

Performance of a LVL2 Trigger Feature Extraction Algorithm for the Precision Tracker

John Baines¹

Rutherford Appleton Laboratory, Chilton, Didcot, UK.

Reinier Dankers²

NIKHEF, Amsterdam, The Netherlands.

Sergey Sivoklokov³

Institute of Nuclear Physics, Moscow State University, Moscow, Russia

Abstract

An algorithm to search for isolated high p_T tracks in the ATLAS precision tracker suitable for use in the LVL2 trigger is described. A histogramming method employing a Hough transform is used to select sets of points forming track candidates. For each selected set of points, fits are performed to all combinations of one point per detection plane. The best candidate is chosen on the basis of the fit residuals. The algorithm has been implemented in the t2scFex package of the ATRIG trigger simulation program and also for benchmarking studies in a stand-alone program SCTFEX. Efficiency, fake track rate and timing measurements are presented for a number of algorithm options.

1. email: j.t.m.baines@rl.ac.uk

2. email: r78@nikhef.nl

3. email: s.sivoklokov@cern.ch

1 Introduction

The first level trigger (LVL1) uses information from the calorimeter and muon systems to reject un-interesting events and to identify Regions of Interest (RoI) within potentially interesting events. These RoIs are analysed in more detail at higher trigger levels where finer granularity calorimeter information is available and where the information from the inner detector can be utilised. The LVL1 trigger produces RoIs for jets, e/γ , τ/hadron and μ . In this note, a Feature Extraction (FEX) algorithm designed to search within an e/γ RoI for a high p_T track is described.

For jet events after the LVL2 calorimeter selection, $\sim 90\%$ of e/γ RoI are due to a photon. Although $\sim 20\%$ of these photons subsequently undergo a conversion in the inner detector, requiring a track provides a means to reduce the trigger rate due to jets significantly. This is important as it allows a lower calorimeter E_T threshold to be used for the electron trigger than that used for the photon trigger.

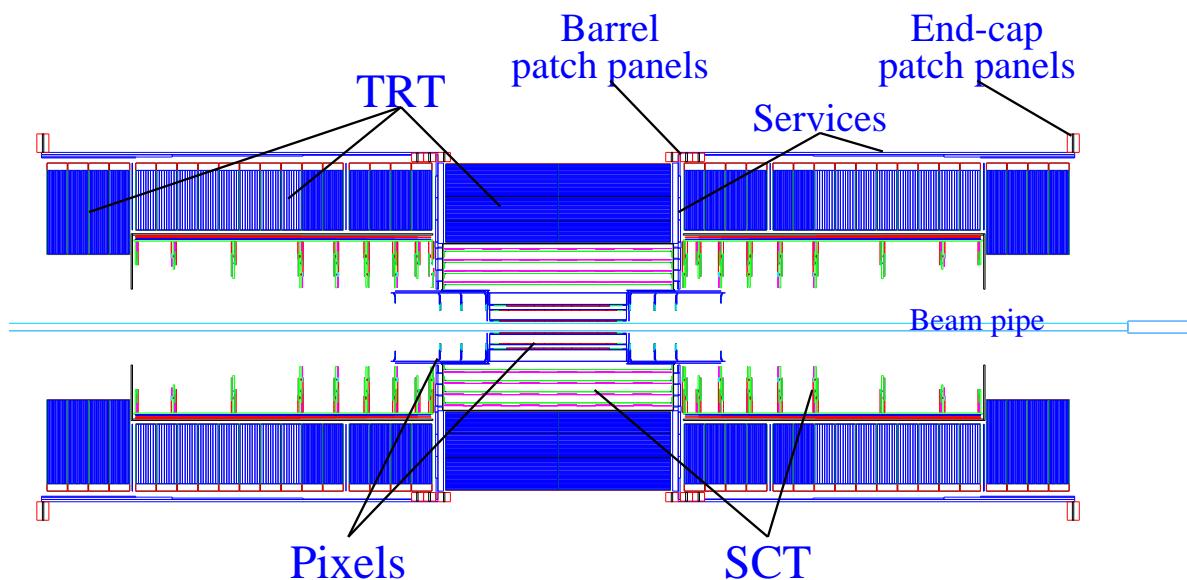


Figure 1-1 The ATLAS inner detector

The inner detector of ATLAS consists of three parts, the pixel detector system, the Semi-Conductor Tracker (SCT) and the Transition Radiation Tracker, see Figure 1-1. In one possible trigger implementation, track searches are performed independently in the precision tracker (SCT and pixel detector) and in the TRT. In this note an algorithm to search for tracks in the precision tracker is presented. The geometry of the SCT and pixel systems is described in more detail in Section 3.

Prior to the track search the raw data passes through several processing steps. The complete set of steps from raw data through to the reconstruction of track candidates are as follows:

- **Pre-selection:** The input data consists of the information from all Read Out Buffers (ROB) which have at least one module lying within the geometrical bounds of the RoI. A number of modules are connected to each ROB (in the current design, 24 SCT modules or 48 pixel modules¹). Typically only a fraction of these modules lie within the RoI. The pre-selection step selects the data from the subset of modules which lie within the RoI.

- **Clustering:** a track can produce hits, signals above the electronic read-out threshold, on more than one pixel or strip. The clustering stage combines the hits on adjacent strips/pixels into a single cluster.
- **Association of stereo information (SCT only):** An SCT module consists of two layers (ϕ and stereo) each providing a 1-dimensional measurement. Detectors in the stereo-layer are rotated with respect to those in the ϕ -layer in order to provide information in a second dimension. Clusters from the ϕ -layer are combined with compatible clusters in the stereo-layer in order to produce a 2-dimensional point represented in the local coordinate system of the detector module. For this study, only points formed from the association of a ϕ and a stereo cluster are used in the subsequent steps. Un-associated clusters are rejected.
- **Space-point formation:** Transformation from the two-dimensional representation in the local coordinate system of the module to a three-dimensional point in the global coordinate system.
- **Post-Selection:** Once three-dimensional points have been formed a further level of geometrical selection is possible. This time only selecting space points lying within the ROI. However this post-selection has not been used for the results presented here.
- **Track Candidate Search:** Sets of points likely to lie on a track are selected using a histogramming method.
- **Fit and track candidate selection:** For each set of points selected by the histogramming stage, fits are made to all combinations of one point per detection plane. Track quality cuts are applied and the best candidate is chosen on the basis of p_T and the residuals of the points with respect to the fitted track.

The first 5 steps, up to and including post-selection, can be considered as data preparation and are described in a separate note [1]. The details of the implementation of these steps have yet to be finalised. All steps might be executed in the FEX processor or it is possible that one or more steps might be performed before the data is transferred to the FEX processor. The last two stages, which constitute the track reconstruction algorithm, are the subject of this note.

Note that the input to the algorithm described here is a list of three-dimensional space-points. For the SCT, these points are formed by combining information from the ϕ and stereo layers. This is not the method used offline, where ϕ and stereo layer clusters are treated independently through-out the pattern recognition process. An advantage of the method used here is that after the data preparation steps, points from the pixel detector and SCT have the same format. Barrel and end-cap points also have the same format and differ only in the precision of the R , z and ϕ measurements. Hence another benefit is that the transition from barrel to forward geometry is treated automatically.

A disadvantage of the use of space-points is that ghost points will be formed when there is more than one stereo-layer cluster in the region of overlap with the ϕ -layer cluster (each stereo strip crosses about 60 ϕ -strips). A ghost point arises from the incorrect association of a cluster in one layer with a cluster in the other layer that is due to a different track or due to noise. Since the ϕ -stereo association is performed on a module-by-module basis, there is also a small loss of efficiency at the edges of the module due to the cases where tracks pass through only one layer of the module. This efficiency loss could be reduced with a more complex algorithm which searches between neighbouring modules for stereo associations.

1. These numbers may be doubled in the final design.

The algorithm described in this note has been implemented in C as the module `t2scFex` in the trigger simulation package ATRIG [2] and also in C++ as part of the stand-alone package SCTFEX [3]. Within ATRIG, the LVL1 trigger is simulated and ROI created. The LVL2 algorithms are then run on the data inside these ROI. ATRIG also has the capability to write out the data from within the ROI to an ASCII data file. These ASCII data files provide the input for the SCTFEX package. The ATRIG implementation was used for the efficiency measurements shown in this note. The probability of reconstructing a fake track has also been measured for artificial ROI containing only the pile-up of minimum bias events. Care was taken in SCTFEX to implement the algorithm in a time-efficient way. The measurements of execution time are shown for the SCTFEX implementation. The algorithms are almost identical in the two cases. Where there are differences, these will be highlighted. Efficiency measurements were made with the SCTFEX algorithm as a cross-check that the same results are obtained from the two packages¹.

Results are given for a number of algorithm options illustrative of the trade-off between efficiency and execution time and efficiency and fake track probability. An overall assessment of performance can only be made after the information from different systems has been combined to form the trigger-objects used in the final trigger decision. This is done in Ref. [6] where efficiencies and trigger rates are presented for the electron trigger-object formed from the combination of information from the precision tracker (using the algorithm presented here), TRT and calorimeter.

2 Data-sets

The performance of the FEX algorithm has been determined using samples of single particles with and without pile-up for efficiency measurements and di-jet events with pile-up for timing measurements. The latter were produced from 10^6 generated PYTHIA di-jet events. To save computation time, a particle level filter was applied to reject events outside the inner detector acceptance or with low p_T , details can be found in Ref. [7]. The surviving sample of jets and the full samples of single particles were passed through a full GEANT simulation of the ATLAS Detector. The version of the ID design used for the simulation is that described in the section 3 of Ref. [8]. Details of the simulation of the calorimeter are given in [9]. Following the detector simulation, a loose set of calorimeter cuts, designed to reject events that would clearly fail the LVL1 e/ γ trigger, were applied to the jet sample. The pile-up of minimum bias events corresponding to running at design luminosity ($10^{34} \text{ cm}^{-2}\text{s}^{-1}$) were added to the surviving sample of jet events.

The following data-sets were used for the measurements presented in this note:

- Single particle events were generated at two p_T values, 20 GeV (μ, e^+, e^-) and 30 GeV (e^+, e^-), corresponding to the E_T threshold values currently envisioned for the LVL1 e/ γ trigger at low and high luminosity respectively. The $p_T = 30$ GeV data-sets were produced both with and without the addition of pile-up.
- Jet events with the addition of pile-up corresponding to design luminosity. The trigger rate after the LVL1 trigger will be dominated by background from jet events. It was this sample, therefore, which was used for the timing measurements.

1. The `t2scfex` version of the algorithm has also been implemented in the C/C++ algorithm development environment, CTrig [4], and in the LVL2 test-bed software framework [5]. Timing and efficiency measurements have also been made in these frameworks.

3 Geometry

The data-sets used for these studies were generated for the ID TDR [8]. The geometrical layout used in this simulation is described below. There have been a few changes to the detailed design of the inner detector since these data-sets were generated and the major changes will be listed at the end of this section. More details can be found in the Pixel TDR [10].

