Low-Pt Tracking for ATLAS in Nominal LHC Pileup

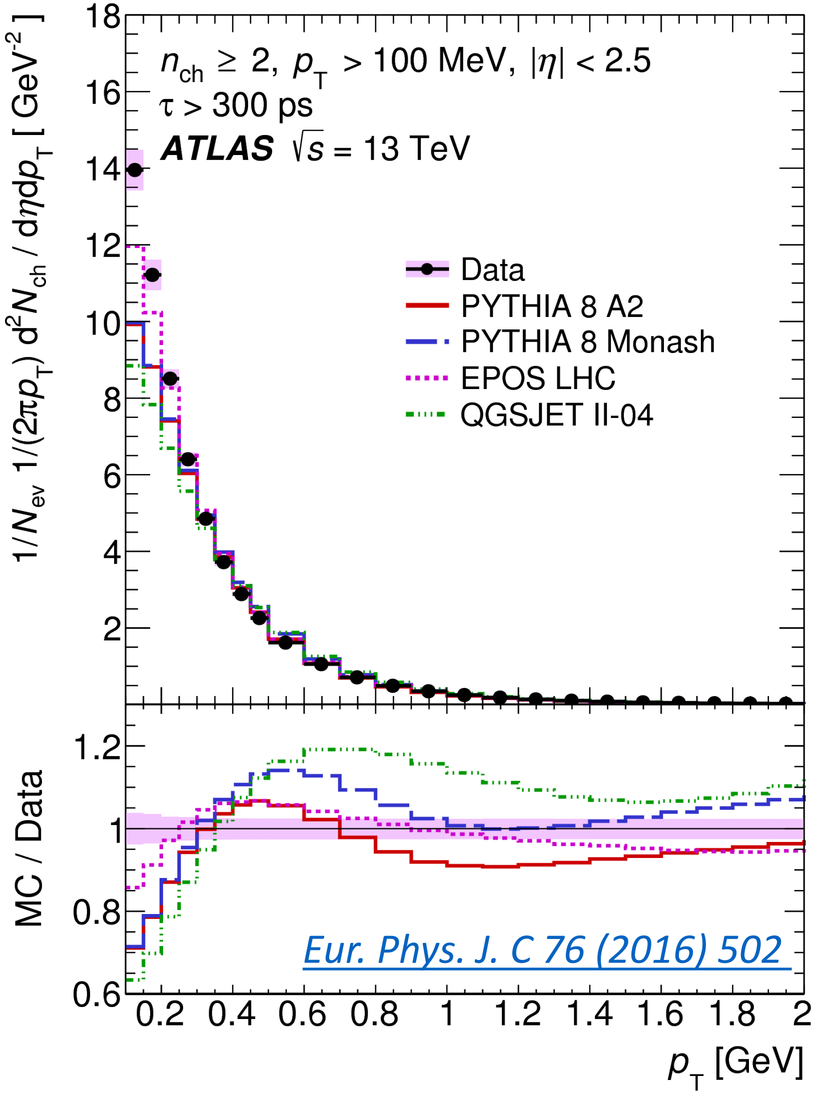
Patrick McCormack on behalf of the ATLAS Collaboration

April 20, 2020

Connecting the Dots

ATLAS Tracking

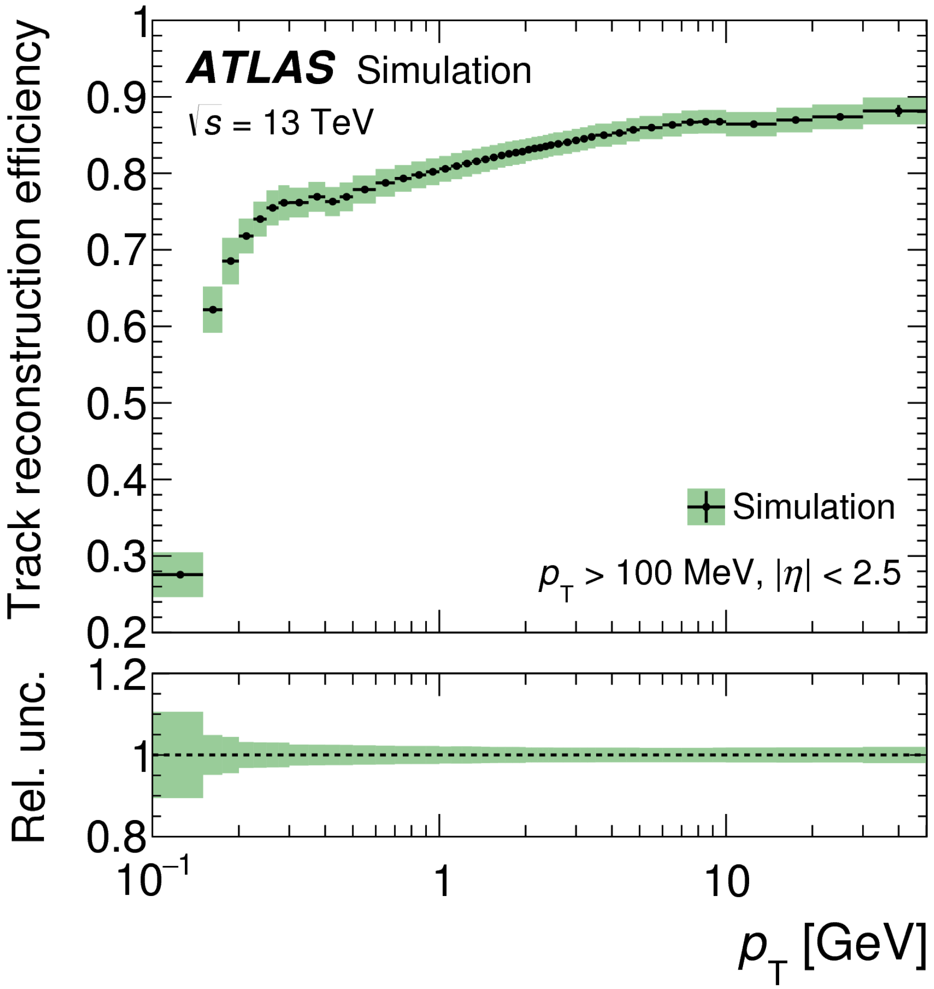
- Current ATLAS tracking framework attempts to reconstruct particles with p_T down to 500 MeV
- The particles from pileup interactions and from the underlying event (UE) of high-q^2 pp interactions at the LHC are in fact MOST LIKELY to have p_T< 500 MeV
  - For most analyses, these particles are not interesting
  - There is also a steep reconstruction time and storage cost for incorporating low-p_T tracks



Stable primary charged particle multiplicity distribution as a function of particle p_T (for particles with p_T > 100 MeV).

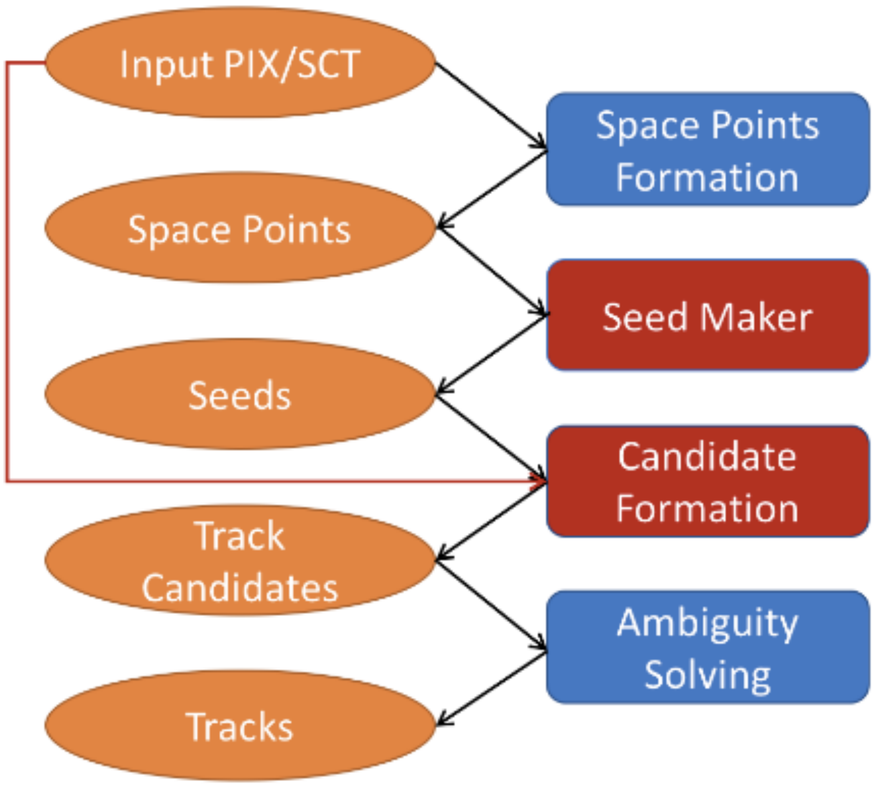
Why reconstruct low-p_T tracks?

