Finding, Fitting.. What’s New?

- A new paradigm: Go 3D!
- New Tools: use MC to propagate trajectories
- A recast of Traditional GLAST Finding: Combo
- A recast of the Kalman Filter
- Sea Trials
- Setting the e^+e^- Energies
- Vertexing: How to put the tracks together
- Present status w.r.t PSF & A_{eff}

3D vs 2 x 2D

Detector Structure of GLAST suggest that we track separately in the X & Y Projections (X hits in a particular layer/tower aren't correlated with the Y hits)

Tracks crossing between Towers, Track lengths, and multiple scattering reduce the ambiguities

First GLAST Fitting/PR (1993) was 3D. X-Y ambiguities were a matter of combinatorics at Tower level. Towers were smaller and the X-Y ambiguity error was a few percent.

In 1995 PR/Fitting re-written as 2 quasi-independent projections.
- PSF never recovered!

Fitting really needs 3D to assess the material audit along trajectories

Became convinced that 3D really needed another try.

Strips/Clusters to Space Points

A basic to GLAST is the 3-in-a-row trigger: 3 consecutive X-Y planes firing within a microsecond.

This yields possible space points.

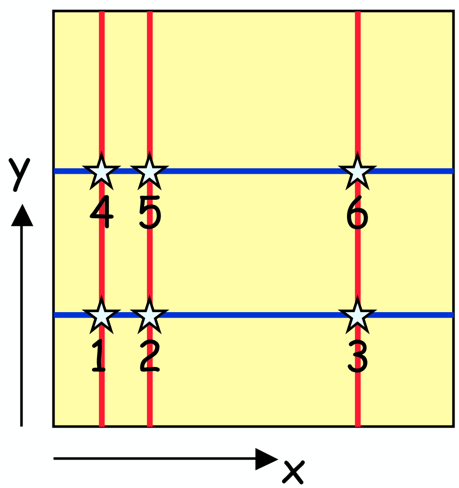
Step one: build an object which can cycle over the allowed X-Y pairing in a given GLAST measuring layer

a) Ordered just as they come X's then Y's

b) Ordered with reference to closeness to a given space point

Case a)

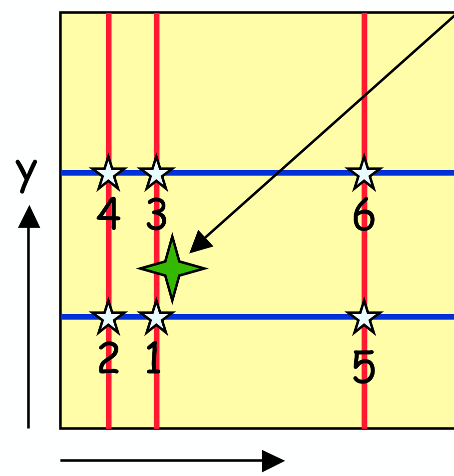
(nextHit)



Case b)

(x,y) Point about which to search

(nearestHit)



From Space Points to Track Hypothesises

Three Approaches Under investigation:

1) "Combo" - Cycle over space points - build an ordered list of tracks

2) "Link & Tree" - Join space points allowing branches - forming a Tree like structure

3) "Neural Net" - Link close by space points (forming "neurons"). Link neurons by rules weighting linkages.

First method is "track by track" -

Pro's: Simple to understand (although details add complications)

Con's: Find bogus tracks early on throws off the remaining by missing assigning hits

Can be quite time consuming depending of depth of search

Methods 2) & 3) are "whole event"

Pro's: Optimized finding across entire event

Con's: Neural Nets can be quite time consuming

Combo Pat Rec - Kalman Overview

Pattern Recognition

First Pass Best Track — Calorimeter Based or Blind Search

Hit Flagging — Allow up to 5 shared Clusters

Second Pass All the others — Blind Search

Set Energies — Global Energy Constrained Track Energy

Track Fitting

Final Fits — Kalman Track Fit

γ Creation — Track Averaging (Not Vertexing)

The “Combo” Pat. Rec. (Details)

Starting Layer: One furthest from the calorimeter

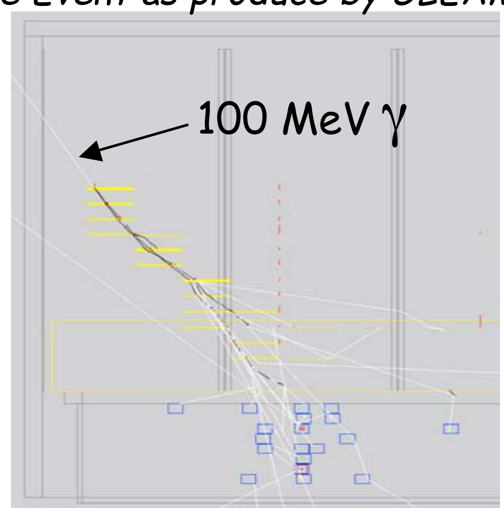
Two Strategies:

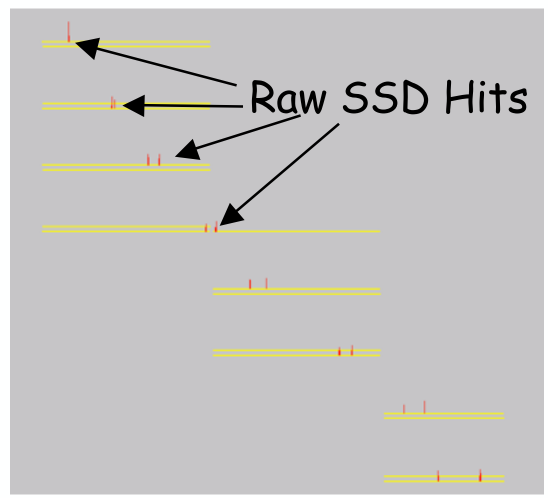
1) Calorimeter Energy present ⇒ use energy centroid (space point!)

2) Too little Cal. Energy ⇒ use only Track Hits

“Combo” Pattern Recognition - Processing an Example Event:

The Event as produce by GLEAM

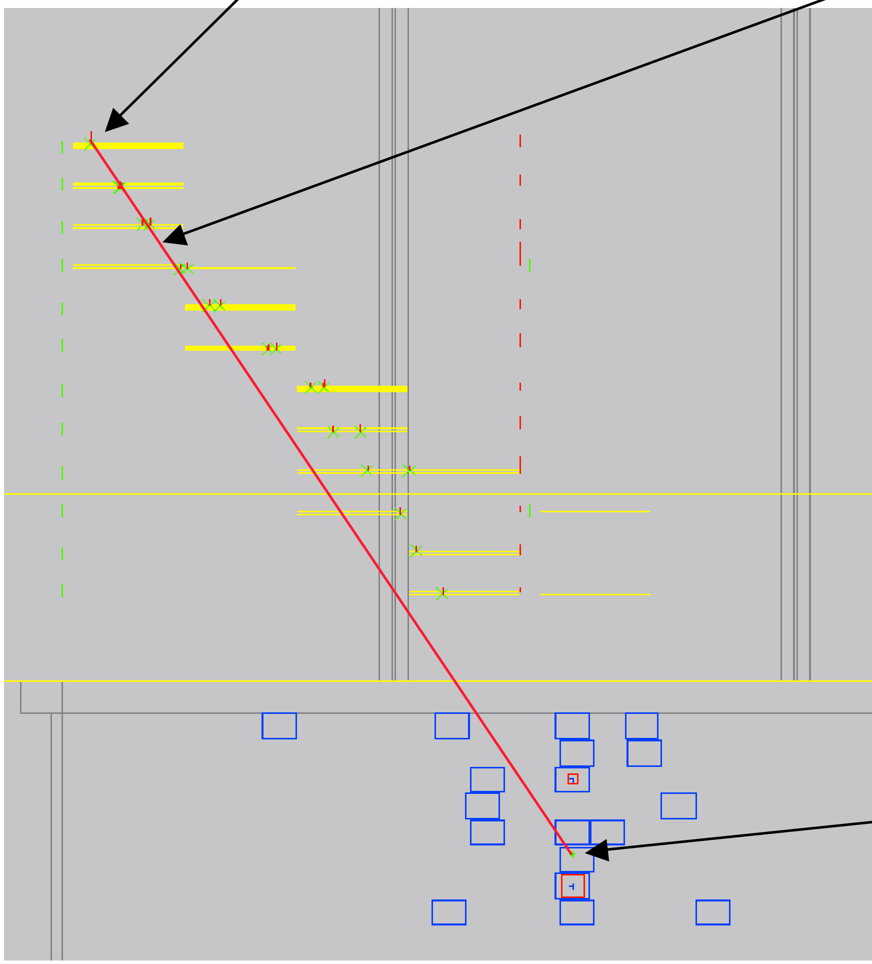


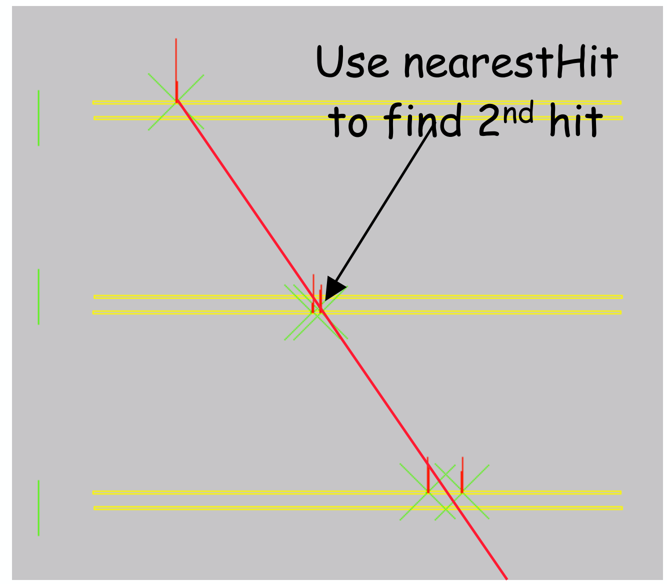


The "Combo" Pat. Rec. (Details)