The pixel detector provides 3 - 4 measurements along a track, and the silicon micro-strip detectors 4 - 5 measured points (3 in a small transition region from barrel to end-cap geometry). Individual detectors are arranged in concentric cylinders in the barrel, and in disks in the end-cap regions, see Figure 1-1. There are overlaps between adjacent detectors in z (barrel) or R (end-cap) and in $R-\phi$. It is therefore possible for a single track to give two measured points in a single plane in small regions of the detector.

The inner “B-layer” of the pixel detector is at a radius of 4 cm and provides coverage over the entire pseudo-rapidity range $|\eta| < 2.5$. At least two additional measurements on a track are provided by barrel layers at radii of 11 cm and 14.2 cm and by disks, of which there are four in each end-cap in the ID-TDR design. The individual sensitive element in the barrel (end-cap) is a pixel 50 μm in $R-\phi$ and 300 μm in z (R). The positional resolutions are given in Table 3-1. The efficiency and noise occupancy used for the simulation are given in Table 3-2.

The SCT is composed of four layers of modules in the barrel and nine discs in each end-cap. The transition region from barrel to end-cap geometry occurs in the pseudo-rapidity range $1.16 < |\eta| < 1.64$. An SCT module consists of two back-to-back detectors, one with strips parallel to the beam-pipe in the barrel and radial in the end-caps, the other with strips rotated by ± 40 mRad to give a stereo measurement.

There have been several sets of changes to the material description and to the details of the detector design since the data-sets were generated for the ID TDR. The main changes are as follows (further details can be found in [10],[11],[12]):

- The sign of the tilt angle has changed for the pixel barrel modules. This has the effect of reducing the width of the cluster in $R-\phi$ and reducing the occupancy. Due to an increase in the signal to noise ratio the $R-\phi$ position resolution is also improved;
- The description of the material in the inner detector has been improved in the simulation and the routing of the services from the B-layer has been changed;
- There is an additional 5% X_0 of material at the outer radius of the end-cap TRT straws.
- The pixel detector thickness has been increased from 150 μm to 250 μm (200 μm for the B-layer). The material has increased to 1.76% X_0 per layer in the barrel and to 1.65% X_0 in the end-caps.

Table 3-1 Resolutions for the pixel and SCT subsystems. The values given for the SCT are for a module.

detector	$\sigma(R\phi)$ (μm)	$\sigma(R)$ (μm)	$\sigma(z)$ (μm)
pixel	12	66	77
SCT	16	580	580

Table 3-2 The detector efficiencies and noise occupancies used in the simulation.

Detector	Efficiency (%)	Noise (%)
Pixel	97	0.001
SCT	97	0.01

- In each end-cap there are now five pixel discs each with a single ring of detectors as opposed to the ID TDR arrangement of three discs with two rings and one disc with a single ring.
- The radii of the barrel pixel layers have been reduced slightly. All layers are now the same length.

The overall effect on performance of the changes in the geometrical layout are small. The change of sign of the tilt angle for the Pixel barrel layers will result in smaller clusters which will reduce the time taken for the pixel clustering step described in [1]. The changes in the amount of material represent a small fraction of the total material inside the radius of the calorimeter and hence the impact on performance is minor.

4 Algorithm Description

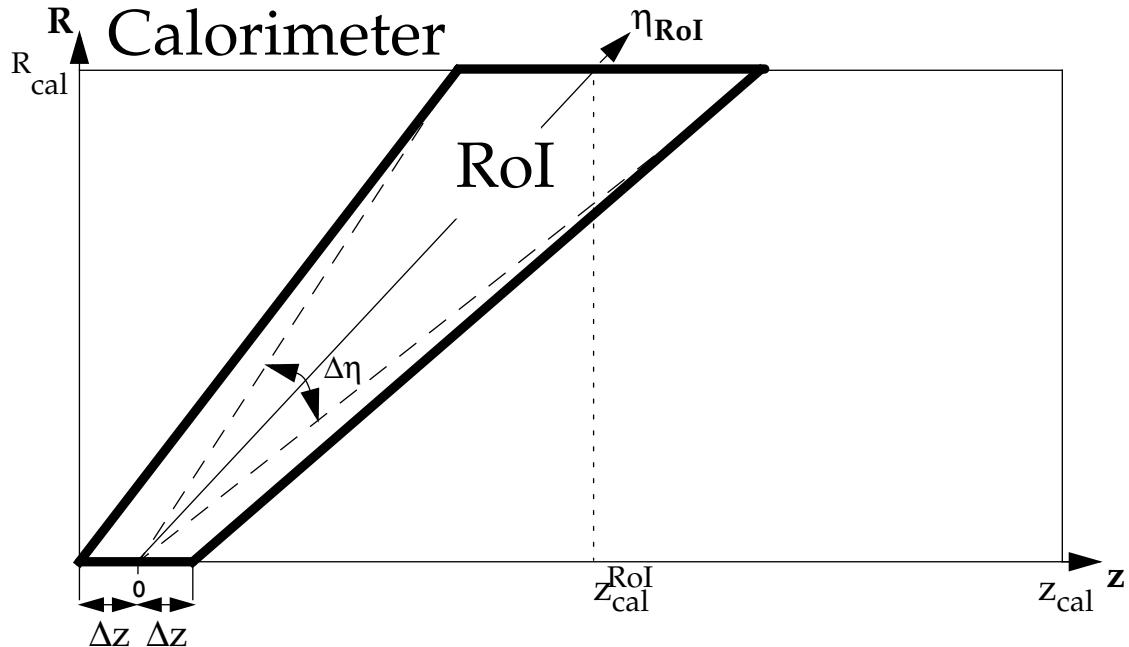


Figure 4-1 RoI definition used for the precision tracker FEX. Dimensions are $\Delta\phi \times \Delta\eta = 0.2 \text{ Rad} \times 0.2 \text{ Rad}$, $\Delta z = 11.2 \text{ cm}$.

The algorithm consists of a track candidate search followed by track-fits performed to select the best candidate and to determine the track parameters. The purpose of the candidate search is to find sets of points that lie within roads thereby reducing the number of point combinations to be investigated at the track fitting stage. A histogramming method based on a Hough transform is used. The roads correspond to bins of the histogram in transform space. The candidate search is performed within a RoI identified by an external system. The shape and dimensions of the RoI used in this study are given in Figure 4-1. This corresponds to the dimensions of a LVL1 e/ γ RoI. In a trigger architecture with a sequential processing scheme it would be possible to use a smaller RoI based on the LVL2 calorimeter information.

4.1 The Hough Transform

The algorithm makes use of the fact that the trajectory of a charged particle is linear in the ϕ - z plane and in the ϕ - R plane approximates to a straight line for particles with high p_T produced close to the origin. The candidate search looks for points which lie on a straight line in the appropriate projection; ϕ - z for the end-cap region ($|\eta_{\text{RoI}}| > 1.5$) and ϕ - R for the barrel region ($|\eta_{\text{RoI}}| < 1.5$). The trajectory of a particle produced at $(R, \phi, z) = (R, \phi_0, z_0)$ can be described in terms of an intercept, ϕ_0 (ϕ'_0), and a slope, C_t (C_z), in the ϕ - R (ϕ - z) projection respectively:

$$\begin{aligned} \phi &\approx \phi_0 + C_t R & (\phi\text{-}R \text{ projection}) \\ \phi &= \phi'_0 + C_z z & (\phi\text{-}z \text{ projection}) \end{aligned} \quad 4-1$$

where

$$\phi'_0 = \phi_0 - z_0 C_z ,$$

The slope is inversely proportional to the p_T of the track (in units of GeV):

$$\begin{aligned} C_t &\approx 0.003 B_z q / 2 p_T & (\phi\text{-}R \text{ projection}) \\ C_z &= 0.003 \tan(\theta) B_z q / 2 p_T & (\phi\text{-}z \text{ projection}) \end{aligned} \quad 4-2$$

where θ is the polar angle of the track, B_z is z component of the magnetic field strength (in units of T) and q is the particle charge (in units of e).

The principle of the Hough transform is to use each point in turn to define a series of entries in a histogram constructed in (ϕ_0, C) space. For each point, the value of ϕ_0 is calculated for each value of C in steps between $-C_{\text{max}}$ and C_{max} and an entry is made in the corresponding histogram bin. The value of C_{max} is determined by entering the minimum p_T track to be sought in Equation 4-2. Track candidates are identified as peaks in the histogram at the bin with the values of (ϕ_0, C) corresponding to the track parameters.

A two-dimensional histogram is constructed in terms of slope C and intercept at the calorimeter face, ϕ_{cal} (the calorimeter face is represented by a cylinder of radius 140 cm and half-length 360 cm). A value of $p_T^{\text{min}} = 5$ GeV is used as the minimum p_T for the track search. Only one point per detector plane is allowed to contribute to a given histogram bin. This restriction is applied in order to minimise the impact of points due to low p_T tracks from minimum bias events, noise or ghost points. Thus, after the histogram has been filled, the number of entries in a bin represents the number of detector planes with at least one point on the trajectory represented by the parameters (ϕ_{cal}, C) of the centre of the bin.

The histogramming process is illustrated diagrammatically in Figure 4-2. The process consists of the following steps:

- Construct a histogram with N_ϕ ϕ -bins in the range $\phi_{\text{RoI}}^{\text{min}}$ to $\phi_{\text{RoI}}^{\text{max}}$ and N_C slope-bins in the range $-C_{\text{max}}$ to C_{max} .
- For each space point in turn, calculate the intercept, ϕ_{cal} , for each value of slope in steps of size ΔC_{step} between $-C_{\text{max}}$ and C_{max} . Where the step size is $\Delta C_{\text{step}} = 2 C_{\text{max}} / (N_C + 1)$.
- If the intercept is in the range of the histogram, the values (ϕ_{cal}, C) define a histogram bin. The bin contents are incremented if there have been no previous entries from that detector plane. The information for each space point contains the address of the module that gave

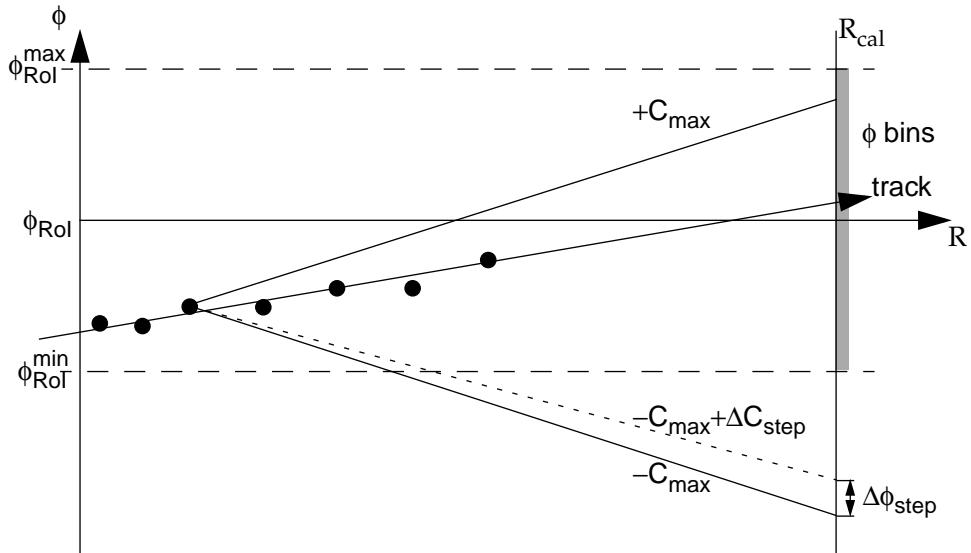


Figure 4-2 Diagram illustrating the histogramming process in the ϕ - R projection. The range of the histogram is from ϕ_{RoI}^{\min} to ϕ_{RoI}^{\max} . The minimum and maximum slopes considered, $-C_{\max}$ and C_{\max} , are shown. The step, $\Delta\phi_{\text{step}}$, resulting from a single step in slope, ΔC_{step} , is also indicated.

rise to it. A check is made to determine whether the detector plane containing this module has already contributed an entry to the histogram bin.

