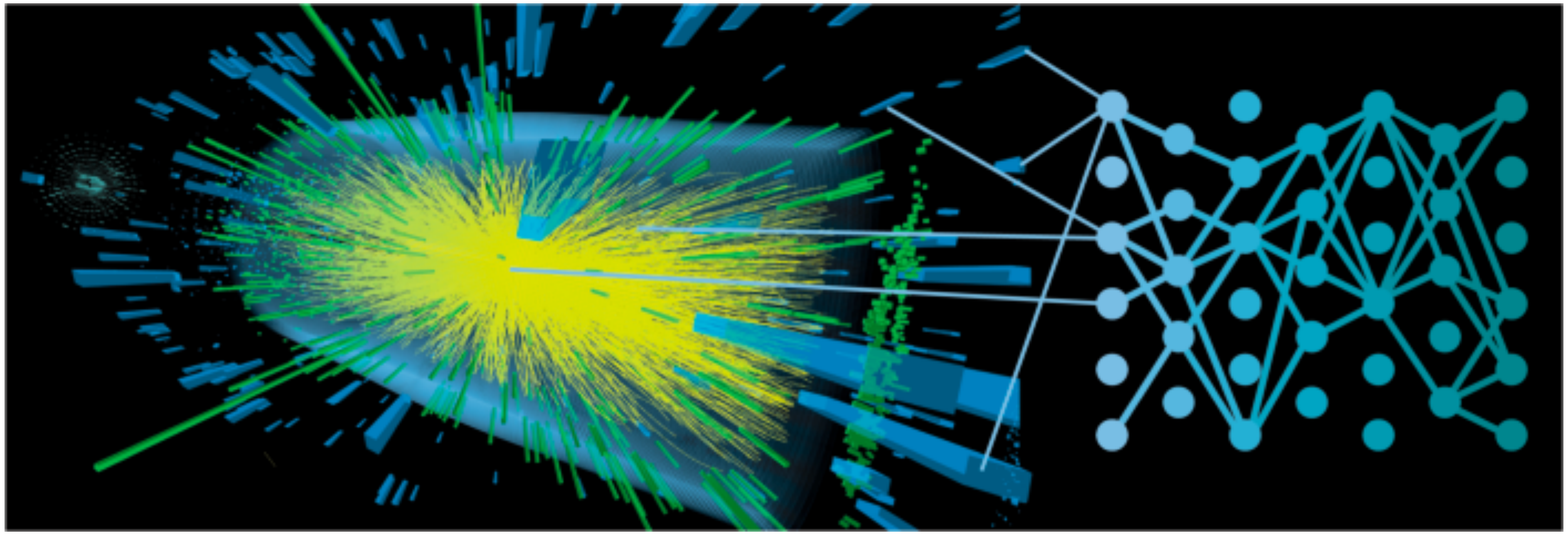
Application of Machine Learning at HEP colliders

Sanmay Ganguly
IIT-Kanpur

ICHEPAP -2023, Saha Institute

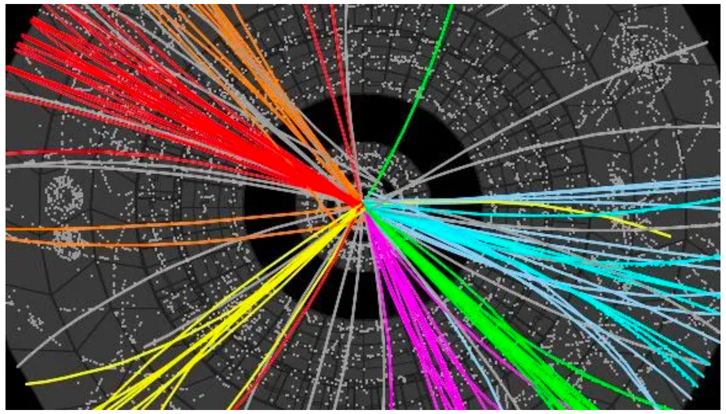


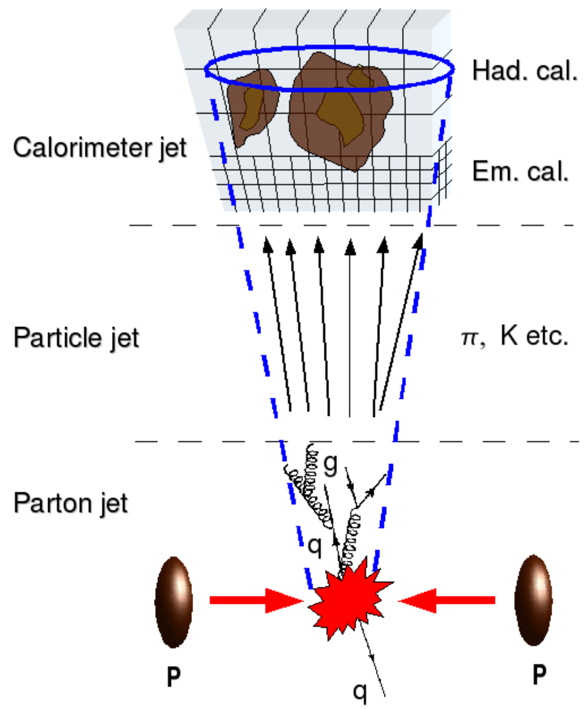
Outline

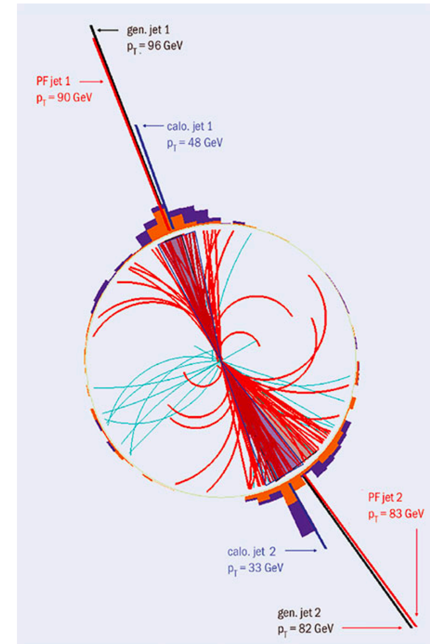
- **This is a summary about areas where we apply ML in collider physics.**
- **This seminar is not about what is GNN/GAN**
- **The emphasis will be on latest architectures and SOTA performances (biased with CMS/ATLAS results)**
- **Will also have some discussion on open data which is crucial for HEP-ML R&D**

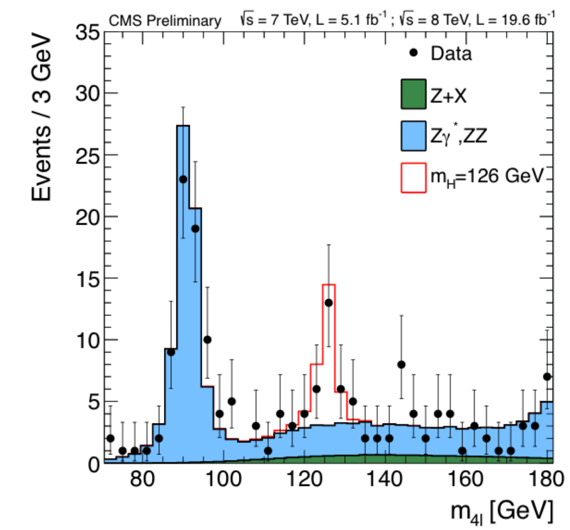
The LHC data flow-chain

ML can play a role at every instance of this flow chain.

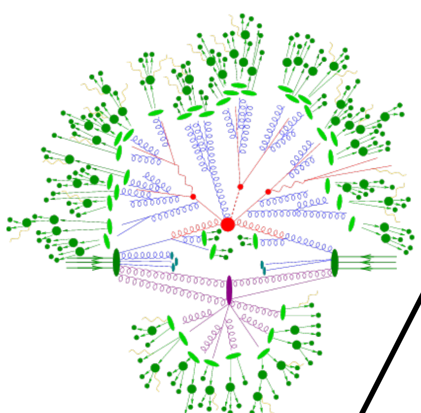


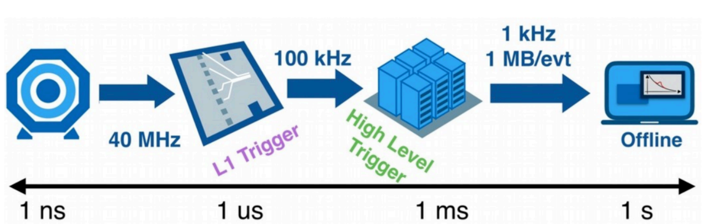


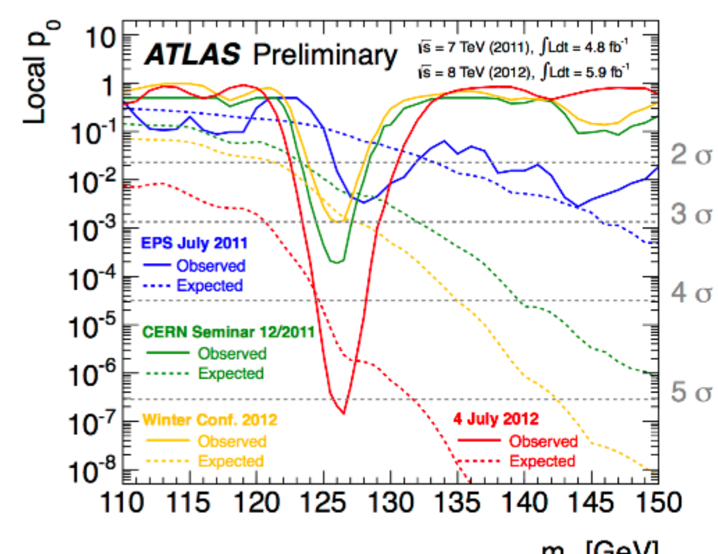




Trigger → Object rec → Event rec → Analysis

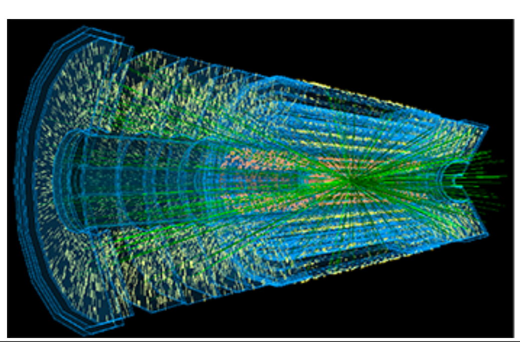




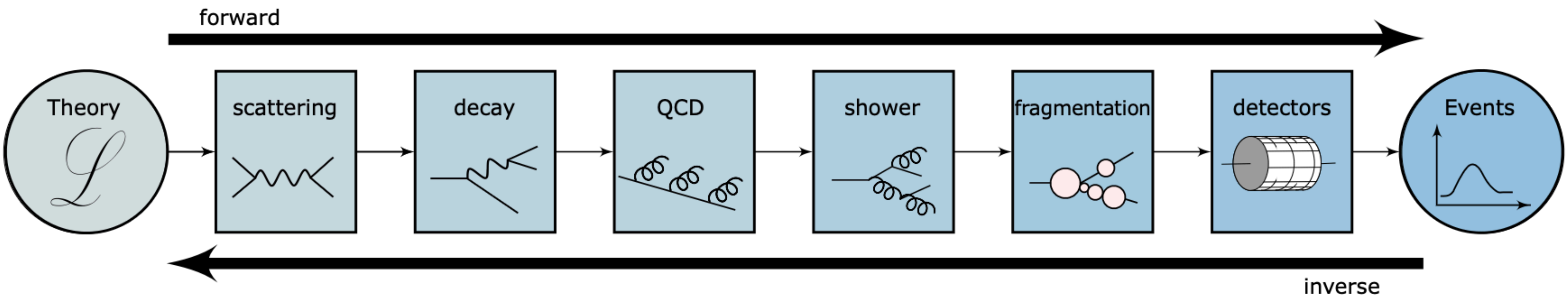


Inference

ME + PS generation
Detector Simulation



Monte-Carlo modelling using ML



arXiv > hep-ph > arXiv:2203.07460

Search...

Help | Advance

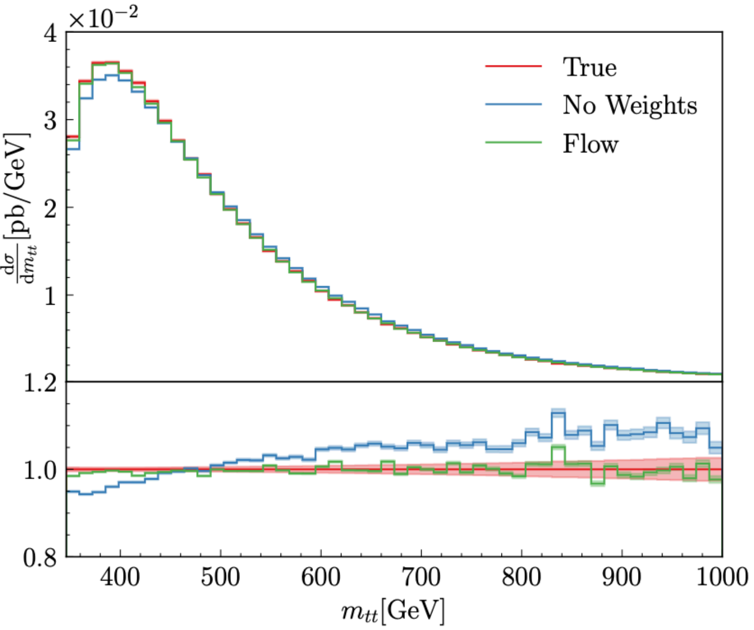
High Energy Physics – Phenomenology

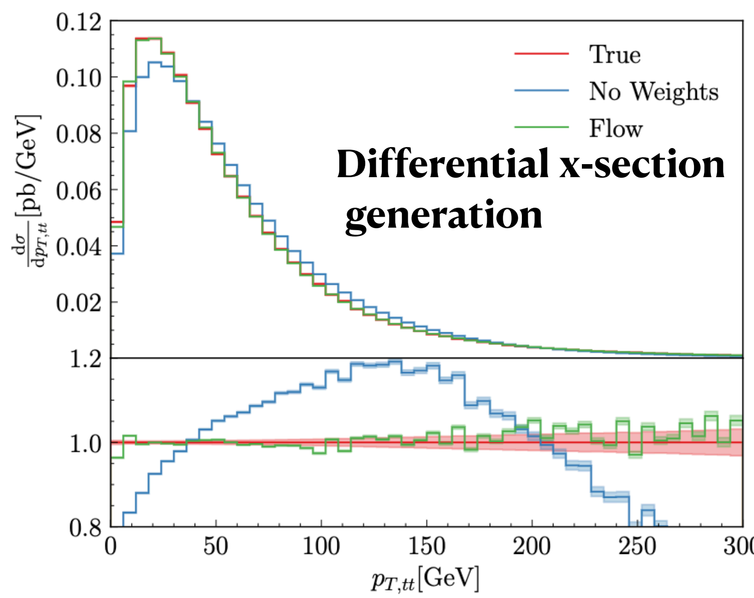
[Submitted on 14 Mar 2022 (v1), last revised 28 Dec 2022 (this version, v2)]

Machine Learning and LHC Event Generation

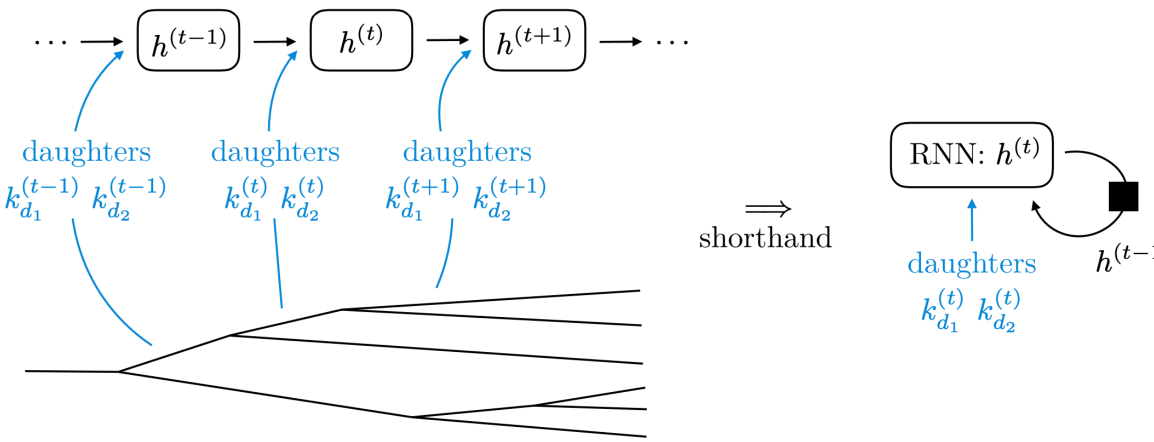
Anja Butter (ed), Tilman Plehn (ed), Steffen Schumann (ed), Simon Badger, Sascha Caron, Kyle Cranmer, Francesco Armando Di Bello, Etienne Dreyer, Stefano Forte, Sanmay Ganguly, Dorival Gonçalves, Eilam Gross, Theo Heimel, Gudrun Heinrich, Lukas Heinrich, Alexander Held, Stefan Höche, Jessica N. Howard, Philip Ilten, Joshua Isaacson, Timo Janßen, Stephen Jones, Marumi Kado, Michael Kagan, Gregor Kasieczka, Felix Kling, Sabine Kraml, Claudius Krause, Frank Krauss, Kevin Kröninger, Rahool Kumar Barman, Michel Luchmann, Vitaly Magerya, Daniel Maitre, Bogdan Malaescu, Fabio Maltoni, Till Martini, Olivier Mattelaer, Benjamin Nachman, Sebastian Pitz, Juan Rojo, Matthew Schwartz, David Shih, Frank Siegert, Roy Stegeman, Bob Stienen, Jesse Thaler, Rob Verheyen, Daniel Whiteson, Ramon Winterhalder, Jure Zupan

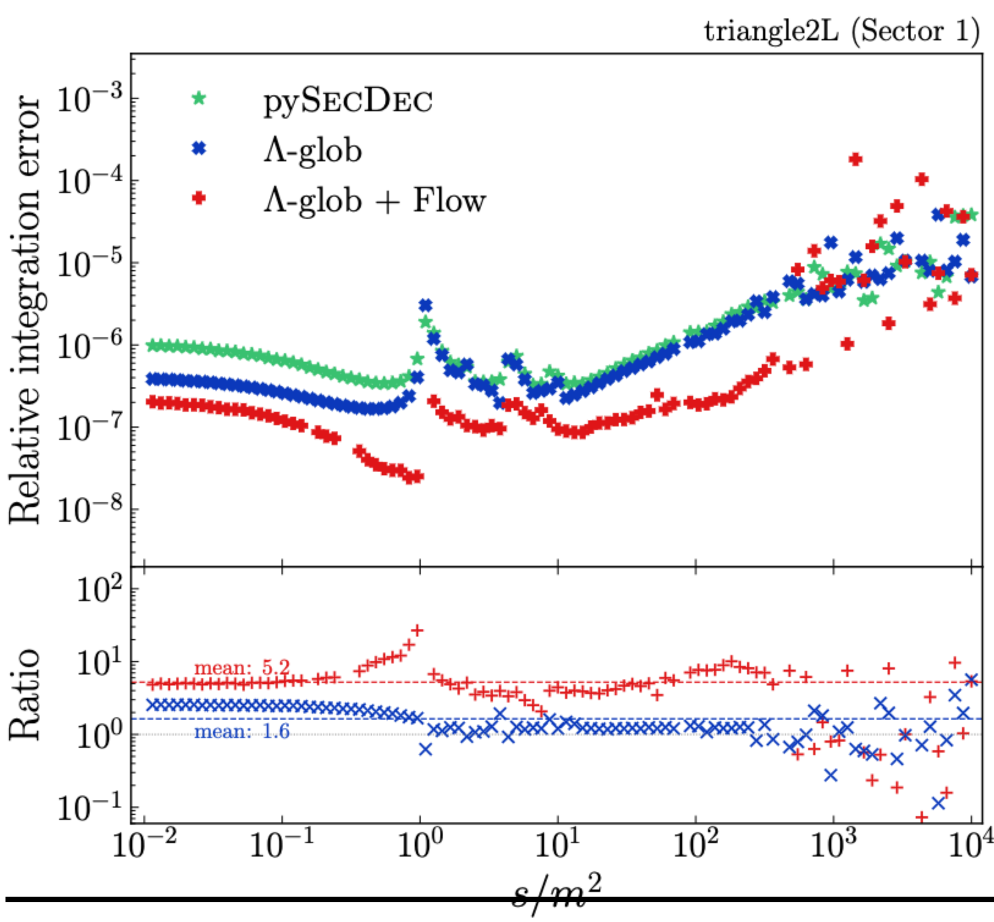
ML based ME + PS

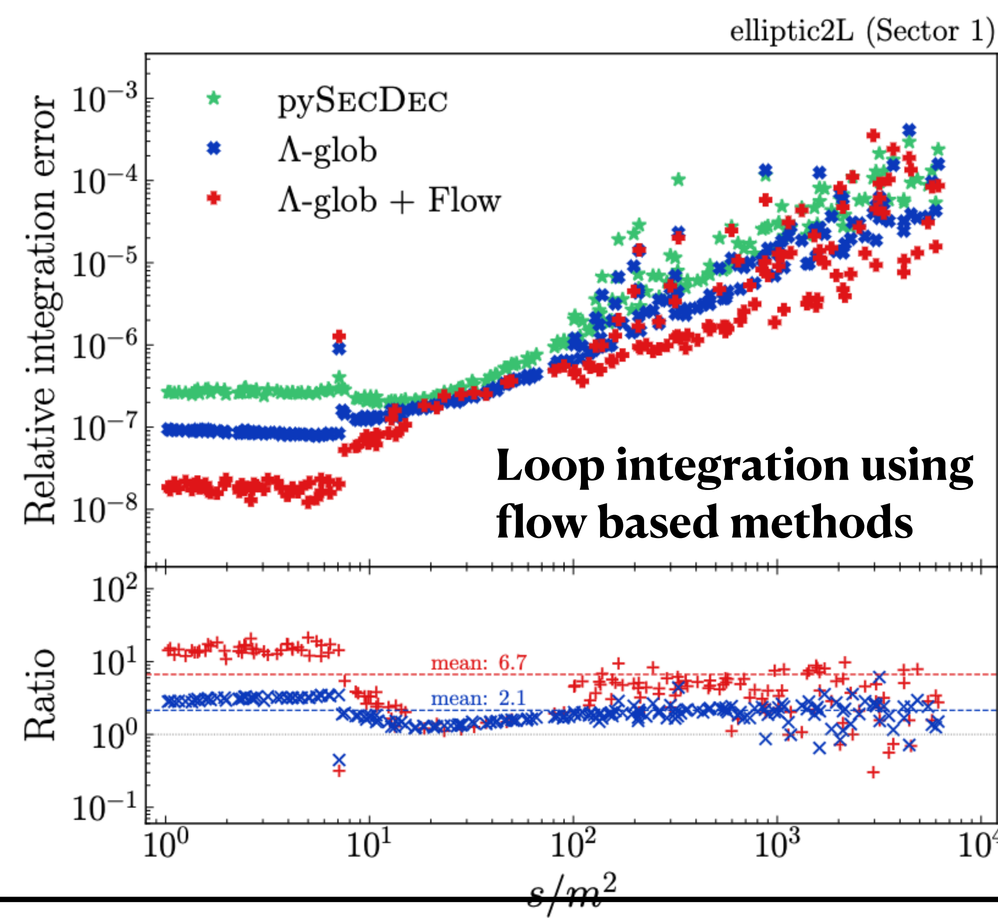




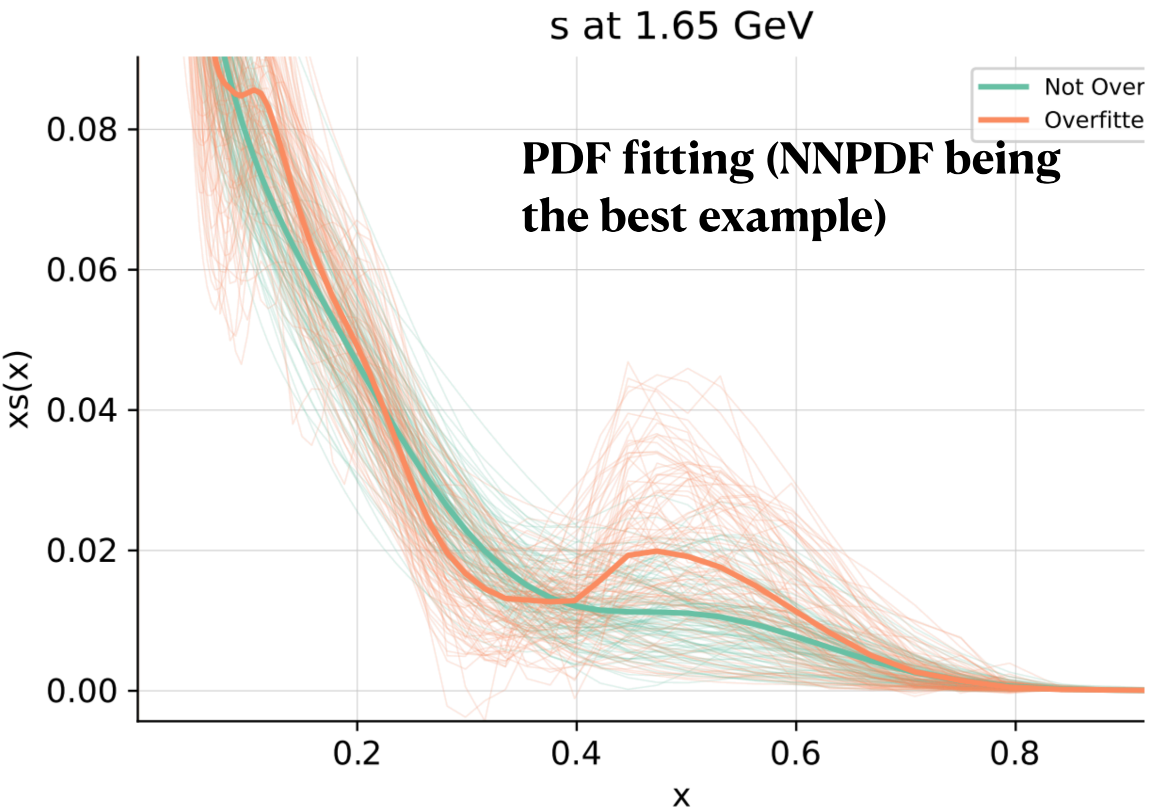
Markovian approach for Parton Shower (JUNIPR)



