Vertex Reconstruction in ATLAS Run II

Matt Zhang, for the ATLAS Collaboration

Background

Vertex reconstruction is the process of taking reconstructed tracks and using them to determine the locations of proton collisions.

Reconstruction occurs in two steps: **seed finding** and **vertex fitting**. In the seed finding step we find a point representing a possible vertex location. Then we associate nearby tracks and adjust the final vertex position in the fitting step.

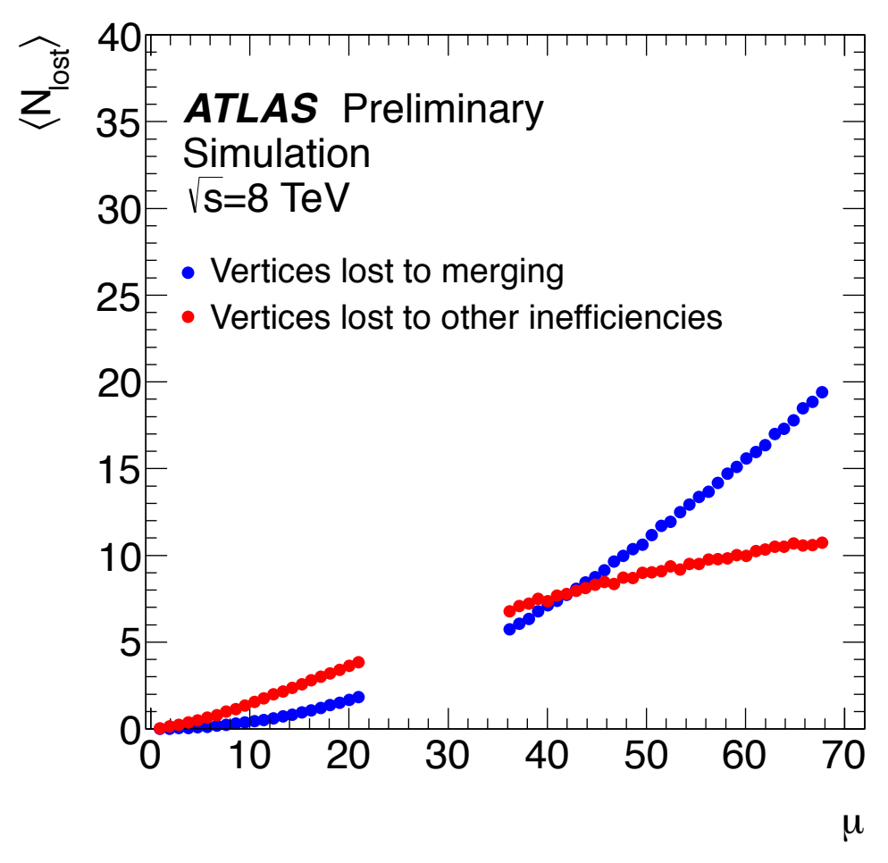
Current Method

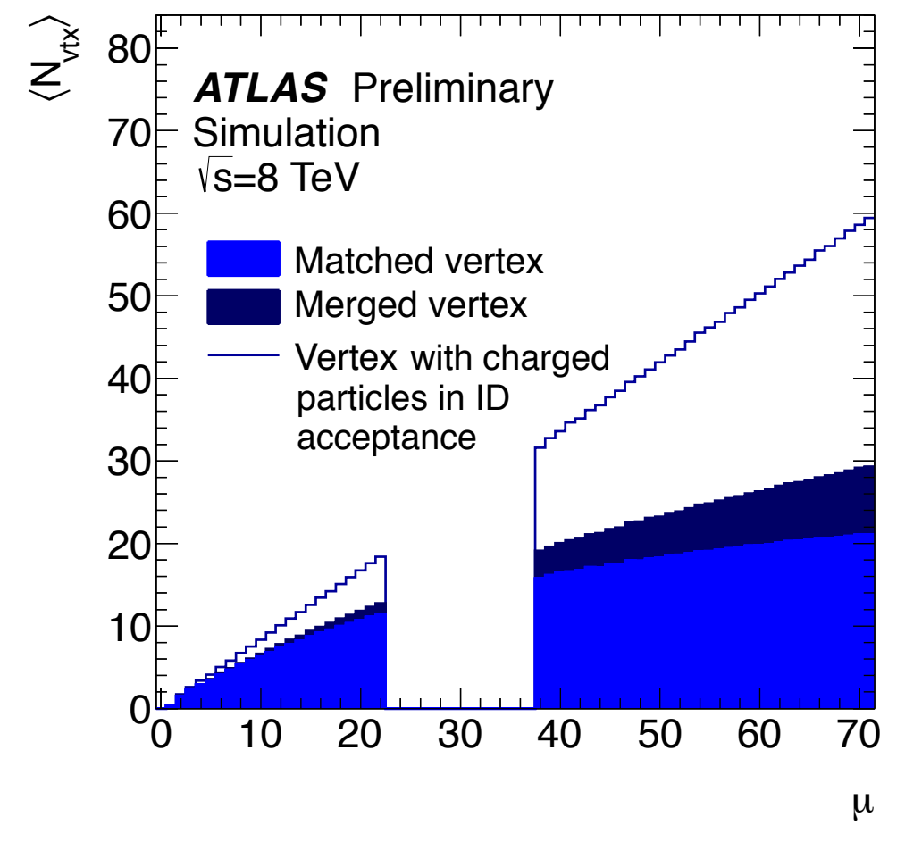
In the current vertex reconstruction method we use an iterative seed-finding procedure. The method is as follows:

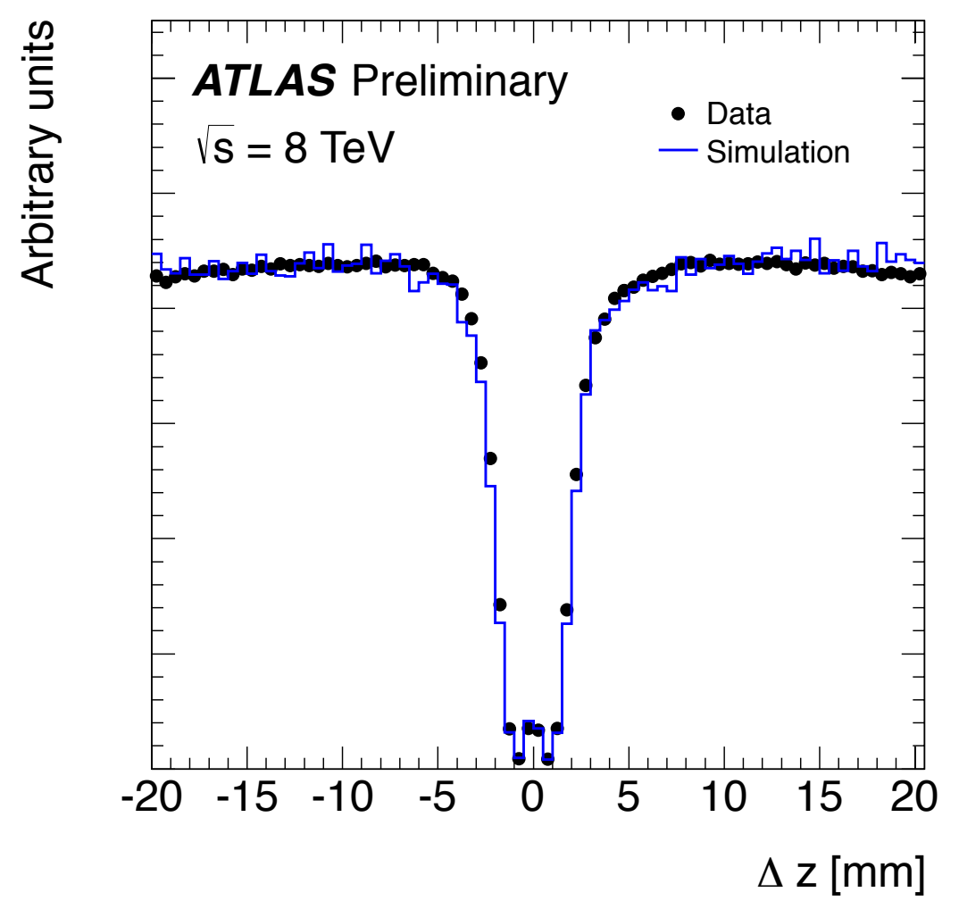
- Find reconstructed tracks passing a good track selection.
- Using these tracks, determine the location of a single seed.
- Fit nearby tracks to the seed. The fit is an iterative procedure, and in each iteration less compatible tracks are down-weighted and the vertex position is recomputed.
- Use the leftover tracks to find another seed and repeat.

Issues in Run II

In Run II, we expect a higher pileup environment with increased luminosity. As can be seen in the plots below (from our Run I results), **merging** is expected to be a major issue with higher pileup. This is when two or more truth vertices are reconstructed as a single vertex. As can be seen in the lower plot, vertices within about 5 mm of each other along the **z (beam) axis** are generally merged.



