LHCb note 2007-023

Updated Performance of the T-Seeding

R. Forty, M. Needham
CERN

March 21, 2007

Abstract

Improvements to the performance of the track seeding since the start of the DC '06 data challenge are described. For tracks above 2 GeV an efficiency of 96.3 % is achieved for a ghost rate of 8.1 %. In addition, the expected performance of the algorithm during the 2007 pilot run is discussed.

1 Introduction

An efficient algorithm for standalone reconstruction in the T-stations was described in [1]. In this note improvements to the algorithm that occurred after the DC '06 data challenge are described. The algorithm described here corresponds to that released in Brunel v31r0.

This note is organized as follows. First, the improvements to the algorithm that have been made are discussed. This is followed by a discussion of the performance. Finally, the performance of the algorithm in the conditions expected during the 2007 pilot run is described.

2 Improvements to the Algorithm

Since the time of studies presented in [1] many changes to the seeding algorithm and code have been made. This has resulted in a large improvement in performance as will be seen in Section 3. The most important of these changes are summarized below.

The first type of changes are purely technical. To reduce the number of mathematical operations two tricks are now used throughout the code. The first is to use Horner's form [2] for polynomial evaluation. The second is to replace repeated divisions by a given value with multiplication by the reciprocal of that value. These changes gain a significant amount of time — especially in the x search where the amount of numerical operations made is large. In addition, in the case of the x search a penalty of 20 ms was observed due to the creation of a STL vector [1] deep within a nested loop. Better performance was obtained by creating the vector once outside the loop and clearing it when necessary. Finally, as discussed in [3, 4] in the in the majority of the algorithm STL binary search methods are used to give fast data access. Previously, these tricks had not been implemented in the case of the stub finding and linking steps as they contributed only a small proportion of the reconstruction time. Implementing these tricks the stub finding steps reduce in time from around 5 ms to 1 ms.

Several algorithmic improvements have been made mainly to the x search. First, the collection of hits in the window defined by the initial pair of hits in T1 and T3 has been improved. For the Outer Tracker case the centre of the window in T2 is calculated including a correction that takes account of the displacement of the particle in the fringe field [5]. This is calculated under the assumption that the track originated close to the interaction point and observing that:

$$\frac{\Delta x_{T2}}{\Delta x_0} = \frac{\int_0^{T3} Bdl}{\int_{T1}^{T3} Bdl} \simeq 0.0022.$$

It is also required in the case of Outer Tracker seeds that there be more hit found in the T2 station. This gives slightly improved efficiency and also speeds up the algorithm.

It was observed that after the x search there was a non-negligible rate of clone tracks. To remove these after the x search the tracks are ranked according

[1] In general the performance of STL vectors is very good. The reason the penalty is large in this case is due to the very high combinatorics in hot events. For example in one hot event profiling showed that the vector was created and destroyed 300,000 times !

to the number of measurements and fit χ^2. Starting with the best candidate each track is considered in turn. If less than 10 % of the hits on the track are used the track is selected and the corresponding hits flagged as used. On the other hand if more than 10 % of the hits are used the track is removed from the list of good candidates. This change reduces the algorithm speed by 20 ms without effecting the efficiency. Finally, the stub finding and linking steps have been re-tuned to reduce the ghost rate whilst maintaining high efficiency. The tuning and cuts used now more closely follows that discussed in [4]. As part of this work the calculation of the likelihood has been extended to all track types.

The net effect of these changes both to speed up the algorithm significantly and also to reduce the ghost rate. Since the performance of the algorithm for hot events is now much improved it was decided to loosen the cut on the total number of hits in the T-stations to 12000. This means that at the nominal luminosity of 2×10^{32} cm^{-2}s^{-1} all events are reconstructed. In addition, the criteria used to removed IT hot-spots has been loosened from 32 to 48 strips.

3 Performance

With the default settings of the algorithm an event weighted efficiency of 92.7 % is found [2]. Fig 1 shows the efficiency as a function of momentum. It can be seen that above momenta of 5 GeV the efficiency plateaus at 96 %. Below 2 GeV the efficiency of the algorithm falls rapidly. As discussed in [1] another source of inefficiency is the 7 cm region around $y = 0$ cm in the Outer Tracker where the detector is only 50 % efficient. If this region is excluded the efficiency at high momentum increases to 97 %. The efficiency for reconstructing all particles in the T acceptance is 78.4 %.

The efficiency for reconstructing tracks that originate from B decays has also been investigated. The results are summarized in Table 1. For both muons from $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$ and electrons from $\mathrm{B_d} \rightarrow J/\psi(e^+e^-)\mathrm{K_S}(\pi^+\pi^-)$ a performance comparable to that obtained with the inclusive track sample is found. For the electron case the performance is slightly worse reflecting the fact that bremsstrahlung in the material of the detector makes them harder to reconstruct. The performance for pions from $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$ where the pion give sufficient hits in the VeLo to be reconstructible as Long

[2]The track weighted efficiency would be 92.4 %.

