GENFIT - a Generic Track-Fitting Toolkit

Johannes Rauch [1]   Tobias Schlüter [2]

[1] Physik Department E18, Technische Universität München

[2] Ludwig-Maximilians-Universität München

ACAT 2014

Outline

Track Fitting - GENFIT
- Overview
- Design of GENFIT

Track Fitting Algorithms
- Kalman Filter
- Kalman Filter with Reference Track
- Deterministic Annealing Filter

Summary

GENFIT

GENFIT - A Generic Track Reconstruction Toolkit

What is GENFIT?

- Modular track-fitting framework.
- Suitable for a wide variety of experiments and detectors.
- Interface to vertex-fitting-framework RAVE.
- Interface to alignment-code MILLIPEDE II.
- Open source `C++` code (LGPL v3, `http://sourceforge.net/projects/genfit/`).

History and Status

- Originally developed in the `PandaROOT` framework at TUM (C. Höppner, S. Neubert et al., NIMA 620, 2-3, 1121 Aug. 2010, P. 518-525).
- Major update (“GENFIT2“) based on experience gained esp. in Belle II.
- Large user community (e.g. Belle II, PANDA, GEM-TPC, FOPI, ...).

Design of GENFIT

Modular Design

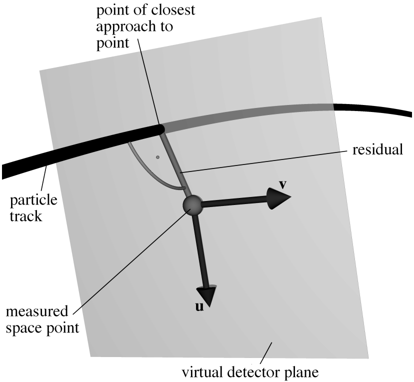
- Measurements
  - E.g. strip-, pixel-, wire-, spacepoint-measurements.
  - Provide (virtual) detector planes and measurement coordinates and covariance projected into that plane.
- Track representations (“TrackReps”)
  - Track parametrization.
  - Extrapolation through material and magnetic field.
  - Particle hypothesis.

Track

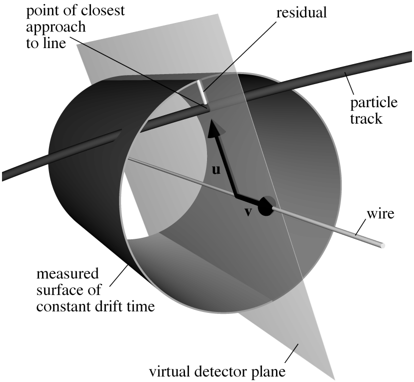
- Contains measurements (can be from different detectors).
- Can be fitted with several TrackReps simultaneously (esp. particle hypotheses).

- Track fitting algorithms
  - Use measurements and TrackReps to calculate fit results.
  - Start value for fit needed, e.g. from pattern recognition.

Virtual Detector Planes



Spacepoint measurement
(e.g. from TPC).



Wire measurement
(e.g. from drift-chamber or STT).

Track Fitting Algorithms

Kalman Filter

Algorithm

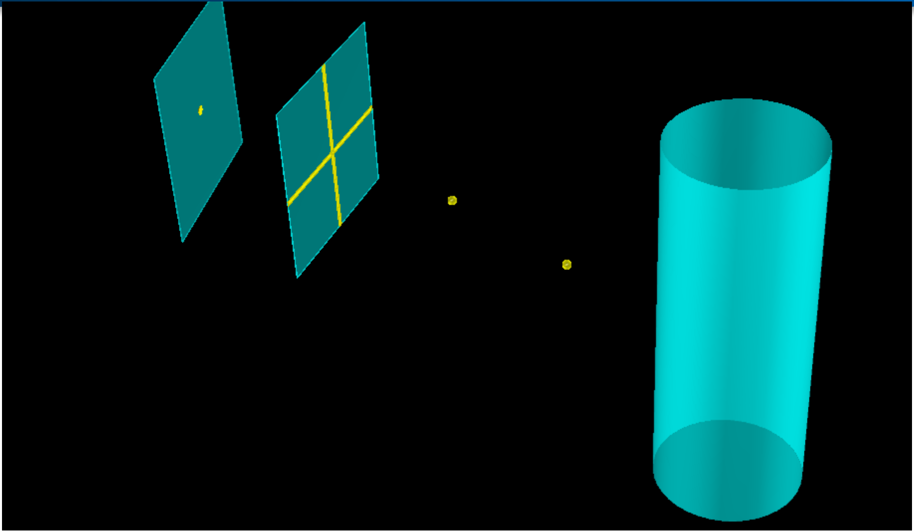
- Iterative algorithm to produce an optimal estimate of a system state (with covariance) from a series of noisy measurements.
- Prediction step: extrapolate state and covariance to next measurement.
- Update step: Calculate a “weighted average” between prediction and measurement.

