Development and performance of track reconstruction algorithms at the energy frontier with the ATLAS detector

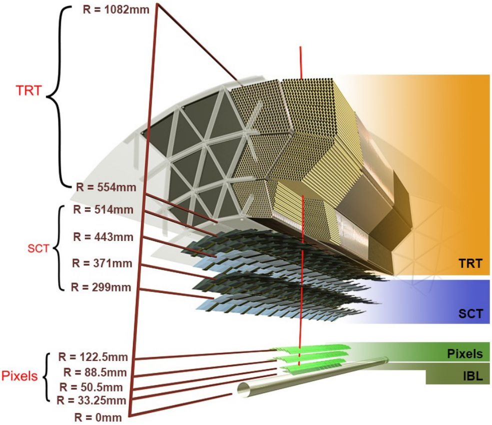
Louis-Guillaume Gagnon
on behalf of the ATLAS collaboration

CHEP '16
2016/10/12

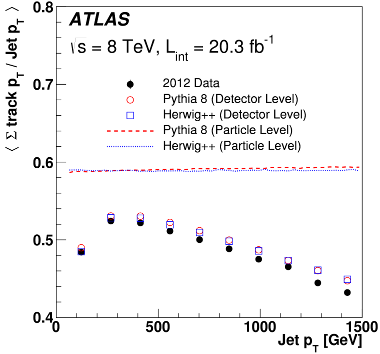
Université de Montréal

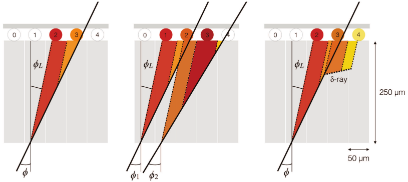
Groupe de Physique des Particules
Université de Montréal

ATLAS EXPERIMENT



- ATLAS: general purpose detector at the CERN LHC
- At center: inner detector
- TRT: Drift tube detector
- SCT: silicon strip detector
- PIXEL: silicon pixel detector





- Charged particles typically deposit charge in > 1 pixel or strip per layer
- dense environments: distance between particles close to detector granularity
  - e.g. high-pt b-hadron, τ decays or top quark jets
- Charge clusters can merge
- High $\sqrt{s} \rightarrow$ more dense environments $\rightarrow$ difficult tracking!
- figures: [STDM-2014-17], [1406.7690]

Outline

Tracking in ATLAS
- Track finding
- Ambiguity solving
- Track fitting

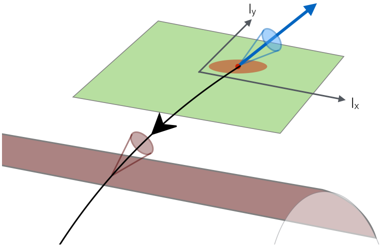
Performance
- Track reconstruction
- Lost tracks

Conclusion

Tracking in ATLAS

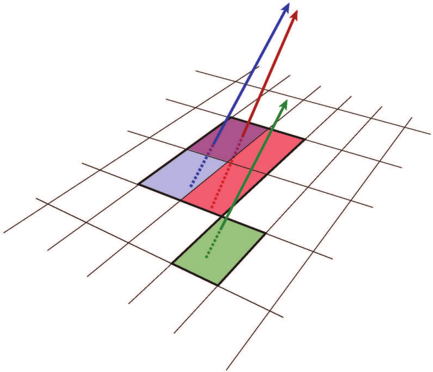
First step: track finding

- Detector measures space-points ≡ 3D coordinates of hits
- Next, define seeds: sets of 3 space-points passing p_T, impact-parameter cuts
- Use Combinatorial Kalman Filter to produce Track candidates
  - Get track parameters from seed (position, angle, charge)
  - Predict probability distribution of parameters for next hit
  - Use compatible space-points to update/create track candidate



Tracking in ATLAS

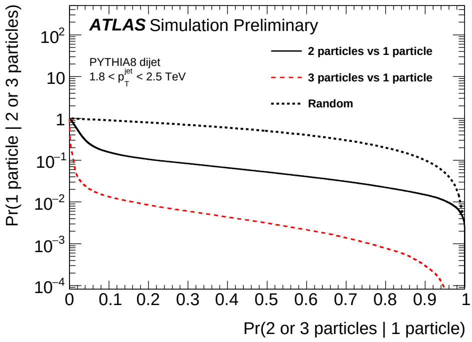
Second step: ambiguity solving



[1406.7690]

