The Compact Muon Solenoid Experiment

CMS Note

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

February 24, 2005

A Gaussian-sum Filter for Vertex Reconstruction

T. Speer

University of Zürich, Switzerland

R. Frühwirth

Institute of High Energy Physics, Vienna, Austria

Abstract

A vertex reconstruction algorithm that is based on the Gaussian-sum Filter (GSF) was developed and implemented in the framework of the CMS reconstruction program. While linear least-square estimators are optimal in case all observation errors are Gaussian distributed, the GSF offers a better treatment of non-Gaussian distributions of track parameter errors when these are modelled by Gaussian mixtures. The algorithm has been verified and evaluated with simulated data. The results are compared to the Kalman filter and to an adaptive vertex estimator.

1 Introduction

One of the most frequently used algorithms for vertex reconstruction is the Kalman Filter [1]. It is mathematically equivalent to a global least-squares (LS) minimization, which is the optimal estimator when the model is linear and all random noise is Gaussian. In that case, the estimator is unbiased and has minimum variance, residuals and pulls of estimated quantities have Gaussian distributions, and the objective function obeys a χ^2 -distribution. For non-linear models or non-Gaussian noise, it is still the optimal linear estimator.

A method that takes non-Gaussian distributions of measurement errors better into account is the Gaussian-sum filter (GSF). In many cases, the distribution of the measurement errors shows a Gaussian core and tails, so that a Gaussian mixture model is appropriate. The core corresponds to the principal component of the Gaussian mixture, and the tails can be modeled by one or several additional Gaussians. The distribution of the estimated parameters is a Gaussian mixture as well. The GSF is a non-linear estimator, as the weights of the components of this mixture depend on the measurements.

This approach has been applied to track reconstruction [2], where it has been seen that in the presence of large tails, the GSF has smaller variance than the Kalman filter. It is particularly useful for electron reconstruction, as the Bethe-Heitler distribution of bremsstrahlung energy loss is highly non-Gaussian and can be modelled by a Gaussian mixture. This has been successfully implemented in the CMS reconstruction software [3], where a significant improvement of the track parameter resolution is observed. This method has nevertheless never been applied to vertex reconstruction.

2 The Gaussian-sum filter for vertex reconstruction

In the context of vertex reconstruction, the measurements are the estimated track parameters. Their distribution is, as stated above, modelled by a mixture of multi-variate Gaussians probability density functions. The distribution of the estimated vertex state vector is thus also distributed according to a mixture of Gaussians.

In the fit, as in the Kalman filter, an iterative procedure is applied, i.e. the estimate of the vertex is updated with one track at the time. Each component of the vertex state mixture is updated with each component of the track parameter mixture by a Kalman filter, effectively doing an exhaustive combination of the components of the two mixtures. The GSF is thus a weighted sum of several Kalman filters, and it is implemented as a number of Kalman filters run in parallel. Only the weights of the components have to be calculated in addition.

As stated above, in a vertex fit, the parameter vectors p_k (for tracks $k = 1, \dots, n$) describing the tracks to be fitted to a common vertex are considered as the virtual measurements, obtained in a prior track reconstruction phase. The quantities to be estimated, the state vector, are the vertex position x and, for each track, its momentum q_k at the vertex. In contrast to the filter for track reconstruction, the system equation is the identity, as the position of the vertex is the same for each track.

The dependence of the track parameters on the state vector (x, q_k) is given by the measurement equation

$$p_k = h_k(x, q_k) + \epsilon_k, \quad (1)$$

where h_k is the functional dependence on the state vector and ϵ_k the measurement error. As the track model is usually non-linear, a first-order Taylor expansion of h_k at an expansion point (x_e, q_e) is used:

$$h_k(x, q_k) \approx h_k(x_e, q_e) + A_k(x - x_e) + B_k(q_k - q_{k,e}) = c_{k,e} + A_k x + B_k q_k, \quad (2)$$

where $A_k = [\partial h_k / \partial x]_e$ is the Jacobian w.r.t. to the position and $B_k = [\partial h_k / \partial q_k]_e$ is the Jacobian w.r.t. to the momentum, calculated at the expansion point.

