Tracking at High Energy Physics Detectors

Sebastian Fleischmann, Univ. Bonn

- Introduction
- Motion of charged particles in magnetic fields
- Track fitting
  - Propagation of track parameters
  - Methods
    - Global Least Squares Methods
    - Recursive Filters
  - Material Effects
- Pattern recognition and track finding
- Summary

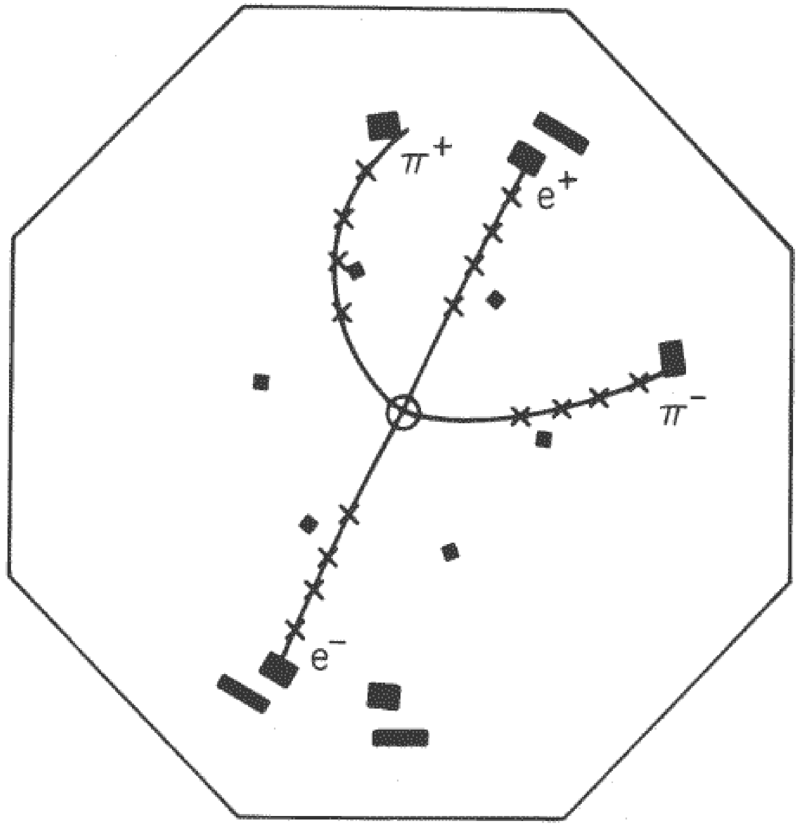
PHYSIKALISCHES INSTITUT

universitätbonn

ELPHI
26024 / 1730
0.0 cm

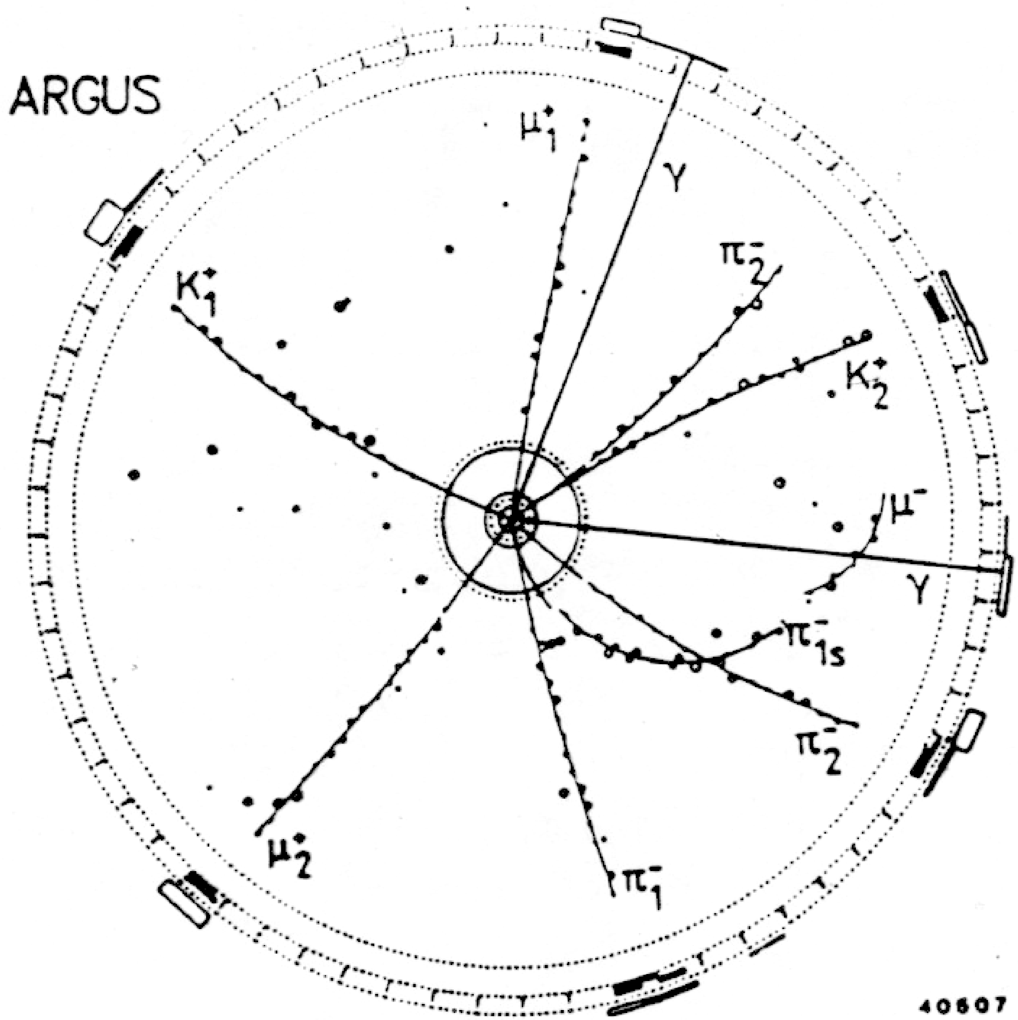
Introduction:
History of Tracking in HEP experiments

- J/Psi event at MARK-I detector (SLAC); e^+e^- @ 3 GeV



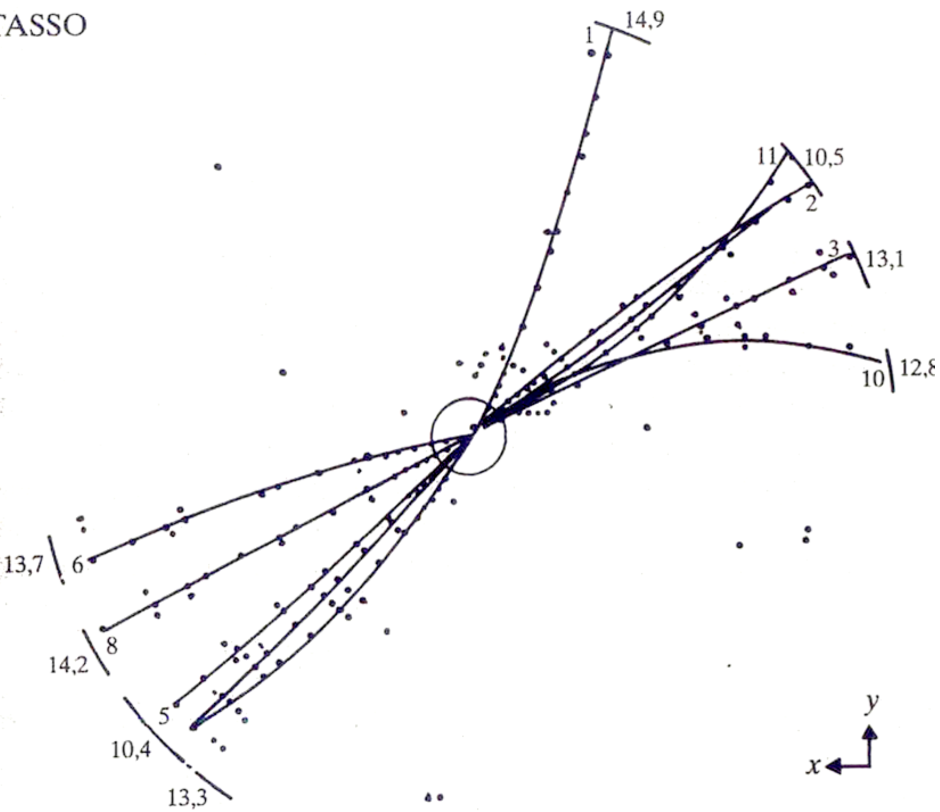
Introduction:
History of Tracking in HEP experiments

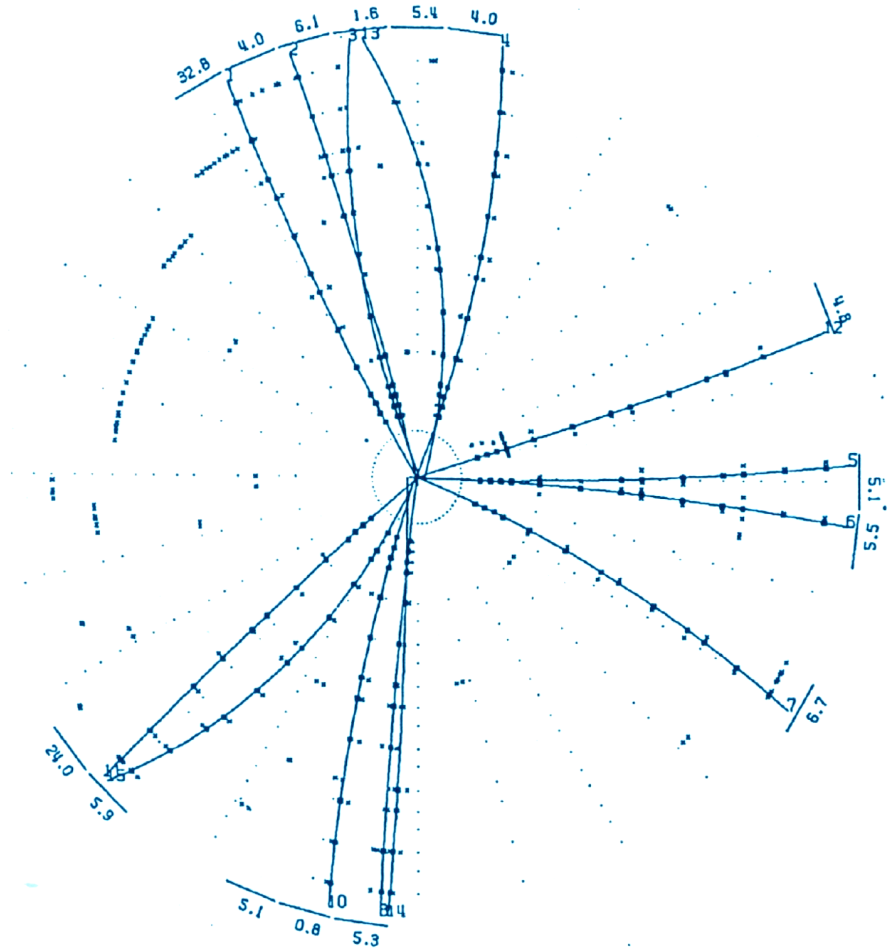
- Discovery of $B/\bar{B}$-mixing of ARGUS at DORIS (1987); e^+e^- @ 10 GeV



Introduction:
History of Tracking in HEP experiments

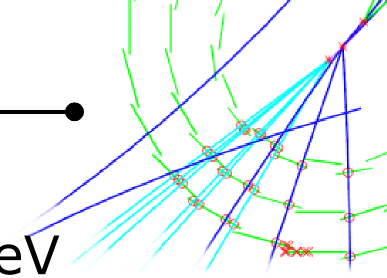
- 2-jet / 3-jet event (gluon discovery) of TASSO at PETRA (1979); e^+e^- @ 30 GeV



