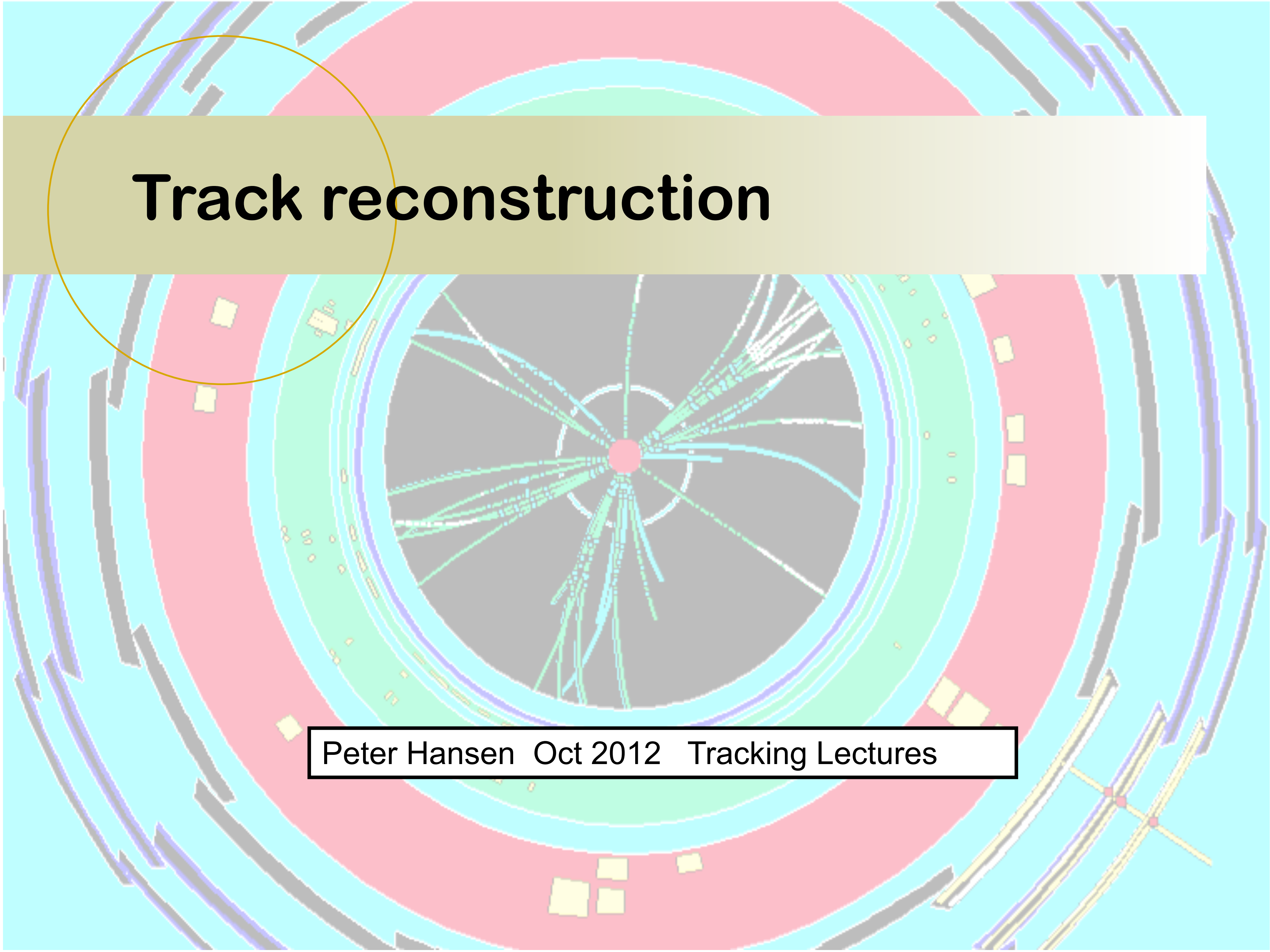
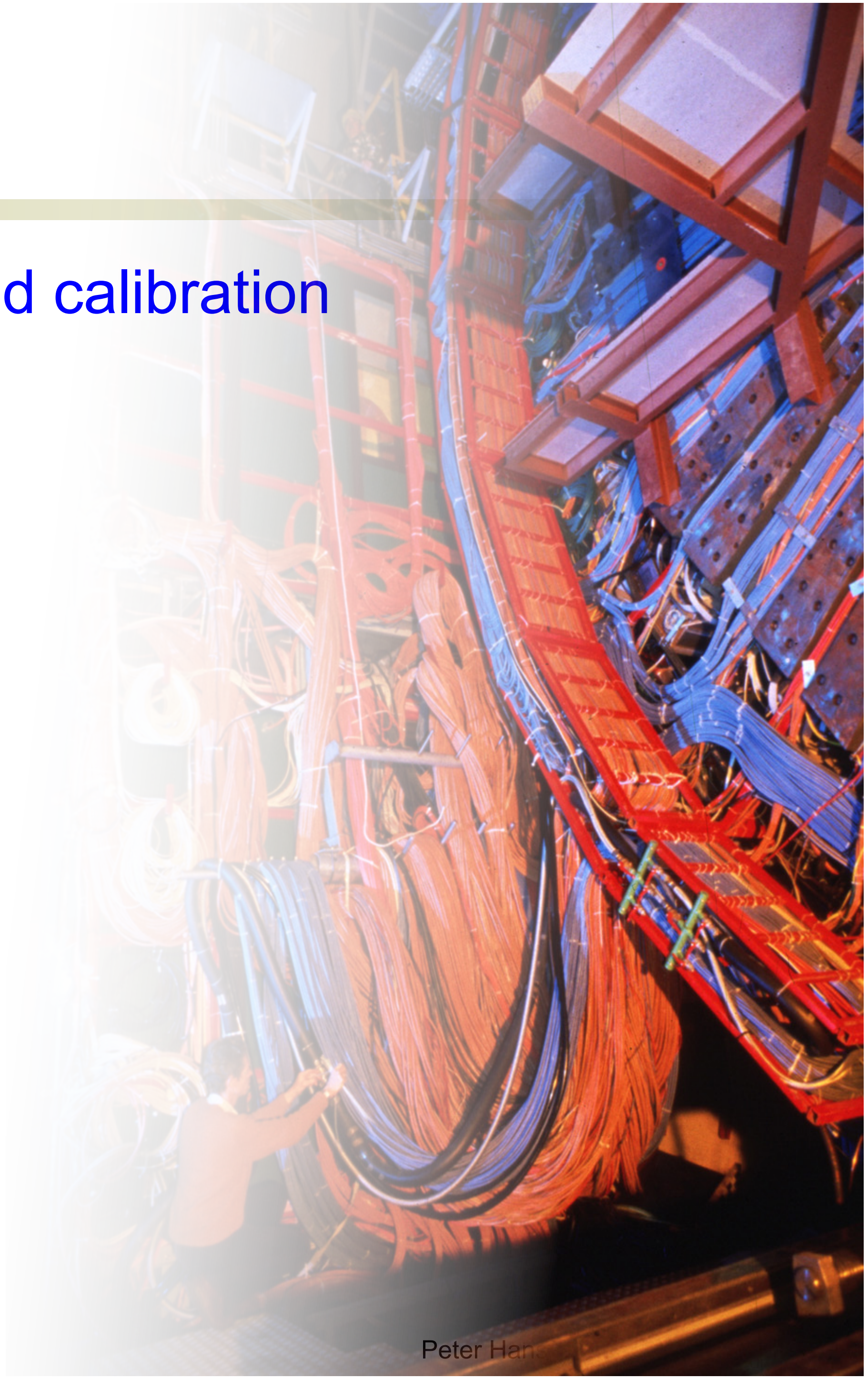
Track reconstruction



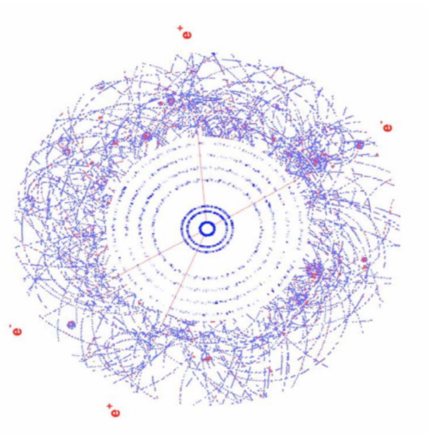
Peter Hansen Oct 2012 Tracking Lectures

Overview

- Spacepoint formation and calibration
- Pattern recognition
- Track fitting methods
- Vertexing
- Alignment



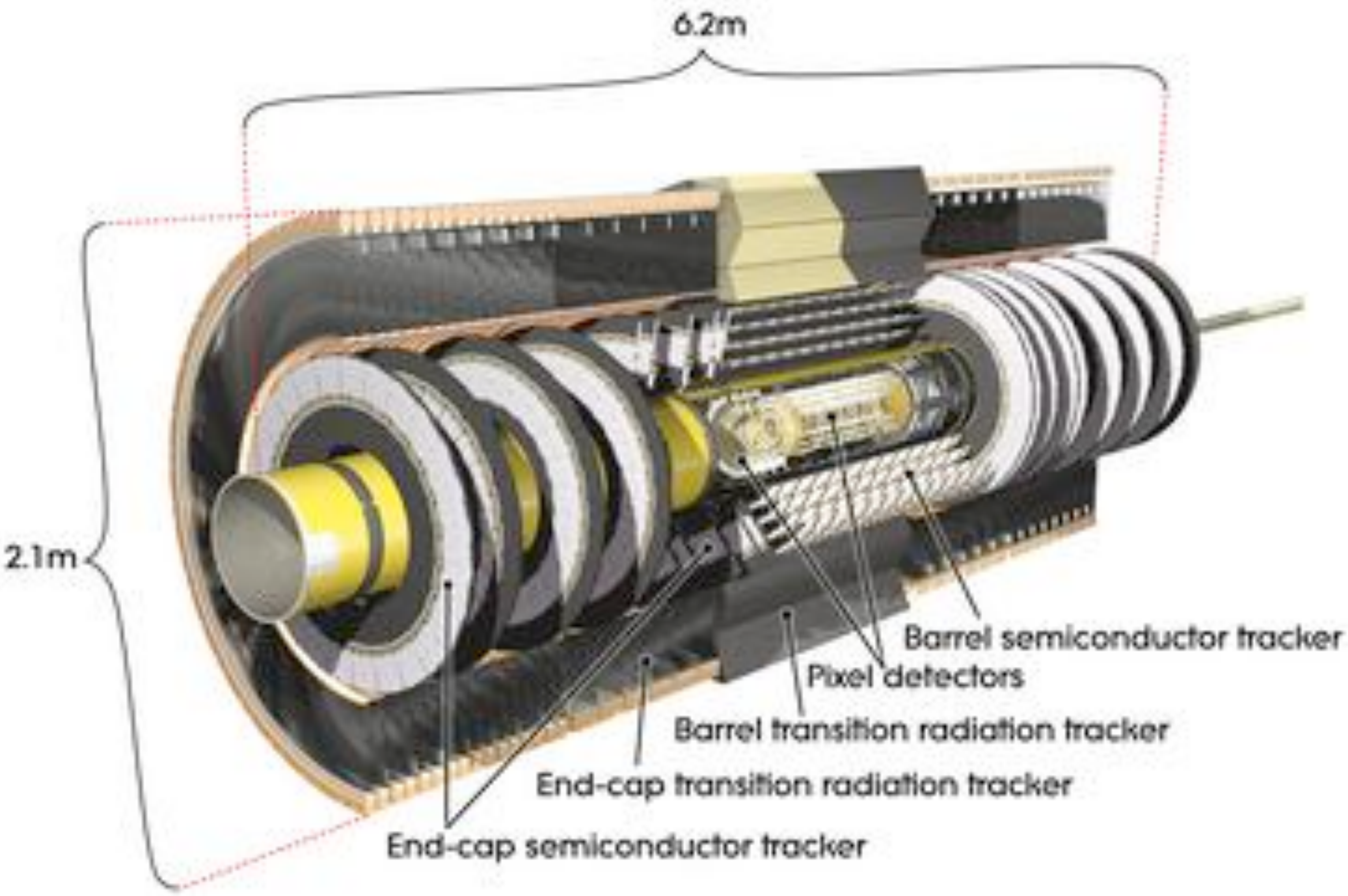
The tracking challenge



- Every second, 40 millions events are happening at the LHC, with thousands of tracks from up to 40 individual collisions. A few hundred of these are to be selected to mass-storage for later processing.
- To cope with the high density and momenta of the tracks very many channels are needed which cause rather large amounts of material in the tracking detectors.
- Thus, we need highly sophisticated and error-tolerant track-finders and –fitters, good calibration and alignment methods, robust vertexing and particle identification.

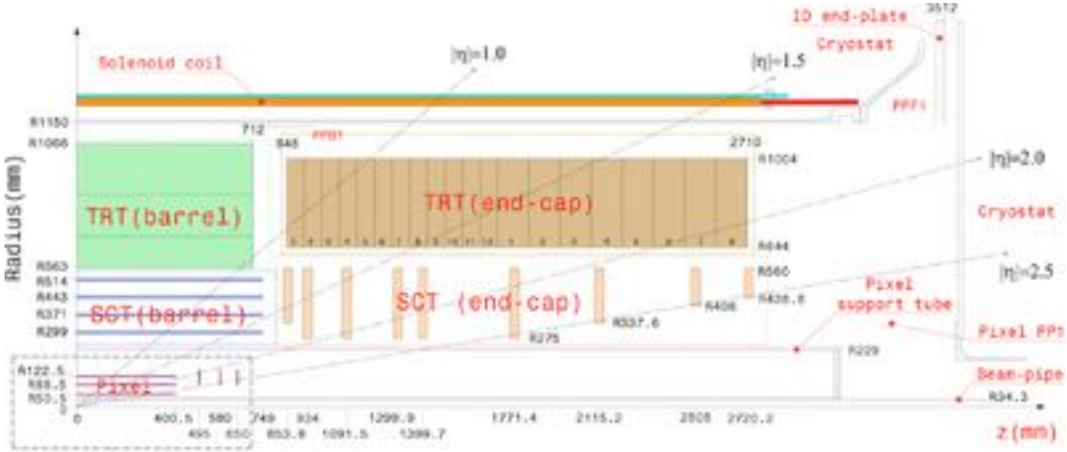
◆ This lecture covers a few standard tracking methods.

The ATLAS Inner Detector

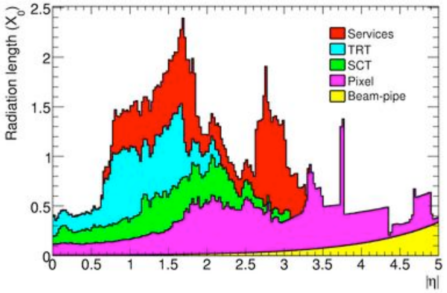

6.2m
2.1m
Barrel semiconductor tracker
Pixel detectors
Barrel transition radiation tracker
End-cap transition radiation tracker
End-cap semiconductor tracker

I use many ATLAS examples in this lecture since I come from the ATLAS TRT community, but the points made are completely general.
For example, one point is to build planes perpendicular to the tracks…

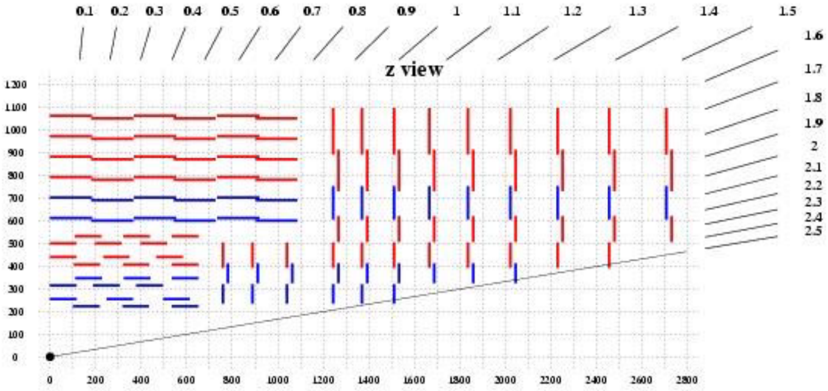
The Inner Detector



The ATLAS ID



ATLAS ID material



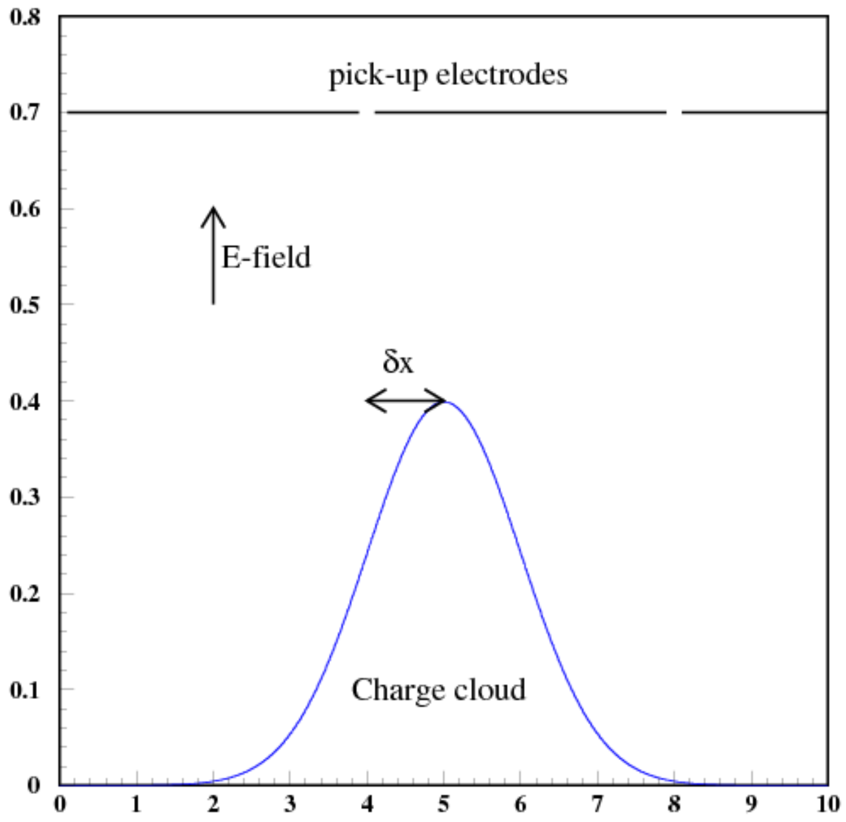
The CMS silicon strips

Spacepoint formation

- Most tracking detectors register “hits” from signals induced on pickup electrodes by an *electron cloud* made by a track.
- In case of a hit on only one electrode, the precision is $\Delta/\sqrt{12}$ (Δ = the electrode size).
- Much better is it if the signal is distributed over two electrodes.

In that case $$\frac{\delta x}{\Delta} = f(P_1, P_2, \sigma)$$

This gives higher accuracy, but you need to know both P and σ..