Let us assume that the distribution of the estimated track parameters of track k is a Gaussian mixture with M_k components:

$$P(p_k) = \sum_{i=1}^{M_k} \gamma_k^i \Gamma(p_k; p_k^i, V_k^i), \quad \sum_{i=1}^{M_k} \gamma_k^i = 1, \quad (3)$$

where $\Gamma(p_k; p_k^i, V_k^i)$ denotes a multivariate Gaussian *pdf* with mean p_k^i and covariance matrix V_k^i .

Each component corresponds to a hypothesis on the measurement, with a corresponding estimate and spread, and the component weight γ_k^i is the probability of the hypothesis. No assumptions need to be made on these hypotheses, and the means are therefore allowed to be different. For instance, electron reconstruction with the GSF yields a Gaussian mixture with a different mean value in each component. It is only when the mixture is used to describe long-tailed observation errors that the components may have the same mean.

Each hypothesis is considered in turn. For hypothesis i , the *pdf* of the observation p_k^i , conditional on the state vector (x, q_k) , is given by

$$P(p_k^i | x, q_k) = \Gamma(p_k^i; A_k x + B_k q_k + c_k, V_k^i). \quad (4)$$

The *pdf* of the estimate of the vertex position based on tracks $\{1, \dots, k-1\}$, x_{k-1} , can be assumed to be a mixture of Gaussians with N_{k-1} components:

$$P(x_{k-1}) = \sum_{j=1}^{N_{k-1}} \pi_{k-1}^j \Gamma(x; x_{k-1}^j, C_{k-1}^j), \quad \sum_{j=1}^{N_{k-1}} \pi_{k-1}^j = 1. \quad (5)$$

By inclusion of track k to the vertex estimate x_{k-1} conditional on hypothesis i , the *pdf* of the new state vector (x_k, q_k) is

$$P_i(x_k, q_k) = P_i(x_k, q_k | p_k^i, x_{k-1}, q_k^0), \quad (6)$$

where q_k^0 is the prior assumption for the momentum vector of the considered track. By applying Bayes's theorem, this *pdf* can be written as

$$P_i(x_k, q_k | p_k^i, x_{k-1}, q_k^0) = \frac{P(p_k^i | x, q_k) P(x_{k-1}) P(q_k^0)}{\int P(p_k^i | x, q_k) P(x_{k-1}) P(q_k^0) dx dq_k^0}. \quad (7)$$

Taking the *pdf* of the prior momentum $P(q_k^0)$ as a single Gaussian distribution with covariance matrix D_k^0 , a calculation similar to the Lemma 2 of [2] yields for Eq.(7) a mixture of Gaussians with N_{k-1} components:

$$P_i(x_k, q_k) = \sum_{j=1}^{N_{k-1}} w_k^{ij} \cdot \Gamma \left((x, q_k); (x_k^{ij}, q_k^{ij}), \begin{pmatrix} C_k^{ij} & E_k^{ij} \\ (E_k^{ij})^T & D_k^{ij} \end{pmatrix} \right). \quad (8)$$

where the weights are given by

$$w_k^{ij} \propto \pi_{k-1}^j \Gamma(p_k^i; A_k x_{k-1}^j + B_k q_k^0 + c_{k,e}, (V_k^i)^{-1} + A_k C_{k-1}^j A_k^T + B_k D_k^0 B_k^T). \quad (9)$$

The constant of proportionality is determined by the requirement that the weights sum to 1. The final posterior *pdf* of the state vector is computed by summing over all hypotheses using their respective probabilities:

$$P(x_k, q_k) = \sum_{i=1}^{M_k} \sum_{j=1}^{N_{k-1}} \gamma_k^i w_k^{ij} \cdot \Gamma \left((x, q_k); (x_k^{ij}, q_k^{ij}), \begin{pmatrix} C_k^{ij} & E_k^{ij} \\ (E_k^{ij})^T & D_k^{ij} \end{pmatrix} \right). \quad (10)$$

The final mixture has $N_k = N_{k-1} \cdot M_k$ components.

