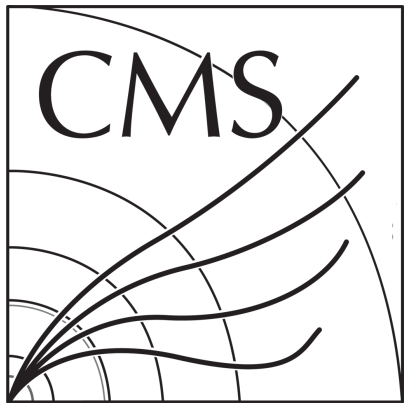
Offline Electron Seeding Validation - Update

Caleb Fangmeier
Ilya Kravchenko, Greg Snow

University of Nebraska - Lincoln

EGamma Workshop — November 21, 2017



Introduction

- Our goal is to study **seeding** for the **offline** GSF tracking with the **new pixel detector**.
- Specifically, we want to optimize the window sizes used in the new pixel-matching scheme already implemented in HLT.
- Since last update[1],
  - Migrated Code from `9_0_2` to `9_2_8`
  - Integrated the new pixel matching into the trackingNtuple. (although still a work-in-progress)
  - Regenerated `trackingNtuples` for dataset

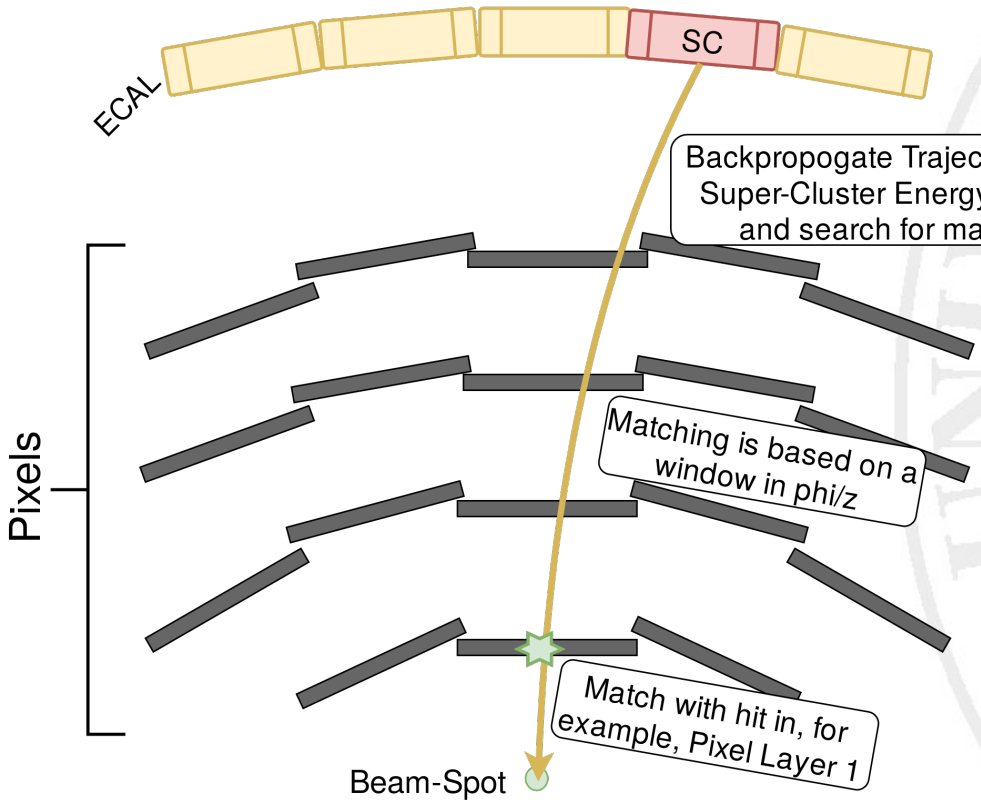
`/ZToEE_NNPDF30_13TeV-powheg_M_120_200/`

