Event Filter Tracking for the Upgrade of the ATLAS Trigger and Data Acquisition System

Viviana Cavaliere for the ATLAS TDAQ Collaboration

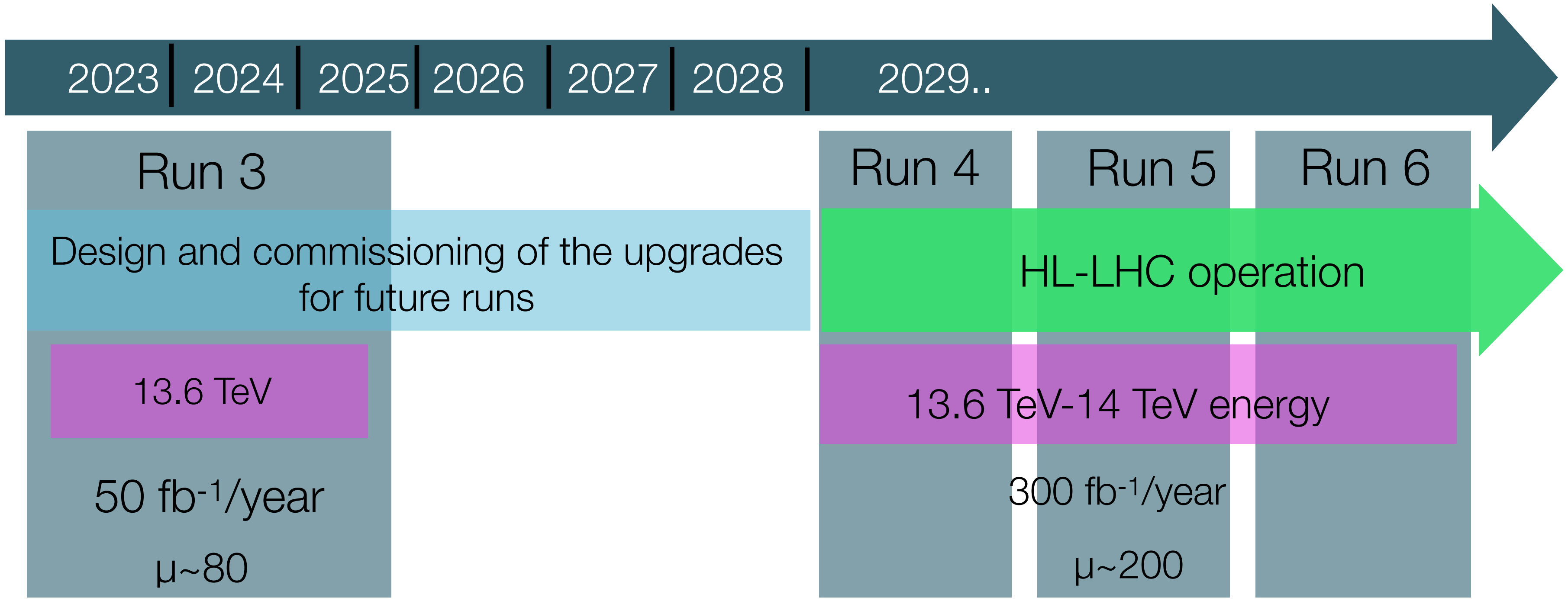
July 8th, 2022

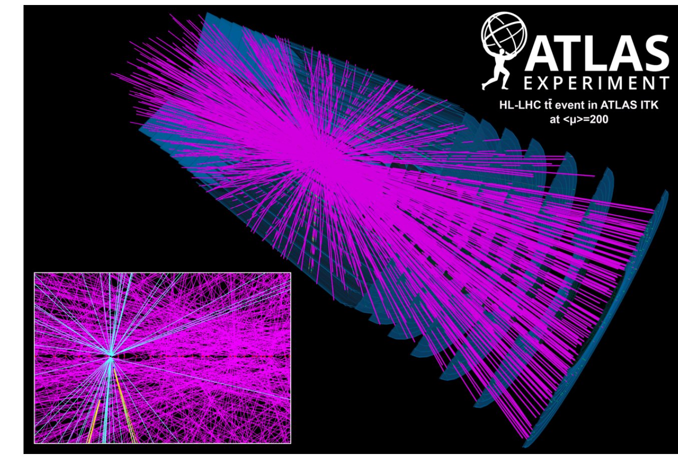
ICHEP 2022

Brookhaven National Laboratory

ATLAS EXPERIMENT

ATLAS Phase II upgrade

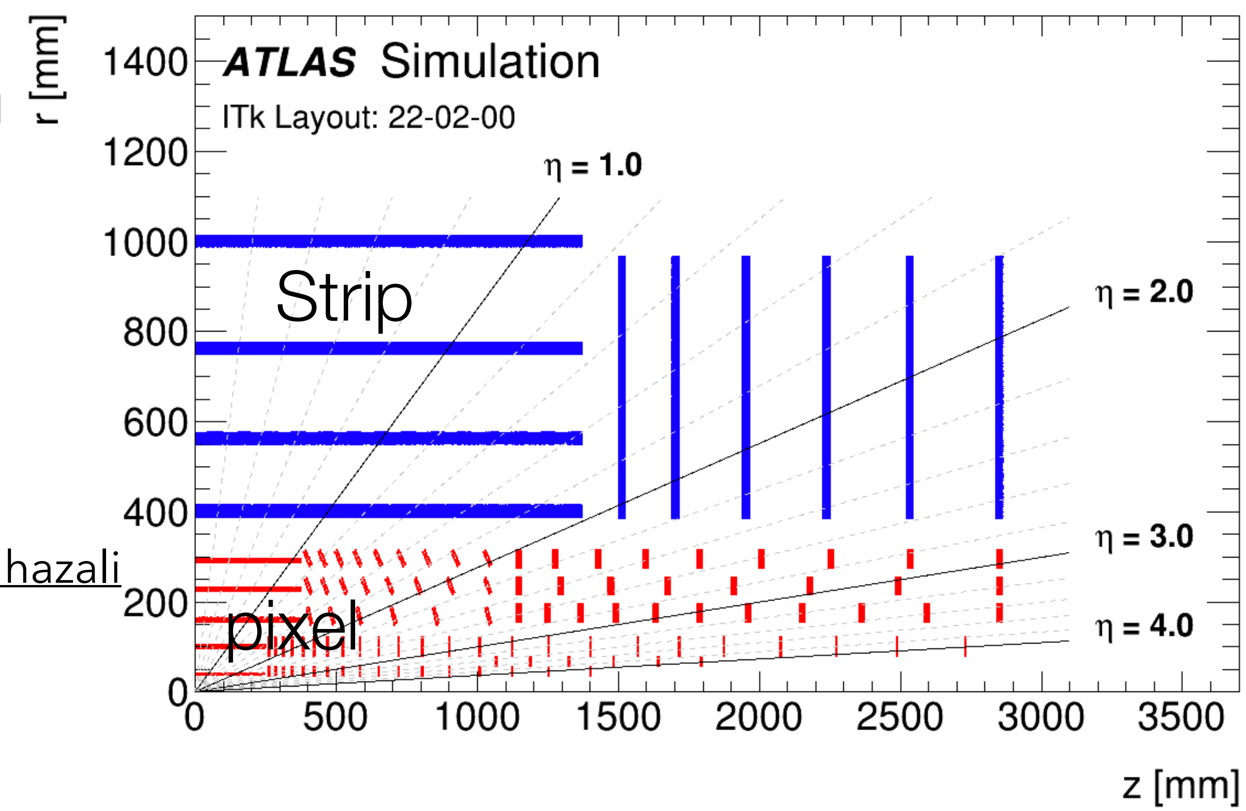




$t\bar{t}$ event at average pile-up of 200 collisions per bunch crossing.

- Conditions at the **HL-LHC,** with an average of 200 simultaneous collisions (pile-up) per bunch crossing expected, will be challenging for experiments:
  - **ATLAS is planning a major, including a new inner tracking detector, a lighter and more granular all-silicon tracking detector to allow high-precision reconstruction of charged particle tracks (ITk)**

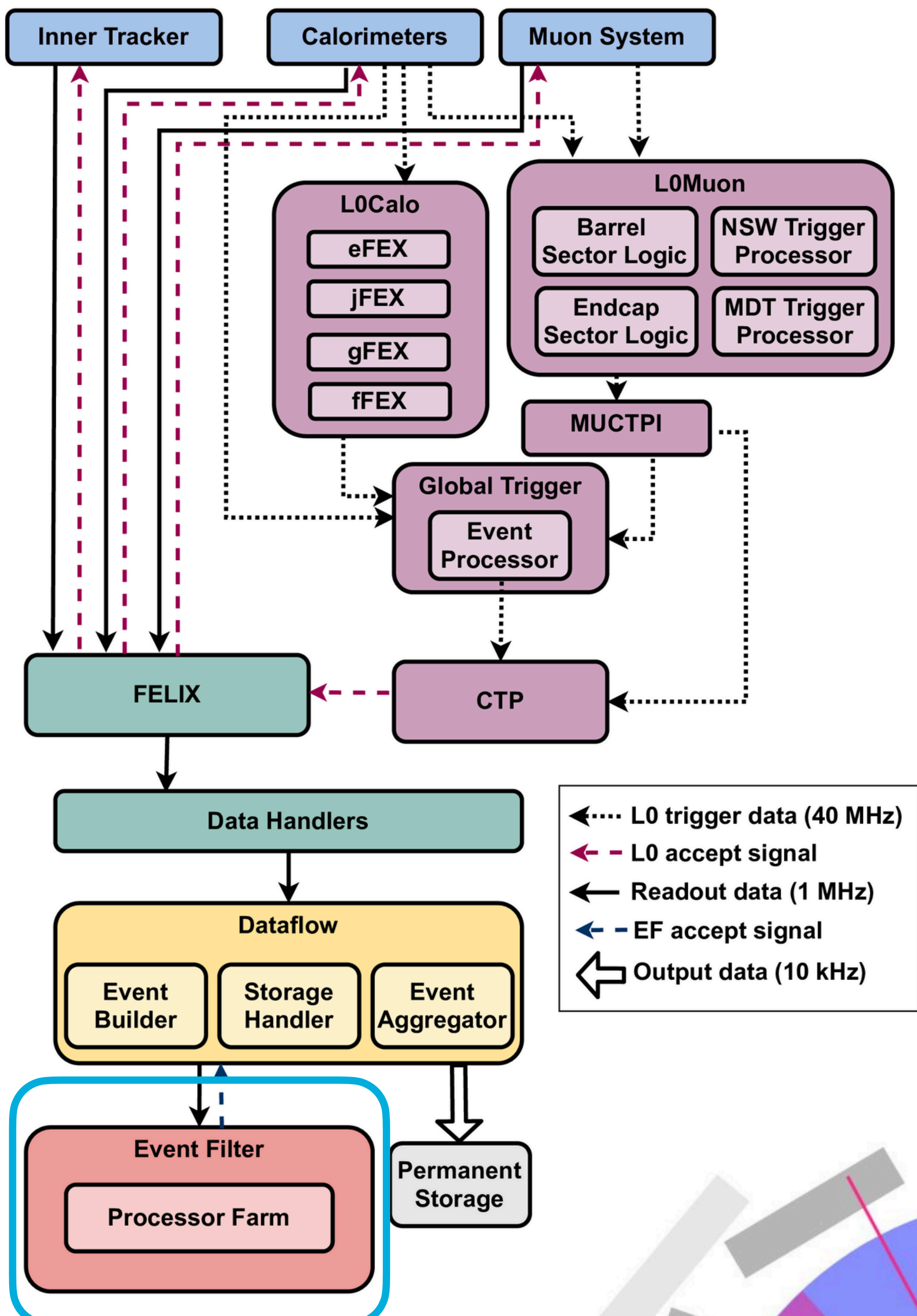
    See talk by Yassine El Ghazali
  - **Triggering will become more difficult and time consuming**



