PROCEEDINGS OF SCIENCE

# Tracking at High Level Trigger in CMS

**Mia TOSI**[*] [†]
*Universitá degli Studi di Padova e INFN (IT)*
*E-mail:* mia.tosi@gmail.com

The trigger systems of the LHC detectors play a crucial role in determining the physics capabilities of the experiments. A reduction of several orders of magnitude of the event rate is needed to reach values compatible with detector readout, offline storage and analysis capability.
The CMS experiment has been designed with a two-level trigger system: the Level-1 Trigger (L1T), implemented on custom-designed electronics, and the High Level Trigger (HLT), a streamlined version of the CMS offline reconstruction software running on a computer farm. A software trigger system requires a trade-off between the complexity of the algorithms, the sustainable output rate, and the selection efficiency. With the computing power available during the 2012 data taking the maximum reconstruction time at HLT was about 200 ms per event, at the nominal L1T rate of 100 kHz.
Track reconstruction algorithms are widely used in the HLT, for the reconstruction of the physics objects as well as in the identification of b-jets and lepton isolation. Reconstructed tracks are also used to distinguish the primary vertex, which identifies the hard interaction process, from the pileup ones. This task is particularly important in the LHC environment given the large number of interactions per bunch crossing: on average 25 in 2012, and expected to be around 40 in Run II. We will present the performance of the HLT tracking algorithms, discussing its impact on CMS physics program, as well as new developments done towards the next data taking in 2015.

*Technology and Instrumentation in Particle Physics 2014,*
*2-6 June, 2014*
*Amsterdam, the Netherlands*

[*]Speaker.
[†]on behalf of the CMS collaboration

 

PoS(TIPP2014)204

## 1. Introduction

The Compact Muon Solenoid, CMS, is one of the two general-purpose experiments installed at the Large Hadron Collider (LHC) at CERN [1]. The collision rate at the LHC is heavily dominated by large cross section QCD processes, which are not interesting for the physics program. The processes we are interested in usually occur at a rate of the order of 10 Hz. Since it is not possible to register all the events and to select them later on, it becomes mandatory to use a trigger system in order to select events according to physics-driven choices.

CMS has a wide physics program in RunII, from the re-discovery of the Standard Model at 13 TeV to the search of possible new physics as well as precision measurements of rare processes which characterize the physics of the b-quarks. The main goal of the CMS trigger system is to keep the largest as possible number of events for analysis while keeping the event rate within the system limitation, namely 500 Hz. In order to reduce the difference between online and analysis selection cuts, the online reconstruction and calibrations have to match even better the offline and analysis objects. One of the key ingredients is to make a wider use of the tracking and particle-flow based techniques. The CMS High Level Trigger (HLT) [2] uses a processor farm running C++ software to achieve large reductions in data rate. The HLT filters events selected at rates of up to 100 kHz using the Level-1 (hardware) trigger. Whereas Level-1 uses information only from the CMS calorimeters and muon detector, the HLT is also able to capture information from the tracker, thereby adding the powerful tool of track reconstruction to the HLT.

In 2015, data taking operations are expected to re-start at a centre-of-mass energy of 13 TeV with an instantaneous luminosity which should reach the peak value of $1.4\times10^{34}\text{cm}^{-2}\text{s}^{-1}$. In such conditions we are expecting an increase of event rate of about a factor of 4 with respect to the last period of data taking in 2012, regardless the applied trigger selections. Moreover, the expected average number of overlapping proton-proton interactions will be around 40. In these conditions the CMS tracker is crossed by thousands of charged particles in each bunch crossing. In such a high occupancy environment design tracking algorithms with high tracking efficiency and a low fraction of fake tracks is very challenging. In addition the tracking code must run sufficiently fast that it can be used at the HLT.

