A Gaussian-sum Filter for vertex reconstruction

R. Frühwirth - HEPHY Vienna
T. Speer - University of Zurich

IX International Workshop on Advanced Computing and Analysis Techniques in Physics Research
Tsukuba
4th December 2003

Vertex reconstruction

- Standard tool for vertex reconstruction is the Kalman Filter (also implemented in the reconstruction software of the CMS experiment at LHC, CERN)
- The Kalman Filter is mathematically equivalent to a global least square minimization (LSM)
- If the model is linear and random noise is Gaussian:
  - LS estimators are unbiased and have minimum variance
  - Residuals and pulls of estimated quantities are also Gaussian
- For non-linear models or non-Gaussian noise, it is still the optimal linear estimator
- Non-Gaussian measurement errors degrade results!

The Gaussian-sum Filter

- Gaussian-sum Filter (GSF)

  Measurement error distributions modelled by mixture of Gaussians:

  - Main component of the mixture would describe the core of the distribution
  - Tails would be described by one or several additional Gaussians.

- First proposed by R. Frühwirth for track reconstruction

  (Computer Physics Communications 100 (1997) 1.)

- Successfully implemented in the CMS reconstruction software for electron track reconstruction:

  - Bethe-Heitler energy loss distribution modeled by a mixture of Gaussians

- GSF for vertex reconstruction now also implemented in the CMS reconstruction software.

The Gaussian-sum Filter for vertex reconstruction

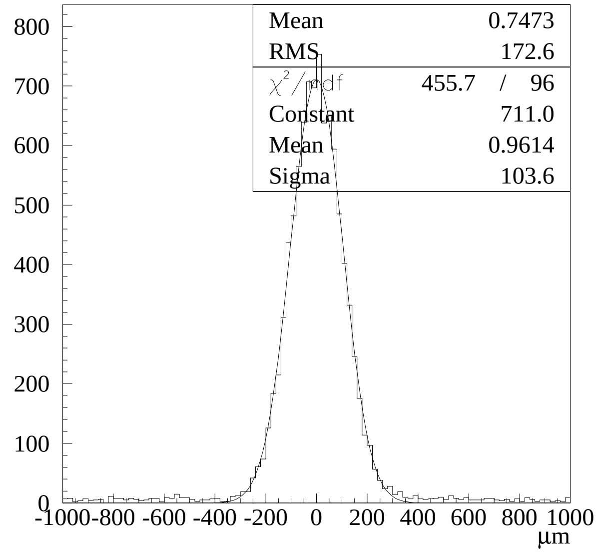
- Track parameter error distributions modeled by a mixture of Gaussians
- Vertex State vector x , is also distributed according to a mixture of Gaussians
- Iterative procedure: estimate of the vertex is updated with one track at the time
- Add new track to vertex, each component of the Vertex State is updated with each component of the track (Combinatorial combination of all track components)
- The new Vertex State x_k is therefore distributed according to a mixture of N_k $(= N_{track - k} * N_{vertex - k-1})$ Gaussians
- The filter is a weighted sum of several Kalman Filters
  - GSF is implemented as a number of Kalman filters run in parallel
  - The weights of the components are calculated separately
- Non-linear estimator: weights depend on the measurements

Simulation

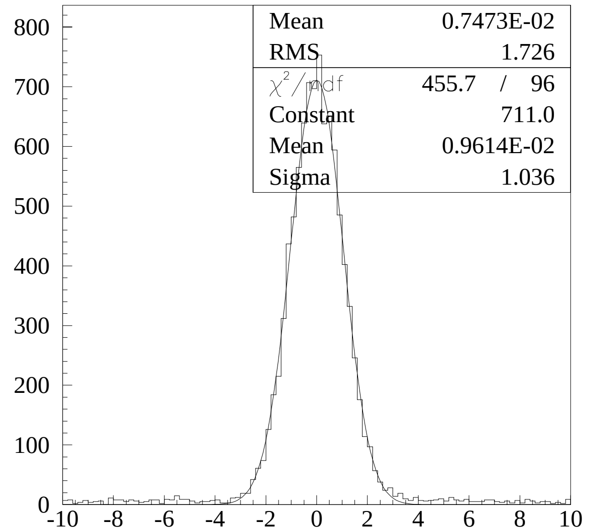
Simplified simulation in a fully controlled environment:

- Tracks generated at a common vertex
- No track reconstruction
- Track parameters are smeared according to known distributions:
- E.g. 2 component Gaussian mixture:
  - Narrow component: 90 % Relative weight
    (Standard deviation of Impact parameter = 100µm)
  - Wide component: 10 % Relative weight
    Std dev. 10x larger (Impact parameter = 1000µm)
    - ➻ Ratios of Standard deviation = 10

- For the Kalman Filter:
  - tracks smeared according to two-component mixture
  - single component used in the fit:
    - ➢ track parameter variance of dominating component
    - ➢ estimated position independent of scaling of variance (but not position uncertainty or χ^2)



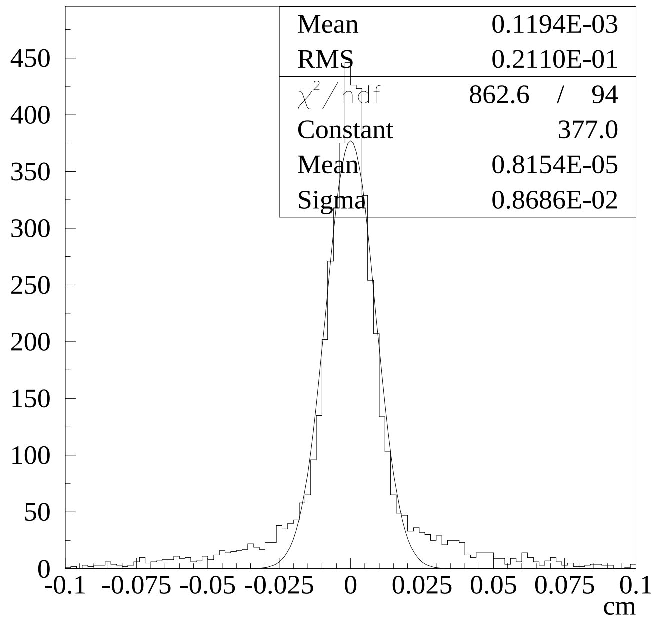
Transverse IP - residuals (µm)

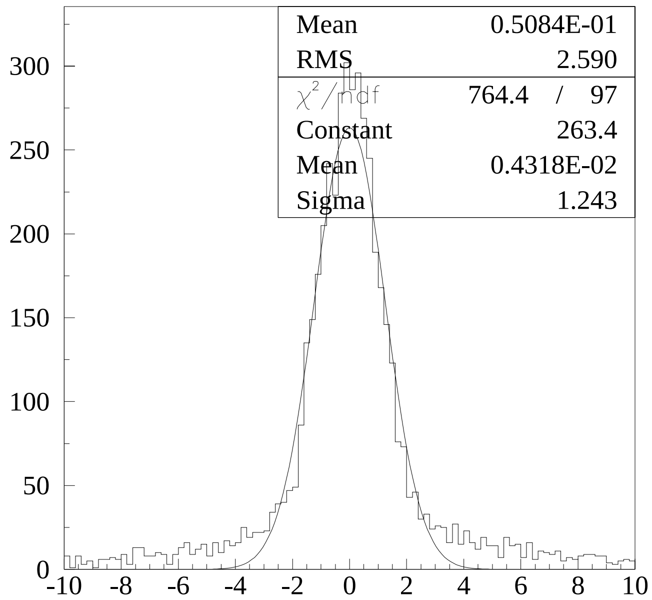


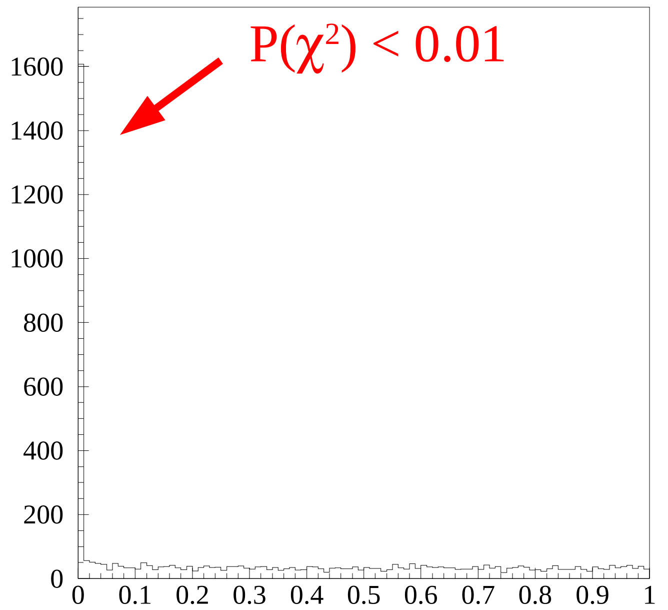
Transverse IP - pulls for Kalman Filter

Kalman Filter fit

Four track-vertex fit with the Kalman Filter:





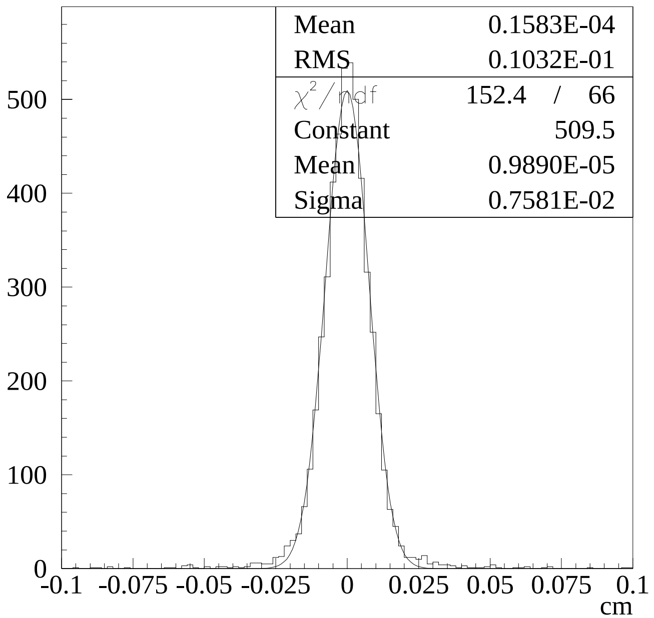


x Residuals (μm) *x* Pull $P(\chi^2)$

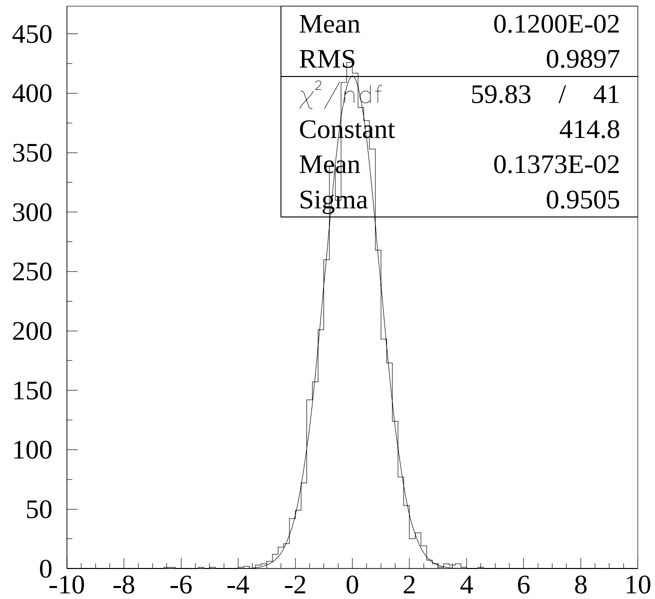
- Non-Gaussian tails in the distributions of residuals and pulls
- Large number of fits with $P(\chi^2) < 0.01$

Gaussian-sum Filter fit

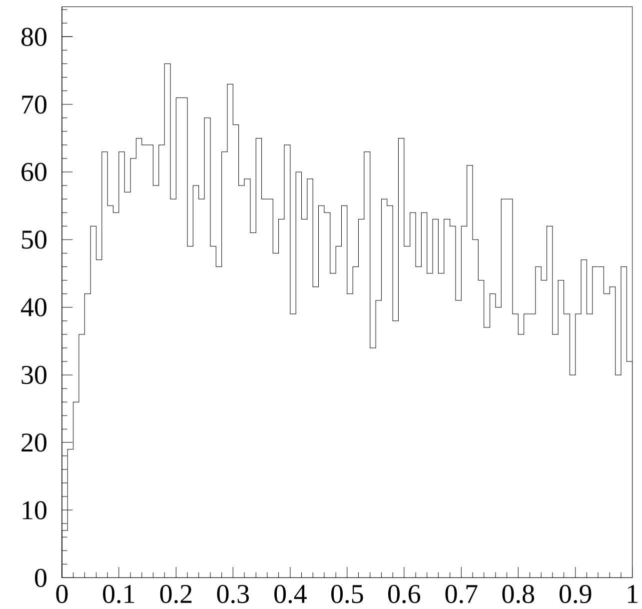
Four track-vertex fit with the GSF (using the full Gaussian mixture)



x Residuals (μm)



x Pull



$P(\chi^2)$

Residuals: smaller tails then with the Kalman Filter, smaller resolution

The remaining tails are due to events with several outliers.

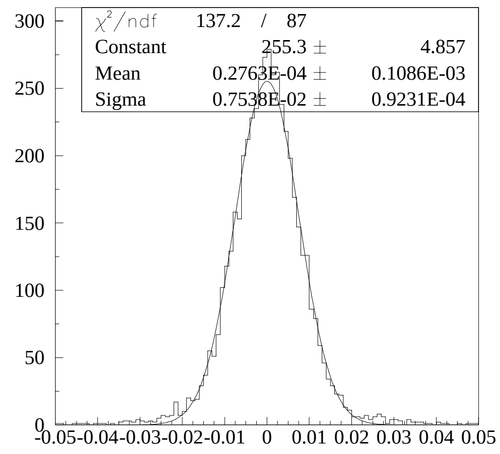
No outliers in the pull distributions: error on the outliers correctly taken into account

$P(\chi^2)$: dip at 0. - in early stages of the fit, bias towards components with a low χ^2

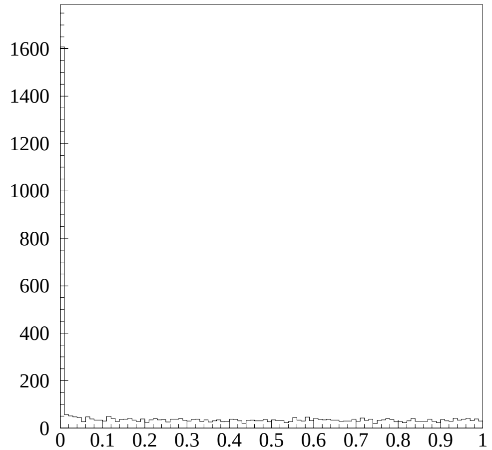
The filters needs several iterations (tracks) to stabilise and select the correct vertex component (combination of track components)

Measures of improvement of vertex fits

- Two-component Gaussian mixtures with different ratios of standard deviations and relative weights (4-track vertices)
- Measures:
  - ➢ 50% and 90% coverage: half-widths of the symmetric intervals covering 50% and 90% of the residual distribution (*x*-coordinate)
  - ➢ Relative efficiency: ratio of the mean (3D) distances of the estimated vertex from its simulated position, for fits with the Kalman Filter and the GSF
  - ➼ For Kalman Filter: estimated position independent of scaling of track parameter variance
  - ➢ Fraction of Kalman Filter fits with $P(\chi^2) < 0.01$
  - ➼ For Kalman Filter: estimated uncertainty dependent of scaling of track parameter variance