Iterate over measurements, forth and back, until converged.

Linearization

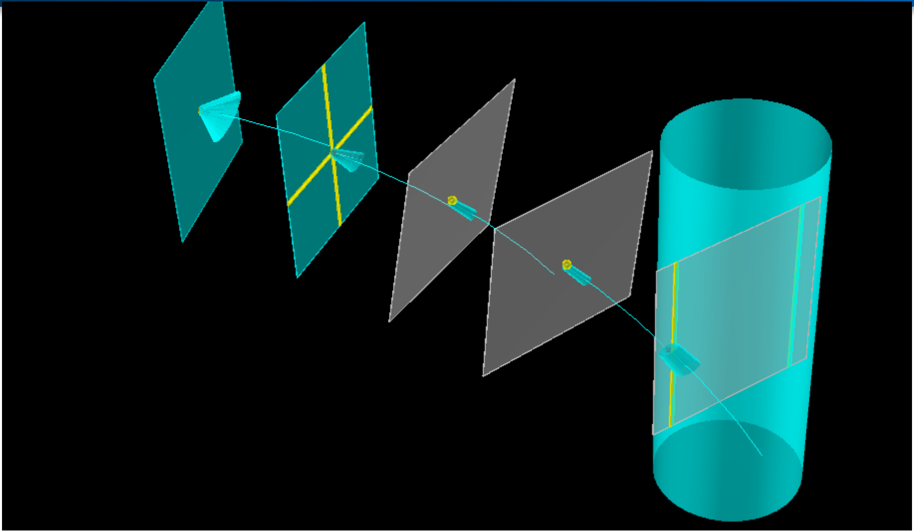
- Kalman filter is a linear estimator $\rightarrow$ need to linearize transport.
- Expansion point: e.g. state prediction.

Kalman Filter



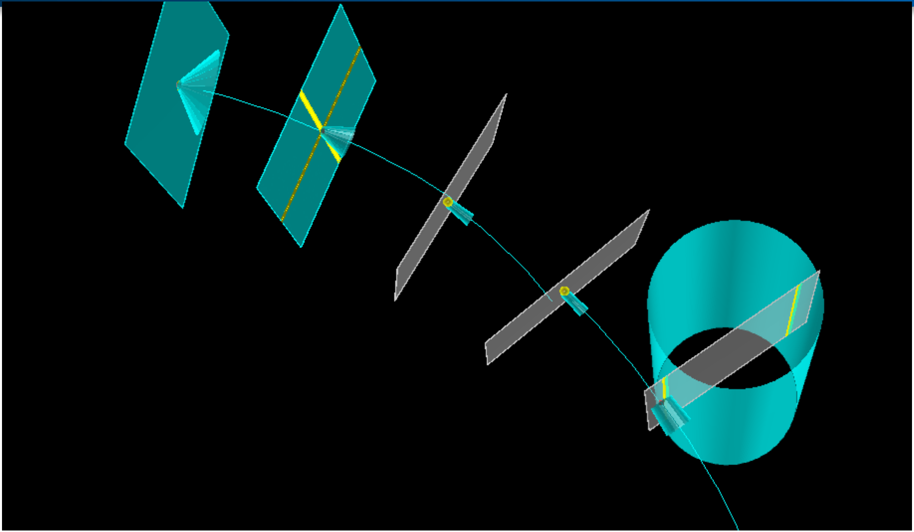
Series of noisy measurements.

