XII International Conference on New Frontiers in Physics

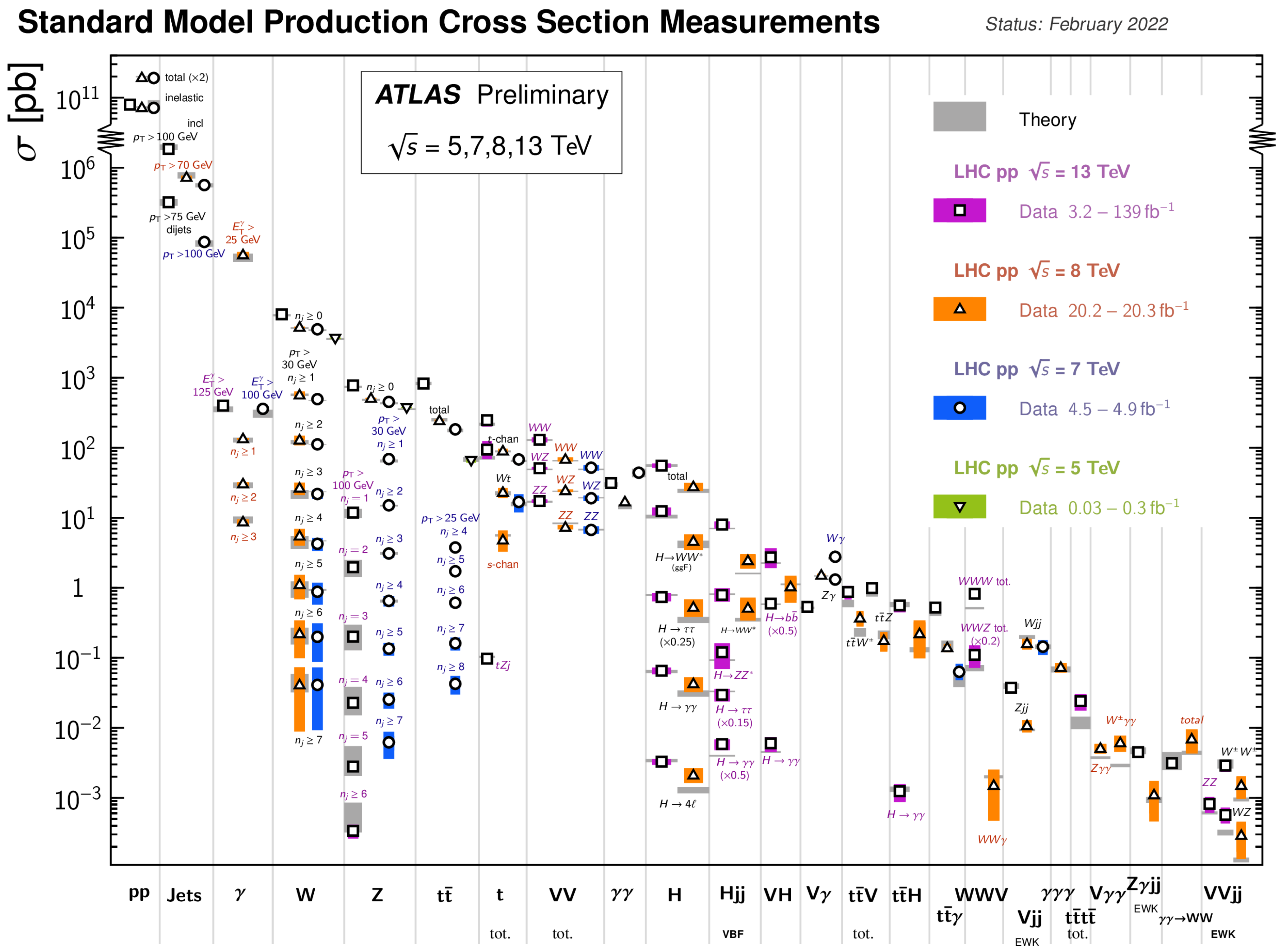
Improving ATLAS Hadronic Object Performance with ML/AI Algorithms

Francesco Cirotto
On behalf of ATLAS Collaboration

Jets in High energy Physics

Precision is crucial

- Jets are formed by showers of particles originating from the hadronization of quarks and gluons
- Fundamental objects in Standard Model analyses and Beyond Standard Model searches
- A good hadronic object reconstruction translates into significant physics results
- Jets in ATLAS: Produce tracks in the trackers and energy deposits in the calorimeters, that are clustered together to obtain the properties of the initial quark or gluon



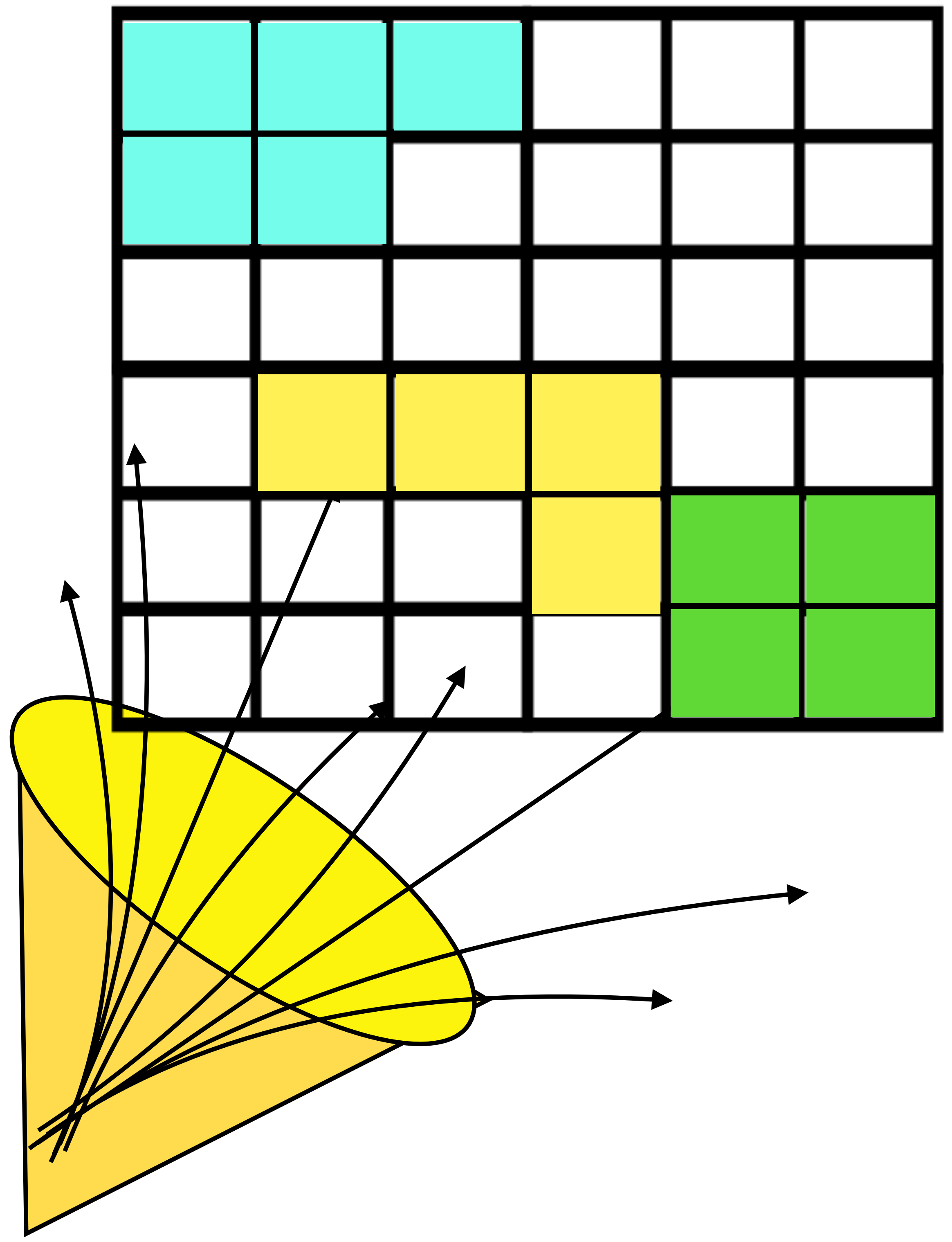
ATL-PHYS-PUB-2022-009

JETS IN ATLAS

Towards full jet building

- **Inputs/constituents**: jets are built from energy deposits in ID tracker and calorimeters
- **Reconstruction**: group constituents with a dedicated algorithm
- **Calibration**: techniques to determine and measure the detector response to jets
- **Tagging**: study of jet substructure to identify the originating jet particle
- The full chain is really long and difficult to process

  ↪A great field for Machine Learning application!



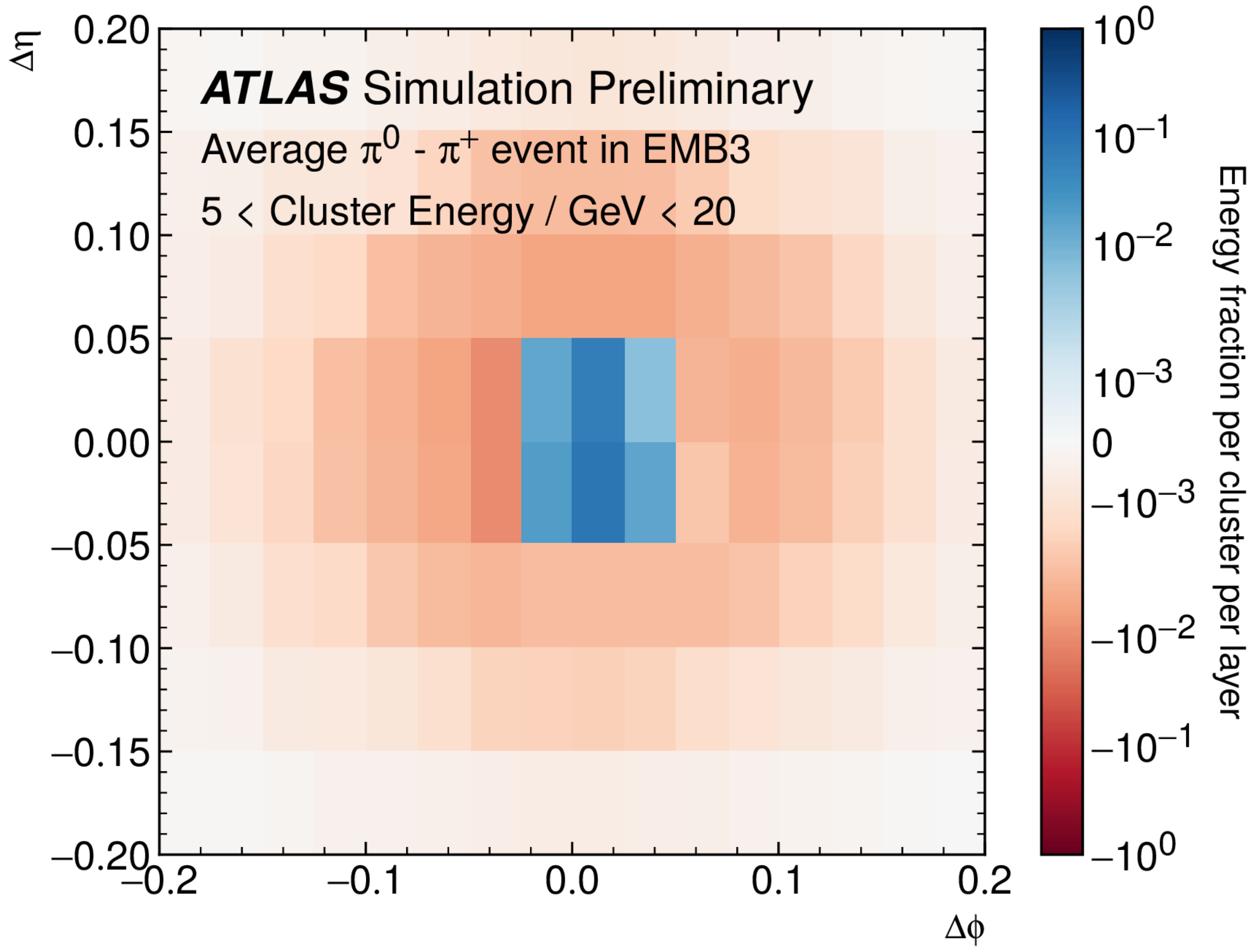
ML application covered in this talk

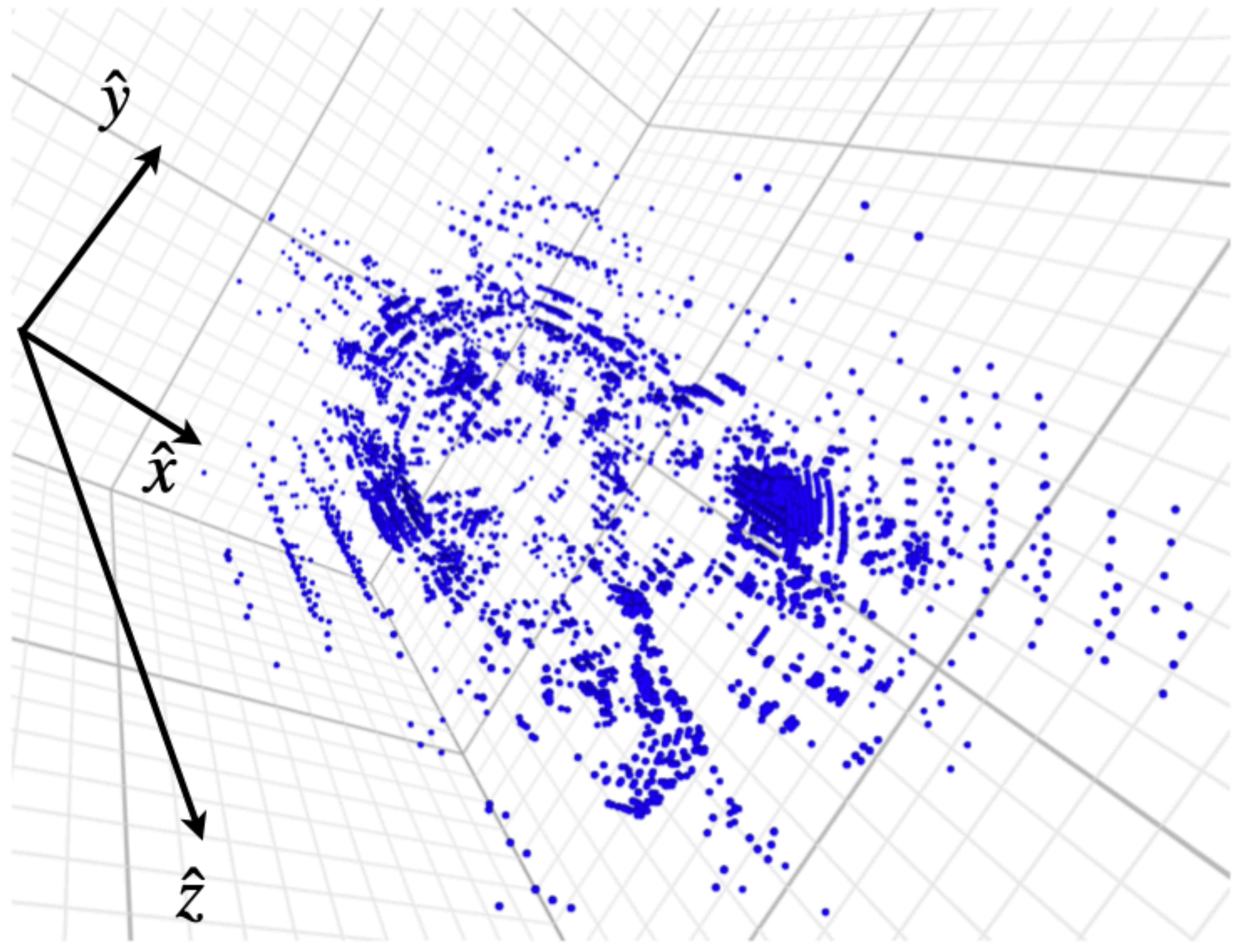
- Regress truth-level quantities from detector-level information
- Classify type of object

CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

Pion reconstruction

- Classify pions as charged or neutral
  - ↪ATLAS has non compensating calorimetry: energy deposits from charged and neutral pions need to be restored to different scales
- Starting point **topoclusters** definition: three-dimensional clusters of topologically-connected calorimeter cells
- Current approach:
  - ↪Cluster classification based on geometric and signal moments per-cluster
  - ↪Cluster Calibration through Local Cell Weighting (LCW) based on local properties
- New developments: find a new representation for topoclusters as input for ML algorithms
  - ↪Topoclusters as Images → Convolutional Neural Network (CNN)
  - ↪Topoclusters as point clouds → Deep Sets/ ParticleFlow Network (PFN)
  - ↪Topoclusters as graphs → Graph NN (GNN), Transformer
  - ↪Topoclusters 4-momenta → Deep Neural Network (as baseline)
  - ↪Architectures employed for different tasks (classification/calibration)