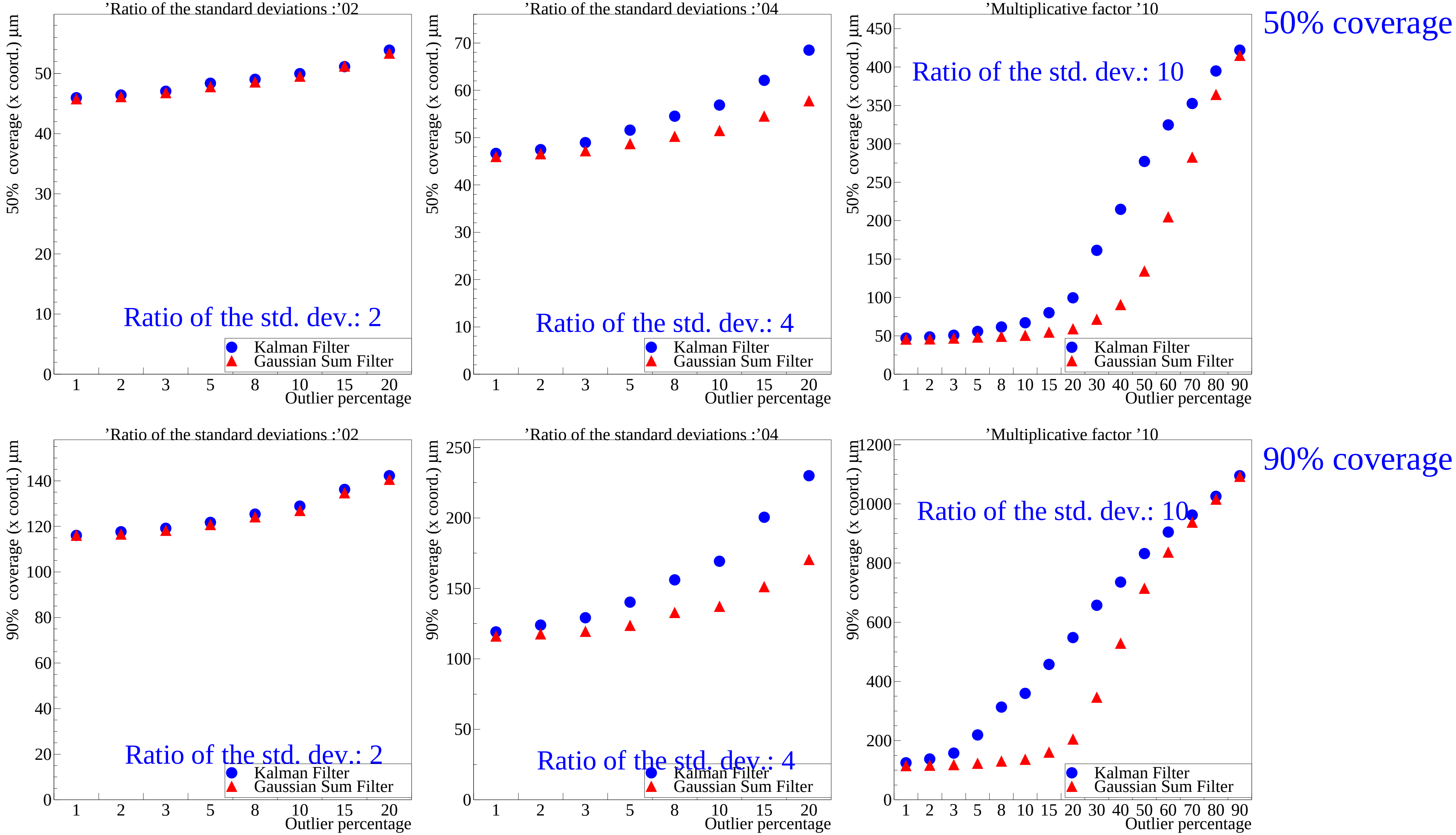


x Residuals (μm)

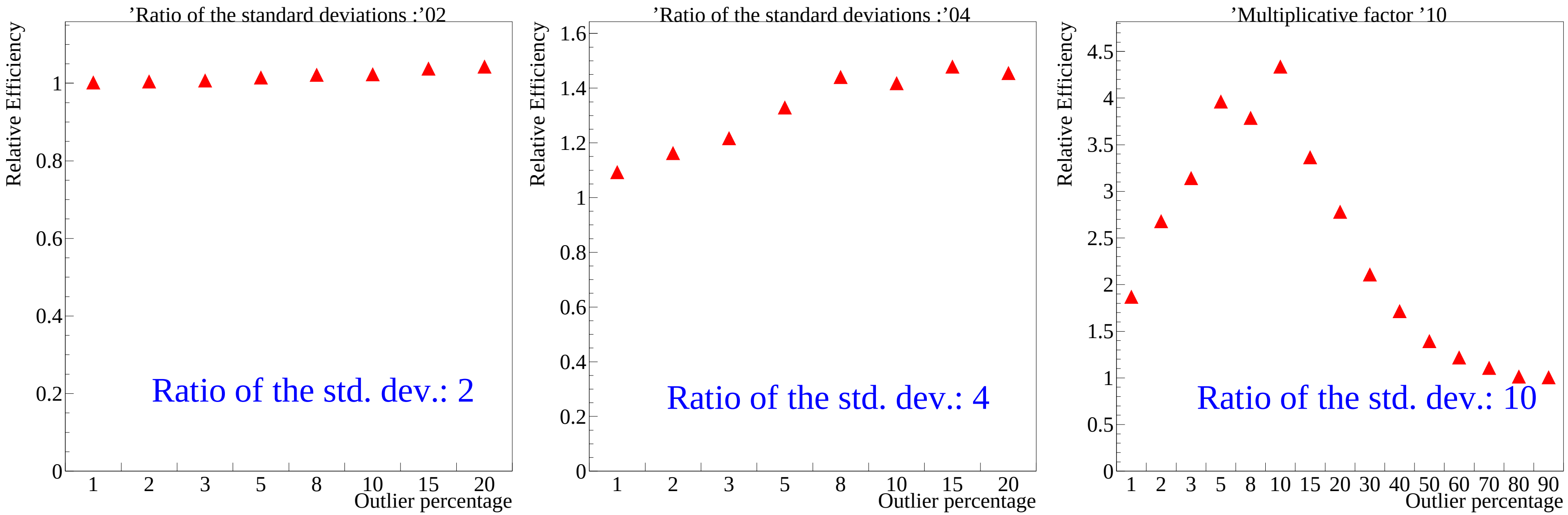


$P(\chi^2)$

Coverage


'Ratio of the standard deviations :'02
'Ratio of the standard deviations :'04
'Multiplicative factor '10
50% coverage (x coord.) µm
90% coverage (x coord.) µm
Outlier percentage
Kalman Filter
Gaussian Sum Filter
Ratio of the std. dev.: 2
Ratio of the std. dev.: 4
Ratio of the std. dev.: 10
50% coverage
90% coverage

Relative efficiency



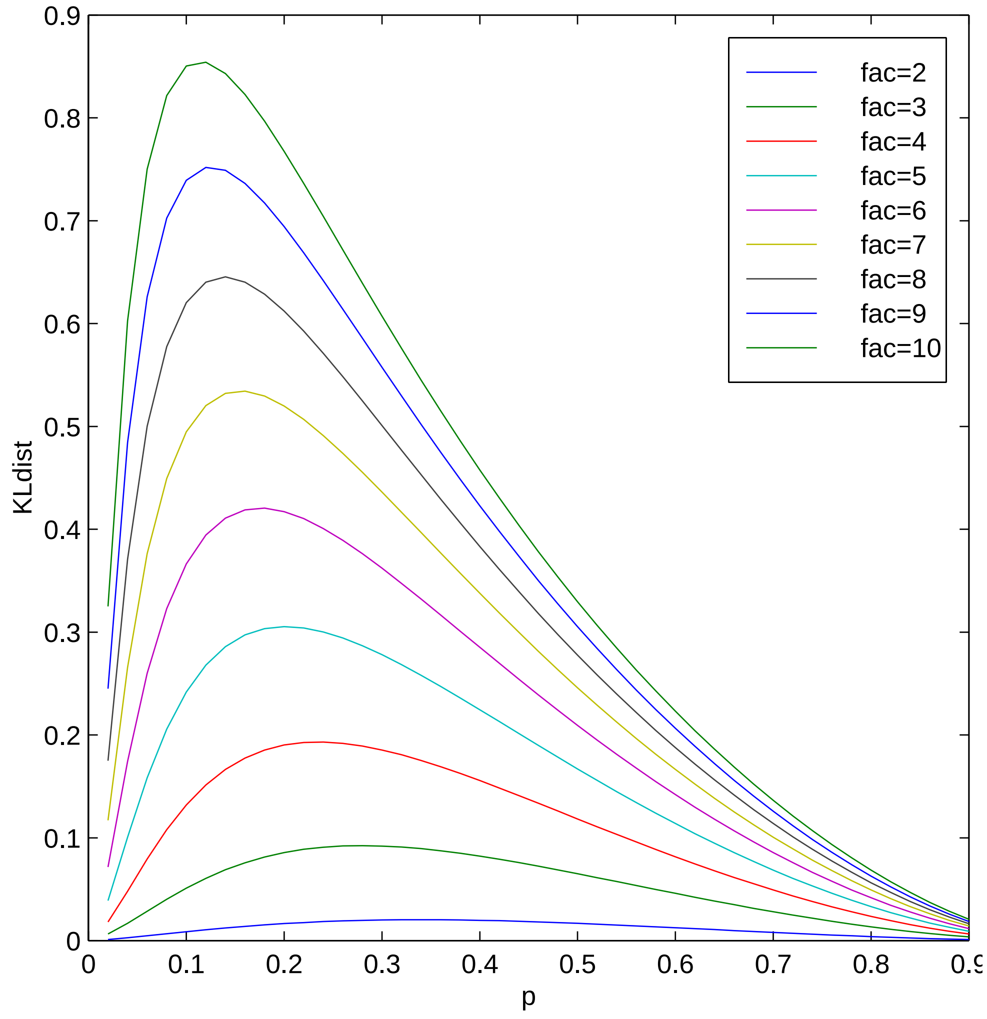
Relative efficiency: ratio of the mean distances (in three dimensions) of the estimated vertex from its simulated position, for fits with the KVF and the GSF