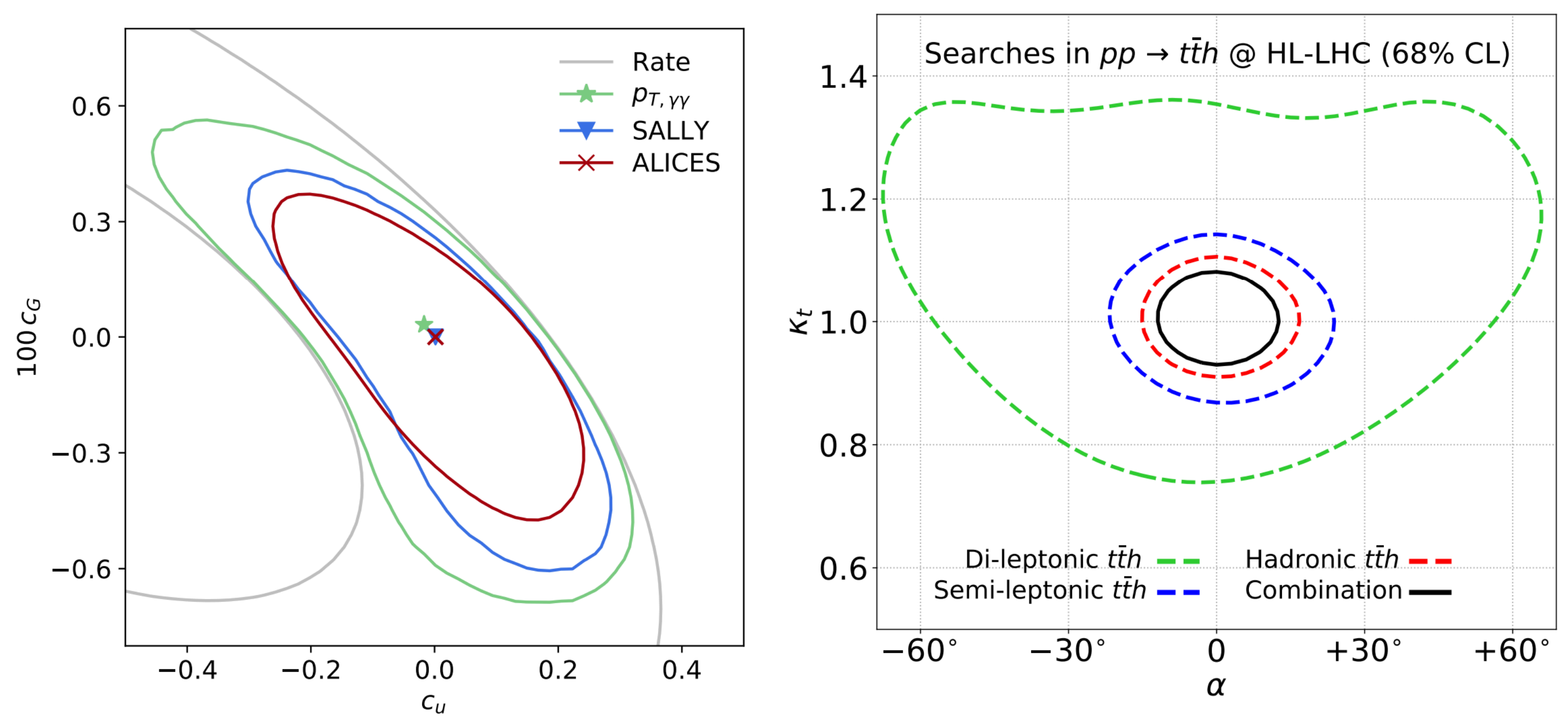


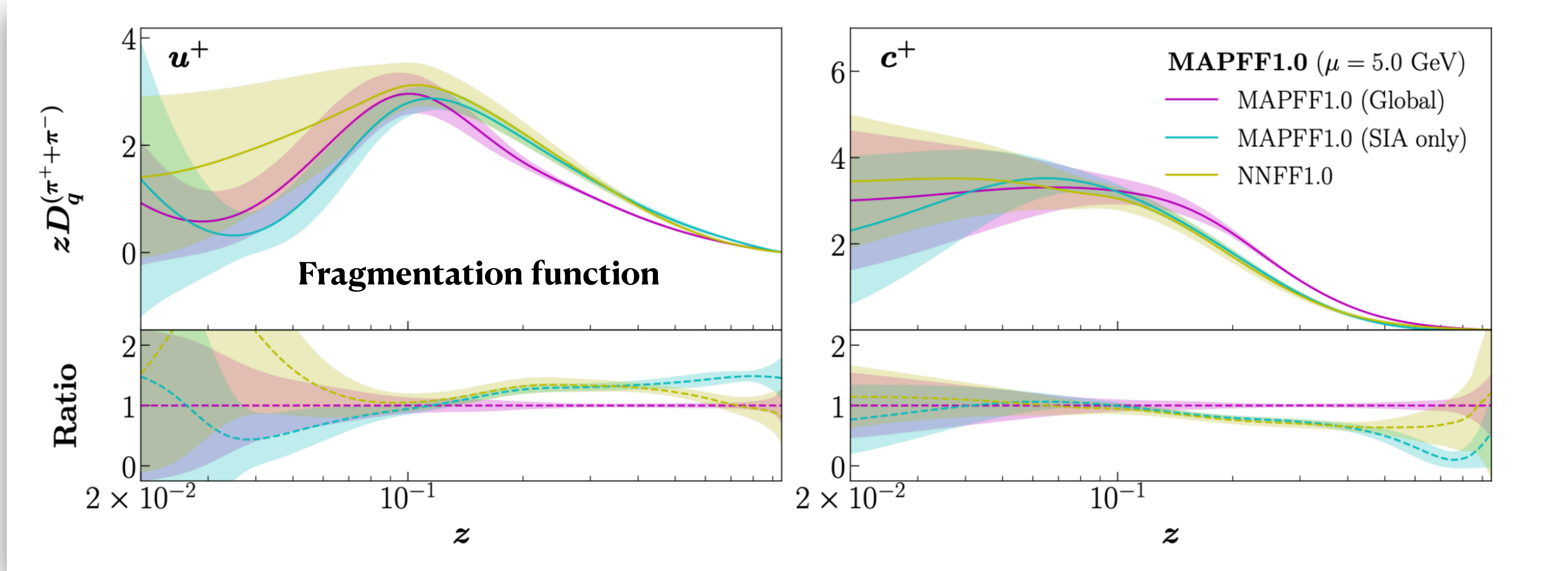


Generative models in MC



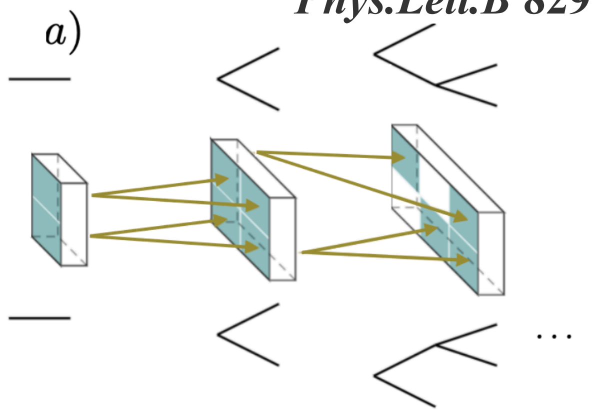
EFT parameter fitting using Madminer

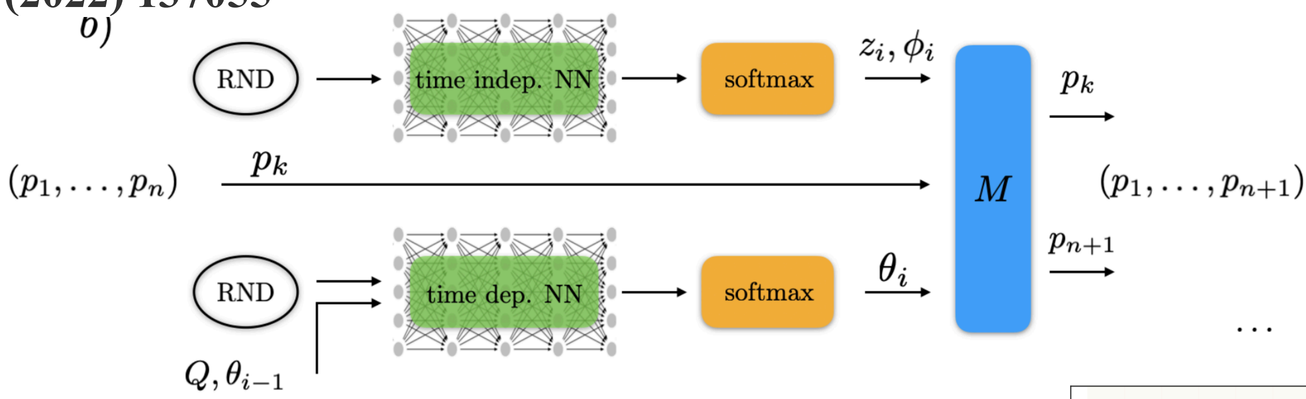


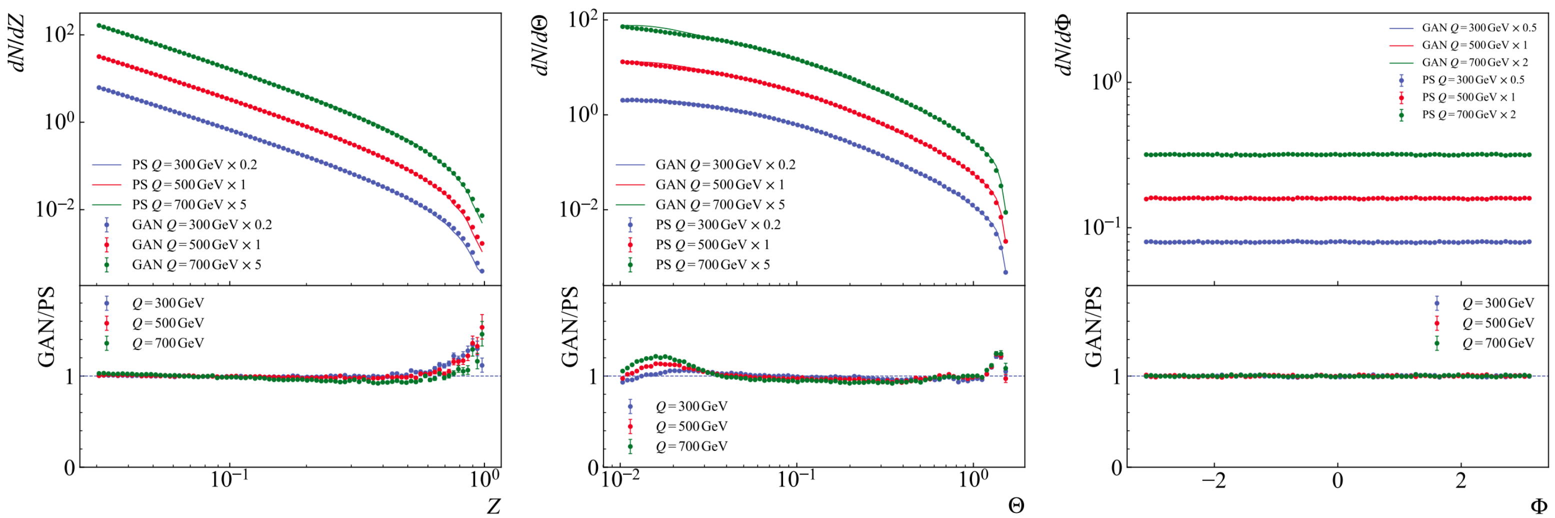


PS + Hadronization with ML

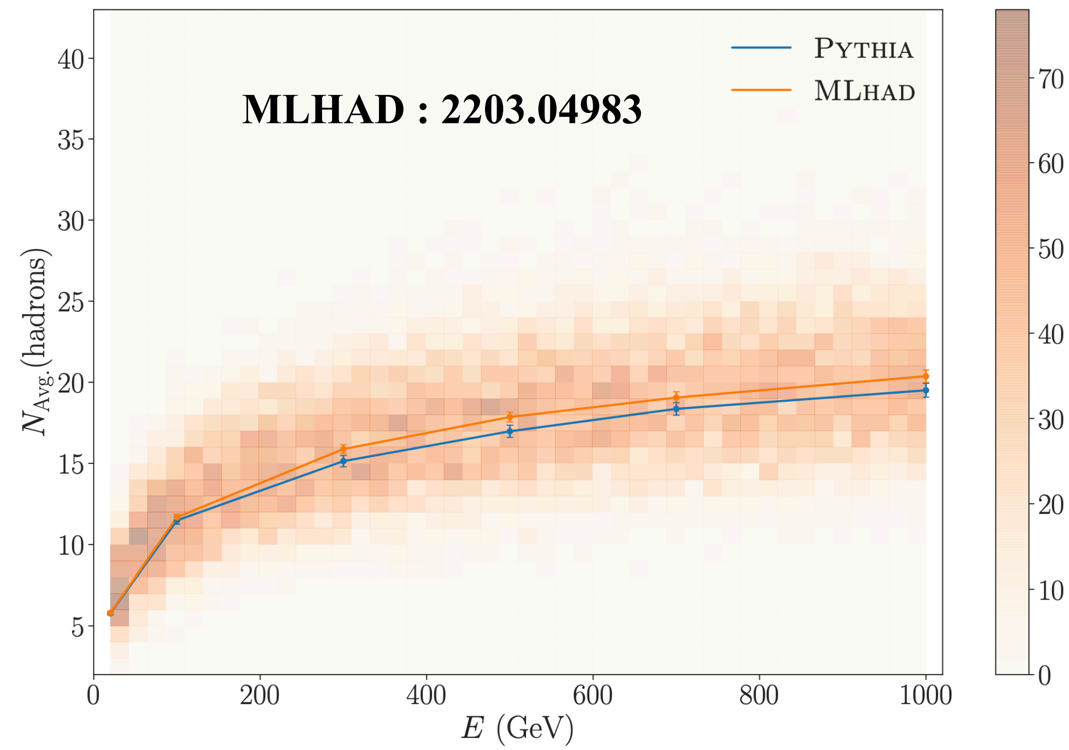
Phys.Lett.B 829 (2022) 137055



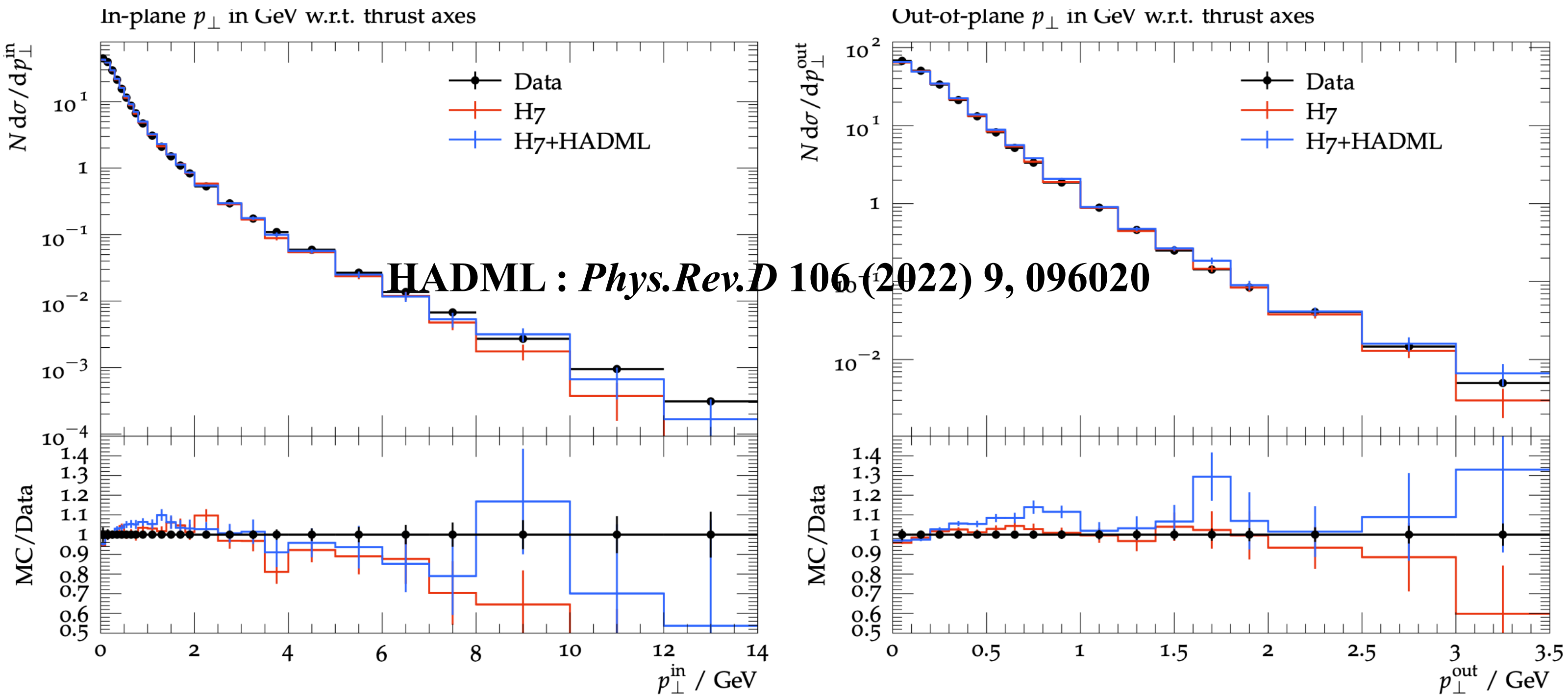




MLHAD : 2203.04983



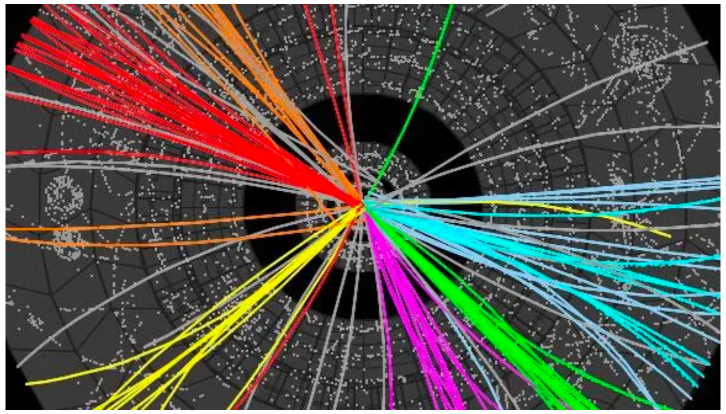
HADML : *Phys.Rev.D* 106 (2022) 9, 096020

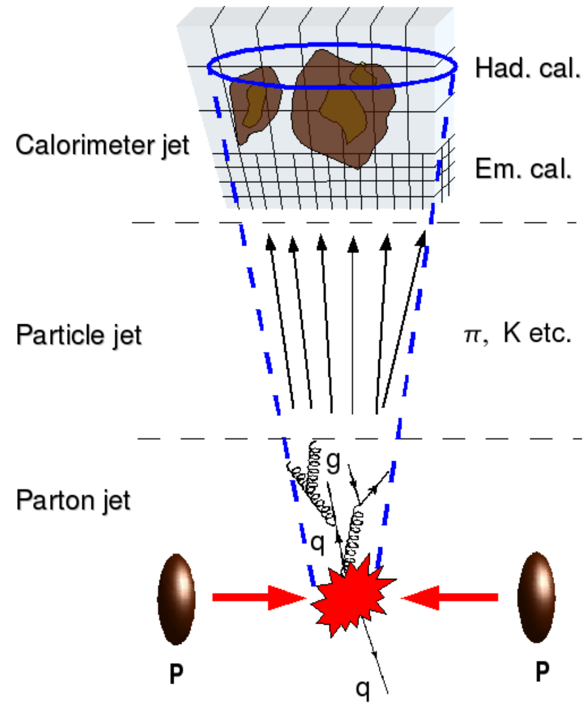


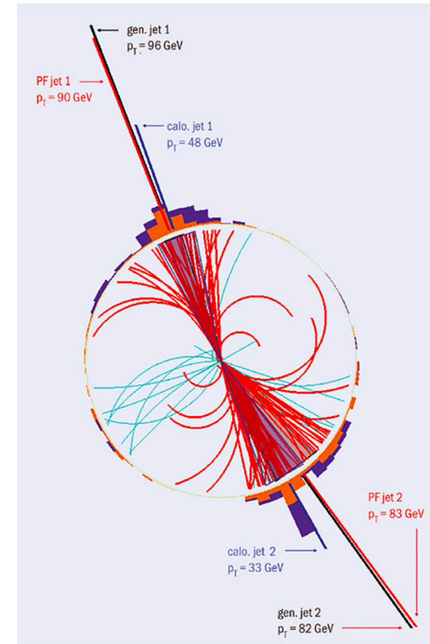
Overall shape normalization from Hadronization can be generated by generative models.

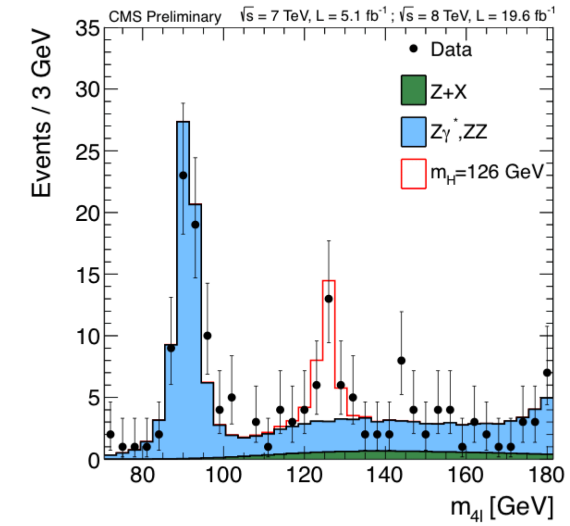
The LHC data flow-chain

ML can play a role at every instance of this flow chain.

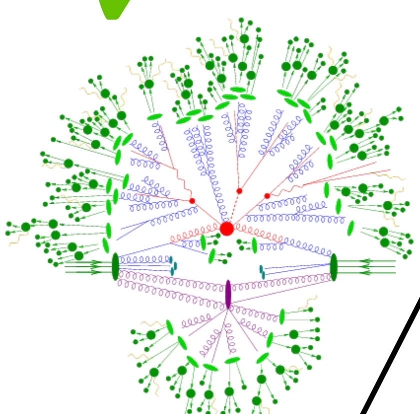


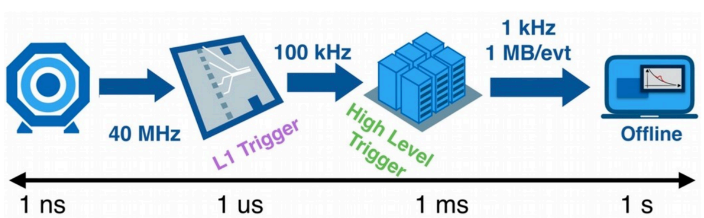


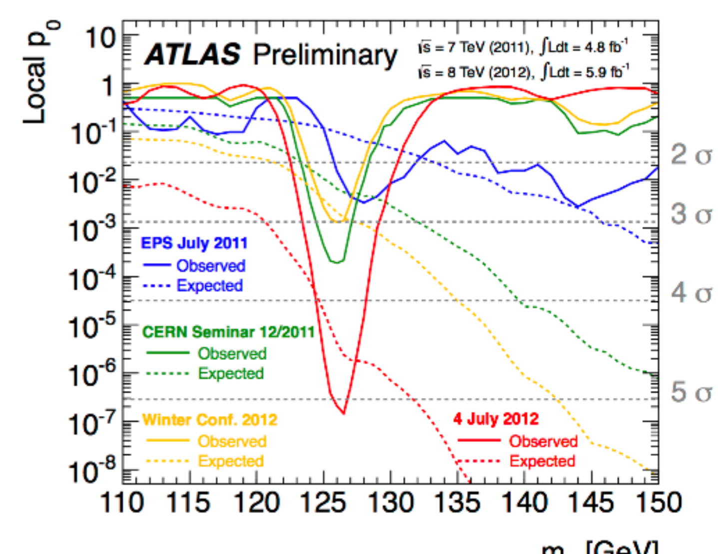




Trigger → Object rec → Event rec → Analysis

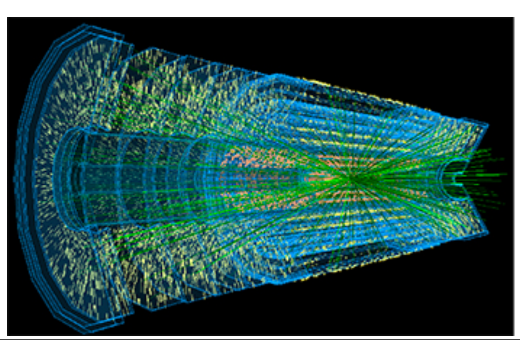




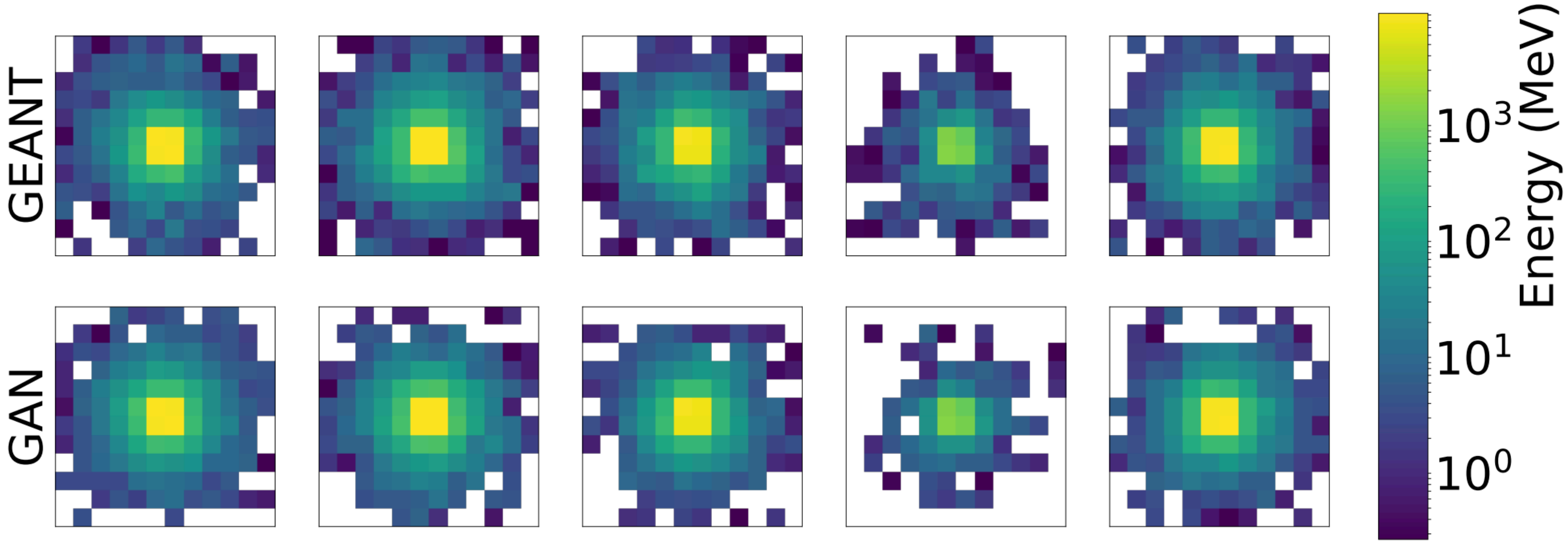


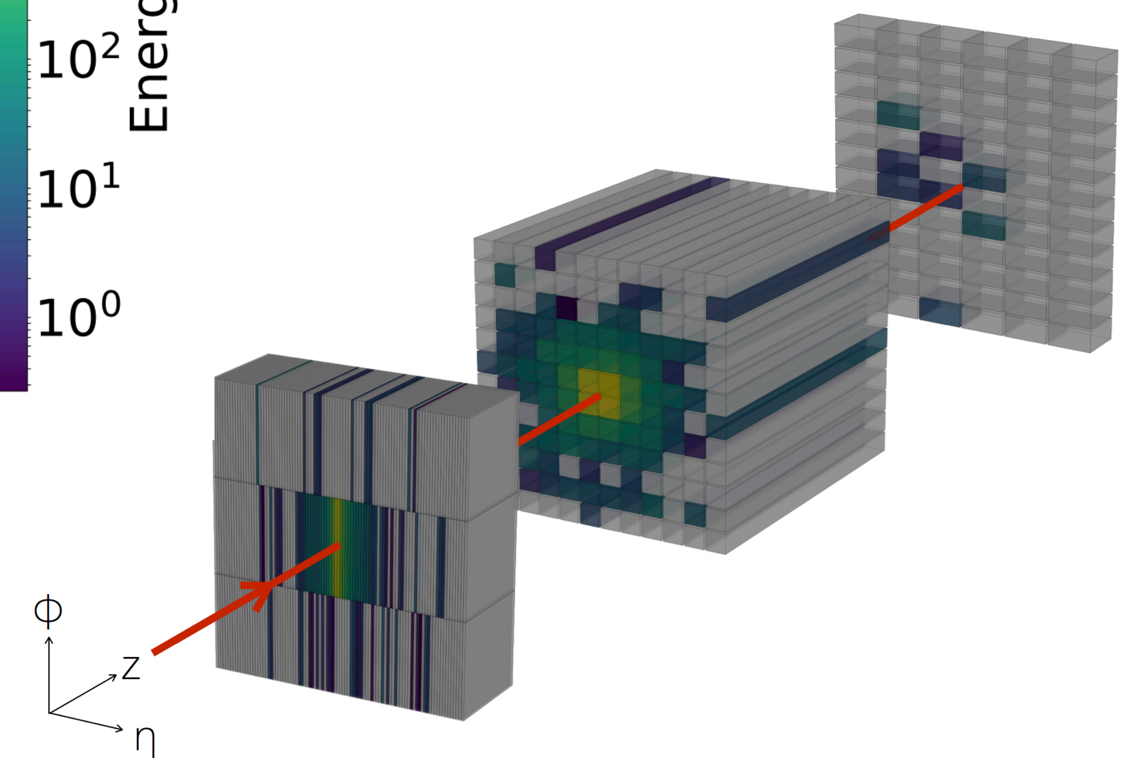
Inference

ME + PS generation
Detector Simulation

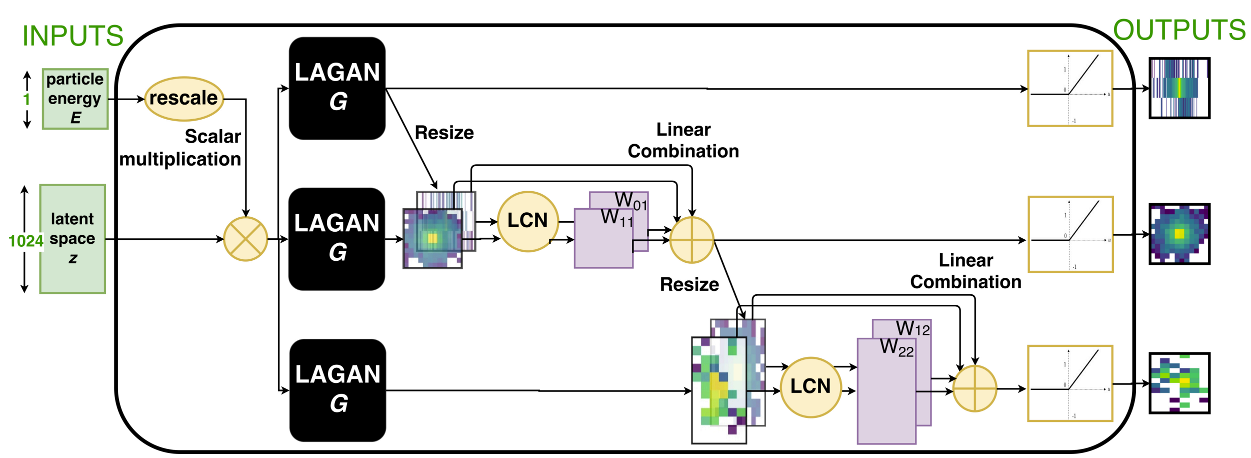


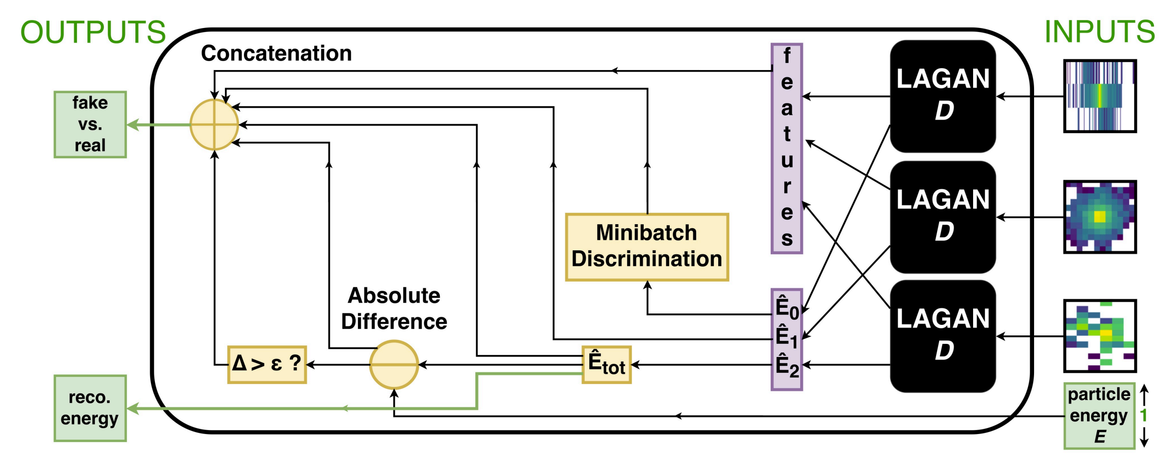
Detector simulation using ML





CaloGAN 1705.02355

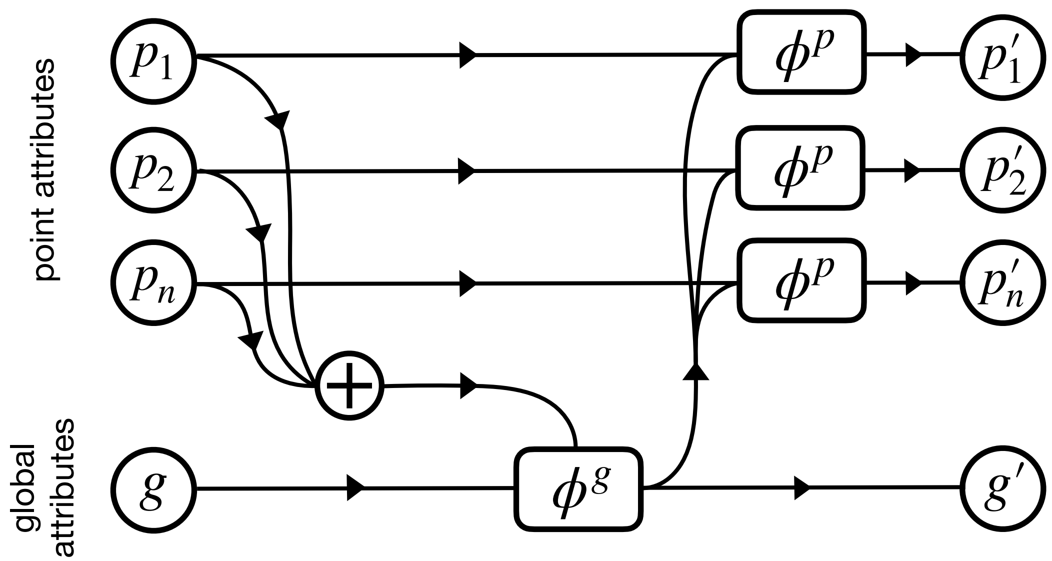


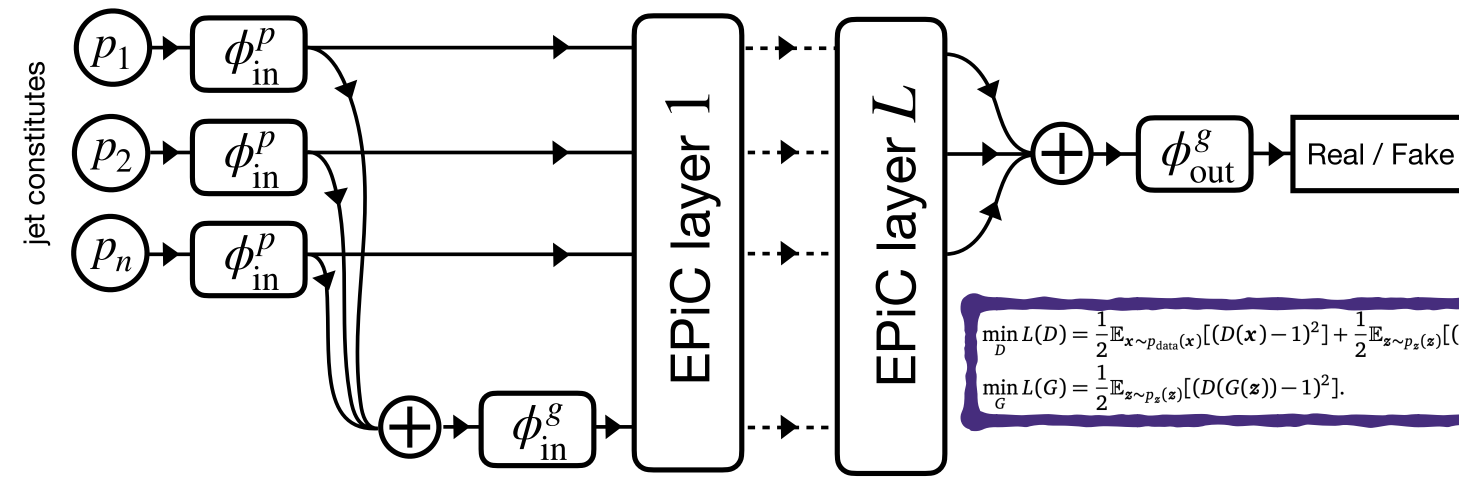


Generator

Discriminator

Detector simulation using ML

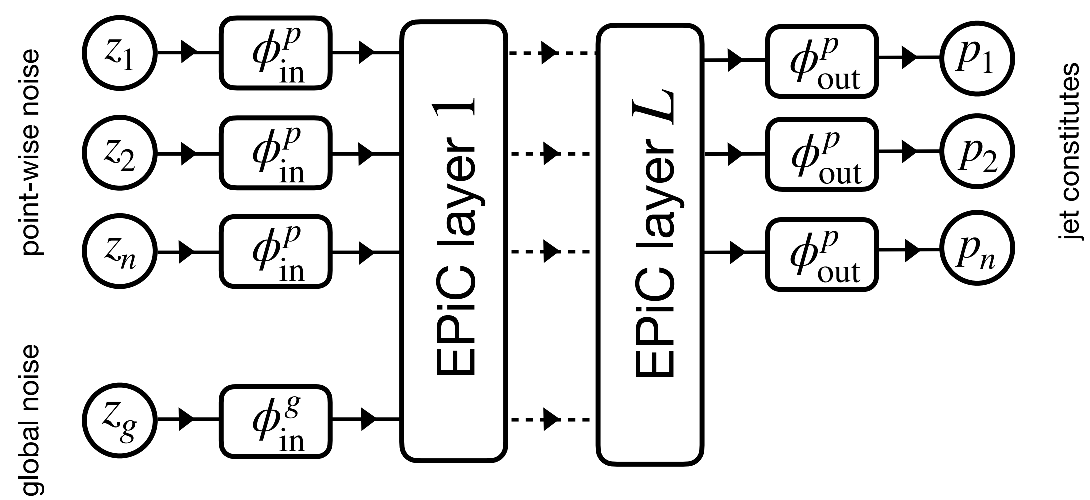


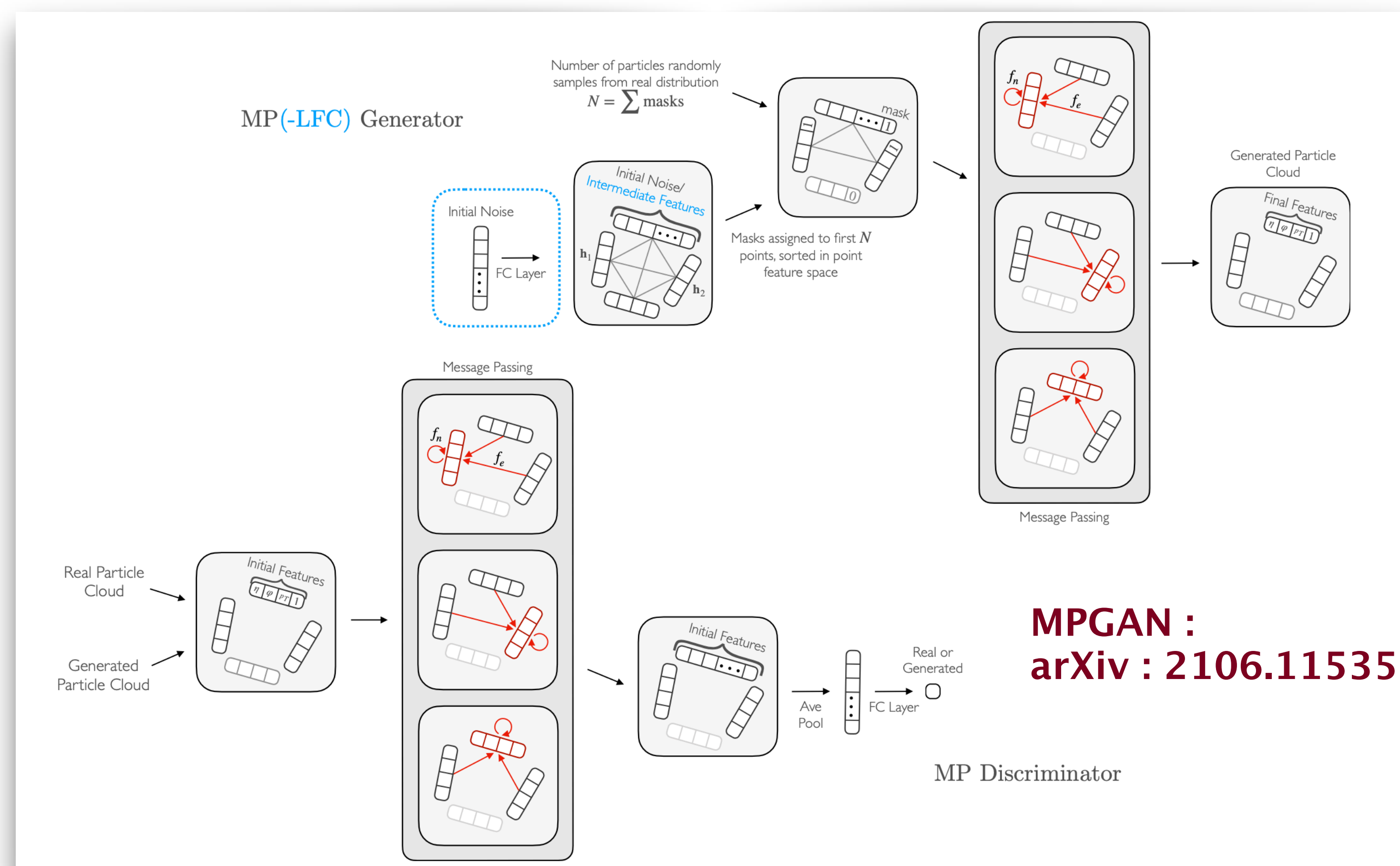


$$\min_D L(D) = \frac{1}{2}\mathbb{E}_{x\sim p_{\text{data}}(x)}[(D(x)-1)^2] + \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(z)))^2],$$

$$\min_G L(G) = \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(z))-1)^2].$$