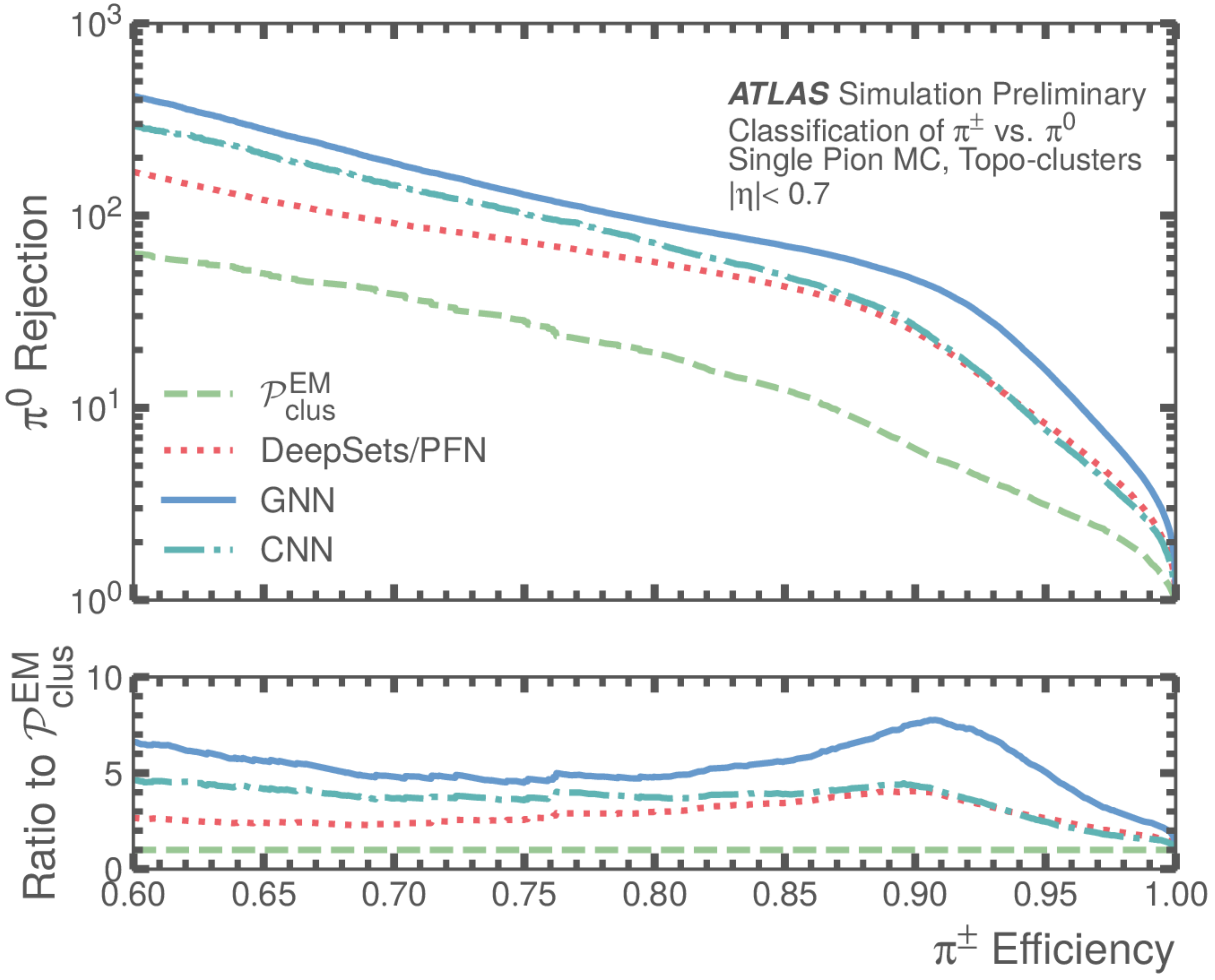
Point clouds: collections of points in space representing a three-dimensional object

ATL-PHYS-PUB-2022-040

CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

Pion classification

- First step in cluster calibration: differentiate EM from hadronic clusters
- Performance evaluated by comparing the rejection rate of π^0 as function of $\pi^\pm$ efficiency
  ↪Baseline method $\mathscr{P}^{EM}_{clus}$
- GNN is the best in overall: improving ~5 times rejection in the full pseudo-rapidity range

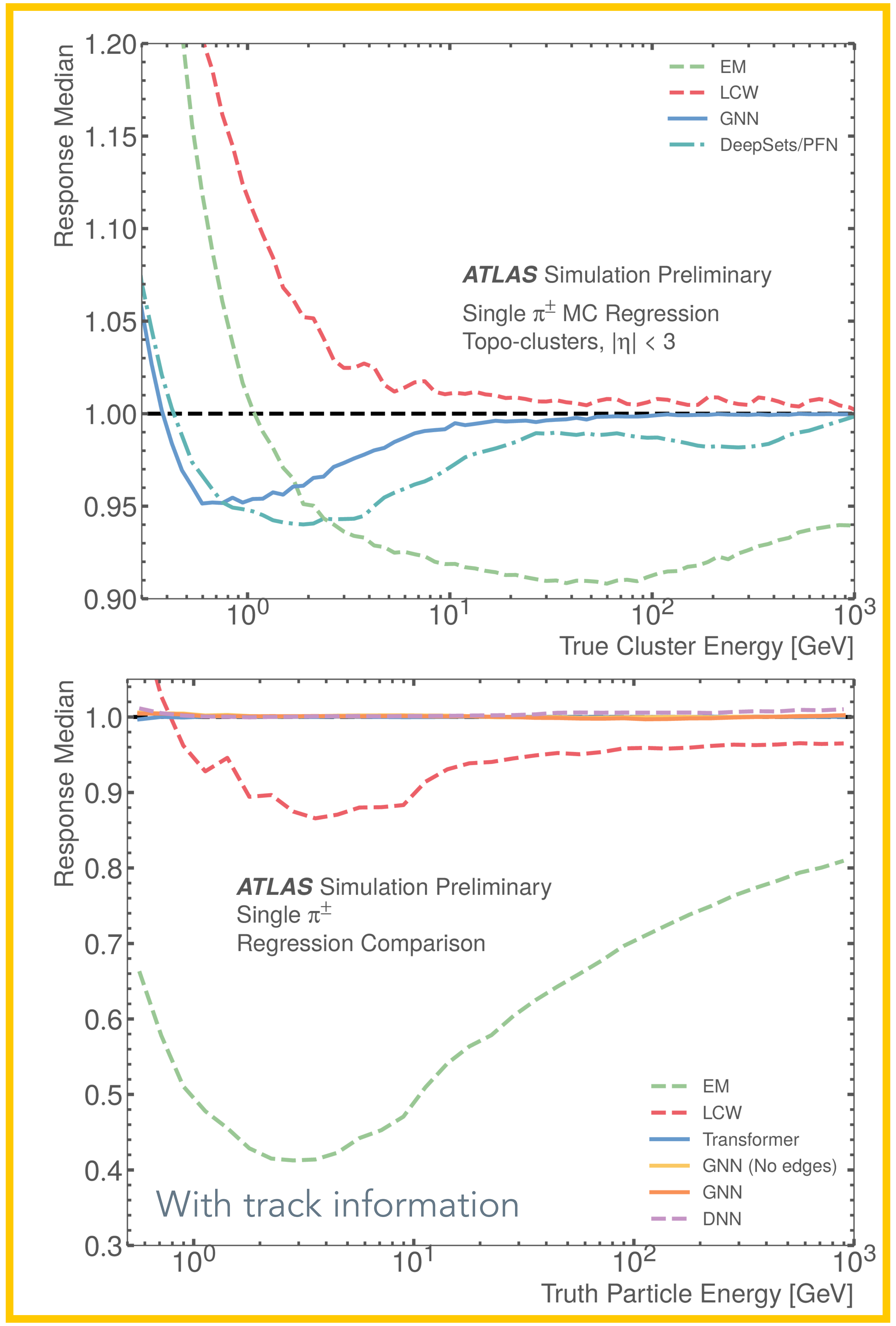


CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

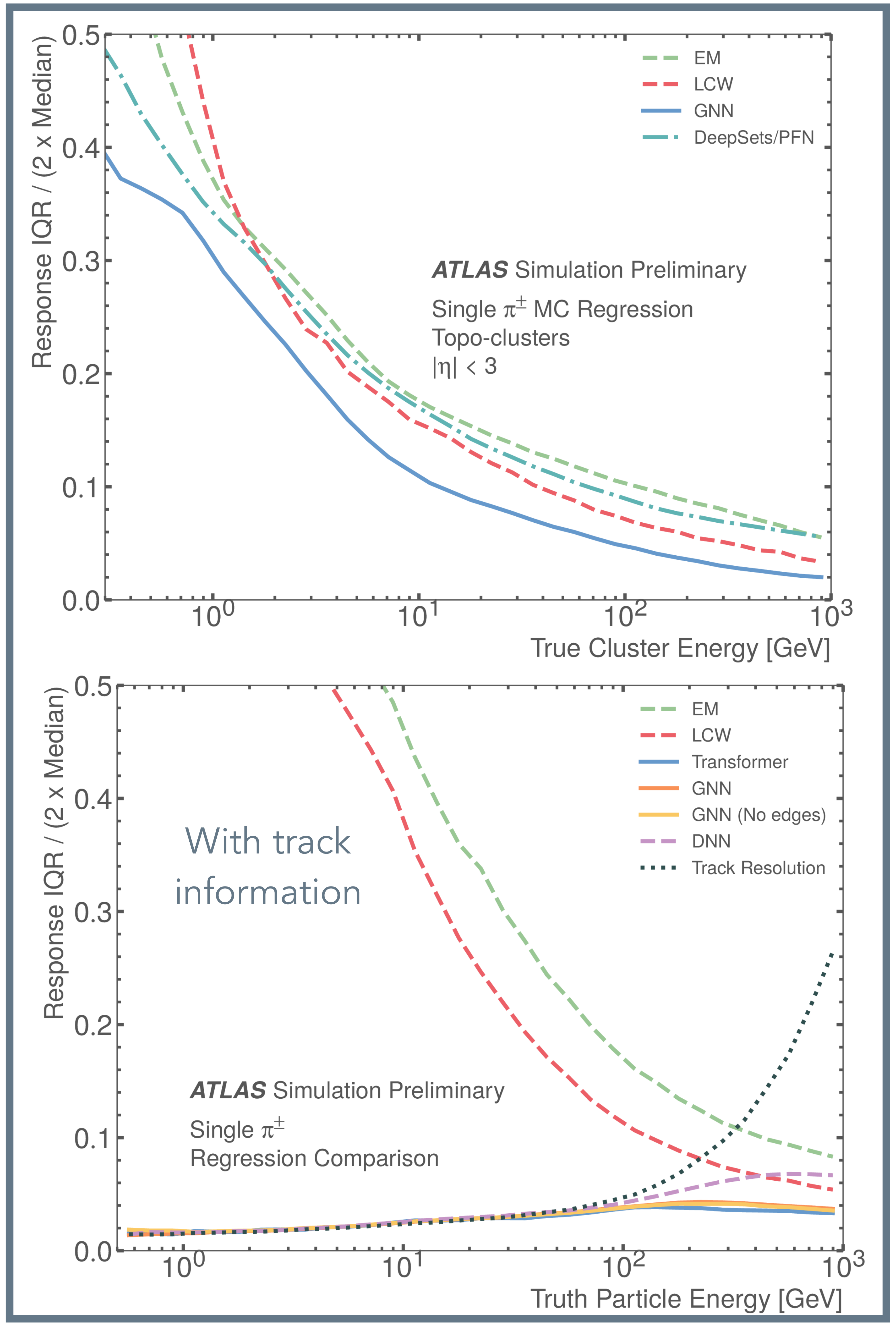
Pion energy calibration

- Calibrating the classified cluster energy response
- Comparison with respect to calibrated (LCW) and uncalibrated (EM)
  - ↪All ML models significantly improves baseline methods
- Including track information
  - ↪Expected to complement and interplay differently with model performance than calorimeter cluster information
  - ↪Further improvements in resolution

Energy response



Energy resolution

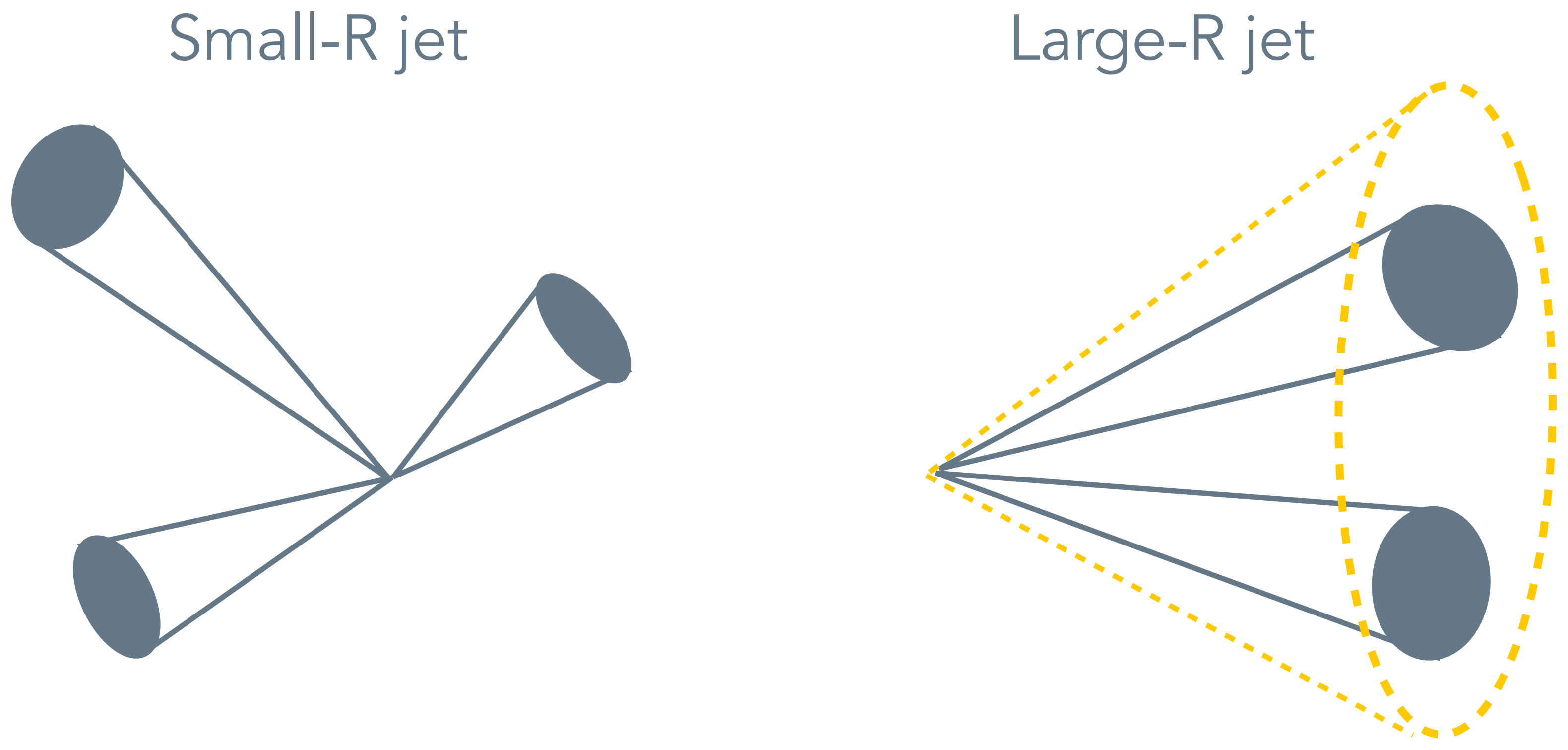


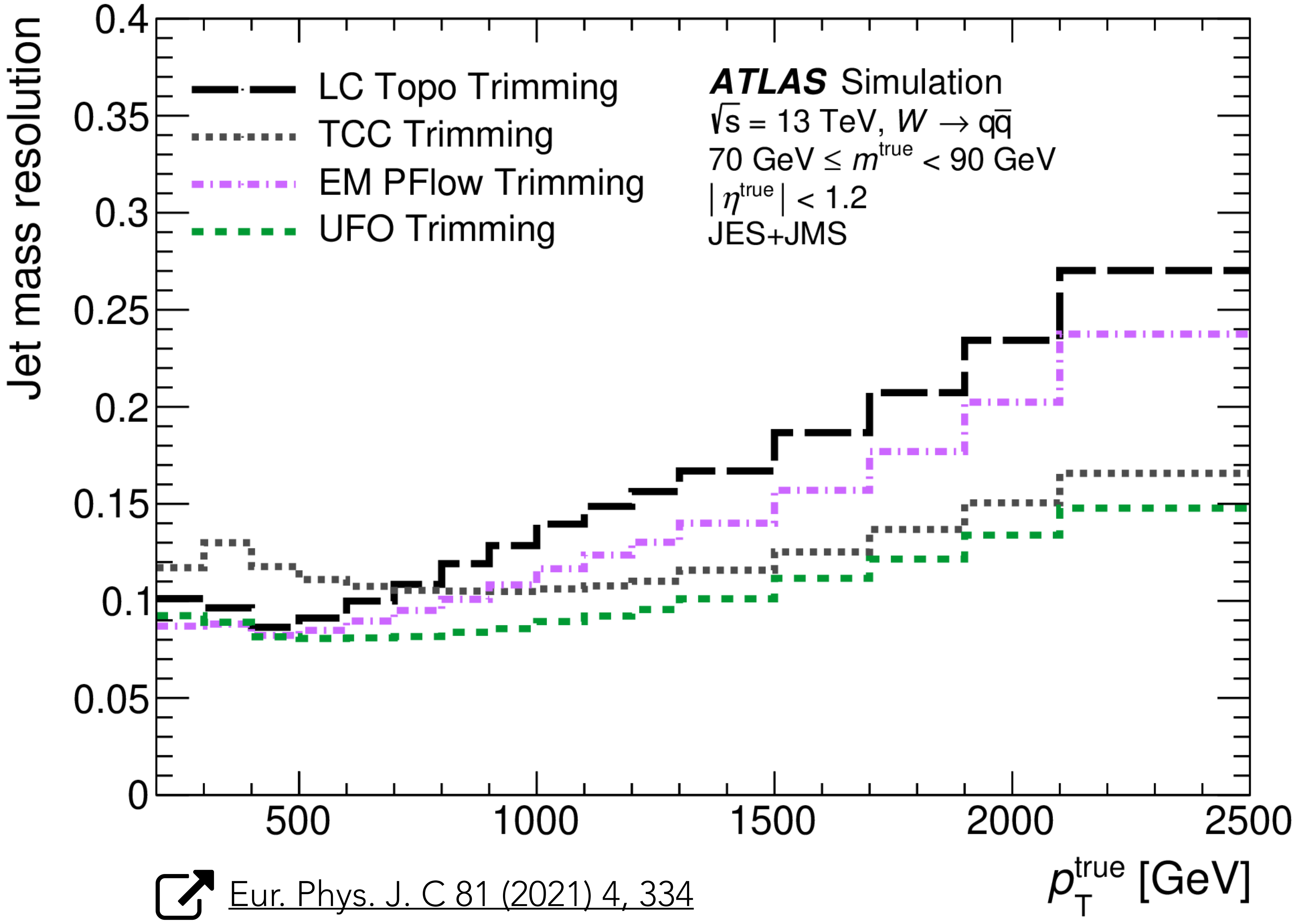
ATL-PHYS-PUB-2022-040

BOOSTED JET TAGGING

UFO: a new jet definition for large-R jet

- Massive particles (W/Z/H/top) with large Lorentz boosts have decay products collimated in the direction of progenitor particle
  - ↪Advantageous to reconstruct hadronic decay products as a single large-radius (large-R, where R = 1.0) jet
  - ↪Allows to better capture multi-pronged jet substructure information
- Unified Flow Objects: current state of the art on large-R jet definition. It is a combination of two algorithm:
  - ↪Particle Flow (PFlow): combine track and topocluster information; tracks with good momentum resolution extrapolated to calorimeter, cell-by-cell subtraction of their deposited energy. **Better resolution at low p_T**, pileup separation
  - ↪Track-CaloClusters (TCC): use tracks to split up large clusters based on the energy flow and their direction. **Improvement of the mass resolution at high p_T**
  - ↪UFO combines both advantages: improved pile-up resilience and jet mass resolution
- Taggers distinguish large-radius jets from massive particles from light quark/gluon-initiated jets
  - ↪Use jet substructure information
  - ↪Enhances performances of BSM searches and precision SM measurements
  - ↪ML techniques can improved standards cut-based taggers



