中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

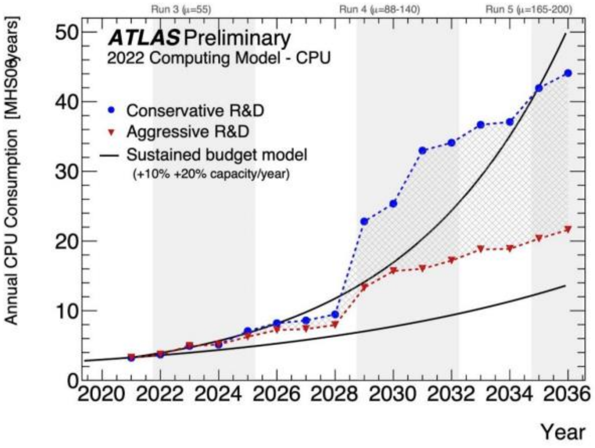
ML for fast calorimeter simulation

Wenxing Fang (IHEP)

第十四届全国粒子物理学术会议 2024.08.15

Background

- HL-LHC data challenge:
  - Event rate increased: luminosity from $2 \times 10^{34}\mathrm{s}^{-1}\mathrm{cm}^{-2}$ to $7.5 \times 10^{34}\mathrm{s}^{-1}\mathrm{cm}^{-2}$
  - Event size increased: finer-grained detector readout
  - Aggressive R&D is needed, otherwise the resources will be a problem
- The MC simulation takes most CPU resources, dominated by the detector simulation of calorimeter
  - Traditional fast calorimeter simulation methods: shower parameterization (PCA), frozen shower, …
  - In recent years, ML based methods show significant promise as a replacement



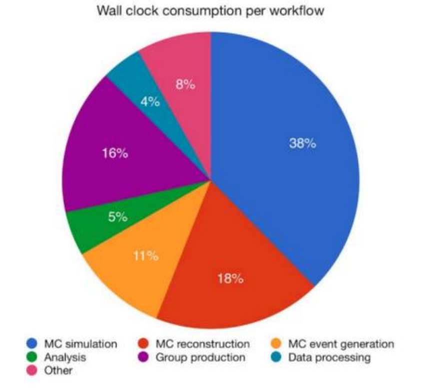
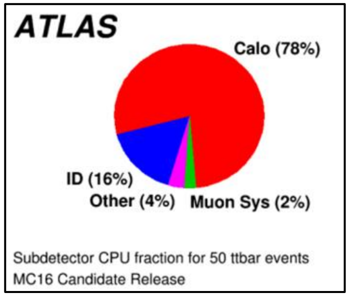
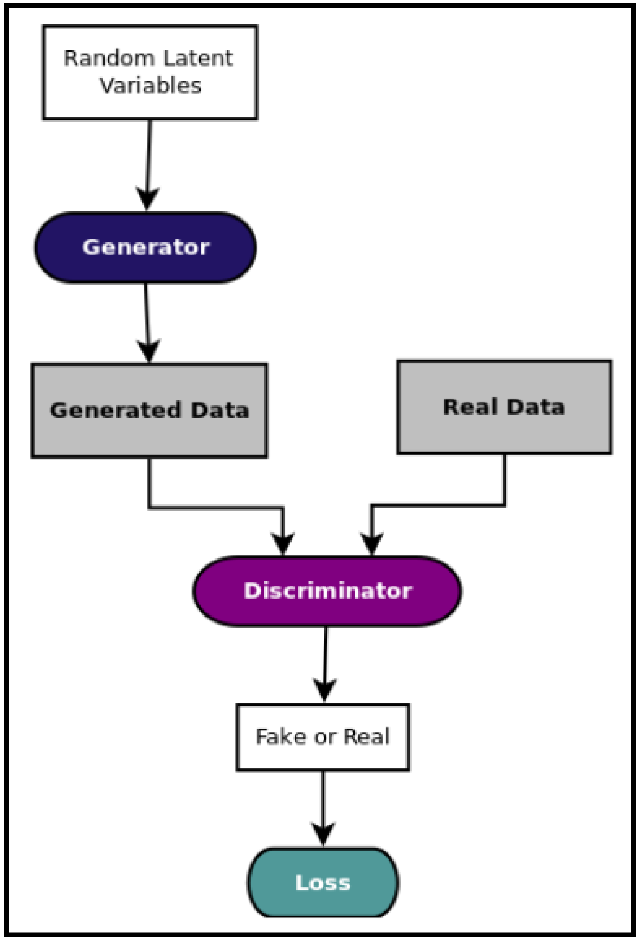


Figure 1: ATLAS CPU hours used by various activities in 2018



Generative Adversarial Networks (GAN)

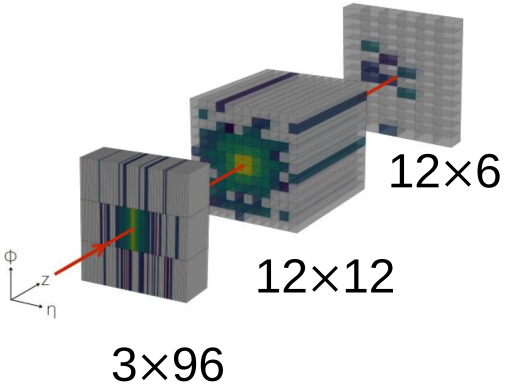
- Discriminator tries to discriminate the real data and generated data
- The generator tries to produce generated data which can confuse the discriminator
- At the end of training, the discriminator can not discriminate the real or generated data. The generator learns the true underlying data distribution