- That being said, there *are* ATLAS analyses where low-p_T track information can be useful
  - Searches for photon-induced physics can use better reconstruction of an UE to help distinguish between photon- and parton-induced interactions
  - Charm tagging can be improved, as D meson decays often result in low-p_T tracks
  - Some SUSY models predict low-p_T tracks (e.g. small chargino-neutralino LSP mass splitting)
- ATLAS *has* reconstructed tracks with p_T down to 100 MeV in the past, but that was done in low pileup (PU) conditions (~1 pp collision per bunch crossing)
  - In order for an analysis to use low-p_T tracks in nominal data-taking conditions, **the implementation must work in high PU condition** (up to 70 pp collisions per crossing)



Implementing low-p_T tracking (for high PU)

- High PU results in a large number of detector hits in each layer of the ATLAS Inner Detector (ID)
  - Seeds for tracks are formed from combinations of 3 pixel-layer hits or 3 SCT hits (and sometimes an extra confirmation hit)
  - High-p_T charged particles curve relatively little, so seed finding is a question of finding three hits that form a line in space (or small deviations from a line)
  - Going to lower p_T opens up a large deal of phase space for the curvature of seeds resulting in a large combinatorics problem- now can combine hits that are significantly non-linear
- To mitigate this problem, run tracking in two passes: first do nominal tracking, then *remove hits used by these tracks*, and then perform low-p_T tracking with these hits as input
  - Low-p_T tracks have slightly relaxed hit requirements

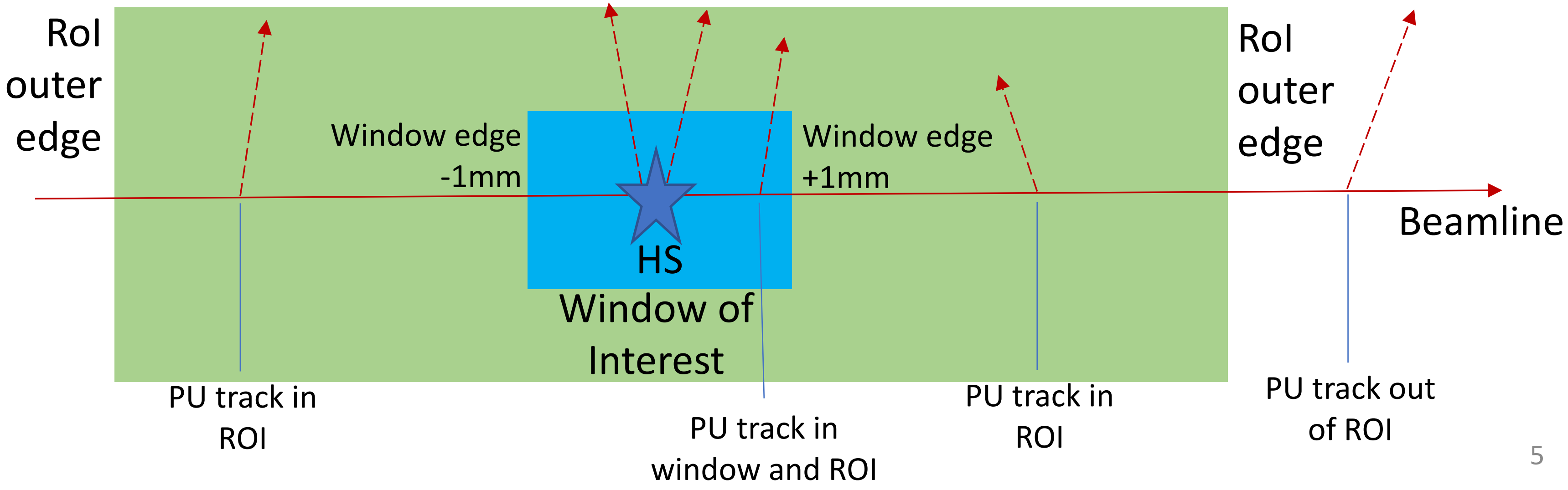


Hits used by nominal track ARE NOT in the input pix/SCT for the low-p_T

Implementing low-p_T tracking (for high PU)

- An additional trick to reduce combinatorics has also been implemented: only consider track seeds pointing into a region of interest (RoI)
  - RoI position can be chosen from reconstructed objects (data and MC) or from a truth-level position in MC: no need to find low-p_T tracks far from collision of interest
  - Iteratively find seeds and reject them if pointing away. Only for the successful ones extend and move to ambiguity solving
  - RoI size is configurable- choice will be explored in coming slides

Tracks are reconstructed for seeds that point into the RoI (green region). However, the tracks that are **most relevant for analysis** are those near the hard scatter (HS), in what I will call the "window of interest" (blue region). Most of the following plots are only of particles or tracks in the blue region, which we typically take to be $\pm$1mm from the HS for tracks.



Performance metrics

1. Efficiency of reconstructing charged particles (truth-level objects)

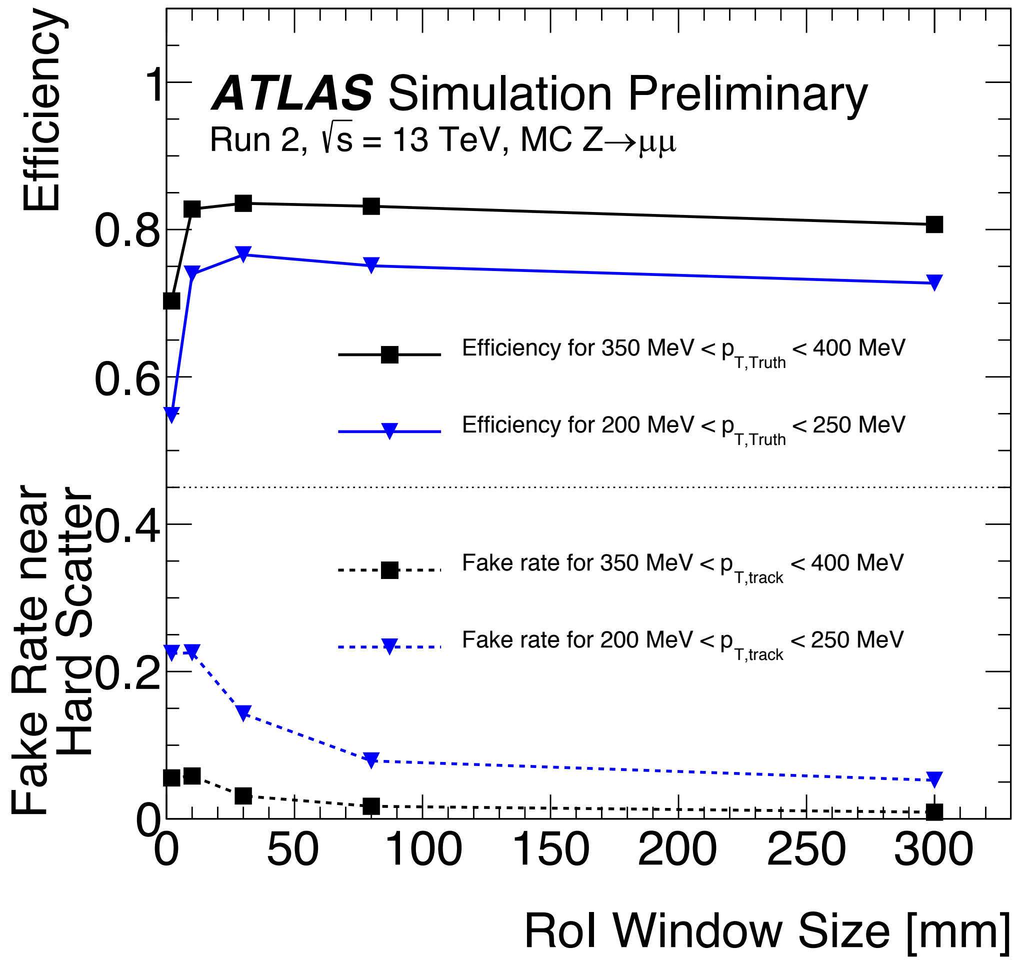
$$eff = \frac{\#charged\ particles\ with\ at\ least\ one\ reconstructed\ track}{\#charged\ particles}$$

2. Fake rate: a fake track is one constructed from detector hits created by multiple different charged particles- i.e. it does not correspond to any truth level object's trajectory