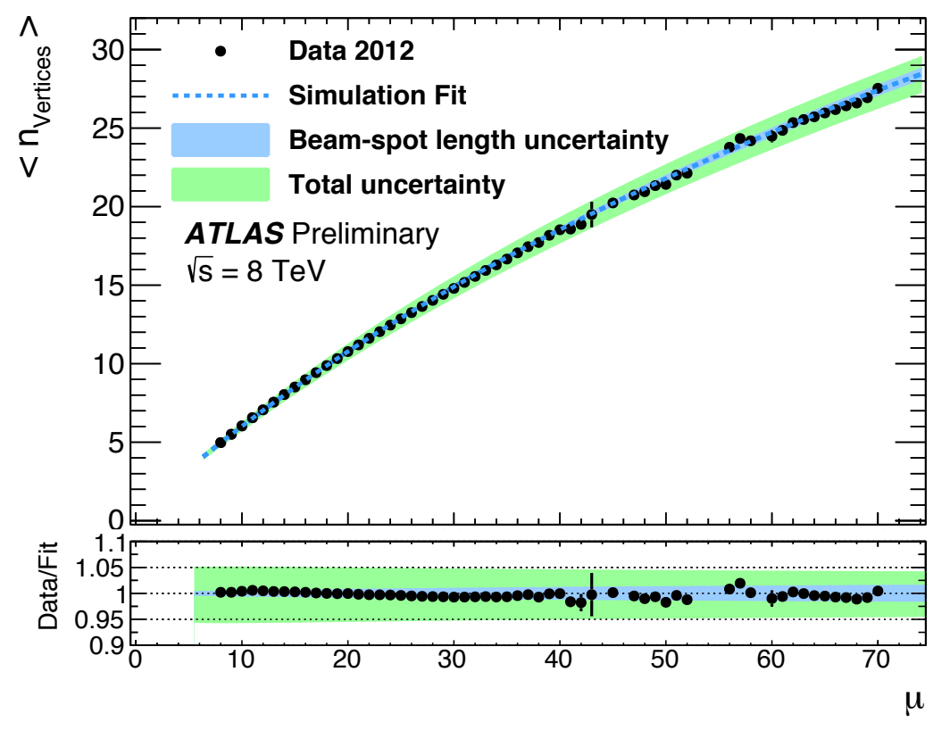




A new seed finding algorithm may improve this behavior in Run II.

