

Tracking and Vertexing performance in CMS

A. Tropiano^{*†}

Università degli Studi di Firenze

E-mail: antonio.tropiano@cern.ch

The raw data from the CMS silicon pixel and strip detectors are used to reconstruct tracks, which, in turn, are used to reconstruct vertices. Tracking and vertexing algorithms used in CMS are designed to operate in a very difficult environment, with a high number of interactions per bunch crossing and a high number of tracks originating from pp collisions. The performance of the standard CMS tracking reconstruction algorithm, the iterative tracking, is very good in terms of efficiency, fake rate and resolution for muons, electrons and pions. Muon track reconstruction is also measured with tag and probe and found to be very high and in good agreement with simulation. Tracking is also widely used at trigger level, where it helps in improving the resolution of all physics objects, from leptons to jets.

*The 21st International Workshop on Vertex Detectors
16-21 September 2012
Jeju, Korea*

^{*}Speaker.

[†]on behalf of the CMS collaboration.

1. Introduction

During these years of data taking LHC luminosity has exceeded $10^{33} \text{ cm}^{-2} \text{ s}^{-1}$, with a maximum of 30 overlapping proton-proton interactions. In these conditions the CMS tracker is crossed at each bunch crossing by about 1000 charged particles and track reconstruction in such a high occupancy environment is very challenging. First of all it is not easy to design tracking algorithms with high tracking efficiency and a low fraction of fake tracks. In addition the tracking code must run sufficiently fast that it can be used not only for offline event reconstruction but also for the CMS High Level Trigger.

The physics goals of CMS [1] place strong requirements on the performance of the tracking. Searches for high mass dilepton resonances require tracks to have good momentum resolution for transverse momenta p_T of up to 1 TeV/c. At the same time, efficient reconstruction of very soft tracks with $p_T < 1 \text{ GeV}/c$ is needed for studies of hadron production rates and to obtain optimum jet energy resolution with particle flow techniques. In addition, it must be possible to resolve very close tracks, such as those from 3-prong tau decay. Furthermore, excellent impact parameter resolution is needed for a precise measurement of the primary vertex position and for b-jet identification.

The CMS tracker was designed with these requirements in mind and the track finding algorithms are designed to fully exploit its capabilities and deliver the desired performance. The software algorithms used to achieve this are described and the performance obtained are shown, in terms of tracking efficiency, fake rate and resolution. The performance of tracking and vertexing at the trigger level are also shown and discussed.

2. The CMS silicon tracker

The core of the CMS detector is the super-conducting solenoid, that produces a magnetic field of 3.8 T. Inside the solenoid volume are placed the various subdetectors, the hadronic calorimeter, the electromagnetic calorimeter and the Silicon tracking detector for the reconstruction of charged particles. The detectors to identify and track the muons are placed outside the solenoid, embedded in the iron yoke.

The Tracker is shown in Figure 1. The inner part is composed of a Pixel Silicon detector with three barrel layers at radii between 4.4 cm and 10.2 cm and two endcap disks at each end. The single pixel size is $100 \times 150 \mu\text{m}^2$ for a total of 66 M channels.

The Silicon Strip Tracker covers the radial range between 20cm and 110cm around the interaction point. The barrel region ($|z| < 110 \text{ cm}$) is split into a Tracker Inner Barrel (TIB) made of four detector layers, and a Tracker Outer Barrel (TOB) made of six detector layers. The TIB is complemented by three Tracker Inner Disks per side (TID). The forward and backward regions ($120 \text{ cm} < |z| < 280 \text{ cm}$) are covered by nine Tracker End-Cap (TEC) disks per side covering the region of $|\eta|$ up to 2.5. In some of the layers and in the innermost rings special double-sided modules are able to provide accurate three-dimensional position measurement of the charged particle hits. The Silicon Strip Tracker is made of modules with different strip pitches ranging from 80 to 180 μm for about 9.6 M channels.