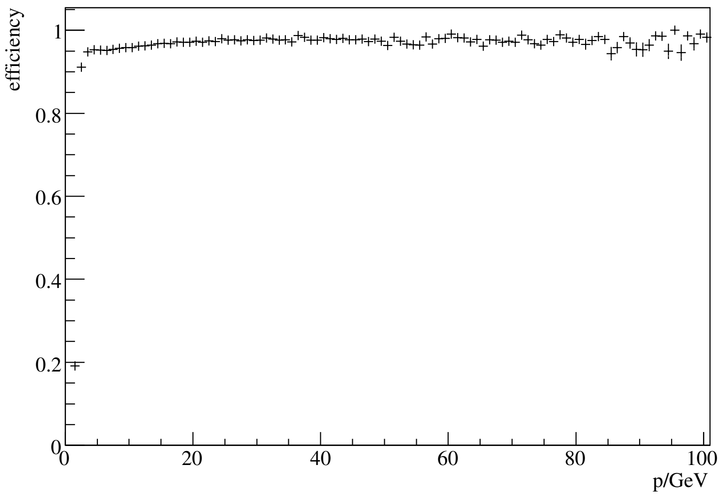

Figure 1: Track finding efficiency as a function of the track momentum.

Track type	$\overline{p}$/GeV	Track efficiency %
$\mu^{\pm}$ from $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$	33	96 ± 0.4
$e^{\pm}$ from $\mathrm{B_d} \rightarrow J/\psi(e^+e^-)\mathrm{K_S}(\pi^+\pi^-)$	34	94 ± 0.3
$\pi^{\pm}$ from $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$ in Long acceptance	12	92 ± 1
$\pi^{\pm}$ from $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$ in Downstream acceptance	14	94 ± 1

Table 1: Efficiencies for reconstructing tracks from specific B final states.

Tracks is slightly worse. This may be partially explained by the lower momenta of these tracks. The performance for $\mathrm{B_d} \rightarrow J/\psi(\mu^+\mu^-)\mathrm{K_S}(\pi^+\pi^-)$ where the pions are reconstructible as Downstream Tracks is comparable to that obtained with the inclusive track sample.

The event weighted ghost rate is 8.1 % [3]. If a higher purity is needed the ghost rate can be reduced, at some cost in efficiency, by cutting harder on the likelihood variable. Fig. 2 shows the efficiency versus ghost rate for various

[3]The corresponding track weighted ghost is 11.1 %.

cuts on this variable. It can be seen from this plot that ghost rate can be reduced by a factor of almost two for a loss in efficiency of ~ 1 %. As noted in [1] well reconstructed tracks from previous spills are classified as ghosts. In dedicated studies it is found that such tracks account for 20 % of the observed ghost rate. This is a larger fraction than before and indicates that whilst the ghost rate has decreased the efficiency to reconstruct tracks from previous spills has remained constant.

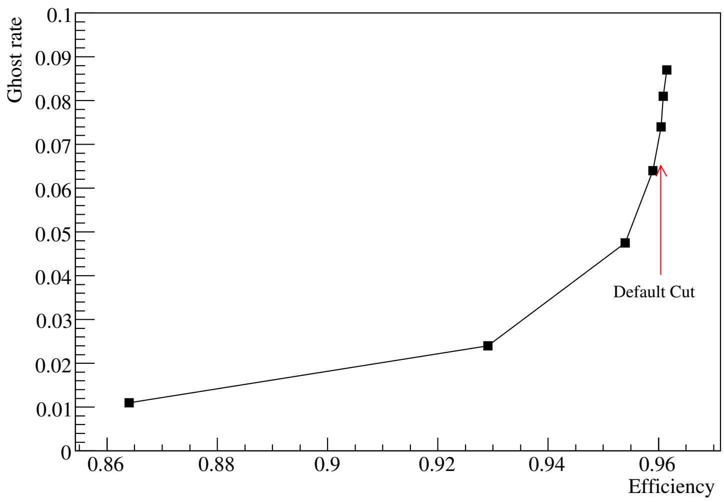

Figure 2: Efficiency for tracks with p > 2 GeV versus ghost rate for various cuts on the Likelihood. From left to right the cut values are -15 , -20, -25, -28, -30, -32 and -35.

The performance as a function of the number of visible interactions as defined in [6] has been investigated. Fig. 3 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 $\sim$ 0.9 % The ghost rate shows a clear dependence on the number of visible interactions in the detector. For each additional interaction the ghost rate increases by ~ 3.1 %.

