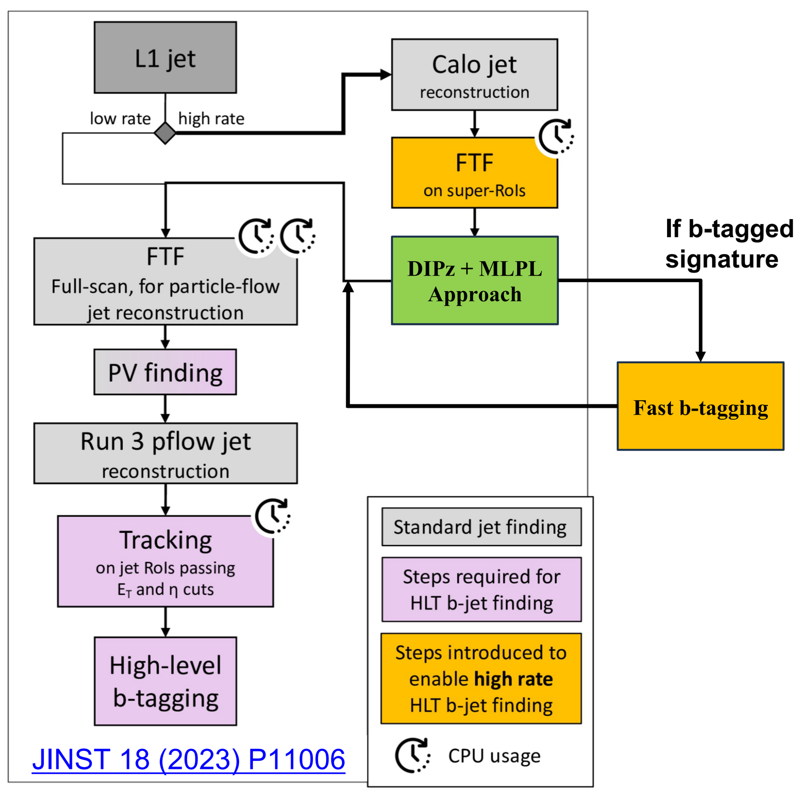
Fast Pile-Up Jet Rejection in the ATLAS HLT Using the DIPz[1] + MLPL[2] Approach

[1] Deep-sets model leveraging Impact Parameter information to regress jet's origin along the beamline z-axis

[2] Maximum Log Product of Likelihoods discriminant variable for event-level pile-up rejection

Motivation

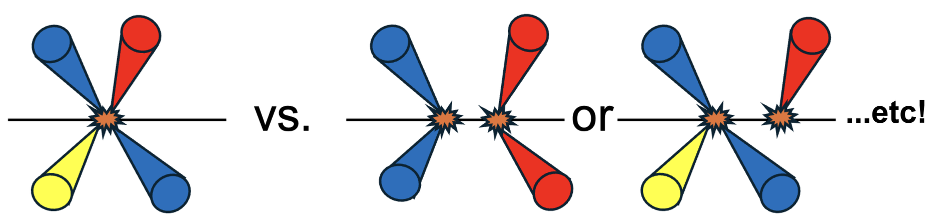
Jet pathway inside the ATLAS High-Level Trigger (HLT):



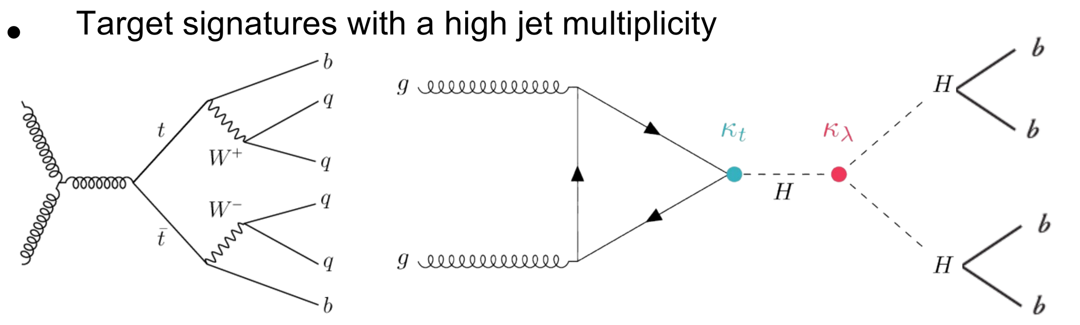
JINST 18 (2023) P11006

- CPU difficulties → Running full-scan FTF (Fast Track Finding) on all the interesting jet events that pass L1 is computationally infeasible
- Background events (events that contain pileup jets) are reducible if we can quickly identify and remove them before full-scan FTF
- Fast b-tagging to pre-select events provides a nice mitigation to such problem for b-tagged jet signatures, but:
  - Can we do better?
  - What about other flavour-agnostic jet signatures?