The basic performance of the tracking detector is a transverse momentum resolution $\sigma(p_T)/p_T$ around 1-2% for muons of p_T around 100 GeV, an impact parameter resolution of 10-20 μm for

tracks with $p_T = 10\text{-}20$ GeV and the ability to reconstruct tracks in jets with an efficiency of about 0.85-0.90 and a few percent fake rate.

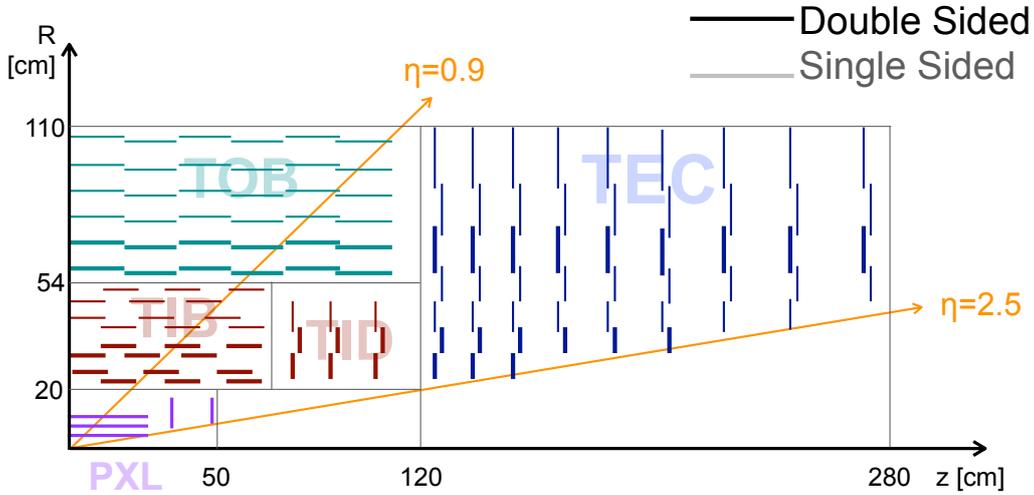


Figure 1: A simplified view of the Rz section of the CMS Tracker (bold lines represent double sided modules).

3. Track and vertex reconstruction

The CMS tracking software is known as the Combinatorial Track Finder (CTF). The collection of reconstructed tracks is produced by multiple iterations of the CTF track reconstruction sequence, in a process called iterative tracking. In the early iterations tracks with relatively high p_T , produced near the interaction region, are reconstructed. After each iteration, hits associated with tracks already found are removed, reducing the combinatorial complexity and thus allowing later iterations to search for lower p_T or highly displaced tracks. For 2011 data reconstruction, the iterative tracking consisted of 6 iterations. Iteration 0 is the source of most tracks and is designed to reconstruct prompt tracks with $p_T > 0.8$ GeV and which have three pixel hits. Iteration 1 is used to recover prompt tracks which only have two pixel hits or slightly lower p_T . Iteration 2 is configured to find low p_T prompt tracks. Iterations 3 to 5 are intended to find tracks which originate outside the beamspot and to recover tracks not found by the previous iterations. At the beginning of each iteration, hits associated with high purity tracks found in previous iterations are discarded.

Each iteration proceeds in four steps:

- The seed generation provides initial track candidates found using only a few (2 or 3) hits. A seed defines the initial estimate of the trajectory parameters and their uncertainties.
- The track finding is based on a global Kalman filter [3]. It extrapolates the seed trajectories along the expected flight path of a charged particle, searching for additional hits that can be assigned to the track candidate.
- The track fitting is used to provide the best possible estimate of the parameters of each trajectory by means of a Kalman filter and smoother.

- The track selection sets quality flags and discards tracks that fail certain criteria.

The main differences between the 6 iterations lie in the configuration of the seed generation and final track selection steps.

3.1 Seed generation

The trajectory seeds define the starting trajectory parameters and associated uncertainties of potential tracks. In the magnetic field present in the tracker, charged particles follow helices and therefore five parameters (including the trajectory curvature) are needed to define a starting trajectory. To obtain these five parameters requires at least 3 hits, or 2 hits and a beam constraint. To limit the number of hit combinations, seeds are required to satisfy loose criteria, for example on their minimum transverse momentum and consistency with originating from the proton-proton interaction region.