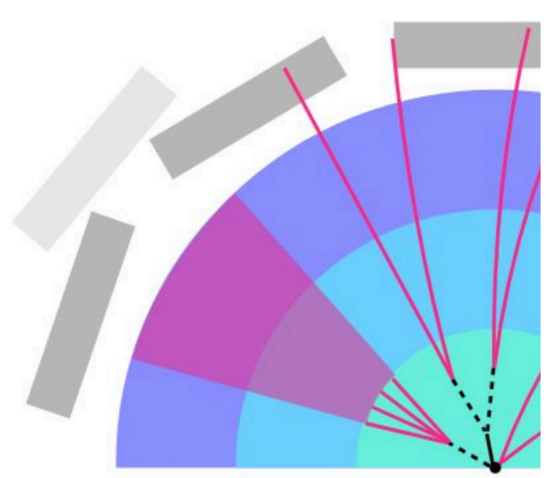
Tracking at trigger level is essential to control rates while maintaining good efficiency for relevant physics processes

Brookhaven National Laboratory

ATLAS TDAQ system for HL-LHC



- The Event Filter is a farm of the commodity CPUs / commodity accelerators
  - **Receive input from Level-0 at 1 MHz**
- Perform **regional tracking** in Regions of Interest defined based on objects identified at Level-0
  - used to verify the presence of high-p_T tracks in single lepton triggers and to associate objects to a common vertex
- Run **full-scan tracking** at a reduced rate after the regional tracking for jets and E_T miss
- Large radius tracking for exotics signal
  - focuses on tracks with high impact parameters like those resulting from the decays of Long Lived Particles

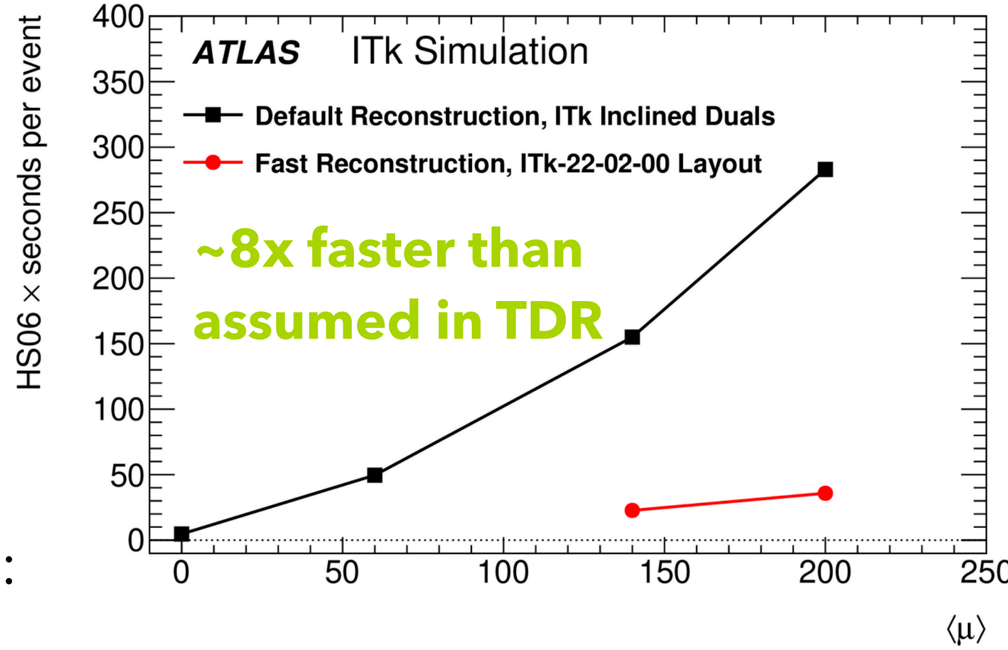


K. Pachal

Brookhaven National Laboratory

Tracking @ Event Filter

- The Technical Design Report (CERN-LHCC-2017-020) assumed a hardware-based track reconstruction based on associative memory ASICs and FPGAs
  - Plan to possibly process higher input rate (1MHz→4MHz)
    - Decided not to pursue the evolution scenario
  - Software tracking improvements
  - The rise of commercial accelerator cards



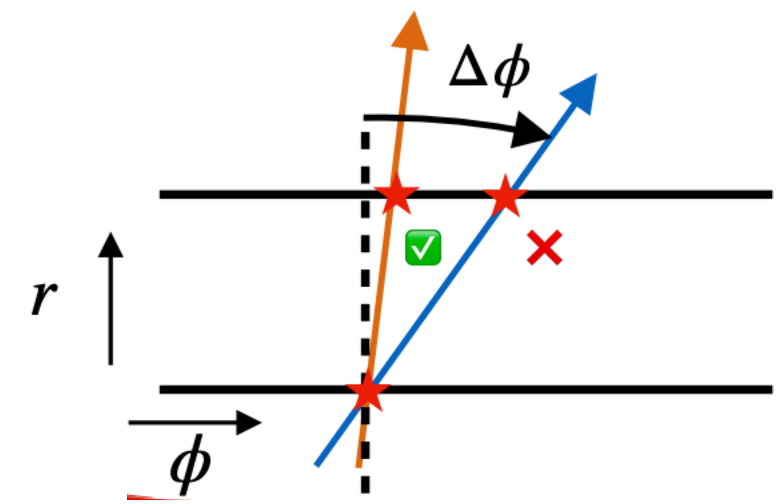
- Revisited the solution for EF Tracking (ATLAS-TDR-029-ADD-1):
- New proposed EF design==> flexible, heterogeneous commercial system consisting of CPU cores and possibly accelerators
- **Develop demonstrators for CPU/GPU/FPGA** with a decision of the technology of the final system in 2025 driven by
  - requirements on the tracking performance (efficiency, resolutions & fake rate)
  - Must come in within budget and satisfying space and power constraints
  - **The final system is unlikely to be one of the exact demonstrators outlined in these slides but working prototypes provide confidence in our ability to build a system that meets all specifications**

