The Forward Tracking: Algorithm and Performance Studies

LHCb Note

Issue: 1
Revision: 0

Reference: LHCb-015-2007
Created: December 19, 2006
Last modified: May 7, 2007

Prepared by: Olivier Callot,
Laboratoire de l'Accelerateur Lineaire - Orsay;
Stephanie Hansmann-Menzemer,
Physikalisches Institut Heidelberg

Abstract

This note describes the so-called Forward Tracking. This algorithm uses as input track seeds reconstructed in the VELO (Vertex Locator) and searches for associated tracks in the inner and outer tracker of the T stations based on a Hough Transformation approach. We describe in detail the implementation of the algorithm as it is in Brunel version 30r14 and the according performance on DC06 [1] data.

1 Introduction

A track of a charged particle in a magnetic field can be described by five parameters, namely its position (x, y), its direction $(dx/dz, dy/dz)$ and its momentum at a given position z. If we know the field map of the magnetic field (and neglect material effects) the path of the particle through the detector can be precisely determined from those parameters. The algorithm documented here is the so-called Forward Tracking. It uses tracks reconstructed in the vertex detector (VELO) as input. Since there is essentially no magnetic field in the VELO only the position and the direction of the track are known from the seed track. One additional x measurement of the particle trajectory in or behind the magnetic field[1] is sufficient to determine the missing momentum information. The Forward Tracking algorithm combines all measurements in the T stations with the VELO seed. It computes for each combination the x coordinate of the potential particle trajectory at a fixed z value (z = 8520 mm). The measurements are thus projected along the trajectory onto this reference plane. The measurements which belong to the same particle as the VELO seed will be projected (within uncertainties) on the same x position. Projections from random measurements are expected to be distributed uniformly. Technically the extrapolated track positions at the reference plane are filled in a sorted list and selecting track candidates corresponds to identifying a significant cluster in this list. This pattern recognition approach is called Hough Transformation.
The basic idea of this method is quite simple. However several technical details and performance optimizations resulted in a rather complex implementation of the algorithm. Once we move on to pattern recognition on real data we need to be aware of all cuts and decisions applied throughout the tracking strategy as some of them - e.g. sizes of search windows - will depend on our understanding of the detector and might need retuning. This note is an attempt to describe the Forward Tracking and its implementation as detailed as possible.
We first briefly remind of the LHCb tracking system. Then we introduce the propagation model to compute the intersection of the track candidates with the reference plane. Next we follow the various steps through the tracking algorithm. Finally, we will report the performance of the algorithm as it has been measured with Brunel v30r14 based on DC06 [1] data. The results presented here supercede studies based on an earlier version of this algorithm [2].

[1] The direction of the principle component of the magnetic field in the LHCb detector is along the y direction.

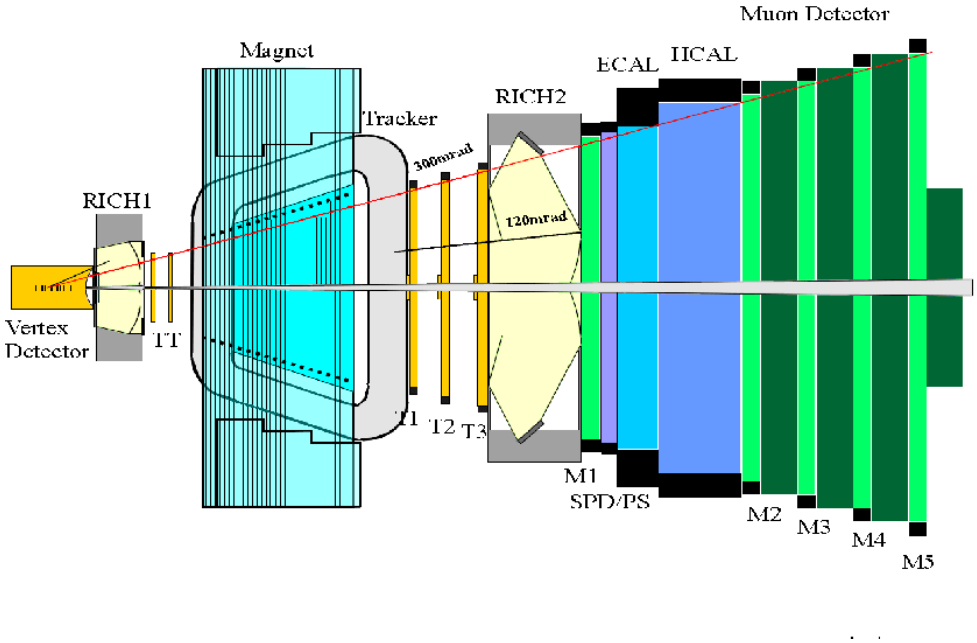

Figure 1 Top view (x/z) of the LHCb forward spectrometer. The z axis is defined along the beamline, the interaction point is located around z = 0. The direction of the principle component of the magnetic field is along the y axis.

2 Brief Introduction to the LHCb Tracking System

The LHCb Tracking System consists of a silicon vertex detector close to the interaction point (Vertex Locator - VELO), two tracking station with two silicon layers each in the fringe field before the magnet (Trigger Tracker - TT) and three tracking stations (T stations) behind the magnet (Fig. 1). Each T station consists of four layers each with a (x, u, v, x) structure. x represents a tracking layer with approximately vertical detection elements and u, v are tilted layers with $\theta = \pm 5^\circ$ stereo angle. For the outer tracker (OT) these are double-layers of straws, while for the inner tracker (IT) they are single layers of silicon strips. Once the drift times are resolved the outer tracker single measurement resolution is ≈ 200 μm, the inner tracker resolution is ≈ 50 μm. The hit efficiency in an outer tracker monolayer is 92 % while the one in the inner tracker is 99.5 %. A detailed description of the various LHCb detector components can be found in [3].

3 Pattern Recognition

The Forward Tracking algorithm proceeds as follows. First we search for potential measurements in the the x planes of the T stations, applying loose cuts based on the information from the VELO seed. Next hit candidates are projected on the reference plane. We then search for clusters of hits which potentially belong to the VELO seed. Next a third order polynomial is fitted to the hits in the cluster. Based on the contribution to the χ^2 of the fit, hits which potentially do not belong to the track are identified and removed from the cluster. Track candidates which fulfill certain quality

criteria such as minimum number of hits or maximum χ^2/dof are then passed to the search for stereo hits. Here again first a rough preselection of potential stereo hits is performed. In combination with the y information from the VELO seed the x position can be derived out of the u, v measurements. The deviations in x of the hit from the track are stored in a sorted list. Then again a cluster search is performed. Finally a parabolic fit of the x hits and the x information of the stereo hits and a straight line fit to the y information of the stereo hits is performed. Again the hits with the largest contribution to the χ^2 of the fit are removed until the track candidate fulfill certain quality criteria or it is discarded. Finally a quality variable based on momentum, χ^2/dof, compatibility in y of the VELO seed and the track in the T stations and number of hits is introduced which is then used to select the best track.
The various pattern recognition steps are described in detail in the following.