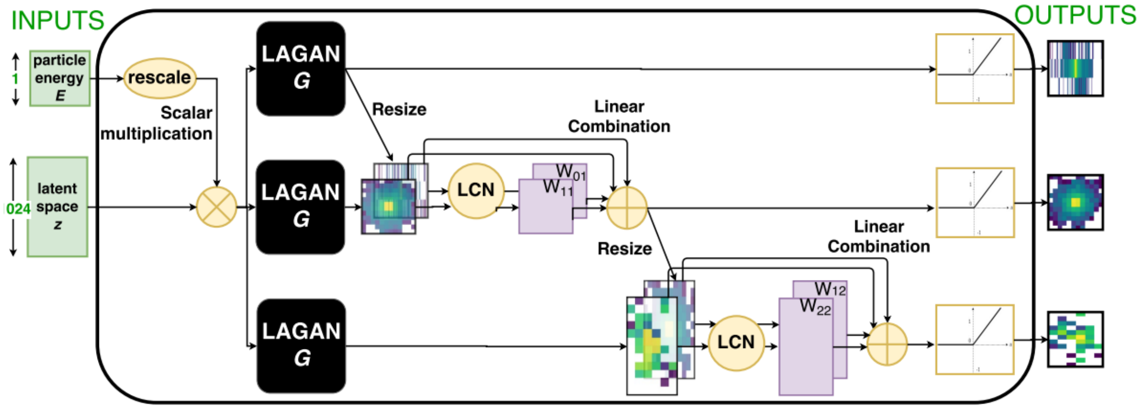


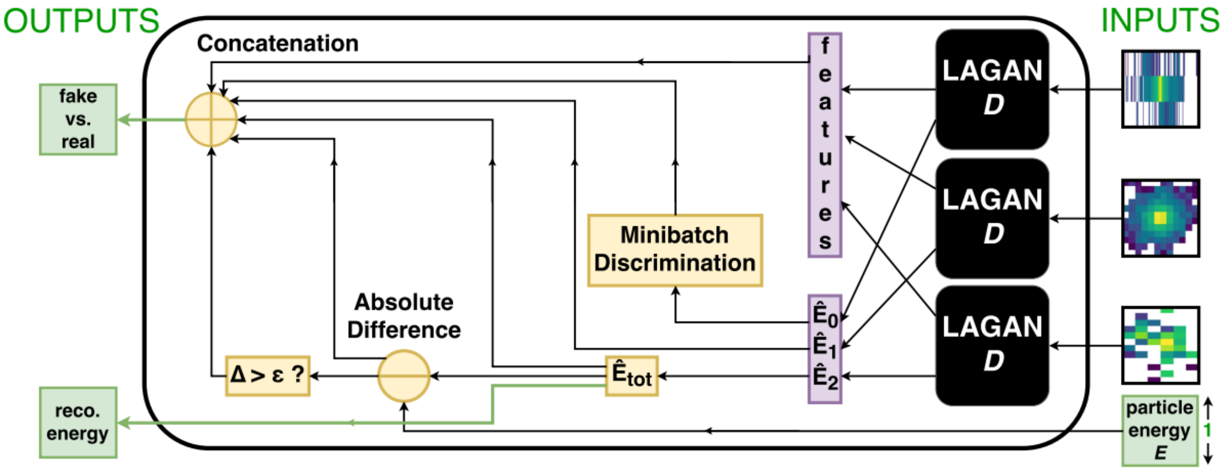
$$\min_{G} \max_{D} V(D,G) = E_{x\sim p_{data}(x)}[logD(x)] + E_{z\sim p_z(z)}[log(1 - D(G(z)))]$$

CaloGAN

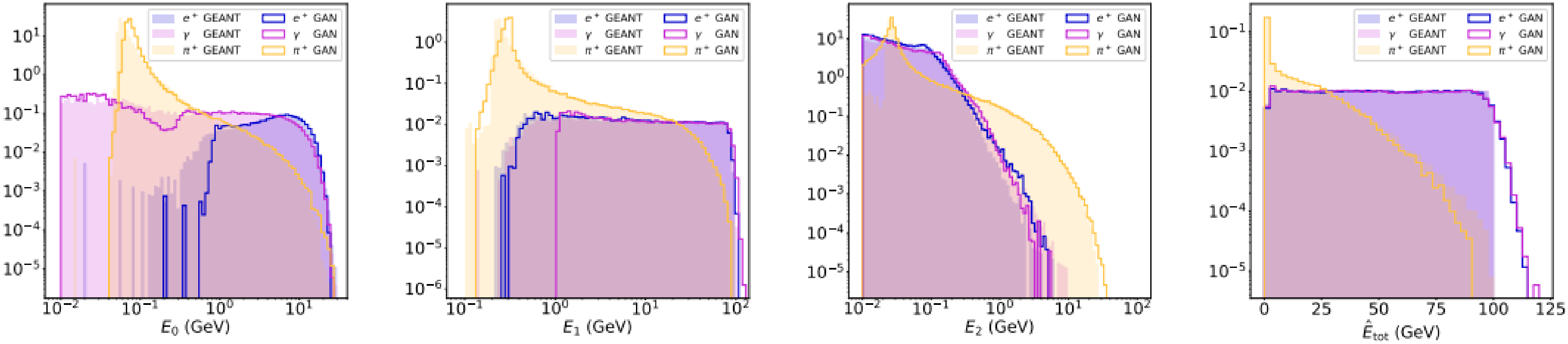
- The [CaloGAN](#) (2017) achieved a fast calorimeter simulation based on GAN







CaloGAN performance



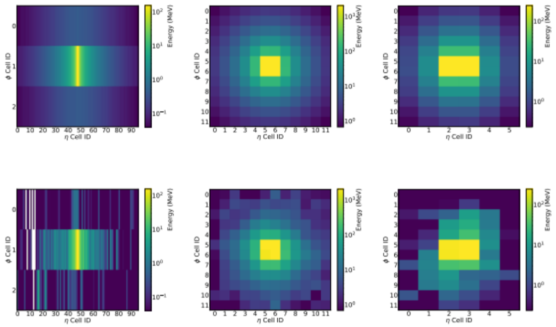
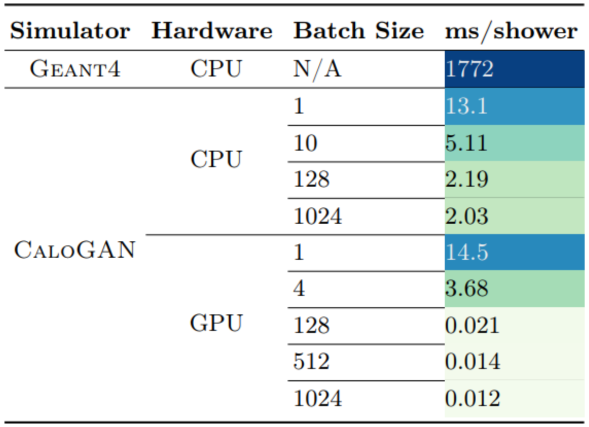


FIG. 8: Average π^+ GEANT4 shower (top), and average π^+ CALOGAN shower (bottom), with progressive calorimeter depth (left to right).



Simulator	**Hardware**	**Batch Size**	**ms/shower**
GEANT4	CPU	N/A	1772
CALOGAN	CPU	1	13.1
		10	5.11
		128	2.19
		1024	2.03
	GPU	1	14.5
		4	3.68
		128	0.021
		512	0.014
		1024	0.012