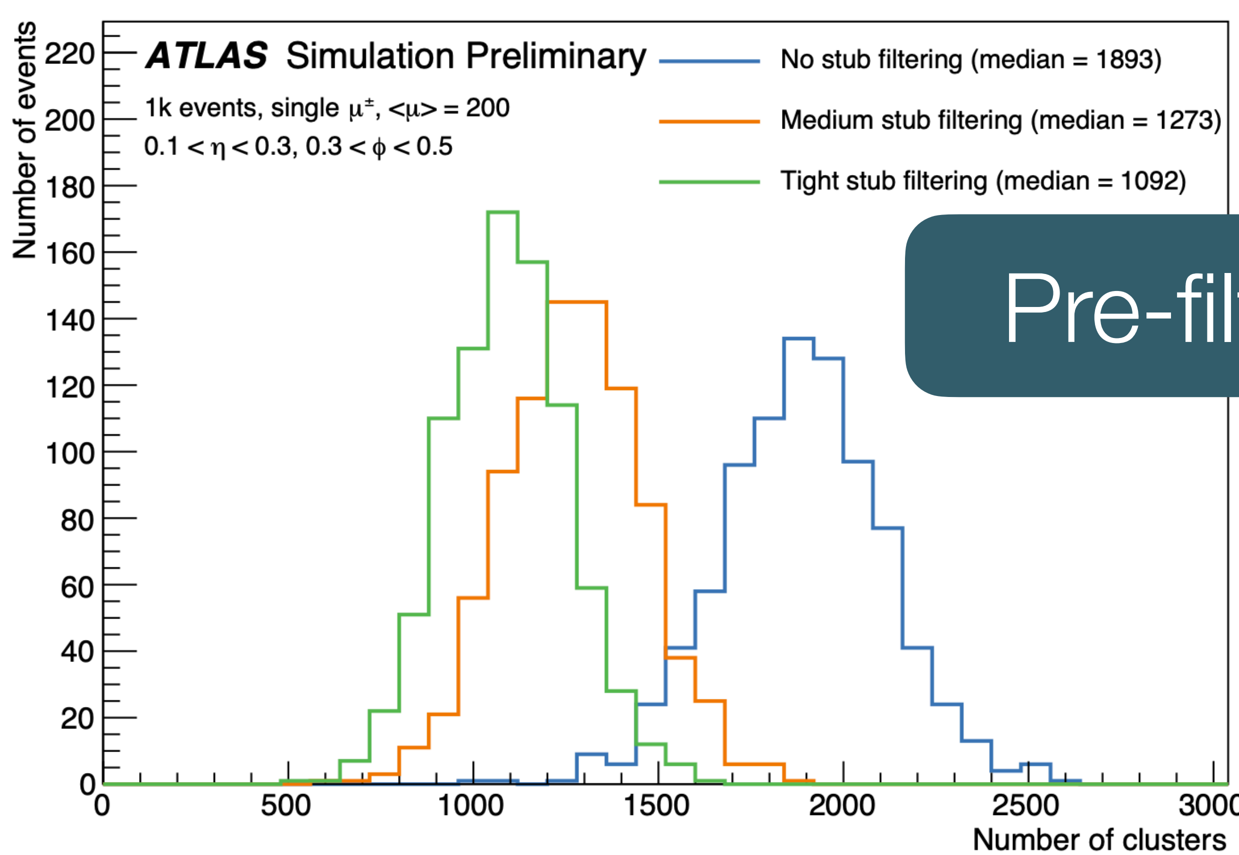
Brookhaven National Laboratory

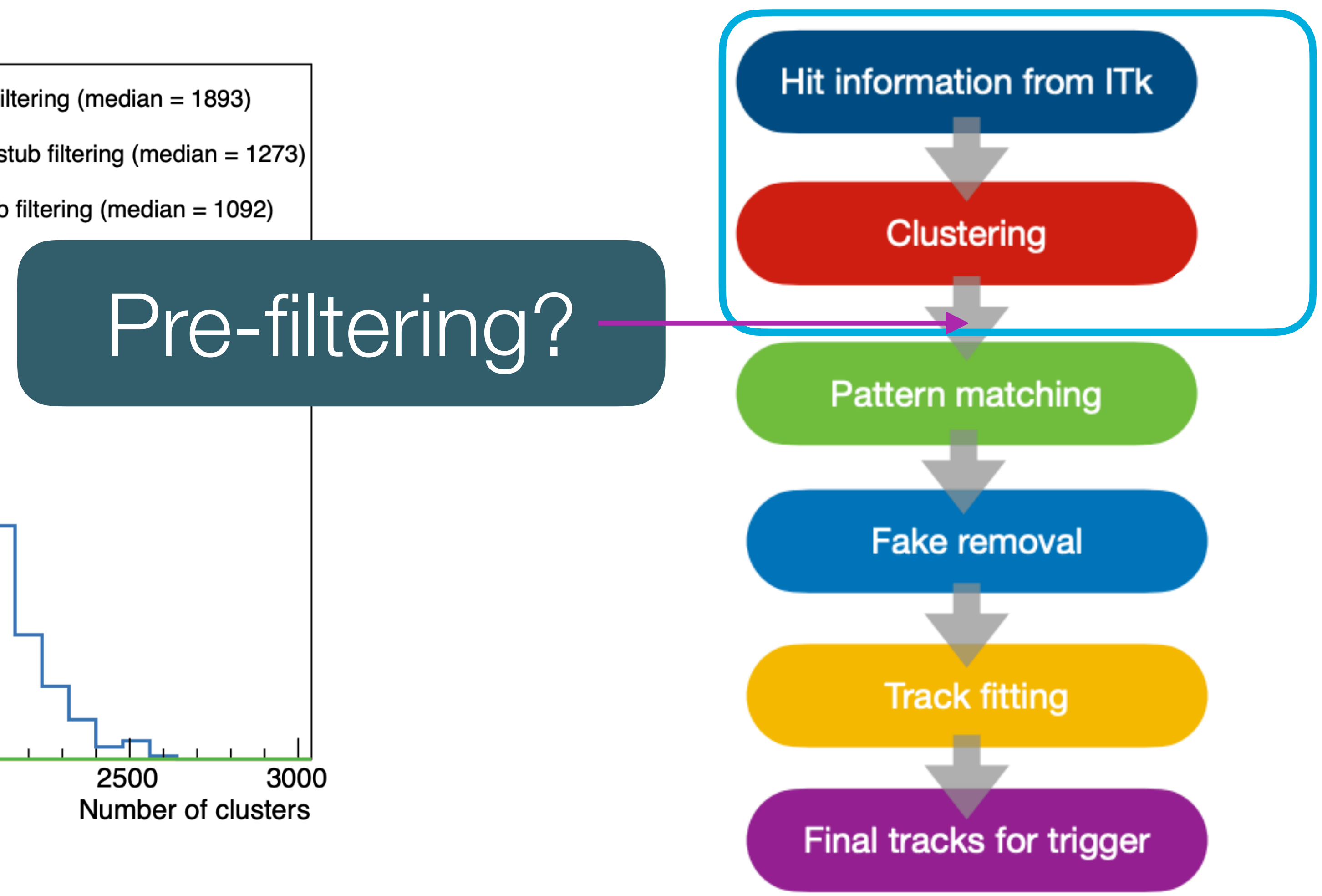
Tracking @ Event Filter

- Several common elements between various system designs of EF Tracking

- **ITk Data Preparation:**
    - Decode the ITk raw data and cluster the hits in each ITk sensor
    - Pre-filter hits and/or to create space points using the ITk Strip clusters from each side of a stave [silicon sensor with readout +cooling], which reduces the number of measurements to be handled later in the reconstruction chain



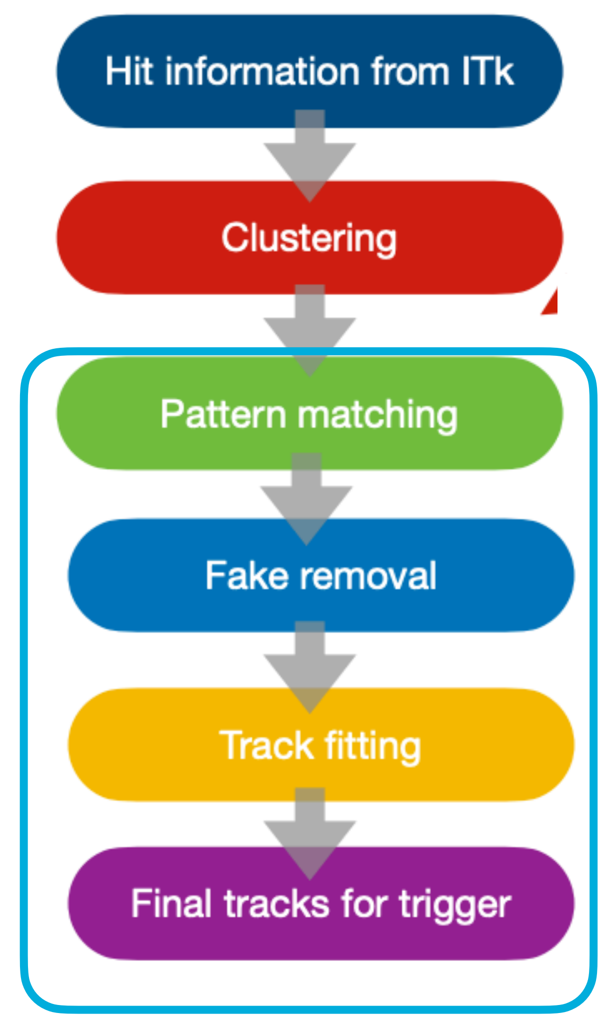




Tracking @ Event Filter

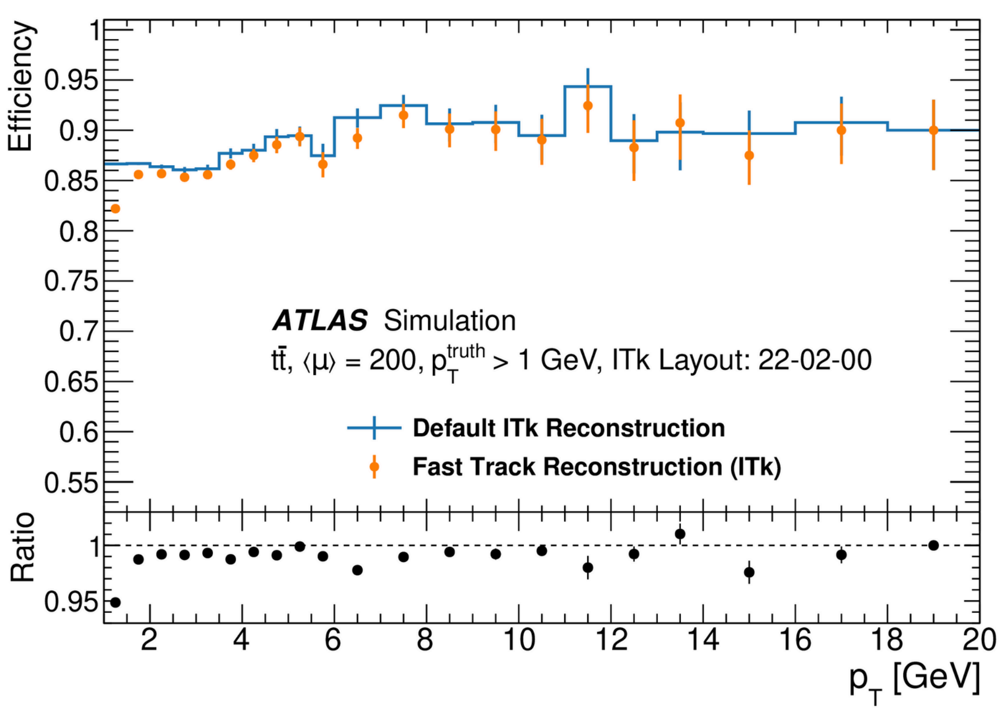
- Several common elements between various system designs of EF Tracking