`RunIISummer17DRStdmix-NZSFlatPU28to62_92X_upgrade2017_realistic_v10-v1/GEN-SIM-RAW`

  - Ongoing work happening here: https://github.com/cfangmeier/cmssw/tree/ValidationGsfTracks928_dev
- This Talk:
  - Description of current Offline electron seeding
  - Description of current HLT (future Offline) electron seeding
  - Plans for 2018

[1] https://indico.cern.ch/event/662743/contributions/2744847/attachments/1534642/2403597/main.pdf

Pair Electron Seeding I

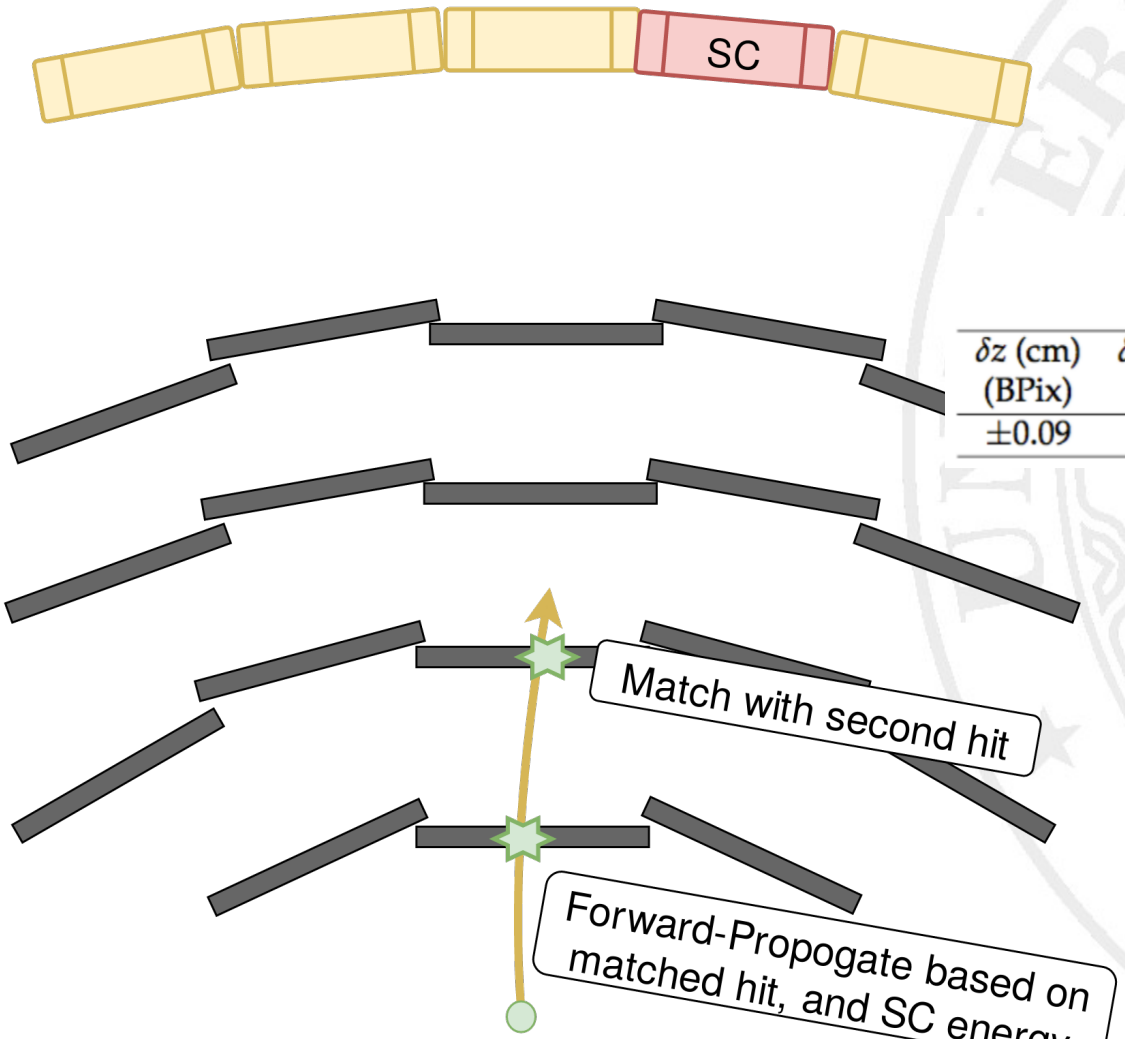


First window

E_T^{SC}(GeV)	δz (BPix)	δr (FPix or TEC)	$\delta\phi$ (rad) (positive charge)
≤ 5	$\pm 5\sigma_z$	$\pm 5\sigma_z$	$[-0.075; 0.155]$
10	$\pm 5\sigma_z$	$\pm 5\sigma_z$	$[-0.046; 0.096]$
≥ 35	$\pm 5\sigma_z$	$\pm 5\sigma_z$	$[-0.026; 0.054]$

Windows from https://indico.cern.ch/event/611042/contributions/2464057/attachments/1406271/2148742/ElectronTracking30112016.pdf