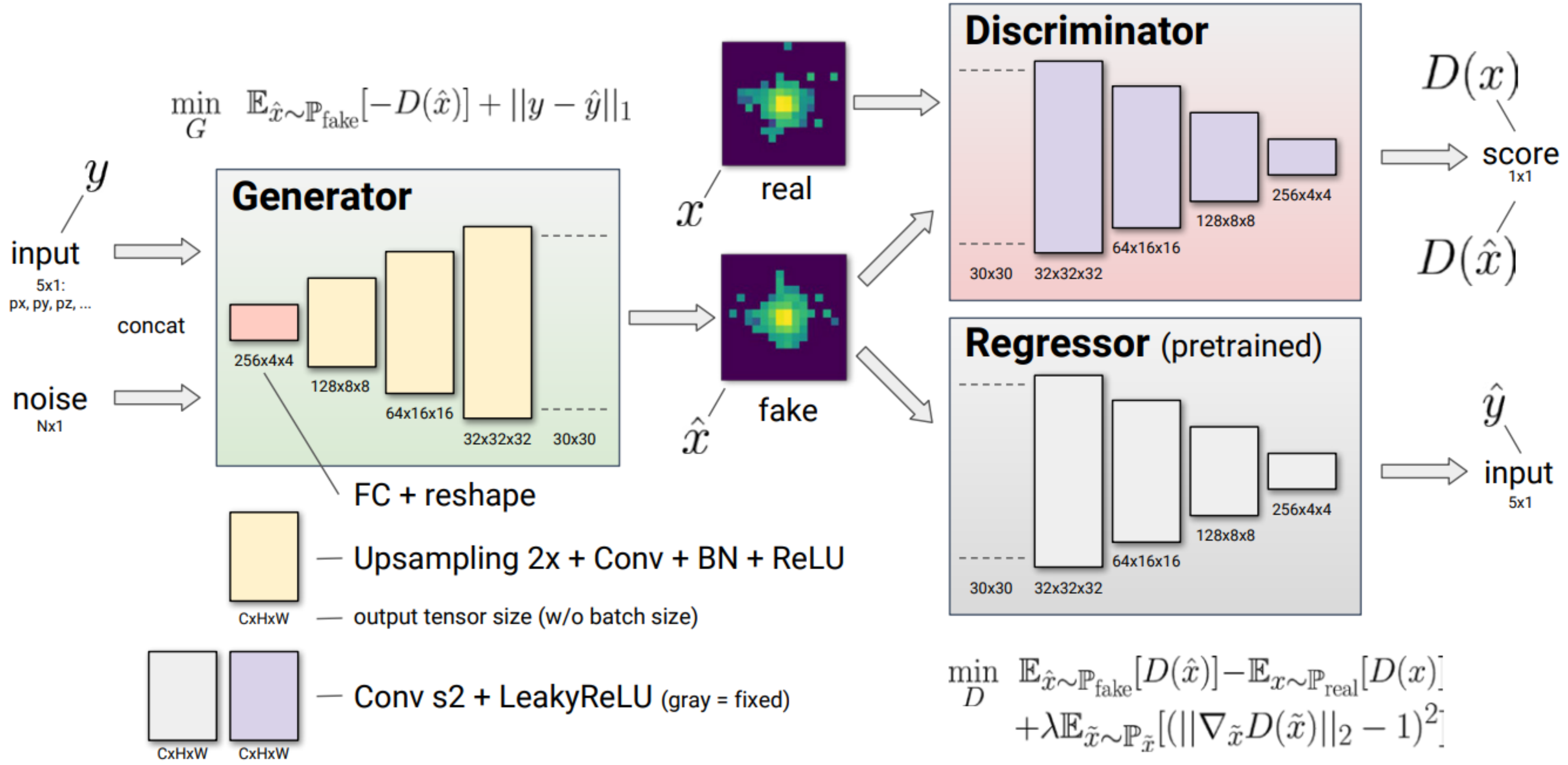
e^+ vs. π^+

		Test on	
		GEANT4	CALOGAN
Train on	GEANT4	99.6% ± 0.1%	96.5% ± 1.1%
	CALOGAN	98.2% ± 0.9%	99.9% ± 0.2%

e^+ vs. γ

		Test on	
		GEANT4	CALOGAN
Train on	GEANT4	66.1% ± 1.2%	70.6% ± 2.6%
	CALOGAN	54.3% ± 0.8%	100.0% ± 0.0%

The LHCb case

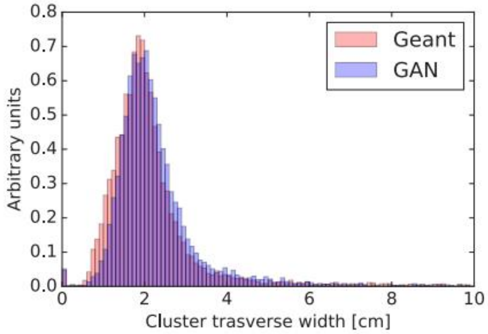


$$\min_G \; \mathbb{E}_{\hat{x}\sim\mathbb{P}_{\text{fake}}}[-D(\hat{x})] + ||y - \hat{y}||_1$$

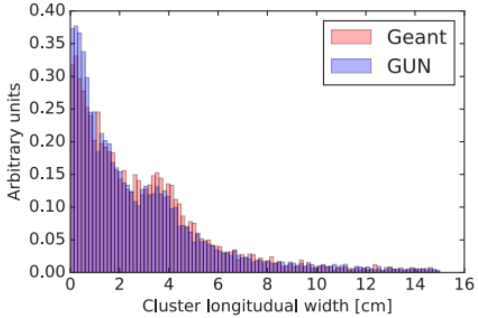
$$\min_D \; \mathbb{E}_{\hat{x}\sim\mathbb{P}_{\text{fake}}}[D(\hat{x})] - \mathbb{E}_{x\sim\mathbb{P}_{\text{real}}}[D(x)] + \lambda \mathbb{E}_{\tilde{x}\sim\mathbb{P}_{\tilde{x}}}[(||\nabla_{\tilde{x}} D(\tilde{x})||_2 - 1)^2]$$

link

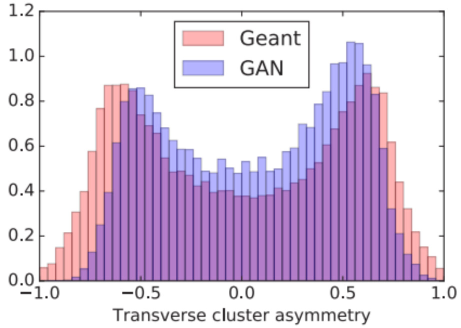
The LHCb case (performance)



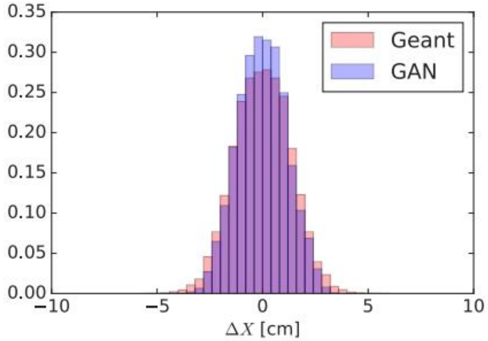
(a) The transverse width of real and generated clusters



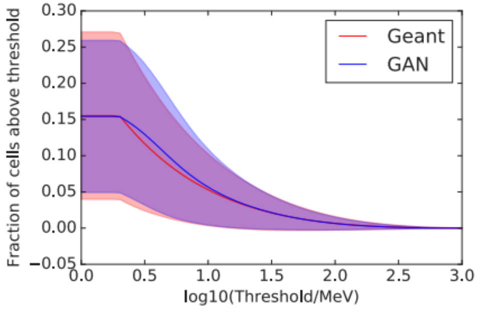
(b) The longitudinal width of real and generated clusters



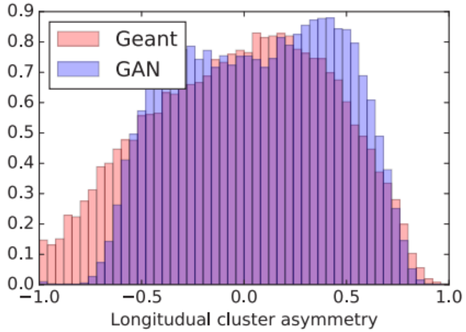
(e) The transverse asymmetry of real and generated clusters