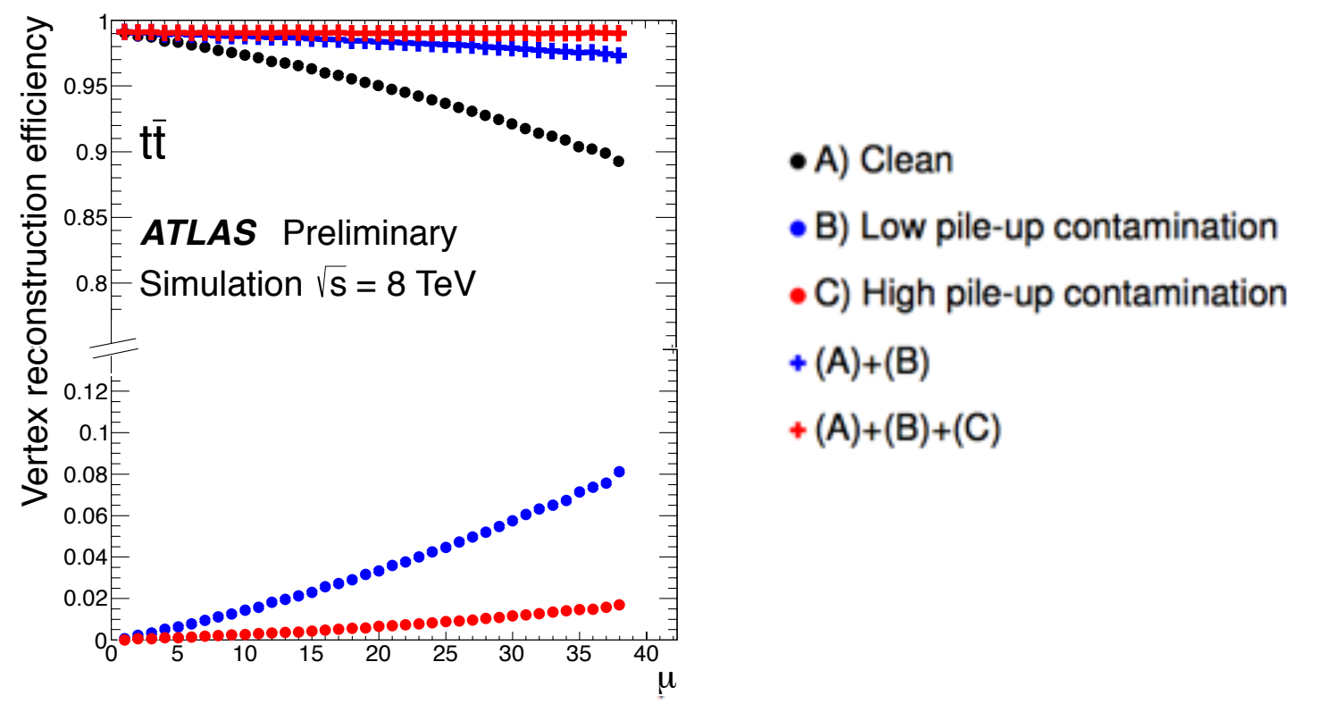
Current Algorithm Performance

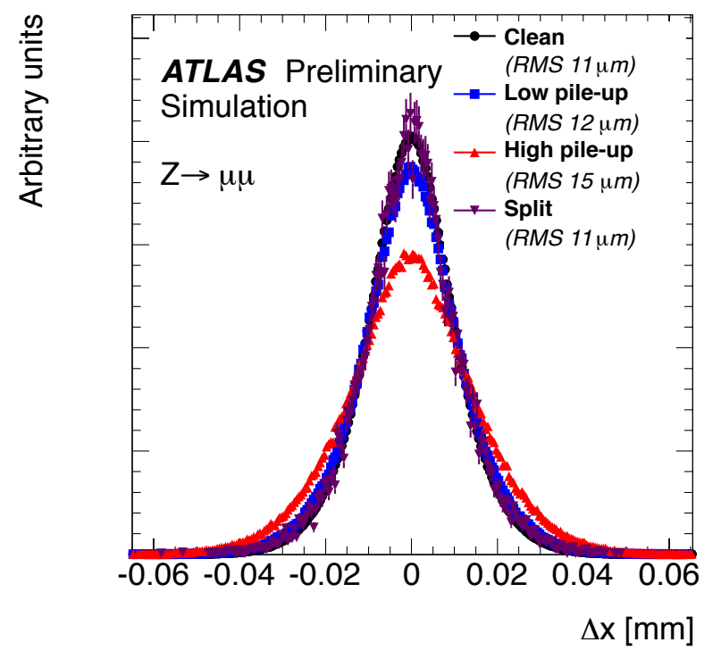
Here we show the average number of reconstructed vertices as a function of **μ, the average number of interactions per bunch crossing**. Up to μ=40, we reconstruct on average about 50% of the vertices. However, note that the curve is concave down. Vertices lost to merging impact the reconstruction performance more heavily at higher pileup.

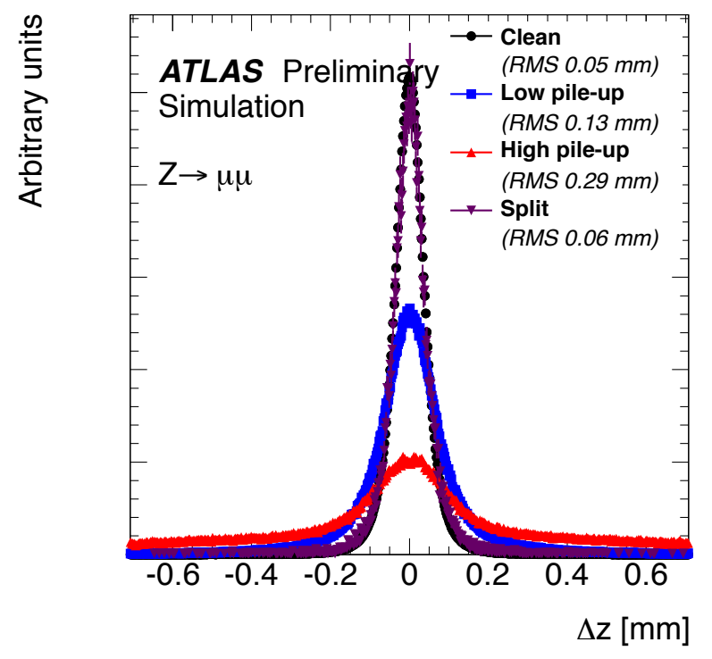


We look at the quality of reconstructed hard scatter vertices in the following plots. The vertices reconstruct with high efficiency, but losses due to merging increase with pileup. Furthermore, more contaminated vertices are reconstructed with less accuracy. The plots use the following definitions:

- Clean: Exactly one reconstructed vertex corresponding to the truth hard scatter vertex.
- Low pile-up: Exactly one merged vertex where hard scatter contributes more than 50% of the track weight.
- High pile-up: No vertex where the hard scatter contributes more than 50% of the track weight.
- Split: Two or more vertices with more than 50% weight.







Below, we compare the number of tracks in reconstructed vertices, with the iterative method performed on data and on MC. Agreement is fairly good, but slightly off at a high and low number of tracks.