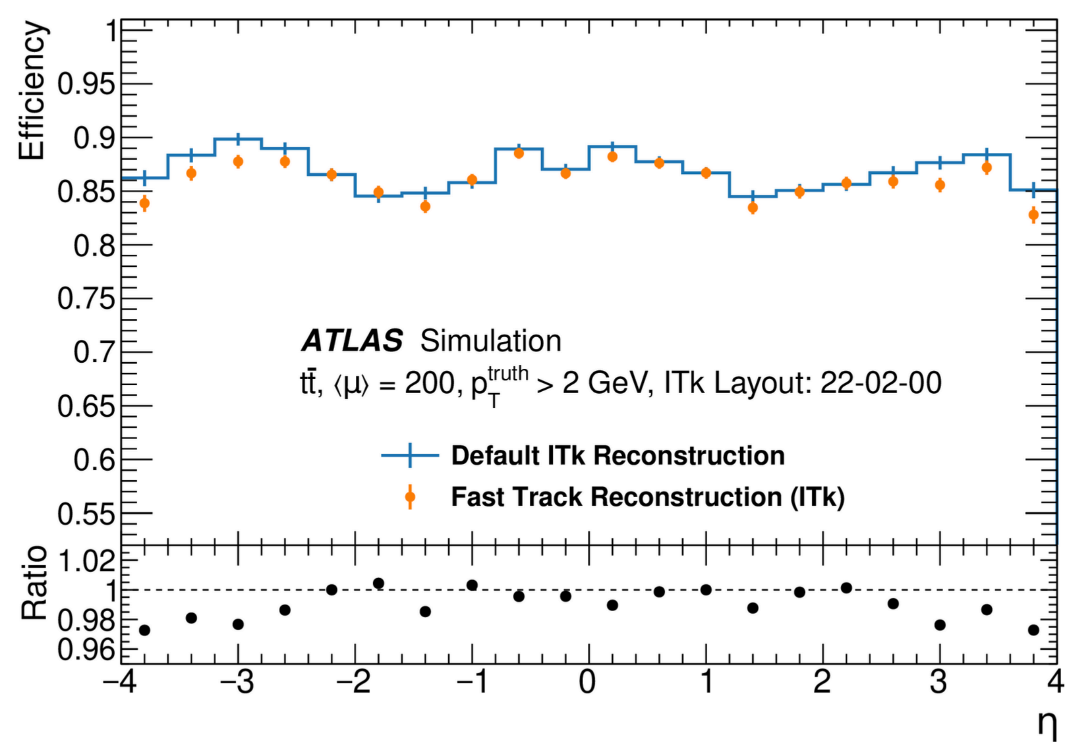
- **Tracking seeding and pattern recognition:**
  - Take subset of hits to find likely track candidates (pixel-seeded or strip-seeded)
- **Track extension, fitting and ambiguity resolution:**
  - extend track seeds into complete track candidates adding any unused silicon layers
  - Algorithms for duplicate removal, fake rejection
  - Final high precision track fit to determine the track parameters



Fast-tracking prototype for CPU

- Based on recent fast ITk reconstruction that makes use of updates to ITk geometry and new performance improvements to **speed up total track reco from earlier estimates by ~x8:**
  - **Remove ambiguity resolution** by implementing tighter track selection upstream, make use of fast Kalman filter (some approximations in material model), seed finding only in pixels (no strips)
  - **Regional tracking for η-φ coverage of 5%**, corresponding to 15% of ITk detector elements
    - Raise p_T threshold to 2 GeV for region tracking (800 MeV in forward) for regional tracking

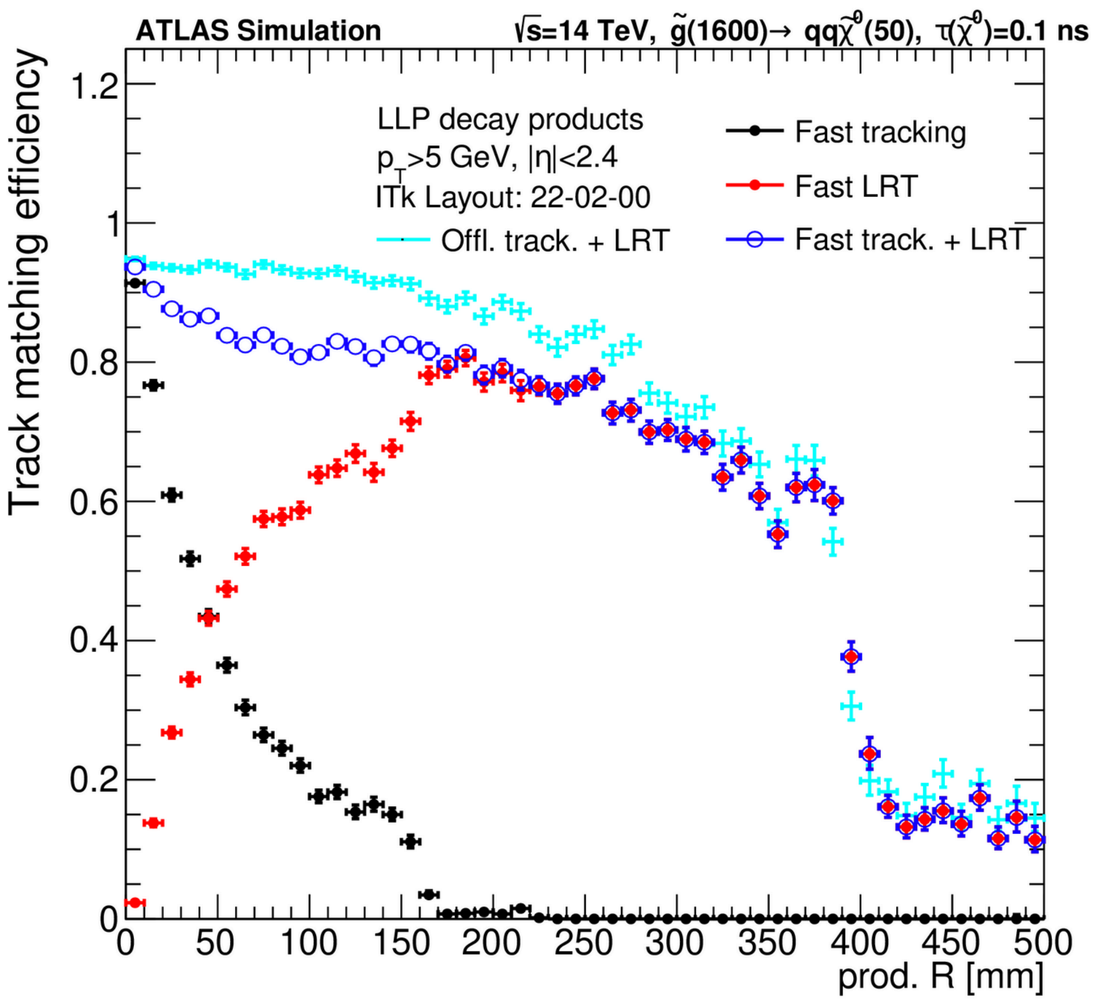




Brookhaven National Laboratory

CPU-based demonstrator Large Radius Tracking

- Runs as second pass after prompt tracking, removing used clusters and seeded with space points from stereo strip hits.
    - **Still being tuned, but already promising performance**



Pair production of gluinos→pair of neutralinos + quarks. Neutralinos decay via RPV coupling to qqq, τ=0.1 ns. Different masses change whether decay products point to PV

Efficiency for long-lived neutralino decay products as a function of production radius

plan is to run in regional tracking and on fraction of events to control the CPU cost :

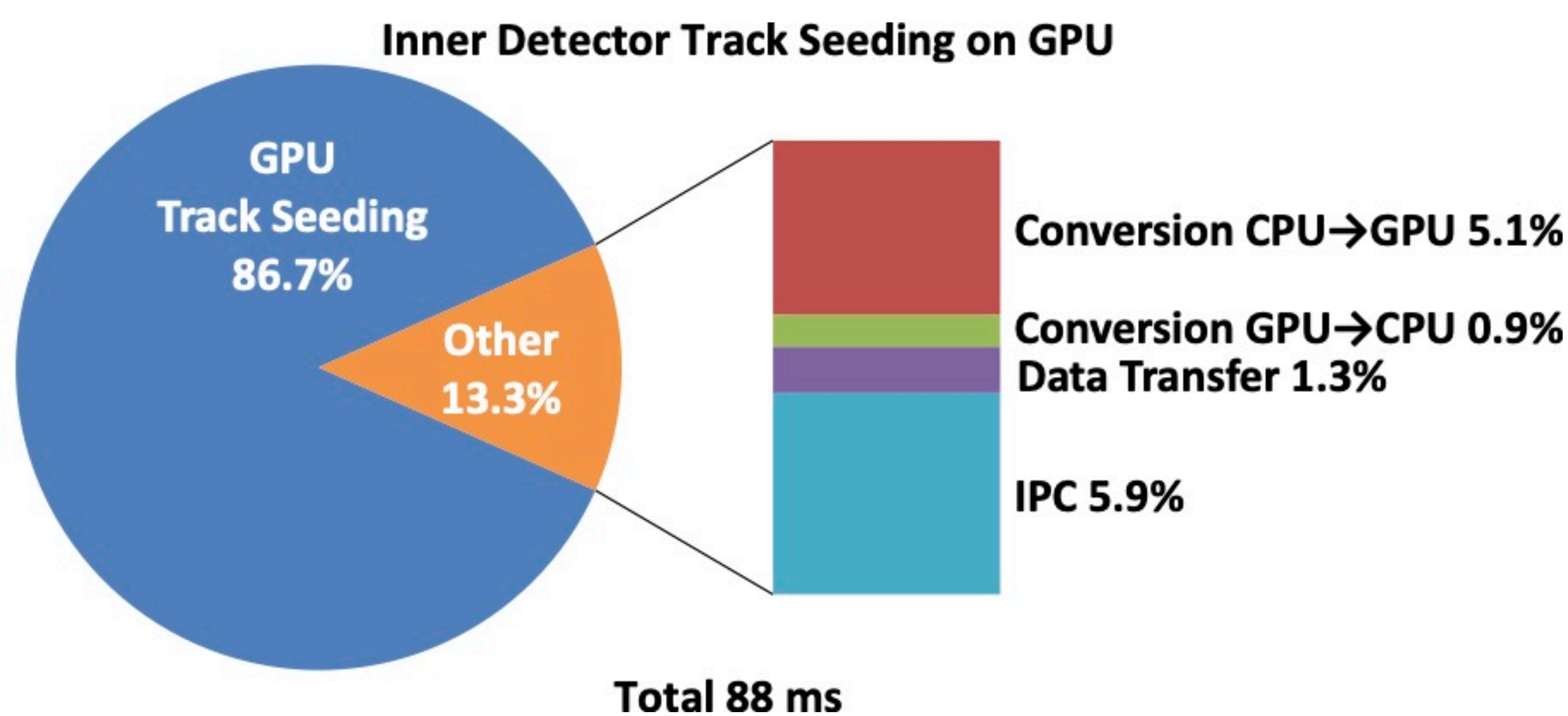
- Running this configuration on 20 kHz of events at pile-up of 200 would increase the CPU cost for tracking only marginally (<4%)