Seeds are built in the inner part of the tracker and the track candidates are reconstructed outwards. The reason for this approach is that, although the track density is much higher in the inner region of the tracker, the high granularity of the pixel detector ensures that the average occupancy of the inner pixel layer is much lower than the average occupancy of the outer strip layer. Secondly, the pixel layers provide three-dimensional space-point measurements which allow for more constraints and better parameter estimation.

3.2 Track finding

The track finding module of the CTF algorithm is based on the Kalman filter method. The filter begins with a coarse estimate of the track parameters provided by the trajectory seed and then builds track candidates by adding hits from successive layers one by one. The information provided at each layer includes the location and uncertainty of any found hit as well as the amount of material crossed, which is used to estimate the uncertainty arising from multiple Coulomb scattering. The Kalman filter method is implemented in four steps.

The first step, navigation, uses the parameters of the track candidate, evaluated at the current layer, to determine which adjacent layer(s) of the tracking detector are intersected by the extrapolated trajectory.

The second step is a search for compatible detectors in the layers returned by the navigation step. A detector is considered compatible with the trajectory if the position at which the latter intercepts the detector surface is no more than a given number (currently three) of standard deviations outside the detector boundary.

The third step forms groups of hits, obtained by collecting all the hits compatible with a certain trajectory. A configurable parameter also allows the addition of an invalid hit to represent the possibility that the particle failed to produce a hit in the detector group, for example, due to detector inefficiency. A χ^2 test is used to check which of the hits are compatible with the extrapolated track trajectory.

The fourth and last step is to update the trajectory state. New track candidates are formed from each of the original ones, by adding to them exactly one of the compatible hits from each detector grouping (where this hit may be an invalid hit). The candidate trajectory parameters are

then updated at the new detector surface, by combining the information from the hit with the extrapolated track trajectory of the original candidate.

3.3 Track fitting

The aim of the Track fitting stage is to obtain a precise estimate of the track parameters, given the set of hits obtained in the previous stage. For each trajectory, the track finding stage results in a collection of hits and also an estimate of the track parameters. However, the full information about the trajectory is only available at the last hit of the trajectory and the estimate can be biased by constraints applied during the seeding stage. Therefore the trajectory is refitted using a Kalman filter and smoother.

The Kalman filter is initialized at the location of the innermost hit with the trajectory estimate obtained during seeding. The fit then proceeds in an iterative way through the full list of hits, updating the track trajectory estimate sequentially with each hit. For each valid hit, the hit position estimate is re-evaluated using the current values of the track parameters. This first filter is complemented by the smoothing stage: a second filter is initialized with the result of the first one and is run backward toward the beam line. The track parameters at the surface associated with any of its hits can then be obtained from the average of the track parameters of these two filters, evaluated on this same surface.

4. Tracking and vertexing performance

The performance of the CTF tracking algorithm has been evaluated on isolated particles (a single muon, pion or electron) and also on simulated $t\bar{t}$ events. Simulated particles and reconstructed tracks are associated for evaluating the tracking efficiency, fake rate and track parameter resolutions. A simulated track is associated to a reconstructed one if at least 75% of the hits assigned to the reconstructed track were produced by the simulated particle. The tracking efficiency is also measured using a data driven technique.

The tracking efficiency is defined as the fraction of simulated charged particles that can be associated with a reconstructed track. It depends not only on the quality of the track finding algorithm, but also on the intrinsic properties of the tracker, such as its geometrical acceptance and material budget. The fake rate is defined as the fraction of reconstructed tracks that are not associated with any simulated particle. This quantity represents the probability that a track produced by the reconstruction algorithm is either a combination of unrelated hits or a genuine trajectory that is badly reconstructed by including a large number of spurious hits.