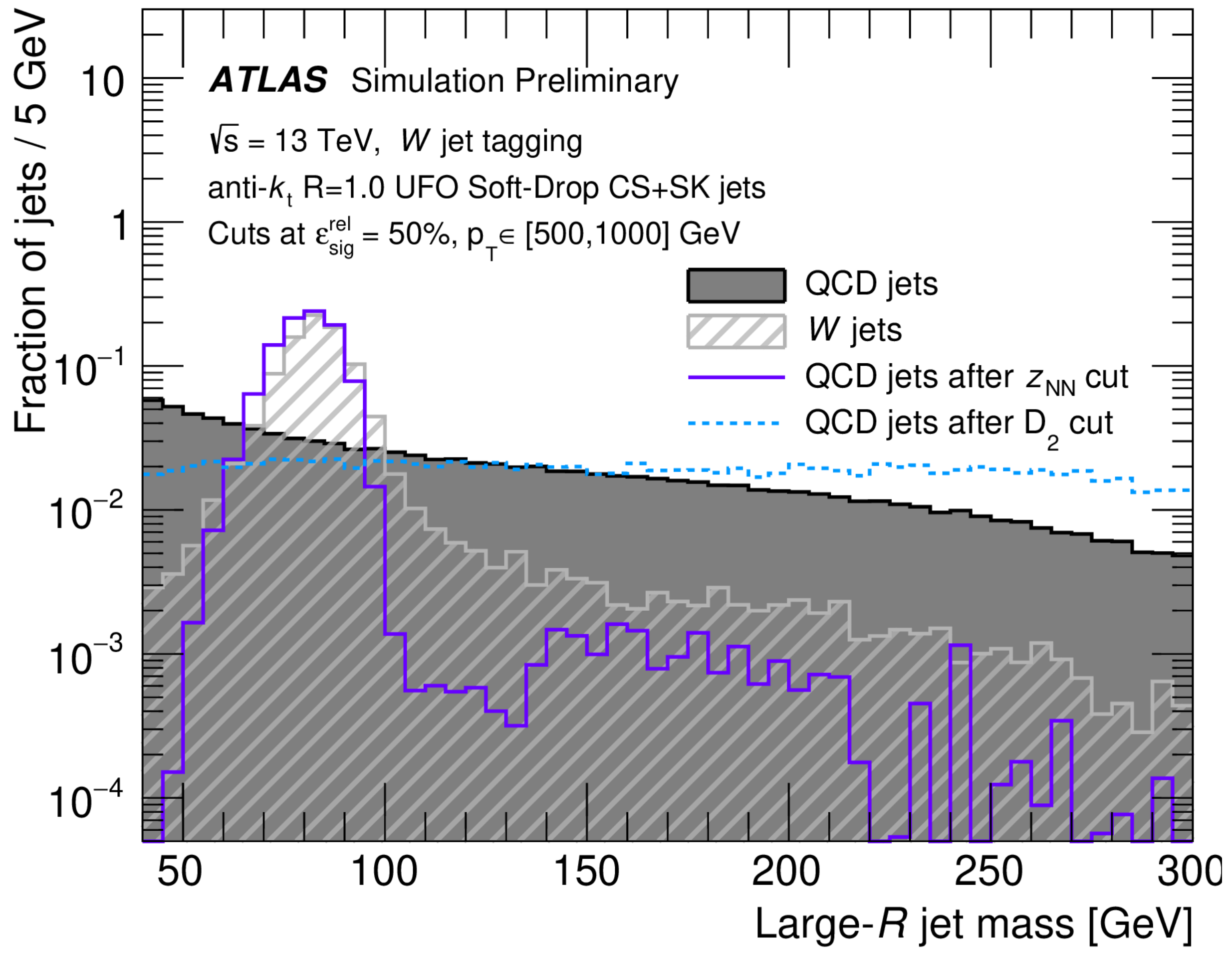


Eur. Phys. J. C 81 (2021) 4, 334

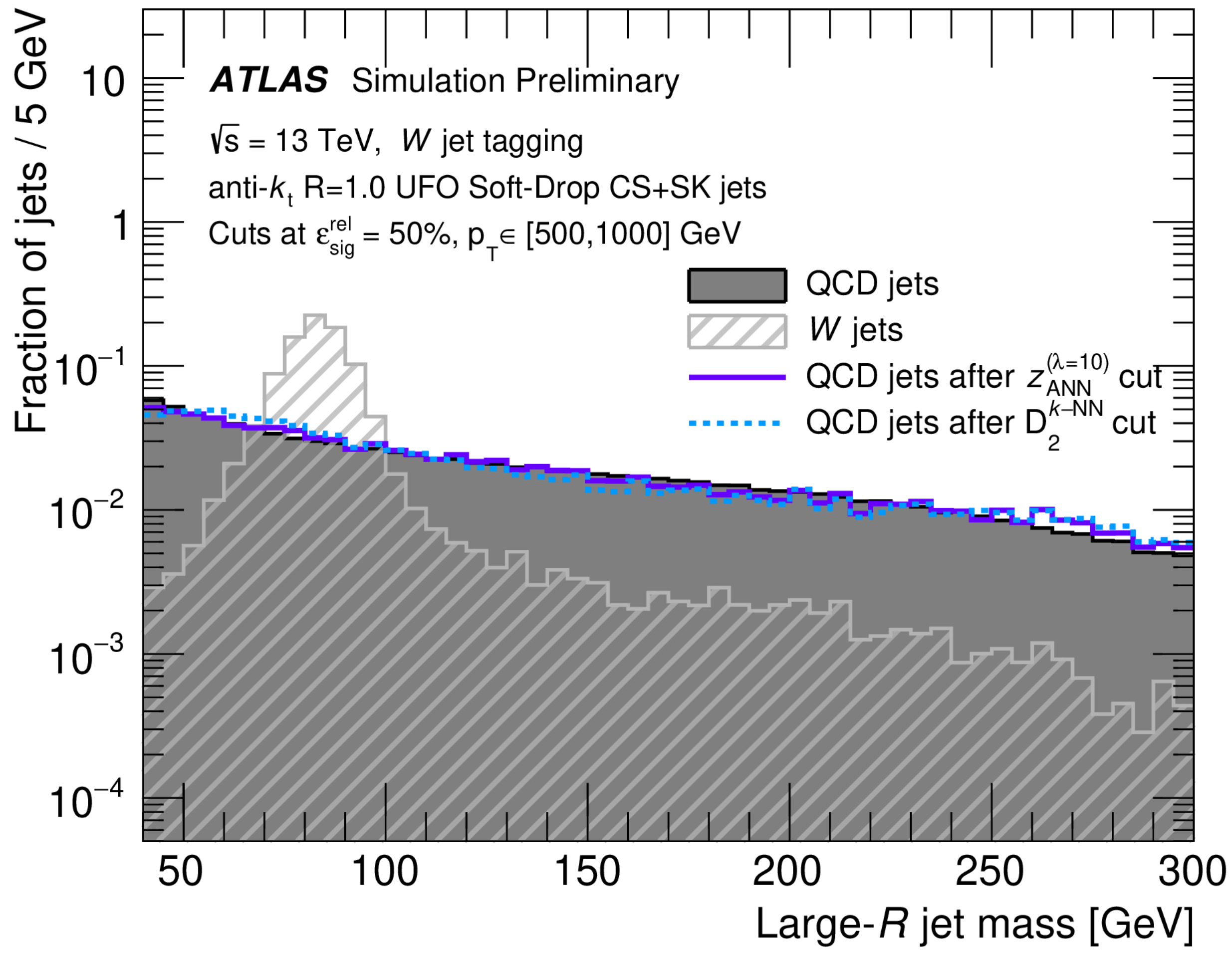
Boosted Jet Tagging

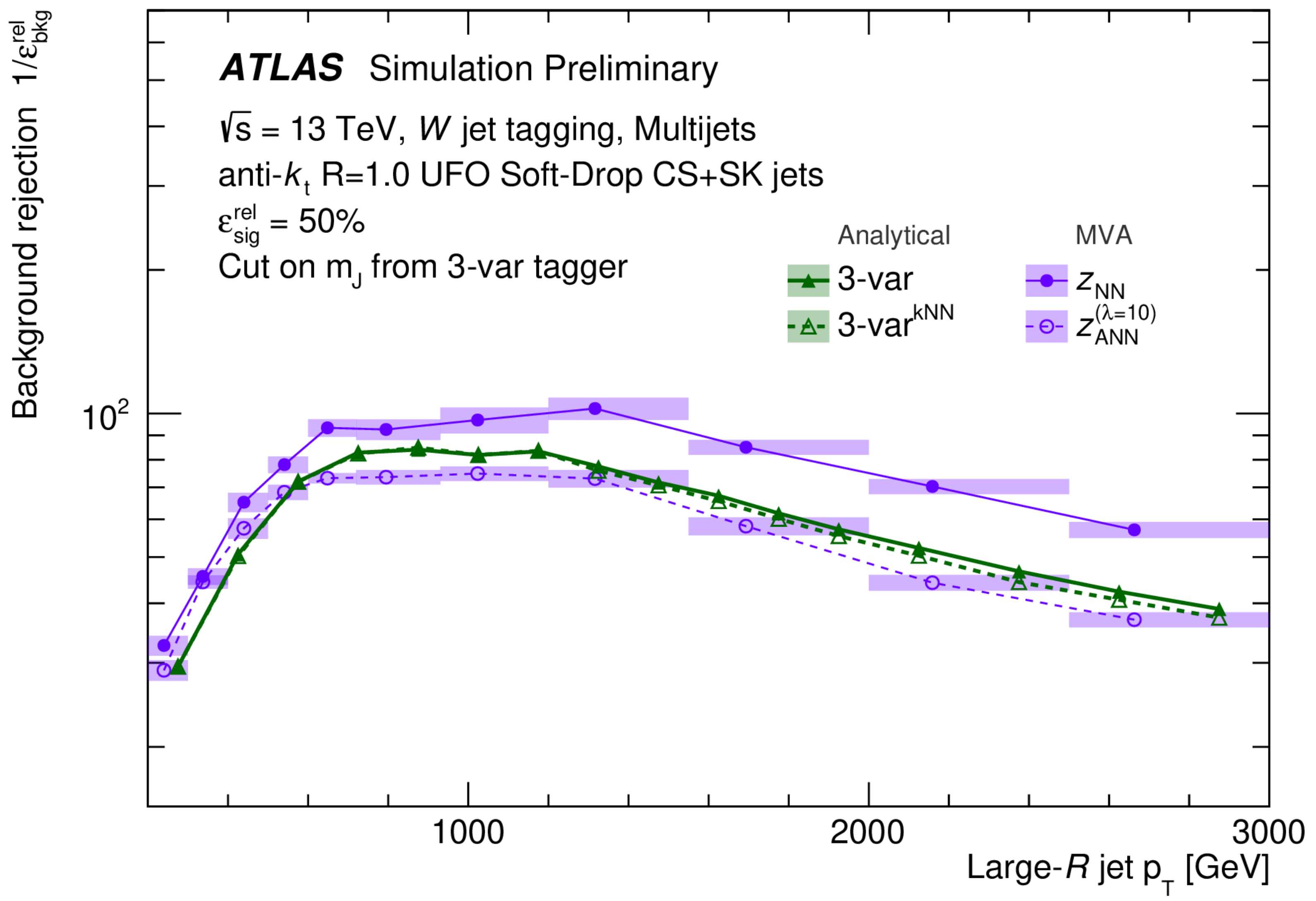
W/Z taggers

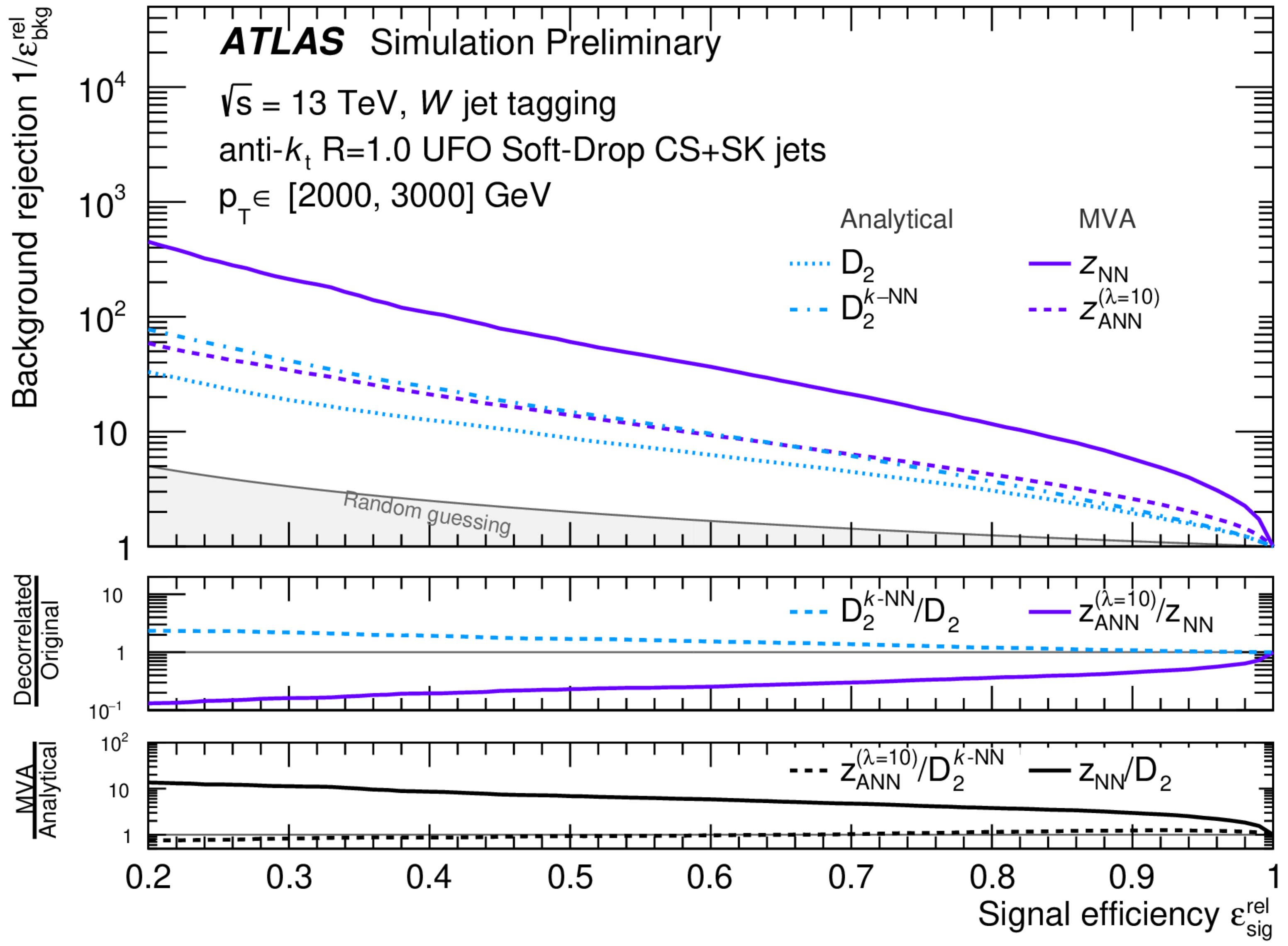
- Two taggers used at the moment:
  - ↪ Three variables cut-based tagger uses rectangular cuts on substructure variables
  - ↪ ML based tagger based on a DNN using substructure variables and track information
- Both methods introduced high mass correlation
  - ↪ The mass sculpting is problematic for background estimation
  - ↪ A decorrelation procedure needs to be applied to increase