The covariance matrix of the prior momentum D_k^0 has to be carefully chosen. As no prior information on the momentum from an independent source exists, the prior covariance matrix has to be large enough for the prior momentum not to bias the estimated state vector. Furthermore, the *pdf* of the prior momentum has to be a single Gaussian, to ensure that the weight of each component of the state vector is calculated with the same standard deviation. If instead a Gaussian mixture was used, where the different standard deviations scaled with the magnitude of the covariance matrix of the track parameters, track components with a small covariance would be artificially favoured, and vertex state components composed of these would have an artificially high weight.

With a suitably large matrix D_k^0 , the choice of the prior momentum becomes irrelevant and the inverse of D_k^0 can later be neglected. The calculation of the mean and covariance matrices of Eq.(8), which are the estimate of the state vector and its covariance matrix, yields the well known expressions of the Kalman filter:

$$x_k^{ij} = C_k^{ij}[(C_{k-1}^j)^{-1}x_{k-1}^j + A_k^T G_k^{B^i}(p_k^i - c_{k,e})] \quad (11)$$

$$q_k^{ij} = W_k^i B_k^T G_k^i (p_k^i - A_k x_{k-1}^j - c_{k,e}) \quad (12)$$

$$\text{cov}(x_k^{ij}) = C_k^{ij} = ((C_{k-1}^j)^{-1} + A_k^T G_k^{B^i} A_k)^{-1} \quad (13)$$

$$\text{cov}(q_k^{ij}) = D_k^{ij} = W_k^i + (E_k^{ij})^T C_k^{ij} E_k^{ij} \quad (14)$$

$$\text{cov}(x_k^{ij}, q_k^{ij}) = E_k^{ij} = -C_k^{ij} A_k^T G_k^i B_k W_k^i \quad (15)$$

$$W_k^i = (B_k^T G_k^i B_k)^{-1} \quad (16)$$

$$G_k^{B^i} = G_k^i - G_k^i B_k W_k^i B_k^T G_k^i \quad (17)$$

Although the Gaussian-sum filter is a non-linear filter, a “chi-square” can be defined in analogy with the Kalman filter. This pseudo-chi-square will obviously not be chi-square distributed. The total “chi-square” of the fit is the sum of the “chi-square” increment of each track-update, which is itself the weighted mean of the individual “chi-square” of each parallel filter:

$$\chi^2 = \sum_{k=1}^n (\chi_F^2)_k \quad (18)$$

$$(\chi_F^2)_k = \sum_{i=1}^{M_k} \sum_{j=1}^{N_{k-1}} \gamma_k^i w_k^{ij} \cdot (\chi_F^2)_k^{ij} \quad (19)$$

$$(\chi_F^2)_k^{ij} = (x_k^{ij} - x_{k-1}^j)^T C_{k-1}^j (x_k^{ij} - x_{k-1}^j) + \quad (20)$$

$$(p_k^i - A_k x_k^{ij} - B_k q_k^{ij} - c_{k,e})^T G_k^i (p_k^i - A_k x_k^{ij} - B_k q_k^{ij} - c_{k,e}) \quad (21)$$

3 Validation

To validate the algorithm, a simplified simulation in a fully controlled environment has been used. A single four-track vertex is generated per event. No track reconstruction is done, and track parameters are smeared according to a two-component Gaussian mixture model to simulate error distributions with a Gaussian core with tails. The component that models the tails of the distribution (the *wide component*) has a standard deviation ten times larger than the one modelling the core (the *narrow component*). The weights are 90% for the narrow component and 10% for the wide component, and the standard deviations of the impact parameters are 100 μm and 1000 μm , respectively.

In the fit with the Kalman Filter, where only one component is used, the variance of the track parameters used is that of the dominant (narrow) component¹⁾. The distributions of the vertex-coordinate residuals and pulls (Figure 1) have a Gaussian core with tails, and 32% of the fits have a χ^2 probability below 0.01.