Four-jet Signatures Example:



- Target signatures with a high jet multiplicity



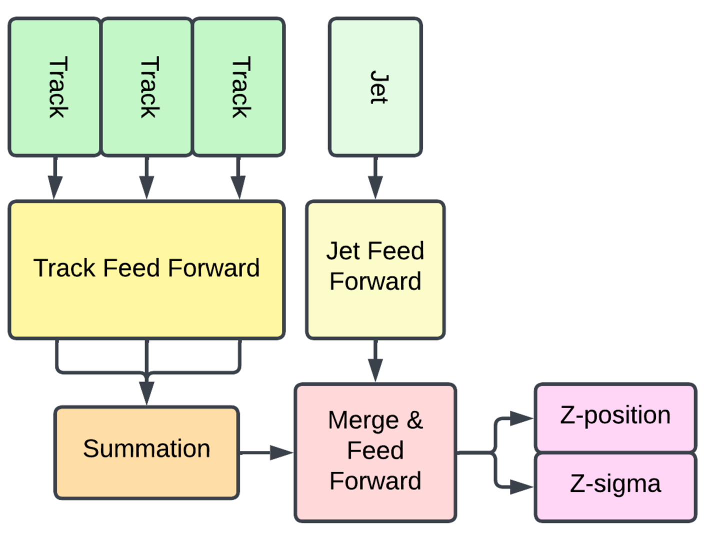
DIPz Architecture

DIPz is an uncertainty-aware ML model that leverages the permutation-invariant Deep Sets architecture and uses Impact-Parameter information as inputs to regress a jet's vertex beamline z-position, summarized formally as:

$$z_{pred} = F\left(\Phi_{jet}(j) \oplus \sum_{i=1}^{n} \Phi_{track}(t_i)\right)$$

Where:

- t_i: feature vector of the i-th track,
- j: vector of jet-level features;
- Φ_{track}: shared feed-forward network applied to each track;
- Φ_{jet}: feed-forward network applied to jet features;
- $\sum_{i=1}^{n}\Phi_{track}(t_i)$: sum over all processed tracks (Deep Sets aggregation)
- $\oplus$: concatenation operator (applied for the jet and track latent representations);
- F: post-merge feed-forward network that predicts the vertex z-position and uncertainty;
- z_{pred}: two predicted quantities μ_z and σ_z



DIPz Loss Function

$$Loss\,(z_{target}, \mu_z, \sigma_z) = -\log\left(\mathcal{L}\,(z_{target}, \mu_z, \sigma_z)\right)$$

derived from the Gaussian likelihood "$\mathcal{L}$":

$$\mathcal{L}(z_{target}, \mu_z, \sigma_z) = \frac{1}{\sqrt{2\pi}\,.\,\sigma_z}\, e^{-\frac{1}{2}\left(\frac{z_{target}-\mu_z}{\sigma_z}\right)^2}$$

