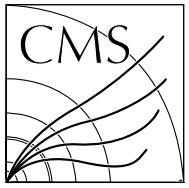
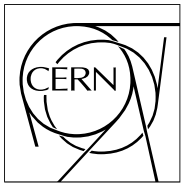
Available on CMS information server **CMS NOTE 2006/026**



The Compact Muon Solenoid Experiment

CMS Note

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



January 31, 2006

Track reconstruction, primary vertex finding and seed generation with the Pixel Detector

S. Cucciarelli[a)],

CERN, Geneva, Switzerland

M. Konecki[a)],

Institute of Experimental Physics, Warsaw University, Warsaw, Poland

D. Kotliński,

Paul Scherrer Institute, Villigen, Switzerland

T. Todorov,

Institut de Recherches Subatomiques, IN2P3-CNRS-ULP, Strasbourg, France / CERN, Geneva, Switzerland

Abstract

The Pixel Detector is the innermost detector of the tracking system of the Compact Muon Solenoid (CMS) experiment. It provides the most precise measurements which contribute to the full track reconstruction. It also allows standalone track reconstruction, which is particularly useful for online event selection (called High-Level Triggering). The reconstruction algorithms using the Pixel Detector are presented, including pixel track reconstruction, primary vertex finding and seed generation.

[a)] formerly at University of Basle, Switzerland

1 Introduction

The Pixel detector provides high resolution, three-dimensional space points allowing for precise pattern recognition. With three pixel hits per charged particle, using only the pixel data tracks can be approximately reconstructed and primary vertices (PVs) can be found. Such "pixel" reconstruction is useful for track seeding, primary vertex finding and in a variety of High Level Trigger (HLT) algorithms. The Pixel detector is the most suitable for these tasks due to its good spatial resolution. The typical position resolution is around 10–20 μm in the transverse and longitudinal coordinates.

The Pixel detector layout considered here consists of three barrel layers with two endcap disks on each side. The three barrel layers are located at mean radii of 4.4, 7.3 and 10.2 cm and are 53 cm long. The two disks are placed at 34.5 and 46.5 cm from the interaction point. To achieve a similarly good resolution of the vertex position in the transverse and the longitudinal planes, a design with a rectangular pixel shape of dimensions 150×100 μm^2 and a thickness of 290 μm is used. To enhance the spatial resolution by analog signal interpolation, the effect of charge sharing induced by the large Lorentz drift in the 4T magnetic field is used. Hence the detectors are deliberately not tilted in the barrel layers but are tilted in the endcap disks, resulting in a turbine like geometry. The whole pixel system consists of about 66 Mpixels in 1400 detector modules. A more detailed description of the Pixel layout, its simulation and hit reconstruction can be found in Ref. [1].

The track finding based on pixel hits is carried out in two steps: defining a pair of hits compatible with a track, and adding a third hit which matches the pair. Two hits (pairs) are enough to define a track seed but the ghost rate might be high. Track candidates based on three pixel hits (triplets) allow reconstruction of the primary vertex and can be used in simple algorithms for online event selection, but are not fully efficient. This note describes the various algorithms used to find pixel pairs, triplets and primary vertices. The performance of these algorithms are presented and their application in the CMS reconstruction software ORCA [2] are outlined.

In the simulation, the spread of the p-p interactions in the x, y (σ =0.0015 cm) and z (σ =5.3 cm) directions are included; however the effect of detector misalignment has not been yet studied.

When discussing the performance of our algorithms two types of efficiencies are used. The "algorithmic" efficiency is defined as the ratio of the events where a pixel pair/triplet/track is found, to events where the relevant pixel hits exist. The "absolute" efficiency is defined as the ratio of events where the pairs/triplets/tracks are found to all events. Thus the absolute efficiency includes detector inefficiencies.

2 Components of the Reconstruction Software

An effort has been made to design the software in a modular way in order to make it robust and allow for flexible usage. A major asset of the current implementation is the possibility to perform reconstruction in a restricted region (*regional reconstruction*) which may be useful for the online software.

The key concept in the regional-based reconstruction is the `TrackingRegion`. It is defined as a geometric region of interest with some kinematic constraints. The `TrackingRegion` specifies the direction around which the region is defined, the (signed) inverse transverse momentum range and the allowed range of track impact points. The allowed range of impact points is defined by the vertex position along the beam axis and the maximum allowed distance from that vertex in both the transverse plane and along the beam axis. Two concrete implementations are provided, `GlobalTrackingRegion` and `RectangularEtaPhiTrackingRegion`. The latter allows the direction of the track to be constrained to be within a given range of η and φ.

In CMS, the constraints on η and φ for `RectangularEtaPhiTrackingRegion` would most likely come from the other detectors, e.g. from a candidate μ in the muon system. In this study, which is specific to the Pixel Detector, the constraints are taken from the knowledge of the true η and φ directions of the track which gives rise to the hit(s).

The Pixel Reconstruction software is split into several packages, as is illustrated in Fig. 1. Its action can be decomposed into four steps.

- *Estimation of the track position uncertainties* – parametrization of hit errors, multiple scattering and longitudinal bending effects (Sections 3.1, 3.2, 3.3),
- *Construction of pixel track candidates from pixel hits* – searching for hits, hit pairs and hit triplets (Sections 4 and 5),

- *Fitting of pixel tracks and finding primary vertices* – assignment of kinematic parameters and their errors to pixel tracks, finding vertices by associating groups of tracks (Sections 6 and 7),
- *Higher-level applications* – combinations of basic functionalities with additional filtering, cleaning and efficiency recovering, conversion into standard ORCA object (`RecTracks`).

At each step the results can be used to construct track seeds (Section 8) and start the full tracker reconstruction.

Single muon events are used for all algorithmic efficiencies presented in this note. The muons were generated with transverse momenta $p_T(\mu) \in [1.0,\ 10.0]$ GeV/c, pseudorapidity $|\eta(\mu)| < 2.5$ and transverse impact parameter $d < 0.2$ cm. For all algorithmic efficiencies the search region is defined using the generated track information. The maximum transverse impact parameter of a candidate track is set to $\Delta r = d/0.95$. The allowed z range $[z_{\rm MIN}, z_{\rm MAX}]$ has a size $\Delta z = z_{\rm MAX} - z_{\rm MIN}$ and is set to 0.2 cm. This range is selected in such a way that it contains the muon true vertex position randomly distributed with uniform probability over the $[z_{\rm MIN}, z_{\rm MAX}]$ range. Similarly, the region direction is not perfectly centred on the muon but is selected randomly to contain the muon within the region $\Delta\eta$ and $\Delta\varphi$. The minimum transverse momentum is set to $0.95 \cdot p_T^{\rm TRUE}$.

For most other performance plots more complex h→ee$\mu\mu$ events are used. For these events, in the case of global reconstruction, no generator-level information is used during the reconstruction process and the reconstruction region is constrained by $p_T^{\rm min} = 1$ GeV/c, $z \in [-15, +15$ cm], $\Delta r = 0.2$ cm. In the case of regional reconstruction the direction of the region is still selected based on the generator information, since no data from other detectors was used in the reconstruction.

3 Estimate of track position uncertainties

3.1 Parametrization of Hit Errors

The average spatial resolutions of the barrel pixel hits are about 20 μm and 10 μm in the longitudinal and transverse directions respectively. The forward disks provide measurement points with a resolution of around 17 μm in the φ direction and 12 μm in the transverse plane pointing towards the beam axis. The spatial resolution of pixel hits strongly depends on the cluster size, and a more precise evaluation of the resolution can be achieved with knowledge of the track impact angle onto the detector module. The description of the technique used to perform a precise parametrization of pixel hit errors can be found in Ref. [3].

3.2 Parametrization of Multiple Scattering

A charged particle passing through material is affected by multiple scattering ([4] and Appendix of this note). The track deflection due to multiple scattering can be substantial and should be taken into account wherever an accurate prediction of the particle position in space is needed. In the full online or offline track reconstruction, multiple scattering effects are included in the error matrix during state propagation. For the approximate prediction needed for track seeding purposes, a faster parametrization of multiple scattering is required. Such a parametrization has been developed and implemented in the ORCA reconstruction software. The goal of the parametrization is to generate

1. the track position uncertainty in a given detector layer when the track originates from the interaction point;
2. the track position uncertainty in a given detector layer when the track is constrained (by its measurement) in an inner detector layer;
3. the track position uncertainty in a given detector layer when both inner and outer constraints (measurements) are given.

The simplified track reconstruction is based on the Pixel Detector. The parametrization is thus optimized for the inner part of the Tracker. The approximations and algorithms used for the parametrization are explained in the Appendix.

The reliability of the parametrization is demonstrated in Figs. 2–5. Figure 2 shows the test of item 1 above. The r.m.s. displacement predicted by the parametrization is compared with that obtained from the detailed simulation. Similarly, the tests of items 2 and 3 are presented in Figs. 3 and 4 respectively. The predicted spread matches well that of the detailed simulation. The effect of non-Gaussian tails in the multiple scattering is displayed in

Fig. 5, which shows that the number of standard deviations needed to obtain a given efficiency does not follow the expectation from a Gaussian distribution.

3.3 Track bending in the longitudinal plane

A charged particle trajectory in a uniform magnetic field follows a helix. Its projection onto the plane transverse to the magnetic field is a circle. The projection onto the rz (longitudinal) plane during its initial trajectory can be approximated by a straight line. This approximation is good for high p_T tracks. However for transverse momenta of $\mathcal{O}(1$ GeV/$c)$ and a position measurement accuracy of $\mathcal{O}(10$ μm) it might be insufficient, and the second order term has to be taken into account:

$$r_{\text{straight line}} = r + \frac{r^3}{24\rho^2}, \tag{1}$$

where r is the true radial position of the particle, $r_{\text{straight line}}$ the radius from the straight line prediction and ρ the inverse curvature (i.e. the bending radius in the transverse plane). The effect of projecting a helix onto the longitudinal plane is shown in Fig. 6. The deviation between the straight line propagation and the helix projection is shown as a function of radius for tracks of 1, 2 and 3 GeV/c transverse momentum. The second order corrections coincide well with the exact helix computation, except for large radii and low transverse momenta.

4 Finding Hit Pairs

A hit pair consists of an *outer* hit (with larger radius) and an *inner* hit (smaller radius) coming from two different detector layers. The various combinations of layers and disks used to find hit pairs are shown in Fig. 7. The combinations were chosen to provide redundancy in searching for pairs of hits, to satisfy the requirement of maximizing the efficiency for generating seeds. Tracks that lose a hit in any one of the layers may still be reconstructed in the other layers. To find pairs of hits, first an outer hit is searched for. The outer hit and the vertex constraint are then used to search for a second (inner) hit in a layer between the vertex and the outer hit.

Searching for an *outer* hit within a given layer is straightforward in the case of the `GlobalTrackingRegion`, where all hits from a `DetLayer` (ORCA terminology corresponding in the case of the Pixel Detector to a disk or a layer) are available. For the `RectangularEtaPhiTrackingRegion`, the allowed ranges for the position of the hits are predicted analytically from the minimum allowed momentum, the range of possible track directions and the interaction point constraint (Fig. 8). The search area is extended to account for multiple scattering (3σ level), hit errors (3σ level) and the non-linear projection of the helix. Independent constraints on the hit position are constructed for the φ and z (barrel) or φ and r (endcap) coordinates. During the search, first the φ and then the z (r) constraint is checked. In order to speed up the algorithm, first the set of detectors (`DetUnits`) compatible with the tracking region is defined and then the hits from those detectors are examined.