(c) ΔX between cluster center of mass and the true particle coordinate



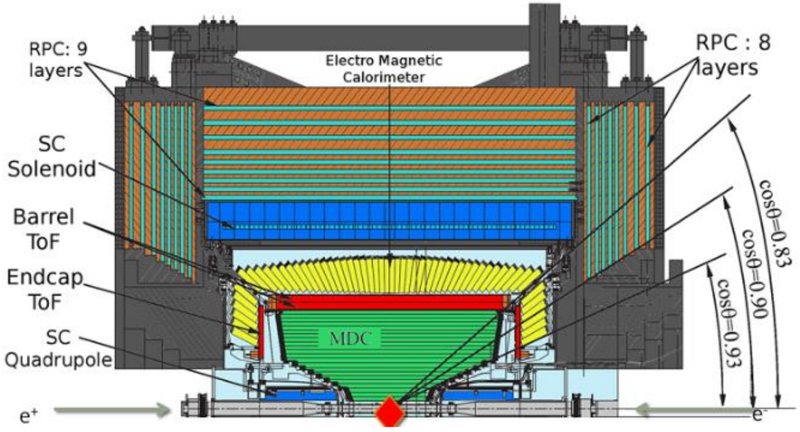
(d) The sparsity of real and generated clusters



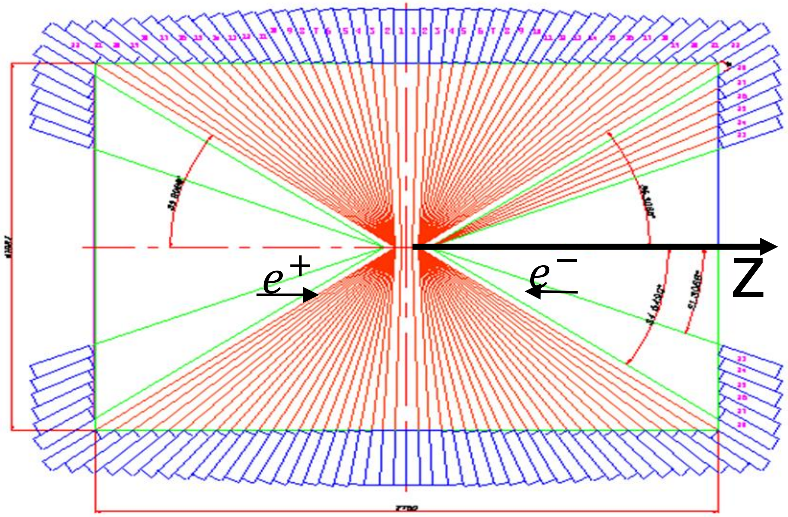
(f) The longitudinal asymmetry of real and generated clusters

BESIII experiment

- The BESIII experiment focuses on physics at the tau-charm region, such as non-perturbative QCD, exotic hadrons, BSM, …



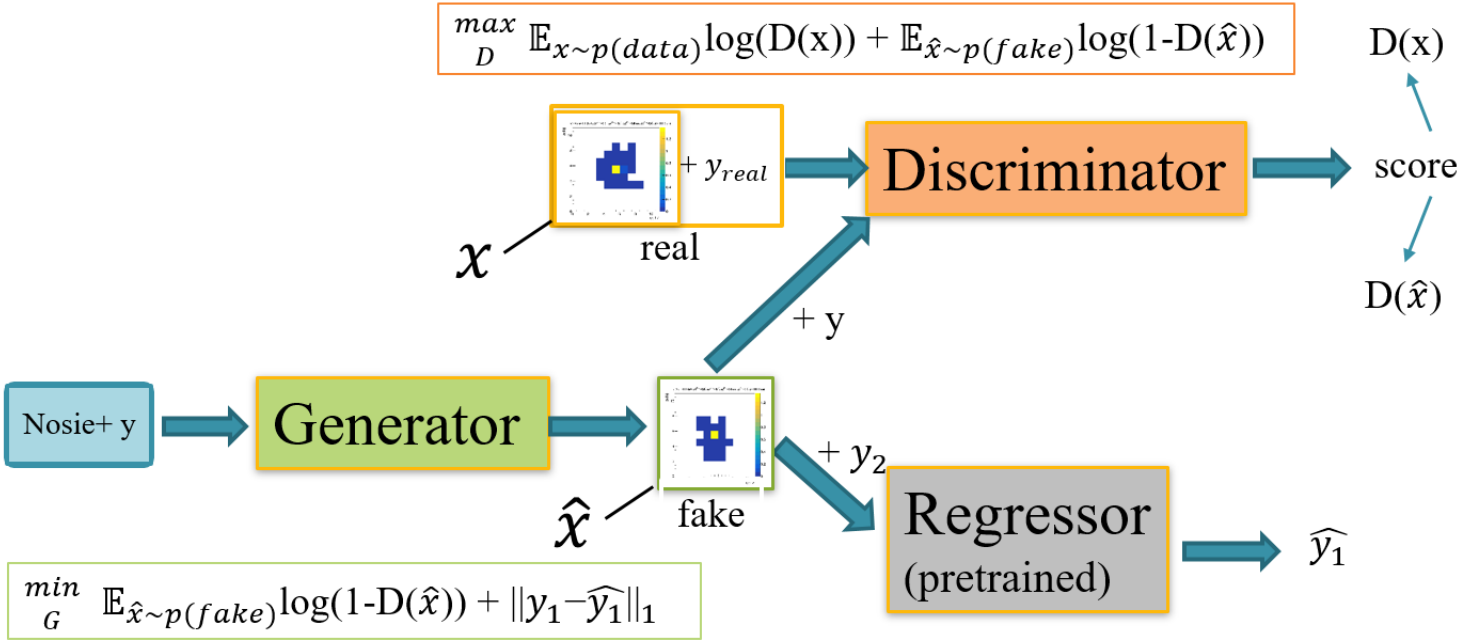
- BESIII EMC:
    - 44 rings of crystal in barrel and 120 crystals in each ring. The front size of each crystals is 5×5 cm^2
    - 6 rings of crystal in each endcap



