Stacked Track Trigger Simulations

Track Trigger Simulations Working Group
(D. Acosta, A. Ballado, L. Fields, M. Fisher,R. Frazier, C. Fountas, I.K. Furić, I. Furic, C. Harder, T. Kamon, L. Kaplan, A. Madorskiy, D. Newbold, A. Rose, A. Ryd, B. Scurlock, M. Weinberger, B. Williams)

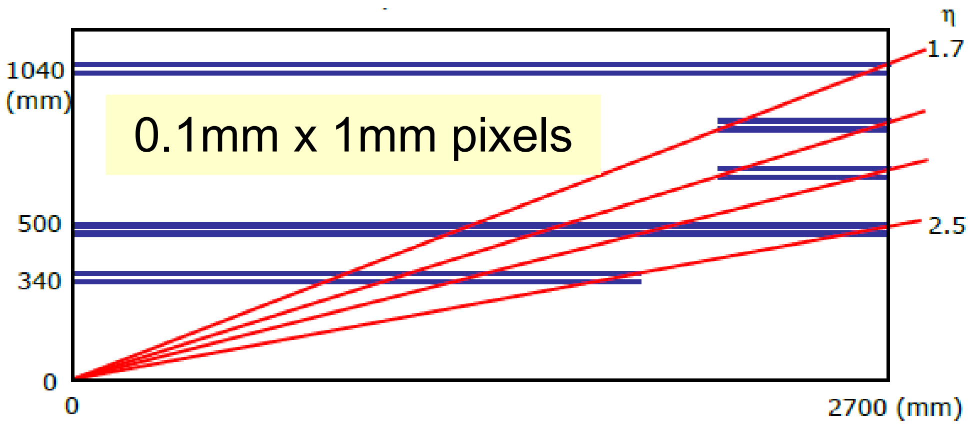
29 Oct 2009

Outline

- Basic Track Trigger Primitive (TTP) Simulations
- Algorithm Studies
  - Electrons
    - Fast Simulation
  - Muons
  - Taus
- Conclusion

TTP Simulations in Fast Sim

- Start w/ CMSSW and geometries provided by the tracker upgrade simulation group:



0.1mm x 1mm pixels

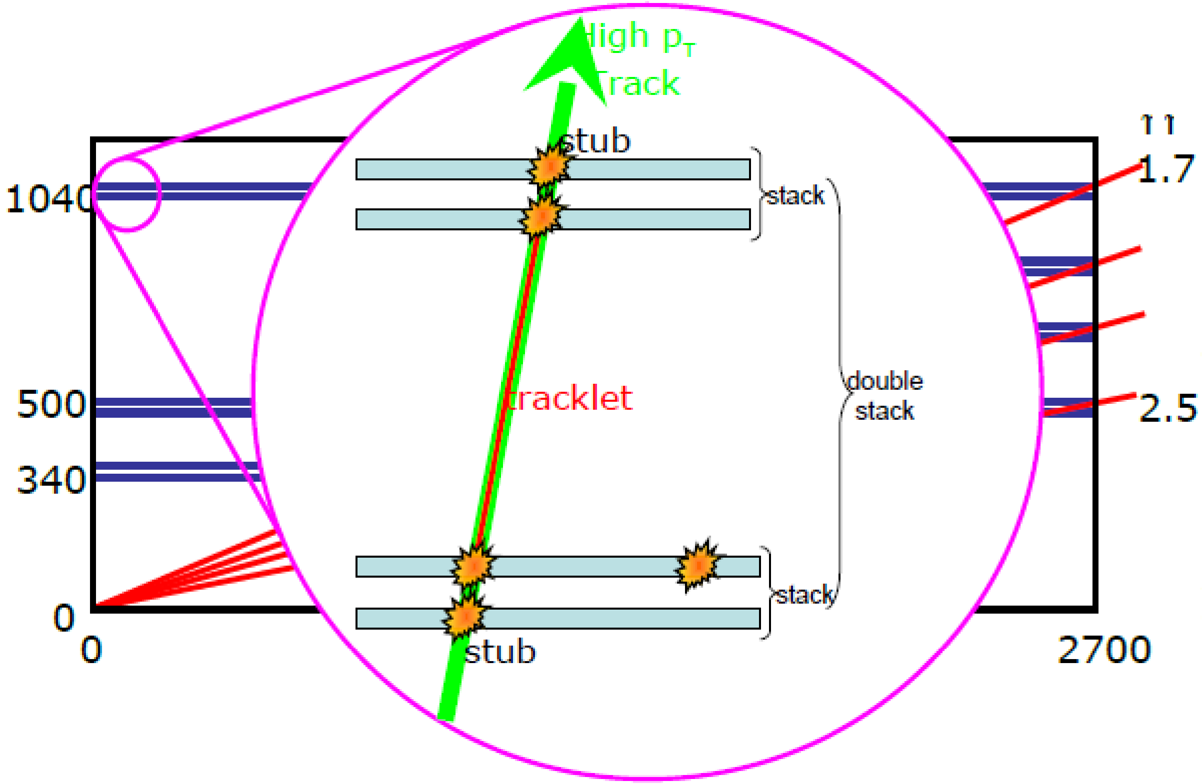
“Long Barrel”
(“Hybrid” and
“Strawman B”
also exist)

- What we get from CMSSW:
  - ”SimHits” - Exact position of each particle
  - ”Digis” (a.k.a. ”Trigger Hits” or just ”Hits”): simhit info is pixelized, charge sharing is implemented (and inefficiencies are added)

Only very recently implemented;
results today do not include this

TTP Simulations in Fast Sim

TTP Terminology:



Stack:pair of closely spaced sensors (~1mm)

Stub:correlated pair of hits in stack

Double stack:Two stacks separated by few cm.

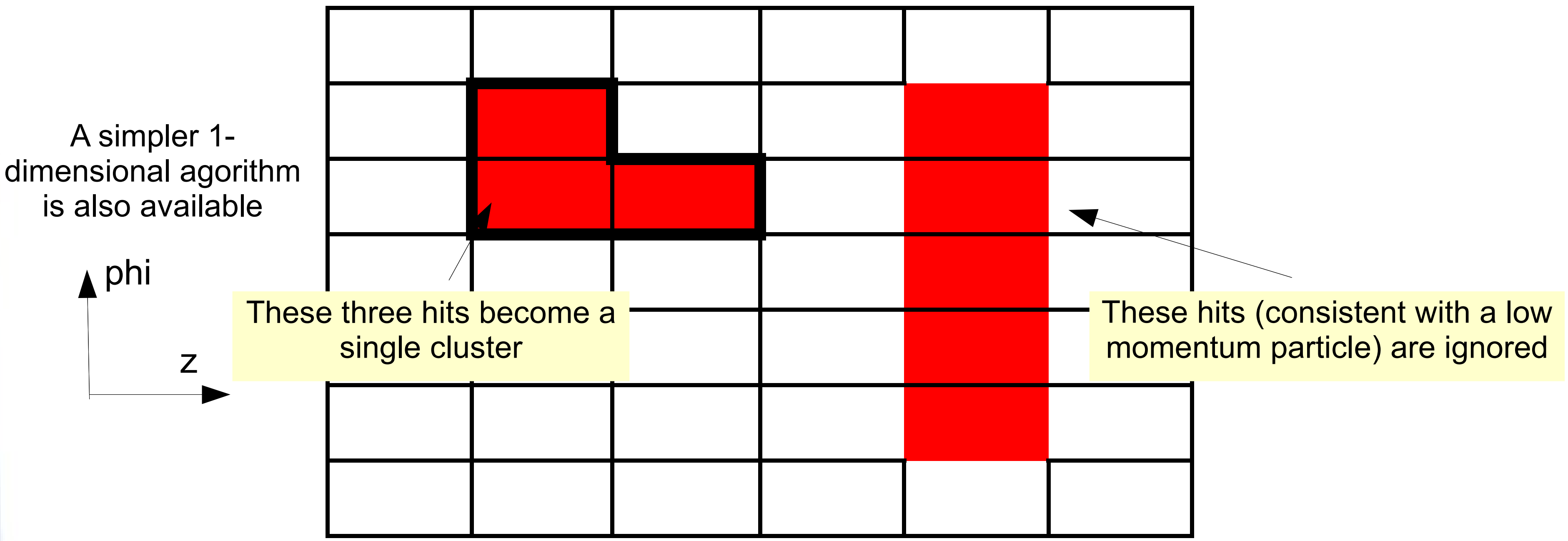
Tracklet A matched pair of stubs.

A layer is one stack in this talk.

https://twiki.cern.ch/twiki/bin/view/CMS/SLHCTrackerTerminology

TTP Simulations in Fast Sim

- What the stacked track trigger software (written by A. Rose and others) does:



A simpler 1-dimensional agorithm is also available

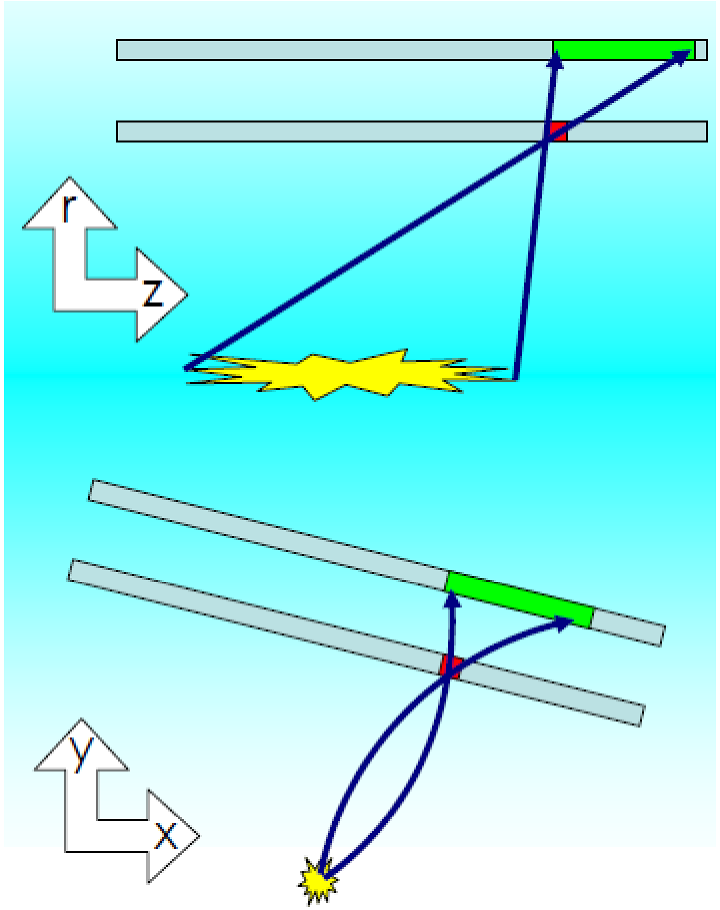
These three hits become a single cluster

These hits (consistent with a low momentum particle) are ignored

- First performs clustering on digis. The "2d” clustering algorithm:
  - Identifies 2d clusters that are less than 3 pixels wide in phi
  - Assumes that each pixel will have knowledge of its immediate neighbors
    - Also, a ”kill bit” that supresses clusters 3+ pixels wide

TTP Simulations in Fast Sim

- More on what the stacked track trigger software does:

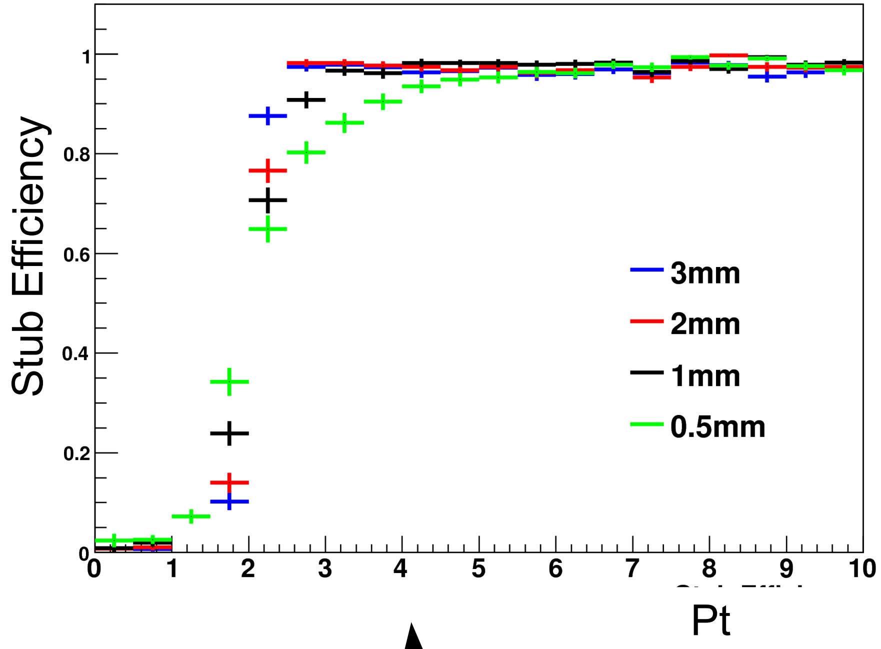


- Once clusters are identified, clusters within a single stacked module are matched using a "hit matching algorithm”
- Hit matching algorithms start with each cluster found on an inner layer and search for clusters on the outer layer that are within a matching window
  - The matching window can be defined in a number of ways
  - Current default: phi window is defined using a Pt threshold; z window requires stubs point to luminous region

Could be implemented as look-up tables on-detector

TTP Simulations in Fast Sim

TTP Efficiencies:

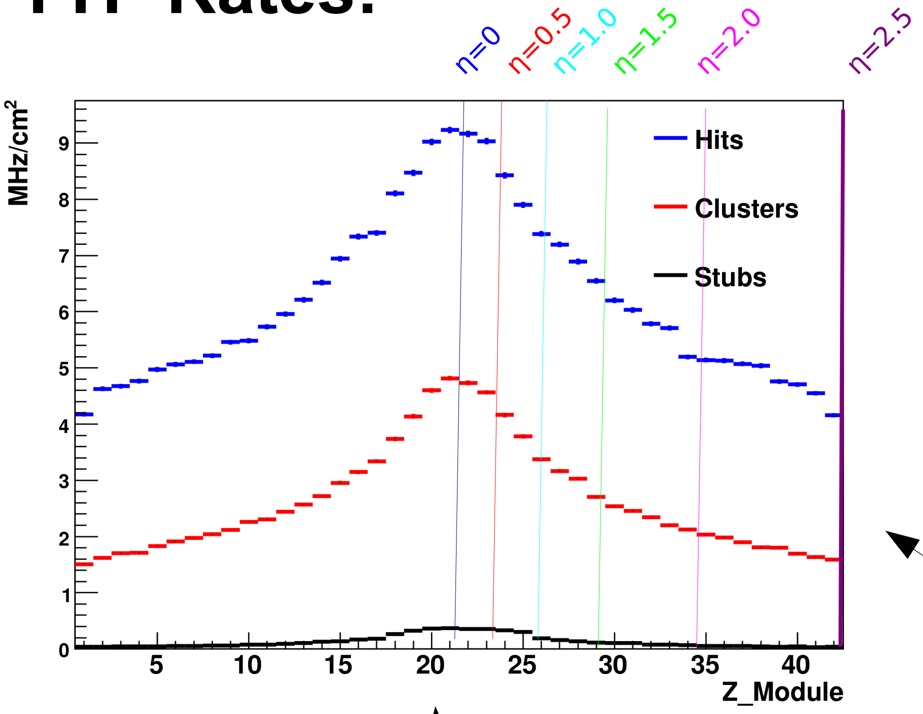


Stub Efficiencies using the standard Long Barrel Geometry/ 2d Clustering algorithm (First layer)

- Recall: default hit matching algorithm calculates phi-matching window based on Pt threshold
- Small efficiency at 0-1 GeV mostly independent of stack separation
- Turn on curve is sharper for wider stack separation
- Above turn on, efficiencies are O(1%) smaller for wider stack separations

TTP Simulations in Fast Sim

TTP Rates:



- Clustering reduces rates by ~factor of 3 (from hit rates)
- Stub rate is factor of ~20 less than cluster rate

Divide these numbers by 40 MHz to get rate per bunch crossing

Hit/Cluster/Stub Rates using the standard Long Barrel Geometry/ 2d Clustering algorithm (First layer)

These rates should be taken with a grain of salt; rates in fastsim are factor of ~3 lower than in fullsim (M. Pesaresi)

TTP Simulations in Fast Sim

Peak TTP Rates (in MHz/cm2):

2 GeV Stub/Tracklet Threshold ↓

Stack Sep →	0.5mm	1mm	2cm	3cm
SimHits	5	5	5	5
Digis	9	9	9	9
Clusters	4	4	4	4
Stubs	0.3	0.3	0.27	0.25
Tracklets	0.1	0.15	0.18	0.17

1 mm Stack Separation ↓

Stub/Tracklet Threshold →	1 GeV	2GeV	3GeV	5GeV
SimHits	5	5	5	5
Digis	9	9	9	9
Clusters	4	4	4	4
Stubs	0.9	0.3	0.14	0.06
Tracklets	0.3	0.15	0.05	0.01

- Peak rates occur at η=0
- Simhit rates are equivalent to charged particle rates
- Digi rates are larger than simhit rates due to charge sharing
- Cluster rates are smaller than simhit rates because large clusters have been ignored
- Stub/Tracklet rates vary modestly with stack separation, strongly with stub threshold
- Stub rates are dominated by efficiencies below threshold
- Tracklet rates are dominated by efficiencies above threshold

Divide these numbers by 40 MHz to get rate per bunch crossing

Electron Studies

L. Fields, A. Ryd
(Cornell University)

Electron Studies @ Cornell

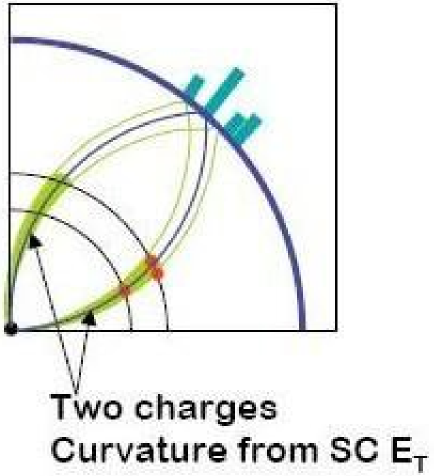
The Goal: **To study efficiencies and rates associated with EID at L1**

- How do effs/rates change when tracker information is added?
- How do results change when more layers are added?