- Tracking finding can create an excess of track candidates
- Score tracks according to holes, fit χ^2, $log(p_T)$, ...
- If cluster used by multiple tracks, use neural network to decide if correct or not
  - Clusters identified as merged (i.e., multi-particle): no penalty $\rightarrow$ shared
  - Clusters not identified as merged are penalized
- Neural networks estimate probability that cluster comprises 1, 2 or ≥ 3 particles

Clustering neural network



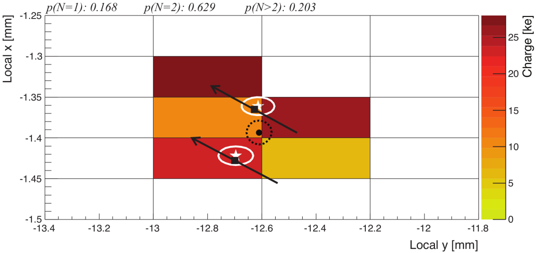
- Trained using the Keras python library
  - https://keras.io/

- Inputs:
  - 7x7 discretized charge matrix centered on charge centroid
  - 7D vector of pixel pitches
  - Angles of incidence of track candidate
  - Layer number and detector region (barrel or endcap)
- Output: 3D vector of class probabilities
  - 1-particle, 2-particles, ≥3-particles
- Trained on 12 million clusters from high-p_T dijet sample
  - 22% 1-particle clusters
  - 26% 2-particle clusters
  - 52% ≥ 3-particle clusters

Tracking in ATLAS

Track fitting

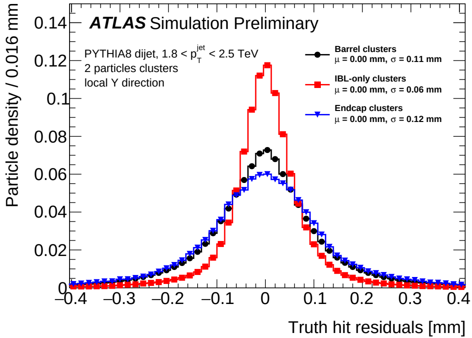
- Finally fit all tracks and measure final parameters
- Problem: shared clusters assigned only one hit position
- Use another set of neural networks to estimate the correct positions and improve track resolution



- ●: position obtained with linear interpolation
- ■: true hit position
- ⋆: Position obtained with neural network

[1406.7690]

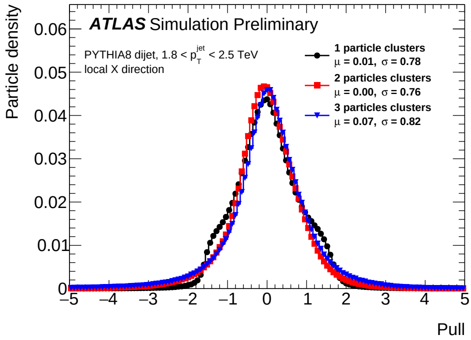
Tracking in ATLAS
Track fitting



- Neural network trained separatly for 1, 2 and 3 particle cases
- > 3 particle not considered
- Inputs: same as number neural network
- Output: $1/2/3 \times$ 2D position vectors
- Regression task: network has linear activation at output layer

Tracking in ATLAS

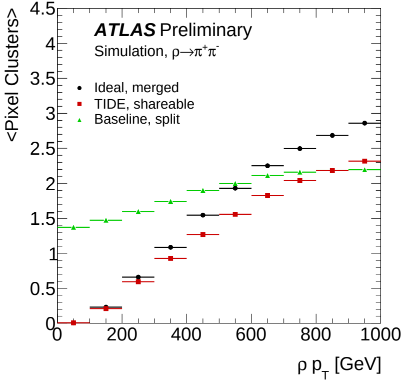
Track fitting



- Also estimate error on measurement using neural networks
- 6 neural networks: one for each number/direction pair
- Inputs: same as before + position estimation
- Output: 1, 2 or 3 binned probability distributions on residual
- Actually a classificaction task over the bins!
- Point estimate of error: rms of distribution