Muons are reconstructed better than any other charged particle in the tracker. They mainly interact with the silicon detector through ionization and, unlike electrons, their energy loss by bremsstrahlung is generally negligible, except when the muon p_T exceeds about 100 GeV. Therefore, muons usually cross the whole volume of the tracking system, producing detectable hits on several sensitive layers of the apparatus. Finally, muon trajectories are altered exclusively by Coulomb scattering, whose effects are straightforward to include inside the Kalman filter formalism. For isolated muons with a transverse momentum between 1 GeV and 100 GeV, the tracking efficiency is higher than 99% over the full η range of the tracker acceptance and does not depend on the transverse momentum (Fig. 2 left), while the fake rate is completely negligible. Charged pions

undergo multiple scattering and energy loss by ionization as they cross the tracker volume. Like all hadrons, pions are also subject to elastic and inelastic nuclear interactions that introduce long tails in the distribution of the scattering angle. Inelastic nuclear interactions are the main source of tracking inefficiency for hadrons, in particular in those regions of the tracker where the material budget is large. Depending on the pseudo-rapidity, up to 20% of the simulated particles are not reconstructed, despite leaving hits in three or more layers of the tracking system. This effect is more significant for low energy hadrons due to their higher cross section for nuclear interactions. The tracking efficiency and fake rate are also affected by the secondary particles produced in inelastic processes.

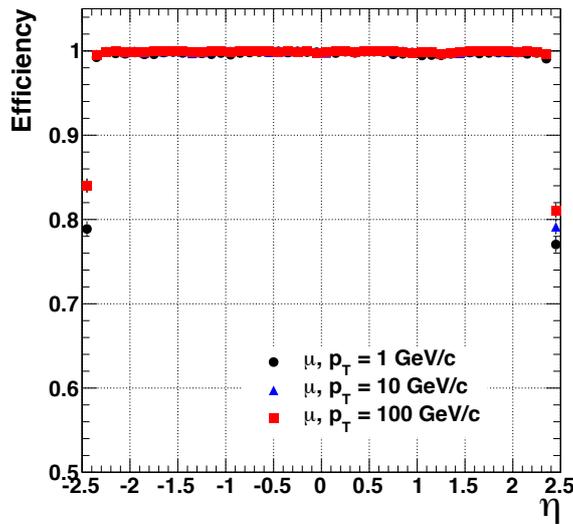


Figure 2: Track reconstruction efficiency for single, isolated muons and pions. Results are shown as a function of η , for transverse momenta of 1, 10 and 100 GeV.

The goal of the primary vertex reconstruction is to measure the location, and the associated uncertainties, of all proton-proton interaction vertices in each event using the available reconstructed tracks. It consists of three steps: selection of the tracks to be used, clustering of the tracks, meaning deciding which ones originate from the same interaction vertex, and fitting the position of each vertex using its associated tracks.

Track selection, which aims to select tracks produced in the primary interaction region, imposes requirements on the maximum allowed transverse impact parameter significance with respect to the beamspot, its number of strip and pixel hits, and its normalized χ^2 . To ensure high reconstruction efficiency, even in minimum bias events, there is no requirement on the minimum allowed track p_T . The selected tracks are then clustered, based on their z coordinates at the point of closest approach to the beamspot. This clustering allows for the possibility of multiple primary interactions in the same LHC bunch crossing. The clustering algorithm must balance the efficiency for resolving nearby vertices in cases of high pileup against the possibility of accidentally splitting a single, genuine interaction vertex into more than one cluster of tracks.

The track clustering is performed with a deterministic annealing (DA) algorithm [2]. Annealing finds the global minimum in a problem with many degrees of freedom analogous to the

way a physical system reaches the state of minimal energy through a series of gradual temperature reductions.