GPU-based demonstrator

ATLAS EXPERIMENT

- Provided a **summary of GPU results** in TDAQ TDR that demonstrate the potential of GPUs
  - Uses current ID system and μ=46 samples
- The most computationally intensive data preparation and track-seeding stages
- Overheads for data conversion, communication between processes and not having every stage moved to GPU limits potential gains.
- TDR found that using GPUs would provide the same cost/benefit as adding more CPUs, but this is already a **demonstration of feasibility**

Minimizing data format conversions critical

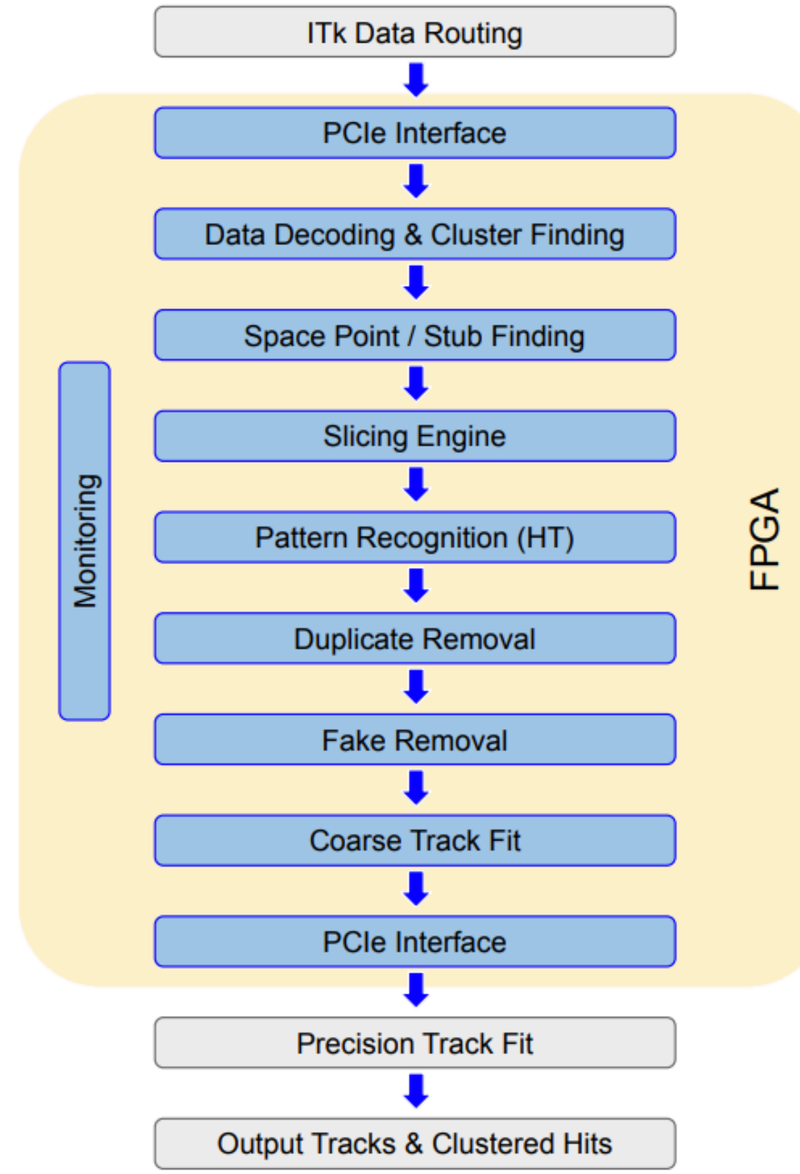


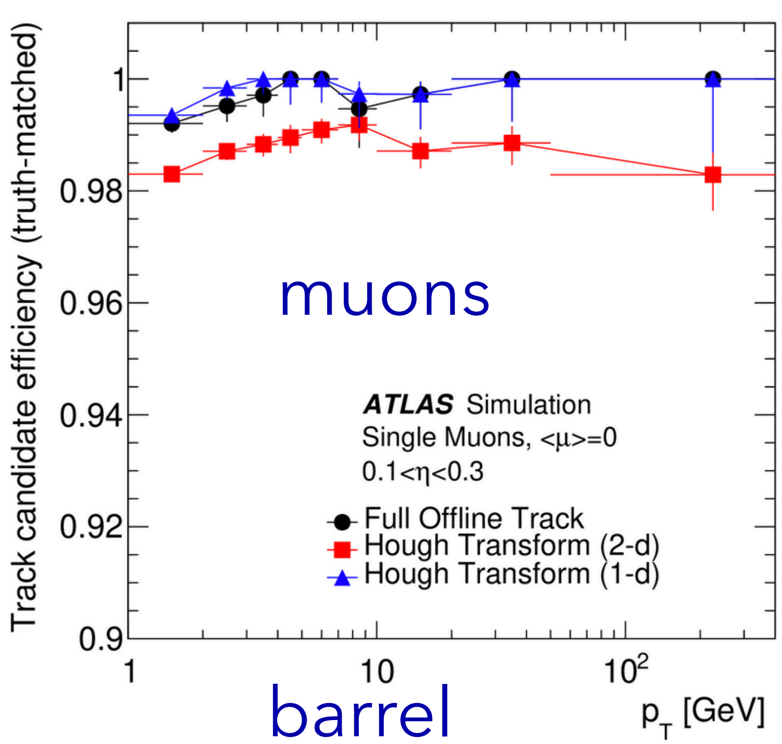
Brookhaven National Laboratory

FPGA-based demonstrator

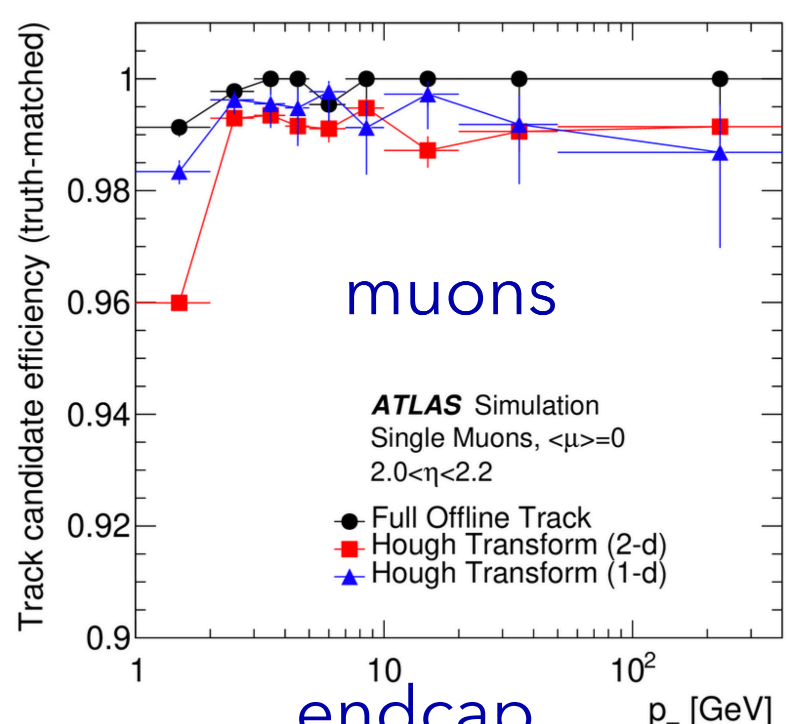
- Results focused on Hough Transforms inside FPGAs (as one example)
- Full track extrapolation to all layers inside FPGA, final CPU-based precision fit for final rejection (but one of many options)
- Target FPGA wherever possible Xilinx Alveo U250
- Neural Network to reject fake+duplicate tracks from Hough Transform

