

Adaptive Multi-vertex fitting

R. Frühwirth, W. Waltenberger

Institute of High Energy Physics of the Academy of Sciences, Vienna, Austria

Abstract

The Multi-Vertex Fitter is an algorithm that fits n vertices simultaneously, allowing competition between the vertices for the tracks. In each vertex fit, each track is weighted according to its distance to the current vertex position. The method is adaptive in the sense that the weight with respect to a vertex depends also on the other vertices that compete for the same track. This paper presents the implementation and verification of the method in ORCA and outlines future developments and possible use cases.

INTRODUCTION

The robustification of vertex estimation is an important issue in the reconstruction of secondary vertices. In order to have the optimal separation between the primary and one or several secondary vertices, it is essential that neither of them, in particular the secondary ones, are distorted by wrongly assigned tracks. A recent development is the Adaptive Vertex Fitter (AVF, [1]) which automatically down-weights outlying tracks by means of soft assignment. For each track assigned to the vertex a weight is computed which is based on the distance of the track from the vertex. The vertex estimation is iterated until the weights stabilize.

The Multi-Vertex Fitter (MVF, [2]) is very similar to the AVF in that it has a soft assignment with a weight function that is a generalization of the AVF weight function. The difference to the AVF is that several vertex candidates compete for the available tracks. Thus the (soft) assignment of a track may change from vertex to another one in the course of the iterations. For the special case of a single vertex candidate the MVF reduces to the AVF.

THE FIT

The fitting procedure is iterative and can be described as follows:

- The user supplies reconstructed tracks (parameters and covariance matrices), initial positions of vertex candidates, and optionally initial assignment probabilities between tracks and vertices.
- Then the following two steps are repeated until convergence:
 1. Fit all vertices, using the assignment probabilities as track weights.
 2. Recompute the assignment probabilities using the most recent vertex positions.

In each iteration, each vertex fit is a reweighted least-squares estimator, which can be computed for example by a Kalman filter.

THE WEIGHT FUNCTION

Assume that there are n tracks to be fitted to m vertices. The weight of vertex j with respect to track i is given by

$$w_{ij} = \frac{\exp(-\chi_{ij}^2/2T)}{\exp(-\chi_{\text{cut}}^2/2T) + \sum_{k=1}^m \exp(-\chi_{ik}^2/2T)}$$

χ_{ij}^2 is the χ^2 -distance between track i and vertex candidate j . Note that χ_{ij}^2 is computed using the original track weight. χ_{cut}^2 is a cutoff which suppresses tracks that do not fit to any of the vertices. T is a temperature parameter which is lowered in the course of the iterations (annealing). This helps in finding the globally optimal solution of the assignment problem. The natural final value of T is 1. If hard assignment is desired, T should be lowered close to 0.

Figure 1 shows the weight of a track with respect to a vertex as a function of the χ^2 -distance to the vertex, both without competition from other vertices and in the presence of a single competing vertex.

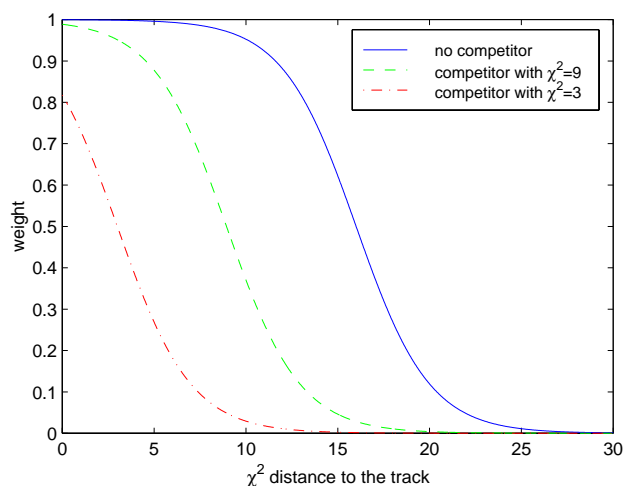


Figure 1: Weight of a track with respect to a vertex as a function of the χ^2 -distance to the vertex. The cut is at $\chi^2=16$. Solid blue line: no competing vertex; dashed green line: one competitor with a $\chi^2=9$ w.r.t. the track; dashed-dotted red line: one competitor with a $\chi^2=3$ w.r.t. the track.

IMPLEMENTATION AND VERIFICATION STRATEGY

The algorithm was implemented in ORCA [3], re-using many basic classes of the existing Kalman Filter and the AVF implementations. Effectively, the MVF consists of several parallel reweighted Kalman filters.