- Highest relative efficiency: largest distance between the two-component Gaussian mixture and the single Gaussian
- Larger weight of the tails: tails start to dominate ➵ lower relative efficiency

Relative efficiency

Kullback-Leibler Distance between a two-component Gaussian mixture and single-Gaussian distribution with identical moments:

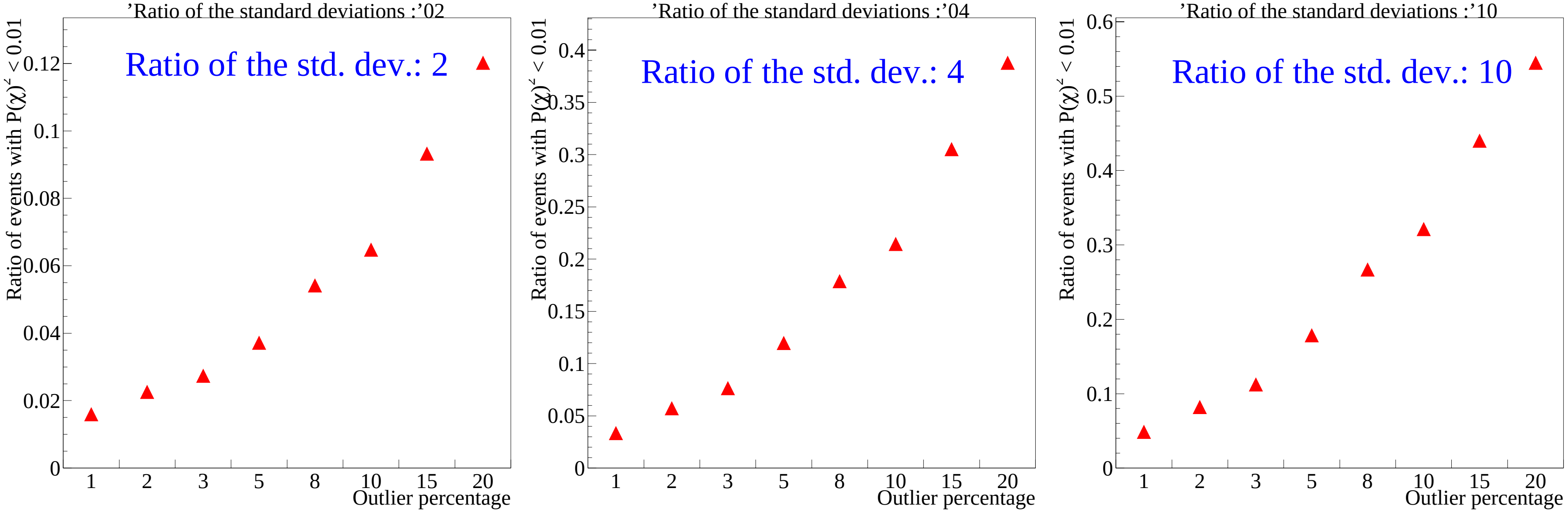
$$D_{KL}(p_1, p_2) = 2 \cdot \left(\int_{-\infty}^{\infty} \ln\left(\frac{p_1}{p_2}\right) p_1\, dx + \int_{-\infty}^{\infty} \ln\left(\frac{p_2}{p_1}\right) p_2\, dx \right)$$



p: relative weight of the second Gaussian

f: ratio of their standard deviations

$P(\chi^2)$



Fraction of Kalman Filter fits with $P(\chi^2) < 0.01$

- Estimated uncertainty dependent of scaling of track parameter variance

Component limitation

The number of components increases exponentially:

- n measurements, with m components: n^m components at the end!
  - Combinatorial explosion!
- Keep only M components at each step:
  - Keep components with the largest weight, discard the rest
  - Cluster (collapse) components with the smallest 'distance'

    2 Distance measurements were used:

    – Kullback-Leibler Distance

$$D_{KL}(p_1, p_2) = tr\left[\left(V_1 - V_2\right)\left(V_1^{-1} - V_2^{-1}\right)\right] + \left(\mu_1 - \mu_2\right)^T \left(V_1^{-1} + V_2^{-1}\right)\left(\mu_1 - \mu_2\right)$$

Mean

Standard deviation

    – Mahalanobis Distance

$$D_M(p_1, p_2) = \left(\mu_1 - \mu_2\right)^T \left(V_1 + V_2\right)^{-1} \left(\mu_1 - \mu_2\right)$$

The GSF vertex filter shows little sensitivity to the number of components kept

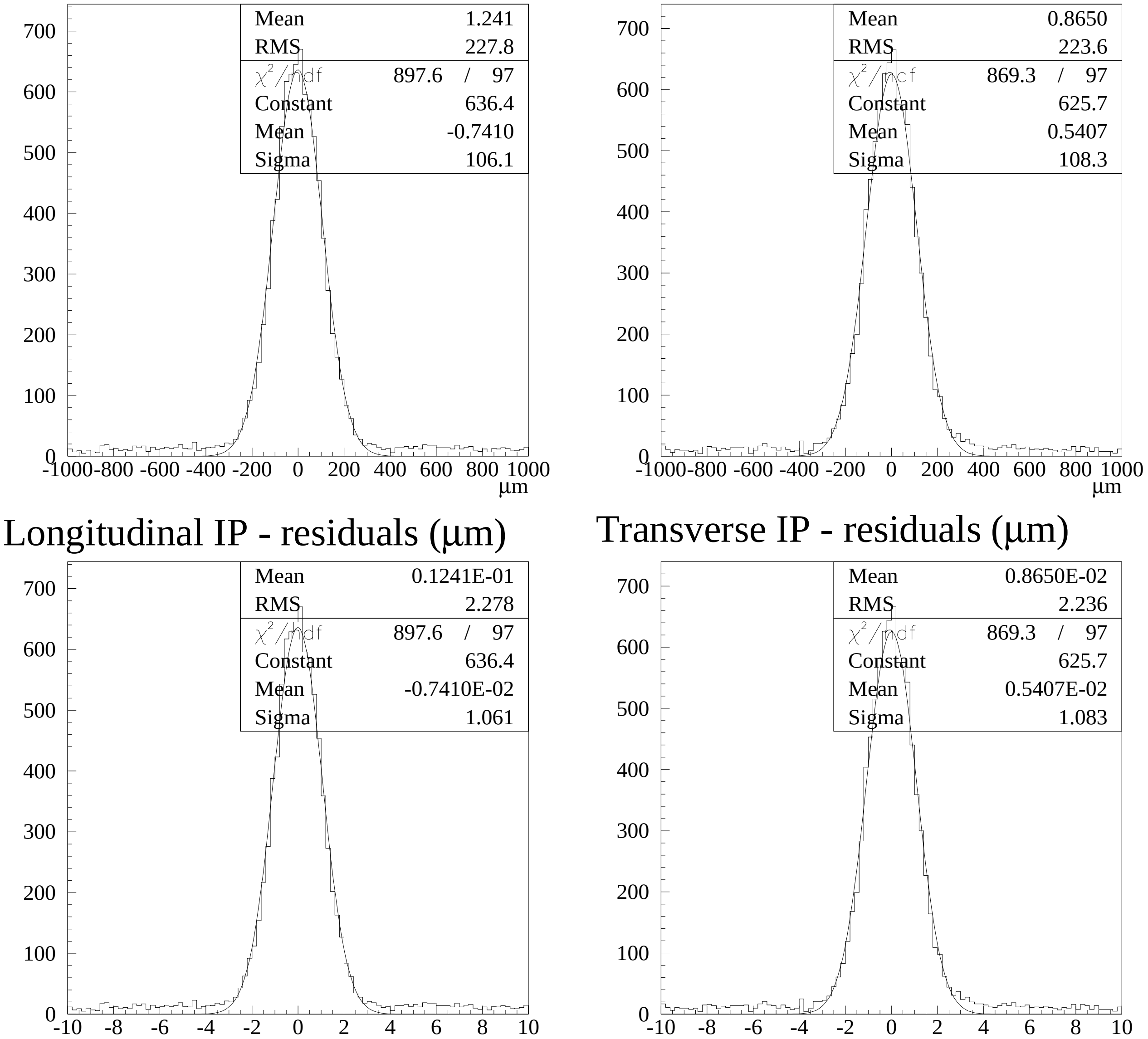
Component limitation

2 component Gaussians mixture:

- Narrow comp.: 80% rel. weight
- Wide comp.: 20% rel. weight
- Ratios of Standard deviation = 10

With 4 tracks: up to 16 components

Pulls when a single component is used (Kalman filter) →



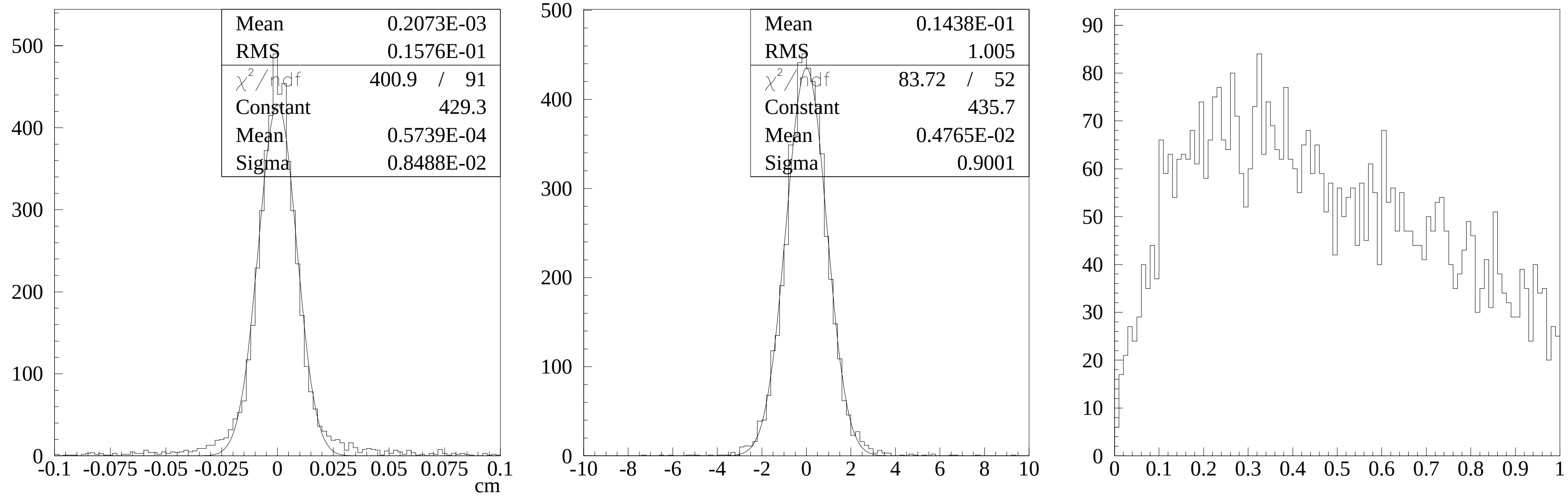
Longitudinal IP - residuals (µm)

Transverse IP - residuals (µm)

Longitudinal IP - pulls

Transverse IP - pulls

Component limitation



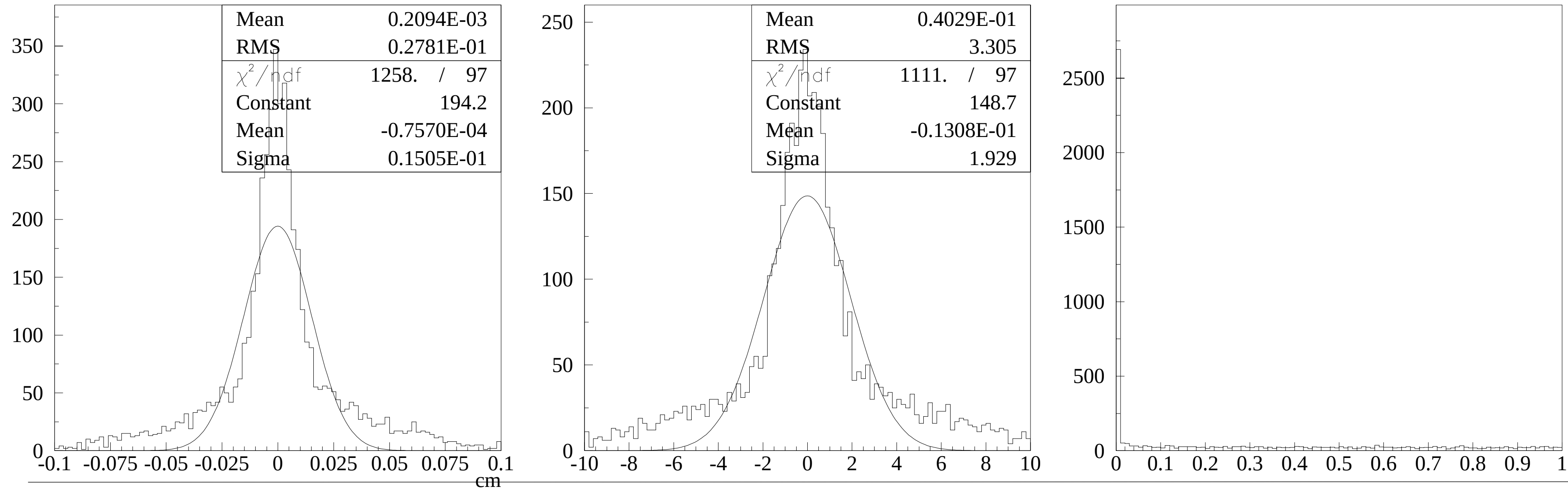
GSF - No limitation of the number of components

x Residuals (μm)

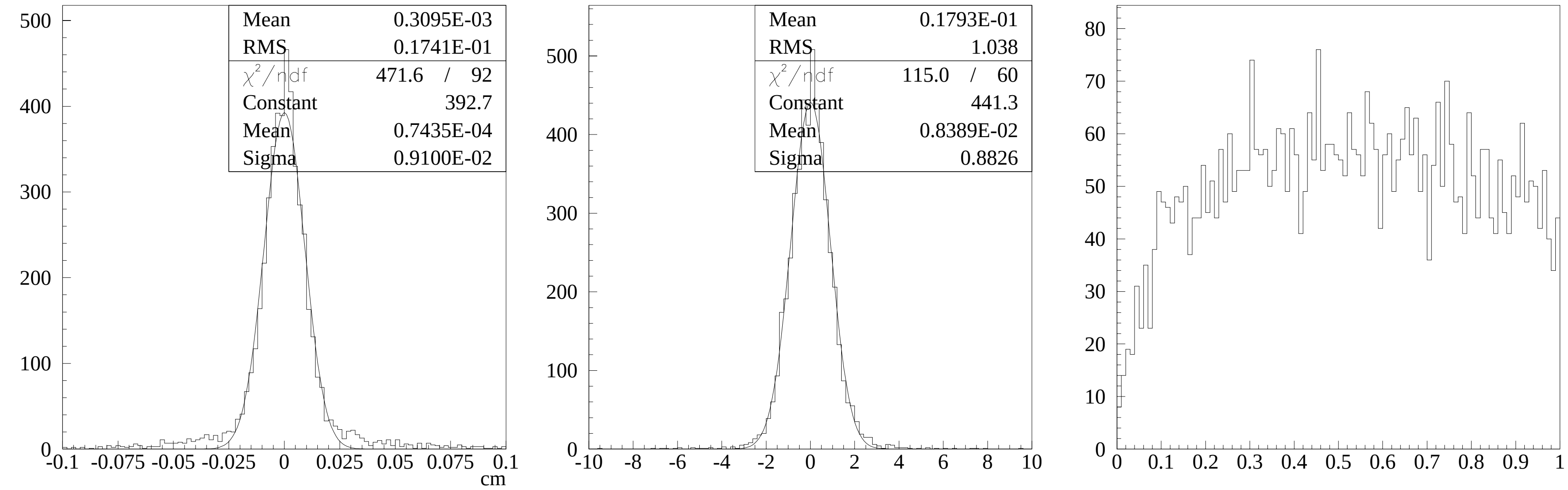
x Pull

P(χ^2)