3.1 Selection of Potential x Hits

As a first preselection criteria a y search window is defined in each x plane. As x planes do not provide any y information beside their geometrical acceptance the restriction to a search window in y mainly restricts for the OT the search to the upper or lower half of the detector and for the IT to the four boxes, which are read out separately. The center y_{center} of the search window is computed as straight line extrapolation of the VELO seed to $z_{0,plane}$, the central z position of the x plane.

$$y_{center} = y_{0,VELO\ seed} + dy/dz_{VELO\ seed} \times z_{0,plane} \tag{1}$$

The size of the search window Δy is defined as follows:

$$\Delta y = \texttt{yCompatibleTol} + 50\ \text{mm} \times dy/dz_{VELO\ seed} \tag{2}$$

where `yCompatibleTol`[2] = 10 mm. Only hits which lie within this window are further considered.
With the additional y information from the VELO seed the x position of the hits can be corrected for small y slopes of the planes[3]. First the y position and z position of the intersection point of the VELO seed with the measurement plane is recomputed. Then the corrected x and z position of the hit (x_{meas}, z_{meas}) is derived.

$$y = \frac{y_{0,VELO\ seed} + dy/dz_{VELO\ seed} \times z_{0,plane}}{1 - dy/dz_{VELO\ seed} dz/dy_{plane}} \tag{3}$$

$$z_{meas} = z_{0,plane} + dz/dy_{plane} \times y \tag{4}$$

$$x_{meas} = x_{meas} + dx/dy_{plane} \times y \tag{5}$$

For hits in the OT detector the drift distance r is given:

$$r = (drift\_time - (wire\_length - |y|) \times wire\_velocity) \times drift\_velocity \tag{6}$$

where $drift\_time$ is the raw measurement time, $wire\_velocity$ and $drift\_velocity$ are read out from the OT geometry. $wire\_length$ is an offset of a specific channel, which is 0 per default but can be

[2] All parameters which can be adjusted via option files are written in type writer style.
[3] The nominal slope of the planes is 3.6 mrad, which corresponds to the rotation angle of the beamline in the LHCb detector with respect to the horizontal plane.

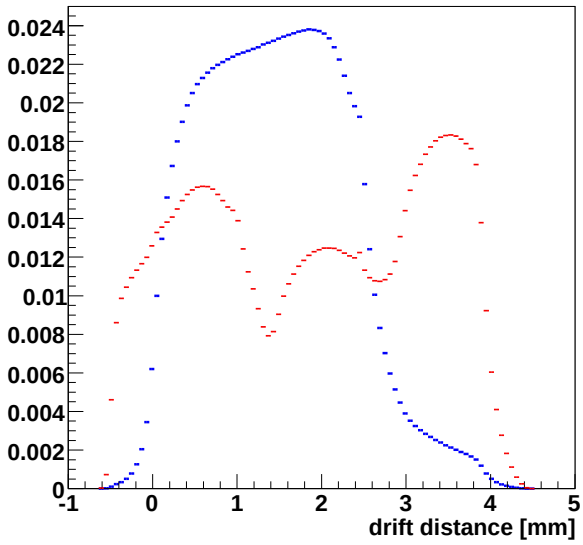

Figure 2 Drift distance of OT hits corrected for y position of seed track. In blue/bold the distribution for good hits, red/thin for wrong ones.

adjusted according to the information in the calibration data base. In case the drift distance r is smaller than `minOTDrift` = -0.1 mm or larger than `maxOTDrift` = 2.6 mm, this hit is not taken into account for the pattern recognition. The distribution of the drift distance for good (associated to VELO seed) and wrong hits (not associated to VELO seed) is displays in Fig. 2. Those cuts cause a loss in single hit efficiency ($<$ 8%) but reduce significantly the rate of wrong hits (44 %). As those hits are unphysical (outside the cell radius of 2.5 mm) they are potentially anyhow removed at a later step of the algorithm e.g. outlier removal in the fit.

Next the search window in the reference plane is defined. Therefore the VELO seed is extrapolated as straight line to the reference plane at z_{ref} = 8520 mm.

$$x_{extrapolated} = x_{0,VELO\ seed} + dx/dz_{VEL0\ seed} \times z_{ref} \tag{7}$$

For tracks without any momentum estimate a symmetric search window is opened around the extrapolated x position. The search window is adjusted to correspond to the deviation from the straight line due to the magnetic field for a particle with at least `minPt` = 80 MeV transverse momentum and at least `minMomentum` = 1 GeV total momentum. This results in a search window size Δx which is the minimum of

$$\Delta x_1 = \texttt{rangePerMeV} \times dx/dz_{VELO\ seed} \times 1/\texttt{minPt} \text{ and} \tag{8}$$

$$\Delta x_2 = \texttt{rangePerMeV}/\texttt{minMomentum} \tag{9}$$

where `rangePerMeV` = 5.25 mm. In case the momentum of the seed is known (this is possible for VELO tracks with additional information in the TT stations[4]) the extrapolated position is corrected

[4]The standard pattern recognition uses VELO only seeds; hits in the TT stations are added later.

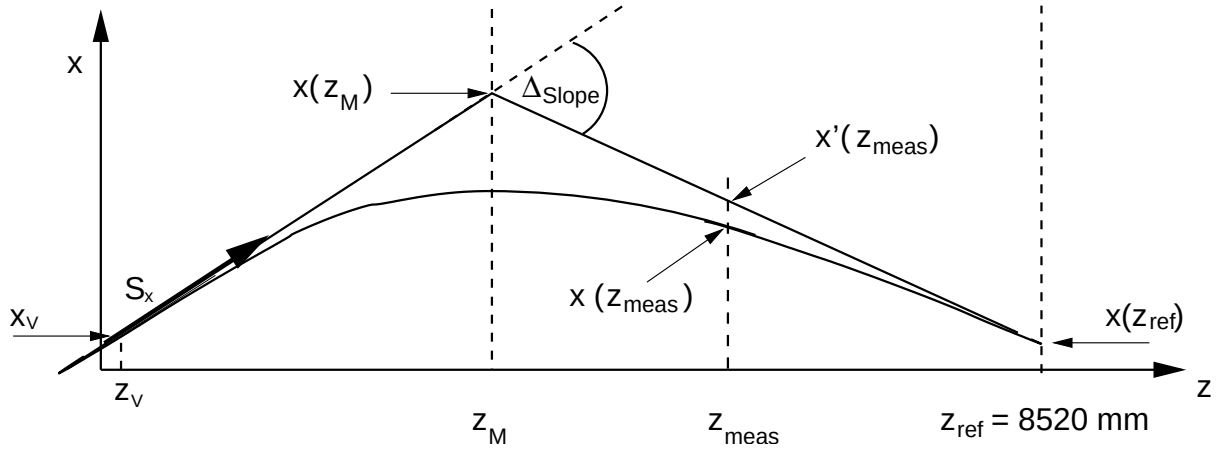

Figure 3 Sketch of relevant quantities for the track parameterization used in the pattern recognition strategy.