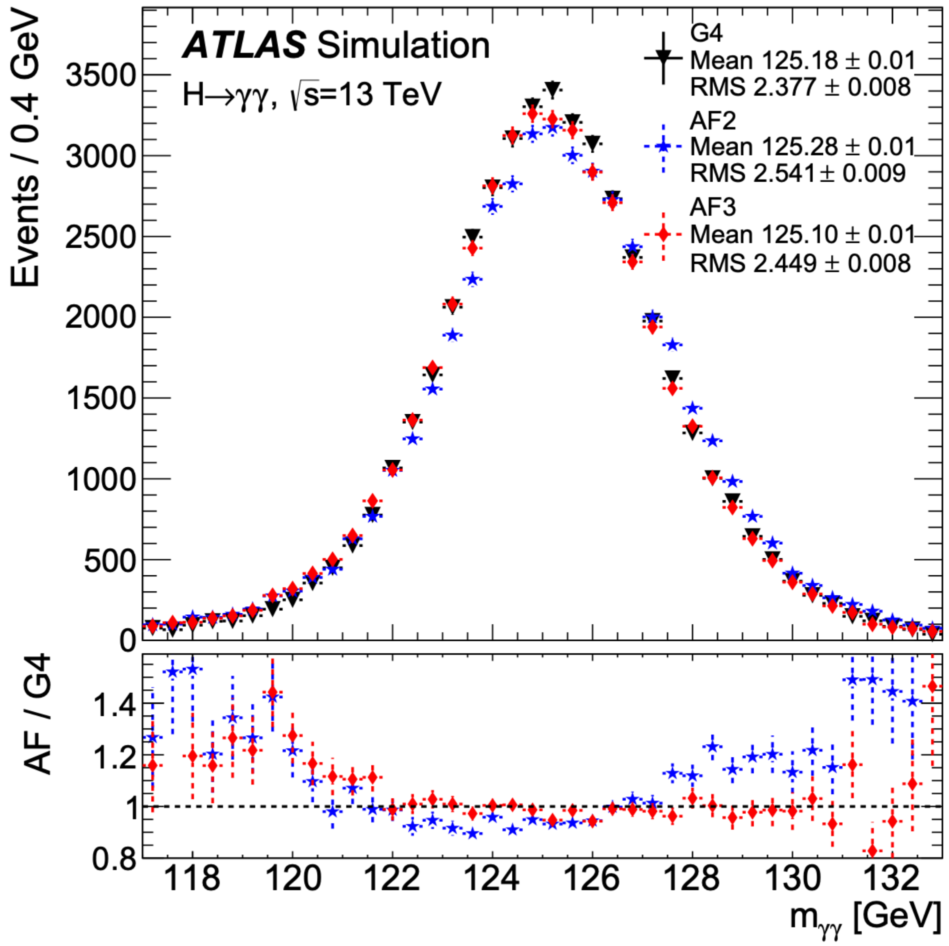
EPiC-GAN : SciPost Phys. 15, 130 (2023)

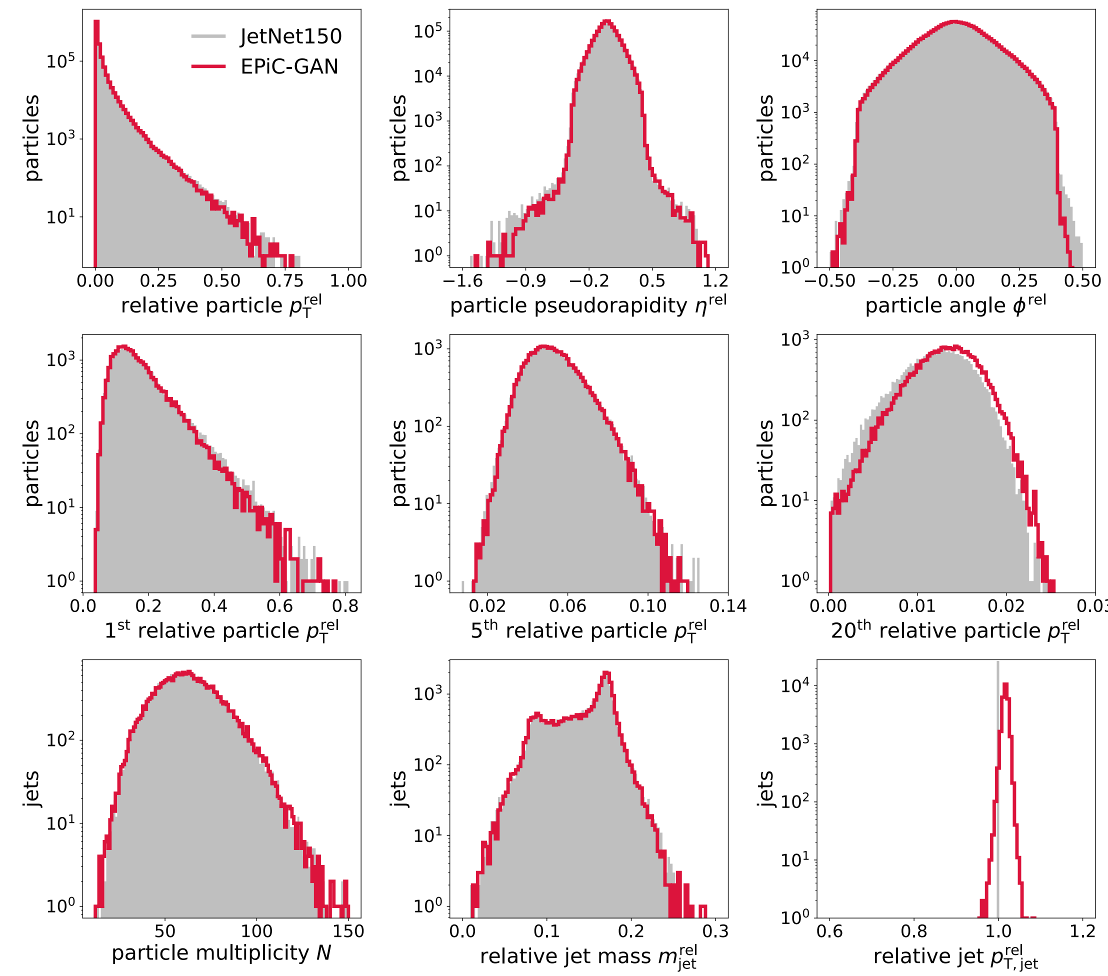




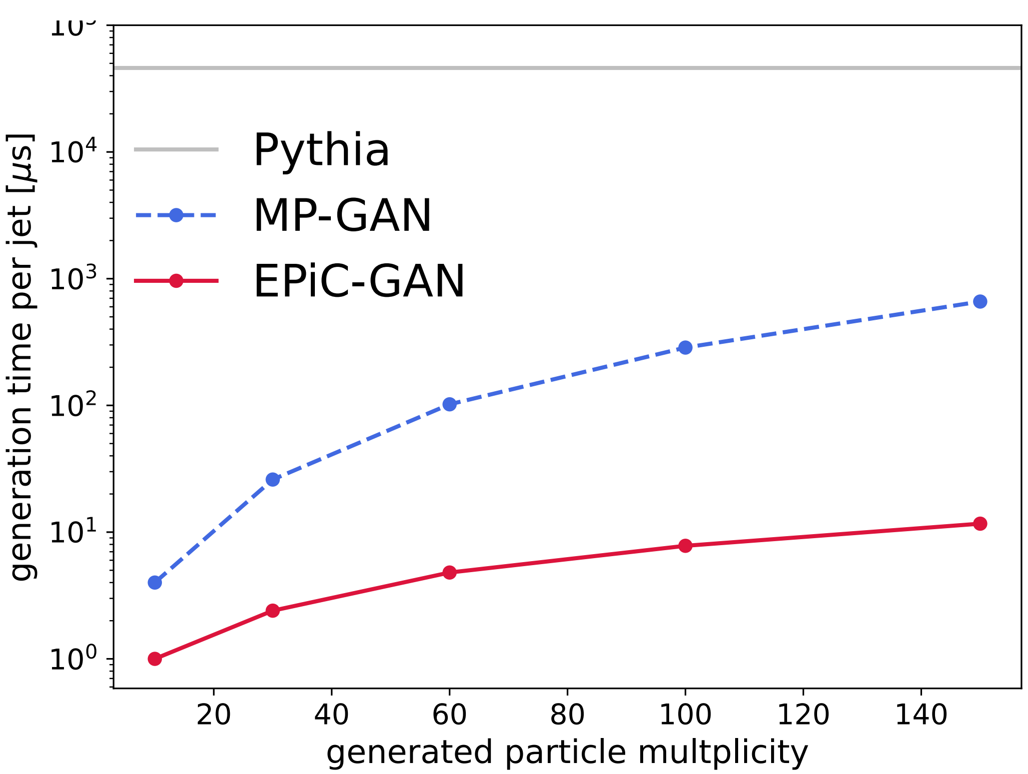
MPGAN : arXiv : 2106.11535

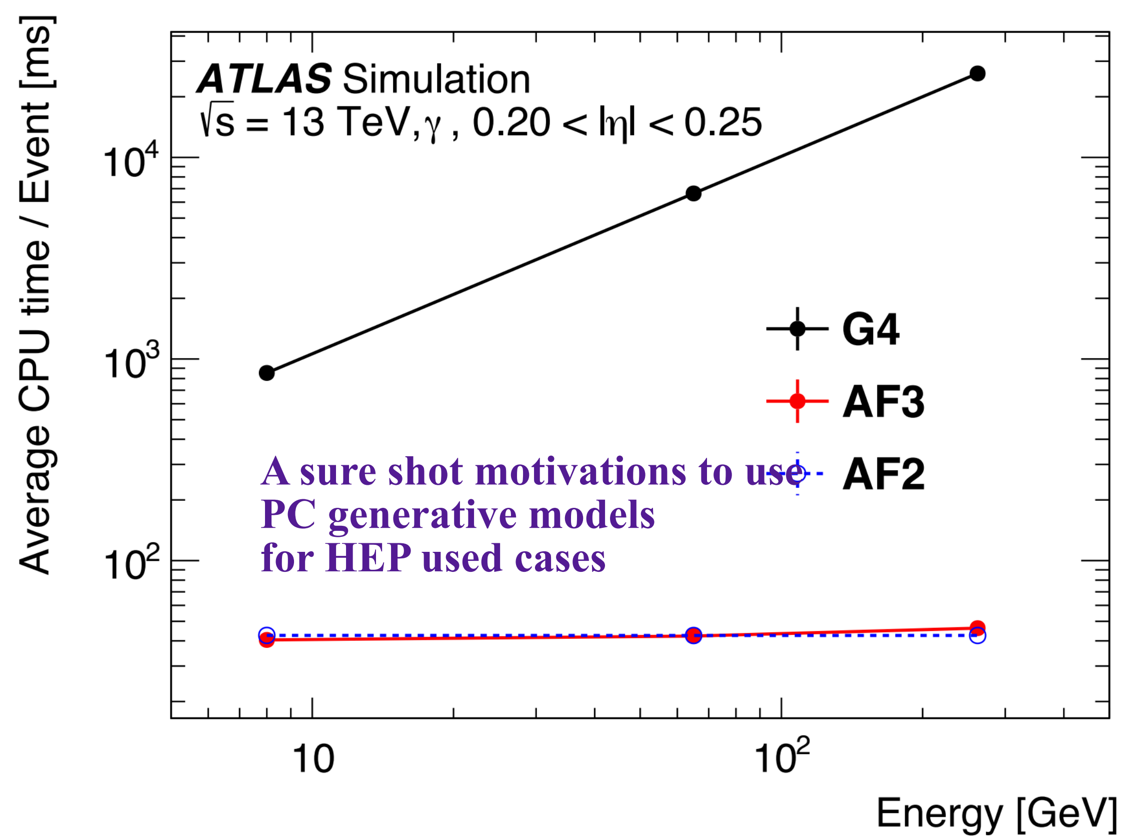
The major gain





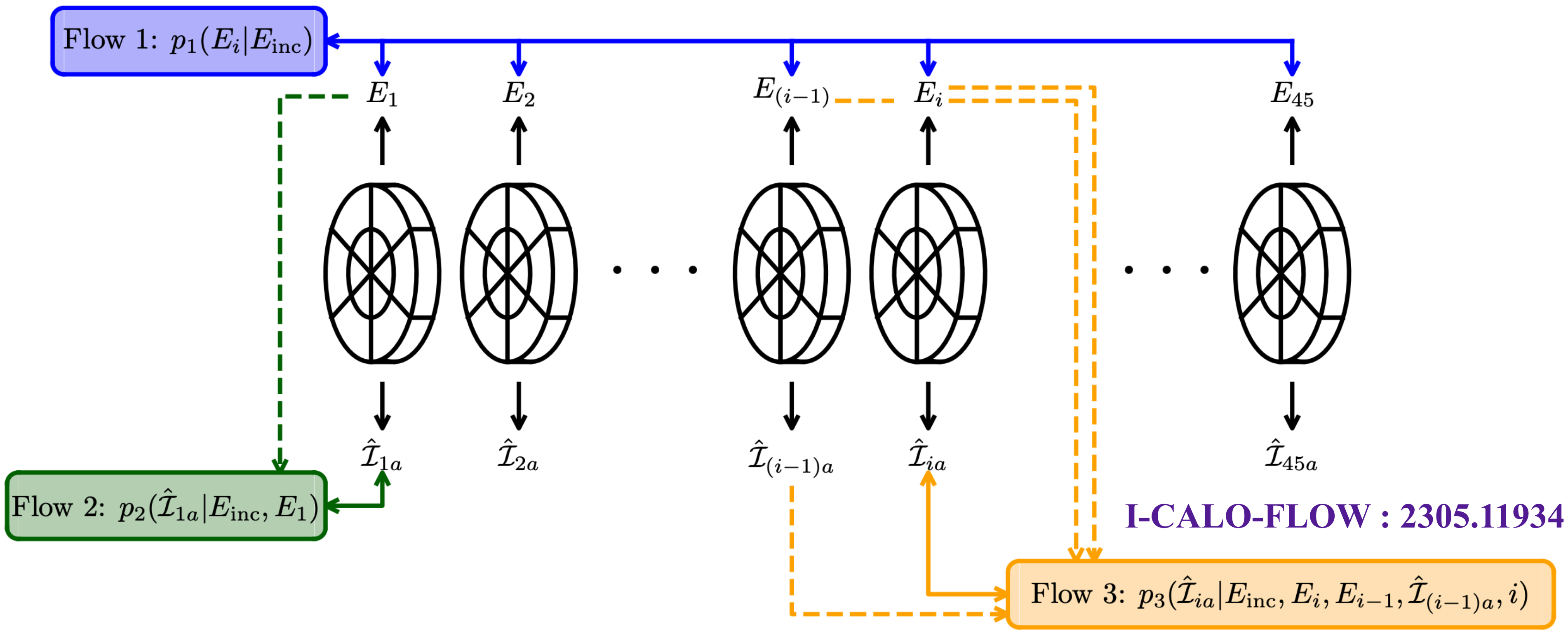
arXiv : 2109.02551



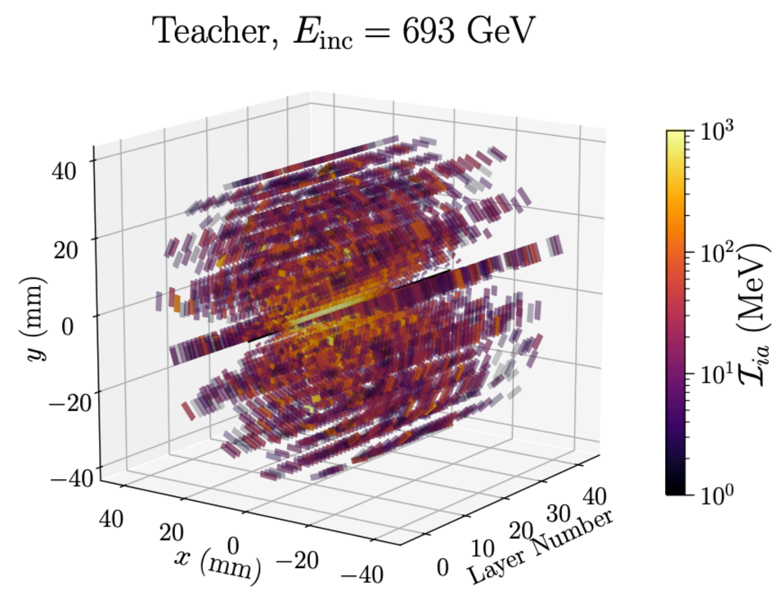


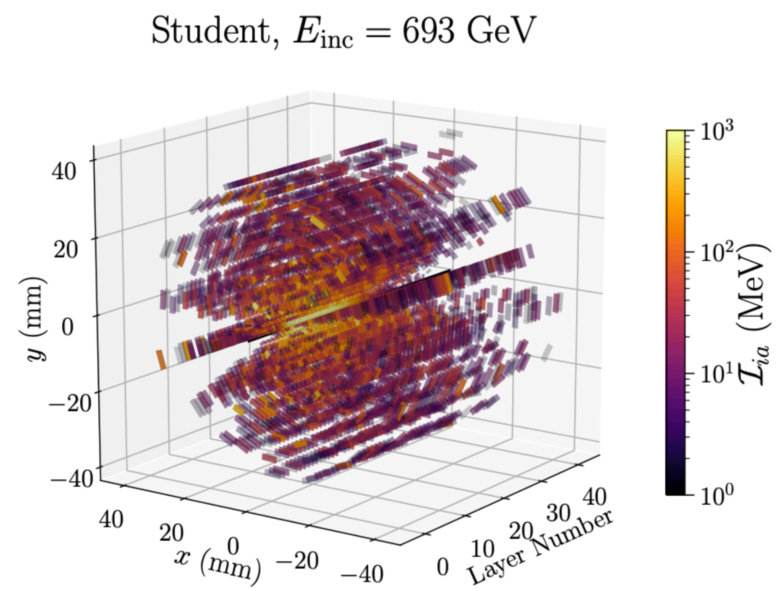
A sure shot motivations to use PC generative models for HEP used cases

The latest architectures

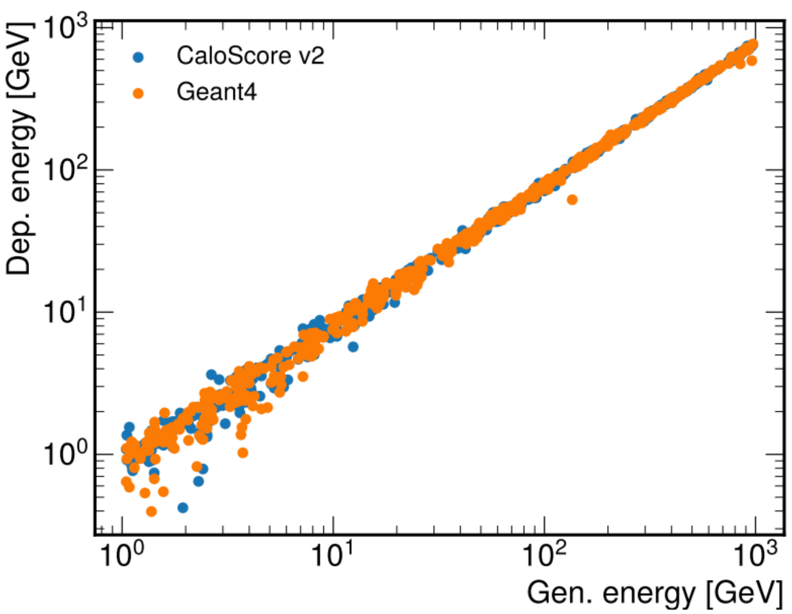


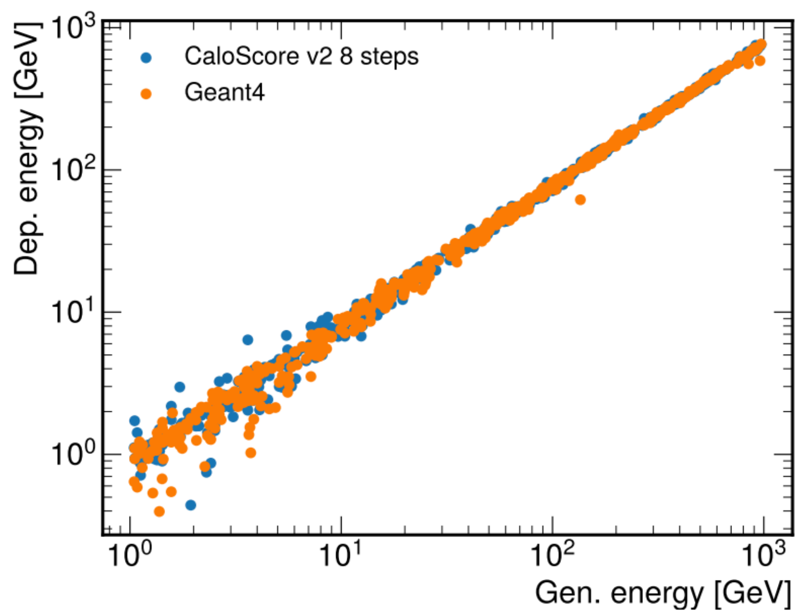
I-CALO-FLOW : 2305.11934

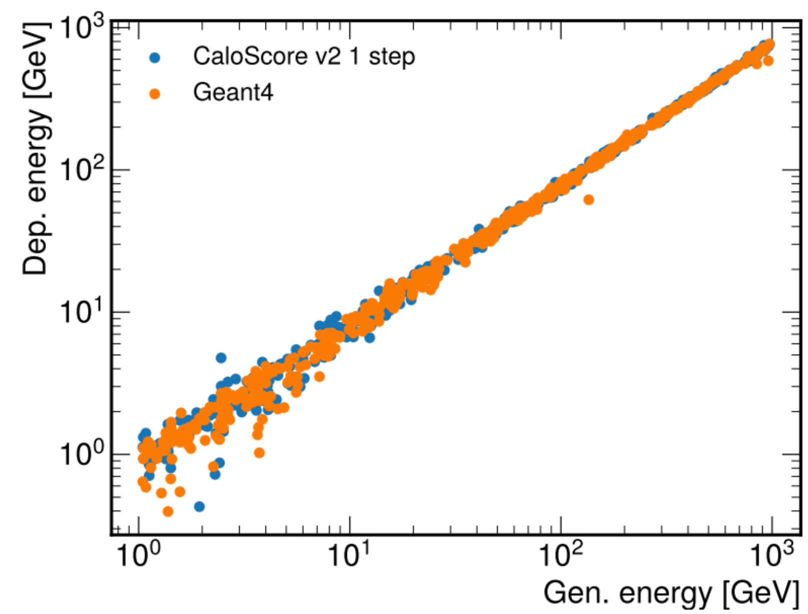




Diffusion model based CaloSimulation : 2308.03847

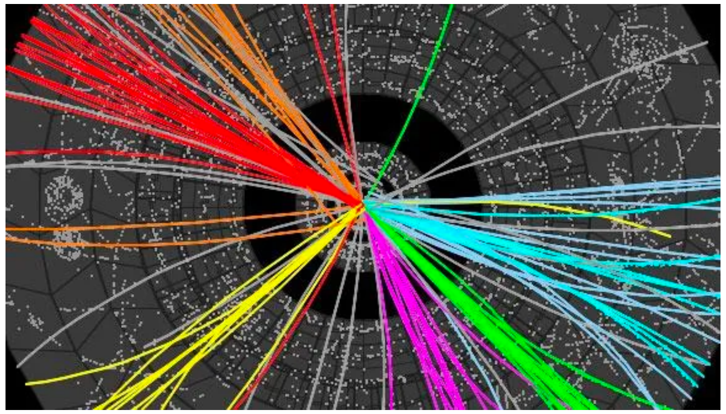


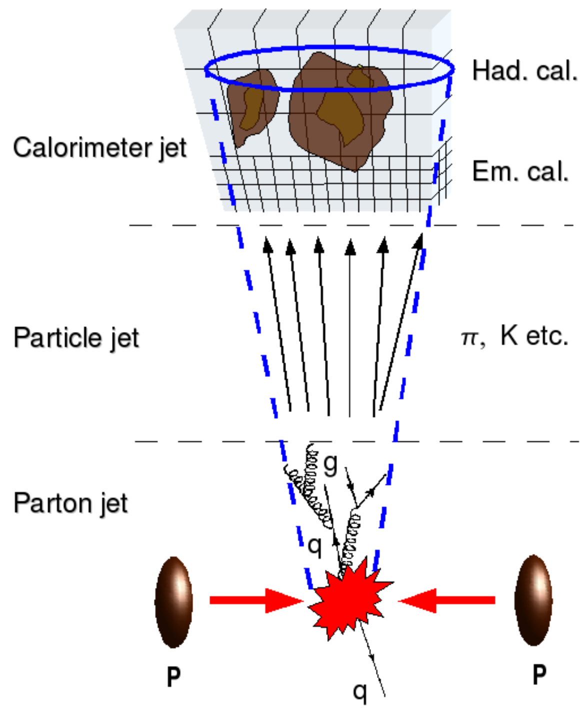


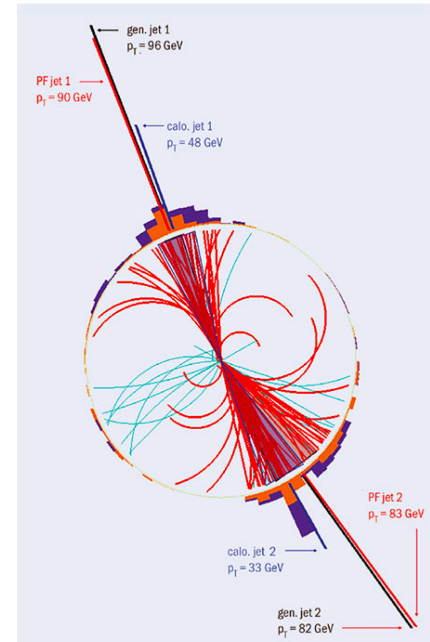


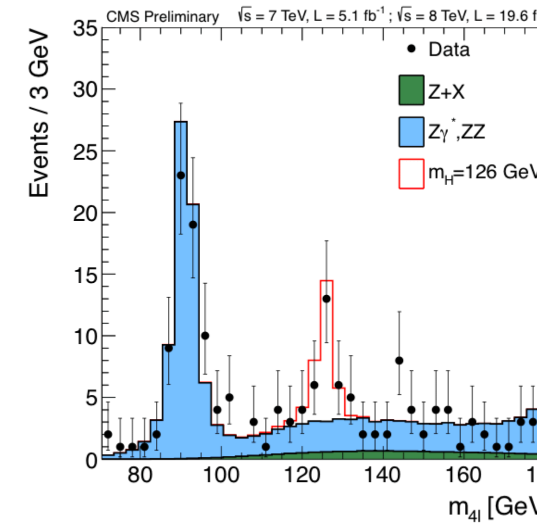
The LHC data flow-chain

ML can play a role at every instance of this flow chain.