Blum, Ronaldi: TPC tracking book

Spacepoint formation

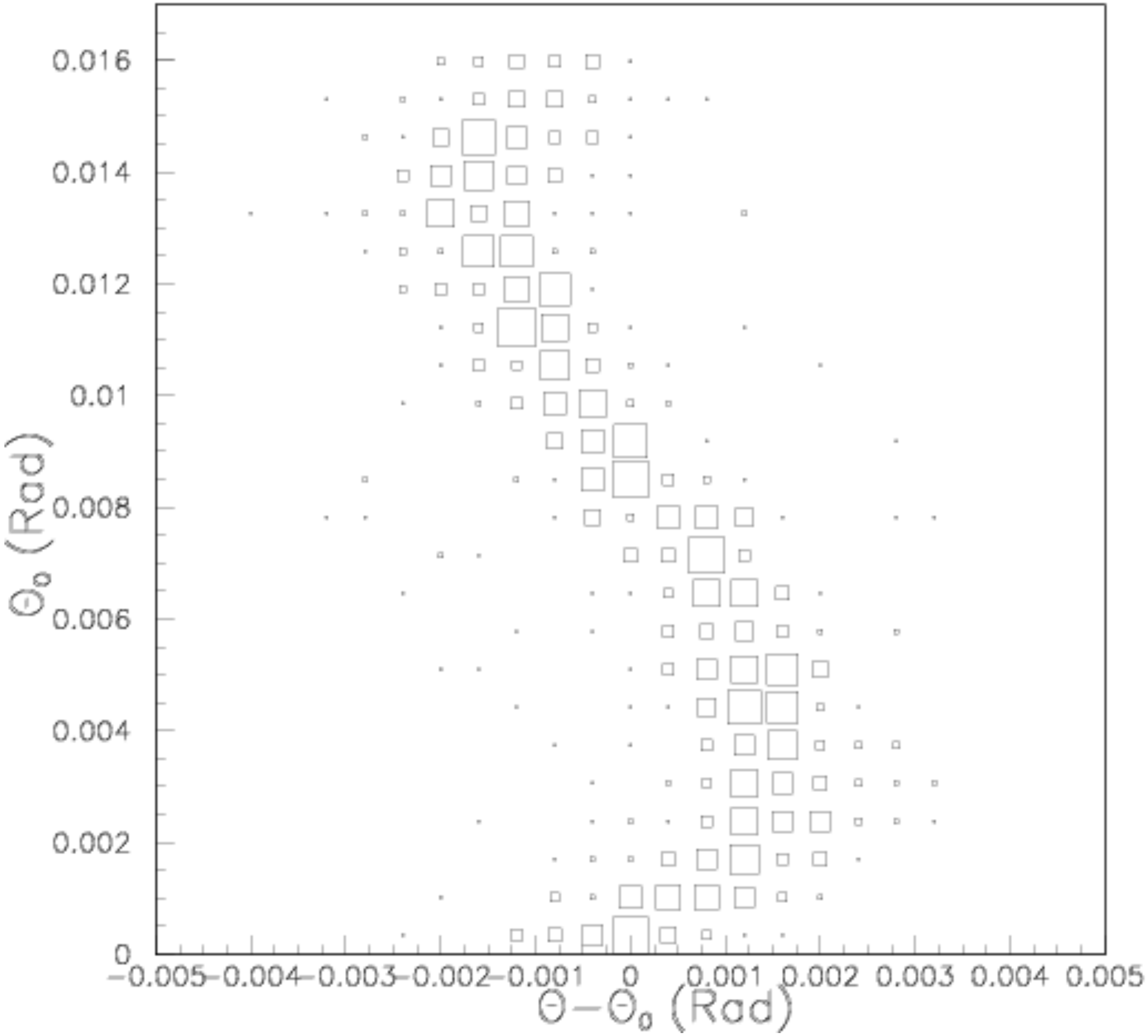
- If 3 or more electrodes pick up signal for one track passing the detector layer, it is possible to fit the width of the electron cloud. In that case the barycenter (pulseheight weighted average of the electrode centers) is a popular estimator of the track position.
- The pulseheights, P, must exceed a certain threshold and the electrodes must be spatially connected , forming a cluster. Summing over cluster cells, we get the barycenter:

$$x = \sum P_i x_i / \sum P_i$$

Spacepoint formation

- The barycenter is not perfect because of the finite size of the electrodes.
- Example:

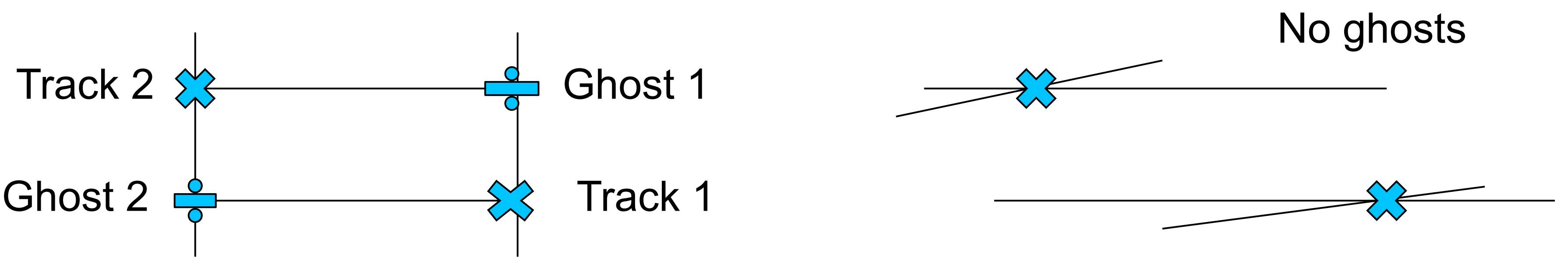
3x3cm electrodes in a lead-gas sampling calorimeter. You see that the estimate is only unbiased at the border between two or at centre of one.



Aleph EM calorimeter

Stereo view

- If you do not have pixels, only wires or strips, what about the second coordinate?
- In strip detectors double sided wafers are often used with strips on both sides having an angle between them. But large angles gives ghost hits!
- At high track densities, 20-80 mrad is a good choice, avoiding too many ghost hits, having good resolution in the bending plane and still some resolution in the second coordinate.



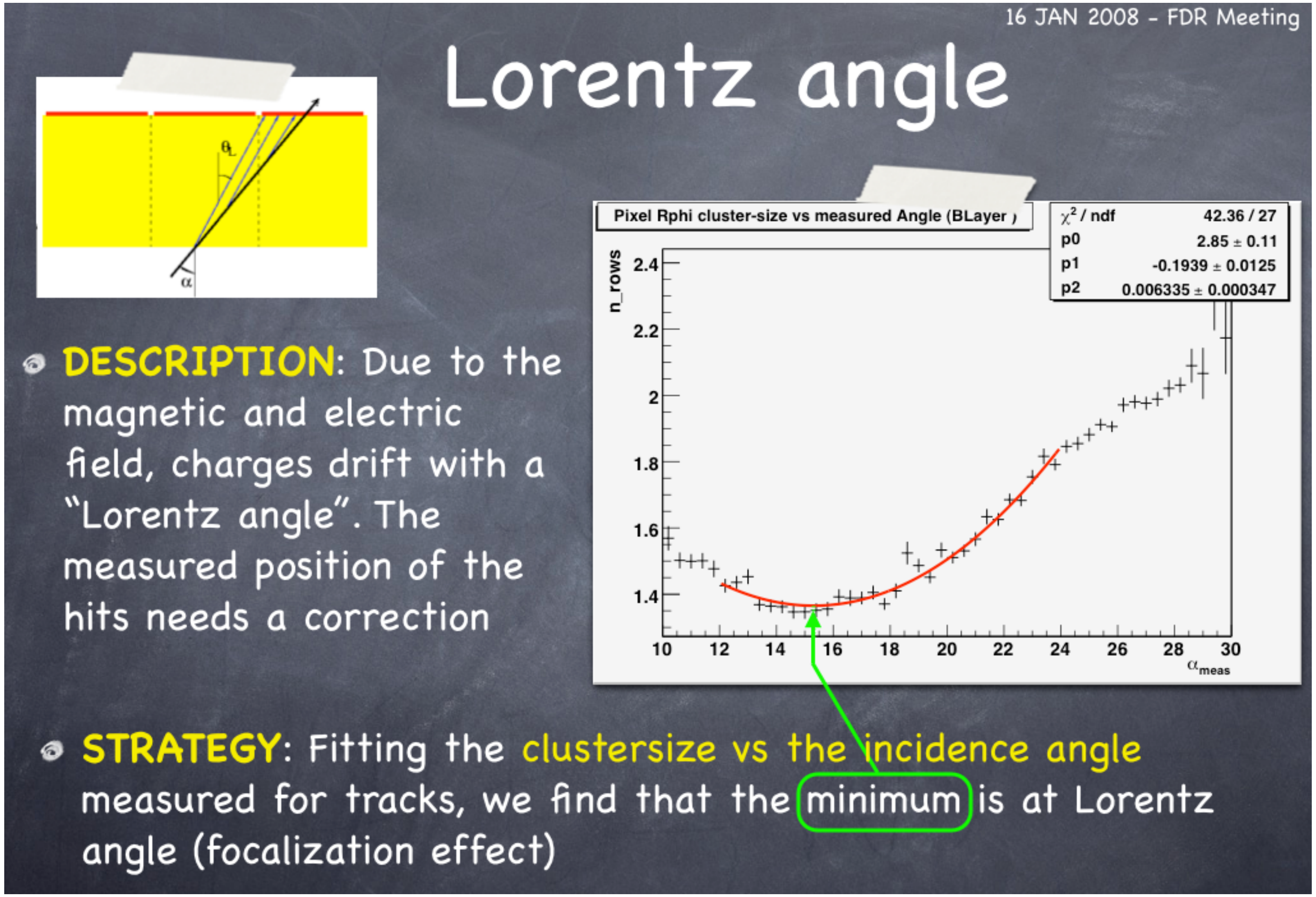
Spacepoint calibration

- In general we must know *the response function* , the probability distribution of induced pulse-heights for a given track impact
- (Actually, we would like the inverse: the pdf for the track impact, given the pulseheights. But we can not get that from testbeam..)
- The response function may vary from channel to channel and even vary in time. It must be *calibrated from data*.

Spacepoint calibration example

- Corrections are needed for crossing angle and change in electron drift direction caused by B-fields.

(picture from Simone Montesano)



16 JAN 2008 - FDR Meeting

Lorentz angle

- DESCRIPTION: Due to the magnetic and electric field, charges drift with a "Lorentz angle". The measured position of the hits needs a correction

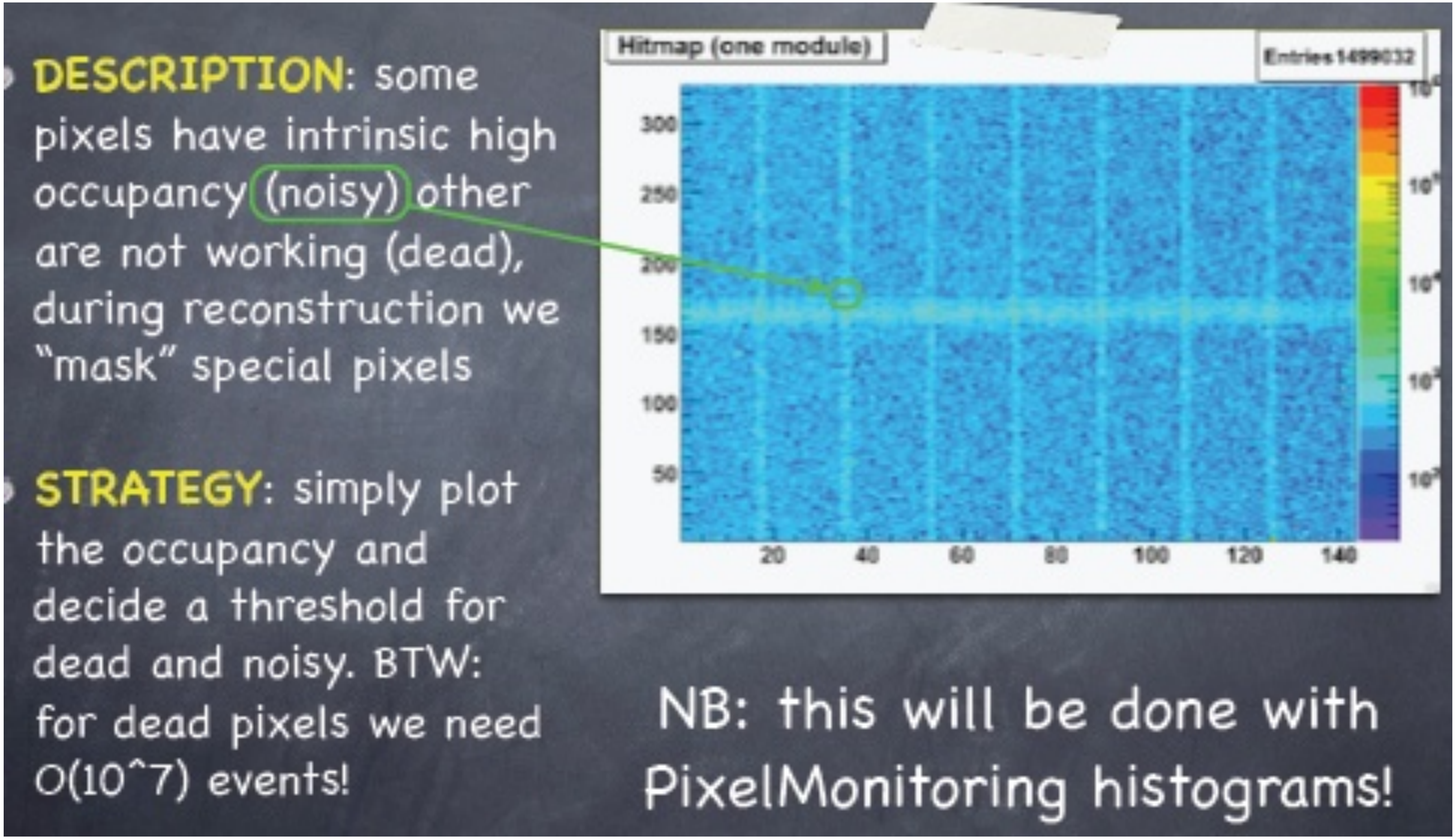
- STRATEGY: Fitting the clustersize vs the incidence angle measured for tracks, we find that the minimum is at Lorentz angle (focalization effect)

Splitting merged clusters

- At high track densities, clusters of fired detector cells from two different tracks may merge.
- For example, a jet with pT=1TeV has a minimum distance of only 0.1mm (on average) between two tracks at the innermost ATLAS pixel layer.
- A NN algorithm has been used to split them again (Prokofieff and Selbach 2012)
- In the calorimeters a search for local minima is used.

Dead and noisy channels

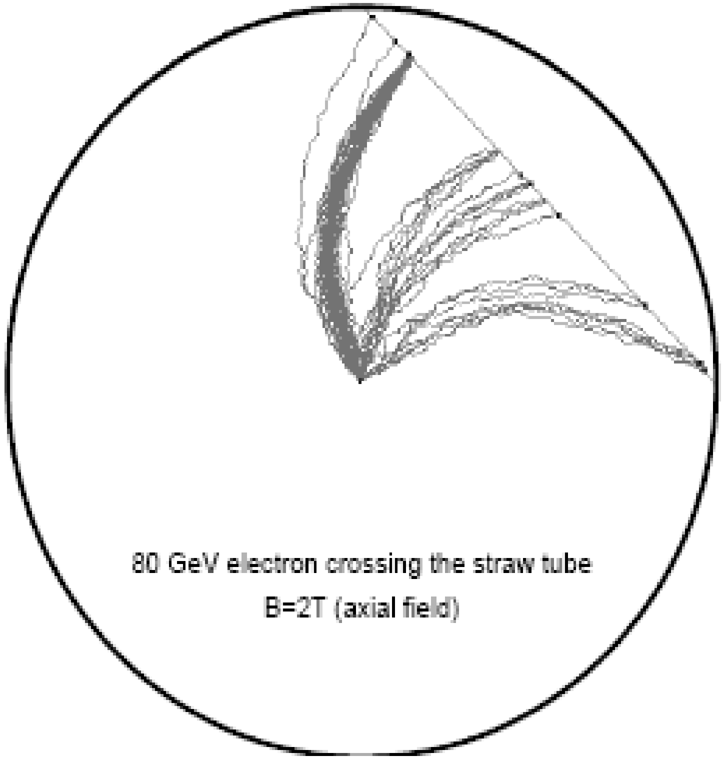
- Any clustering algorithm must handle dead or noisy channels to avoid false or split clusters.



- **DESCRIPTION**: some pixels have intrinsic high occupancy (noisy) other are not working (dead), during reconstruction we "mask" special pixels
- **STRATEGY**: simply plot the occupancy and decide a threshold for dead and noisy. BTW: for dead pixels we need O(10^7) events!

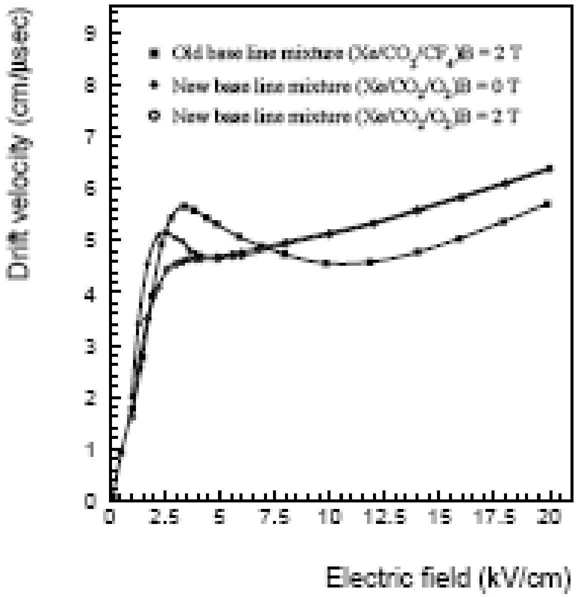
NB: this will be done with PixelMonitoring histograms!

Spacepoints in drift-tubes

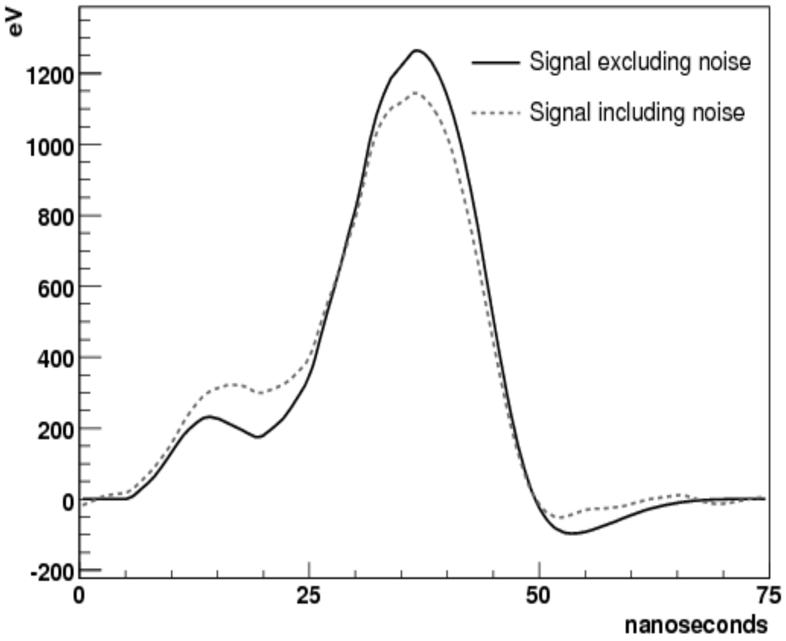


ATLAS TRT

In the ATLAS TRT we flag time-bins t in which the signal exceeds some threshold. Must calibrate the distance R(t-t0) from the track to the wire (in particular t0 varies by channel.)



Refinements of drift radii



ATLAS TRT

Large pulses will trigger the threshold sooner for the same track impact -> small correction for large time-over-threshold or High-Threshold hit.

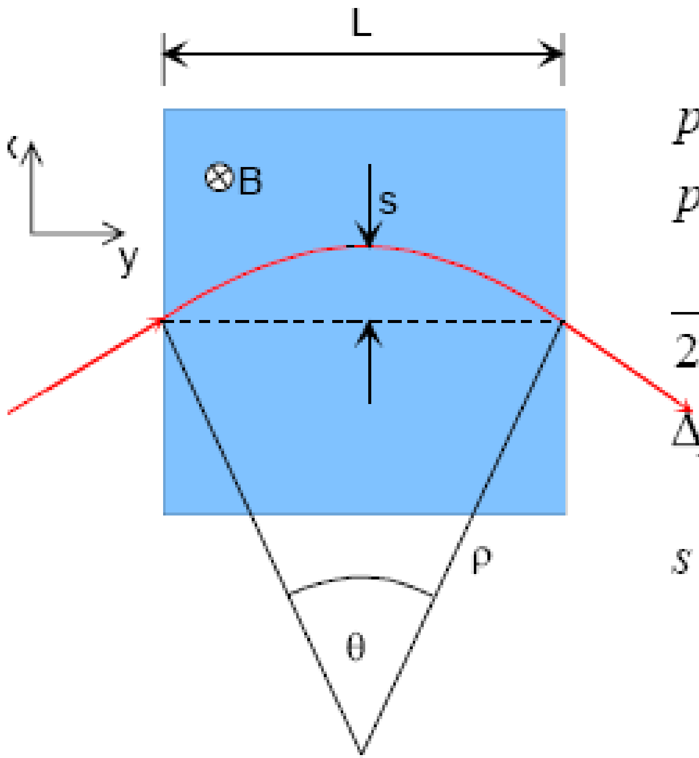
At a track refit, the track impact along the wire, angle and other info is available.

Small corrections for time-of-flight, signal propagation and other effects can be made at this point.

From space-points to tracks

- Given a collection of space-points we need to group together those space-points that belong to the same track and determine the track features.
- An important feature of a track is its momentum, so let us open a parenthesis on momentum measurement
- Then we will look at pattern recognition at the trigger level
- Then study two track fit algorithms: the Kalman filter and the global chi-squared fit.

