Vertexing Algorithms with the ATLAS Detector for the HL-LHC Upgrade

Ian T. Lim,[1] Benjamin Nachman,[2] Simone Pagan Griso,[2] and Maurice Garcia-Sciveres[2]

[1]*Department of Physics, Stanford University, Stanford, California 94305, USA*
[2]*Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA*

(Dated: December 4, 2017)

We evaluate the performance of the standard vertexing algorithms used in the LHC Run 1 analyses. With the analysis framework ROOT, we develop metrics for vertexing performance and quantitatively compare the current algorithms to possible alternatives in the high pile-up regime ($\langle\mu\rangle \sim 200$). Our results will guide algorithm development in preparation for the HL-LHC upgrade, which will begin operation in mid-2026.

INTRODUCTION

The ATLAS Experiment is one of two general purpose detectors at the Large Hadron Collider (LHC), a particle accelerator near Geneva, Switzerland run by the European Organization for Nuclear Research, CERN. Currently observing at $\sqrt{s} = 13\,\text{TeV}$ center-of-mass energy,[1] ATLAS focuses on the analysis of proton-proton interactions, using a system of silicon pixel and strip detectors to detect the trajectories of particles generated by collisions and decays within the detector chamber. Using algorithms to reconstruct the tracks left by particles such as electrons and pions, one can determine the positions and energies of so-called hard scatter events, proton collisions of potential interest. A few of these (at most one in 10^{12}) hard scatter events may provide indications of new physics beyond the Standard Model, such as supersymmetry and the nature of dark matter.

ATLAS regularly collides bunches of protons with a period of 25 nanoseconds,[1, 2] and each bunch crossing currently results in roughly 20-40 proton-proton collisions. All but one of these collisions are a source of noise and are known as known as pile-up. Therefore, a central challenge for ATLAS is the reconstruction of vertices, which are the spatial locations of energetic proton-proton collisions, including hard scatter events. This process is also called vertexing. The correct association of outgoing particle tracks to reconstructed vertices is important to separate pileup from hard scatter events in the ATLAS detector as a whole.

Moreover, because the events of particular interest at the LHC are so rare, future upgrades to the accelerator and the detector have been designed with an increased luminosity, i.e. a higher rate of collisions. Specifically, the HL-LHC (High-Luminosity LHC) upgrade will increase the rate of proton-proton collisions by a factor of 10,[1] but will also increase the relative proportion of pile-up so that events of interest become harder to distinguish from the background. In light of this, improved vertex reconstruction algorithms will be needed to keep up with the increased luminosity in order to continue processing the massive amounts of data being produced by the collider.

Here, we investigate the performance of existing vertexing algorithms and study possible alternatives for use at the HL-LHC. By benchmarking current methods against several basic metrics for vertex reconstruction and track association, we establish a standard of comparison for new approaches to vertexing.

MATERIALS AND METHODS

For this project, we used ROOT 6.10.04,[3] the standard analysis framework used to process experimental particle physics datasets like those produced at the LHC. ROOT is primarily written in C++ but can also be used with other languages such as Python and R. The ROOT framework contains functionality to read, process, and visualize the large datasets generated by the ATLAS detector and Monte Carlo simulations thereof. Its data structures (the most important of which is the TTree) are designed for efficient data retrieval.

In order to evaluate the performance of existing vertexing algorithms, we use Monte Carlo simulations of different events in the ATLAS detector. The general procedure is as follows. By simulating proton-proton collisions, we can compute the trajectories of collision byproducts in our detector, which we call truth tracks. Using these trajectories, we then simulate the response of our detector to these sprays of incident particles, and run our vertexing algorithms on the simulated hits in our detector. From these hits we can reconstruct particle tracks, just as we would with experimental data, and based on these tracks we can determine the most likely locations of the collisions in the event. We can additionally match these reconstructed, or reco, tracks to truth tracks by matching the detector hits assigned to a reco track to the simulation truth particle that produced those hits. In this way, we can use track matching to match reco vertices to so-called truth vertices.