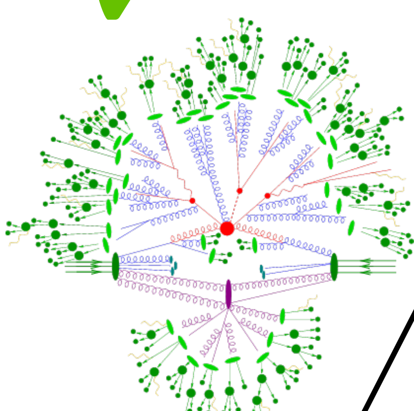


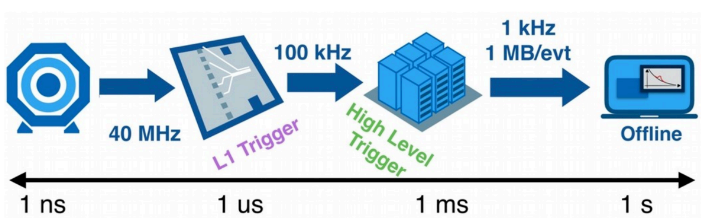


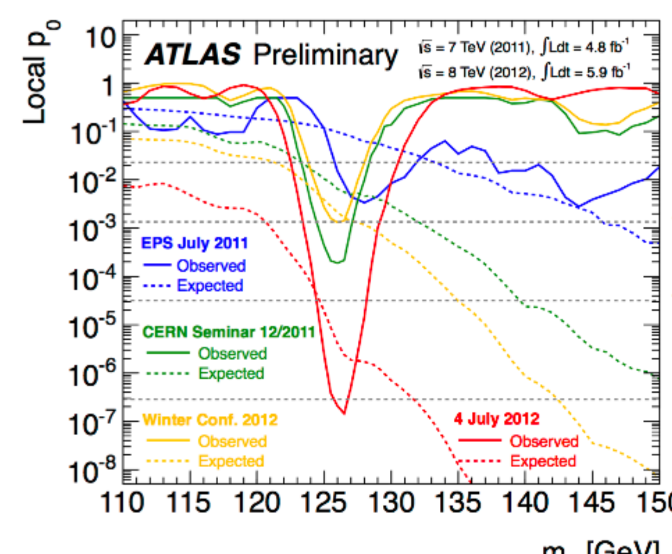




Trigger → Object rec → Event rec → Analysis

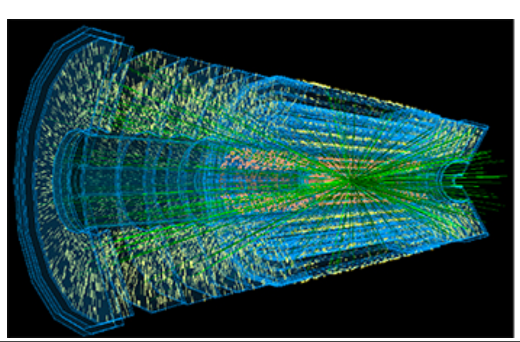




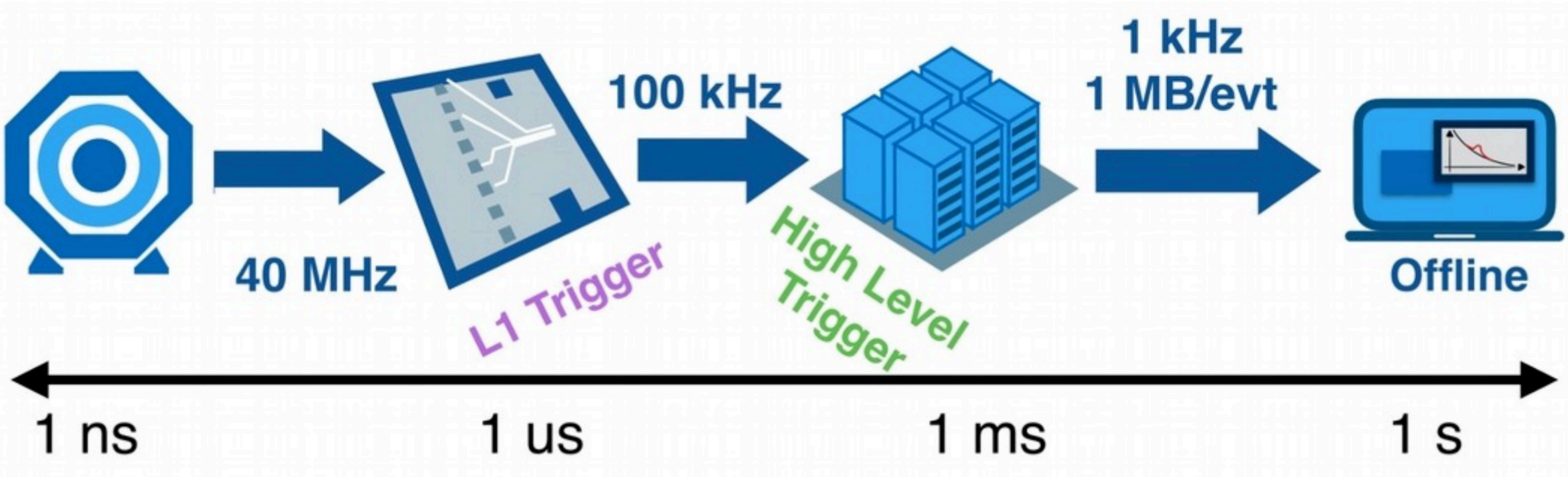


Inference

ME + PS generation
Detector Simulation



Improving trigger using ML



J. Duarte *et al* 2018 *JINST* 13 P07027

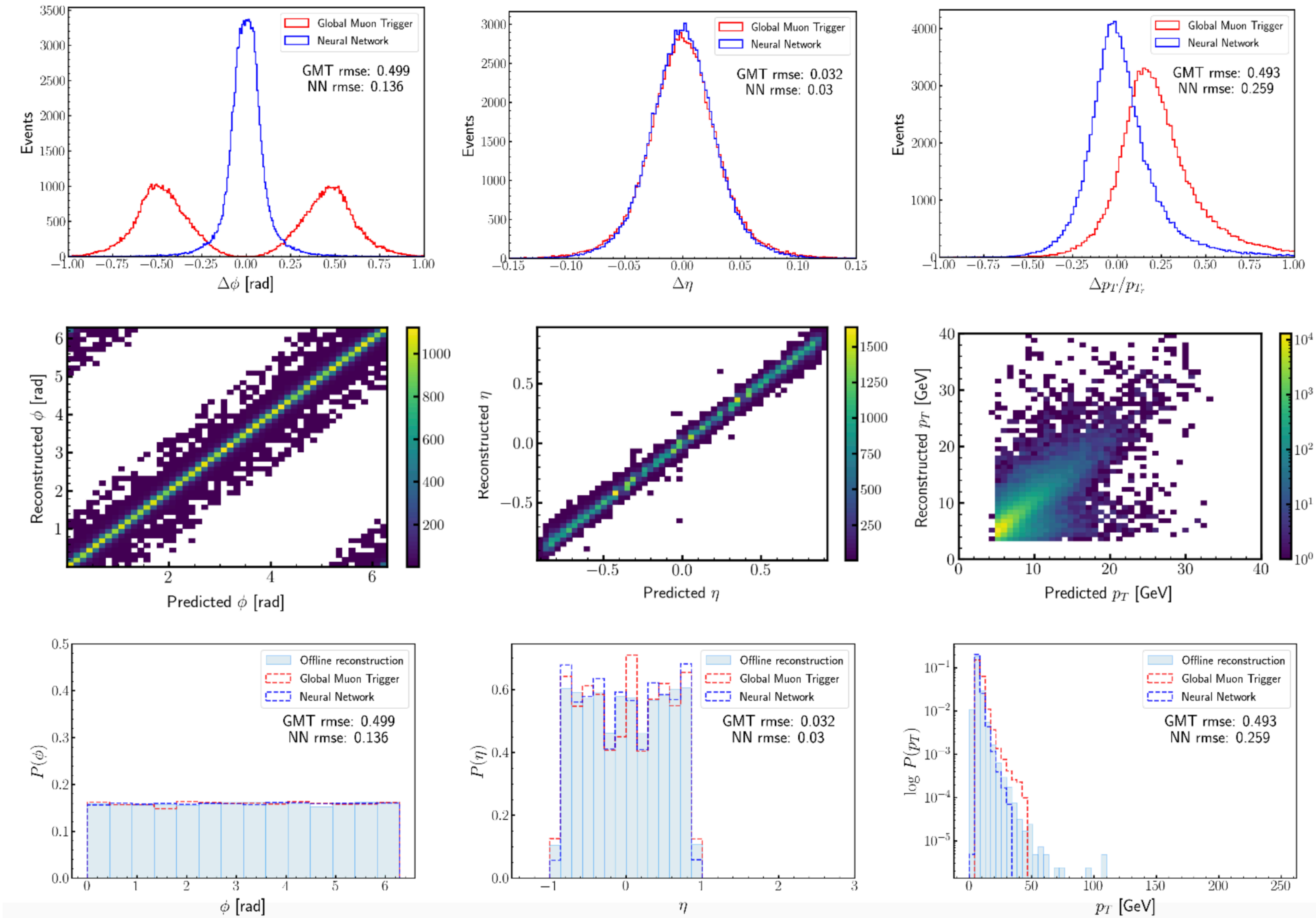
- **L1 Trigger** (hardware: FPGAs)
  - O(μs) hard latency. Typically coarse selection, BDT used for muon p_T assignment
- **HLT** (software: CPUs)
  - O(100 ms) soft latency. More complex algorithms (full detector information available), some BDTs and DNNs used
- **Offline** (software: CPUs)
  - > 1 s latencies. Full event reconstruction, bulk of machine learning usage in CMS

hls4ml

Original slide

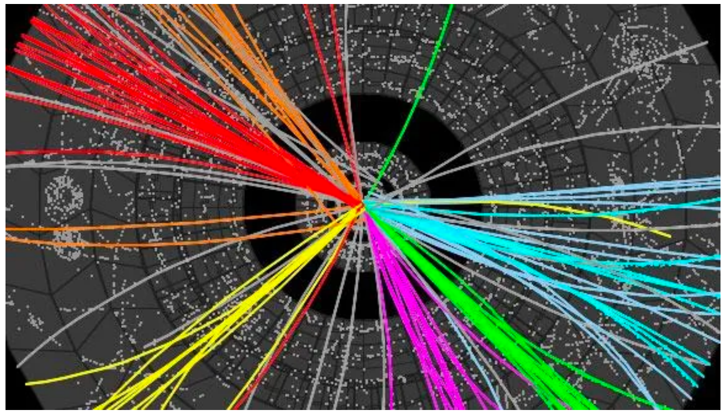
Example of muon trigger

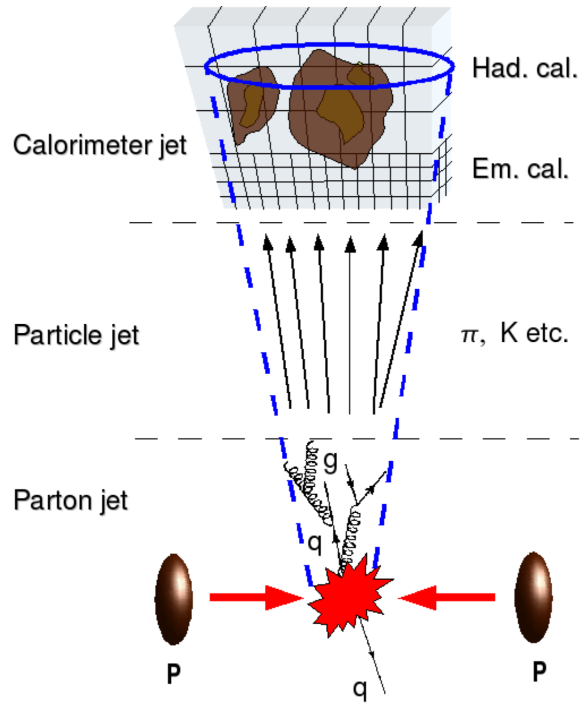
Link

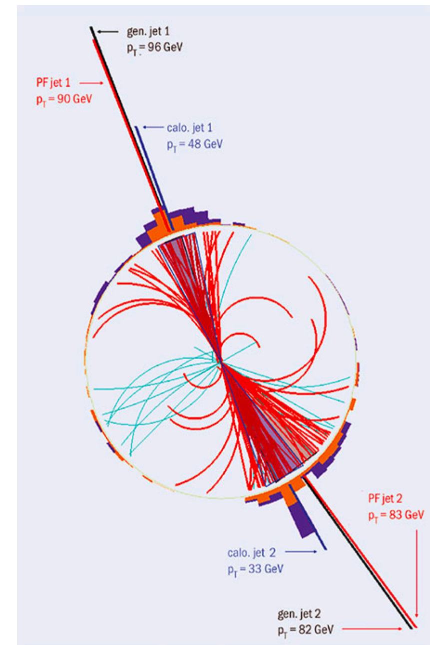


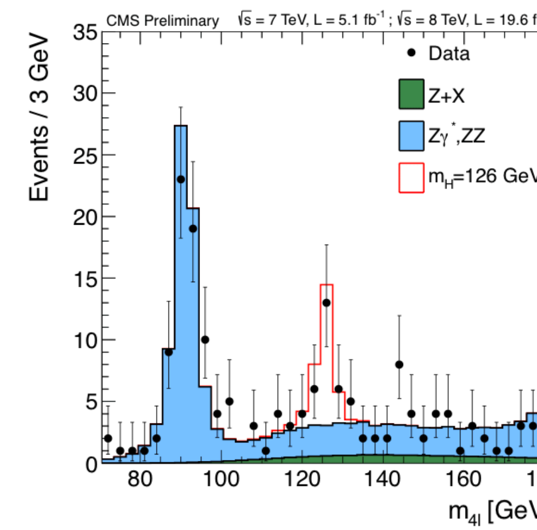
The LHC data flow-chain

ML can play a role at every instance of this flow chain.

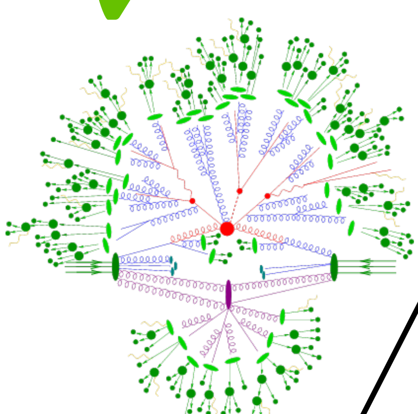


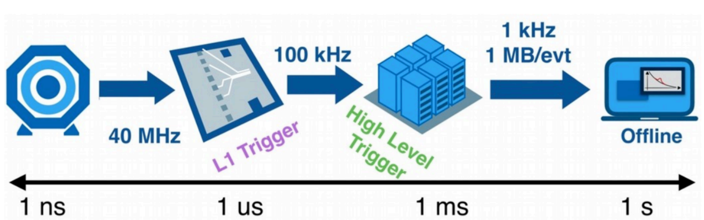


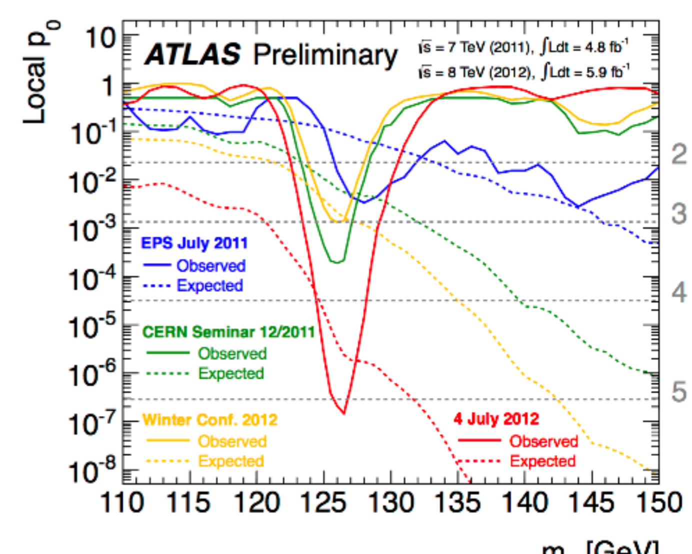




Trigger → Object rec → Event rec → Analysis

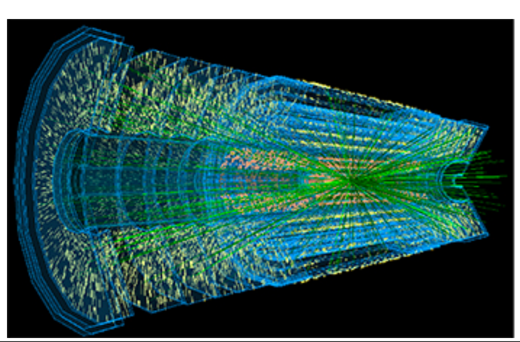




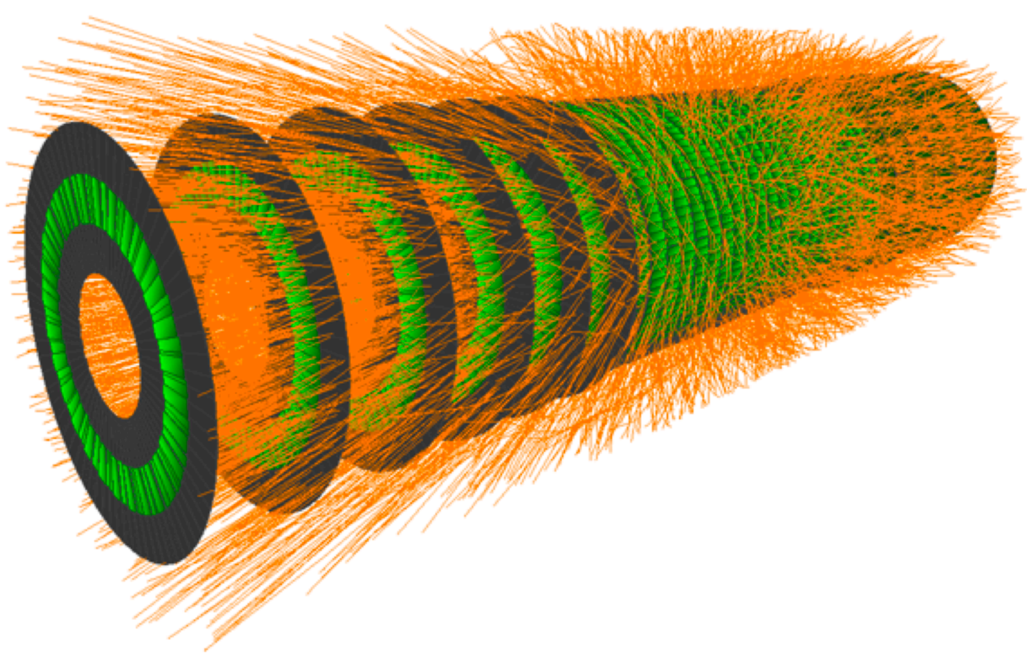


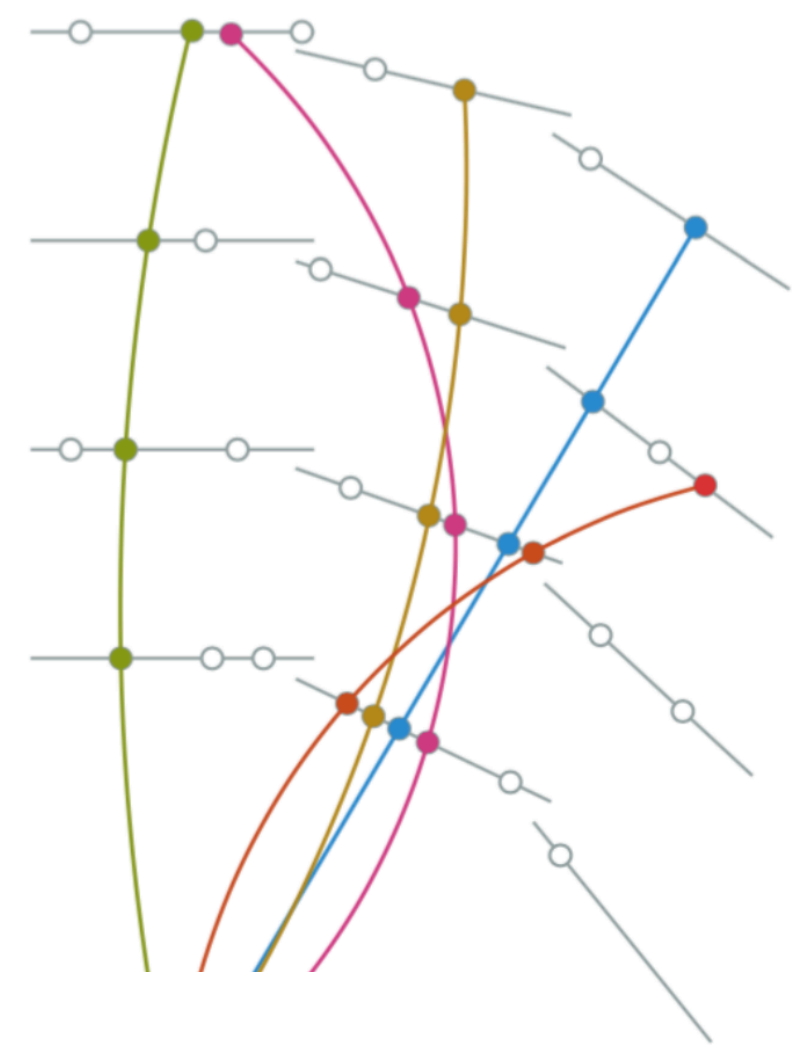
Inference

ME + PS generation
Detector Simulation

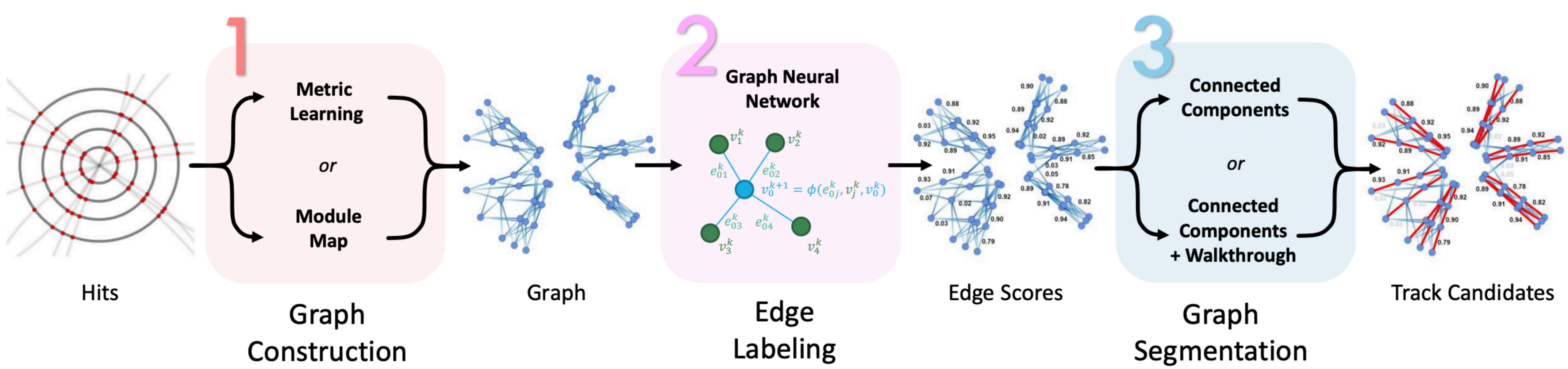


Tracking & ML

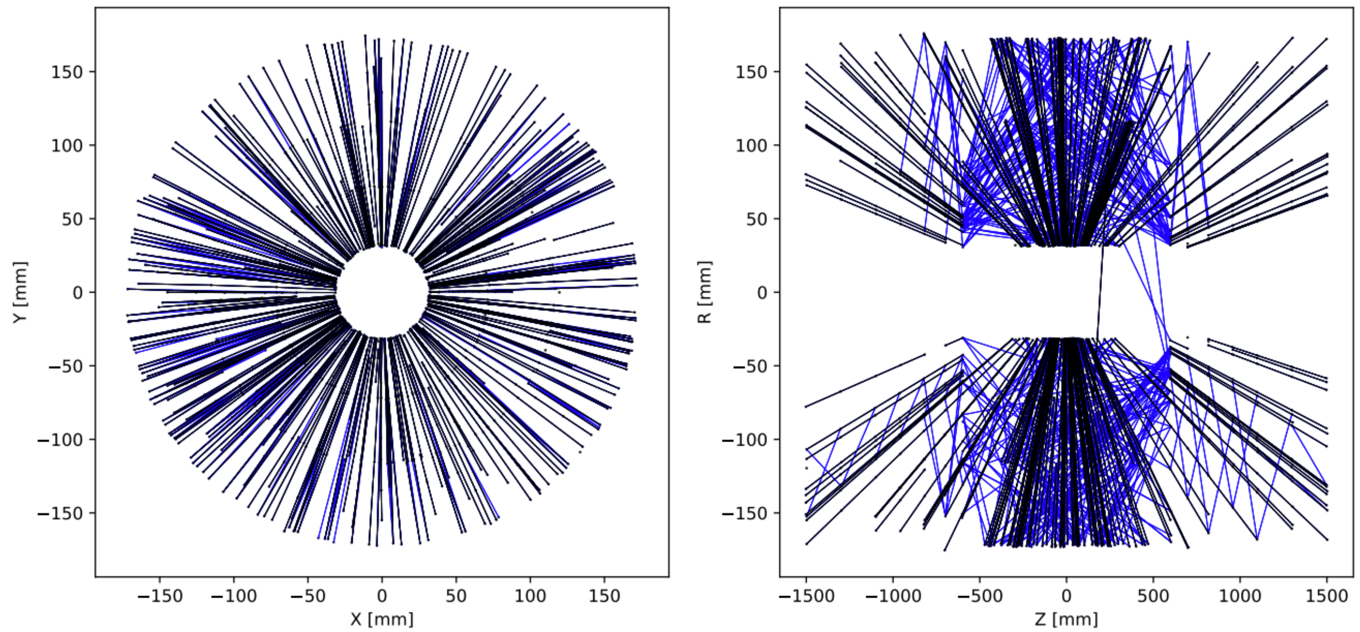


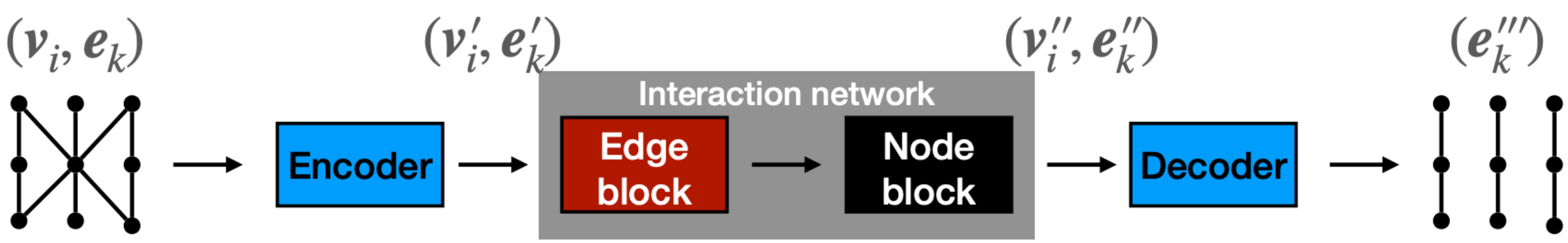


An exponentially large edge finding problem



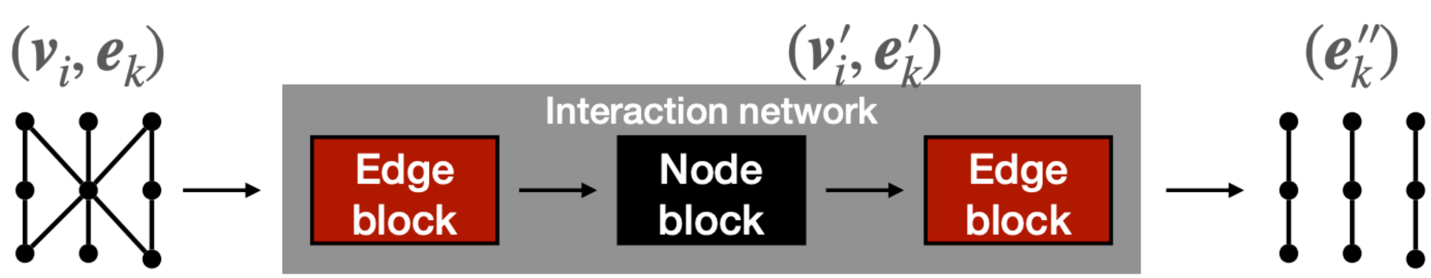
Tracking & ML





$e'_k = \phi_1^e(e_k)$ $\quad e''_k = \phi_2^e(e'_k, v'_{r_k}, v'_{s_k})$ $\quad v''_i = \phi_2^v(\bar{e}''_i, v'_i)$ $\quad e'''_k = \phi_3^e(e''_k)$

$v'_i = \phi_1^v(v_i)$ $\quad \bar{e}''_i = \rho^{e\to v}(E'_i)$

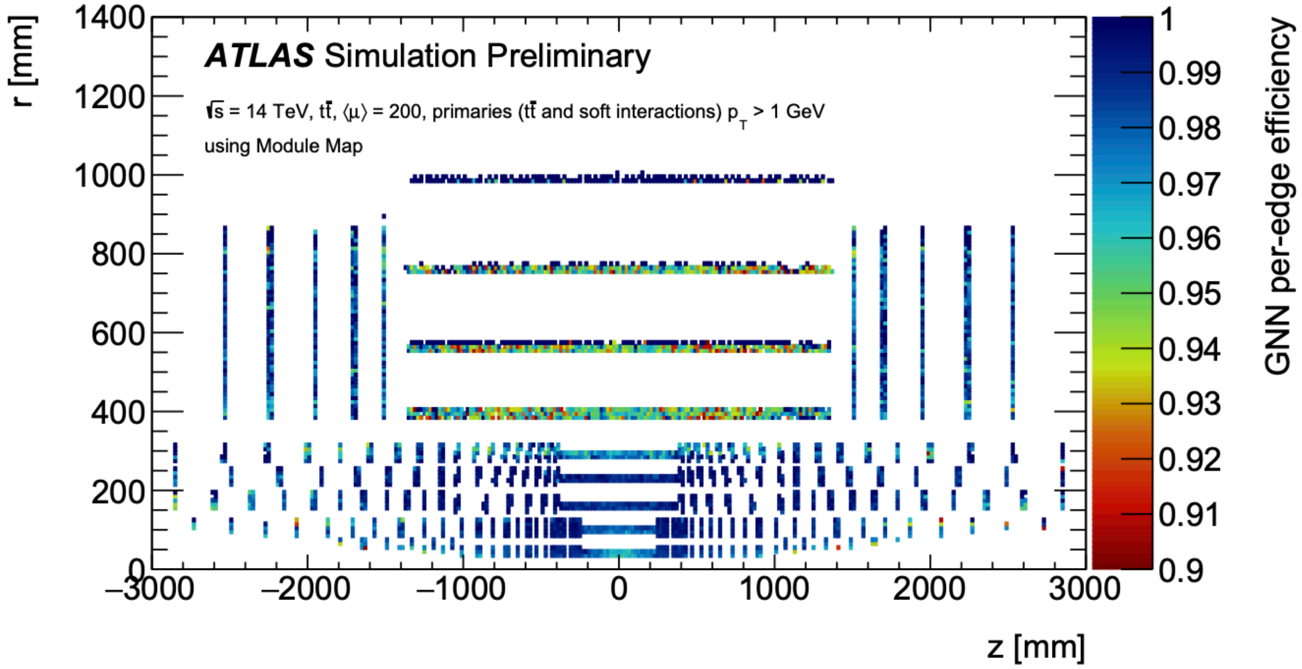


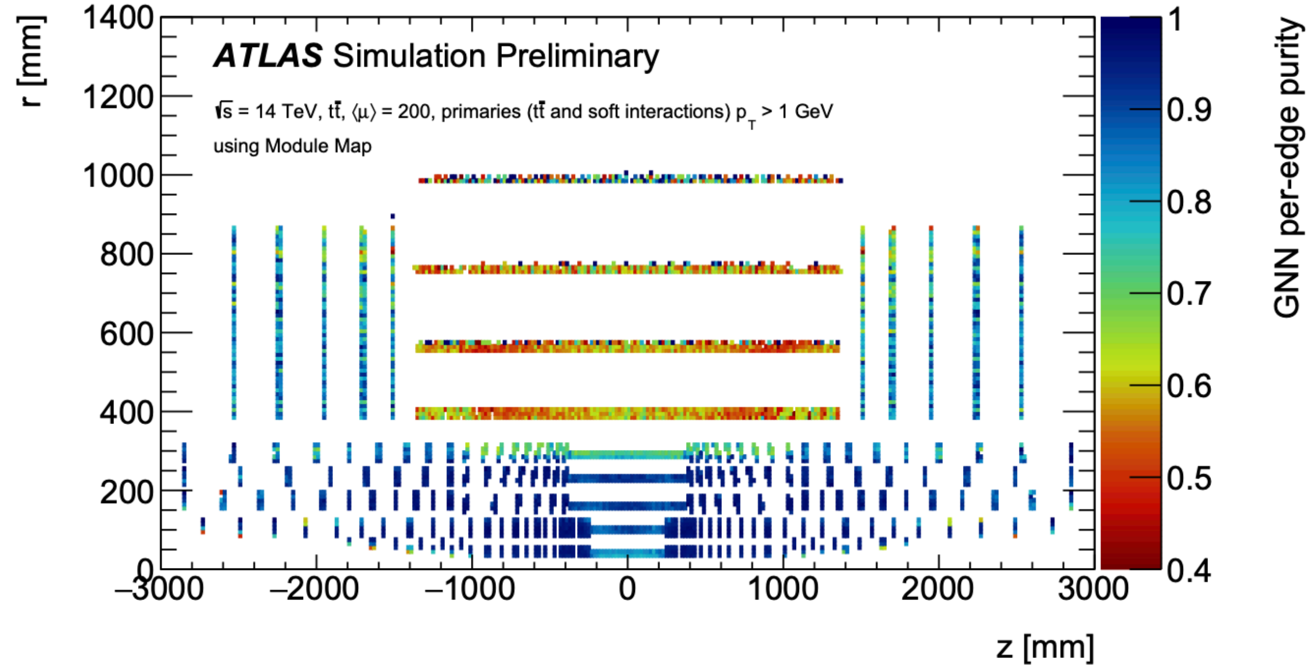
$e'_k = \phi_2^e(e_k, v_{r_k}, v_{s_k})$ $\quad v'_i = \phi_2^v(\bar{e}'_i, v_i)$ $\quad e''_k = \phi_2^e(e'_k, v'_{r_k}, v'_{s_k})$