¹⁾ In case of a high percentage of tails, it is arguable what the best choice for that variance would be. The

Figure 1: Residual (left) and pull (middle) of the x -coordinate of the reconstructed vertex and χ^2 probability (right) of the vertex fit using the Kalman Filter.

Figure 2: Residual (left) and pull (middle) of the x -coordinate of the reconstructed vertex and χ^2 probability (right) of the vertex fit using the GSF, without limiting the number of components.

The GSF uses both components, each with the correct weight and variance. In these first tests, no limit is placed on the number of components kept in the posterior distribution of the estimated vertex. The distributions of the residuals (Figure 2) of the estimated vertex position have fewer outliers, and the core, when fitted with a Gaussian, has a smaller standard deviation (called hereafter the *resolution*). The remaining outliers are due to vertices with several track-outliers. The distributions of the pulls show no tails and are nearly perfectly Gaussian with a standard deviation very close to 1. This indicates that the errors on the track-outliers are correctly taken into account and that a high weight is assigned to the correct component.

The distribution of the pseudo- χ^2 probability shows a dip at 0, large values of χ^2 occurring less frequently than expected. This can be explained by the fact that the tails of the narrow core component are well within the range of the wide component. Observations in the tails of the core are therefore interpreted as coming from the wide component, and their contribution to the χ^2 is accordingly small. Large values of χ^2 can occur only if all observations are in the tails of the wide component, and this has a very low probability already with four tracks.

These tests have been repeated with different two-component Gaussian mixture models, varying both the ratio of the standard deviations and the relative weight of the second component. To assess the improvement of the GSF with respect to the Kalman filter, the half-widths of the symmetric intervals covering 50% and 90% of the residual distribution (called *50%* and *90% coverages*) of the x -coordinate and the *relative efficiency* are used. The relative efficiency is defined as the ratio of the generalized variances of the fits with the Kalman filter and the GSF.

As the relative efficiency is very sensitive to outliers that are far away, the presence or absence of even

position of the estimated vertex, and hence the residuals, are nevertheless independent of the scaling factor of the variance used by the fit, whereas the uncertainty of the estimated vertex, and thus the pulls and the χ^2 , are no longer independent of that scaling factor.

Figure 3: 50% (left) and 90% coverage (middle) and relative efficiency (right) of vertex fits for tracks with two components as functions of the relative weight of the second component.

Figure 4: Kullback-Leibler Distance between a two-component Gaussian mixture and single-Gaussian distribution with identical moments, as a function of the relative weight p of the second Gaussian and the ratio f of their standard deviations.

one such vertex can have a large influence on this number. Fluctuations of this number can therefore be expected. These fluctuations are even more important when very distant outliers are present, as will be the case in the robustness tests of the next sections. The relative efficiency will thus not always be a reliable measure of the performance of an algorithm.

The relative efficiency and the 50% and 90% coverages are shown in Figure 3 for the same ratio of the standard deviations of 10 as used in the first example. As can be expected, vertices fitted with the GSF always show an improvement over those fitted with the Kalman filter. The coverages show the stability of the GSF with respect to the Kalman filter. The highest relative efficiency is reached when the distance between the two-component Gaussian mixture and the single Gaussian is the largest, as measured for example by the *Kullback-Leibler Distance* [4]. In Figure 4 is shown the Kullback-Leibler Distance between a two-component Gaussian mixture and Gaussian distribution with identical moments, as a function of the relative weight and the ratio of the standard deviations. With a higher proportion of outliers, the relative efficiency decreases, as the tails of the track parameter error distributions start to dominate.

Apart from the improvement of the position-resolution, another important improvement is the distribution of the χ^2 probability. As already shown, fits with the GSF do not display the large peak at small values.

Figure 5: *Ratio of events with $P(\chi^2) < 0.01$ for fits with the Kalman Filter.*

For the Kalman filter, the fraction of fits with $P(\chi^2) < 0.01$ is shown in Figure 5, although this will depend on the exact choice of the variance of the track parameters.