Momentum measurement (..



$$p_T = qB\rho$$

$$p_T\ (\mathrm{GeV/c}) = 0.3B\rho \quad (\mathrm{T \cdot m})$$

$$\frac{L}{2\rho} = \sin\theta/2 \approx \theta/2 \quad \rightarrow \quad \theta \approx \frac{0.3L \cdot B}{p_T}$$

$$\Delta p_T = p_T \sin\theta \approx 0.3L \cdot B$$

$$s = \rho(1-\cos\theta/2) \approx \rho\frac{\theta^2}{8} \approx \frac{0.3}{8}\frac{L^2 B}{p_T}$$

This and next three slides are from Christian Jorams summer student lectures

Momentum accuracy

the sagitta s is determined by 3 measurements with error $\sigma(x)$:

$$s = x_2 - \frac{x_1 + x_3}{2}$$

$$\left.\frac{\sigma(p_T)}{p_T}\right|^{meas.} = \frac{\sigma(s)}{s} = \frac{\sqrt{\frac{3}{2}}\sigma(x)}{s} = \frac{\sqrt{\frac{3}{2}}\sigma(x) \cdot 8 p_T}{0.3 \cdot BL^2}$$

for N equidistant measurements, one obtains

(R.L. Gluckstern, NIM 24 (1963) 381)

$$\left.\frac{\sigma(p_T)}{p_T}\right|^{meas.} = \frac{\sigma(x) \cdot p_T}{0.3 \cdot BL^2} \sqrt{720/(N+4)}$$ (for N $\geq \approx$10)

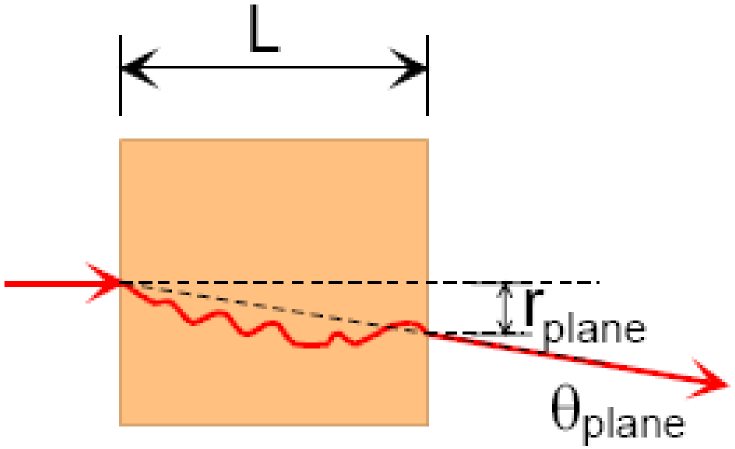
ex: p_T=1 GeV/c, L=1m, B=1T, $\sigma(x)$=200μm, N=10

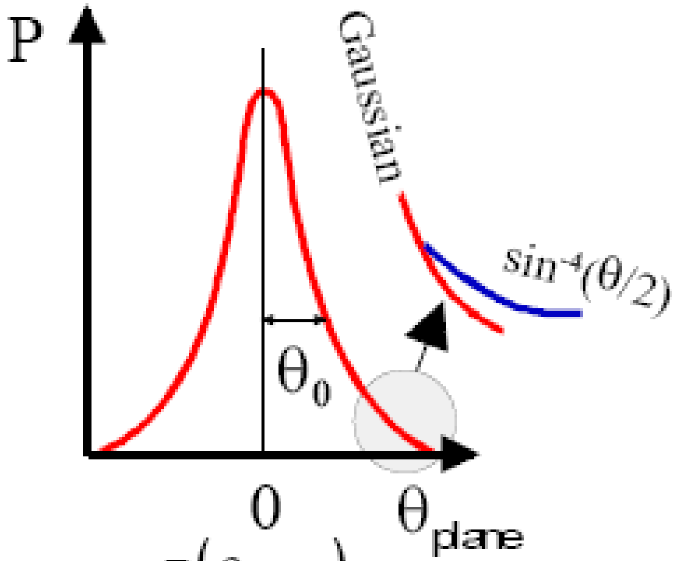
$$\left.\frac{\sigma(p_T)}{p_T}\right|^{meas.} \approx 0.5\%$$ (s ≈ 3.75 cm)

Multiple scattering

Sufficiently thick material layer

→ the particle will undergo multiple scattering.





$$\theta_0 = \theta_{plane}^{RMS} = \sqrt{\langle \theta_{plane}{}^2 \rangle} = \frac{1}{\sqrt{2}} \theta_{space}^{RMS}$$

$$P(\theta_{plane}) = \frac{1}{\sqrt{2\pi}\theta_0} \exp\left\{ -\frac{\theta_{plane}{}^2}{2\theta_0^2} \right\}$$

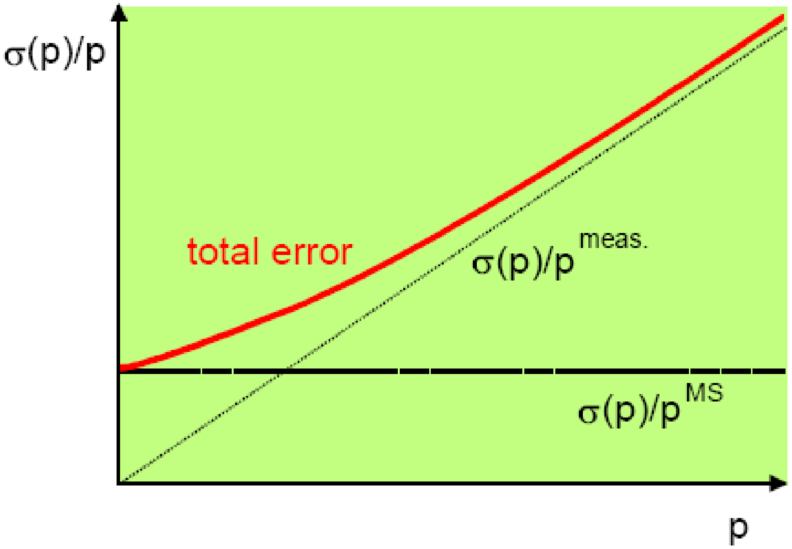
Total momentum error ..)

contribution from multiple scattering

$$\Delta p^{MS} = p\sin\theta_0 \approx p\cdot 0.0136\frac{1}{p}\sqrt{\frac{L}{X_0}}$$

$$\left.\frac{\sigma(p)}{p_T}\right|^{MS} = \frac{\Delta p^{MS}}{\Delta p_T} = \frac{0.0136\sqrt{\frac{L}{X_0}}}{0.3BL} = 0.045\frac{1}{B\sqrt{LX_0}}$$

independent of p !

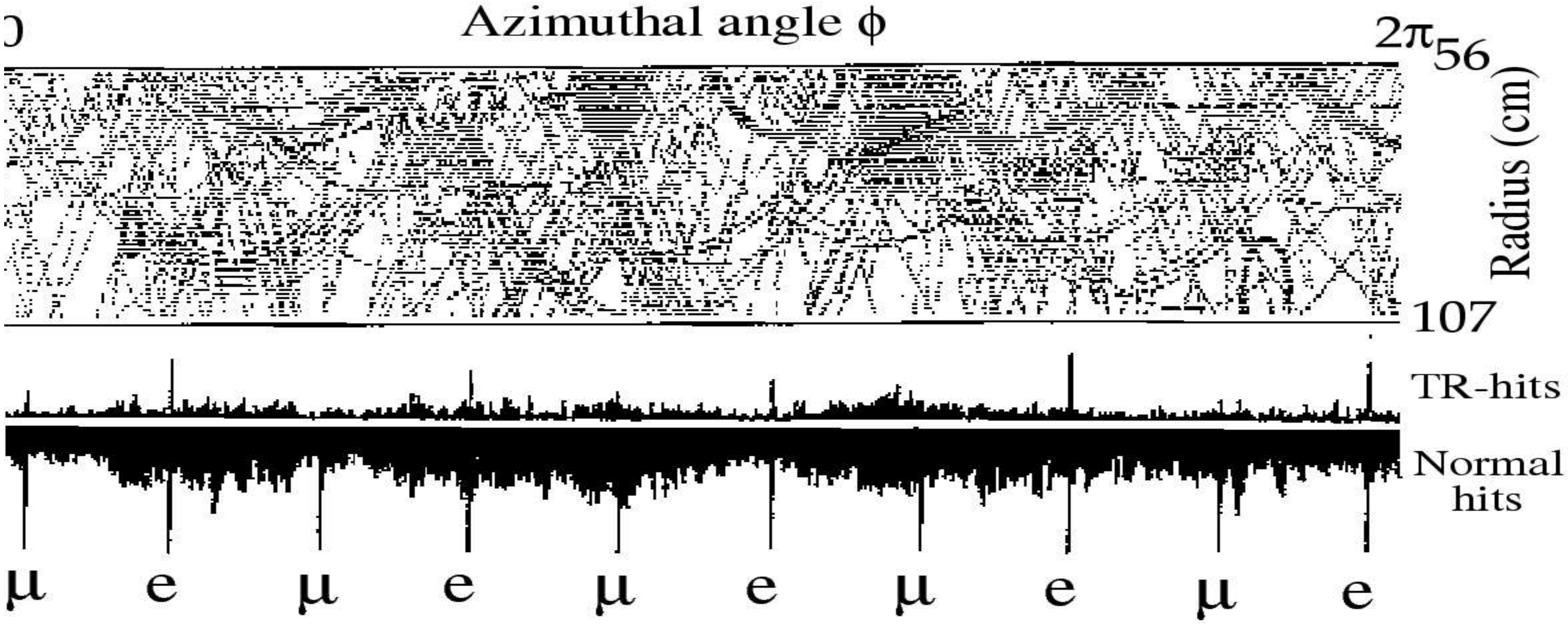


Pattern recognition

- How to associate a subset of the hits to some track?
- *Predefined templates*, i.e. patterns of fired cells defining an allowed track. Used in fast trigger algorithms.
- The *cell tower* is an example from the calorimeter world.
- The *associative memory* where each hit is seen by all possible templates is the most advanced example.
- The *Hough transform* is another method. For straight tracks in two dimensions, each hit corresponds to a straight line in the slope-intercept plane.
- Peaks in this plane where many lines intercept reveal the hits-on-tracks. This is also relatively fast.

Pattern recognition (Hough)

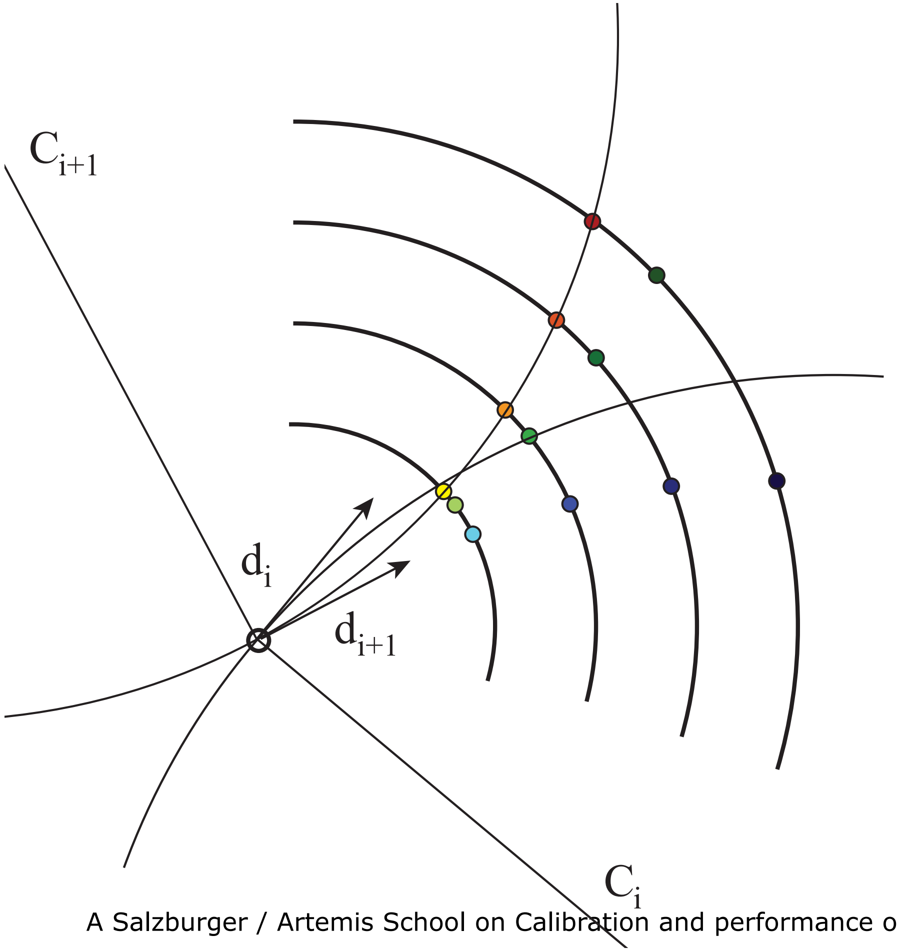
- Such histogramming methods may provide fast seeds for high momentum tracks – here an example:

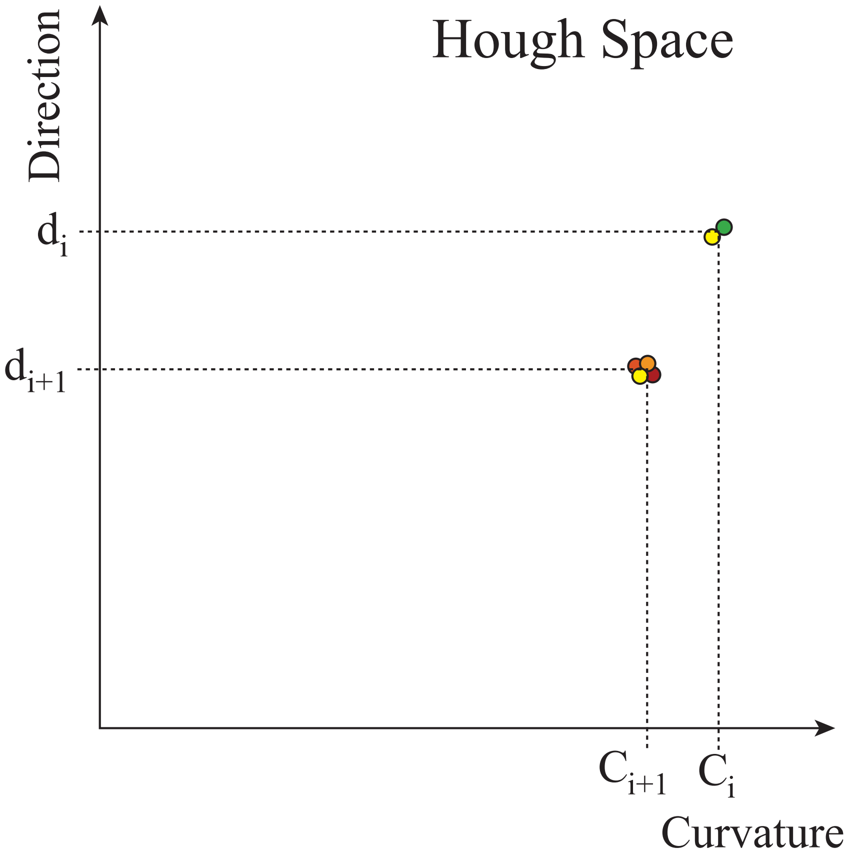


ATLAS TRT

General Hough transform

- Scan in two dimensions (an angle and a curvature)
- Find maxima





The state vector

- Let at each detector surface the track be given by a vector (position, direction, 1/p), along with its uncertainties:

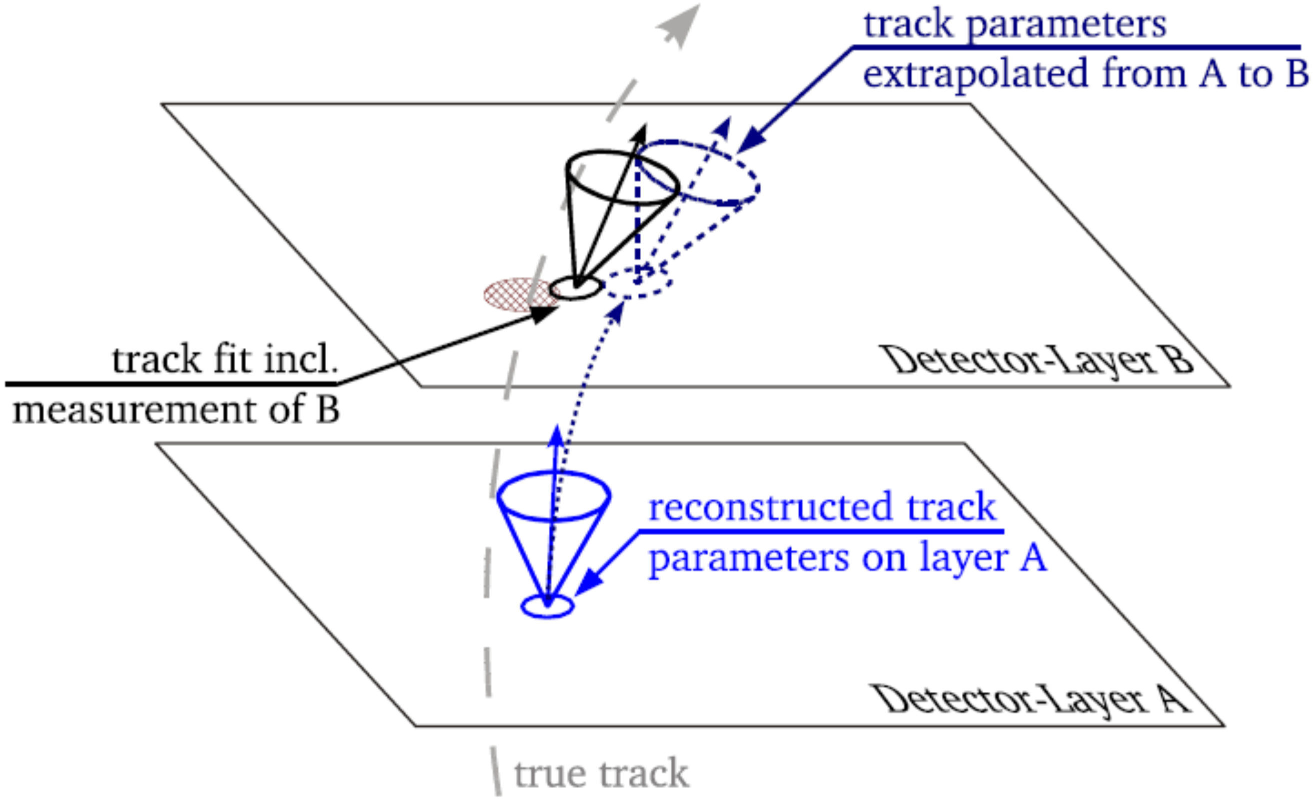


Figure from ATLAS reconstruction group

The helix

- An example of a *state vector* of *track parameters* is
- *The helix*, where 90º-λ is the track angle to the B field, R is the radius, s the path length and h is a sign

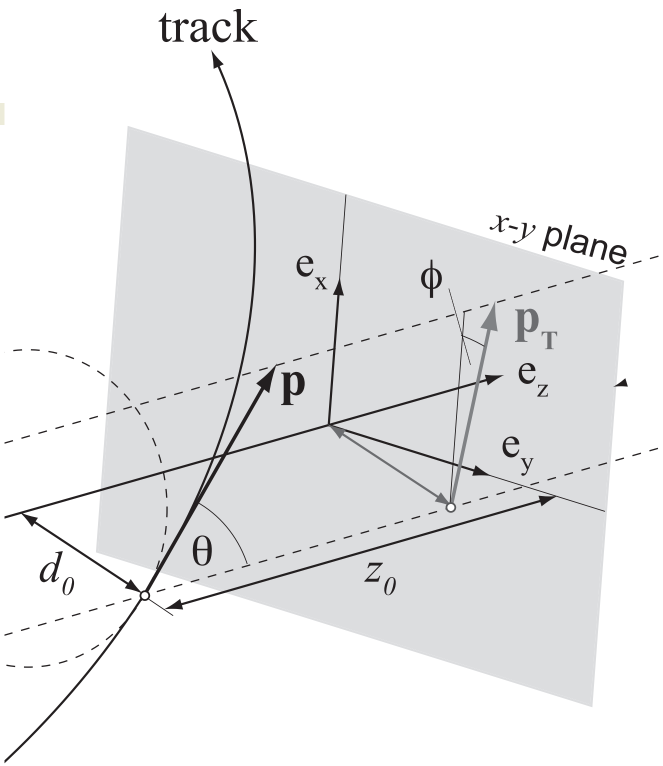
$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x^0 + R \cdot (\cos(\alpha_0 + \frac{hs}{R} \cdot \cos\lambda) - \cos\alpha_0) \\ y^0 + R \cdot (\sin(\alpha_0 + \frac{hs}{R} \cdot \cos\lambda) - \sin\alpha_0) \\ z^0 + s \cdot \sin\lambda \\ \end{pmatrix}$$

- A track in a detector with cylinder symmetry is a collection of helices at each “cylinder surface”.

Perigee parameters

- The perigee parameters

$$\bar{x} = (d_0, z_0, \phi_0, \theta, \frac{q}{p})$$



are often used to describe the track state at the closest point to the beam (z) axis.