The algorithmic efficiency for finding (outer) hits within the `RectangularEtaPhiTrackingRegion` is shown in Fig. 9. The efficiencies are shown for the *Narrow Region*, defined only by the size of the hit position errors (3σ) and multiple scattering (3σ), i.e. the allowed $\Delta\eta$ and $\Delta\varphi$ of the direction of the particle are set to 0. This serves as a test of the correctness of the position error and multiple scattering parametrizations. The algorithmic efficiencies obtained are better than 99.6% and depend neither on the track pseudorapidity nor on its transverse momentum (p_T).

The second hit is required to be *inner* with respect to the first one. The analytical prediction for its position is computed independently in the φ and z (barrel) or φ and r (endcap) coordinates (Fig. 10), taking into account the same uncertainties as for the outer hit. Since hits from a given `DetLayer` are accessed more than once, the relevant hits from a given layer are stored in a cache. They are stored in $\Delta\eta \times \Delta\varphi$ bins, sorted in φ, which allows faster hit accessing.

The algorithmic efficiency for finding hit pairs is shown in Fig. 11 for `RectangularEtaPhiTrackingRegion` of size $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$ ($\Delta\varphi$ in this note is in radians). The efficiency does not depend on the track p_T and there are small (less than 1%) inefficiencies in the endcap regions. The absolute efficiency to find a hit pair per generated track is shown in Fig. 12. The effect of pile-up[1)] is below 1%, as shown in Fig. 13.

[1)] In this note "low luminosity pile-up" is defined to be the pile-up at a luminosity of 2×10^{33} cm^{-2}s^{-1} and corresponds to about 5 "minimum-bias" events per beam-crossing. "High luminosity pile-up" is defined to be for a luminosity of 10^{34} cm^{-2}s^{-1} and corresponds to about 25 minimum-bias events per beam-crossing.

The number of reconstructed pairs depends on the luminosity and the size of the reconstruction region. Figure 14 shows the number of pairs per region for more complex $\mathrm{h}\rightarrow\mathrm{ee}\mu\mu$ events as a function of region size. There are two contributions to the increasing number of pairs with increasing size of the region: the number of real tracks increases with increasing size of the region, and the combinatorial background increases with increasing number of hits in the region. Algorithms optimized for efficiency obviously generate more fake pairs. The ratio of ghosts to real tracks increases by a factor of about 100 for the global region (`GlobalTrackingRegion` shown in the right-most bin). This can also be seen in Fig. 15 where the hit pair purity, defined as the ratio of the number of fully reconstructed tracks to the number of initial pairs, is shown.

The CPU time needed to reconstruct hit pairs is shown in Fig. 16. This was measured using a 2.8 GHz Xeon processor, and includes hit accessing, reconstruction (but not digitization) and hit-pair searching and creation. For low luminosity the regional reconstruction time is a few milliseconds, reaching about 30 msec for global reconstruction. At high luminosity the regional reconstruction time is well below 20 msec and the global one is about 100 msec.

5 Finding Hit Triplets

In contrast to pair finding, finding pixel triplets requires hits from three different layers or disks. Unavoidable detector and readout data losses introduce significant inefficiencies which dominate over the algorithmic inefficiencies. Since an efficiency close to 100% is not achievable, the detailed and time-consuming corrections due to multiple scattering, hit errors and longitudinal bending are not used. Instead, effective tolerances, adjusted empirically, are applied.

Triplet finding is based on adding a third hit to pairs of hits. Each combination of pixel layers used in pair finding defines the layer in which the search for a third hit is performed. The layers and disks chosen are shown in Fig. 17. In order to find hits compatible with a given hit pair, a search window in the relevant layer or disk is defined. This window is formed by the tolerances in $\Delta\varphi$ and Δz in the barrel and $\Delta\varphi$ and Δr in the endcaps. The $\Delta\varphi$ window is centred on the outer hit of the pair. The Δz (Δr) tolerance is centred on the line connecting the two hits. The values for the $\Delta\varphi$ and Δz (Δr) tolerances (0.03 rad and 0.03 (0.02) cm, respectively) are optimized for tracks with transverse momentum above 1 GeV/c.

The φ coordinate of the selected third hit is checked using the region constraints and a more accurate φ prediction coming from a fast circle fitting method based on the parabolic approximation [5] applied in inverted [cm^{-1}] coordinates. These coordinates map the Cartesian x, y to $u = x/(x^2 + y^2), v = y/(x^2 + y^2)$, with the relation $v = A + B \cdot u + C \cdot u^2$, where A, B and C are related to the curvature, initial direction and impact parameter. In the procedure described above, the hit cache is used for fast access to the hits.

The performance plots for finding triplets are shown in Figs. 18–21. The algorithmic and absolute (including detector data losses) efficiencies are shown in Fig. 18. The lack of multiple scattering corrections is responsible for a few percent loss of efficiency for muons of p_T less than 2.5 GeV/c. The absolute efficiencies are approximately 10% lower than the algorithmic ones, which have a plateau at 98%. Triplet finding rapidly loses efficiency for tracks with pseudorapidity greater than about 2.1, due to the detector geometrical configuration. The efficiency is shown in Fig. 19 for selected tracks in pure signal events, and with low and high luminosity pile-up superimposed. The luminosity dependent losses cause up to a 10% reduction in efficiency. The number of reconstructed hit triplets (Fig. 20) is roughly a factor 100 smaller than the number of hit pairs (Fig. 14). This reflects the fact that hit triplets are much cleaner than hit pairs, since the combinatorial background of hit pairs is much reduced by requiring a third hit and the number of "ghost" triplets is small. The purity of these tracks for different region sizes and the two luminosities is shown in Fig. 21. It varies from 80 (90)% to almost 100% for high (low) luminosity, depending on the region size.

The CPU time required is shown in Fig. 22. It is comparable with the time to find hit pairs.

6 Evaluation of the Parameters of Pixel Tracks

In this section, each triplet is used to form a track and to subsequently determine its transverse momentum and longitudinal and transverse impact parameters (IP) [6]. Results of the studies presented in this section refer to single muon tracks.

A triplet defines a circle in the transverse plane from which the value of the radius of curvature R and transverse momentum $p_T \simeq 0.003 \cdot \mathrm{B} \cdot R$ (p_T in GeV/c, B in T and R in cm) are extracted. However the small lever arm of the

Pixel detector allows a reliable such estimate to be made only for p_T below $\sim$10 GeV/c. In Fig. 23a, the resolution $\sigma(p_T)/p_T$ is shown as a function of p_T, where $\sigma(p_T)$ is the width of a Gaussian fit to the p_T distribution. The relative momentum resolution is well parametrized by $\sigma(p_T)/p_T = 0.055 + 0.17 p_T$ (p_T in GeV/c). In Fig. 23b $\sigma(p_T)/p_T$ is shown as a function of pseudorapidity for p_T of 1 and 10 GeV/c. For 1 GeV/c, the relative p_T resolution is as good as 7%, but it already amounts to 22% for 10 GeV/c tracks.

The track transverse impact parameter is defined as the distance of the point of closest approach to the beam axis and is denoted $\mathrm{IP}_{r\varphi}$. The transverse impact parameter is determined from the coordinates of the centre $(x_\mathrm{C}, y_\mathrm{C})$ and radius R of the unique circle that passes through the three pixel hits. The unsigned distance of closest approach of the circle to the beam axis is used as an estimate of $\mathrm{IP}_{r\varphi}$:

$$\mathrm{IP}_{r\varphi} = |\sqrt{x_\mathrm{C}^2 + y_\mathrm{C}^2} - R|.$$

For large transverse momentum, this expression contains a difference between two large terms, which may lead to numerical inaccuracies. To alleviate this potential problem, the circle passing through the three pixel hits can be approximated by a parabola [5]. The equation of the parabola, expressed in the reduced coordinates defined as

$$u = \frac{x}{x^2 + y^2}, \quad v = \frac{y}{x^2 + y^2},$$

is $v = p_1 + p_2 u + p_3 u^2$, with

$$p_1 = \frac{1}{y_\mathrm{C}}, \quad p_2 = -\frac{x_\mathrm{C}}{y_\mathrm{C}}, \quad p_3 = -\left(\frac{R}{y_\mathrm{C}}\right)^3 \mathrm{IP}_{r\varphi}.$$

The transverse impact parameter resolution is shown in Fig. 24a as a function of p_T for pseudorapidity $|\eta| < 1.7$ and $|\eta| > 1.7$. Above 6 GeV/c, the transverse impact parameter resolution is around 80 μm. When hits from the Silicon Strip Tracker detector are included in the reconstruction, this resolution significantly improves to about 20 μm.

To estimate the longitudinal impact parameter, z_IP, a helix parametrization is used. The three pixel hits are projected onto the (ψ, z) plane, where ψ is the azimuthal angle between the hit and the point of closest approach around the circle defined by the three hits in the transverse plane. In this plane, the helix projection is expected to be exactly a straight line, up to uncertainties due to the hit position measurement and multiple scattering in the detector material. The longitudinal impact parameter is defined as the point of intercept between the line joining the first two pixel hits $(\psi_{1,2}, z_{1,2})$ and the z axis :

$$z_\mathrm{IP} = z_1 - \frac{\psi_1}{\psi_1 - \psi_2}(z_1 - z_2). \tag{2}$$

The longitudinal IP resolution, σ_{z_IP}, is shown in Fig. 24b as a function of pseudorapidity η for three different values of p_T. The poorer resolution at low η reflects the fact that the pixel position resolution in the z direction is poor for low η tracks, due to the absence of charge sharing between pixels. The longitudinal IP resolution can be improved by a factor of two if the full Tracker information is used.

7 Pixel Vertexing

Primary-vertex finding based on the pixel hits provides a simple and efficient method for measuring the position of the primary-vertex (PV). This measurement is subsequently used for track seeding and in many High-Level Trigger (HLT) analyses. It must therefore be sufficiently accurate and fast. For this reason PV finding is reduced here to a one-dimensional search along the z axis.

The two PV finding algorithms described below make use of the pixel tracks reconstructed from hit triplets (Section 5). Usually triplets found in the full pixel detector acceptance (global region) are used. However the triplet finding can also be restricted to a selected region in order to make the vertex finding faster. The search for the primary vertex along the z axis is based on the longitudinal impact point, z_IP, evaluated from pixel tracks (Section 6). Only pixel tracks reconstructed with p_T in excess of 1 GeV/c and a transverse impact point smaller than 1 mm are used for primary vertex finding.

Two vertex-finding algorithms have been implemented and tested. The *histogramming method* progressively merges tracks close enough to each other according to their longitudinal impact points, z_IP, to form PV candidates. The *divisive method* looks for large z_IP intervals without tracks to divide the z axis into several regions. An

average PV position is computed from all tracks in each region. Tracks not compatible with that average position are discarded. The discarded tracks are recovered to make a new vertex candidate. This iterative procedure stops when all tracks are found to be compatible with the corresponding PV positions. Both PV finding algorithms are described in more detail in Ref. [6].

After vertex finding, the vertex candidates are sorted in decreasing order according to the sum of their track p_T^2, with a p_T ceiling at 10 GeV/c to account for the imprecision in the p_T measurement for three-hit tracklets. A detailed description of the algorithms can be found in Ref. [6].

Among the primary-vertex candidates, the *closest* primary vertex is defined as that closest in z to the simulated signal primary vertex, and the *tagged* primary vertex as the one with the largest p_T sum. The efficiency to find the primary vertex (tagged $\varepsilon_{\rm tag}$ or closest $\varepsilon_{\rm close}$) is defined with respect to vertices reconstructed inside a window of 500 μm around the position of the generated signal primary vertex. The primary vertex finding efficiency achieved for the signal primary vertex is given in Table 1 for both algorithms, with low-luminosity pile-up.

Table 1: Efficiencies of signal primary vertex finding for the histogramming and divisive pixel primary-vertex finder algorithms, at low luminosity.