Kalman Filter



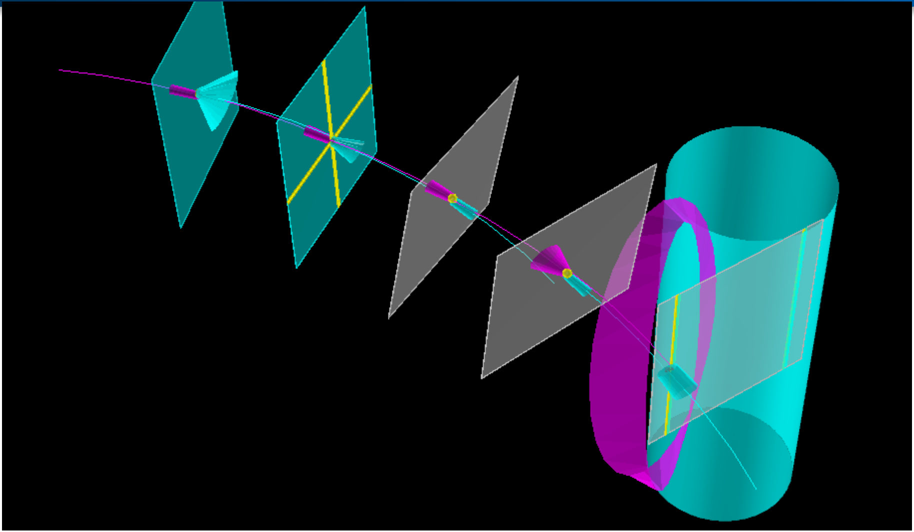
Forward fit with virtual detector planes.

Kalman Filter



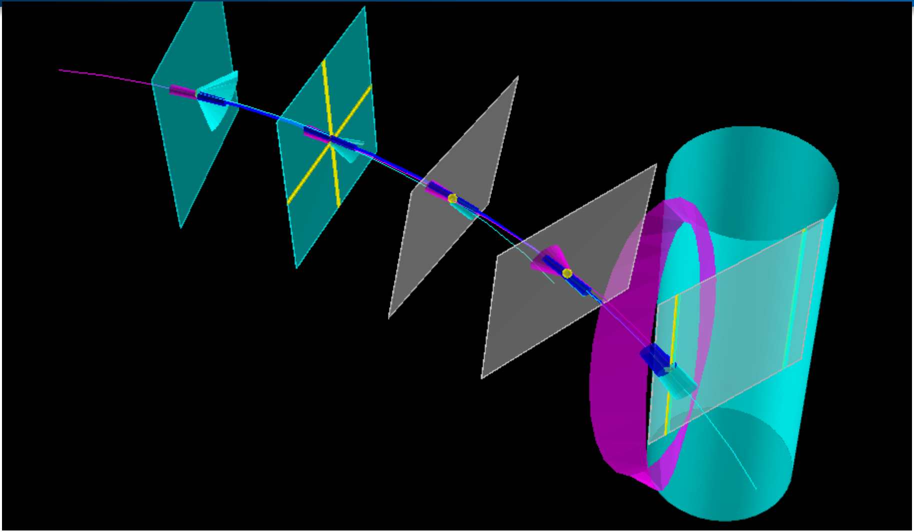
Forward fit.

Kalman Filter



Backward fit.

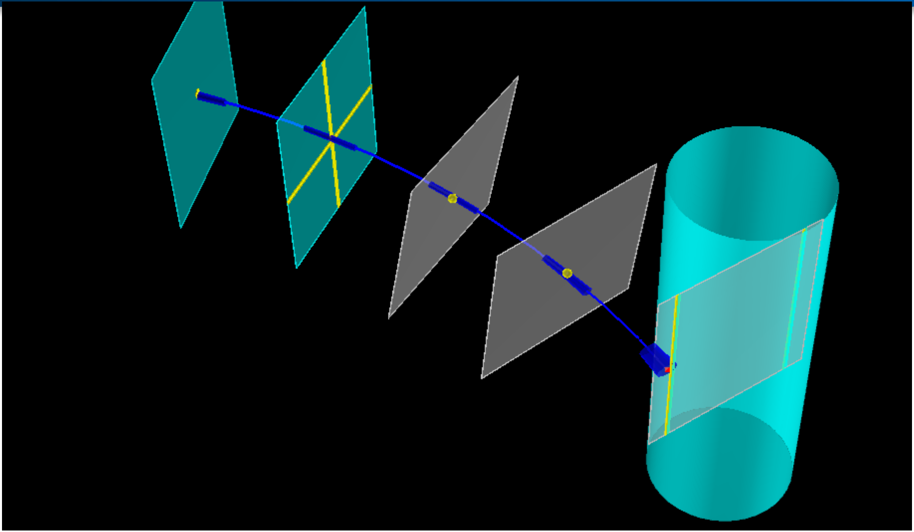
Kalman Filter



Smoothed track.

Kalman Filter

Technische Universität München



Smoothed track.