- Use q/p instead of qp, because q/p is generally measured with a gaussian uncertainty.
- d0 has some sign convention according to the angular momentum of the track wrt the z axis

The projection matrix H

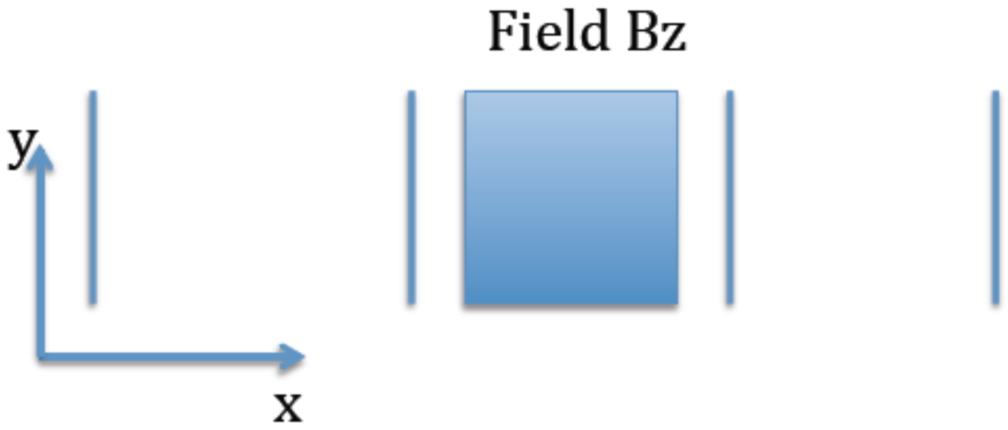
- In order to compare with measurements $\underline{m}$, the track state $\underline{x}$ needs to be projected onto the “measurement frame”. This is accomplished by the matrix H=δm/δx (assuming for simplicity that $\underline{x}$=0 corresponds to $\underline{m}$=0)
- Consider, for example, a set of strips forming a small angle α with the x axis. Let the track parameters be x and y at each plane of strips. Then we must do

$$H\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -\sin\alpha & \cos\alpha \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

in order to arrive at the y’ -coordinate perpendicular to the tilted strip. This y’ measures the hit strip number which is one of the raw measurements $\underline{m}$.

Spectrometer example

4 y-z pixel planes   distance between planes d



$$\tan(\Delta\phi) = 0.3 \times \int B_z \, dx \times q / p = b(q / p)$$

Using units of Tesla,m, and GeV/c

Spectrometer example

Projection matrix (linearized)

measurements

State vector at plane 1

$$m = \begin{bmatrix} z_1 \\ \dots \\ z_4 \\ y_1 \\ \dots \\ y_4 \end{bmatrix}, \qquad x = \begin{bmatrix} z \\ z' \\ y \\ y' \\ q/p \end{bmatrix}, \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & d & 0 & 0 & 0 \\ 1 & 2d & 0 & 0 & 0 \\ 1 & 3d & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d & 0 \\ 0 & 0 & 1 & 2d & bd/2 \\ 0 & 0 & 1 & 3d & 3bd/2 \end{bmatrix}$$

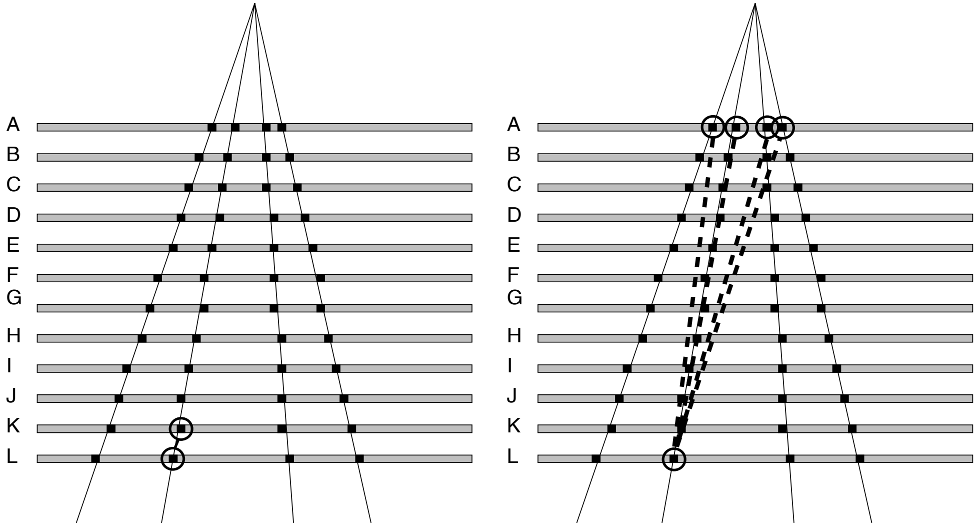
The Kalman filter

- Determines the track state vector dynamically from measurements at each detector surface.
- These are either discarded or used to update the existing state vector.
- Needs only inversion of small matrices. Fast.
- Can account for noise, multiple scattering and energy loss at each surface. Efficient.
- Is equivalent to the least squares fit, but provides pattern recognition integrated in the fit.

Seeding

- We need TRACK SEEDs with a reasonable success rate. Different strategies are used, depending on both geometry and physics:

Figure from Mankel arXiv:0402039v1



- For each seed we may let the filter proceed

The propagator F

- Let the track transport from layer k-1 to k be given by

$$x_k = f(x_{k-1})$$

- This is then the predicted state (denoted by a tilde). If f is not already linear in x, we can try to use its derivatives to get a linear equation:

$$\tilde{x}_k = F_k x_{k-1},$$

$$C_k^{k-1} = F_k C_{k-1} F_k^T + Q_k$$

where C_k is the covariance matrix for the predicted state and Q contains the additional random perturbations in the step, such as multiple scattering and energy loss.

Covariance matrices V and C

- For any random variables x_i and x_j, the covariance matrix:

$$V_{ij} = E((x_i - E(x_i)) \times (x_j - E(x_j))$$

is symmetric and have diagonal elements equal to the variances of the x'es.

and off-diagonal elements describing the degree of correlation between x_i and x_j.

- Any set of functions f_i of the x's has (to first order in a Taylor expansion) the covariance matrix:

$$C_{ij}^{f} = \sum_{kl} \frac{\partial f_i}{\partial x_k} \frac{\partial f_j}{\partial x_l} V_{kl}$$

- This is the chain rule.

The propagator F – simple example

- The F propagator is exactly the same as the transfer matrix of accelerator physics.
- For our example spectrometer we have in the z projection for propagation from the second to the third plane

$$\tilde{x}_3 = F_3 x_2 = \begin{bmatrix} 1 & d \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z_2 \\ z'_2 \end{bmatrix},$$

$$C_3^2 = F_3 C_2 F_3^T + Q.$$

$$C_2 = \begin{bmatrix} \sigma_m^2 & \sigma_m^2 / d \\ \sigma_m^2 / d & 2\sigma_m^2 / d^2 \end{bmatrix}, \qquad Q = \begin{bmatrix} \theta_{MS}^2 d^2 & \theta_{MS}^2 d \\ \theta_{MS}^2 d & \theta_{MS}^2 \end{bmatrix}$$

The propagator F – complex case

- In regions with an inhomogenous B field, the preferred method is Runge-Kutta integration. Here the trajectory derivatives are sampled at a number of intermediate positions, weighted so that the error is 5th power in h, the small time-step to the next plane:

$$y' = f(t, y), \qquad y_0 = y_0(t_0)$$

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$k_1 = f(t_n, y_n)$$

$$k_2 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_1)$$

$$k_3 = f(t_n + \frac{h}{2}, y_n + \frac{h}{2}k_2)$$

$$k_4 = f(t_n + h, y_n + hk_3)$$

The propagator F – complex case

- Let us try a grossly nonlinear case and see how it goes:

$$y' = y, \qquad y_0 = 1$$

$$k_1 = 1$$

$$k_2 = 1 + \frac{h}{2}$$

$$k_3 = 1 + \frac{h}{2}(1 + \frac{h}{2})$$

$$k_4 = 1 + h + \frac{h^2}{2}(1 + \frac{h}{2})$$

$$y_1 = 1 + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) = 1 + h + \frac{h^2}{2} + \frac{h^3}{6} + \frac{h^4}{24}$$

- Victory!

The residual r

- The difference between a measurement m and its prediction by the track state, Hx, is called the residual:

$$r_k^{k-1} = m_k - H_k \tilde{x}_k, \;\; R_k^{k-1} = V_k + H_k C_k^{k-1} H_k^T$$

V is the *covariance matrix* of the measurements

R is the *covariance matrix* of the residuals.

At this point you can reject a measurement m_k on the basis of r_k^2/R_{kk} . This is the *pattern recognition*.

(Note that the contribution from the track is here *added* to the measurement variance. The measurement is not used yet. If the hit contributes to the track, the track variance is instead *subtracted* from the residual variance).

The gain matrix K

- The matrix transforming the residual into a correction to the track state is:

$$K_k = C_k^{k-1} H^T (V_k + H_k C_k^{k-1} H_k^T)^{-1}$$

$$x_k = \tilde{x}_k + K_k r_k^{k-1}$$

$$C_k = (1 - K_k H_k) C_k^{k-1}$$

where x_k is now the updated (filtered) state.

Try it with a one-dimensional update of weighted averages

For a derivation: R.Mankel, Prep. Prog.Phys. &7 (2004) 553

An easier alternative:

- A completely equivalent update algorithm is to take a weighted average of the predicted track state and the state suggested by the new measurement:

$$x_k = C_k((C_k^{k-1})^{-1} x_k^{k-1} + H^T V_k^{-1} m_k)$$

$$C_k^{-1} = (C_k^{k-1})^{-1} + H^T V_k^{-1} H$$

Filtered residuals

- The filtered residual, its covariance and chi2 are

$$r_k = (1 - H_k K_k) r_k^{k-1},\ R_k = (1 - H_k K_k) V_k,\ \chi^2 = r^T R^{-1} r$$

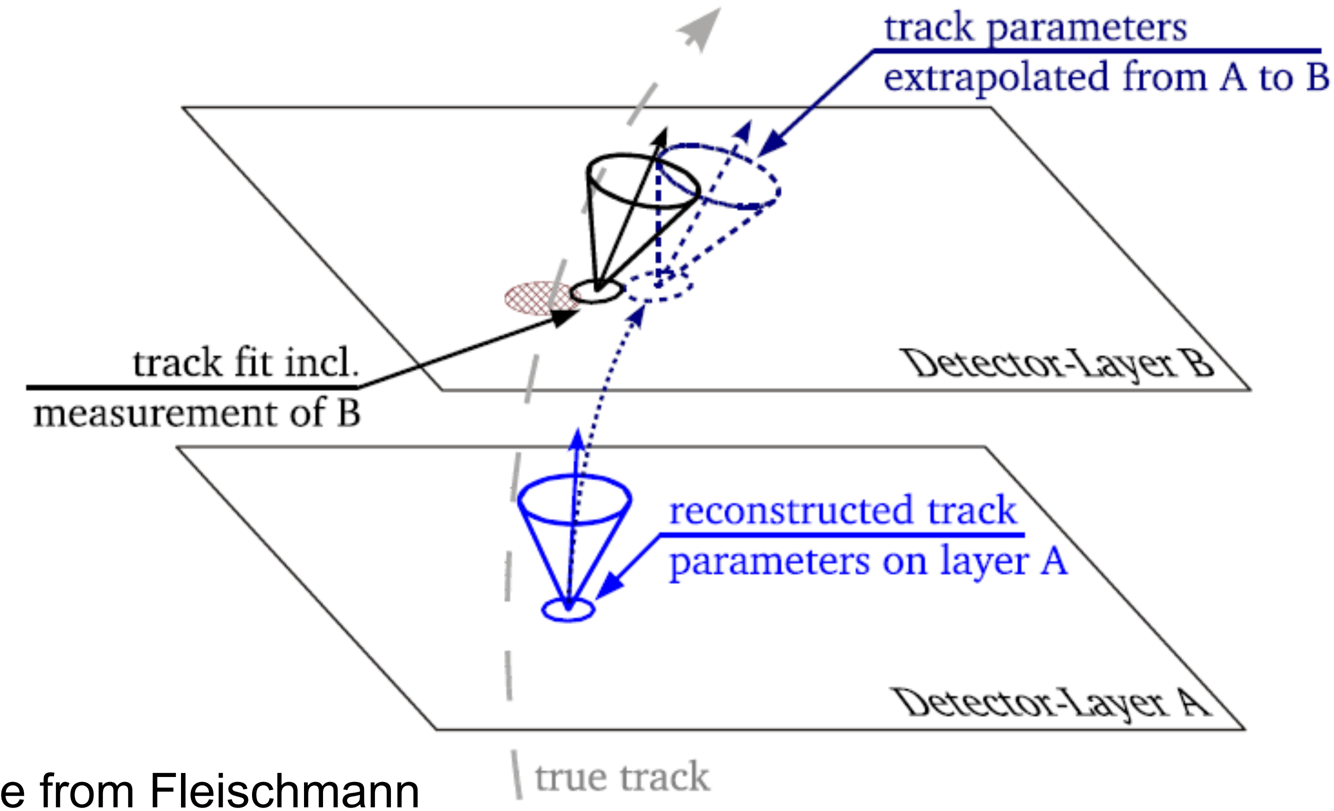


Figure from Fleischmann

Smoothing

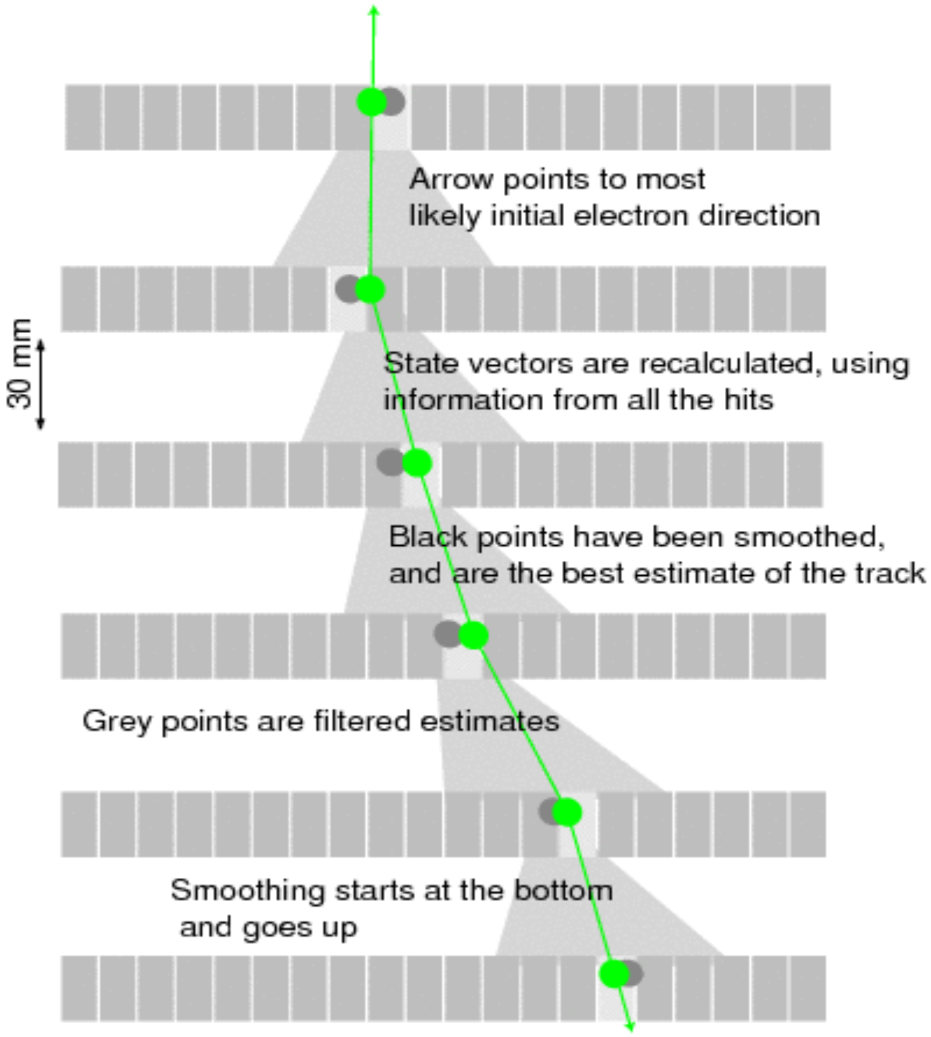
- We have reached the end with n hits. Now the procedure is repeated backwards. This is used to update the state at each k with the information from all the others:

$$C_{k|n}{}^{-1} = C_{k|k}{}^{-1} + (C^{b}_{k|k+1})^{-1}$$

$$x_{k|n} = C_{k|n}(C_{k|k}{}^{-1}x_{k|k} + (C^{b}_{k|k+1})^{-1}x_{k|k+1})$$

If the measurement m at plane k is more than a few sigma from updated x , it may be ignored in the fit. Ignored measurements may, however, be interesting later.

- Finally the innermost surface is extrapolated to the perigee



Arrow points to most likely initial electron direction

State vectors are recalculated, using information from all the hits

Black points have been smoothed, and are the best estimate of the track

Grey points are filtered estimates

Smoothing starts at the bottom and goes up

Fig. 2.— The Kalman smoothing process.

From the GLAST Science Prototype

Global (Newton-Raphson) fits

- As an alternative, the global least-squares fit assumes we know in advance which hits belong to the track.
- It minimizes the weighted sum of distances between the fitted track and the assigned hits, adjusting the track states at each surface.
- It is mathematically equivalent to the Kalman filter-smoother for a fixed selection of hits on a track.

Global fits

- In the approximation where the expected measurements are linear in the track parameters x, we minimize

$$\chi^2 = (\overline{m} - H\overline{x})^T V^{-1} (\overline{m} - H\overline{x})$$

where m is a vector of measurements at *all the surfaces.*

- The parameters minimizing this is

$$\overline{x} = (H^T V^{-1} H)^{-1} H^T V^{-1} \overline{m}$$

For normally distributed m, this is also the maximum likelihood estimate of the parameters.

The factor

$$(H^T V^{-1} H)^{-1} = (\frac{1}{2}\frac{\delta^2 \chi^2}{\delta \overline{x}^2})^{-1}$$

is the covariance matrix of the parameters.

Newton-Raphson fit

- If the projection h(x) is not linear, we can Taylor expand around an initial value x0 obtaining approximately:

$$\frac{d\ \chi^2}{dx}(x_0) = 2H^T V^{-1}(m - h(x_0))$$

$$\frac{d^2\chi^2}{dx^2}(x_0) = 2H^T V^{-1} H = Cov^{-1}(x_0)$$

we insert now instead $x_1 = x_0 - (\frac{d^2\chi^2}{dx^2})^{-1}\frac{d\chi^2}{dx}$

x_1 may not be exactly the minimizing value – but it is, after all, better than x_0. Thus we iterate until convergence

(|x1 -x0|<small number).

Dealing with multiple scattering

- The global chi-squared track fit can allow at each scattering plane a MS angle treated as an extra track parameter with a contribution to chisquared of $\left(\theta/\theta_0\right)^2$
- Alternatively we can introduce correlations between surfaces in the covariance matrix V. This is what we do in the exercise.
- The Kalman filter also does this at each scattering plane j:

$$V \rightarrow V + S\Theta S^T \qquad S = \partial r/d\theta \qquad \Theta_{jj} = \theta_{0j}^2$$

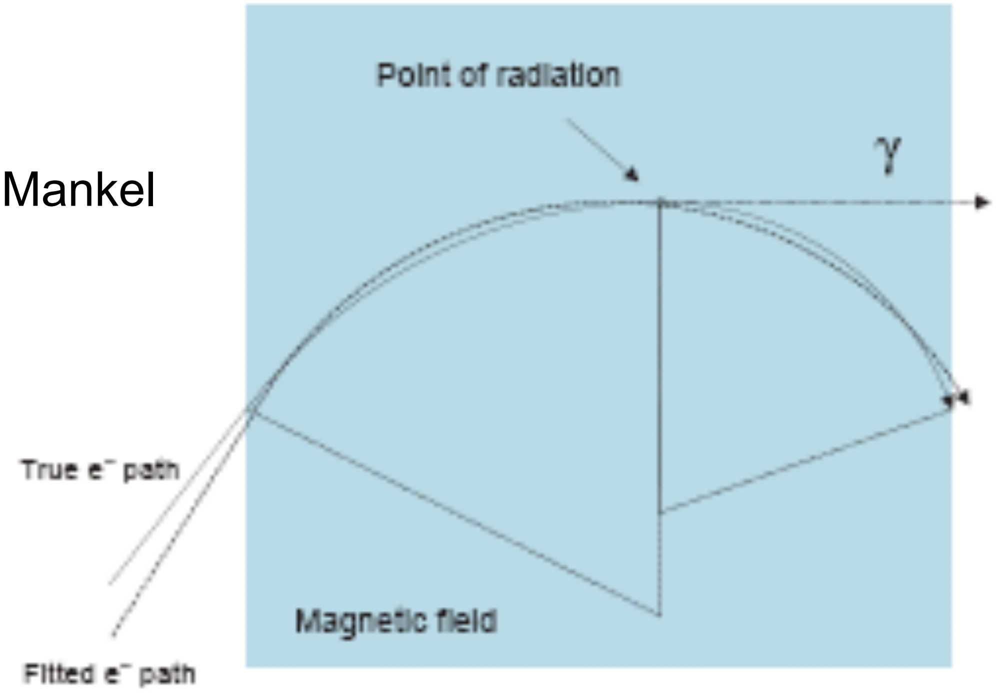
- MS is approximately Gaussian. Luckily, because the Global Chisquared and the Kalman Filter work efficiently only with Gaussian deviations from expectations.