- Apply GAN for EMC barrel fast simulation for $e^\pm$

GAN model for the BESIII

- The structure is similar to the LHCb one

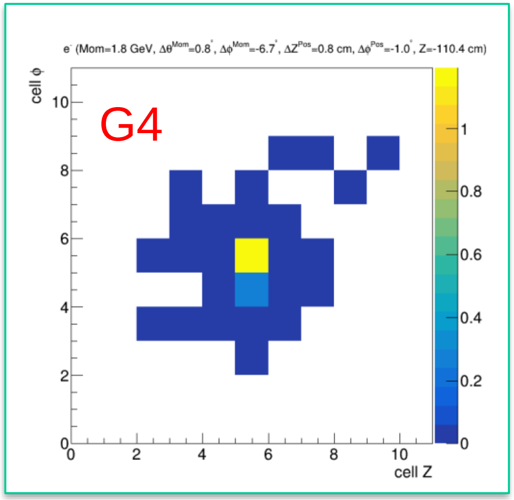


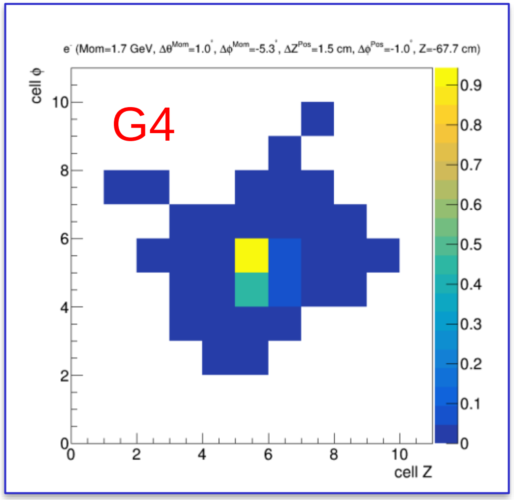
$$\max_{D} \mathbb{E}_{x\sim p(data)}\log(D(x)) + \mathbb{E}_{\hat{x}\sim p(fake)}\log(1\text{-}D(\hat{x}))$$

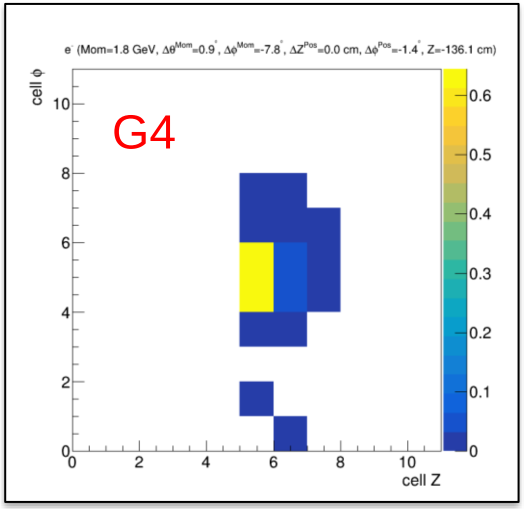
$$\min_{G} \mathbb{E}_{\hat{x}\sim p(fake)}\log(1\text{-}D(\hat{x})) + \|y_1 - \widehat{y_1}\|_1$$

- The y ($y_1 + y_2$) contains the momentum of particle and the relative position and angular between the particle and the calorimeter.
  - y_1
    - Momentum: the momentum of the particle.
    - $\Delta\phi^{\mathrm{Mom}}$: the ϕ difference between the momentum of incoming particle and the direction of the crystal.
    - $\Delta\theta^{\mathrm{Mom}}$: the θ difference between the momentum of incoming particle and the direction of the crystal.
  - y_2
    - ΔZ^{Pos}: the Z difference between the hit point of incoming particle and the z of front center of the crystal.
    - $\Delta\phi^{\mathrm{Pos}}$: the ϕ difference between the hit point of incoming particle and the ϕ of front center of the crystal.
    - Z: the Z value of hit point

Event display (e^-)



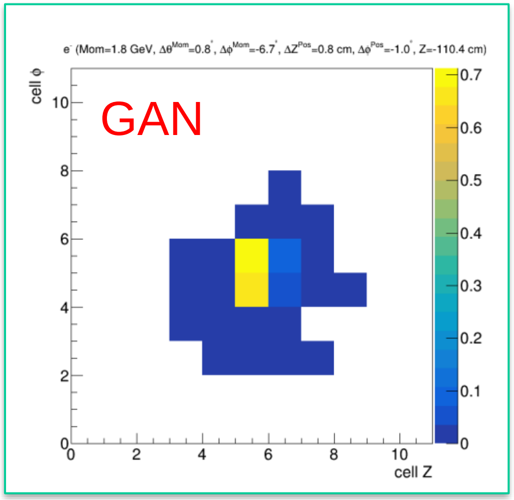


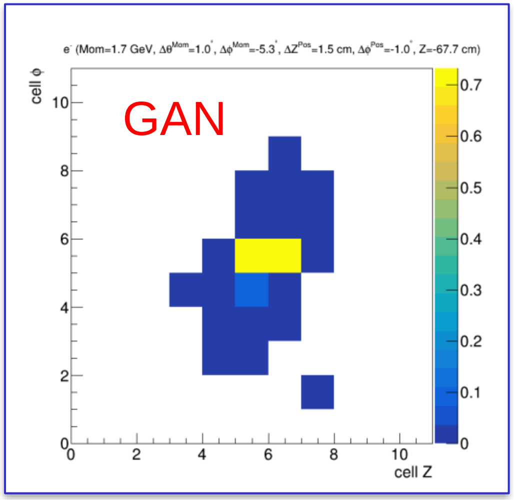


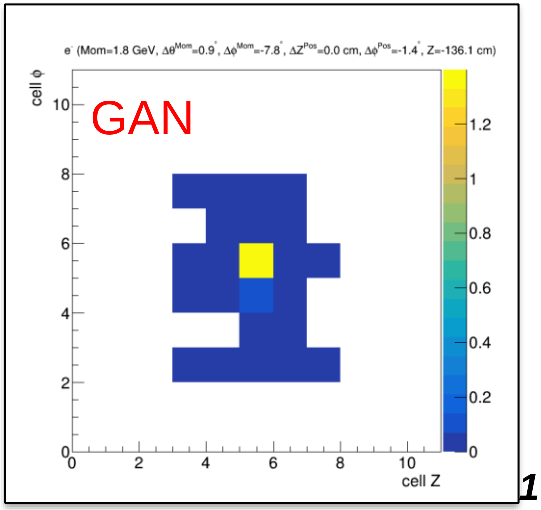
$\mathrm{e}^-(\mathrm{Mom} = 1.8\ \mathrm{GeV}, \Delta\theta^{\mathrm{Mom}} = 0.8^\circ, \Delta\phi^{\mathrm{Mom}} = -6.7^\circ,$
$\Delta Z^{\mathrm{Pos}} = 0.8\ \mathrm{cm}, \Delta\phi^{\mathrm{Pos}} = -1.0^\circ, Z = -110.4\ \mathrm{cm})$

$\mathrm{e}^-(\mathrm{Mom} = 1.7\ \mathrm{GeV}, \Delta\theta^{\mathrm{Mom}} = 1.0^\circ, \Delta\phi^{\mathrm{Mom}} = -5.3^\circ,$
$\Delta Z^{\mathrm{Pos}} = 1.5\ \mathrm{cm}, \Delta\phi^{\mathrm{Pos}} = -1.0^\circ, Z = -67.7\ \mathrm{cm})$

$\mathrm{e}^-(\mathrm{Mom} = 1.8\ \mathrm{GeV}, \Delta\theta^{\mathrm{Mom}} = 0.9^\circ, \Delta\phi^{\mathrm{Mom}} = -7.8^\circ,$
$\Delta Z^{\mathrm{Pos}} = 0.\ \mathrm{cm}, \Delta\phi^{\mathrm{Pos}} = -1.4^\circ, Z = -136.1\ \mathrm{cm})$