Kalman Filter with Reference Track

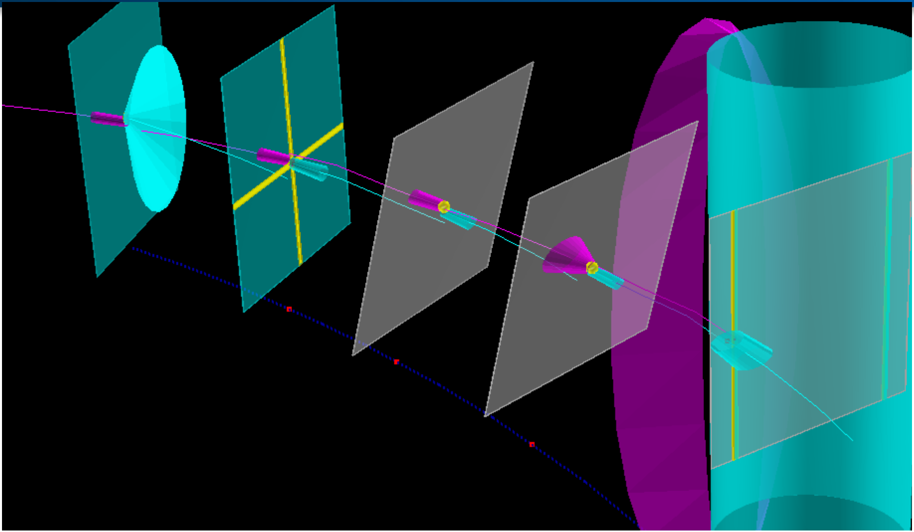
Problems when linearizing around predictions

- Especially for the first few hits, state predictions can be far off the actual trajectory.
- Outliers can bend the prediction away from the actual trajectory.
- Consequences:
  - Linearization makes no more sense.
  - Wrong material lookup.
  - The fit can fail (track can be bent so far from the actual trajectory that detectors with hits can no longer be reached).

Solution: reference track

- Take estimated track parameters from pattern recognition or previous fit as expansion point for linear approximation.
- I.e. linearize around reference track instead of state predictions.

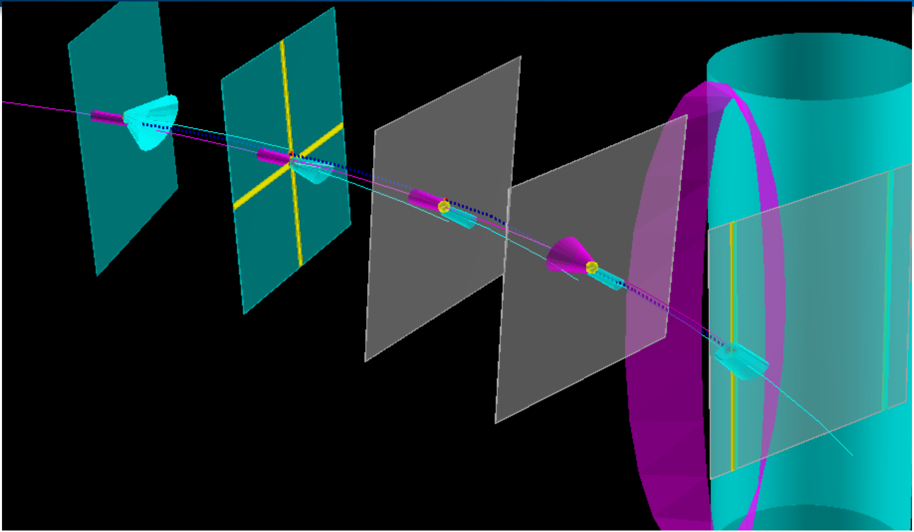
Kalman Filter with Reference Track



1st iteration.