The verification was done using the *Vertex Gun* [2] developed for ORCA. The Vertex Gun is a fast simulation tool that directly generates “reconstructed” tracks from simulated tracks by adding random noise, bypassing event generation, detector simulation, and track reconstruction. Using the vertex gun has the double advantage of speed and perfect knowledge of the track errors.

The verification sample consists of events with two vertices. Five tracks are attached to the primary vertex at $(0,0,0)$. The tracks form a “jet” with total jet momentum $\vec{p}_{\text{pr}}=(0,25,25)$, and opening angle=0.5. A secondary vertex is positioned two millimeters from the primary vertex. It has three tracks, jet momentum $\vec{p}_{\text{sec}}=(-15,0,20)$, and opening angle=0.5. All tracks are “perfectly reconstructed”, i.e. the errors are Gaussian and correctly described by the covariance matrices attached to the tracks.

Before vertex reconstruction the events are degraded in the following manner. The track of the primary vertex that is most compatible to the secondary vertex is selected, and vice versa. Those two tracks are moved to the “wrong” vertices — the primary track is assigned to the secondary vertex and vice versa (Figure 1). These two track bundles with one mis-associated track in each bundle are the input to the MVF.

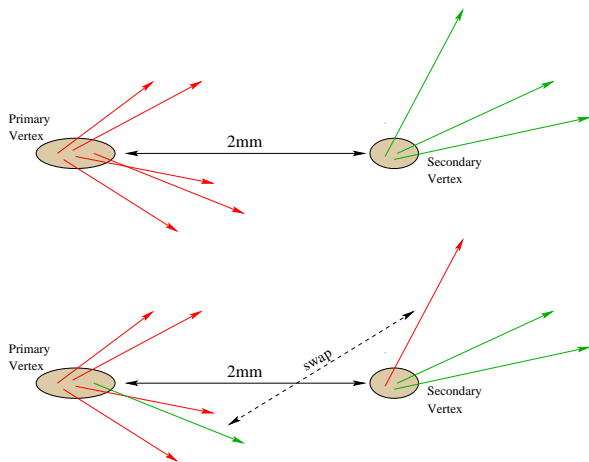


Figure 2: MVF code verification: Track swapping

RESULTS

We have compared the MVF to the AVF and to the Kalman Vertex Fitter (KVF). The most important indicator of the quality of vertex reconstruction is the resolution of the *distance* between the two reconstructed vertices, i.e. the

decay length of a short-lived particle connecting the two vertices.

The KVF and the AVF cannot recover the mis-associated tracks. The KVF estimate (Figure 3, top) shows a huge bias, because all tracks enter the fit with the same weight. The AVF (Figure 3, center) down-weights the outliers, but does not use the tracks that are associated to the wrong track bundle. Its estimate is therefore unbiased, but the distribution has tails, and resolution is worse than the one of the MVF (Figure 3, bottom).