The z -coordinates of the points of closest approach of the tracks to the beamspot are referred to as z_i^T , and their associated uncertainties as σ_i . The tracks must be assigned to some unknown number of vertices at positions z_k^V . Hard assignment can be represented by values p_{ik} that equal 1 if track i is assigned to vertex k and 0 otherwise. In the DA framework, assignments are soft. This means that p_{ik} ranges between 0 and 1 and can be interpreted as the assignment probability of track i to vertex k in a large ensemble of possible assignments. The assignment probabilities are given by:

$$p_{ik} = \frac{\rho_k \exp \left[\frac{-1}{T} \frac{(z_i^T - z_k^V)^2}{\sigma_i^2} \right]}{\sum_{k'} \rho_{k'} \exp \left[\frac{-1}{T} \frac{(z_i^T - z_{k'}^V)^2}{\sigma_i^2} \right]}$$

where ρ_k represents the number of vertices coinciding at position k and the temperature T controls the softness of the assignment and the resolution, as it effectively scales the σ_i by \sqrt{T} . At very high T all p_{ik} are equal and all tracks are compatible with a single vertex. Hard assignment is approached for very low T and for $T \rightarrow 0$ every track is compatible with exactly one vertex. The DA algorithm starts at a very high temperature with a single vertex prototype, then gradually decreases T . The starting point is a temperature above the first critical temperature, evaluated with $\rho_k = p_{ik} = 1$. The temperature is decreased at every step by the cooling factor 0.6. The annealing is continued down to a minimal temperature $T_{min} = 4$, which represents a compromise between the resolving power and the probability of splitting real vertices.

The primary vertex resolution along the x (transverse to the beamline) and z (along the beamline) axis is shown in Fig. 3 as a function of the number of tracks.

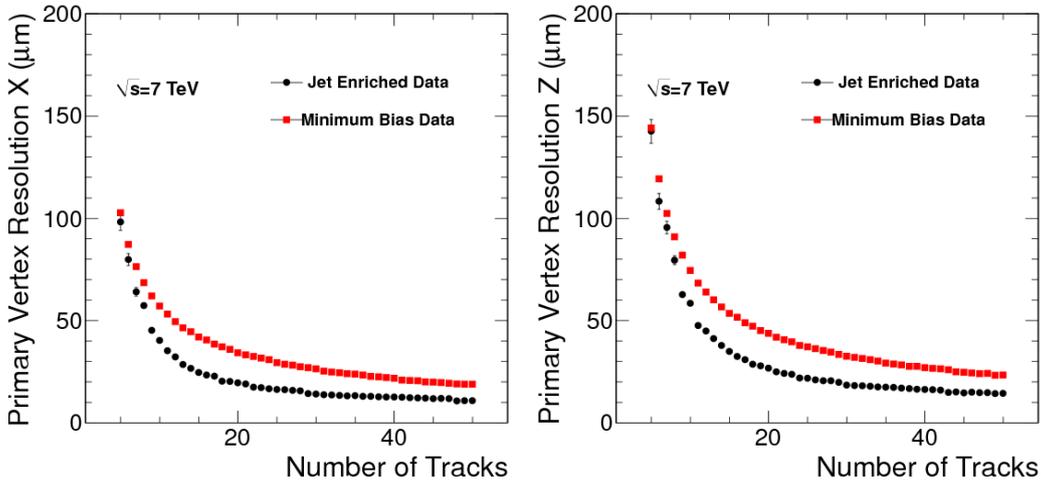


Figure 3: Primary vertex resolution in x (left) and z (right) as a function of the number of tracks, for two different kinds of events with tracks of different average transverse momentum.

4.1 Efficiency estimation from data

The tag-and-probe method allows to extract the muon tracking efficiency directly from data using resonances.

$Z \rightarrow \mu^+ \mu^-$ candidates are reconstructed using pairs of oppositely charged particles, identified as muons using the muon chambers. Each candidate must consist of one tag muon, that have to be reconstructed in both tracker and muon chambers, and one probe muon, reconstructed in the muon chambers, with no requirement to be reconstructed in the tracker. The invariant mass of each candidate is required to be within the [50,130] GeV range. The tracking efficiency can be estimated from the fraction of the probe muons that can be associated to a reconstructed track in the tracker. A correction must be made for the fact that some of the probe muons are not genuine. This correction is obtained by fitting the dilepton mass spectrum in order to subtract the non-resonant background, (since only genuine dimuons will contribute to the resonance). This must be done separately for candidates in which the probe is or is not associated to a track.