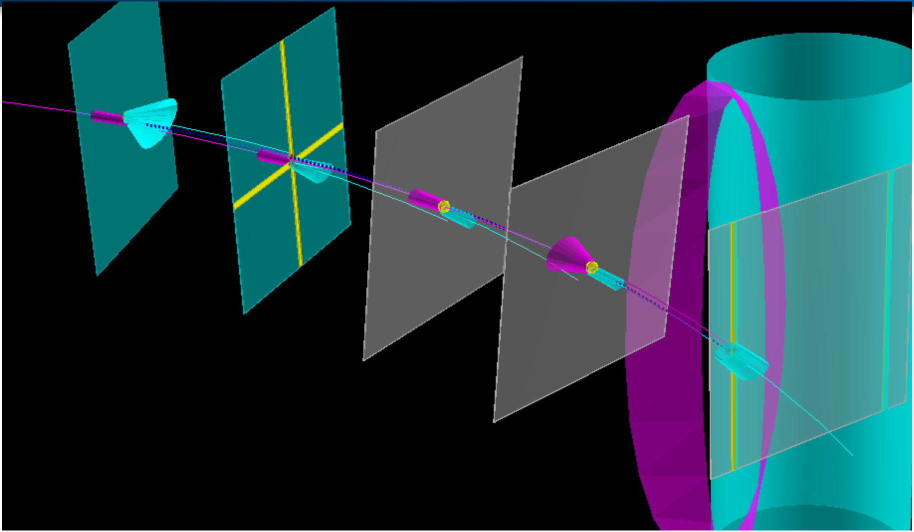
Kalman Filter with Reference Track

Technische Universität München



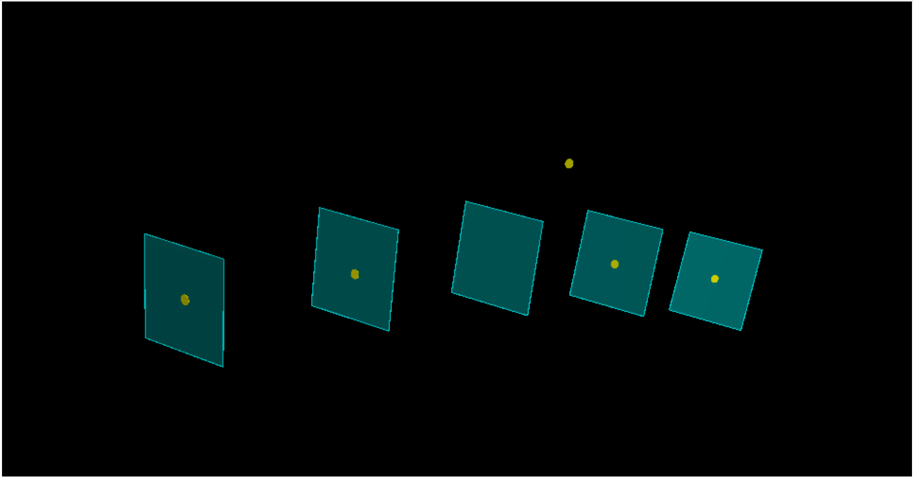
2nd iteration.

Kalman Filter with Reference Track



3rd iteration.

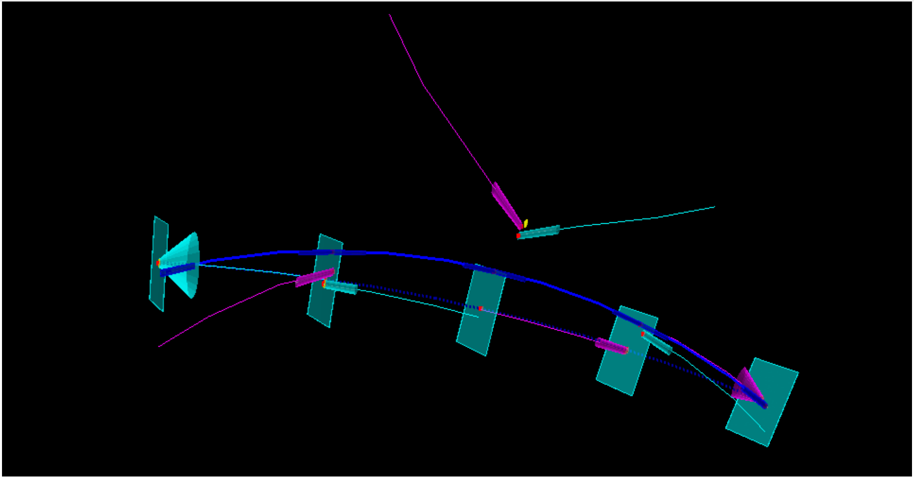
Outlier



But what happens if we have an outlier in the track (here in layer 3)?

Outlier - Fitted with the Kalman

Technische Universität München



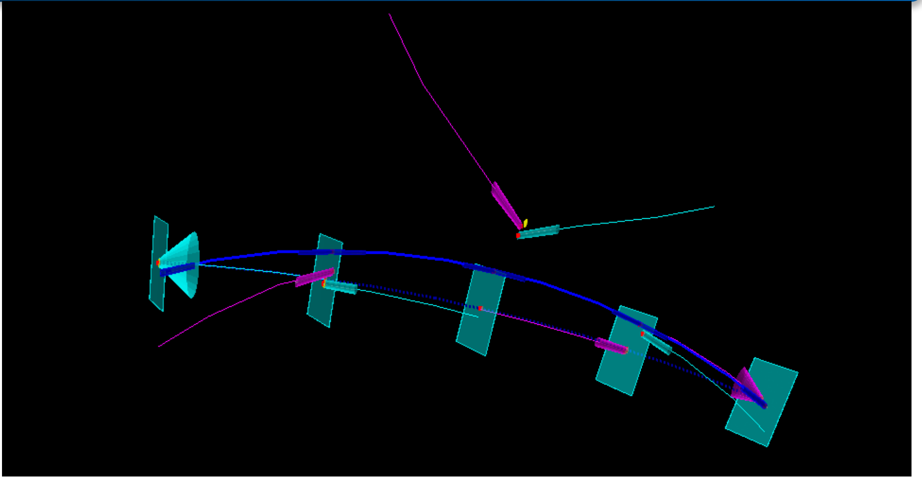
The outlier strongly biases the fit.