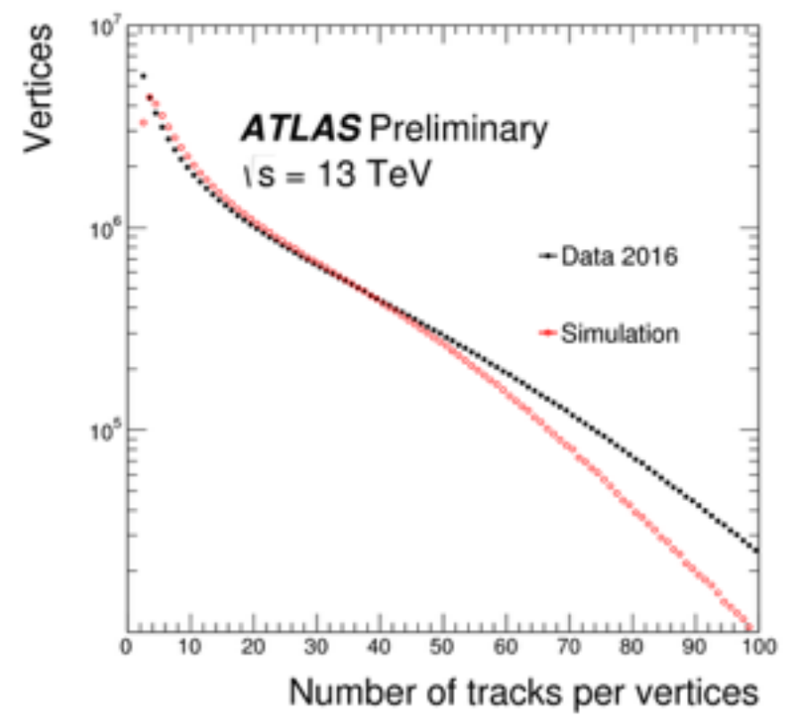
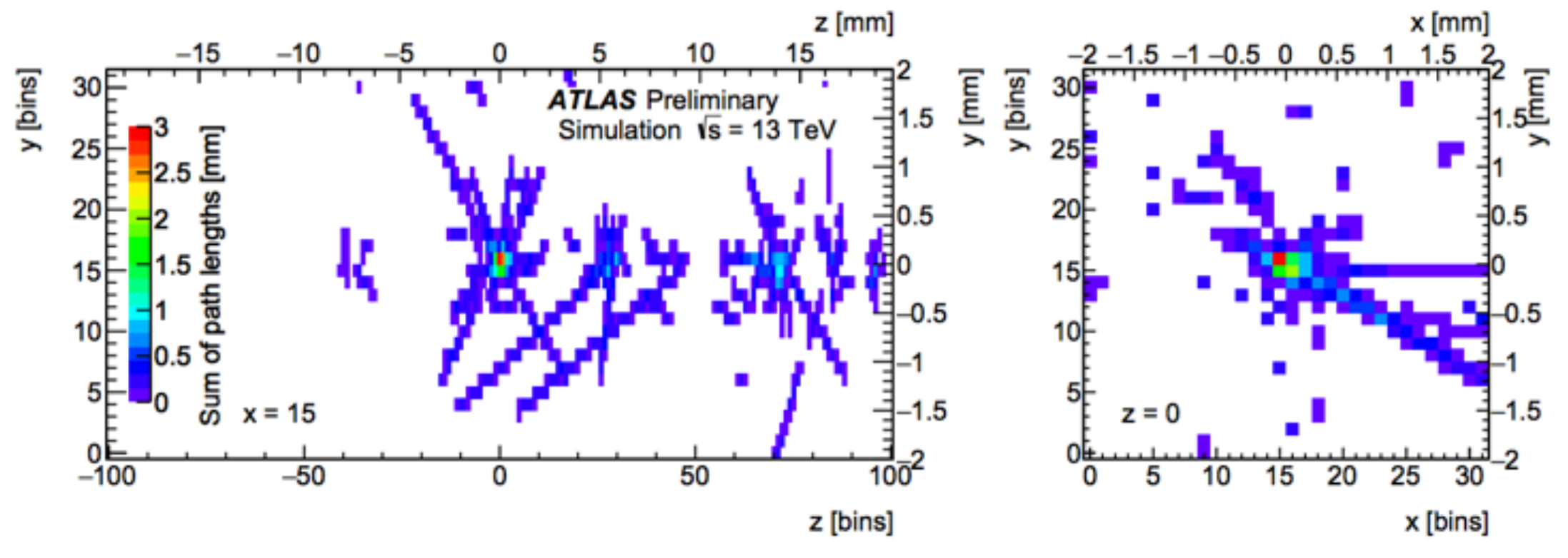