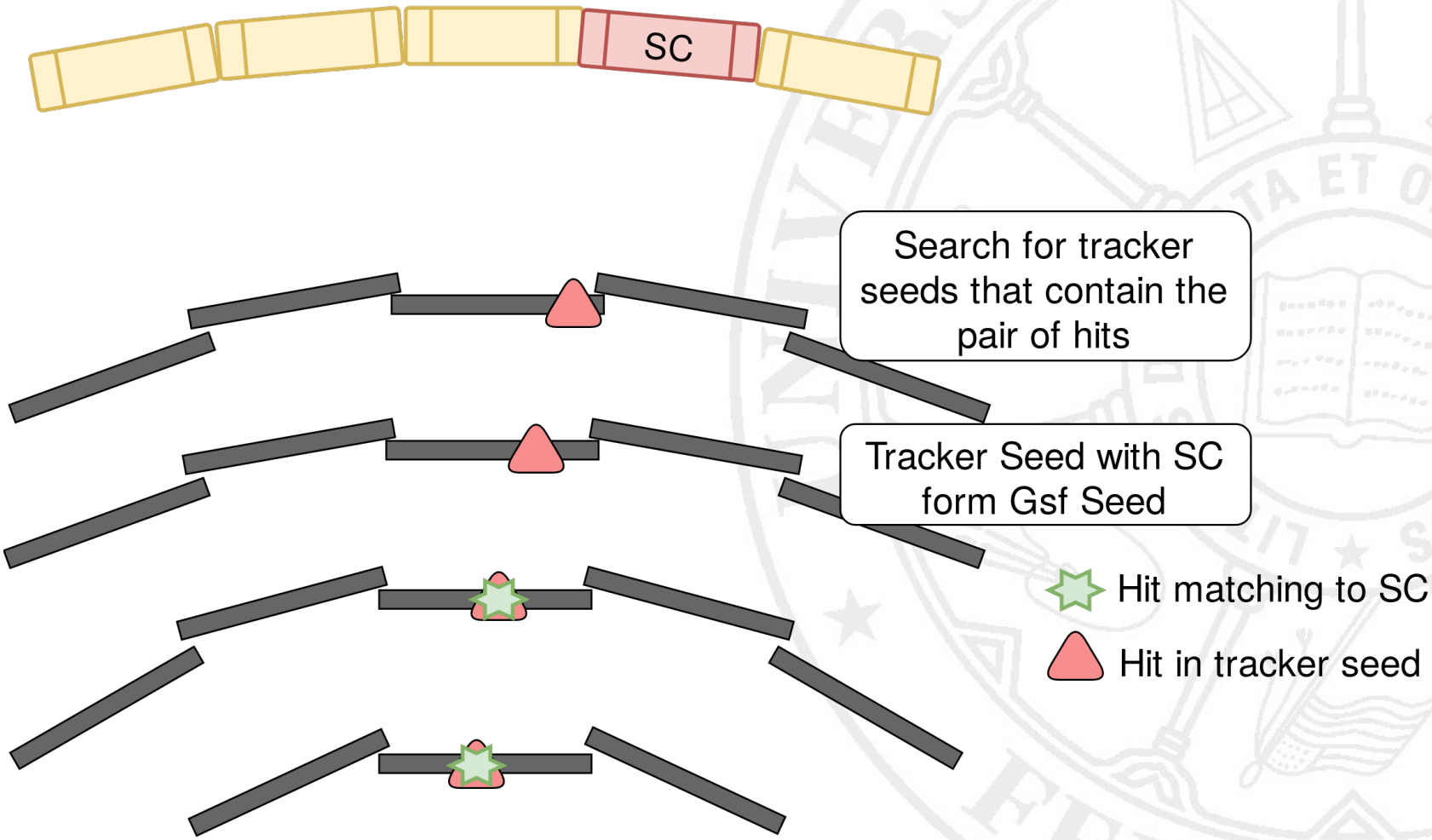
Pair Electron Seeding II



Second window

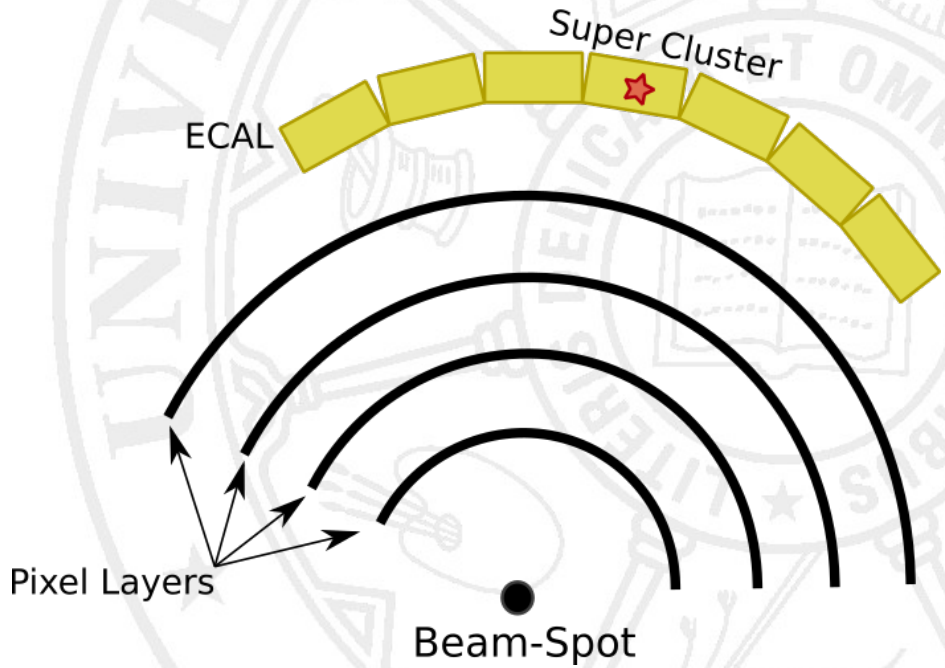
δz (cm) (BPix)	δr (cm) (FPix)	δr (cm) (TEC)	$\delta\phi$ (rad) (BPix)	$\delta\phi$ (rad) (FPix or TEC)
± 0.09	± 0.15	± 0.2	± 0.004	± 0.006