Deterministic Annealing Filter

DAF

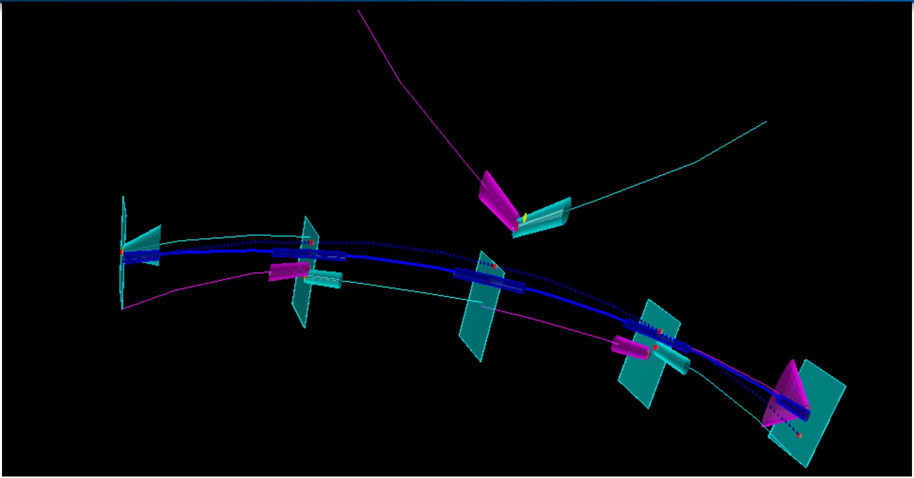
- Robust track fitter.
- Produces assignment probabilities (weights) of measurements.
- Iterative Kalman filter with weighting and annealing to find best fit.
- Can e.g. be used to reject outliers or to resolve left/right ambiguities of wire-measurements.

Outlier - Fitted with the DAF



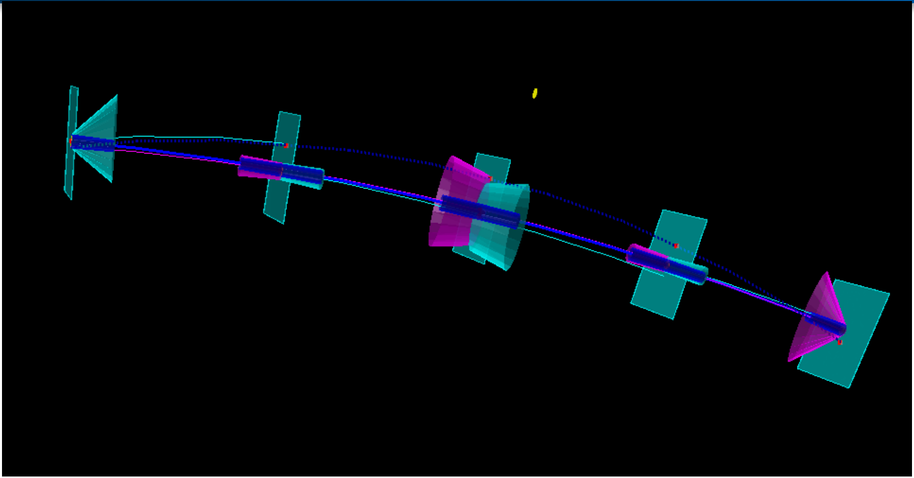
$\beta = 100$	initial weights:	1	1	1	1	1
$\log_{10}\beta = 2$	new weights:	0.4960	0.4238	0.1940	0.4310	0.5003

Outlier - Fitted with the DAF



$\beta = 17.78$	initial weights:	0.4960	0.4238	0.1940	0.4310	0.5003
$\log_{10} \beta = 1.25$	new weights:	0.5426	0.3640	6.052×10^{-6}	0.3913	0.5470