Recent Updates

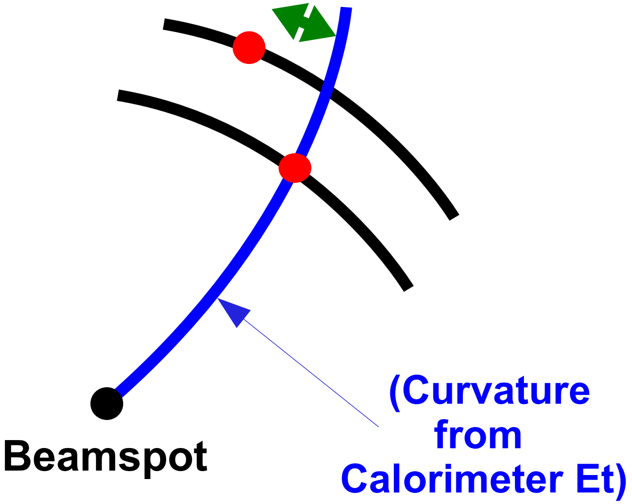
- Now using updated calorimeter code (w/ isolated electrons)
- Clustering has been implemented (via the ”2d” algorithm)
- Full long barrel geometry → Long barrel w/ rings
- Gaps between modules in z (significant source of inefficiency) reduced
- Added configurations with multiple layers (e.g. Requiring stubs on 4/4 or 4/6 layers)

Electron Studies @ Cornell



The algorithm is based on existing HLT strip algorithm and starts with L1Electrons from calorimeter

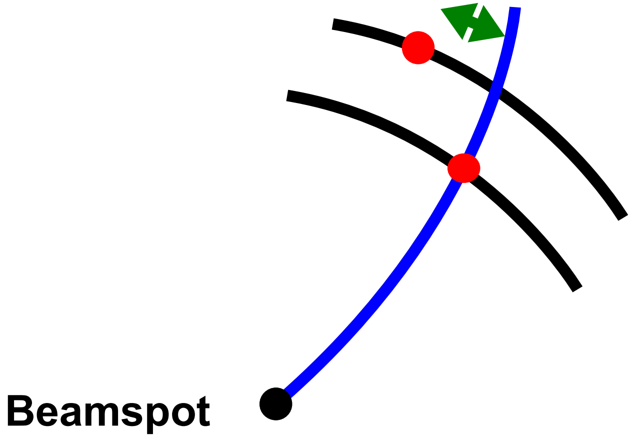
- **First step: Look for *single* stubs consistent with L1Electron**
- **Second step: Match stubs into *pairs* consistent with L1Electron**



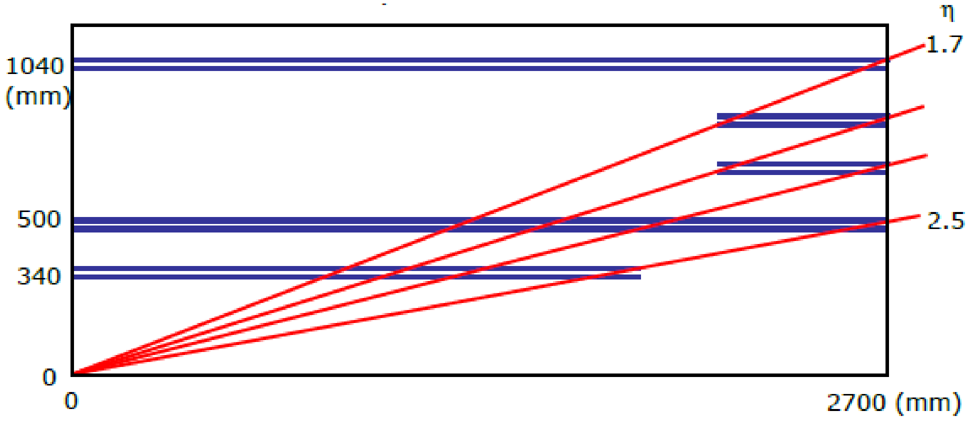
- **Loose z cuts are also made in both steps**
- **For electrons, almost all rate supression comes in the 2nd phi cut, which requires that momentum measured in the tracker and calorimeter match**

Electron Studies @ Cornell

- Our most basic results look at reconstructing electrons by requiring stubs on the first two long barrel layers

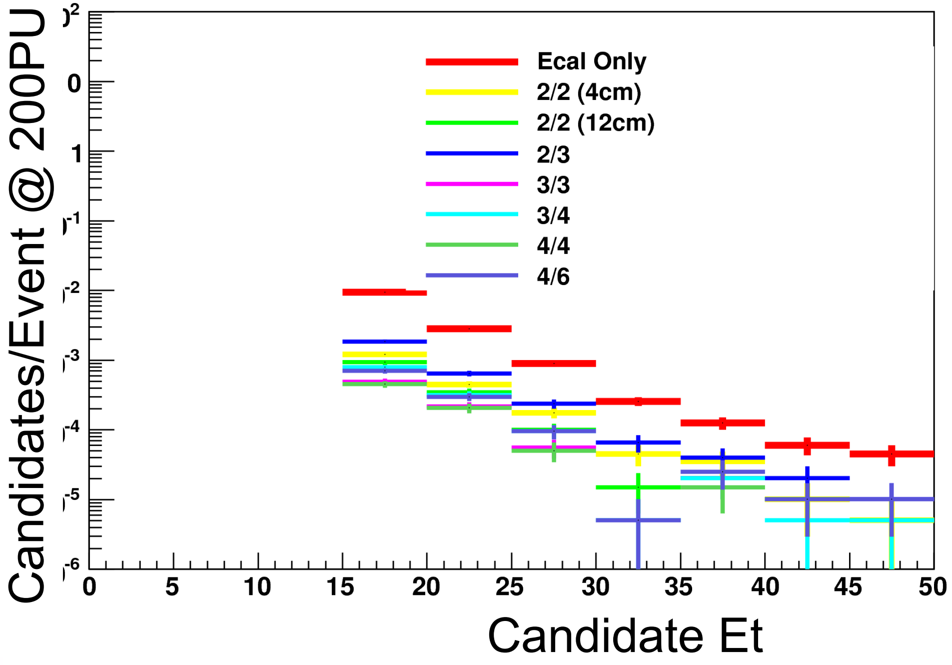


- We have also started to look at more of the many options that would be possible with the long barrel geometry



- How do results change when you require stubs on the e.g. three out the first four layers? Or all four of the first four layers?

Electron Studies @ Cornell



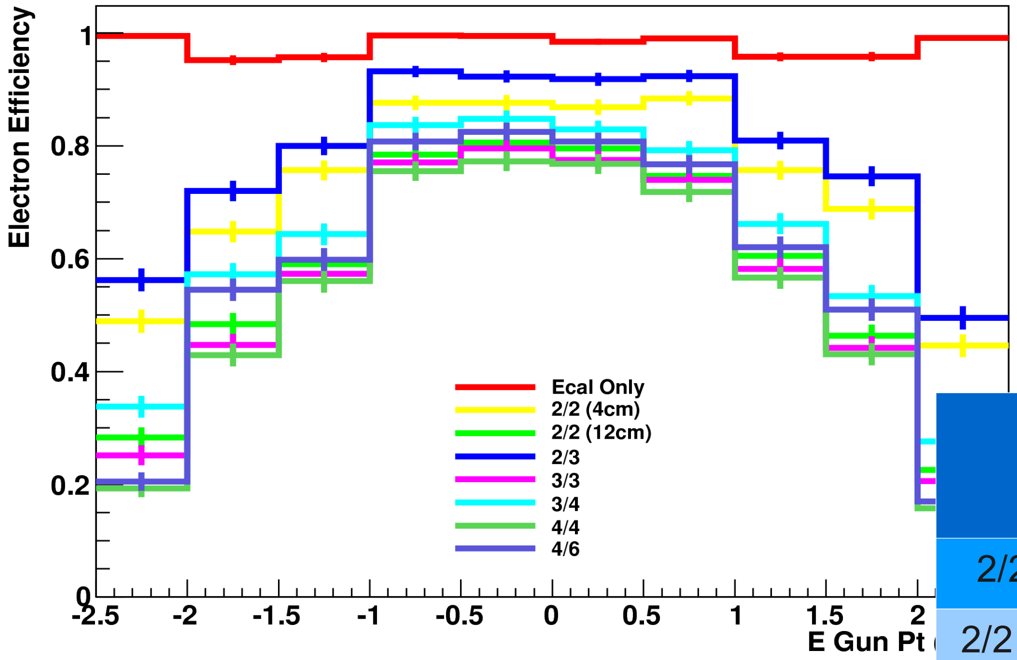
Rates vs Et

Rate Supression Factors
And Efficiencies

Configuration	Rate supression (Efficiency) for 20<Et<50 GeV
2/2 (Layers 1+2 – 4cm sep)	9 (74%)
2/2 (Layers 2+3 – 12 cm sep)	15 (60%)
2/3 (Layers 1+2+3)	8 (80%)
3/3 (Layers 1+2+3)	19 (58%)
3/4 (Layers 1+2+3+4)	15 (62%)
4/4(Layers 1+2+3+4)	20 (51%)
4/6 (all layers)	15 (57%)

Rates are reduced by factors of ~10-20, depending on the number of tracker layers required for reconstruction. Efficiencies also vary, peaking at ~80% → why?

Electron Studies @ Cornell



Efficiencies vs Eta

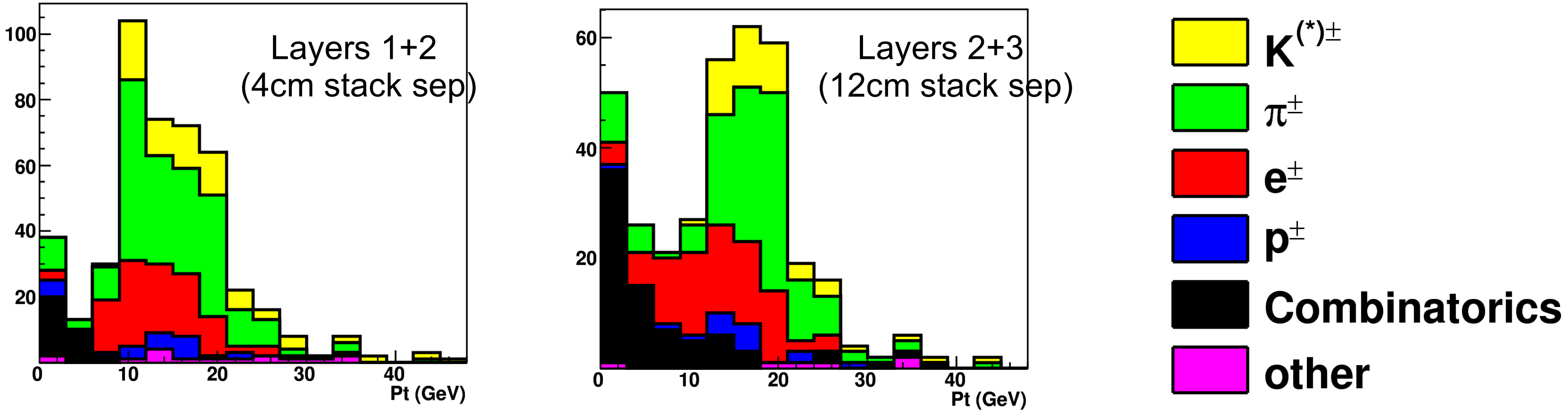
Efficiencies are strongly dependent on rapidity → material effects are a factor. We are working on quantifying all of the sources of inefficiency

Requiring stubs in 2/3 of first three layers gives rate supression of 8 and efficiencies of over 90% in central region

Configuration	Rate supression (Efficiency) for 20<Et<50 GeV
2/2 (Layers 1+2 – 4cm sep)	9 (74%)
2/2 (Layers 2+3 – 12 cm sep)	15 (60%)
2/3 (Layers 1+2+3)	8 (80%)
3/3 (Layers 1+2+3)	19 (58%)
3/4 (Layers 1+2+3+4)	15 (62%)
4/4(Layers 1+2+3+4)	20 (51%)
4/6 (all layers)	15 (57%)

Electron Studies @ Cornell

Electron Backgrounds:



Identity of tracks causing electron backgrounds when stubs are required in 2/2 layers

- Most backgrounds come from real tracks (mostly pions, kaons, electrons)
- Small combinatoric background (mixed hits from 2+ particles) increases with stack separation

(More) Electron Studies

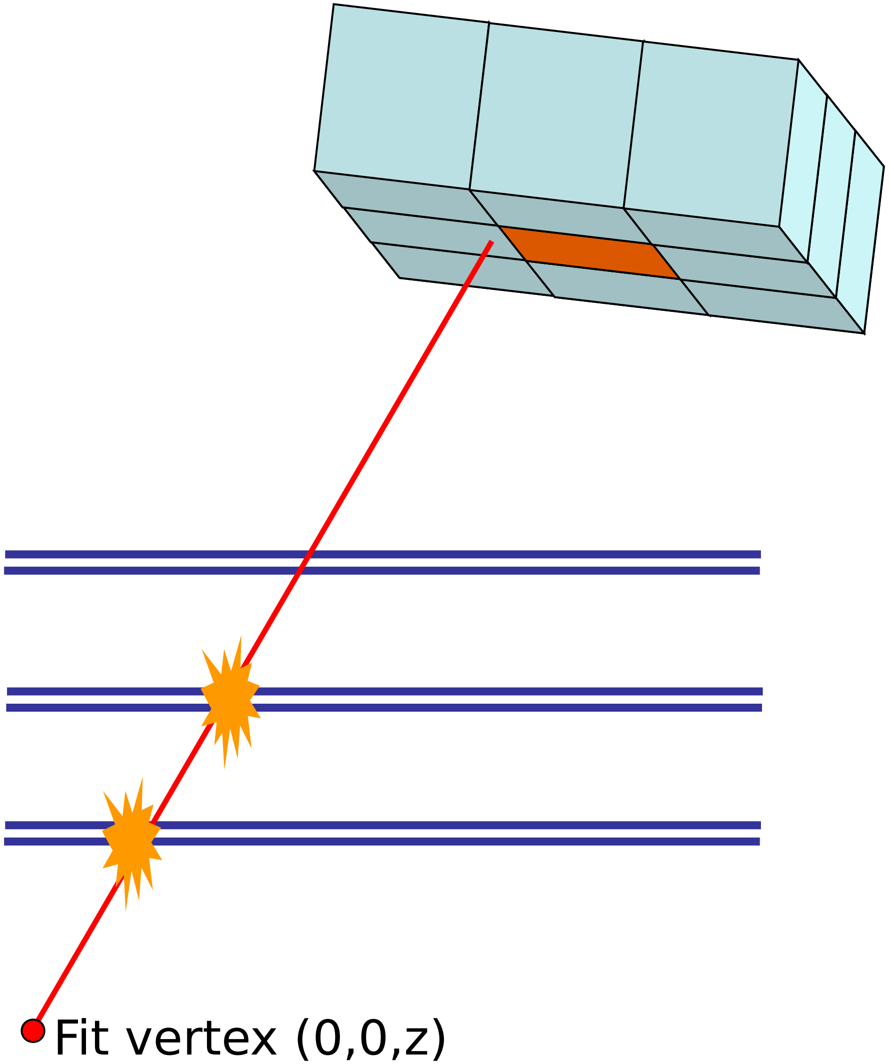
A. Rose, C. Fountas
(Imperial College)

29 Oct 2009

Electron Studies @ Imperial

The “inside-out” electron trigger:

- Start with two point tracklet
- Project to calorimeter
- Match against calo trigger primitive by Δη and ΔΦ
- Match against calo trigger primitive by E_T vs. P_T



Electron Studies @ Imperial

Motivation: Why Inside-Out?