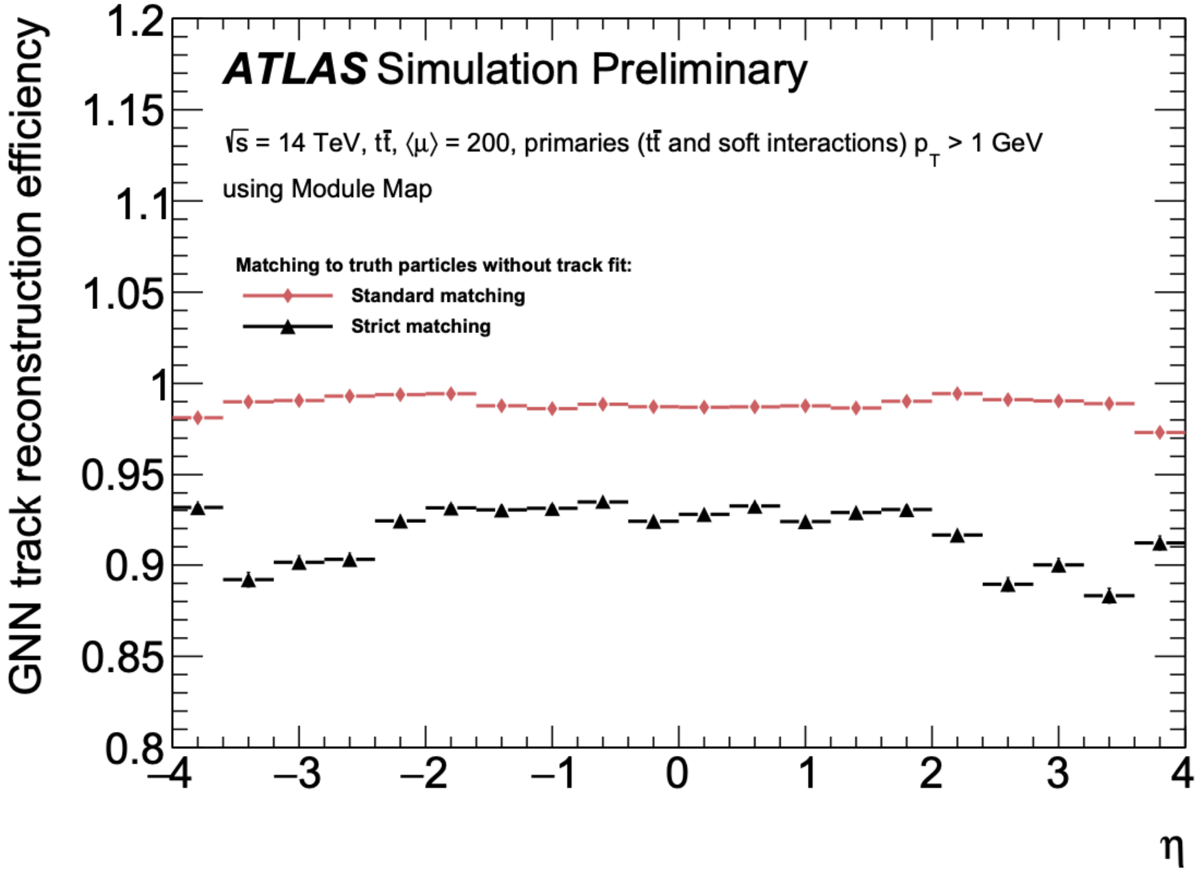
$\bar{e}'_i = \rho^{e\to v}(E_i)$

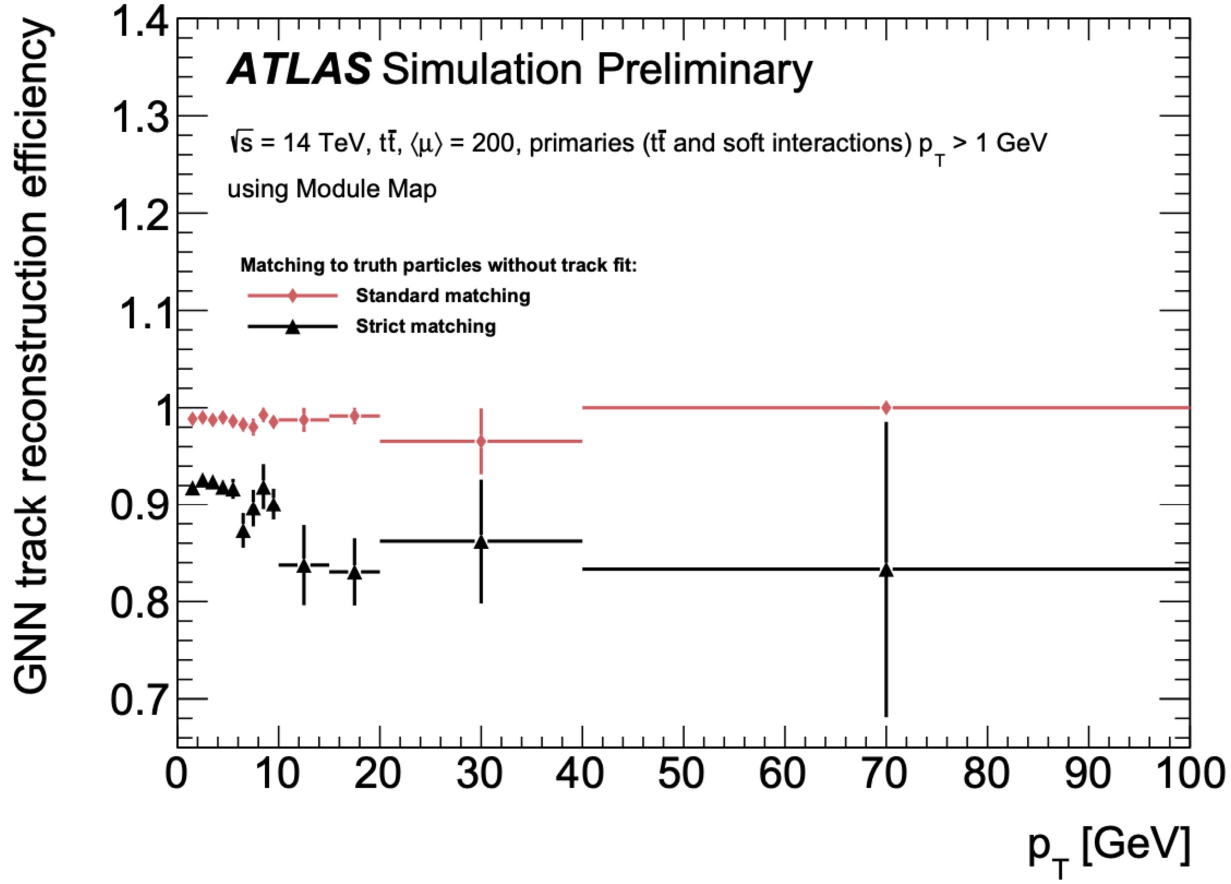
Tracking & ML

ATL-ITK-PROC-2022-006

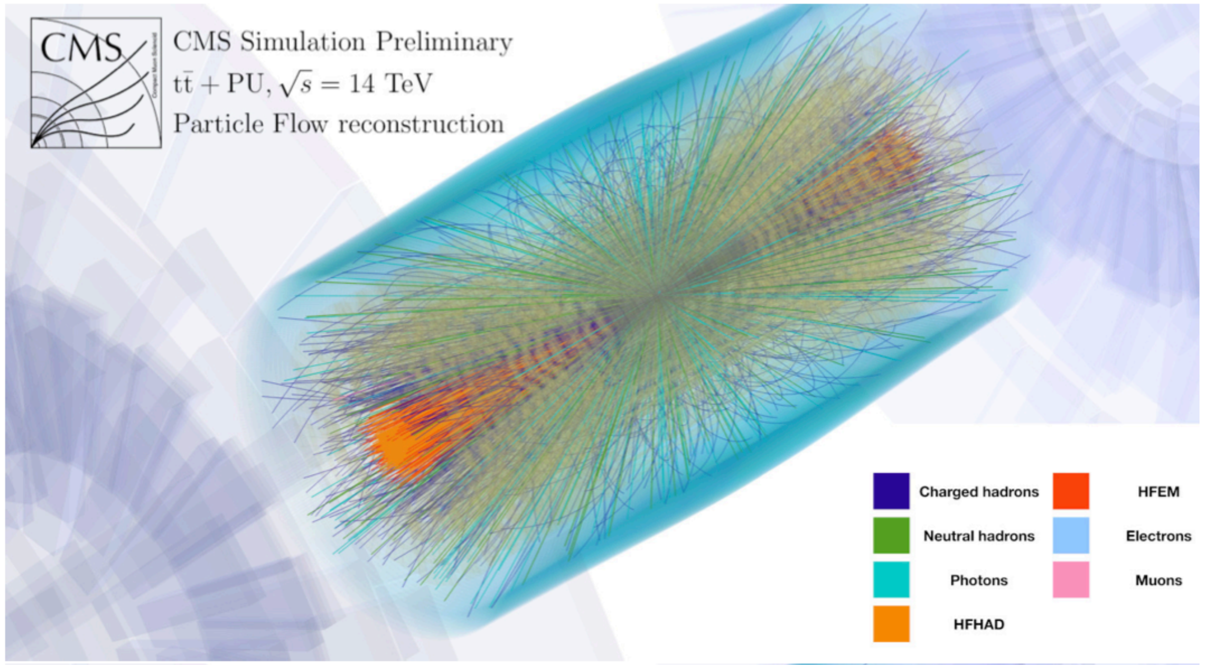


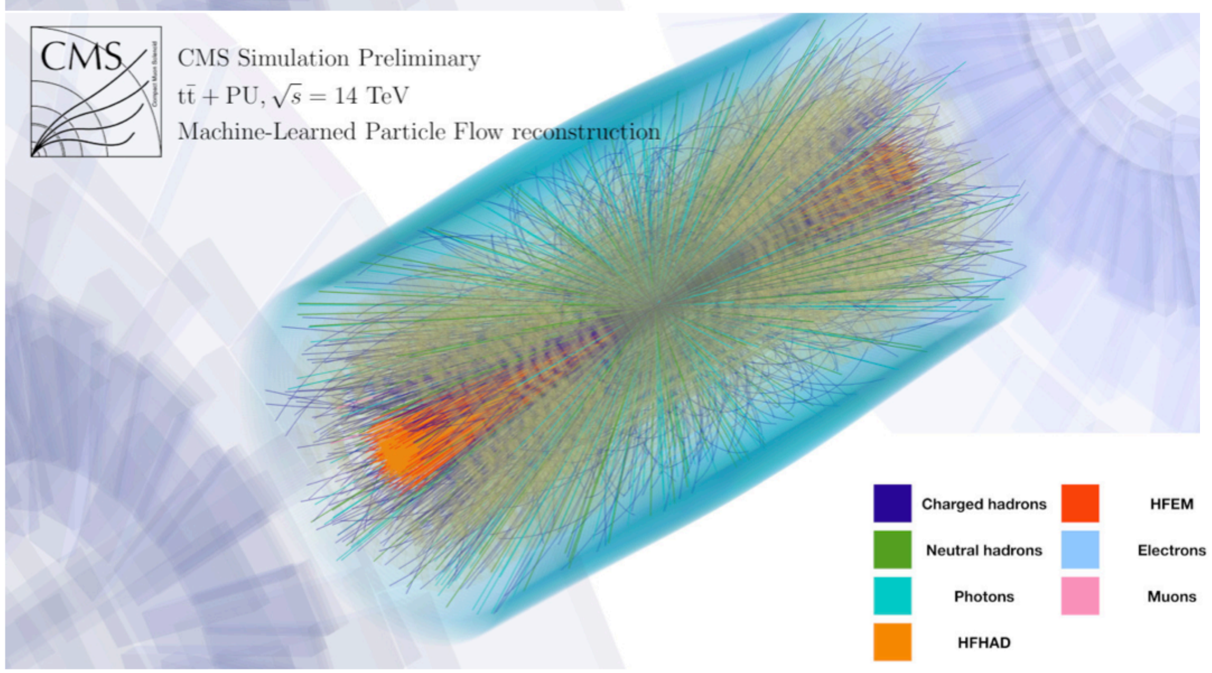


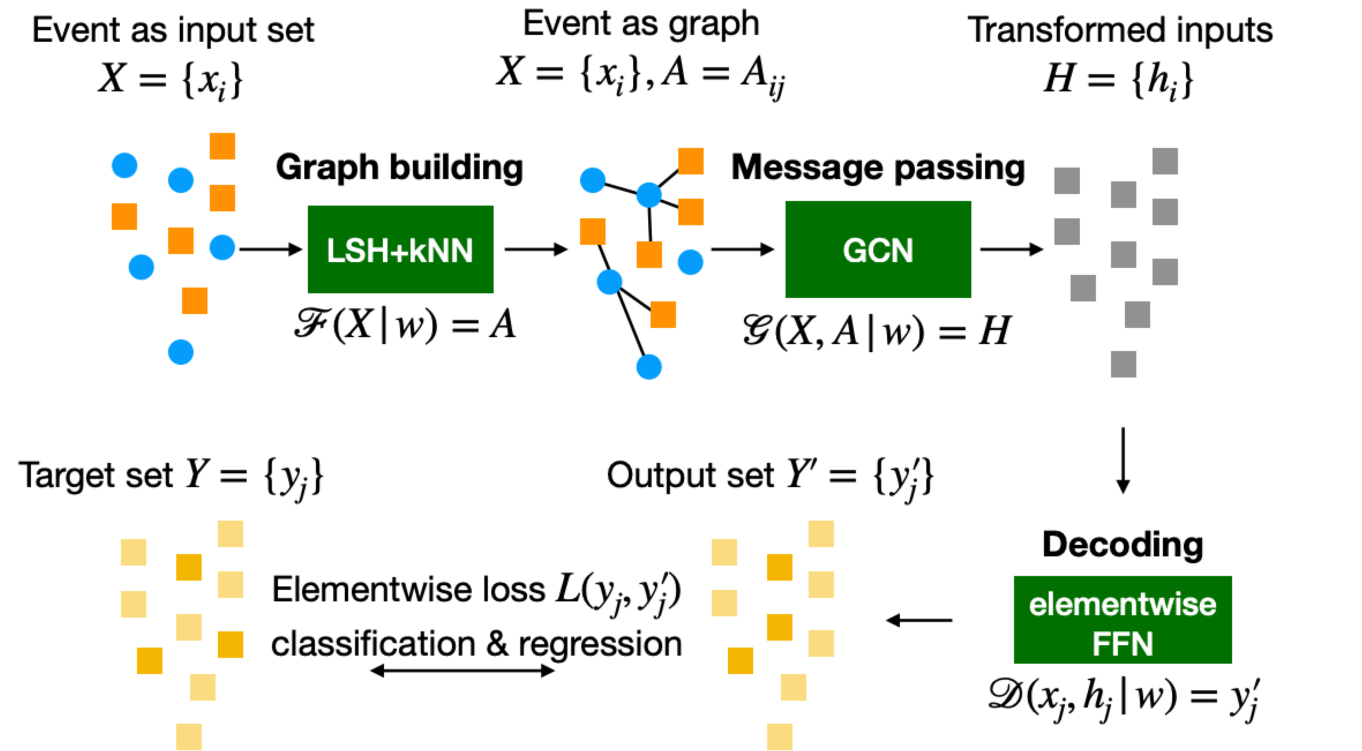




Full ML driven PFlow : MLPF







$x_i = [\text{type}, p_T, E_{ECAL}, E_{HCAL}, \eta, \phi, \eta_{outer}, \phi_{outer}, q, \ldots]$, type $\in$ {track, cluster}

$y_j = [\text{PID}, p_T, E, \eta, \phi, q, \ldots]$, PID $\in$ {none, charged hadron, neutral hadron, $\gamma, e^{\pm}, \mu^{\pm}$}

$h_i \in \mathbb{R}^{256}$

Trainable neural networks: $\mathscr{F}, \mathscr{G}, \mathscr{D}$

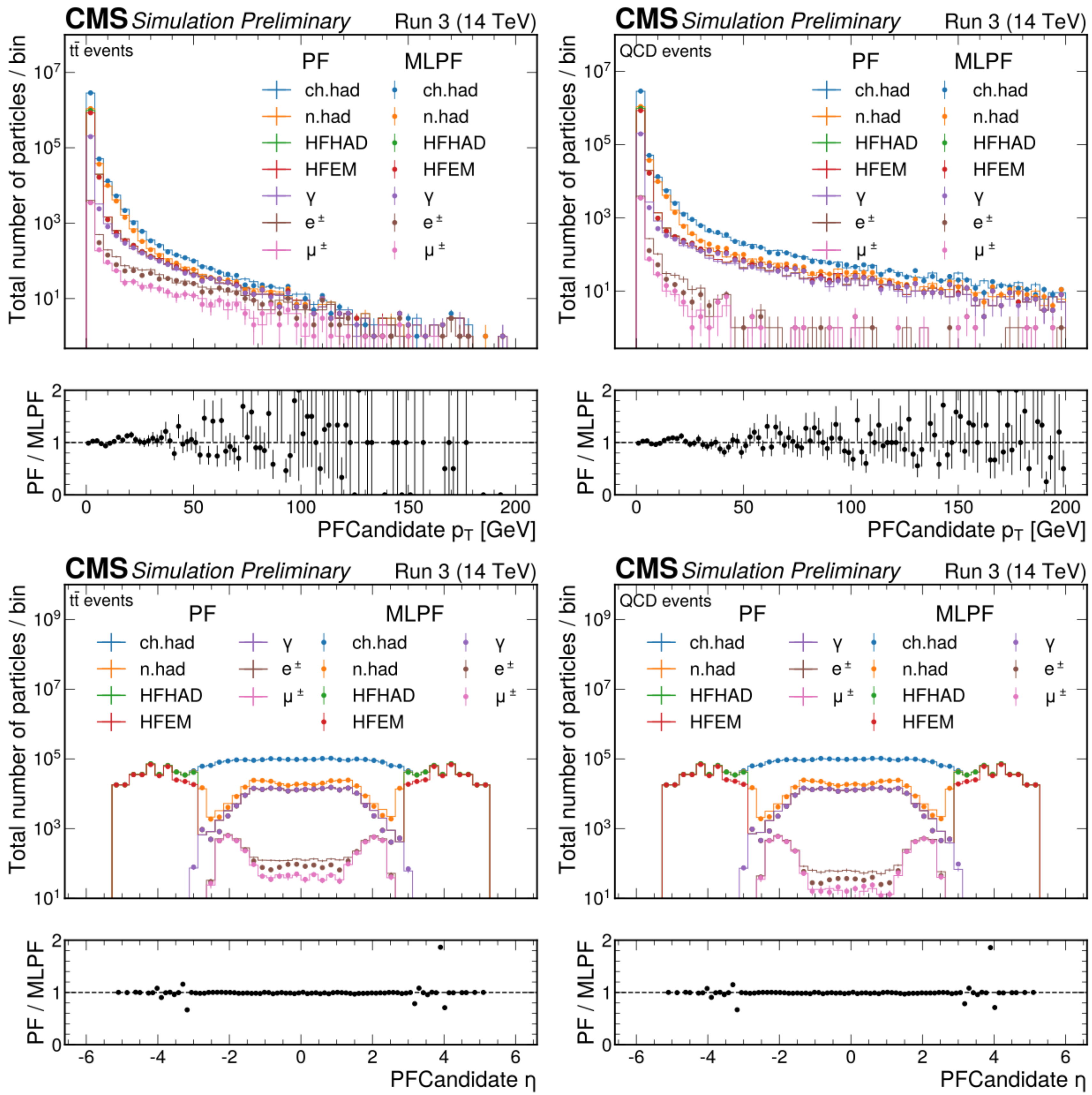
● - track, ■ - calorimeter cluster, ■ - encoded element

■ - target (predicted) particle, ■ - no target (predicted) particle

MLPF
Eur. Phys. J. C (2021) 81: 381
J. Pata et. al.

PF lepton, hadron, photon = F_{PF} (track hits + calo cells)

Combining track + calo for PFlow



MLPF

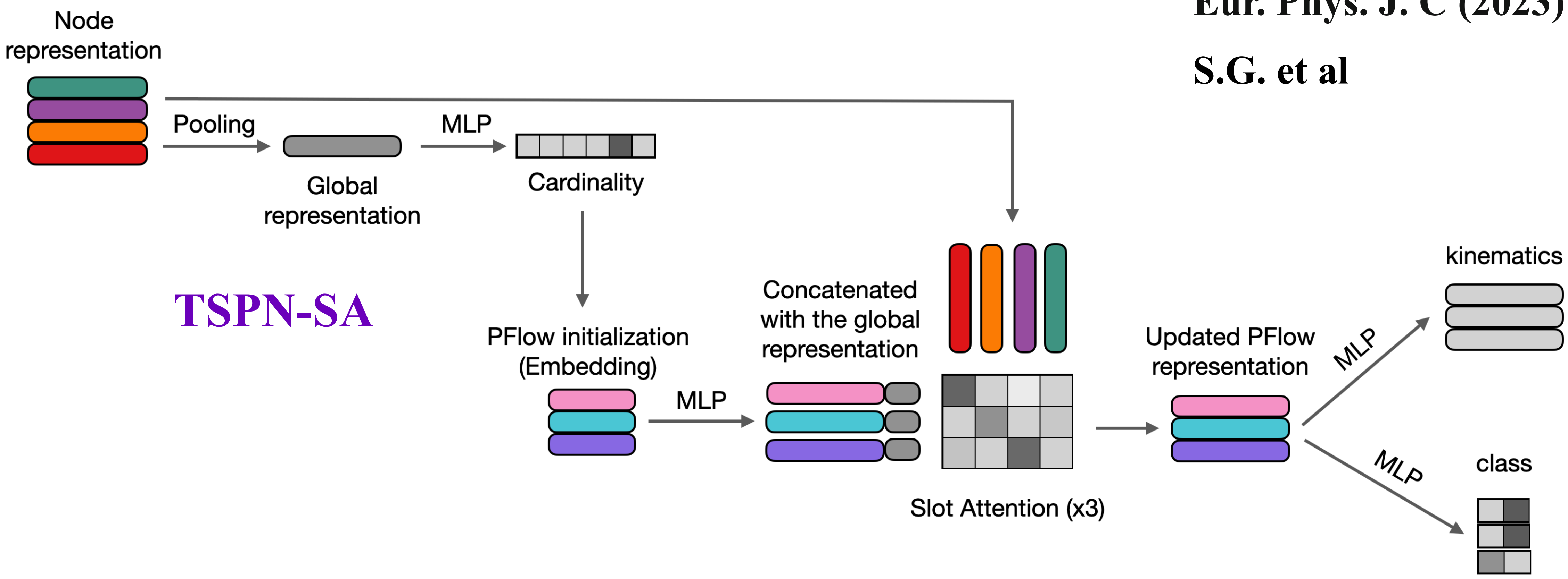
J. Phys.: Conf. Ser. 2438, 012100 (2023)

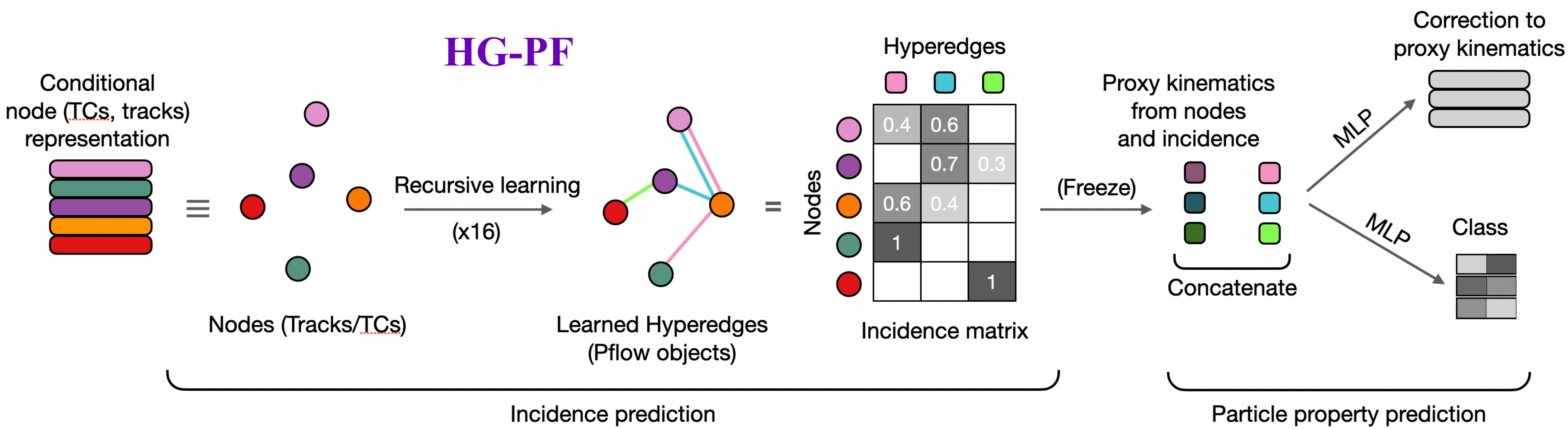
J. Pata et. al.

Attempt for higher order correlation

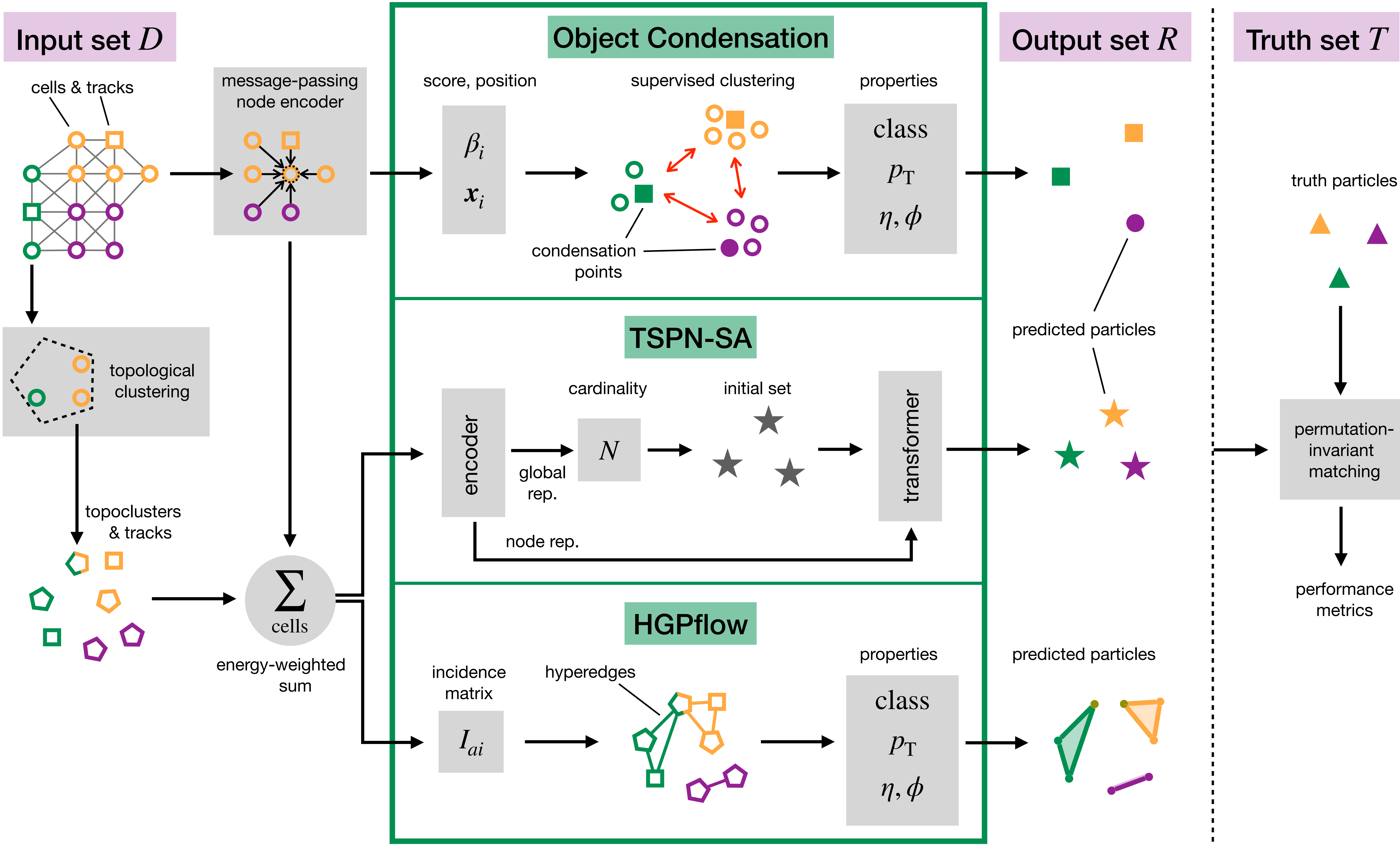
Eur. Phys. J. C (2023) 83:596

S.G. et al

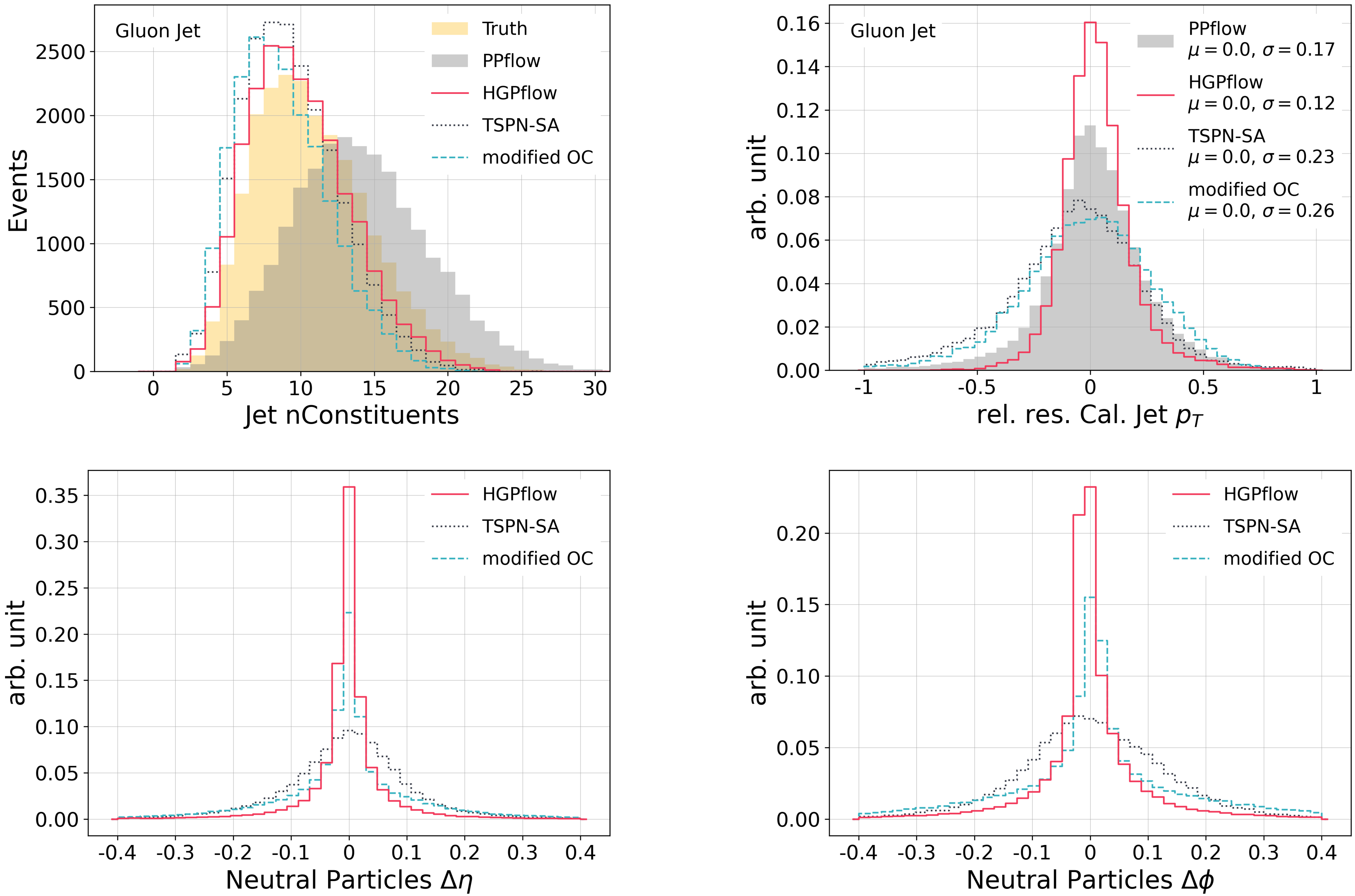




The network flow comparisons

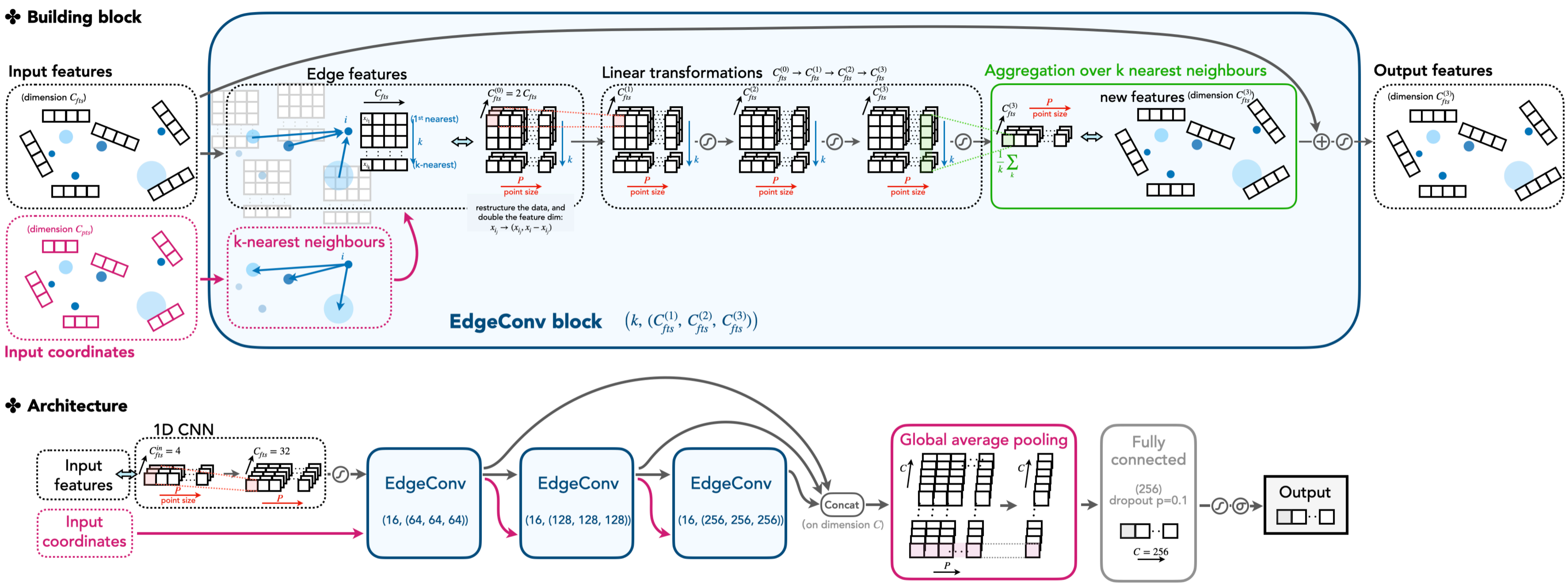


Comparison of the three networks



Object tagging

Particle Net : 1902.08570



arXiv > cs > arXiv:1801.07829

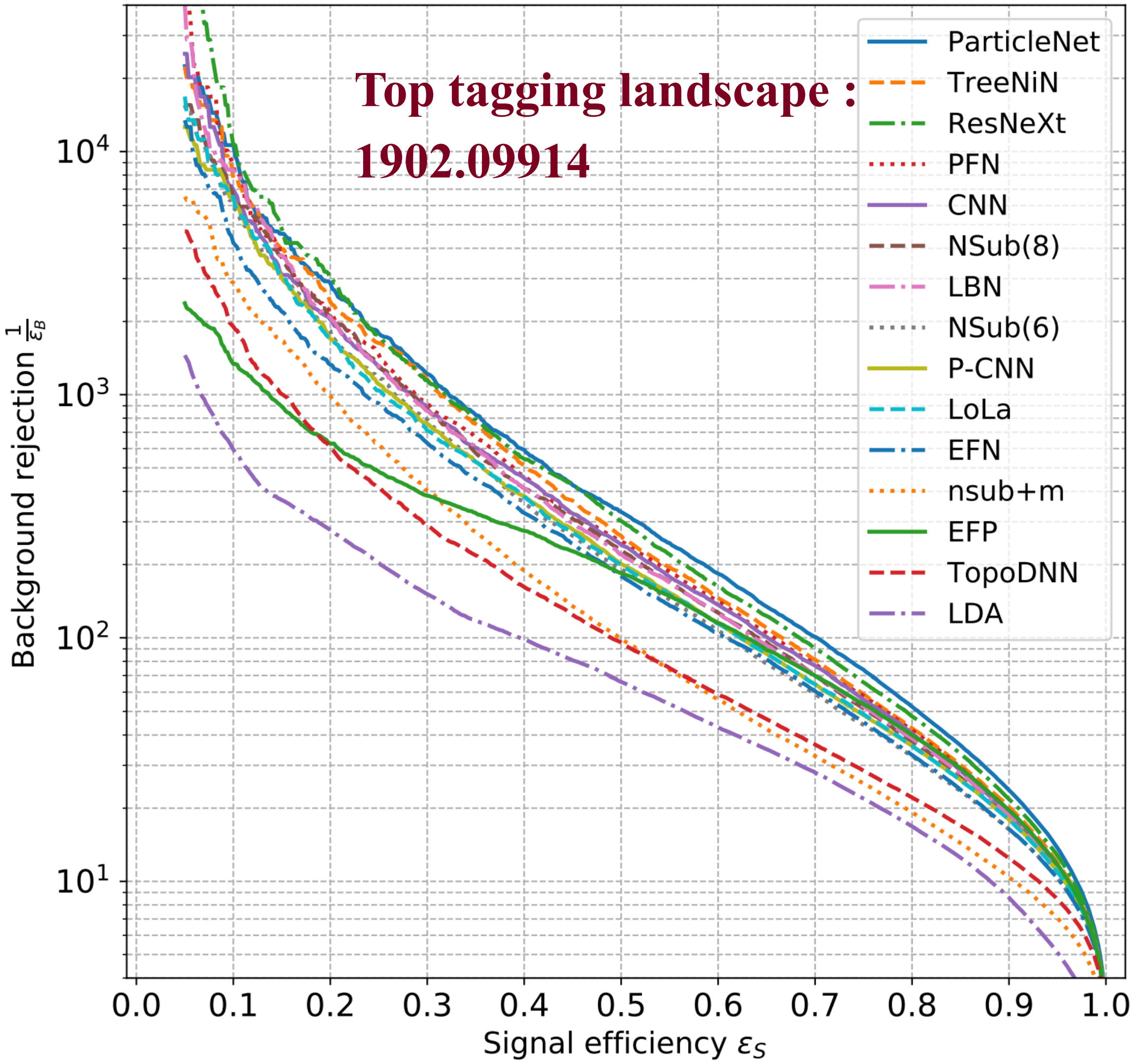
Computer Science > Computer Vision and Pattern Recognition