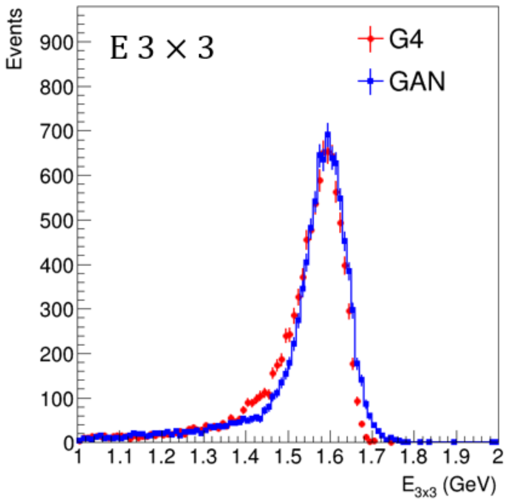


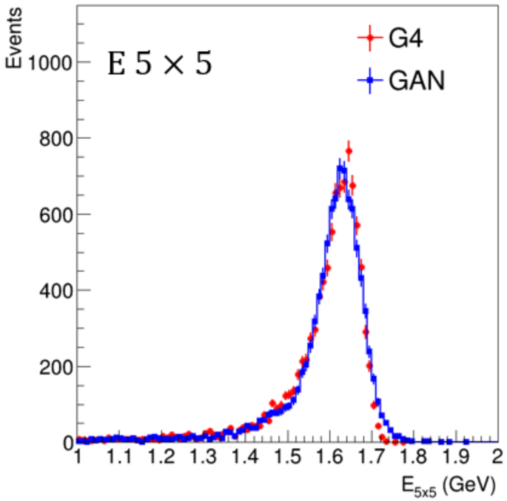
The BESIII case (performance)

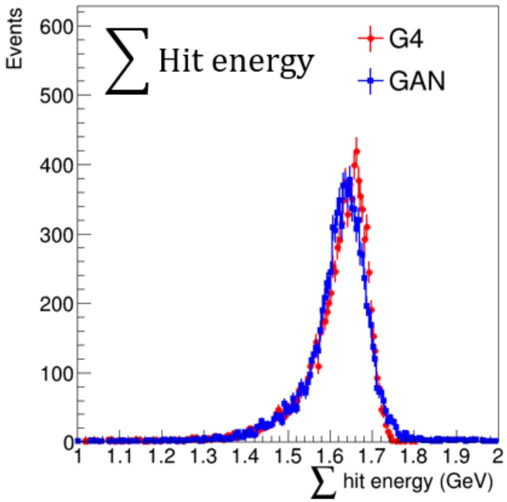
e^-

Dataset:

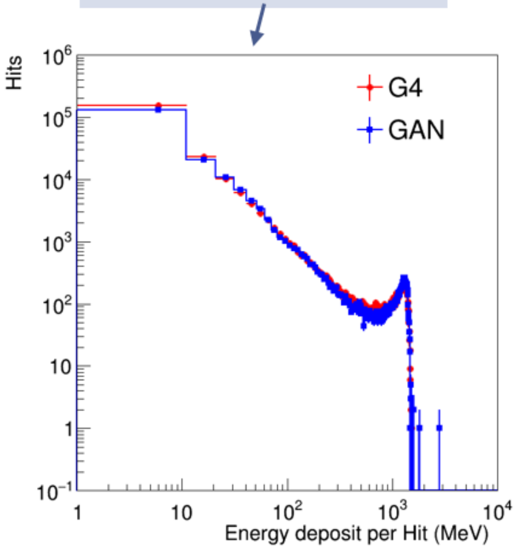
- MC Bhabha events





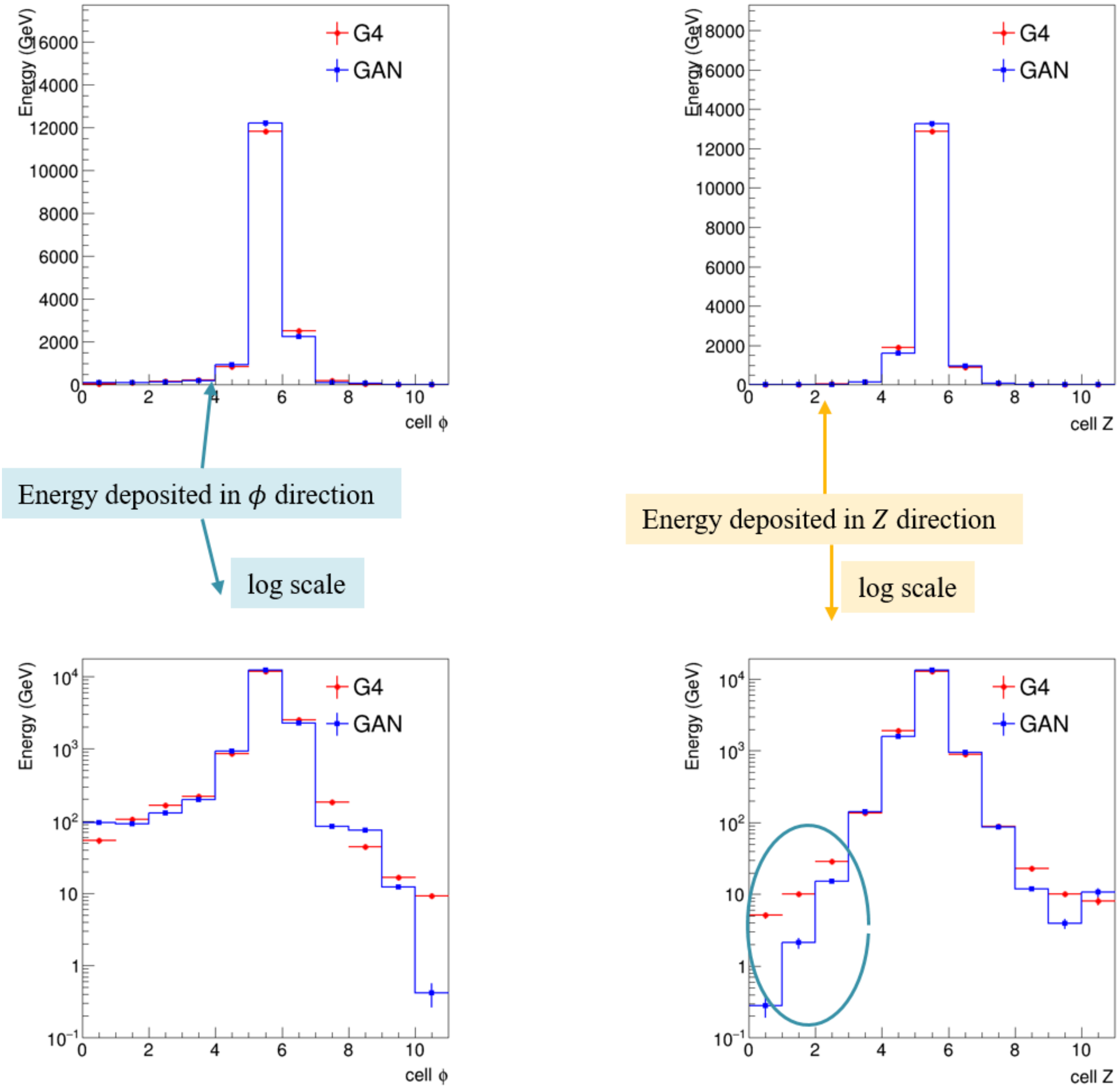


Energy of calorimeter cells



The BESIII case (performance)