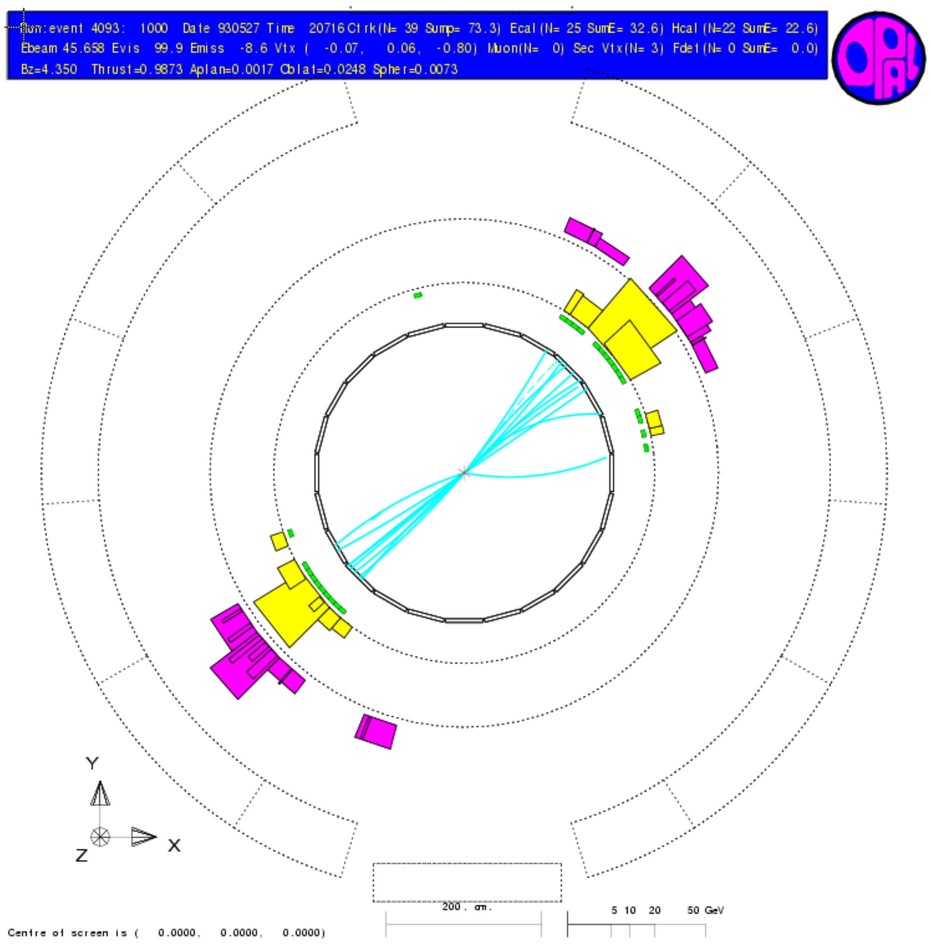


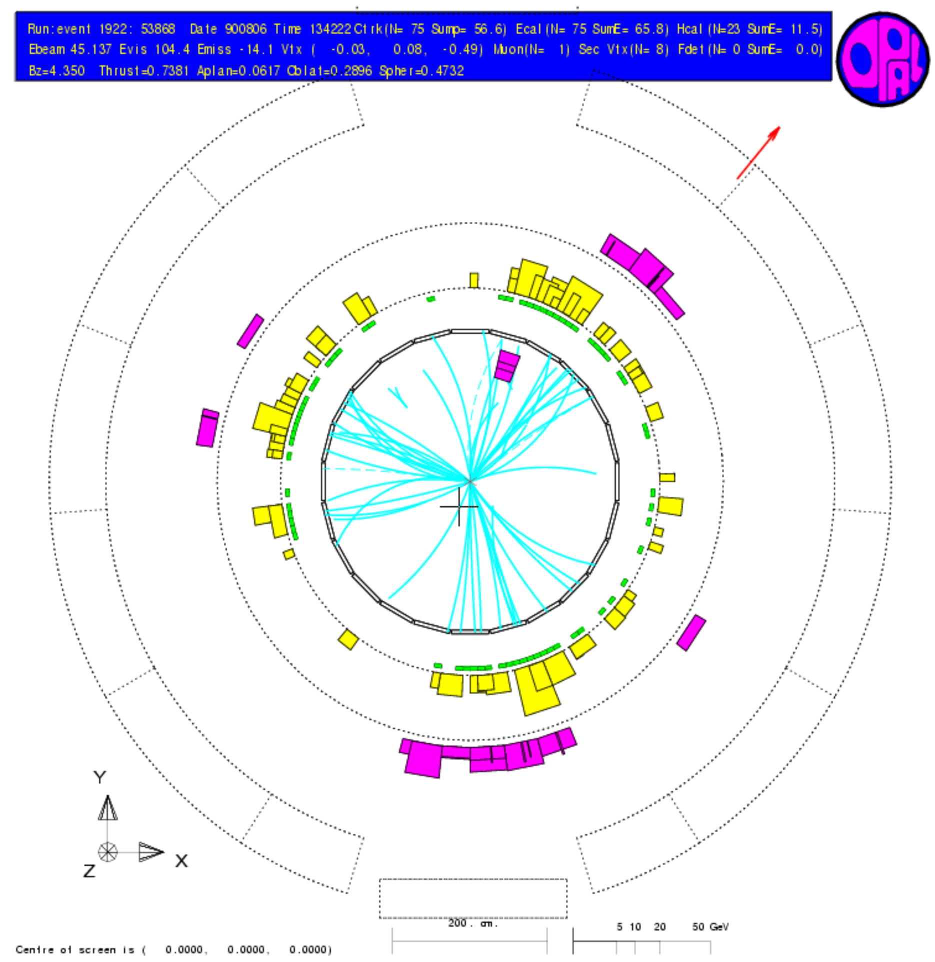
Introduction:

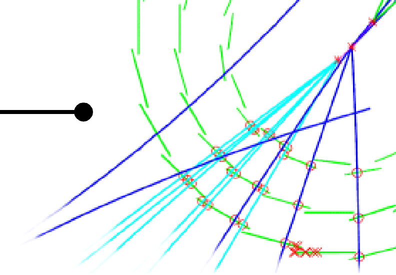
History of Tracking in HEP experiments



- 2-jet and 3-jet events in OPAL at LEP; e^+e^- @ 90 GeV

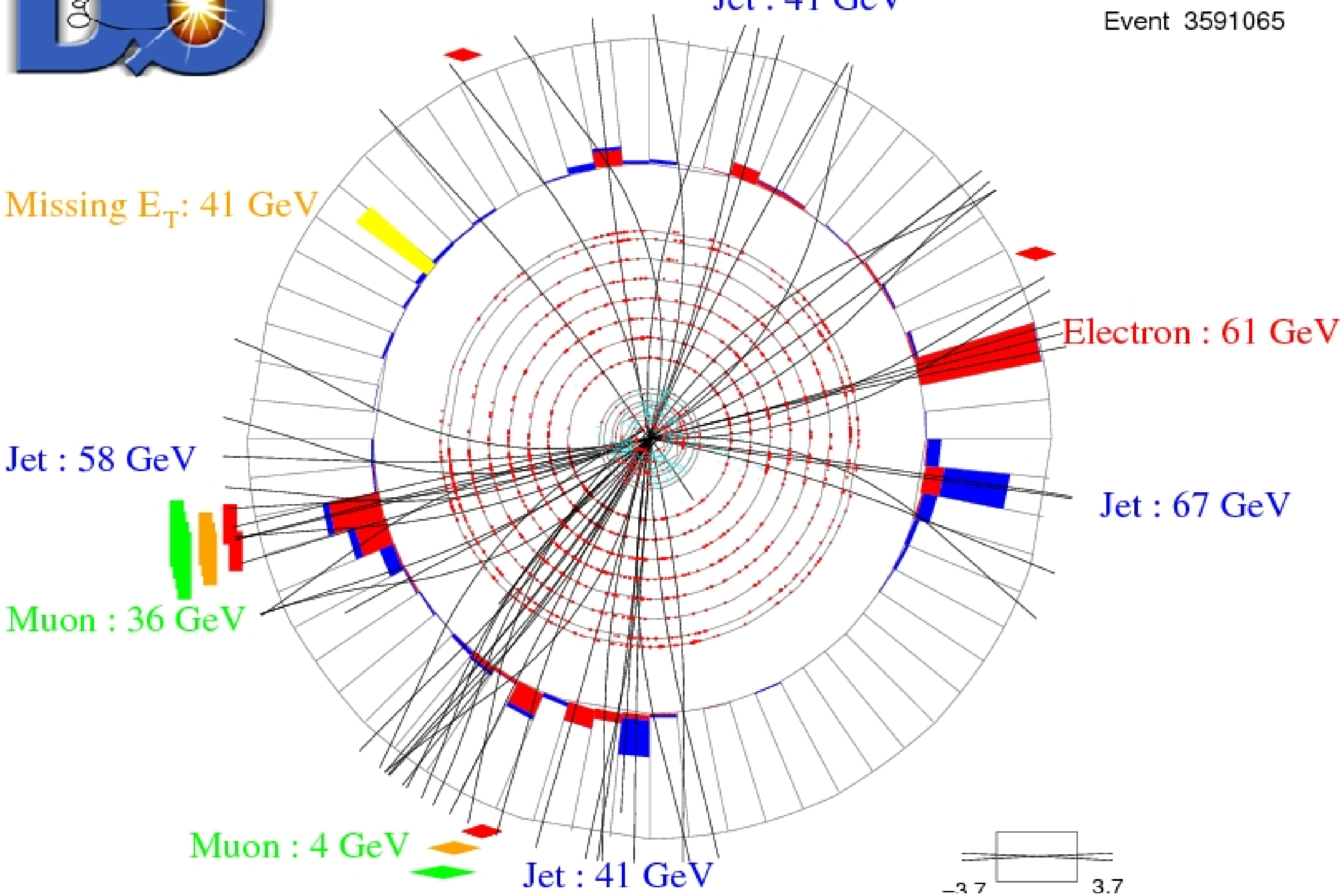






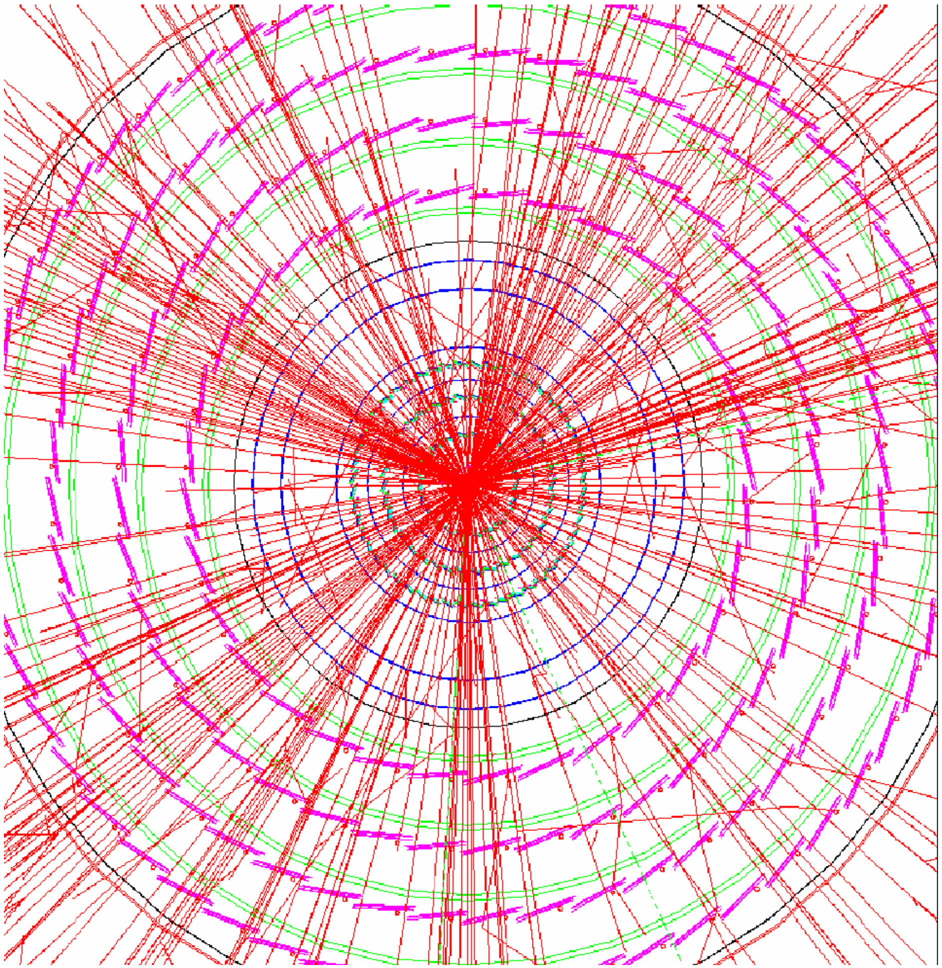
Introduction:
History of Tracking in HEP experiments

- ttH event in DØ at Tevatron; $\bar{p}p$ @ 2 TeV

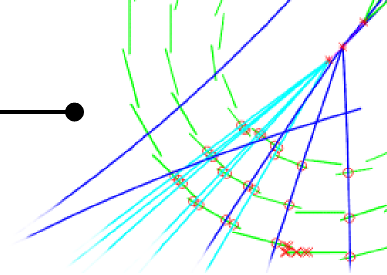


Introduction:
History of Tracking in HEP experiments

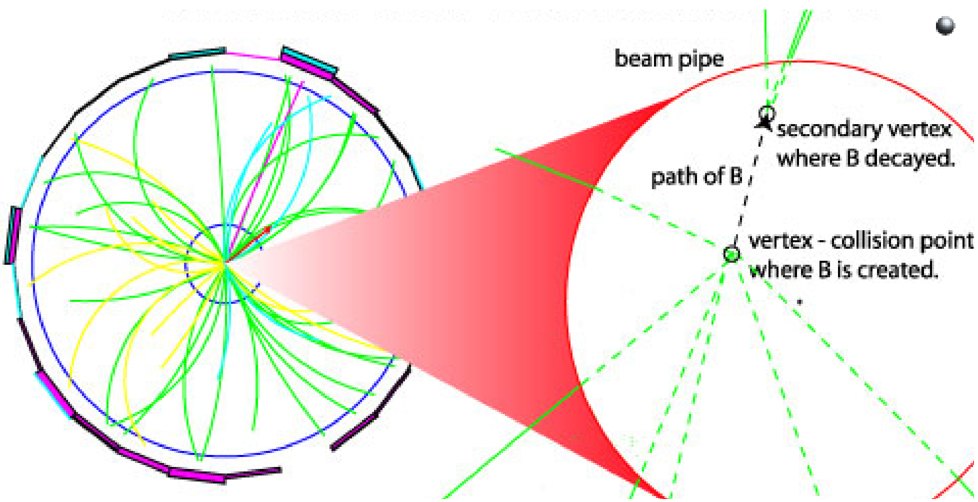
- ATLAS at LHC; pp @ 14 TeV



Introduction: Why Tracking?