Dealing with Non Gaussian errors

- Special methods are needed to take care of *non-gaussian* influences, such as hard photon radiation where the probability density for the electron to retain a fraction z of its energy follows the Bethe-Heitler law

$$f(z) = (-\ln z)^{c-1} / \Gamma(c) \qquad c = X_{rl} / \ln 2$$

Figure from Mankel



Gaussian Sum Filter

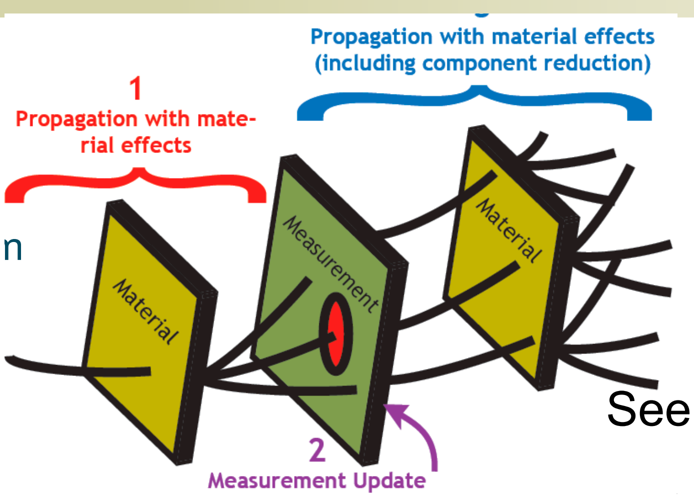

Propagation with material effects
(including component reduction)
1
Propagation with material effects
n
Material
Measurement
Material
2
Measurement Update

Figure from Salzburger lectures

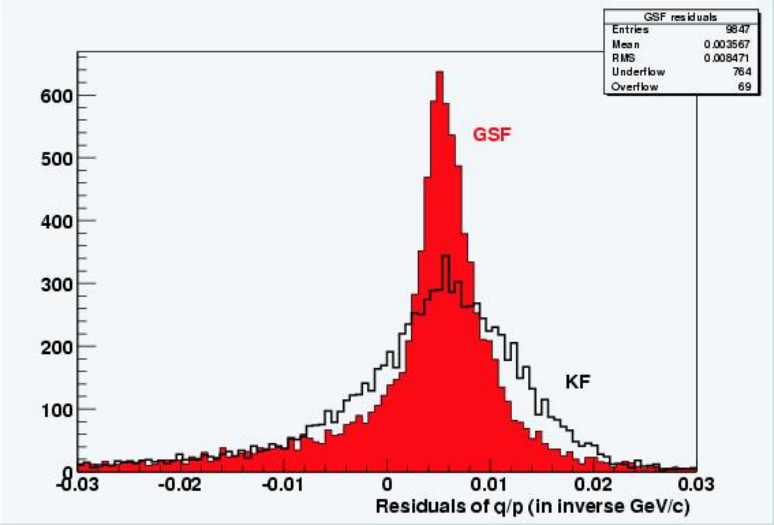
See eg R. Frühwirth and S. Frühwirth-Schnatter, 1998

- Branch the Kalman filter at each surface into parallel paths using a finite number of different Gaussian errors.
- This is the same as modelling e.g. the Bethe-Heitler as a sum of Gaussians

$$f(z) = \sum_{i}^{Nmax} g_i \varphi(z; \mu_i, \sigma_i)$$

where the weights g_i, the average and variance of the energy-loss are determined beforehand from fits to simulation.

Gaussian Sum Filter



From A. Strandlie, CMS simulation 2003

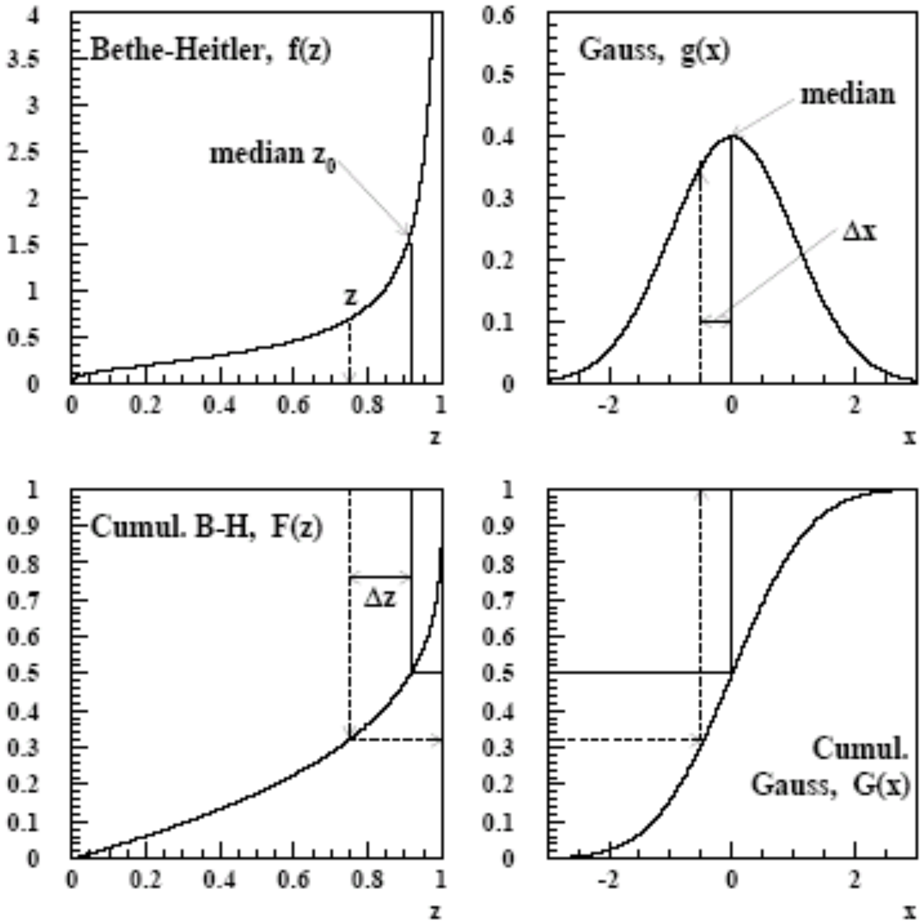
- Effectively the track state branches out into a number of possibilities at each plane.
- Some method, called *component reduction*, must be chosen to keep the number of branches from exploding.
- The resulting algorithm is very efficient in recovering from hard bremsstrahlung, but is also very CPU consuming. Normally only used for good electron candidates.

Dynamic Noise Adjustment

- DNA offers an alternative recovery from bremsstrahlung.
- First, the z retained after a particular surface is estimated using the hits in a few following planes.
- Then an adjustable noise level σ(z) is calculated so that the Bethe-Heitler probability of a deviation from median z to the found one equals that for z=z(median)+xσ(z), where x is drawn from a unit Gaussian.
- This way the Bethe-Heitler is mapped onto a Gaussian

Dynamic Noise Adjustment

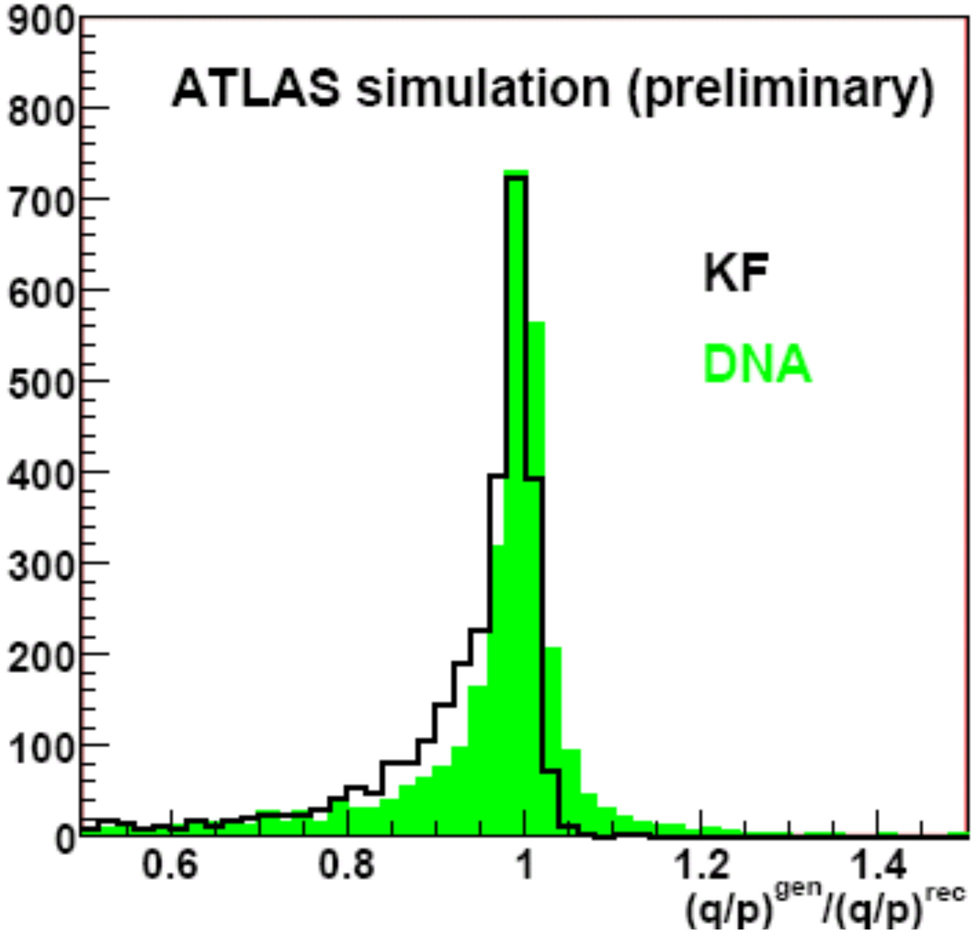
- The dynamically adjusted $\sigma(z)=\Delta z/\Delta x$ noise term is fed back to the Kalman Filter covariance matrix just like for multiple scattering.



from Kartvilishvili

DNA filter

- Using the DNA filter instead of just a fixed noise of width $<(z-z_{median})^2>$ really helps for electrons, because z is not Gaussian:



From Kartvilishvili

Exercise 1

Break to do exercise 1 in

http://www.nbi.dk/~phansen/nordforsk/Exercises.docx

Global optimization

- The problem arises of competing assignments of hits to the different track candidates or to noise. This can be tackled by minimizing a total energy function.
- In the Elastic Arms Algorithm a certain number of "deformable track templates" must first be found. These templates should also include a "noise template".
- A "metric" M_{ia} is typically defined as the chisquared for assigning hit i to track template a.
- One approach is to minimize the "total energy":

$$E = \sum_{a}^{Tracks} \sum_{i}^{Hits} \left[S_{ia} M_{ia} \right]$$

where the "assignment strength" S_{ia} is either 0 or 1.

Elastic arms and annealing

- You try then to optimize the S_{ia}'s, but typically the energy-landscape is very "spiky" with lots of local minima.
- This is tackled by annealing and fuzzy assignment strength:

$$S_{ia} = \frac{e^{-\beta M_{ia}}}{e^{-\beta\lambda} + \sum_{a=1}^{Tracks} e^{-\beta M_{ia}}}$$

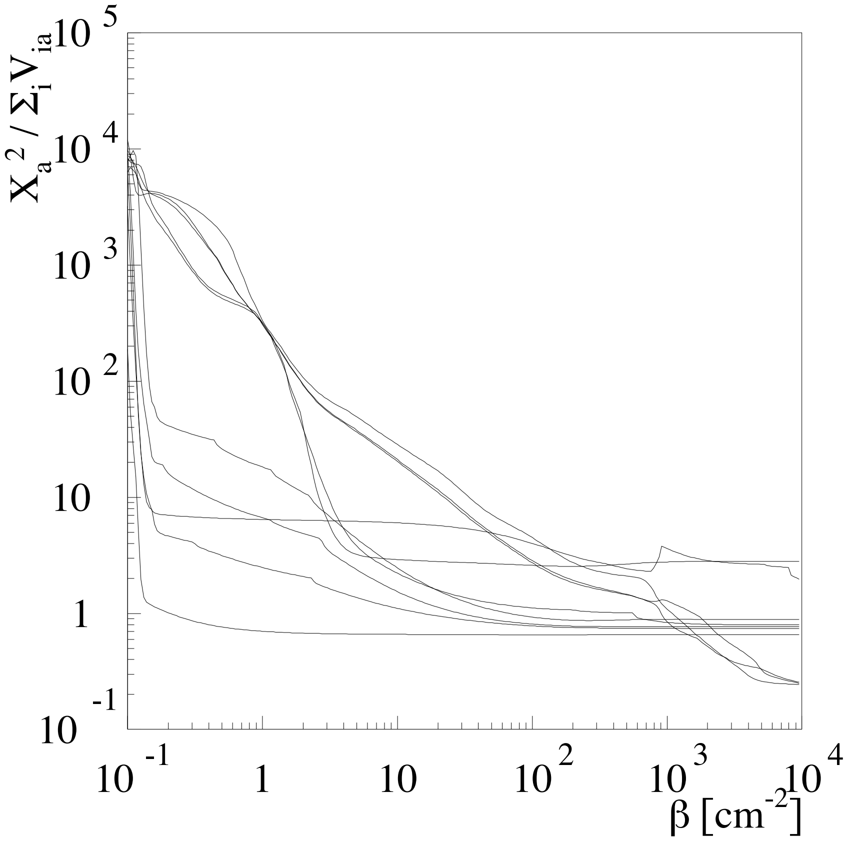
where β=1/T and λ is a "chisquared cut" of the order 10.

S_{i0} is here the assignment strength to noise ($M_{i0}=\lambda$)

- We now start at a high "temperature" where the S_{ia}'s are relatively large, even for distant hits. Few local minima.
- The track parameters are then iterated to the global minimum of E using the derivatives a la Newton-Raphson

Elastic arms and annealing

- We then lower the temperature (by eg 5%), repeat and continue until T<<1.
- At this point, the S_{ia}'s take ~discreet values of 0 and 1.



Chisquared for 10 muon tracks in HERA-B with decreasing temperature. (From Borgmeier Diploma Thesis 1996)

More is found in R.Mankels review (arXiv:040239v1) from 2004.

Deterministic Annealing Filter

- A problem with a *global method* like Elastic Arms is that the approximate number of tracks must be known beforehand. It gives you only better hit sharing.

  Therefore Frühwirth and Strandlie proposed to modify the (*local*) Kalman filter using an assignment probability S_{ik}

  for assigning hit i in plane k to the current track, thus keeping several options open for the propagation of a given seed.
- Investigated for ATLAS in S. Fleischmanns thesis

DAF idea

- The assignment probability for each of nk measurements in layer k to the current track is assumed to be proportional to a multivariate Gaussian:

$$\phi_k^i = \phi(m_k^i ; H_k x_k , TV_k^i + H_k C_k^* H_k^T)$$

σ

where x here is the smoothed track state without involving layer k in the fit and T a temperature parameter. The last term is the “track contribution” to the residual error, which can often be ignored.

DAF details

- Allowing for the hypothesis that no hit is produced by the track in layer k, we normalise the assignment probability as:

$$S_i^k = \frac{\phi_k^i}{\sum_j^{n_k} (\Lambda_k^j + \phi_k^j)}$$

- The cut term is parametrised

$$\Lambda_k^j = \frac{1}{(2\pi)^{\dim(m)} \sqrt{T \det V_k^i}} \exp(-\frac{\lambda}{2T})$$

where λ acts as a χ^2cut-off at low temperature (T=1).

(Frühwirth and Strandlie, Comp.Phys.Comm,133(2000)34)

DAF overview

- Uses the same principles as the Kalman filter, but several measurements per detector layer are taken into account by using their weighted mean.

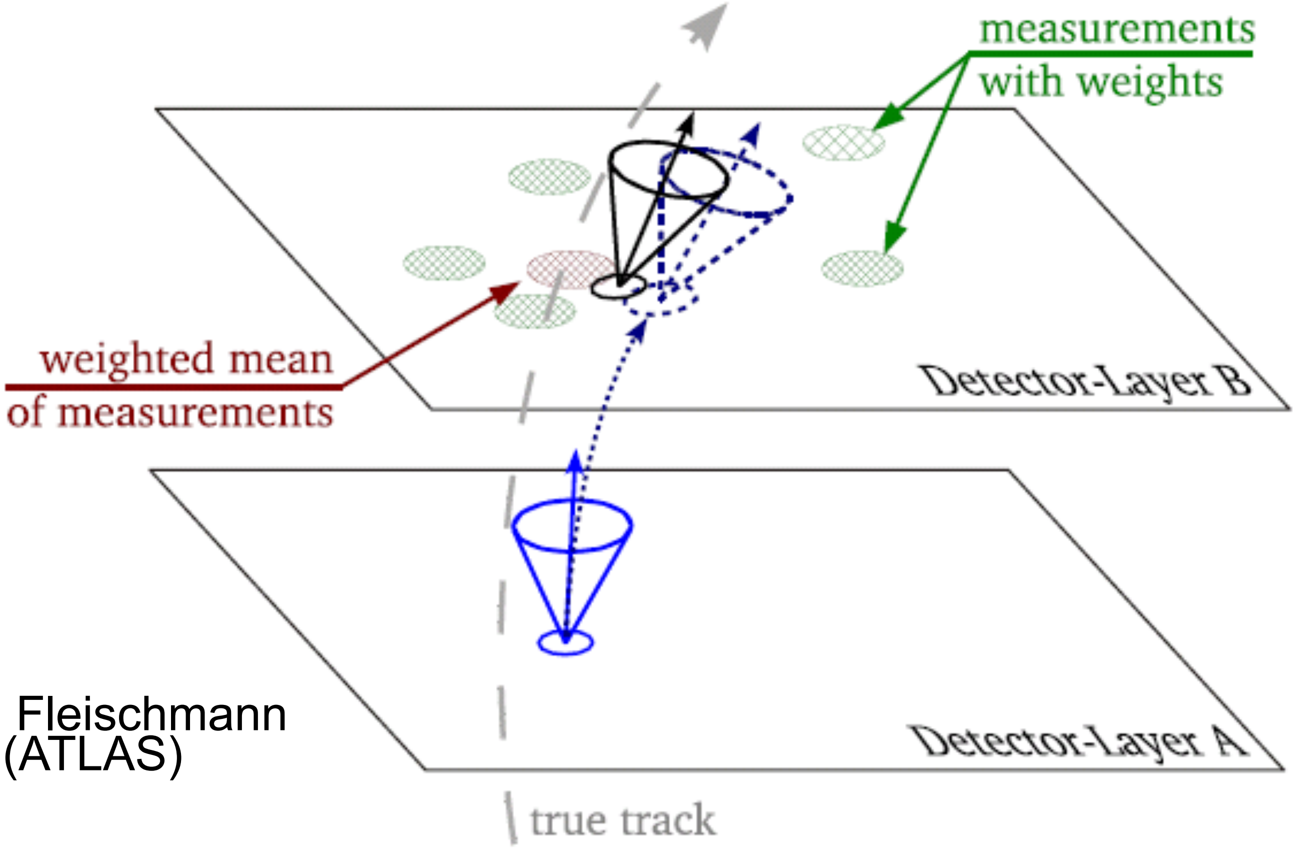


Figure from Fleischmann PhD thesis (ATLAS)

DAF in practice

- The Deterministic Annealing Filter has turned out especially effective in finding the best left-right choices in drift tubes.
- It can be used as an ”afterburner” to the Kalman Filter

  and may significantly improve momentum resolution.
- It can be extended it to a multi-track fitter with built-in pattern recognition.
- In this case the normalisation of assignment probabilities needs to be changed so that the sum runs over all accepted tracks competing for the measurements.
- As in the Elastic Arms a procedure starts at a high temperature and iterates with decreasing tolerance, but without working with a fixed number of tracks.

Track scoring

- We need a quality estimator (a “score”) used to reject or accept the track candidate.
- Several sub-estimators are used, typically
  - Number of precision hits
  - Number of outlier hits
  - Holes
  - Shared hits
  - Total chisquared per degree of freedom
  - Transverse momentum
- In a *second pass* the hits not yet assigned to a track may be considered with larger tolerances (for example a lower P_T cut).