A major benefit of Monte Carlo simulations is the ability to compare our reconstructed vertices to truth vertices, since we do not have direct access to the true collisions and their byproducts in our actual detector. By comparing truth and reco vertices, we can test the accuracy of our algorithms in determining important characteris-

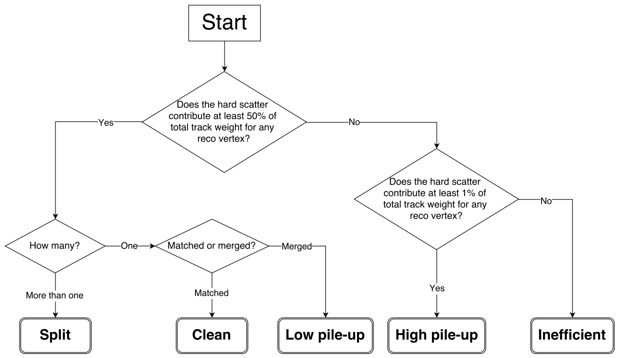

FIG. 1. A flow chart of the Run 1 categories for event classification as described in Aaboud et al.[2] The categories measure the level of pile-up contamination of the hard scatter vertex, and all categories except inefficient are considered to be successful in hard scatter reconstruction. Under current operating conditions, we expect over 99% hard scatter reconstruction efficiency.

tics of the bunch crossing, such as the spatial resolution of our vertex reconstruction and the level of pile-up contamination in the hard scatter reconstruction. We can also vary the parameters of our simulation and the vertexing algorithm itself to test for potential improvements to these metrics.

Furthermore, we can evaluate the overall reconstruction of a given event (bunch crossing) using the categories established for Run 1 analysis by the ATLAS Collaboration. Under this scheme, reco vertices are classified into categories based on the numerical track weights of the reconstructed particle tracks associated to that vertex. These track weights represent the track compatibility with the vertex location, and are computed as part of the fitting process. For example, a reco vertex is classified as "matched" to a truth vertex if over 70% of the cumulative track weight of that vertex is contributed by tracks from a single truth vertex.[2] Vertices without clear truth vertex counterparts are then labeled "merged," and if multiple reco vertices are found to correspond to the same truth vertex, they are further marked as "split."

After classifying individual vertices, we classify the overall event reconstruction based on how well we reconstructed the hard scatter vertex (i.e. the vertex of primary interest). The process of event categorization is shown in Fig. 1. From best to worst, an event may be classified as clean, low pile-up, high pile-up, split, or inefficient. It is then useful to understand the dependence of event categorization on other simulation conditions such as the local density of vertices around the hard scatter vertex and the angular distribution of tracks produced by the hard scatter.

RESULTS AND DISCUSSION

In this work, we primarily study Monte Carlo simulations of a Higgs-to-invisible decay under HL-LHC pileup conditions. Vertexing was performed using the iterative algorithm from Step 1.6, as opposed to the adaptive multi-vertex fitter (AMVF) approach used in later analyses. To model the HL-LHC operating conditions, our simulations were run with a center-of-mass energy $\sqrt{s} = 14\,\text{TeV}$, a beamspot size of $\sigma_z = 50\,\text{mm}$, and an average of $\langle \mu \rangle = 200$ simulated truth vertices per event. Of these 200 vertices, we expect to successfully reconstruct less than half of these vertices using the current algorithm. One notion of overall reconstruction efficiency is displayed in Fig. 2. However, in terms of hard scatter reconstruction efficiency, by the Run 1 standards we consistently reconstruct the hard scatter over 95% of the time, and with an excellent spatial resolution on the order of tens of microns. Our preliminary results reveal that there are still many vertices which we completely fail to reconstruct, even though they are in principle are bright enough (i.e. have enough high-quality tracks) to be reconstructed. Moreover, the spatial resolution of vertex reconstruction increases with the proportion of correctly matched track weight, so our overall reconstruction quality for an event can be improved by better track-vertex association, as seen in Fig. 3.