Start with hits in outer most layer

First Guess: connect hit with Cal. Centroid



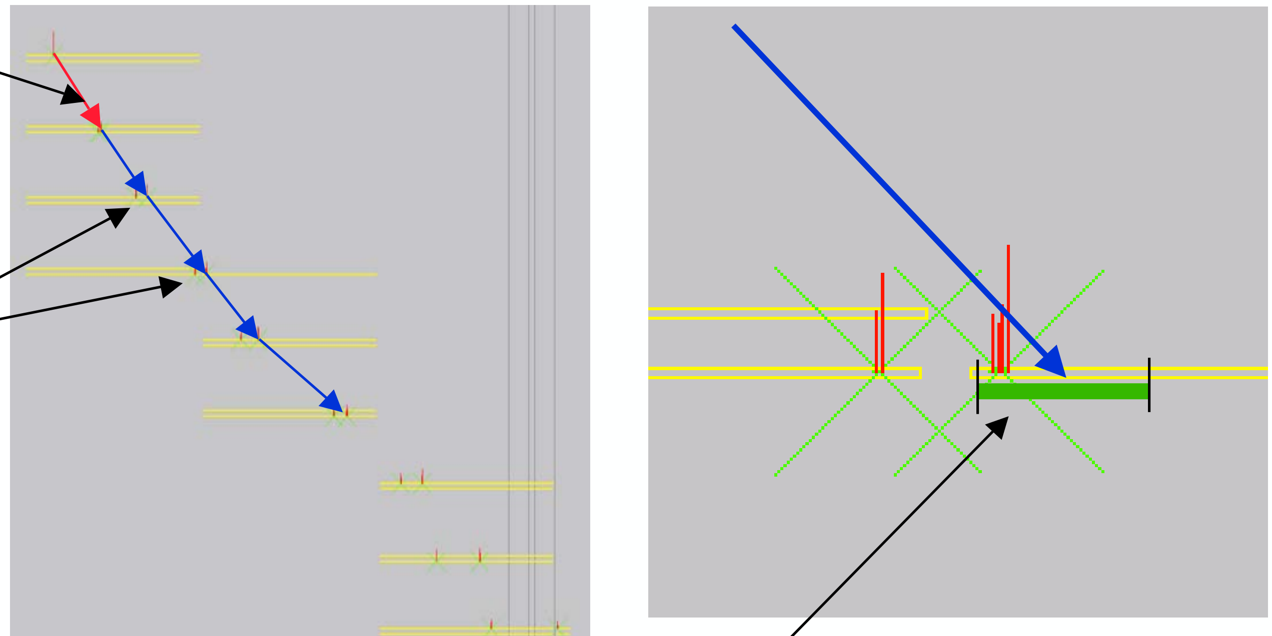


Sufficient Cal. Energy (42 MeV)
Use Cal. centroid

The “Combo” Pat. Rec. (Details)

Initial Track Guess:
Connect first 2 Hits!

Project and Add
Hits Along the
Track within
Search Region



The search region is set by propagating the track errors through the GLAST geometry.

The default region is 9σ (set very wide at this stage)

The "Combo" Pat. Rec. (Details)

The Blind Search proceeds similar to the Calorimeter based Search

- •1st Hit found found - tried in combinatoric order
- •2nd Hit selected in combinatoric order
- •First two hits used to project into next layer -
- •3rd Hit is searched for -
- •If 3rd hit found, track is built by "finding - following" as with Calorimeter search

In this way a list of tracks is formed.

Crucial to success, is ordering the list!

The “Combo” Pat. Rec. (Details)

Track Selection Parameter Optimization

Ordering Parameter

Q = Track-Quality - C_1* Start-Layer - C_2*First-Kink - C_3*Hit-Size - C_4*Leading-Hits

Track_Quality: “No. Hits” - χ^2 : track length (track tube length) - how poorly hits fit inside it

Start-Layer: Penalize tracks for starting late

First-Kink: Angle between first to track segments / Estimated MS angle

Hit-Size: Penalize tracks made up of oversized clusters (see Hit Sharing)

Leading-Hits: These are unpaired X or Y hits at the start of the track. This protects against noise being preferred.

Status: Current parameters set by observing studying single events.
Underway – program to optimize parameters against performance

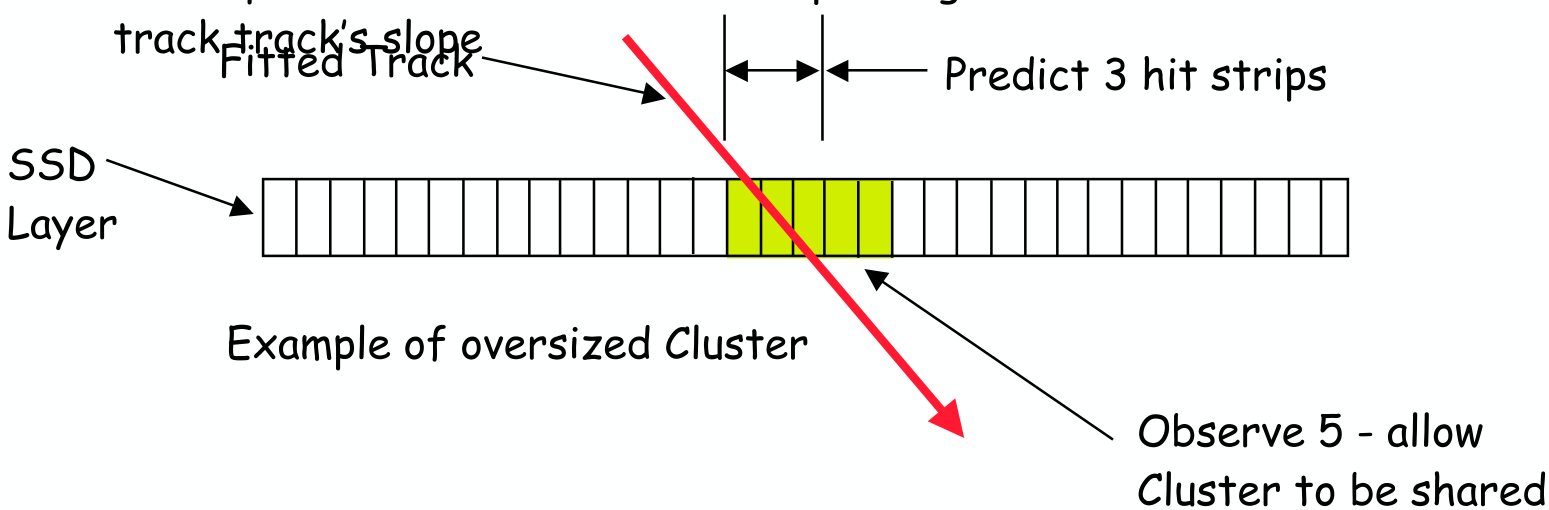
The "Combo" Pat. Rec. (Details)

Hit Flagging (Allowed Hit Sharing)

In order not to find the same track at most 5 clusters can be shared

The first X and Y cluster (nearest the conversion point) is always allowed to be shared

Subsequent Clusters are shared depending on the cluster width and the track track's slope



A Kalman Filter for GLAST

- What is a Kalman Filter and how does it work.
- Overview of Implementation in GLAST
- Validation (or *Sea Trials*)

Reference: *Data Analysis Techniques in HEP* by Fruthwirth et al, 2000

Kalman Filter

The Kalman filter process is a successive approximation scheme to estimate parameters

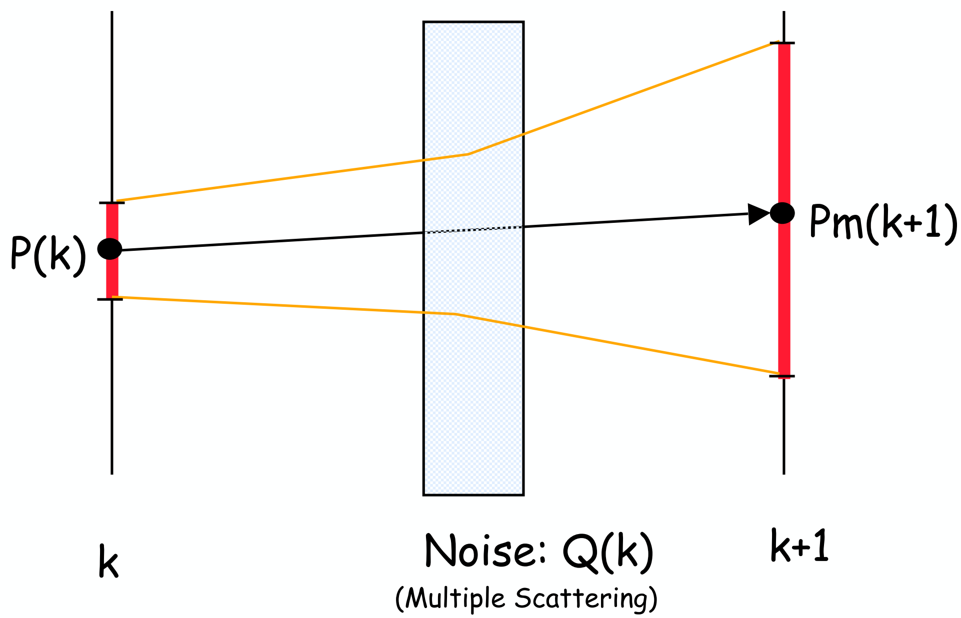
Simple Example: 2 parameters - intercept and slope: $x = x_0 + S_x * z$; $P = (x_0 , S_x)$