- If the number of entries in a bin reaches a pre-defined threshold, the identifier for this bin is added to the output list.

An example is shown in Figure 4-3 (top) of the contents of a histogram for an ROI in a single $p_T = 20$ GeV muon event. Each space-point populates many ($\leq N_C$) bins forming the diagonal lines of entries that can be seen in the figure¹. Provided an appropriate bin size has been chosen, there will be a bin (or bins) where all points on a track contribute an entry. In the figure, several bins with the maximum number of entries (7) can be seen in the region where the diagonal lines of entries meet, indicated by the arrow. Thus track candidates can be identified as peaks in the histogram. If the number of entries in a bin reaches a pre-defined threshold (5), all points lying within the road represented by the bin (i.e. points with ϕ_{cal} in the range of the bin for the given value of C) are considered as forming a potential track candidate and are passed on to the next stage. Shown in Figure 4-3 (bottom) is an example of a histogram for a ROI in a single $p_T = 20$ GeV electron event with pile-up at design luminosity. In addition to the entries due to the points on the electron track, points on tracks from minimum bias events also give diagonal lines of entries. However, since the majority of these tracks have a p_T below 5 GeV, these lines do not meet (they would converge at a slope value outside the range of the histogram). Thus the majority of minimum bias tracks add to the occupying of the histogram, but do not yield peaks. The peak due to the electron track can clearly be seen in the figure (indicated by the arrow) as several bins with the highest number of entries (7). The number of bins forming the peak of the histogram is larger than for the example shown in Figures 4-3 (top) due to the additional contributions from minimum bias.

1. For a negatively charged particle, the slope of the line of entries is negative for the barrel and $+z$ end-cap and positive for the $-z$ end-cap.

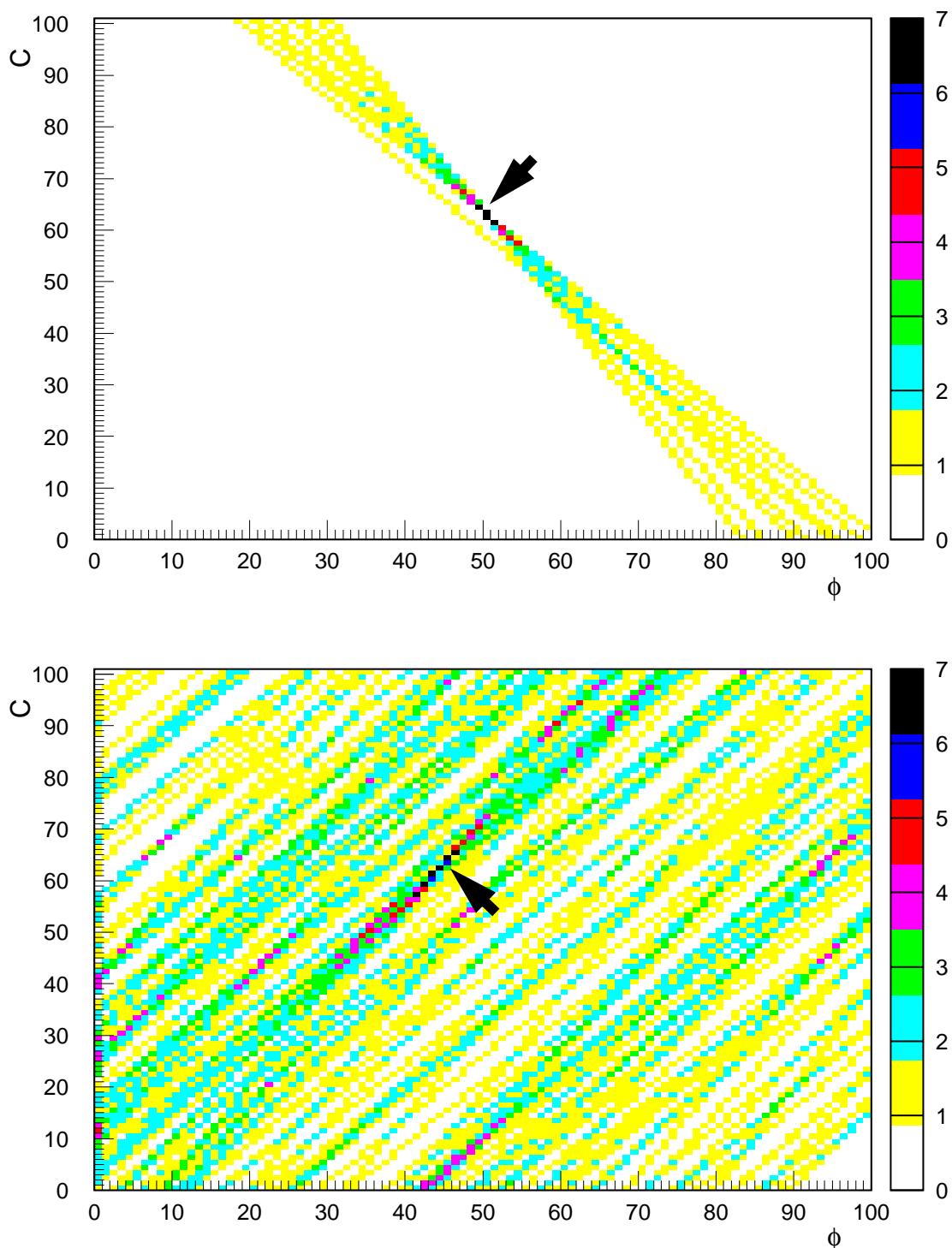


Figure 4-3 A representation of the contents of the histogram for an R0l in a single $p_T = 20$ GeV single muon event (top) and for a R0l in a single $p_T = 20$ GeV electron event with pile-up (bottom).

4.2 Track fitting Step

The input to the track fitting stage is a list of identifiers for histogram bins with at least N_{\min} entries ordered by decreasing number of entries and, for a given number of entries, by decreasing p_T . Contributions from minimum bias tracks increase the occupancy of the histogram and hence the number of bins with at least the threshold number of entries. Some of these bins will contain contributions from several different low p_T tracks, forming ‘fake’ track candidates. The bins with the highest number of entries are preferentially selected as these are more likely to form a ‘real’ track candidate with points entirely or predominantly due to a single track.

The fitting procedure starts with the bin with the most entries and highest p_T and proceeds, as required, to bins with lower p_T . If no good candidate has been found, the fitting procedure continues with bins with a lower numbers of entries. For each bin, the points are reselected, this time without the restriction to a single point per plane. Un-weighted least-squared linear fits are performed independently in the z -R, ϕ -R and/or ϕ -z projections to all combinations of one point per plane. Track quality cuts are applied and the best combination passing these cuts is kept. For bins with the same number of entries, the best overall candidate is chosen, depending on the choice of algorithm options, either as that with the highest p_T or on the basis of the fit-residuals. A subsequent cut, p_T^{cut} , is applied to the transverse momentum of the fitted track. Two values of this cut, 7 GeV and 10 GeV, have been used for the results presented in this note.

The choice of the minimum p_T of the histogram, p_T^{\min} , is an important consideration as it determines a trade-off between execution speed and track reconstruction quality. The memory requirement and execution time can be minimized by setting $p_T^{\min} = p_T^{\text{cut}}$. However this can lead to a higher trigger rate, compared to a lower p_T^{\min} , due to an increased probability of selecting a fake or poorly reconstructed track. Even if a ROI contains no track with $p_T > p_T^{\min}$, a bin containing contributions from two or more tracks, or a track plus noise hits, may pass the threshold on the number of entries. Such a candidate can give rise to a trigger due to a p_T miss-measured to be higher than p_T^{cut} . If a lower value of p_T^{\min} is used, there is an increased probability that the ROI will contain a track with $p_T > p_T^{\min}$, that it will be correctly reconstructed and subsequently rejected by the cut on the reconstructed p_T . A value p_T^{\min} of 5 GeV has been used for the results presented in this note.

5 Implementation

The two implementations, `t2scFex` and `SCTFEX`, are described below. The algorithms are almost identical in both cases. Where there are differences, these will be highlighted.

5.1 t2scFex Implementation

The t2scFex implementation of the histogramming algorithm consists of an initialisation step, performed once per ROI followed by the histogramming step itself. At initialization the following two arrays are allocated and initialized by the C function `calloc`:

Table 5-1 The user definable parameters of the t2scFex algorithm

parameter	default	Meaning	
pTmin	5	Minimum p_T track to be sought (GeV)	
nSlope	101	Number of slope bins (if even, next highest odd value is taken)	
nPhi	100	Number of ϕ -bins	
histThresh	4 ^a	Require more than histThresh entries	
qualCut	2	2	Use associated stereo pairs only
		1	Additionally use un-associated ϕ -layer clusters
fexLevel	1	Word used to select a set of algorithm options, see Table 5-2	
maxShared	-1	-1	Return the parameters of only the best track candidate
		0	Return parameters of all track candidates that have no shared points
		n	Return parameters of all track candidates that have $\leq n$ shared points

a. i.e. at least 5 entries are required ($N_{\min} = \text{histThresh} + 1 = 5$).

Table 5-2 Options for the t2scFex FEX. The meaning of the bit fields of the FEX-level word are shown. The default value is shown in bold (default FEX-level = 1).

Bit	Parameter	Value	Action
0	fitAllComb	0	Stop fitting point-combinations in a bin when a good candidate is found.
		1	Perform fits to all point-combinations in bin; choose best candidate.
1	maxEntriesOnly	0	Select all bins with $>\text{histThresh}$ entries.
		1	Only select bins with the highest number of points.
2	fitAllBins	0	Stop fitting once a bin with a good candidate has been found.
		1	Perform fits in all selected bins and choose best candidate.
3	fitAllNclus	0	Only fit bins with lower number of points when no good candidate.
		1	If maxonly = 0, perform fits in all bins with $>N_{\min}$ points
4	overlap	0	Use non-overlapping ϕ bins.
		1	Use overlapping ϕ bins.
6	useZcut	0	Do not use z information to select points
		1	Only use points that lie within the RoI in the z-R projection.