Image Seeding

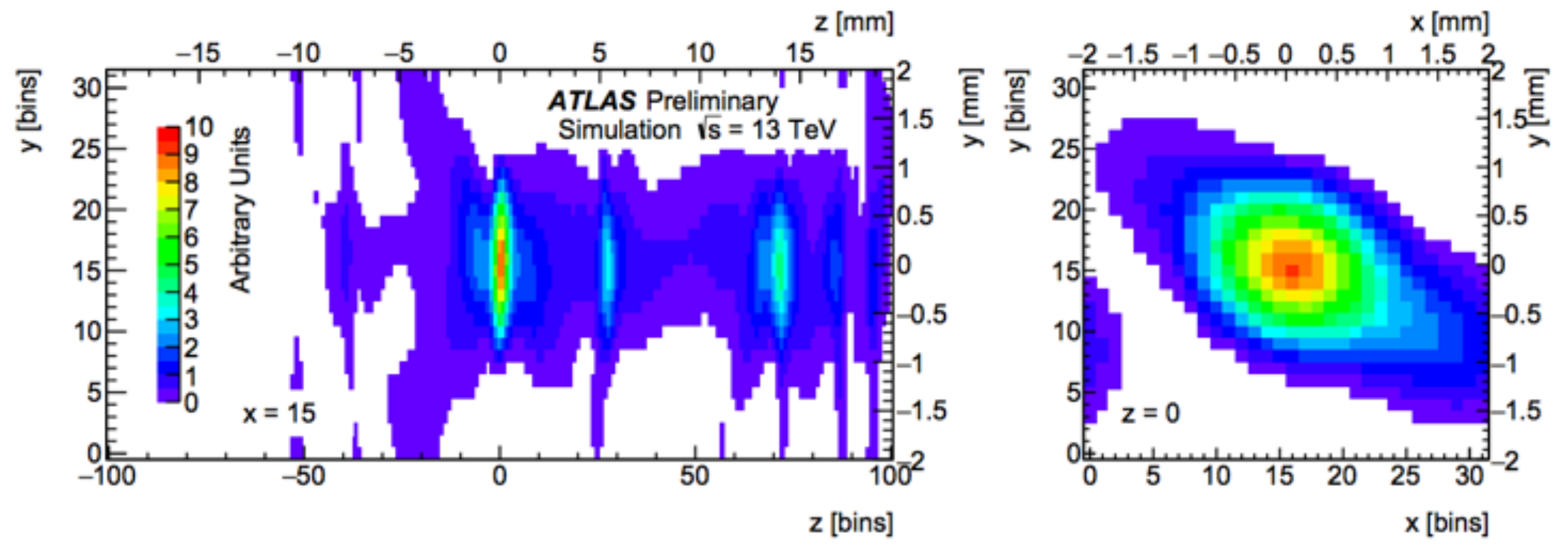
We are investigating a new method of seed-finding which is based on techniques used in medical imaging [1].

In this method, we take all tracks which pass a good track selection and use them to fill a 3D spatial histogram centered around the beam axis (figure a).

This histogram is sent through a FFT algorithm, a frequency filter is applied, then the FFT is reversed (figure b).

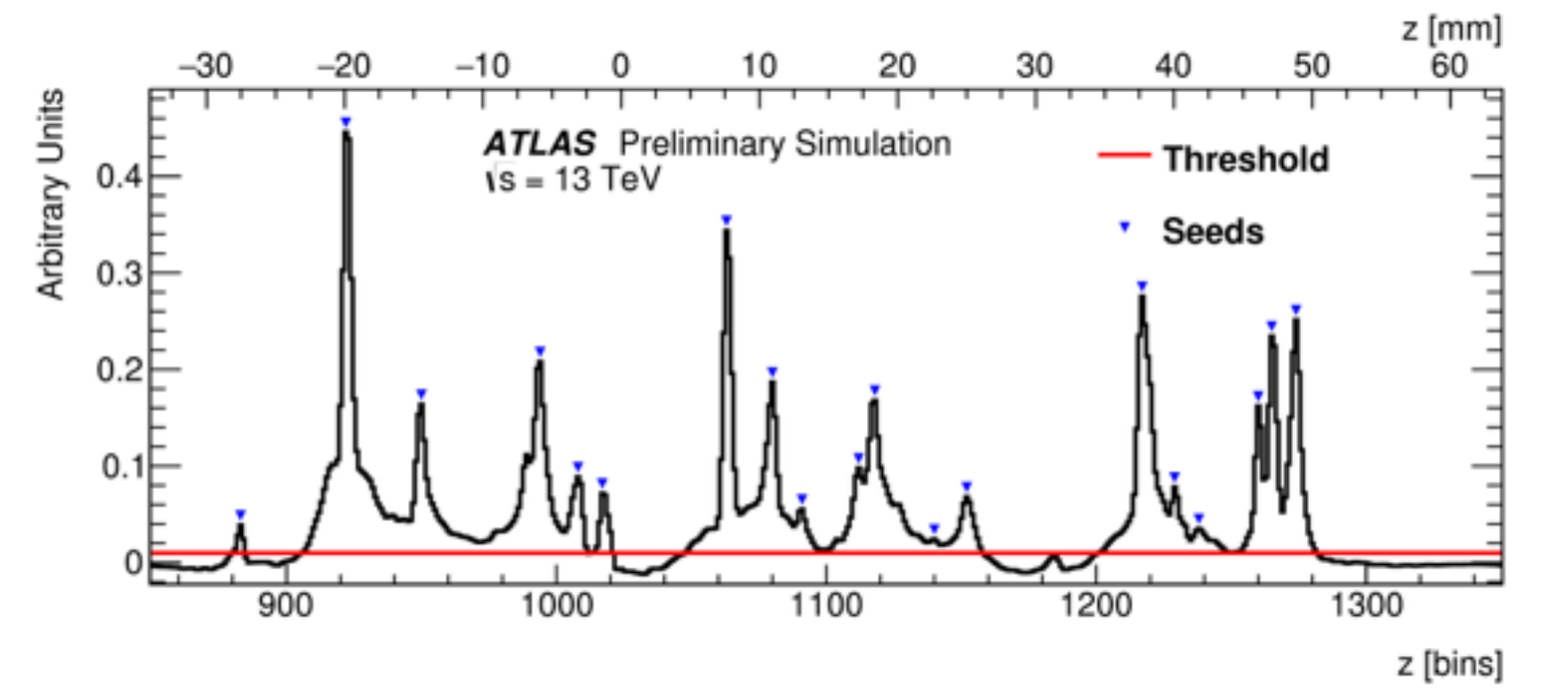


(a) Track back-projection



(b) Reconstructed image

This histogram is collapsed onto the z axis, with bins weighted by distance from the axis, and local maxima are taken as vertex seeds. Tracks are then associated with the seeds as usual.