by the expected deviation. A momentum dependent search window Δx is computed

$$x_{extrapolated} = x_{extrapolated} + \texttt{rangePerMeV} \times q/p[\mathrm{MeV}^{-1}] \quad (10)$$
$$\Delta x = \texttt{minRange} + \texttt{rangeErrorFraction} \times \texttt{rangePerMeV} \times q/p[\mathrm{MeV}^{-1}] \quad (11)$$

where q is the sign of the charge of the track seed and p its momentum; `minRange` = 300 mm and `rangeErrorFraction` = 0.6.
Next the x projection of the hit on the reference plane is computed and checked to be consistent with the x window defined in Eq. 8-11.

3.1.1 Track Parameterization and Computation of x_{ref}

Theoretical the problem of propagating a track through a magnetic field with a given field map is well defined. But for pattern recognition we have to find a reasonably fast approximation to derive the intersection of the track with the reference plane at a fixed position behind the magnet.
The parameters of the VELO seed are $x_V(z_V)$, $y_V(z_V)$, $S_x(z_V) = dx/dz(z_V)$, $T_y(z_V) = dy/dz(z_V)$. z_V is the z position where the VELO seed is parameterized, $x_{meas}(z = z_{meas})$ is the corrected hit position (Eq. 4-5).
For an ideal magnet the track outside the magnetic field could be described as two straight line which intersect in the middle of the magnet (Fig. 3). But there are some weak fringe field as well outside the magnet volume which cover at least the first tracking station T_1. A cubical parameterization has turned out to describe best the path of the track through the remaining magnetic field behind the magnet:

$$x(z) = x_{ref} + B_x(z - z_{ref}) + C_x(z - z_{ref})^2 + D_x(z - z_{ref})^3; \quad (12)$$
$$y(z) = A_y + B_y(z - z_{ref}); \quad (13)$$

B_x is the tangent of the track at z_{ref}. Δ_{slope} is the angle between the x slope of the VELO seed S_x and B_x. Note that the z_M, the position of the focal plane of the magnet, is due to its trapezoidal

shape not anymore right at the center of the magnet but depends on the space parameters of the track. Based on fits to tracks in Monte Carlo the following empirical parameterization has been found:

$$\begin{aligned} B_x &= \frac{x_M - x_{ref}}{z_M - z_{ref}}; && (14)\\ \Delta_{slope} &= B_x - S_x; && (15)\\ z_M &= M_0 + M_1 \times \Delta^2_{slope} + M_2 \times S^2_x + M_3 \times x^2_{meas} + M_4 \times T^2_y; && (16) \end{aligned}$$

z_M depends on Δ_{slope} and the other way round. Therefore the computation of x_{ref} needs two iterations to be solved:

$$\begin{aligned} z_M &= M_0 + M_2 \times S^2_x + M_3 \times x^2_{meas} + M_4 \times T^2_y; && (17)\\ x_M &= x_V + S_x \times z_M; && (18)\\ \Delta_{slope} &= \frac{x_{meas} - x_M}{z_{meas} - z_M} - S_x; && (19)\\ z_M &= z_M + M_1 \times \Delta^2_{slope}; && (20)\\ & && (21) \end{aligned}$$

The x positions of the hit has been already corrected for potential y slopes of the measurement planes (Eq. 5). This correction was at that time based on the assumption that the magnetic field does not have any impact on the y slope, which is only approximately true. A small field component perpendicular to y causes a change in dy/dz which results in a correction δy of the y position of the track at z_{meas} with respect to the earlier computation.

$$\begin{aligned} \delta y &= \Delta^2_{slope} \times T_y \times (Y_1 + (z_{meas} - z_{ref}) \times Y_2); && (22)\\ x_{meas} &= x_{meas} + \delta y \times dx/dy_{plane}; && (23)\\ C_x &= X_1 \times \Delta_{slope}; && (24)\\ D_x &= X_2 \times \Delta_{slope}; && (25)\\ x^{'}_{meas} &= x_{meas} - C_x \times ((z_{meas} - z_{ref}))^2 - D_x \times ((z_{meas} - z_{ref}))^3; && (26)\\ x_{ref} &= x_M + (z_{ref} - z_M) \times \frac{x^{'}_{meas} - x_M}{z_{meas} - z_M}; && (27)\\ B_x &= \frac{x_M - x_{ref}}{z_M - z_{ref}}; && (28)\\ A_y &= y_{0,VELO_seed} + T_y \times z_{ref} + \Delta^2_{slope} \times T_y \times Y_1; && (29)\\ B_y &= T_y + \Delta^2_{slope} \times T_y \times Y_2 && (30) \end{aligned}$$

$x^{'}_{meas}$ is defined in Fig. 3.

The values of $M_0 - M_4$, X_1, X_2, Y_1 and Y_2 have been derived from fits to Monte Carlo data [4].

3.2 Scan for Hough Clusters

The next step of the pattern recognition is to search for clusters of hits in the sorted list of projected x positions. The hits are sorted by increasing x projection. The maximal width Δx (distance between first and last hit) of a cluster is computed in the following way.

$$\Delta x \quad = \quad \texttt{maxSpreadX} + |(x_{first\ hit} - x_{extrapolated}) \times \texttt{maxSpreadSlopeX}|; \tag{31}$$

where `maxSpreadX` = 0.6 and `maxSpreadSlopeX` = 0.011; $x_{extrapolated}$ has been defined in Eq. 7, 10. This definition of Δx takes into account that clusters which are further away from the straight line extrapolation belong to a low momentum track. For those tracks the clusters are expected to be broader. In case the first hit of a cluster is an OT measurement Δx is enlarged by 1.5 mm to take into account the drift times (so far only the center cell positions are used). If not at least hits from `minXPlanes` = 5 different x planes are found in the window $[x_{hit\ i}, x_{hit\ i} + \Delta x]$, the hit $i + 1$ is taken as next starting point of a potential cluster. The same test is then repeated. Once a successful cluster candidate [$x_{hit\ j}$, ..., $x_{hit\ k}$] is found in the list, we search for overlapping clusters. Therefore we add hits $k+1$, $k+2$, ... to the cluster as long as one hit in the range $j+1, ..., k+1-5$ can be chosen as a starting hit of the overlapping cluster and both requirements (hits in at least `minXPlanes` different planes and $x_{start} - x_{end} < \Delta x$) are fulfilled. Once we failed to add a new hit to the cluster, we check if the position of the starting hit j is within Δx to the end position of a previously accepted cluster. If so we merge the both clusters, otherwise we define a new cluster. This procedure is repeated until all hits in the list are checked. Each cluster is in the following considered as potential track candidate.