- A **histogram array** of dimension $N_C \times N_\phi$ words containing the histogram itself;
- A **plane-id array** of dimension $N_C \times N_\phi$ words used to ensure that only one point per detector plane contributes an entry to any single histogram bin. This array contains, for each bin, a bit pattern which represents the detector planes which have contributed an entry to the bin.

The histogramming step consists of a loop over space points, and for each space point, a loop over slope values. For each slope value, C_i , the intercept, $\phi_{\text{cal}}(C_i)$, is calculated as follows:

$$\phi_{\text{cal}}(C_i) = \phi_{\text{cal}}(-C_{\text{max}}) - i \Delta\phi_{\text{step}} \quad (i=0, N_C)$$

where $\phi_{\text{cal}}(-C_{\text{max}})$ is the intercept for slope $-C_{\text{max}}$ and $\Delta\phi_{\text{step}}$ is the change in ϕ_{cal} resulting from one step in slope:

$$\Delta\phi_{\text{step}} = 2 (R_{\text{cal}}^{\text{RoI}} - R_i) * C_{\text{max}} / (N_C - 1) \quad (\phi\text{-R projection}) \quad 5-1$$

$$\Delta\phi_{\text{step}} = 2 (z_{\text{cal}}^{\text{RoI}} - z_i) * C_{\text{max}} / (N_C - 1) \quad (\phi\text{-z projection})$$

For each step, the histogram array index is calculated, the plane-id array is checked and, if this is the first entry for the plane, the histogram bin is incremented and the plane-id array updated. If number of entries in the histogram equals the threshold ($N_{\text{min}} = 5$) the array index, intercept and slope for the bin are added to the output list. Once the histogramming process is completed, the output list is sorted in order of decreasing number of entries and then, for a given number of entries, by increasing absolute slope (i.e. decreasing p_T).

In order to ensure that there exists a bin containing all the points on a track, the value of $\Delta\phi_{\text{step}}$ must be less than the histogram ϕ -bin size, i.e. for the barrel region:

$$N_C > 2 N_\phi (R_{\text{cal}}^{\text{RoI}} - R_i) * C_{\text{max}} / (\phi_{\text{RoI}}^{\text{max}} - \phi_{\text{RoI}}^{\text{min}}) + 1 \quad 5-2$$

for all planes at radius R_i . The most stringent requirement occurs for the inner pixel layer ($R_i \approx 5$ cm). The requirement is less stringent in the forward region due to the $\tan(\theta)$ term in Equation 4-2. Substituting for the width of the RoI (200 mRad) and the value of C_{max} (0.6 mRad cm⁻¹ for a minimum p_T of 5 GeV). The inequality becomes:

$$N_C > 0.8 N_\phi \quad 5-3$$

Coarser steps in slope may be used without loss of track reconstruction efficiency if the histogram is constructed with overlapping ϕ -bins such that each value of slope yields entries in two adjacent histogram bins. However, the use of non-overlapping ϕ -bins with an appropriate choice of the slope step size was found to be more efficient in terms of overall performance. This will be discussed more fully below. For the results presented for the t2scFex implementation a histogram with dimensions of $N_C \times N_\phi = 101 \times 100$ non-overlapping bins has been used.

Various options have been implemented for the t2scFex algorithm. In addition to setting the histogram dimensions and p_T^{min} value for the histogram, a set of options may be selected by setting bits in the in a word used to define the **FEX-level**. The user definable parameters and the meaning of the bit fields of the FEX-level word are given along with their default values in Tables 5-1 and 5-2.

Table 5-3 Cuts used to define a good candidate. The quantities $\Delta_{R\phi}^2$ (mRad²), $\Delta_{z\phi}^2$ (mRad²) and Δ_{zR}^2 (cm²) are the mean-squared residuals in the $R\phi$, $z\phi$ and zR planes respectively. Δz_0 is the cut on the closest approach to the origin in z (this cut. is effectively switched off for the t2scFex implementation).

Cut	t2scFex		SCTFEX	
	barrel	end-cap	barrel	end-cap
$\Delta_{R\phi}^2 <$	1.5	–	1	1.5
$\Delta_{z\phi}^2 <$	–	1.5	10	4
$\Delta_{zR}^2 <$	15	15	50	15
$\Delta z_0 <$	1000	1000	25	25

Following the histogramming step, the algorithm proceeds as follows:

- **Select histogram bins:** The value of the parameter `maxEntriesOnly` determines which histogram bins are selected for the track fitting stage. By default all histogram bins with at least N_{\min} entries are selected.
- **Loop over selected bins:** For each selected bin:
 - **Select points in bin:** The points are reselected this time without the restriction to a single point per plane. The set of selected points is compared with the set selected for the previous bin. If the current set is identical to the previous set, it is skipped and fitting proceeds with the next selected bin.
 - **Fit to point-combinations in bin:** A **point-combination** consists of one point from each plane with a hit. For each point-combination:
 - **Perform un-weighted linear least squared fits** in two planes; the z - R plane and the R - ϕ plane (barrel) or the z - ϕ plane (end-caps).
 - **Apply track quality cuts:** A good combination is defined as one for which the mean-squared residuals satisfy the conditions listed in Table 5-3 (there is also an extremely loose cut on the intercept with the z -axis). Results are also shown for an algorithm variation where, after the z - R fit, points lying outside a road of width 0.9cm about the fitted direction are rejected. The point furthest outside the road is rejected first and the track re-fitted. The process is repeated until there are no points outside the road. The track is accepted if it has at least N_{\min} points and it passes the quality cuts listed in Table 5-3.
- **Select best track candidate in bin:** By default the `fitAllComb` option is selected and fits are performed to all point-combinations for the bin and the best candidate is chosen as that with the minimum value of the mean-squared residual of the R - ϕ fit (barrel) or the z - ϕ fit (end-caps).
- **Select best overall track candidate:** By default the `fitAllBins` option is not selected and fitting stops once a bin with a good candidate has been found, i.e. the candidate passing the quality cuts with the most points and, for that number of points, the highest p_T is chosen. If, instead, `fitAllBins` is set, the best candidate is chosen as that with the minimum value of the mean-squared residual of the R - ϕ fit (barrel) or the z - ϕ fit (end-caps). In both cases, however, if no good candidate is found in the bin, fitting continues with the next selected bin. If no good candidate is found in bins with the highest number of entries, fitting proceeds with the highest p_T bin with the next highest number of entries. To prevent a very large time being spent on a bin with a very large number of combinations, if the number of fits performed in a bin reaches a maximum of 10000, fitting in that bin stops. There is also a limit of 30000 imposed on the total number of fits that are performed per ROI. It is very rare for these limits to be reached. Such events could be accepted by the trigger with negligible impact on the trigger rate.

Optionally the value of `maxShared` may be changed from the default so that more than one track candidate is returned. In this case, when a good candidate is found it is compared with each of the track candidates in the output list in turn. If the new candidate shares less than `maxShared` points with any other candidate, it is considered to be a distinct track and is added to the list of output track candidates. However, if two candidates share at least `maxShared` points, the mean-squared residual of the R - ϕ fit (barrel) or the z - ϕ fit (end-caps) are compared and only the candidate with the better fit is retained.

5.2 SCTFEX implementation

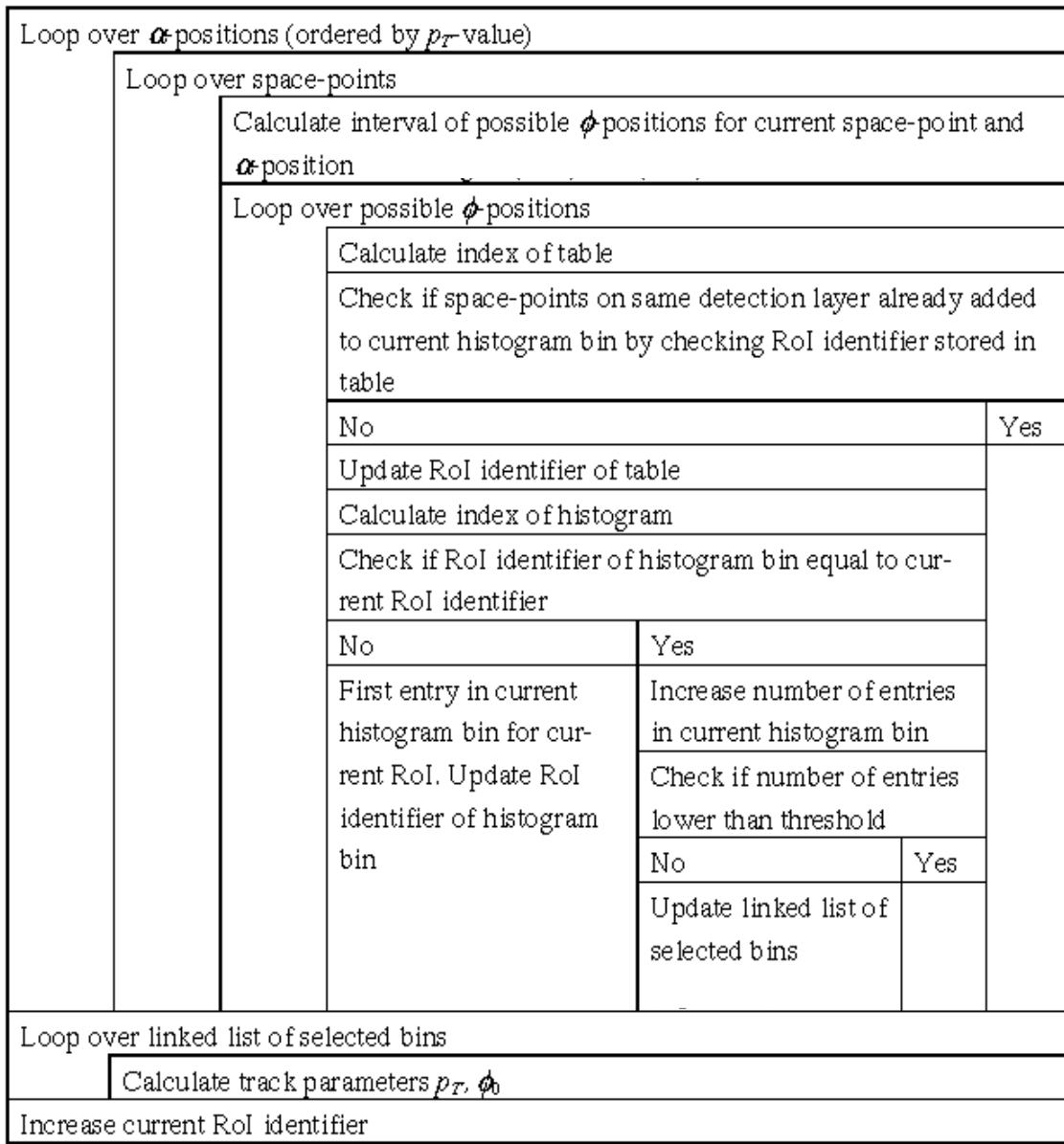


Figure 5-1 Program Structure Diagram for the histogramming algorithm as implemented in SCTFEX.