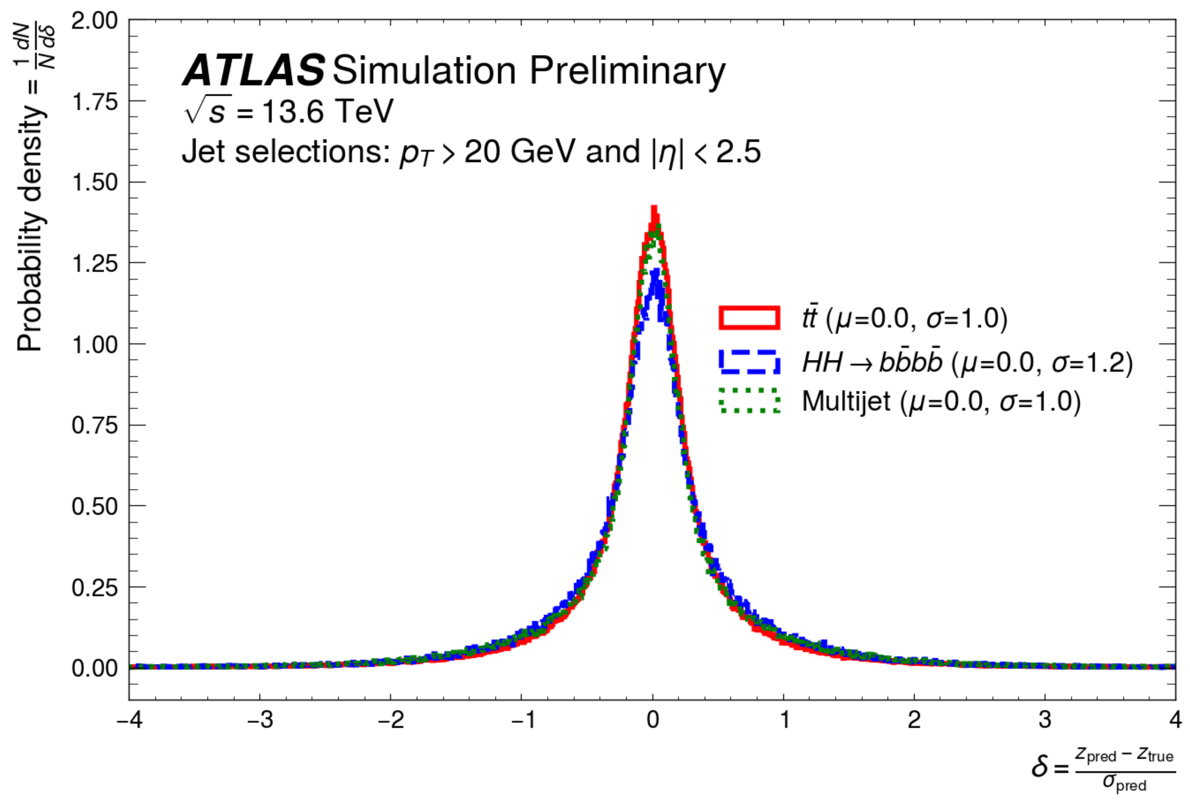
DIPz Inputs and Training

- Jets inputs: Jet's transverse momentum and pseudo-rapidity
- Track inputs:

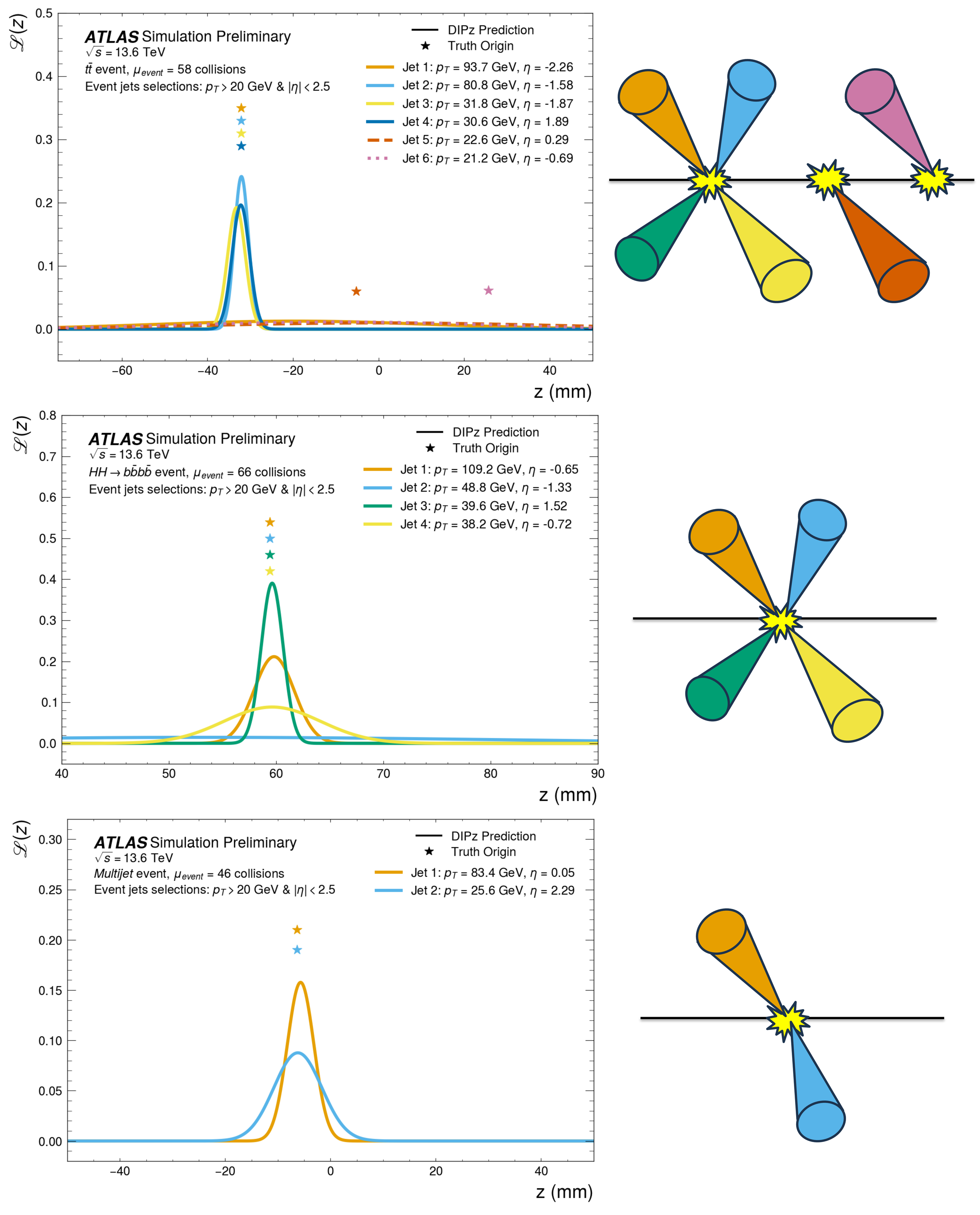
Track Feature	Description
p_T	Transverse momentum
d_0	Distance of closest approach of track to the beamline in the transverse plane
z_0 relative to beamspot	Displacement between beamspot center and closest approach to the beamline, projected along beamline
$\Delta\eta$	Pseudorapidity of track relative to jet η
$\Delta\phi$	Azimuthal angle of track relative to jet ϕ
q/p	Track constituent particle charge divided by its momentum
χ^2	$\sum_{\text{hits on track}}(r/\sigma_r)^2$, where r ≡ hit residual, σ_r ≡ residual uncertainty
number DoF	Number of degrees of freedom in track fit

- The training is performed using 3 million jets (with a separate validation set of 600,00 jets) from simulated top-pair production events from proton-proton collisions at a centre-of-mass energy of 13.6 TeV