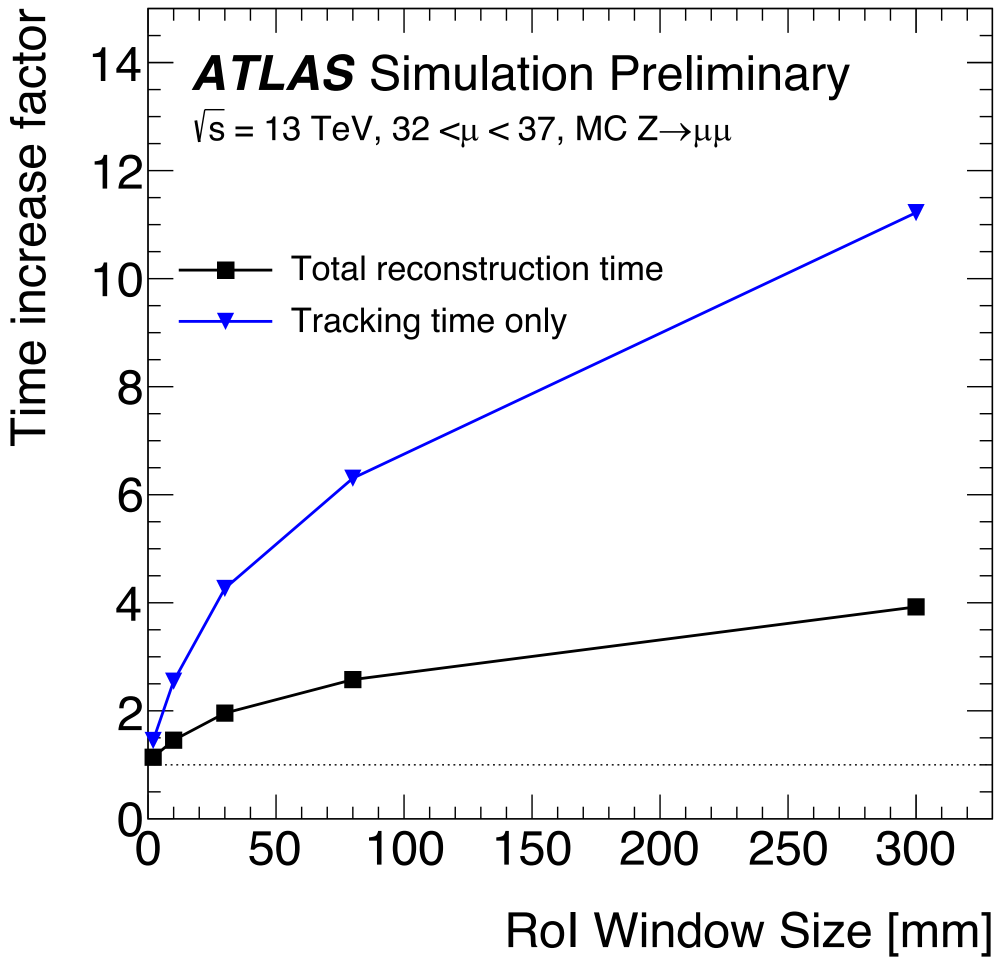
3. Reconstruction time. How much longer does the reconstruction take when low-$\mathrm{p_T}$ tracking is included?

- We will use these metrics to determine an optimal RoI size and to evaluate the performance of our algorithm

Determining RoI size



Efficiency and fake rate near HS vs RoI size for two p_T slices of RoI tracks



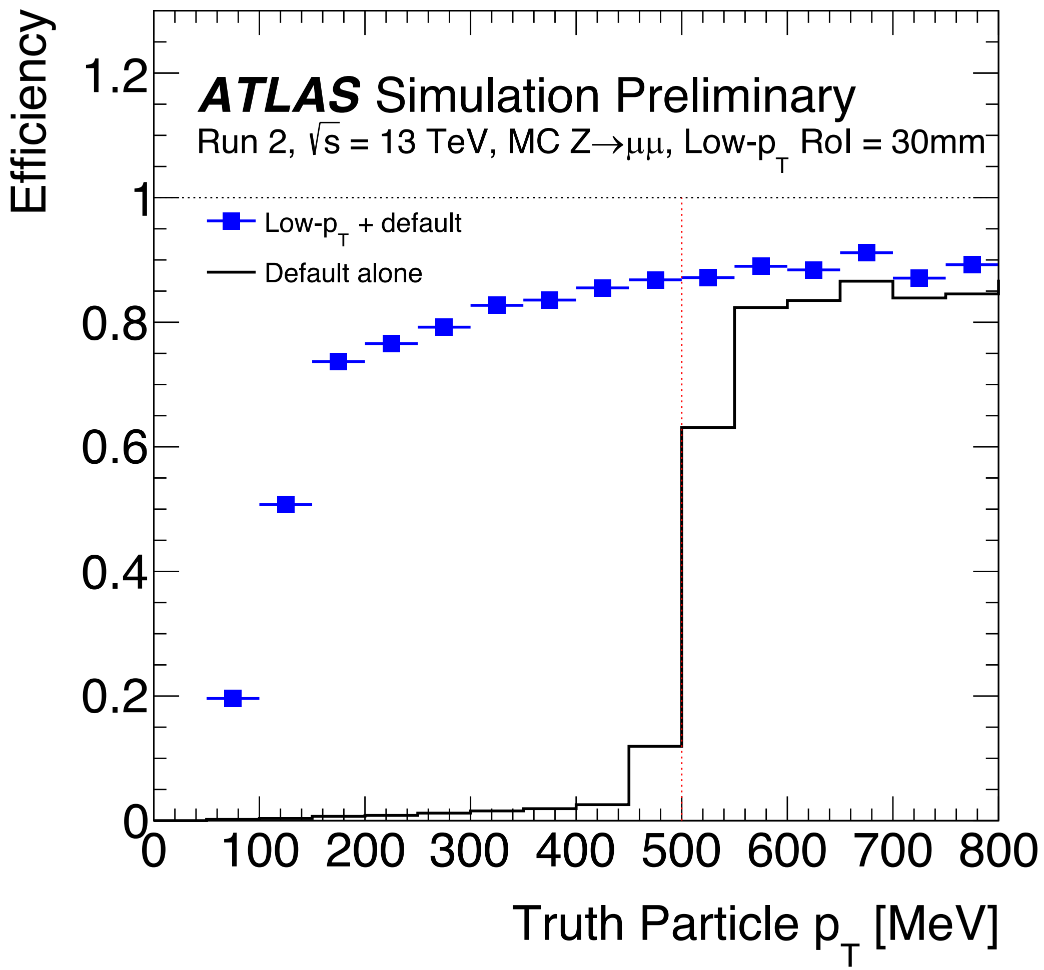
Total time increase and time increase of the tracking component of reconstruction alone vs RoI size. The factor is taken as a multiplicative factor over reconstruction without low-p_T tracking.