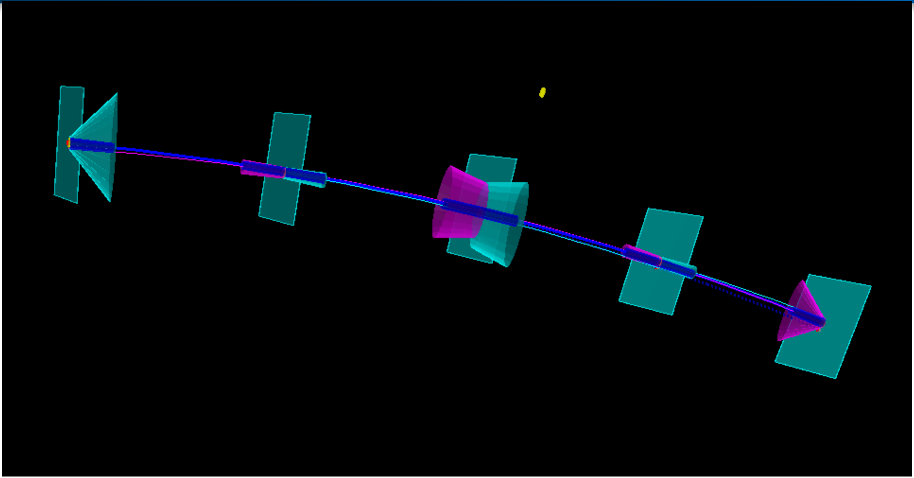
Outlier - Fitted with the DAF



$\beta = 3.162$	initial weights:	0.5426	0.3640	6.052×10^{-6}	0.3913	0.5470
$\log_{10}\beta = 0.5$	new weights:	0.8111	0.8093	4.106×10^{-52}	0.8099	0.8109

Outlier - Fitted with the DAF

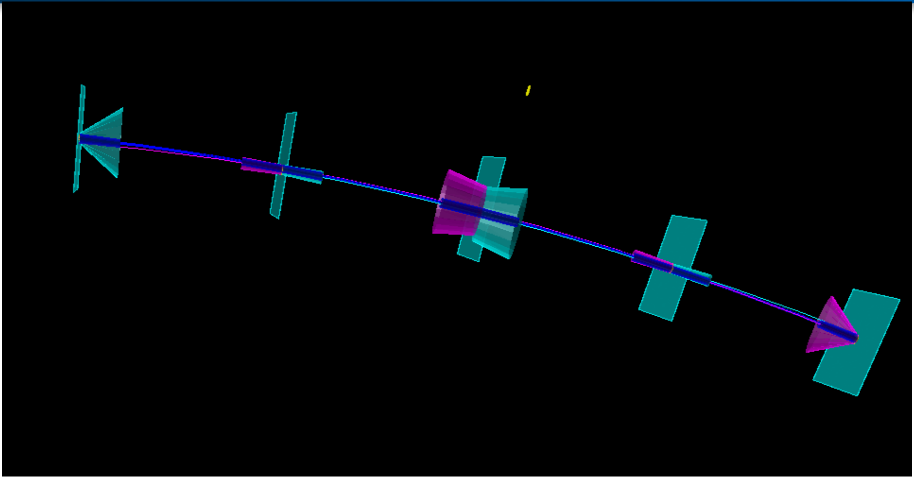
Technische Universität München



$\beta = 0.5623$	initial weights:	0.8111	0.8093	4.106×10^{-52}	0.8099	0.8109
$\log_{10} \beta = -0.25$	new weights:	0.9997	0.9997	1.725×10^{-290}	0.9997	0.1000

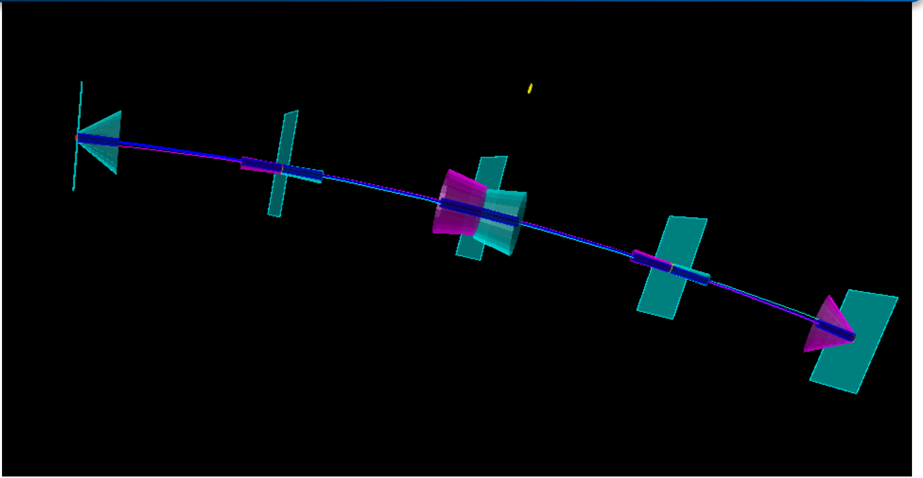
Outlier - Fitted with the DAF

Technische Universität München



$\beta = 0.1$	initial weights:	0.9997	0.9997	1.725×10^{-290}	0.9997	0.1000
$\log_{10} \beta = -1$	new weights:	1	1	0	1	1