Kalman Filter

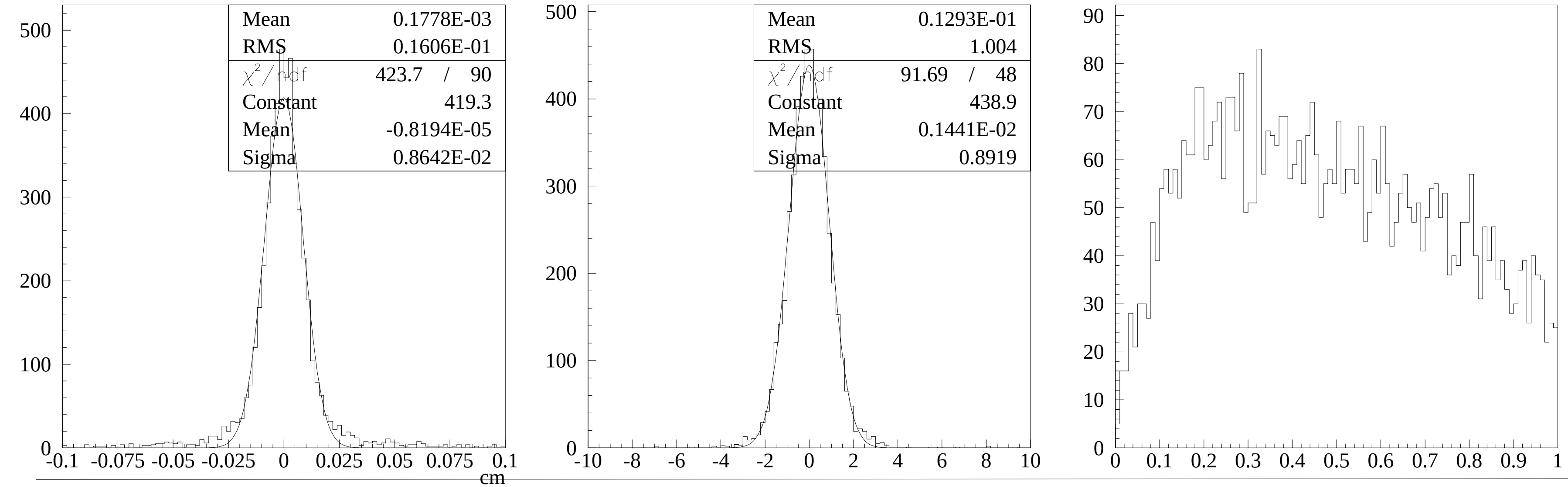


Component limitation



x Residuals (μm) *x* Pull $P(\chi^2)$

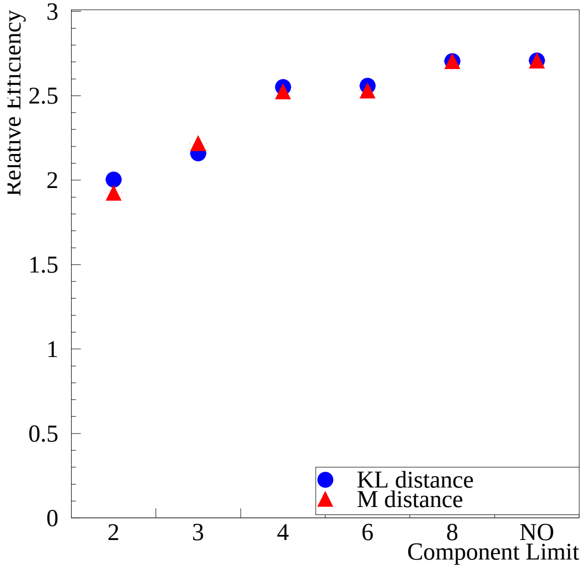
GSF - Limit of 2 components (using Kullback-Leibler Distance)



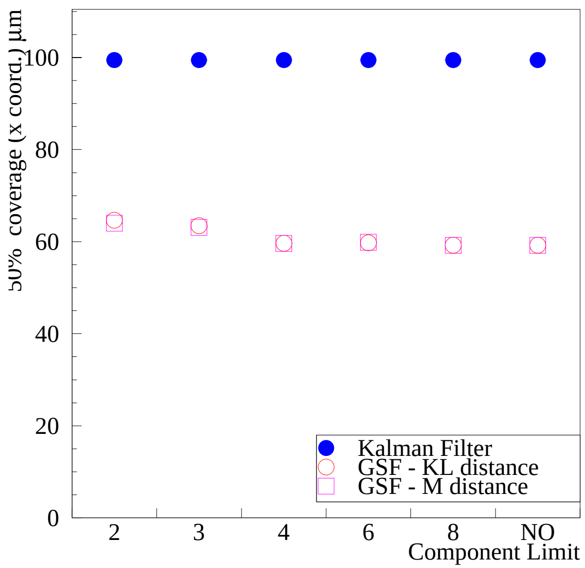
GSF - Limit of 4 components (using Kullback-Leibler Distance)

Component limitation

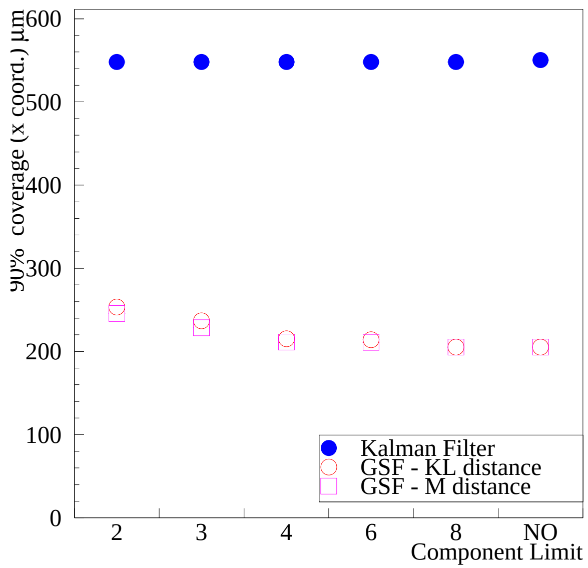
Nbr Comp.	Average χ^2	Res. [µm]	Pull	Average χ^2	Res. [µm]	Pull
No limitation	0.99	84.8	0.9			
	Kullback-Leibler Distance			Mahalanobis Distance		
2	0.91	90.5	0.88	0.93	92.2	0.84
3	0.94	90.5	0.88	0.95	89.7	0.85
4	0.95	85.4	0.89	0.95	84.9	0.89
6	0.96	84.9	0.89	0.97	84.6	0.89
8	0.96	83.9	0.9	0.97	83.9	0.9



Relative efficiency



50% coverage



90% coverage

Component limitation

4 component Gaussians mixture:

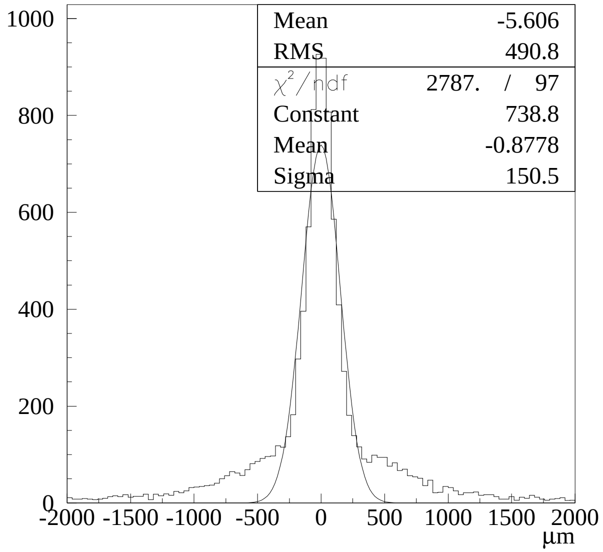
· 1st (narrow) comp.: 50% rel. weight (σ_1)

· 2nd comp.: 30% rel. weight ($\sigma_2=5*\sigma_1$)

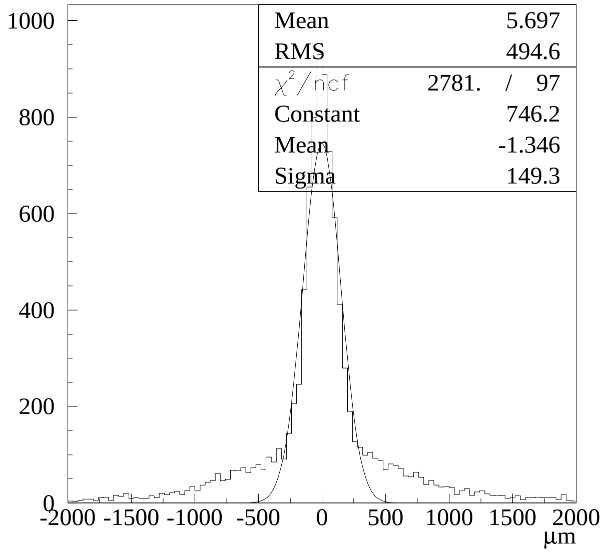
· 3rd comp.: 10% rel. weight ($\sigma_3=10*\sigma_1$)