e^-



Energy deposited in ϕ direction

log scale

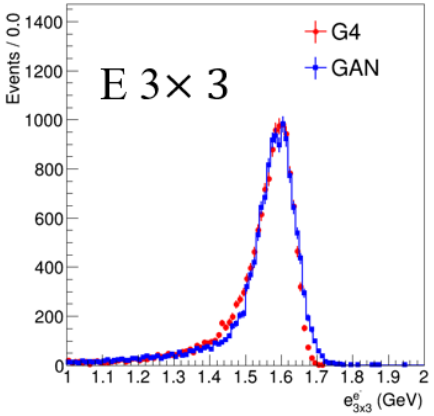
Energy deposited in Z direction

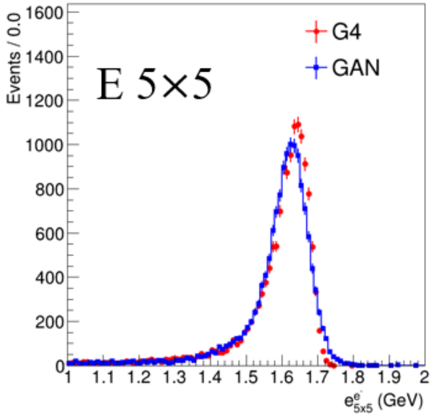
log scale

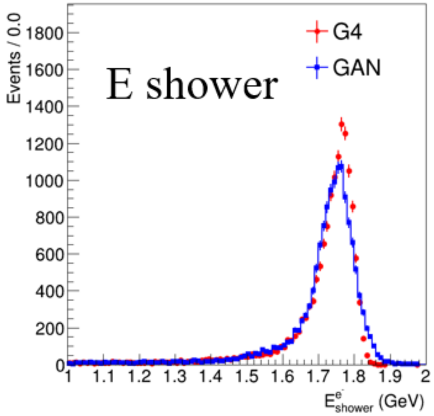
The BESIII case (performance)

e^-

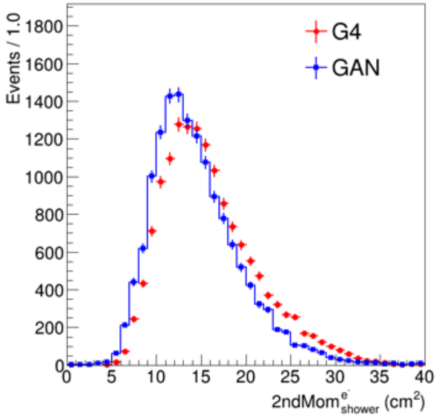
- Apply the GAN simulation in BESIII offline software

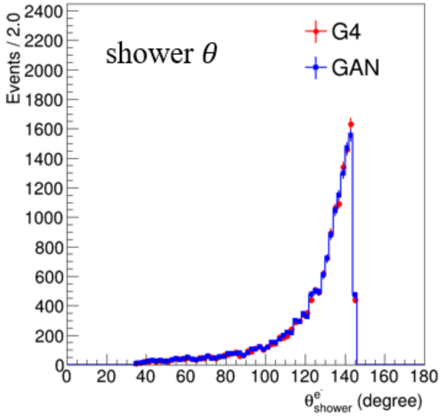


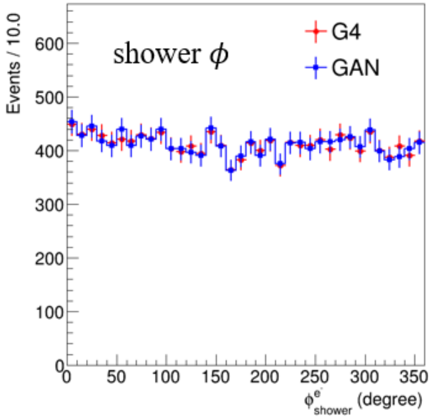




Shower 2nd mom

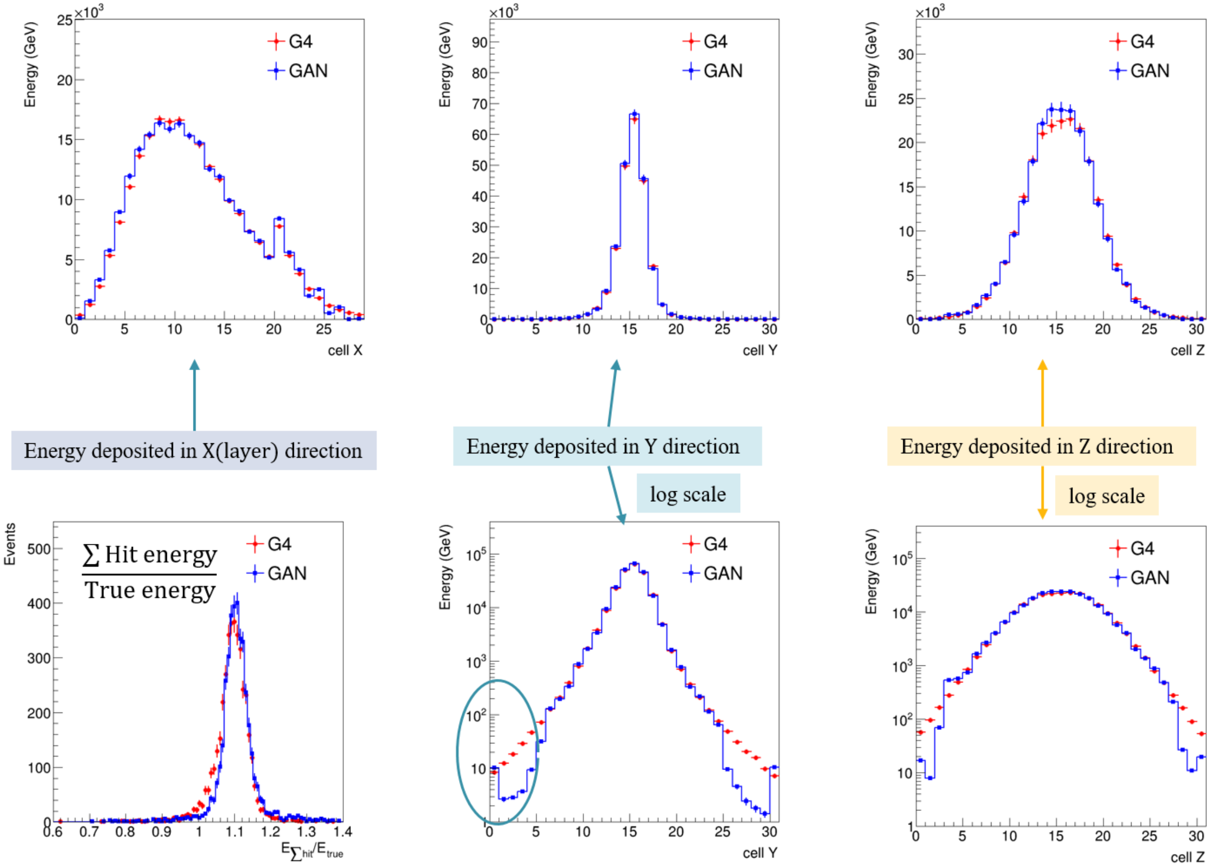






For CEPC experiment

- Apply the model for silicon-tungsten ECAL (29 layers). The model is extended from 2D to 3D (mainly replace 2D convolution operation by 3D convolution)