4 Limitation of the number of components

An obvious problem is that the number of components of the estimated state vector rises exponentially with the number of tracks, as at each step it is multiplied by the number of components modelling the current track. It has therefore to be limited to an acceptable number. This is achieved by pair-wise clustering (collapsing) of components which are close according to a certain distance measurement, until the desired number of components is reached. Two distance measurements are used, the *Kullback-Leibler Distance* [4] and the *Mahalanobis Distance* [5].

To test the sensitivity of the GSF with respect to the maximum number of components kept, the track parameters are smeared with a higher number of outliers, by increasing the relative weight of the wide component to 20%. Obviously, vertices fitted with the Kalman Filter are further degraded, with a higher number of outliers in the residual and pull distributions and a high number of fits (53%) with a χ^2 probability below 0.01. The same vertices fitted with the GSF without limiting the number of components show only a modest increase of the number of outliers and of the RMS of the core of the residual distribution. For a four-track fit, the final estimate of the vertex has 16 components.

In Table 1, the main features of the fits are shown for different maximum number of components, for either of the two distance measurements. When limiting the maximum number of components kept during the fit to two, the distribution of the residuals is only slightly degraded, with a modest increase of the width and of the tails. Only the distribution of the χ^2 probability changes noticeably. When increasing the maximum number of components to four, the results are nearly identical to those without limitation. There is no significant difference between the two distance measurements, and good results can be achieved even with a small number of components.

5 The Gaussian-sum smoother

As with the Kalman filter, the Gaussian-sum *smoother* recomputes the track parameters and their covariance matrices with the final estimate of the vertex position at the end of the vertex fit. As can be seen in the previous section, this trivially follows from the formalism of the GSF, in a way similar to the Kalman filter. Indeed, it is implemented as a number of Kalman smoothers run in parallel.

The main problem is that because of the merging or discarding of vertex-components, the one-to-one relationship between the vertex-components and the track-components is not unique anymore and is

Table 1: Comparison of the average χ^2 , resolution, pulls, 50% and 90% coverages and relative efficiency of the distributions of the estimated vertices, for different maximum number components using either the Kullback-Leibler Distance (KLD) or the Mahalanobis Distance (MD).

Nbr Comp.	Avg. χ^2	Resolution [μm]	RMS of pull	50% Cov. [μm]	90% Cov. [μm]	Rel. Eff.
Kalman Filter	21.35	150.5	1.93	99.1	546	-
Gaussian-sum filter						
No limitation	0.99	64.6	0.90	59.2	205	2.71
2 - KLD	0.91	90.5	0.66	64.7	253	1.93
3 - KLD	0.94	90.5	0.66	63.5	237	2.22
4 - KLD	0.95	65.4	0.69	59.6	215	2.53
5 - KLD	0.96	64.9	0.69	59.6	214	2.53
6 - KLD	0.96	63.9	0.90	59.2	205	2.71
2 - MD	0.93	92.2	0.64	64.0	246	2.00
3 - MD	0.95	69.7	0.65	63.1	226	2.16
4 - MD	0.95	64.9	0.69	59.6	211	2.55
5 - MD	0.97	64.6	0.69	59.6	205	2.56
6 - MD	0.97	63.9	0.90	59.2	205	2.70

lost. In an ideal case, if the number of vertex-components was not limited during the fit and all vertex-components were kept and none merged, each final vertex-component would have used exactly one component of each track. The smoothing procedure would then be quite straightforward, as each track-component would be smoothed with the vertex-components to which it has contributed.

Refitting the vertex keeping all the components is clearly not feasible, as this entails just the combinatorial explosion which the component-limitation mechanism tries to alleviate! Nevertheless, by refitting the vertex with the usual limitation of components, but taking care to add the track which is to be smoothed as the last one, the final mixture of the vertex estimate will retain the one-to-one relationship between the vertex-components and the track-components to be smoothed. As a full vertex fit for every track would be very time-consuming, an approach similar to the track fit has been adopted, where the results of a forward and a backward fit are combined. First, a fit in the descending order of the list of tracks is done, keeping each intermediary estimate (*forward filter*). Next, a fit in the ascending order of the list of tracks is done, keeping each intermediary estimate (*backward filter*). For each track that is to be refitted, a weighted mean of the forward and backward filter estimates which *do not* include that track is done. This temporary vertex is then updated with the relevant track through the GSF, and, at the same time, each component of the track is smoothed by a Kalman smoother. This ensures that each track component is smoothed with the correct vertex components.