Pair Electron Seeding III



Triplet Electron Seeding - Setup

- Begin with ECAL super cluster and beam spot

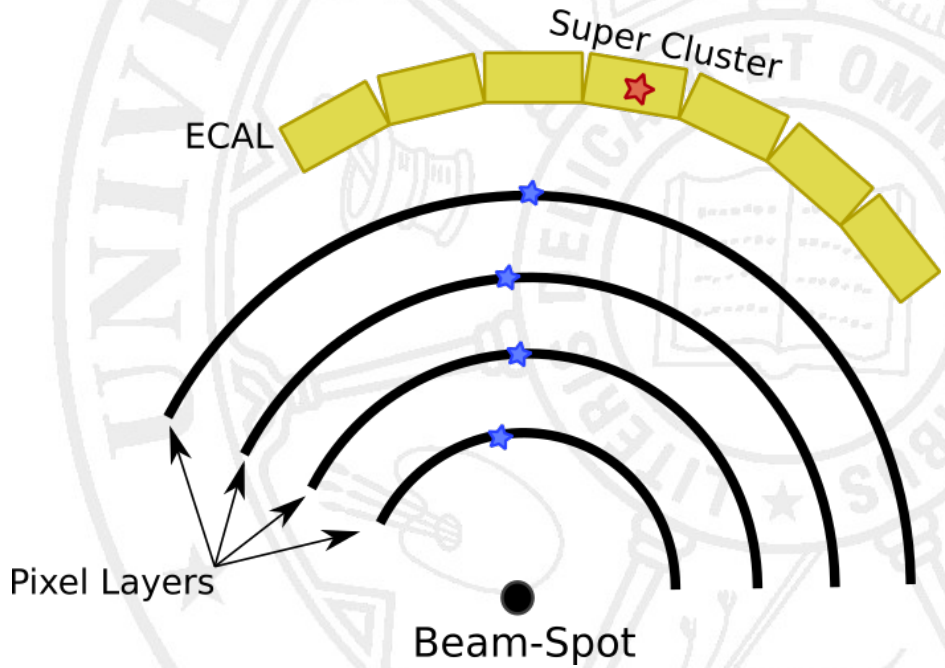


Triplet Electron Seeding - Introduce Seed

- Now, examine, one-by-one seeds from general tracking*
- Note that we do not look at all hits in an event, but rather rely on general tracking to identify seeds.

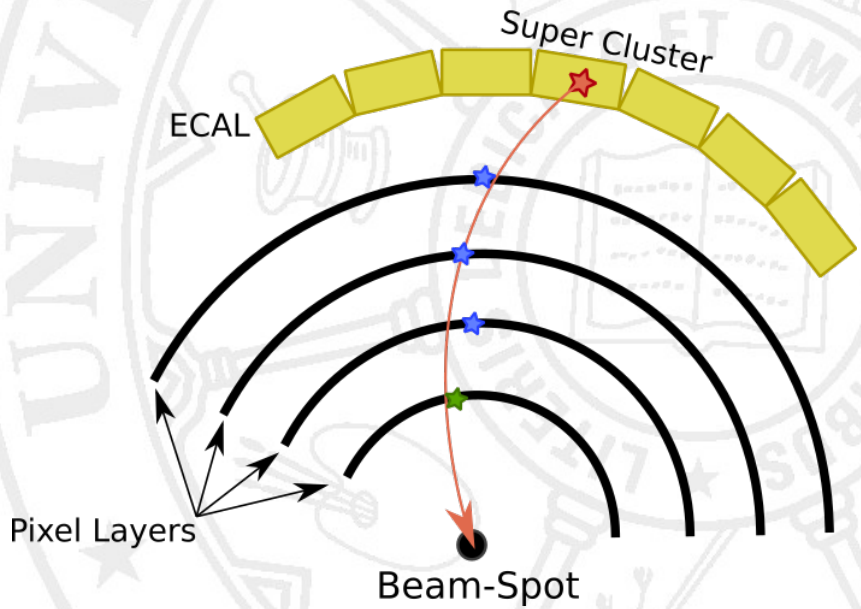
---

*initialStepSeeds, highPtTripletStepSeeds, mixedTripletStepSeeds, pixelLessStepSeeds, tripletElectronSeeds, pixelPairElectronSeeds, stripPairElectronSeeds