After decorrelation







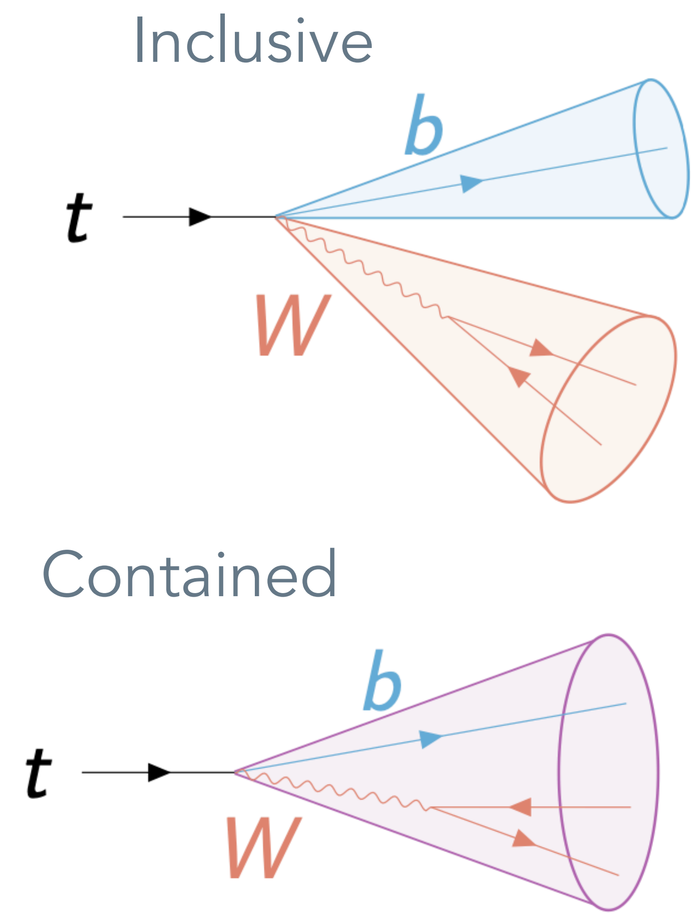
- For DNN approach an Adversarial NN used to decorrelate jet mass
  - ↪ Small decrease in performance recovered with analysis-specific mass window cuts

ATL-PHYS-PUB-2021-029

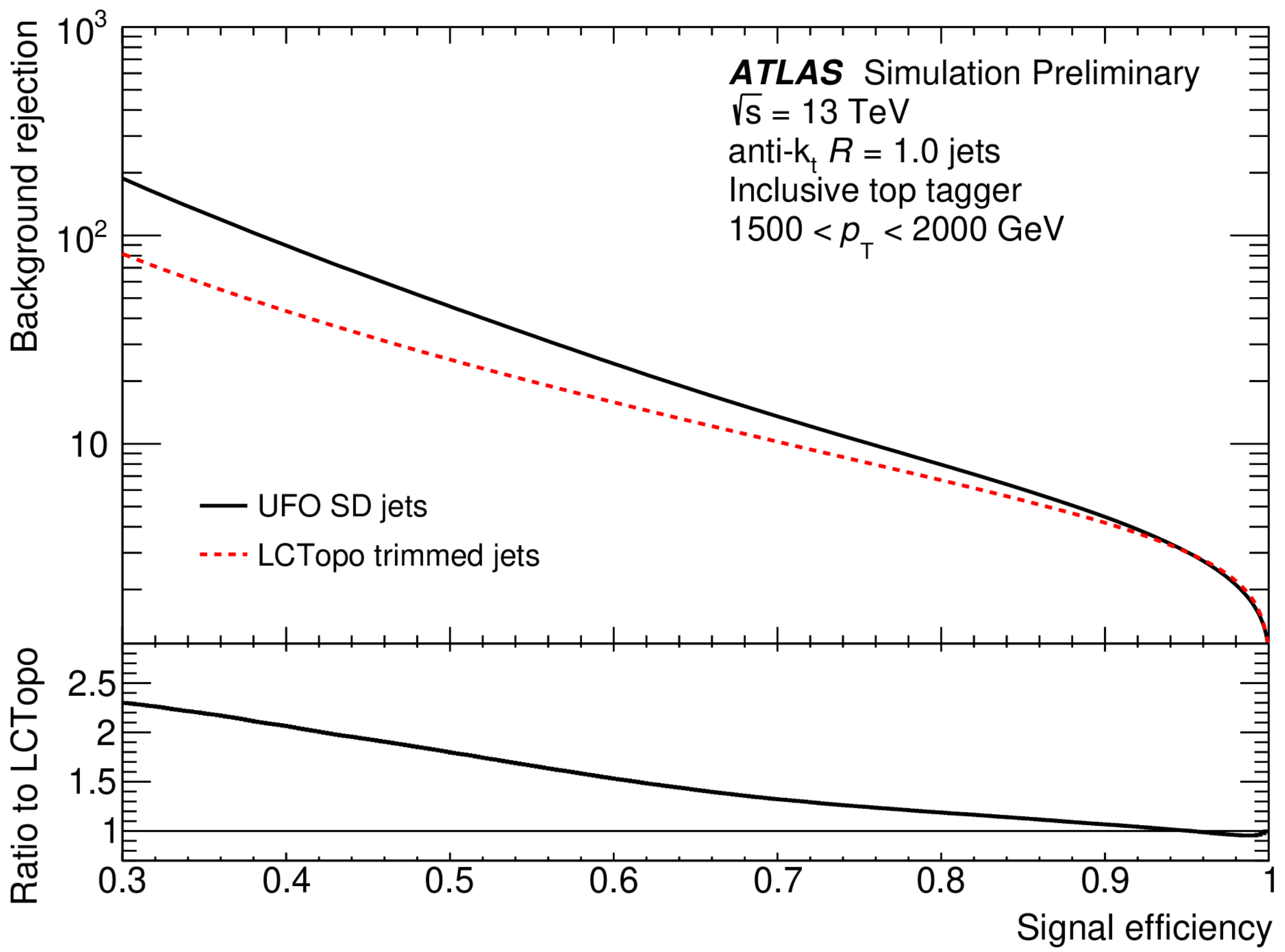
JETM-2022-06

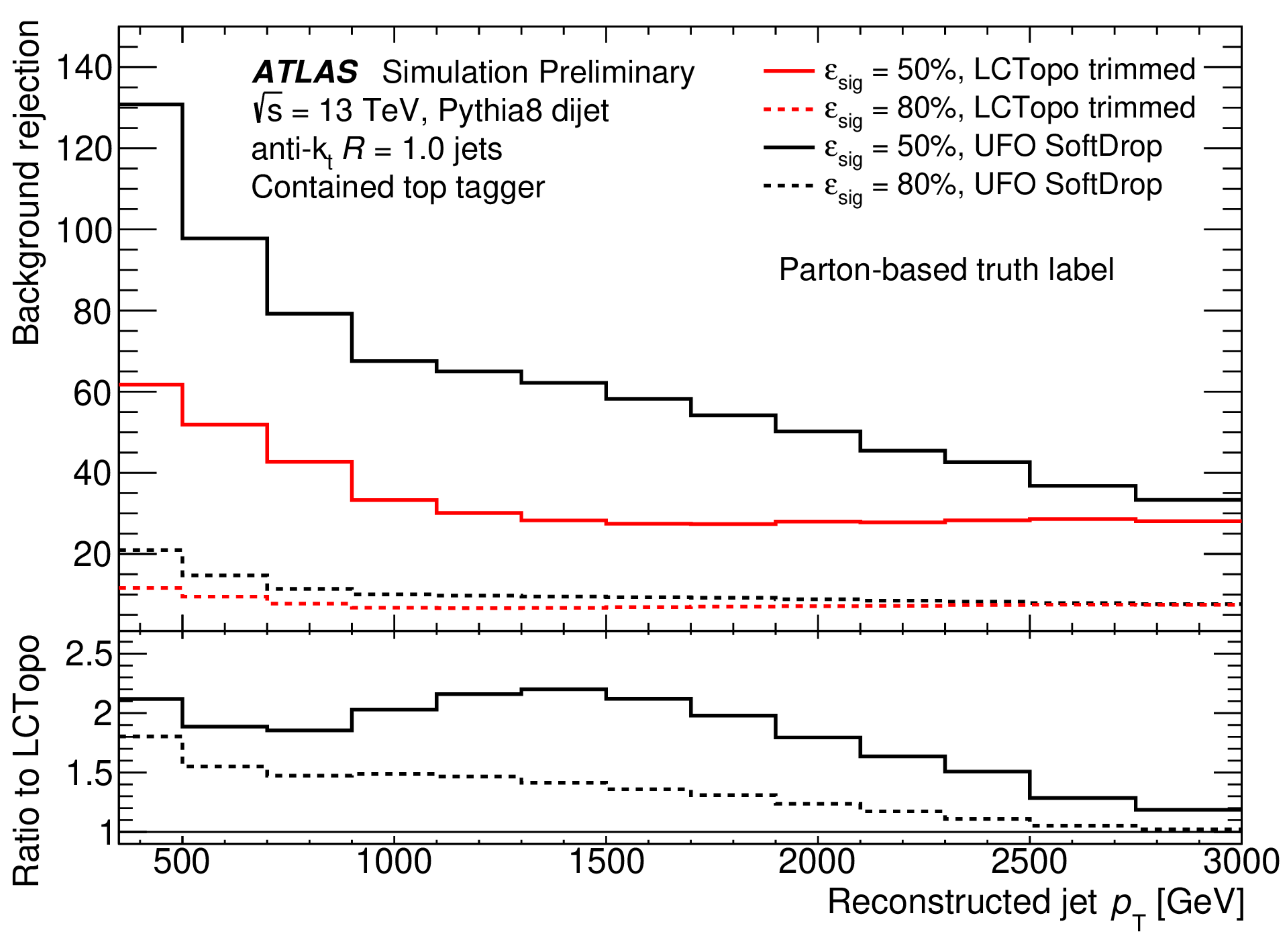
Boosted Jet Tagging

Top taggers: from high level features to low level features



- Initial ML algorithm (DNN) exploits 15 high-level jet substructure quantities
  - ↪Applied to contained and inclusive (a jet not containing the whole decay) tops
- Two working points under study (50% and 80%)
- Comparison with previous jet definition (LCTopo)
  - ↪Inclusive top tagger: background rejection is improved by almost a factor 2 for 50% WP
  - ↪Contained top tagger: UFO SD tagger outperfoms the LCTopo tagger for both working points





ATL-PHYS-PUB-2021-028

Boosted Jet Tagging

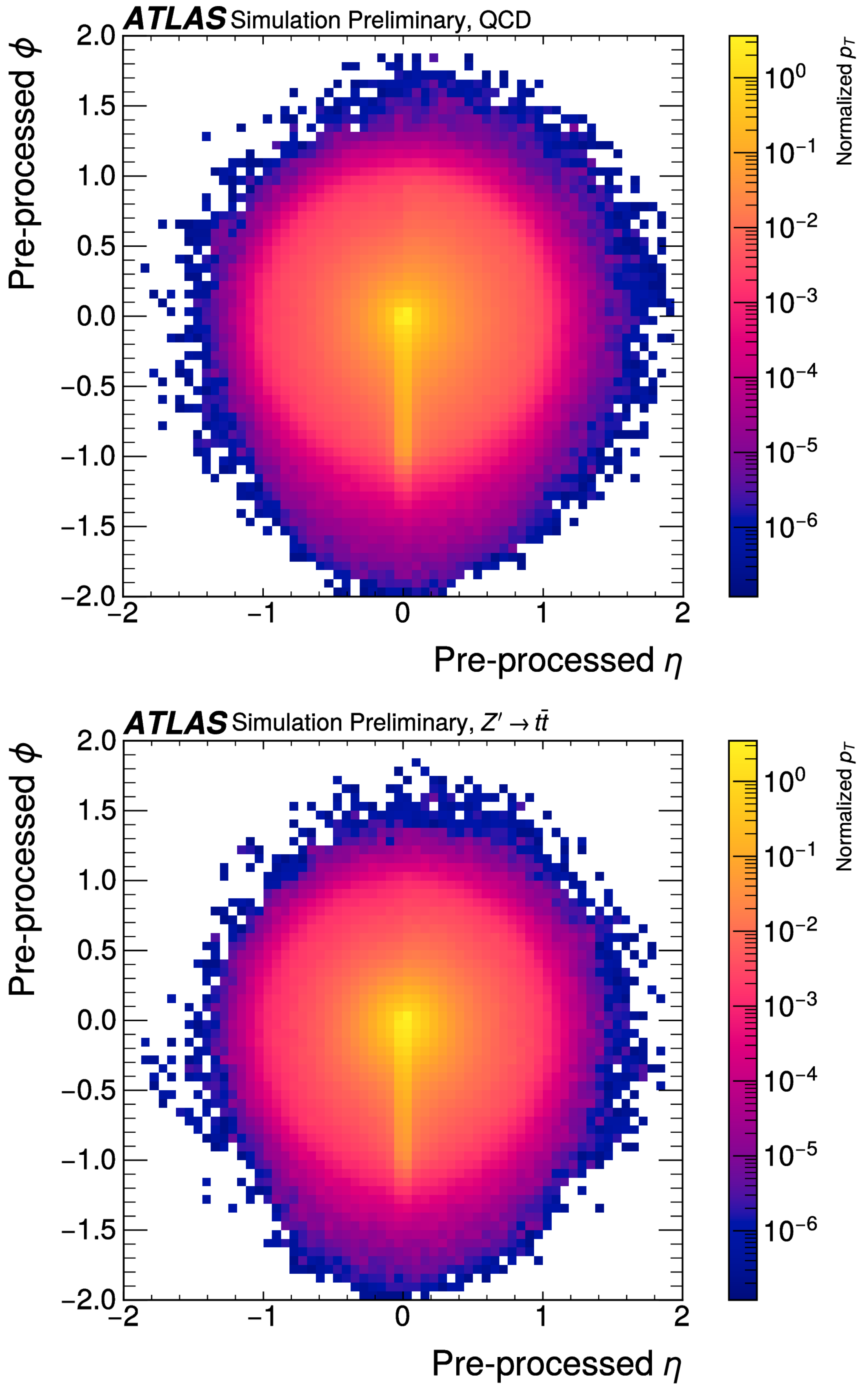
Top taggers: from high level features to low level features

- Moving to low level features: 4-vector jet constituents as inputs
  - ↪Features are pre-processed to exploit known symmetries
  - ↪Contained boosted top only considered
- 5 new ML architectures to be compared with baseline high-level DNN

The architectures

- Baseline DNN trained on high-level input features (**hlDNN**)
- **DNN** trained on constituent inputs
- Energy Flow Network (**EFN**) using DeepSets structure
  - ↪Ensures permutation invariance with respect to network inputs
  - ↪Can manage variable length, but only quantities linear in p_T
- Particle Flow Network **(PFN)** using DeepSets structure
  - ↪Similar to EFN, but all constituent inputs can be used
- **ResNet50**, a convolutional neural network, jets treated as images
- **ParticleNet**, a Graph NN, jet represented as graphs