The HLT uses track reconstruction software that is identical to that used for offline reconstruction [9], but it has to fulfill the CPU timing constraint: the event selection has to be done in about 200 ms. Because the track reconstruction is a sophisticated and complex algorithm and it is one of the most time consuming step, the HLT approach is to apply the track reconstruction only at the end of the event selection, generally after having applying other requirements based on the fast reconstruction of the physics object with only information from the calorimeter and muon system. Moreover, again for optimizing the CPU timing of the online event selection, the track reconstruction is done only within regions-of-interest, defined by the direction of the already available physics object. In order to guarantee the best physics performance, CMS applies a specialized track and vertex reconstruction for each specific case.

## 2. The tracking system

The silicon tracking system, shown in Figure 1, is immersed in a magnetic field of 3.8T and is composed of a pixel silicon detector and a silicon micro-strip one. The pixel detector consist of three barrel layers at radii between 4.4 cm and 10.2 cm and two endcap disks at each side and

sensors feature single pixel size of 100×150 $\mu m^2$ for a total of 66M channels. The strip tracker covers the radial range between 20 cm and 110 cm around the LHC interaction point. The barrel region ($|z| < 110$cm) is split into a Tracker Inner Barrel, made of four detector layers, and a Tracker Outer Barrel, made of six detector layers. The TIB is complemented by three Tracker Inner Disks per side. The forward and backward regions (120 cm < $|z|$ < 280 cm) are covered by nine Tracker End-Cap disks per side, thus extending the overall acceptance to cover the region $|\eta| < 2.5$. 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 strip tracker is instrumented by about 15,000 modules with different strip pitches ranging from 80 to 180 $\mu$ m, for a total 9.6 million channels[1][3][4].

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$\mu$m for tracks with $p_T$ = 10-20 GeV and the reconstruction of tracks belonging to a jet has an efficiency of about 85-90% and a few percent fake rate.

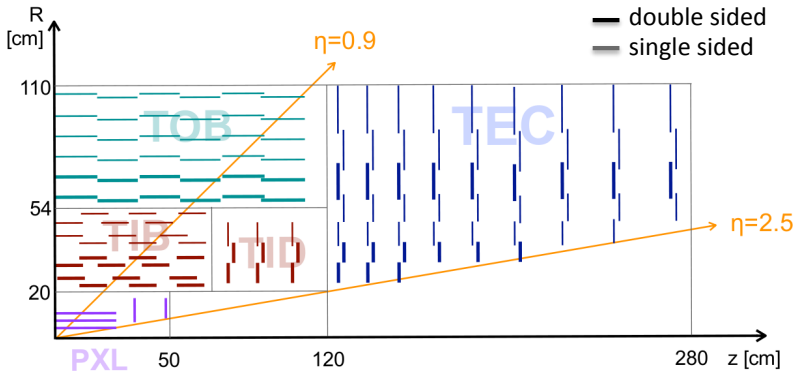


**Figure 1:** A simplified sketch of a quadrant of the Rz section of the CMS Tracker (bold lines represent double sided module assemblies).

Because the silicon strip unpacking is known to take a long time and to have a strong dependence on strip occupancy, regional and *on-demand* unpacking is performed at HLT, in which only modules requested during the pattern recognition are actually unpacked. The strip modules are grouped into geometrical regions, defined by a grid on the $\eta$-$\phi$ plane (where $\eta$ is the pseudorapidity and $\phi$ the azimuthal angle) with configurable dimensions (typically 0.5×0.5), and raw data is considered only from regions-of-interest. More specially, any raw data packet with at least one channel connected to a region-of-interest is fully unpacked. These regions-of-interest can be defined by physics objects identified in external sub-detectors (as muons, electrons, jets and taus). This is not used in the offline reconstruction as the track reconstruction searches the entire $\eta$-$\phi$ region and therefore needs all hits.