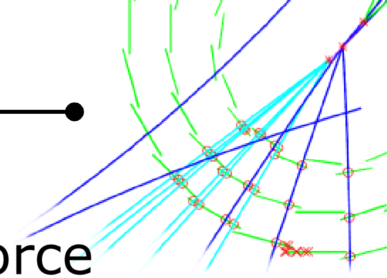


- Detector Output ⟶
  - Layer-based position measurements
    - 2D: Pixel
    - 1D: Silicon Strip
    - 1D: Drift Chambers
  - continuous position measurements (Time Projection Chambers,...)
- Analysis Input
  - (four-)momentum of charged particles
  - sign of charge
  - ID tags of particle type



- lifetime tags to identify B-hadrons (or tau leptons,...)
  - Impact parameter (IP)
  - measurement of a secondary vertex
  - high demands on reconstruction precision ($c\tau$ = 500 µm)

Charged particles in magnetic fields



- Motion of charged particle determined by Lorentz force

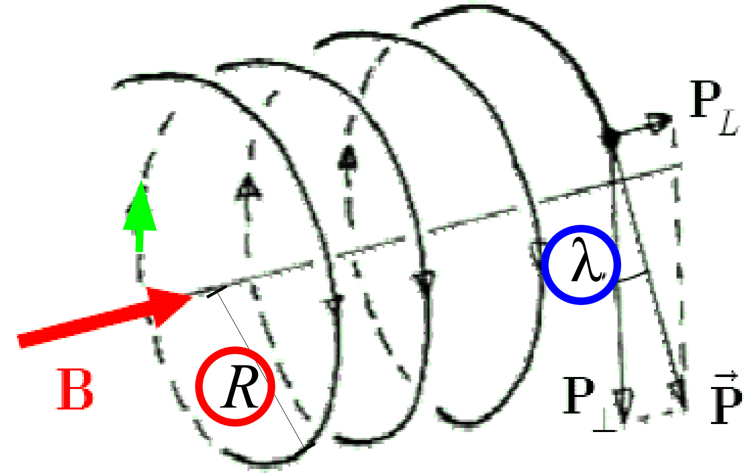
$$m_0 \gamma \frac{d^2 \vec{r}}{dt^2} = e \frac{d\vec{r}}{dt} \times \vec{B} \qquad \frac{d^2 \vec{r}}{ds^2} = \frac{e}{p} \frac{d\vec{r}}{ds} \times \vec{B} \qquad (ds = v\,dt)$$

- Solution in homogeneous field: Helix

$$x(s) = x_0 + R\left[\cos\left(\Phi_0 + \frac{h\cos\lambda}{R} s\right) - \cos\Phi_0\right]$$

$$y(s) = y_0 + R\left[\sin\left(\Phi_0 + \frac{h\cos\lambda}{R} s\right) - \sin\Phi_0\right]$$

$$z(s) = z_0 + s\sin\lambda$$

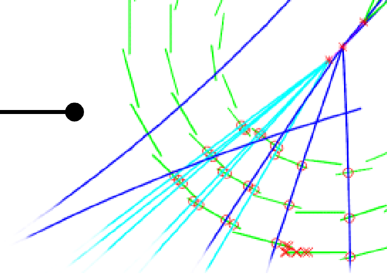


$$P_T = P\cos\lambda = 0.3\,B\,R \quad ([P] = \mathrm{GeV}, [B] = \mathrm{T}, [R] = \mathrm{m})$$ (transverse momentum)

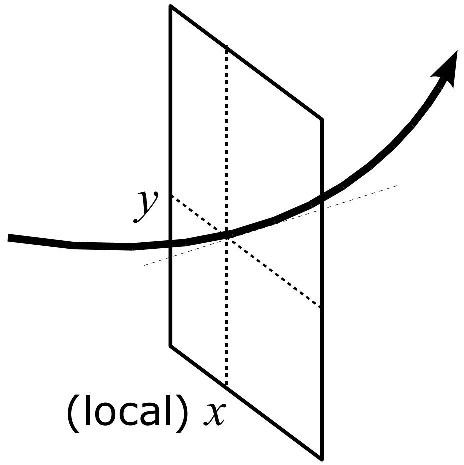
$h = \pm 1$ sense of rotation of helix ⇒ sign of charge

$$(\lambda = \pi/2 - \theta)$$

Track Parameters and Propagation

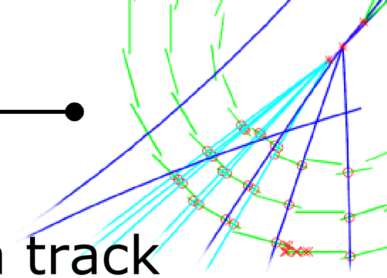


- Track Parameters $\vec{\lambda}$ are given at reference surfaces
  - point of closest approach to origin of coordinate system
  - estimated interaction point
  - measurement layers
- 5 degrees of freedom, e.g. $\vec{\lambda} = \left(\frac{q}{p}, \theta, \phi, x, y \right)$

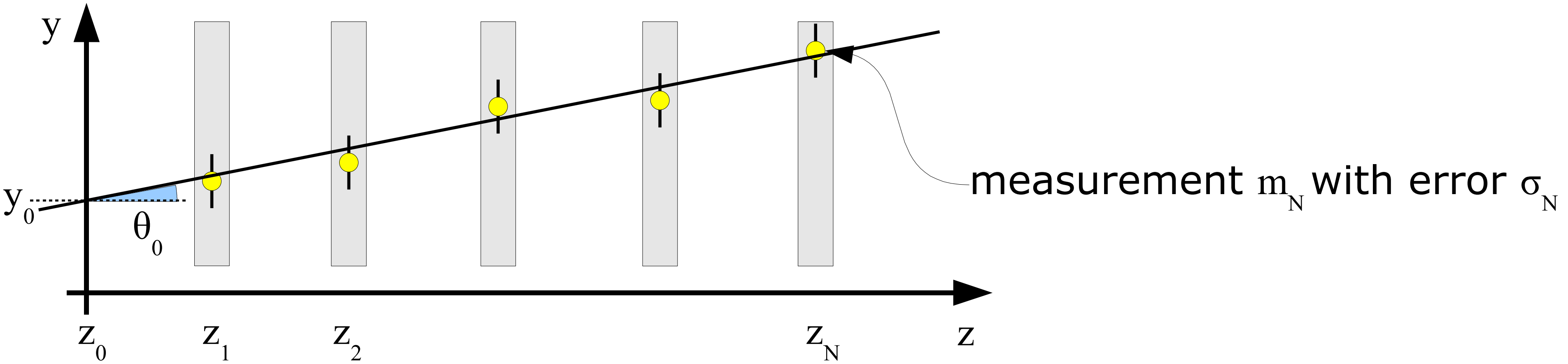


- Some definitions and notation:
  - Measurement positions given by
    $\vec{m} = f(\vec{\lambda}^{true}) + \vec{\epsilon}, \quad \langle \vec{\epsilon} \rangle = 0$
  - Covariance Matrix (“error matrix”) of measurements
    $cov(\vec{m}) = \mathbf{V} = \langle (\vec{m} - \langle \vec{m} \rangle)(\vec{m} - \langle \vec{m} \rangle)^T \rangle$
  - Covariance Matrix of estimated Track Parameters
    $cov(\vec{\lambda}) = C_{\lambda} = \langle (\vec{\lambda} - \vec{\lambda}^{true})(\vec{\lambda} - \vec{\lambda}^{true})^T \rangle$
- Propagation of parameters and related covariance matrix from one surface to another can be done
  - analytically, e.g. helix
  - by numerical solution of ODE, e.g. in inhomogeneous field with Runge-Kutta integration

Track fits: Estimating track parameters

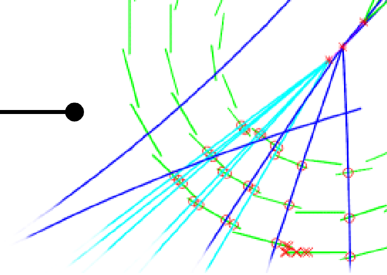


- Estimate the parameters $\vec{\lambda}$ of the trajectory using a track model $f_{\vec{\lambda}}(x)$
  - The track model may be helical, but it can even be non-analytical
- A straight-line fit in a plane will serve as an example in the following, assuming a layer-based detector layout



  - In this example the track parameters $\vec{\lambda}$ at the reference surface z_0 are one local position (y_0) and one angle (θ_0)

Track fits:
Least Squares Methods



- A well known method is to minimize the χ^2:

$$\chi^2 = \sum_{i=1}^{N} \frac{(m_i - f_{\vec{\lambda}}(x_i))^2}{\sigma_i^2}$$

which can be written as

$$\chi^2 = (\vec{m} - \vec{f})^T \mathbf{V}^{-1} (\vec{m} - \vec{f}) \qquad \mathbf{V} = diag\{\sigma_i^2\}$$

requiring the derivative to vanish leads to

$$\mathbf{F}^T \mathbf{V}^{-1} \vec{f} = \mathbf{F}^T \mathbf{V}^{-1} \vec{m} \qquad \mathbf{F} = \frac{\partial \vec{f}}{\partial \vec{\lambda}} = \left(\frac{\partial f_{\vec{\lambda}}(x_i)}{\partial \lambda_j} \right)$$

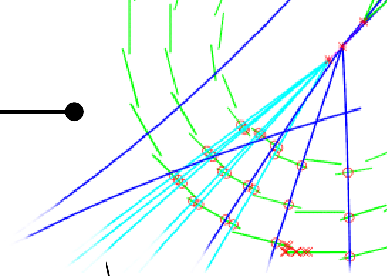
- In case of a linear function $\vec{f} = \mathbf{F} \vec{\lambda}$ this can directly be inverted to

$$\vec{\lambda} = (\mathbf{F}^T \mathbf{V}^{-1} \mathbf{F})^{-1} \mathbf{F}^T \mathbf{V}^{-1} \vec{m}$$

- The covariance matrix of the parameter estimate is

$$cov(\vec{\lambda}) = C_\lambda = (\mathbf{F}^T \mathbf{V}^{-1} \mathbf{F})^{-1}$$

- C_λ^{-1} is of dimension $N_\lambda \times N_\lambda$ and therefore inexpensive to invert
- V diagonal ⇒ inversion trivial



Track fits:
Least Squares Methods – example

- estimate parameters of

$$y = f(z) = a + bz \qquad \vec{f} = \mathbf{F}\vec{\lambda} = \begin{pmatrix} f_0 \\ \vdots \\ f_N \end{pmatrix} = \begin{pmatrix} a + bz_0 \\ \vdots \\ a + bz_N \end{pmatrix} = \begin{pmatrix} 1 & z_0 \\ 1 & \vdots \\ 1 & z_N \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix}$$

  using the measurements $\vec{m} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}$

$$\mathbf{V} = diag\left\{\sigma_i^2\right\}$$

  requiring the derivative to vanish leads to

$$\vec{\lambda} = \left(\mathbf{F}^T \mathbf{V}^{-1} \mathbf{F}\right)^{-1} \mathbf{F}^T \mathbf{V}^{-1} \vec{m}$$

$$= \underbrace{\left(\begin{pmatrix} 1 & 1 & 1 \\ z_0 & \cdots & z_N \end{pmatrix} \begin{pmatrix} 1/\sigma_1^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1/\sigma_N^2 \end{pmatrix} \begin{pmatrix} 1 & z_0 \\ 1 & \vdots \\ 1 & z_N \end{pmatrix} \right)^{-1}}_{\left(\begin{pmatrix} \sum_{i=1}^{N} \frac{1}{\sigma_i^2} & \sum_{i=1}^{N} \frac{z_i}{\sigma_i^2} \\ \sum_{i=1}^{N} \frac{z_i}{\sigma_i^2} & \sum_{i=1}^{N} \frac{z_i^2}{\sigma_i^2} \end{pmatrix} \right)^{-1}} \underbrace{\begin{pmatrix} 1 & 1 & 1 \\ z_0 & \cdots & z_N \end{pmatrix} \begin{pmatrix} 1/\sigma_1^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1/\sigma_N^2 \end{pmatrix} \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}}_{\begin{pmatrix} \sum_{i=1}^{N} \frac{y_i}{\sigma_i^2} \\ \sum_{i=1}^{N} \frac{y_i z_i}{\sigma_i^2} \end{pmatrix}}$$