3.3 2D Fit & Outlier Removal

The IT and OT detector planes are split up in regions as it is shown in Figure 4. With the configuration of the magnetic field given in the LHCb detector it is - at least for tracks in the OT - unlikely that at track crosses regions. Therefore we check first for subsets of hits in the cluster which are all in the same region and which additionally have at least one hit in each of the six x planes. In case we find several subclusters which fulfill these requirements we chose the narrowest subcluster.
In case there is no region with one hit in each plane we count how many different planes are represented in the cluster. We then search the narrowest subcluster which contains hits from each of them, independently to which region they belong. This list of hits is then our central subcluster.
Once a subcluster is defined by one of those criteria, additional hits are tested. Hits which are within a tolerance of 0.2 mm from the first or the last hit of this central subcluster are merged in the cluster. In case the subcluster contains mainly OT hits the tolerance is enlarged to 2.0 mm to account for drift times. Once new hits are merged to the cluster the tolerances are counted from the new starting and end hit and additional hits are tried to be merged as well.
Next a first track parameterization based on the VELO seed and the central hit in the subcluster is derived (Eq. 12-30). All hits are corrected according to the new best track parameterization. The same formulas as in Eq. 3-5 are used but instead of $y_{0,VELO\ seed}$ and $dy/dz_{VELO\ seed}$ this time $A_y - B_y \times z_{ref}$ and B_y of the current track are used.

Next we try to resolve drift ambiguities of OT measurements already before the fit. Therefore only layers with at least two OT hits (one per monolayer) are studied. All possible track/hit combinations

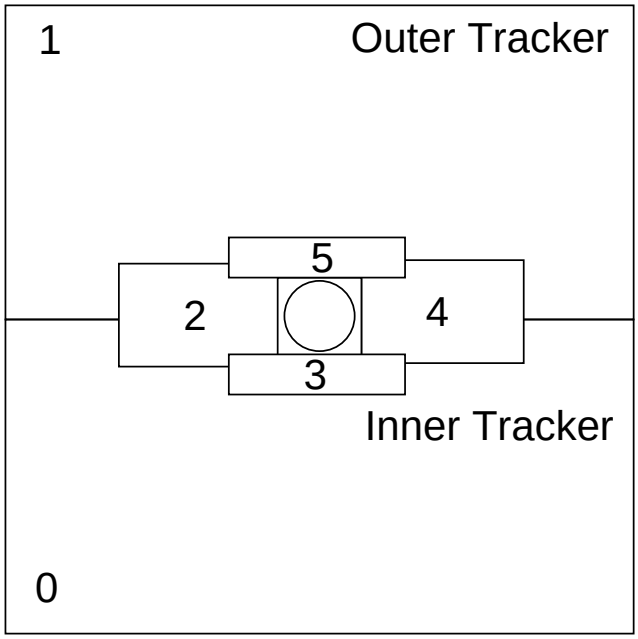

Figure 4 Definition of OT & IT regions in the T stations.

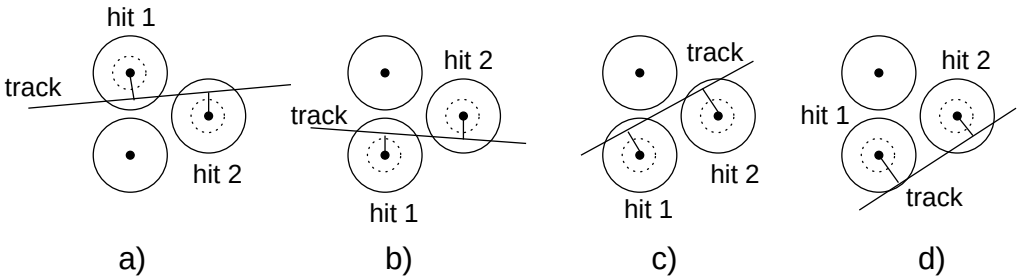

Figure 5 Possible hit/track configurations in OT double layers.

are displayed in Fig. 5. While in case a) and b) the left-right ambiguity can be easily resolved in case c) and d) the precision of the current track is not good enough to make any decision. Mathematically the criteria for fixing the ambiguity before the fit is

$$|dist(track, hit_1, +1)| + |dist(track, hit_2, -1)| < 0.3 \text{ mm} \quad (32)$$
$$|dist(track, hit_1, -1)| + |dist(track, hit_2, +1)| < 0.3 \text{ mm} \quad (33)$$

Where the $\pm$ 1 corresponds to the case where the track passes below/above the center of cell. Eq. 32 corresponds to drawing a) and Eq. 33 corresponds to drawing b). In those cases the ambiguities are fixed before the fit. For all other OT hits the solution which is closest to the starting track of each fit iteration is chosen. Then a fit of the deviations from the previously determined track parameterization is performed. Note that a cubical track model is used although here we only fit for three of the parameters.

$$x(z) = (A_x + \delta A_x) + (B_x + \delta B_x) \times (z - z_{ref}) + (C_x + \delta C_x) \times (z - z_{ref})^2 + (D_x) \times (z - z_{ref})^3; \quad (34)$$

A_x, B_x, C_x, D_x are fixed and we fit then for δA_x, δB_x and δC_x. All hits are included in the fit with its weight, additionally the x position at z_M of the extrapolated VELO seed is added to the fit with the following uncertainty:

$$\sigma^2_{x_M} = \texttt{xMagnetTol} + \Delta slope^2 \times \texttt{xMagnetTolSlope} \quad (35)$$

where `xMagnetTol` = 3 mm^2 and `xMagnetTolSlope` = 40 mm^2. Adding the VELO seed as additional constraint to the fit does not introduce any additional bias (e.g. on K_s reconstruction efficiency) which is not anyhow present by the fact that the forward tracking requires reconstructed VELO seeds to be present.