	Histogramming		Divisive	
	$\varepsilon_{\rm close}$	$\varepsilon_{\rm tag}$	$\varepsilon_{\rm close}$	$\varepsilon_{\rm tag}$
u-jets; $E_T = 10$ GeV	0.99	0.98	1.00	0.99
u-jets; $50 < E_T < 100$ GeV	0.97	0.90	0.99	0.94
b-jets; $E_T = 100$ GeV	0.98	0.96	0.99	0.99
b-jets; $30 < E_T < 50$ GeV	0.97	0.89	1.00	0.96
H (115 GeV/c^2) $\rightarrow \gamma\gamma$, g fusion	0.89	0.75	0.94	0.80
H (150 GeV/c^2) $\rightarrow$ ZZ $\rightarrow 2e2\mu$	0.97	0.96	1.00	1.00
$B_s^0 \rightarrow J/\psi\phi$	0.81	0.61	0.97	0.78
$t\bar{t}$	0.99	0.98	1.00	1.00
$t\bar{t}H$, H (120 GeV/c^2) $\rightarrow b\bar{b}$	1.00	0.99	1.00	1.00

The divisive method generally has a better primary vertex efficiency for the different event samples. This is because it takes into account the compatibility of the tracks with the estimate of the vertex, instead of considering only the compatibility of adjacent tracks. For most event types the primary vertex is recovered with an efficiency close to 100%. The primary-vertex finding efficiencies are significantly below 100% for events such as H $\rightarrow \gamma\gamma$, where the low charged-particle multiplicity does not allow the signal primary vertex to be always distinguished from the pile-up primary vertices. Other methods to find the most likely signal PV specific to these physics channels are under investigation. At high luminosity, the inefficiency roughly doubles with respect to the results of the divisive algorithm at low luminosity, as explained in Ref. [6].

The PV z position is reconstructed typically with a resolution of better than 50 μm for both the histogramming and divisive methods [6]. Examples of the position resolution of the reconstructed PV obtained with the divisive method are shown for 100 GeV u-jet events in Fig. 25a and for 100 GeV b-jet events in Fig. 25b. The position resolution for both algorithms, for several event types, with low-luminosity pile-up, is summarized in Table 2.

The average time for PV finding is 0.7 ms per event, for both the histogramming and divisive methods. The time was measured on a 2.8 GHz PentiumIV, for $q\bar{q}$ events with $E_T^{\rm Jet} = 100$ GeV, at high luminosity. The time quoted does not include the contributions from hit reconstruction and triplet finding.

Table 2: Spatial resolution of the z coordinate of the signal primary vertex for the histogramming and divisive pixel primary-vertex finder algorithms, at low luminosity.

	Histogramming		Divisive	
	$\sigma(\mu\mathrm{m})$	RMS (μm)	$\sigma(\mu\mathrm{m})$	RMS (μm)
u-jets; $E_T = 100\,\mathrm{GeV}$	33.3	37.0	27.1	28.7
u-jets; $50 < E_T < 100\,\mathrm{GeV}$	41.5	55.8	33.2	42.0
b-jets; $E_T = 100\,\mathrm{GeV}$	45.9	54.1	38.1	48.3
b-jets; $30 < E_T < 50\,\mathrm{GeV}$	50.4	65.6	44.4	55.9
H ($115\,\mathrm{GeV}/c^2$) $\rightarrow \gamma\gamma$, g fusion	51.1	64.9	39.1	53.3
H ($150\,\mathrm{GeV}/c^2$) $\rightarrow$ ZZ $\rightarrow 2\mathrm{e}2\mu$	39.7	45.5	30.3	34.4
$\mathrm{B_s^0} \rightarrow \mathrm{J}/\psi\phi$	69.4	90.2	74.8	98.5
$\mathrm{t\bar{t}}$	40.1	51.4	32.5	40.2
$\mathrm{t\bar{t}H}$, H ($120\,\mathrm{GeV}/c^2$) $\rightarrow \mathrm{b\bar{b}}$	38.6	51.3	28.7	32.5

8 Generation of Seeds

A trajectory seed is the starting point for the pattern recognition in the tracker. The seed should constrain all five track parameters, the estimate of these parameters should be sufficiently close to their true value to allow the use of linear fitting algorithms (e.g. Kalman filter) and the uncertainties of the parameters should be sufficiently small to allow a reasonably compact search region for hits.

8.1 Seeds from hit pairs

The minimal information from the pixel detector needed to construct a trajectory seed is a hit pair. Since hit pairs do not constrain the momentum, the additional assumption that the track passes through either a known vertex or the centre of the beam spot must be used.

The parameters of the seed are first estimated at the centre of the beam spot, using the equations of an ideal helix passing through the two hits and the beam axis (the beam spot does not constrain the seed in z). The uncertainties assigned to the helix at the beam spot are

- 1 rad in the polar and the azimuthal angles,
- 1 $(\mathrm{GeV}/c)^{-1}$ on the inverse momentum,
- the transverse and longitudinal uncertainty on the position of the vertex, or the dimensions of the "region of interest" defined by the tracking region.

The initial track parameters are then propagated to the surface of the closest hit and updated with the hit measurement information. The updated track state is again propagated to the surface of the outer hit, and updated with its measurement information. The resulting track state includes an error matrix which takes into account the size of the region, the hit errors, and the multiple scattering. The procedure is slow (compared to hit pair finding) due to the two propagations and updates, and takes about 0.3 msec per seed.

8.2 Seeds from hit triplets

The same sequence of operations can be applied to transform a hit triplet into a trajectory seed, the only difference being an additional propagation and update on the surface of the third hit.

The seeding from triplets versus seeding from pairs has the same advantages and disadvantages as the hit triplet finding versus hit pair finding described in the previous sections: the number of seeds is much smaller and the purity of the seeds is high, but their efficiency is significantly lower. For these reasons the standard track reconstruction uses seeds from hit pairs and seeding from triplets is used in special cases, such as reconstruction of events from heavy-ion collisions.

9 Conclusions

Reconstruction algorithms using the Pixel Detector, including pixel track reconstruction, primary vertex finding and track seed generation have been developed. The key feature of these algorithms is a track reconstruction based on two or three pixel hits. Track parameters are calculated with a simple helix approach. It has been demonstrated that a high track reconstruction efficiency ($\sim 90\%$) and a low fake rate can be achieved. The signal primary vertices can be found and their position along the z axis can be measured with a resolution better than $50\ \mu\mathrm{m}$. The pixel standalone reconstruction is useful for the online HLT event selection.

Acknowledgments

Two of us (MK, SC) are grateful to Ludwig Tauscher for the nice and fruitful collaboration during their stay at the University of Basle.

We would like to thank Marcel Vos for his comments and careful testing of our code.

We also thank Ken Bell for his help in the preparation of this note.

References

[1] CMS Data Acquisition & High Level Trigger TDR, Volume II, CERN/LHCC 02-26.

[2] http://cmsdoc.cern.ch/orca/.

[3] S. Cucciarelli, D. Kotlinski and T. Todorov, *Position Determination of Pixel Hits*, **CMS NOTE 2002/049**.

[4] V. Highland, Nucl. Instrum. and Methods **129** (1975) 497;
G.R. Lynch and O.I. Dahl, Nucl. Instrum. and Methods **B58** (1991) 6.

[5] M.Hansroul, H.Jeremie, D.Savara, Nucl. Instrum. and Methods **A270** (1998) 490.

[6] S. Cucciarelli *et al.*, *Track-Parameter Evaluation and Primary-Vertex Finding with the Pixel Detector*, **CMS NOTE 2003/026**.

Appendix: Multiple Scattering Parametrization - technicalities

Thin layer approximation

Multiple scattering results from a large number of deflections dominated by the Coulomb scattering described by the Moliere theory. The effect can be well described [4] with a Gaussian approximation of the projected angular distribution. The width of the Gaussian distribution for a particle of unit charge can be written as:

$$\sigma'_\theta = \sin\theta \cdot \sigma_\theta = \sin\theta \cdot \frac{0.0136\,\mathrm{GeV}}{\beta c\, p_T}\sqrt{x/X_0}[1 + 0.038\ln(x/X_0)], \tag{3}$$

where σ'_θ is the sigma of the direction deflection projected onto any plane containing the initial direction, θ the polar angle[2)], $p_T/\sin\theta$ the momentum, βc the velocity and x/X_0 the thickness of the medium in radiation lengths along the track direction.

If the detector is composed of several thin detector layers the multiple scattering can be considered as a product of scatterings on thin surfaces. The sigma of the position deflection in detector m, projected onto a plane transverse to the beam line (i.e. in the $r-\varphi$ plane), for a particle coming from the interaction point and crossing layers i to m at transverse radii $r_i \ldots r_m$, is given by:

$$\sigma^2_{r\varphi} = \sum_{k=i}^{k<m} [\sigma_{\theta k} \cdot (r_m - r_k)]^2 \tag{4}$$

One can show that if an additional constraint (measurement) is known in detector n, outer to m, the formula for the transverse deflection $\sigma_{r\varphi}$ takes the following form:

$$\sigma^2_{r\varphi} = \left(\frac{r_n - r_m}{r_n - r_i}\right)^2 \sum_{k=i+1}^{k\le m} [\sigma_{\theta k} \cdot (r_k - r_i)]^2 + \left(\frac{r_m - r_i}{r_n - r_i}\right)^2 \sum_{k=m+1}^{k<n} [\sigma_{\theta k} \cdot (r_n - r_k)]^2 \tag{5}$$

These formulae are valid for barrel layers and endcap disks and allow one to estimate the multiple scattering for all three cases listed in Section 3.2. Although there is no explicit dependence on the polar angle θ, the contributing thickness in radiation lengths should be calculated along the track, i.e. one has $x_k = x_{\perp k}/\sin\theta$ for barrel layers and $x_k = x_{\perp k}/\cos\theta$ for endcap disks (where $x_{\perp k}$ is the medium thickness).

Algorithm

The Tracker layout can be approximated be a set of thin layers (which will be called *MS-layers* to distinguish them from the standard ORCA *DetLayers*) representing various detector layers and their support. The sets (containers) of MS-layers are constructed in pseudorapidity bins (ranges). The MS-layers representing tracker sensitive detectors are attached to the sets. The sets are then supplemented with additional (not present in the reconstruction geometry) MS-layers representing the beam line and all pixel detector support structures. Each MS-layer is then assigned a value of x/X_0 by associating the material between physical layers to the closest MS-layer. The x/X_0 information is extracted from the simulation geometry, to benefit from the fact that the Tracker description in the CMS detector simulation program includes not only sensitive material but also supports, cabling, cooling pipes and electronic chips. The summed thickness of material in radiation lengths (x/X_0) is stored in an external datafile and is used by the multiple scattering algorithm in ORCA.

[2)] Note that this formula does not depend on θ since it also enters into the p_T. The splitting of the $\sin\theta$ out of σ'_θ is useful in the detector coordinate frame.