Performance

Track reconstruction

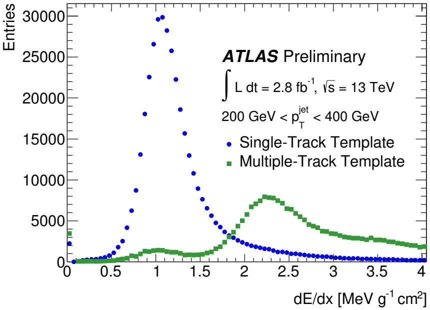


[ATL-PHYS-PUB-2015-006]

- measure average multiplicity of shared clusters in simulation
- True distribution in black
- Measured distribution with algorithms optimized for dense environment in red
- Measured distribution before optimization in green
- Optimized selection follows much more closely the true distribution

Performance

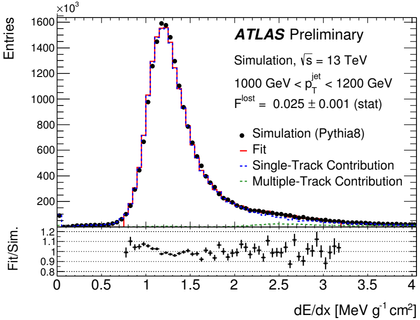
Lost tracks

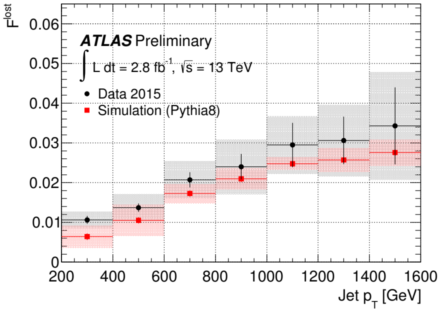


[ATL-PHYS-PUB-2016-007]

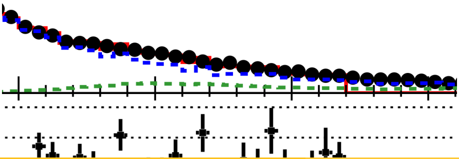
- dE/dX of MIP $\sim$ Landau
- Shifted if > 1 charge-particle fit to a Landau
- Blue: dE/dX of single-track clusters away from jet core
- Green: dE/dX of merged clusters inside jet core
- Select single-track clusters inside jet core, fit to weighted sum of both single and merged clusters templates

- If multiple-charged-particles cluster not split: used by one track only
- Merged clusters template will then be needed in dE/dX fit
  - Measures the inefficiency of tracking in dense environments





Zoom:



- Left: Fit in $1 < p_T^{jet} < 1.2$ TeV range. $F_{lost} \approx 0.03$
- Right: F_{lost} value in all p_T^{jet} bins
- Good agreement between data and simulation (up to 25%)

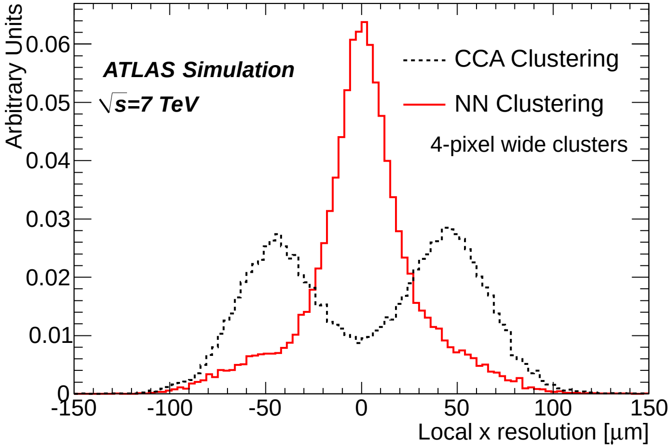
Conclusion

- The ATLAS track reconstruction has been optimized for dense environments that are more and more frequent at energy frontier
- Notably: now use three sets of neural networks to estimate number of particles contributing to clusters and to measure the true hit positions
- Performance very good: never lose more than 4% of tracks in high-p_T jet core for tracks up to $\approx$ 1.5 TeV
  - Measured by dE/dX fit
- All done without increasing CPU time

Thank you!