In addition, the performance with data generated at a luminosity of 5×10^{32} cm^{-2}s^{-1} has been studied. In this case an efficiency of 91.6 % and a ghost rate of 15 % is found. It should be noted that if only the number of visible interactions in the event spill effects the performance of the track reconstruction then efficiencies and ghost rates for an arbitrary luminosity can be derived

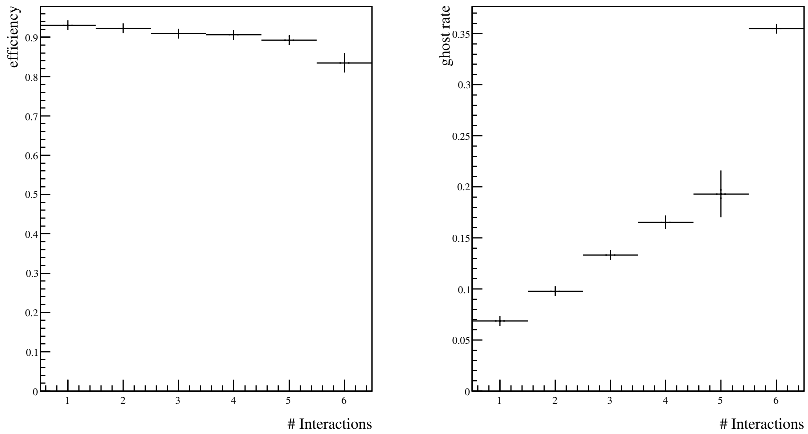

Figure 3: Efficiency (left) and ghost rate (right) versus the number of visible interactions.

directly from Fig. 3. At a luminosity of 5×10^{32} cm^{-2}s^{-1}, on average there are two visible interactions per B event. From Fig. 3 the efficiency in this case would be expected to be 92.4 % and the ghost rate 11 %. The difference between the predicted and observed ghost rates is explained by the increase in spillover between the two luminosities [4].

Finally, the CPU performance of the algorithm has been evaluated. On a 2 GHz Intel Centrino machine the algorithm runs in a time of 79 ms per event using the standard LHCb compilation options for gcc. Fig. 4 shows the time per event versus the total number of hits in the T-stations. The observed behaviour can be parameterized as:

$$t = 5.9 \times 10^{-9} \times N^2[s].$$

The improvements discussed in Section 2 mean that the algorithm time now goes quadratically with the number of hits rather the the cubic behaviour found in [1]. As noted in [1] a 15 % improvement in speed is observed if the compiler flags that allow the use of the SSE registers are enabled [5].

[4] Recall that spillover constitutes a sizeable fraction of the ghost rate.

[5] The gcc compiler flags: -msse2 -mfpmath=sse.

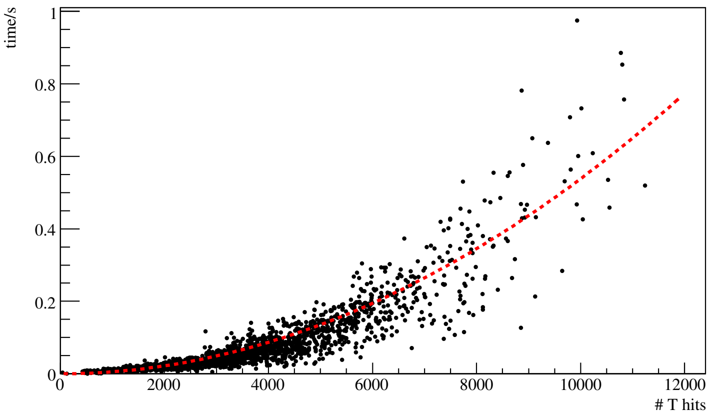

Figure 4: Algorithm time versus the total T station multiplicity. The dashed line is the parameterization given in the text.

4 Pilot Run Conditions

The performance of the algorithm in the conditions expected during the 2007 pilot run has been investigated. During this data taking period 450 GeV beams will be collided at low luminosity. It is forseen that data will be taken both with magnet on and magnet off. The latter data will be used primarily to align the detector. For these studies samples of 5000 minimum bias events generated in these conditions have been used [7].

For the case of magnet on, an efficiency of 92.4 % is found rising to 98 % above 5 GeV. The ghost rate is 1.0 % and the algorithm takes 2.9 ms to run on a 2 GHz machine. The improved performance compared to standard running can be attributed to the much reduced occupancies in the T-stations under these conditions: there is only one proton-proton interaction per crossing, producing on average six tracks within the acceptance and no spillover.