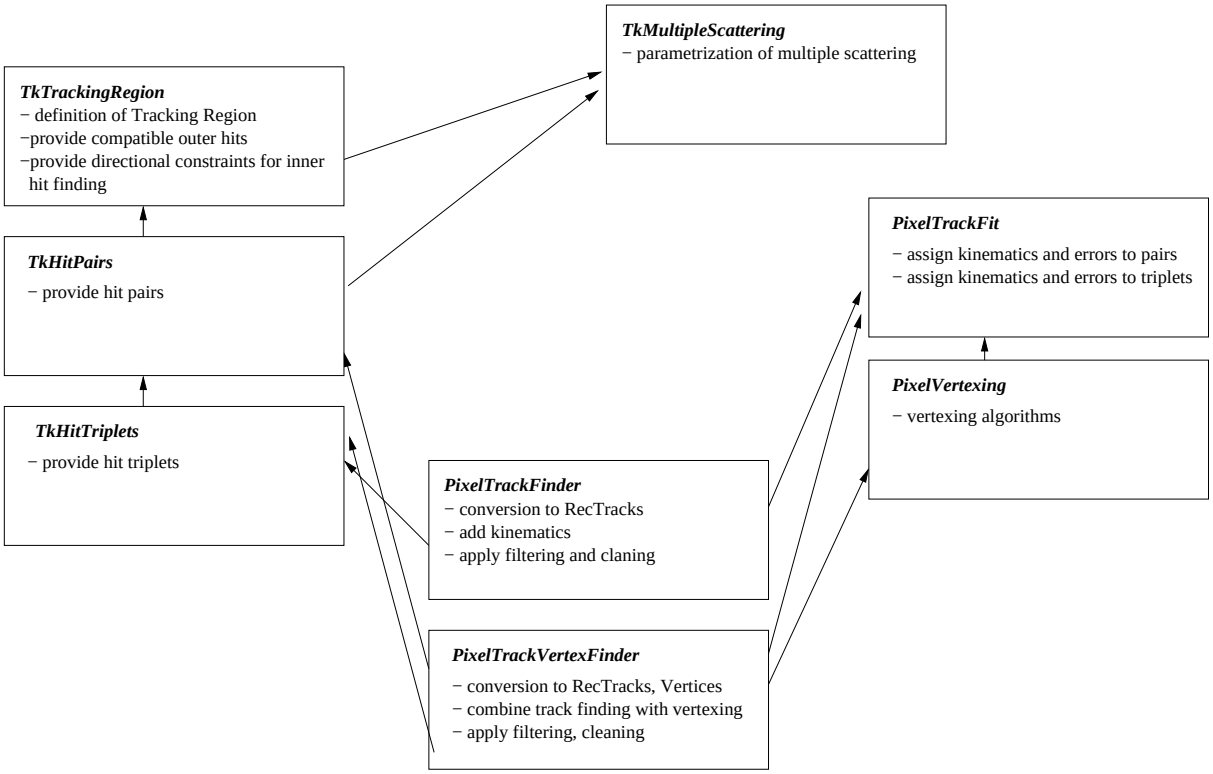


Figure 1: Code components and their dependencies.

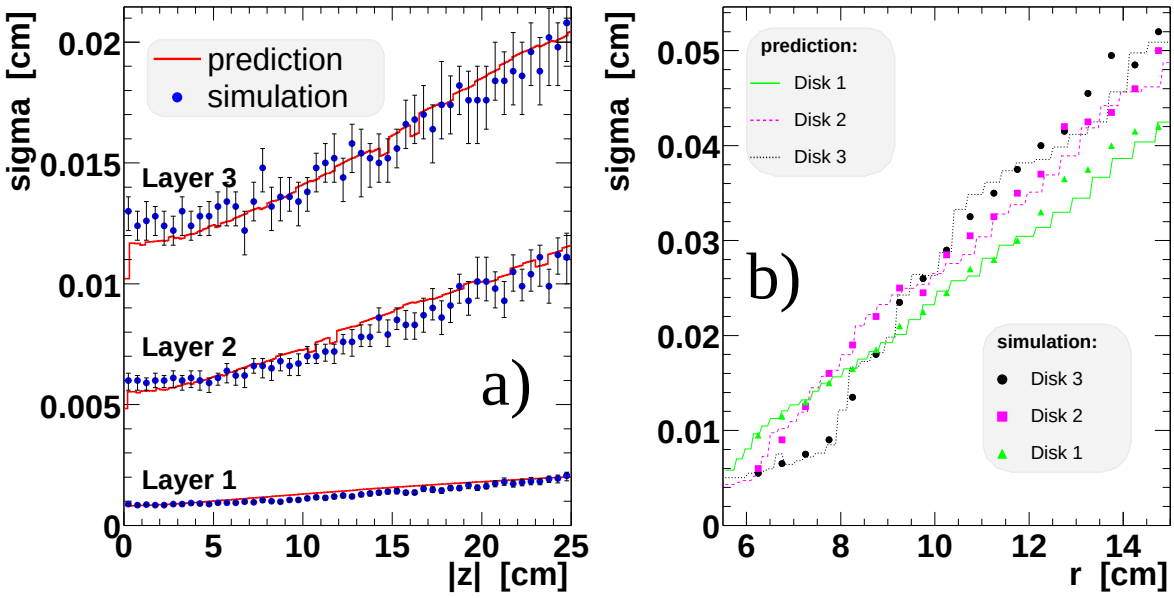


Figure 2: Parametrization of multiple scattering for (a) pixel barrel layers 1, 2 and 3, and (b) pixel endcap disks 1, 2 and 3, shown as the r.m.s. trajectory deflection in the $r\varphi$ plane for tracks constrained by the primary vertex. The points show the 1σ differences between the positions given by the full simulation and the ideal helix, for $p_T = 1$ GeV/c tracks. The solid lines show the spread predicted by the parametrization. For the purposes of this plot, a third pixel disk is added (but is not otherwise used in the simulations presented in this note and is not foreseen to be installed in CMS).

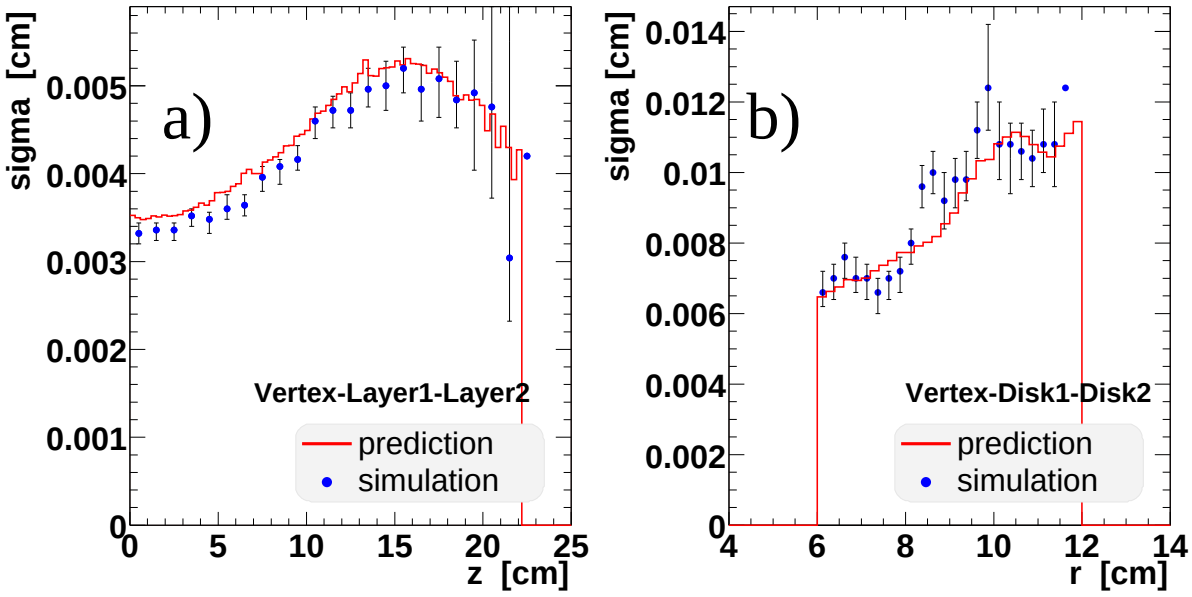


Figure 3: Parametrization of multiple scattering. (a) The 1σ trajectory deflection in barrel layer 1 for tracks constrained by the primary vertex and a measurement in barrel layer 2. (b) The 1σ trajectory deflection in endcap disk 1 for tracks constrained by the primary vertex and a measurement in endcap disk 2. The points show the 1σ differences between the positions given by the full simulation and the ideal helix, for $p_T = 1$ GeV/c tracks. The solid lines show the predictions of the parametrization.

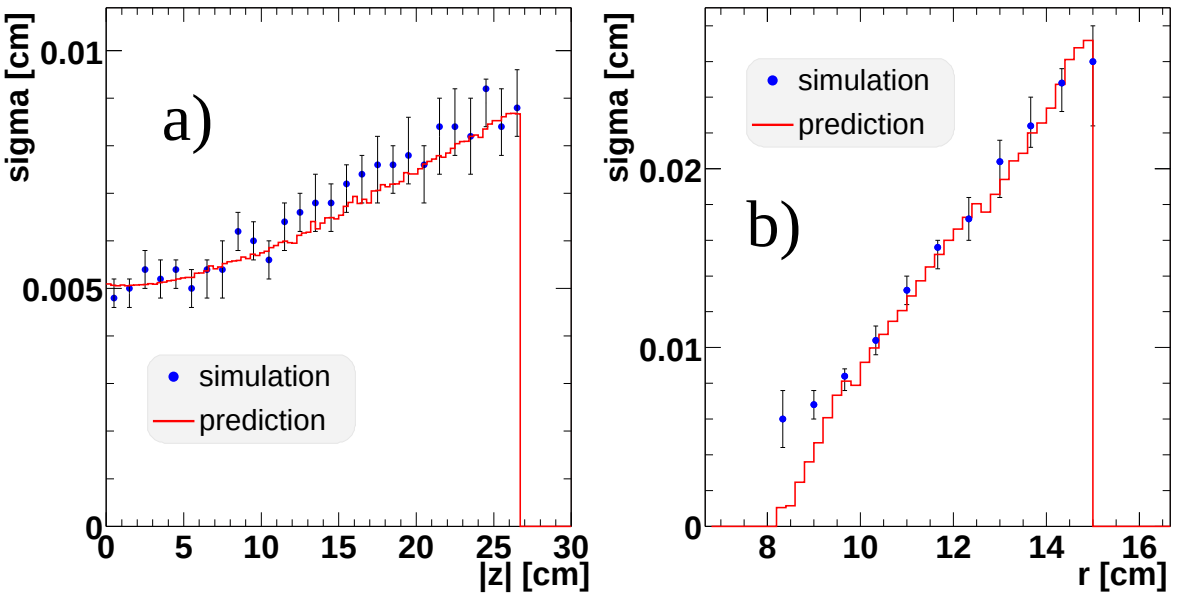


Figure 4: Parametrization of multiple scattering for (a) barrel pixel layer 2, and (b) forward pixel disk 2, for tracks constrained by a measurement in the preceding layer or disk. The points show the 1σ differences between the positions given by the full simulation and the ideal helix, for $p_T = 1$ GeV/c tracks. The solid lines show the predictions of the parametrization.

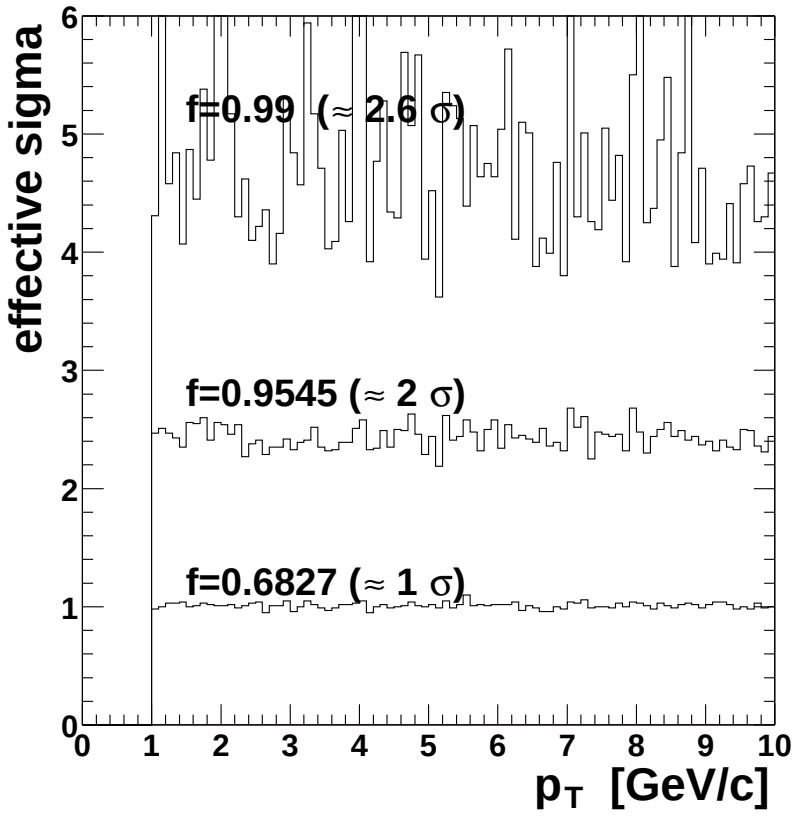


Figure 5: The effect of non-Gaussian tails in multiple scattering. The number of effective sigmas, given by the multiple scattering parametrization, necessary to accumulate the fraction f of the deflected track population, is shown as a function of track transverse momentum, for a single charged particle in a barrel layer. For a perfect Gaussian distribution the chosen values of f would correspond to 1, 2 and about 2.6 standard deviations.