The results of the tag and probe fit are shown in Fig. 4 as a function of the probe η and number of reconstructed primary vertices in the event. The measured tracking efficiency for muons from Z decay is well over 99% in both data and simulation. The data displays a small drop in tracking efficiency with increasing pileup, which is not reproduced by the simulation. This may be explained by the dynamic (pileup dependent) inefficiency of the pixel detector, not modeled in the simulation.

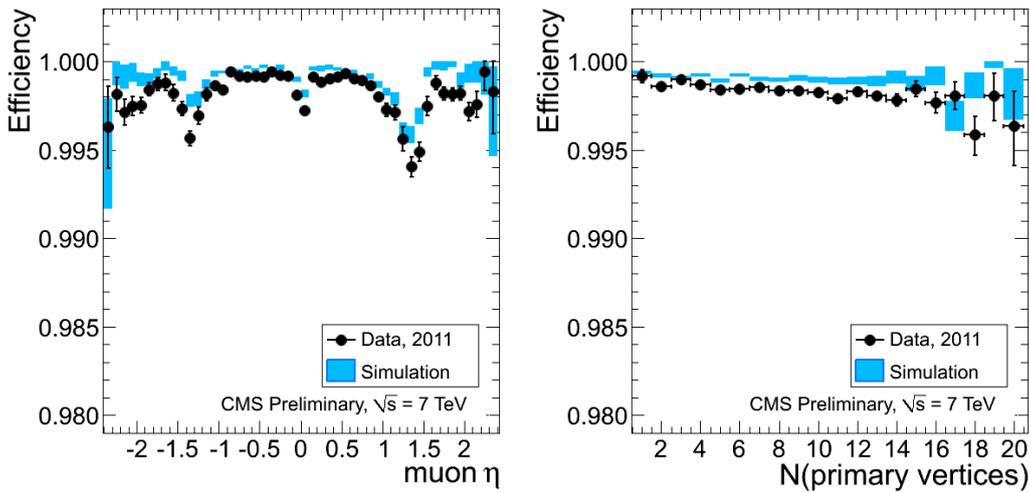


Figure 4: Tracking efficiency measured with a tag-and-probe technique, for muons from Z decay, as a function of the muon η (left) and the number of reconstructed primary vertices in the event (right) for data (black) and simulation (blue). The decrease of efficiency in the region $1 < |\eta| < 1.5$ is due to dead material in that region of the tracker, while the small efficiency drop around 0 reflects dead channels in that region of the pixel detector.

5. Tracking at HLT

The CMS high-level trigger (HLT) uses a processor farm to achieve large reductions in the data rate. The HLT runs on events selected by the Level 1 (hardware) trigger, at rates of up to 100 kHz. Track reconstruction is a powerful tool in the HLT for the following reasons.

Demanding that muon/electron candidates reconstructed in the CMS muon chambers or electromagnetic calorimeter should be confirmed by the existence of a corresponding track in the

tracker, greatly reduces the fake rate. The background rejection rate of the lepton triggers can be enhanced further by requiring that leptons should be isolated. One method of doing this is to put a veto on the presence of (too many) tracks in a cone around the lepton. It is possible to trigger on jets produced by b-quarks, by counting the numbers of tracks in a jet, which have a transverse impact parameter that is statistically incompatible with the track originating from the beam-line. The HLT uses identical track reconstruction software to that used for offline event reconstruction. However, it must run much faster. This is achieved by modifying various configuration parameters that control the tracking algorithms. Tracks can be reconstructed from triplets of hits found using only the pixel tracker. This is extremely fast. One vital role for which these pixel tracks are required is the reconstruction of the primary vertex position in the HLT.

Tracks can also be reconstructed in the HLT using hits from both pixel and strip trackers. Such tracks have superior momentum resolution and a lower probability of being fake. However, they take much more CPU time to reconstruct than the pixel tracks, since the strip tracker does not provide the precise, 3-D hit measurements of the pixel tracker and suffers from a higher hit occupancy. The pixel tracking and other track reconstruction uses about 20% of the total HLT CPU time. This is kept low by only performing track reconstruction when necessary and only after other requirements have been satisfied to reduce the rate at which tracking must be done.