After each fit A_x, B_x and C_x are updated according to the fit result and the fit procedure is repeated up to at most 10 times. This is needed as the OT ambiguities that remain unresolved can be flipped during the fit. The iterations are stopped in case the fitted deviations of the track parameter fulfill the following requirements:

$$\delta A_x < 5.0 \times 10^{-3} \quad (36)$$
$$\delta B_x < 5.0 \times 10^{-6} \quad (37)$$
$$\delta C_x < 5.0 \times 10^{-9} \quad (38)$$

In case there is one or more hits which contribute more than `maxChi2` = 20 to the overall χ^2 of the fit, the hit with the largest χ^2 contribution (= maxχ^2_{hit}) is removed. If

$$\max\chi^2_{hit} < 20 \times \texttt{maxChi2}, \quad (39)$$

all hits in the total cluster (not only the previously defined subcluster) which have a distance to the fitted track which is smaller than maxχ^2_{hit} are added to the subcluster. Then we restart the fitting procedure. The check for hits outside the central subcluster is done only the first time requirement Eq. 39 is valid. In later iterations no hits are added to the central subcluster. The iterations stop when either no hit has a χ^2 contribution to the fit which is larger than `maxChi2` or we have less than `minXPlanes` different planes represented in the hit list of the track candidate. In the first case the x candidate is accepted in the later one it is discarded. If a track is accepted we repeat the search for a subcluster and the fit procedure with the not yet used hits of the original cluster.

In average we are left with two to five x candidates per VELO seed which are then passed to the stereo pattern recognition part.

3.4 Selection of Potential Stereo Hits

The track parameterization obtained by the fit of the x candidate is now extrapolated to every u/v plane via Eq. 12-13. We check that this position is within $\pm$ `yCompatibleTol` (= 10 mm) consistent with the geometrical acceptance of the plane.
Next the u/v hits are combined with the knowledge of the track parameters and thus transformed into x measurements. Again Eq. 3-5 are used and again $A_y - B_y \times z_{ref}$ and B_y of the x candidate track are used instead of the parameters of the VELO seed. Here Eq. 5 corresponds to the transformation of the u/v into an x measurements where dx/dy is the tilt of the stereo angle.
OT hits are again required to have drift distances between `minOTDrift` = -0.1 and `maxOTDrift` = 2.6 mm.
Then the extrapolated position $x_{extrapolated}$ is compared to the position of the hit x_{meas}. The hit is accepted if the difference is within:

$$\Delta x = \texttt{maxSpreadY} + \texttt{maxSpreadSlopeY} \times (q/p)^2[MeV^{-2}] \tag{40}$$

where `maxSpreadY` = 1.5 mm and `maxSpreadSlopeY` = 70 mm. Δx is enlarged by 1.5 mm in case of OT hits. For all hits fulfilling this cut, $x_{extrapolated} - x_{meas}$ is filled in a list.

3.5 Scan for Hough Clusters in Stereo

The scan for hit clusters of this list, is performed in the very same way as the scan for the x hit clusters. The minimum number of stereo planes required is four and the maximum size of the cluster is 3 mm for IT hits as starting hit and 4.5 mm for OT hits. In case several good stereo clusters are found the one with largest number of different planes is taken. In case of two clusters with equal number of planes the narrower cluster is selected as starting cluster for the 3D fit.

3.6 3D Fit & Outlier Removal

First all hits (both x and stereo) are updated for the current track parameterization. Then if possible the left right ambiguities of OT hits are resolved in the same way as described above (Eq. 32-33). Next the x projection is fitted. Here the x position of the stereo hits is included in the fit. Otherwise the same iterative fitting procedure as described above is used. Then a straight line fit to the y component of the stereo measurements is performed Eq. 13. Based on the new y parameterization of the track all hits are updated and a new fit in y space is performed. Again up to ten iterations are tested mainly to take into account the flip of left-right ambiguities from OT measurements. Iterations are stopped once the change between iterations in A_y is smaller than 0.05 mm and the one in B_y is smaller than 0.00005.
Once the fit was successful the stereo hit with the largest χ^2 contributions in case it is larger than `maxChi2` = 20 is removed. As well all stereo hits with a χ^2 contribution larger than 1000 are removed. In case hits in more then `maxPlanes` = 9 different (x and stereo) planes are left after the removal, the 3D fitting procedure is repeated with the left over hits. In case the highest χ^2 is smaller than 2 $\times$ `maxChi2`, the following iterations are performed with check on χ^2 contribution from the x hits as well. Once the highest χ^2 of all hits on the track is smaller than 20 the iterations

are stopped and we check once more for at least nine different (x and stereo) planes. As well the geometrical compatibility in y of the fitted tracks with the detector planes is tested again. All tracks which survived till here will compete in the final track selection.

3.7 Final Track Selection

The best track candidates are selected in several steps. First the change in the y position of the VELO seed and the track candidate in the T stations is tested. Deviations Δy between the straight line extrapolation of the VELO track (corrected for small changes due to the magnetic field) and the track candidate parameterization at $z = z_{ref}$ which are larger than tol_{Δ_y} are discarded.

$$tol_{\Delta_y} = \texttt{maxDeltaY} + \frac{1}{p^2}[GeV^{-2}] \times \texttt{maxDeltaYSlope} \tag{41}$$

where `maxDeltaY` = 30 mm and `maxDeltaYSlope` = 300 mm. Additionally a minimum number of hits is required. Here IT hits count with a weight of two (# hits = 2 $\times$ # IT hits + # OT hits). For tracks which mainly pass the IT region at least `minHits` = 14 hits have to be on the track, for tracks crossing the OT detector the according number is `minOTHits` = 16. Then the number of hits in different (x and stereo) planes are counted. Only track candidates with the largest number and with one plane less are considered further. Then a quality variable Q is defined which combines the deviation in y between VELO seed and track candidate in the T stations, the χ^2/ndf and the momentum of the track. Small values of Q indicate good tracks.

$$Q = \frac{5 \times \Delta y}{tol_{\Delta y}} + \frac{\chi^2/ndf}{10} + \frac{10}{|p|}[GeV^{-1}] \tag{42}$$

The track with the lowest Q value Q_{min} is then identified. Only tracks with a Q value within $[Q_{min}, Q_{min} + 1]$ are considered further. The distribution of the quality variable and its components is displayed in Fig. 6.

The last cut is on the hit content of the tracks. The track with the largest number of hits ($\#hits_{max}$) and all tracks with at least $\#hits_{max}$-2 or at least 22 hits are considered as final track candidates. Most of the time zero or one track survive this selection. In about 5% of the cases more than one track candidate is selected. The final track candidates are then put into the output track collection and the pattern recognition is repeated for the next VELO seed.