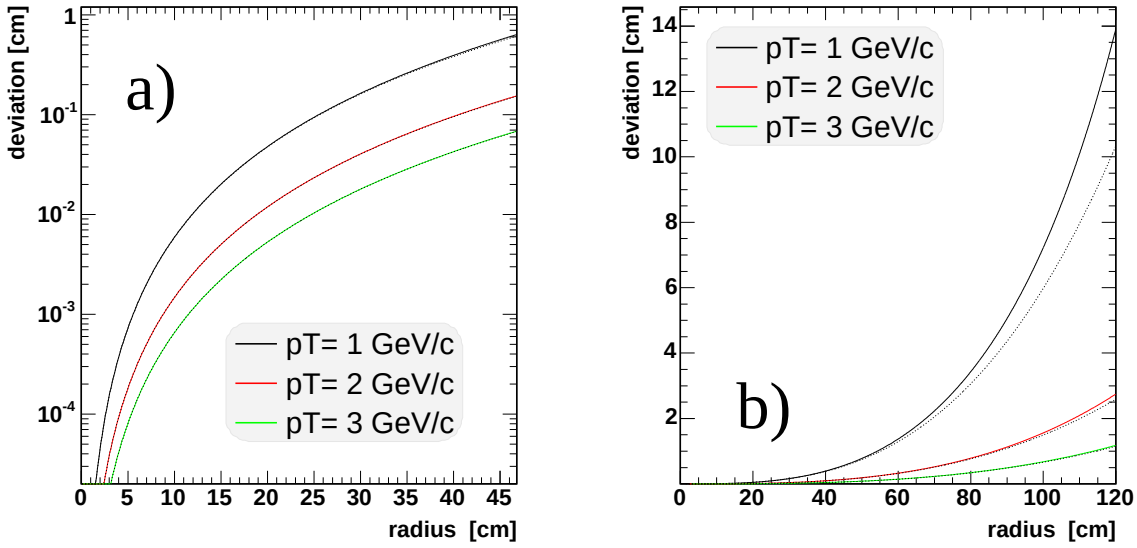


Figure 6: The difference between straight line propagation and helix projection is shown as a function of radius, in log scale (a) and linear scale (b) for tracks of 1, 2 and 3 GeV/c transverse momentum (full curves). The estimate from Eq. 1 is shown by dotted curves. These coincide with the full curves, except at large radii and low p_T.

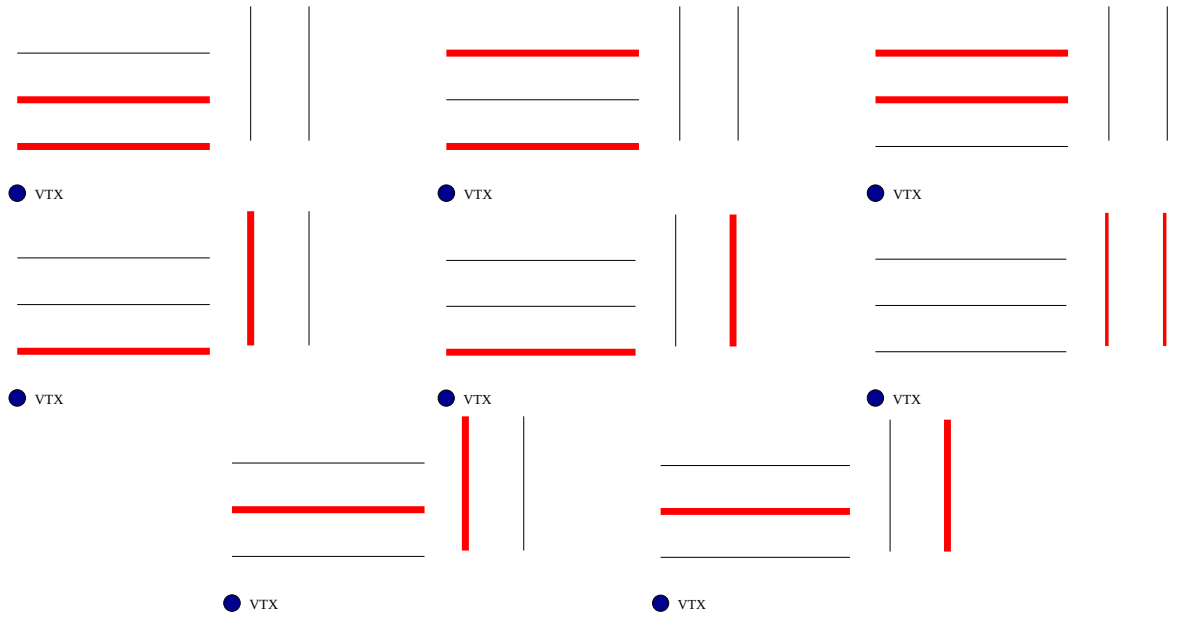


Figure 7: The combination of layers (bold lines) used for finding pairs of hits.

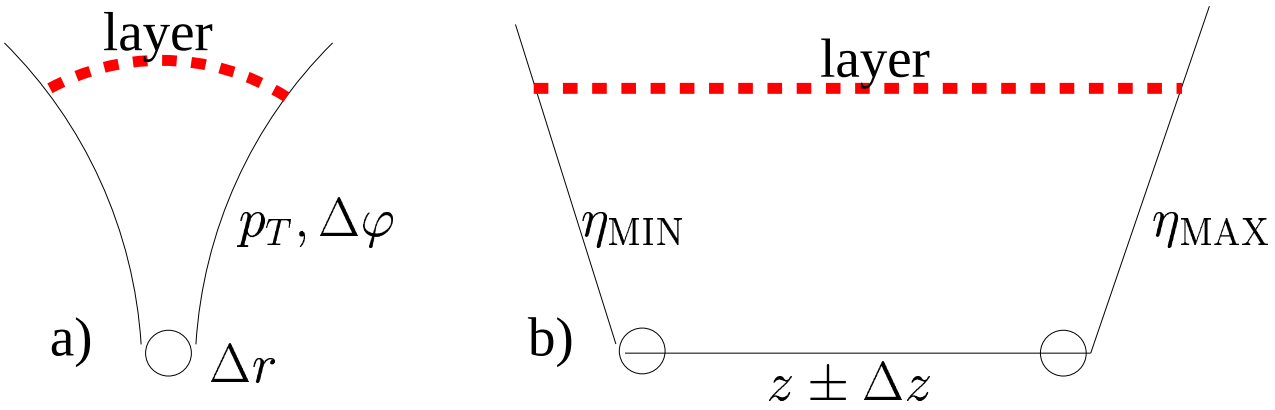


Figure 8: Illustration of the (initial) *outer* hit finding in the `RectangularEtaPhiTrackingRegion`. The allowed ranges for the positions of the hits are marked with bold-dashed lines. The φ interval is determined by the range of the momentum direction, the range of the value of the transverse momentum and the primary vertex constraint. The range in the z or r coordinate is given by the constraints on the track direction and primary vertex. Track prediction uncertainties are also taken into account in both cases, but are not shown in the plot.

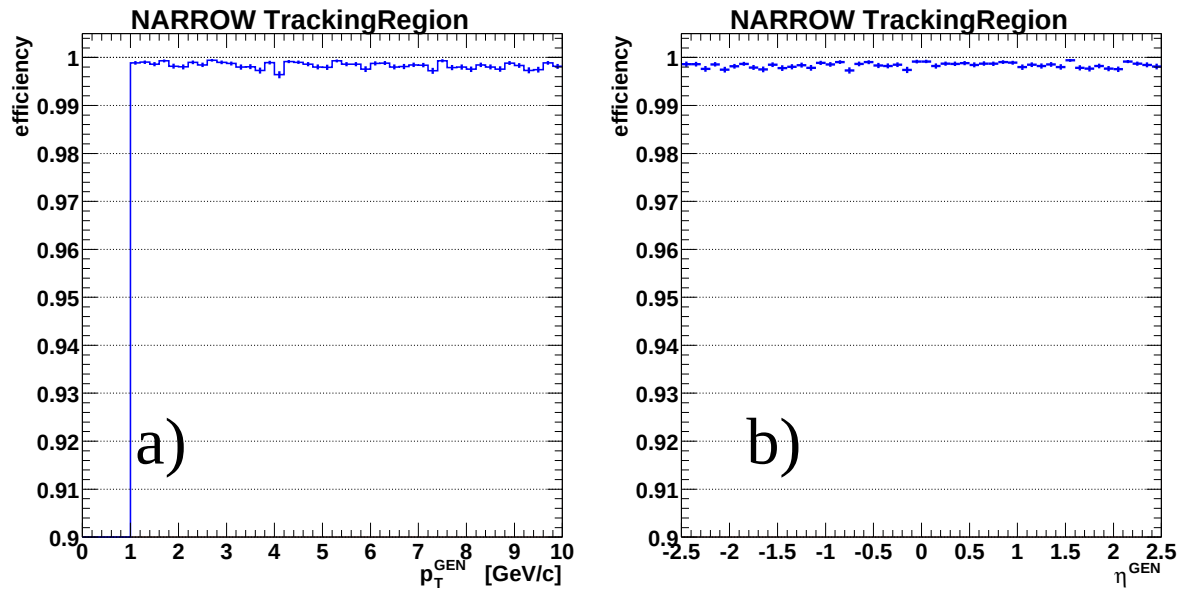


Figure 9: The algorithmic efficiency for outer hit finding for the `RectangularEtaPhiTrackingRegion` as a function of (a) transverse momentum, and (b) pseudorapidity. In order to test the correctness of the hit errors and treatment of multiple scattering, the "narrow" region described in the text was used. The plots are made for single muon events.