## 3. Track reconstruction at HLT

Tracks can be reconstructed from triplets of hits found using only the pixel tracker. This is extremely fast, and can be used with great effect in the reconstruction of the primary-vertex position in the HLT. Tracks can also be reconstructed in the HLT using hits from both the pixel and strip detectors. In CMS the tracks are reconstructed in four steps:

- the seed generation provides initial track candidates (see Figure 2). Seeds are built in the inner part of the tracker and the track candidates are reconstructed out-wards. At HLT the seeding is done also

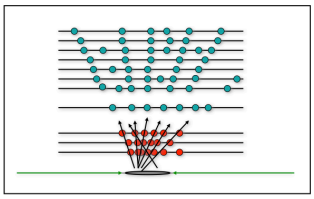

**Figure 2:** Seeding.

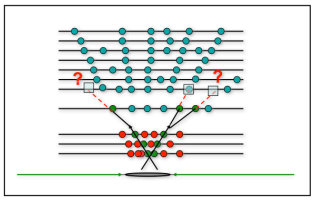

**Figure 3:** Building.

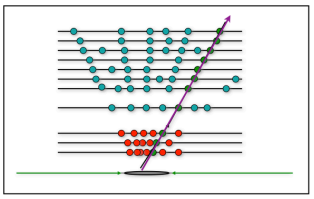

**Figure 4:** Fitting.

by using already reconstructed pixel tracks, pixel or the inner layers of the strip tracker hit triplets or pairs (plus the beam spot constraint). A seed defines the initial estimate of the trajectory parameters and their uncertainties. To limit the number of hit combinations, seeds are required to satisfy loose criteria (minimum $p_T$ and consistency with originating from the proton-proton interaction region);

- the pattern recognition (building), when track candidates are propagated using a Kalman filter technique [5] to find new compatible hits and the track parameters are updated, as shown in Figure 3. 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. In this step track candidates are rejected if not enough hits are found;

- the final track fitting is used to provide the best estimate of the parameters of each trajectory combining all the associated hits by means of a Kalman filter and smoother (see Figure 4). At this stage hits can be rejected if they look incompatible to the fitted track. 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 selection sets quality flags based on a set of cuts sensitive to fake tracks and based on track normalized $\chi^2$, track compatibility with interaction region, track length and number of missed hits. Tracks which do not fulfill the loosest set of cuts are discarded. This step is particularly important in order to mitigate the fake rate (fraction of reconstructed tracks that are fake).

Reconstruction efficiency relies on several iterations (steps) of the tracking procedure; every step, except the first, works on the not-yet-associated hits surviving the previous step. This recursive procedure is referred to Iterative Tracking, and it is the standard track reconstruction adopted by CMS. 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 discarded, reducing the combinatorial complexity and thus allowing later iterations to search for lower $p_T$ or highly displaced tracks.

Tracks reconstructed by using both pixel and strip hits have superior momentum resolution and a lower probability of being fake. However, this requires much more CPU time than just reconstructing pixel tracks, since the strip tracker does not provide the precise 3-D hits of the pixel tracker, and suffers from a higher hit occupancy. The track reconstruction absorb about 20% of the total CPU time. The HLT uses track reconstruction software that is identical to that used for offline reconstruction, but it must run much faster. This is achieved by using a modified configuration of the track reconstruction, in particular by

- performing track reconstruction only when necessary, and only after other requirements have been satisfied, so as to reduce the rate at which tracking must be performed;
- a regional track reconstruction, where the software is used to reconstruct tracks lying within a specified $\eta$-$\phi$ region around some object of interest (as muon, electron, or jet candidate);
- increasing the $p_T$ requirement when forming the seeds (usually $\sim$ 1 GeV). These stringent requirements reduce the number of seeds, and thereby the amount of time spent building track candidates;
- selecting only the track phase-space in which tracks mostly comes from the primary interaction;
- stopping the track candidate building once a specific condition is met, for example, a given minimum number of hits (typically eight), or a certain precision requirement on the track parameters. As a consequence, the hits in the outermost layers of the tracker tend not to be used. While such partially reconstructed tracks will have slightly poorer momentum resolution and higher fake rates than fully reconstructed tracks, they also take less CPU time to construct.
- decreasing the maximum number of built candidates from a given seed (at HLT is 2);
- decreasing the number of track reconstruction iterations.