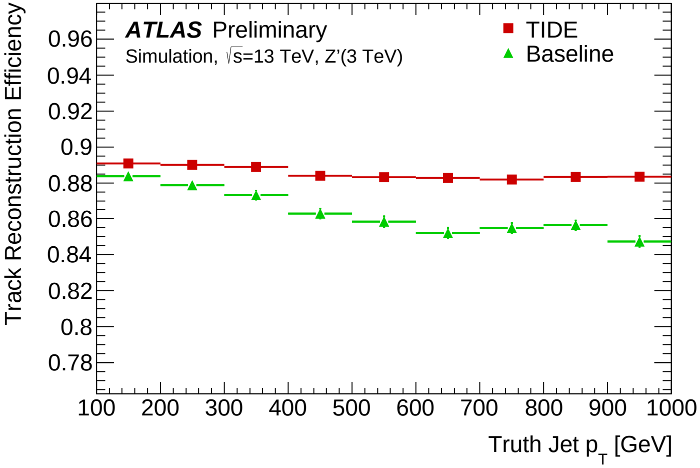
Backup slides

Truth hit resolution with/without neural network



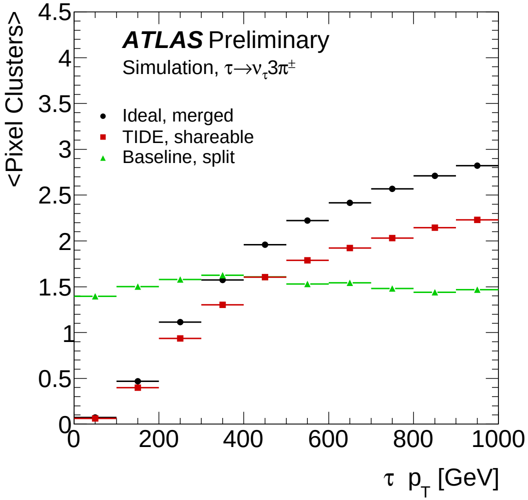
[1406.7690]

Tracking efficiency vs p_T



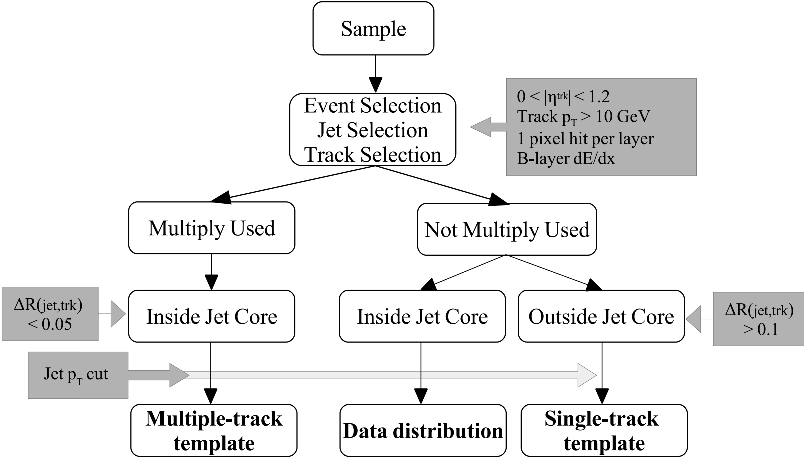
[ATL-PHYS-PUB-2015-006]

Shared cluster multiplicity, $\tau \to \nu 3\pi^{\pm}$



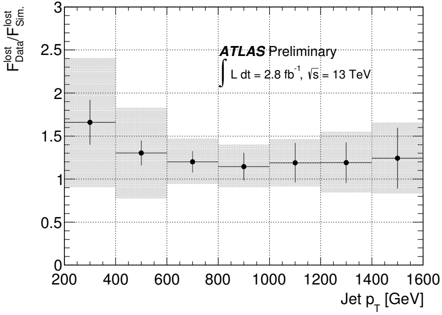
[ATL-PHYS-PUB-2015-006]

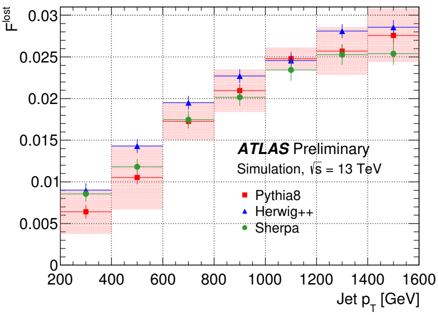
dE/dX templates definitions



[ATL-PHYS-PUB-2016-007]

F_{lost}: data & simulation agreement





[ATL-PHYS-PUB-2016-007]