}_k = (\boldsymbol{V}_k)^{-1}$
- (Smooth...)

See: Keisuke Fujii, Extended Kalman Filter, The AFCA-SIM-J Group

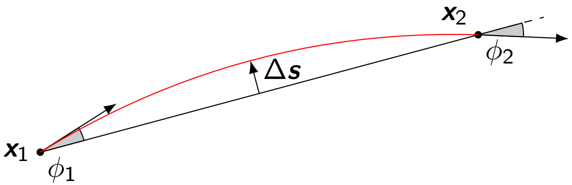
Parabolic interpolation

The position $\boldsymbol{x}(t)$ between the points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is parametrised as a function of $0 \geq t \geq 1$

$$\boldsymbol{x}(t) = \boldsymbol{x}_1 + t(\boldsymbol{x}_2 - \boldsymbol{x}_1) + 4t(1-t)\Delta\boldsymbol{s}, \qquad (1)$$

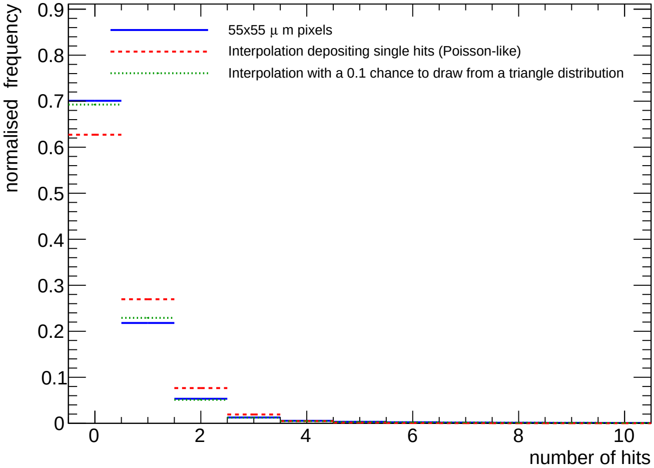
where $\Delta\boldsymbol{s}$ is the deflection midway given by

$$|\Delta\boldsymbol{s}| = \frac{|\boldsymbol{x}_2 - \boldsymbol{x}_1|}{4} \sin(\Delta\phi_{12}/2). \qquad (2)$$

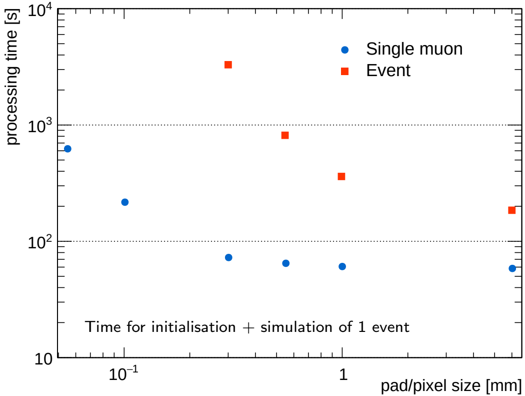


Distribution of hits along the track

- Ionization in gas follows roughly a Landau distribution
- Approximate by a combination of a Poisson and a triangle (for now)



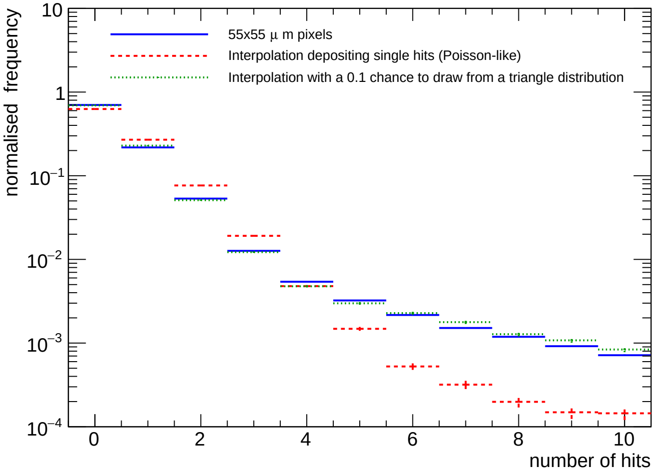
Simulating 55 × 55 μm^2 pixels as small pads costs too much processing time



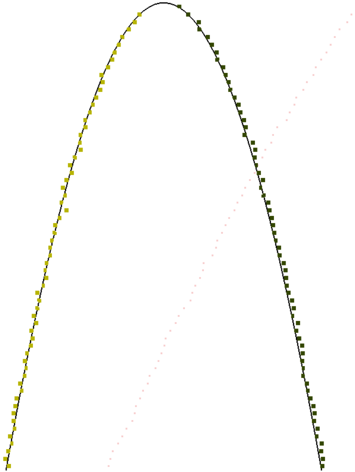
- Processing time increases rapidly at smaller pixel sizes

Distribution of hits along the track

Distribute hits with a $P(N_{\text{hits}} = N) \simeq 0.1 \cdot \frac{2N}{N_{\text{total}}^2}$ chance to deposit multiple hits



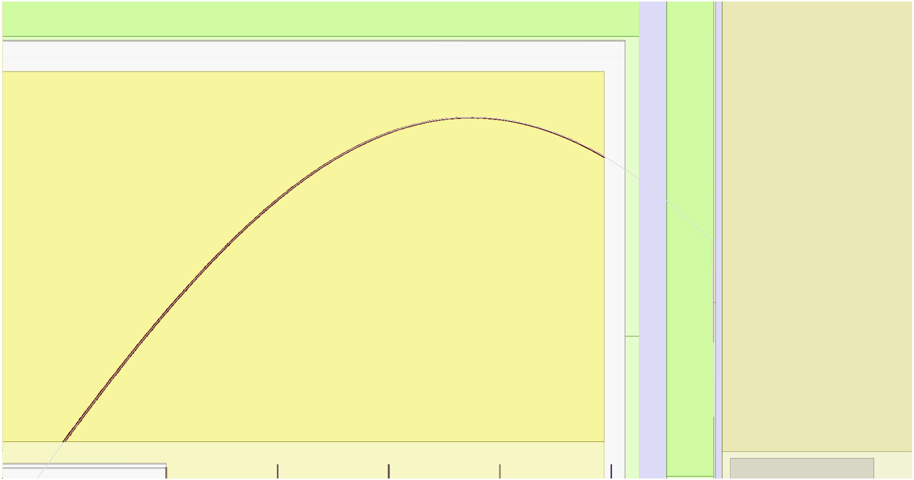
Track fitting for pads



Track fit: For curled (low momentum) tracks, cluster inward and outward parts separately and merge

Fit of curled track

1 GeV muon without energy loss



Fit of curled track

1 GeV muon without energy loss