- Each tracklet will point to either one or zero calo trigger primitives. The normal out-to-in approach may have multiple stubs per calo trigger primitive. Is one scenario easier to handle than the other?
- With this method, tracklets may be produced in parallel with calo trigger primitives rather than as part of a sequential reconstruction.
- The number of tracklets is lower than the number of stubs – less data to handle

Electron Studies @ Imperial

The Technical Details:

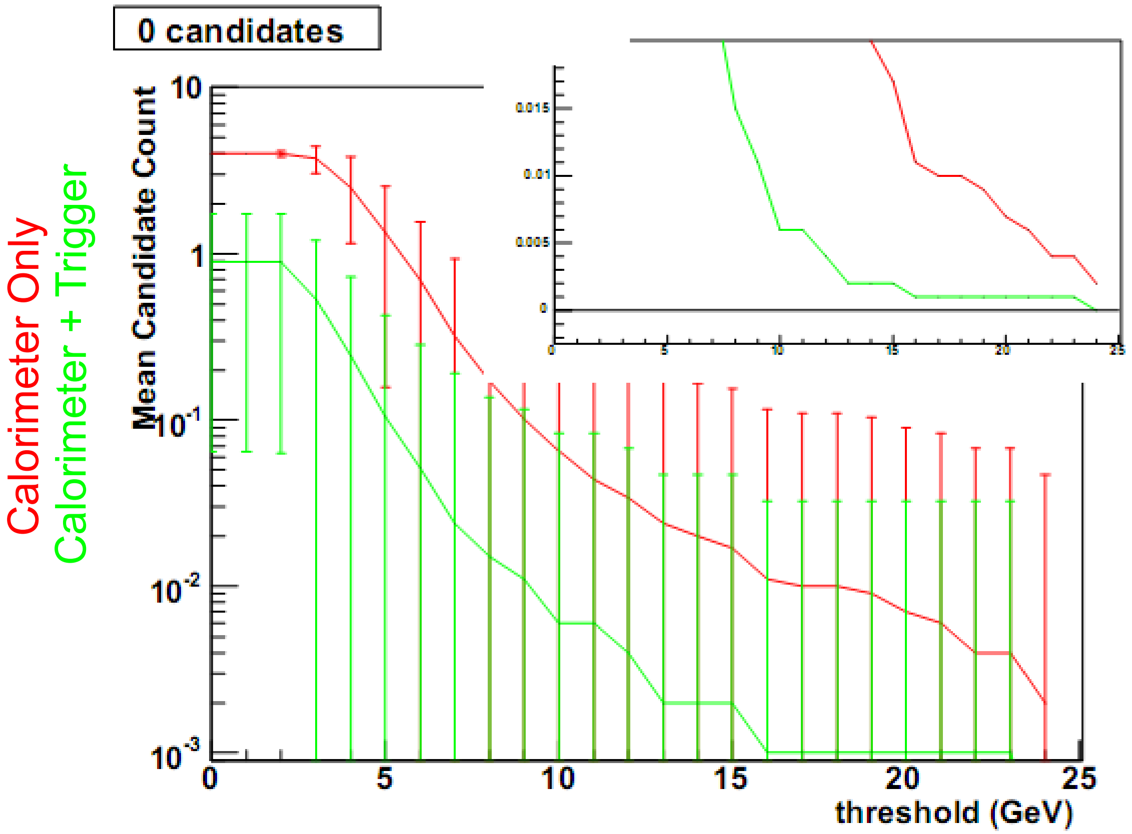
- CMSSW 2_2_6
- Fast Simulation

- Strawman-B Geometry
  - Due to some issues with Long Barrel when studies started
  - Includes correction to the geometry to reduce module z-spacing from 4mm (standard but incorrect) to 800um (correct)

- 2D clustering algorithm
- Global geometry hit matching algorithm

Electron Studies @ Imperial

Candidate Counts @ 200 PU (no signal)



The number of calo trigger primitives with...

At least one tracklet pointing to it

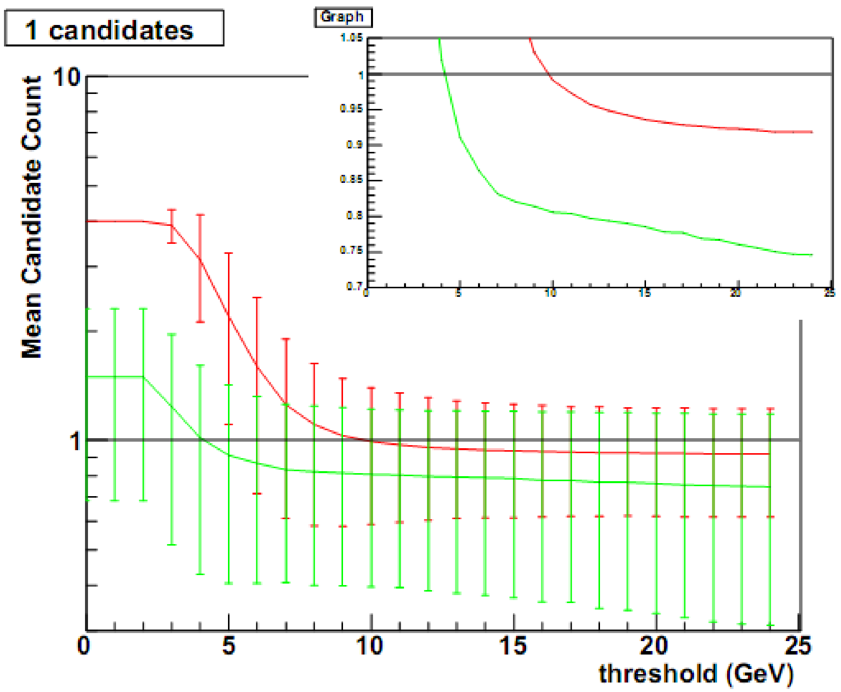
...where...

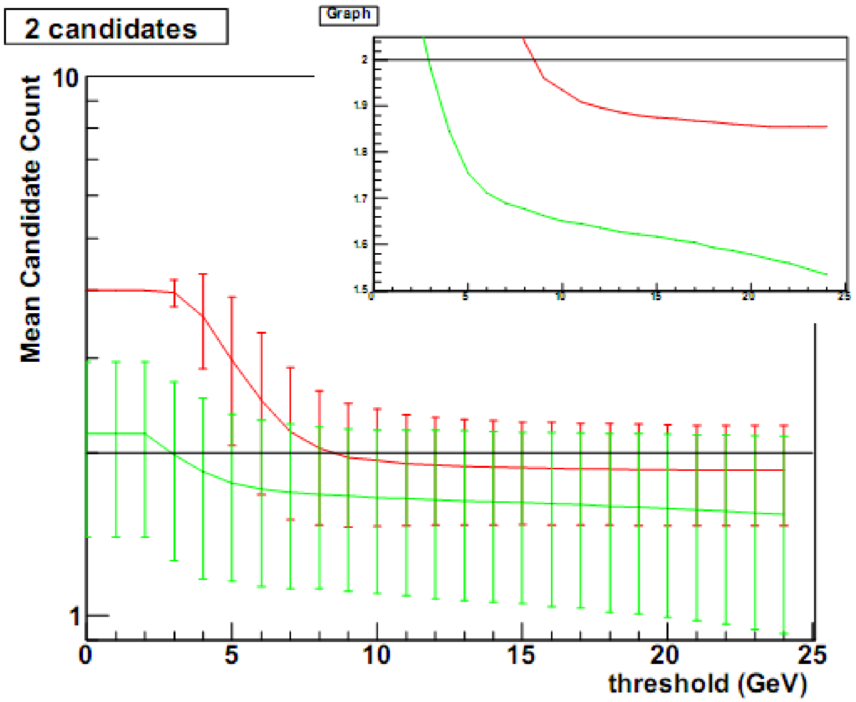
Both the tracklet and calo trigger primitive over threshold

Clearly, statistics are limited; larger samples to follow

Electron Studies @ Imperial

Candidate Counts @ 200 PU w/ 1 or 2 50 GeV electrons





Although the rate reduction is good, efficiency is low!!!
Statistics are limited → larger samples to follow.

Electron Studies @ Imperial

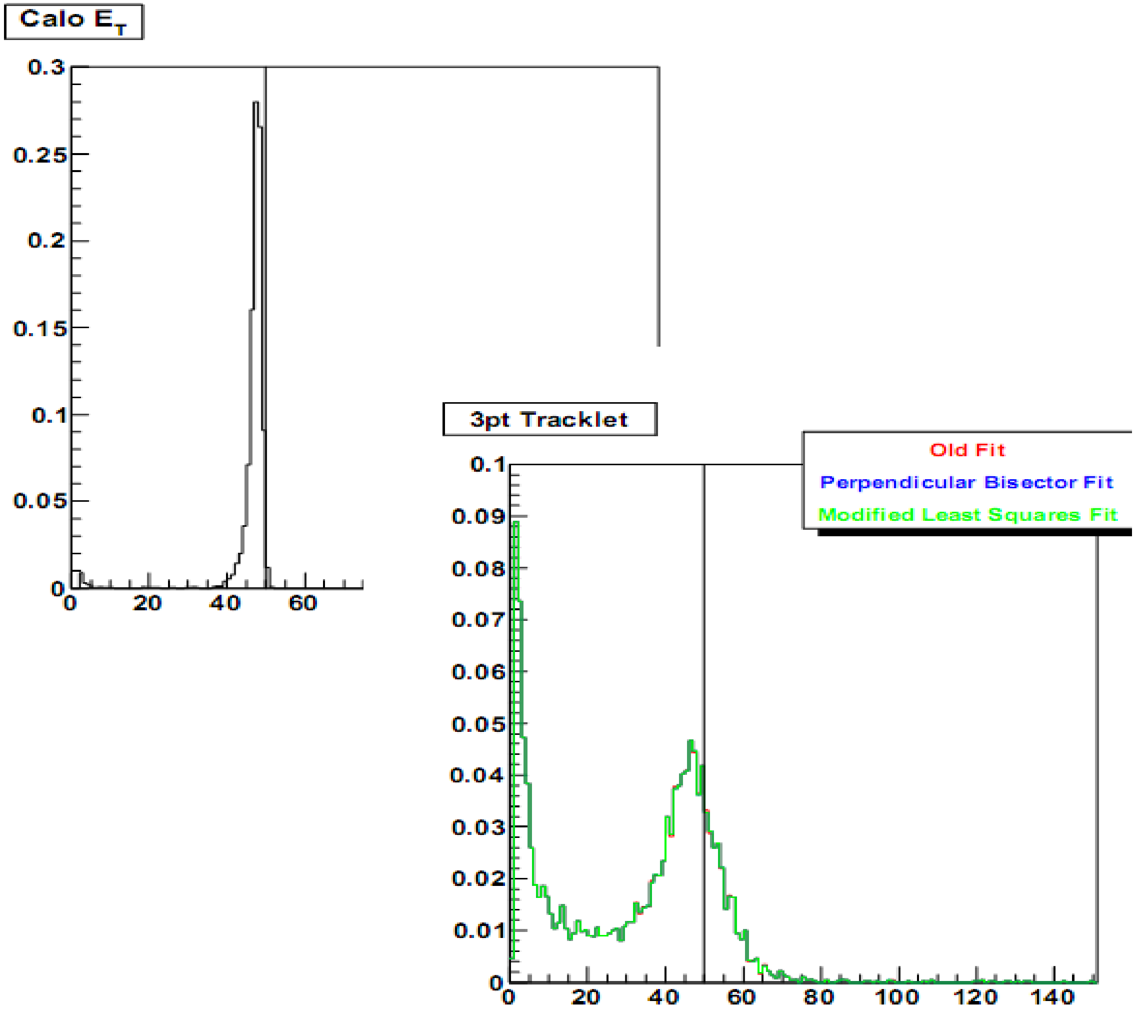
Why are efficiencies so low?

- Studies show that inefficiency is due to tracklets failing the P_T cut.
- That is, for 50GeV electrons, around 30% of reconstructed tracks have a momentum below 20GeV.

- Why?
- How can we improve this?

Electron Studies @ Imperial

Pt Resolution (50 GeV electrons, no PU)



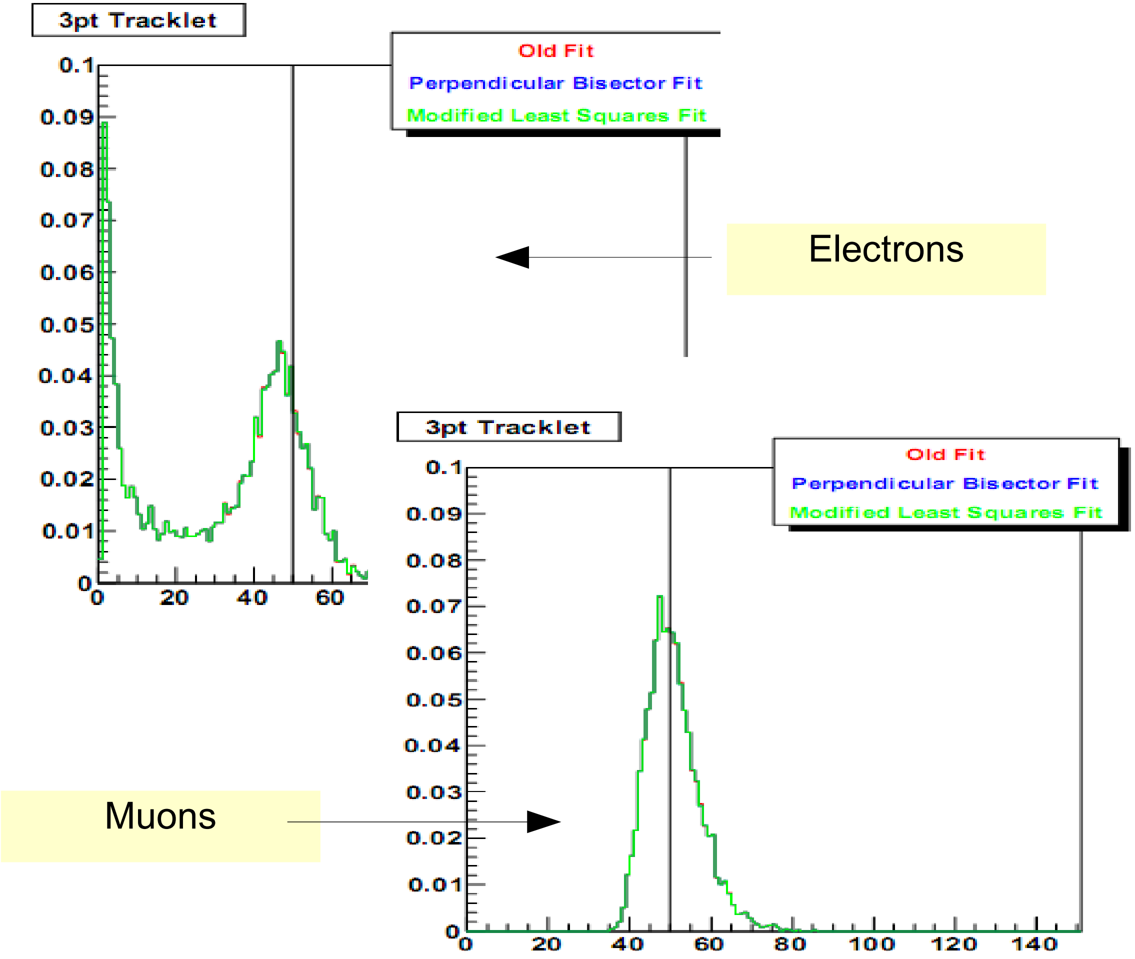
- Calo trigger primitive energy resolution is excellent.
- Tracklet P_T measurement has poor resolution due to close spacing of stacks.
- Where does the low P_T tail come from?

Hypothesize Bremsstrahlung…

(try with muons instead)

Electron Studies @ Imperial

Pt Resolution (50 GeV muons, no PU)



Electrons

Muons

- Three point tracklet resolution for muons is fair
- Most significantly: there is no low P_T peak

- Bremsstrahlung and material effects have a significant impact on tracklet P_T measurement.
- Causes ~30% efficiency loss for E_T/P_T cut!