Finding the primary vertex

- Typically a limited ”beam-spot” is given by the machine-parameters , possible pick-up electrodes or pre-processing.
- Hereafter, just two tracks suffices to provide an accurate seed for the vertex

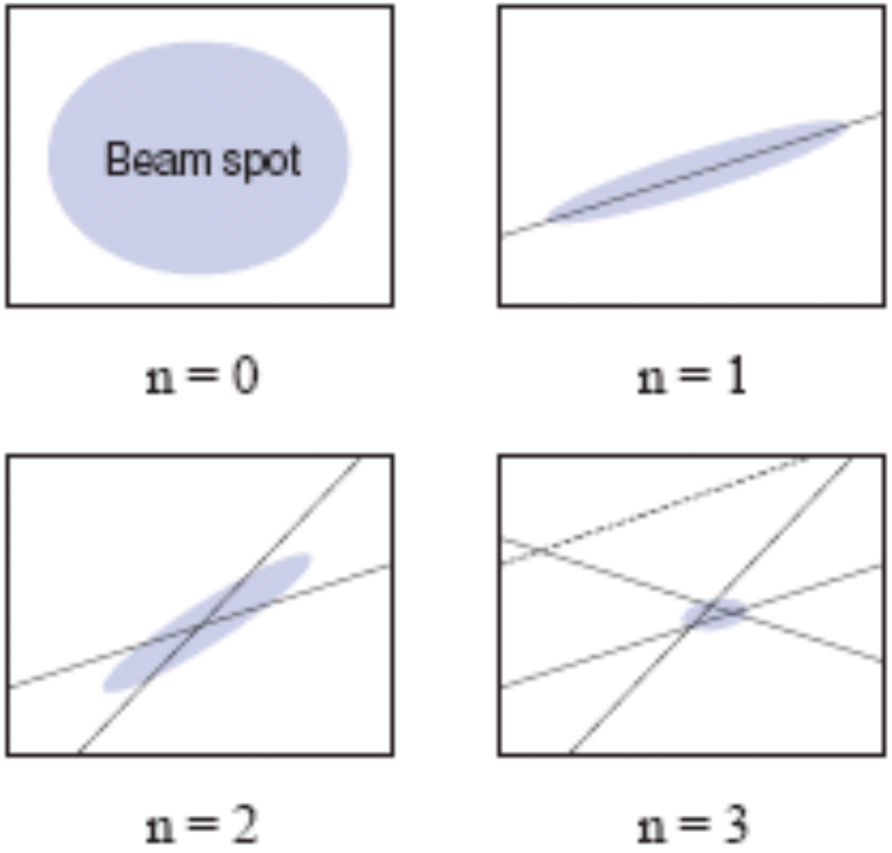


Figure from Mankel

Finding the first vertex seed

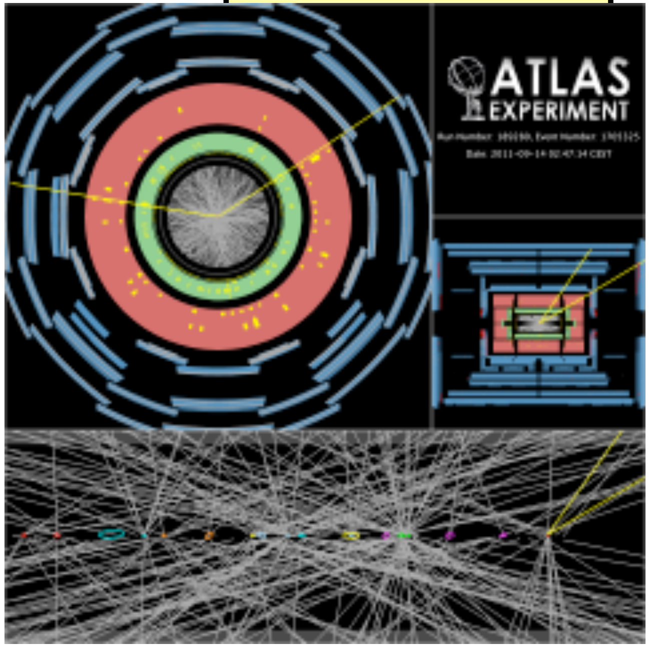
- The danger is of course that the initial seed is wrong, so great care must be taken in this very first step.
- In ZEUS, for example, all candidate track pairs were checked for compatibility with a common vertex on the proton beam-line. They were then ranked according to how many other pairs they agreed with. The best pair then started the chi-squared fit.
- In CMS, a similar procedure is used were the coordinates with the highest density of track crossings is found. In this evaluation, each track pair is weighted by a decreasing function of the distance between their two perigees.

ATLAS Multi Vertex Finder

INPUT

Tracks

Track selection



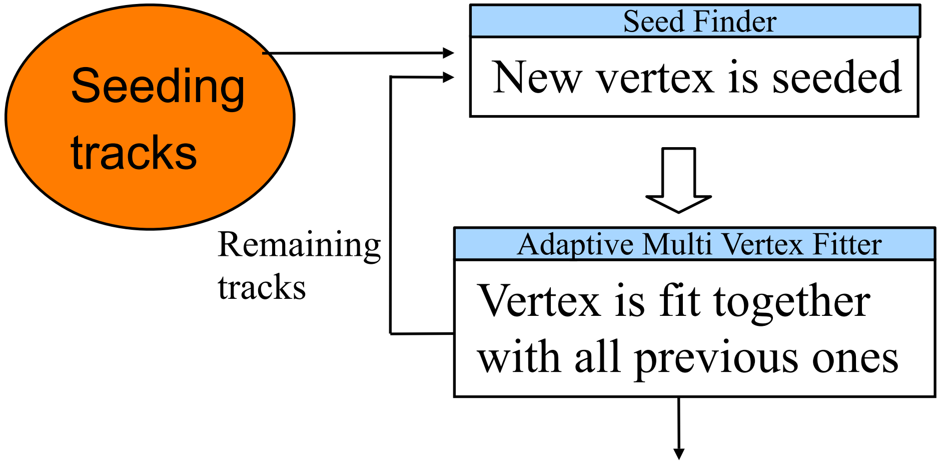

Seeding tracks
Seed Finder
New vertex is seeded
Adaptive Multi Vertex Fitter
Vertex is fit together with all previous ones
Remaining tracks

Figure from Saltzburger

- Several vertices fitted simultaneously (20 vertices above)
- Several iterations of Kalman filtering with decreasing tolerance for assigning a track to a vertex a la the DAF.

Billoir vertex fit

- Also here, the alternative to a Kalman Filter is the Newton-Raphson least squares fit (for vertices called a Billoir fit):

- Let $\overline{v} = (x_v, y_v, z_v)$ be the vertex position for n tracks
- Let $\overline{p}_i = (p_{xi}, p_{yi}, p_{zi})$ be the i'th track momentum
- Let $\overline{x}_i = F(\overline{v}, \overline{p}_i)$ be the 5 helix parameters of the i'th track
- To first order: $F = F(\overline{v}_0, \overline{p}_{0i}) + D_i \delta\overline{v} + E_i \delta\overline{p}_i$
- Let $\delta\overline{x}_i = \overline{x}_{i,meas} - F(\overline{v}_0, \overline{p}_{0i})$ and V_i be its covariance matrix.

- Then $\chi^2 \approx \sum (\delta\overline{x}_i - D_i\delta\overline{v} - E_i\delta\overline{p}_i)^T V_i (\delta\overline{x}_i - D_i\delta\overline{v} - E_i\delta\overline{p}_i)$

Billoir vertex fit

- Just like in the track fit (slide 44), we minimize chisquared at:

$$\overline{v} = \overline{v}_0 + (A - \sum B_i C_i^{-1} B_i^T)^{-1} (\overline{t} - \sum (B_i C_i^{-1})^T \overline{u}_i)$$

$$\overline{p}_i = \overline{p}_{0i} + C_i^{-1} (\overline{u}_i - B_i^T \delta \overline{v})$$

$$A = \sum D_i^T V_i D_i \qquad B_i = D_i^T V_i E_i \qquad C_i = E_i^T V_i E_i$$

$$\overline{t} = \sum D_i^T V_i \delta \overline{x}_i \qquad \overline{u}_i = E_i^T V_i \delta \overline{x}_i$$

- Where all the real work is in the calculation of derivatives D and E -and in the initial guess of v_0

Billoir vertex fit

- Now interchange $\bar{v}_0$ with $\bar{v}$ and continue until convergence
- The covariance of the fitted parameters is now

$$\operatorname{cov}(\bar{v}) = \left(A - \sum B_i C_i^{-1} B_i^T\right)$$

$$\operatorname{cov}(\bar{p}_i) = C_i^{-1} + (B_i C_i^{-1})^T \operatorname{cov}(\bar{v}) B_i C_i^{-1}$$

$$\operatorname{cov}(\bar{v}, \bar{p}_i) = -\operatorname{cov}(\bar{v}) D_i E_i^{-1}$$

- We also get correlations between the track momenta:

$$\operatorname{cov}(\bar{p}_i, \bar{p}_j) = \delta_{ij} E_j^{-1} - E_i^{-1} D_i^T \operatorname{cov}(\bar{v}, \bar{p}_j)$$

Billoir vertex fit - constraints

If you have some prior knowledge about the beam collision position b, just add an extra contribution to the chisquared which changes the derivatives D_i:

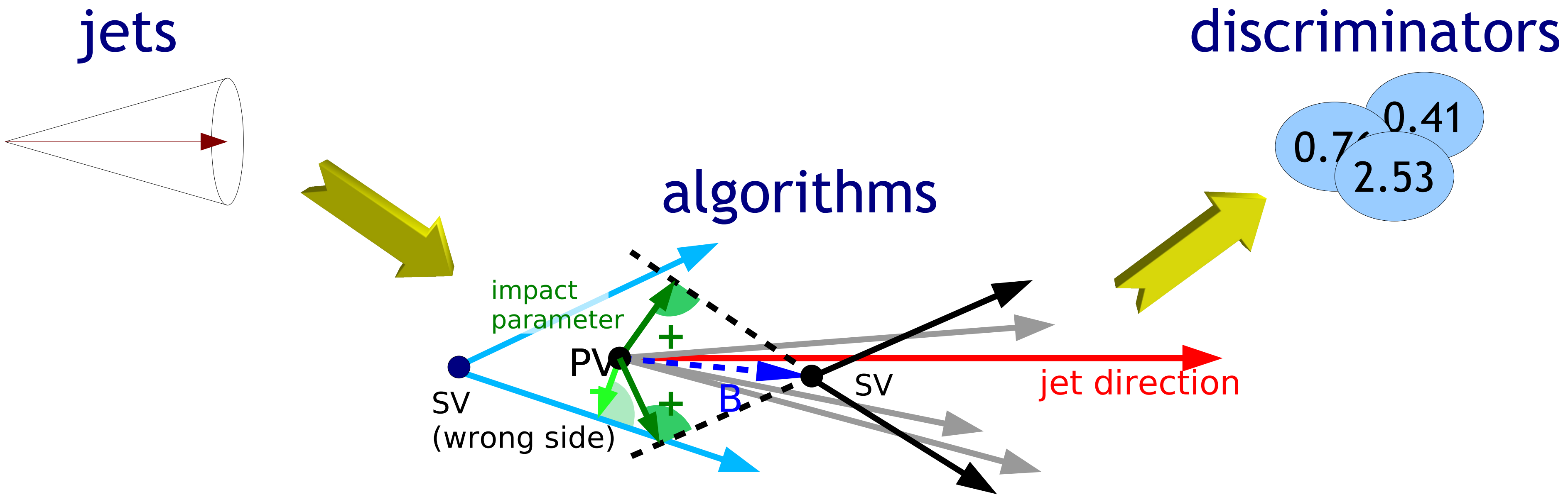
$$\delta\chi^2 = (\bar{v} - \bar{b})^T V_b (\bar{v} - \bar{b})$$

- If we deal with a secondary vertex and expect the sum of track momenta to point back to the primary vertex, then we can use the method of Lagrange Multipliers (more later):

$$\delta\chi^2 = -\bar{\lambda} \bullet \bar{d}(\bar{v}, \sum \bar{p}_i)$$

- Where λ are 3 new arbitrary fit parameters and d is the minimal vector distance between the primary vertex and the line pointed by the momentum sum.

B-jet tagging



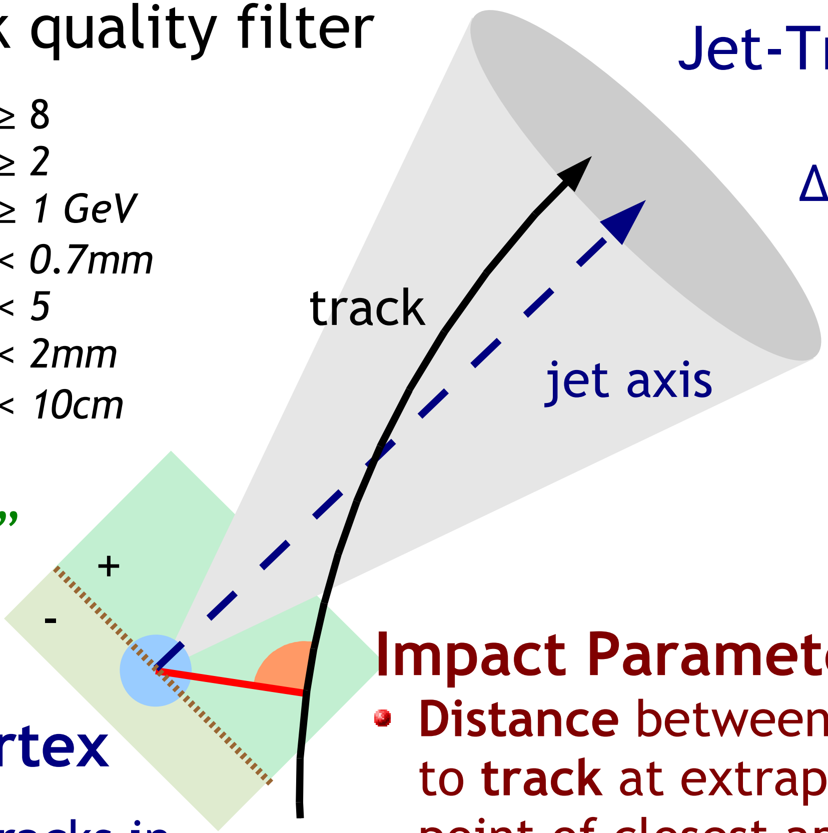
- Jets with a B-hadron can be identified by the lifetime (1.5 ps) and high mass of the b quark (about 4.2 GeV)

From the ATLAS B-physics group

Impact parameter (CMS)

Track quality filter

total tracker hits	≥ 8
pixel hits	≥ 2
P_T	≥ *1 GeV*
jet axis dist.	< *0.7mm*
$x^2/ndof$	< *5*
IP_{xy}	< *2mm*
decay length	< *10cm*



Jet-Track Association

ΔR_{max} to jet axis:
0.5 or 0.3

Primary Vertex

reconstructed using all tracks in event using the

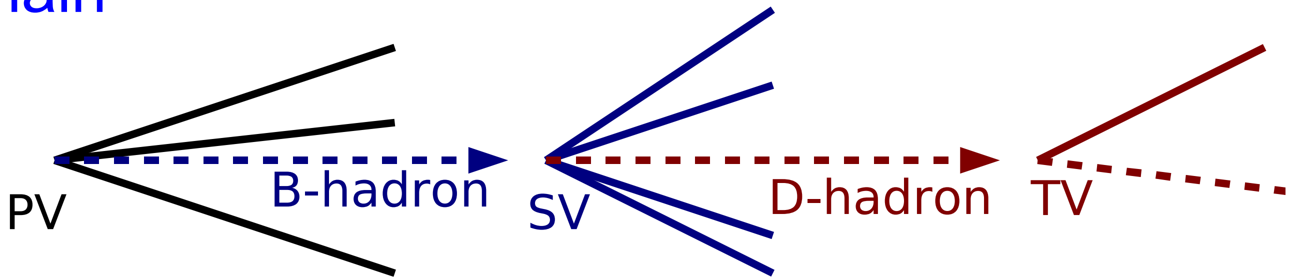
“Adaptive Vertex Fitter”:

Impact Parameter

- **Distance** between **Primary Vertex** to **track** at extrapolated point of closest approach
- **Signed**
- Transverse **r-φ** or full **3D** value
- **Significance:** distance / error

Secondary vertex finding

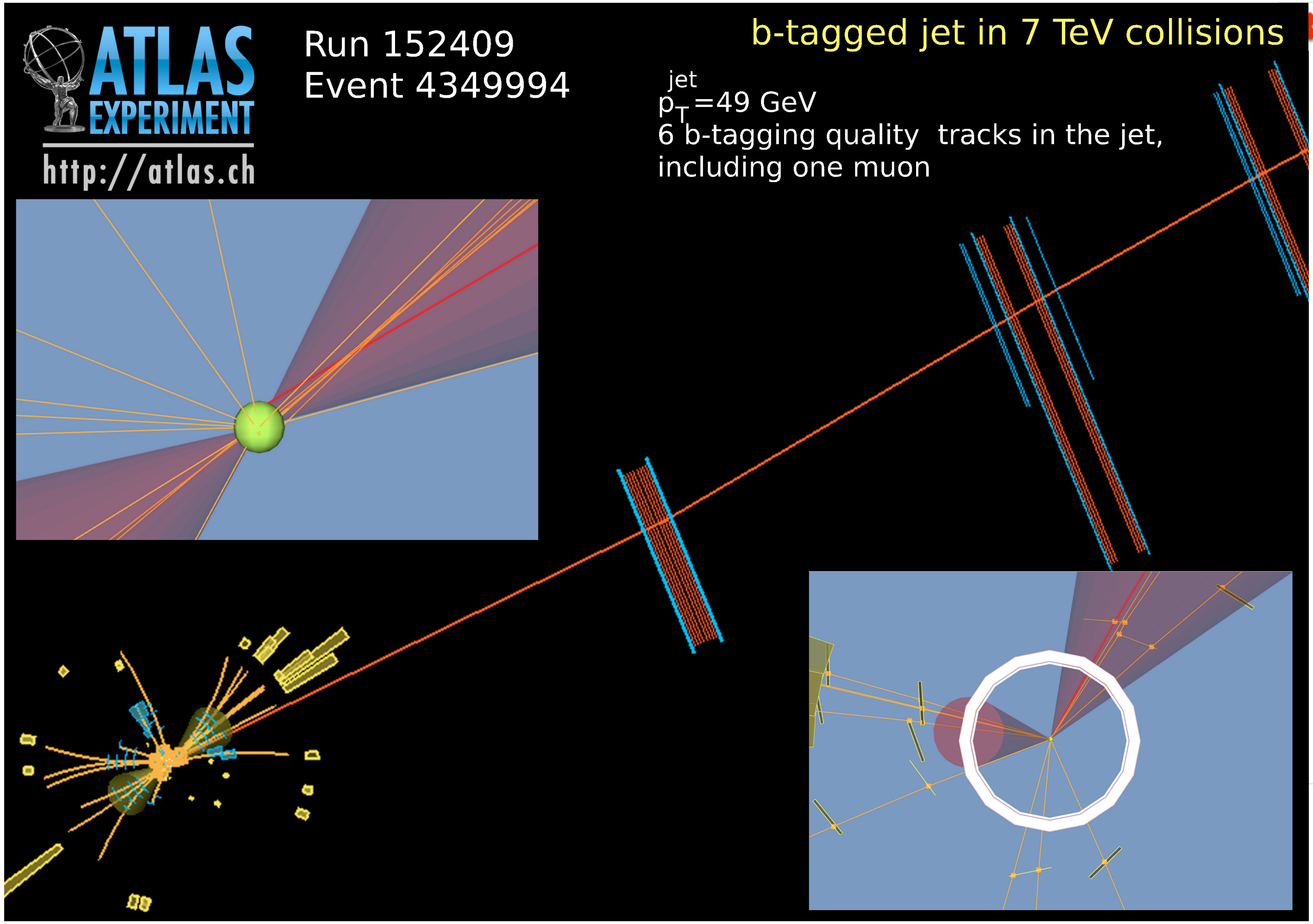
- Alternatively one can try directly to reconstruct the decay chain



- The number of found vertices along the jet-axis, their distances from PV, the mass and number of tracks at each vertex are all examples of variables with power to discriminate between b-quarks and lighter partons.

Soft lepton tag

- ❑ Due to the high B-meson mass, its leptonic decay (~10%) has a higher p_T^{rel} than leptons from light parton jets



Run 152409
Event 4349994

b-tagged jet in 7 TeV collisions

p_T^{jet}=49 GeV
6 b-tagging quality tracks in the jet, including one muon

The image cannot be displayed. Your computer may not have enough memory to open the image, or the i
may have been corrupted. Restart your computer, and then open the file again. If the red x still appears, y
have to delete the image and then insert it again.