Errors on parameters x_0 & S_x (covariance matrix): $C = \begin{pmatrix} Cx\text{-}x & Cx\text{-}s \\ Cs\text{-}x & Cs\text{-}s \end{pmatrix}$

$Cx\text{-}x = \langle(x-x_m)(x-x_m)\rangle$

In general

$C = \langle(P - P_m)(P-P_m)^T\rangle$



Propagation:

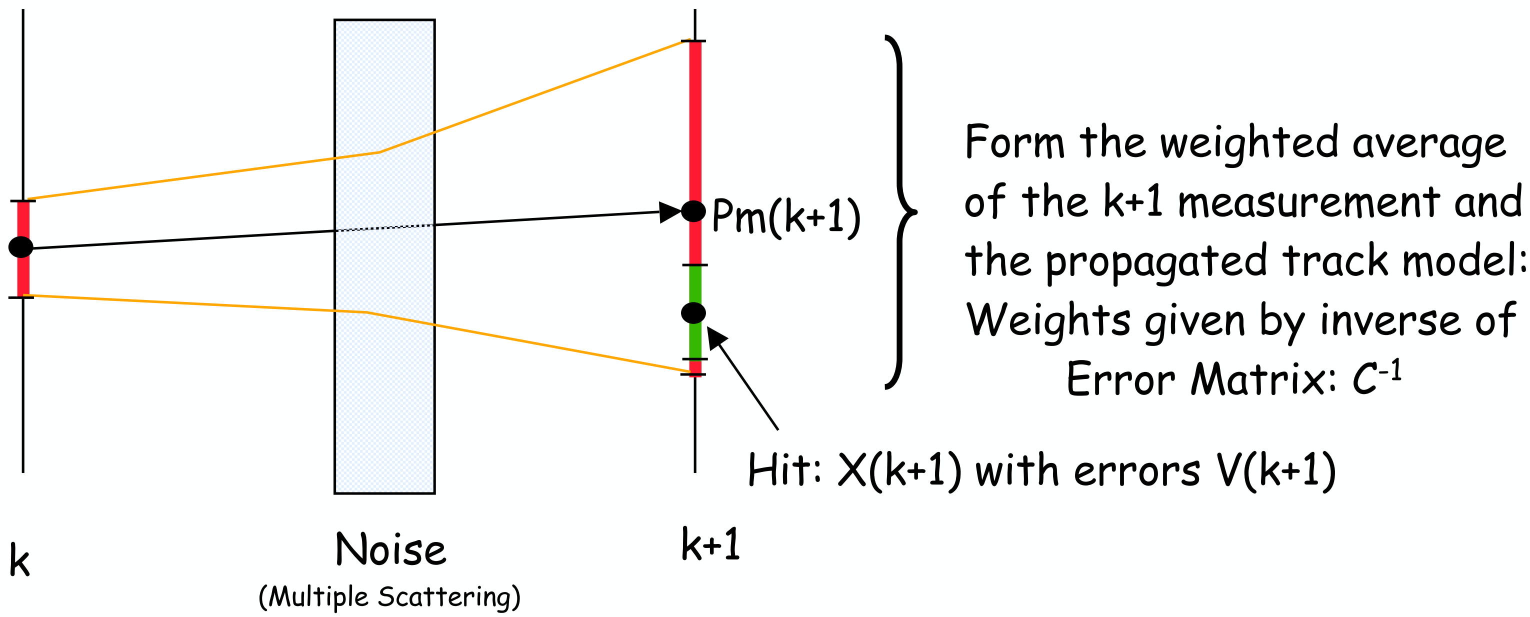
$x(k+1) = x(k)+Sx(k)*(z(k+1)-z(k))$

$Pm(k+1) = F(\delta z) * P(k)$ where

$F(\delta z) = \begin{pmatrix} 1 & z(k+1)-z(k) \\ 0 & 1 \end{pmatrix}$

$Cm(k+1) = F(\delta z) * C(k) * F(\delta z)^T + Q(k)$

Kalman Filter (2)



$$P(k+1) = \frac{Cm^{-1}(k+1)*Pm(k+1)+ V^{-1}(k+1)*X(k+1)}{Cm^{-1}(k+1) + V^{-1}(k+1)} \quad \text{and} \quad C(k+1) = (Cm^{-1}(k+1) + V^{-1}(k+1))^{-1}$$

Now its repeated for the k+2 planes and so - on. This is called FILTERING - each successive step incorporates the knowledge of previous steps as allowed for by the NOISE and the aggregate sum of the previous hits.

Kalman Filter (3)

We start the FILTER process at the conversion point

BUT... We want the best estimate of the track parameters
at the conversion point.

Must propagate the influence of all the subsequent Hits backwards to the beginning of the track - Essentially running the FILTER in reverse.

This is call the SMOOTHER & the linear algebra is similar.

Residuals & χ^2:

Residuals: $r(k) = X(k) - Pm(k)$

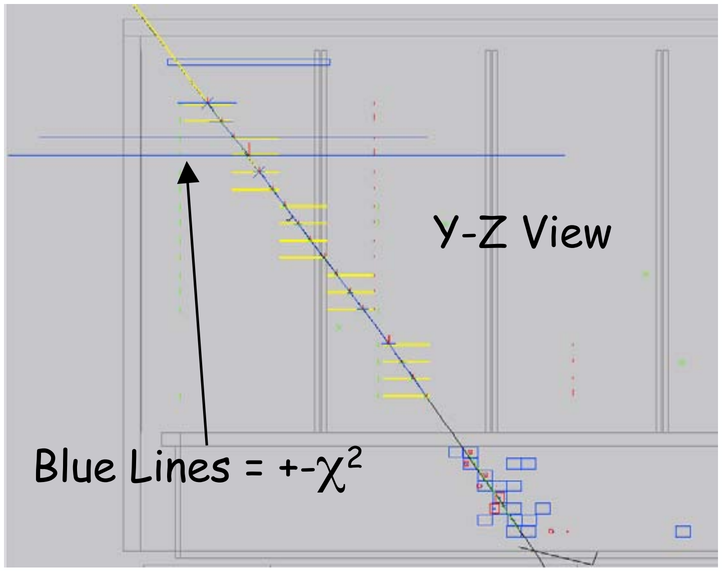
Covariance of r(k): $Cr(k) = V(k) - C(k)$

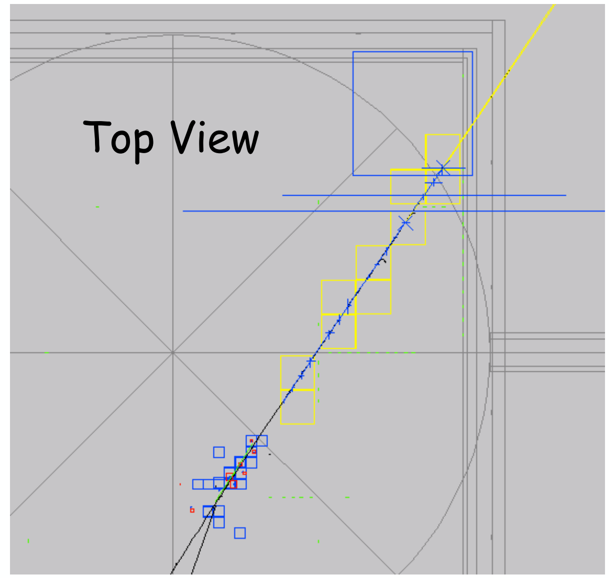
Then: $\chi^2 = r(k)^T Cr(k)^{-1} r(k)$ for the k^{th} step

Implementation in GLAST

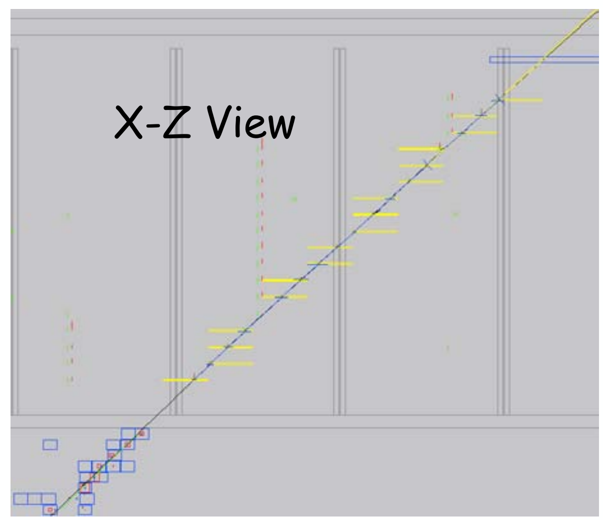
- 3 Dimensional: Essentially GLAST is composed of 2 - 2D trackers however multiple scattering mixes x & y. This creates correlations between the two projections and hence the covariance matrix (C) has significant off (block) diagonal terms.
- Difference between two separate projections and 3D projection becomes increasingly important as BOTH the S_x & S_y become large.
- Calculation of χ^2 involves both x & y and their correlation
- The SMOOTHed χ^2 is not a true χ^2 as errors are correlated point to point (not so for the FILTER χ^2). However since the smallest errors (and hence the largest weights) are the measurement errors the difference between them is small. (Presently we use the SMOOTHed χ^2)