Jets as images



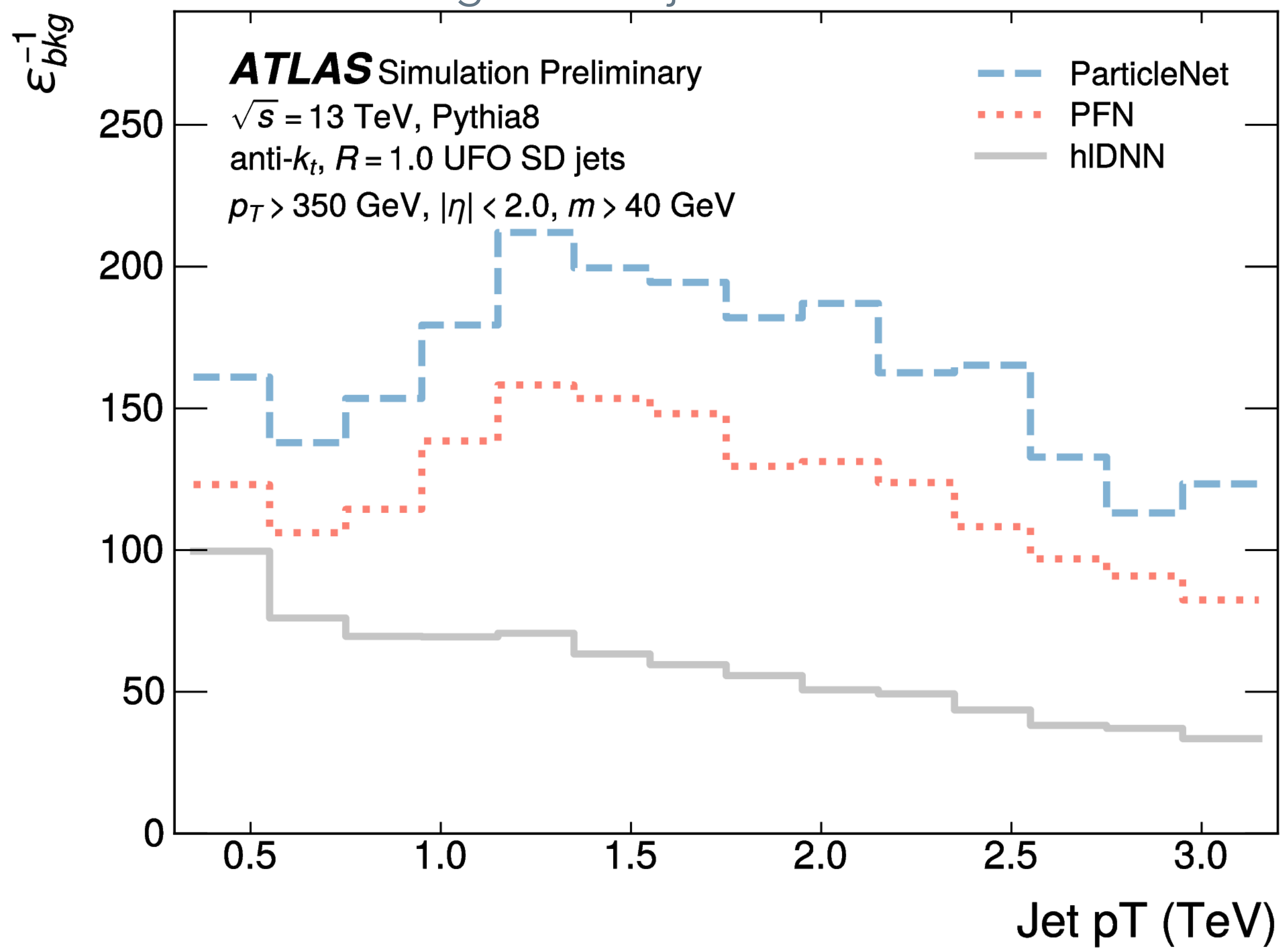
ATL-PHYS-PUB-2022-039

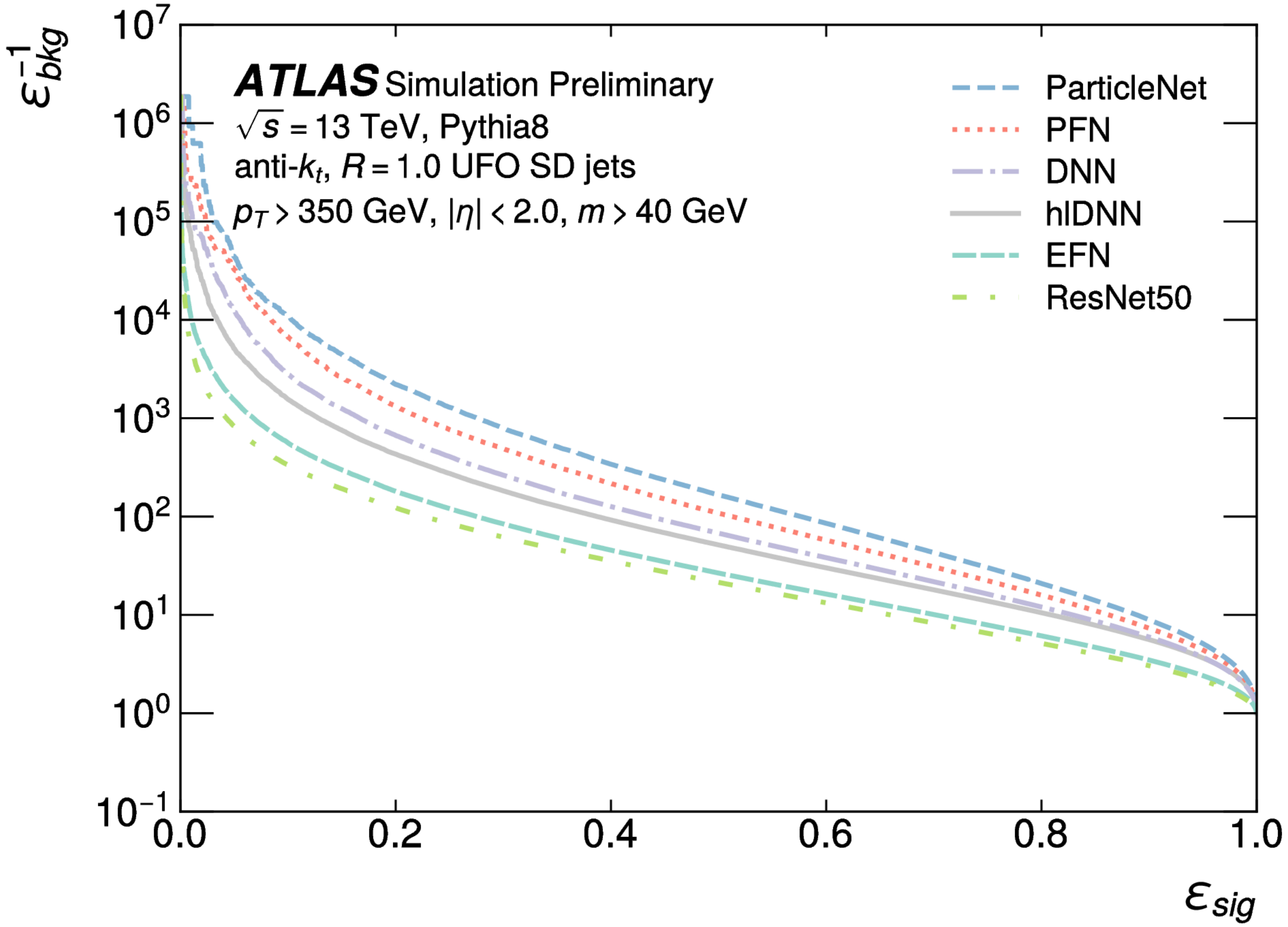
Boosted Jet Tagging

Top taggers: from high level features to low level features

- Three new architectures show better performances than baseline hlDNN: DNN, PFN and ParticleNet
- In all metrics ParticleNet achieved the best performance
- Constituent based taggers take advantage of addition information contained in jet constituents

Background rejection at 50% WP





Model	AUC	ACC	ε_{bkg}^{-1} @ $\varepsilon_{sig} = 0.5$	ε_{bkg}^{-1} @ $\varepsilon_{sig} = 0.8$	# Params	Inference Time
ResNet 50	0.885	0.803	21.4	5.13	1,486,209	9 ms
EFN	0.901	0.819	26.6	6.12	1,670,451	4 ms
hlDNN	0.938	0.863	51.5	10.5	93,151	3 ms
DNN	0.942	0.868	67.7	12.0	876,641	3 ms
PFN	0.954	0.882	108.0	15.9	689,801	4 ms
ParticleNet	0.961	0.894	153.7	20.4	764,887	38 ms

ATL-PHYS-PUB-2022-039

Boosted Jet Tagging

Top taggers: from high level features to low level features

- Model dependence has been evaluated
  - ↪PFN and ParticleNet are more dependent on QCD modeling than baseline hlDNN model

ATL-PHYS-PUB-2022-039

MISSING ENERGY IN ATLAS

Finding working point using NN

- Missing transverse momentum p_T^{miss} represents the total transverse momentum of undetected particles produced
  ↪Could indicate the production of invisible particles such as Standard Model neutrinos or BSM particles that escape ATLAS undetected
  ↪Detector resolution/acceptance, wrong particle assignement can lead to a different p_T^{miss} reconstructed with respect to its true value
- Traditionally p_T^{miss} is calculated by negative sum of objects:
  ↪existing working points to meet analysis requirements but not flexible

$$\mathbf{E}_\mathrm{T}^\mathrm{miss} = \underbrace{\underbrace{-\sum_{\substack{\text{selected}\\\text{electrons}}} \mathbf{p}_\mathrm{T}^{e}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},e}} \underbrace{-\sum_{\substack{\text{accepted}\\\text{photons}}} \mathbf{p}_\mathrm{T}^{\gamma}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},\gamma}} \underbrace{-\sum_{\substack{\text{accepted}\\\tau\text{-leptons}}} \mathbf{p}_\mathrm{T}^{\tau_\mathrm{had}}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},\tau_\mathrm{had}}} \underbrace{-\sum_{\substack{\text{selected}\\\text{muons}}} \mathbf{p}_\mathrm{T}^{\mu}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},\mu}} \underbrace{-\sum_{\substack{\text{accepted}\\\text{jets}}} \mathbf{p}_\mathrm{T}^{\mathrm{jet}}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},\mathrm{jet}}}}_{\text{hard term}} \underbrace{\underbrace{-\sum_{\substack{\text{unused}\\\text{tracks}}} \mathbf{p}_\mathrm{T}^{\mathrm{track}}}_{\mathbf{E}_\mathrm{T}^{\mathrm{miss},\mathrm{soft}}}}_{\text{soft term}}$$

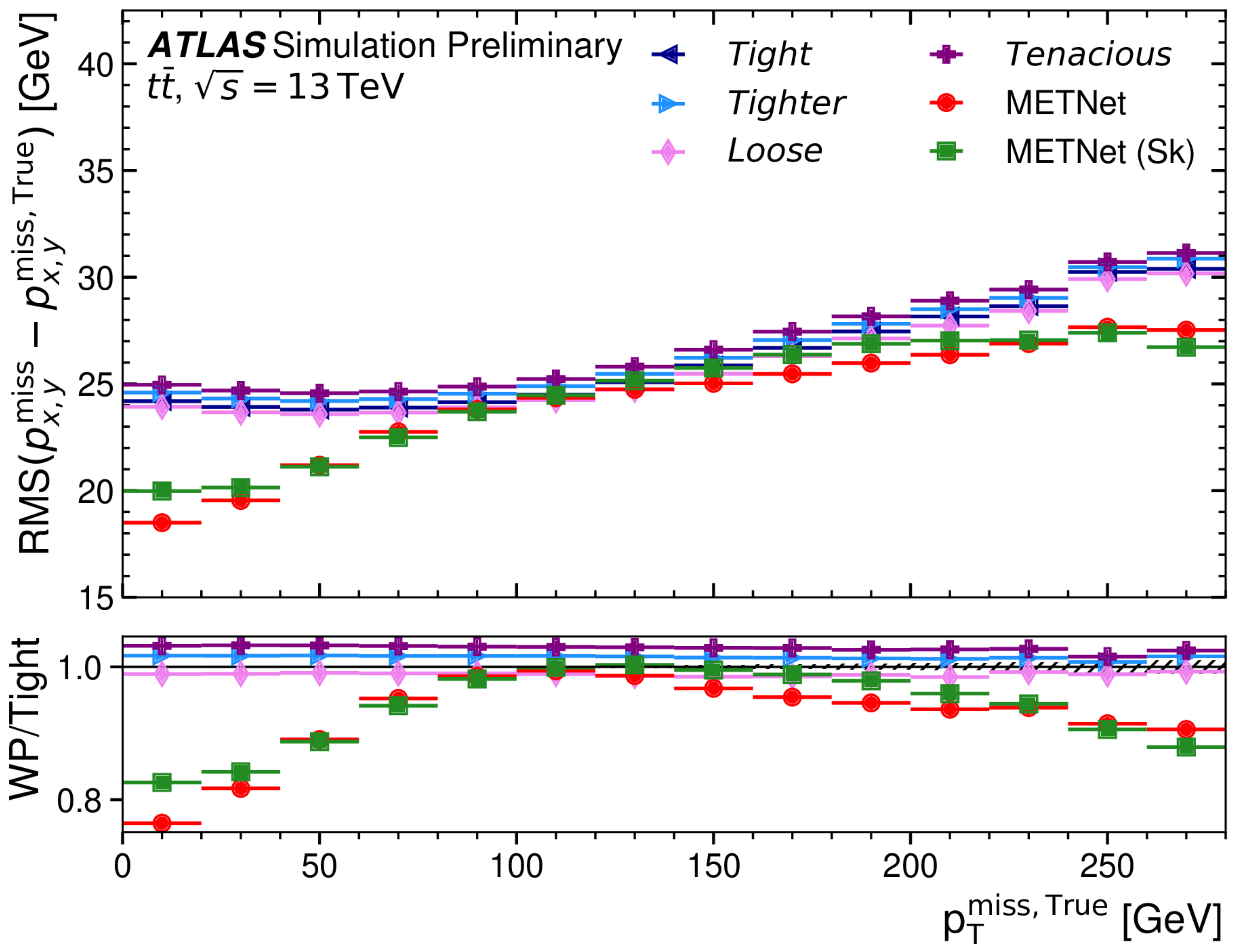
- Use of regression to combine complementary information from each working point on an event-by-event basis to produce p_T^{miss} prediction with improved resolution
- Train NN to predict $p_x^{miss,true}$ and $p_y^{miss,true}$ → METNet
  ↪The NN receives 60 event variables
  ↪Use an optimized loss function (Sinkhorn) to prevent negative bias (METNetSk)
  ↪Develop METNetSig, a "variant" to separates real and fake p_T^{miss}: defined as p_T^{miss} over its resolution

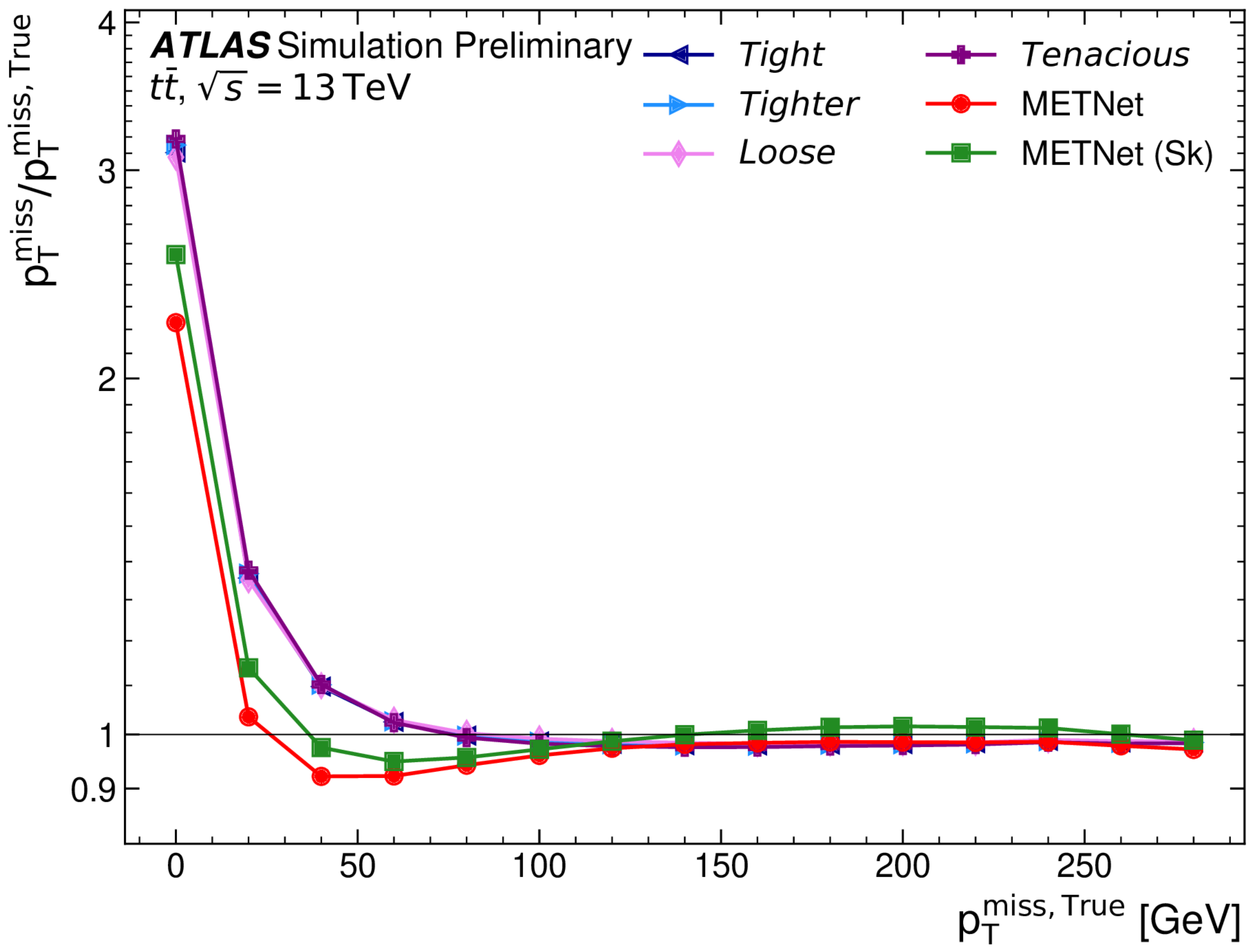
ATL-PHYS-PUB-2021-025

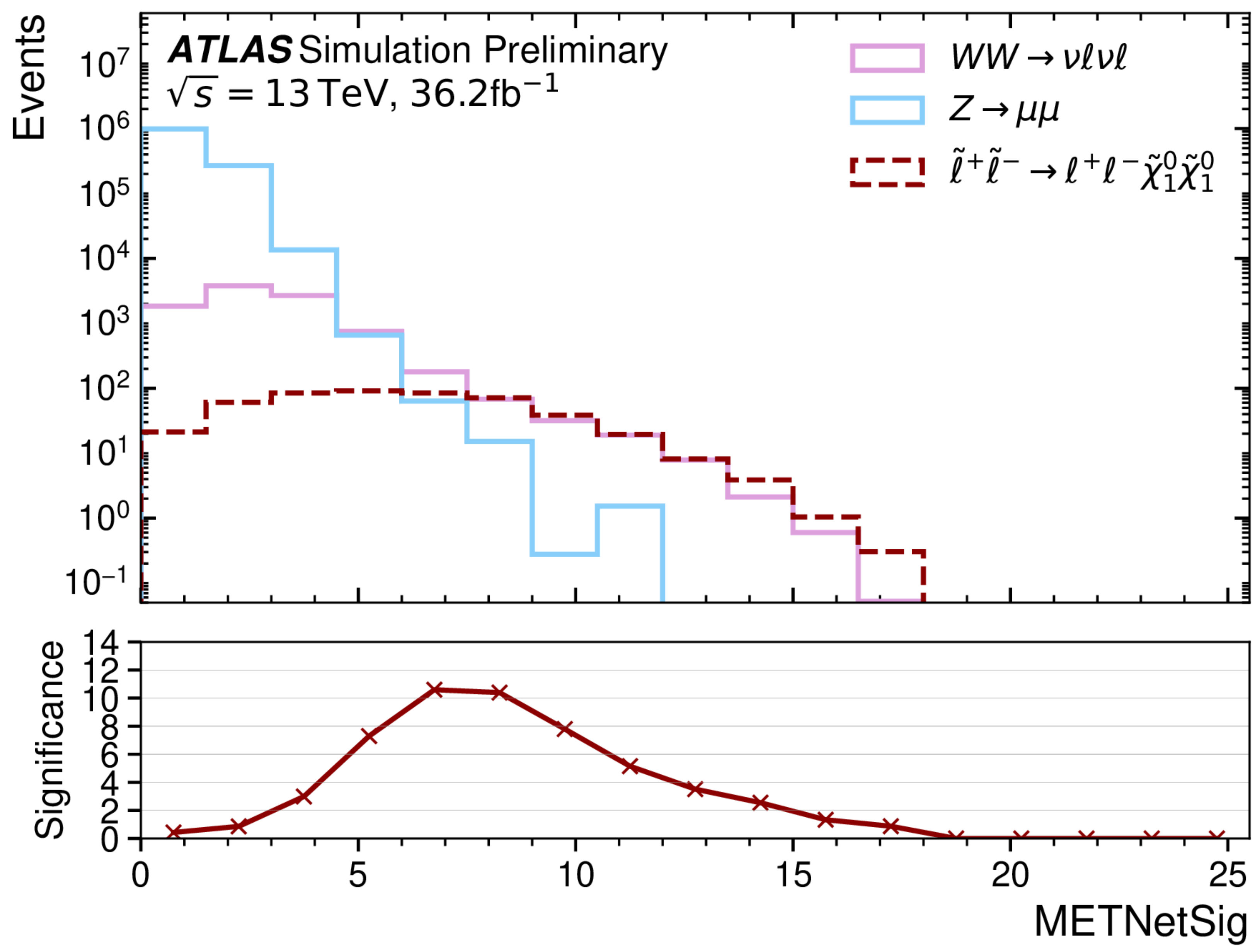
Missing Energy in ATLAS

Finding working point using NN

- p_T^{miss} definition depends on process but METNet performs best for all
- METNet has a significantly improved resolution
- Large potential to significantly improve missing energy reconstruction using machine learning techniques.