This procedure has been tested on four-track vertices, where the track parameters are smeared with the default two-component mixture. The residuals and the pulls of the track impact parameters before (i.e. as they were smeared) and after smoothing are shown in Figure 6.

6 Robustness tests

As can be seen in this study, the GSF is very efficient when outliers are accurately described by the *pdfs* modelling the errors of the parameters, whereas, as is well known, LS-estimators such as the Kalman filter are very sensitive to those same outliers. The robustness with respect to true outliers, i.e. tracks which are not modelled correctly, can be tested by adding mis-measured tracks (type 1 outliers) or tracks from another vertex (type 2 outliers) to the list of correct tracks. In this respect, the GSF can be compared to another non-linear filter, the *Adaptive Vertex Filter* (AVF) [6, 7]. The Adaptive filter is an iterative re-weighted Kalman filter which down-weights tracks according to their χ^2 distance to the vertex. It has been shown that this filter is very stable with a high break-down point. Both filters can actually be

Figure 6: *Residuals (left) and pulls (right) of the track impact parameter in the local frame before (upper figures) and after (lower figures) smoothing the track, for tracks with two components (90/10 ratio).*

combined. Indeed, as in the Adaptive filter the computation of the vertex position is independent of the computation of the track weights, the Kalman filter used in the default implementation can be replaced by a GSF. In this way, the complete mixture modelling the track measurements is taken into account, instead of only a single component. This filter is referred to as the *Adaptive-GSF* (A-GSF).

For these tests, vertices with four tracks are simulated, where the direction of the jet is parallel to the x -axis, and the tracks are in a cone of 0.5 rad. The parameters of inlying tracks are smeared with the default two-component mixture. The Adaptive filter being sensitive to only a single component, the variance used for this filter is that of the narrow component.

For vertices without outliers, the distributions of the residuals and of the pulls of the vertices fitted with the Adaptive Filter and the Adaptive-GSF are similar to those obtained with the GSF. Table 2 shows that the results of the three non-linear filters are similar, although the 90% coverage indicates that the residual distribution for the Adaptive filter has slightly heavier tails. In addition, some 19% of the fits performed

Table 2: *Comparison of the average χ^2 probability, resolution, pulls, 50% and 90% coverages of the y -coordinate of vertices without outliers estimated with the different filters.*

Filter	Mean $P(\chi^2)$	Resolution [μm]	Pull	50% Cov. [μm]	90% Cov. [μm]	Rel. Eff.
Kalman Filter	0.32	70	1.41	48	229	-
GSF	0.47	54	1.00	36	90	5.8
Adaptive Filter	0.30	59	1.08	40	113	3.4
Adaptive-GSF	0.27	54	0.94	36	91	1.1

Table 3: Comparison of the average χ^2 probability, resolution, pulls, 50% and 90% coverages of the y -coordinate and relative efficiency of vertices estimated with the different filters. One type 1 track-outlier is included in the four-track vertex.

Filter	Mean $P(\chi^2)$	Resolution [μm]	Pull	50% Cov. [μm]	90% Cov. [μm]	Rel. Eff.
Kalman Filter	0.18	115	1.61	297	326	-
GSF	0.37	63	1.11	54	167	2.5
Adaptive Filter	0.18	92	1.34	59	160	2.0
Adaptive-GSF	0.24	64	1.00	55	166	2.1

with the Adaptive filter still result in a χ^2 probability below 1%, as compared to 32% with the Kalman Filter. The χ^2 probability distribution for the vertices fitted with the Adaptive-GSF is very similar to the one of the GSF.