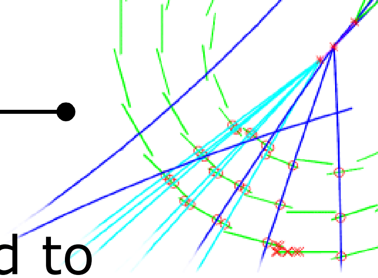
Track fits:
Kalman Filter

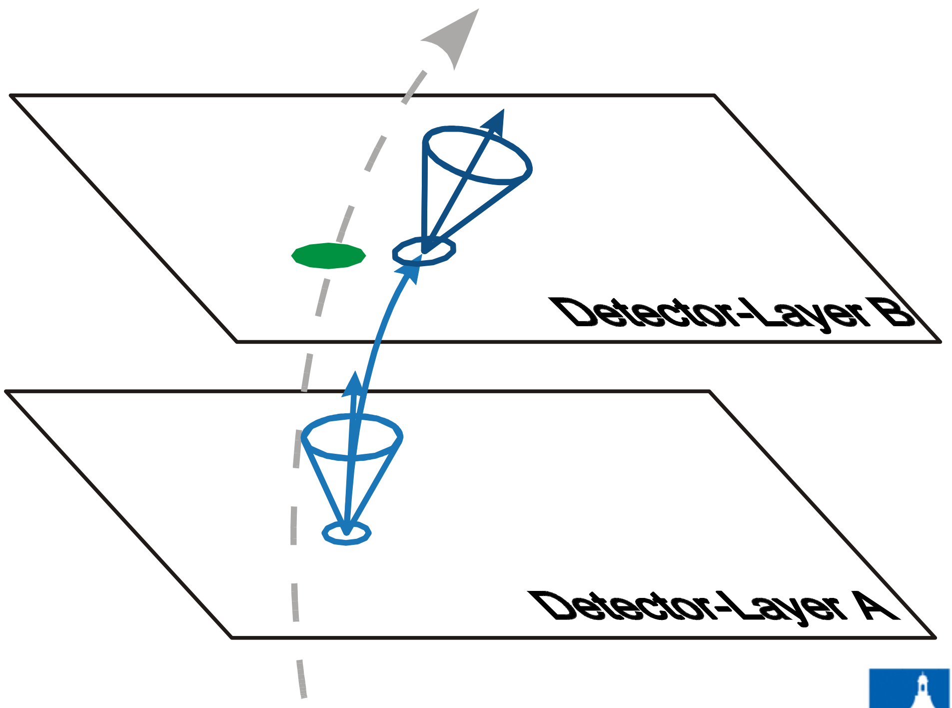
- The track parameters on surface k-1 are propagated to surface k (*prediction*)

$$\vec{\lambda}_k^{k-1} = \mathbf{F}_k \vec{\lambda}_{k-1} \qquad \left(\mathbf{C}_k^{k-1}\right)^{-1} = \mathbf{F}_k \left(\mathbf{C}_{k-1}\right)^{-1} \mathbf{F}_k^T$$

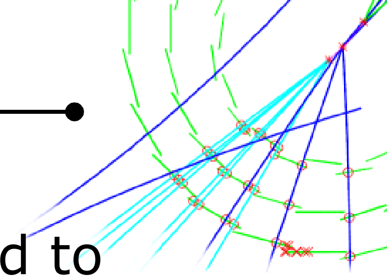
- Those track parameters on layer k can be projected to get the predicted measurement position

$$\mathbf{H}_k \vec{\lambda}_k^{k-1}$$





Track fits:
Kalman Filter



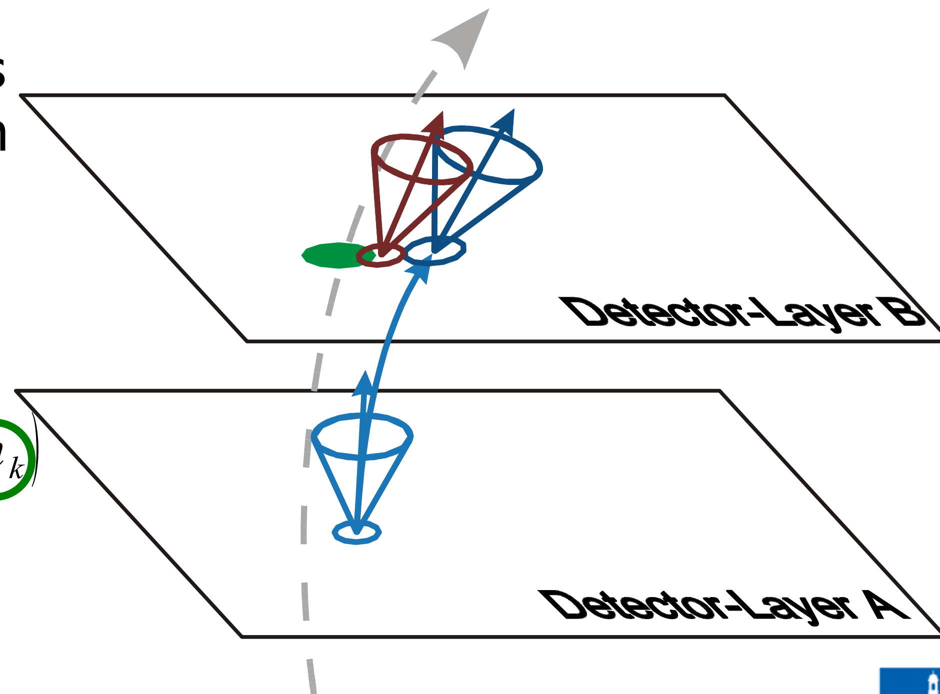
- The track parameters on surface k-1 are propagated to surface k (*prediction*)

$$\vec{\lambda}_k^{k-1} = \mathbf{F}_k \vec{\lambda}_{k-1} \qquad (\mathbf{C}_k^{k-1})^{-1} = \mathbf{F}_k (\mathbf{C}_{k-1})^{-1} \mathbf{F}_k^T$$

- Those track parameters on layer k can be projected to get the predicted measurement position

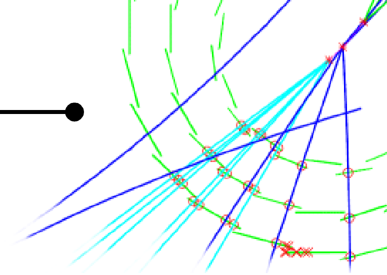
$$\mathbf{H}_k \vec{\lambda}_k^{k-1}$$

- The updated track parameters including the measurement on layer k are calculated as weighted mean (*update*)



$$\vec{\lambda}_k^k = \mathbf{C}_k^k \left((\mathbf{C}_k^{k-1})^{-1} \vec{\lambda}_k^{k-1} + \mathbf{H}_k^T (\mathbf{V}_k)^{-1} m_k \right)$$

$$(\mathbf{C}_k^k)^{-1} = (\mathbf{C}_k^{k-1})^{-1} + \mathbf{H}_k^T (\mathbf{V}_k)^{-1} \mathbf{H}_k$$



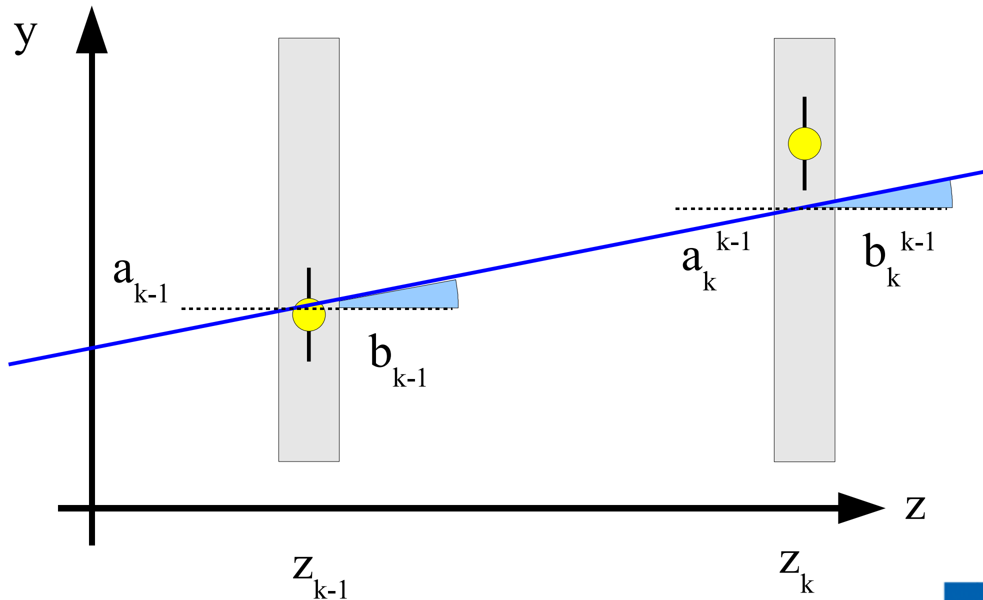
Track fits:
Kalman Filter – example

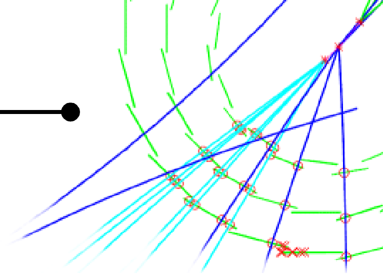
- Propagated track parameters (*prediction*)

$$\vec{\lambda}_k^{k-1} = \begin{pmatrix} a_k^{k-1} \\ b_k^{k-1} \end{pmatrix} = \mathbf{F}_k \vec{\lambda}_{k-1} = \begin{pmatrix} a_{k-1} + b_{k-1}(z_k - z_{k-1}) \\ b_{k-1} \end{pmatrix} = \begin{pmatrix} 1 & z_k - z_{k-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_{k-1} \\ b_{k-1} \end{pmatrix}$$

- Predicted measurement, i.e. the expected position

$$\mathbf{H}_k \vec{\lambda}_k^{k-1} = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} a_k^{k-1} \\ b_k^{k-1} \end{pmatrix} = a_k^{k-1}$$





Track fits:
Kalman Filter – example

- Propagated track parameters (*prediction*)

$$\vec{\lambda}_k^{k-1} = \begin{pmatrix} a_k^{k-1} \\ b_k^{k-1} \end{pmatrix} = \mathbf{F}_k \vec{\lambda}_{k-1} = \begin{pmatrix} a_{k-1} + b_{k-1}(z_k - z_{k-1}) \\ b_{k-1} \end{pmatrix} = \begin{pmatrix} 1 & z_k - z_{k-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} a_{k-1} \\ b_{k-1} \end{pmatrix}$$

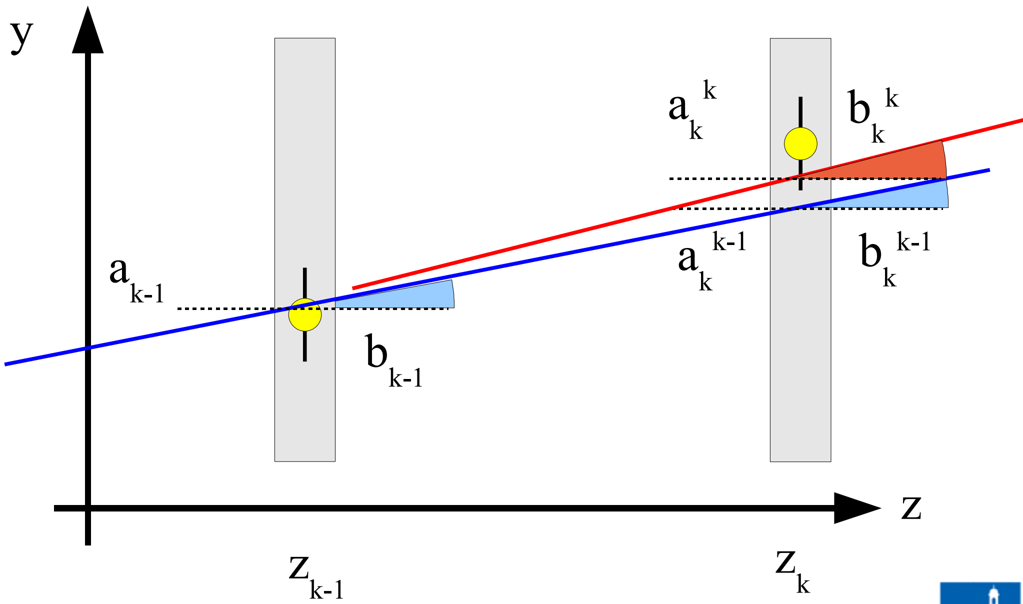
- Predicted measurement, i.e. the expected position

$$\mathbf{H}_k \vec{\lambda}_k^{k-1} = \begin{pmatrix} 1 & 0 \end{pmatrix} \begin{pmatrix} a_k^{k-1} \\ b_k^{k-1} \end{pmatrix} = a_k^{k-1}$$

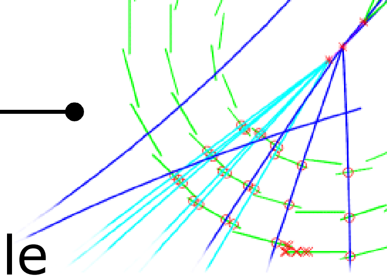
- Updated track parameters

$$\vec{\lambda}_k^{k} = \mathbf{C}_k^k \left((\mathbf{C}_k^{k-1})^{-1} \begin{pmatrix} a_k^{k-1} \\ b_k^{k-1} \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \frac{1}{\sigma_k^2} y_k \right)$$