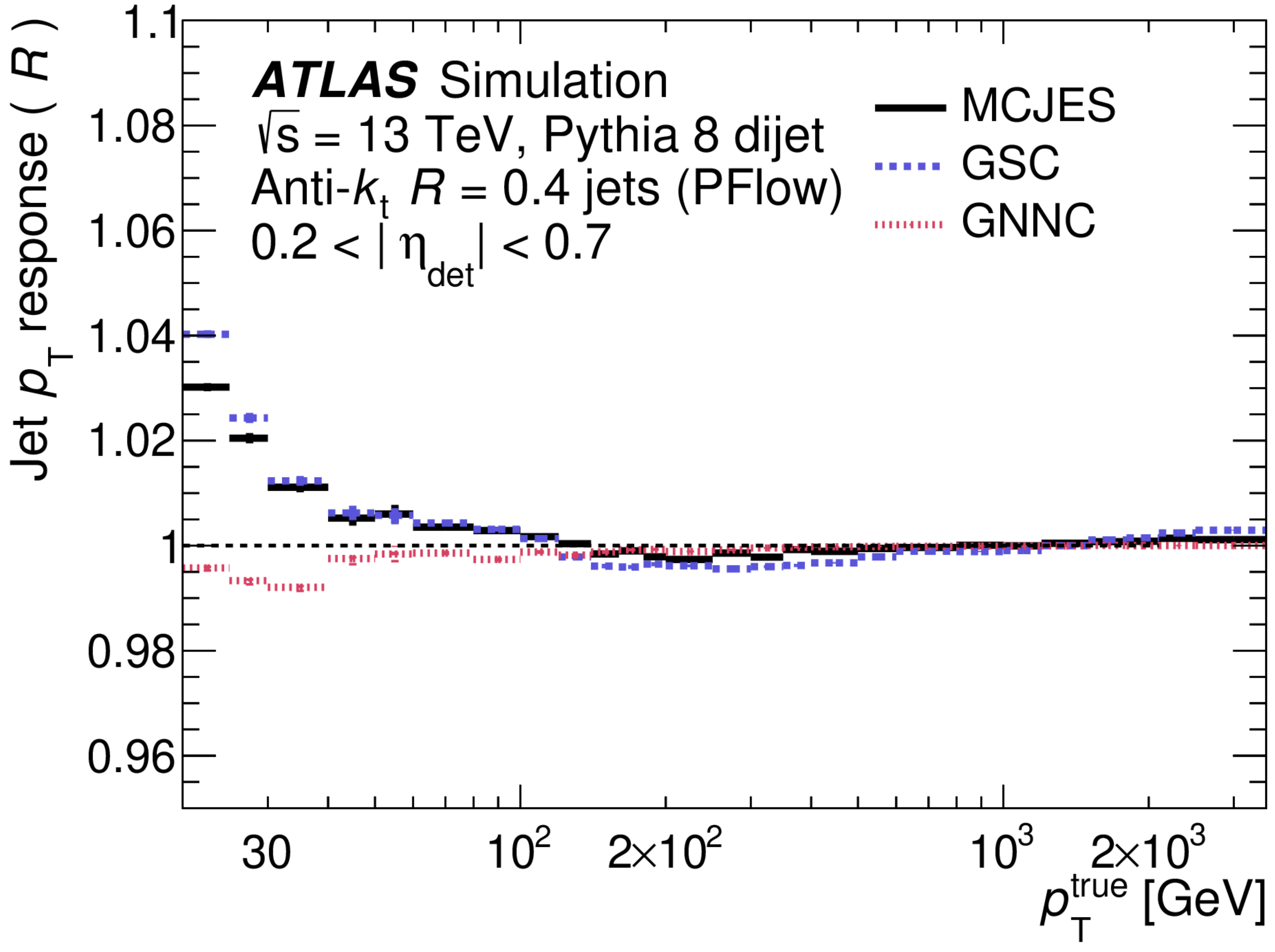


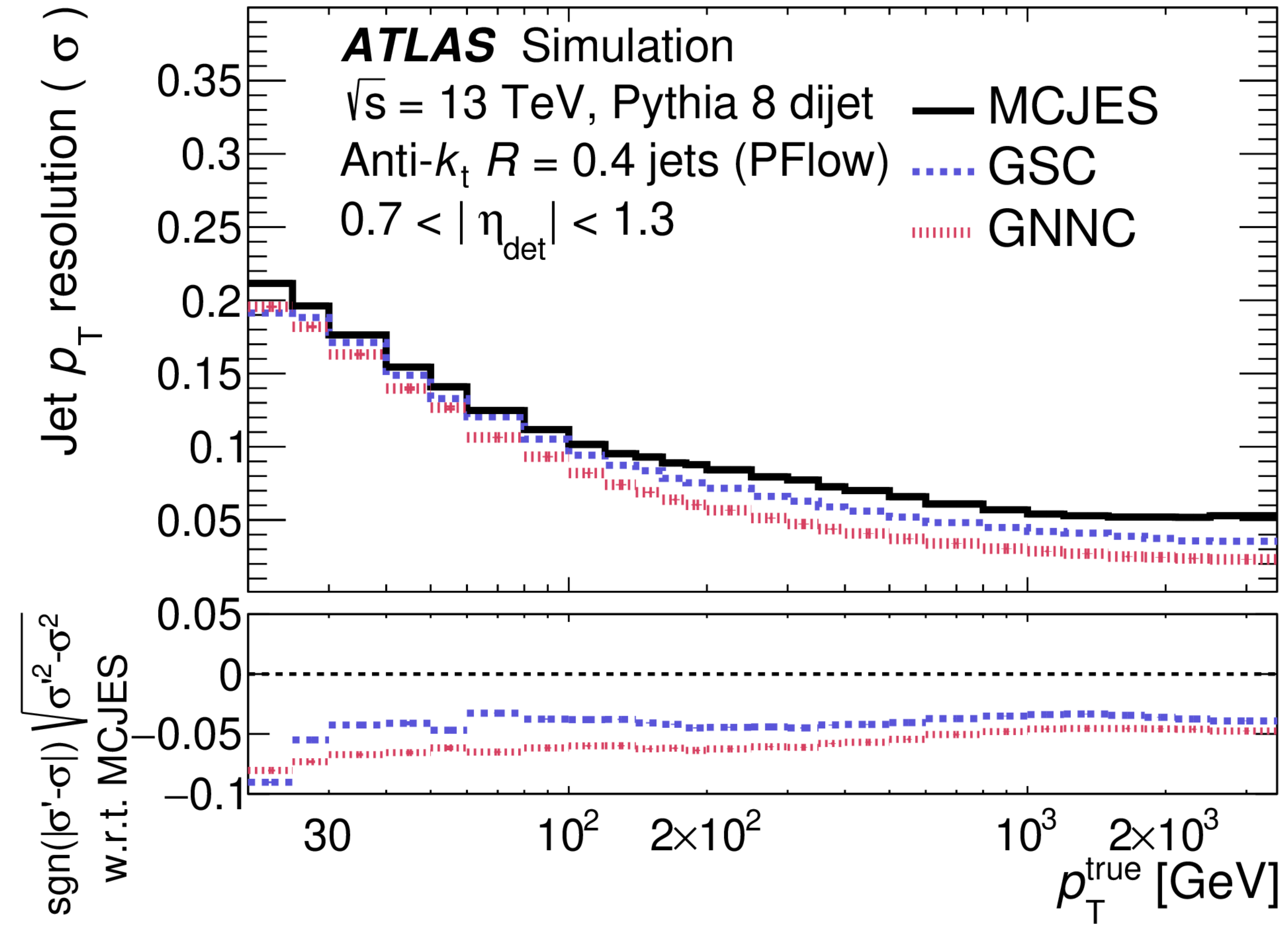
- METNetSig shows similar behaviour per-topology to object-based $p$miss significance
- Current implementation performs slightly worse than the object-based variable
- It remains a promising approach as its performance can be expected to improve further with more optimisation and training statistics.

ATL-PHYS-PUB-2021-025

JET CALIBRATION: GLOBAL CORRECTION

- Jet response could be jet-type dependent. Need a correction to account for impact of different types of jets on the jet response
  - ↪Typically using E and η (MCJES)
- The global jet property calibration applies further corrections to jets based on their individual characteristics
  - ↪Global sequential calibration (GSC): using 6 observables to improve jet energy response
  - ↪Global neural network calibration (GNNC): DNN trained with more variables
  - ↪Corrections derived in η bins





GNNC has better jet pT closure and has over 15% improvements in jet pT resolution

arXiv:2303.17312

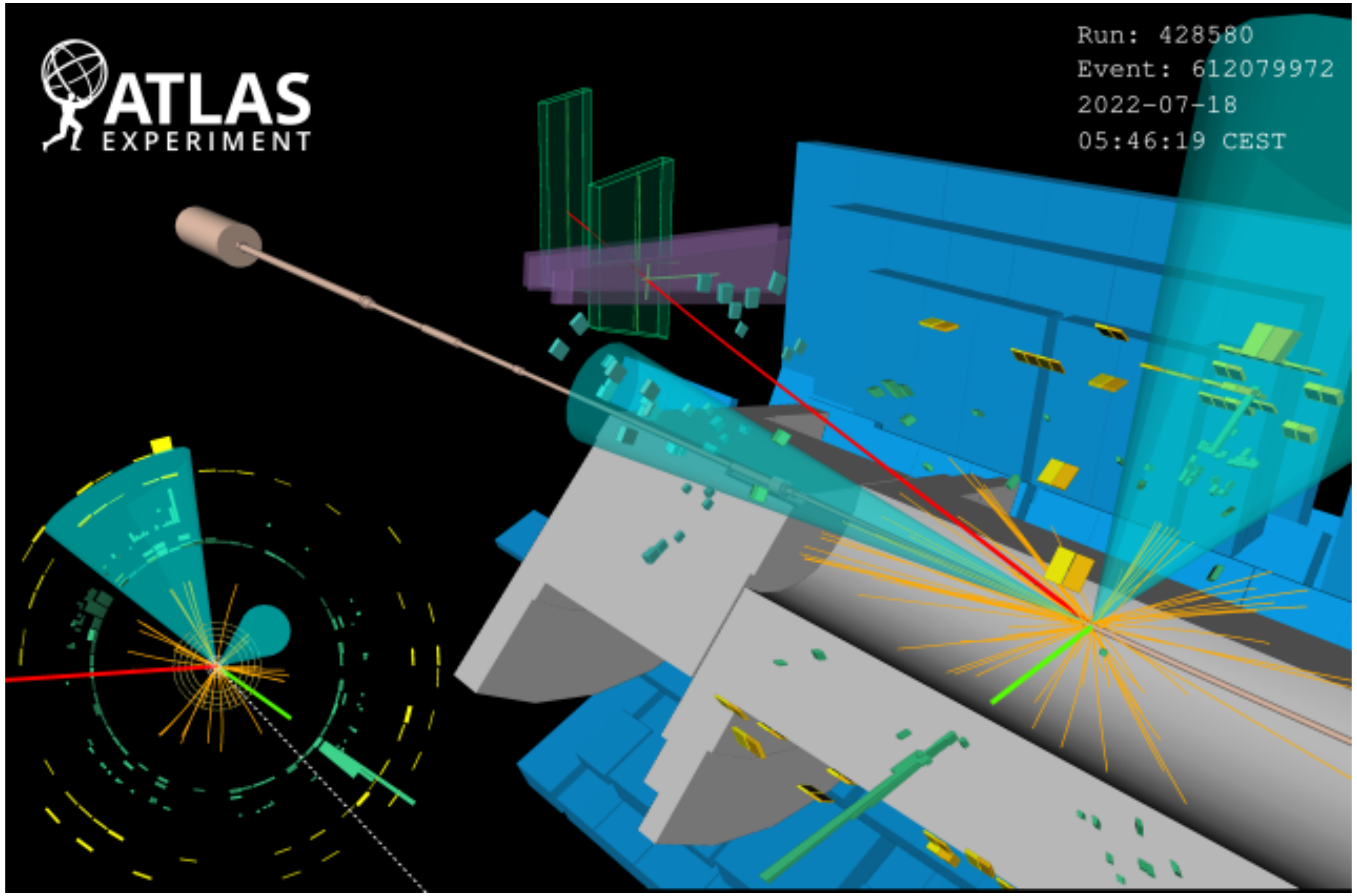
Conclusions

Jet understanding is fundamental

- Hadronic object reconstruction, calibration and tagging fundamental for precision measurements and BSM searches

- During Run-2 and Long Shutdown many efforts to exploits Machine Learning potential for improving current state of the art
  - ↪Pion reconstruction, W/Z/top taggers, calibration, METNet
  - ↪Many architectures have been tested: from “simple” DNN to GNN

- Run-3 started, a great opportunity to test this improvements on new data and continue developing these tools