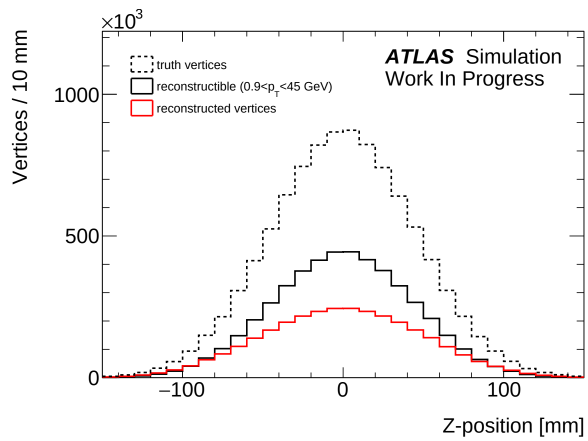

FIG. 2. A comparison of the populations of truth, reconstructible, and reconstructed vertices in the Higgs-to-invisible samples. Vertices are normally distributed around the center of the beamspot at $z = 0$ with a spread of $\sigma_z = 50\,\text{mm}$. The dashed line represents the total truth vertex population over all events, but not all these vertices are reconstructible. If a vertex produces only particle tracks very close to the beam line, it becomes very difficult for us to detect its tracks and reconstruct the origin vertex. The same is true of low-energy tracks from soft interactions. Once we apply a cut to our truth particles and require that they produce at least three high-quality tracks, we see that the populations of reconstructible (solid black) and reconstructed vertices (red) are much closer.

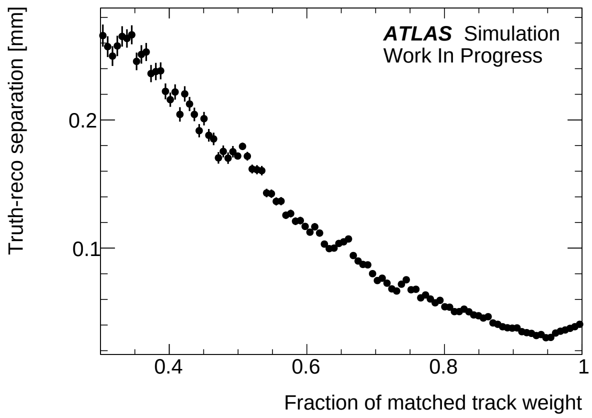

FIG. 3. As the fraction of correctly matched track weight within a reco vertex increases, the separation between the truth and reco z-positions of that vertex decreases. Therefore, better track-vertex association will improve our capacity to accurately reconstruct the locations of vertices in space. For a correctly matched weight fraction near 1, our capacity to resolve the z-position of a given vertex is on the order of tens of microns.

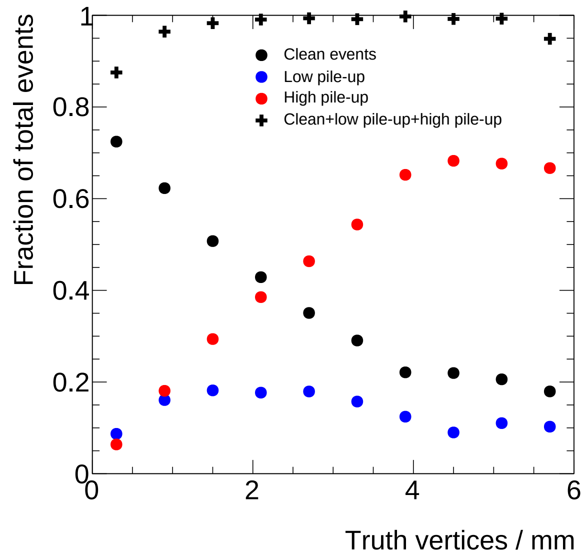

FIG. 4. The Run 1 categories for event classification depend strongly on the local vertex density around the hard scatter. Here, high pile-up dominates for densities higher than 2 vertices/mm, but overall hard scatter reconstruction efficiency remains high at over 95%. Densities were computed for a 1 mm radius around the hard scatter vertex.