[Submitted on 24 Jan 2018 (v1), last revised 11 Jun 2019 (this version, v2)]

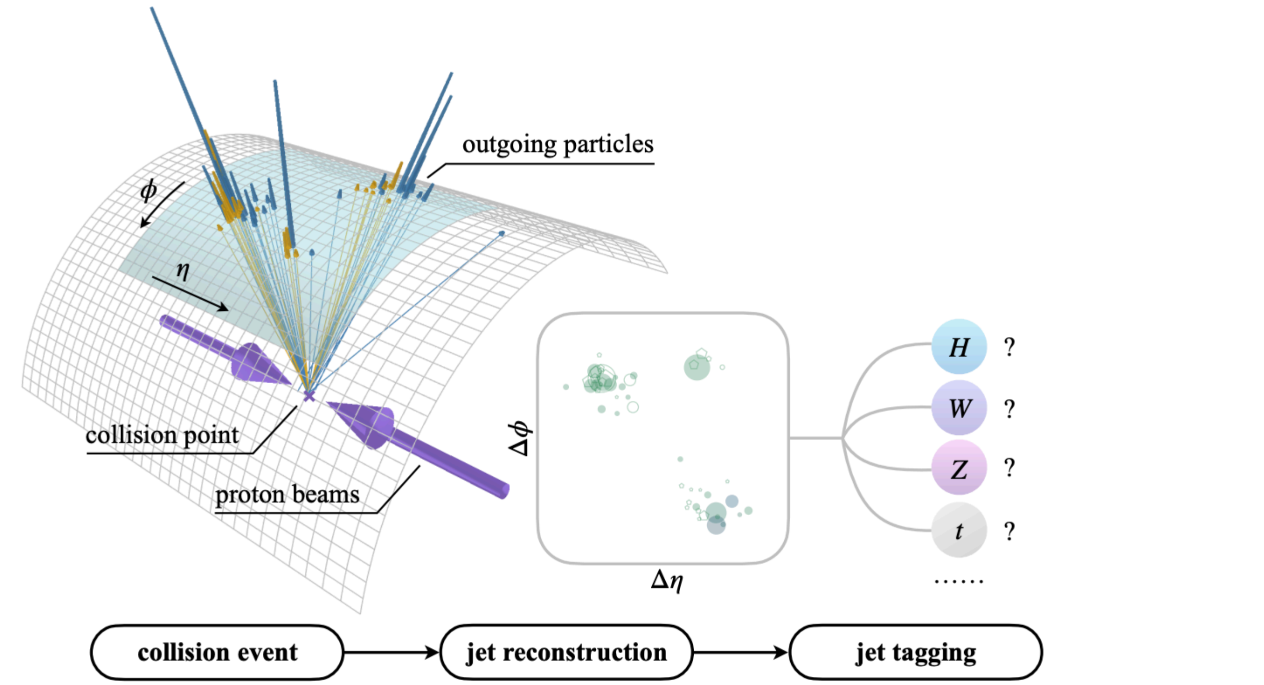
Dynamic Graph CNN for Learning on Point Clouds

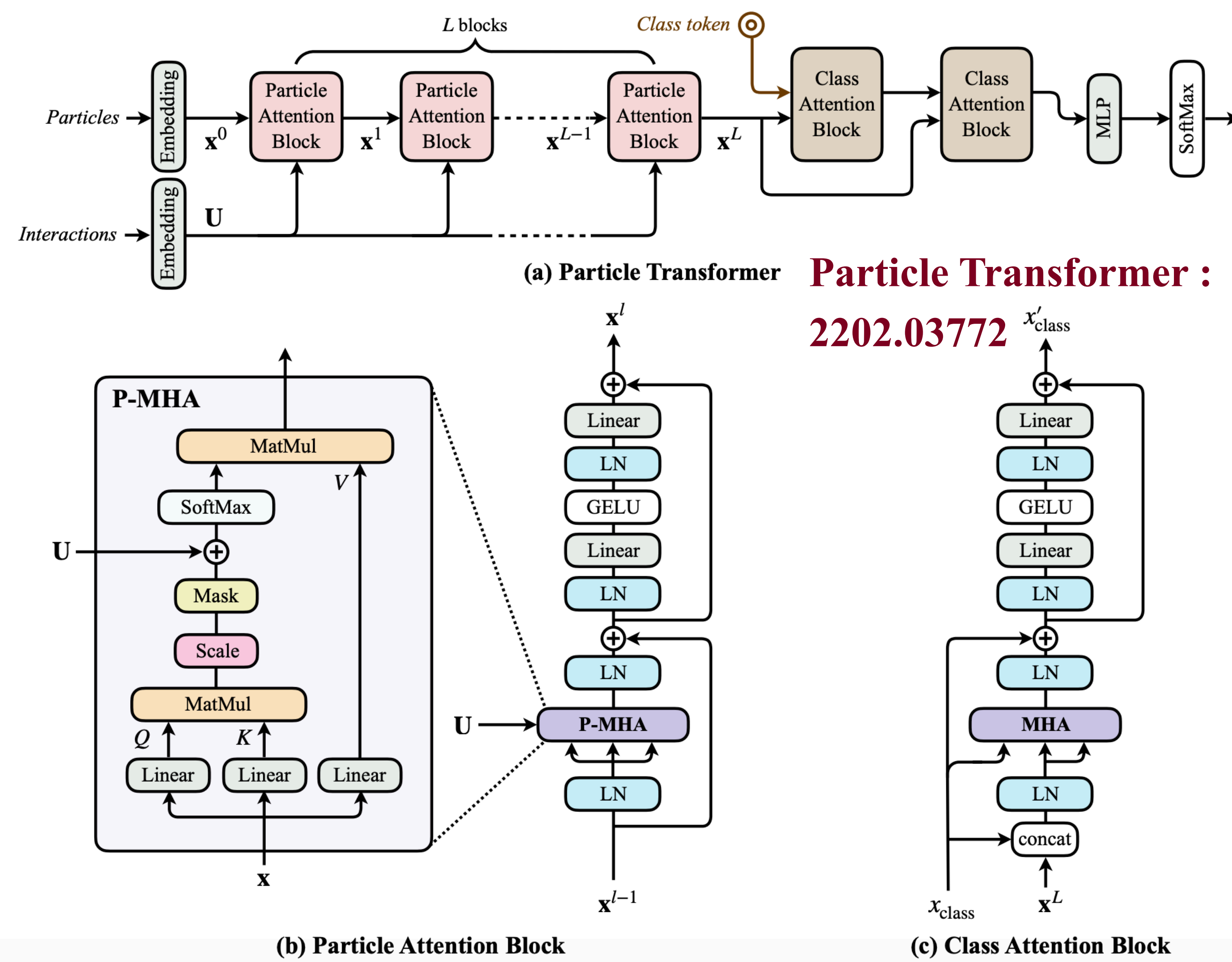
Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, Justin M. Solomon

Object tagging



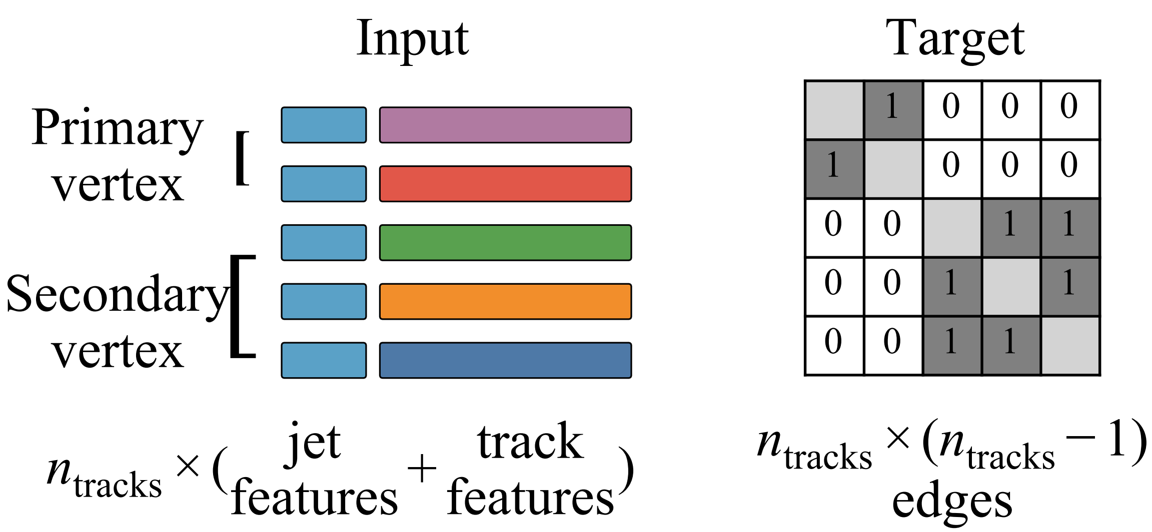
	Accuracy	AUC	$\text{Rej}_{50\%}$	$\text{Rej}_{30\%}$
P-CNN	0.930	0.9803	201 ± 4	759 ± 24
PFN	—	0.9819	247 ± 3	888 ± 17
ParticleNet	0.940	0.9858	397 ± 7	1615 ± 93
JEDI-net (w/ $\sum O$)	0.930	0.9807	—	774.6
PCT	0.940	0.9855	392 ± 7	1533 ± 101
LGN	0.929	0.964	—	435 ± 95
rPCN	—	0.9845	364 ± 9	1642 ± 93
LorentzNet	0.942	0.9868	498 ± 18	2195 ± 173
ParT	0.940	0.9858	413 ± 16	1602 ± 81
ParticleNet-f.t.	0.942	0.9866	487 ± 9	1771 ± 80
ParT-f.t.	**0.944**	**0.9877**	**691 ± 15**	**2766 ± 130**

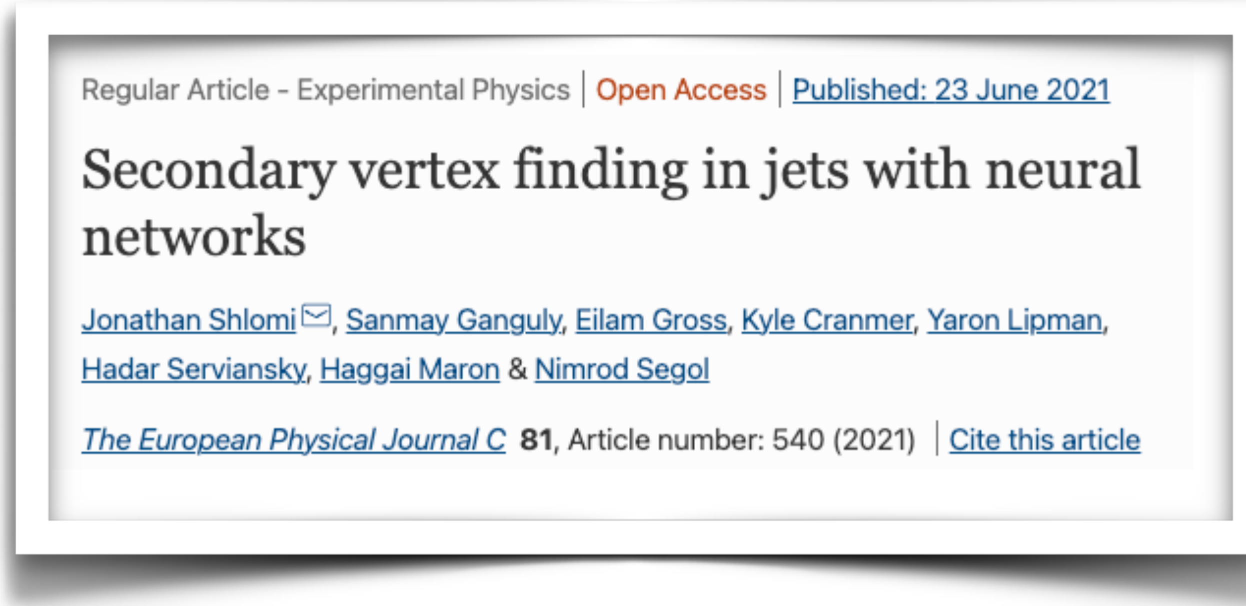




Particle Transformer : 2202.03772

Set2Graph proposal for flavor-tagging



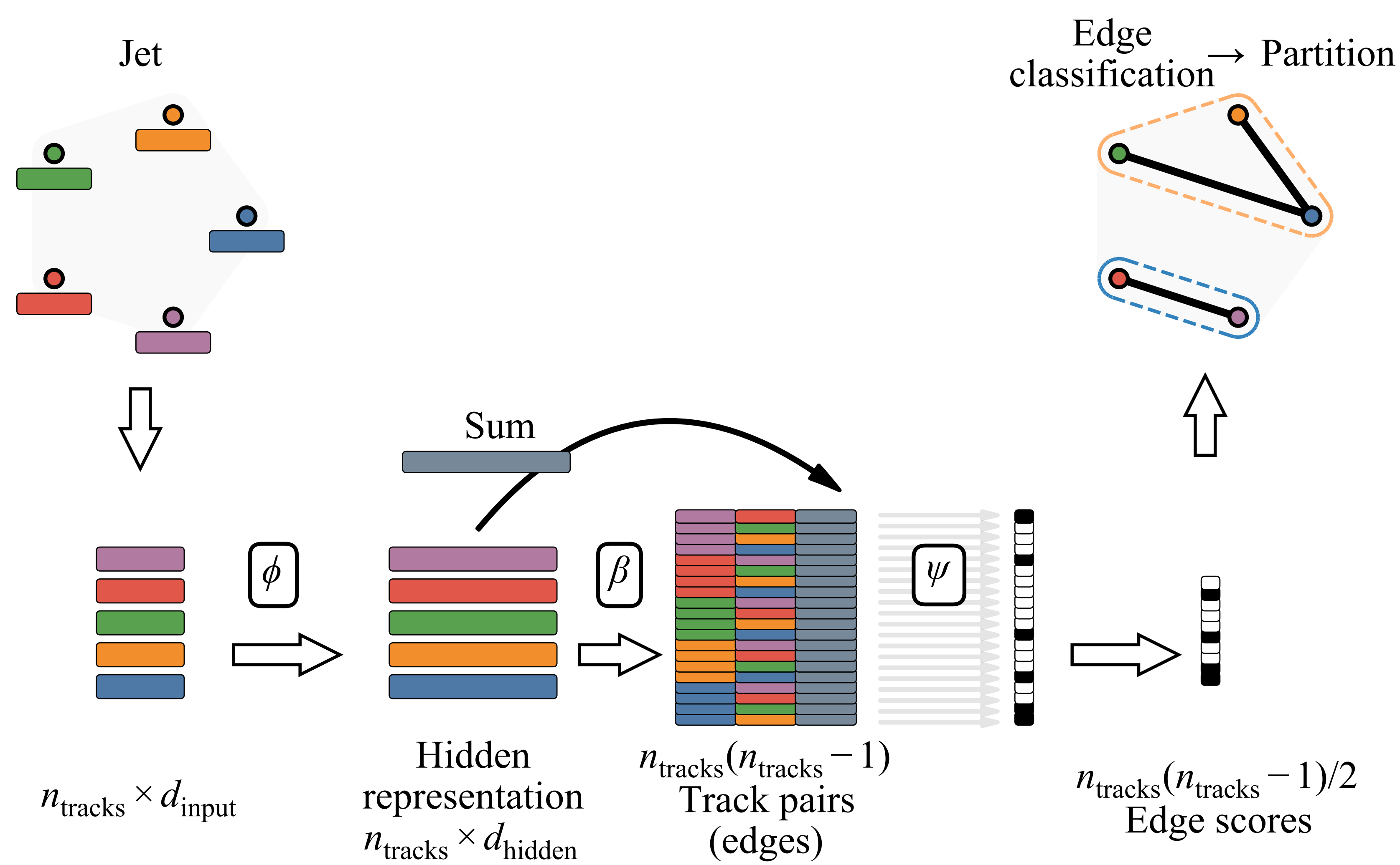


Regular Article - Experimental Physics | Open Access | Published: 23 June 2021

Secondary vertex finding in jets with neural networks

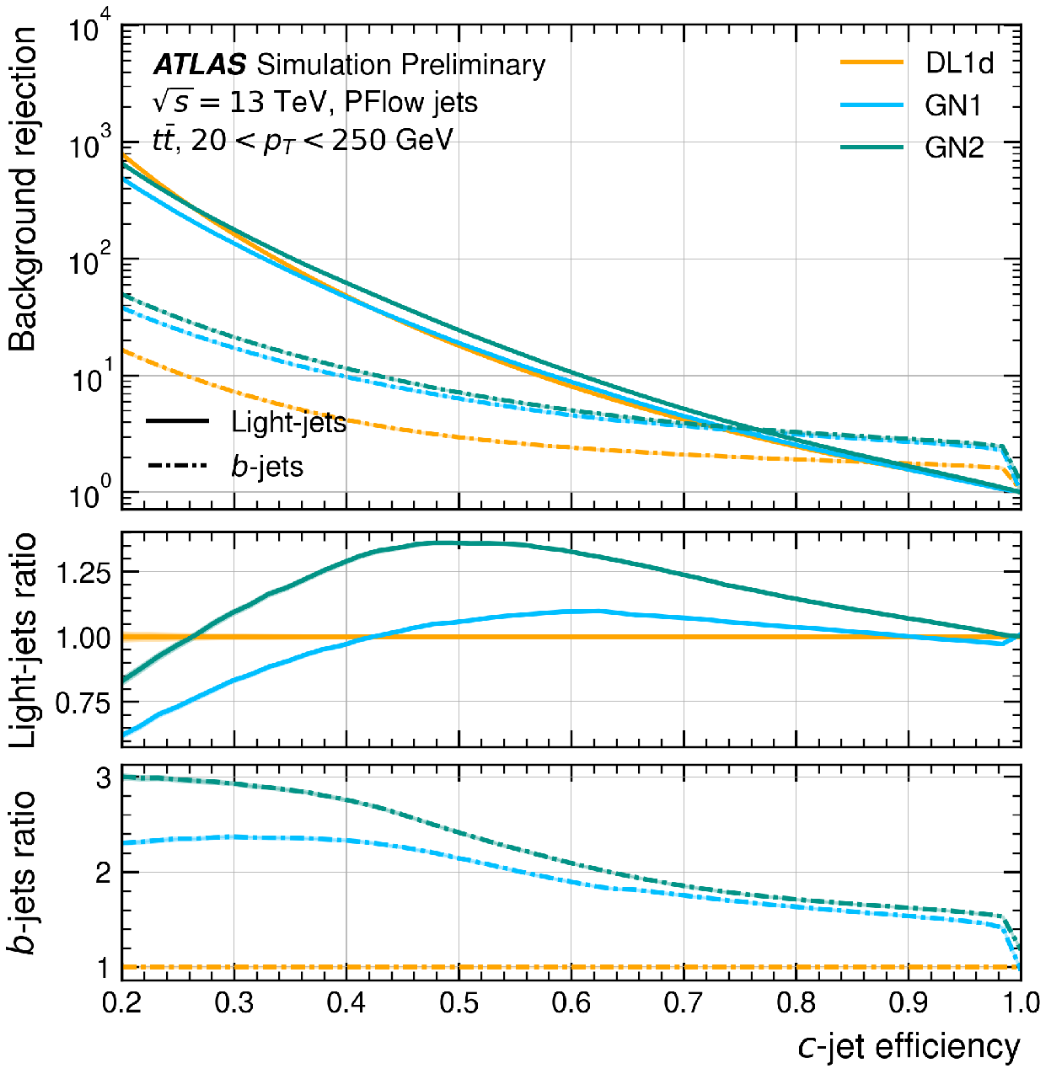
Jonathan Shlomi, Sanmay Ganguly, Eilam Gross, Kyle Cranmer, Yaron Lipman, Hadar Serviansky, Haggai Maron & Nimrod Segol

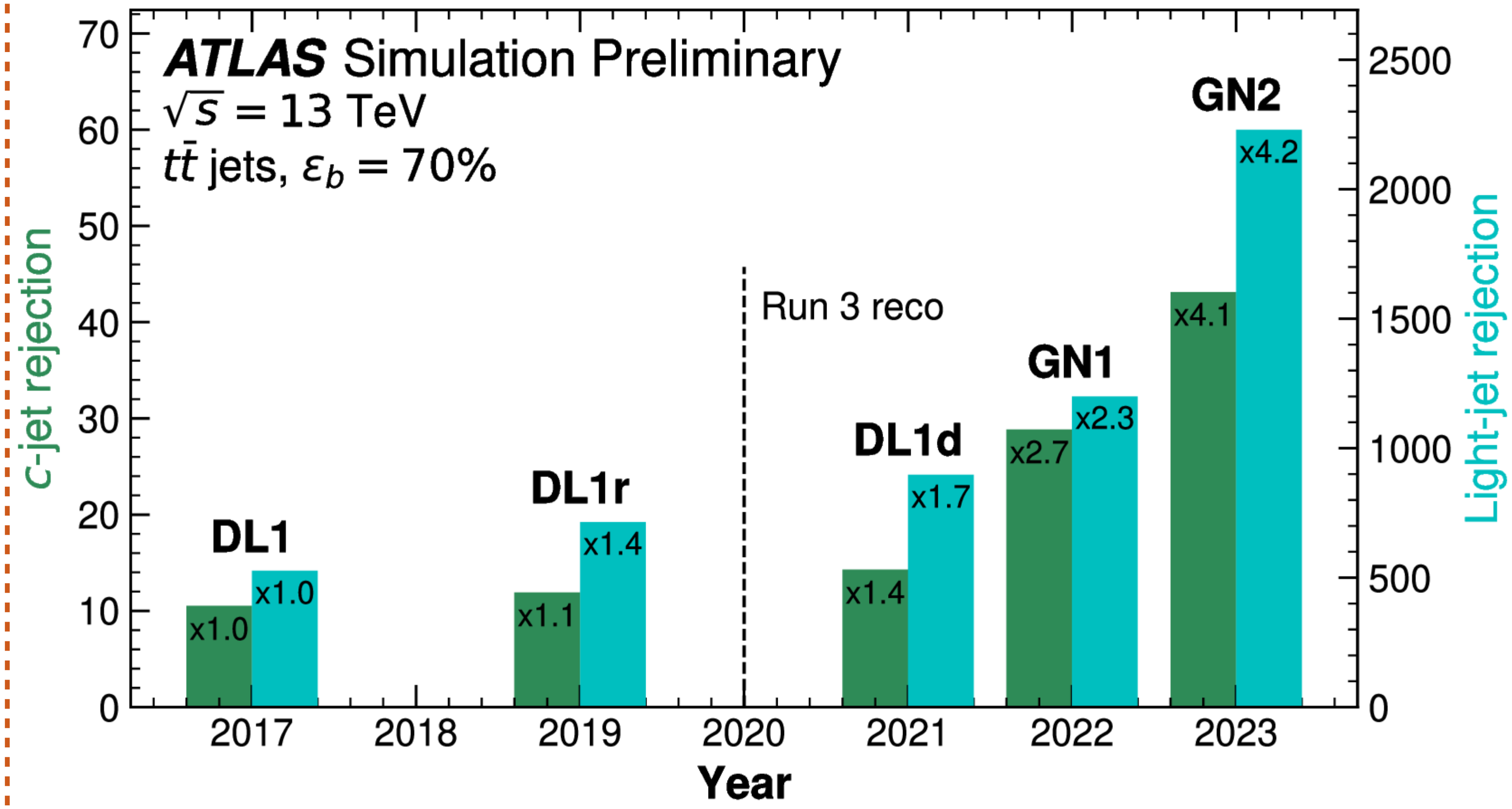
The European Physical Journal C **81**, Article number: 540 (2021) | Cite this article

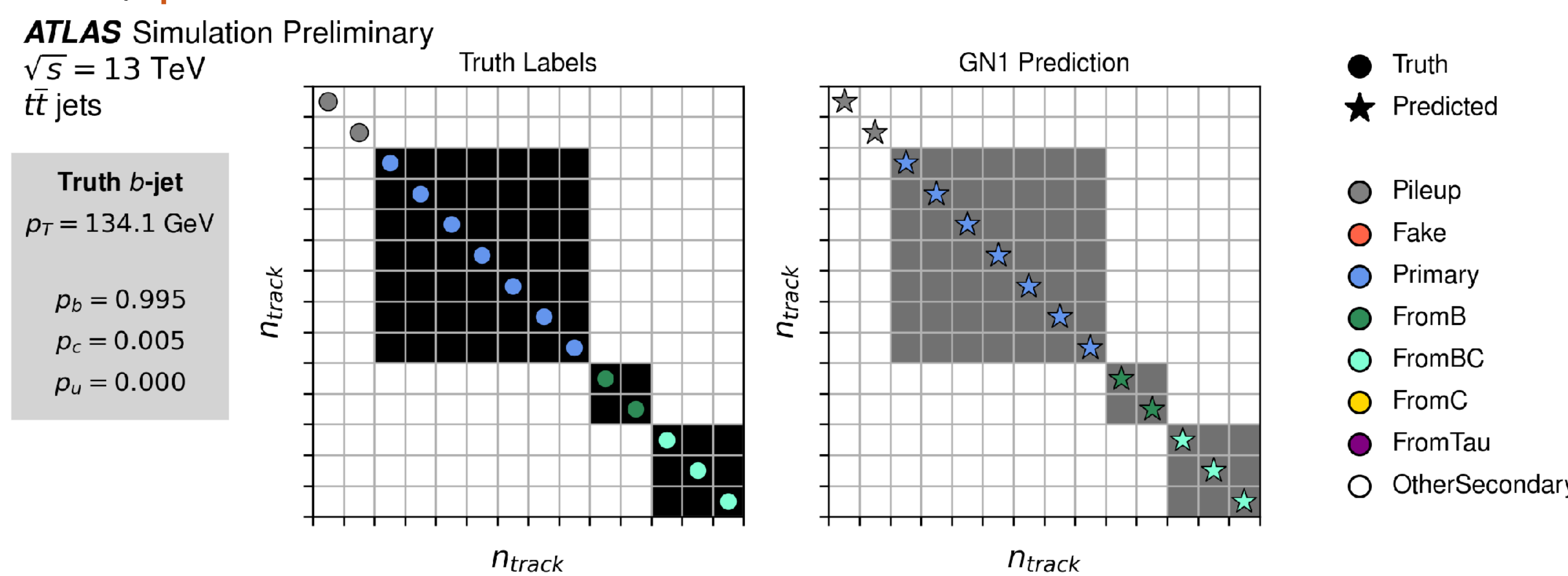


Set2Graph model within ATLAS

FTAG-2023-001





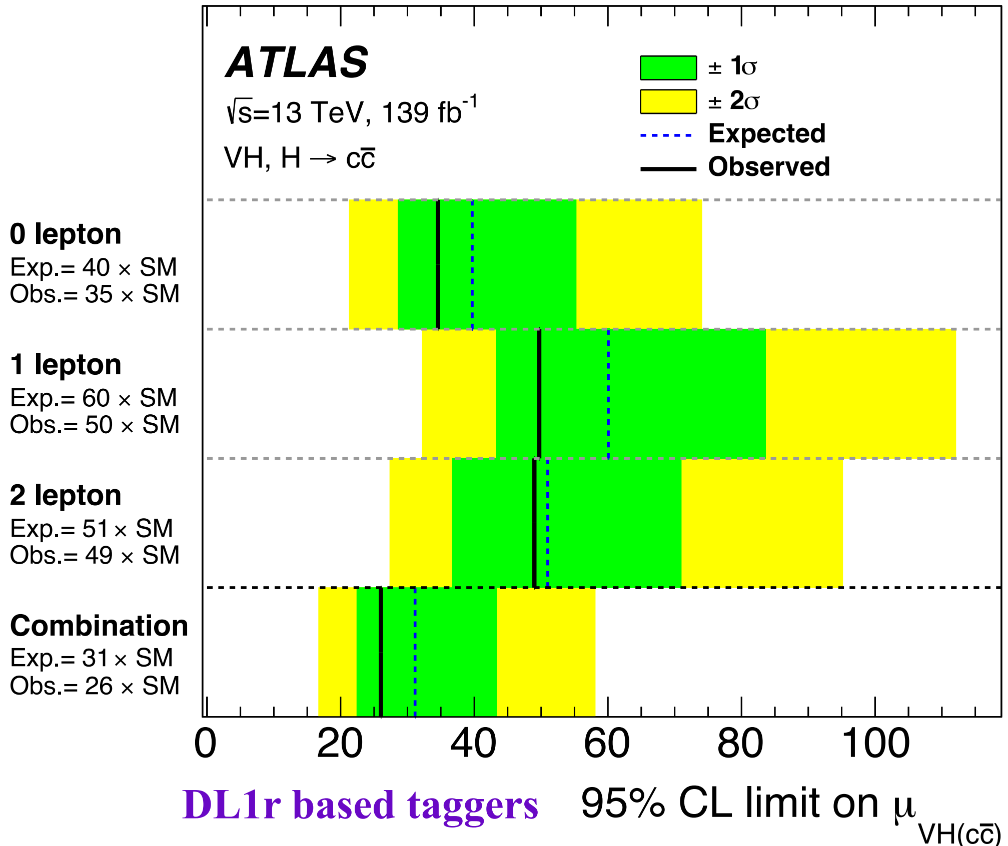


Sizable improvement over the current DL1r algorithm.

For a c-tagging working point ~ 30%, a significant gain in Rejection rate is obtained.