In the SCTFEX implementation, to save time the memory for the histogram and associated arrays are allocated and initialised once only, rather than once per event as in the t2scfex implementation. The following arrays are allocated:

- A **histogram array** of dimension $N_\phi \times N_C$ words containing the histogram bin contents.
- An **RoI identifier array** with the same dimension as the histogram array containing, for each histogram bin, a RoI identifier consisting of the event number and RoI number within that event. At run time, before incrementing a histogram bin, this array is checked. If the current RoI identifier does not match that stored in the RoI identifier array, the histogram bin contents are initialised to one and the current RoI identifier is stored in the RoI

identifier array. In this way, the necessity to initialises the whole histogram array once per ROI is avoided, at the cost of additional memory usage.

- A **detector plane array** of dimension $N_\phi \times N_C \times N_{\text{plane}}$ where N_{plane} is the number of planes in the barrel plus one end-cap (13). This array is used to ensure that only one point per plane can contribute to a histogram bin. To avoid the need to zero this array once per ROI, there are N_{plane} words per histogram bin which each store an ROI identifier. Before a histogram bin is incremented for a point from a given plane, the appropriate entry in the detector plane array is checked. If the stored ROI identifier does not match the current ROI id, the histogram bin contents are incremented and the current ROI id is stored in the detector plane array.

Before the histogramming step, the slope values, C_i , corresponding to the histogram bins are calculated and stored in a Look Up Table (LUT). For the $R\phi$ projection (i.e. for ROI in the barrel region), the slope values are the same for all ROI and so this LUT could be constructed at initialisation. However for the $z\phi$ projection (ROI in the end-cap) the bins depend on $|\eta_{\text{ROI}}|$. Alternate positive and negative slope values are stored, corresponding to the trajectories of positive and negative charged particles. The values are ordered by increasing $|C_i|$, i.e. in order of decreasing p_T of the track candidate. The histogram is filled via a loop over the slope values in this array, and for each slope value, a loop over all the space-points in the ROI. For each space-point, the value of the intercept, ϕ_{cal} , is calculated and the corresponding histogram bin identified. The ROI identifier and detector plane arrays are checked and, if appropriate, the histogram bin is incremented. When a histogram bin is incremented to the threshold value ($N_{\text{min}} = 5$), the bin is added to the end of a linked list of selected bins. The ordering of the values in the slope array ensures that the output list contains bin identifiers ordered by decreasing p_T . If the histogram bin to be incremented already has N_{min} or more entries, the bin identifier is moved up the link list such that the list is ordered by decreasing number of entries and for a given number of entries, by decreasing p_T . The Program Structure Diagram for the implementation of the histogramming algorithm is given in Figure 5-1.

The track fitting algorithm implemented in SCTFEX is almost the same at that implemented in t2scFex. In the case of SCTFEX, four algorithm options have been implemented. These are equivalent to the following FEX-levels:

- **FEX-level 1** (t2scFex default): All bins with at least N_{min} entries are selected. Fitting starts with the highest p_T bin with the highest number of entries. Fits are performed to all combinations within a bin, but fitting stops as soon as a bin with a good candidate is found (i.e. for a given number of entries, the good candidate with the highest p_T is chosen). If, however, no good candidate is found fitting proceeds with the next highest p_T bin and, if necessary, continues to bins with a lower numbers of entries.
- **FEX-level 3:** (`maxEntriesOnly`) As for FEX-level 1, but only the histogram bins with the highest number of entries are selected. Fitting is abandoned if no good candidate is found in these bins.
- **FEX-level 5:** (`fitAllBins`) As for FEX-level 1, but fits are performed to all bins with the same number of entries and the best candidate is chosen on the basis of the mean-squared fit residuals. Fitting proceeds to lower numbers of entries if no good candidate is found.
- **FEX-level 7:** (`maxEntriesOnly ∩ fitAllBins`) As for FEX-level 5, but only the histogram bins with the highest number of entries are selected.

The SCTFEX implementation has the following features which differ from t2scFex:

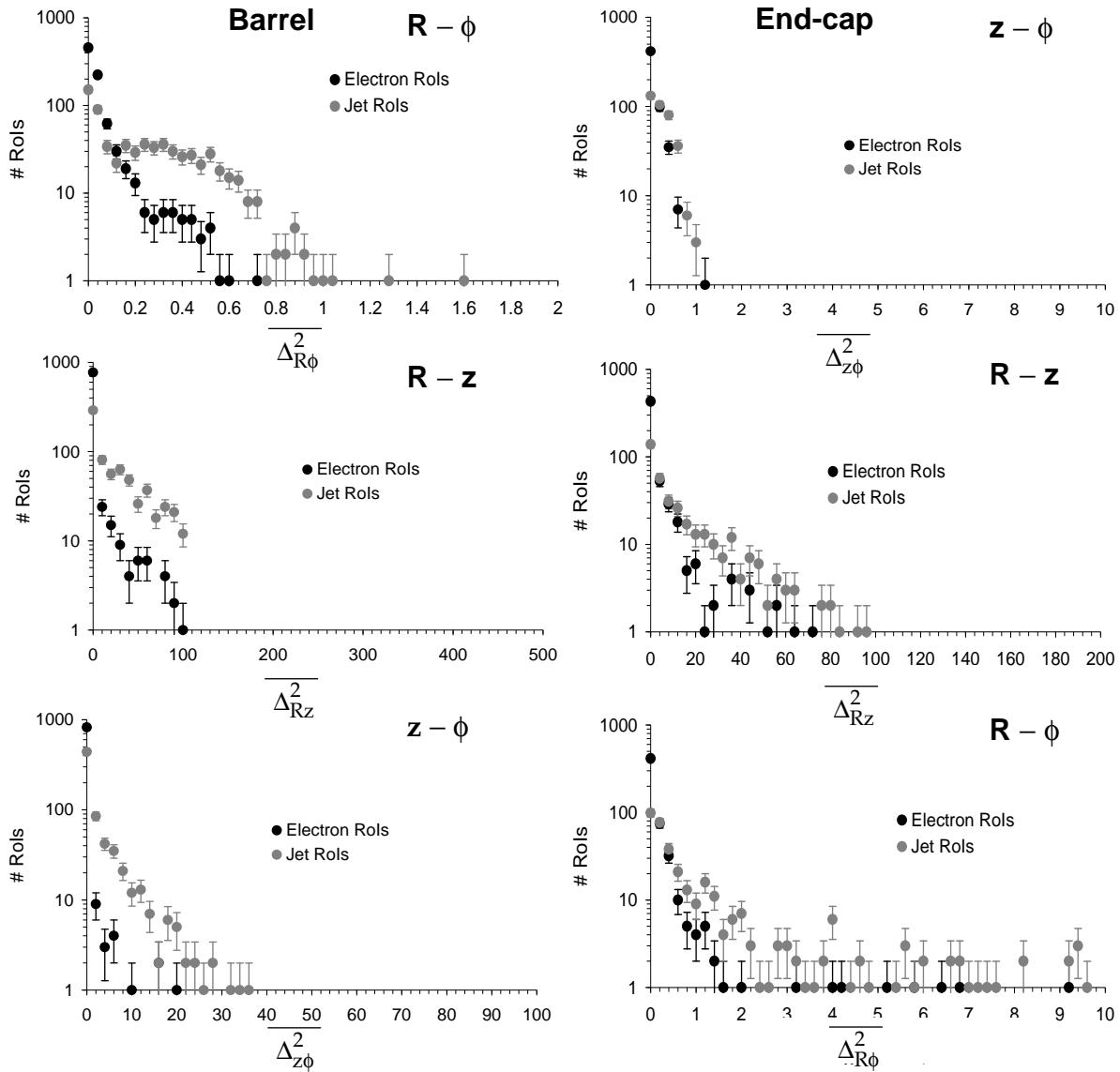


Figure 5-2 Distributions of the mean-squared residuals for the barrel region (left) and the end-cap region (right). The top plots are for the residuals in the projection used for the histogram (barrel: $R - \phi$, end-cap: $z - \phi$). The middle plots are for the fits in the z - R projection. The bottom plots show the distributions for the additional fits performed in SCTFEX in the ϕ - z plane (barrel) and the R - ϕ plane (end-cap). Note the different horizontal scales.

- before any fitting is performed for a bin, the number of combinations of one point per plane is calculated, and the bin is skipped if the number of combination exceeds 1000 (this is rare).
- for each point-combination, fits are performed in the three projections $R - \phi$, $z - \phi$ and $z - R$ and quality cuts are applied to the mean squared residuals calculated in each of these fits. The values of these cuts differ from those used in the `t2scfex` implementation and are given in Table 5-3.
- The best candidate is chosen as that which passes the quality cuts and has the minimum value for the quantity $Q = \frac{\Delta_{R\phi}^2}{\Delta_{R\phi}^2} \times \frac{\Delta_{z\phi}^2}{\Delta_{z\phi}^2} \times \frac{\Delta_{zR}^2}{\Delta_{zR}^2}$, where $\Delta_{R\phi}^2$, $\Delta_{z\phi}^2$, and Δ_{zR}^2 are the mean-squared residuals of the fits in the $R\phi$, $z\phi$ and zR planes respectively.

Including the residuals in the z - R plane in the calculation of the quality parameter provides discrimination against point-combinations which lie on a line in the 2-D projection used for the histogram, but in fact originate from several different tracks. This is particularly important in the end-caps.

The distributions of the mean-squared residuals (obtained with the SCTFEX implementation) are shown in Figure 5-2 for track candidates from ROI in electron events with pile-up and jet events with pile-up. There is only one entry per ROI in these plots which is for the candidate with the minimum value of Q . The quality cuts listed in Table 5-3 were chosen so as to give good efficiency for the best candidate in electron ROI. The top two plots show the distributions for the mean-squared residuals of the fit in the R - ϕ (z - ϕ) plane for the barrel (end-cap) respectively. The middle two plots show mean-squared residuals for the R - z fits applied to ROI in the barrel and end-cap regions. In both implementations, SCTFEX and t_2scFex , cuts are applied in the tails of the distributions to reject point-combinations unlikely to be due to a single track. In the bottom two plots of Figure 5-2 the mean-squared residual distributions are shown for the additional fits performed in the SCTFEX implementation alone.

The intercept of the track in the z - R projection also provides some discrimination against bad candidates. The distribution is shown in Figure 5-3. A cut of $|z_0| < 25$ cm is applied in the SCTFEX implementation.