Track reconstruction is used for example to improve the resolution of jets, by performing a customized particle flow reconstruction for the HLT. In these cases the rate at which tracking is performed is kept low by selecting events with high p_T calorimetric jets. Tracking is also performed to select events coming from b -jets. For these events tracking is run only in a region around the jet of interest, to reduce the timing. The b -tagging algorithm is then applied firstly on pixel tracks, with loose requirements. If these requirements are met the full track reconstruction is performed. This allows to reduce the rate at which tracking is run in these cases.

In the reconstruction of the LHC data at trigger level the most used vertex reconstruction algorithm is a simple gap clustering algorithm. All tracks are ordered by the z coordinate of their point of closest approach to the beamspot. Then, wherever two neighboring elements in this ordered set of z coordinates had a gap between them exceeding a distance cut z_{sep} , this point is used to split the tracks on either side of it into separate vertices. In such an algorithm, interaction vertices separated by a distance less than z_{sep} are merged, thus this algorithm is not an optimal choice for high pileup LHC running. However in 2012 data taking, where the number of interactions per bunch crossing reached 30, the number of reconstructed vertices still shows a linear dependence on the number of interactions (Fig. 5).

6. Conclusions

CMS has developed sophisticated tracking software, based on Kalman filter techniques. It is able to reconstruct tracks over the full rapidity range of the tracker and is sensitive to charged particles with p_T as low as 100 MeV/c and also to those produced as much as 60 cm from the beam line. For promptly produced charged particles the average tracking efficiency typically 95 (85)% in the tracker barrel (endcap), with most of the efficiency loss being caused by hadrons undergoing nuclear interactions in the tracker material. The fraction of reconstructed tracks which is not genuine is at the few percent level. For high p_T tracks (100 GeV), for which multiple

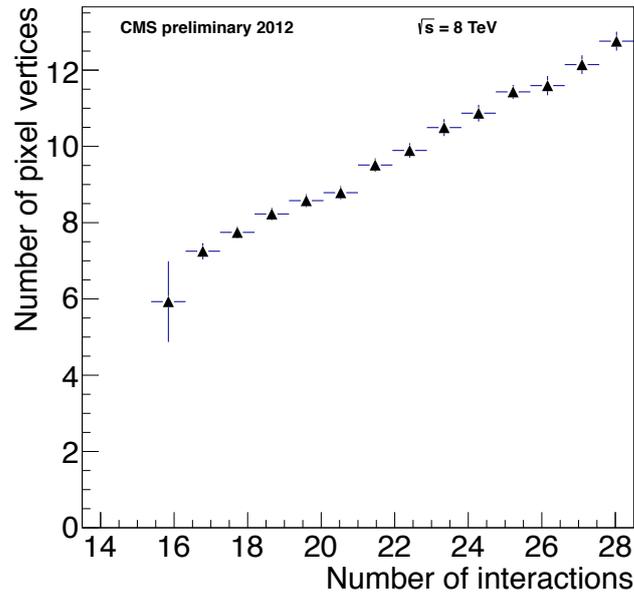


Figure 5: Number of vertices as a function of the pile up, for a single fill.

scattering is of little importance, the resolution in the central region ($|\eta| < 1$) is approximately 2% in p_T and $10 \mu\text{m}$ ($35 \mu\text{m}$) in transverse (longitudinal) impact parameter.

The tracks are also used to reconstruct the primary interaction vertices in each event. For the multiparticle vertices typical of interesting physics events, the vertex resolution achieved is 10-12 μm in all three dimensions.

References

- [1] CMS Collaboration, J. Inst. **3** (2008) 361.
- [2] K. Rose, Proceedings of the IEEE **86** (1998), 11.
- [3] R. Fruhwirth, Nucl.Instrum.Meth. **A262** (1987) 444D450.