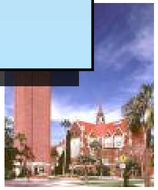
Muon Studies

D. Acosta, A. Ballado, M. Fisher, I. K. Furić, L. Kaplan,
A. Madorskiy, B. Scurlock, B. Williams

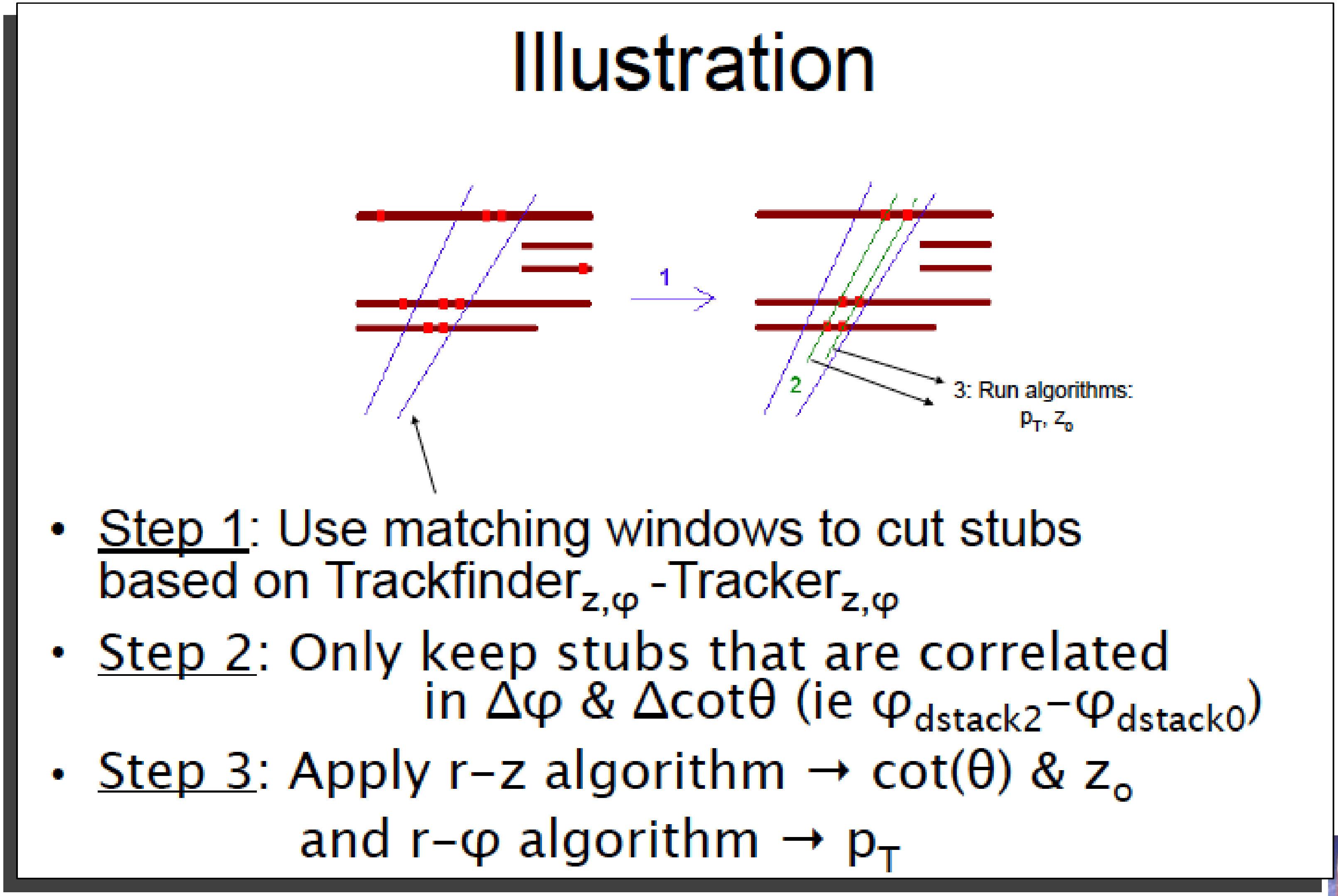
(University of Florida)

CSCTT Algorithm

- Define regions of interest to help pre-sparsify tracker readout
- Assume stub information is read out from tracker
- Define narrow roads in ϕ, z to further filter tracker readout
- Tracker stubs have excellent positional resolution - utilize internal correlations
- Attempt fit using tracker-only information (best measurement at low momenta)
- **Current CSCTT model developed in context of the Long barrel geometry**
- **CSCTT code should be in CVS this week**

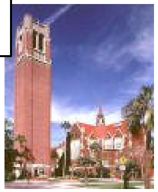


CSCTT Algorithm



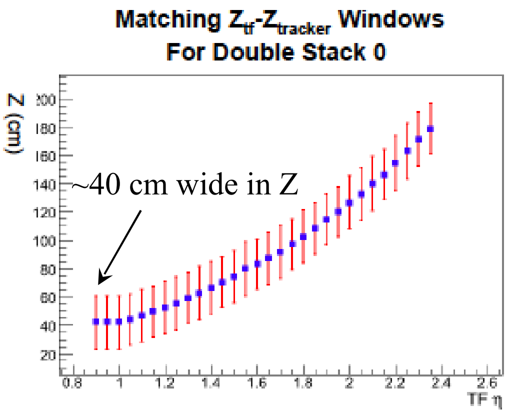
Illustration

- Step 1: Use matching windows to cut stubs based on Trackfinder$_{z,\varphi}$ -Tracker$_{z,\varphi}$
- Step 2: Only keep stubs that are correlated in $\Delta\varphi$ & $\Delta\cot\theta$ (ie $\varphi_{dstack2}-\varphi_{dstack0}$)
- Step 3: Apply r−z algorithm → $\cot(\theta)$ & z_o and r−φ algorithm → p_T



CSC+Trigger Matching Windows

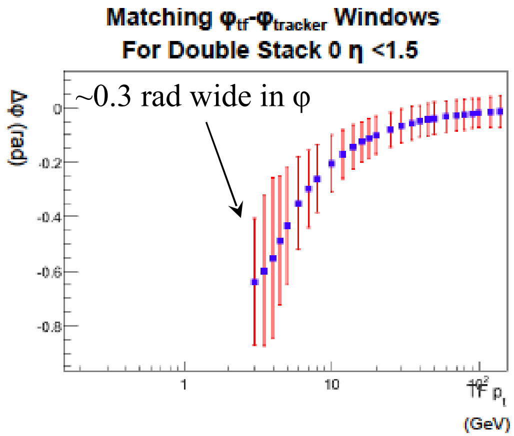
Examples of **For Double Stack 0 :**



Matching Z_{tf}-$Z_{tracker}$ Windows
For Double Stack 0

~40 cm wide in Z

Widths ~ 6 cm



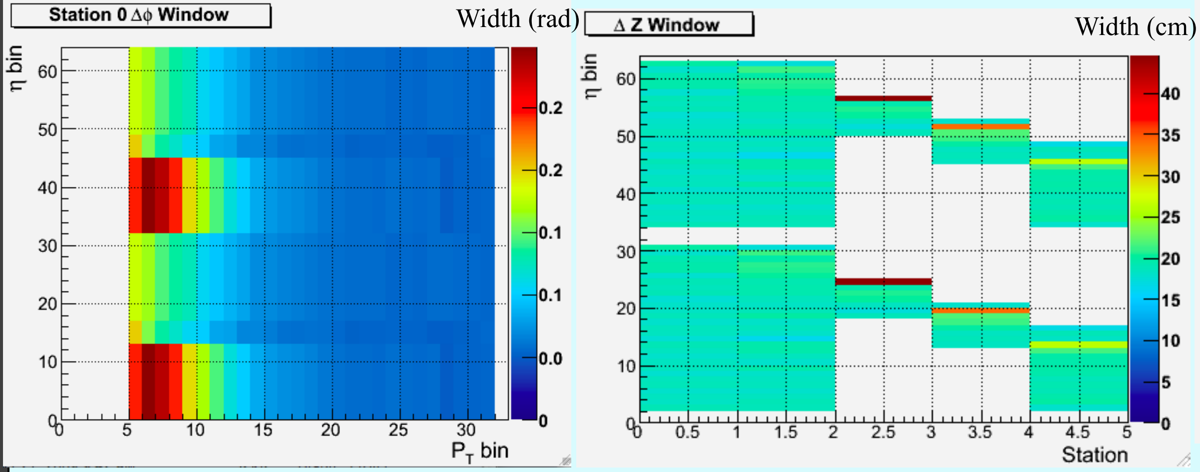
Matching φ_{tf}-$\varphi_{tracker}$ Windows
For Double Stack 0 η <1.5

~0.3 rad wide in φ

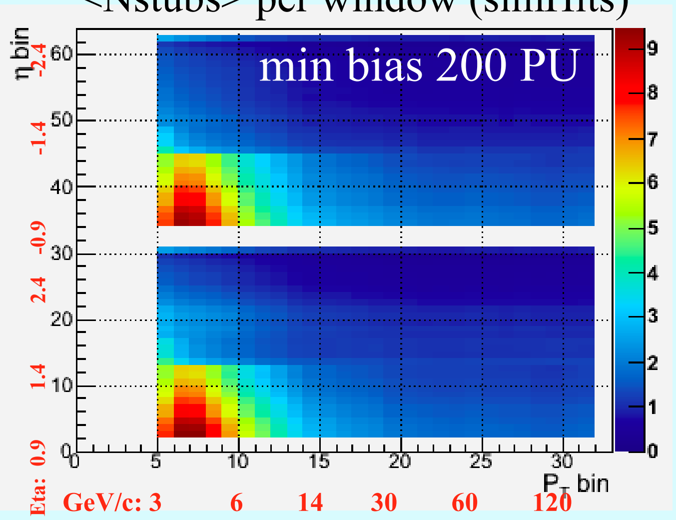
Widths =O(~0.1) - O(~0.01) rad
η dependence low p_T due to inhom. B-field
Can be tightened if necessary6

Matching windows are defined for all possible CSCTF-P_T (5 bits) and CSCTF-η (5 bits per endcap) values. Average match-window-occupancy plots shown below are a function of these CSCTF bins and were made with min bias events (200 PU).

Example Match window sizes versus CSCTF bins



<Nstubs> per window (simHits)

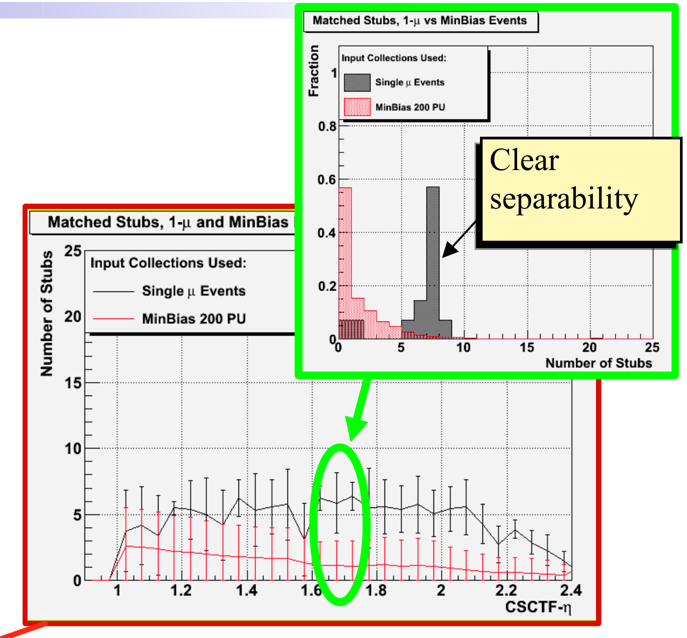


Matching Windows: Signal versus Background

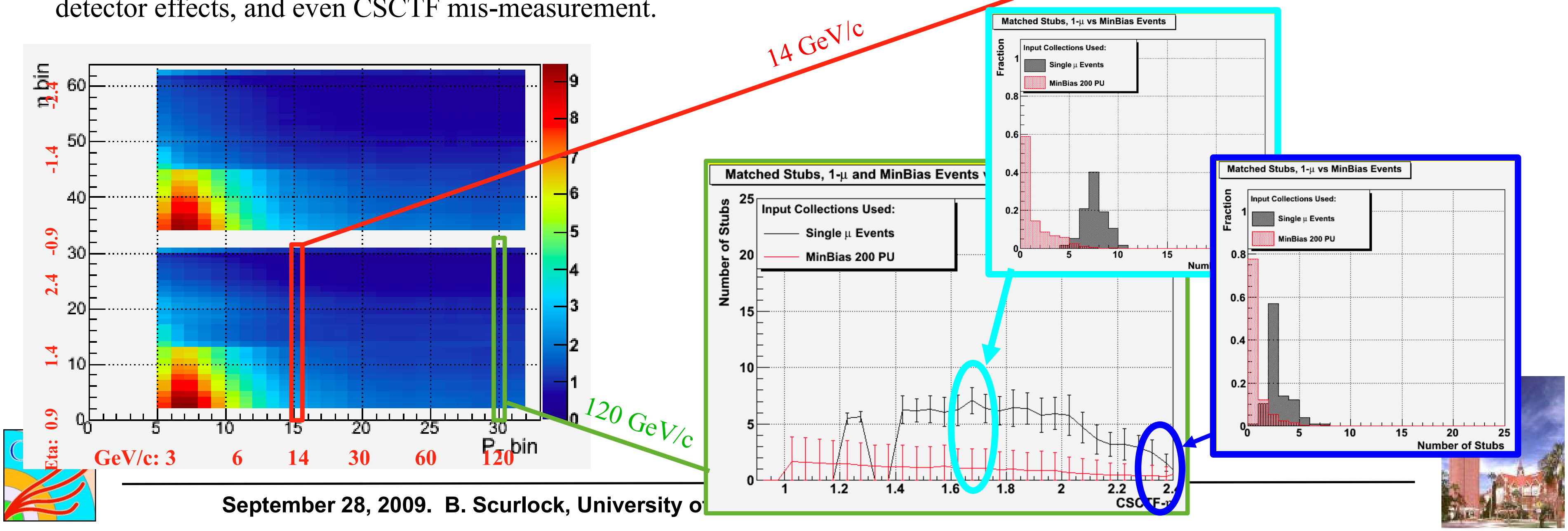
Examine stub content within CSC-Tracker matching windows looking at “signal” stubs (from **1-mu events**) versus “background” stubs (from **min bias 200 PU events**). Background stubs are counted within randomly chosen windows. Signal stubs are counted within window directly seeded by CSC Track-Finder track.

Green and red boxes show resulting <Nstubs> vs CSCTF-η graph for **signal** versus **background** for CSCTF-PT ~ 14 and 120 GeV/c. Zoom into particular CSCTF-η bin, and one observes the signal versus background distributions, which show clearly separation between signal and background for the selected matching windows.

By simply counting stubs within matching windows, we expect this can provide a power discriminator against fake rate from accidentals, noise, detector effects, and even CSCTF mis-measurement.

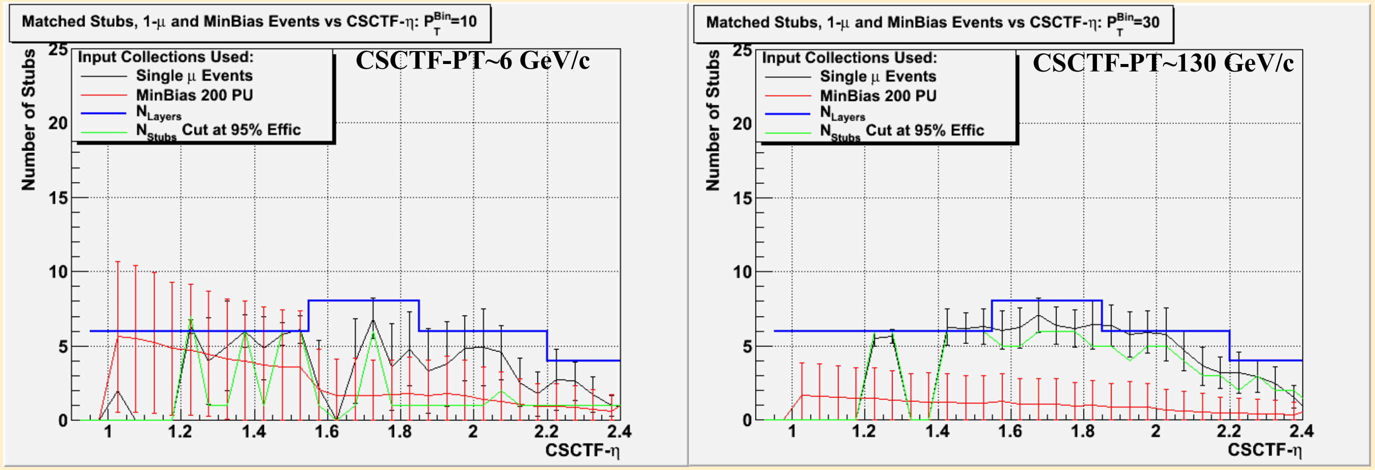


Clear separability



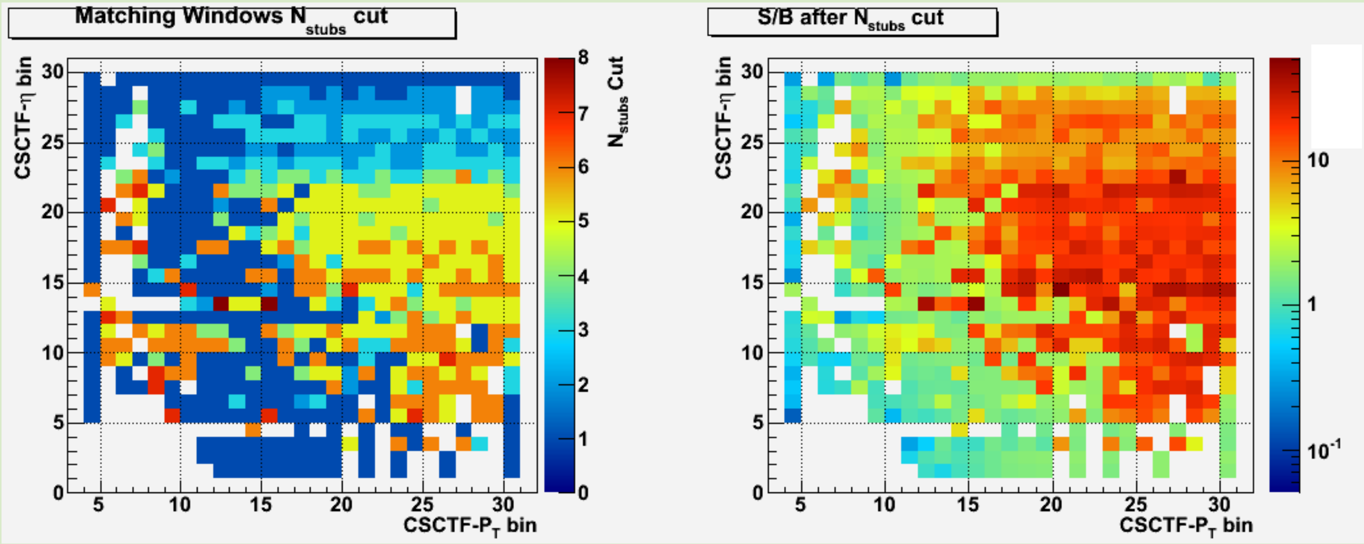
Matching Windows: Separating Signal from Background

Signal and Background vs CSCTF-η

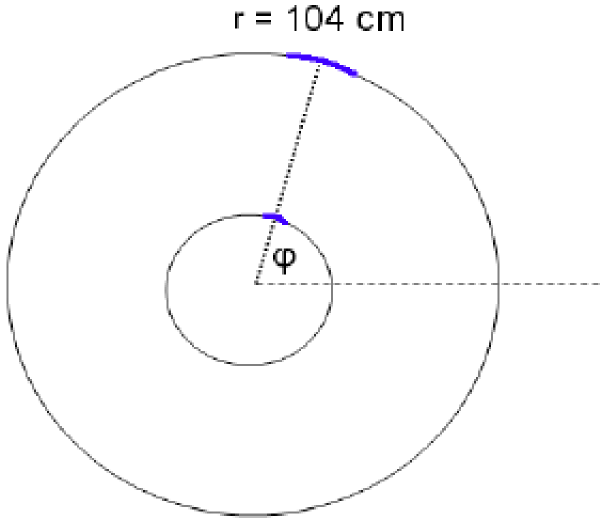


Once matching windows are re-tuned, expect that counting can provide a powerful handle for rate reduction from noise and CSCTF mis-measurement.