χ^2 and the 1-Event Display





3 Views of a 1 GeV μ^+



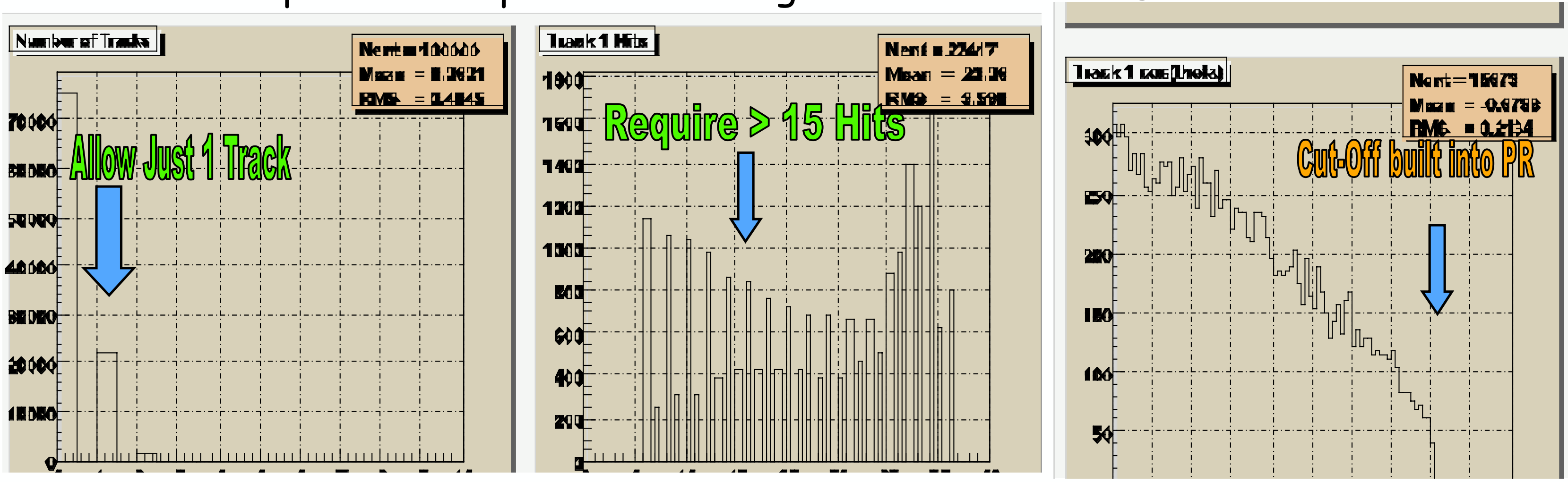
End-to-End Testing

Objective: Test if the implementation of the errors in the Kalman Filter Routines is Correct.

Method: Use Monte Carlo μ's (KE = 100 MeV, 1 GeV, & 10 GeV)
Provide the Kalman Filter with the correct energy ($p\beta$)

Test: If Monte Carlo generation of multiple scattering is the same as that in the Kalman Filter AND the calculation of the covariance matrices is correct AND their usage is correct THEN we except $\langle\chi^2\rangle \sim 1.0$ independent of position and angle.

μ's generated over $-1 < \cos(\theta) < 0$



The First Problem: $\chi^2 > 1$

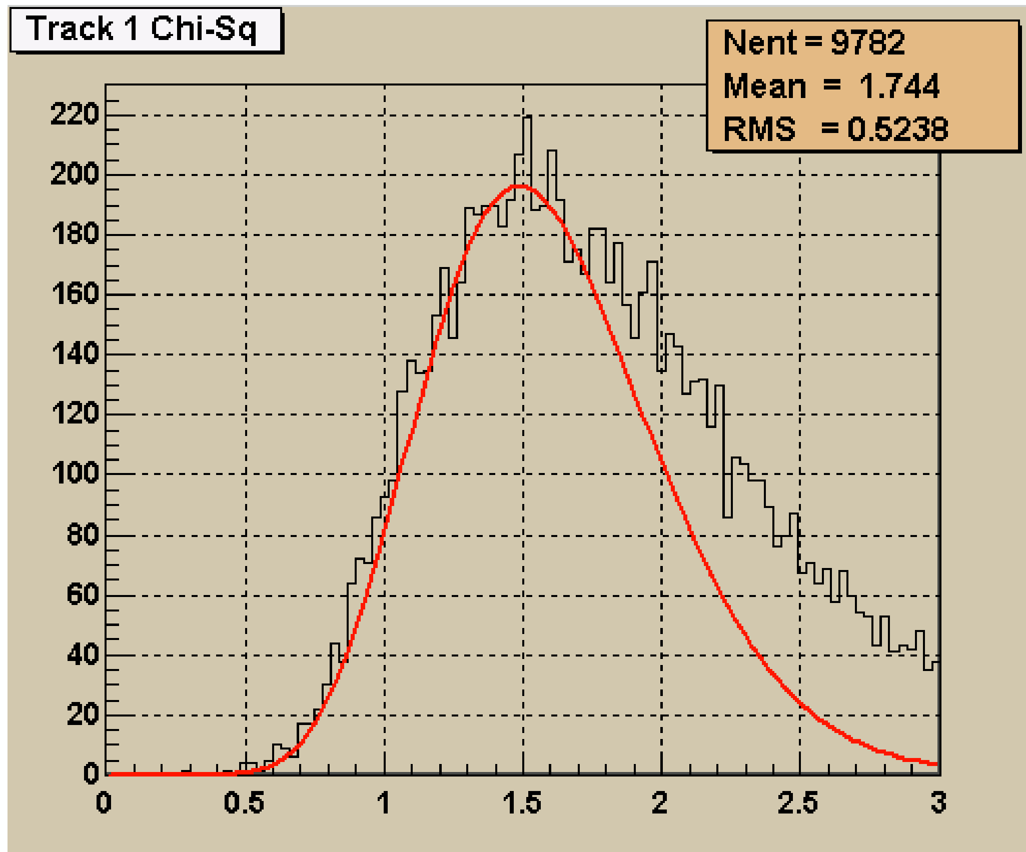
100 MeV - Normal Inc.

N_{hits} = 36

$\langle\chi^2\rangle$ = 1.6

Note: μ's generated with 100 MeV KE. This implies E_{tot} = 205.7 MeV and $p\beta$ = 151.4 MeV

Red Line: χ^2 function with parameters as above



Partial Solution - Include Energy Loss

100 MeV μ's entering the Tracker exit with ~ 65 MeV

First guess would give (assuming β ~ 1):
~ 22 MeV (50:50 for Si+C : W)

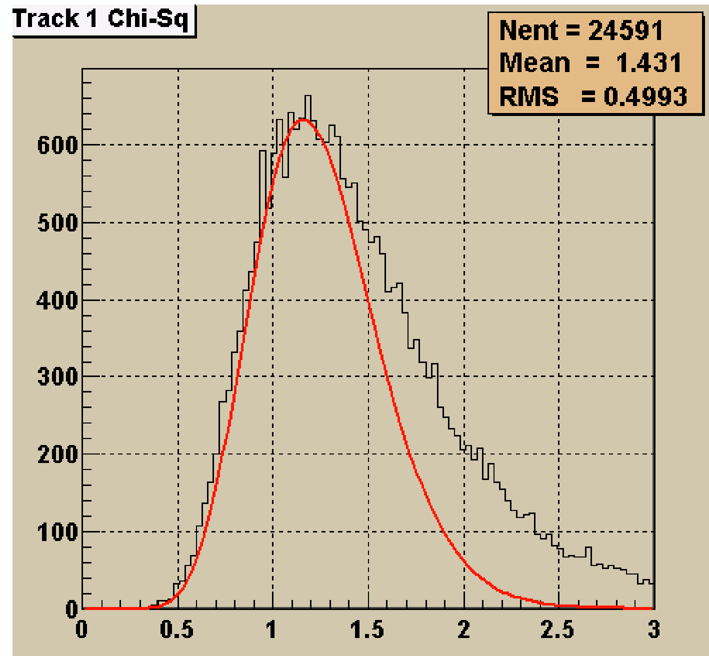
Correcting for β (= .85 const.):
~ 30 MeV

Integrating over path
~ 35 MeV

Implemented Bethe-Block Energy Loss in Kalman Filter (see results)