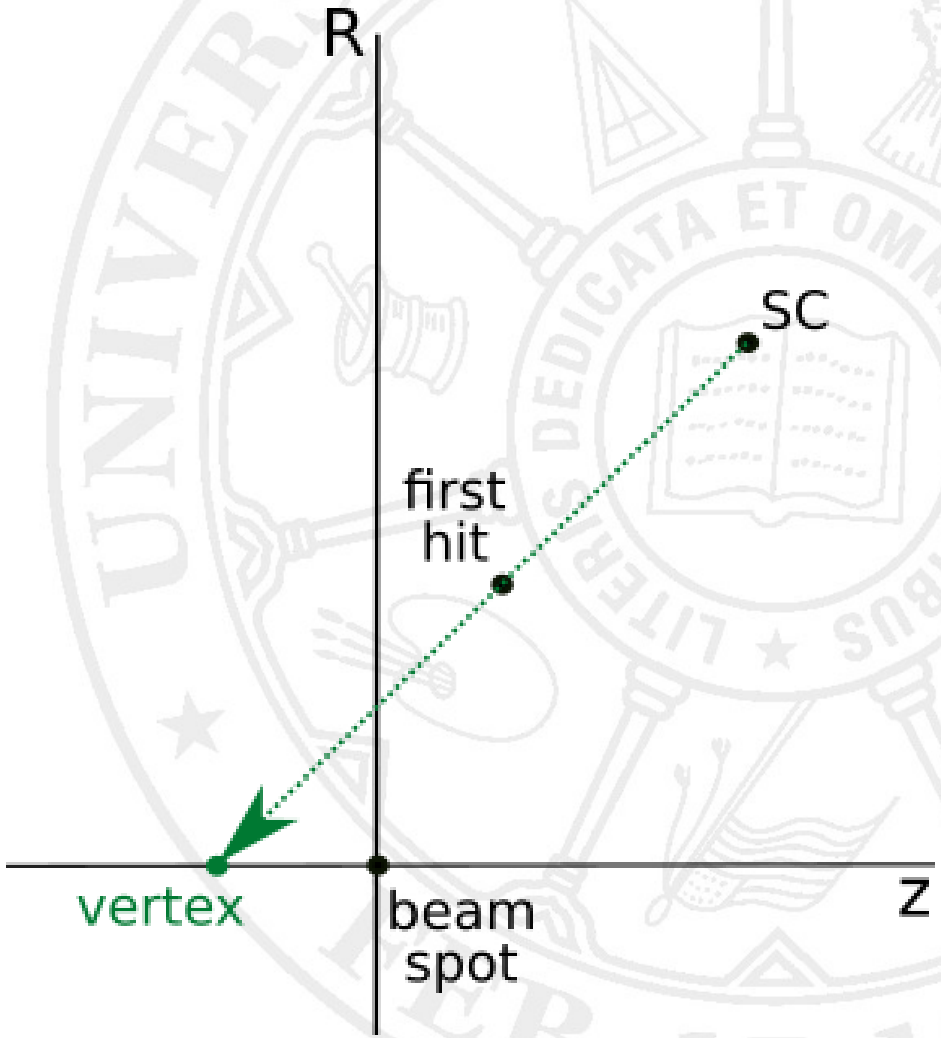
Triplet Electron Seeding - Match First Hit

- Using the beam spot, the SC position, and SC energy, propagate a path through the pixels.
- Next, require the first hit to be within a $\delta\phi$ and δz window. ($\delta\phi$ and δR for FPIX)
- δz window for first hit is huge as SC and beam spot positions give very little information about z.