Example exercise: tune matching window bin-by-bin N_{stubs} threshold to accept 95% of signal stubs. Cuts and S/B versus bin seen on right ➙



P_T Estimate 1: Using Δφ



Circle Fit Approximation:

$$\phi = \phi_0 + \arcsin(\zeta R / p_T)$$

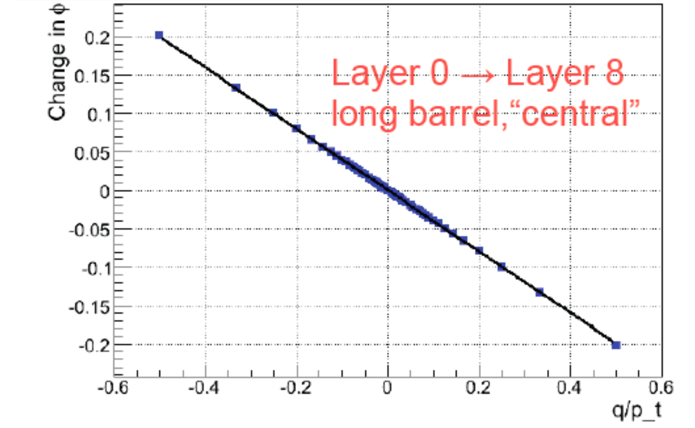
linear approximation:

$$\Delta\phi \sim 1/p_T$$

$$\Delta\phi \sim \Delta R$$

- sensors report local coordinate → global ϕ
- measure ϕ in 100 μm units of arc length at 104 cm
- $\Delta\phi_{09} = \Delta\phi_{ij} \cdot \Delta R_{09} / \Delta R_{ij}$
- $\Delta\varphi_{09} \rightarrow 1/p_T \rightarrow p_T$

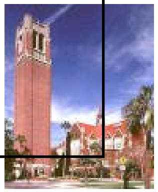
Approach demonstrated to achieve 2% P_T resolution



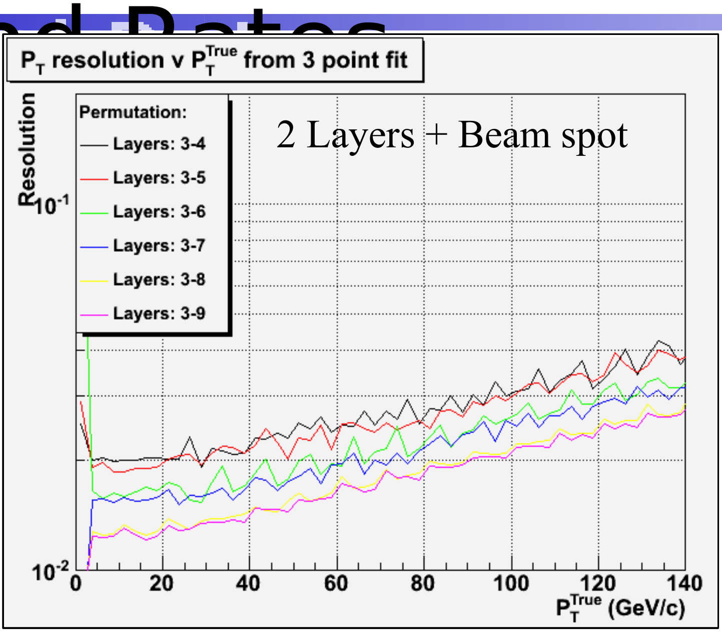
Layer 0 → Layer 8
long barrel,"central"

P_T with Beam spot drift

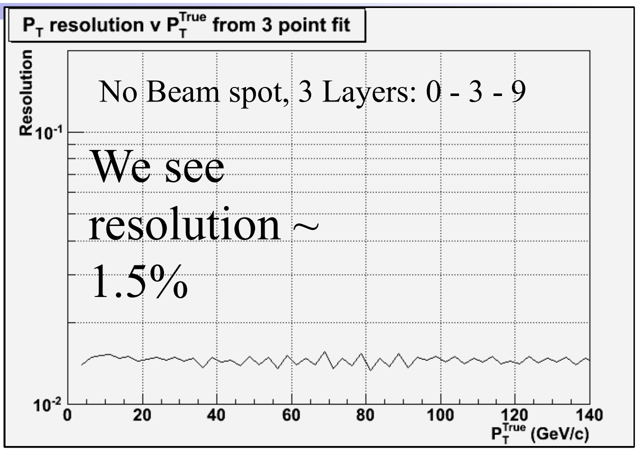
- Current algorithm takes filtered stub candidates and assigns P_T by finding effective $\Delta\varphi_{09}$ between tracker Layers
  - Uses linear fit between $1/P_T$ and $\Delta\varphi_{09}$, with (0,0,0) beamspot
  - This algorithm can be re-tuned to accommodate off-center (not investigated yet)
- Can we use the CSCTT model framework to accommodate beam spot drift?
  - Take filtered stub candidates and use a 3-point circle fit to find P_T
    - Algorithm 1: Assume a known beam spot and use stubs available from two tracker Layers
    - Algorithm 2: Assume unknown beam spot and use stubs available from three tracker Layers
      - Can then use DCA to provide beam spot location
    - Both algorithms fit two lines: L1 = Point_i to Point_{i+1} and L2 = Point_{i+1} to Point_{i+2} (points increasing in radius). Solve for the intersection of the two orthogonal lines which bisect L1 and L2
    - Working with engineer to understand how we can implement algorithm in HW



P_T Estimate 2: Circle-Fit Resolutions and Rates



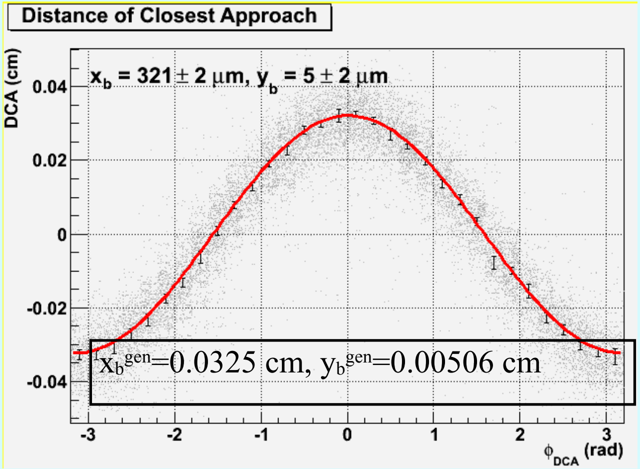
2 Layers + Beam spot



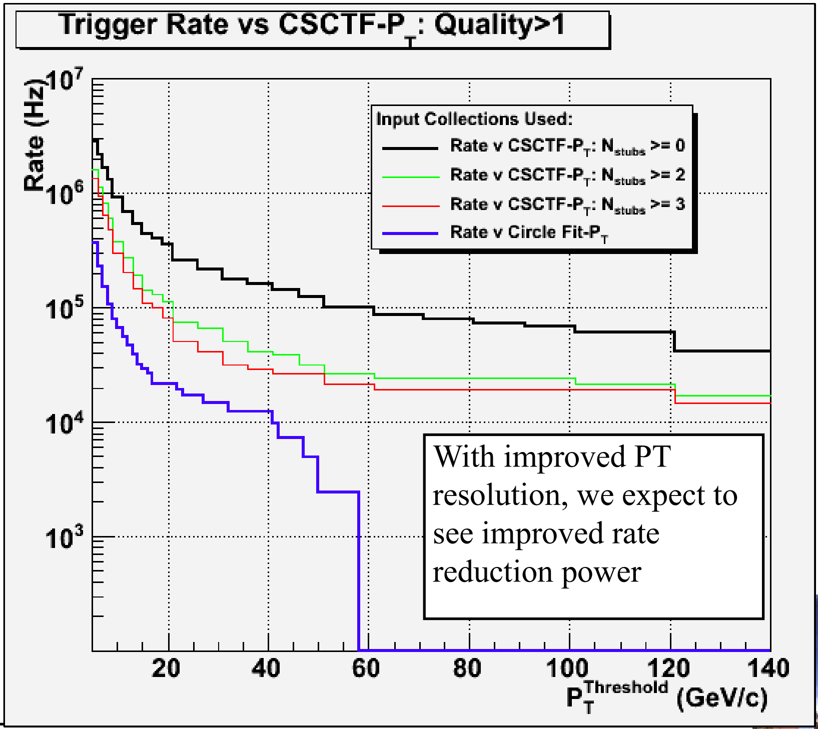
No Beam spot, 3 Layers: 0 - 3 - 9

We see resolution ~ 1.5%

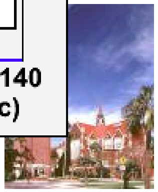
Use DCA to find beam spot



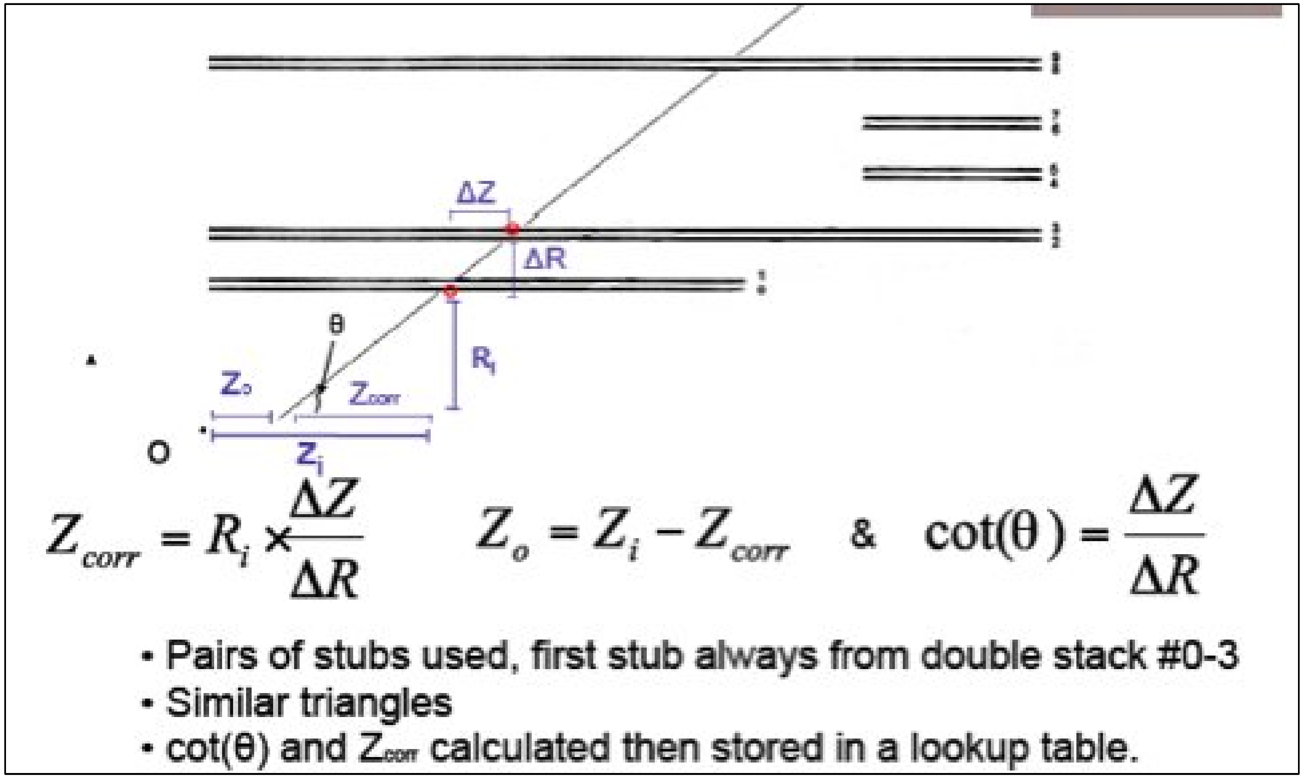
x_b^{gen}=0.0325 cm, y_b^{gen}=0.00506 cm



With improved PT resolution, we expect to see improved rate reduction power



cot(θ) & Z_0



$$Z_{corr} = R_i \times \frac{\Delta Z}{\Delta R} \qquad Z_o = Z_i - Z_{corr} \quad \& \quad \cot(\theta) = \frac{\Delta Z}{\Delta R}$$

- Pairs of stubs used, first stub always from double stack #0-3
- Similar triangles
- cot(θ) and Z_{corr} calculated then stored in a lookup table.

(1) Get cotθ from

$\Delta z_i = z_{stub1} - z_{stub2}$ and

$\Delta R_i = R_{stub1} - R_{stub2}$

(2) Get Z_{corr} from R_i (known) and cotθ

(3) Get Z_0 from Z_{stub1} and Z_{corr}

CSTT model has been demonstrated to achieve Z_0 resolution 640 µm and cot(θ) resolution 0.002

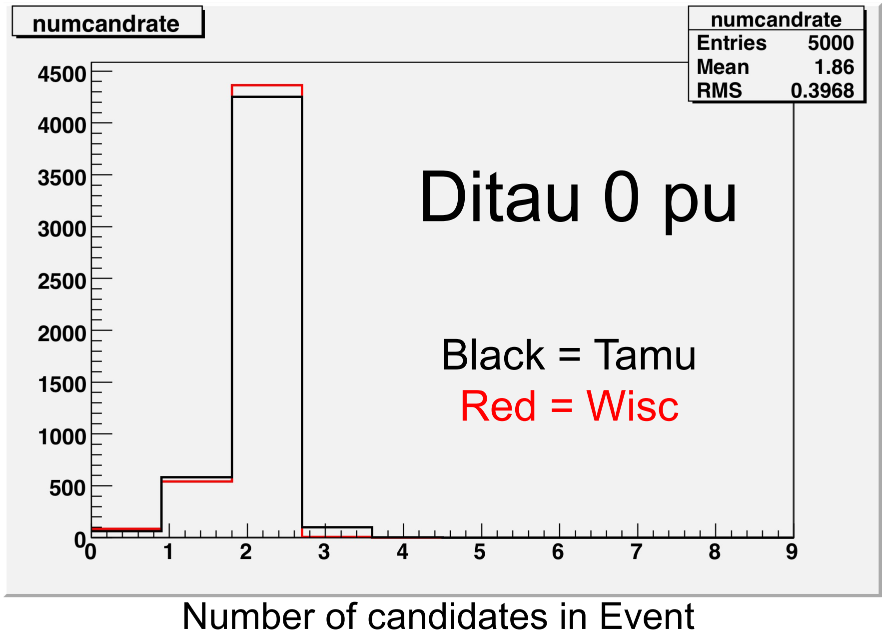
Tau Studies

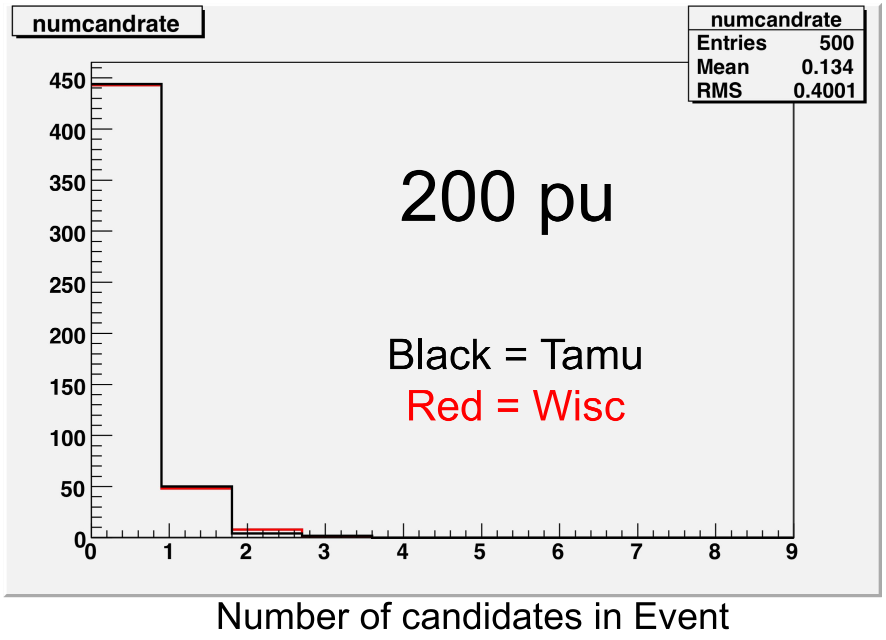
M. Weinberger, T. Kamon, A. Safonov, A. Marotta
(Texas A&M University)

Outline

- A lot of progress has been made to make our previous results more robust
  - Moved from Mctau to L1 trigger objects
  - Wisconsin and Tamu candidates are now virtually identical
- Repeated studies of single tau trigger rate and isolation studies.