Direct physics application of the taggers

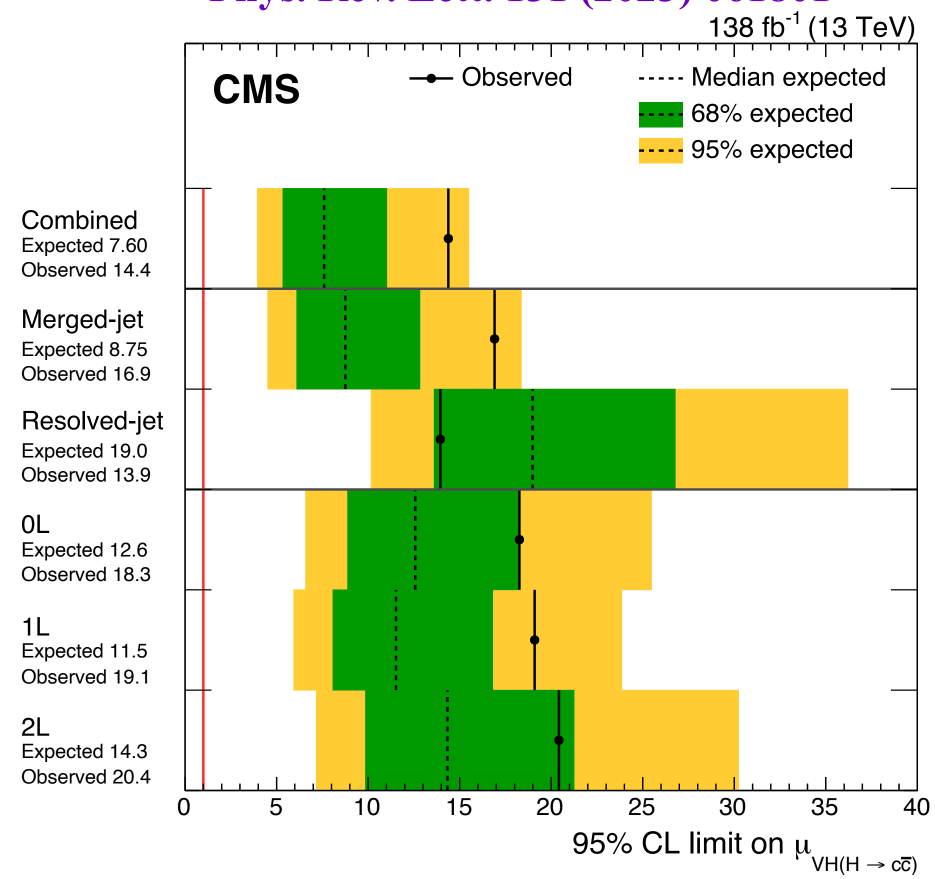
Eur. Phys. J. C (2022) 82:717



DL1r based taggers

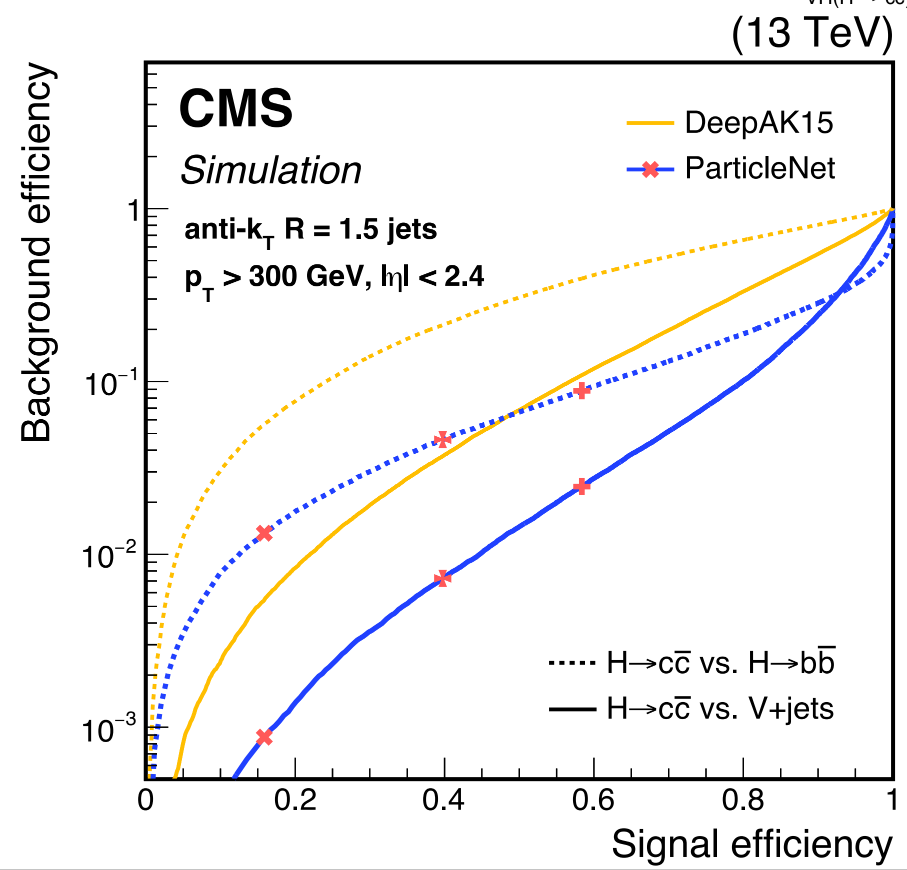
Future iteration will use GN2

Phys. Rev. Lett. 131 (2023) 061801



ATLAS bound : $|\kappa_c| < 8.5$

CMS bound : $1.1 < |\kappa_c| < 5.5$

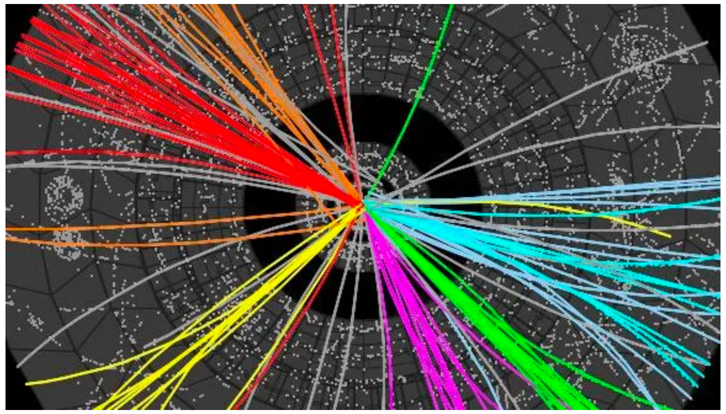


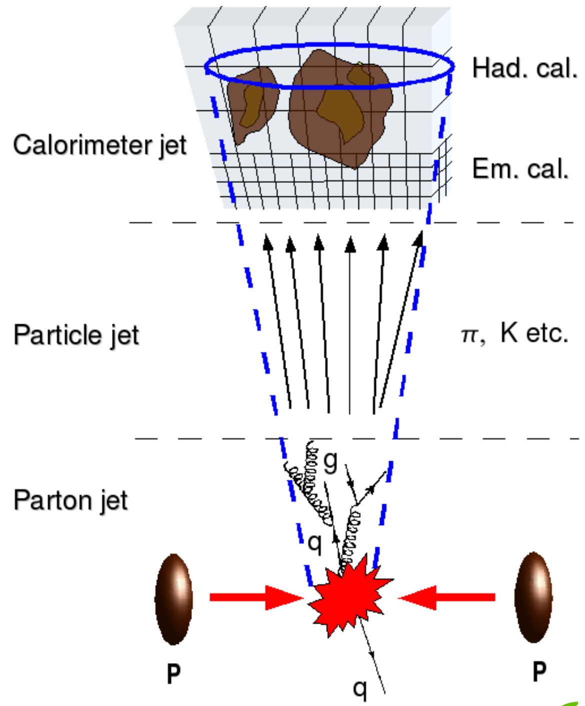
Future direction of tagger improvement:

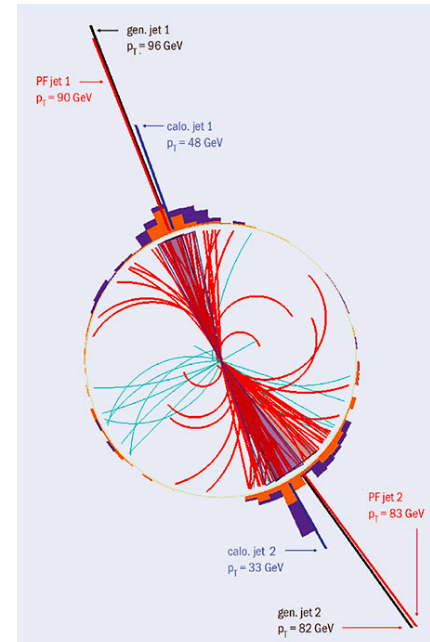
1. Explainable taggers on heterogeneous pc
2. A systematic uncertainty extraction.
3. How much universal taggers can be made across topologies?

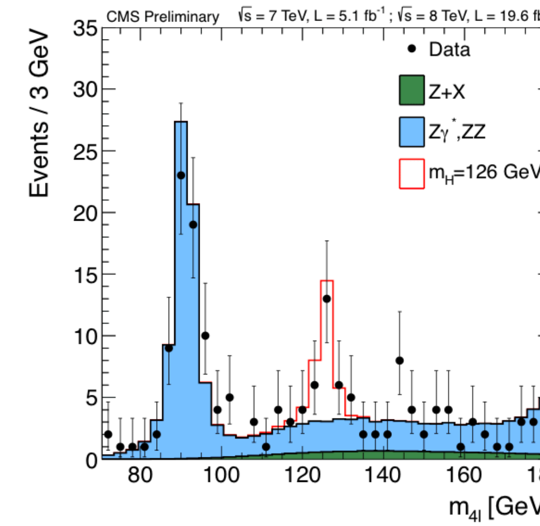
The LHC data flow-chain

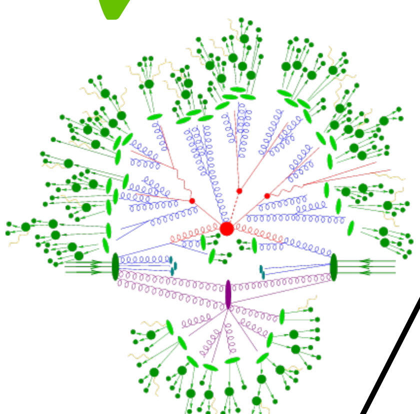
ML can play a role at every instance of this flow chain.



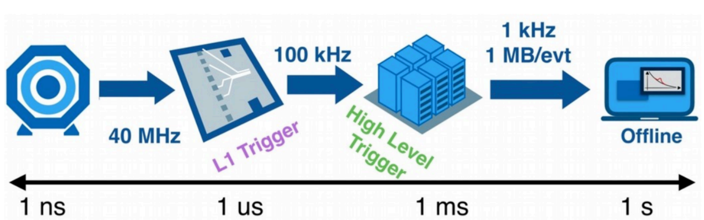


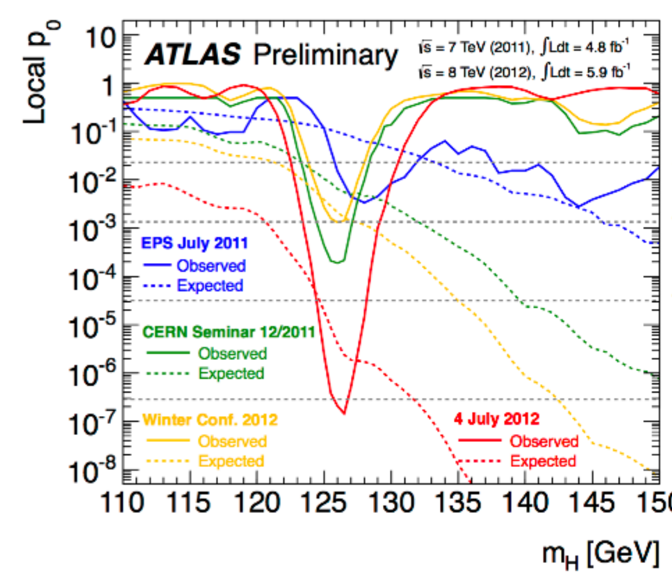






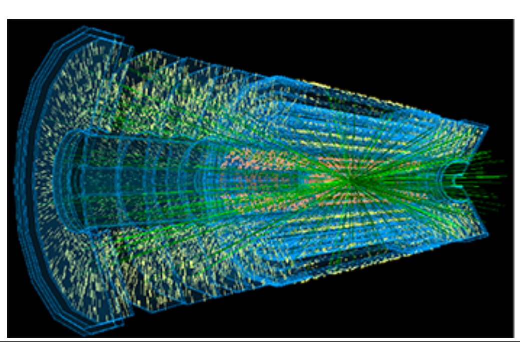
Trigger → Object rec → Event rec → Analysis



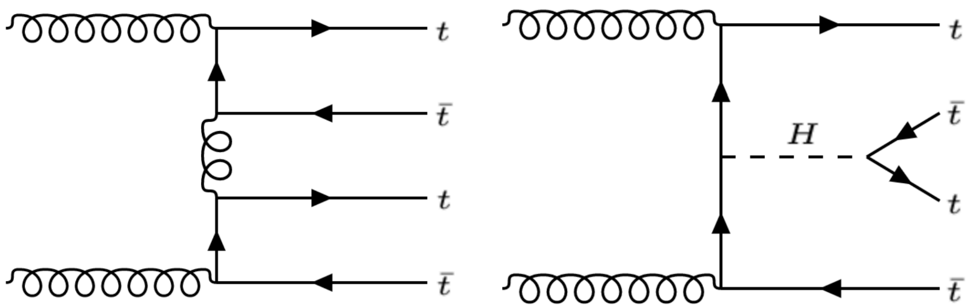


Inference

ME + PS generation
Detector Simulation



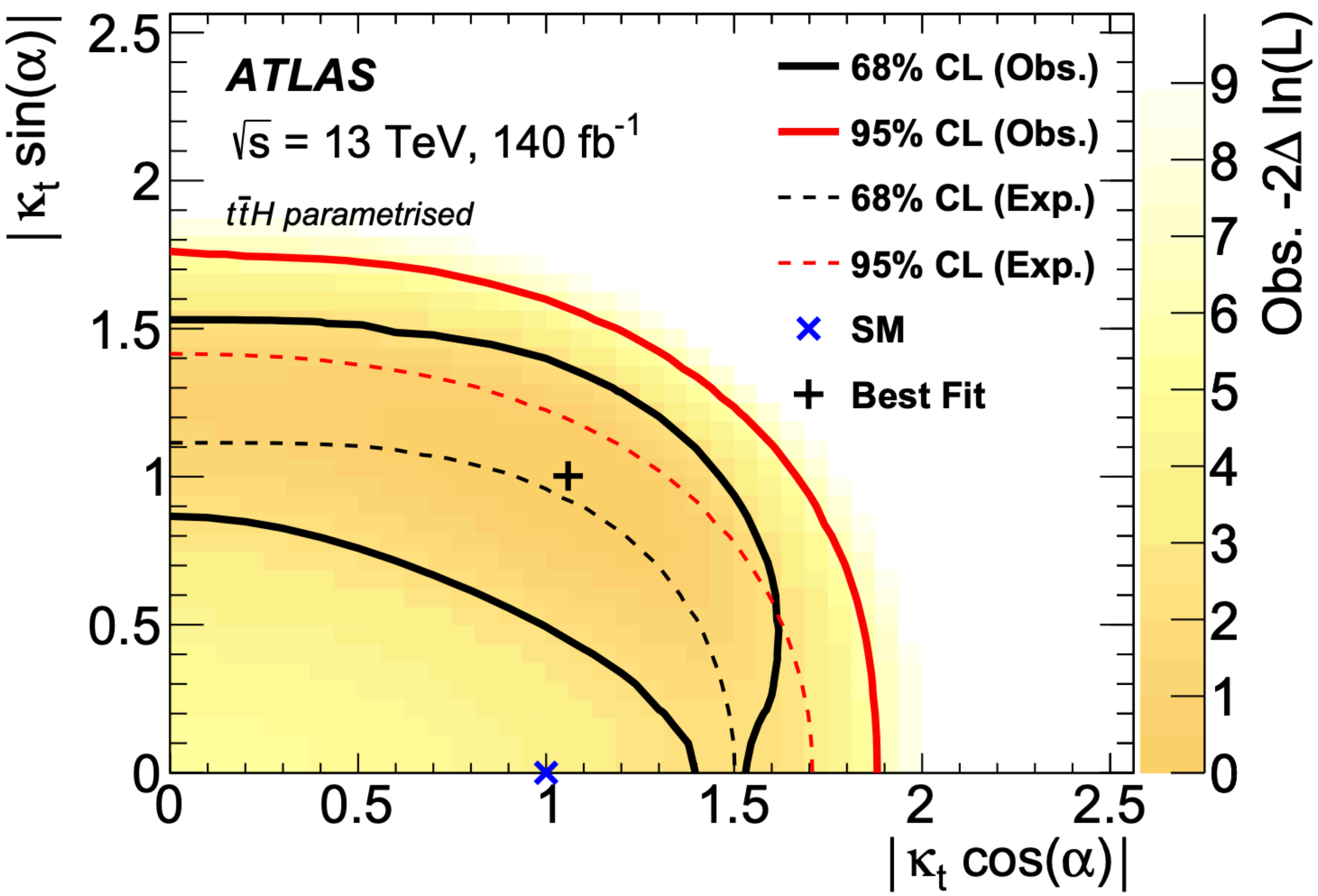
Event construction using NN

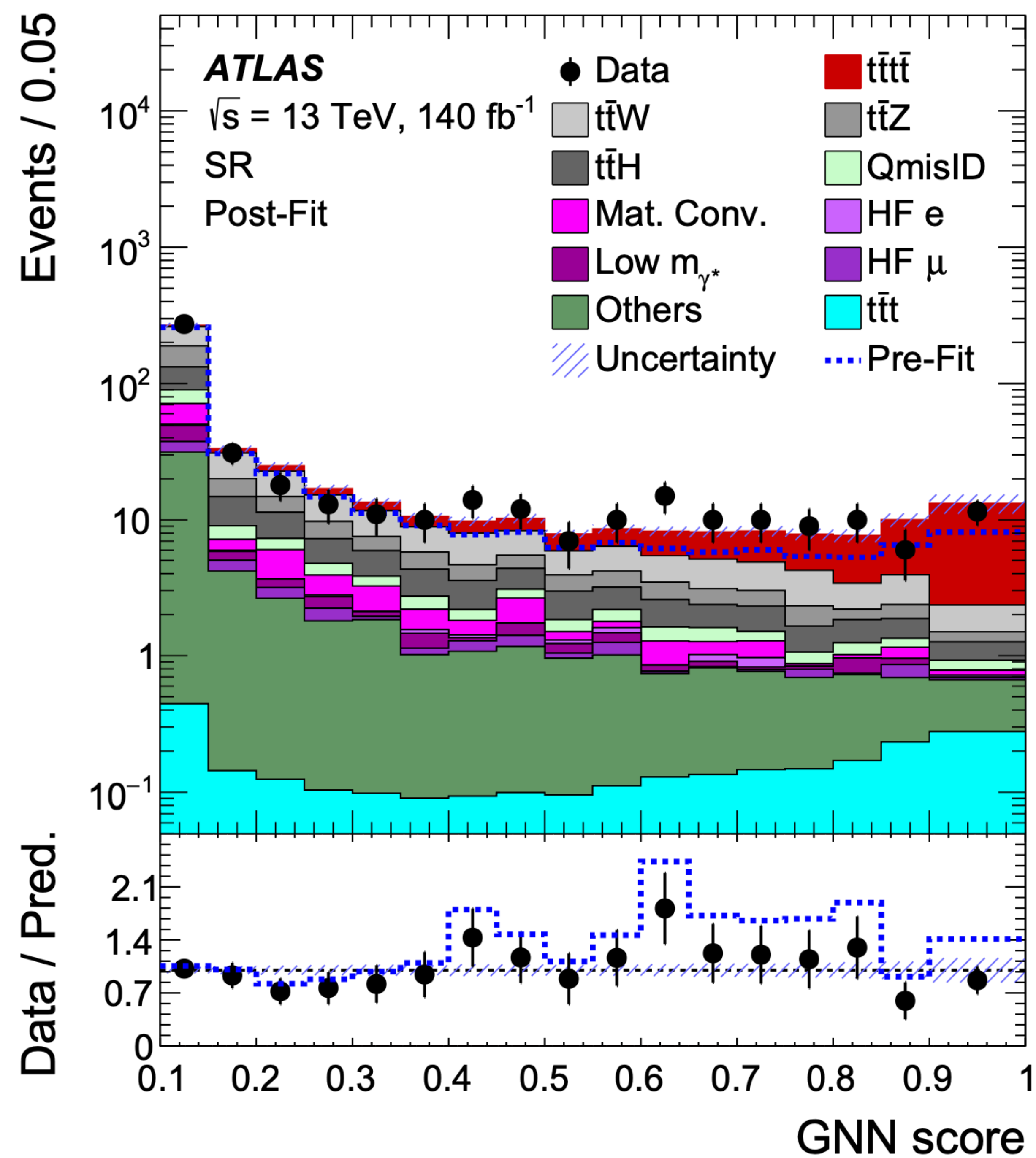


$$\sigma_{t\bar{t}t\bar{t}} = 22.5^{+4.7}_{-4.3}\,(\text{stat})\,^{+4.6}_{-3.4}\,(\text{syst})\ \text{fb} = 22.5\,^{+6.6}_{-5.5}\ \text{fb}.$$

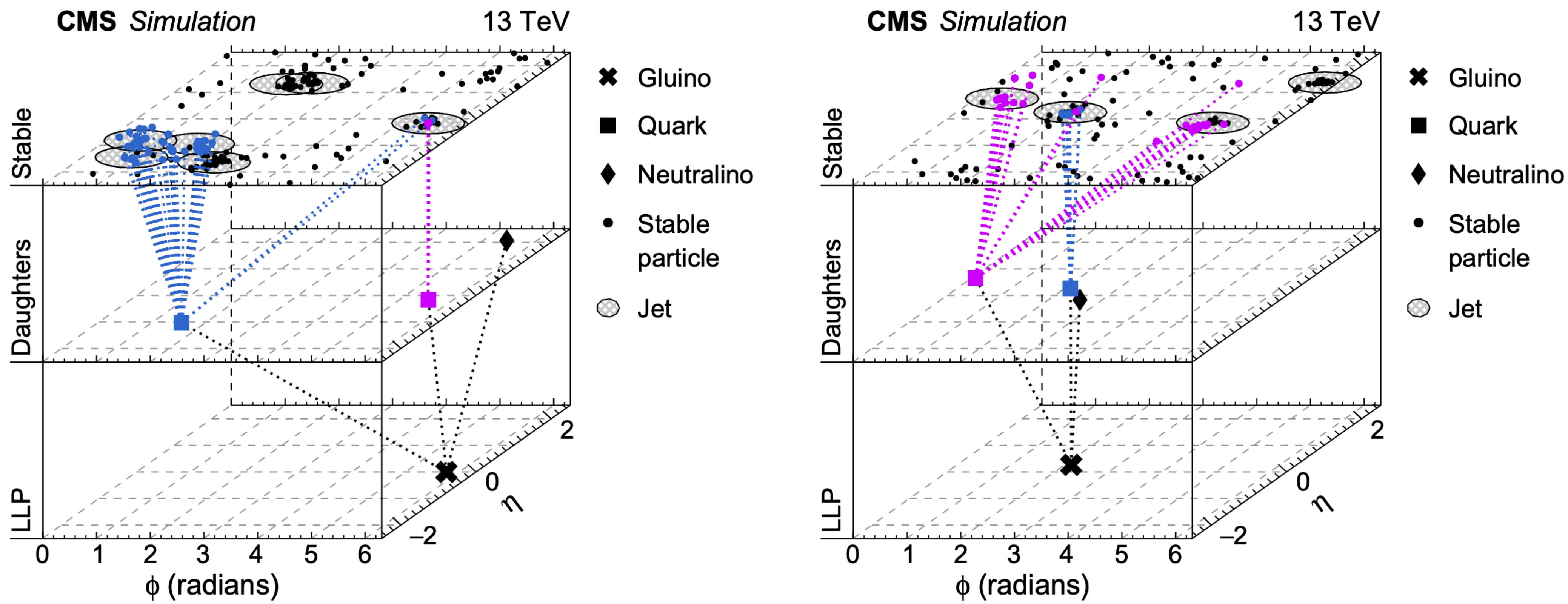
Eur. Phys. J. C 83 (2023) 496

Operators	Expected C_i/Λ^2 [TeV $^{-2}$]	Observed C_i/Λ^2 [TeV $^{-2}$]
O^1_{QQ}	[-2.4, 3.0]	[-3.5, 4.1]
O^1_{Qt}	[-2.5, 2.0]	[-3.5, 3.0]
O^1_{tt}	[-1.1, 1.3]	[-1.7, 1.9]
O^8_{Qt}	[-4.2, 4.8]	[-6.2, 6.9]