$\langle N_{hits} \rangle = 36$

$\langle \chi^2 \rangle = 1.25$

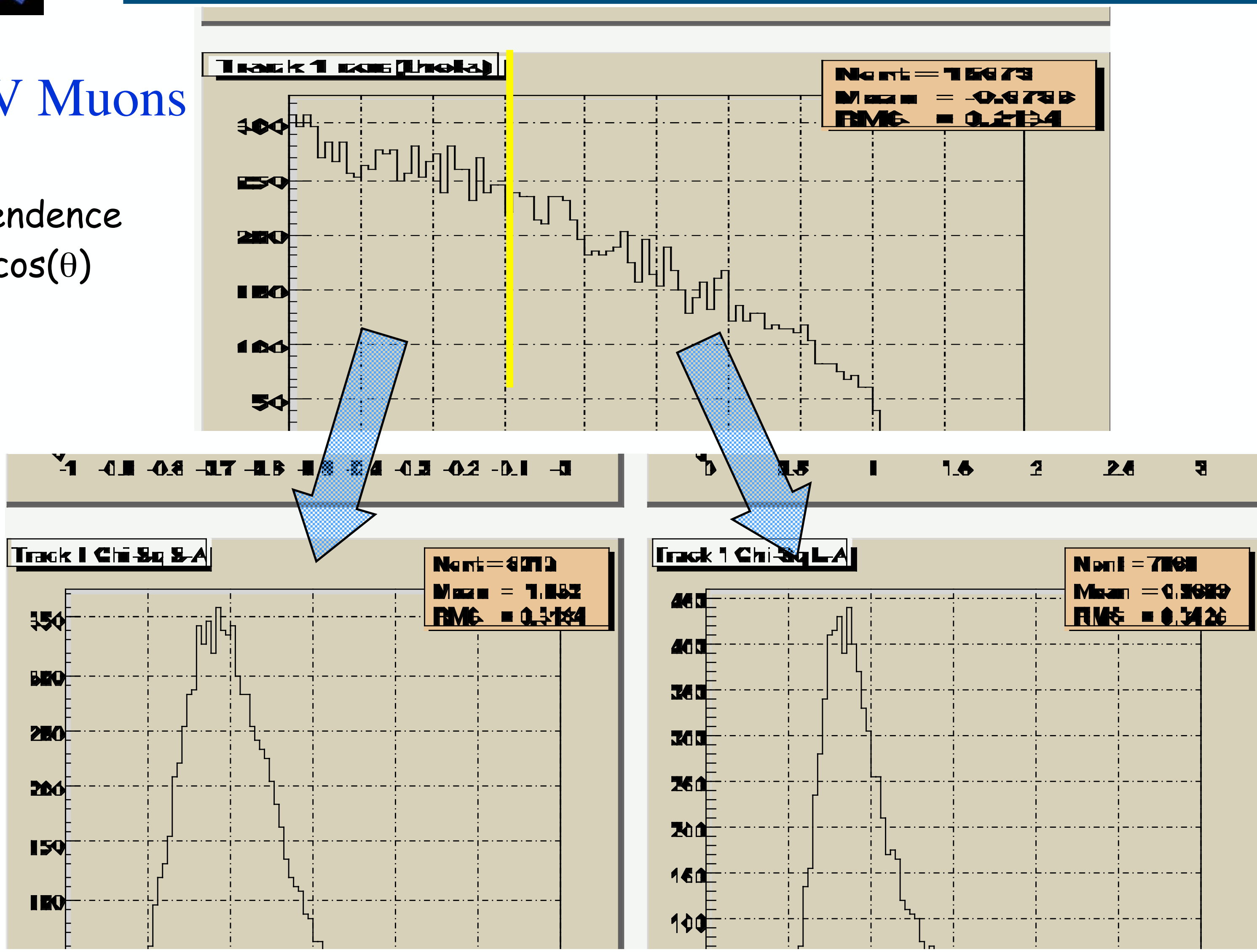


Problem becomes small by 1 GeV

Second Problem: χ^2 Depends on Angles

1 GeV Muons

Dependence on cos(θ)



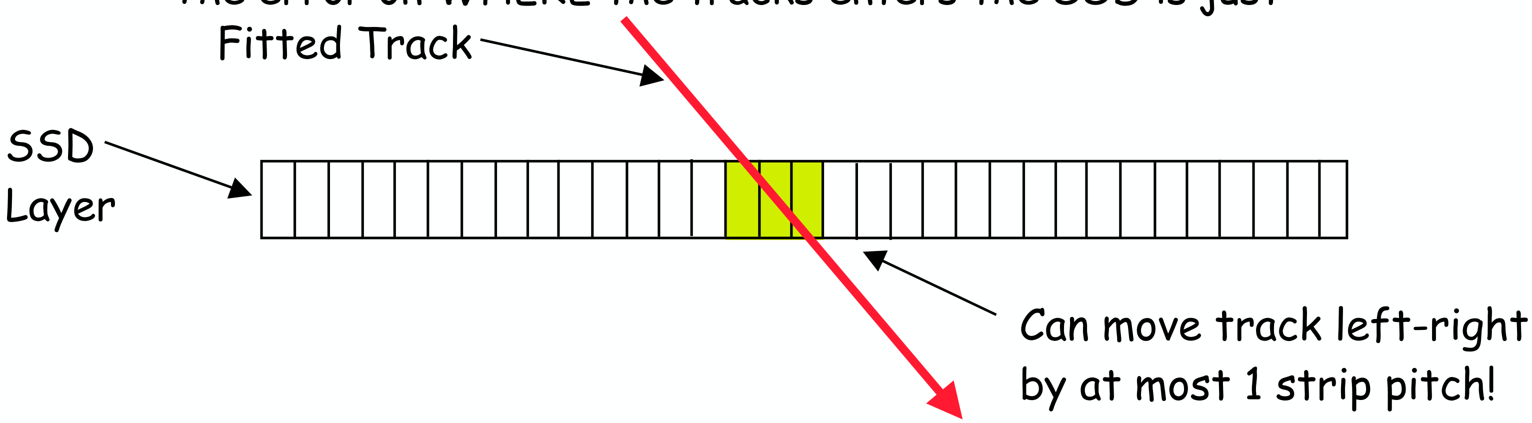
Measurement Errors (First Round)

The position error on a "square" distribution: $\frac{Width}{\sqrt{12}}$

Naively expect the error on a Cluster to be $\frac{ClusterWidth}{\sqrt{12}}$

But… Consider a track going through an SSD -
The Cluster edges determine the centroid
AND the are ~ 100% Correlated.

The error on WHERE the tracks enters the SSD is just $\frac{\sigma_P}{\sqrt{12}}$



10 GeV Muons - ϕ Dependence

Cluster Size
Error Dependence

Upper Plots:
Error ~ (Size * σ_P)

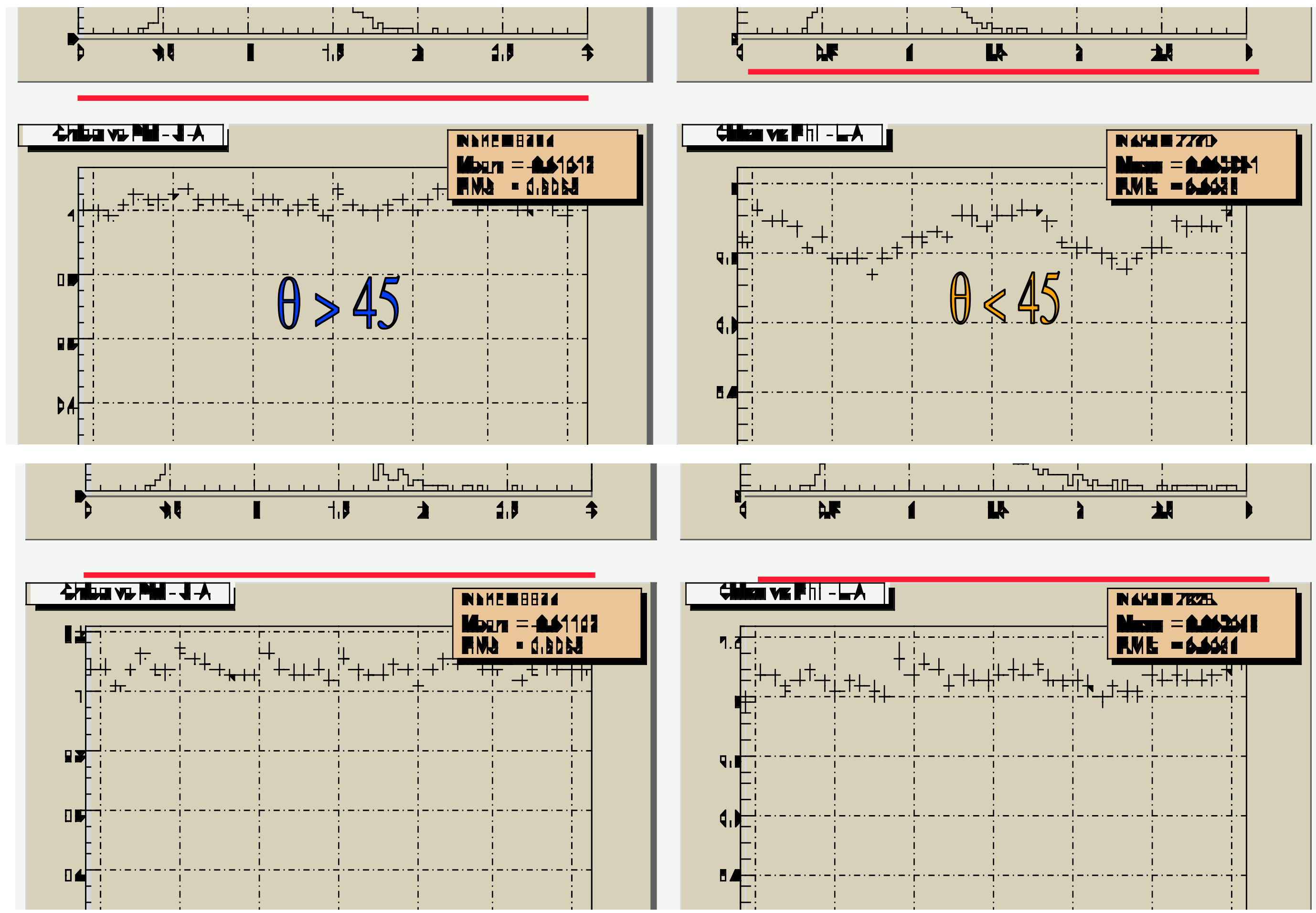
Resolution:
Meas. Errors

Lower Plots:
Error ~ σ_P

Where: $\frac{\sigma_P}{\sqrt{12}}$

σ_P =

RED Line at $\langle\chi^2\rangle = 1$



Third Problem: Tower Co-ordinate

Issue: To include or **not** include theTower Co-ordinate as well as the Strip Co-ordinate.