Number of Candidates

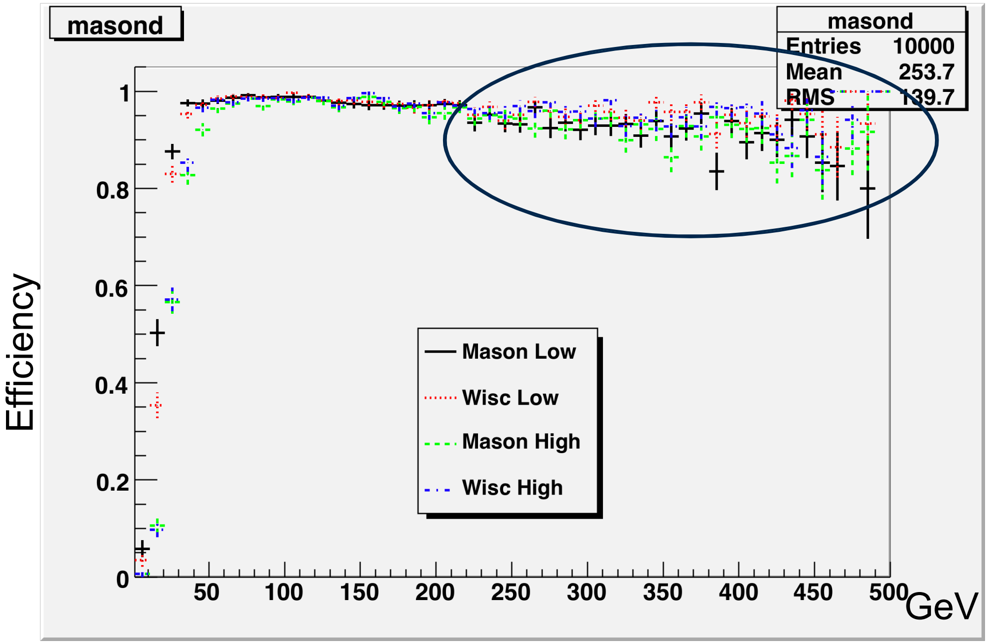


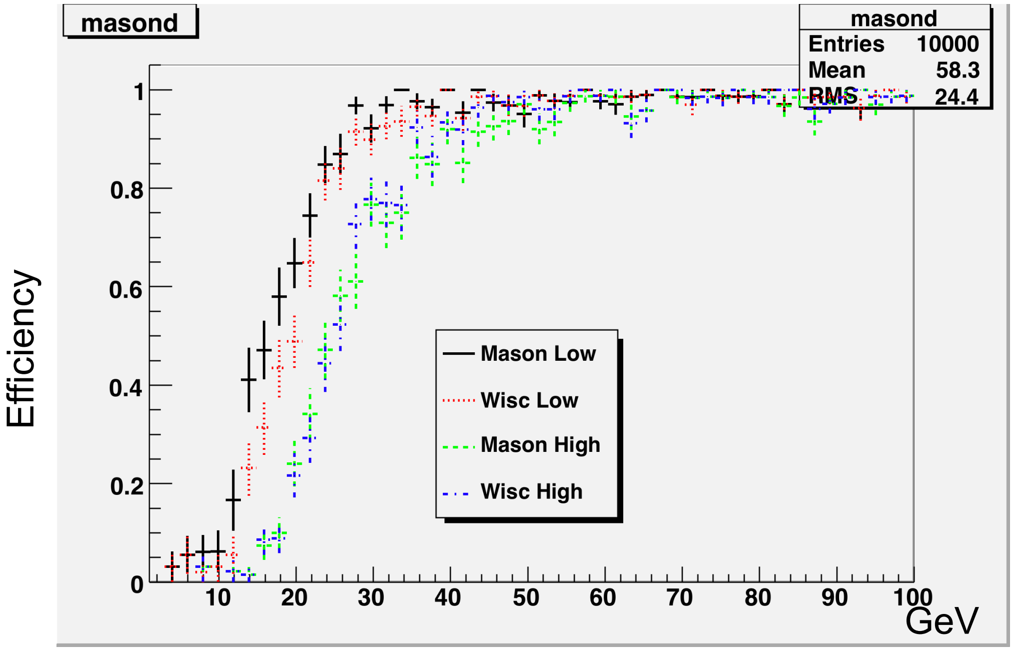


- With all cuts and the higher seed values, the number of candidates found for 0pu ditau and 200pu both match very well
  - Seed Tower Et = 10GeV [previous 5GeV]
  - 2x2 candidate Et = 15GeV [previous 10 GeV]

0pu Ditau Turn On Curves

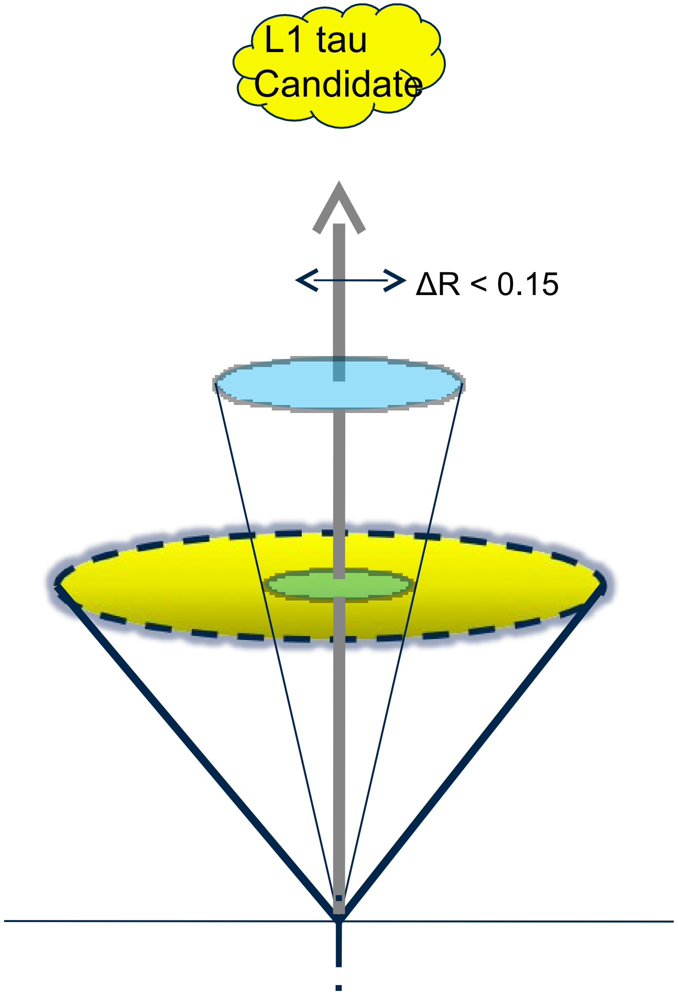
Low and high refer to the seed and 2x2 cuts on page 2.





- Plots show the turn of curve for finding efficiency of the MC candidate
- By 35 GeV bin, the samples are very close together, at lower Et, differences due to shift in energy between candidates, due to difference in calculations
- With the same cuts, Mason and Wisc are very similar for signal eff
- Drop in efficiency [circled] at high Et explored due to MC issue. The lost Mctaus deposit very little [<4GeV] in the calorimeter. Working with fastsim people to resolve this issue

Seeding Calo Clusters



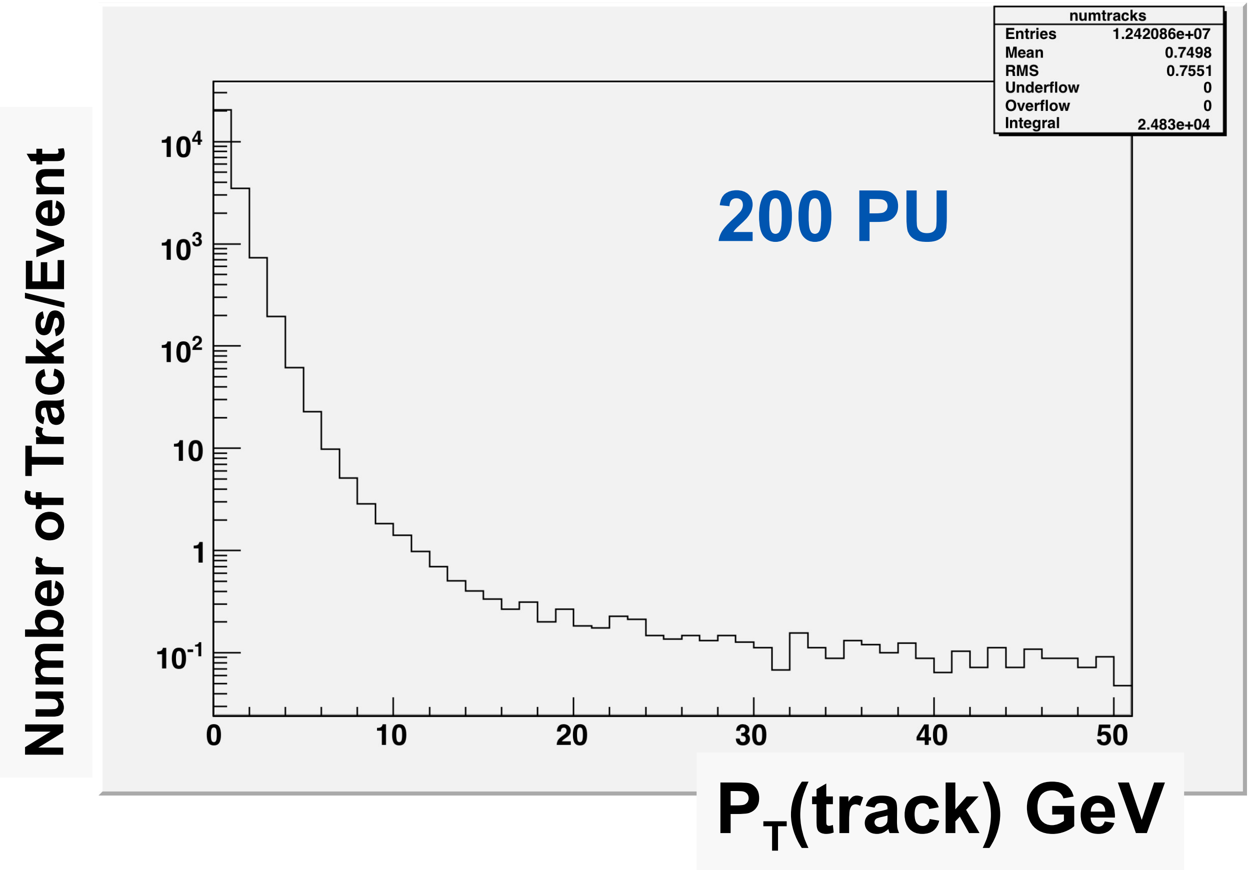
- Next step for rate reduction is to attempt seeding

- Look for highest pt track with pt>10GeV in a cone of ΔR < 0.15 around the calo cluster
  - Extrapolate the track to the Hcal to get eta phi for comparisons

- This should reduce the backgrounds from fakes

- The question is how much will this help

Estimate Number of Tracks From Pileup



Each of these tracks could be a fake-tau track to produce a seed if matches a L1 tau

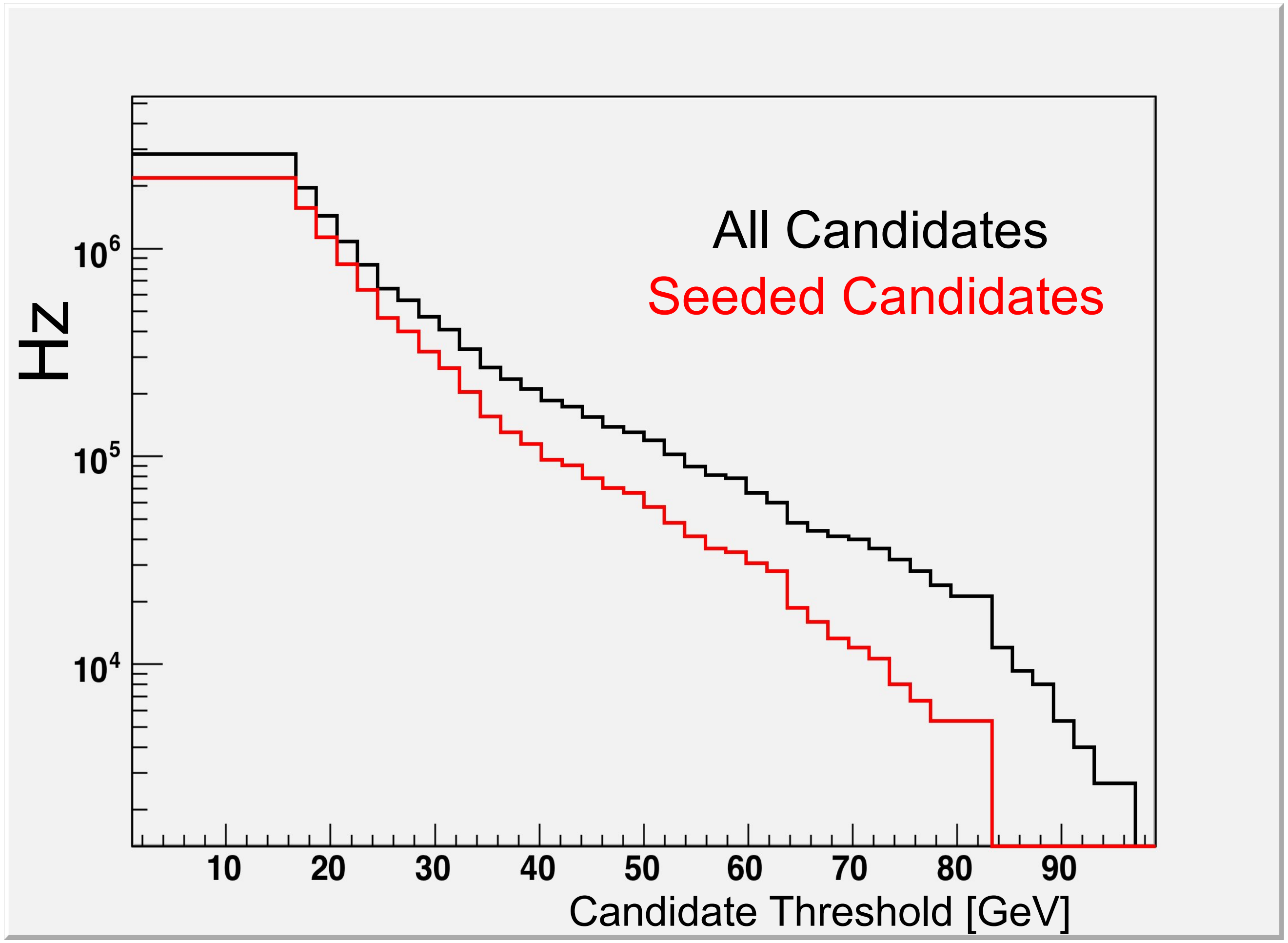
P_T > 1 GeV → 4500 tracks/event

P_T > 2 GeV → 1040 tracks/event

P_T > 5 GeV → 50 tracks/event

P_T > 10GeV → 4.5 tracks/event

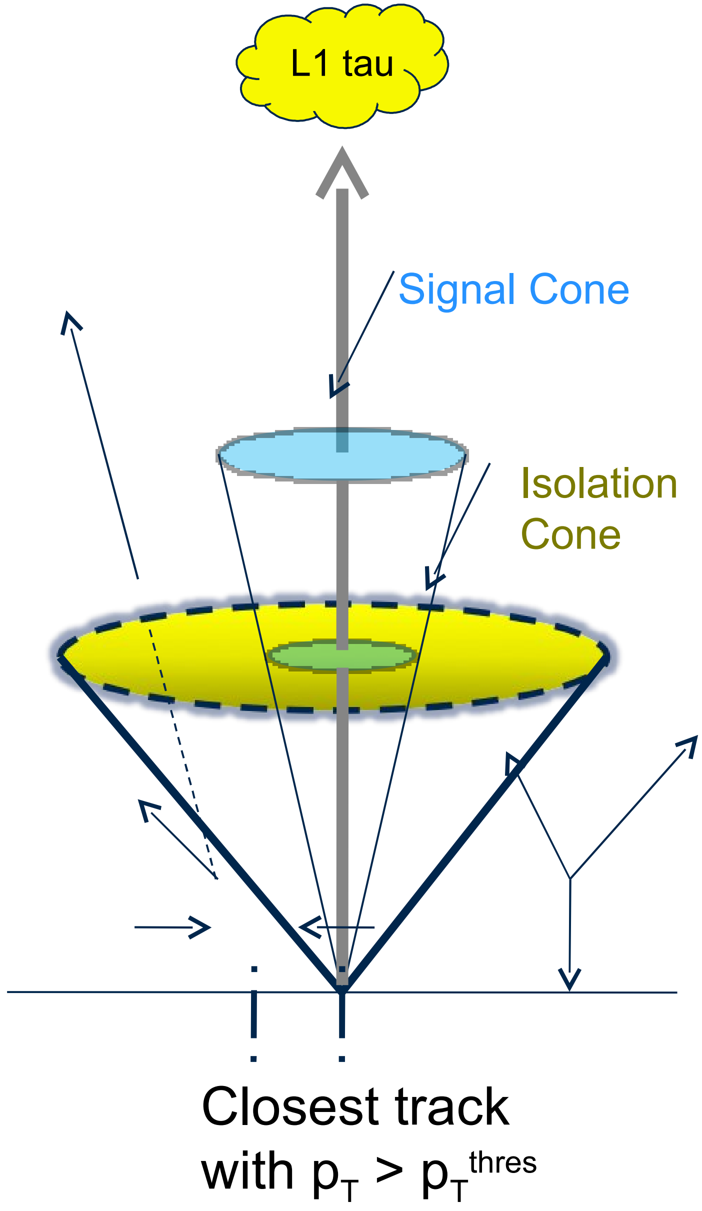
Seed Track rate reduction



- Look at fastsim sample with 200 pu with no signal taus
- Use the Wisc L1 tau candidate
- Measure two things
  - No seed requirement: just look at L1 tau
    - Save highest candidate > 20 GeV.
  - With seeding: find highest pt seed track above 10 GeV with in $\Delta R < 0.15$ (between L1tau and seed track)