Selecting an RoI size

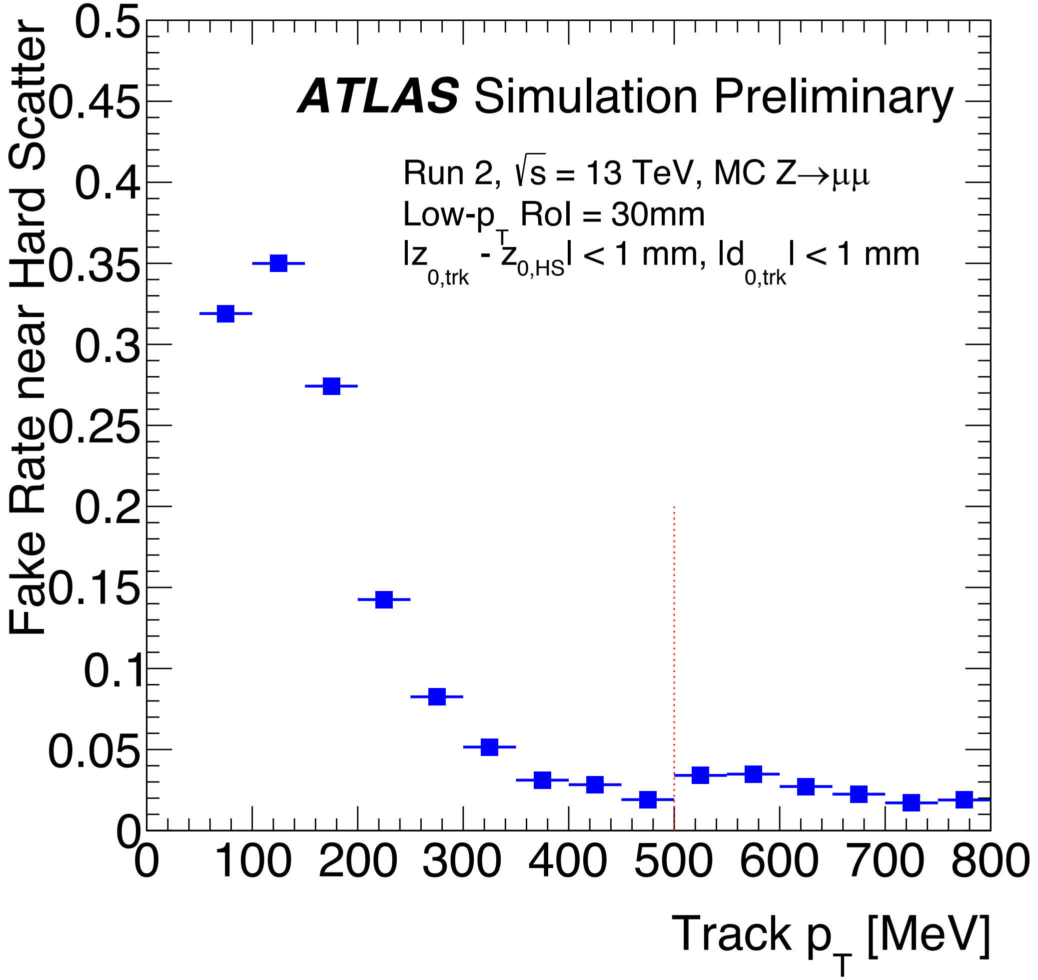
- A small RoI (< 10 mm) **is fast** but has lower efficiency and **high fakes**
- A large RoI (>100 mm) has **low fakes**, has generally lower efficiency, and is **time intensive**
- A medium RoI size (~30 mm) has **close to maximal efficiency**, at the cost of medium fakes and a moderate reconstruction time impact

- We adopt 30mm as a tentative default, though this is an adjustable parameter when the algorithm is called
- Now, how do efficiency and fake rate perform as a function of p_T?

Efficiency and fake performance



Efficiency for reconstructing charged, stable, non-lepton truth particles as a function of truth particle p_T. Steep dropoff in performance below 200 MeV.



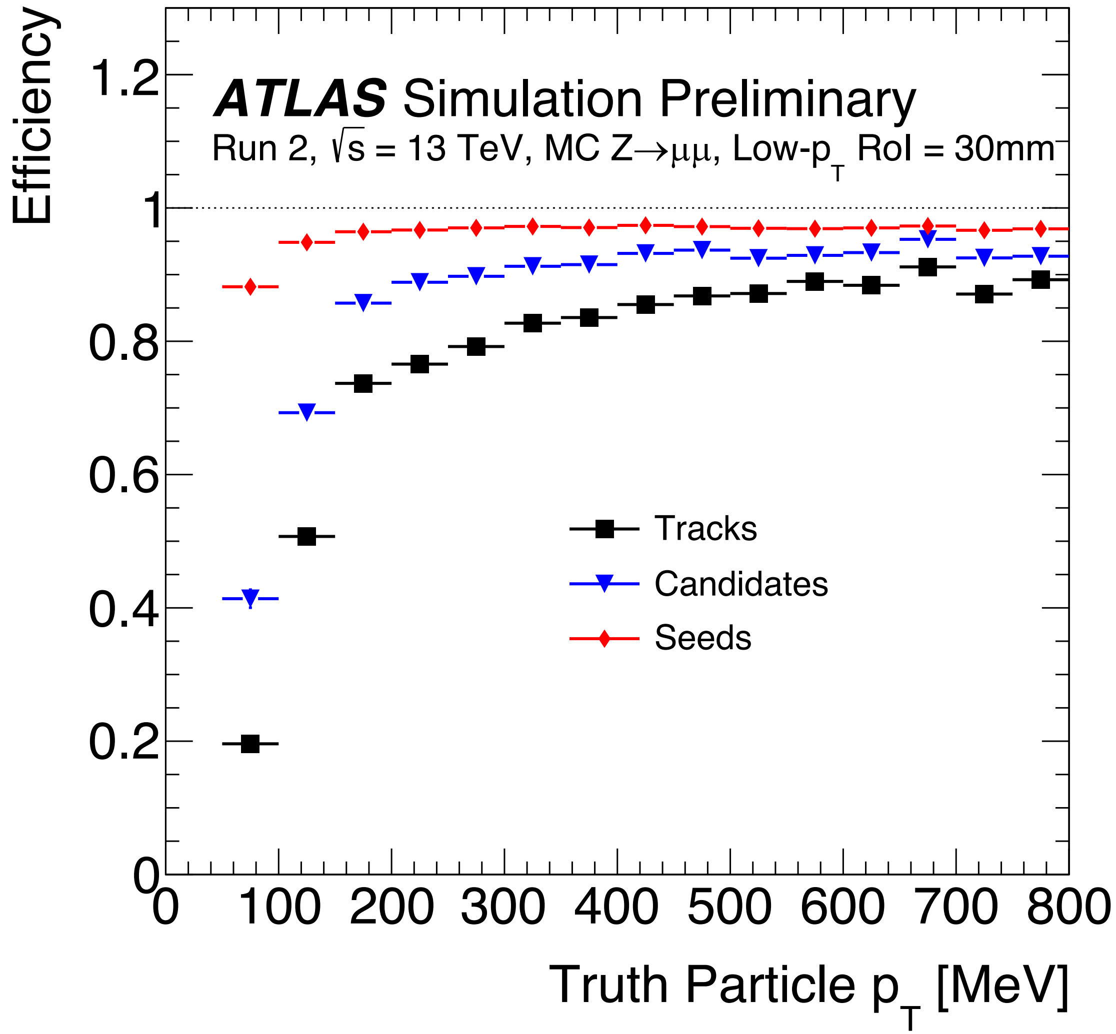
Fake rate for tracks within 1mm of the HS vs the track p_T. Relatively steep increase as p_T decreases.

Remarks

- The efficiency decreases slightly but is fairly stable down to ~200 MeV
  - Steep dropoff for extremely low-p_T particles
- Fake rate stays below 10% for tracks down to ~250 MeV
  - Below 200 MeV, ~1 out of every 3 of tracks are fakes
- **How could we improve efficiency and decrease fake rate?**
  - ATLAS tracking is performed in three stages
    1. Create “seeds” from 3 space point (either 3 pixel hits or 3 SCT hits)
    2. Extend seeds through additional hits using Kalman filter to create “candidates”
    3. Perform ambiguity solving on candidates to create final tracks (want only 1 track per particle!)
  - To increase efficiency, you want more seeds (or perhaps looser ambiguity solving)
  - To decrease fake rate, you want fewer seeds (or perhaps tighter ambiguity solving)
- Let’s examine where efficiency is being lost and our seed quality

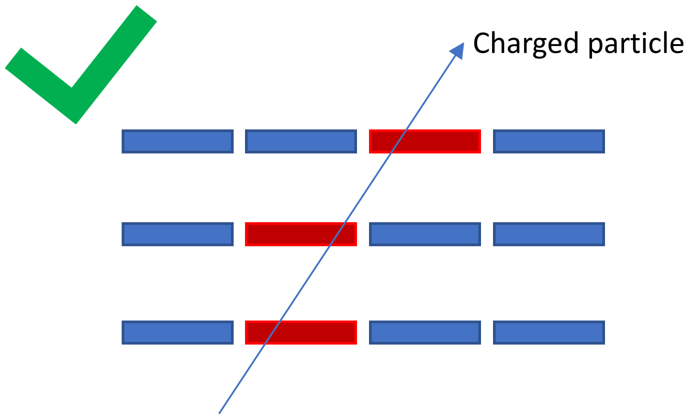
Where is efficiency lost?

- Here we plot the probability for a charged, stable, non-lepton truth particle to have at least one track/candidate/seed as a function of the truth particle p_T
  - Most particles have at least one seed
  - About half of the seed-to-track loss is lost in the candidate stage, except at very low p_T, where loss is more dramatic
- Does this mean we should just accept a higher fraction of seeds to increase our efficiency?
  - Why is the seed-to-candidate drop worse at lower-p_T?