Classification studies

In order to study the suitability of the Run 1 event categories for the HL-LHC operating conditions, we implemented event classification in ROOT and considered the dependence of categories on various event parameters such as local vertex density and the z-position of the hard scatter vertex. For instance, the relationship between event category and local vertex density is depicted in Fig. 4. As the density of vertices around the hard scatter increases, the proportion of clean events drops sharply, while high pile-up becomes the most common classification for densities larger than 2 vertices / mm.

It is also illuminating to understand how classification depends on the location of the hard scatter within the beam spot. In Fig. 5, we see that the relative populations of the three best event categories vary significantly with the position of the hard scatter. Towards the center of the beamspot, vertex densities are highest and the likelihood of a high pile-up reconstruction is maximized.

However, it turns out to also be true that at such high pile-up levels, event classification depends very little on the actual number of truth vertices in the event. Over a range of $170 < \mu < 230$, the relative populations of clean, high pile-up, and low pile-up are essentially constant, as shown in Fig. 6. This is likely because parameters which are strongly correlated with classification like the local vertex density vary minimally in this range of μ values. Additional vertices in an event contribute to pile-up but do not significantly impact our capacity to reconstruct the hard scatter vertex– we may effectively treat this range as a single operating regime.

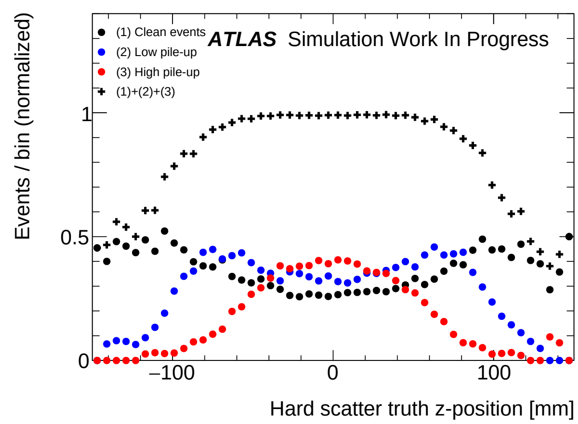

FIG. 5. Here, we see the dependence of event classification (and overall reconstruction) on the truth z-position of the hard scatter. Clean events are most common towards the edges (where the vertex density tends to be lower), high pile-up is peaked in the center of the beam spot, and low pile-up is peaked in between.

Vertex substructure

As part of our exploration of vertex merging, we did a number of studies looking for substructure within a vertex. We might reasonably ask whether there are any characteristics of merged vertices which we can use to distinguish them from matched vertices, and one way this might manifest is as internal structure within a single

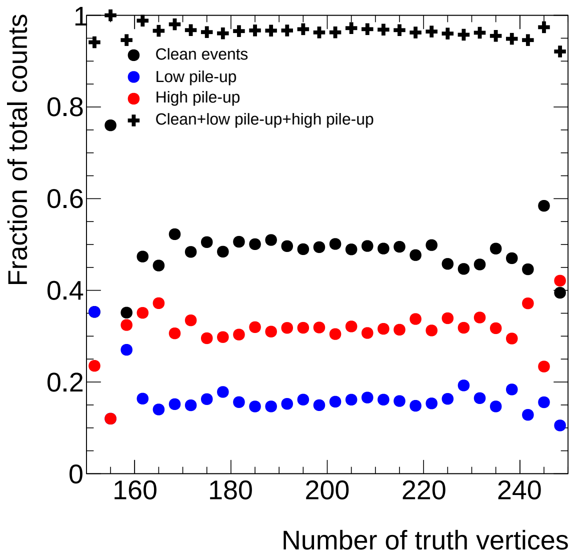

FIG. 6. Over a broad range of truth vertex counts, the relative proportions of clean, high pile-up, and low pile-up events are basically constant. This is very different from the low pile-up regime which corresponds to Run 1 operating conditions. For a range $0 < \mu < 40$, event classification is strongly correlated with the number of truth vertices μ in a given event.