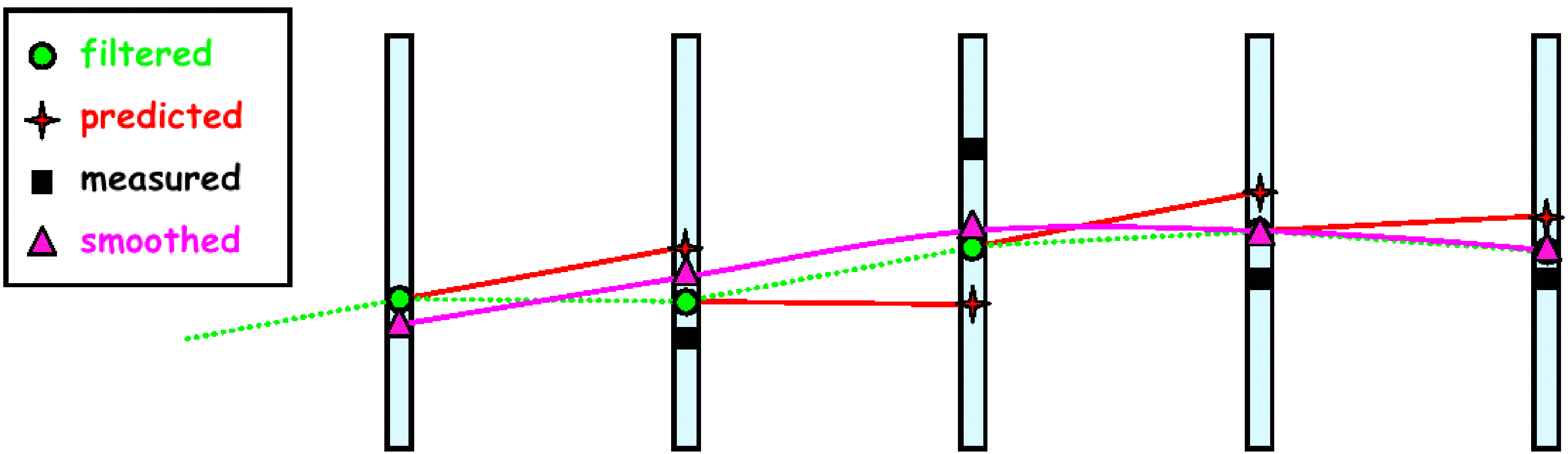
$$(\mathbf{C}_k^k)^{-1} = (\mathbf{C}_k^{k-1})^{-1} + \mathbf{H}_k^T (\mathbf{V}_k)^{-1} \mathbf{H}_k$$
$$= (\mathbf{C}_k^{k-1})^{-1} + \begin{pmatrix} 1 \\ 0 \end{pmatrix} \left(\frac{1}{\sigma_k^2} \right) \begin{pmatrix} 1 & 0 \end{pmatrix}$$

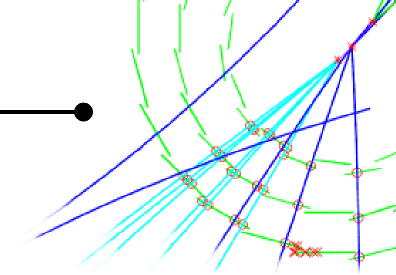


Track fits:
Kalman Filter – Smoothing

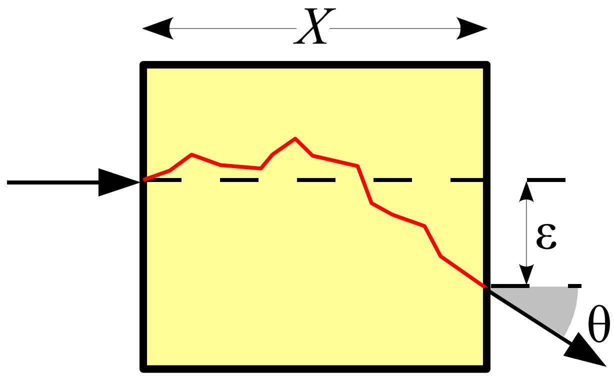


- Only last updated state $\vec{\lambda}_N^N$ at layer N containes whole information from all measurements
- *Smoothing* to get whole info at every layer:
  - Run second filter in backward direction
  - Weighted mean of forward and backward filter gives smoothed estimate





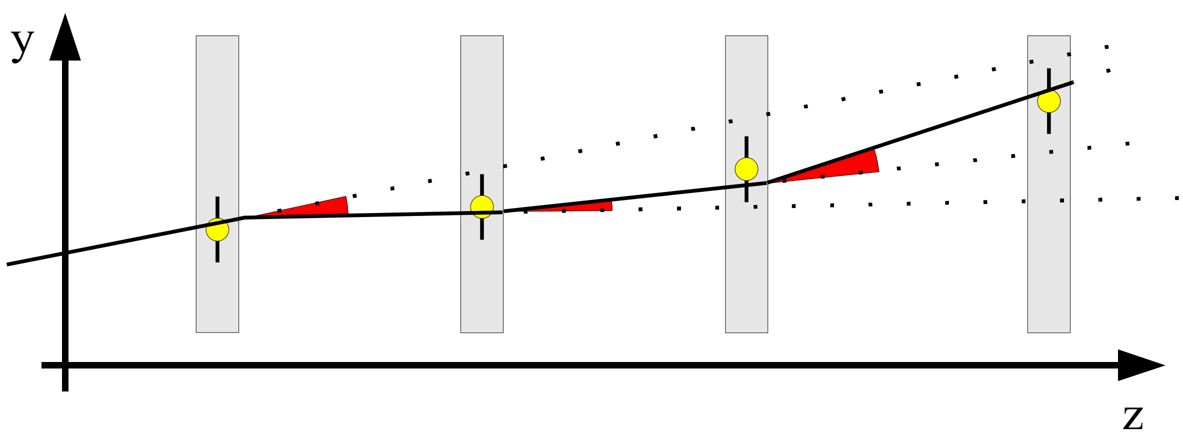
Track fits: Multiple scattering

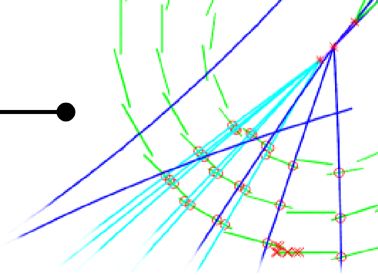


$$\langle\theta\rangle = 0, \quad \langle\theta^2\rangle \propto \frac{X}{X_0}\frac{1}{p^2}$$

$$\langle\epsilon\rangle = 0, \quad \langle\epsilon^2\rangle \propto \frac{X}{X_0}\frac{1}{p^2}X^2$$

- Distribution almost Gaussian + tails
- Assume thin detector layers:
  - lateral displacement can be neglected

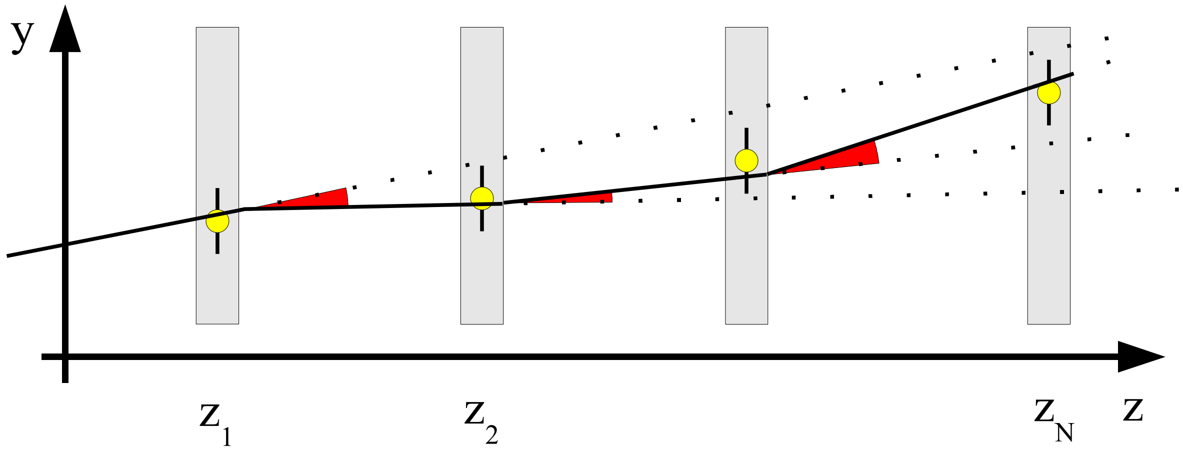


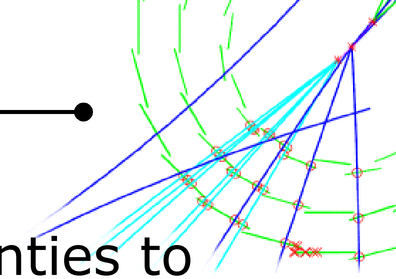


Track fits:
Multiple scattering in Global Methods

- Break Point method
  - adequate for limited number of strong scatterers
  - scattering angles are additional parameters of the fit

$$\chi^2 = (\vec{m} - \vec{f}_{\vec{\lambda},\vec{\theta}})^T \mathbf{V}^{-1} (\vec{m} - \vec{f}_{\vec{\lambda},\vec{\theta}}) + \vec{\theta}^T \left(cov(\vec{\theta}) \right)^{-1} \vec{\theta}$$



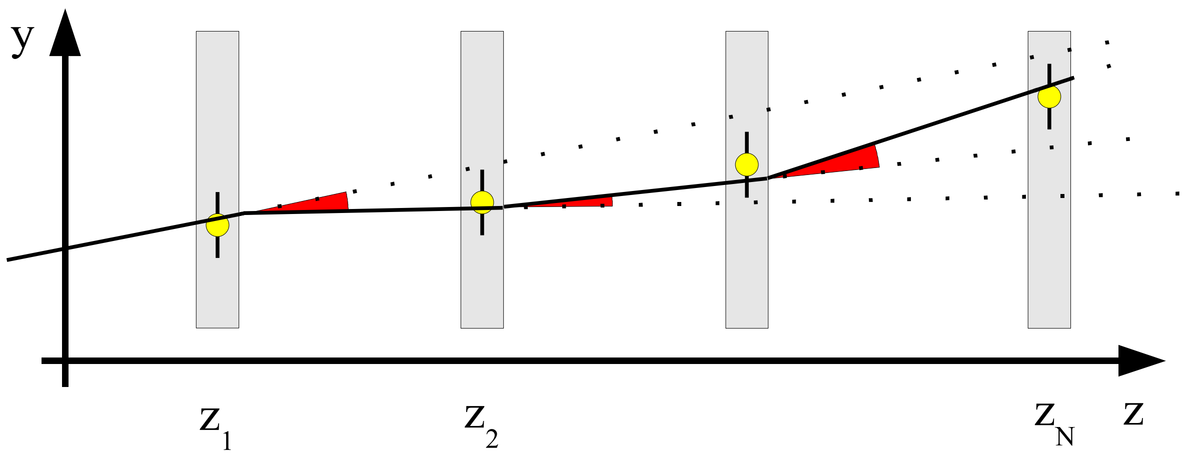


Track fits: Multiple scattering in Global Methods

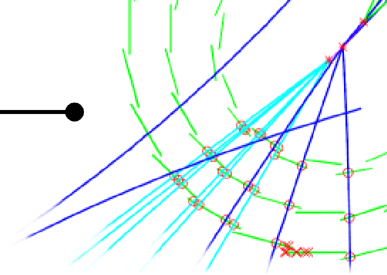
- Scattering angles can be handled as extra uncertainties to measurements

$$\mathbf{V} = diag\left\{\sigma_i^2\right\} + M\left(\left\langle\theta_i^2\right\rangle, dist\left(z_1, \ldots, z_N\right)\right)$$

  - BUT: Introduces correlations
  - Remember: V ($N \times N$) needs to be inverted, which gets very costly for large N
  - Other disadvantage: track fit follows ideal extrapolation of track parameters at reference surface, not scattered path



Track fits:
Multiple scattering in Kalman Filter



- Introduction of material effects rather simple
  - for multiple scattering just add term in propagation of covariances of track parameters

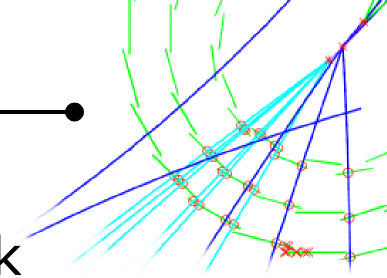
$$(\mathbf{C}_k^{k-1})^{-1} = \mathbf{F}_k (\mathbf{C}_{k-1})^{-1} \mathbf{F}_k^T + \mathbf{Q}_k$$

  - example

$$(\mathbf{C}_k^{k-1})^{-1} = \begin{pmatrix} 1 & z_k - z_{k-1} \\ 0 & 1 \end{pmatrix} (\mathbf{C}_{k-1})^{-1} \begin{pmatrix} 1 & 0 \\ z_k - z_{k-1} & 1 \end{pmatrix} + \begin{pmatrix} (z_k - z_{k-1})^2 \langle \theta_k^2 \rangle & \\ & \langle \theta_k^2 \rangle \end{pmatrix}$$

- Other effects like energy loss can be considered similarly
- Advantage: Track fit follows scattered path closely, including prediction at each layer allowing for
  - precise material estimation
  - propagation in inhomogeneous magnetic fields possible
  - pattern recognition (see next slides...)