· 4th comp.: 10% rel. weight ($\sigma_4=15*\sigma_1$)

With 4 tracks: up to 256 components



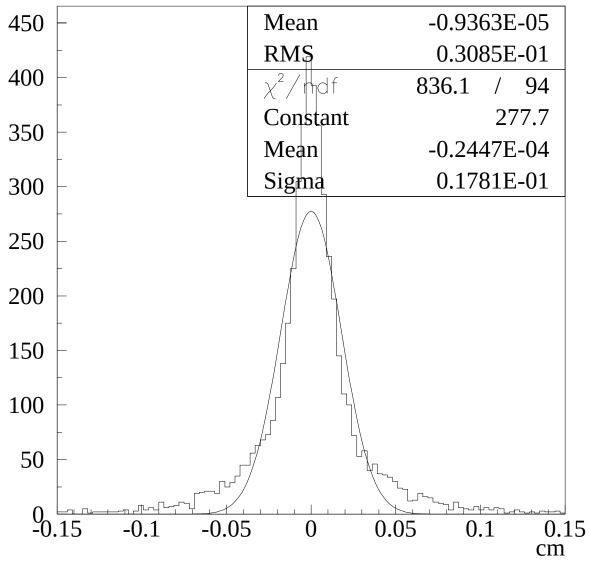
Longitudinal IP - residuals (µm)

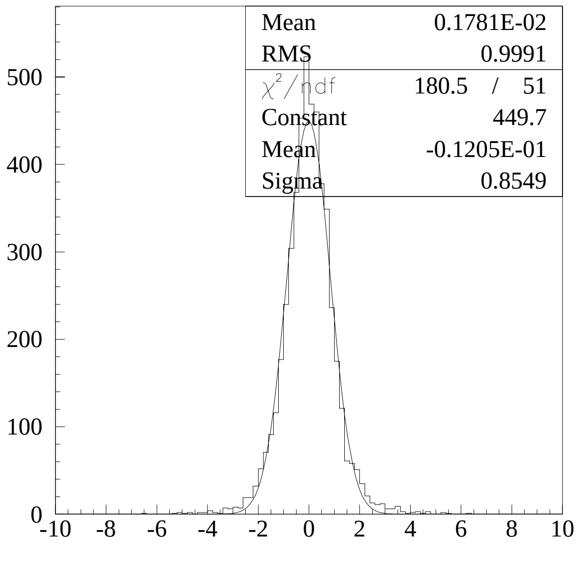


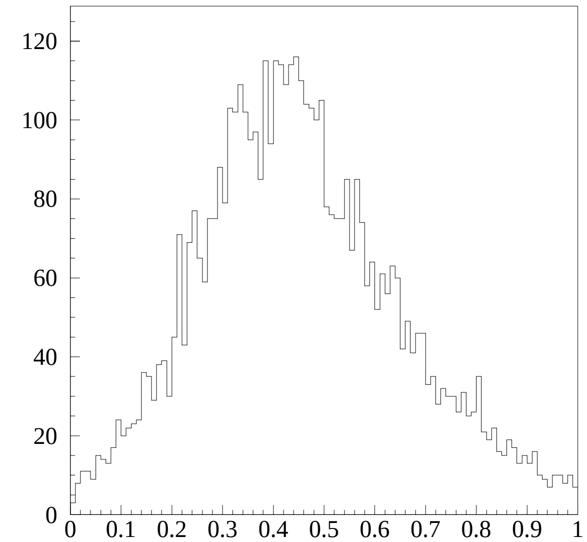
Transverse IP - residuals (µm)

For the Kalman filter, the collapsed state of the track has been used

Component limitation







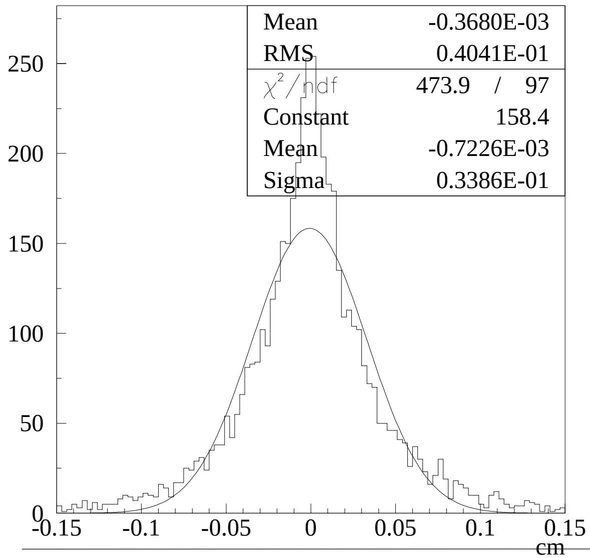
GSF -
No limitation
of the number
of components

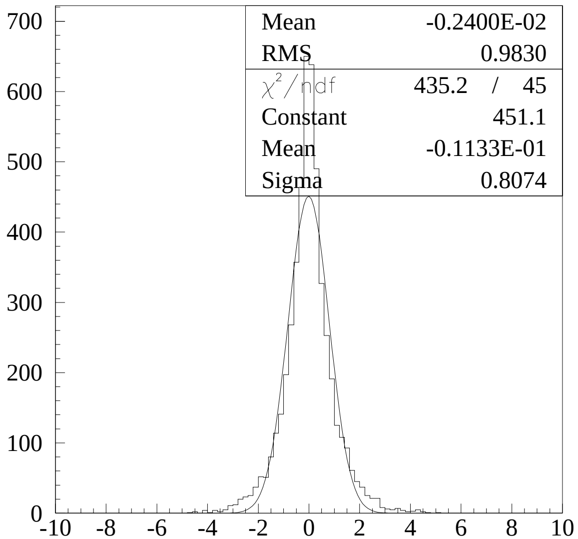
x Residuals (µm)

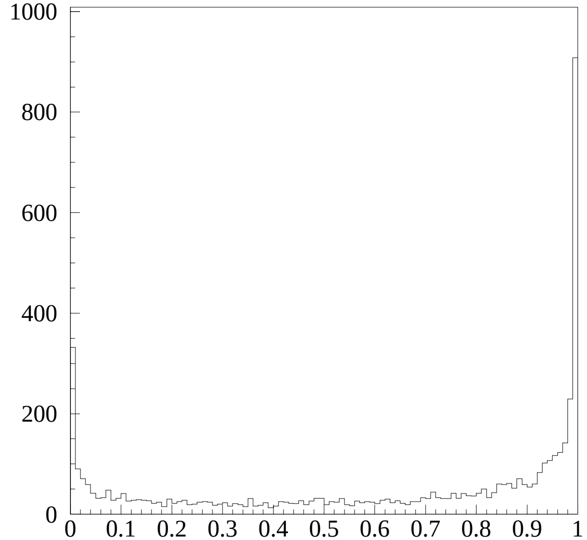
x Pull

P(χ^2)