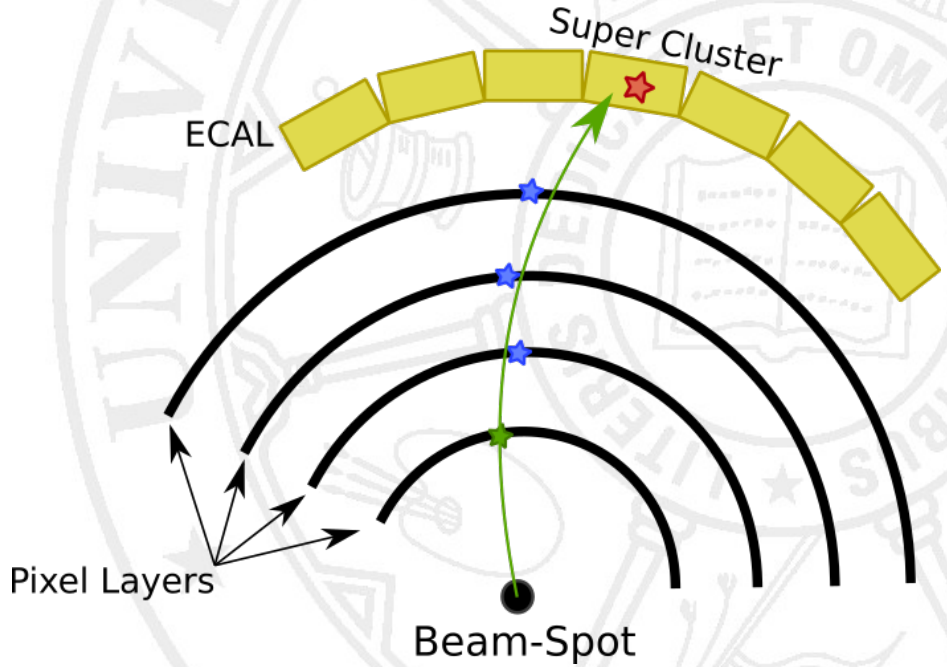
Triplet Electron Seeding - Extrapolate Vertex

- Once we have a matched hit, use it with the SC position, to find the vertex z.
- Vertex x and y are still the beam spot's.
- Just a simple linear extrapolation.



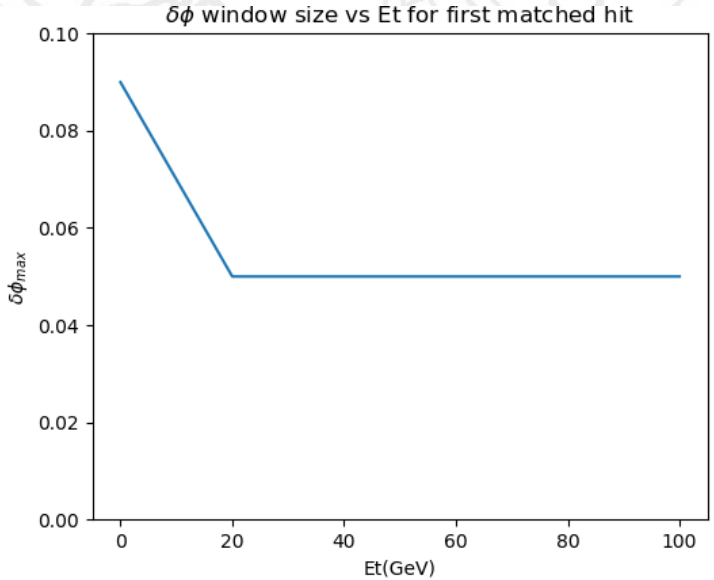
Triplet Electron Seeding - Match Other Hits

- Now forget the SC position, and propagate a new track based on the vertex and first hit positions, and the SC energy.
- Progress one-by-one through the remaining hits in the seed and require each one fit within a specified window around the track.
- Quit when all hits are matched, or a hit falls outside the window. No skipping is allowed.
- However, *layer* skipping is not ruled out if the original seed is missing a hit in a layer