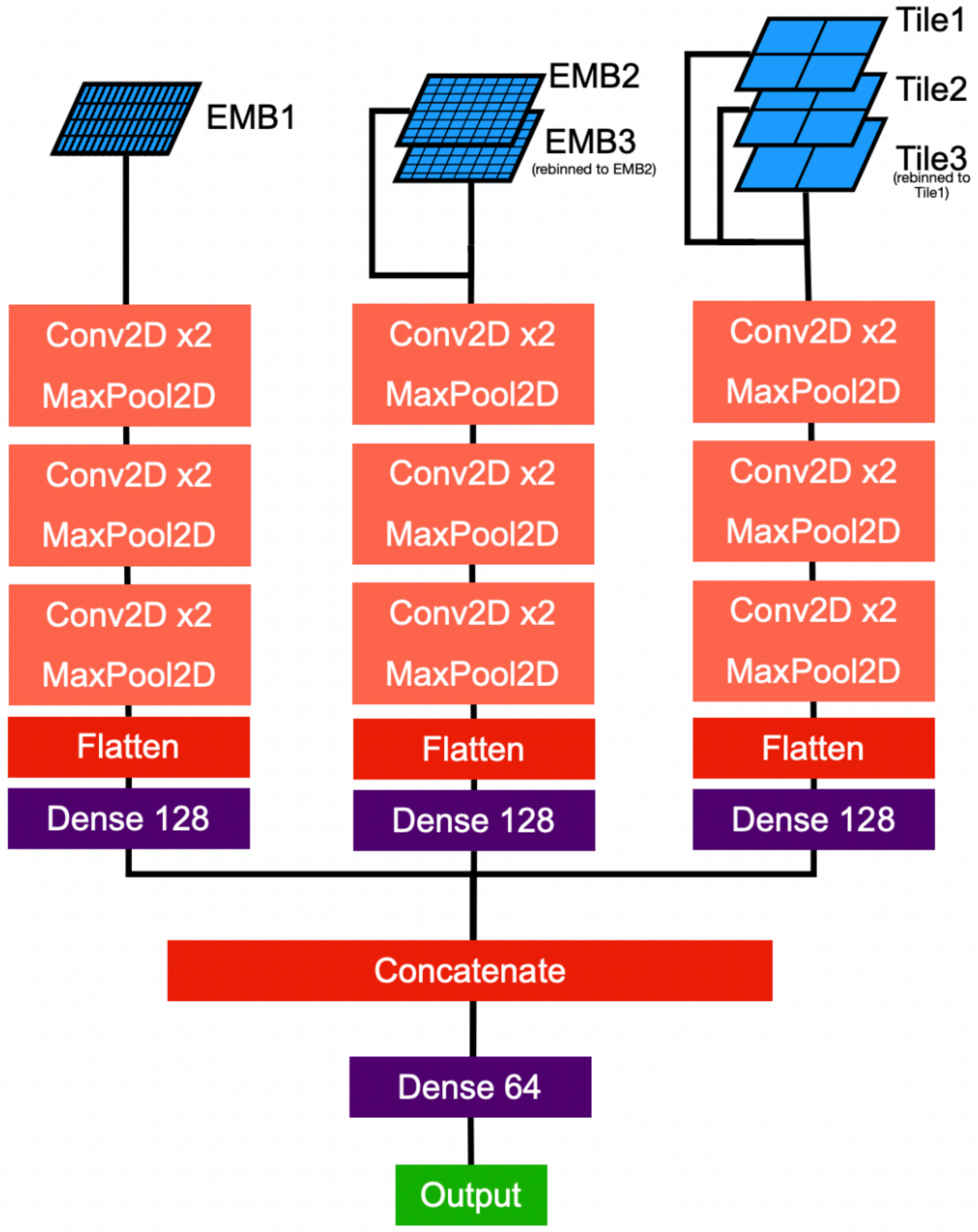


ATLAS-PHOTO-2022-061

Backup

CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

ML architectures - CNN



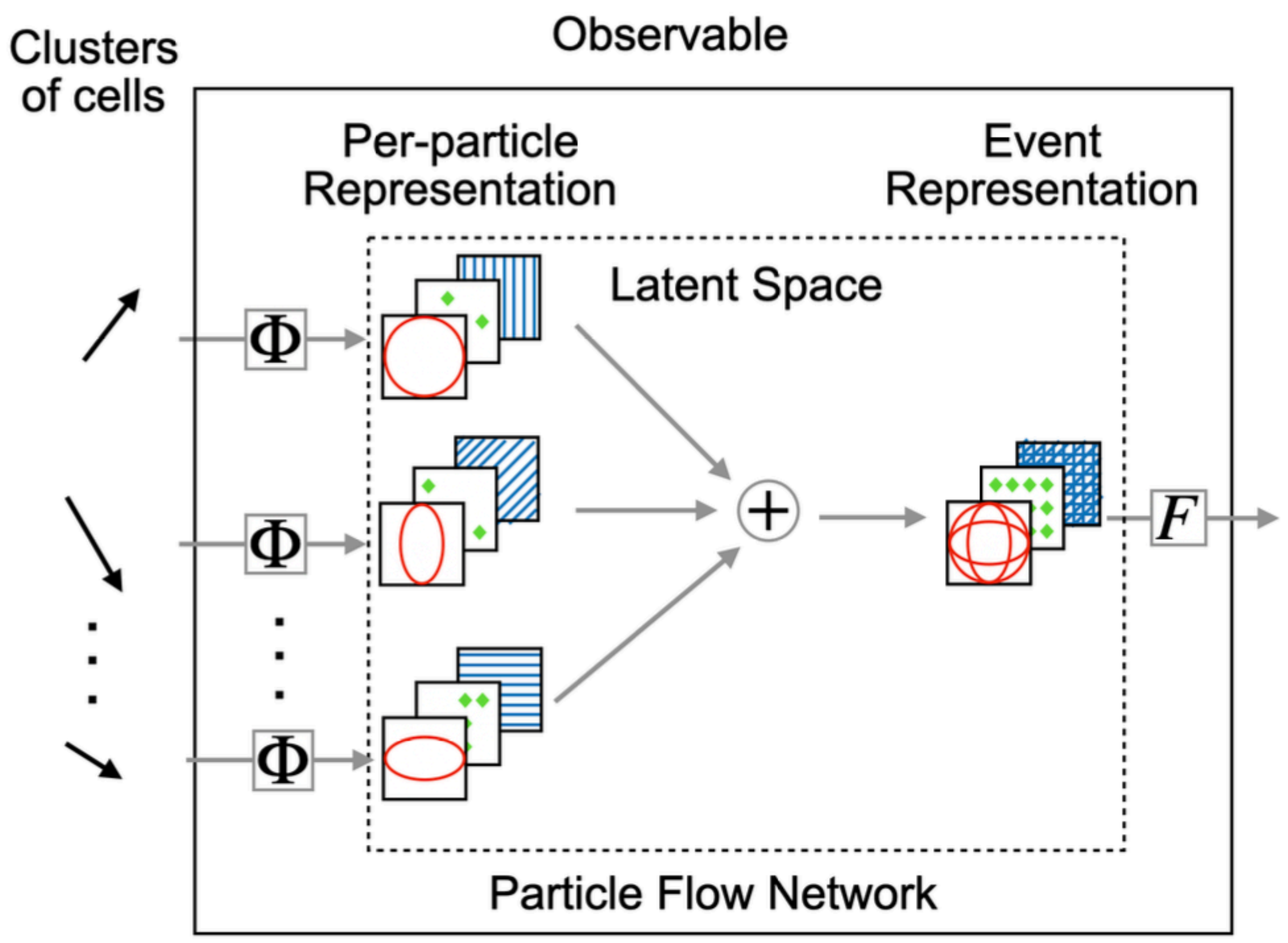
Calorimeter Layer	$(\Delta\eta, \Delta\phi)$ Granularity
EMB1	128 × 4
EMB2	16 × 16
EMB3	8 × 16
Tile1	4 × 4
Tile2	4 × 4
Tile3	2 × 4

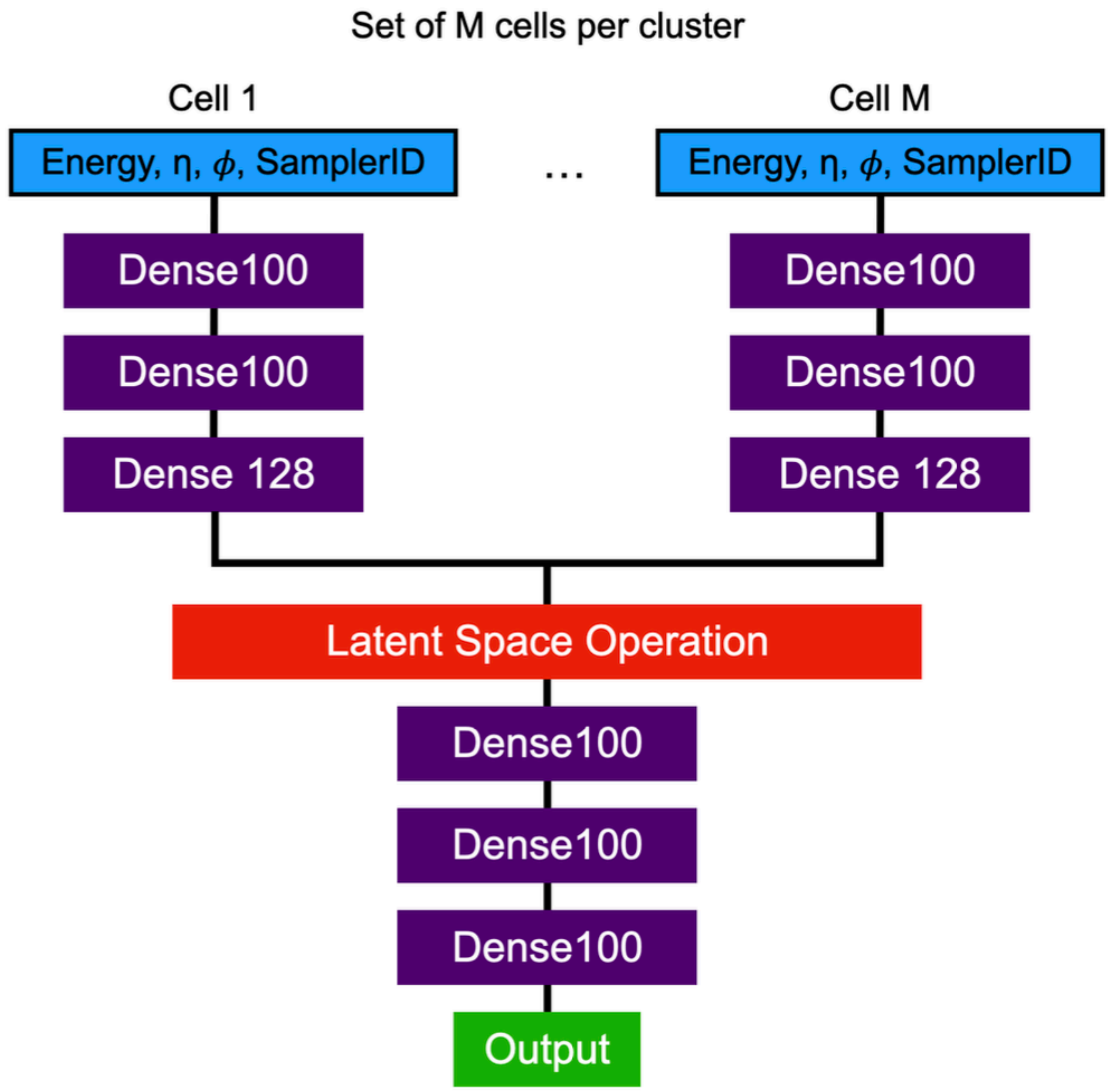
- Energy deposits in calorimeters treated as pixel densities (η, ϕ)
- 3 layers of the EM calorimeter + 3 layers of hadronic calorimeter

CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

ML architectures - DeepSets

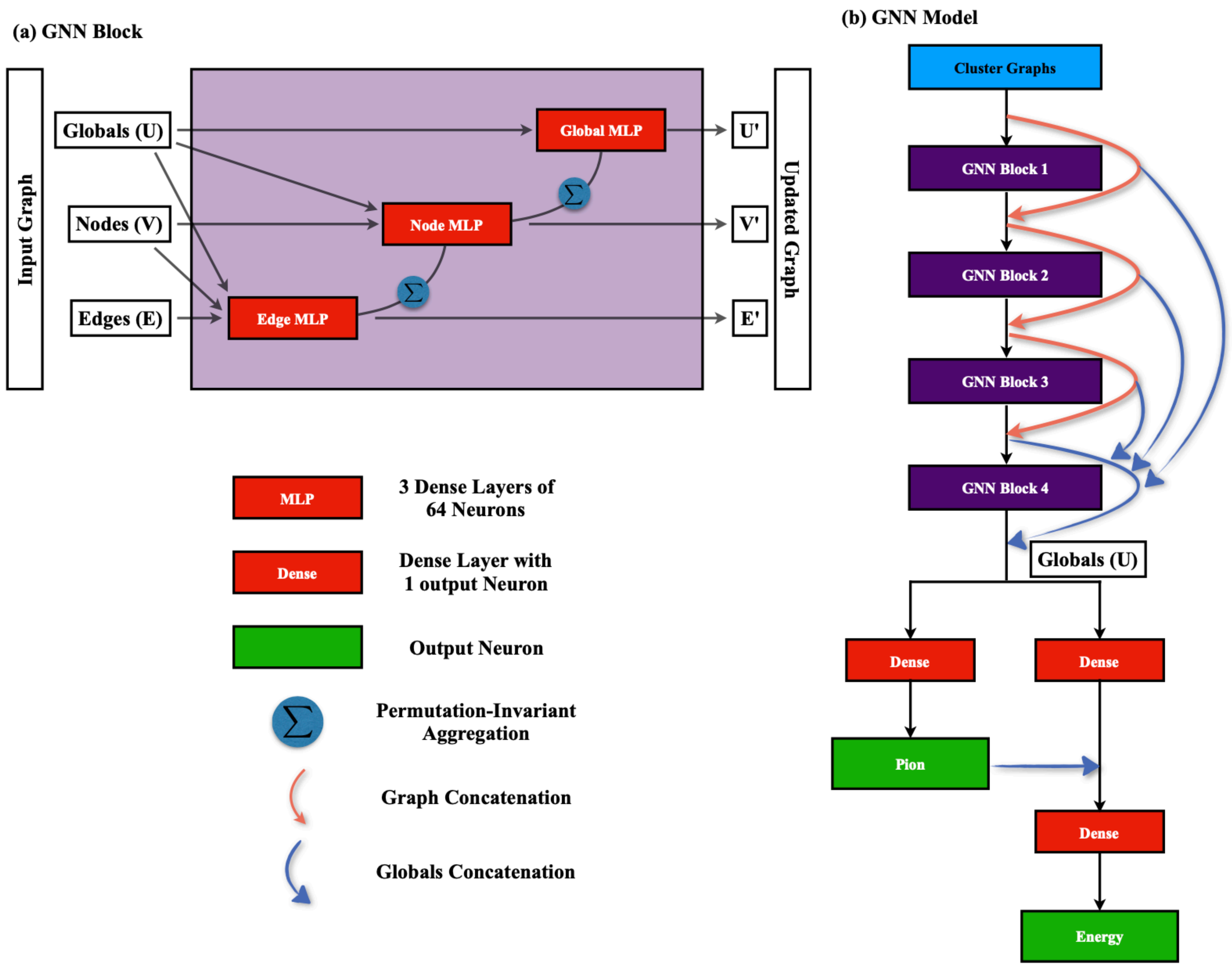
- any physics observable that is symmetric with respect to the ordering of the considered particles, can be approximated arbitrarily well with a parameterization of permutation-invariant functions of variable-length inputs
- observables are viewed as functions of sets of clusters composed of calorimeter cells
-





CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

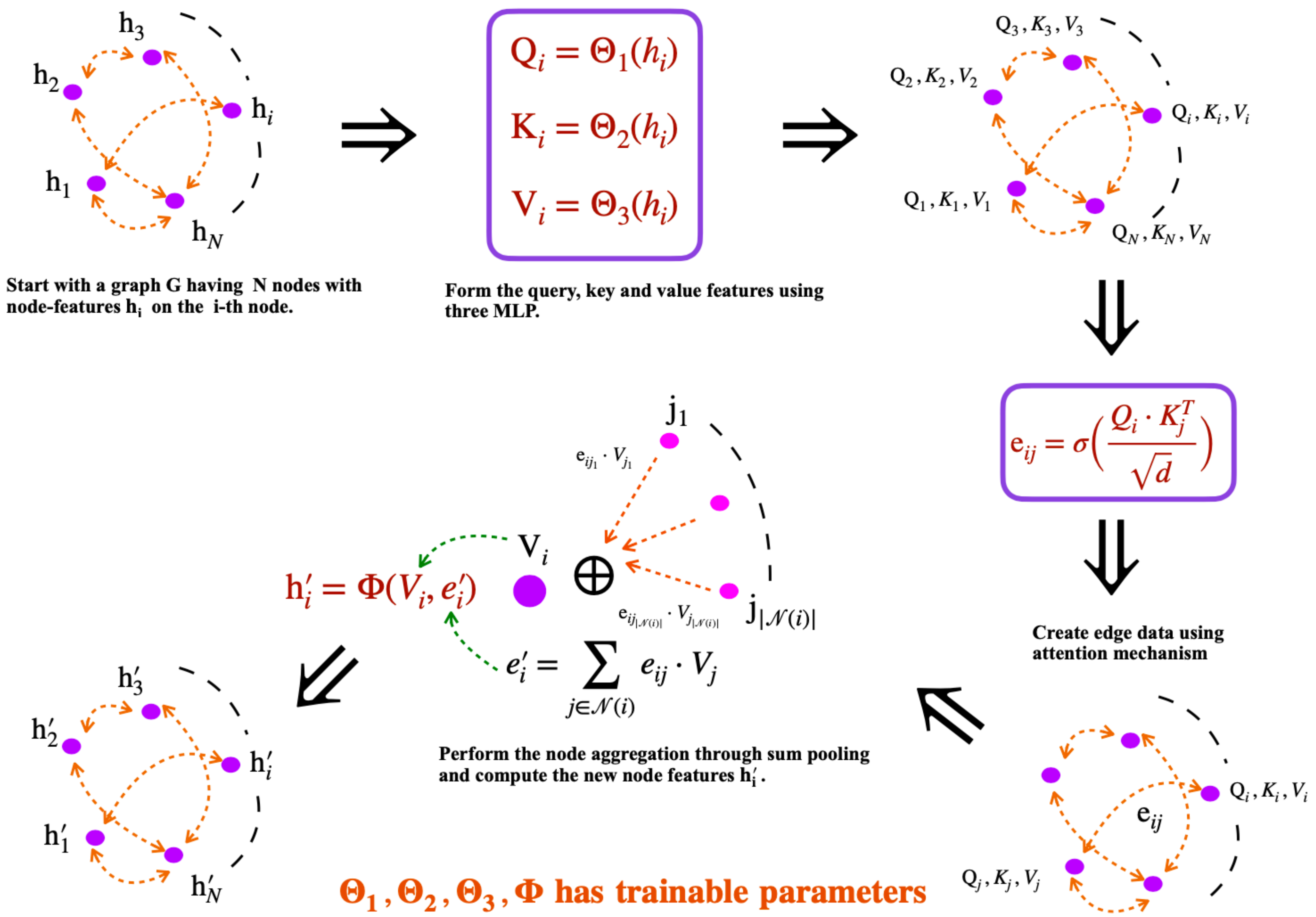
ML architectures - GNN



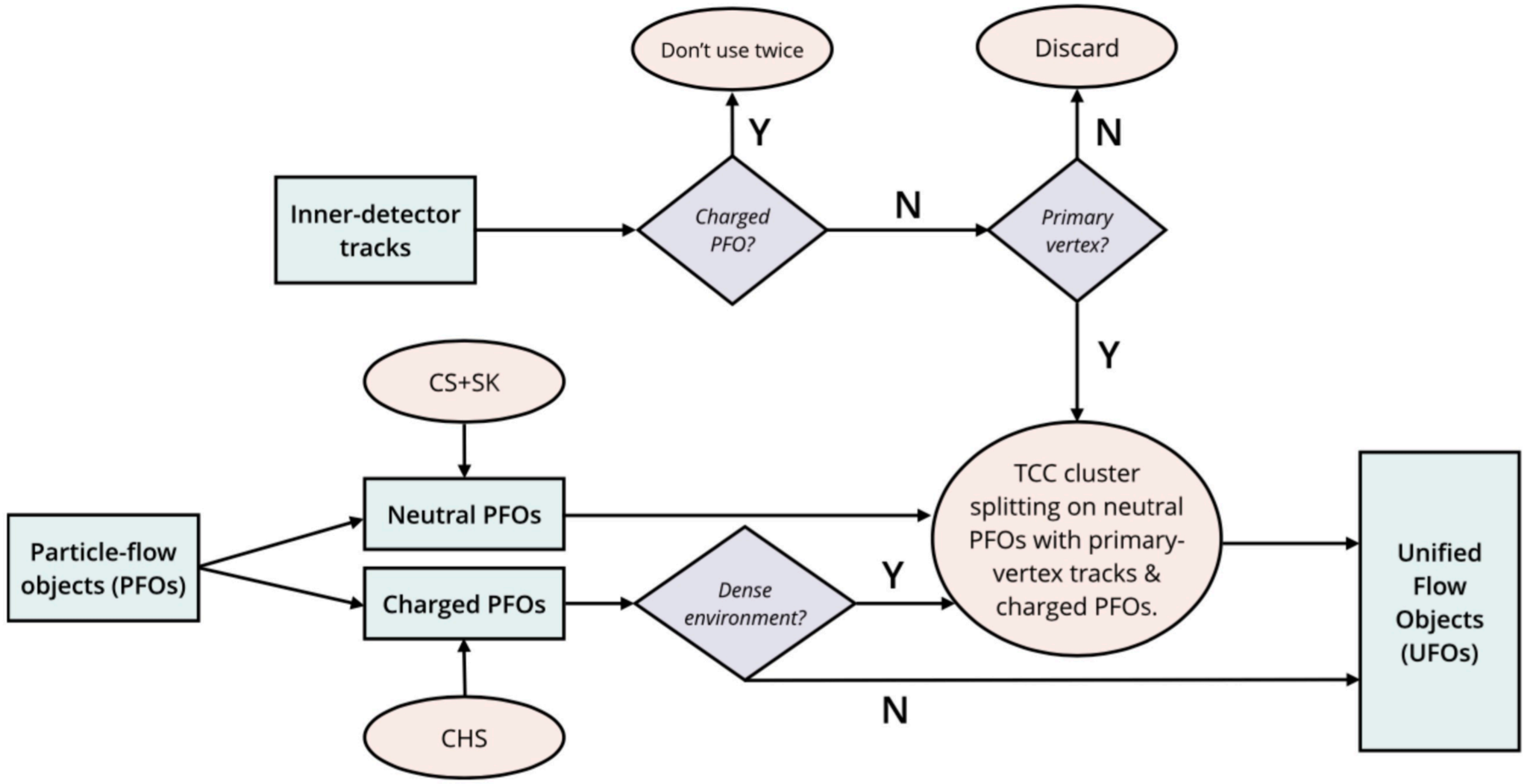
- takes a graph-structured input $G = (V, E)$ and learns a hidden representation of the graph that is repeatedly updated via message passing
- classification task of pion identification ($\pi 0$ versus $\pi\pm$) and regression task of energy calibration.
- Each pion topo-cluster is represented as a graph with nodes, edges, and a global node
  - ↪Features: cell energy, sampling layer, $\eta, \phi, \Delta\eta, \Delta\phi$, shortest radial distance of the cell to the shower axis
  - ↪Edge: neighboring cells are connected to one another
- four GNN blocks that use multi-layer perceptrons (MLP), each MLP in the GNN block consists of three dense layers of size 64 each
-