3.8 Adding TT Hits to the reconstructed Track

The search for additional hits in the TT station is performed as follows. First the VELO seed is extrapolated as straight line to the z position of the TT station. We then check for consistency of the extrapolated y position with the active region of the station within a tolerance tol_{TT}

$$tol_{TT} = \texttt{ttTol} + \texttt{ttTolSlope}/p[GeV^{-1}] \tag{43}$$

where `ttTol`= 2 mm, `ttTolSlope` = 20 mm; p is the momentum of the track as it is found by the fit of the VELO seed and the hits in the T stations. The positions of all hits in the station are then updated according to the track parameters to correct for potential slopes of the measurement planes

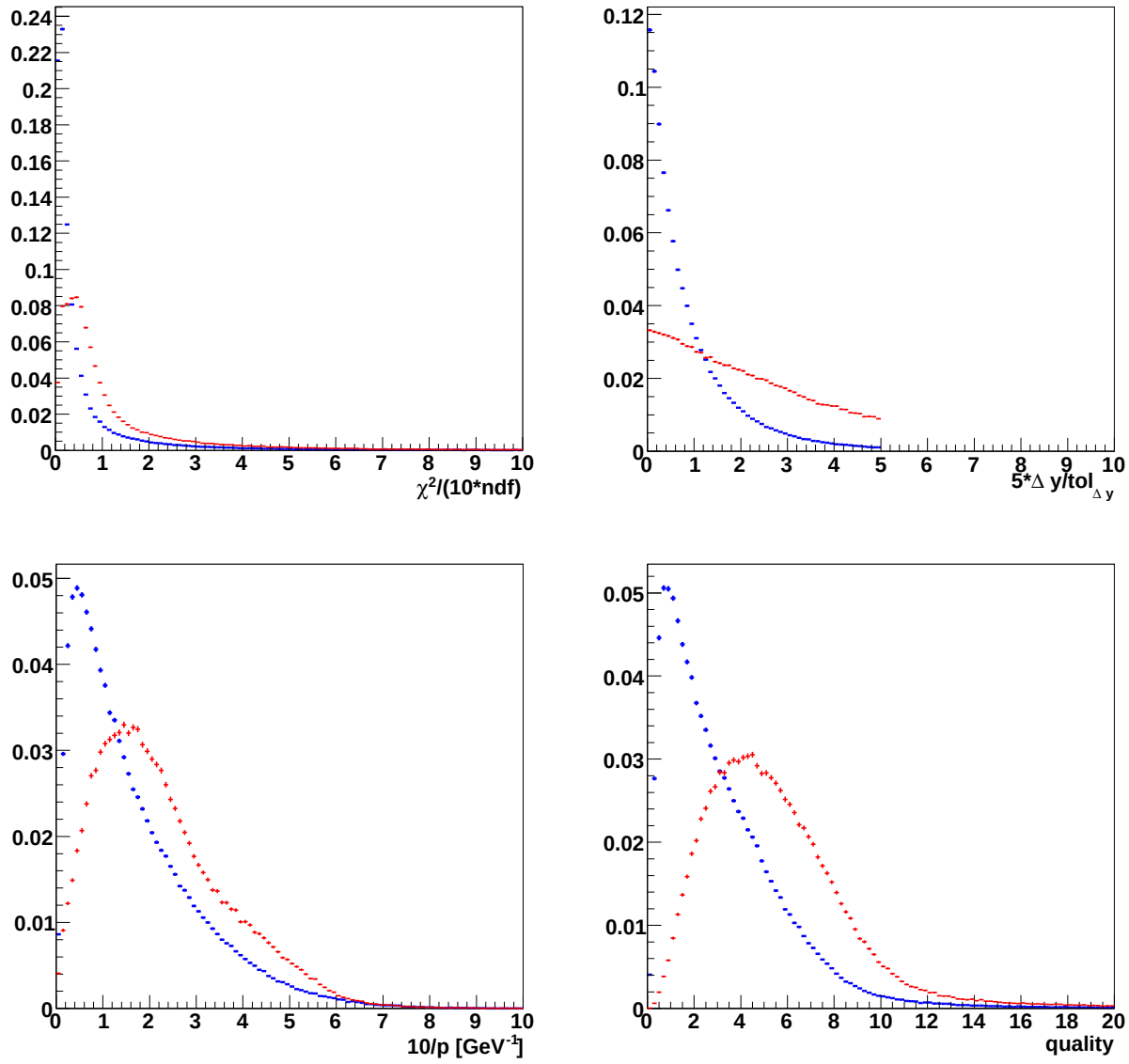

Figure 6 Quality variable and various of its contributions for good and bad track candidates. In blue/bold the distribution for good track candidates, red/thin for bad candidates.

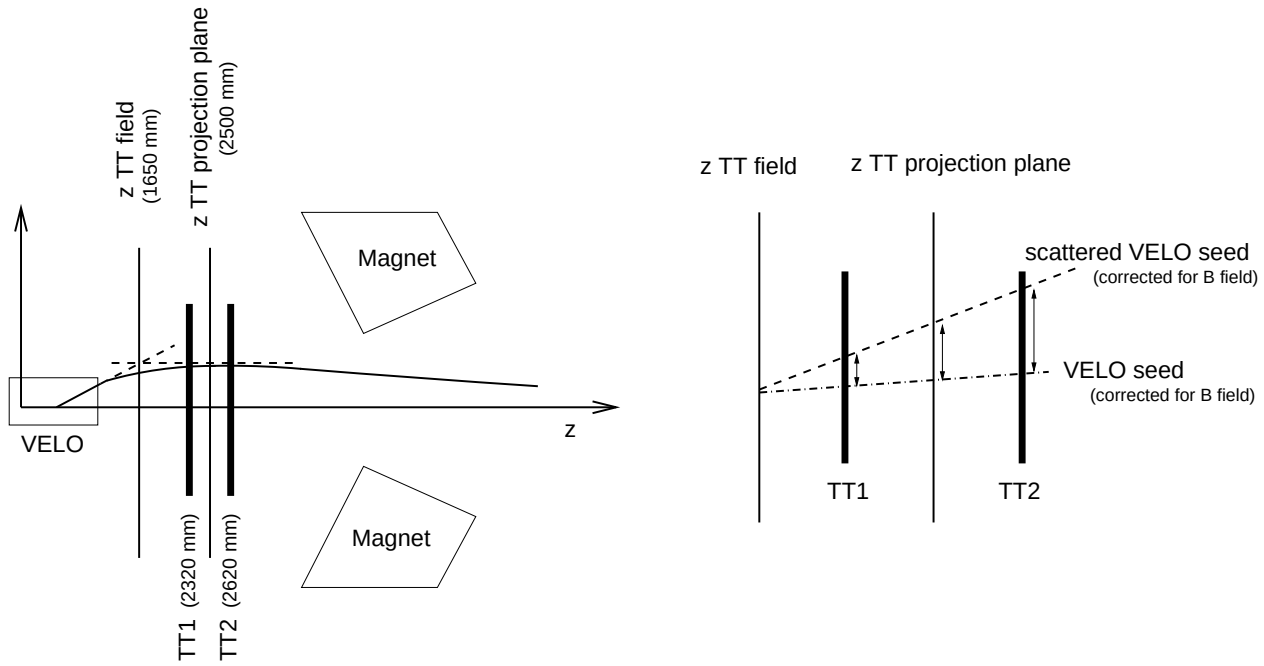

Figure 7 Left: Deviation from straight line extrapolation of VELO seed and real path of particle through the detector due to magnetic field. Right: Deviation from VELO seed and scattered seed. Multiple scattering changes mainly the slope of the track.