Track fits:
Quality of fit and Outlier identification



- Estimate of fit quality is important criterion for track selection
  - for an individual measurement use the filtered residual

$$r_k^k = m_k - \mathbf{H}_k \vec{\lambda}_k^k$$

$$\mathbf{R}_k^k = cov(r_k^k) = \mathbf{V}_k - \mathbf{H}_k \mathbf{C}_k^k \mathbf{H}_k^T$$

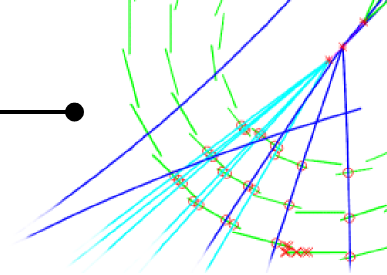
$$\chi_{kF}^2 = (r_k^k)^T (\mathbf{R}_k^k)^{-1} r_k^k$$

  - giving the total X^2

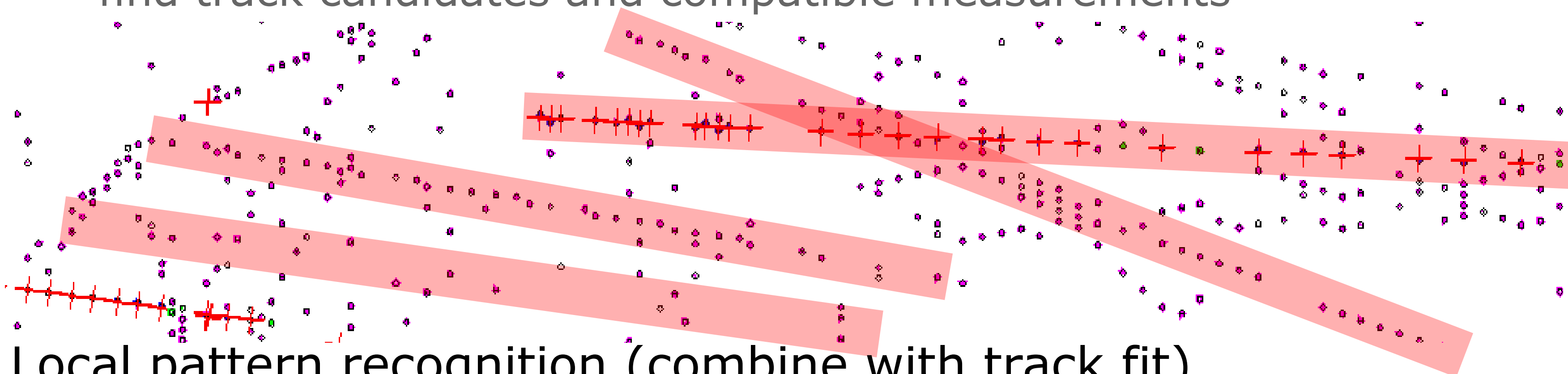
$$\chi_k^2 = \chi_{k-1}^2 + \chi_{kF}^2$$

- χ_{kF}^2 can be used to identify outliers, i.e. bad or wrongly assigned measurements

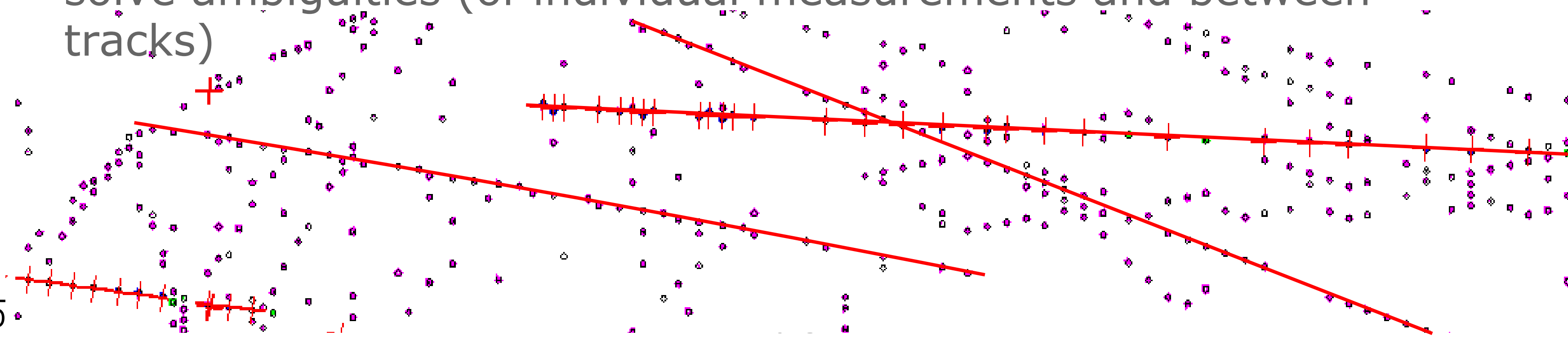
Pattern recognition and Track Finding



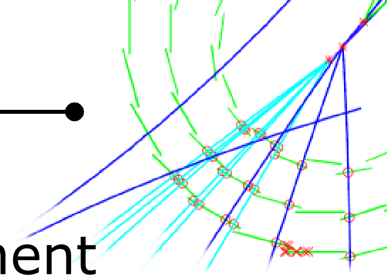
- Pattern recognition needed before track fit
  - Which measurements belong to a track?
  - How many tracks exist?
- Global pattern recognition (the “rough estimate”)
  - find track candidates and compatible measurements



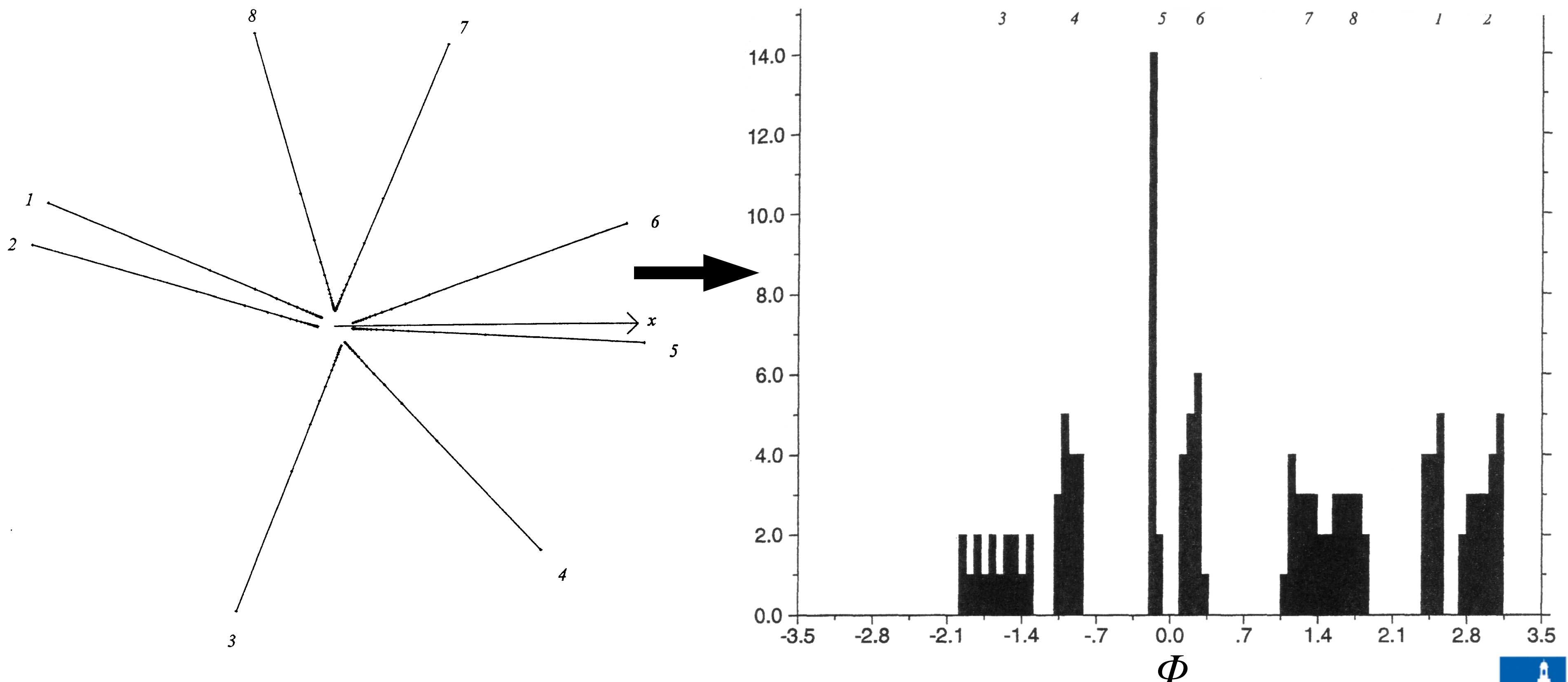
- Local pattern recognition (combine with track fit)
  - drop far-off measurements (outliers)
  - solve ambiguities (of individual measurements and between tracks)



Pattern recognition and Track Finding: Histogramming

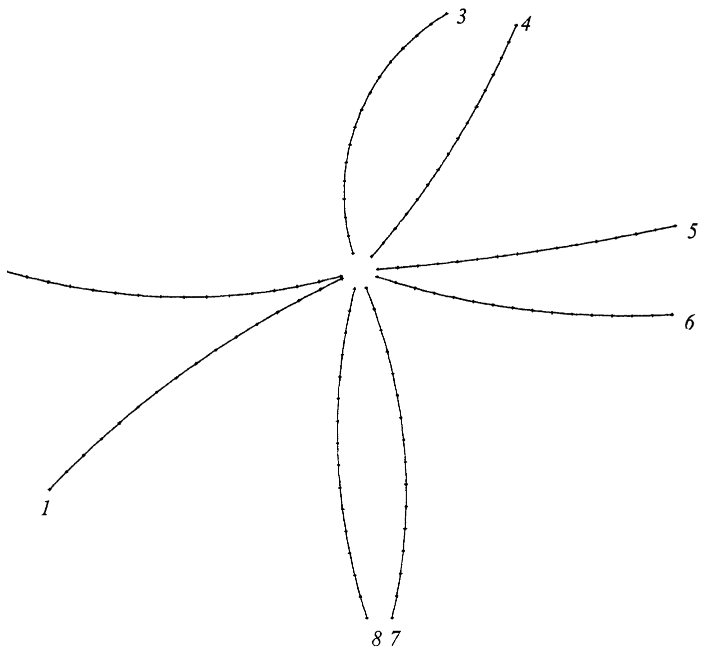


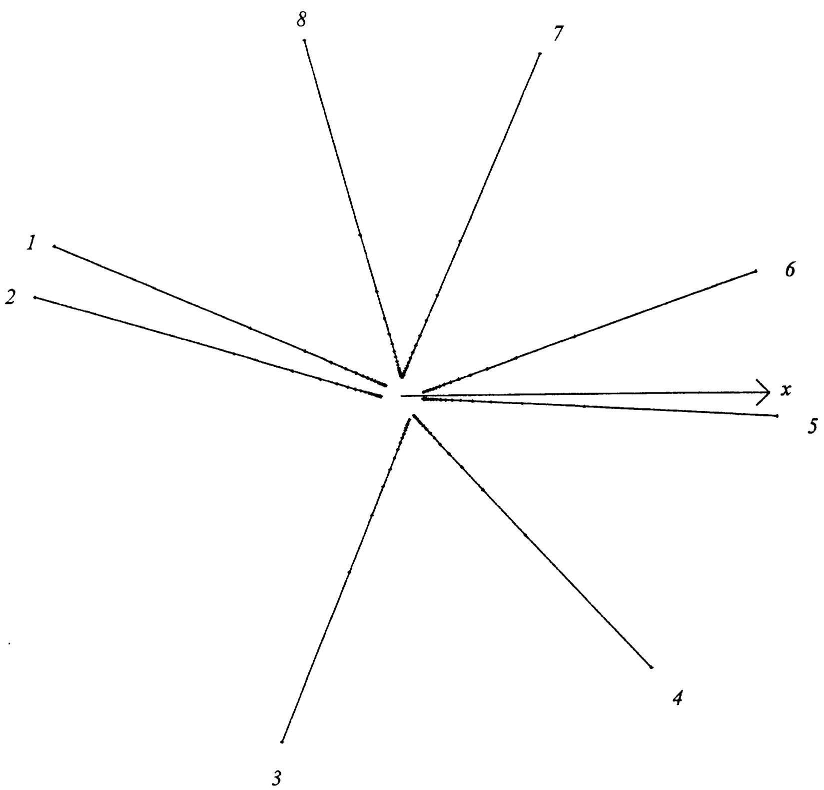
- Use appropiate coordinates to histogram measurement positions
- Peak search gives position/direction of track candidates



