Available online at www.sciencedirect.com
ScienceDirect
Nuclear Physics B Proceedings Supplement 00 (2014) 1–3
Nuclear Physics B Proceedings Supplement

Tracking at High Level Trigger in CMS

M. Tosi
Dipartimento di Fisica, Universitá degli Studi Di Padova & INFN
via Marzolo 8, 35122 Padova (Pd), Italy

Abstract

The trigger systems of the LHC detectors play a crucial role in determining the physics capabilities of 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 HLT tracking algorithms, discussing its impact on CMS physics program, as well as new developments done towards the next data taking in 2015.

Keywords:
HEP, trigger, tracking, CMS, LHC

1. Introduction

CMS [1] has a wide physics program for RunII (rediscovery of the Standard Model at 13 TeV, search of possible new physics, precision measurements of rare processes), therefore the main goal of the CMS trigger system is to keep the largest as possible number of events for analyses while keeping the event rate within the system limitation, namely 500 Hz. 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], which uses a processor farm running C++ software to achieve large reductions in data rate, filters events selected at rates of up to 100 kHz.

In 2015, data taking operations are expected to restart 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. Moreover, the expected average number of overlapping proton-proton interactions (PU) 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 designing tracking algorithms with high efficiency and a low fraction of fake tracks is very challenging. In addition the tracking code must run sufficiently fast so that it can be used at the HLT.

2. Track reconstruction at HLT

The HLT uses track reconstruction software that is almost identical to that used for offline reconstruction [8], but it has to fulfill the CPU timing constraint: the event selection has to be done in about 200 ms. Tracks are reconstructed using hits from both pixel and strip detectors, but they can be reconstructed from hits found using only the pixel tracker. This is extremely fast, and is used with great effect in the reconstruction of the primary-vertex position. In 2012, for instance, the number of reconstructed vertices shows a linear dependence on the number of interactions without saturating (see Figure 1).

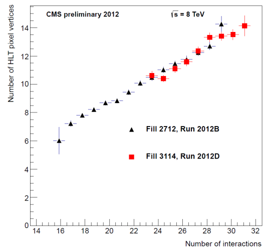

Figure 1: Number of reconstructed pixel vertices as function of the number of pile-up interactions.

In CMS, tracks are reconstructed in four steps:

- the seed generation (*seeding*) provides initial track candidates (see Figure 2), defining the initial estimate of the trajectory parameters and their uncertainties.
- 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 final track fitting (*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).
- the track selection sets quality flags based on a set of cuts sensitive to fake tracks, on the track normalized χ^2, and on its compatibility with interaction region.

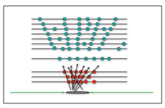

Figure 2: Seeding.

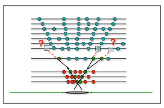

Figure 3: Building.

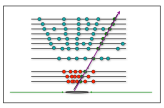

Figure 4: Fitting.

Reconstruction efficiency relies on several iterations of the tracking procedure; each step, except for the first one, works on the not-yet-associated hits surviving the previous step, reducing the combinatorial complexity. In the early iterations tracks with relatively high p_T, produced near the interaction region, are reconstructed, discarding hits associated with those tracks, later iterations can search for lower p_T or highly displaced tracks. This recursive procedure is referred to as *Iterative Tracking*.

Because the tracking is a sophisticated and complex software and it is one of the most time consuming step (about 20% of the total CPU time), the HLT has to run it 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 at the end of the event selection process, (after having applyed other requirements based on the fast reconstruction of the physics object, as muon, electron, jet and tau);
- using a regional track reconstruction, where tracking is done only within regions-of-interest defined by the direction of the already available physics object;
- increasing the p_T requirement when forming the seeds (usually $\sim$ 1 GeV);
- selecting only the track phase-space in which tracks mostly comes from the primary interaction;
- stopping the track candidate building once specifics condition are met, for example, number of hits (typically 8), or the track parameters precision;
- limiting the maximum number of built candidates from a given seed (typically 2);
- limiting the number of iterations.

Moreover, because the silicon strip unpacking takes a long time and has 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.

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 for prompt tracks 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 have only two pixel hits or slightly lower p_T. Iteration 4 is intended to find displaced tracks with respect to the beamspot.