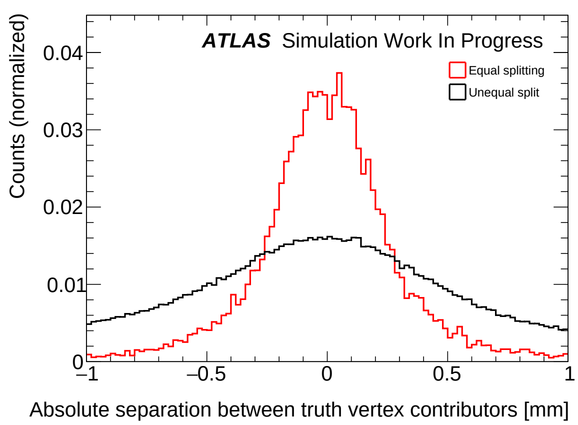

FIG. 7. Here, we examine the internal structure of highly merged vertices. Reco vertices whose track weight is contributed equally from two separate truth vertices exhibit a much narrower peak in the separation between their contributing truth vertices as compared to all other reco vertices.

reco vertex.

To begin with, we study the positions of the two most important truth vertices contributing to a given reco vertex. In the case of a well-matched reco vertex, it is possible that a single truth vertex dominates and the second most important vertex is quite distant in space. However, when the reco vertex is merged and two truth vertices contribute about equally to the reco vertex, their expected separation is much less– at some point, we simply fail to resolve the two vertices as separate, and reconstruct a single merged vertex. We see exactly this result in Fig. 7.

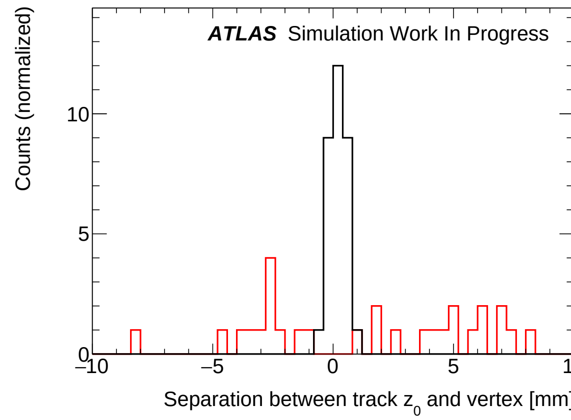

FIG. 8. In this plot, we have a single well-matched vertex in black, and a 50/50 merged vertex in red. Each vertex has 32 tracks. The merged vertex has a wide separation between the two contributing truth vertices, which contribute 50.6% and 49.3% of the total track weight. The matched vertex has 83.5% of its track weight matched to a single truth vertex, and it is clear that the tracks for the matched vertex are much more localized.

Furthermore, if two distantly separated vertices were merged and their tracks contributed about equal weight to the resulting reco vertex, we would expect to see a distinctly bimodal spread in the tracks associated to that reco vertex. In fact, such merged vertices exist and can be studied; we observe one example in Fig. 8. In this figure, the track spreads around the vertex position for the matched vertex (black) and the merged vertex (red) differ quite significantly. While the tracks from the matched vertex are closely localized around zero (i.e. the determined vertex position), the merged vertex has a much broader spread due to its tracks coming from two distinct truth vertices which are separated by almost 6 mm.

Tracks at high η

Particle tracks from the forward region of our detector near the beamline are known to be more difficult to associate to vertices. In experimental particle physics, it is conventional to denote particle track direction not by the conventional polar angle θ but by the pseudorapidity η, a dimensionless quantity defined in terms of θ as

$$\eta \equiv -\ln\left[\tan\left(\frac{\theta}{2}\right)\right]. \quad (1)$$

Thus a pseudorapidity $\eta = 0$ corresponds to a particle track at $\theta = 90^\circ$, transverse to the beamline, while a pseudorapidity $\eta = 2.44$ corresponds to a track at $\theta =$

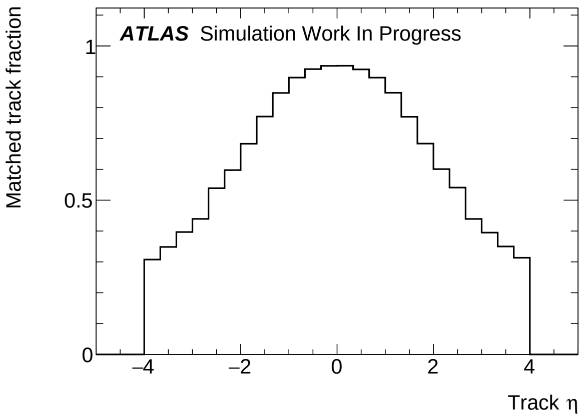

FIG. 9. Track-vertex association as a function of η. For tracks at large η, it is less likely that we associate them to the correct reco vertex.