Inclusion controlled by the mapping of the measurements onto the parameters and visa-versa. (usually called the **H** matrix).

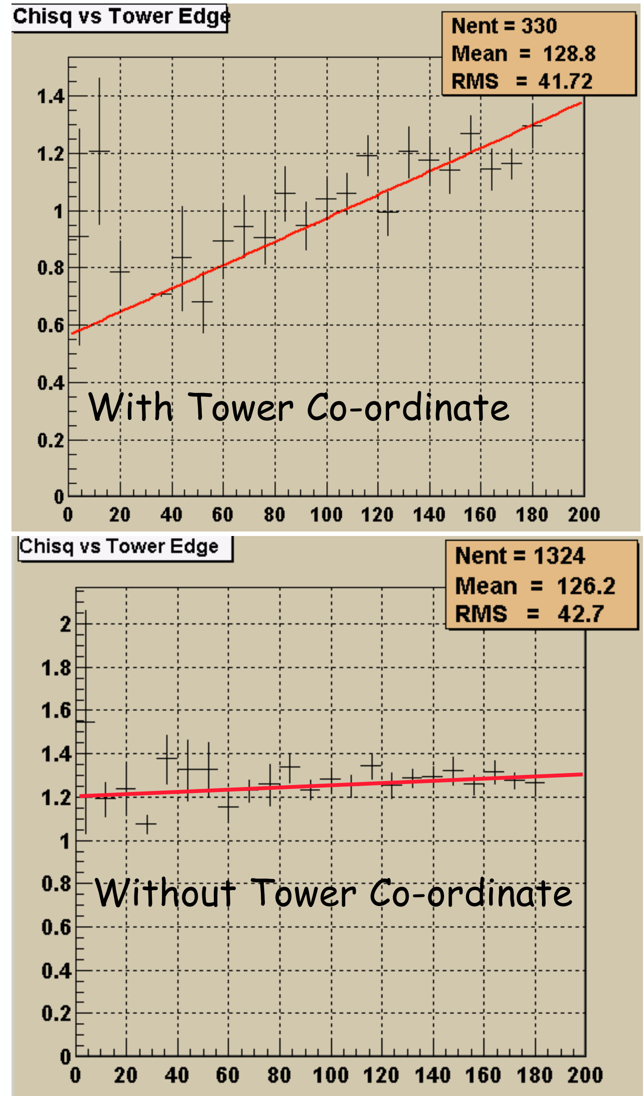
Reason not to include:

1) When results examined on a scale commensurate with bin size (Tower) binning effects appear.

2) Slight pull of fit toward center of tower at normal incidence.

3) Masks the χ^2 behavior of the strip co-ordinate.

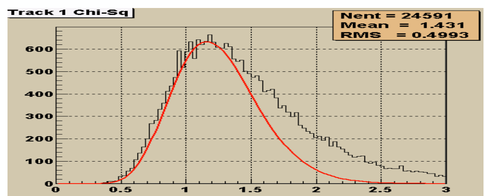
Tower Edge:
0. mm at Tower Center
187.5mm at Tower Edge



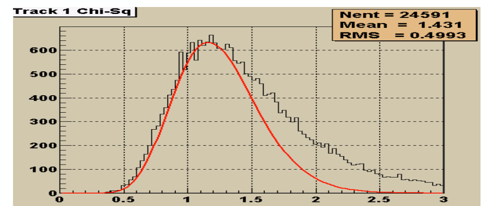
Energy and Angle Dependence

$\cos(\theta) = -1$ | **$-1 < \cos(\theta) < 0$**

100 MeV

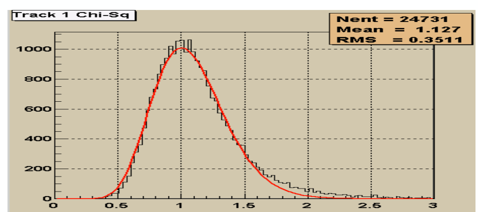


$\langle N_{hits} \rangle = 36 \quad \langle \chi^2 \rangle = 1.25$

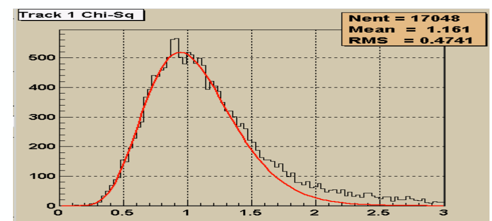


$\langle N_{hits} \rangle = 20 \quad \langle \chi^2 \rangle = 1.4$

1 GeV

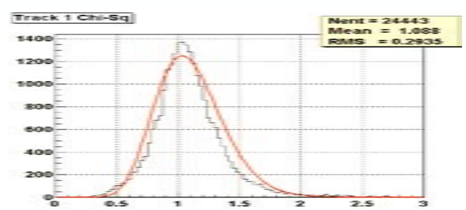


$\langle N_{hits} \rangle = 36 \quad \langle \chi^2 \rangle = 1.05$

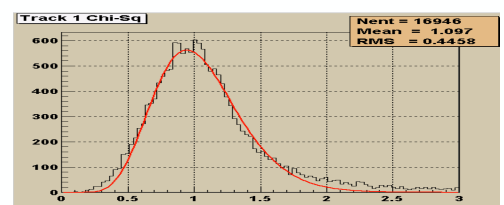


$\langle N_{hits} \rangle = 22 \quad \langle \chi^2 \rangle = 1.06$

10 GeV



$\langle N_{hits} \rangle = 36 \quad \langle \chi^2 \rangle = 1.08$



$\langle N_{hits} \rangle = 24 \quad \langle \chi^2 \rangle = 1.05$

Conclusions

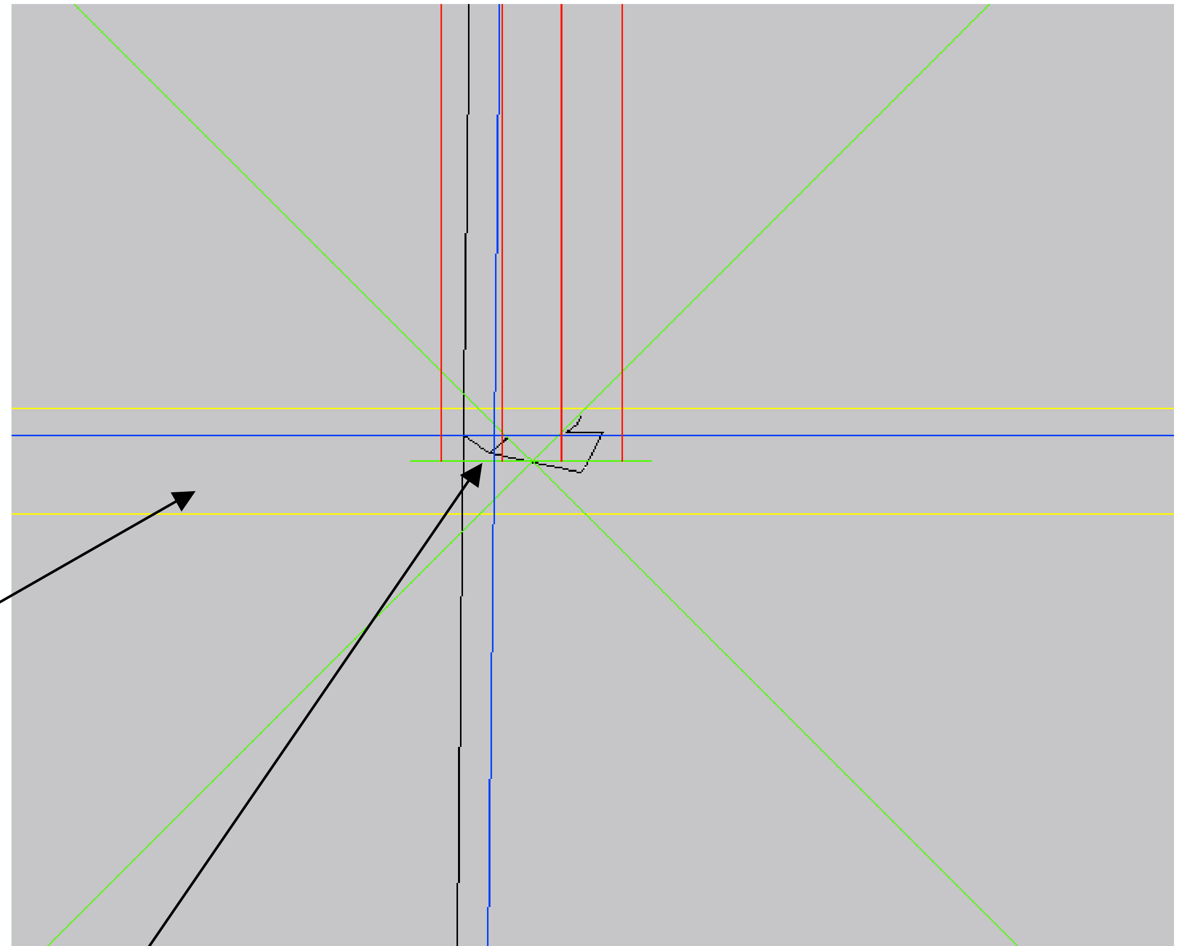
A 3 dimensional Kalman Filter has been implemented

The errors, as reflected in χ^2

- Are ~ not dependent on the Polar Angle (θ)
- Are ~ not dependent on the Azimuthal Angle (ϕ)
- **DO** depend on energy:
  - remaining error in Kalman Multiple Scattering?
  - G4 give MS 10% larger then Wallet Card Formulas?