(Eq. 3-5). Next the x position of the track at the z position of the TT measurement is computed. Again the magnet field effect is described by one kink between a straight line in the VELO and a straight line in the TT stations at a focal plane (`zTTField` = 1650 mm) as shown in Fig. 7.

$$\begin{aligned} x_{extrapolated} &= x_{0,VELO\ seed} + (z_{meas} - z_{0,VELO\ seed}) \times dx/dz_{VELO\ seed} \\ &\quad + \texttt{ttParam} \times q/p[GeV^-1] \times (z_{meas} - \texttt{zTTField}) \end{aligned} \tag{44}$$

where `ttParam` = 30 mm. If the distance between $x_{extrapolated}$ and x_{meas} is smaller than tol_{TT} the distance is projected on a plane between the two TT stations at a z position of `zTTProj` = 2500 mm. The idea is that after the effect of the magnetic field is corrected the only difference between the VELO seed and the true path of the particle comes from multiple scattering (e.g. in RICH1) which mainly results in a change of the slope of the seed. Therefore the deviation in TT1 is expected to be smaller than the one in TT2 and the size of the deviation is supposed to scale linear in z. Therefore the following projection is performed:

$$x_{proj} = (x_{extrapolated} - x_{meas}) \times (\texttt{zTTProj} - \texttt{zTTField})/(z_{meas} - \texttt{zTTField}) \tag{45}$$

Next we search for a cluster of hits in the projection plane with at least hits in three different TT planes and a width of Δx:

$$\Delta x = 2\ \text{mm} + 0.25 \times |x_{first\ hit}| \tag{46}$$

where $x_{first\ hit}$ is the projected position of the starting hit of the cluster. Then a straight line fit in x (and a fit to a constant value in y is performed (Note, we only fit for deviations from the expected

curve due to multiple scattering here):

$$x(z) = offset + slope \times (z - \texttt{zTTProj}); \quad (47)$$
$$y(z) = offset_y; \quad (48)$$

where $offset$, $slope$ and $offset_y$ are the fit parameters. All hits are added to the fit. Additionally the x slope and the (x, y) position of the VELO seed enters as a constraint with $weight$ = 9.0/tol_{TT}^2. The $\chi^2/ndof$ of the fit is then derived as

$$\chi^2/ndof = (weight \times (offset^2 + offset_y^2 + (slope \times (\texttt{zTTProj} - \texttt{zTTField}))^2) \quad (49)$$
$$+ \sum_i (w_{hit\ i} \times dist_{hit_i}^2)/n_{hits} \quad (50)$$

$w_{hit\ i}$ is the weight of the measurement of hit i; n_{hits} is the number of hits on the track and $dist$ is defined as:

$$dist_{hit\ i} = x_{proj,hit\ i} - offset - slope \times (z_{hit\ i} - \texttt{zTTProj}) - offset_y \times sin(\theta) \quad (51)$$

where θ is the stereo angle. Hits with the largest χ^2 contribution to the fit are removed in case $\chi^2/ndof <$ `ttMaxChi2`, where `ttMaxChi2` = 3. If one cluster with hits from at least 3 different planes is found the TT hits are added to the track. In case several clusters which fullfill the requirement are found the one with the smallest $\chi^2/ndof$ contribution is choosen and its hits are added to the track.

3.9 Elimination of Clones/Ghosts

A significant fraction of ghost tracks[5] are related to wrong extrapolation through the magnetic field. Meaning the VELO part and the OT/IT part of the track are two well reconstructed track pieces but they actually do not belong to the same particle. During pattern recognition, we can only compare different OT/IT tracks which are potentially belonging to the same VELO seed. After the pattern recognition has been performed we can compare OT/IT track segments of all VELO seeds. In case two tracks have a significant fraction of OT/IT hits in common but are associated to two different VELO seeds we know that at least one of them is a ghost track. By comparing the two tracks we then try to identify the better one.
Each pair of two reconstructed forward tracks is examined. A clone couple is identified if more than 70 % of the IT+OT hits of at least one of the tracks are used in both tracks. In case one of the tracks has more than `deltaNumberInT` = 3 OT+IT hits or more than `deltaNumberinTT` = 1 TT hits more than its clone partner, the partner is removed from the track list. If non of the tracks is identified as significantly better, both tracks are kept. This procedure removes about 2% of the ghost rate with negligible drop in efficiency (<0.1%).

4 Performance Studies

The performance of the algorithm has been studied using data generated for the DC06 [1] production. Two data samples have been studied:

[5] track which can not be associated to a Monte Carlo particle, for precise definition see next section

- A sample of 20,000 $B^0 \rightarrow J/\psi(\mu^+\mu^-)K_s(\pi^+\pi^-)$ events generated at the default LHCb luminosity of 2×10^{32} cm^{-2}s^{-1};

- and a sample of 500 $B^0 \rightarrow J/\Psi(\mu^+\mu^-)K_s(\pi^+\pi^-)$ events generated at a luminosity of 5×10^{32} cm^{-2}s^{-1}.

The majority of the results were obtained with the first sample.

4.1 Definitions

Efficiency and ghost rates characterize the performance of a pattern recognition algorithm. In order to define efficiency and ghost rate we first have to introduce the definition of reconstructible particles and associated tracks. The standard definition for so-called reconstructible long tracks (tracks with associated measurements in the VELO and in the T stations) is

- The particle momentum at its production vertex is more than 1 GeV.

- The particle has at least three reconstructed clusters in the r sensors of the VELO

- and at least three reconstructed clusters in the ϕ detectors of the VELO.

- It has at least one reconstructed x and stereo hit in each of the tracking stations T1-T3.

- The particle does not interact hadronically before the end of the T stations.

- It is not an electron.