Putting it all together

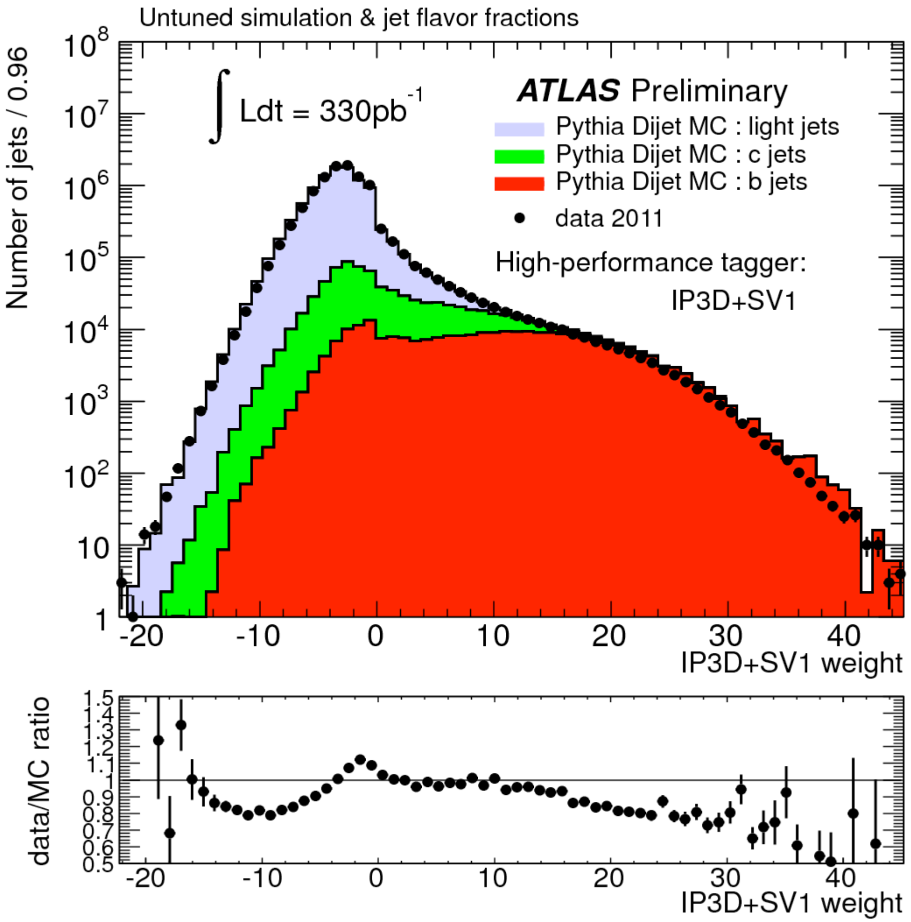
❑ Combine impact parameter significances for jet tracks

$$P(jet) = \Pi \cdot \sum_{i=0}^{N_{trk}} \frac{-\ln(\Pi)^i}{i!} \quad where \quad \Pi = \prod_{i \in jet} \int_{S_i}^{+\infty} f(S)dS$$

S is the impact parameter significance and f(S) its light jet probability.

The P(jet) estimator has many nice properties.

❑ Combine it with other input variables using either combined Likelihood or Multi-Variate Analysis



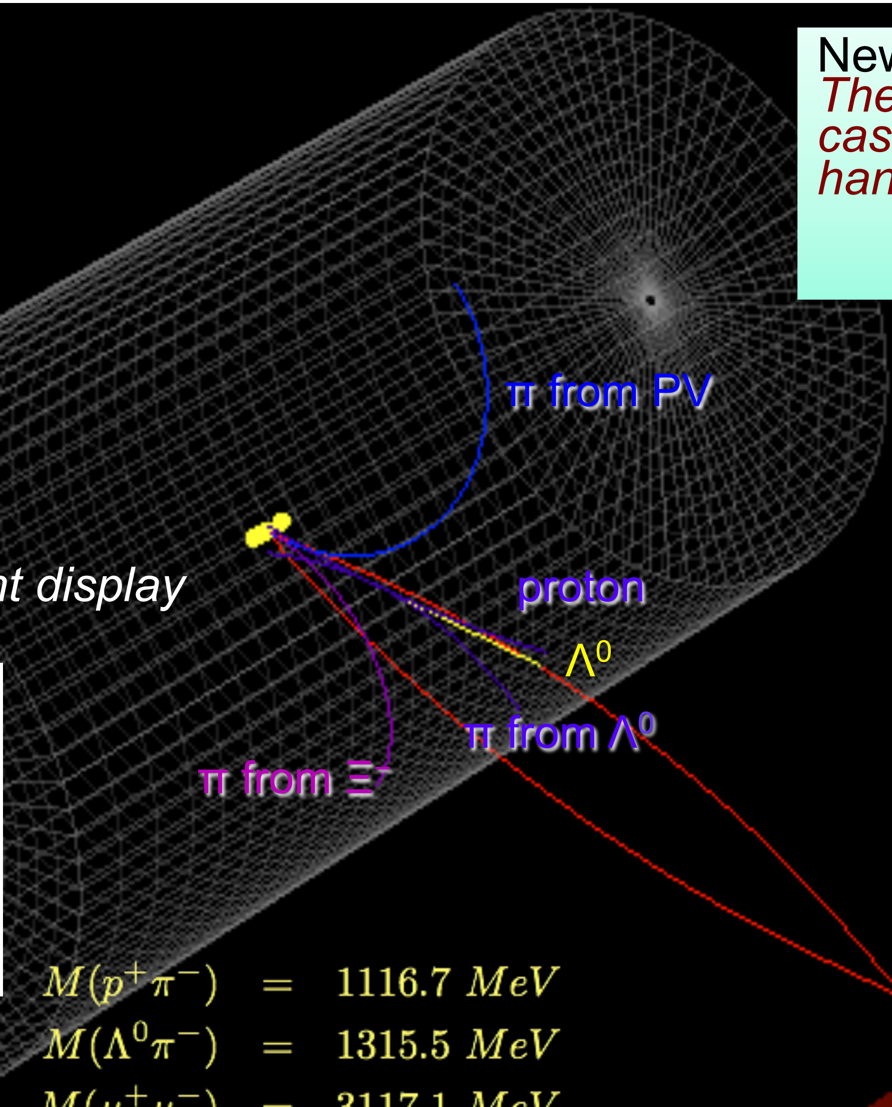
Constraints from priors

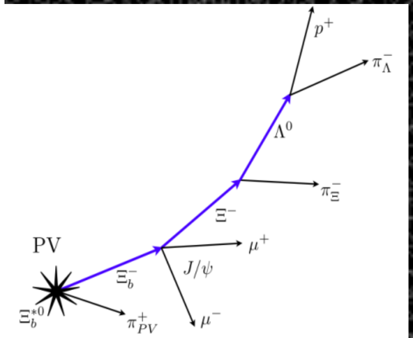
- In the further reconstruction of the event *a priori* knowledge can be used with advantage.
- An example is the beam energy constraint used in the reconstruction of tracks from the decay of $\Upsilon(4S)$ in b-physics experiments
- Another is the reconstruction of B meson cascade decay where the known masses of the D mesons are used as a constraint.
- Another is fitting electron-positron tracks from photon conversions. The knowledge of zero photon mass can be imposed by using Lagrange Multipliers.
- (a fantastic example on the next slide. For more see eg www.phy.ufl.edu/~avery/fitting/kinfit_talk1.doc)

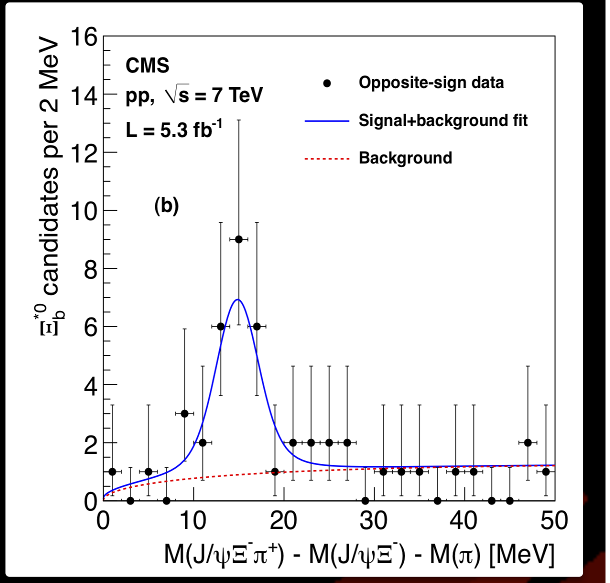
New ParticleDiscovery

The □$_b$^{*0} involves elegant cascade that CMS tracker handles beautifully.

Candidate event display







$$M(p^+\pi^-) = 1116.7\ MeV$$
$$M(\Lambda^0\pi^-) = 1315.5\ MeV$$
$$M(\mu^+\mu^-) = 3117.1\ MeV$$
$$M(J/\psi\Xi^-) = 5787.8\ MeV$$
$$Q(J/\psi\Xi^-\pi^+) = 15.7\ MeV$$

□$_b$^{*0}□□$_b$$^-$□$^+$□□$^-$J/□ □$^+$□L□$^-m^+m^-$□$^+$□p$^+$

Lagrange Multipliers

- Let again x be the track parameters of the two tracks that form a conversion candidate. The constraints must be expressed as some functions, H(x), being equal to zero.
- We again expand around an approximate solution xA:

$$\frac{\partial \overline{H}}{\partial \overline{x}}(\overline{x}-\overline{x}_A)+\overline{H}(\overline{x}_A)=\overline{D}\Delta\overline{x}+\overline{d}=\overline{0}$$

- If the two tracks should emerge parallel from a common point, the expression would be something like

$$\overline{d}=\overline{H}(\overline{p}_1,\overline{p}_2,\overline{r}_1,\overline{r}_2)_A=(\frac{\overline{p}_1}{E_1}-\frac{\overline{p}_2}{E_2},\overline{r}_1-\overline{r}_2)_A$$

- Where the p’s and r’s refer to the start points of the tracks.

Lagrange Multipliers

- The function to be minimized is now (dropping vector bars):

$$\chi^2 = (x - x_0)^T V_0^{-1} (x - x_0) + 2\lambda^T (D(x - x_A) + d)$$

- The minimum is found in the space of the track parameters x=(p,r) and the real constants λ.
- The ”0” refer to the unconstrained solution and the “A” to the previous iteration.
- The solutions have to be iterated since the constraint equations were linearized.

Solution to the constrained fit

- (put derivatives of chi2 to zero and solve by substitution):

$$x = x_0 - V_0 D^T \lambda$$

$$\lambda = V_D (D(x_0 - x_A) + d)$$

$$V_D = (D V_0 D^T)^{-1}$$

$$V_x = V_0 - V_0 D^T V_D V_0$$

$$\chi^2 = \lambda^T V_D^{-1} \lambda$$

Exercise 2

- Break to do exercise 2

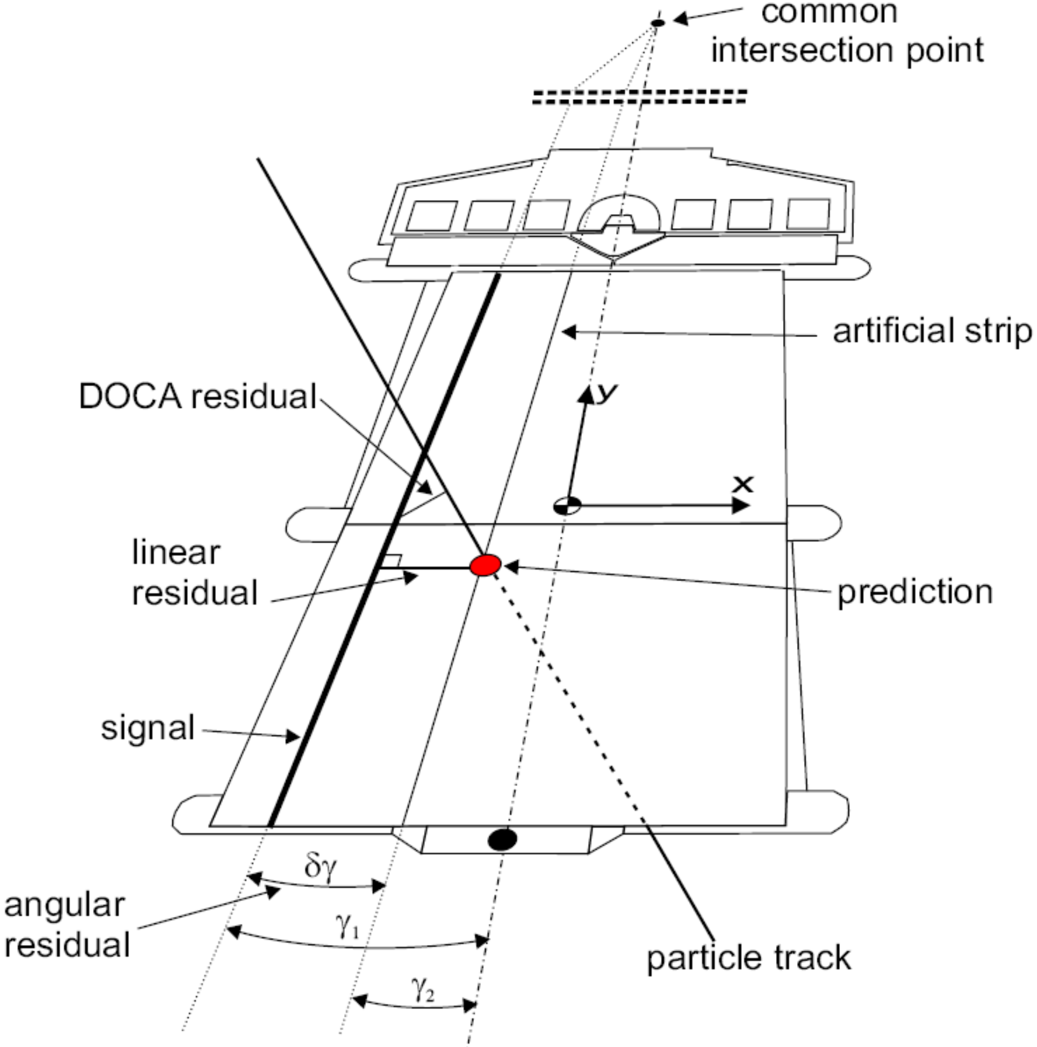
Alignment

- In order to have high resolution unbiased tracking, the detector elements must be correctly aligned.
- This is partly achived by optical survey, and for example laser alignment systems, to track short-term movements.
- The ultimate alignment precision is best achieved by using the fitted tracks themselves.

Alignment with tracks

- Consider a tracker with n planes along the x-axis. From the reconstructed tracks, we want to determine the alignment corrections to the position and orientation of each plane.
- Consider a single measured coordinate y_i and a track model $y=h(\mathbf{x},\boldsymbol{\alpha})$, where **x** are the track parameters and **α** are the alignment corrections.
- A straight forward estimate of $\delta\alpha_i$ is simply the average residual $<r_i = y_i – h(x,\alpha)>$, averaged over all fitted tracks.
- If the considered plane does not take part in the fitted track, r_i is an unbiased residual.
- This *"local"* approach requires in general many iterations because the correlations between planes induced by the fitted tracks are ignored with this method.

Alignment with tracks



From ATLAS alignment paper

Global alignment with tracks

- In the *"global"* approach we define a total *chi2* of a large track sample:

$$\chi^2 = \sum_{tracks} r^T R^{-1} r$$

$$r(x,\alpha,m) = m - h(x,\alpha)$$

where r are the residuals, α the alignment parameters of the detector elements and x the individual track parameters.

What we want is to simultaneously minimise *chi2* both with respect to all the millions of x's and to the many α's.

Sounds impossible, but it isn't!

Alignment with tracks

- It can be shown that the *total* derivative of the *chi2's of each track,* with respect to the alignment parameters are:

$$\frac{d\chi^2}{d\overline{\alpha}} = 2A^T R^{-1}\overline{r}$$

Where *A* is the partial derivative of *r wrt α.*

- Thus the first derivative is *local:* the derivative wrt some α receives only contributions from the local detector element for which the partial derivative A=δr/δα is non-zero.
- Finding δα so that the sum of the derivatives over all tracks be zero thus results in M (alignment parameters) coupled equations – in general non-linear.

Alignment with tracks

- If the summed *chi2* is not already at minimum, we linearize the problem to these M equations:

$$\frac{d\chi^2}{d\overline{\alpha}} + \frac{d^2\chi^2}{d\overline{\alpha}^2}\Delta\overline{\alpha} = \overline{0}$$

- Several algorithms exists for solving them iteratvely. MILLIPEDE is a well-known example (google Blobel).
- Another example is MINRES, minimising the distance between the two sides of the equation (used by CMS).
- Others calculate eigenvectors and eigenvalues of the second derivative exploiting the sparseness of this matrix (used by ATLAS, see ATL-INDET-PUB-2005-002).

Alignment with tracks

- The explicit solution to the alignment problem is thus

$$\Delta\alpha = -\left(\frac{d^2\chi^2}{d\alpha^2}\right)^{-1}\frac{d\chi^2}{d\alpha}$$

or equivalently (assuming r is linear in α):

$$\Delta\alpha_i = -\left[\sum_{tracks}\frac{\partial\overline{r}}{\partial\alpha_i}R^{-1}\frac{\partial\overline{r}^T}{\partial\alpha_j}\right]^{-1}\sum_{tracks}\left[\frac{\partial\overline{r}}{\partial\alpha_j}R^{-1}r^T\right]$$

where $R = V - HCH^T$ is the covariance matrix of the residual vector of a track. See ATL-INDET-PUB-2007-009.

Weak modes

- Diagonalising (half the) the second derivative you get the covariance:

$$C_{kl}(\Delta\overline{\alpha}) = \sum_{j}^{M} \frac{1}{d_j} u_k^{(j)} u_l^{(j)}$$

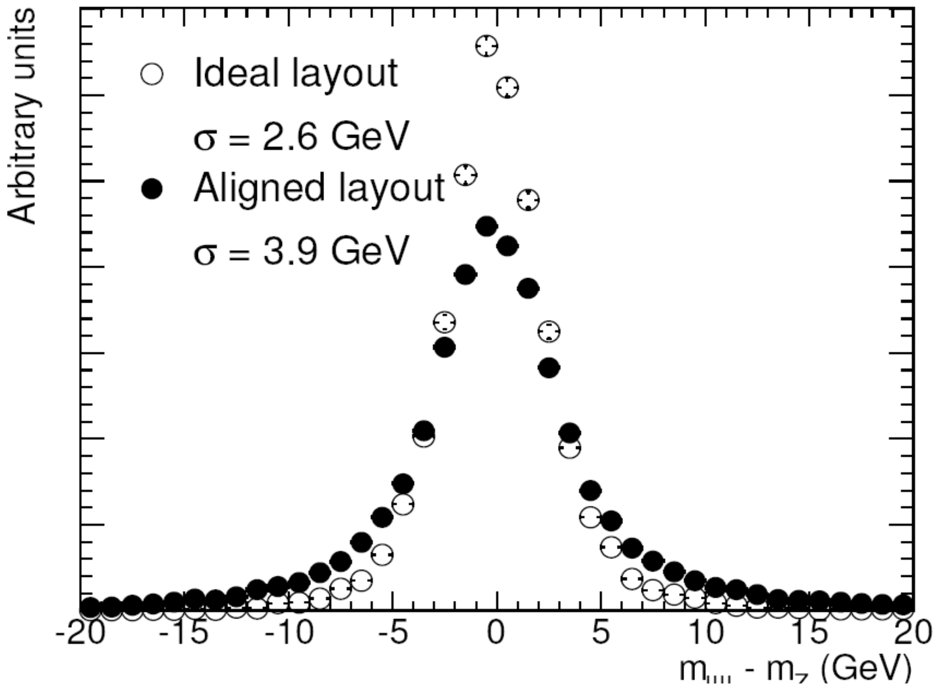
where u_j are eigenvectors of independent collective alignment distortions and d_j are the eigenvalues.

Clearly there is a problem if $d_j=0$. Small eigenvalues corresponds to weak modes poorly constrained by the data.

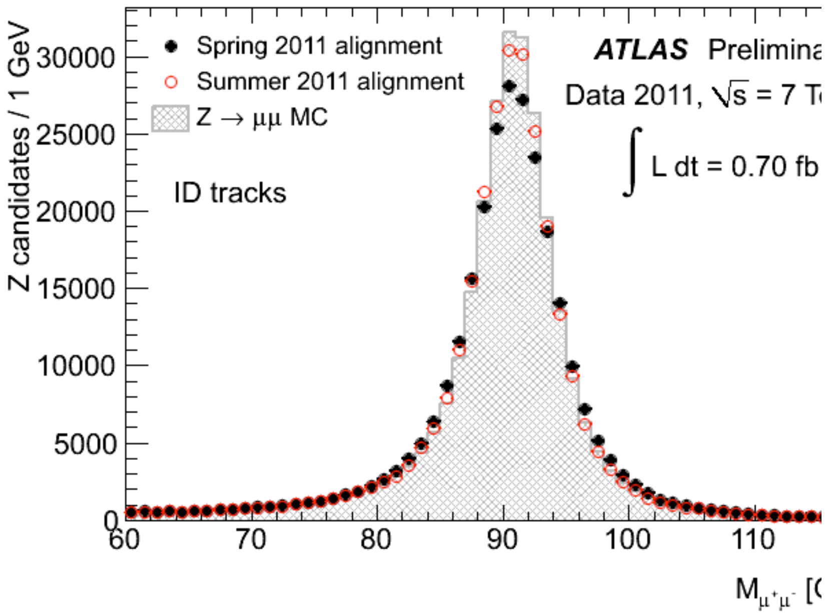
So while most distortions can be corrected by track alignment there are some which may ruin the momentum resolution without being seen in the residuals.