DIPz Regression Performance



DIPz Event Displays

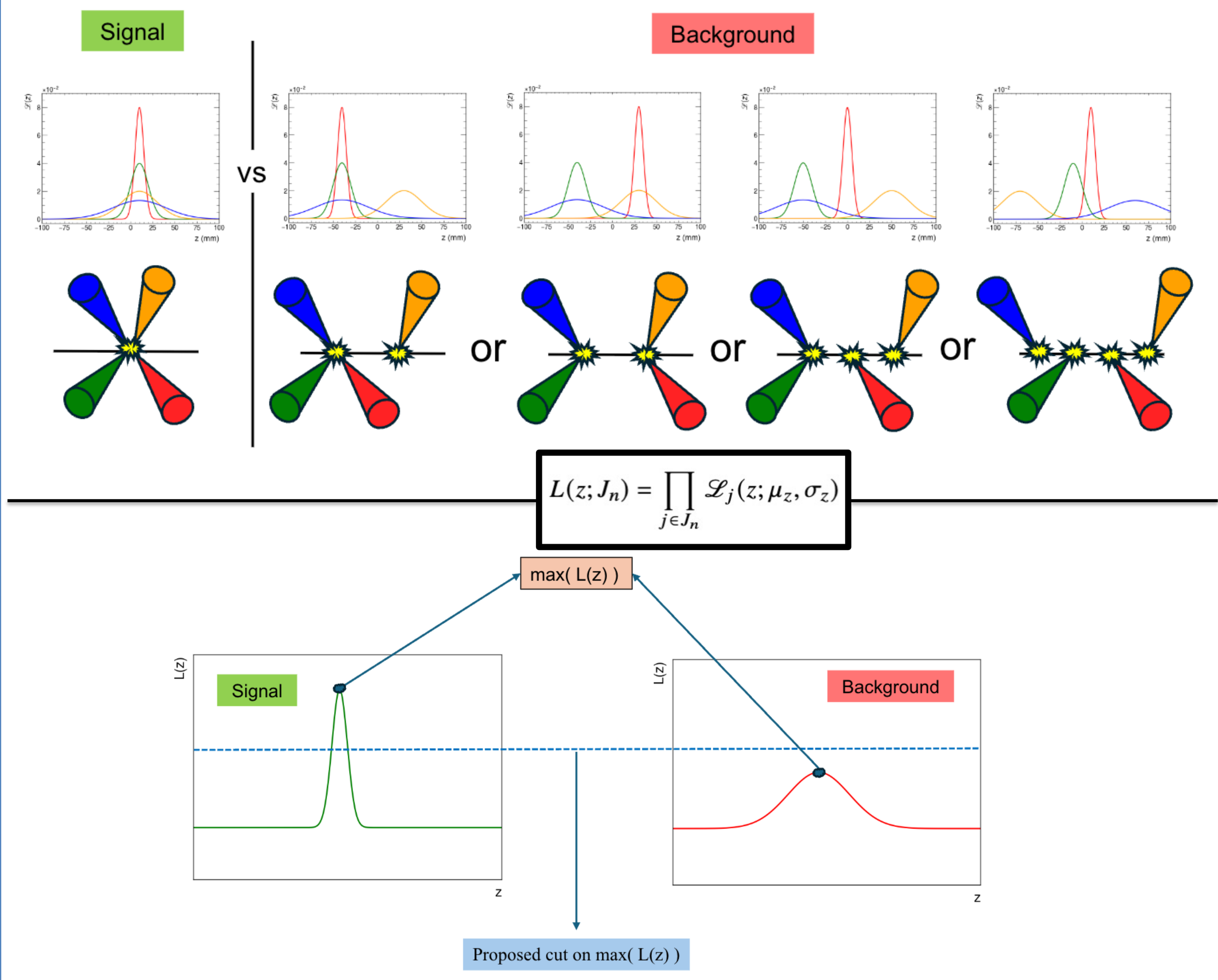


MLPL Algorithm Description

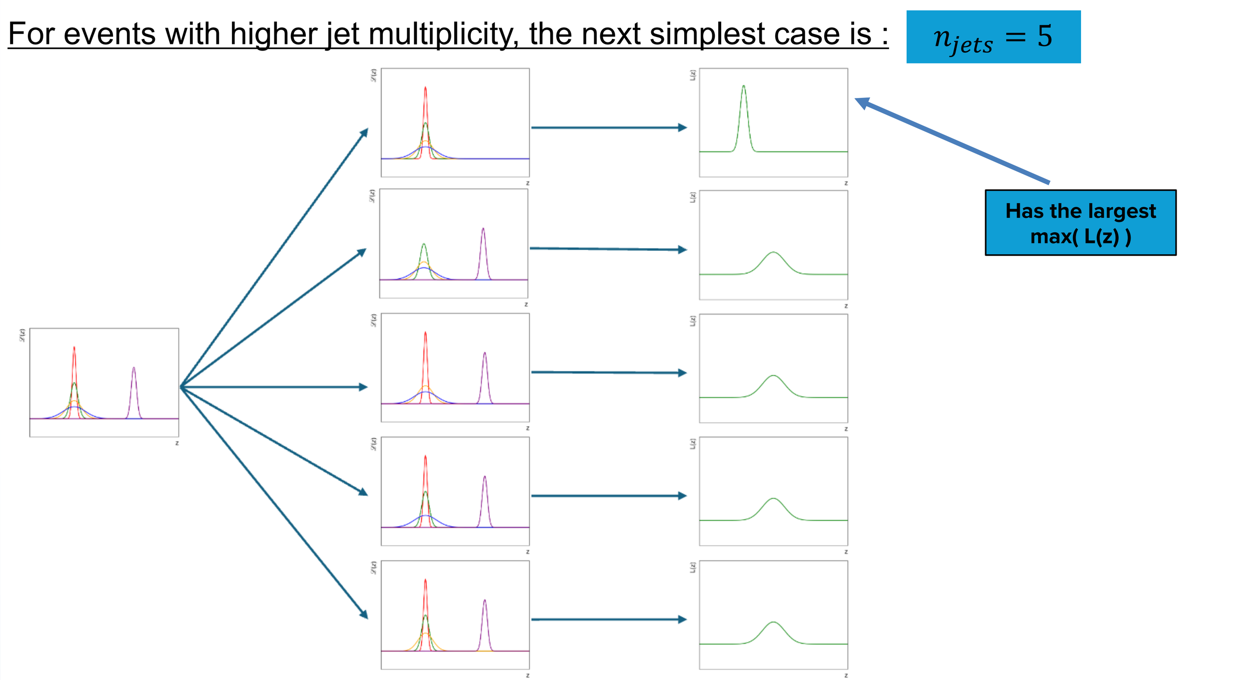
The approach relies on constructing an event-level discriminant based on the per-jet vertex predictions provided by DIPz. For a given n-jet final state selection, a cut is optimized and applied to this discriminant to reject pile-up-dominated events efficiently, enabling fast decision-making within tight latency constraints.