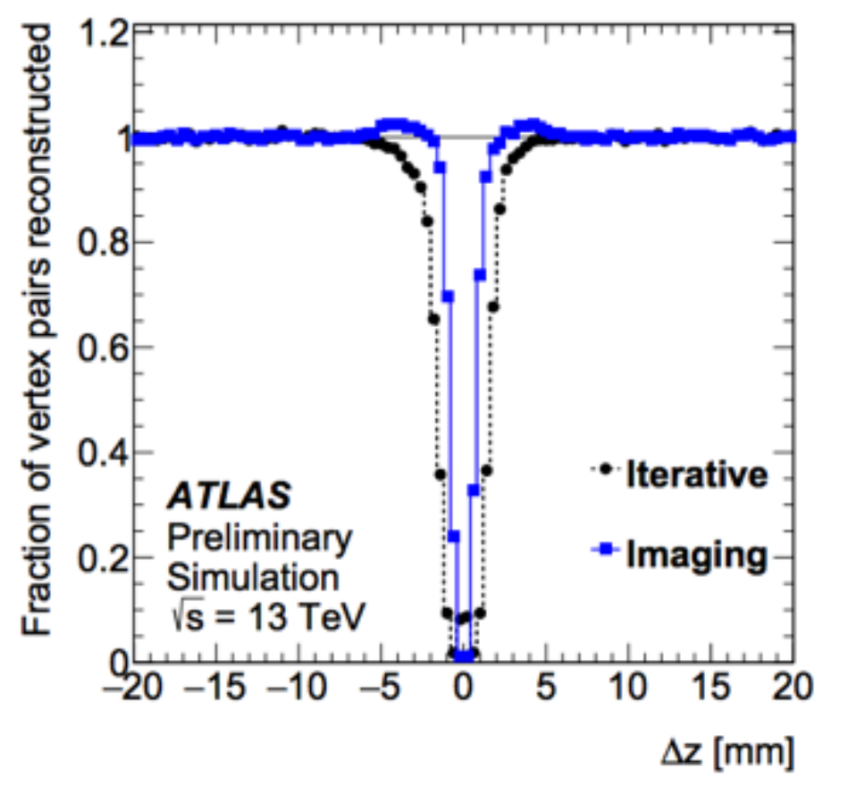


We are still investigating the effect of the new seeding method on vertex reconstruction. Due to updates in the old method, it is currently unclear how much improvement the new seeder provides.

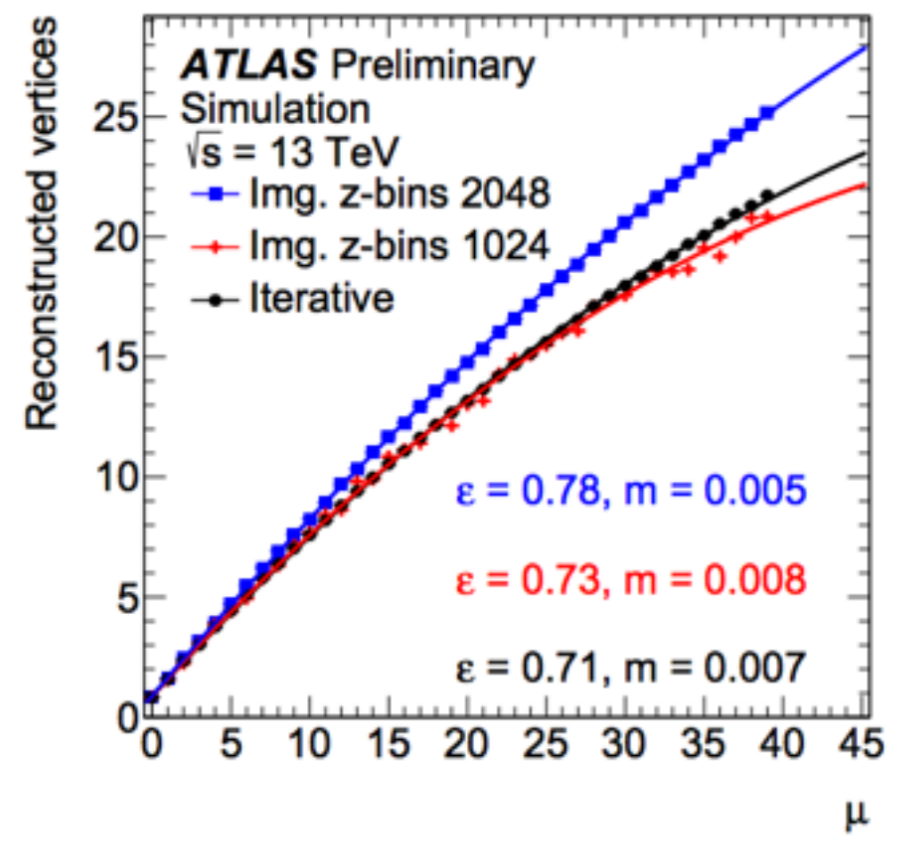
The new seeder can produce seeds with finer spatial resolution but with more **splits (one truth vertex reconstructed as two or more vertices)**. Our strategy is to use the imaging seeder to produce many split vertices, and then to perform an additional recombination step in order to put the pieces back together. We hope that this may reduce the number of merged vertices at the end.

