Primary vertex reconstruction with the ATLAS detector

F. Meloni on behalf of the ATLAS collaboration

Albert Einstein Center for Fundamental Physics, Laboratory for High Energy Physics, University of Bern, Sidlerstrasse 5, 3012 Bern, Switzerland

E-mail: federico.meloni@cern.ch

Abstract: Efficient and precise reconstruction of the primary vertex in a LHC collision is essential for determining the full kinematic properties of a hard-scatter event and of soft interactions as a measure of the amount of pile-up. The reconstruction of primary vertices in the busy, high pile-up environment of Run-2 of the LHC is a challenging task. The algorithms developed by the ATLAS experiments to reconstruct multiple vertices with small spatial separation are presented.

Keywords: Performance of High Energy Physics Detectors; Pattern recognition, cluster finding, calibration and fitting methods
Contents

1 Introduction 1
2 The ATLAS detector and the LHC luminous region 1
3 Primary vertex reconstruction 2
4 Imaging-inspired seed finding algorithm 3
5 Vertex reconstruction efficiency parameterisation 4
6 Conclusion 5

1 Introduction

Efficient and precise reconstruction of the primary vertex, defined as the point at which a proton-proton (pp) interaction occurred, is an important element of data analysis at the LHC. The correct assignment of charged particle trajectories to their source vertex is essential to reconstruct the full kinematic properties of the hard-scatter. An aspect of primary vertex reconstruction requiring special attention at high luminosity is the superposition of multiple inelastic pp that occur during the same LHC bunch crossing. These additional interactions are referred to as pile-up. The average number of inelastic pp interactions per bunch crossing under constant beam conditions is denoted as $\mu$ and is directly related to the instantaneous luminosity.

This document describes the primary vertex reconstruction with the ATLAS detector, presenting studies based on the data collected in 2015 at a proton-proton centre-of-mass energy of $\sqrt{s} = 13$ TeV.

2 The ATLAS detector and the LHC luminous region

The ATLAS detector [1] is a multi-purpose detector with a cylindrical geometry. It is composed of an inner detector (ID) surrounded by a thin superconducting solenoid, a calorimeter system and a muon spectrometer embedded in a toroidal magnetic field. The ID is the primary detector used for vertex reconstruction and covers the pseudorapidity\(^1\) range $|\eta| < 2.5$. It is composed of three subdetectors: the innermost silicon pixel detector, the surrounding silicon microstrip semiconductor tracker (SCT), and the transition radiation tracker (TRT). All three subsystems are divided into a barrel section and two end-caps. The barrel sections consist of several cylindrical layers of sensors

\(^1\)ATLAS uses a right-handed coordinate system with its origin at the nominal interaction point (IP) in the centre of the detector and the $z$-axis along the beam pipe. The $x$-axis points from the IP to the centre of the LHC ring, and the $y$-axis points upward. Cylindrical coordinates ($r, \phi$) are used in the transverse plane, $\phi$ being the azimuthal angle around the beam pipe. The pseudorapidity is defined in terms of the polar angle $\theta$ as $\eta = -\ln \tan(\theta/2)$. 

---

2016 JINST 11 C12060
(four pixel and four double-sided microstrip layers) while the end-caps are composed of a series of wheels of sensors (three pixel and nine double-sided microstrip layers).

The size, position and shape of the luminous region, also called beam spot, are determined by the operating parameters of the LHC. The transverse size is determined by the focusing of the LHC beams near the interaction region and by the emittance of the protons within the colliding bunches. The longitudinal size of the luminous region is determined by the bunch length and by the angle at which the two beams are brought into collision. The typical Run 2 sizes of the luminous region, which is described by a three-dimensional gaussian, are respectively of about 14 µm in the transverse direction, and 45 mm in the longitudinal direction.

3 Primary vertex reconstruction

The input to the vertex reconstruction is a collection of reconstructed charged particle trajectories (tracks) in the ID. For tracks to be considered in the construction of a vertex they must pass the following requirements:

- Transverse momentum $p_T > 400$ MeV and $|\eta| < 2.5$
- Number of silicon hits $\geq 9$ (11) if $|\eta| < 1.65$ ($|\eta| > 1.65$)
- Hits in the first two pixel layers $\geq 1$
- A maximum of 1 shared module (1 shared pixel hit or 2 shared SCT hits)
- Pixel holes = 0
- SCT holes $\leq 1$

where a hole represents a measurement on a detector surface that is expected, given the trajectory predictions, but not observed.

The procedure of primary vertex reconstruction is divided in two stages: vertex finding and vertex fitting. It consists of the following steps:

1. A set of tracks passing the vertex selection criteria above is defined.
2. A seed position for the first vertex is selected.
3. The tracks and the seed are used to fit the best vertex position. The fit is an iterative procedure, and in each iteration less compatible tracks are down-weighted and the vertex position is recomputed.
4. After the vertex position is determined, tracks that are incompatible with the vertex are removed from it and allowed to be used in the determination of another vertex. Vertices are required to have at least two associated tracks.
5. The procedure is repeated until no unassociated tracks are left in the event or no additional vertex can be found in the remaining set of tracks.
The output of the vertex reconstruction algorithm is a set of three dimensional vertex positions and their covariance matrices. The reconstructed position and width of the beam spot can be used as an additional measurement during the primary vertex fit, to help constraining the transverse position resolution of vertices reconstructed from a small number of tracks.

The vertex reconstruction efficiency is determined from data by taking the ratio between events with a reconstructed vertex and events with at least two reconstructed tracks. Vertex reconstruction becomes fully efficient for five or more tracks as shown in figure 1.