In 2015 the iterative tracking will consist of 4 iterations at HLT. The main differences between the 4 iterations lie in the configuration of the seed generation and final track selection steps. Iteration 0 reconstructs the most part of the tracks (around 80%) and is designed to reconstruct prompt tracks with $p_T > 0.9$ GeV by using the already reconstruct pixel tracks as seed. Iteration 1 is configured to find low $p_T$ prompt tracks and it is seeded by pixel triplets. Iteration 2 is used to recover prompt tracks which only have two pixel hits or slightly lower $p_T$. Iteration 4 is intended to find tracks which originate outside the beamspot and to recover tracks not found by the previous iterations. The last iteration will be run only when displaced tracks are needed.

A factor 3.5 of improvement in the CPU time at <PU>$\sim$ 40 has been obtained by optimizing the iterative tracking at HLT, as shown in Figure 5. This reduction is particularly clear in Figure 6, where the comparison between the configuration used in 2012 is compared to the configuration foreseen for 2015 is shown for each iteration of the track reconstruction at the detail level of the single reconstruction step (HLT module).

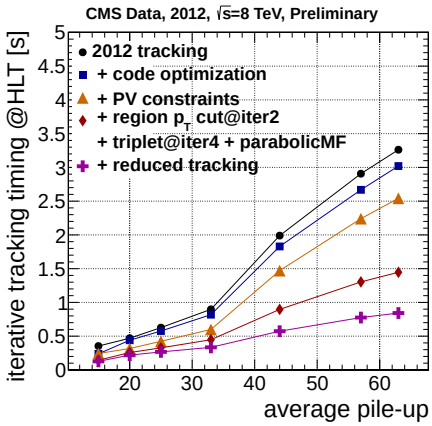


**Figure 5:** Tracking time per event vs average pile-up. The black curve refers to the tracking configuration used in 2012, while the other distributions refer to different tracking configurations, adding sequentially the changes foreseen for 2015.

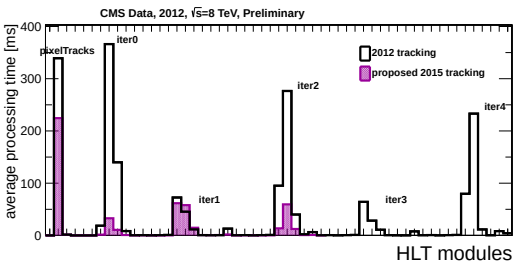


**Figure 6:** The black and magenta distributions show the average timing per event as function of the HLT module before and after the improvements foreseen for 2015, respectively.

### 3.1 Tracking performance

The performance of the iterative tracking algorithm has been evaluated on simulated $t\bar{t}$ events at $\sqrt{s}$ = 13 TeV, with average pile-up 20 and bunch spacing 25 ns. 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 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. The efficiency is obtained for simulated particles generated within $|\eta| < 2.5$, with a production point < 35 cm and < 70 cm from the centre of the beam spot for r and |z|, respectively. We also require $p_T > 0.4$ GeV. The fake rate is measured using all reconstructed tracks. Figure 7 shows the track reconstruction efficiency as function of the main kinematics variables for each iteration, where the different phase-space of tracks reconstructed by each iteration can be clearly appreciated. Moreover, it is worth to be mentioned that the overall tracking efficiency at HLT, which is the result of a regional reconstruction and an *ad hoc* configuration in order to fit the timing constraint, is around the 80%, while for the tracking algorithm used in offline reconstruction is above the 90%.