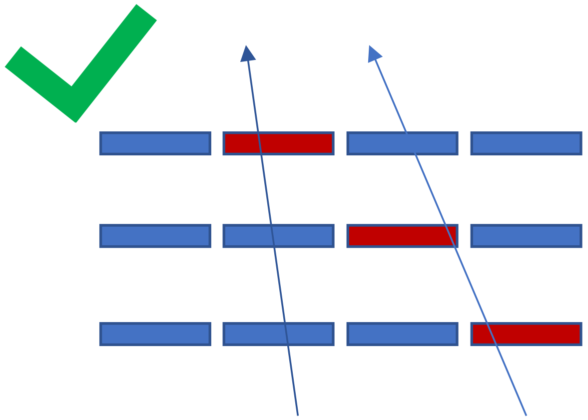


Examining seed quality

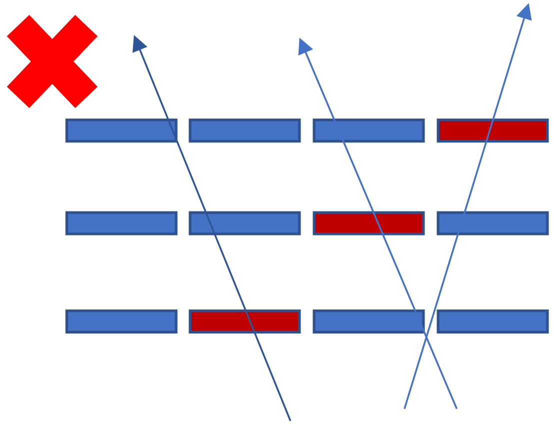
- We will use "truth match probability" as the primary metric for seed quality
  - This is weighted fraction of hits on a track created by the same truth particle
  - Recall that seeds are typically formed from hits on 3 pixel layers or 3 SCT layer (with an extra hit or two possible as "confirmation hits")
  - If all three hits are from different truth particles, truth match probability = .333
  - Typically consider a seed fake if fewer than half of hits come from the same particle



"TRUE": Seed formed from hits in red has truth match prob of 3/3



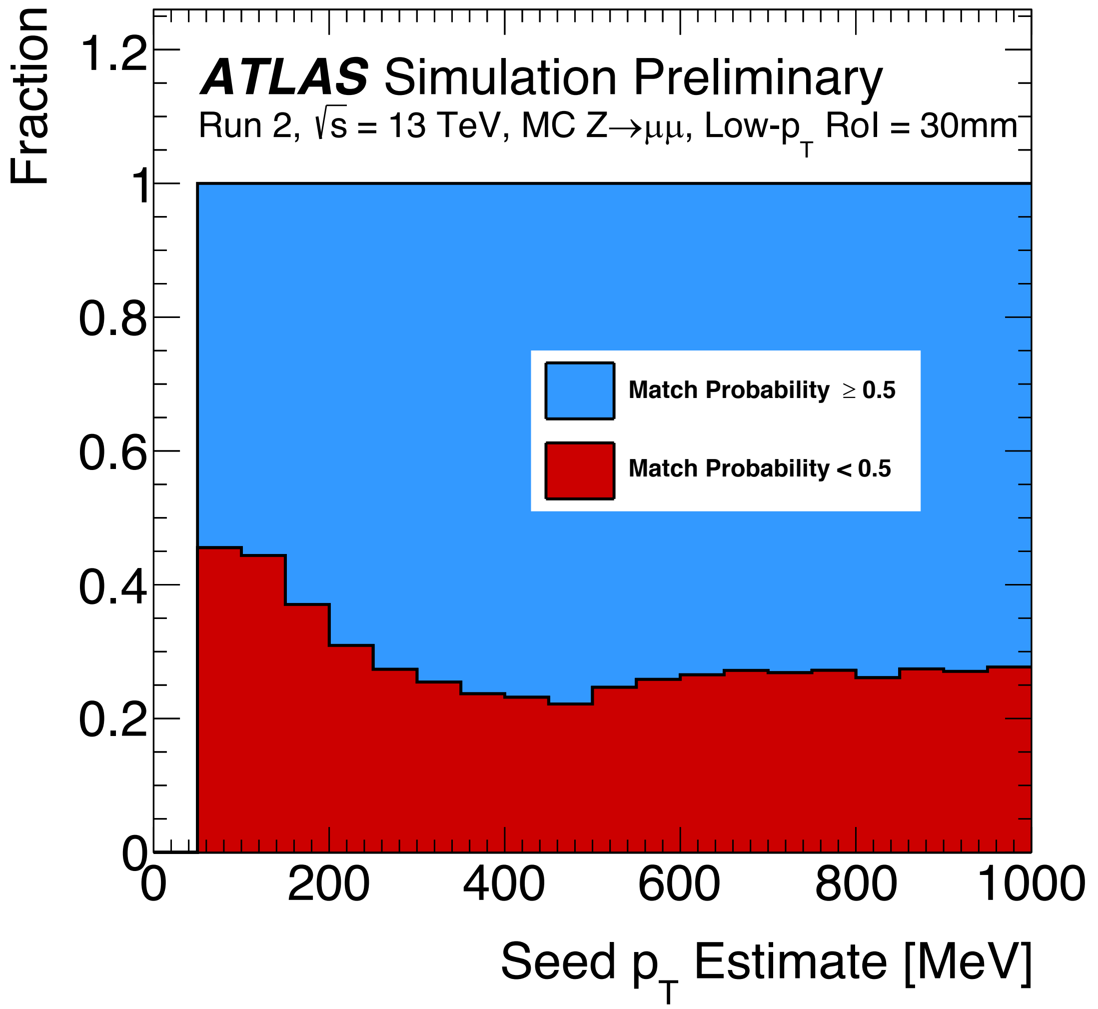
"TRUE": Seed formed from hits in red has truth match prob of 2/3



"FAKE": Seed formed from hits in red has truth match prob of 1/3

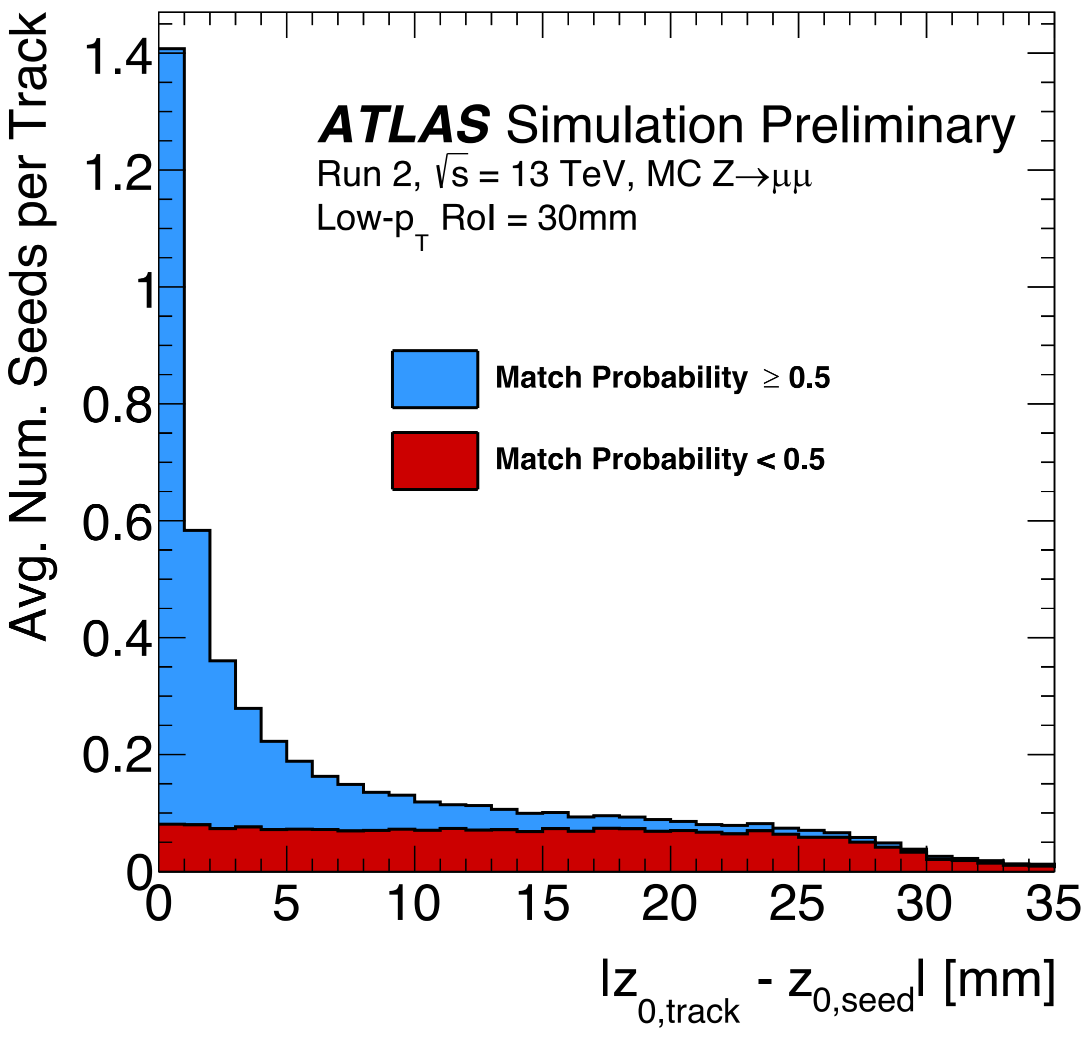
Examining seed quality

- Could we increase efficiency by loosening seed-to-candidate selection?
  - If the seeds are generally poor quality, this is not necessarily the case though
  - If truth match prob is low we don't necessarily expect the seed to be able to be properly extended through hits on other layers to make a candidate
- There is a high fake rate for very low-p_T seeds; loosening selection requirements here will increase fake rate