Pattern recognition and Track Finding: Histogramming

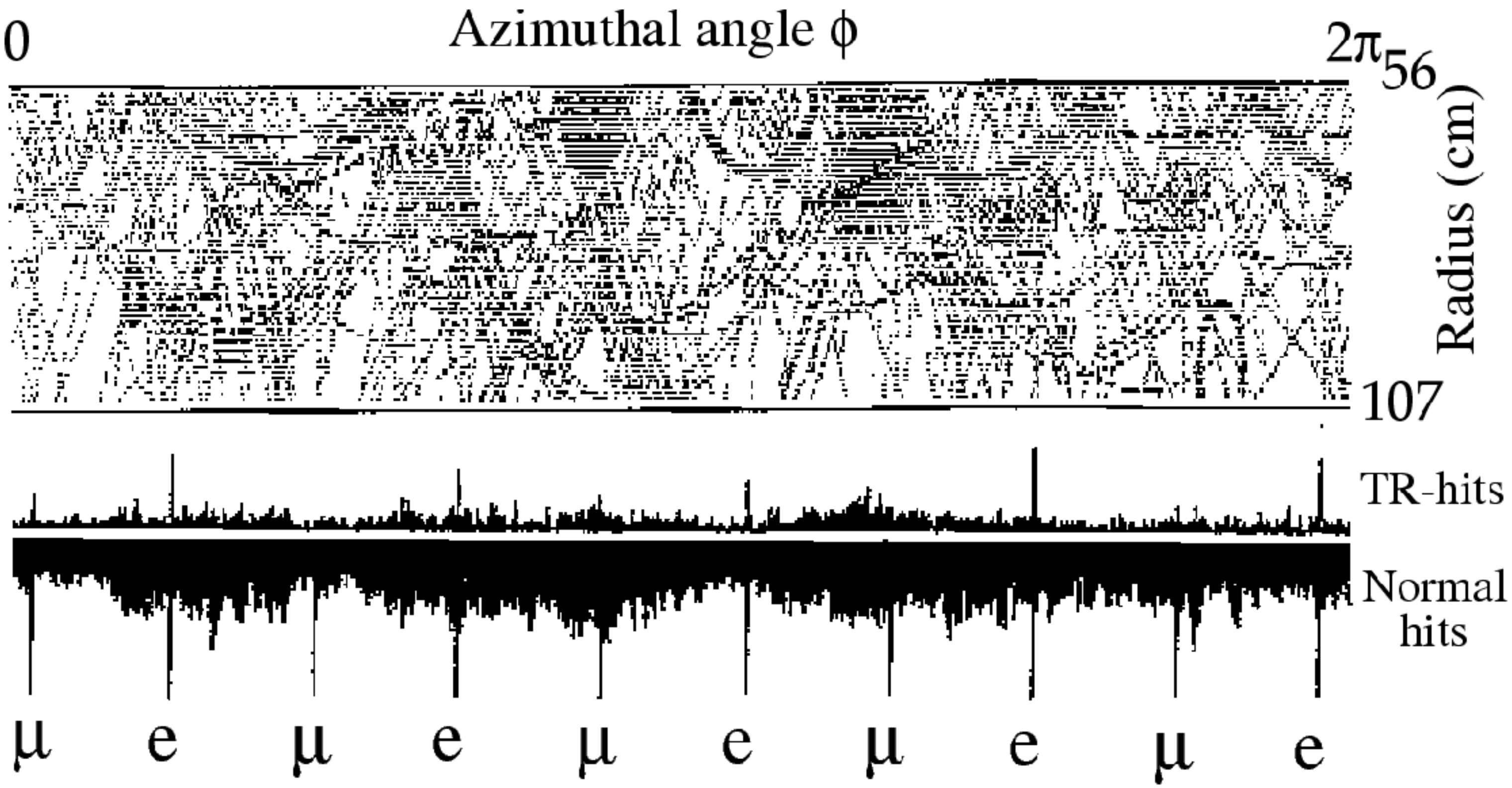
- Conformal transformation can be used for curved tracks





Pattern recognition and Track Finding: Histogramming

- Histogramming in the ATLAS TRT

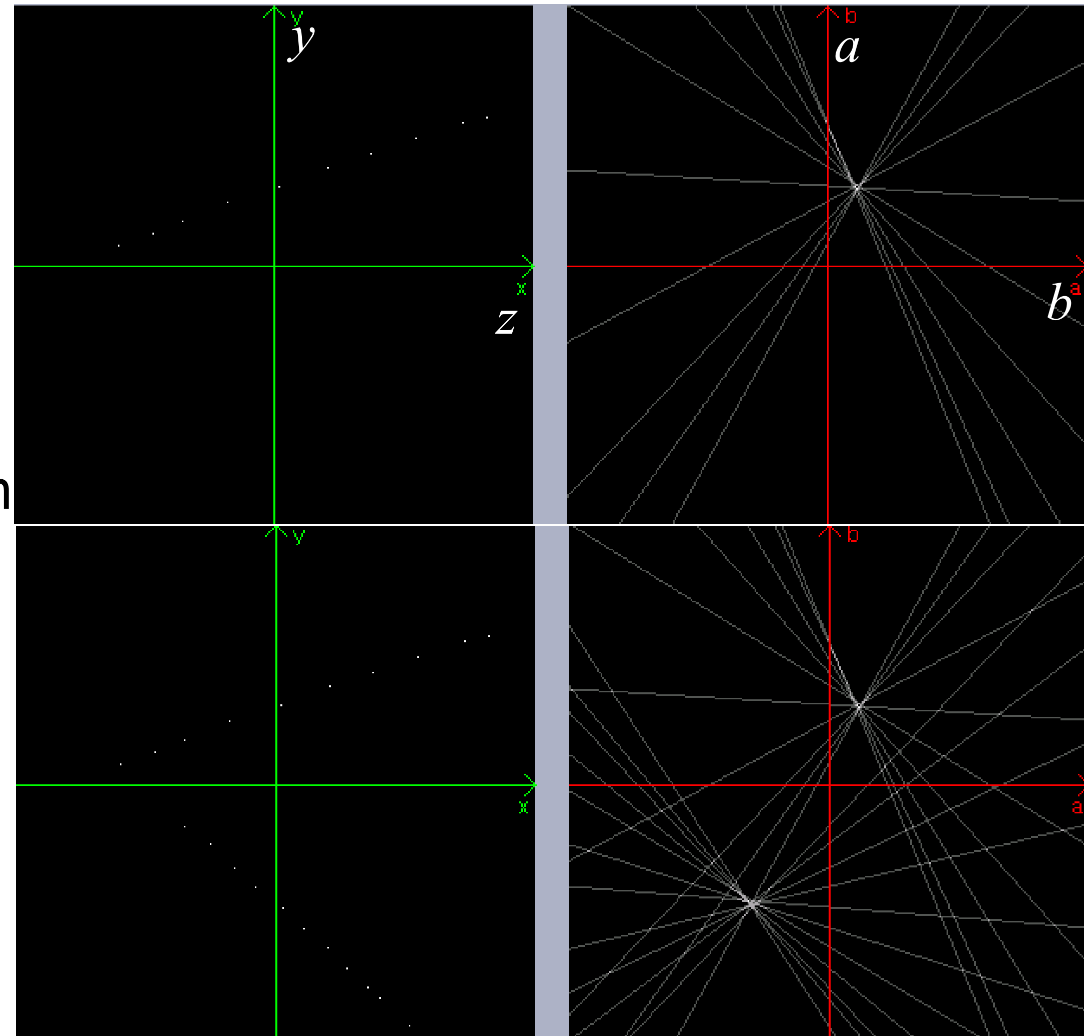


Pattern recognition and Track Finding: Hough transform

- Invert measurement function $\vec{f} : \vec{\lambda} \rightarrow \vec{m}$:
  - each measurement belongs to a hypersurface in parameter space
  - hypersurfaces intersect at "true" track parameters
- Find maxima in histogram to get seeds of track parameters

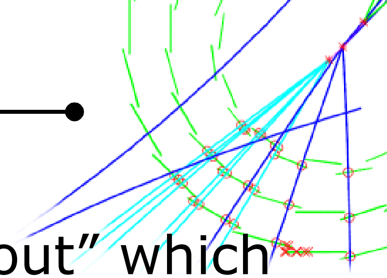
$$y = f(z) = a + bz$$

Pattern Space　　Parameter Space

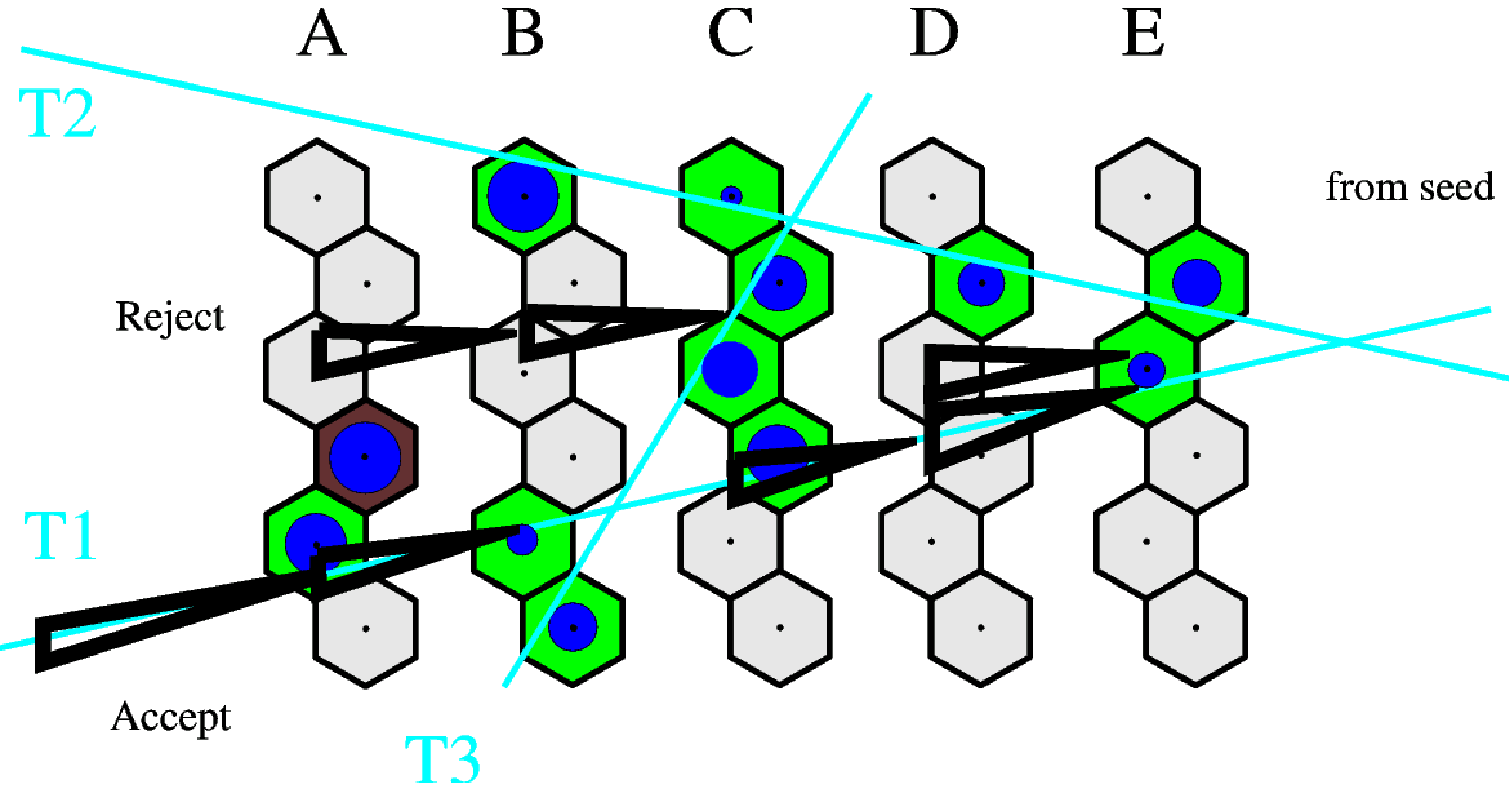


http://www.cs.tu-bs.de/rob/lehre/bv/Hough.html

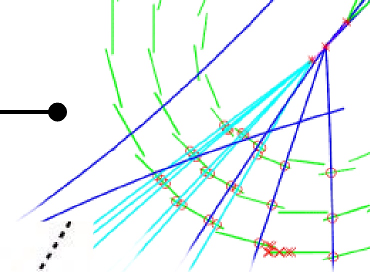
Pattern recognition and Track Finding:
Combinatorial Kalman Filter



- Properties of the Kalman Filter can be used to “try out” which measurement fits best