A track is associated to a Monte Carlo particle if at least 70% of its hit in the VELO and at least 70% of its T station hits are associated to the same particle. A reconstructed track which can not be associated to a Monte Carlo particle is a so-called ghost track.
With those definitions it is possible to introduce the definition for efficiency and ghost rate:

$$\begin{aligned} \text{efficiency} &= \text{N(reconstructible and reconstructed)/N(reconstructible)} \\ \text{ghost rate} &= \text{N(reconstructed and not associated to Monte Carlo)/N(reconstructed)} \end{aligned}$$

Both the efficiency and the ghost rate can be calculated in two ways. The first is to calculate these quantities on an event-by-event basis ("event weighted"). If values for the whole event sample are required the averages of the resulting distributions are used. The alternative is simple to calculate the efficiency and ghost rate on the whole sample of tracks ignoring which event the track came from ("track weighted"). Since there are large event-to-event fluctuations in the case of the ghost rate we will quote in the following the results of both methods.

	event weighted		track weighted	
momentum	efficiency	ghost rate	efficiency	ghost rate
> 1 GeV	85.9%	11.1%	84.8%	15.3%
> 5 GeV	92.9%		92.2%	

Table 1 Performance of the forward tracking algorithm.

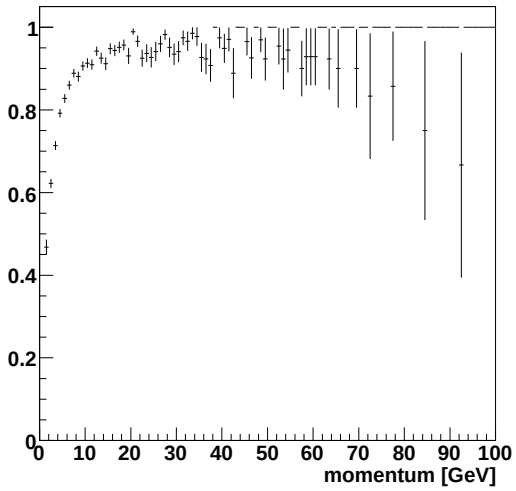

Figure 8 Track weighted efficiency as a function of particle momentum.

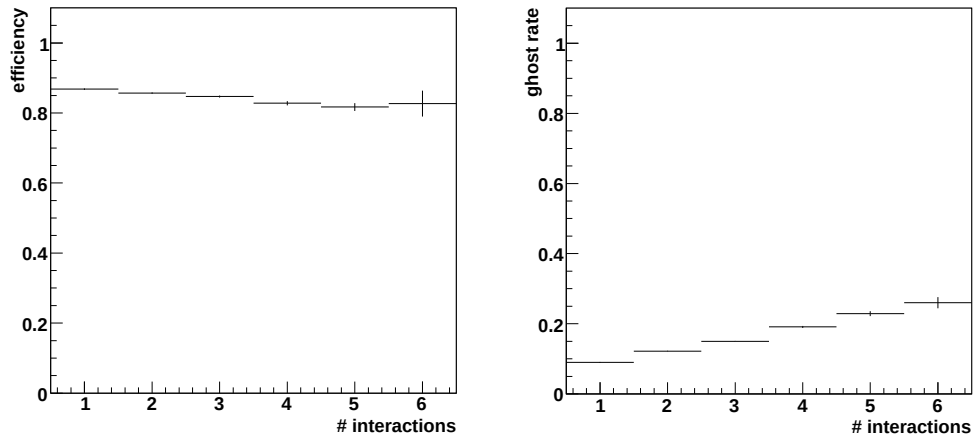

Figure 9 Event weighted efficiency (left) and ghost rate (right) versus the number of visible interactions.

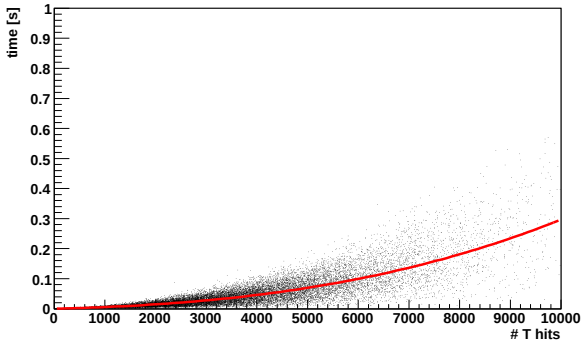

Figure 10 Algorithm time versus the total T station multiplicity.

4.2 Efficiency & Ghost Rate

The results obtained with the default settings of the algorithms are listed in Table 1.

Due to multiple scattering low momentum tracks are significantly harder to reconstruct, which causes the lower performance in the low momentum region (Fig. 8).

The performance of the forward tracking as a function of the number of visible interactions as defined in [3] has been investigated. Fig 9 shows the dependence of the efficiency and ghost rate on this quantity. It can be seen that the dependence of the efficiency on the number of visible interactions is quite weak. For each additional visible interaction in the detector the efficiency decreases by ≈ 1 %. The ghost rate shows a slightly stronger dependence on the number of visible interactions. For each additional interaction the ghost rate increases by $\approx 3\%$.
In addition, the performance with data generated at a higher luminosity of 5×10^{32} cm^{-2}s^{-1} has been studied. An average of about 2 interactions per event are expected at this luminosity. In this case an efficiency of 82.5/84.7% and a ghost rate of 22.4/15.0% is found, track weighted/event weighted respectively. If only the number of visible interactions in the event spill effects the performance of the track reconstruction then efficiency and ghost rates for arbitrary luminosity can be derived directly from Fig. 9. At higher luminosities how ever this extrapolation will break down due to increased spillover that further increases occupancies and detector dead-time.

4.3 Time Performance

Finally, the CPU performance of the algorithm has been evaluated on a machine[6], which is about a factor 1.8 slower than the lxplus cluster at CERN. The algorithm runs in a time of 55 ms per event using the standard LHCb compilation options. Fig 10 shows the time per event versus the total number of hits in the T stations. The dependency of the time spent per event on the T station hit multiplicity can be described by a third order polynomial:

$$t(n_{T\ hits}) = (4.17 \cdot 10^{-6} \times n_{T\ hits} + 1.34 \cdot 10^{-9} \times n^2_{T\ hits} + 1.21 \cdot 10^{-13} \times n^3_{T\ hits})\ s \qquad (52)$$

[6]Dual Core and Opteron Processor 280, 2.4 GHz

5 Summary

We presented a detailed description of the forward tracking algorithm. An event weighted reconstruction efficiency of 85.9% with a ghost rate of 9.0% for tracks with a momentum above 1.0 GeV has been obtained.

6 References

[1] Gauss v25r7, Bool v12r10, Brunel v30r14, XmlDDDB v30r14

[2] M. Benayoun, O. Callot, *The Forward Tracking, An Optical Model Method*, LHCb-note 2002-008

[3] The LHCb Collaboration, *Reoptimized Detector Design and Performance*, CERN/LHCC LHCC-2003-030

[4] The parameters are set in `PatFwdTool.opts` in the `Pat/PatForward` package of the LHCb software