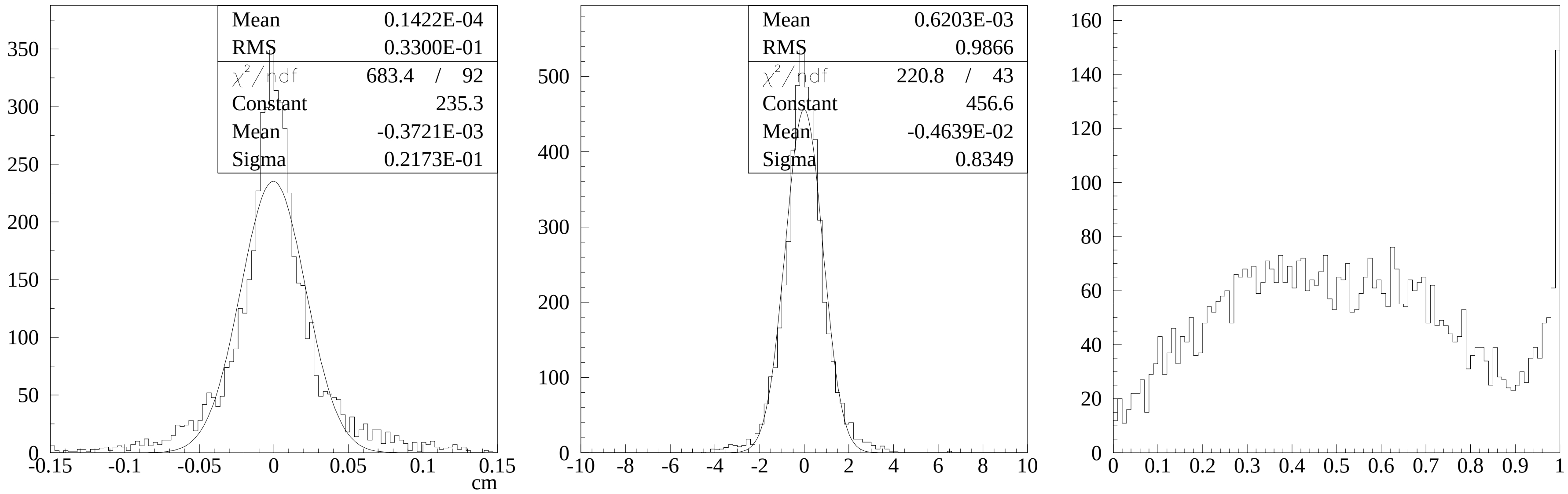
Kalman Filter





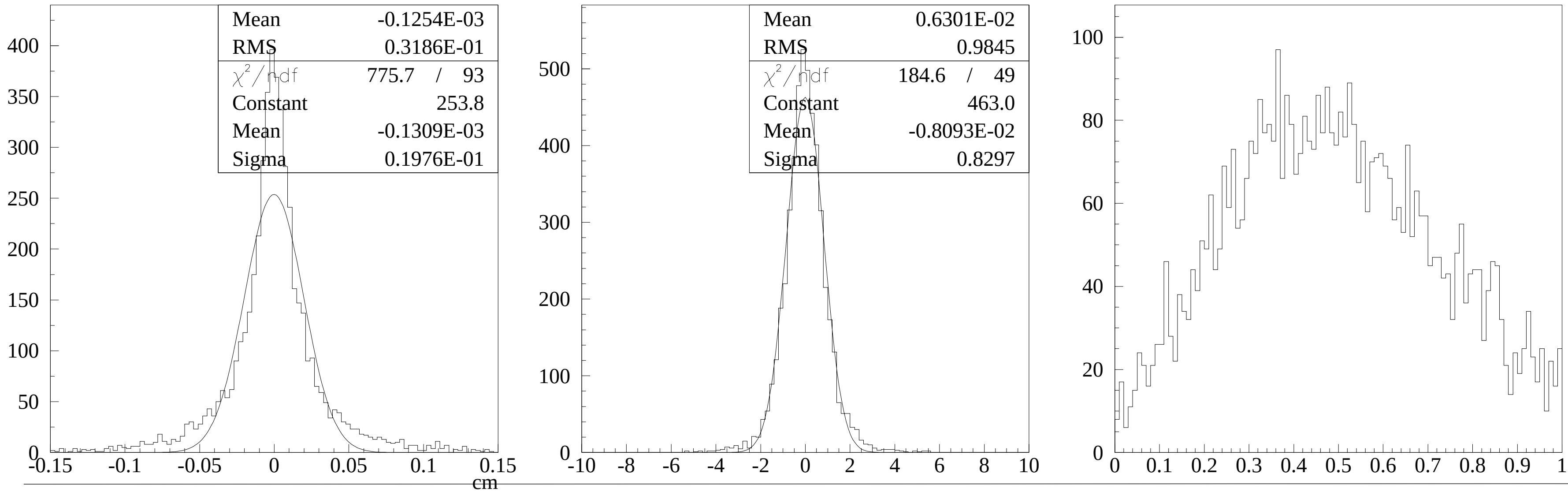


Component limitation



GSF -
Limit of
2 components
(using
Kullback-
Leibler
Distance)

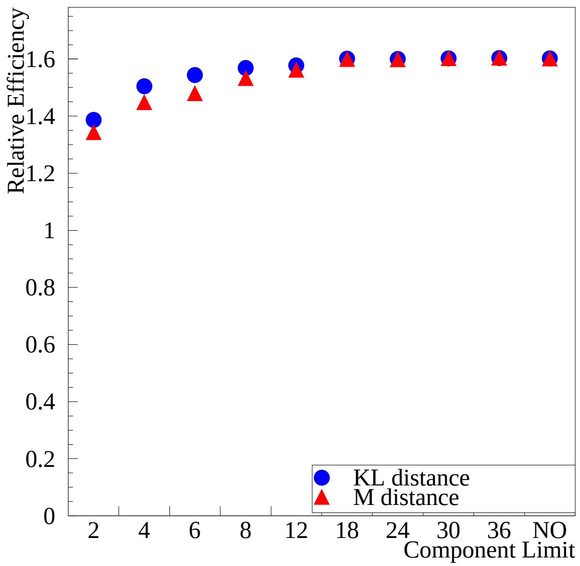
x Residuals (μm) *x* Pull $P(\chi^2)$



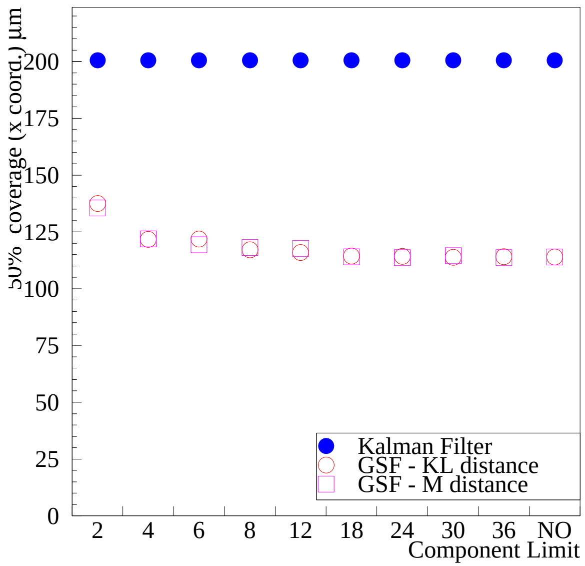
GSF -
Limit of
4 components
(using
Kullback-
Leibler
Distance)

Component limitation

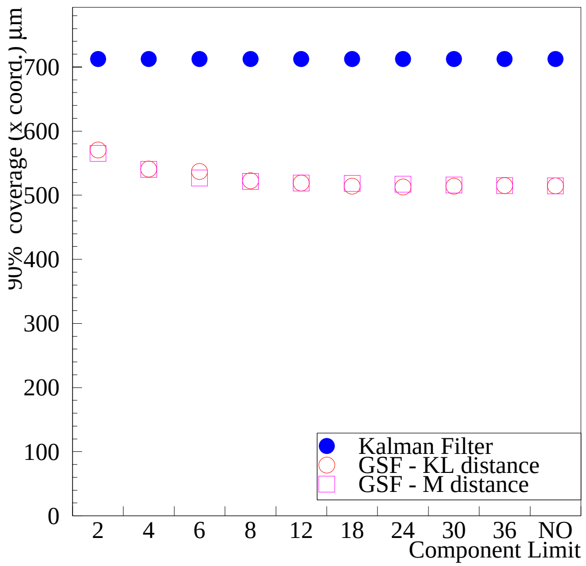
Nbr Comp.	Average χ^2	Res. [µm]	Pull	Average χ^2	Res. [µm]	Pull
No limitation	0.99	178	0.85			
	Kullback-Leibler Distance			Mahalanobis Distance		
2	0.91	217	0.81	0.93	217	0.81
4	0.94	197	0.83	0.95	191	0.82
6	0.95	197	0.83	0.95	188	0.82
8	0.96	191	0.84	0.97	184	0.82
12	0.96	188	0.84	0.97	181	0.83
18	0.99	179	0.85	0.99	178	0.85



Relative efficiency



50% coverage



90% coverage

Conclusion

- A Gaussian-sum Filter for vertex reconstruction has been implemented in the CMS reconstruction software
- Shows an improvement of the resolution and error estimate of the fitted vertex and of the χ^2 of the fit with respect of the Kalman Filter when the track parameters residuals have non-Gaussian tails.
- For electrons reconstructed with the GSF:

  Allows to use the full mixture, and not only the single collapsed state.
- Shows little sensitivity to the number of components kept during fit.
- A small number of components can be kept without degrading the fit too much.