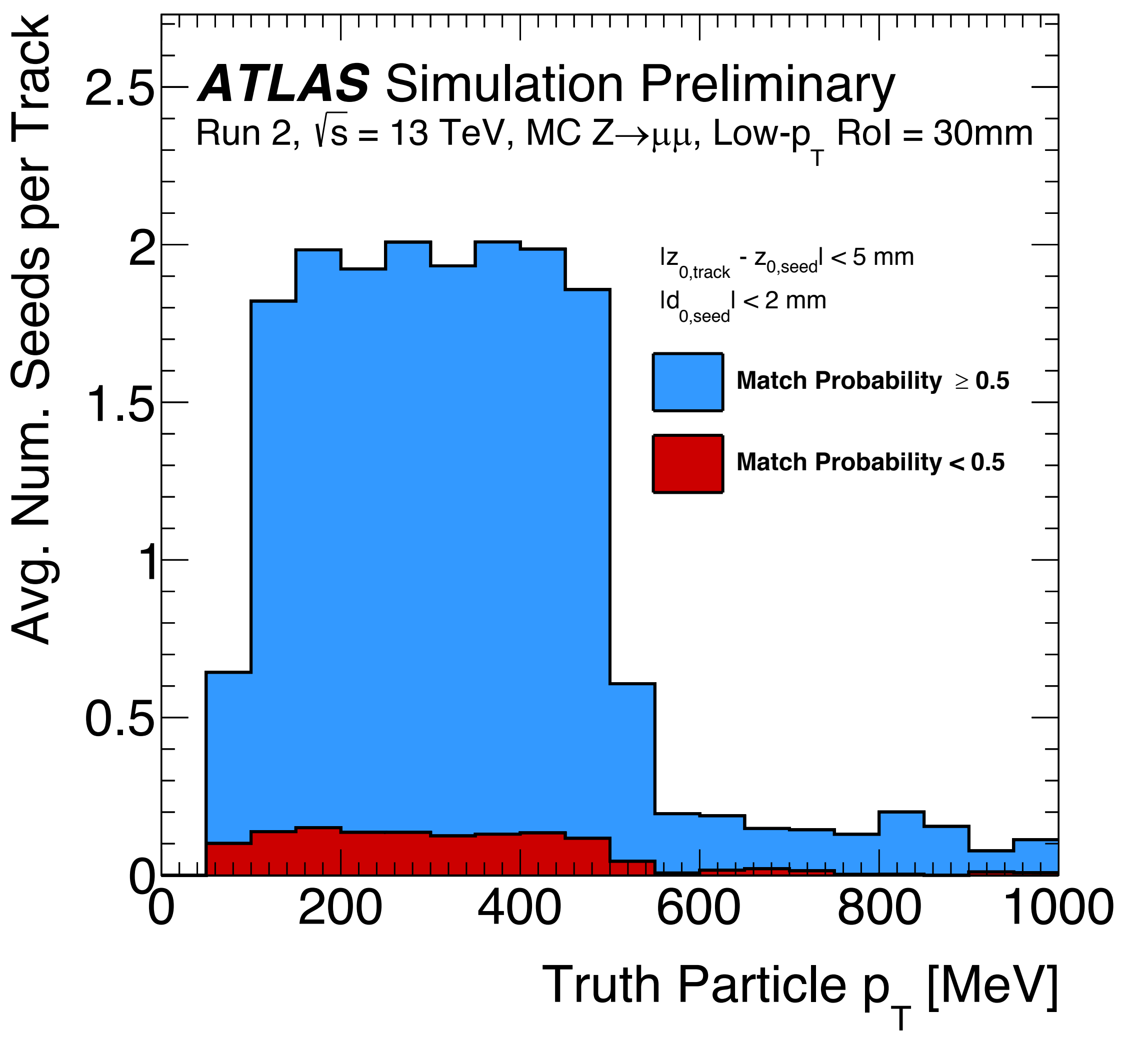
Seed resolution and number of seeds per track

- Conversely, to decrease fake rate, we would want to tighten seed-to-candidate-to-track selections. Do we have enough seeds per track to consider doing this?
- On the right: plot of $|z_{seed} - z_{track}|$ for tracks coming from low-p_T tracking pass
- Fake seeds can point almost anywhere, but we DO see that low-p_T tracks typically have ~2 good seeds pointing within 5mm along the beamline
  - Note that this is seeds per **track**, not per truth particle- i.e. the track *exists* for seeds considered here
  - This is not meant to imply that we should be highly efficient at track level



Seed resolution and number of seeds per track

- Given the seed "resolution" plot on the previous slide, let's restrict ourselves to look at seeds with $|z_{seed} - z_{track}| <$ 5mm and seeds with a transverse displacement from beamline of < 2mm
  - How many of such seeds per track do we see per track?
  - Interestingly, we see that for truth particles that DO have a track, there's typically about 2 good pointing seeds per track
  - Perhaps this means there is a little wiggle room for reducing fake rate



Moving forward

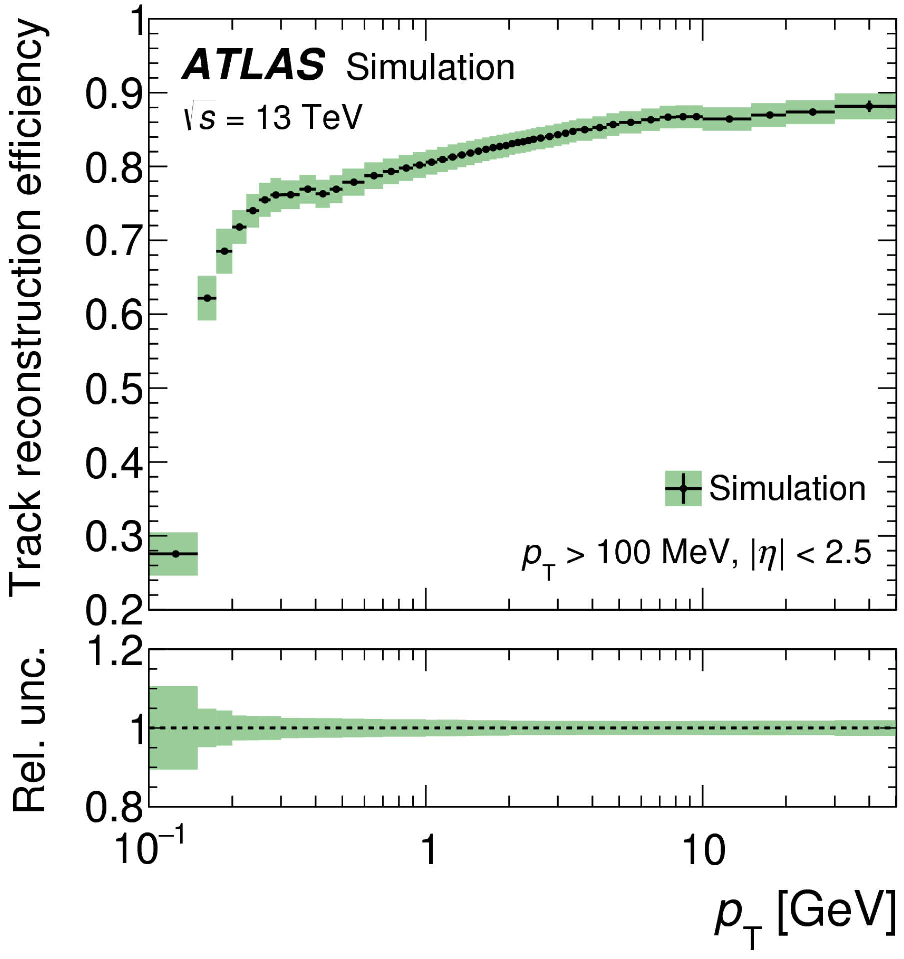
- Increasing the efficiency for very low-p_T particles would be difficult
  - There is a large combinatorial background here, where almost half of seeds are comprised of hits of random origin. Relaxing seeding selections here would only make this worse
- Reducing the fake rate might be possible by tightening selections
  - This could be done at reconstruction level, potentially improving reconstruction time too
  - Alternatively could also be done offline at the track level
    - E.g. make some set of track selections based on track p_T, η, number of pixel hits, number of SCT hits, number of holes, etc.
    - Such selection can decrease the fake rate, but will have a strictly negative effect on efficiency
  - Balance between fake rate and efficiency is somewhat analysis-specific question