A Four-Jet Selection Example

For events with 4 (and only 4) selected jets: $n_{jets} = 4$



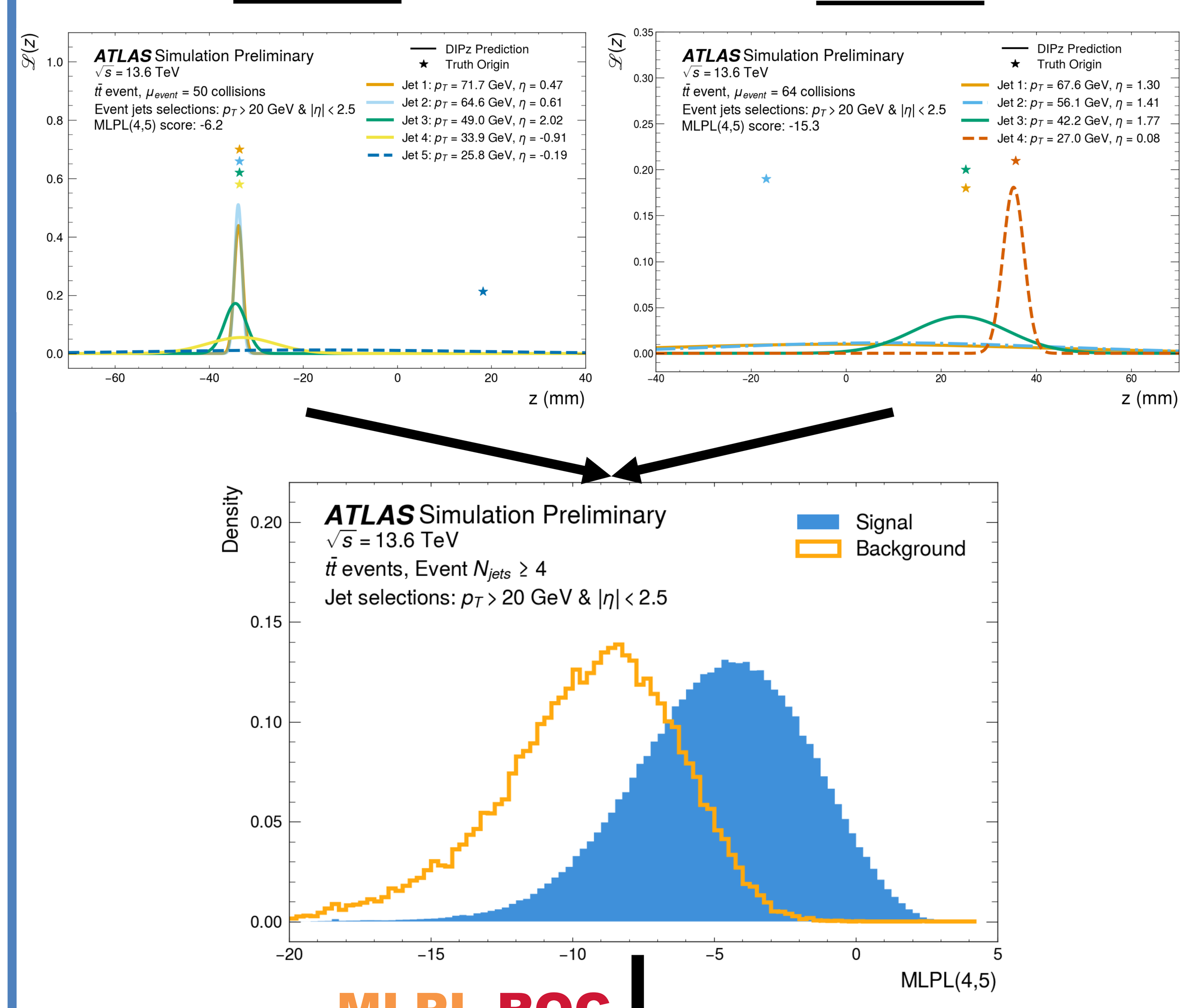
For events with higher jet multiplicity, the next simplest case is : $n_{jets} = 5$