Effect of weak mode misalignment

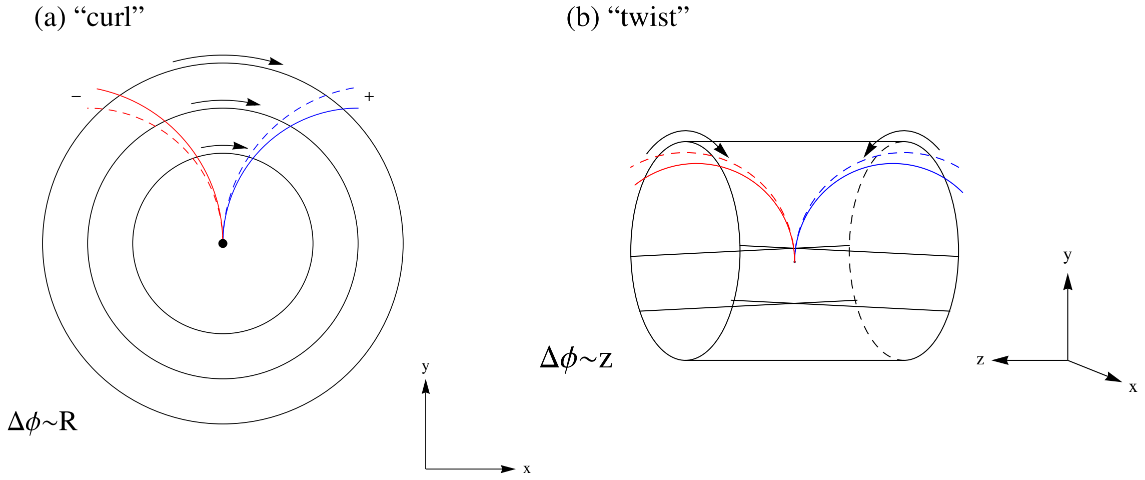
- ATLAS saw the troubles from weak modes (distortions not affecting the track chisquared) already in simulation. Methods are now employed using extra constraints from cosmics, surveys, resonances and multiple detectors in order to correct for weak modes.
- 2008



2011



Curls and twists (q anti-sym)

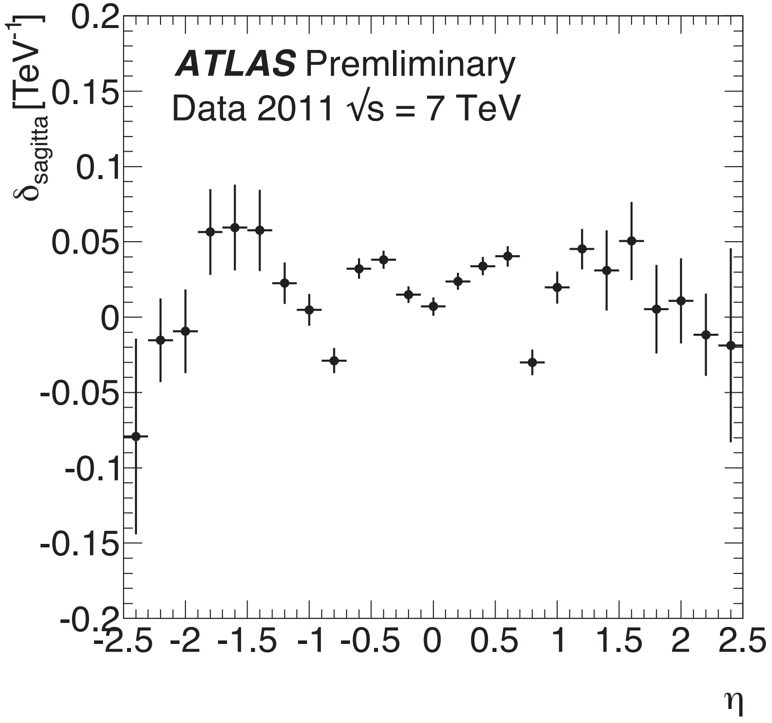
- Imaging a rotation of the various layers in phi proportional with R. That would approximately conserve the helix-shape but bias the momentum (different for positive and negative charge). That is called a curl.
- 

- Imagine a rotation of the end of a layer cylinder. That would approximately conserve the helix-shape but bias the momentum in an eta dependent way. That is called a twist.

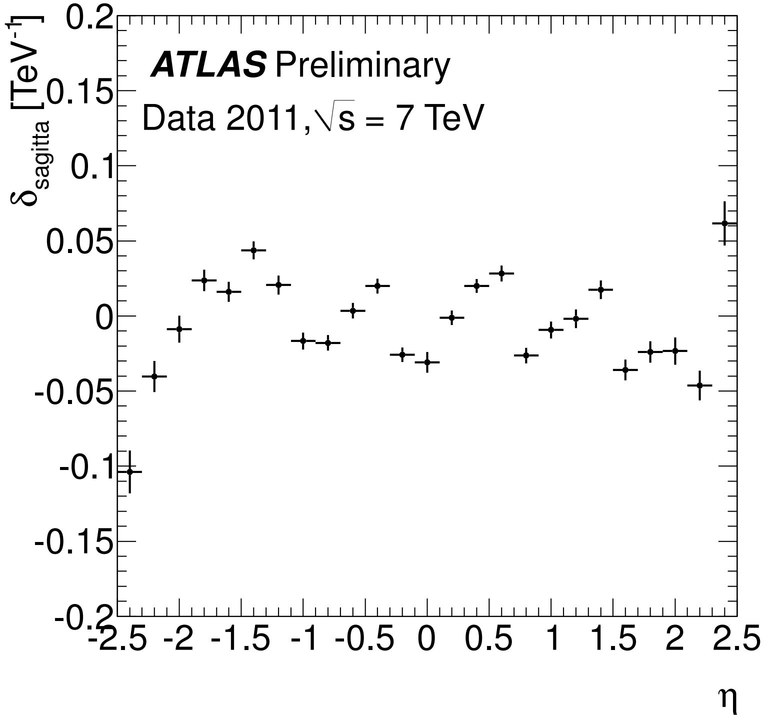
From the ATLAS Silicon alignment group, Bruckman et al

Checking for twists and curls

- No effect of twists and curls seen after latest alignment



E/p (e+ e-)

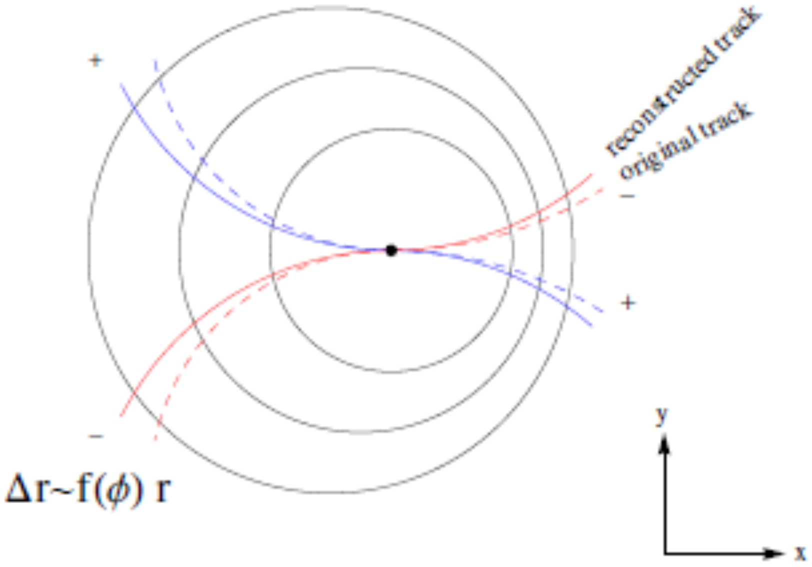


Z->mu+mu- mass

$$q/p_{Corrected} = q/p_{\mathrm{Re}constructed}(1 - qp_T\delta_{Sagitta})$$

Radial deformations (q sym)

- There are also weak mode distortions affecting charges symmetrically – but different at different phi.



$$p_T \longrightarrow p_T(1 + 2\epsilon_{\text{radial}})$$

From the ATLAS Silicon alignment group, Bruckman et al

Checking radial deformations

- Phi dependence of low mass resonances

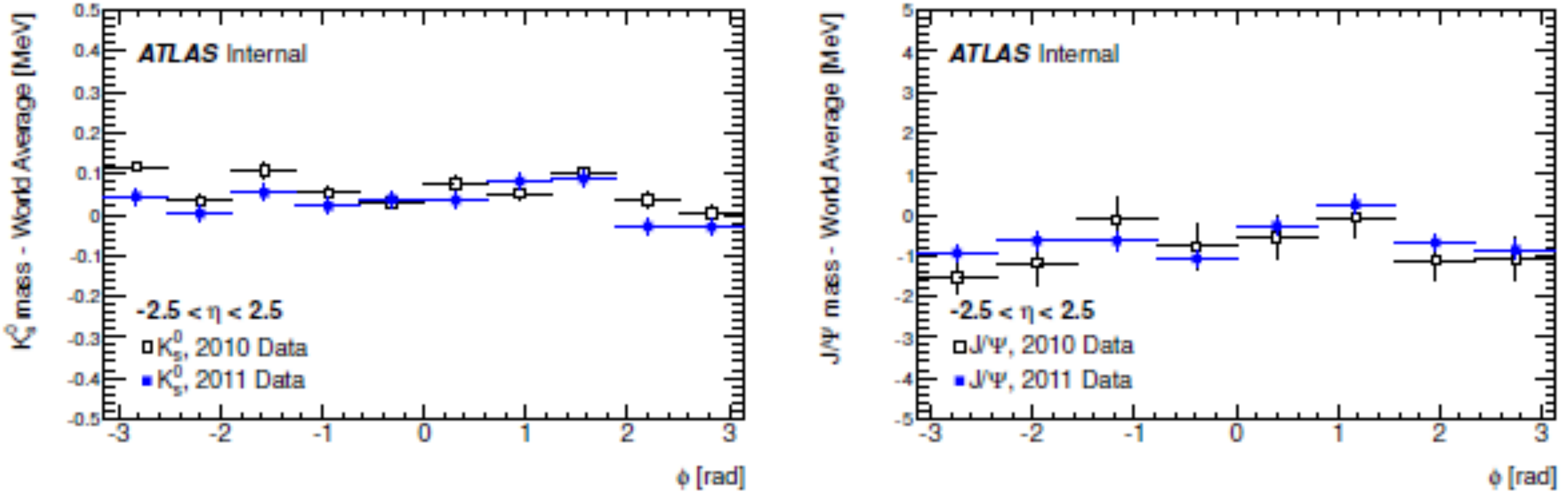
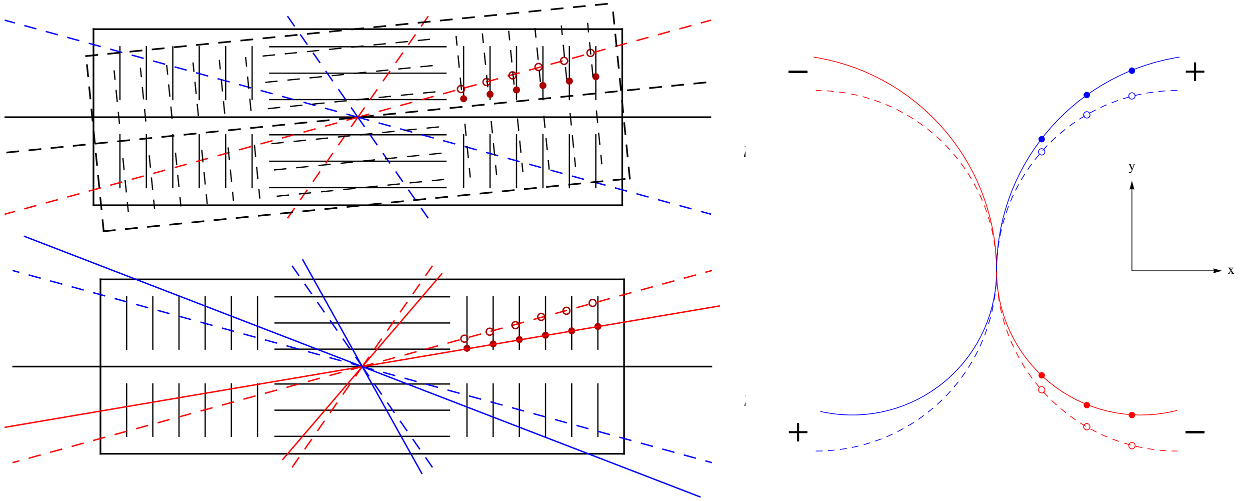


Figure 8: $m_{K_S^0}(\phi_{K_S^0})$ (left) and $m_{J/\psi}(\phi_{J/\psi})$ (right) as a function of ϕ-direction of the resonance.

From the ATLAS Silicon alignment group, Bruckman et al

B field rotations

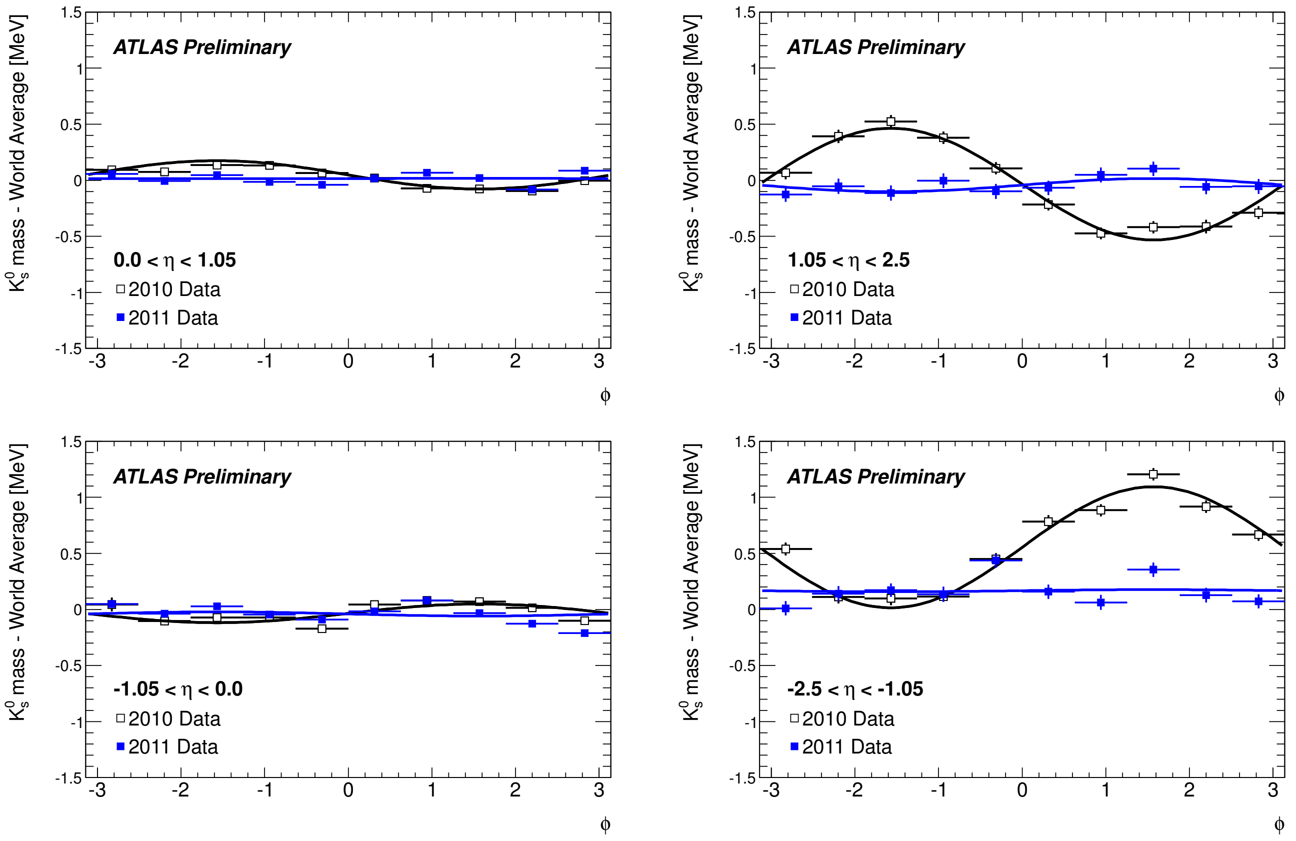


$$p \longrightarrow p(1 - \cot\theta \sin\phi\,\alpha_{rot})$$

From the ATLAS Silicon alignment group, Bruckman et al

Correcting for B field rotation

- The needed extra constraint is here provided by the K0 mass.
- The rotations in data are found from interpolation among simulated rotated samples.



Checking for B field rotation

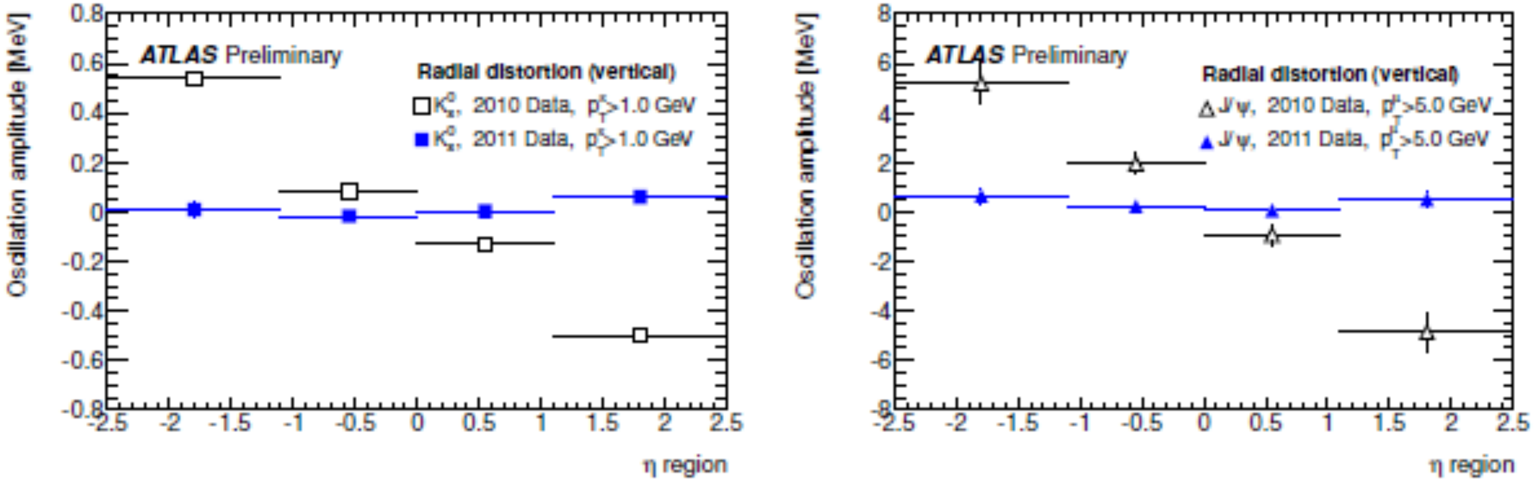
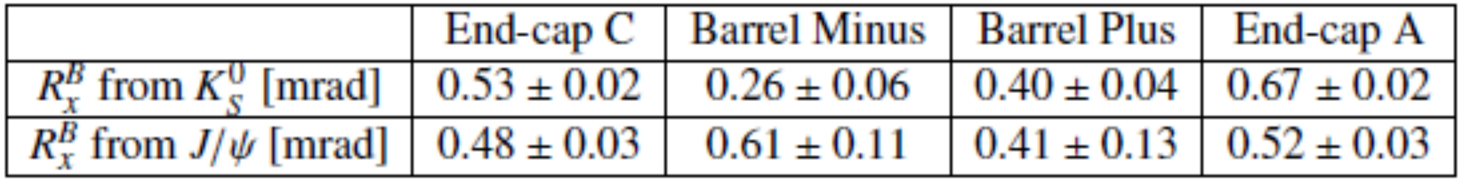


Figure 11: Comparison of the fitted radial-deformation amplitudes in 2010 and 2011 data for the K_S^0 (left) and the J/ψ (right) resonances.

Table 2: Rotation values obtained in different detector regions. Quoted errors are statistical only (from the amplitude fit).



	End-cap C	Barrel Minus	Barrel Plus	End-cap A
R_x^B from K_S^0 [mrad]	0.53 ± 0.02	0.26 ± 0.06	0.40 ± 0.04	0.67 ± 0.02
R_x^B from J/ψ [mrad]	0.48 ± 0.03	0.61 ± 0.11	0.41 ± 0.13	0.52 ± 0.03

From the ATLAS Silicon alignment group, Bruckman et al

Exercise 3

- Break for exercise 3.

Summary

- Good spacepoints is the most important thing! It requires a good detector and knowledge of ALL processes.
- Conditions may depend on time and need to be frequently CALIBRATED and ALIGNED!
- Pattern recognition saves CPU time.
- Fits maximize the likelihood of a set of track and vertex states, given the spacepoints. The Kalman Filter and Global Chisquared minimization are standard local methods.
- Non-linearities are handled by iteration and non-Gaussianity by time-consuming procedures, such as the GSF.
- Further refinements are possible by using global methods as well as mass and vertex constraints.