Talk by Kazuki Todome on Hough Transform





barrel

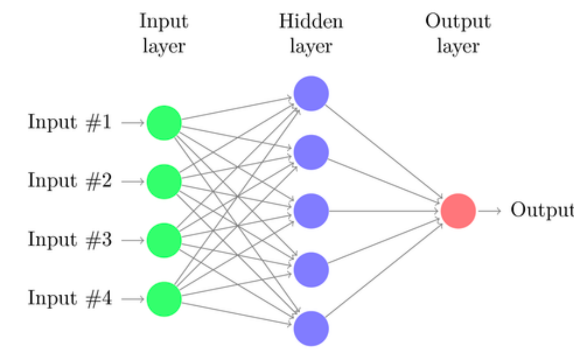


endcap

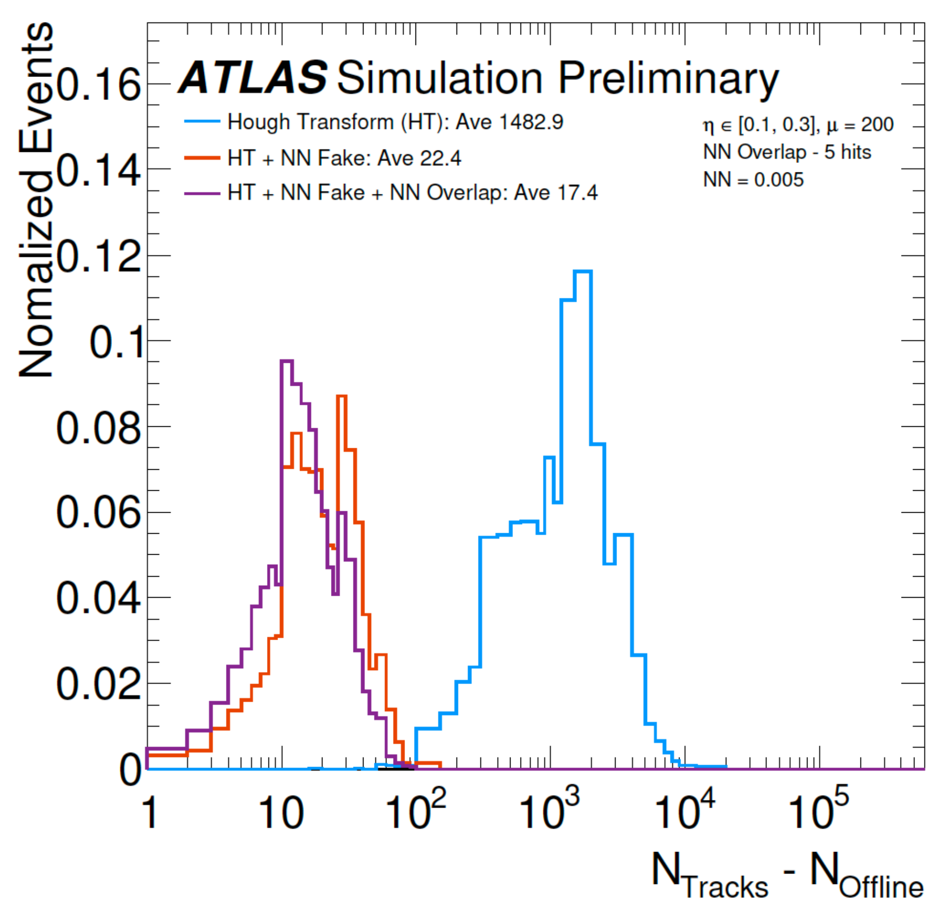
Brookhaven National Laboratory

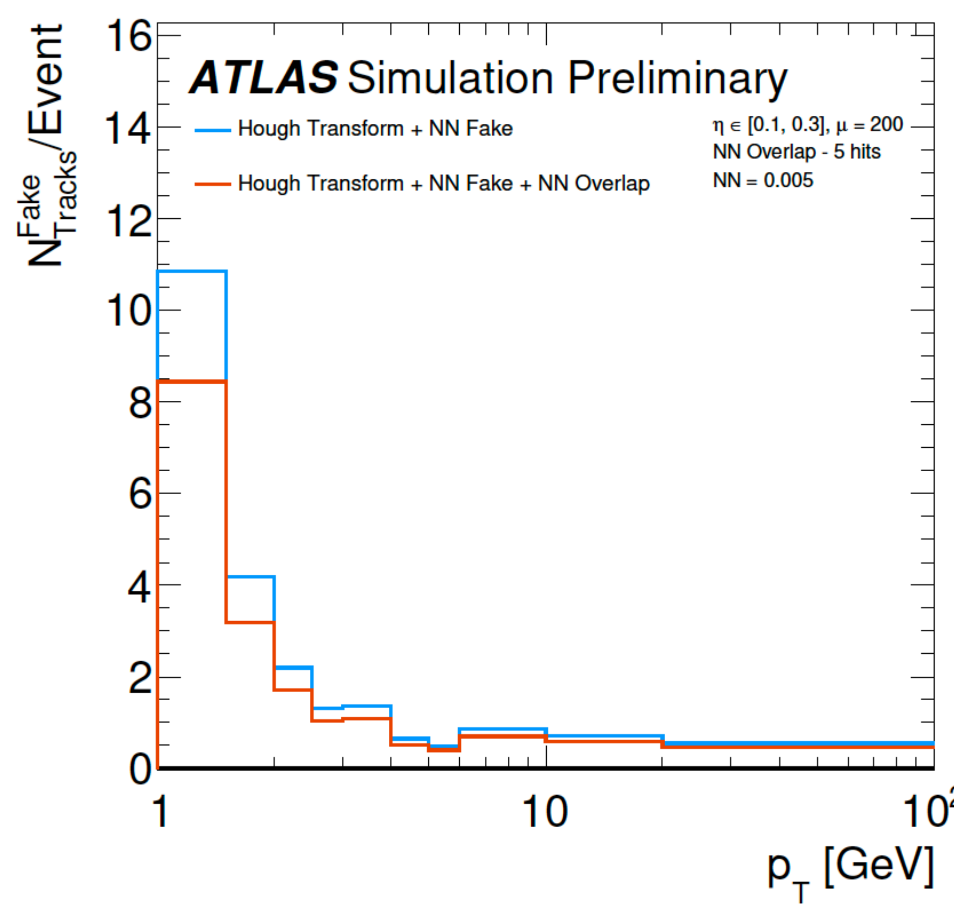
Fake and duplicate removal

- After road filter, very promising first results from NN fake/overlap removal
  - Classification problem that uses correlations between hits in a track candidate
  - Requirements are high background rejection (>10^{-4}) while keeping a high signal efficiency (>99%)
  - While the main considerations are to fit the NN on a FPGA, ensured that it can be used on CPUs/GPUs
- Inputs to a MLP: Scaled and rotated x,y,z positions on track
- Given this choice, NN has two tasks:
  - Reject fake tracks to improve purity (NNFake)
  - Pick the best track candidate if multiple candidates share N hits (NN overlap)



Performance on FPGA evaluated with hls4ml: only a relatively small fraction of FPGA resource is taken up by the NN





Brookhaven National Laboratory

Summary

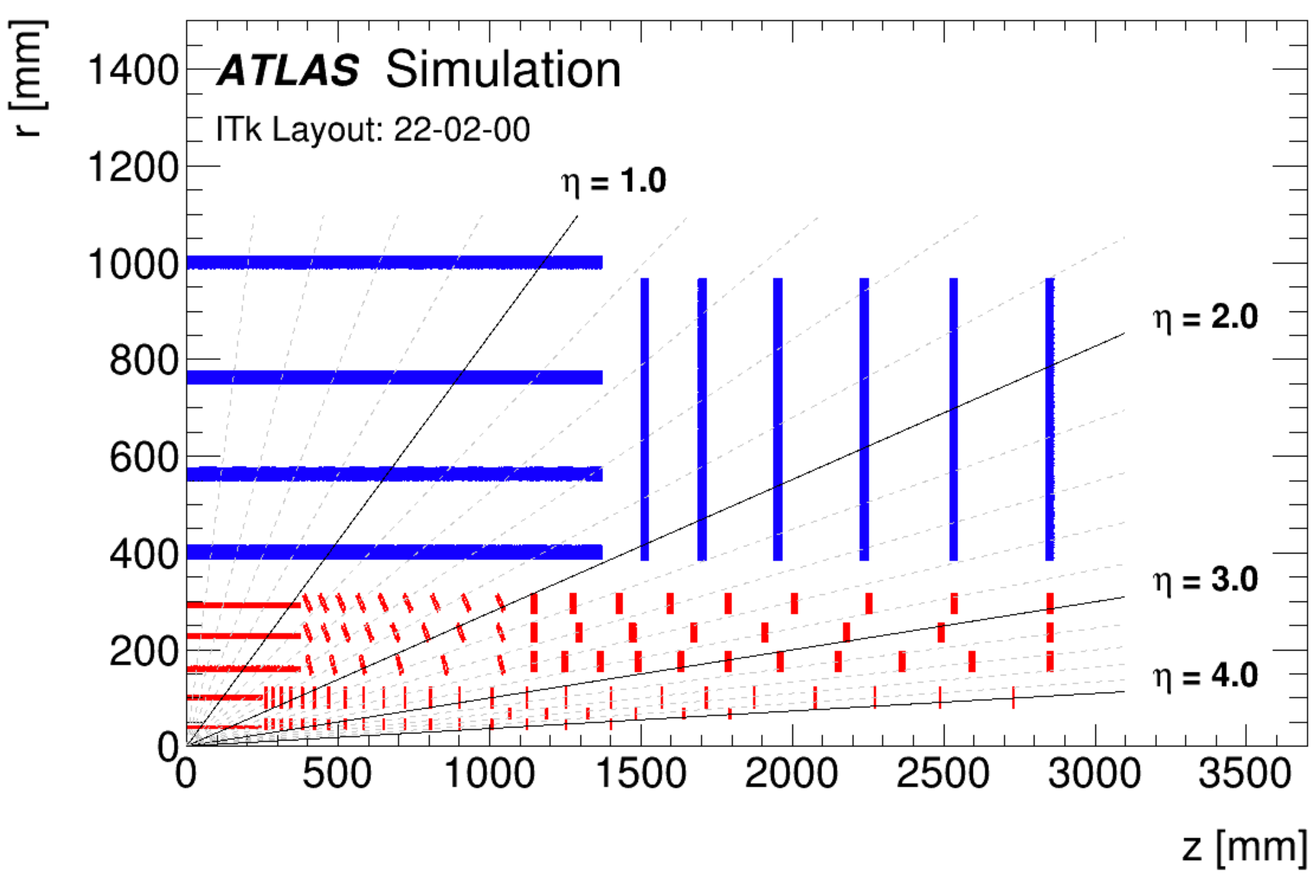
- Ongoing redesign of the EF Tracking for Phase-II upgrade of ATLAS
- Commodity systems based on CPUs/GPUs/FPGAs identified as viable solutions in the amendment of the TDAQ technical design report
- Fast tracking on CPUs demonstrated as a possible solution matching the requirements at the time of the TDR amendment
  - A speedup by factor of 8 achieved in the prototype
- Initial demonstrators using accelerators exist
  - Ongoing work on demonstrators of these (heterogeneous) systems will lead to decision on technology of the final system in 2025
  - Demonstrate delivery of the performance requirements in terms of track reconstruction performance, cost and power consumption