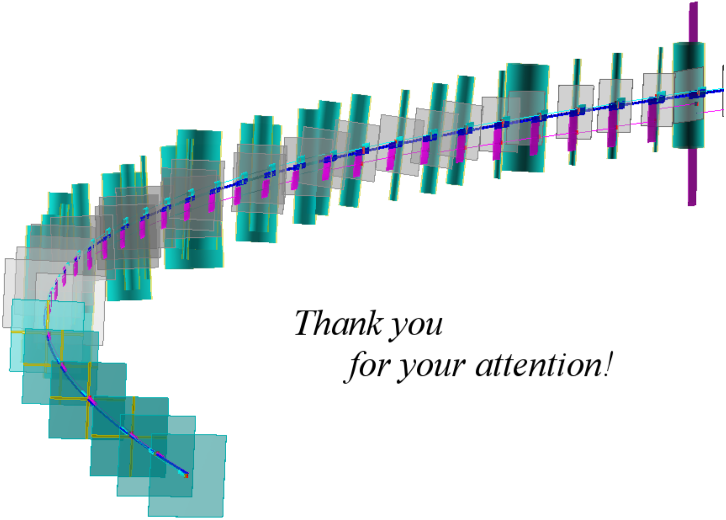
Outlier - Fitted with the DAF



$\beta = 0.1$	initial weights:	1	1	0	1	1
$\log_{10} \beta = -1$	new weights:	1	1	0	1	1

Summary

- Open-source, experiment-independent track-fitting framework.
- Validated code.
- Successfully running in various experiments.
- `http://sourceforge.net/projects/genfit/`



Thank you
for your attention!

Backup Slides

Kalman Filter Equations

Technische Universität München

Prediction:

$$p_{k|k-1} = F_k p_{k-1|k-1} + c_k$$

$$C_{k|k-1} = F_k C_{k-1|k-1} F_k^T + N_k$$

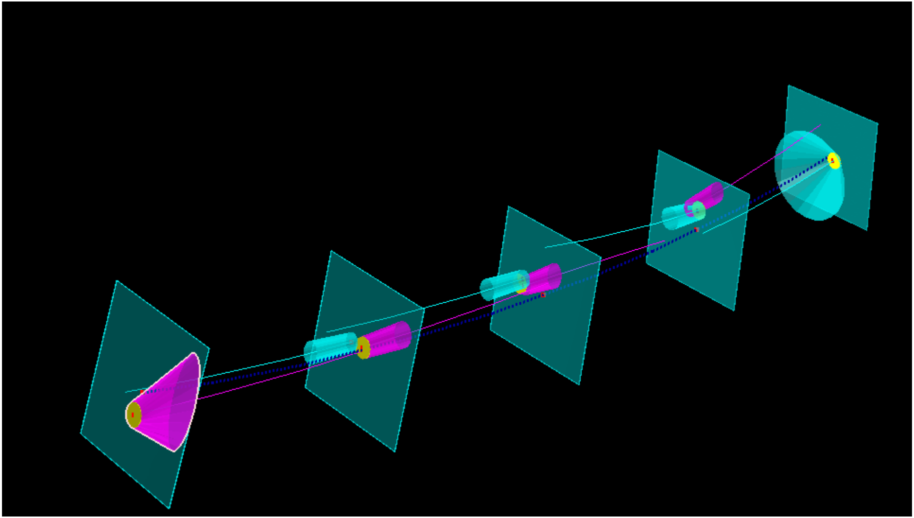
Update:

$$p_{k|k} = p_{k|k-1} + K_k \left(m_k - H_k p_{k|k-1} \right)$$

$$K_k = C_{k|k-1} H_k^T \left(V_k + H_k C_{k|k-1} H_k^T \right)^{-1}$$

$$C_{k|k} = \left(I - K_k H_k \right) C_{k|k-1}$$

Kalman Filter with Reference Track



GENFIT Data Structure

Technische Universität München