6.1 Mismeasured tracks (Type 1 outliers)

Type 1 outliers have been simulated by smearing the tracks parameters with a different mixture than the one used in the vertex fit, the latter being the default mixture. Such outliers simulate tracks with errors that have been seriously underestimated. First, one of the four tracks in the vertex is replaced by an outlier. This outlier is smeared with a two-component mixture which is obtained from the default mixture by increasing the standard deviations by a factor of three.

Vertices fitted with the Kalman Filter (Table 3) are significantly degraded by the outlying track. The core of both the residual and the pull distributions are significantly broader with a higher number of outliers, and 50% of the fits have a χ^2 probability below 1% (Figure 7).

The same vertices fitted with either of the non-linear filters are significantly better than those fitted with the Kalman filter. For vertices fitted with the GSF or the Adaptive-GSF, the RMS of the cores of the residual distributions are obviously somewhat broader than those found for vertices without the outlier. There is only a modest increase of the tails, and the pull distributions are nearly unchanged. The χ^2 probability distributions are shifted to lower values with respect to those obtained without outliers. Vertices fitted with the AVF feature broader residual and pull distributions, as can also be seen in the coverages, and the χ^2 probability distribution reveals a significant peak at 0.

Repeating these tests with ratios of standard deviations of 2, 3 or 4, with 1, 2 or 3 outliers in a vertex of

Figure 7: χ^2 probability distributions for the four filters when one type-1 outlier is added to the four inliers.

Figure 8: Resolution (upper figures) and pulls (lower figures) of the x (left) and y -coordinate (right) of vertex fits for tracks with two components for different number of outliers among the four tracks, and different ratios of standard deviations between the outliers and the inliers.

4 tracks confirms this pattern. The values of the resolutions, pulls and the 50% and 90% coverages are summarized in Figures 8 and 9. The non-linear filters show a consistent improvement of the distribution of the residuals with respect to the Kalman Filter, both in terms of resolution and coverages. If the three non-linear filters also show a better pull distribution, it can be noted that those of the GSF, and in particular those of the Adaptive-GSF, are very stable and the variance remains close to 1.

6.2 Tracks from another vertex (Type 2 outliers)

To test the sensitivity to type 2 outliers, an outlying track originating from a second vertex is added to the four tracks from the main vertex. This second vertex is displaced by 3 mm in the transverse direction (y) relative to the jet-axis of the main vertex. The track parameters of all tracks, both inliers and outliers,

Table 4: Comparison of the average χ^2 probability, resolution, pulls, 50% and 90% coverages of the y -coordinate and relative efficiency of vertices estimated with the different filters. One type 2 track-outlier is added to the four-track vertex.

Filter	Mean $P(\chi^2)$	Resolution [μm]	Pull	50% Cov. [μm]	90% Cov. [μm]	Rel. Eff.
Kalman Filter	0.06	224	4.63	573	653	-
GSF	0.12	63	0.97	42	113	19.2
Adaptive Filter	0.41	69	0.99	46	155	3.6
Adaptive-GSF	0.43	63	0.66	42	112	12.6

Figure 9: 50% (upper figures) and 90% coverage (lower figures) of the x (left) and y -coordinate (right) of vertex fits for tracks with two components for different number of outliers among the four tracks, and different ratios of standard deviations between the outliers and the inliers.

are smeared with the default two-component mixture.

Vertices fitted with the Kalman filter are shifted towards the second vertex, with the mean of the residual distribution at 0.55 mm, as can be expected by the configuration of the tracks. The distributions of the residuals and pulls of vertices fitted with either of the non-linear filters are nearly not affected by the presence of the outliers, and remain remarkably stable. Again, the residual and pull distributions for fits with the Adaptive filter have somewhat heavier tails, as is confirmed by the 90% coverage (Table 4), and the χ^2 probability distribution reveals a significant peak at 0 (Figure 10). While the χ^2 probability distribution for the GSF is shifted to lower values, the corresponding distribution for the Adaptive-GSF is nearly unchanged.