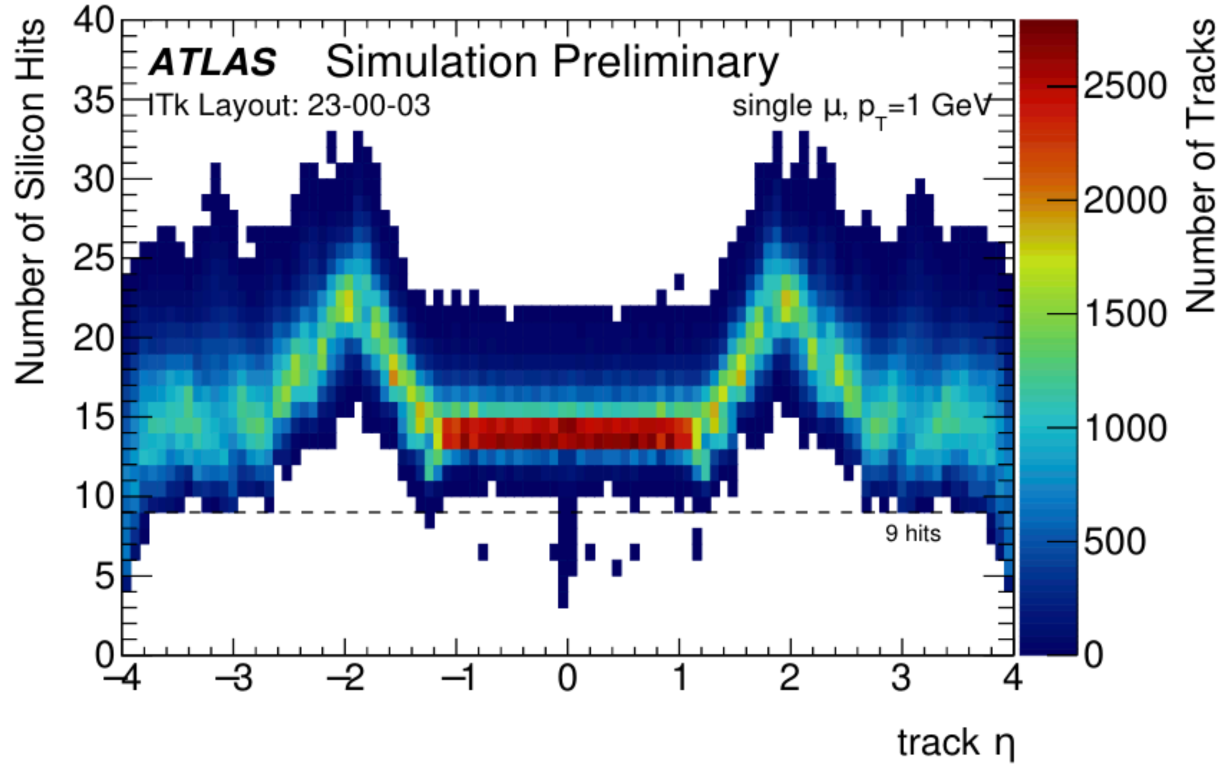
Backup

ITK detector

- ITk detector design optimized substantially since TDAQ TDR
- Simpler layout with ring design and inclined barrel sections optimized where to provide trajectory measurements and reduce CPU in reconstruction
- Pixel detector with a minimum of 5 layers in eta gives sufficient redundancy for efficient & robust seeding

ATL-PHYS-PUB-2021-024





FPGA-based demonstrator-Hough Transform

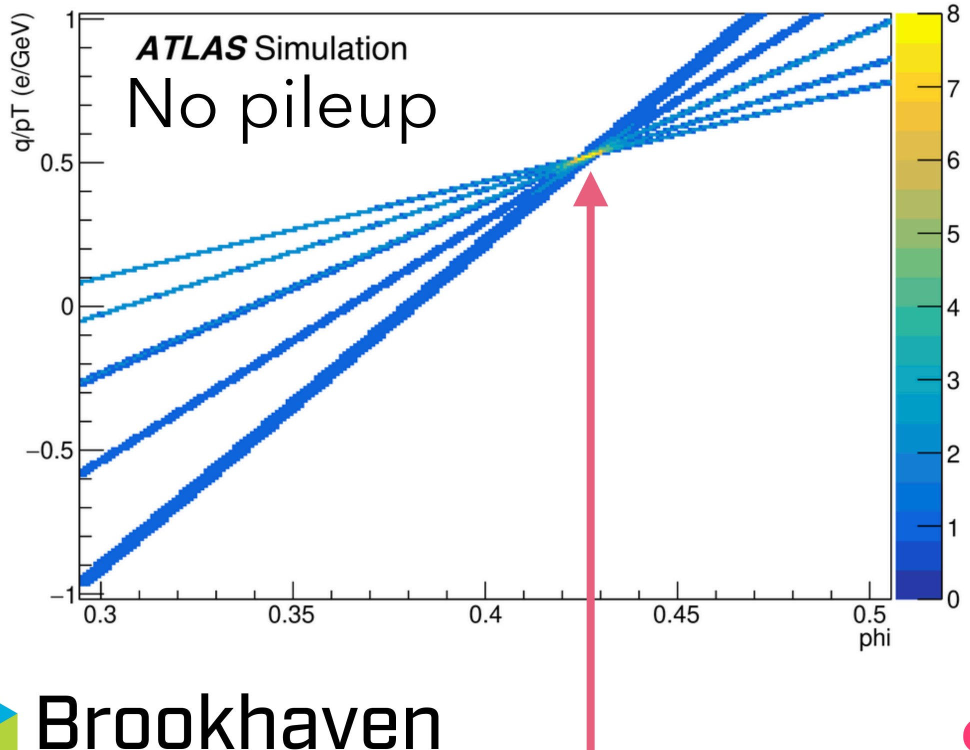
- The track of a charged particle in the transverse plane (x-y plane) of the ATLAS tracker has the shape of a circular arc which can be described by p_T and its initial angular direction φ_0.
- If a vertex constraint is imposed, the clusters on track obey:

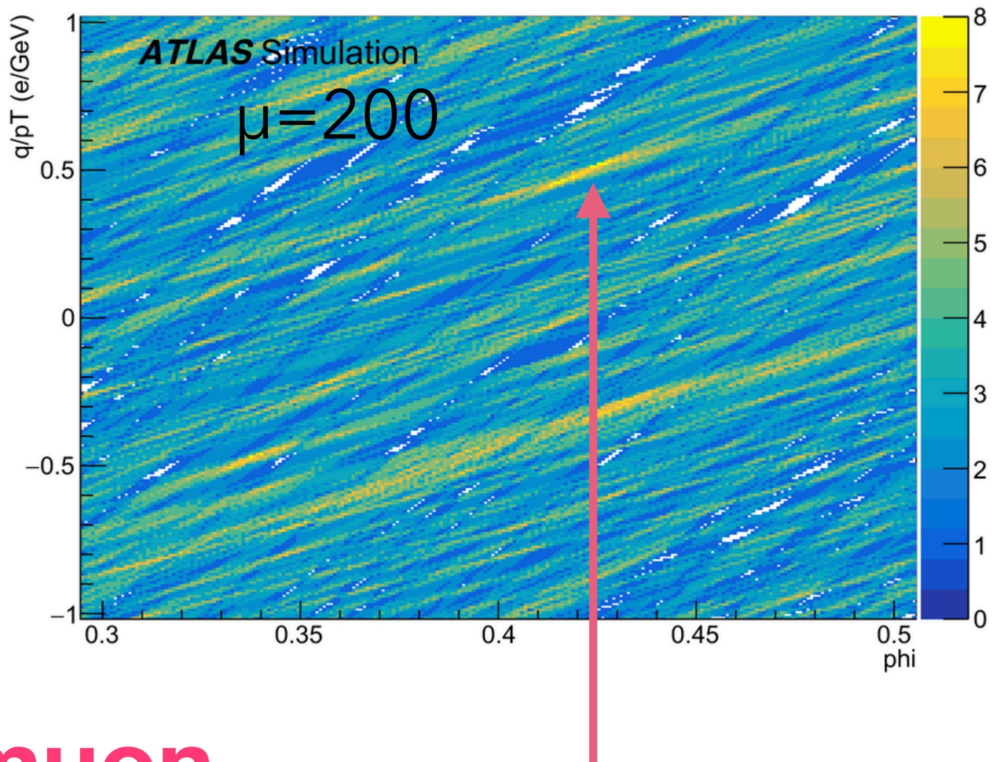
$$\frac{qA}{p_T}(\phi_0) = \frac{\sin(\phi_0 - \phi_1)}{r_1}$$

[Duda& Hart, J. Gradin et al 2018 JINST 13 P04019]

- where (r_1, ϕ_1) are the cluster coordinates, q is the charge of the particle, and $A \approx 3\times10^{-4}$ GeV mm^{-1} is the curvature constant for the 2T magnetic field of the tracker.

- Initialize a histogram (accumulator) with the parameter space to search.
  - Group hits into super strips in ϕ
- For each point, increment the histogram for all possible curves going through that point.
- Points on the same curve will intersect in the parameter space
- Threshold accumulator at a certain value.
- Extract the hits for all bins passing the threshold.