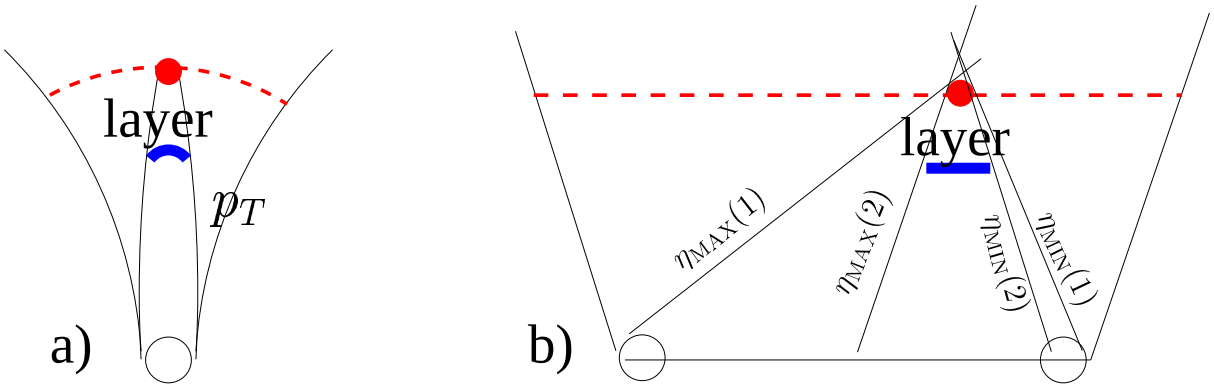


Figure 10: Illustration of the *inner* hit finding in the (a) φ, and (b) z or r coordinates. The allowed ranges for the positions of the hits are marked with bold lines. The constraints on hit position arise from the position of the outer hit and the allowed ranges of momentum and direction. The range of directions is constrained by the vertex position $(\eta_{\mathrm{MIN}}(1), \eta_{\mathrm{MAX}}(1))$, and, for the `RectangularEtaPhiTrackingRegion`, can be additionally constrained by the direction of the region $(\eta_{\mathrm{MIN}}(2), \eta_{\mathrm{MAX}}(2))$. Track prediction uncertainties are also taken into account in both cases, but are not shown on the plot.

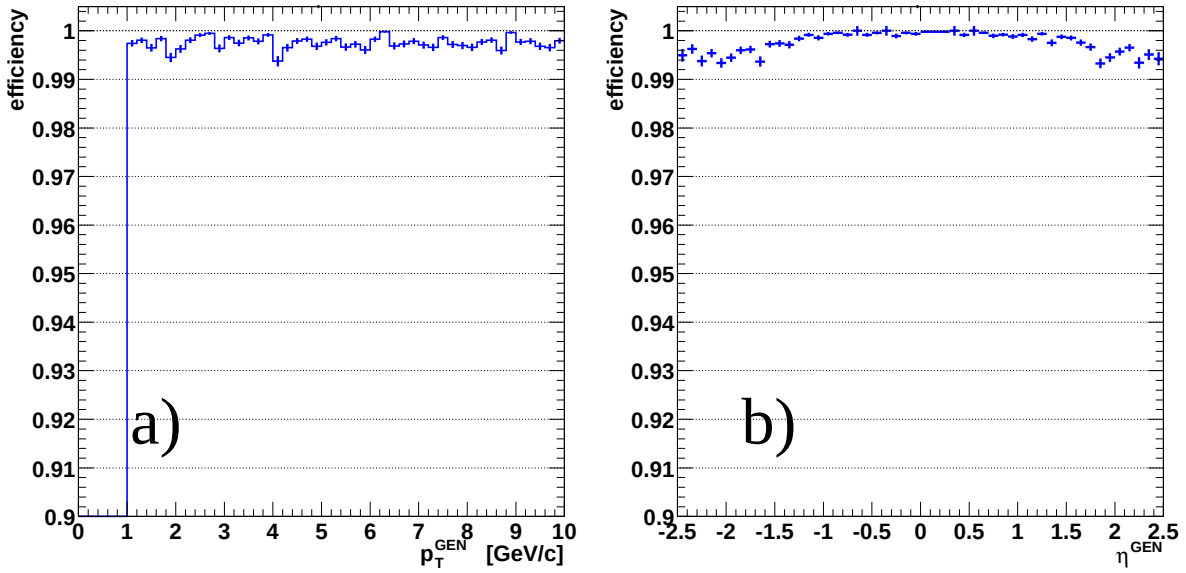


Figure 11: The algorithmic efficiency for finding hit pairs for a `RectangularEtaPhiTrackingRegion` of size $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$, as a function of (a) generated transverse momentum, and (b) pseudorapidity. The plots are made for single muon events.

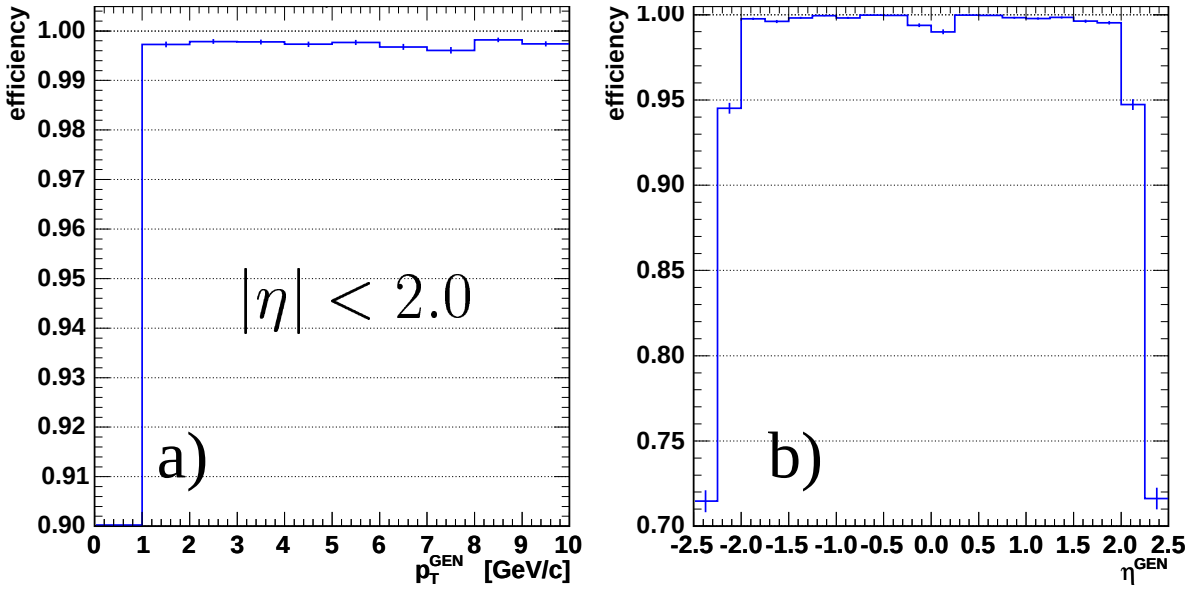


Figure 12: The absolute efficiency for finding hit pairs for a `RectangularEtaPhiTrackingRegion` of size $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$, as a function of (a) generated transverse momentum, and (b) pseudorapidity. The plots are made for single muon events.

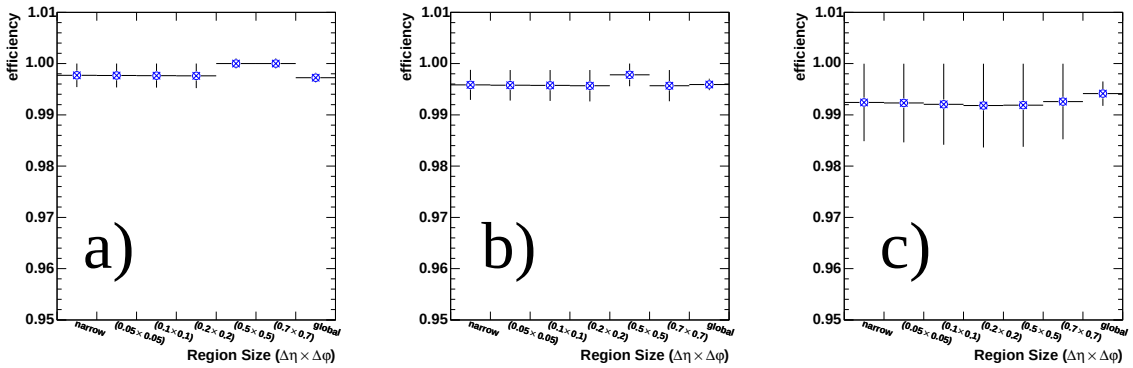


Figure 13: The luminosity dependent efficiency losses for hit pair finding. The absolute efficiency as a function of region size is shown for the same $\mathrm{h}\rightarrow\mathrm{ee}\mu\mu$ events with (a) no pile-up, (b) low-luminosity pile-up, and (c) high-luminosity pile-up. The region is centred on the most energetic muon in the event. The efficiency shown is that of *all selected* generated tracks. This selection includes cuts on pseudorapidity ($|\eta| < 2.1$) and transverse momentum ($p_T > 2.5$ GeV/c), and the momentum directions and impact points of all selected tracks must be inside the region. A pair is matched to a simulated track if both hits of the pair match those of the simulated track.

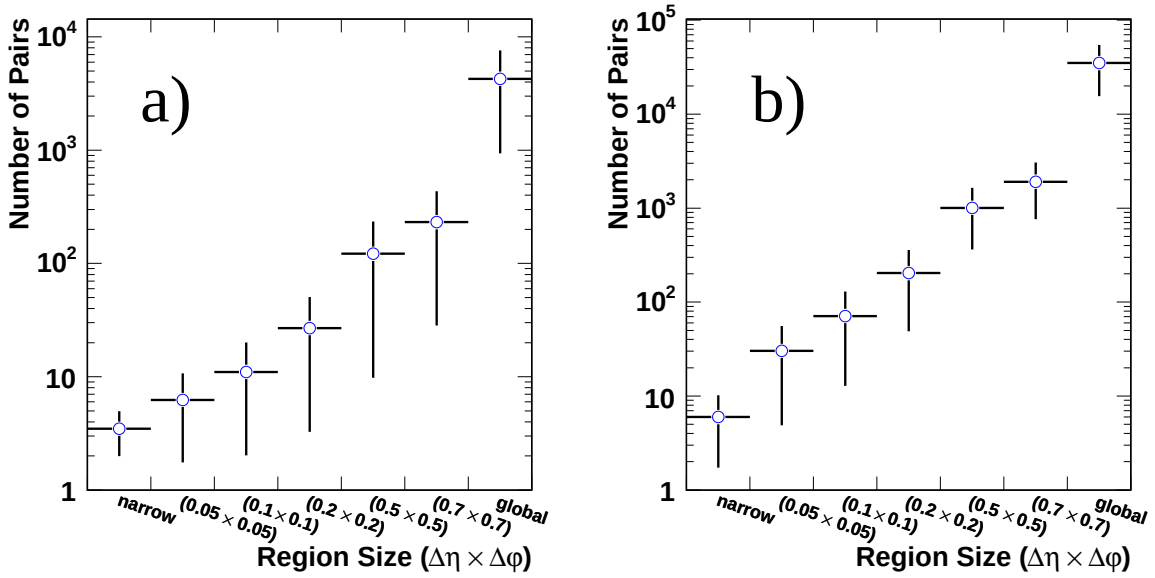


Figure 14: The number of hit pairs reconstructed as a function of region size for (a) low- and (b) high-luminosity pile-up. The plots are made for $\mathrm{h}\rightarrow\mathrm{ee}\mu\mu$ events. The vertical bars indicate the spread. The region is centred on the highest p_T muon.