Michael Weinberger

Probability of Accepting Tau



Closest track with $p_T > p_T^{thres}$

•Importance of z0 resolution

- •Need to be able to associate the tracks with appropriate vertex
- •Z0 resolution determines distance we can tell the vertexes apart
- •Otherwise will get tracks in isolation cone from pileup events
- •Gives a geometry independent way to see what z0 resolution is needed

Method

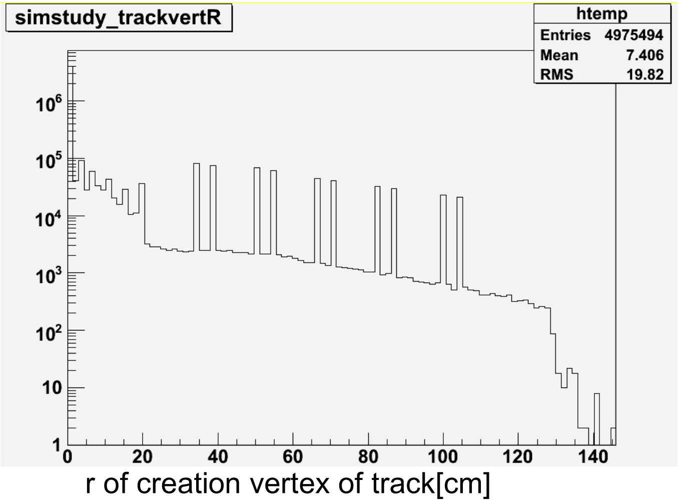
- •Use L1 tau to get tau direction and z0
- •Loop all tracks and propagate them to x=y=0 to get z0
- •Keep the track within the isolation annulus that is the closest to the tau vertex in z
- •Use this to test isolation reductions with varying cuts on Delta z0

Samples

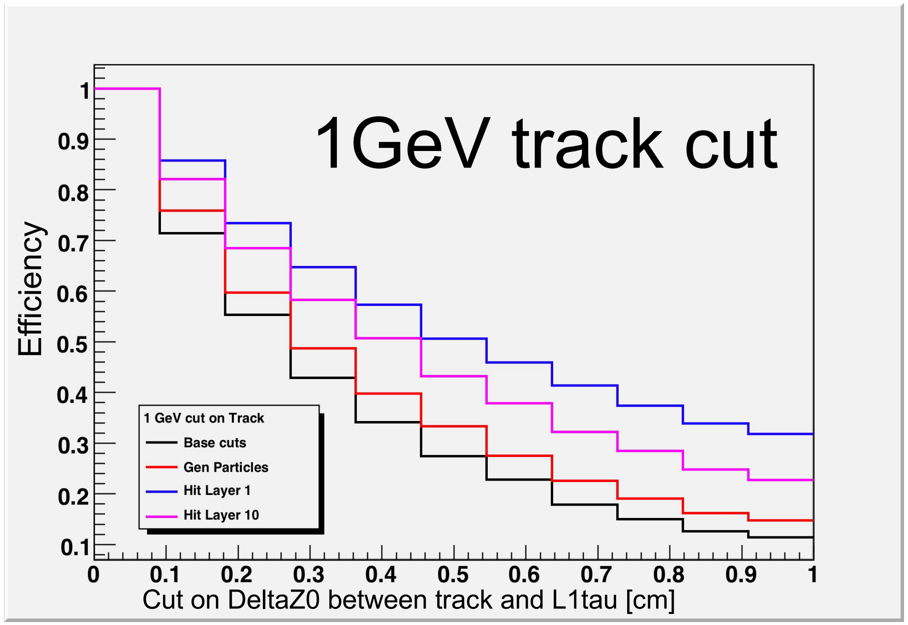
5 sets of cuts used for this study

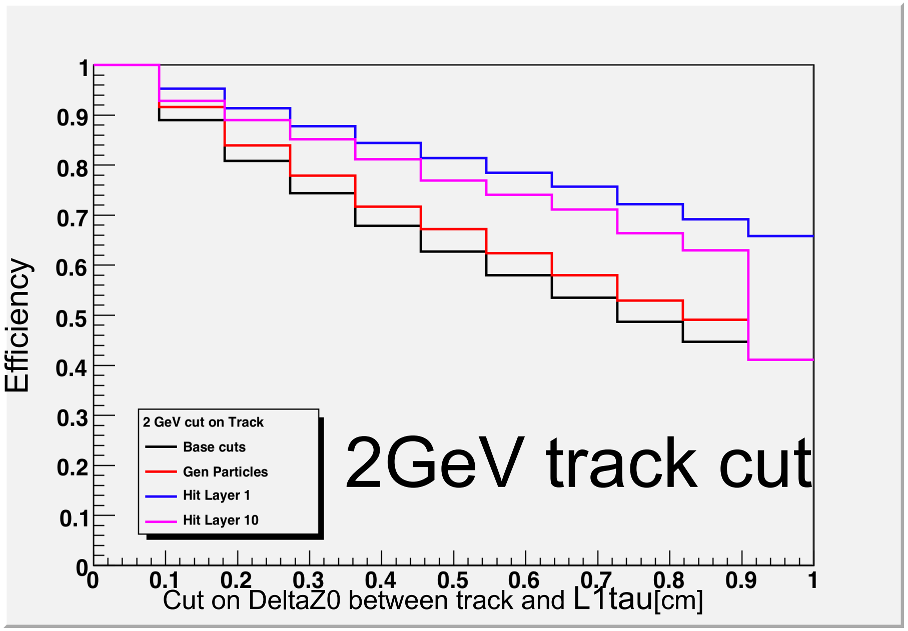
  - Fastsim Pythia tau gun plus 200 pu

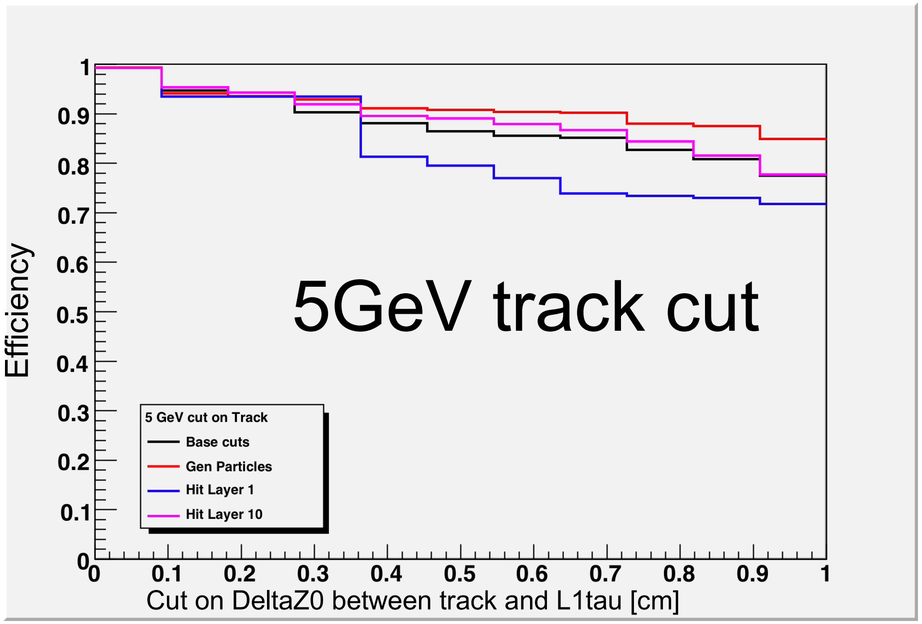
- Base
  - Cut on Cand Et>10, 0.15 < Δr <0.50, pt of track
- Vertex [Gen particles]
  - Require simtrack vertex near beamline to get only generator like particles
    - Plus Base cuts
- Layer1
  - Track must hit the first tracker layer
    - Match simhits to layers and isolation track
    - Plus Base cuts
- Layer10
  - Track must hit the last layer
    - Plus base cuts
- All above samples are done for cuts on track of 1, 2, and 5 GeV
  - Will show plots of all samples for each track cut at end



Effect of Z0 res on tau efficiency







- These plots show how the z0 res affects the tau finding efficiency.
- Want low track cut, which requires a better resolution
- More restrictive cuts on tracks leads to better efficiency

Conclusions

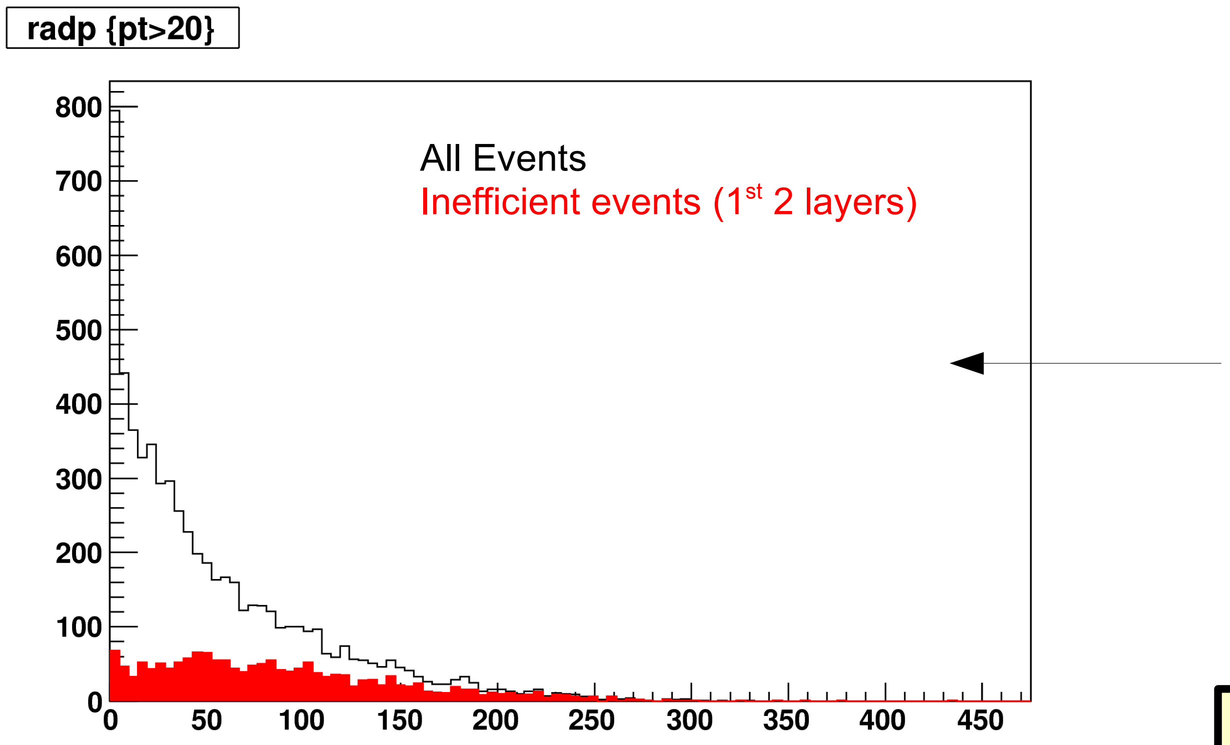
- Wisc vs Tamu tau candidates comparison is finished, and code now uses Wisc for testing
- The seeding studies and isolation studies have been redone with the new candidates
- They show the single tau trigger rate needs a high limit to be usable
  - Need add another trigger object to use low threshold on tau
- Z0 Resolution plots indicate how good the tracker has to be to allow for usable isolation for taus
- Next step is to switch to stubs/tracklets which will tell us where on the resolution plot we are, and how much improvement we can expect by getting better resolution
  - Either by adding more stubs from different layers, or changing the pixel size

Conclusion

- Simulations of Track Trigger Primitives in FastSim continue to mature
  - We understand basic results much better than 6 months ago
- Algorithm studies, linking track-trigger information with calorimeter/muon chambers to reconstruct electrons, muons and taus, are all making excellent progress
- A note documenting all this work is underway

Backup Slides

Electron Studies @ Cornell

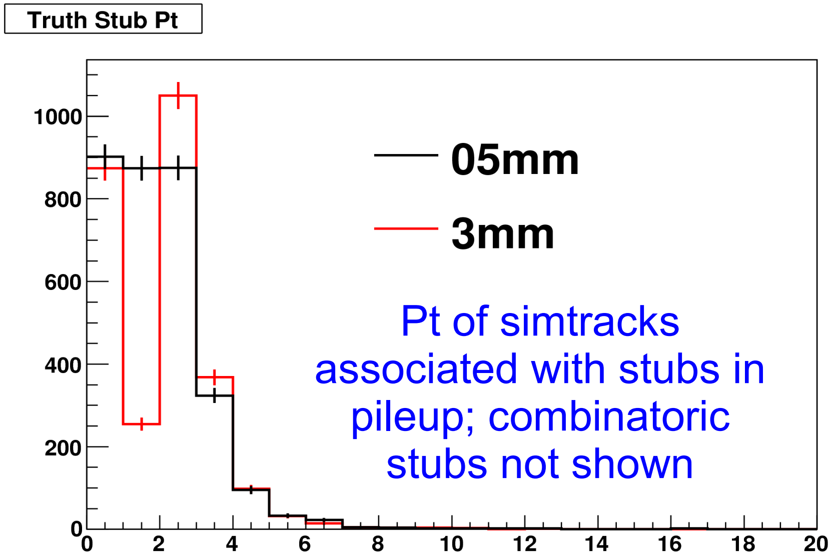


Ratio of red to white shows fraction of inefficienct events as a function of the amount of momentum radiated by the electron

Other sources of inefficiency: z matching cuts and gaps between modules in z

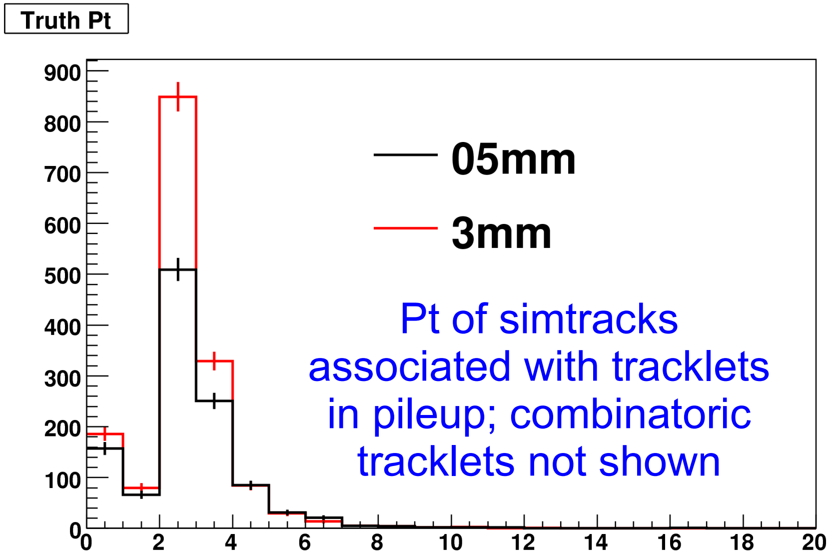
A large portion of electron inefficiencies is due to bremstrallung → This causes Pt measured in the tracker and calorimeter to differ