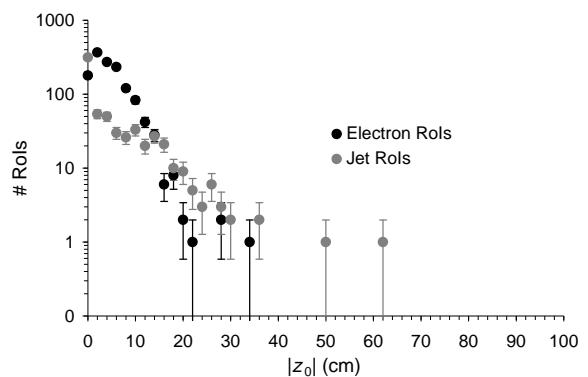


Figure 5-3 The distribution for the absolute value of the z -impact parameter with respect to the origin $|z_0|$.

The intercept of the track in the z - R projection also provides some discrimination against bad candidates. The distribution is shown in Figure 5-3. A cut of $|z_0| < 25$ cm is applied in the SCTFEX implementation.

6 Algorithm performance

Efficiencies have been measured for ROI in single particle events with and without pile-up. Timing measurements were made for ROI in jet events with pile-up on a Dell PC with an 400 MHz Intel Pentium II processor, 512 kbyte cache and 64 Mbytes of internal memory. The system bus speed was 100 MHz. The operating system was Windows NT 4.00. More details of how the timing measurements were made can be found in [1]).

6.1 Histogramming step:

As a result of the histogramming step, track candidates are identified as peaks in the histogram. The bins with at least 5 entries are se-

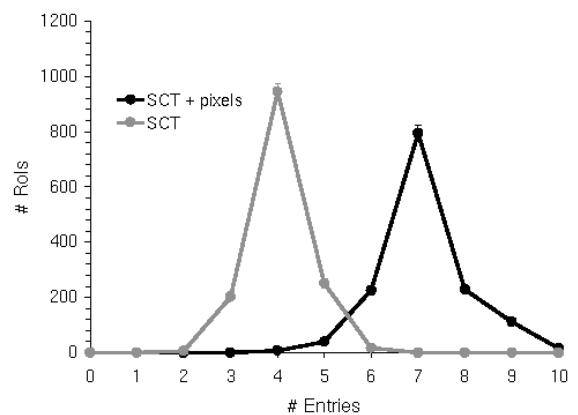


Figure 6-1 The distribution of the maximum number of entries per ROI for a histogram dimension of $N_C \times N_\phi = 101 \times 100$ and non-overlapping ϕ -bins for single $p_T = 30$ GeV electron events with pile-up (for the histogram definition, see Section 5.1). Distributions are shown for the SCT alone and for the SCT and pixel detectors combined.

lected as track candidates (an additional selection of only the bins with the peak number of entries is imposed if the option `maxEntriesOnly` is chosen). In the barrel region of the precision tracker, the maximum possible number of entries in any bin is seven (three pixel layers and four SCT layers). However, in the forward region, due to the disc geometry, there can be as many entries as there are detector planes contained within the ROI. Up to 10 entries are possible at large $|\eta|$. A single track from the interaction point cannot cross all 10 detector planes, but points from two or more tracks at different η may contribute to the same bin if they have similar trajectories in the two-dimensional projection used for the track search. A further level of point selection is possible using 3-D information at the track-fitting stage.

The distribution of the maximum number of entries in a histogram bin is shown in Figure 6-1 for ROI in single $p_T = 30$ GeV electron events with pile-up. Distributions are shown for the SCT alone and for the SCT and pixel detectors combined. The distribution for the SCT plus pixels peaks at 7 entries.

The efficiencies for the histogramming step are given in Table 6-1 for ROI in electron events with pile-up. For the default histogram dimension (101×100 non-overlapping bins), the efficiency is 98.4%. The efficiency is defined as the fraction of ROI for which at least one histogram bin has at least five entries. When the number of slope steps is reduced from 101 to 25, i.e. so that the inequality of Equation 5-3 is no longer satisfied, the efficiency for finding at least one bin above threshold is reduced by 2%. This is due to entries from the histogram peak being divided between two or more adjacent ϕ -bins. The loss of efficiency can be partially recovered by the use of overlapping ϕ -bins, but at the cost of more bins being selected as track candidates, increasing the work to be done at the track fitting stage, as will be discussed later. When the inequality of Equation 5-3 is satisfied, as is the case with the default histogram dimension (101×100 bins), there is no gain in efficiency by using overlapping ϕ -bins.

Table 6-1 The efficiency for the histogramming stage of the SCTFEX algorithm ($N_{\min} = 5$) for ROI in single $p_T = 30$ GeV electron events with pile-up. Efficiencies are given for two different histogram dimensions ($N_C \times N_\phi$) and for histograms constructed with non-overlapping or overlapping ϕ -bins.

Histogram dimensions	E (%)
101×100 , non-overlapping (default)	98.4
25×100 , non-overlapping	96.5
101×199 , overlapping	98.5
25×199 , overlapping	97.8

Table 6-2 The mean number of bins selected per ROI for two options of the SCTFEX algorithm i) when bins with at least 5 entries are selected and ii) when only the bins with the highest number of entries are selected. Results are shown for ROI in single $p_T = 30$ GeV electron events with pile-up and for ROI in jet events with pile-up.

	electrons ($p_T = 30$ GeV) + pile-up		jets + pile-up	
	≥ 5 entries	Highest no. entries	≥ 5 entries	Highest no. entries
non-overlapping ϕ -bins	23.8	3.6	26.4	4.7
overlapping ϕ -bins	47.6	7.0	52.5	8.5

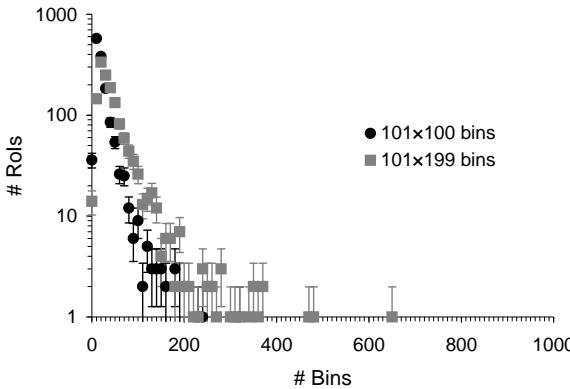


Figure 6-2 The number of histogram bins that are selected per RoI when at least 5 entries are required for single $p_T = 30$ GeV electron events with pile-up. Distributions are shown for histograms constructed with non-overlapping (101×100) and overlapping (101×199) ϕ -bins. The bin-width is the same in both cases. The SCTFEX implementation was used.

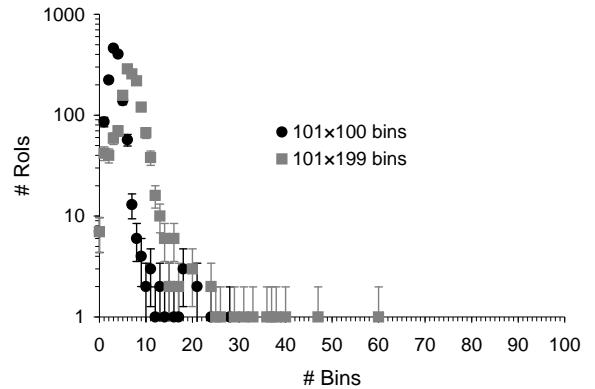


Figure 6-3 The number of histogram bins selected per RoI when only the bins with the highest number of entries are selected. Distributions are shown for electron events with pile-up for histograms constructed with non-overlapping (101×100) and overlapping (101×199) ϕ -bins. Note the different horizontal scale cf. Figure 6-2. The SCTFEX implementation was used.

The distributions of the number of bins per RoI with at least 5 entries are shown in Figure 6-2 for RoI in single $p_T = 30$ GeV electron events with pile-up both for non-overlapping (101×100) and overlapping (101×199) ϕ -bins. The means of the distributions are given in Table 6-2. For non-overlapping bins (the default algorithm option), the mean numbers of bins selected are 23.8 (26.4) for RoI in electron (jet) events with pile-up respectively. If overlapping bins are used these numbers are doubled. If an additional selection is made of only the bins with the highest number of entries (corresponding to the `maxEntriesOnly` option), the number of bins selected per RoI is significantly reduced. The corresponding distributions are shown in Figure 6-3. For non-overlapping bins, the average number of bins selected per RoI is reduced to 3.6 (4.7) in electron (jet) events with pile-up respectively. However this is accompanied by a drop in efficiency, as will be discussed in Section 6.2.

Timing results are given in Table 6-3 for the SCTFEX implementation of the histogramming step. For the default histogram dimensions (101×100 non-overlapping bins) the algorithm has a mean execution time of 8.64 ms on a 400 MHz Pentium II for RoI in jet events with pile-up. Since the algorithm contains a loop over slopes, the execution time is proportional to the number of slope steps. The execution time also scales linearly with the number of space points, see Figure 6-4. A parameterisation of the execution time as a function of the number of space points in the RoI is given in Table 6-4.

Table 6-3 The mean execution time for the SCTFEX histogramming algorithm for a 400 MHz Pentium II.

Histogram Dimensions	Mean Time (ms) Jet+pile-up RoI
101×100	8.64
25×100	1.90

Table 6-4 The parameters of a fit to the execution time of the SCTFEX histogramming algorithm as a function of the number of points in the RoI for jet events with pile-up. The execution time was measured on a 400 MHz Pentium II PC.

Bins: $N_C \times N_\phi$	gradient (ms)
101×100	40.3
25×100	8.58

The timing results are based on the assumption that the input to the candidate search algorithm consists of points from all modules lying wholly or partially within the RoI. The number of space points, and hence the execution time can be reduced by about 45% for a 3% loss of efficiency by selecting, prior to the histogramming stage, the subset of points that lie within the RoI (see [1]).

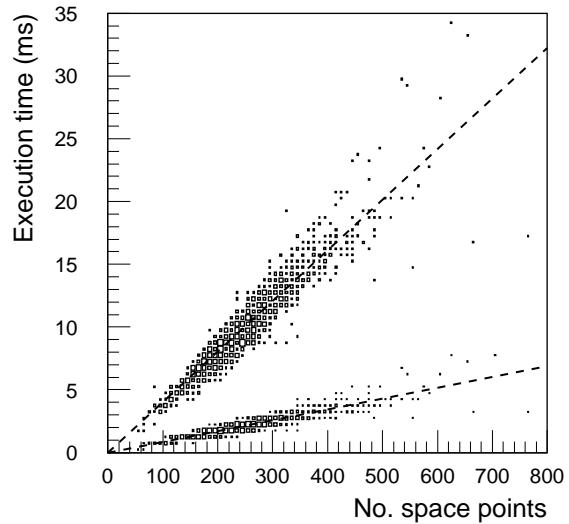


Figure 6-4 The execution time of the SCTFEX histogramming algorithm on a 400 MHz Pentium II for RoI in jet events with pile-up. Results are shown for two histogram sizes; 101×100 (upper distribution) and 25×100 (lower distribution). For each point, the box size is proportional to the number of entries. Fitted lines are shown for each distributions. The fit parameters are given in Table 6-4.