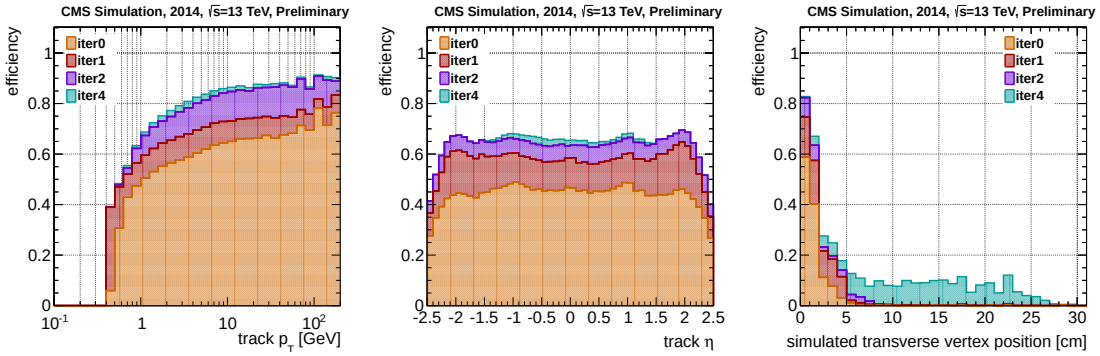


**Figure 7:** Tracking efficiency as function of $p_T$ (left), $\eta$ (centre) and the transverse distance from the beam axis to the production point of each particle (right).

The amount of material that a particle has to cross in the silicon tracker volume is far from being negligible: about 0.4 radiation lengths for tracks with pseudorapidity $\eta = 0$ and about 1.7 $X_0$ at $\eta = 1.5$. This cause a sizeable amount of photon conversions into $e^+e^-$ pairs, bremsstrahlung emission by electrons and nuclear interactions of the charged hadrons with the tracker material. Figure 8 (right) shows the fake rate as function of the track $\eta$. As expected, the largest tracking inefficiency comes from those regions of the tracker where the material budget is large. This effect is more significant for low energy hadrons due to their higher cross section for nuclear interactions (see Figure 8 (left)).

Electrons, being charged particles, can be reconstructed through the standard track reconstruction. However, as electrons lose energy primarily through bremsstrahlung, rather than ionization, large energy losses are common. The energy loss distribution is highly non-Gaussian, and therefore

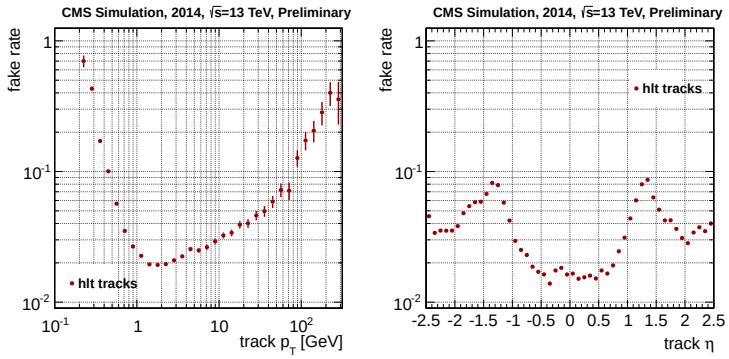


**Figure 8:** Fake-rate as function of $p_T$ (left) and $\eta$ (right).

the standard Kalman filter, which is optimal when all variables have Gaussian uncertainties, is not appropriate. As a result, the efficiency and resolution of the standard tracking are not particularly good for electrons. To obtain the best parameter estimates, the final track fit is performed using a modified version of the Kalman filter, called the Gaussian Sum Filter (GSF) [6]. In essence, the fractional energy loss of an electron, as it traverses material of a given thickness, is expected to have a distribution described by the Bethe-Heitler formula. The GSF technique approximates the energy-loss distribution as the sum of several Gaussian functions.The performance of the GSF electron tracking has been studied at HLT and Figure 9 shows the comparison between the Kalman filter and the GSF techniques in terms of efficiency as function of the electron energy, The introduction and commissioning of GSF tracking in the HLT is a major improvement, giving in this case a 25% rate reduction for no efficiency loss.

### 3.2 Physics object performance

CMS plans to extend the usage of Particle Flow technique at HLT for Run2 and to use tracking also in lepton isolation and b-tagging in order to improve the signal efficiency and background rejection. The background rejection rate for lepton triggers can be enhanced by requiring leptons to be isolated. One method of doing this is to use a veto on the presence of (too many) tracks in a cone around the lepton direction. Triggering on jets produced by b quarks can be done by counting the number of tracks in a jet that have transverse impact parameters statistically incompatible with the track originating from the beam-line. For such algorithms, the high efficiency and low track fake rate is the key ingredient, therefore the iterative tracking procedure applied in the region-of-interest is the best choice.