Triplet Electron Seeding - Window Sizes

- The $\delta\phi$ and $\delta R/z$ windows for each hit are defined using three parameters.
  - `highEt`
  - `highEtThreshold`
  - `lowEtGradient`
- From these, the window size is calculated as $\texttt{highEt} + \min(0, \texttt{Et} - \texttt{highEtThreshold}) * \texttt{lowEtGradient}$.
- For the first hit, these parameters for the $\delta\phi$ window are,
  - $\texttt{highEt} = 0.05$
  - $\texttt{highEtThreshold} = 20$
  - $\texttt{lowEtGradient} = -0.002$
- For the first hit, these parameters for the $\delta\phi$ window are,

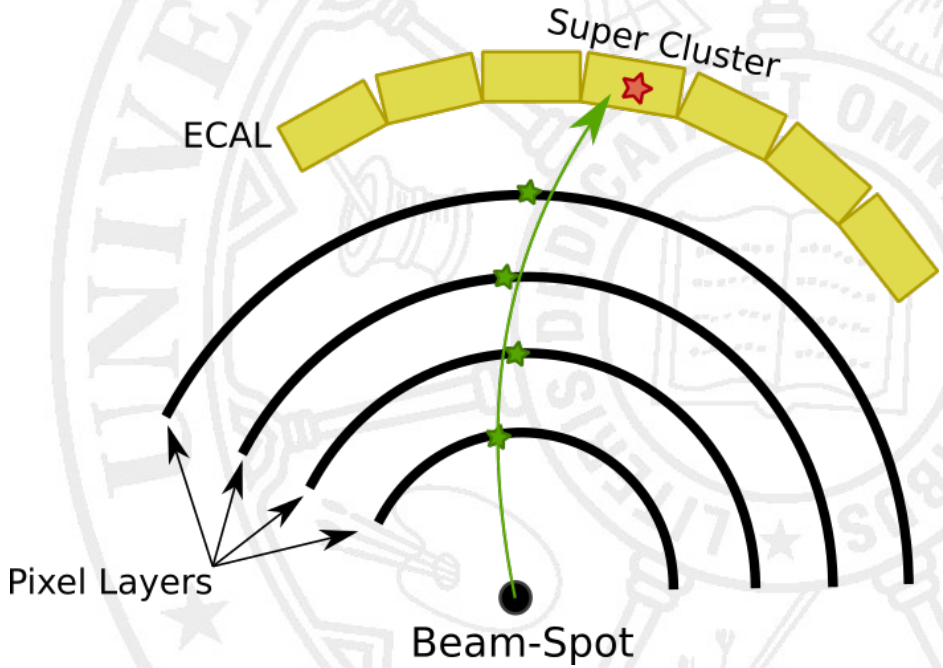


These parameters can be specified for each successive hit, and in bins of η, so optimization is a challenge!

Triplet Electron Seeding - Handle Missing Hits

- Finally, calculate the expected number of hits based on the number of working pixel modules the track passes through.
- Require exact[1] number of matched hits depending on the expected number of hits.
  - If $N_{\text{exp}} = 4$, require $N_{\text{match}} = 3$
  - If $N_{\text{exp}} < 4$, require $N_{\text{match}} = 2$
- If the seed passes all requirements, all information, including
  - Super cluster
  - Original Seed
  - Residuals (For both charge hypotheses)

  are wrapped up and sent downstream to GSF tracking



[1] Exact, rather than minimum to deal with duplicate seeds in input collection. Could switch to minimum with offline cross-cleaned seeds.

Outlook and Plans for 2018

- Construct framework to measure efficiencies and fake-rates using MC-truth information.
- Use this framework to identify sources of inefficiency.
- Finally, optimize the window sizes for offline reconstruction.