Single muon

Brookhaven National Laboratory

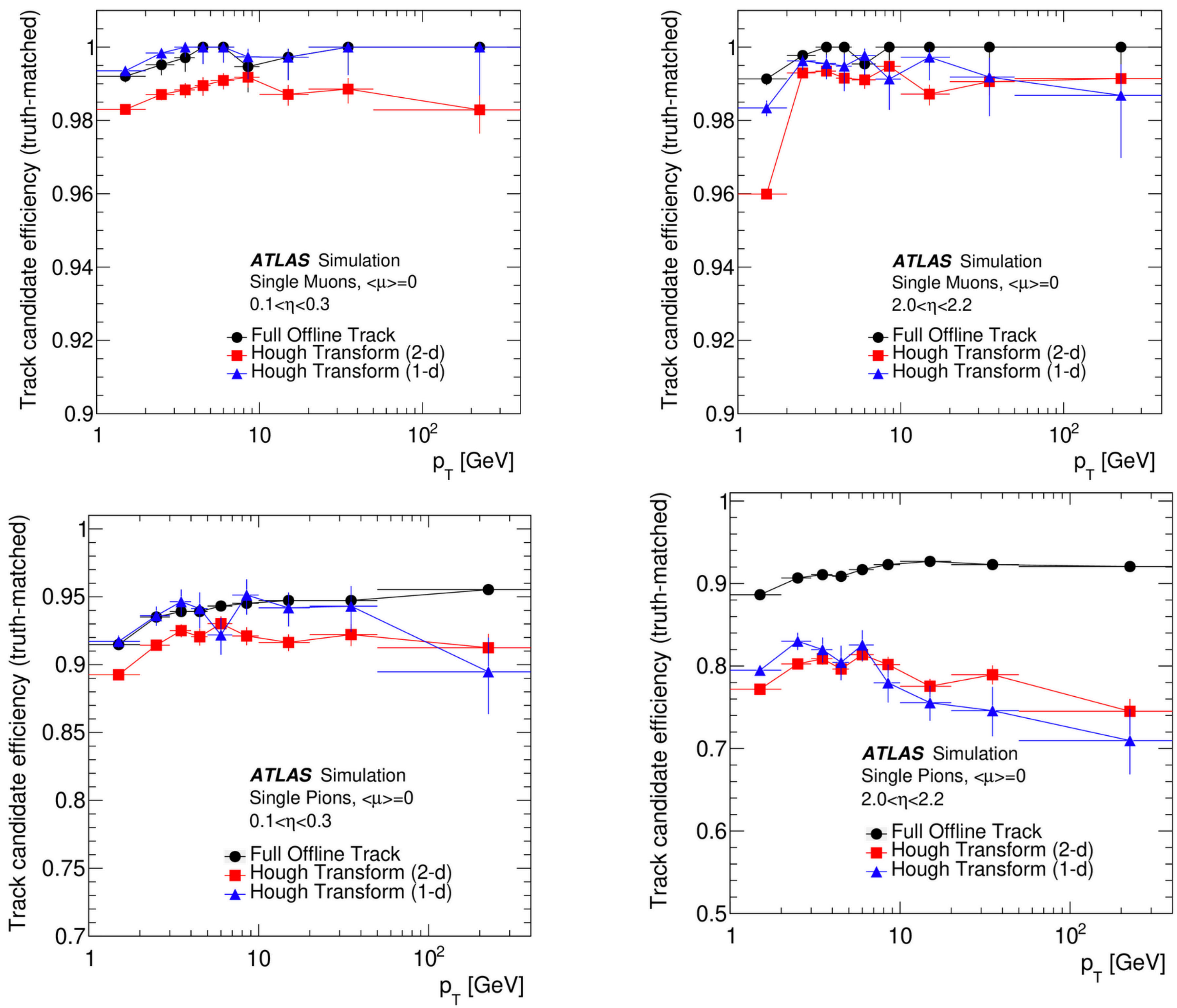
FPGA-based demonstrator-Hough Transform

- FPGA Hough transform efficiency for single particles, no pileup
- Good perf for muons in all eta regions

barrel

endcap

muons

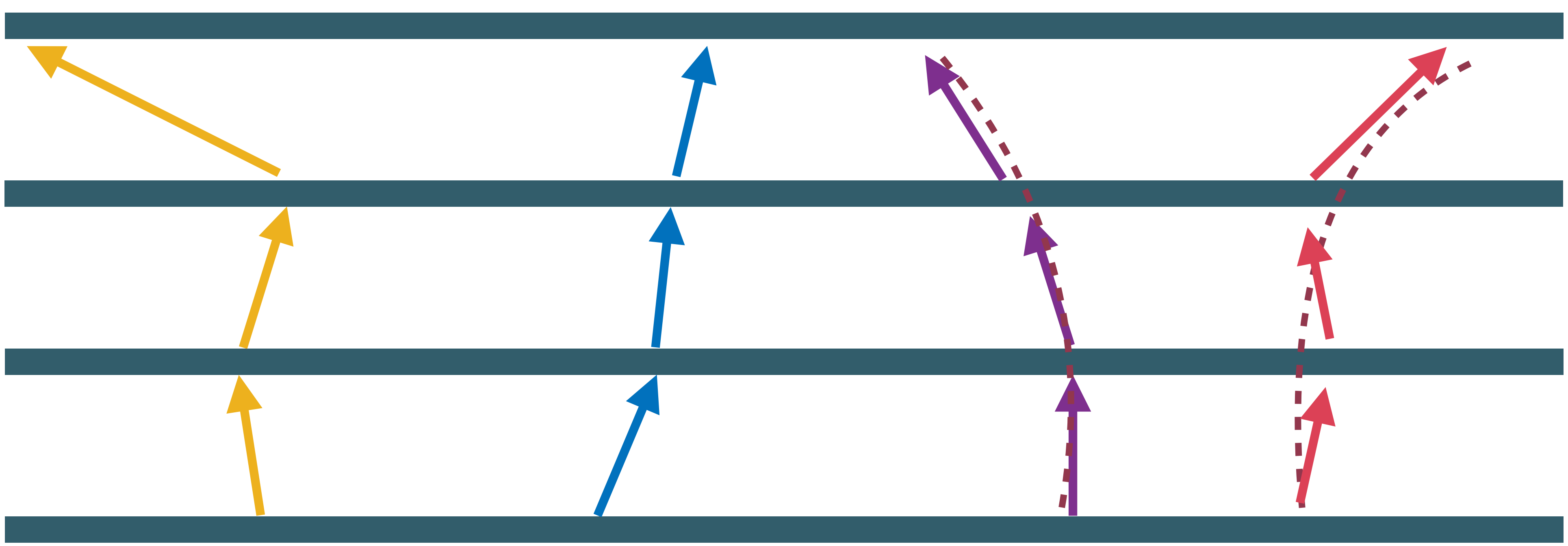


pions

Problem setup

- Initial studies performed **without** assuming what algorithm the track candidate will come from
  - To keep the conclusions as generic as possible
- **Problem:** Classify a vector of x/y/z position coordinates as coming from a 'track'
  - Train the network in the 1 pixel + 7 strip layer configuration
  - ● Fake tracks: HT tracks with fraction of true hits from a particle < 0.5
  - ● True tracks: HT tracks with fraction of true hits from a particle > 0.9

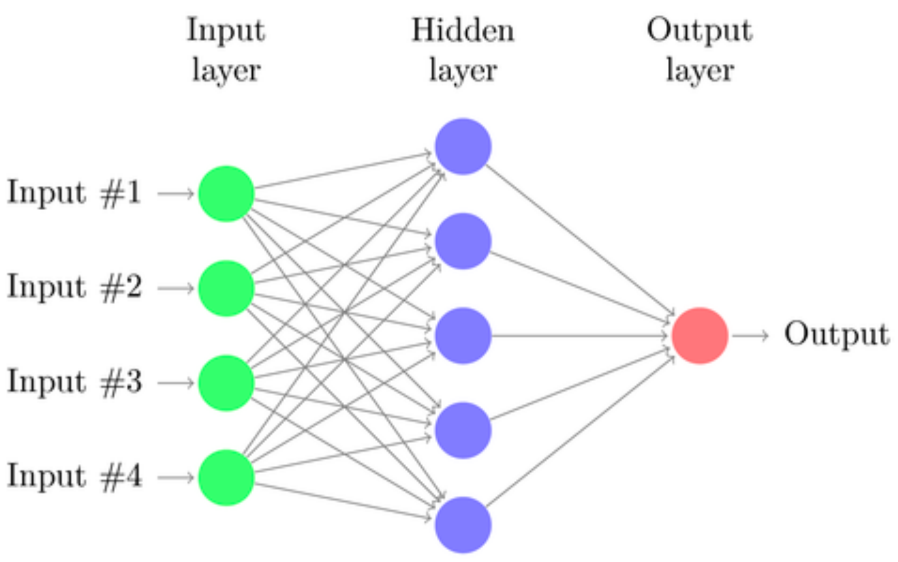
Fake Hit combinations

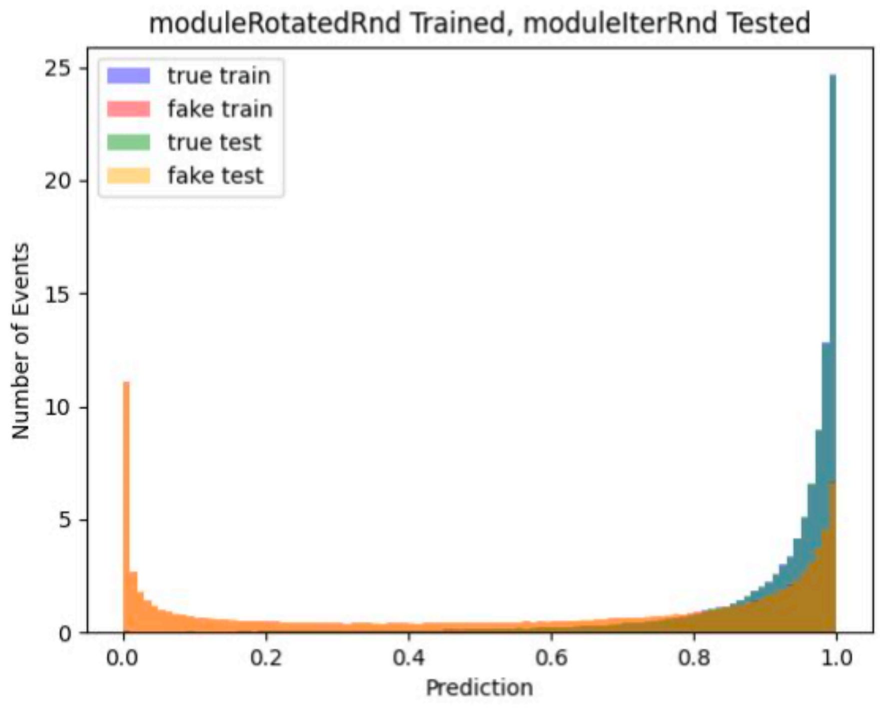


Layer	Module	Close by	Phi rotated
Combine hits that pass through successive layers	Combine random hits in modules that a true track would traverse	Replacing random hits with other nearby ones around a true track	Find a hits modules that a random phi rotated track would traverse

Initial Results

- Very promising results - allowed us to fine tune the recipe
- Architecture - simple MLP performs well
- Pre-processing - Some is required
    - Rotate hits to remove the phi DOF
    - Scale X/Y/Z coordinate such that max value is O(1)
    - Order hits by R





Removing the rotational degree of freedom