Figures 10,11 and 12 show the achieved improvements with respect to the 2012 configuration in terms of the main physics objects performance. This achievement is mainly driven by the higher tracking efficiency and lower fake rate. The iterative tracking improves both the signal efficiency and the background rejection, and it also guarantees a more robust response with respect to the pileup.It is worth to be noticed that by exploiting the iterative tracking and tuning its configuration, preliminary studies on events with high pileup collected in special fills in 2012 show that there is also a gain in the timing (between the 15% and 40%).

## 4. Vertex reconstruction

The reconstruction of the collision primary vertices is one of the important measurements performed with the CMS silicon tracker. A precise determination of the position of the primary interaction is needed for the track reconstruction, when a constraint to the interaction point is required, and for object identification which need to identify tracks from displaced vertices, like the tagging

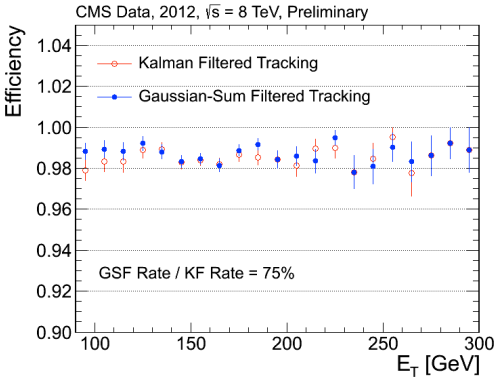


**Figure 9:** Comparison between the Kalman filter and the Gaussian Sum filter techniques. The efficiency of a well identified electron passing offline selection to pass the online electron selection with $E_T > 80$ GeV is reported as function of the electron $E_T$.

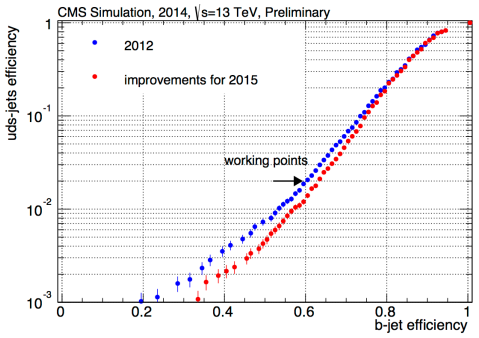


**Figure 10:** Efficiency of the Combined Secondary Vertex b-tagging algorithm at HLT for light-jets vs b-jets. The blue and red curves show the performance before and after the improvements foreseen for 2015, respectively.

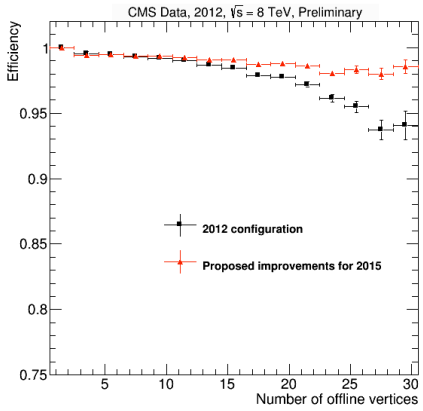


**Figure 11:** Muon isolation efficiency as a function of the number of reconstructed vertices before (black) and after (red) the improvements in muon isolation at HLT foreseen for 2015.

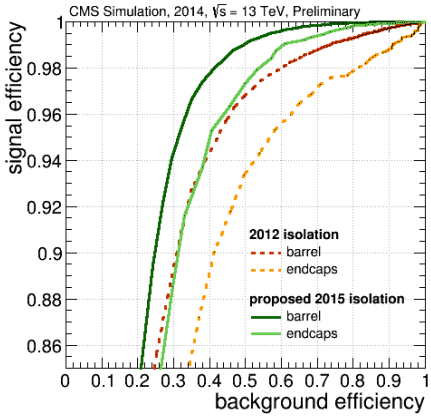


**Figure 12:** Performance curves of electron isolation at HLT, comparing the detector based setup used in RunI to a new Particle Flow based method. Working points used at HLT typically have a signal efficiency of about 99%.

of b-quark jets. Furthermore the determination of the interaction primary vertex longitudinal (z) positions and their multiplicity is a key element to deal with the pileup in each bunch crossing: an efficient vertex reconstruction and an accurate z position determination allow to discriminate among tracks produced by the hard interaction and those produced by the additional soft interactions.