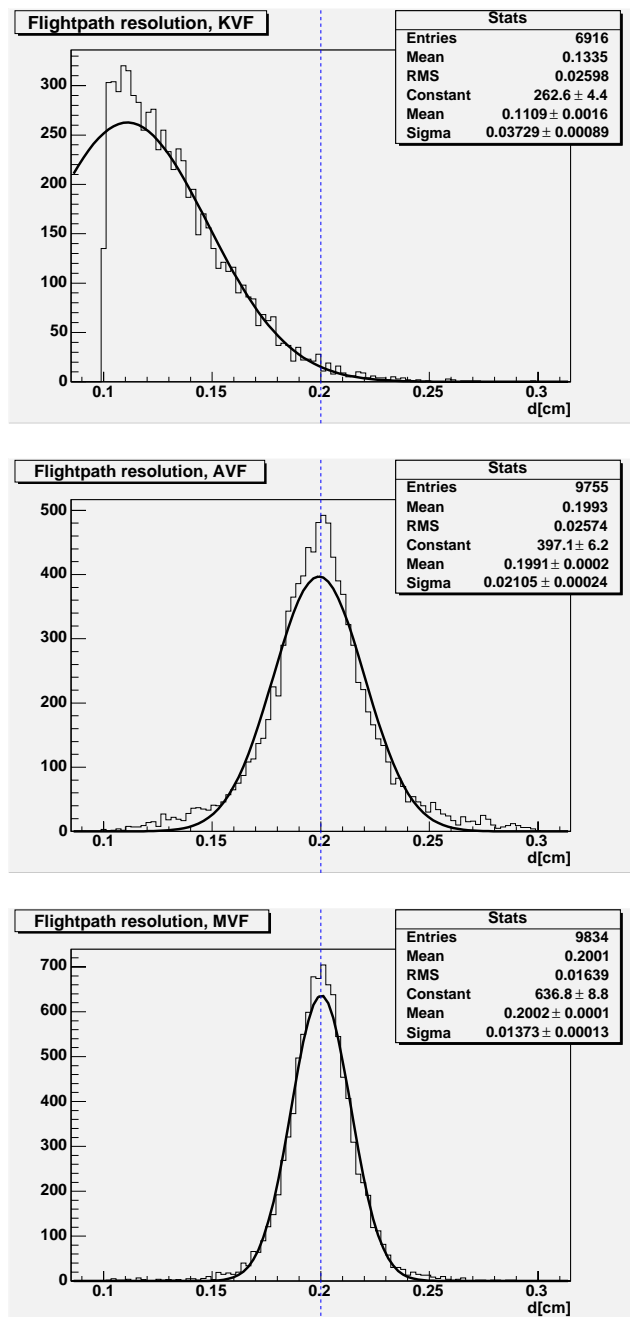


Figure 3: Flightpath distributions for the different fitters. The dashed blue line denotes the true value.

It is instructive to directly compare the assignment probabilities of the AVF and the MVF. The table in Figure 4 shows 10000 events with two vertices and eight tracks, resulting in 80000 correct track-to-vertex associations (“inliers”) and 80000 wrong associations (“outliers”). It can be seen that both algorithms almost always identify the outliers. AVF fails to identify only three outliers; MVF assigns about 30 tracks to the wrong vertex. In the case of inliers an ideal AVF can only do right in 60000 cases; 20000 tracks are mis-associated from the very beginning and cannot be recovered. It can be seen that the AVF comes very close to the theoretical limit. Only about 450 inlying tracks (less than one percent) are down-weighted when they should not be. The MVF performs similarly with respect to its own theoretical limit, only its theoretical limit is much higher, namely 80000. About 400 tracks have a weight close to zero where the right weight would be one or close to one.

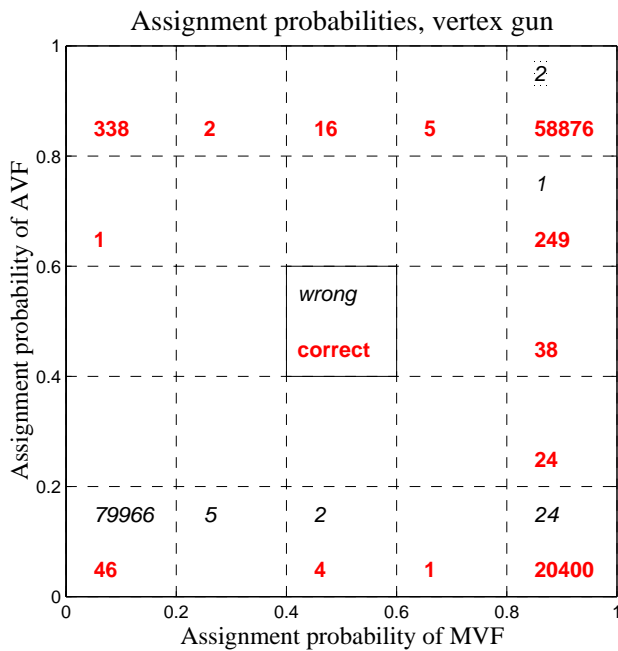


Figure 4: Comparison of the assignment probabilities, MVF versus AVF. Bold: correct track-to-vertex associations; italic: wrong track-to-vertex associations.

FUTURE DEVELOPMENTS

An evaluation of the MVF in terms of physics, especially b -tagging, is under way. The preliminary results are very encouraging.

A short term development of the MVF will be to introduce the possibility of having “hard assigned” tracks, i.e. tracks that are associated to one specific vertex, the association being in a “frozen” state. This additional feature can be used to exploit knowledge that comes from external sources. As a longer term development a generalization of these “hard assigned” tracks seems to be desirable. We

believe it would be useful to implement a framework for any kind of constraints, be it on the tracks, its weights, or a quantity related to the vertex. A way to achieve this could be to interact with the kinematic package [4]. This package implements kinematic constraints via the Lagrange multiplier formalism.

REFERENCES

- [1] J. D’Hondt, P. Vanlaer, W. Waltenberger, and R. Frhwirth: Sensitivity of robust vertex fitting algorithms. CMS Note 2004/002, 2003. To appear in: IEEE Transactions on Nuclear Science.
- [2] W. Waltenberger: Vertex Reconstruction in CMS. PhD thesis, in preparation.
- [3] CMS Collaboration: ORCA, CMS OO Reconstruction. <http://cmsdoc.cern.ch/orca>.
- [4] K.Prokofiev, Th. Speer: A Kinematic fit and a decay chain reconstruction library. CMS Internal Note AN-2004/020.