6.2 Fitting Step

For each of the selected histogram bins, the points are re-selected this time without the restriction to one point per detector plane. The fitting step is used to select a combination of one point per detector plane passing the track quality requirements listed in Table 5-3. Fitting starts with the bin with the highest number of entries and, for that number of entries, the highest p_T and proceeds to other bins as necessary and as determined by the algorithm options described in Table 5-2.

The distributions for the total number of point-combinations per RoI for which fits were performed are shown in Figure 6-5 for four algorithm options. The mean number of fits per RoI are given in Table 6-5 for RoI both in single $p_T = 30$ GeV electron events with pile-up and jet events with pile-up. The total execution time of the fitting step is directly proportional to the number of fits performed. The results for jet events are the most relevant to algorithm execution time, since the background from jet events dominates the LVL1 trigger rate. The higher occupancy in jets results in a larger number of bins above threshold and a larger number of combinations per bin. Also given in Table 6-5 are the efficiencies for reconstructing a track in single $p_T = 30$ GeV electron events with pile-up. Efficiency is defined as the fraction of RoI for which a track is reconstructed with at least five points, satisfying the track quality requirements listed in Table 5-3 for the SCTFEX implementation, and with a reconstructed $p_T > 7$ GeV. Optimization of the algorithm consists of maximizing the efficiency whilst maintaining a reasonably low mean number of fits and hence an acceptable execution time.

For the default algorithm options, fitting stops as soon as a bin is found with a combination of points which satisfy the requirements listed in Table 5-3. Fitting only proceeds to other bins with lower p_T and lower numbers of entries if no satisfactory candidate is found. In this case the mean number of fits performed per RoI in jet events with pile-up is 57. The corresponding efficiency for a track to be found in RoI in electron events with pile-up is 97.3%. If the `maxEntriesOnly` option is chosen, only bins with the maximum number of entries are selected. In this case the number of fits is reduced by a factor 2.4 with a corresponding drop in efficiency of 3.4% for the case of non-overlapping bins.

If the `fitAllBins` option is chosen, rather than stopping when a satisfactory candidate is found, fits are performed in all bins with the same number of entries and the candidate giving the best fit is retained. In this case the mean number of fits for jet events with pile-up is doubled with respect to the default option and there is no improvement in efficiency. However there is an improvement in track quality. The fraction of RoI where all points on the selected track are due to the high p_T electron is increased from 55% when the first good candidate is selected (default option) to 85% when the `fitAllBins` option is chosen. In addition the fraction of tracks which have less than 50% of points from the high- p_T electron is reduced from 73% (default option) to 47% (`fitAllBins` option).

Table 6-5 The mean number of point-combinations per RoI for which fits are performed for the four FEX Levels tested (SCTFEX implementation). Results are shown for single $p_T = 30$ GeV electron events with pile-up and for jet events with pile-up. Also shown are the corresponding efficiencies for finding a track with reconstructed $p_T > 7$ GeV passing the quality cuts given in Table 5-3 for electron events with pile-up.

FEX level	FEX options set	non-overlapping ϕ bins (101 \times 100) ^a		overlapping ϕ bins (101 \times 199)			
		e + pile-up ($p_T = 30$ GeV)		jet + pile-up	e + pile-up ($p_T = 30$ GeV)		
		No. Fits	E (%)	No. Fits	No. Fits	E (%)	
1	default ^b	12	97.3	57	15	97.5	71
3	<code>maxEntriesOnly</code>	9	93.9	24	9	94.1	27
5	<code>fitAllBins</code>	34	97.2	113	61	97.3	164
7	<code>MaxEntriesOnly</code> \cap <code>fitAllBins</code>	32	93.8	92	57	93.9	139

a. Default histogram dimensions.

b. For the default FEX options, see Table 5-2.

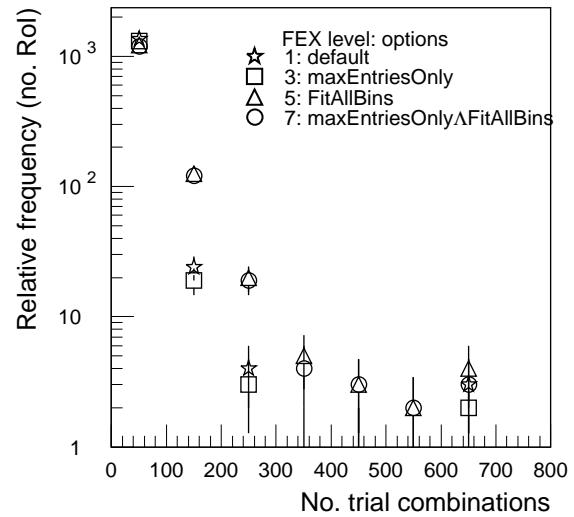


Figure 6-5 The total number of point-combinations fitted for RoI in electron events with pile-up (SCTFEX implementation). Results are shown for four different sets of FEX options.

Table 6-6 Calculation times for RoI in jet events with pile-up and Efficiency for RoI in single $p_T = 30$ GeV electron events + pile-up measured on a 400 MHz Pentium II.

	FEX level	FEX Options	Fit time (ms)	Total time (ms)	Overall Efficiency (%)
101 \times 100 (default)	1	default	1.2	11.6	97.3
	3	maxEntriesOnly	0.6	10.6	93.9
25 \times 100	1	default	0.4	2.6	93.2
	3	maxEntriesOnly	0.2	2.3	90.5

The results of timing measurements are shown in Table 6-6 for two algorithm options, default and `maxEntriesOnly`, and for two histogram sizes with 101 or 25 slope steps. Measurements are shown for the fitting stage and for the total algorithm time, including the histogramming times given in Table 6-3. The corresponding efficiencies are also shown for RoI in electron events with pile-up. When only the histogram bins with the highest number of entries are considered at the fitting stage (`maxEntriesOnly` option), the fit takes an average of 0.6 ms on a 400 MHz Pentium II for the default histogram size (101 \times 100 non-overlapping bins). The total time, including the histogramming time has a mean of 10.6 ms. The distribution of the total execution time is shown in Figure 6-6 for the `maxEntriesOnly` option. The total execution time is less than 20 ms for 98% of RoI in jet events with pile-up.

Since the time required for the fits is only \sim 10% of the overall execution time, the `maxEntriesOnly` option does not offer a significant saving in time. With the default algorithm option, all bins with at least 5 entries are considered at the fitting stage. In this case the mean total execution time is increased by 1 ms to 11.6 ms with a corresponding 3.4% increase in efficiency.

As the execution time for the histogramming step dominates, the time decreases by a factor of four when the number of slope steps is reduced to 25. However there is a 4% loss of efficiency. This is due to a loss of 2% of candidates at the histogramming stage and an additional 2% loss in the fitting step. The latter is a due to RoI where no track is found meeting the quality requirements. This is due to the fact that the points from the track may be split between adjacent ϕ -bins due to the large slope step size. In addition the larger slope-bin size leads to an increased probability that a point from another track is included in the fit.

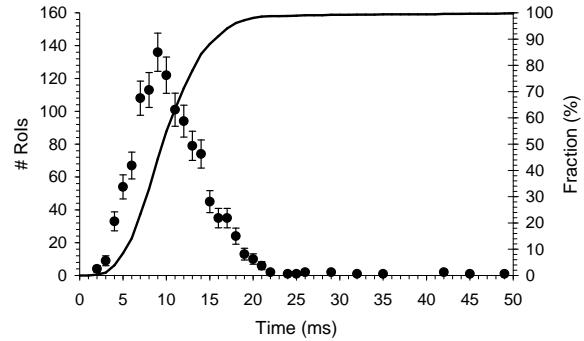


Figure 6-6 The distribution (points) of the total execution time of the SCTFEX algorithm (`maxEntriesOnly` option) for the default histogram dimension (101 \times 100 non-overlapping bins) measured on a 400 MHz Pentium II. The fraction (r.h. scale) of RoI with execution times below a given value is also shown (curve).

6.3 Overall Performance

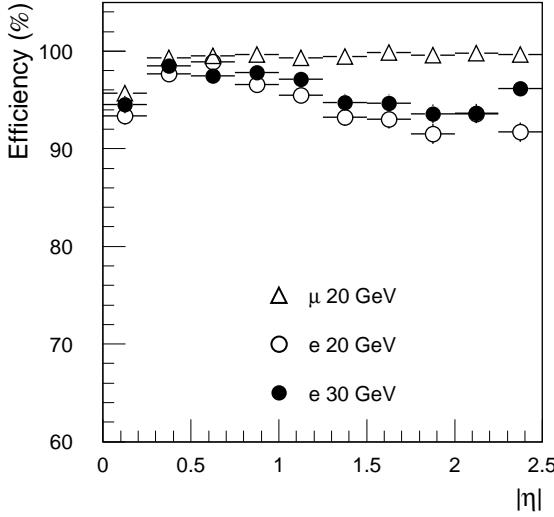


Figure 6-7 Efficiency of the precision tracker FEX (t2scFex implementation) as a function of $|\eta|$ for single particles without the addition of pile-up when a reconstructed track with $p_T > 10$ GeV is required.

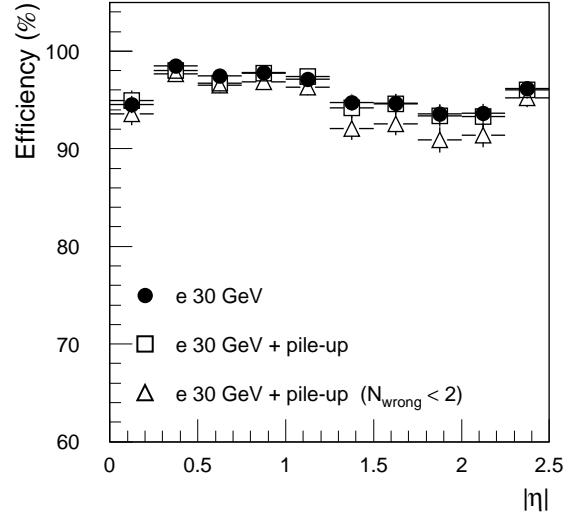


Figure 6-8 Efficiency as a function of $|\eta|$ for single particles of $p_T = 30$ GeV with and without pile-up at design luminosity (t2scFex implementation). Also shown is the efficiency for reconstructing a track with < 2 ‘wrong’ points (points from the track of another particle).

The efficiency of the precision tracker FEX is shown as a function of $|\eta|$ in Figure 6-7 (t2scFex implementation) for single particles without the addition of pile-up. The efficiency is defined as the fraction of ROI for which a track is reconstructed with at least five points, passing the track quality cuts listed in Table 5-3 for the t2scFex implementation, and with a reconstructed p_T greater than 10 GeV. The efficiency for $p_T = 20$ GeV muons is close to 100% except for the region $|\eta| \approx 0$ where there is a 4% loss of efficiency (this is a feature of the algorithm implementation). The overall efficiency for $p_T = 20$ GeV muons is 99%. The efficiency for electrons is 3-4% lower than that for muons; 96% for $p_T = 30$ GeV electrons and 95% for $p_T = 20$ GeV electrons. The lower efficiency for electrons is due to bremsstrahlung energy loss in the material of the beam pipe and in the precision tracker itself. The effect of this loss is more significant for $p_T = 20$ GeV electrons than for $p_T = 30$ GeV electrons as the initial p_T is nearer the threshold.