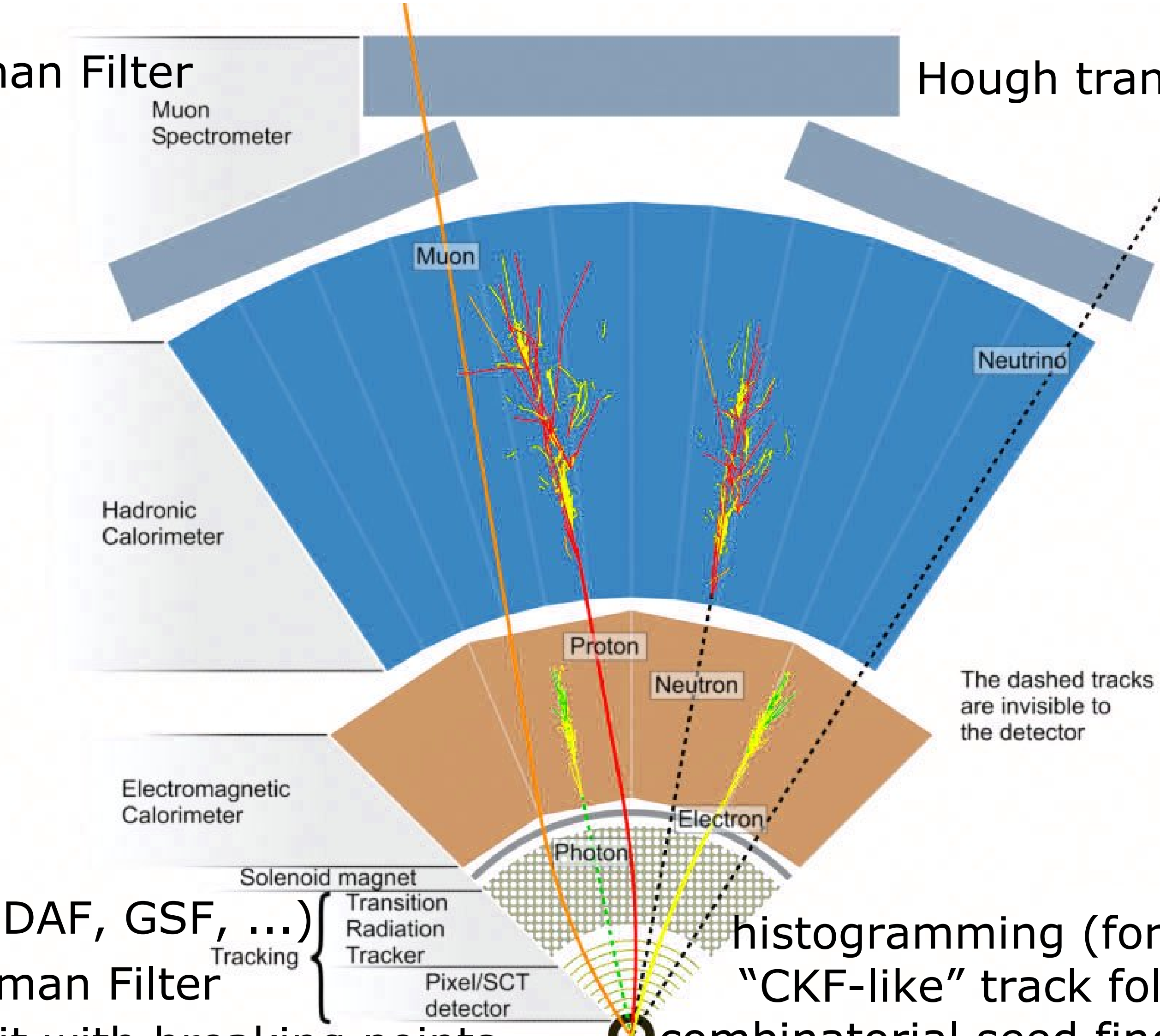


Track Finding and Track Fitting:
What's *really* used



Kalman Filter

Hough transform



(DAF, GSF, ...)

Kalman Filter

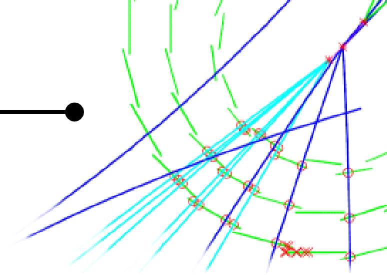
Global fit with breaking points

histogramming (for backtracking)

"CKF-like" track following

combinatorial seed finding

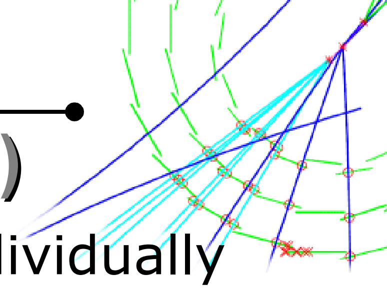
Summary



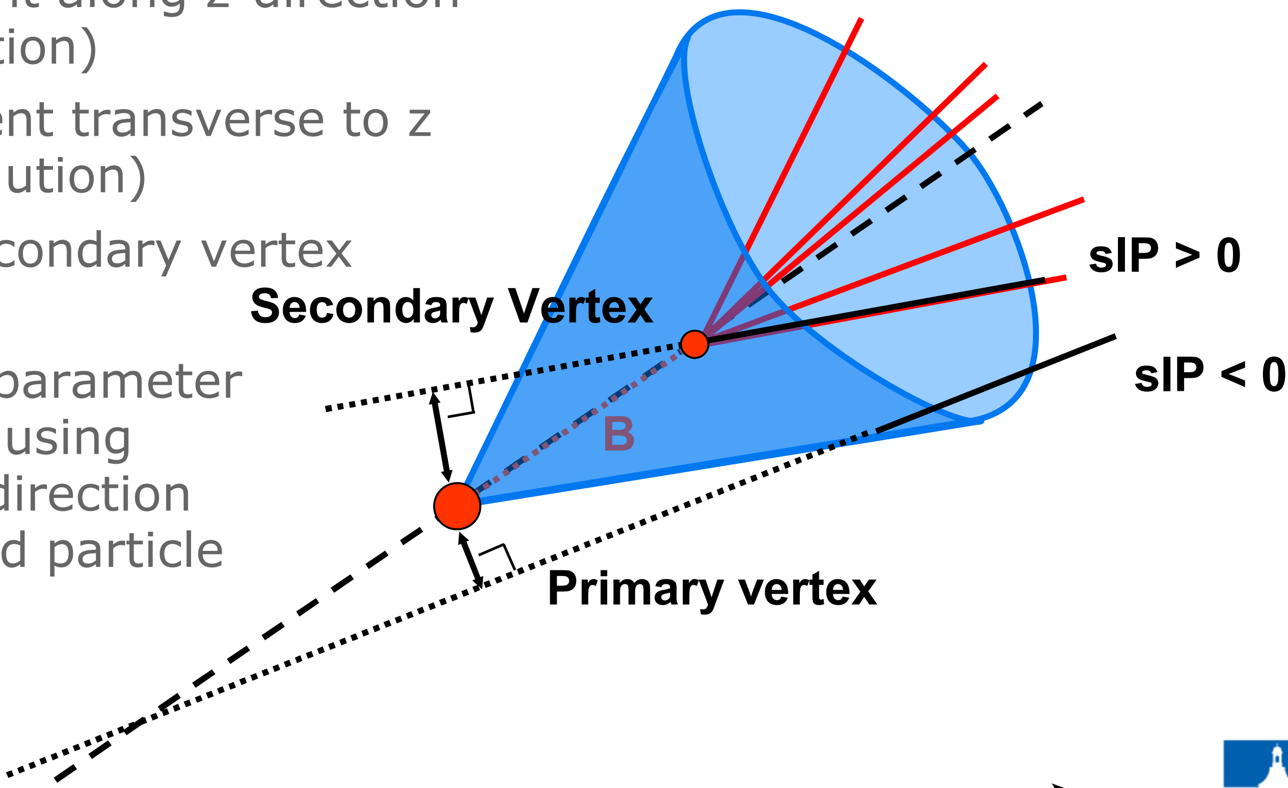
- Track reconstruction in HEP experiments requires
  - track finding (pattern recognition)
  - estimation of track parameters
- Is a very complex task due to high track density in modern experiments, though it is important to
  - allow for precise momentum measurements
  - identify long-lifed particles and multiple vertices
- Pattern recognition methods:
  - Global
  - Local (track-wise) in combination with track fits
- Track parameter estimation
  - Multiple scattering, etc. straightforward incorporated by Kalman Filter-based algorithms

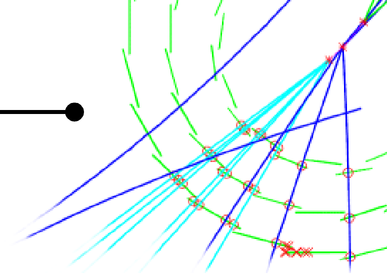
Backup

Introduction:
Lifetime tagging using Impact Parameter (IP)



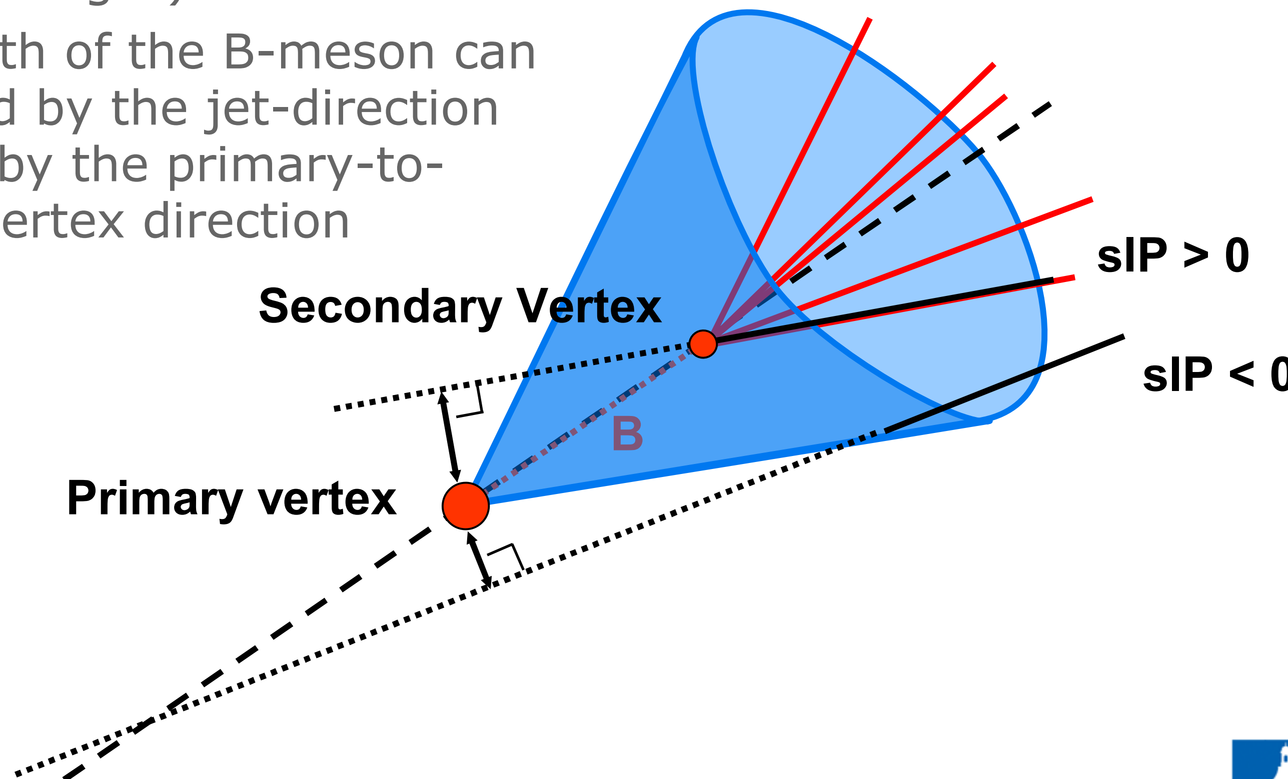
- Impact parameter can be defined for each track individually by the distance between reconstructed track and primary vertex
  - this distance is often split into two independent components because of unequal detector resolutions in those directions:
    - Rz component along z-direction (beam direction)
    - RΦ component transverse to z (higher resolution)
  - Tracks from secondary vertex have larger IP
  - sign of impact parameter can be defined using the estimated direction of the long-lived particle

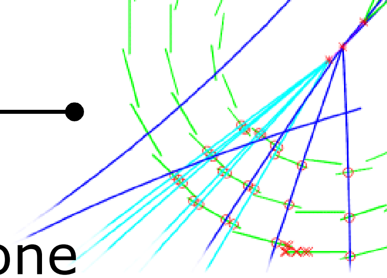




Tagging variables: Impact Parameter (IP) sign definition

- Sign of the impact parameter can be defined
  - using only the track directions ("geometrical sign")
  - including the estimated flight path of the B-meson by calculating the point of closest approach between track and B flight path and separating between upstream and downstream points ("lifetime sign")
    - the flight path of the B-meson can be estimated by the jet-direction or (better!) by the primary-to-secondary-vertex direction





Track Fits:
Kalman Filter

- Track fit can be performed recursively, i.e. adding one measurement after another to the estimate of track parameters