Algorithm Comparisons

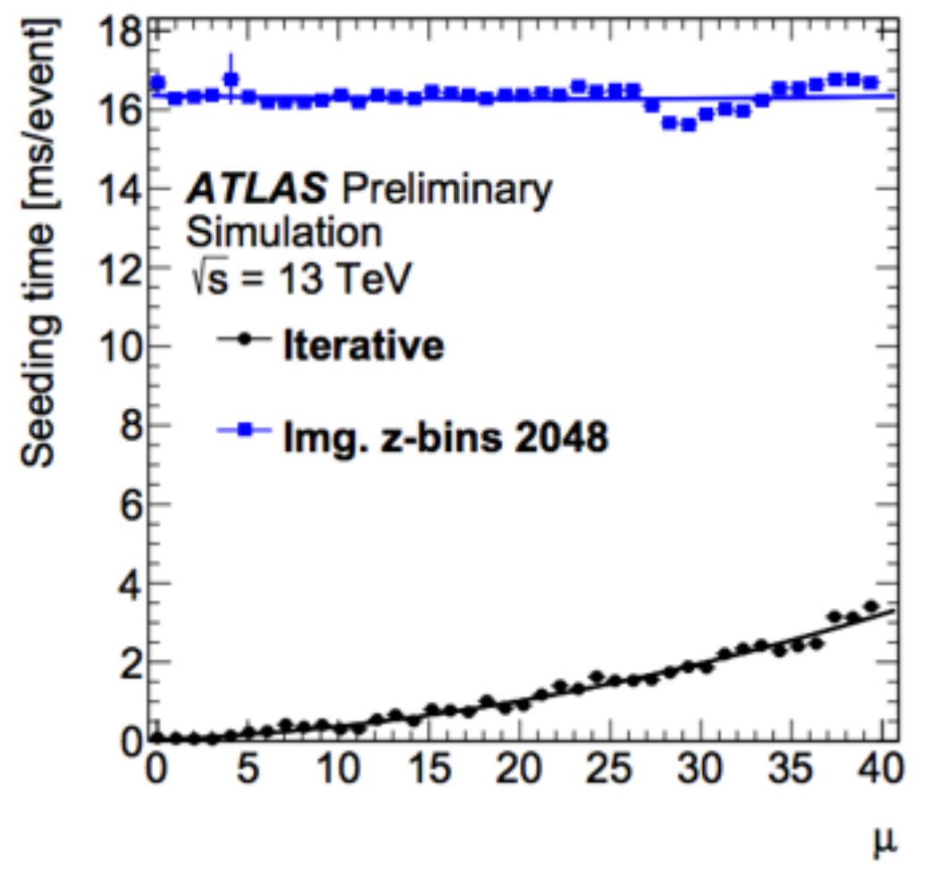
Here we compare the performances of the Run I iterative seeding method with the Run II imaging method. First we see that the imaging method can produce more finely spaced vertices. In the following plot, we see that while the old method has a resolving power of around 5 mm, the imaging method can get to around 2 mm.



The finer resolving power allows us to reconstruct more vertices. In the following plot we show the iterative seeder compared with the imaging seeder run with two different settings. When we use a coarse histogram binning, using 1024 bins in the z direction, we see that the imaging seeder performs about as well as the iterative seeder. However, when we increase the resolution, using 2048 bins in the z direction, we manage to reconstruct many more vertices. The fit lines in this plot use the equation **$N_{vtx} = \varepsilon*\mu*(1-m*\mu)$**.



Increasing the binning further leads to even finer resolving power. However, this also leads to more split vertices. Our current goal is to recombine the split vertices in such a way as to minimize both splitting and merging. Unfortunately, increasing the binning also greatly increases computation time. See the plot below for a comparison. One method of overcoming this problem may be to use sparse matrices in the imaging seeding calculations.



---

[1] **An imaging algorithm for vertex reconstruction for ATLAS Run-2** ATL-PHYS-PUB-2015-008