For the case of magnet off a special tuning of the algorithm has been provided. This tuning take accounts of the fact that with magnet off tracks are straight lines originating close to the primary interaction point. A set of job options that provide a tuning of the T-seeding for these conditions is available from Brunel v31r0 onward. With this tuning an efficiency of 96.4 % is found for all momenta with a ghost rate of 3.6 %. The algorithm takes about 5 ms per event. The higher ghost rate and increased time with magnet off compared to magnet on is expected as the occupancy is higher in this case due to low momentum particles that are no longer swept out of the acceptance by the magnet.

At the end of the algorithm the seeds are fitted using a Kalman Filter [1, 8]. With the magnet off no estimate of the track momentum is available. To obtain a reasonable estimate of the track covariance matrix for the fitted seeds some best guess of the momentum needs to be provided. To achieve this the same solution as used in the fit of VeLo tracks has been adopted. This exploits the correlation of the momentum with polar angle of the track [9]. Using the average p_t of 400 MeV per track and the measured polar angle the track momentum can be estimated with a precision of 50 %. This value is used together with a randomly assigned charge to provide an estimate of q/p for the track fit. The relative error on this value is set to 1×10^{-3} to prevent it being changed during the fit. This procedure leads to reasonable pulls as can be seen in Fig. 5.

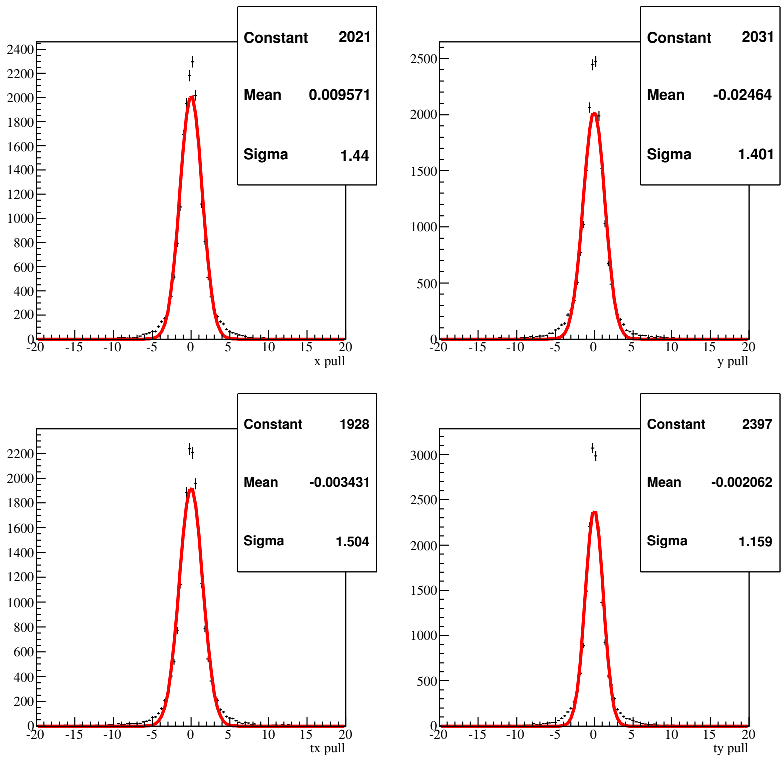

Figure 5: Pulls of the track parameters x, tx, y and ty at the last measured point for field off running.

5 Summary

In this note the improvements to the Tsa seeding that have been implemented since the start of the DC '06 reconstruction phase have been discussed. For tracks above 2 GeV an efficiency of 96.3 % is achieved for a ghost rate of 8.1 %. The timing of the algorithm has also been improved — it now runs in a time of 79 ms per event.

In the future more improvements to the algorithm are forseen. Further optimization of the stereo search and the stub-finding parts of the algorithm seem possible which should lead to still lower ghost-rates and CPU time.

References

[1] M. Needham and R. Forty. Standalone Track Reconstruction in the T-Stations. LHCb-note 2007-022.

[2] W. Horner. A new method of solving numerical equations of all orders by continous approximation. *Philisophical Transactions of the Royal Society of London*, pages 308–335, 1819.

[3] M. Needham. The Tsa Reconstruction Framework. LHCb-note 2007-037.

[4] M. Needham. Tsa: Fast and Efficient reconstruction for the Inner Tracker. LHCb-Note 2004-075.

[5] O. Callot. New approach for a Fast Seeding. T-Rec Meeting 23/10/2006.

[6] The LHCb Collaboration. Reoptimized Detector Design and Performance. CERN/LHCC LHCC-2003-030.

[7] https://twiki.cern.ch/twiki/bin/view/LHCb/SpecialDataSets.

[8] E. Rodrigues . The LHCb Kalman Fit. LHCb-note 2007-014.

[9] F. Teubert J. Holt and E. Tomalin. The LHCb level-2 Trigger. LHC-note 98-047.