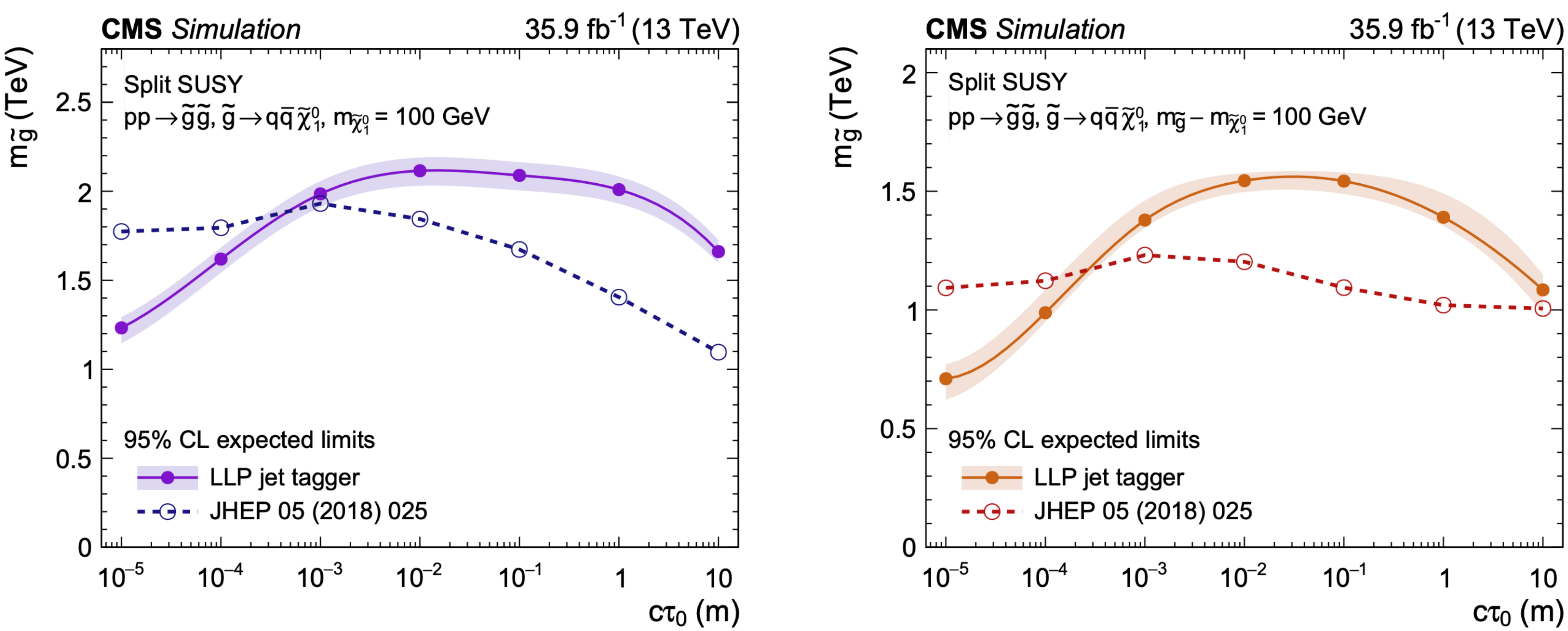




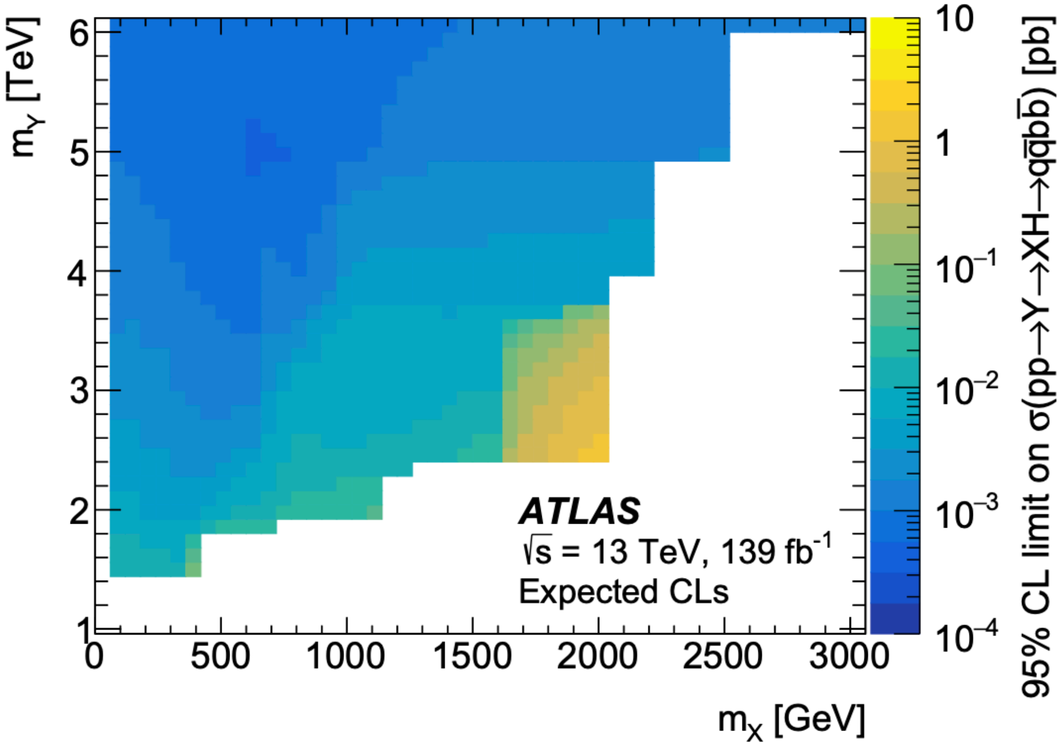
NN's are handy for searching NP

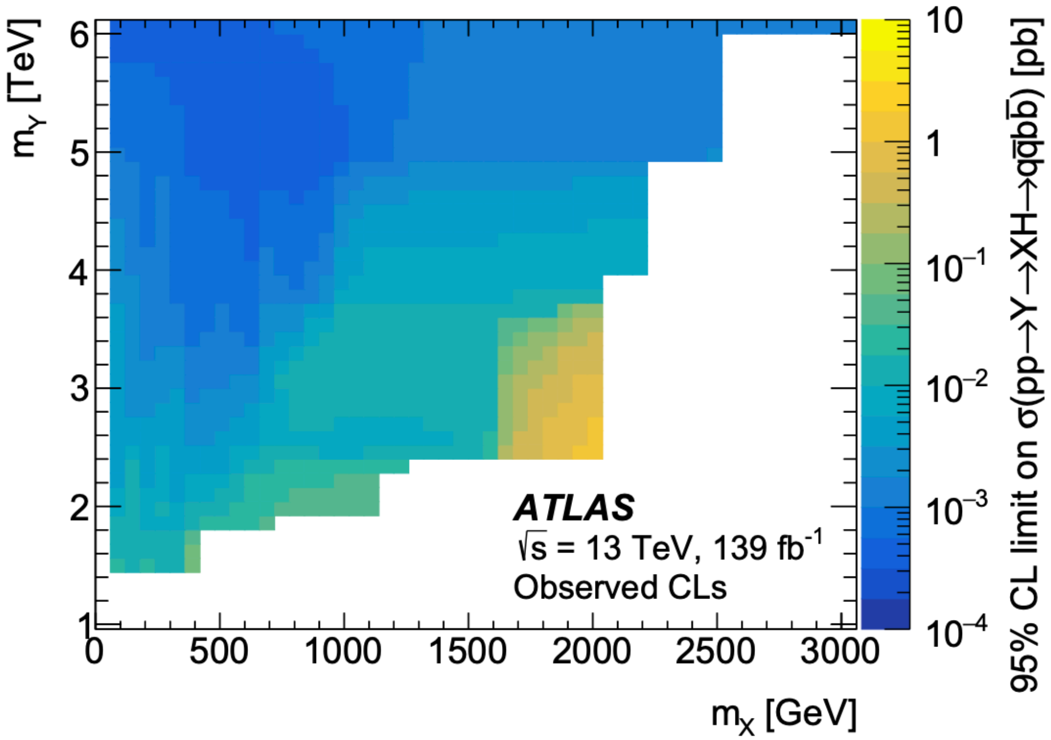


Mach. Learn.: Sci. Technol. 1 (2020) 035012

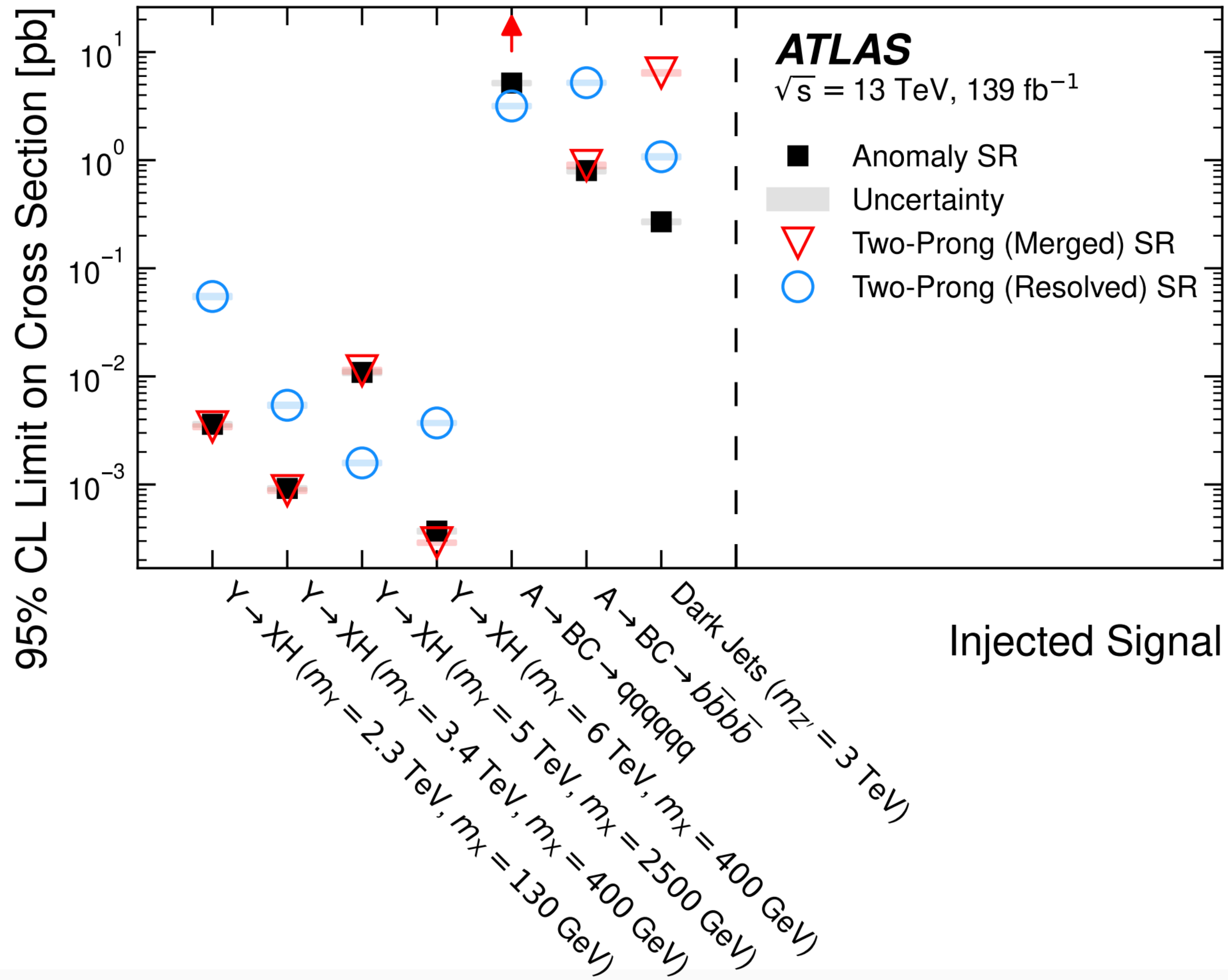


Anomaly detection using NN





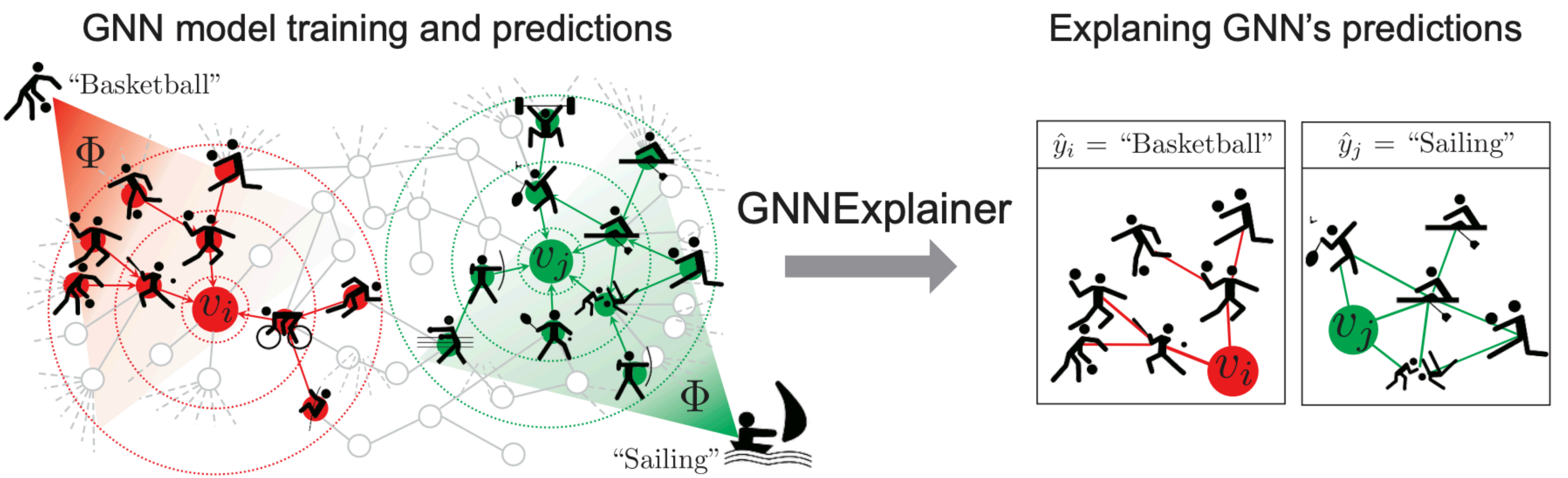
$$\mathcal{L}(t) = |\mathbf{y}(t) - \mathbf{x}(t)|^2 + \lambda D_{\mathrm{KL}}(z||z_t)$$



Phys. Rev. D 108 (2023) 052009

Future Directions

Major thrust in immediate future : Interpretability



Interpretability is a key issue and efforts are ongoing to map the NN explainability to first principle physics intuition

Interpretability : an example attempt

$$\mathbf{R}_j^{(l)} = \sum_k \frac{x_j A_{jk}}{\sum_m x_m A_{mk}} \mathbf{R}_k^{(l+1)} \quad (3)$$

where $\mathbf{R}_j^{(l)}$ represent the R-scores of the features of node j at layer l, while the quantity $x_j A_{jk}$ models the extent to which node j at layer l, with activation x_j, contributes to the relevance of node k at layer $l+1$, where A is the adjacency matrix.

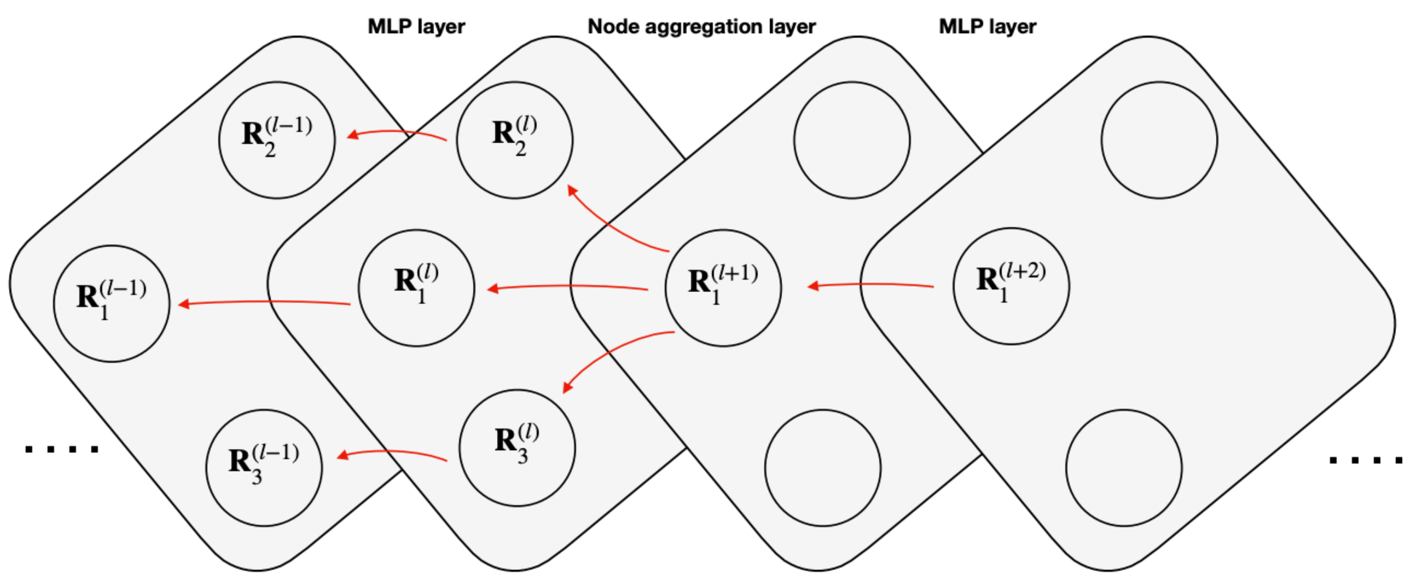
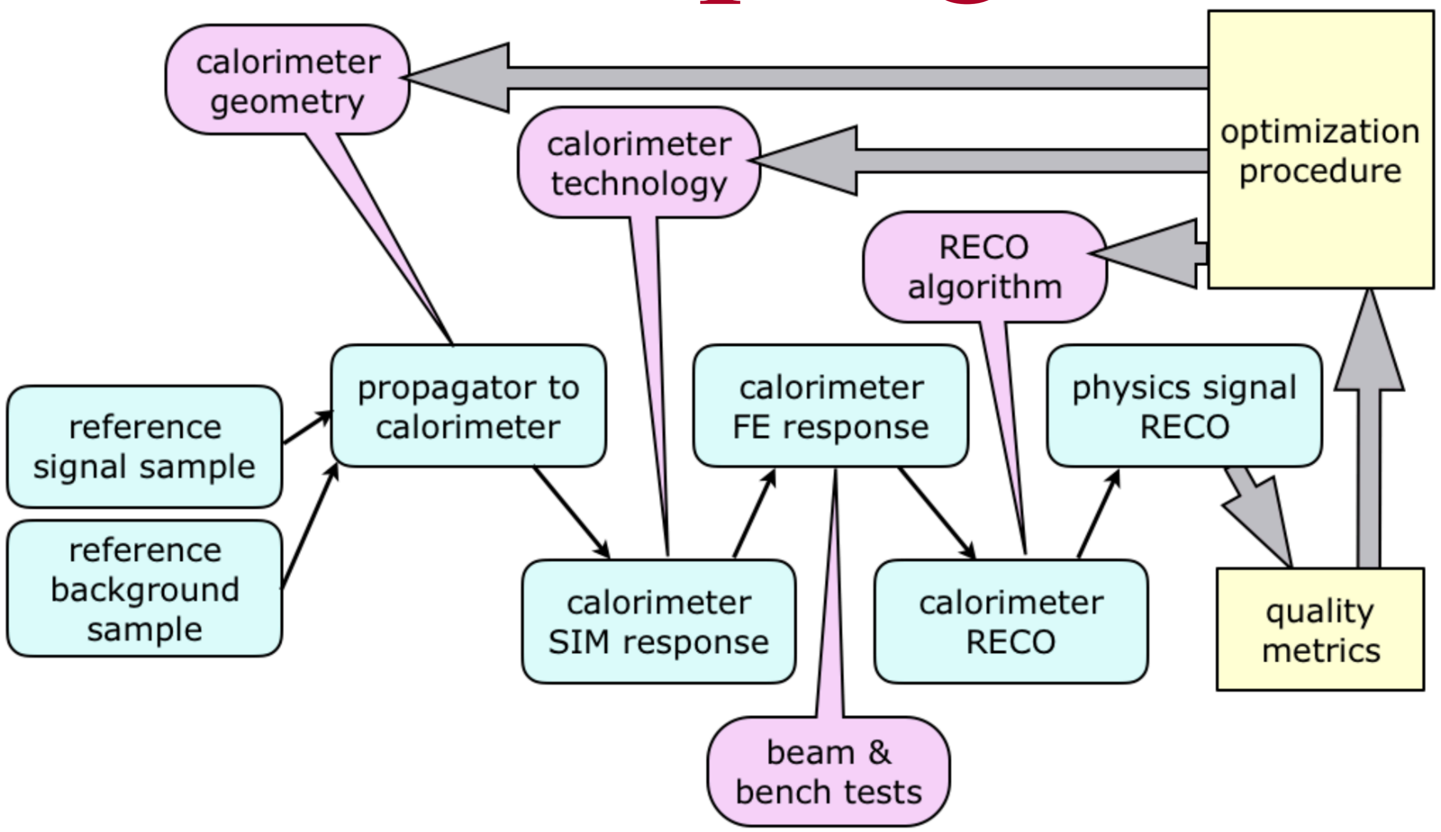


Figure 1: The flow of R-scores of node 1 across the different layers in MLPF. For MLP layers, the redistribution of R-scores follows the standard LRP rules [35, 36]. For the aggregation step in the message passing layer, the redistribution follows Equation 3. We only show three nodes for simplicity.

Explainability for MLPF

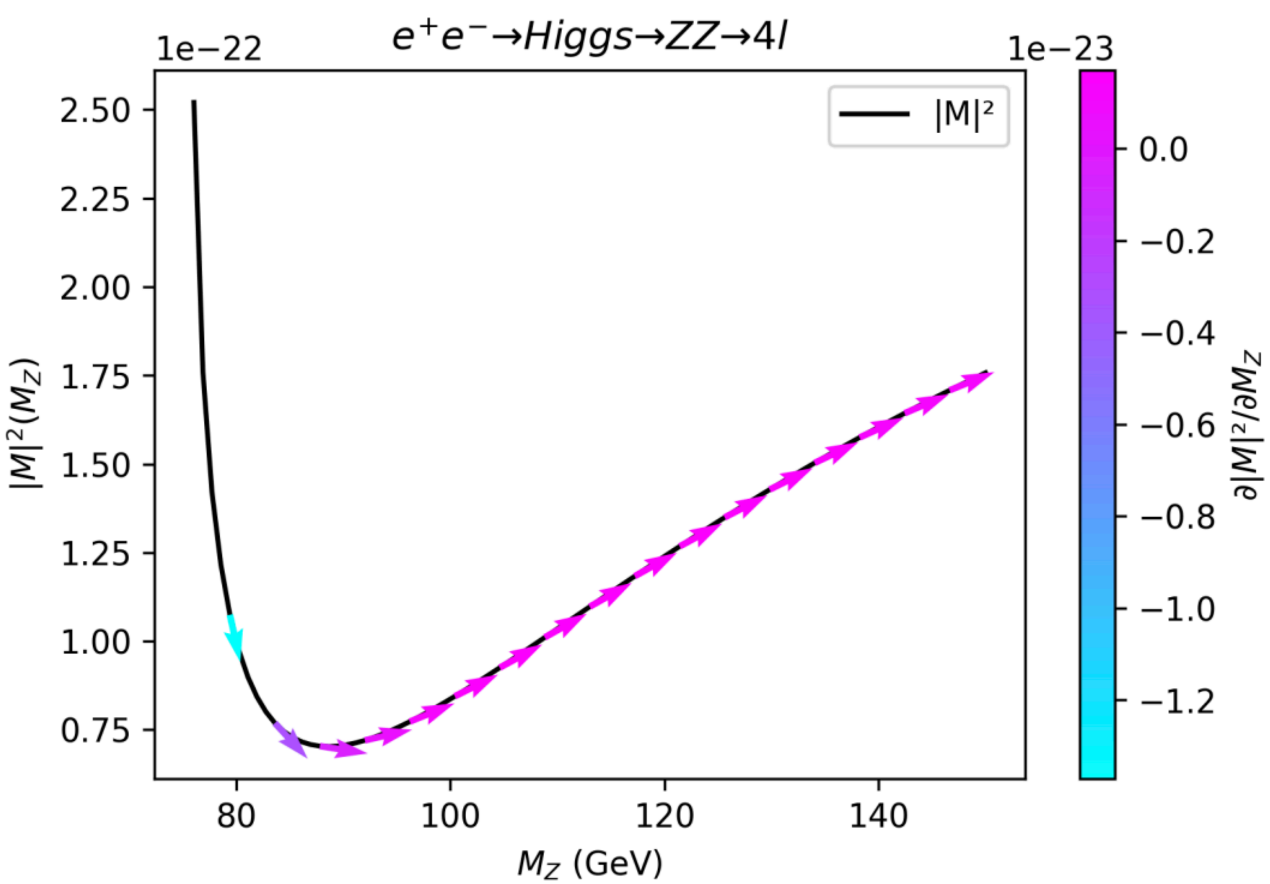
Neur IPS 2021. F. Mokhtar, R. Kansal et al

Differential programming in HEP



MODE · JENAA · iris hep · NSF · APPEC

Third MODE Workshop on
Differentiable Programming for Experiment Design
Princeton University
24-26 July, 2023



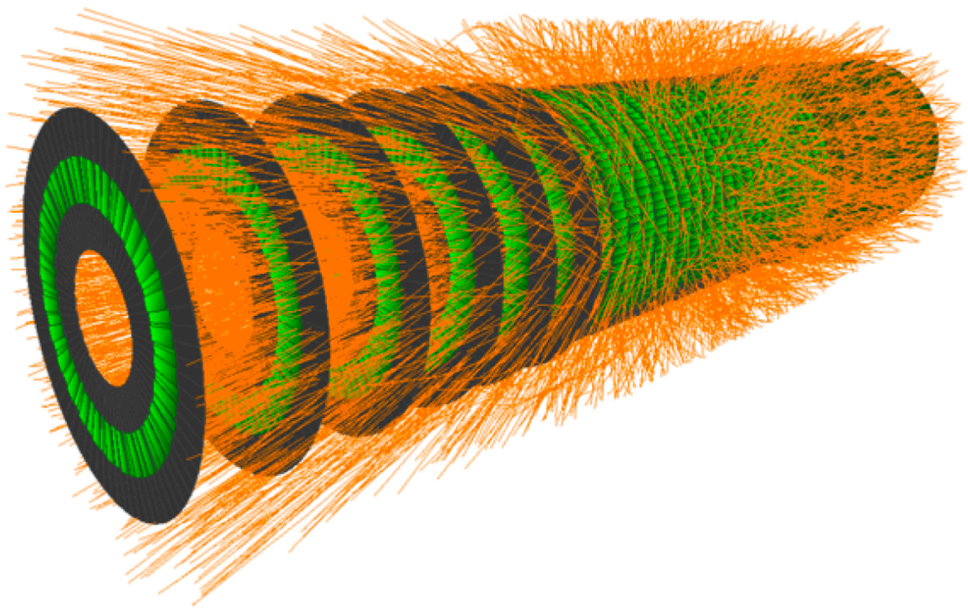

```
generate p p > t t~, t > b udsc udscx , t~ > b~ udsc udscx
output madjax generated_ttbar
set auto_update 0
```

arXiv > hep-ph > arXiv:2203.00057
High Energy Physics – Phenomenology
[Submitted on 28 Feb 2022]
Differentiable Matrix Elements with MadJax
Lukas Heinrich, Michael Kagan

2. Evaluation:

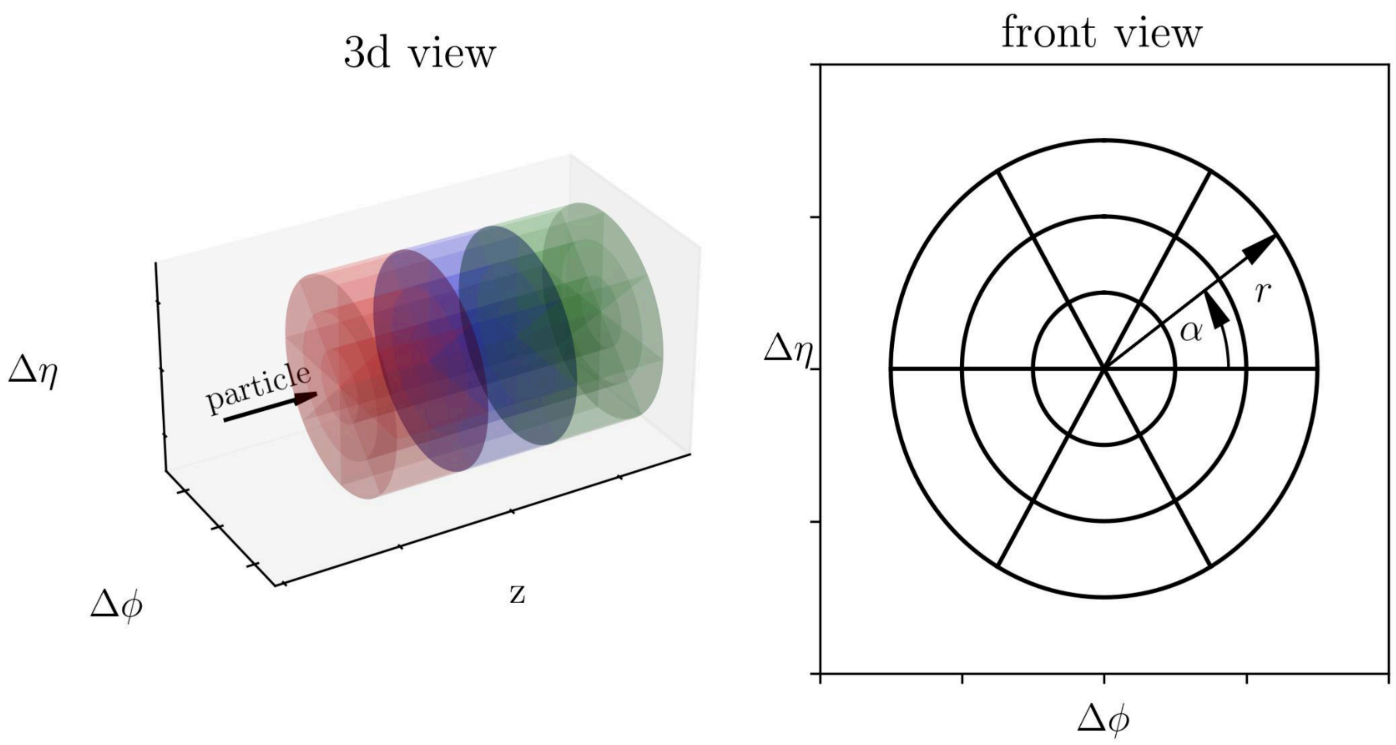
```
import madjax
mj = madjax.MadJax('generated_ttbar')
E_cm = 14000 #GeV
process = 'Matrix_1_gg_ttx_t_budx_tx_bxdux'
matrix_element = mj.matrix_element(E_cm,process)

parameters = ('mass',6): 173.0 #set top mass
phasespace_coords = [0.1]*14 #14D phasespace

val, grad = matrix_element(parameters,phasespace_coords)
grad[('mass', 6)] #gradient wrt top mass
```

Open data for ML R&D at colliders

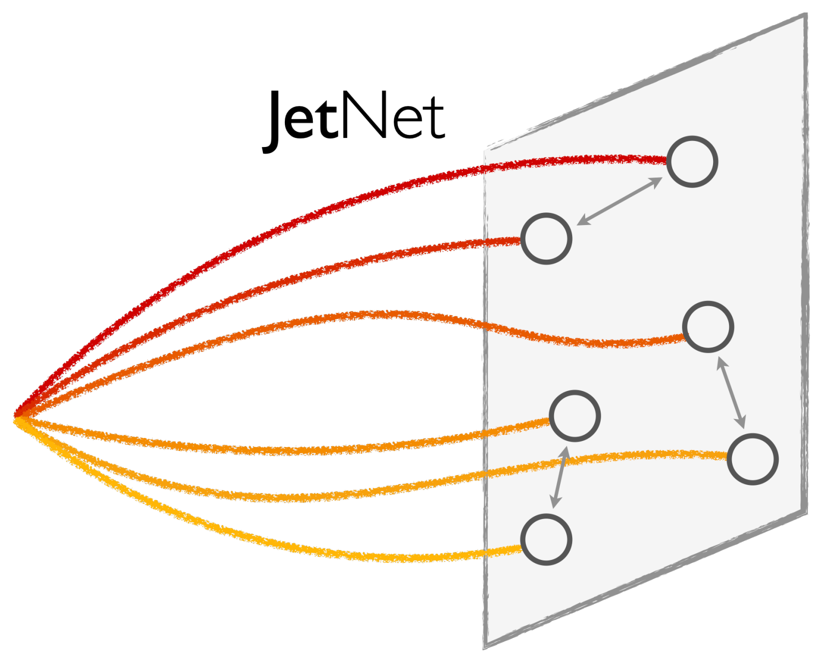


Track-ML challenge :
https://sites.google.com/site/trackmlparticle/home

Calo-ML challenge :
https://sites.google.com/site/trackmlparticle/home



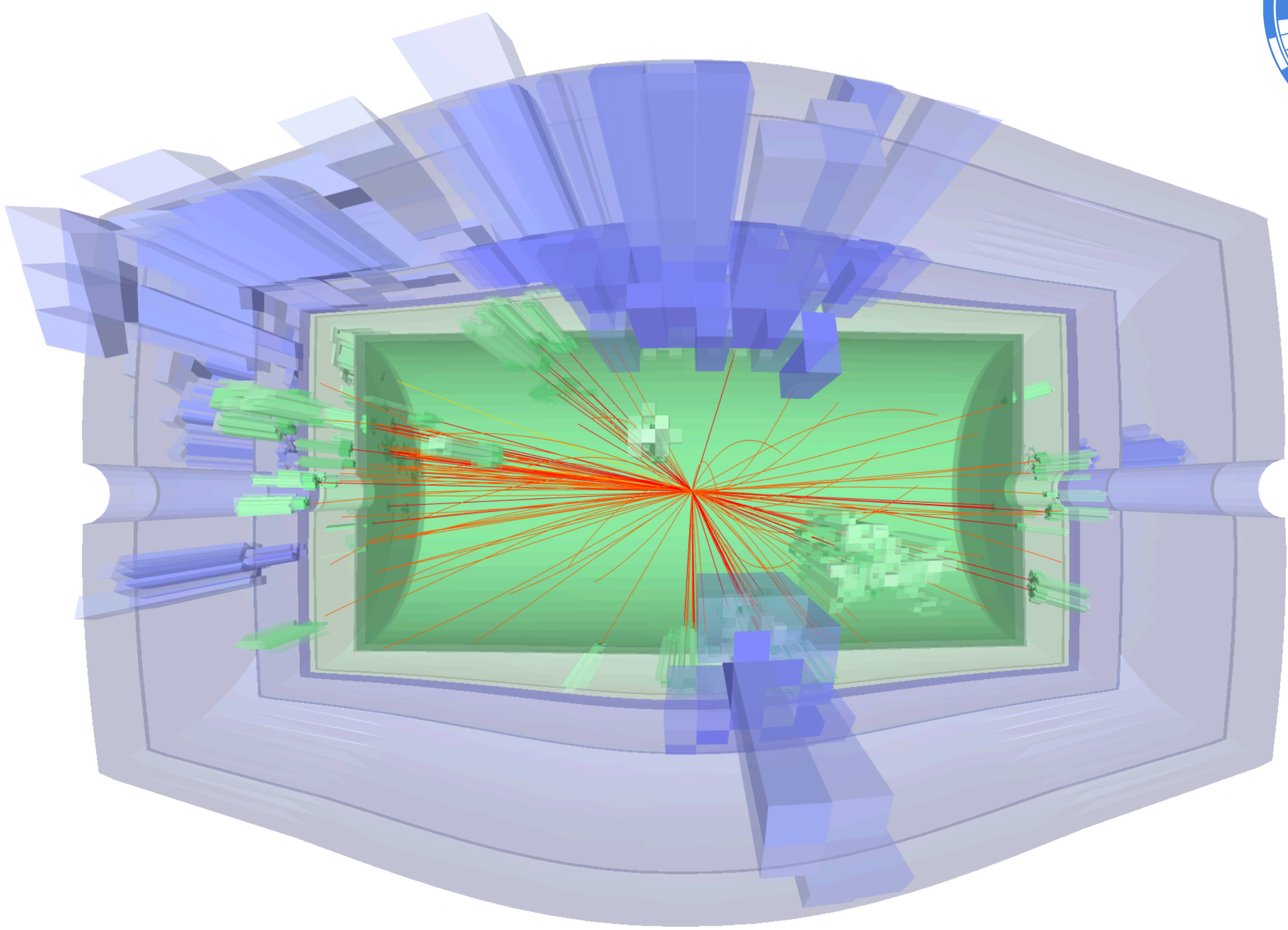
https://github.com/rkansal47/JetNet

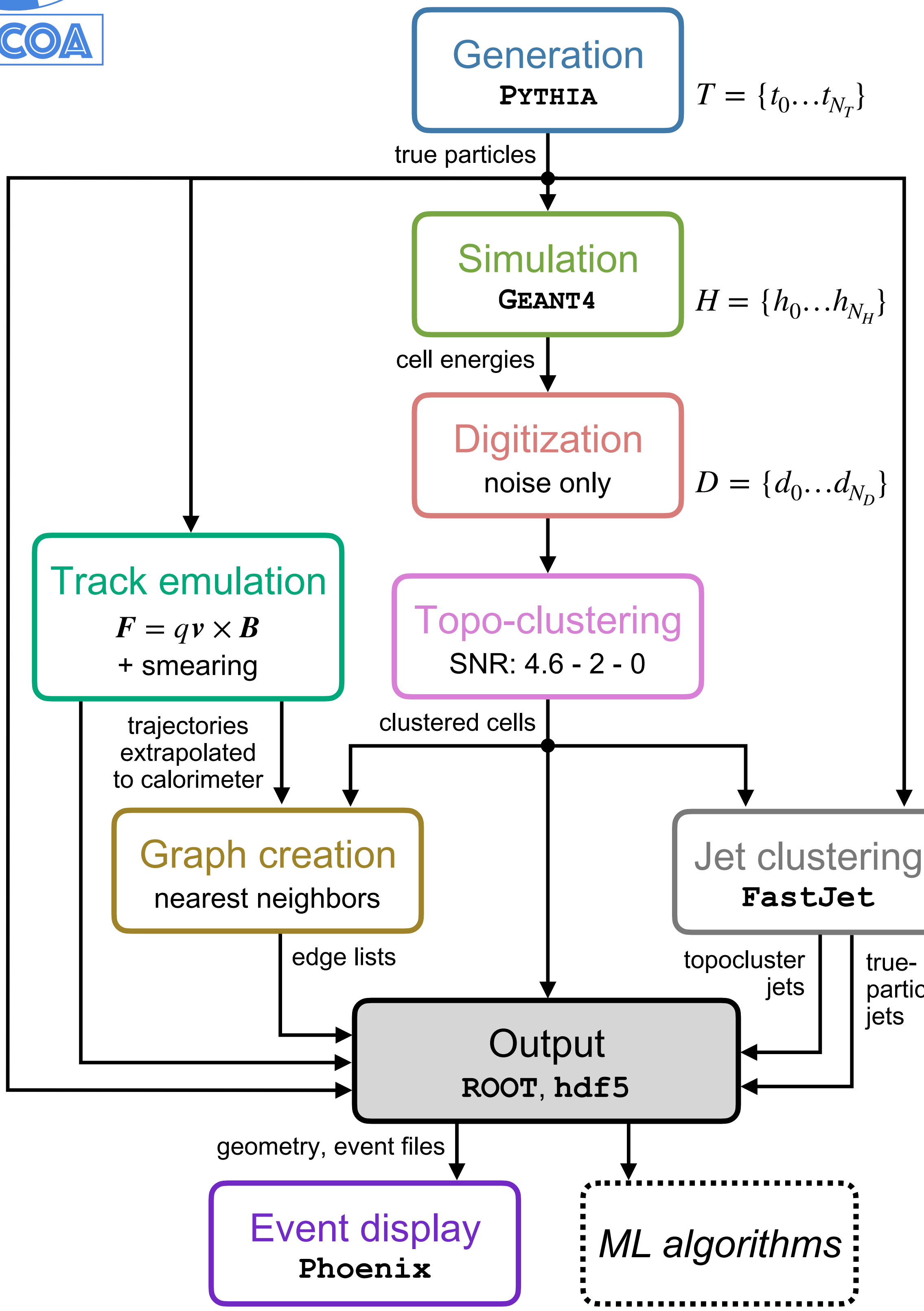


The COCOA

Mach. Learn.: Sci. Technol. 4 035042

https://cocoa-hep.readthedocs.io/en/latest/





COnfigurable **C**al**O**rimeter simulation for **A**I

☑ **A complete hermetic geometry with full GEANT simulation.**

☑ **PYTHIA-8 based ME/PS & Hadronization**

☑ **FASTJET integration is inbuilt.**

☑ **Comes with an ATLAS style pPFlow.**

Take away

- ☑ ML is here to stay with HEP.
- ☑ We can't blindly do a plug & play of the available NN.
- ☑ Interpretability and uncertainty estimations are two key aspects where we the HEP-ML people need to emphasize.
- ☑ Need to keep a close connection with the comp-sc/math community with the latest developments and contribute if possible.
- ☑ Symmetry equivariance and geometric DL methods might play a key role in this field.
- ☑ Many important application of ML are happening in hep-lat and hep-th community as well (See talk by P. Konar)

https://iml-wg.github.io/HEPML-LivingReview/

https://iris-hep.org/