The match of χ^2 distributions to the ideal case is reasonable.

Strip & Cluster Meas. Errors: Round 2



What about δ-rays?

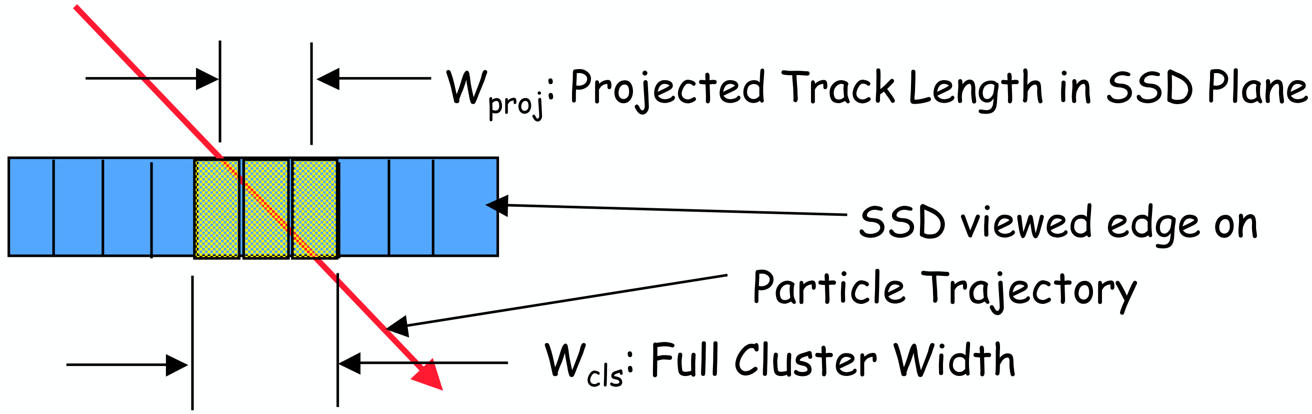
Examples as shown at the left cause large contributions to χ^2

SSD viewed edge on

δ-ray

Strip & Cluster Meas. Errors

This idea is still under development!



Track Measurement Error $\delta_{meas} = (W_{cls} - W_{proj})/\sqrt{12}$

Predicted effects (wish-list):

1) Unweight oversized clusters on track - lessening effect of δ_{rays}

2) Implement the full measurement covariance (position & slope)

Perhaps $\sqrt{2}$ improvement is possible for some clusters!

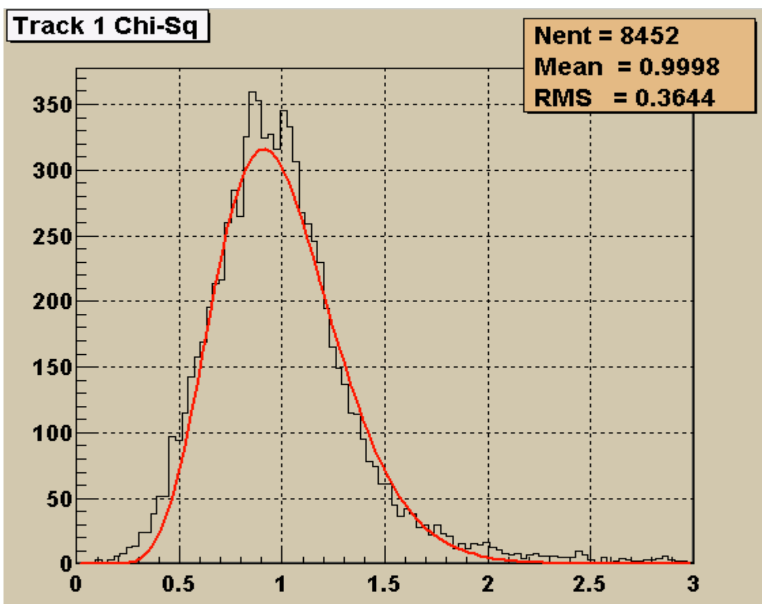
Kalman Filter: Sea Trials

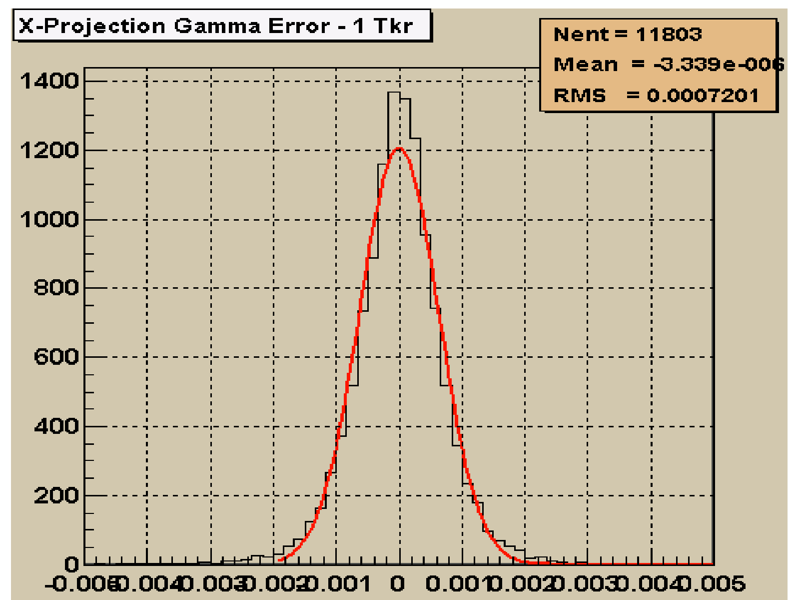
Several Problems discovered During "Sea Trails" Phase

- Proper setting of measurement errors
- Proper inclusion of energy loss (for μ's - Bethe-Block)
- Proper handling of over-sized Clusters

End Results: Example 10 GeV μ's

$\langle N_{hits} \rangle = 24$ $\langle \chi^2 \rangle$ = 1.0/DoF $\langle \sigma_{FIT} \rangle$ = .63 mrad





Setting the Energies

Track energies are critcal in determining the errors
(because of the dominance of Multiple Scattering)

A Three Stage Process:

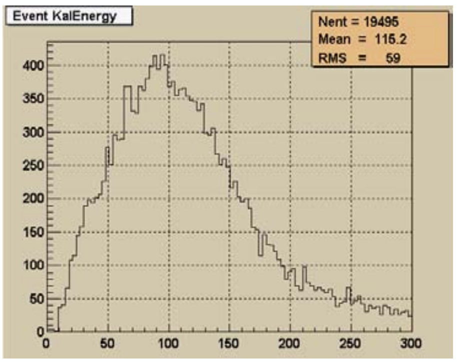
• Kalman Energies: compute the RMS angle between 3D Track segments
Key: include material audit and reference
energy back to first layer
Results: $\sigma_{E\text{-Kalman}}$ ~ 35% @ 100 MeV (!)

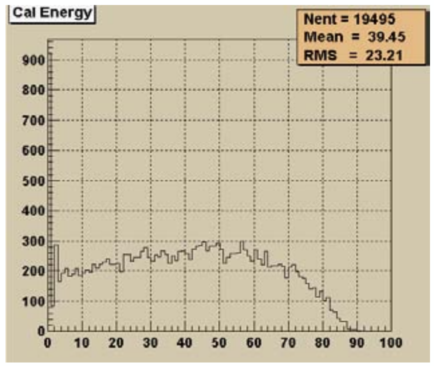
• Determine Global Energy:
E_{Golbal} = Hit counting + Calorimeter Energy
(Resolution limited by Calorimeter response)
Results: Depends on Cuts - Best ~ 12% at 100 MeV

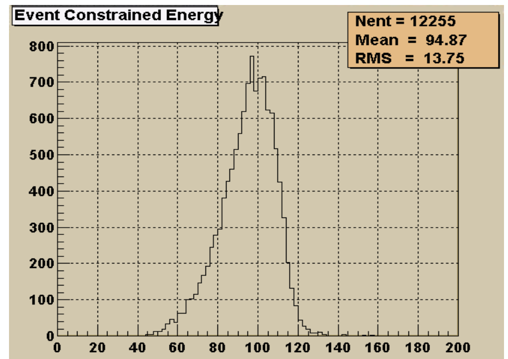
• Use Global Energy to Constrain the first 2 track energies:
$E_{Golbal} = E1_{Kal} + x1*\sigma1_{Kal} + E2_{Kal} + x2*\sigma2_{Kal}$
$\chi^2 = x1^2 + x2^2$
Determine x1 & x2 by minimizing χ^2

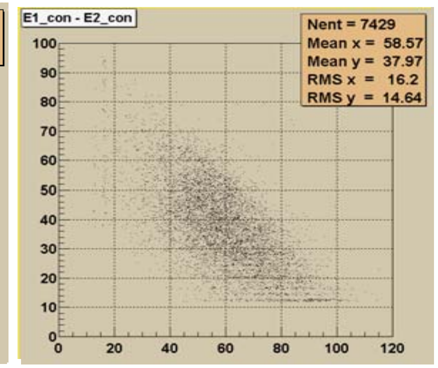
How do Energies Look?