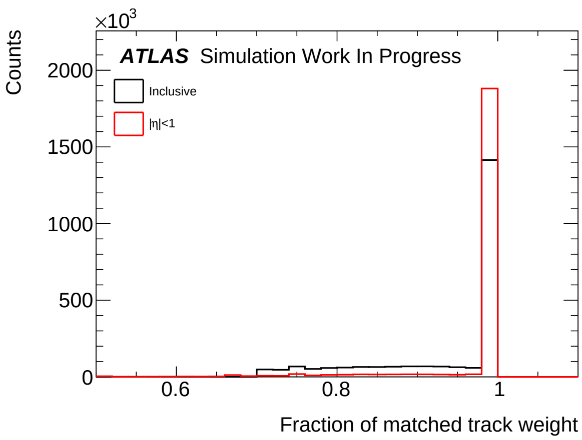

FIG. 10. We consider the fraction of correctly matched track weight within each reco vertex, first including all tracks associated to that vertex (black) and then restricting ourselves to tracks with $|\eta| < 1$ (red). When we consider tracks from $|\eta| > 1$, our probability of a track mismatch is higher and therefore the overall proportion of matched track weight is shifted towards lower values.

10°, very close to the beam axis. It is these tracks at large η, close to the beam axis, which we are especially interested in.

In Fig. 9, we observe the expected trend in track matching as a function of η. Over 90% of all tracks perpendicular to the beam axis are correctly associated to a reco vertex, but tracks at $|\eta| > 3$ are correctly matched less than half the time. Moreover, the track weight contamination introduced by high η tracks is significant, as seen in Fig. 10. The fraction of matched track weight is one good indicator of vertex quality, and so these plots clearly indicate that more care must be taken to correctly associate tracks at high η.

CONCLUSIONS

As we have seen, overall hard scatter reconstruction efficiency remains extremely high even at pile-up levels of $\mu = 200$, and moreover the hard scatter position along the beamline can often be resolved to high precision, on the order of tens of microns. However, it is also true that about half the time, a non-negligible amount of pile-up contamination is introduced to the hard scatter, negatively influencing our vertex fitting process.

Moreover, vertex merging (where multiple truth vertices are reconstructed as a single reco vertex) becomes a dominant source of reconstruction losses in the iterative algorithm. While these merging losses are of secondary importance to the hard scatter reconstruction, they are nevertheless a point of concern for future algorithm development. In our studies of merging, we have confirmed a characteristic length scale for merging and made preliminary steps in looking for substructure within merged vertices.

Finally, we have studied tracks at high η, i.e. from the forward region of the detector. Because these tracks run almost parallel to the interaction region, they are exceptionally difficult to associate to the correct vertex candidates, and we have illustrated that high-eta tracks intoduce an appreciable amount of track weight contamination to reco vertices because of the increased likelihood of a mismatch.

These studies indicate several areas of improvement for future algorithm development. Reductions to vertex merging and more refined track-vertex association methods are needed to produce cleaner, more accurate reconstructions with lower levels of pile-up contamination. However, both vertex resolution and correct track-vertex association are important goals of the vertexing process, and as these goals are often in competition with one another, we should also take care not to incur major performance losses in one in order to obtain modest improvements in the other.

This work was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internship (SULI) program.

[1] A. Collaboration, *Technical Design Report for the ATLAS Inner Tracker Strip Detector*, Tech. Rep. CERN-LHCC-2017-005. ATLAS-TDR-025 (CERN, Geneva, 2017).

[2] M. Aaboud and et al., The European Physical Journal C **77**, 332 (2017).

[3] R. Brun and F. Rademakers, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **389**, 8186 (1997).