Repeating the tests varying the position of the second vertex between 0 and 5 mm along the y -axis confirms this pattern. The values of the resolutions, pulls and the 50% and 90% coverages are summarized in Figures 11, 12 and 13. All three non-linear filters are remarkably stable and show a consistent improvement with respect to the Kalman Filter in terms of resolution, pulls and coverages. Indeed, the range of fluctuations seen for the GSF is small, less than 10 μm , which is to be compared with the displacement of the second vertex, up to 5 mm. For small displacements of this vertex, the AVF is not able to identify all of the outliers and is therefore slightly deteriorated as compared to the GSF and the Adaptive-GSF. This could probably be alleviated by choosing different parameters for the AVF.

The tests show the adaptive power of the GSF, which, in case of an outlier, is able to assign a higher weight to the track-components with the widest standard deviation. This ensures better protection against outliers and offers some robustness. Obviously, as the model does not describe the data adequately, the χ^2

Figure 10: χ^2 probability distributions for the four filters when one type-2 outlier is added to the four inliers.

probability will be distorted with respect to that obtained without outliers. This distortion is nevertheless less severe than for LS estimators. A more robust approach is to combine the AVF with a GSF, as the combined filter is able to both down-weight outliers and use the full mixture for the remaining tracks.

7 Conclusion

A Gaussian-sum filter for vertex reconstruction has been implemented. It has been validated with a simplified simulation, with different distributions of track parameter errors. A smoothing algorithm has also been developed and implemented. While the distributions are not those obtained by a real track reconstruction, the results with the Kalman filter are very close to those observed with fully simulated data, and indicate that similar results could be obtained with the GSF.

Extensive tests have demonstrated the validity of the algorithm and its efficiency in the presence of non-Gaussian noise. An improvement of the resolution and error estimate of the fitted vertex and of the χ^2 of the fit with respect to the Kalman Filter has been obtained, whenever the track parameters errors have non-Gaussian tails. In addition, when using electron tracks reconstructed with the GSF, the full mixture can be employed in the vertex fit, instead of only a single collapsed state. The GSF shows little sensitivity to the number of components kept during the fit, such that only a small number of these needs to be kept without degrading the fit too much. This allows the filter to be used without incurring a high performance penalty.

This filter also demonstrates a higher degree of robustness than LS-estimators such as the Kalman filter in the presence of mismeasured tracks (type 1 outliers) or tracks from another vertex (type 2 outliers). It has been shown that an even more robust algorithm can be constructed by combining the Adaptive filter with the Gaussian-sum filter. The Adaptive-GSF is able to both down-weight outliers and use the full mixture for the remaining tracks.

References

- [1] R. Frühwirth, R. Kubinec, W. Mitaroff and M. Regler, *Comp. Phys. Comm.* **96** (1996) 169
- [2] R. Frühwirth, *Comp. Phys. Comm.* **100** (1997) 1
- [3] W. Adam, R. Frühwirth, A. Strandlie and T. Todorov, eConf **C0303241** (2003) TULT009
- [4] G. Kitagawa, *Comput. Math. Appl.* **16** (1969) 503

- [5] K.V. Mardia, J.T.Kent and J.M.Bibby, *Multivariate Analysis*, Academic Press, London, UK, 1979
- [6] R. Frühwirth, K. Prokofiev, T. Speer, P. Vanlaer and W. Waltenberger, *Nucl. Instrum. and Methods A* **502** (2003) 699
- [7] J. D'Hondt, P. Vanlaer, R. Frühwirth and W. Waltenberger, *IEEE Trans. Nucl. Sci.* **51** (2004) 2037

Figure 11: Means (upper figures), standard deviations (middel figures) and RMS (lower figures) of the residual distributions of the x (left) and y-coordinate (right) of vertex fits for tracks with two components for different positions along the y-axis of the vertex of the outlying track.

Figure 12: Means (upper figures) and standard deviations (lower figures) of the pull distributions of the x (left) and y -coordinate (right) of vertex fits for tracks with two components for different positions along the y -axis of the vertex of the outlying track.

Figure 13: 50% (upper figures) and 90% coverage (lower figures) of the x (left) and y -coordinate (right) of vertex fits for tracks with two components for different positions along the y -axis of the vertex of the outlying track.