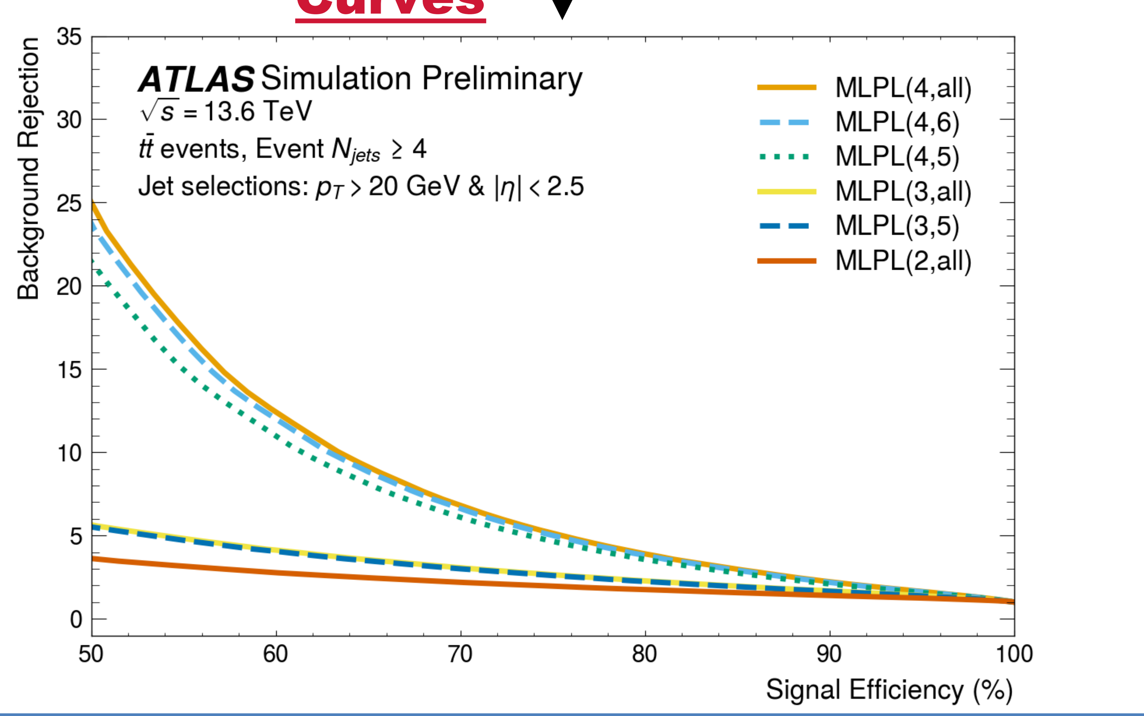
•MLPL(n,m): Maximum "over the combinations of n jets" of the maximum of the Log of the Product of the Likelihood functions of the m highest pT jets in the event.

MLPL Algorithm Performance

Four-jet Signal candidate | **Four-jet Background candidate**



MLPL ROC Curves



Final Remarks

- The DIPz + MLPL approach has been implemented in the ATLAS trigger system software
- Deployed at Point 1, it is used in taking ATLAS collision data, optimizing chains targeting multijet signatures in 2025

Mohammed (Mo) Abdelrazek Aboelela
(mo.abdellrazekk@cern.ch)
On behalf of the ATLAS Collaboration

Scan for public results