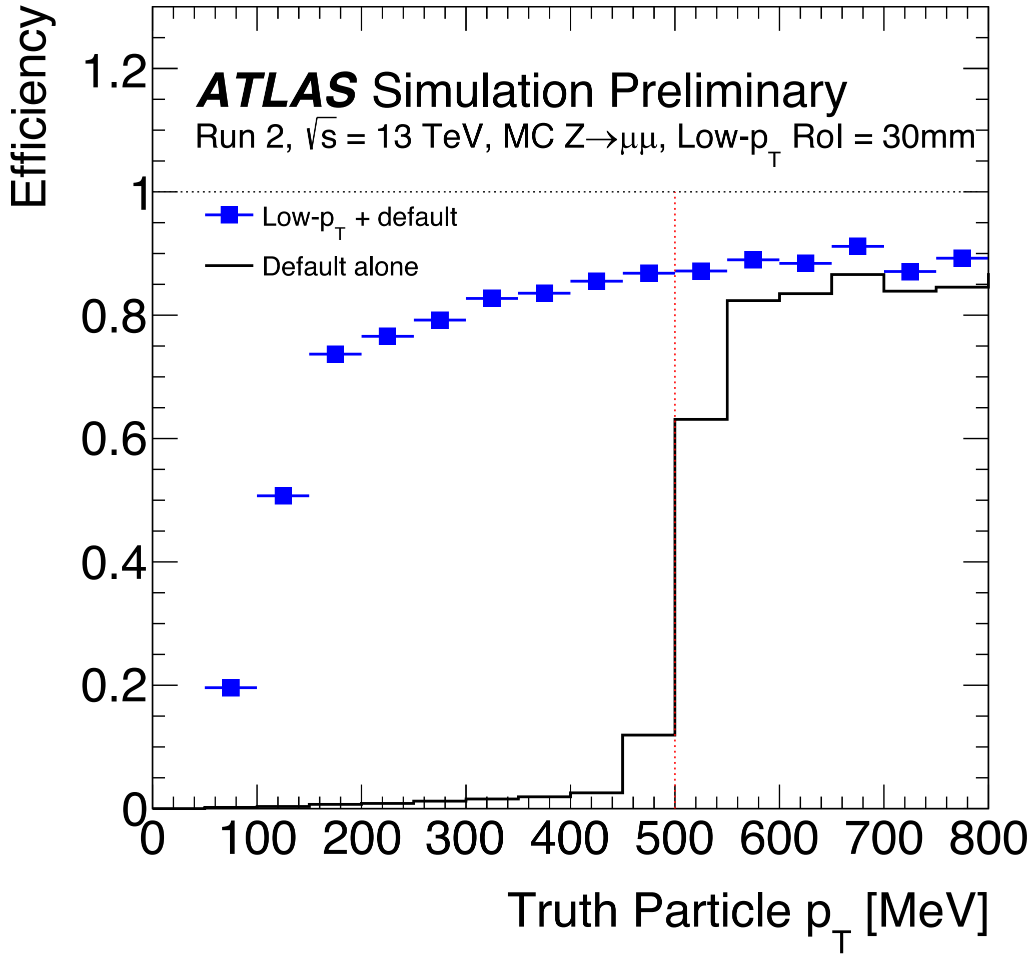
Conclusions

- Low-Pt tracking can be a useful tool for many analyses
  - We have created a framework in the ATLAS software that can be applied on events of interest to specific analyses, making it very portable
- Extending tracking down to ~200 MeV can actually be done in generic run 2 pileup conditions with relatively high efficiency and low fake rate
  - In the very-low Pt regime, fake rate tends to become significant, but this can likely be mitigated with offline selections
- Impact on reconstruction time is not totally negligible, but there's room for improvement by tightening seed selection criteria

Backup

Compare MinBias tracking efficiency to ours

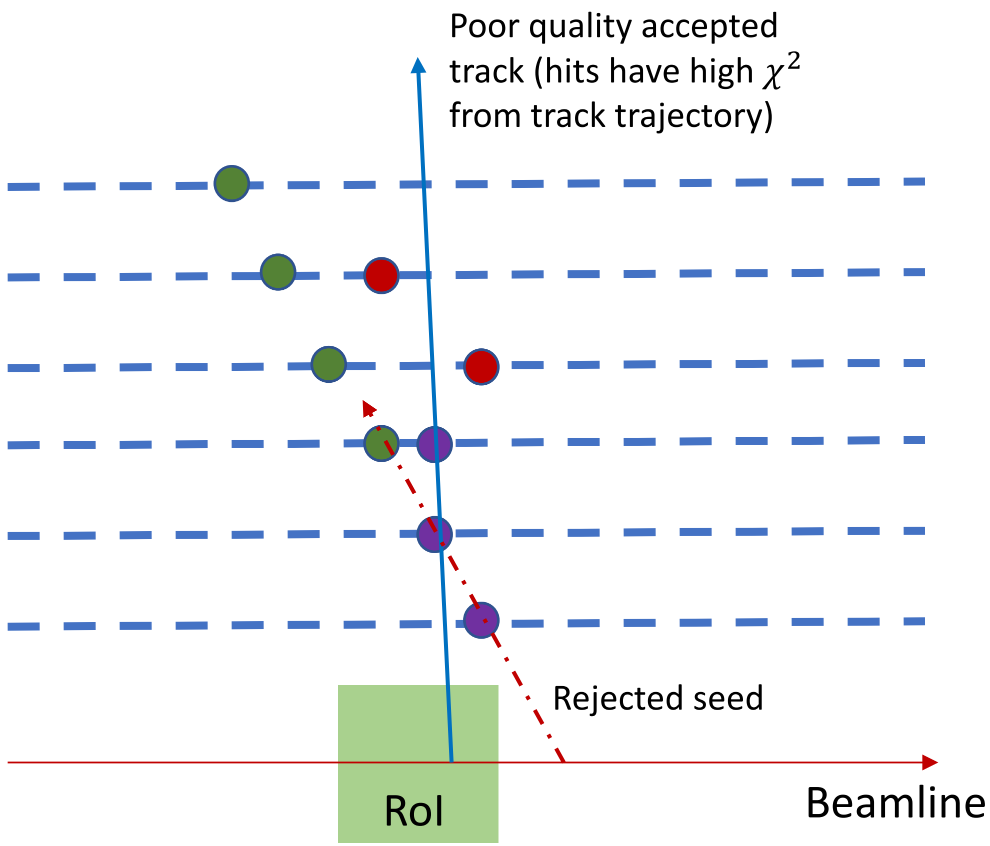




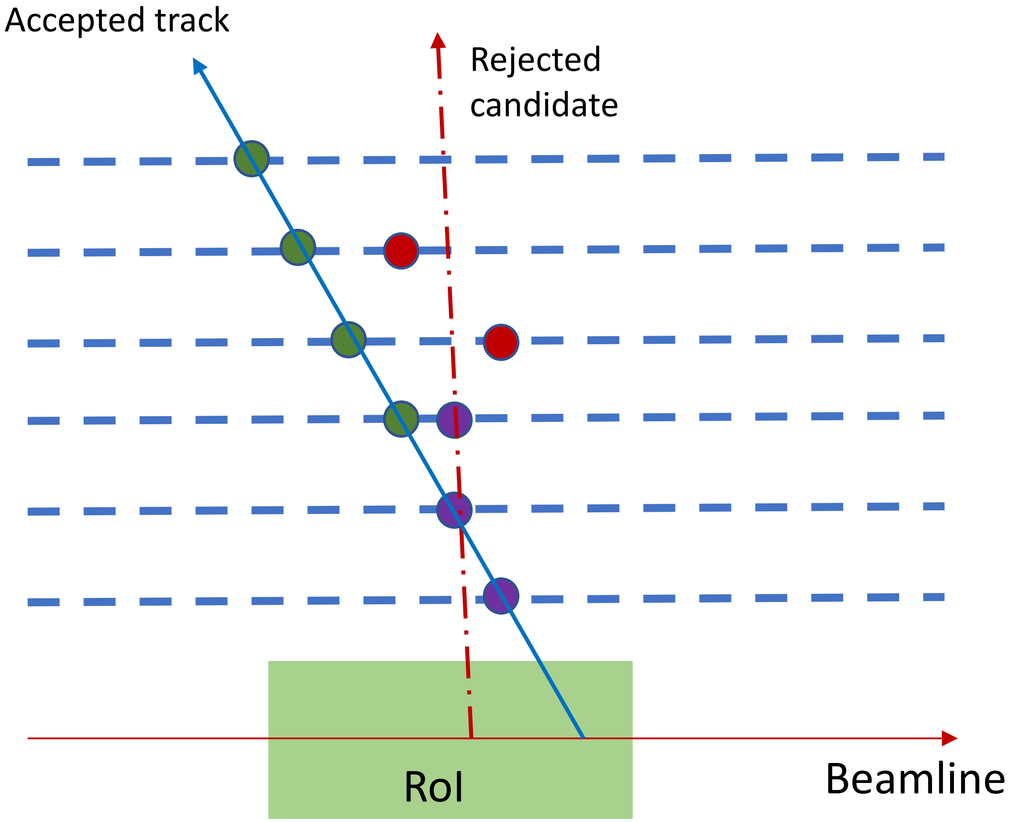
Why does fake rate decrease with RoI size?