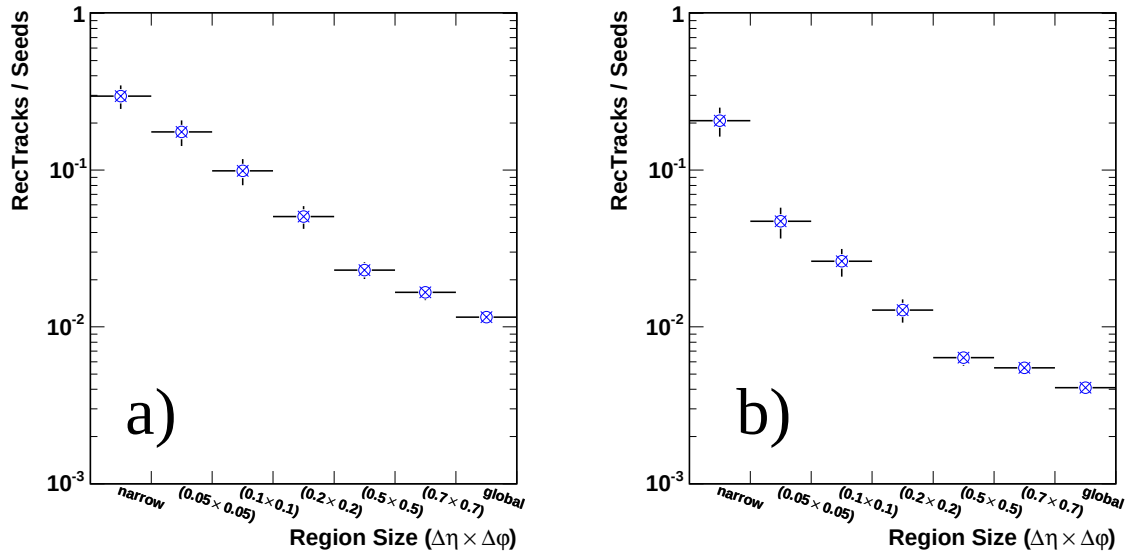


Figure 15: The purity of hit pair finding for $h \to ee\mu\mu$ events with (a) low- and (b) high-luminosity pile-up. The purity is defined as the ratio of the number of fully reconstructed tracks (in the full Tracker) to the initial number of seeds constructed from pairs of hits. Only reconstructed tracks well matching the simulated ones are selected ($\Delta R = \sqrt{\Delta\eta^2 + \Delta\varphi^2} < 0.015$ and $|\Delta z| < 0.15$ cm). The region is centred on the highest p_T muon.

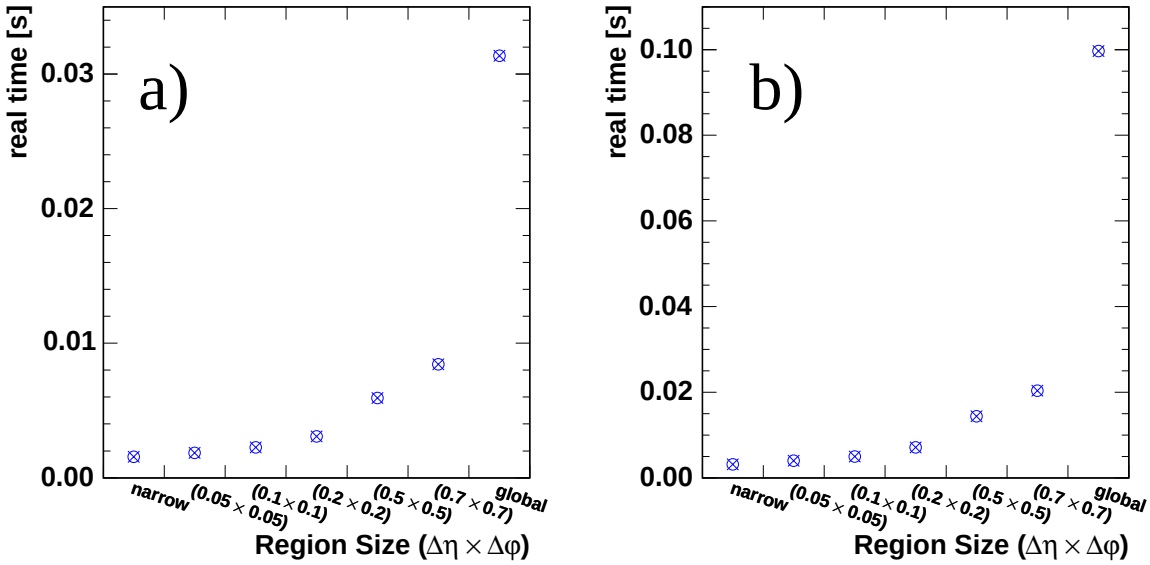


Figure 16: The real CPU time (2.8 GHz Xeon) used to reconstruct hit pairs as a function of region size. The events are $h \to ee\mu\mu$ with (a) low- and (b) high-luminosity pile-up. The region is centred on the highest p_T muon.

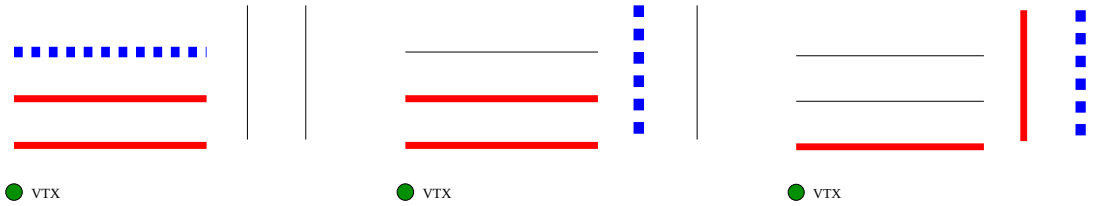


Figure 17: The combination of layers used for finding hit triplets. The pairs of bold lines indicate the layers used for hit pair generation, while the bold-dashed line shows the layer in which the search for the third hit is performed.

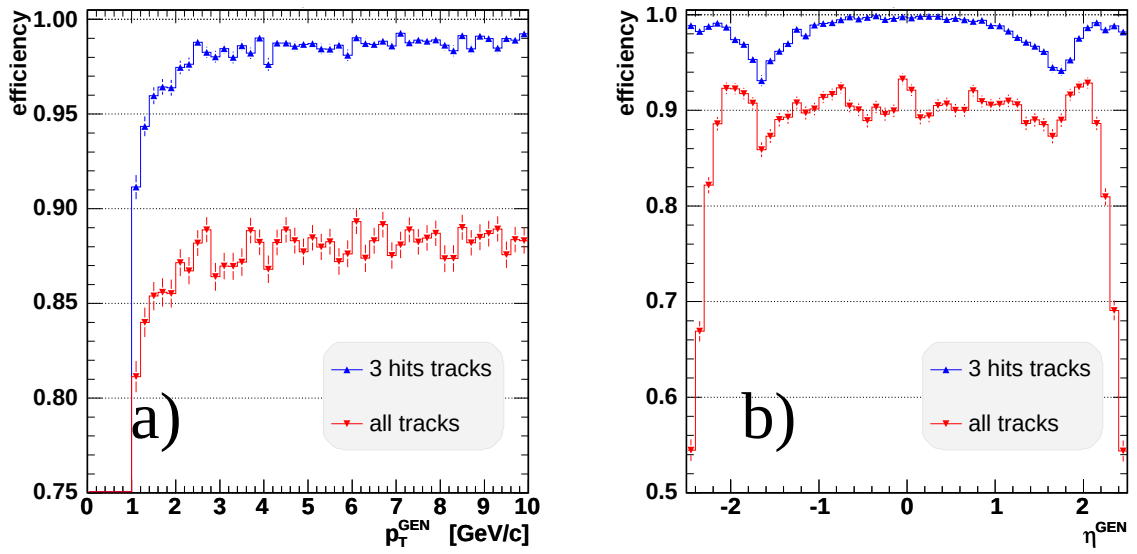


Figure 18: The efficiencies of the triplet finding. The algorithmic (3-hit tracks, upper curves) and absolute (all tracks, lower curves) efficiency is plotted as a function of (a) transverse momentum, and (b) pseudorapidity. The reconstruction is performed with a `RectangularEtaPhiTrackingRegion` of size $\Delta\eta \times \Delta\varphi = 0.2 \times 0.2$. The efficiency is measured for single muon events, and so does not take luminosity dependent data losses into account. The drop in efficiency at the barrel-endcap transition region is caused by the fact that among tracks providing three hits in different planes there are combinations of planes which are not used in the reconstruction (for example barrel layers 1 and 2 with endcap disk 2).

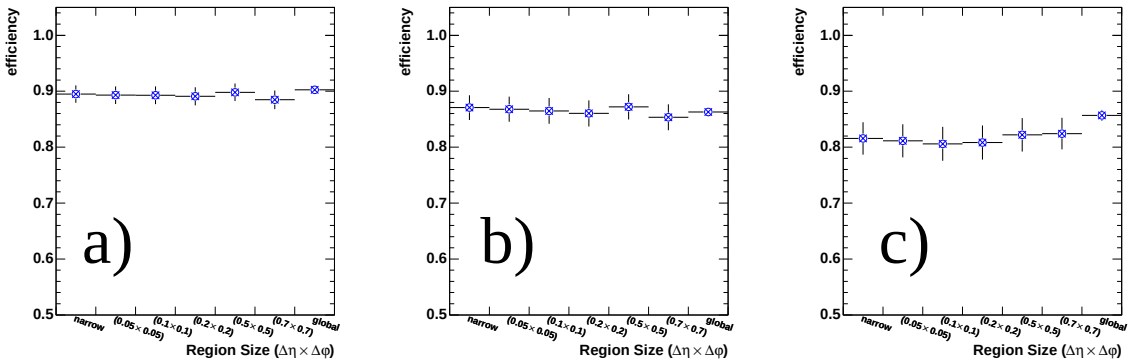


Figure 19: The luminosity dependent efficiency losses. The absolute efficiency as a function of region size is shown for the same $h \rightarrow ee\mu\mu$ events with (a) no pile-up, (b) low-luminosity pile-up, and (c) high-luminosity pile-up. The region is centred on the most energetic muon in the event. The efficiency shown is that of *all selected* generated tracks. This selection includes cuts on pseudorapidity ($|\eta| < 2.1$) and transverse momentum ($p_T > 2.5$ GeV/c), and the momentum directions and impact points of all selected tracks must be inside the region. A triplet is matched to a simulated track if all three hits match those of the simulated track.

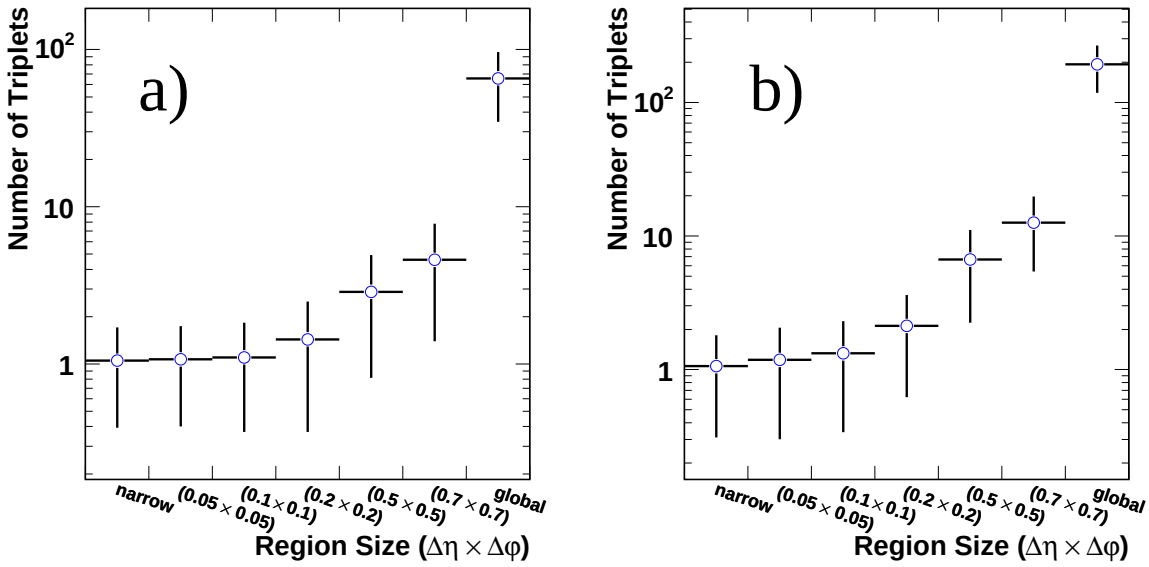


Figure 20: The number of reconstructed hit triplets as a function of region size. The plots are made for h→ee$\mu\mu$ events, with (a) low- and (b) high-luminosity pile-up. The region is centred on the highest momentum muon. The vertical bars show the spread.