CALORIMETER SIGNALS RECONSTRUCTION/CLASSIFICATION

ML architectures - Transformer



UFO ALGORITHM



- applying standard ATLAS PFlow algorithm
- Charged PFOs which are matched to pile-up vertices are removed.
- remaining PFOs are classified into different categories: neutral, charged PFOs which were used to subtract energy from a topocluster, and charged PFOs for which no subtraction was performed
- Jet-input-level pile-up mitigation algorithms may now be applied to the neutral PFOs if desired.
- A modified version of the TCC splitting algorithm applied to the remaining PFOs: only tracks from the hard-scatter vertex are used as input to the splitting algorithm, in order to avoid pile-up instabilities.
- Any tracks which have been used for PFlow subtraction are not considered, as they have already been well-matched and their expected contributions have been subtracted from the energy in the calorimeter.
- The TCC algorithm then proceeds using the modified collection of tracks to split neutral and unsubtracted charged PFOs instead of topoclusters. This approach provides the maximum benefit of PFlow subtraction at lower particle pT, and cluster splitting where the benefit is maximal at high particle pT

Eur. Phys. J. C 81 (2021) 4, 334

Boosted Jet Tagging

W/Z taggers

ATL-PHYS-PUB-2021-029

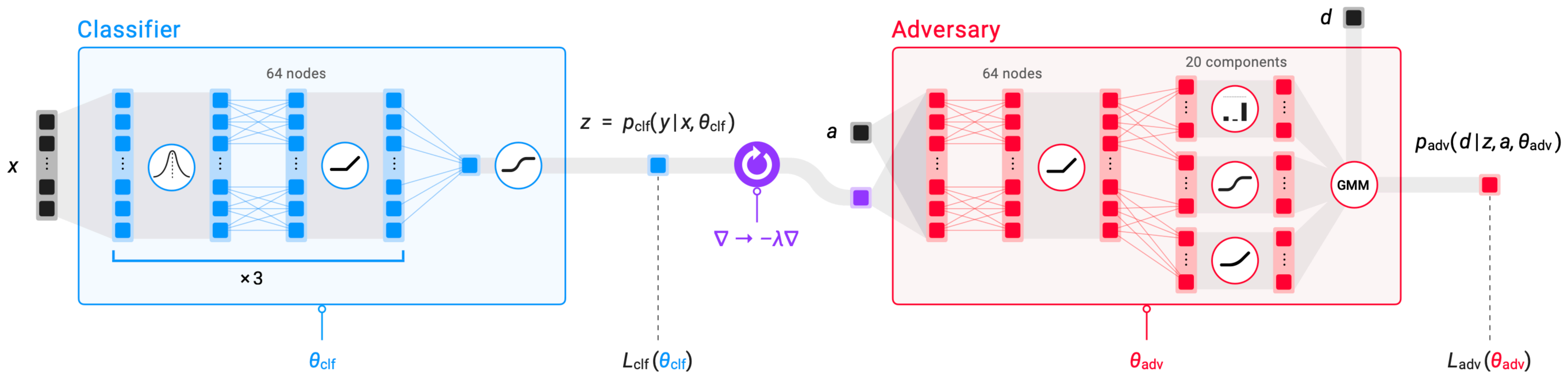
JETM-2022-06

- DNN: three fully-connected 32 node dense layers with a tanh activation function and a single-node output layer with sigmoid activation implemented in Keras and Tensorflow are used in this analysis

- ANN:
  - ↪Adversary trained to infer mJ from classifier score
  - ↪Penalise classifier if adversary predicts mass too well

Table 1: List of substructure variables used in the DNN tagger training.

Variable	Description	Reference
D_2, C_2	Energy correlation ratios	[30]
τ_{21}	N-subjettiness	[41]
R_2^{FW}	Fox-Wolfram moment	[42]
$\mathcal{P}$	Planar flow	[43]
a_3	Angularity	[44]
A	Aplanarity	[45]
Z_{cut}, $\sqrt{d_{12}}$	Splitting scales	[33, 46]
$Kt\Delta R$	k_t-subjet ΔR	[47]

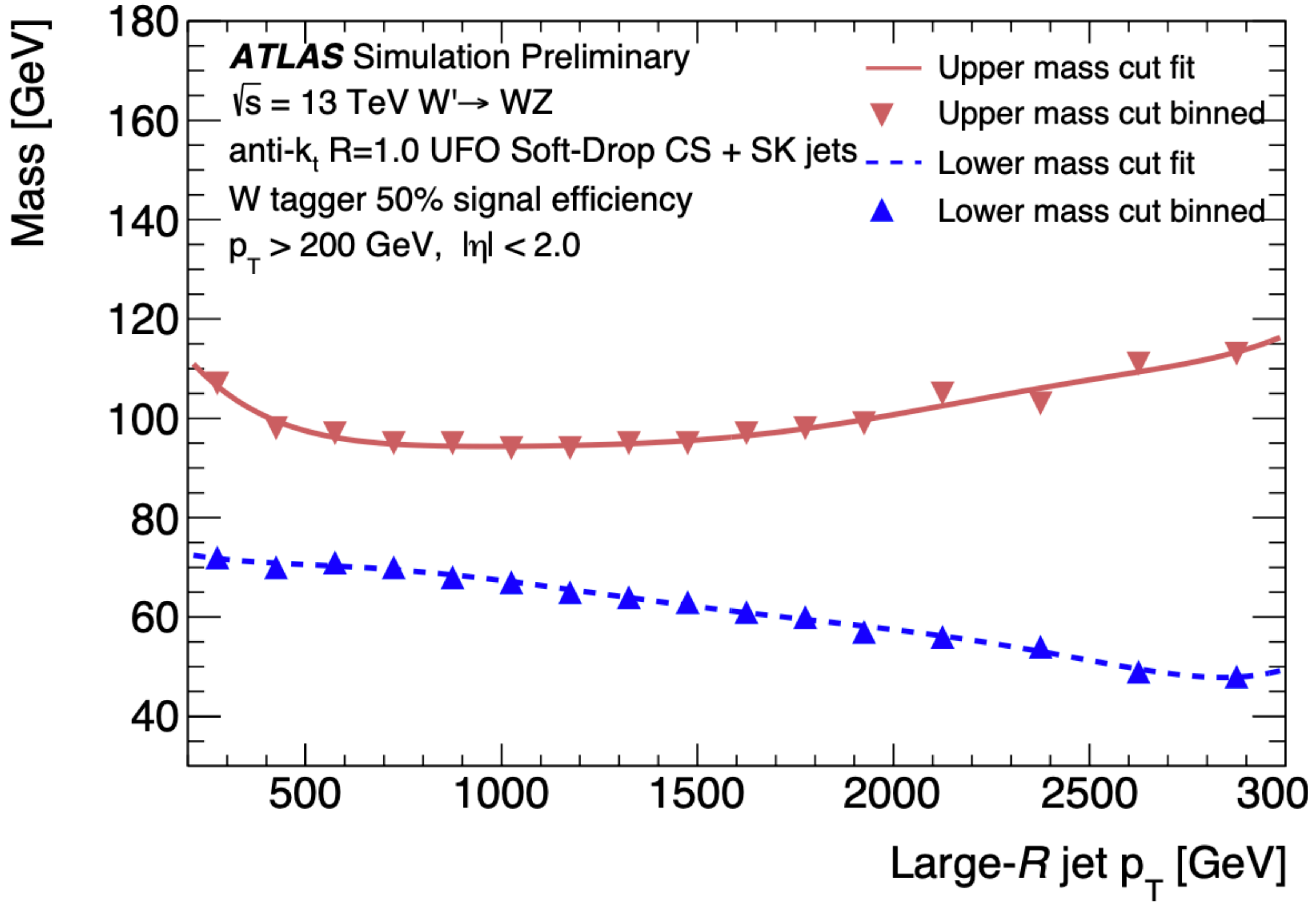


ATL-PHYS-PUB-2018-014

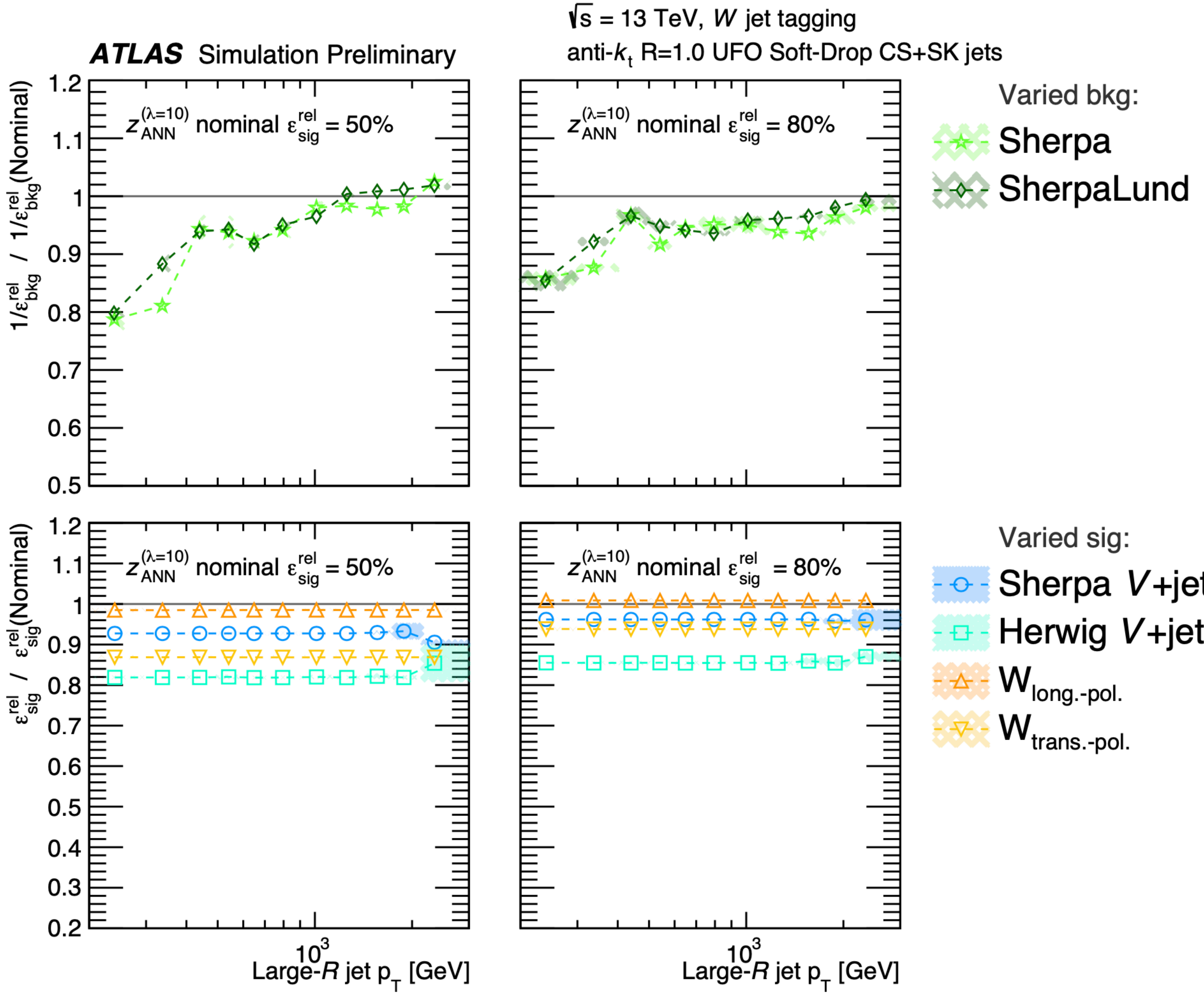
Boosted Jet Tagging

W/Z taggers

- **3-variable tagger Z**
  - ↪pT dependent cuts on 3 features
  - ↪Maximes bkg rejection for a 50% signal efficiency per bin



- **Model-dependence**
  - ↪Sensitive to modeling differences between MC generators



ATL-PHYS-PUB-2021-029

JETM-2022-06

Boosted Jet Tagging

Top taggers: from high level features to low level features

arxiv:1902.08570

- **ParticleNet**: graph neural network (GNN) which represents jets as a graph, composed of nodes and edges. Each constituent in a jet is associated with a node, where are all of the input quantities are taken as features of the node.
  - ↪Each node is connected by an edge to its k nearest neighbors in the η-ϕ plane, where k is a network hyper-parameter. ParticleNet applies a specialized form of the EdgeConv
- **ResNet50**: CNN designed for image classification task. Jet as 64x64 pixel image
- **PFN/EFN:** deep sets models
- **hl-DNN:** 15 high level quantities, standard MLP

Quantity Type	Symbols
N-subjettiness	$\tau_1, \tau_2, \tau_3, \tau_4$
k_t Splitting Scales	$\sqrt{d_{12}}, \sqrt{d_{23}}$
Generalized Energy Correlation Functions	$ECF_1, ECF_2, ECF_3, C_2, D_2, L_2, L_3$
Minimum Pair-wise Invariant Mass	Q_w
Thrust Major	T_m

ATL-PHYS-PUB-2022-039

Boosted Jet Tagging

Top taggers: from high level features to low level features

Model	Hyper-parameters
hlDNN	Hidden Layers: 5 Nodes per Layer: 180 *Activation Functions: ReLU* *Kernel Initialization: glorot uniform* Learning Rate: 4×10^{-5} Batch Size: 250 Batch Normalization: not used
DNN	Hidden Layers: 5 Nodes per Layer: 400 *Activation Functions: ReLU* *Kernel Initialization: glorot uniform* L1 Regularization: 2×10^{-4}, applied to all layers Learning Rate: 1.2×10^{-5} Batch Size: 250 Batch Normalization: applied before activation function for all layers except output layer
EFN	Φ Hidden Layers: 5 Φ Nodes per Layer: 350 Latent Dropout: 0.084 F Hidden Layers: 5 F Nodes per Layer: 300 F Dropout: 0.036 *Activation Functions: ReLU* *Kernel Initialization: glorot normal* Learning Rate: 6.3×10^{-5} Batch Size: 350
PFN	Φ Hidden Layers: 5 Φ Nodes per Layer: 250 Latent Dropout: 0.072 F Hidden Layers: 5 F Nodes per Layer: 500 F Dropout: 0.022 *Acitvation Functions: ReLU* *Kernel Initialization: glorot normal* Learning Rate: 7.9×10^{-5} Batch Size: 250

Model	Hyper-parameters
ResNet 50	Bottom Layer: 7x7 2D convolution with strides (2, 2) and zero padding Number of Stages: 4 Blocks per Stage: (3, 4, 6, 3) *Block Type: bottleneck* Block Output Filters: (64, 128, 256, 512) *Activation Functions: ReLU* *Kernel Initialization: he uniform* Batch Normalization Momentum: 0.1 *Global Pooling: average* Initial Learning Rate: 1×10^{-2} *Scheduler: decrease learning rate by factor of 0.1 every 10 epochs* Batch Size: 256
ParticleNet	Φ Number of Stages: 3 Blocks per Stage: (3, 3, 3) Block Output Features: (64, 224, 384) k Nearest Neighbors: 18 Top Layer Nodes: 125 *Activation Functions: ReLU* *Kernel Initialization: glorot normal* Batch Normalization Momentum: 0.7 Global Pooling: max Learning Rate: 4.2×10^{-4} Batch Size: 250

ATL-PHYS-PUB-2022-039

MISSING ENERGY IN ATLAS

Table 6: METNet neural network achitecture and hyperparameters.

Layer	Nodes	Activation	Parameters
Input	60	None	0
Hidden 1	100	SELU	6100
LayerNorm 1	100	SELU	200
Hidden 2	100	SELU	10100
LayerNorm 2	100	SELU	200
Hidden 3	100	SELU	10100
LayerNorm 3	100	SELU	200
Output	2	Linear	202
Total			27,102

(a) METNet architecture.

Hyperparameter	Value
Optimiser	Adam [67]
Weight initialization	Kaiming He [68]
Learning rate	0.001
Batch size	256
Huber loss δ	1.5

(b) METNet hyperparameters