Stub Rates



Pt of simtracks associated with stubs in pileup; combinatoric stubs not shown

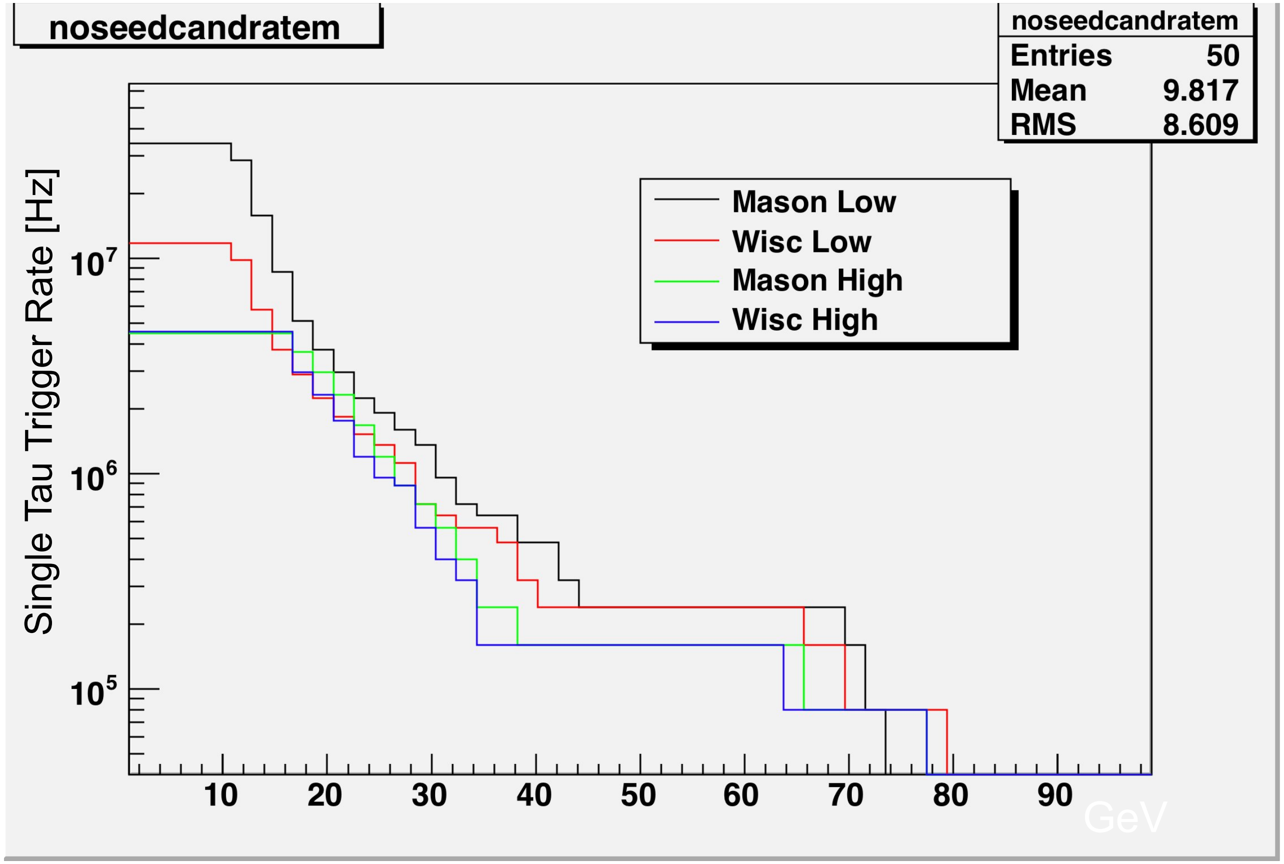
Tracklets have much better Pt resolution than stubs → tracklets below threshold are strongly suppressed



Pt of simtracks associated with tracklets in pileup; combinatoric tracklets not shown

Tracklet rate variations across stack separation are dependent on stub efficiencies just above threshold; Tracklet rates increase more significantly with stack separation than stubs do

200pu trigger rates

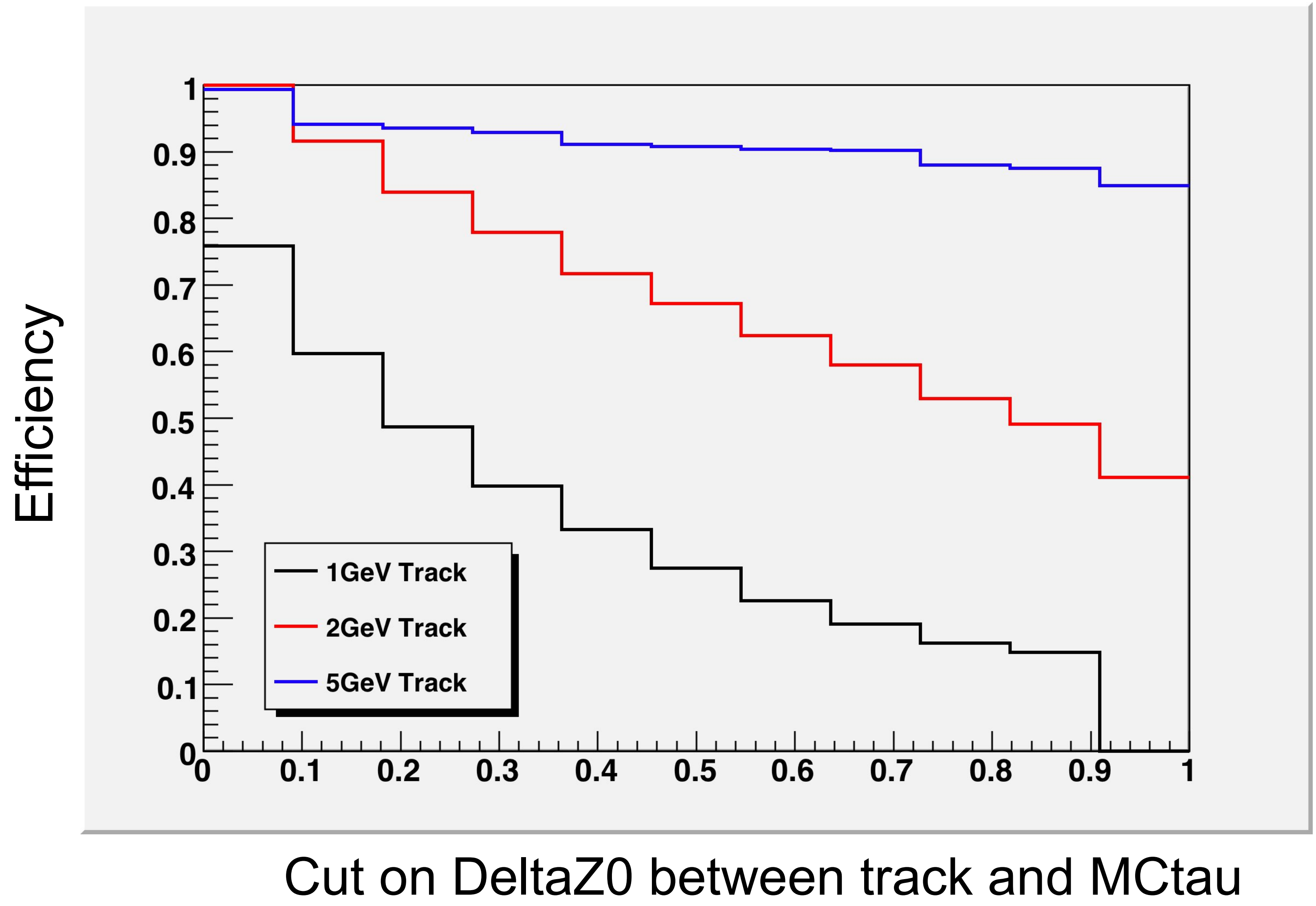


- TAMU for regular seed levels [5GeV seed and 10GeV 2x2]
- Wisc for regular seed levels[5GeV seed and 10GeV 2x2]
- TAMU for high levels [10GeV seed and 15GeV 2x2]
- Wisc for high seeds[10GeV seed and 15GeV 2x2]

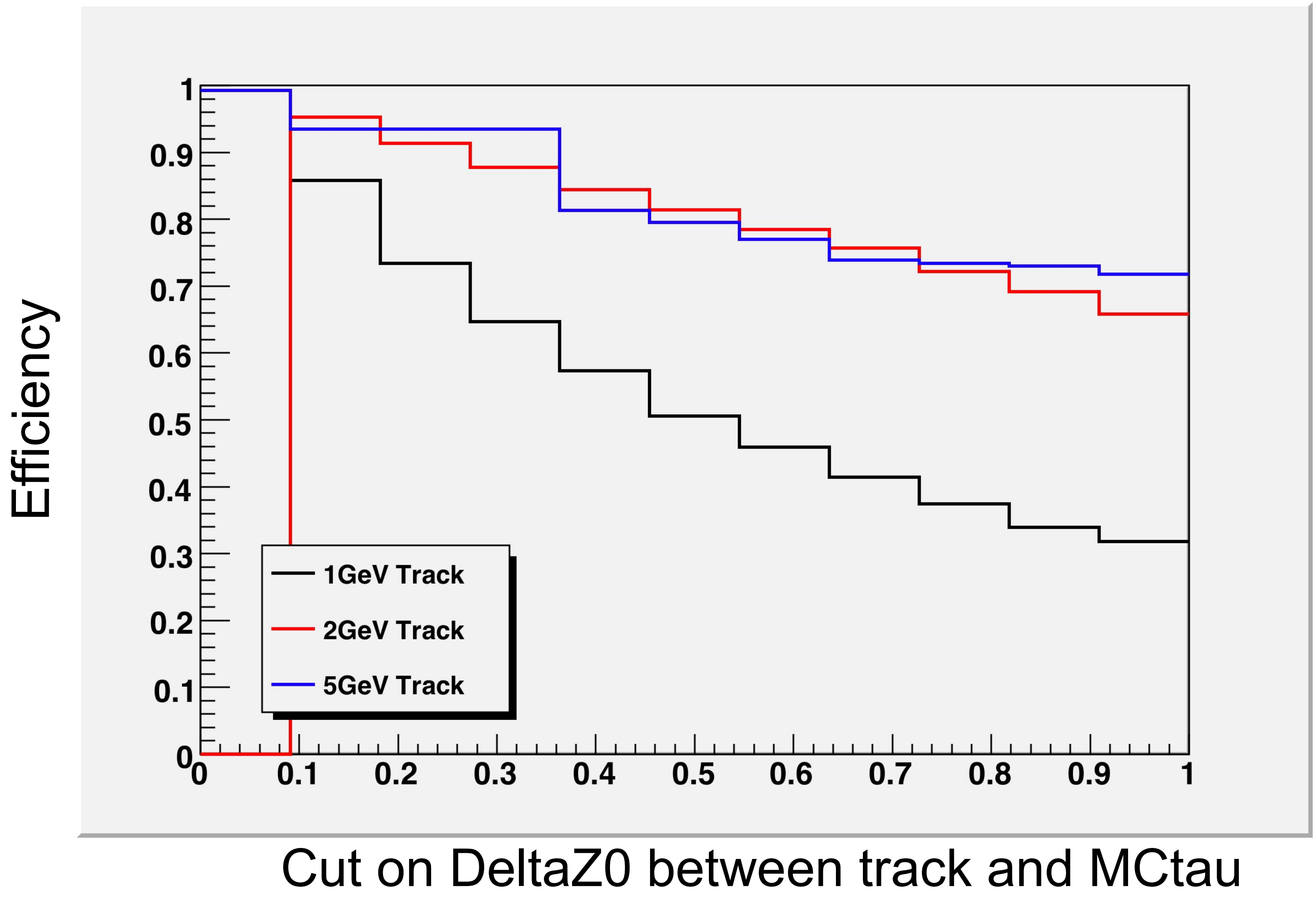
- Above the cut off the the rates are similar
- Below they start to diverge for the low cut due to excess candidates in Tamu code
- Applying the cut prevents this and the rates stay the same
- Difference in rate due to shift in calculated candidate Et

Michael Weinberger

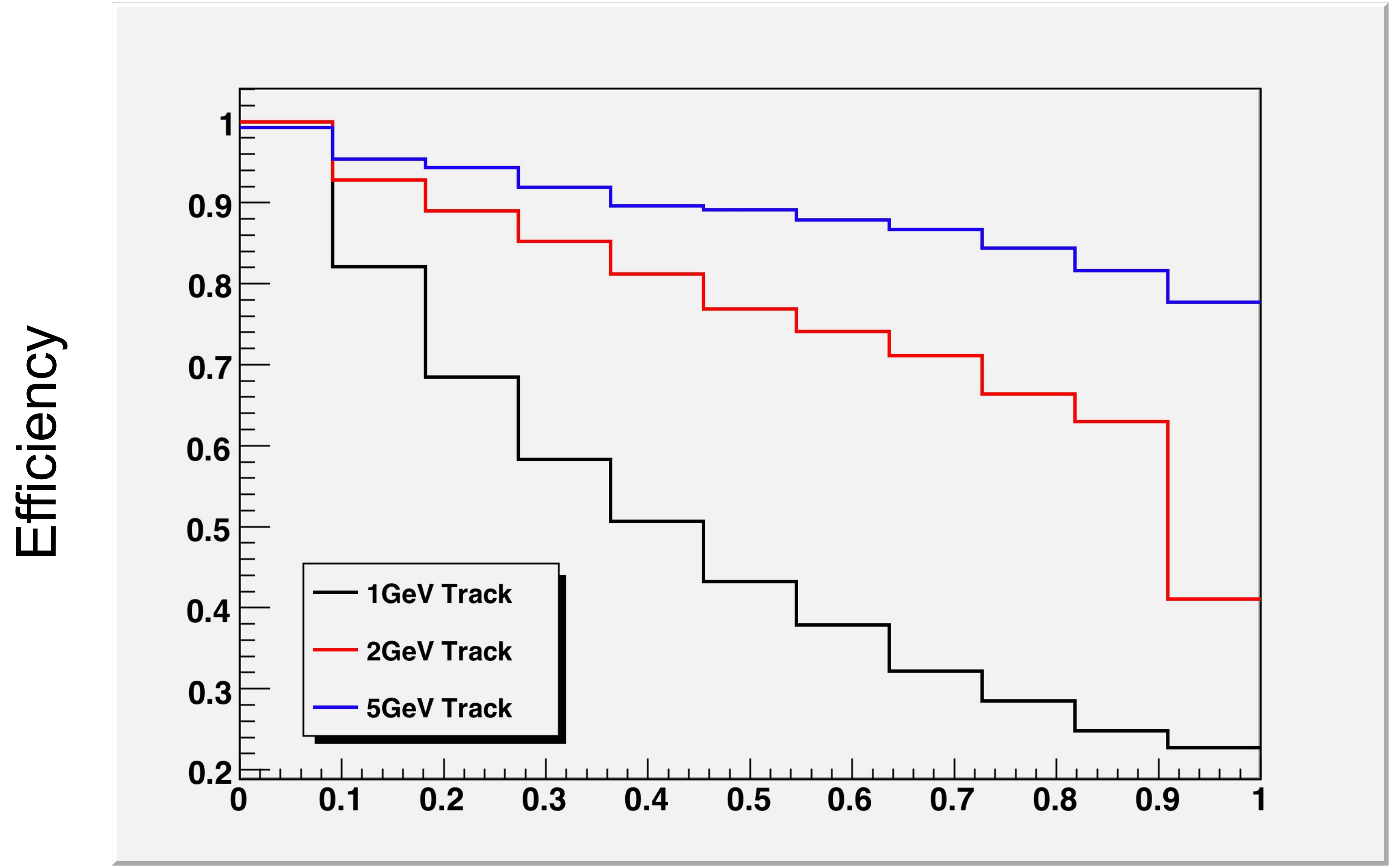
Small Radius of Production [gen tracks]



Base cuts Tau gun +200 pu Hitting Layer 1

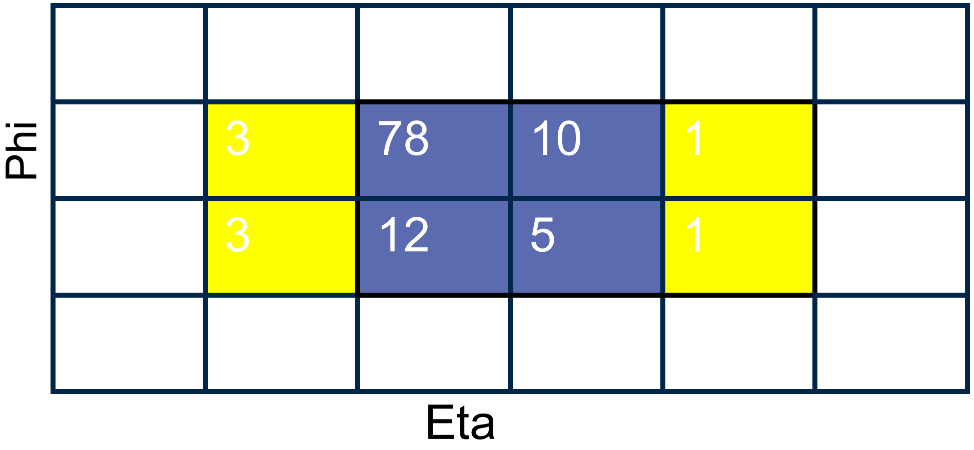


Base cuts Tau gun + 200 pu Fitting Layer 10



Cut on DeltaZ0 between track and MCtau

E6-E4 cut



	3	78	10	1	
	3	12	5	1	

- Blue is 2x2 candidate
- Add on two adjacent towers in plus or minus eta [yellow blocks] to make 2x3 block, and take highest new Et
    - In this case either add 2 or 6 in Et to cand
    - Keep minus eta cand – add 6
- Make difference of new sum to original candidate value
- Cut on difference
    - Old cut was set to 4
        - This candidate would fail
    - New setting is 9
        - This candidate would pass