There is a negligible change in efficiency with the addition of pile-up, see Figure 6-8. However, there is a decrease in track quality due to the inclusion in some track candidates of one or more points from another particle track. The occupancy of the SCT is 0.3% - 0.6% for events at design luminosity (corresponding to between three and five points per module) which leads to the possibility of points being incorrectly assigned to tracks.

Histograms of the number of space-points on a track are shown in Figure 6-9 for single electrons of $p_T = 30$ GeV with pile-up (top plot, solid line) and for fake ROI containing only points from the pile-up of minimum bias tracks (lower plot, open histogram). For the electron events the distribution of the number of points on a track ranges from 5 (the threshold number) to 10 with a peak at 7. For the minimum bias events, however, the p_T of the particles is in general lower than the p_T^{\min} of the histogram and so the distribution of is peaked at 5 points.

Superimposed on the plots in Figure 6-9 are the distributions of the number of the points on the track that are due to a single particle, shown as the dashed line for the electron events (top plot) and the hatched histogram for the fake ROI in minimum bias events (lower plot). For the electron events, there are relatively few tracks with less than 5 points from the same particle; 6% have one and 3.4% have two or more incorrectly assigned points. The efficiency for tracks with no more than one wrong point is shown superimposed on Figure 6-8. It can be seen that the effect of the confusion from other tracks is greatest in the end-caps.

As shown by the hatched histogram in the lower plot of Figure 6-9, tracks in minimum bias events frequently have contributions from more than one particle. In most cases, for the minimum bias events, no more than one or two points are contributed by any one particle.

In addition to maximising the efficiency for isolated high p_T tracks, it is important to be able to discriminate against “fake” track candidates caused by noise points and points from one or more low- p_T minimum bias particles. This is important for reducing the rate of triggers from

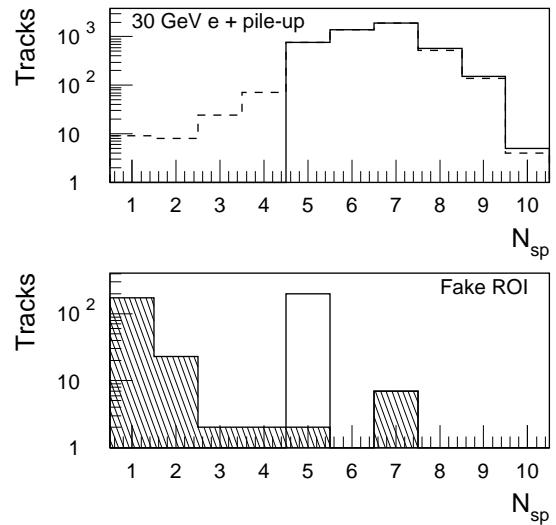


Figure 6-9 The number of space-points on a track (top: solid line, bottom: open histogram) and the number of space-points on the track due to a single particle (top: dashed line, bottom: hatched histogram) for single electrons of $p_T = 30$ GeV with pile-up (top) and for fake RoIs containing only hits from minimum-bias pile-up (bottom).

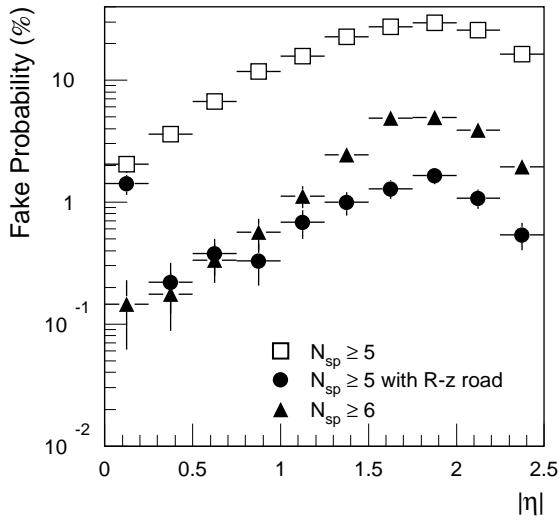


Figure 6-10 The probability of reconstructing a “fake” track in a region $\Delta\phi \times \Delta\eta = 0.2 \times 0.2$ of an event containing only piled-up minimum bias events shown for a threshold of ≥ 5 space-points, with and without the restriction to a ± 0.9 cm road in the z - R plane. The “fake” probability is also shown for a threshold of ≥ 6 space-points.

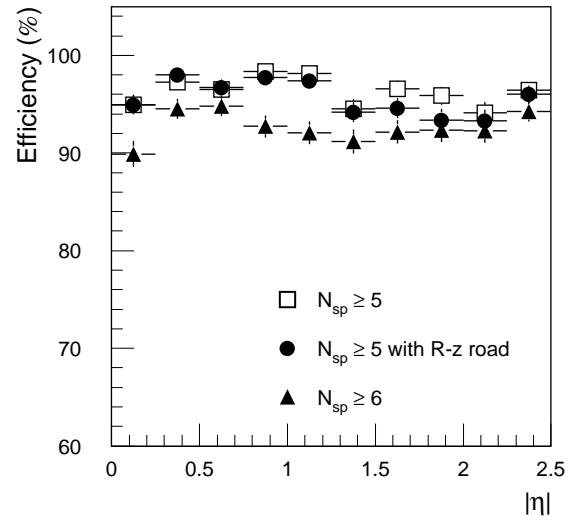


Figure 6-11 The efficiency for reconstructing a track in events with a single $p_T = 30$ GeV electron plus pile-up at design luminosity for the same sets of cuts as applied in Figure 6-10.

jet events. A measure of the discrimination power of the precision tracker FEX against fake tracks is the rate at which tracks are found in fake RoI created randomly in events containing only the pile-up of minimum bias events. The fraction of these RoIs for which a reconstructed track candidate is found is shown in Figure 6-10 for three different sets of cuts. The lowest average fake probability, 0.8%, is obtained when at least five points are required on a track and the points are required to lie within a ± 0.9 cm road in the z-R projection. The probability of finding a fake track increases with $|\eta|$ up to a maximum of 1.5% at $|\eta| \approx 1.8$. This is due to the larger number of detector planes that can contribute to a track, 10 in the end-caps compared to 7 in the barrel, which increases the probability that a fake track can arise due to contributions from two or more particles with similar trajectories. This effect is greatly accentuated when the restriction to a road in the z-R projection is removed, see Figure 6-10. The imposition of the z-R road causes only a small (1%) decrease in efficiency, as shown in Figure 6-11. The fake probability can also be reduced by increasing the number of points required on a track to $N_{\min} = 6$, but this leads to a 4% loss of efficiency.

A disadvantage of the imposition of a z-R road is that it introduces an iterative step in the fitting process and hence an increase in execution time. However this increase is relatively small compared with the time for the histogramming step, for the histogram dimensions used in this note. Another disadvantage is that there is loss of track quality due to the erroneous removal of correctly assigned points from some tracks. The size of this effect will depend on detector alignment. This is an area of future study. Another very powerful method to reject fake candidates is to require a match between the track parameters as measured by the precision tracker and the cluster parameters as measured by the electromagnetic calorimeter. This is the method used for the electron trigger performance measurements reported in [6][8][13].

7 Conclusions

A track finding algorithm for the SCT and pixel detectors has been presented. It is based on an initial search for track candidates using a histogramming method followed by a fitting stage to determine the track parameters and to choose the best candidate in the RoI. Various algorithm options have been investigated. Execution times have been measured on a 400 MHz Pentium II PC using jet events with pile-up corresponding to high luminosity running. The histogram dimension is the most significant factor influencing execution time. For the SCTFEX implementation of the algorithm with the default set of algorithm options, an average execution time of 11.6 ms has been measured with 98% of RoI having an execution time less than 20 ms. The corresponding efficiency for finding a track with reconstructed $p_T > 7$ GeV is 97% for single $p_T = 30$ GeV electron events with pile-up. The execution time can be reduced by using fewer slope steps at the histogramming stage, but with some loss of efficiency. A factor of four reduction in slope steps gives a factor of four reduction in execution time, with a 4% loss of efficiency.

The quality of the reconstructed tracks has been studied by measuring the number of incorrectly assigned points, due to noise or points from a different particle, on tracks in single $p_T = 30$ GeV electron events with pile-up. Of the reconstructed tracks, 6% have one and 3.4% two or more incorrectly assigned points.

Methods of reducing the probability of reconstructing a fake track (composed of points from several different particles) have been investigated. Measurements were made of the probability of reconstructing a track in fake RoI created randomly in events containing only the pile-up of

minimum bias events at high luminosity. A powerful rejection of fake tracks can be achieved by requiring that points lie within a ± 0.9 cm road in the z-R projection. In this case tracks are reconstructed with $p_T > 10$ GeV in 0.8% of fake RoI. Another very powerful method to reject fake candidates is to require a match between the track parameters as measured by the precision tracker and the cluster parameters as measured by the electromagnetic calorimeter. This method is used with the algorithm presented in this note for the electron trigger described in [6].

8 References

1. R.Dankers, J.Baines, *A Data Preparation Algorithm for the Precision Tracker LVL2 FEX* ATLAS Internal Note, ATL-DAQ-99-001 (January 1999).
2. *Atlas TRIGger Simulation Package*
<http://www.cern.ch/Atlas/GROUPS/DAQTRIG/ATRIG/>
3. R.Dankers, *The Physics Performance and Level 2 Trigger for the Inner Detector of ATLAS*, University of Amsterdam Thesis (October 1998).
4. *C environment for Trigger Studies*,
<http://hepunx.rl.ac.uk/atlasuk/simulation/level2/doc/ctrigCVS/>
5. *ATLAS Second Level Trigger Pilot Project Reference Software*,
<http://www.cern.ch/Atlas/project/LVL2testbed/www/>
6. J.Baines et al. *Electron Identification at the Second Level Trigger*, ATLAS Internal Note, ATL-COM-DAQ-99-007 (March 1999).
7. A. Dell'Acqua et al. *1997 ATLAS Jet Production*, ATLAS Internal Note, PHYS-NO-102 (july 1997)
8. *ATLAS Inner Detector Technical Design Report*, Volumes I and II, CERN/LHCC/97-16 and CERN/LHCC/97-17, ATLAS TDR 4, April 1997.
9. *ATLAS Calorimeter Performance Technical Design Report*, CERN/LHCC 96-40, January 1997
10. *ATLAS Pixel Detector Technical Design Report*, CERN/LHCC/98-13, May 1998.
11. D. Barberis et al., *A Comparative Study of Reduced Layouts of the ATLAS Inner Detector*, ATLAS Internal Note, INDET-NO-188 (November 1997).
12. A.Parker et. al. *The Material Budget of the ATLAS Inner Detector*, ATL-INDET-98-207, CERN/LHCC 98-17 (May 1998).
13. *ATLAS Trigger Performance Status Report*, CERN/LHCC 98-15 (June 1998).