- Qualitative explanation:
  - When you run low-p_T tracking, you get a collection of both low and high quality tracks
  - With a larger RoI, hits will be used by high quality tracks, and with ambiguity solving, the hits get "taken away" from the worse tracks. The lower quality tracks get eliminated.
  - Occasionally, eliminate poor track that points near the HS in favor of a good track that points outside the RoI. **This leads to a lower fake rate when you have a larger RoI**.
  - Conversely, with smaller RoI, more poor tracks survive: the good tracks outside of the RoI never get made, so they never take away hits from the poor tracks. Leads to a **higher fake rate when using a smaller RoI**.
- Thought about another way:
  - All of the hits are available for the seeding stage regardless of RoI, so the number of **seeds** near the HS (say within 1mm) won't change much regardless of RoI size, so increasing the RoI is a win for the fake rate because it allows for more ambiguity solving on the candidates from those seeds.

Cartoon explaining fake rate vs RoI size



Blue dashed lines represent inner detector layers



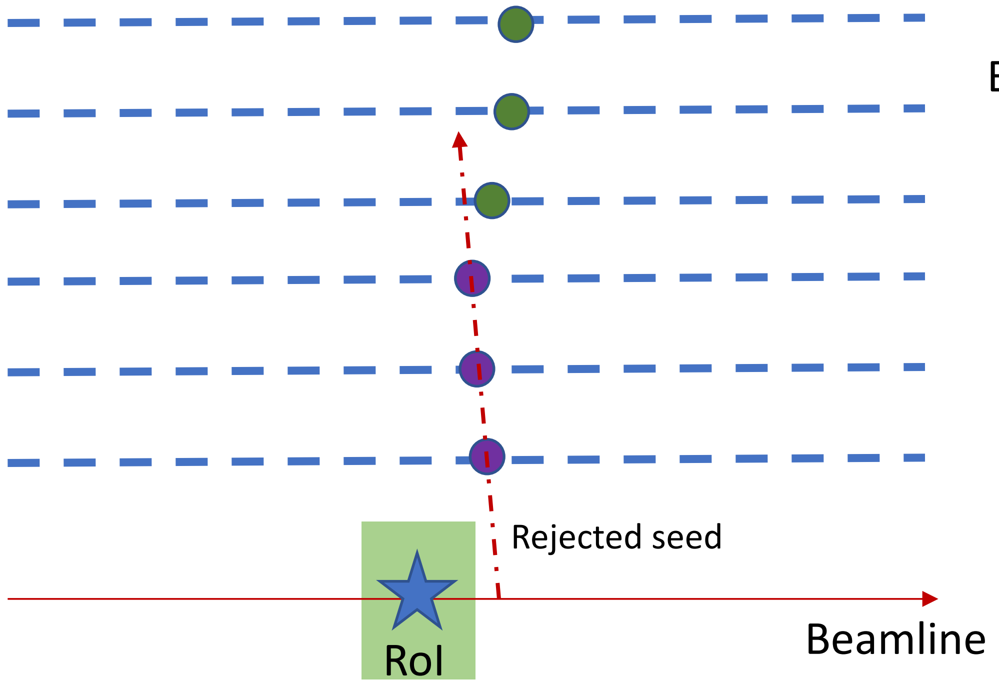
Here a small RoI was used. The seed formed from purple hits was accepted and extended using the red hits becoming a fake candidate and then track. The red dashed seed was rejected because it points outside the RoI, but it would make a good track.

With a larger RoI the fake candidate was rejected in ambiguity solving because it shared hits with a high quality track. **Generally: larger RoI = lower fake rate**

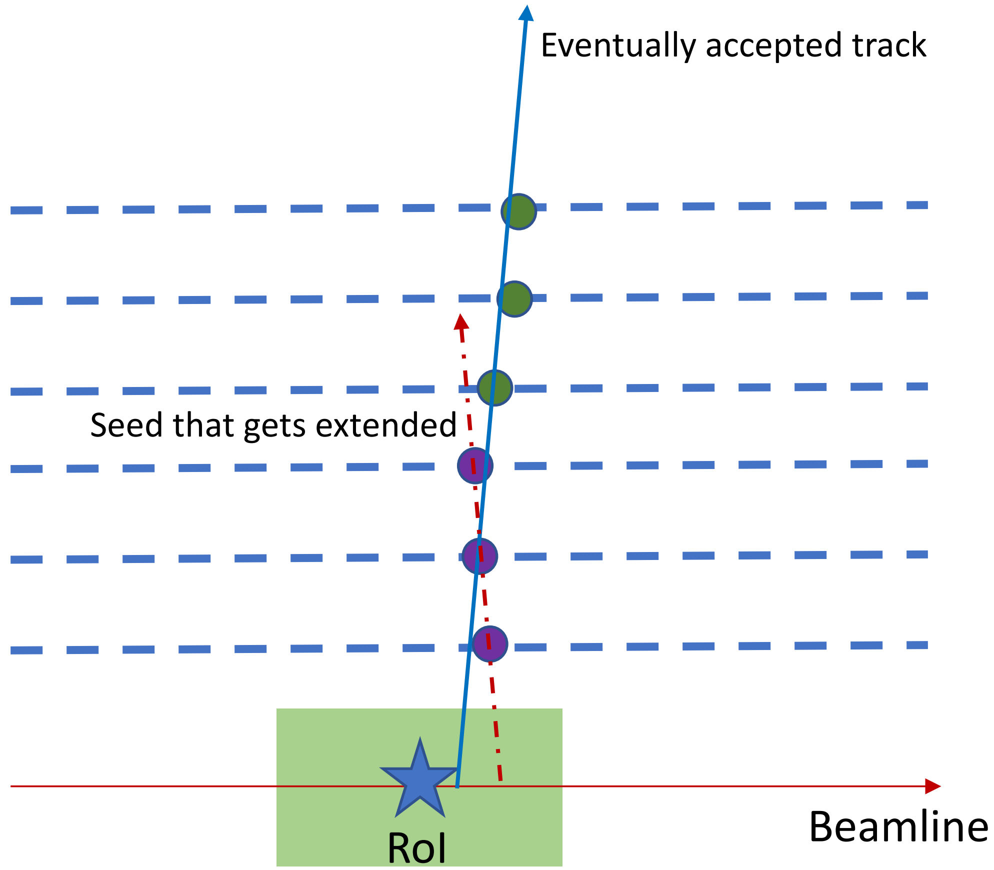
Can you explain eff. vs RoI size shape?

- Qualitative explanation:
  - While the number of **seeds near the HS** isn't very RoI size dependent, the seeds have poor z resolution
  - **If the RoI is too small, you will miss some tracks** that end up pointing towards the HS after being extended through all layers, even though they don't point there at the seed level
  - **Efficiency will increase with RoI size**; there will simply be more candidates, meaning hihger likelihood of having something to match a truth particle
  - However, efficiency drops again for a very large RoI size.
  - The drop is likely due to the ambiguity solving mentioned in the fake rate discussion above
  - Once the RoI is ~30mm, it seems that you don't gain any candidates (near the HS!) by using a larger RoI. **This means that efficiency is NECESSARILY maximal at such an RoI.**
  - A larger RoI will only mean more ambiguity solving, which can only take away tracks, reducing efficiency
  - Admittedly, that means that the ~30mm RoI will have a higher fake rate too (due to less ambiguity solving). A bit of further investigation is needed, but everything seems consistent with this explanation so far.

Cartoon explaining efficiency vs RoI size



Blue dashed lines represent inner detector layers



Here a small RoI was used, so the seed that gets created from the purple hits gets rejected.

With a larger RoI the seed was accepted and eventually extended through the green hits, slightly adjusting the measured z0.