100 MeV γ's - Normal Incidence



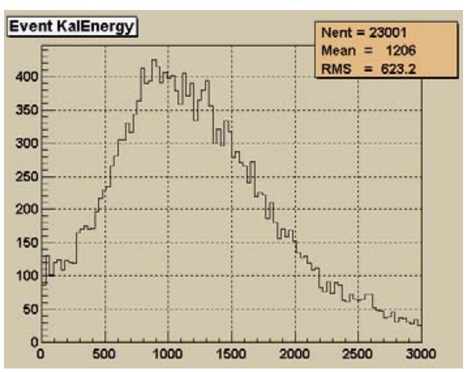


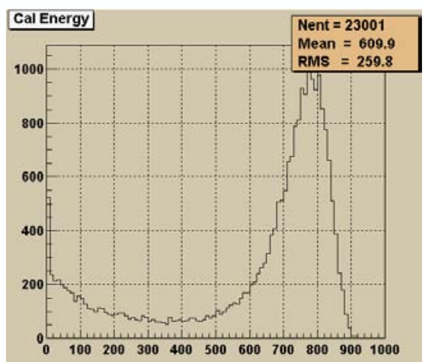


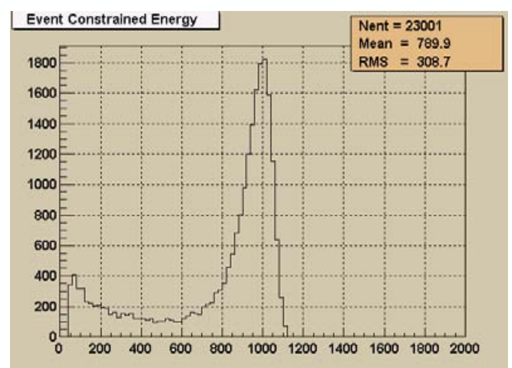


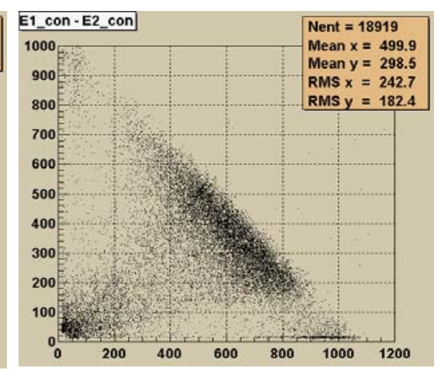
Ecal > 30 MeV

1000 MeV γ's - Normal Incidence

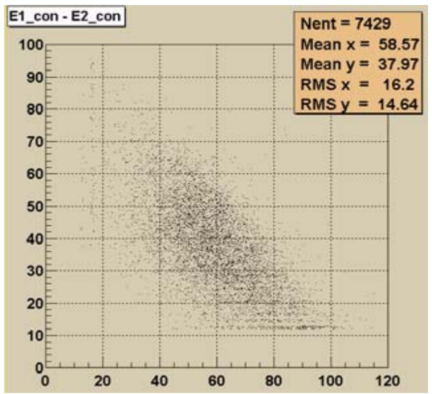


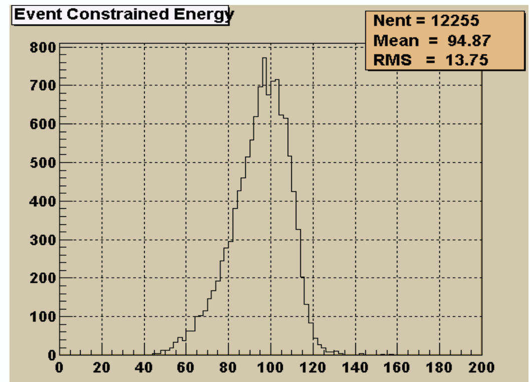






100 MeV - Ecal > 30 MeV





The Final Fits & Creating a γ

A Second Pass through the Kalman Fit is done

- Using the Constrained Energies for the First two tracks - others use the default Pat. Rec. energy

- The Track hits are NOT re-found - the hits from the Pat. Rec. stage are used

Creating a γ: (Note this isn't true "Vertexing")

- Tracks are multiple scattering dominated - NOISE Dominated Verticizing → adding NOISE coherently

- Use tracks as ~ independent measures of γ direction
- Process:
  - Check that tracks "intersect" - simple DOCA Calc.
  - Estimate Combined direction using Track Errors and Constrained Energies to form the weights

The Bottom Line: How does it all Work?

Data for 100 MeV, Nrm. Inc.

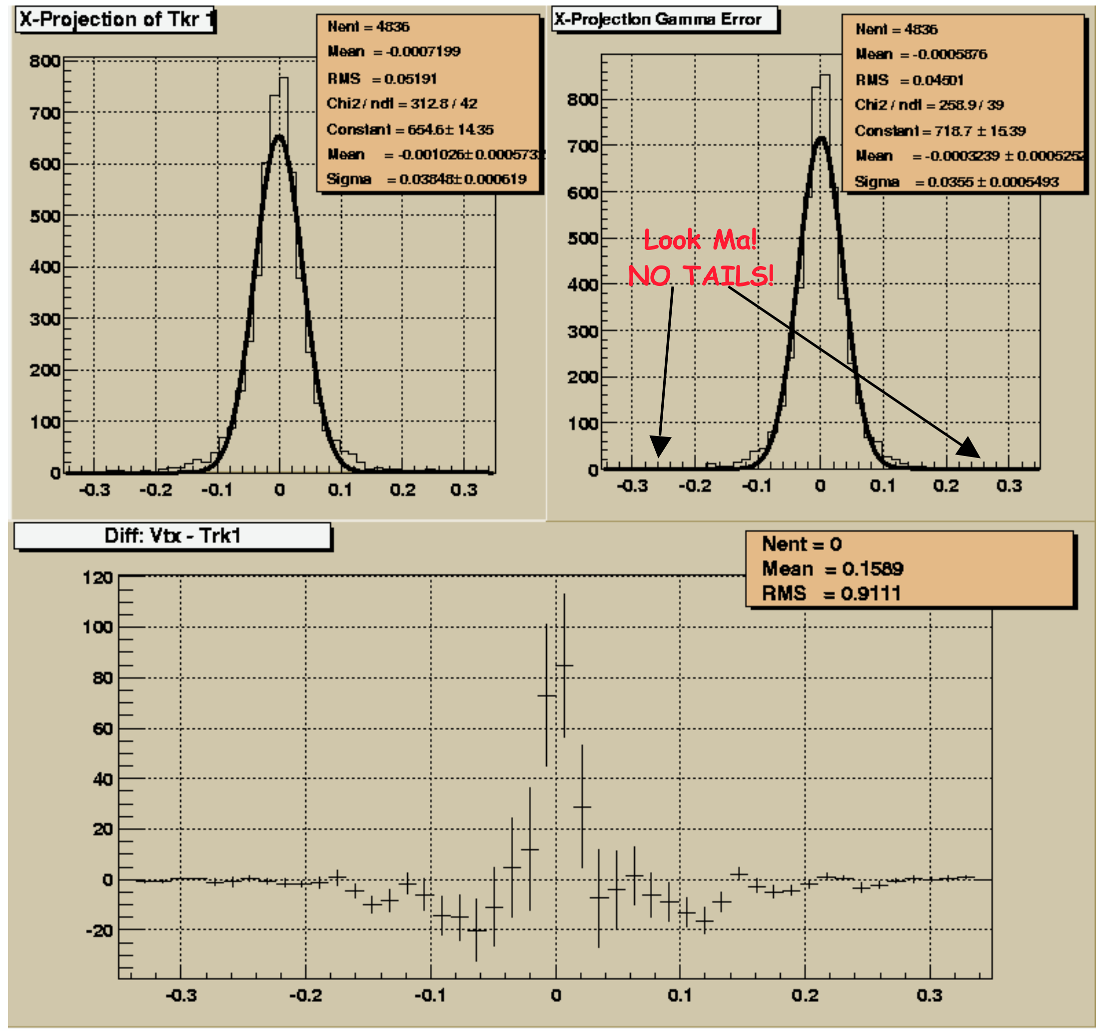
Thin Section Only - Req. All Events to have 2 Tracks which formed a "vertex"
Results: A_{eff} ~ 3000 cm^2

Best Track Resolution:
39 mrad

γ Resolution:
35 mrad

Difference Plot Shows the Improvement!

But… the story is even Better!



Dialing in Your PSF!

The PSF for γ's turn out to depend on the Opening Angle between the 2 Tracks

In retrospect this is now Obvious! - Parallel Tracks → minimal MS!