A factor 3.5 of improvement in the CPU time at $<PU>\sim 40$ has been obtained with respect to the 2012 tracking configuration by optimizing the iterative tracking at HLT, as shown in Figure 5.

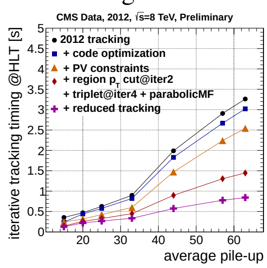

Figure 5: Tracking time per event vs average pile-up. The black curve refers to the tracking configuration used in 2012, while the other distributions refer to different tracking configurations, adding sequentially the changes foreseen for 2015.

2.1. Tracking performance

The performance of the iterative tracking algorithm at HLT has been evaluated on simulated $t\bar{t}$ events at $\sqrt{s}$ = 13 TeV, with average pile-up 20 and bunch spacing 25 ns. The tracking efficiency is defined as the fraction of simulated charged particles that can be associated to 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 amount of material that a particle has to cross in the silicon tracker volume is far from being negligible (from 0.4 X_0 at $\eta \sim 0$ to 1.7 X_0 at $\eta = 1.5$). This cause a sizeable amount of photon conversions, bremsstrahlung and nuclear interactions in the tracker material. Figure 6 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. The overall tracking efficiency at HLT, is around the 80%, while in offline reconstruction it is above the 90%.

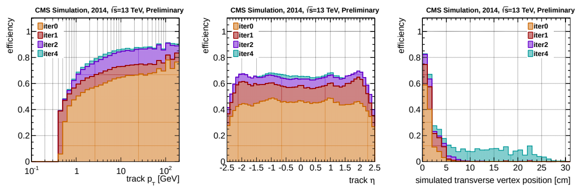

Figure 6: Tracking efficiency as function of p_T (left), η (centre) and the transverse distance from the beam axis to the production point of each particle (right).

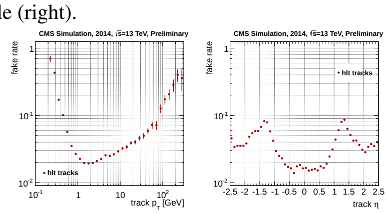

Figure 7: Fake-rate as function of p_T (left) and η (right).

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. Figure 7 (right) shows the fake rate as function of the track η. As expected, the largest tracking fake rate 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, as shown in Figure 7 (left).

2.2. *Impact on the physics object performance*

CMS plans to extend the usage of particle-flow technique at HLT in RunII and to use the tracking also in lepton isolation and b-tagging in order to improve the signal efficiency and background rejection. For such algorithms, high efficiency and low track fake rate are the key ingredients, therefore the *Iterative Tracking* procedure applied in the region-of-interest is the best choice.

Figure 8 shows the achieved improvements with respect to the 2012 configuration in terms of the main physics objects performance. 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.

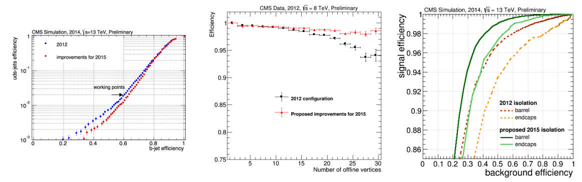

Figure 8: Comparison between object reconstruction performance obtained with 2012 configuration and the one planned to be used in 2015 data taking, which is exploiting the tracking. Left: efficiency of the b-tagging algorithm at HLT for light-jets vs b-jets; centre: muon isolation efficiency as a function of the number of reconstructed vertices; right: performance curves of electron isolation at HLT (typical working points have a signal efficiency of about 99%).

3. 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 possible performance, 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. For promptly produced charged particles the average tracking efficiency is 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] R. Fruhwirth, W. Waltenberger, and P. Vanlaer, CMS Note 2007-008, 2007.
[7] K. Rose, E. Gurewitz, G. Fox, Proceedings of the IEEE 86 (1998) doi:10.1109/5.726788.
[8] CMS Collaboration, CMS-TRK-11-01, Submitted to JINST (2014).
[9] CMS Collaboration, CMS-PAS PFT-09-001(2009).