For dealing with the timing constraint at HLT a dedicated track and primary vertex reconstruction based only on pixel tracker hits is performed to provide a set of primary vertices which can be used for the track reconstruction, thanks to the speed of this simple reconstruction. Vertex finding using pixel tracks provides a simple and efficient method for measuring the position of the primary vertex. The clustering of tracks is performed using a gap clustering algorithm, with vertex candidates having at least two tracks fitted using an adaptive vertex fit.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. In 2012 data taking, where the number of interactions per bunch crossing reached 30, the number of reconstructed vertices shows a linear dependence on the number of interactions without saturating (see Figure 13).

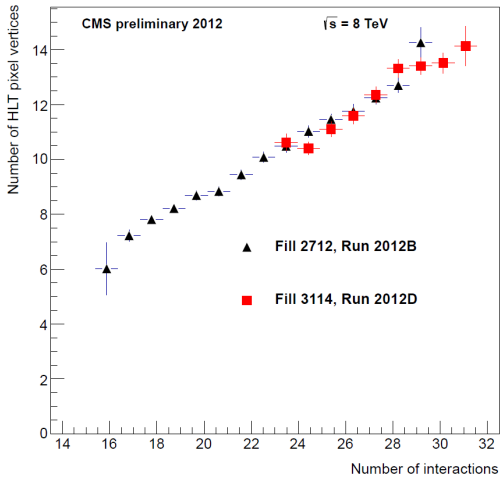


**Figure 13:** Number of reconstructed pixel vertices as function of the number of pile-up interactions.

A more precise estimation of the primary vertex position can be achieved by performing the vertex fitting with an Adaptive Vertex Fitter [7] where the track clustering is made with a deterministic annealing (DA) algorithm [8]. 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. This procedure is particularly expensive in terms of CPU timing, therefore it is run only for estimating with high accuracy observables needed for the b-jet identification, as the impact parameters of tracks within the jet.

## 5. Conclusion

At HLT the track reconstruction is done by using the same algorithm as in the offline reconstruction, which is a sophisticated software, based on Kalman filter techniques. The need of having the highest performance as possible, in terms of high track efficiency and low fake-rate, while keeping the CPU timing within the constraint, forces the development of an *ad hoc* tracking configuration at HLT. This is able to reconstruct tracks over the full rapidity range of the tracker and for promptly produced charged particles the average tracking efficiency typically 80%. This performance is not as high as the offline version, but it has been shown to guarantee a good performance in terms of the physics objects. The application of the iterative tracking at HLT allows, indeed, to decrease the event rate while keeping a high efficiency on selecting events for the physics analysis.

## References

[1] CMS Collaboration, *JINST* **3**, S08004 (2008).
[2] CMS Collaboration, CMS The TRIDAS Project, TDR CERN LHC 02-26, CMS TDR 6 (2002).
[3] CMS Collaboration, The Tracker System Project TDR CERN-LHCC 98-6 (1998).
[4] CMS Collaboration, Addendum to the CMS Tracker TDR CERN-LHCC 2000-16 (2000).
[5] R. Fruhwirth, Nucl.Instrum.Meth. A262 (1987) 444-450.
[6] W. Adam, R. Fruhwirth, A. Strandlie, and T. Todorov, J. Phys. G31(2005) N9, doi:10.1088/0954-3899/31/9/N01.
[7] R. Fruhwirth, W. Waltenberger, and P. Vanlaer, CMS Note 2007-008, 2007
[8] K. Rose, E. Gurewitz, G. Fox, Proceedings of the IEEE 86 (1998) doi:10.1109/5.726788.
[9] CMS Collaboration, CMS-TRK-11-01, Submitted to JINST (2014).
[10] CMS Collaboration, CMS-PAS PFT-09-001(2009).