![Vertex Efficiency Graph](image)

**Figure 1.** Efficiency of vertex reconstruction as a function of the number of tracks in run 267359 [2], a low-\(\mu\) (\(\mu \sim 0.01\)) dataset corresponding to an integrated luminosity of 216.9 \(\mu b^{-1}\). The shaded error bars represent the statistical uncertainty. The mild differences between data and simulation prediction are due to the Monte Carlo tune, which was optimised for higher track multiplicities.

In the ideal case of perfect reconstruction efficiency, the number of reconstructed vertices scales linearly with \(\mu\). In reality there are a number of effects that cause the relation to be non-linear. The most important effect is vertex merging, when two or more interactions are reconstructed as one vertex. Secondary effects include fake and split vertices, reconstruction inefficiencies, detector acceptance, and, at a small level for low track multiplicities, non-collision backgrounds.

### 4 Imaging-inspired seed finding algorithm

A new approach is in development to improve the performance of the iterative strategy described in section 3 at very high \(\mu\). This new technique attempts to simultaneously identify all vertex seeds in one LHC bunch crossing by using all tracks as input to a three-dimensional imaging algorithm. The procedure consists of the following steps:

1. A three-dimensional histogram is created representing the detector volume in which vertex finding will be done. Each bin corresponds to a voxel.
2. The bins of the histogram are filled with the path length of each track transversing the corresponding voxel, after applying a voxel raytracing algorithm [3].
3. The obtained image is transformed into frequency space.

4. A combined filter taking into account the detector angular acceptance and mitigating the effect of high frequency variations in each of the x, y, and z directions is multiplied with the frequency space histogram.

5. The filtered frequency space image is then transformed back to position space.

6. The resulting image is then passed to a separate clustering algorithm where all seeds are identified from peaks in the image.

The simultaneous identification of all seeds allows to reduce the impact of vertex merging. Each resulting group of tracks is then fit with the standard vertex fitting algorithm. If two close-by vertices produce two seeds, both are likely to have some tracks assigned to them and two final reconstructed vertices can be found, improving on the iterative strategy, as shown in figure 2.

![Figure 2](image-url) Fraction of reconstructed vertex pairs versus distance in the z direction between between all pairs of vertices in simulated events with pile-up, zoomed in on the region near 0 mm [4]. The fraction is computed from the reconstructed vertices and the number of interactions in simulation. The dip at low separation comes from merging separate collisions into one reconstructed vertex. The enhancement above 1 for imaging results from split vertices and is due to the binning choice.

5 Vertex reconstruction efficiency parameterisation

An analytical model to compute the average number of reconstructed vertices, \( \langle n_{\text{vertices}} \rangle \), as a function of \( \mu \) has been developed:

\[
\langle n_{\text{vertices}} \rangle = p_0 + \epsilon \cdot \mu - F(\epsilon \cdot \mu, p_{\text{merge}})
\]  

(5.1)

where \( \epsilon \) is the efficiency of the vertex reconstruction algorithm, and \( p_0 \) accounts for any small offset arising from non-collision backgrounds. The quantity \( \epsilon \cdot \mu \) represents the average number of vertices that would be reconstructed in the absence of any pile-up induced vertex merging effects. The function \( F(\epsilon \cdot \mu, p_{\text{merge}}) \) represents the average number of vertices lost due to merging effects,
taking into account the vertex merging probability, $p_{\text{merge}}$. The distribution of $\Delta z$ measured in a low-$\mu$ data sample is used to derive a two-vertex merging probability density function $p_{\text{merge}}(\Delta z)$ that is then be combined with a given beam-spot shape to derive a final prediction, shown in figure 3.

![Figure 3](image)

**Figure 3.** The average number of reconstructed vertices is shown as a function of the average number of interactions $\mu$ [5]. The red triangles and the orange line show respectively the simulation prediction for minimum-bias events and the parameterisation described in this section. The dependence on the density of interactions is exploited to test the effect of a different beamspot longitudinal profile (approximately flat, called Crab Kissing); this is shown in the figure as a blue line. The correction to the number of reconstructed vertices for masking effects $F(\epsilon \cdot \mu, p_{\text{merge}})$ is shown by the dotted lines for the two configurations.

6 Conclusion

The ATLAS primary vertex reconstruction shows high efficiency and good agreement with the simulation predictions. A study of high pile-up effects on the reconstruction performance was performed in preparation for the higher luminosities expected with the foreseen upgrade of LHC. An imaging algorithm has been developed in order to produce vertex seed positions simultaneously for the whole event, differently from the previous approach of iteratively producing single new seeds and fitting them. Preliminary performance studies in simulations corresponding to the conditions expected early in LHC Run-2 show that with this it is possible to identify more closely spaced interactions and thus to lessen the effects of merging in vertex reconstruction.

References

[1] ATLAS collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, 2008 *JINST* 3 S08003.

[2] ATLAS collaboration, *Vertex Reconstruction Performance of the ATLAS Detector at $\sqrt{s} = 13\,\text{TeV},* ATL-PHYS-PUB-2015-026 (2015).
[3] J. Amanatides and A. Woo, *A fast voxel traversal algorithm for ray tracing*, in proceedings of *Eurographics* ’87, Amsterdam, The Netherlands, 24–28 August 1987, pp. 3–10.

[4] ATLAS collaboration, *An imaging algorithm for vertex reconstruction for ATLAS Run-2*, ATL-PHYS-PUB-2015-008 (2015).

[5] ATLAS collaboration, *Vertex reconstruction Plots* (2012), http://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/IDTRACKING/PublicPlots/ATL-COM-PHYS-2012-474/.