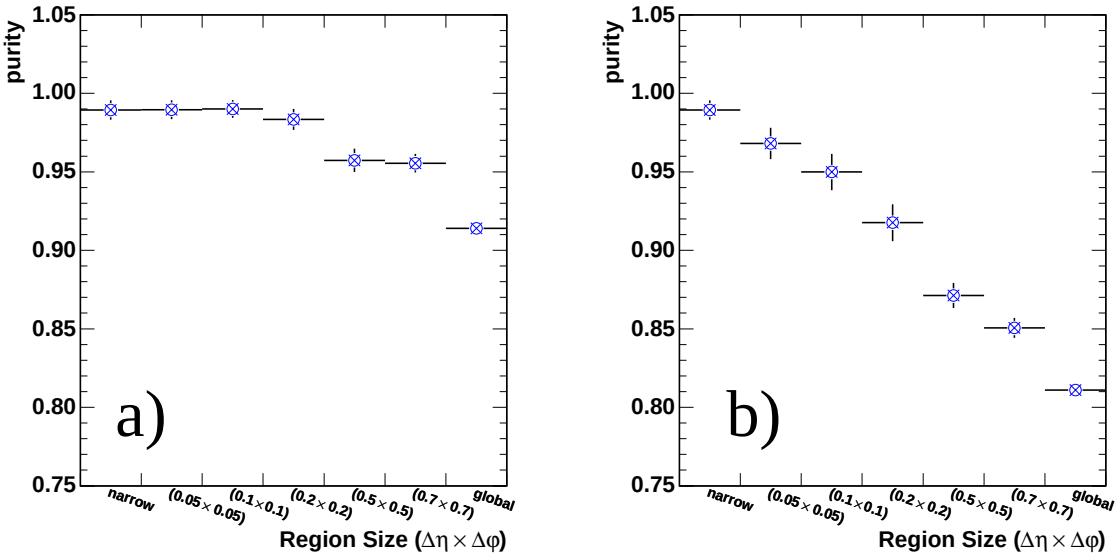


Figure 21: The triplet purity as a function of region size, for (a) low- and (b) high-luminosity pile-up. The purity is defined as the ratio of triplets associated to generated tracks (*both from the signal and the pile-up events*) to the total number of reconstructed triplets. A triplet is associated to a generated track if their directions are close ($\Delta R = \sqrt{\Delta\eta^2 + \Delta\varphi^2} < 0.025$ and $|\Delta\eta| < 0.012$). The simulated events are h→ee$\mu\mu$ and the region is centred on the highest p_T muon.

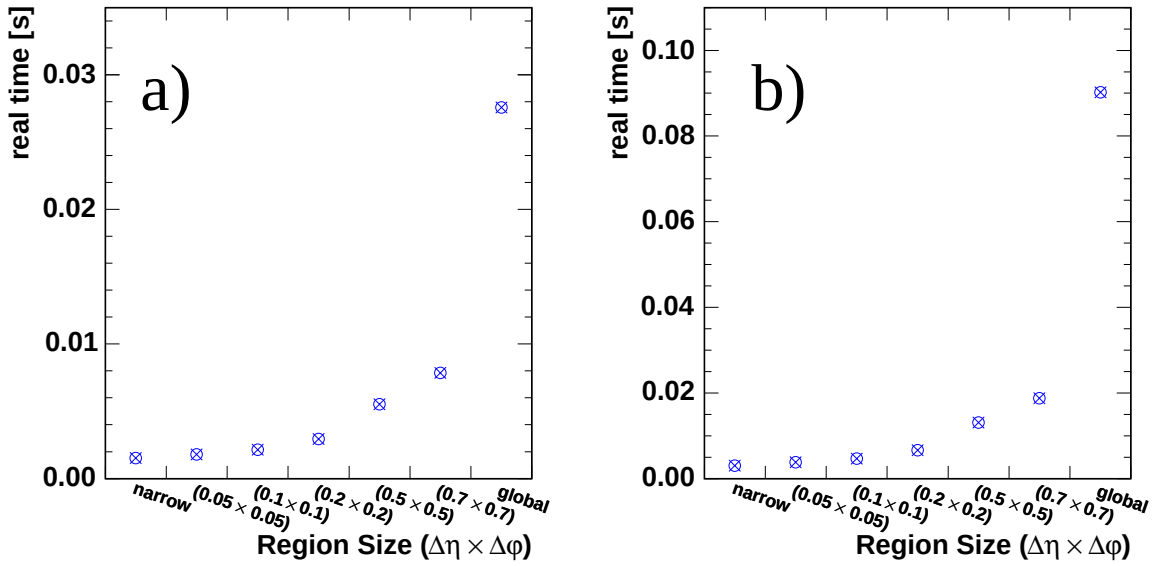


Figure 22: The real CPU time (2.8 GHz Xeon) used to reconstruct hit triplets as a function of region size. The events are h→ee$\mu\mu$ with (a) low- and (b) high-luminosity pile-up. The region is centred on the highest p_T muon.

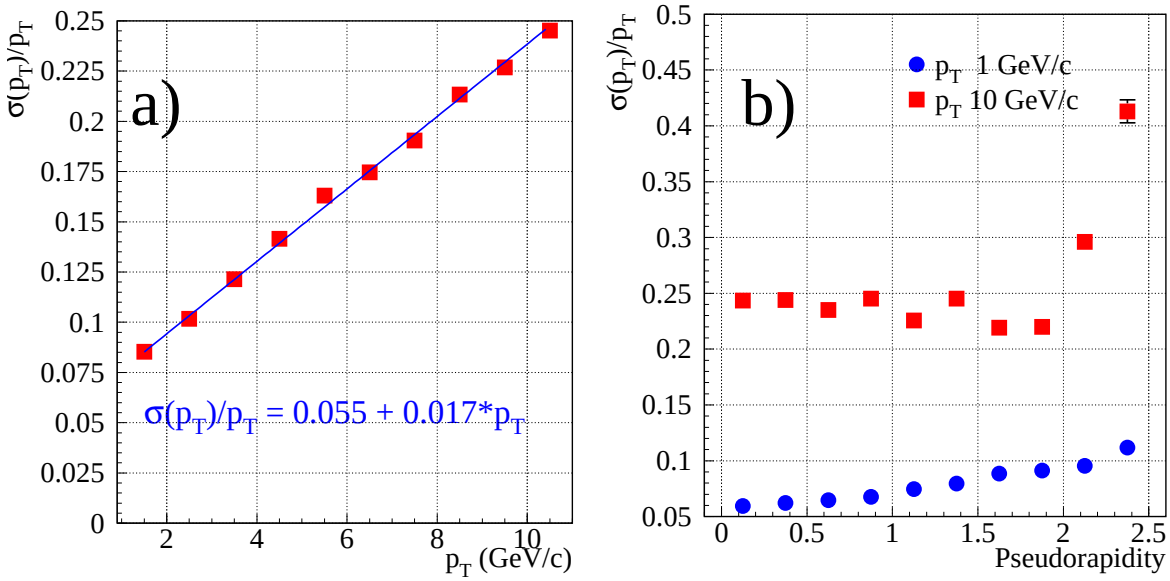


Figure 23: The behaviour of $\sigma(p_T)/p_T$ (a) as a function of p_T and (b) as a function of pseudorapidity for single muon tracks with p_T of 1 and 10 GeV/c.

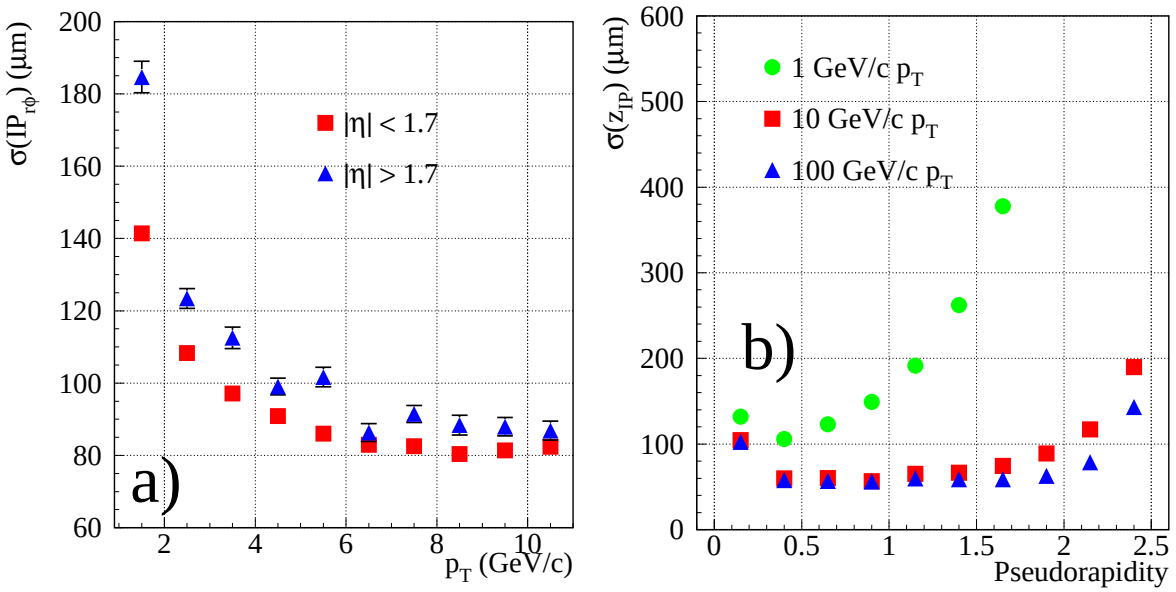


Figure 24: (a) The transverse impact parameter resolution as a function of p_T for two different pseudorapidity regions, and (b) the longitudinal impact parameter resolution as a function of η for p_T of 1, 10 and 100 GeV/c, for single muon tracks.

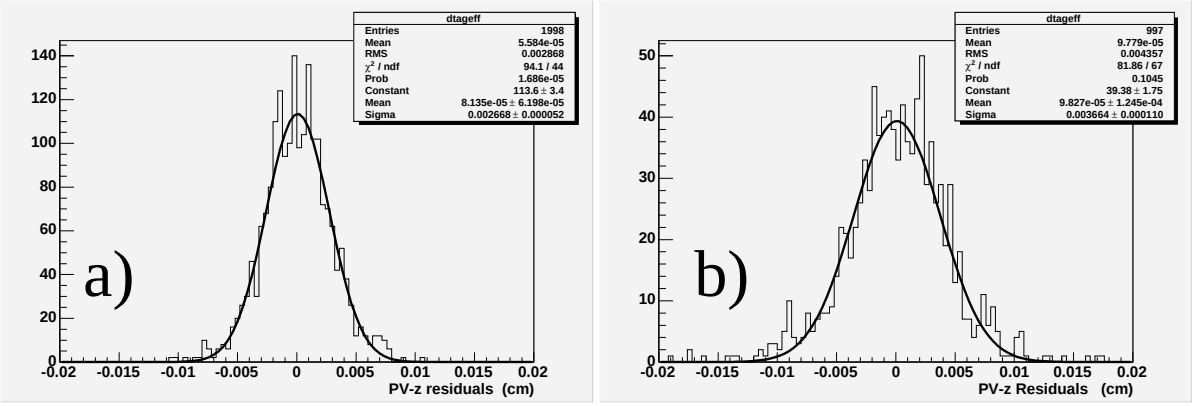


Figure 25: The pixel reconstructed primary vertex z position residuals for (a) u-jet and (b) b-jet events.