Energy deposited in X(layer) direction

Energy deposited in Y direction

log scale

Energy deposited in Z direction

log scale

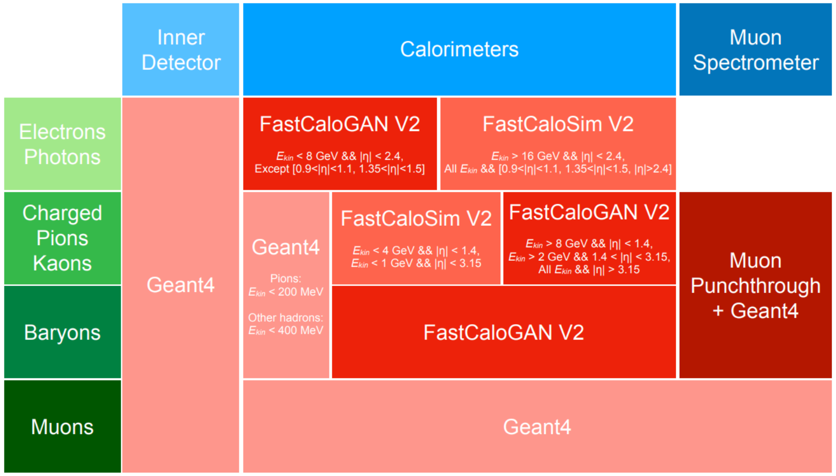
Dataset:

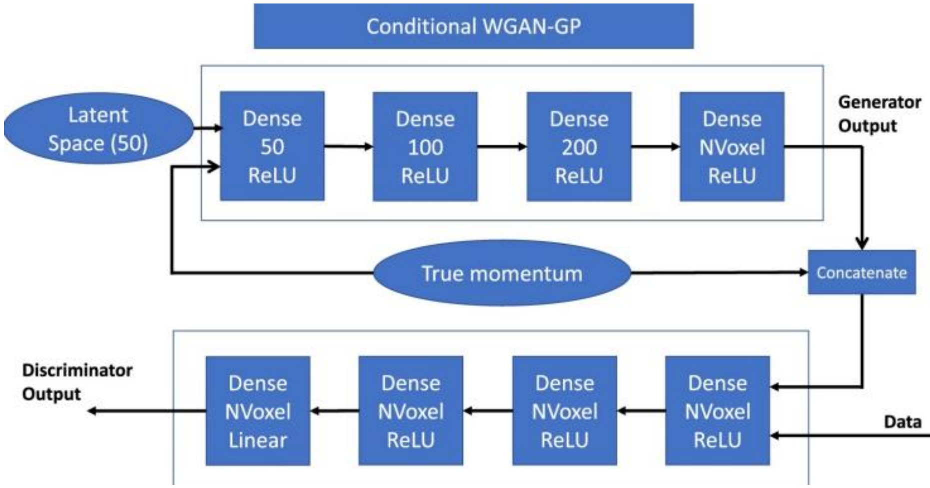
- photon showers in ECAL Barrel
- 31x31x29 voxels

The ATLAS case

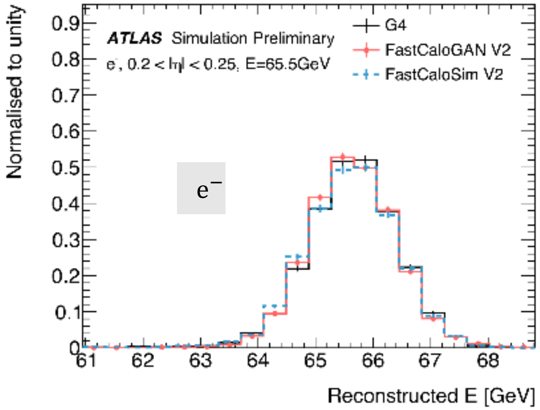
- AltFast3 (a detector response fast simulation system):
  - FastCaloGAN V2 (ML-based)
  - FastCaloSim V2 (parametrization-based)
  - Geant4 (limited to specific cases)
- FastCaloGAN:
  - WGANs trained for each particle type, for each | η | slice.
  - Conditioned on truth momentum
  - Total 300 GANs

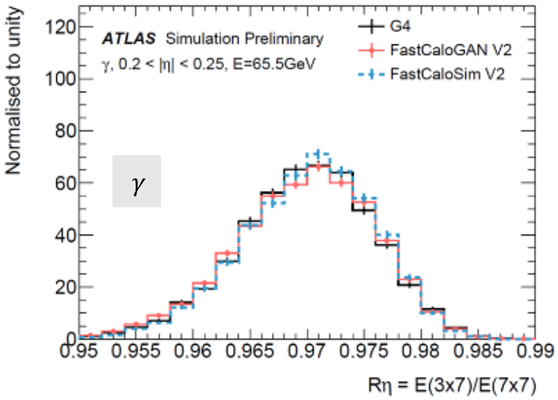
AtlFast3 Configuration for Run 3 SIM-2024-004

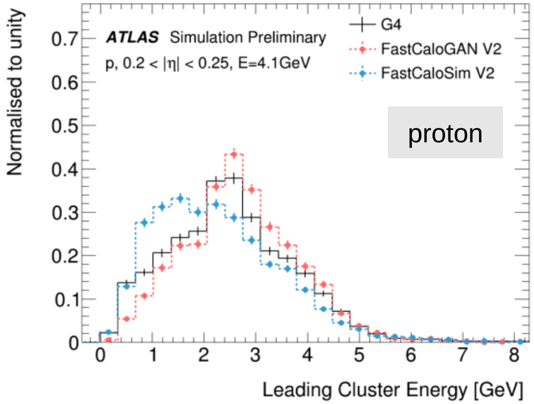


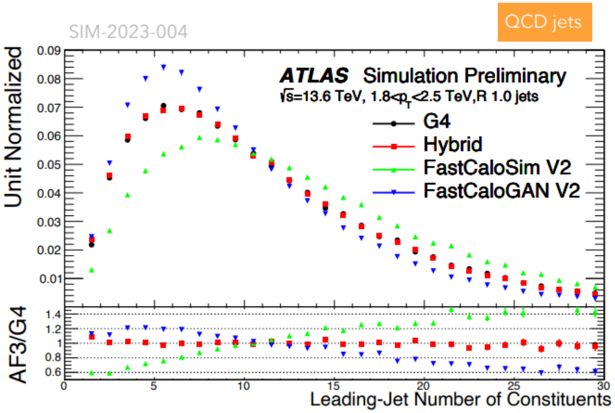


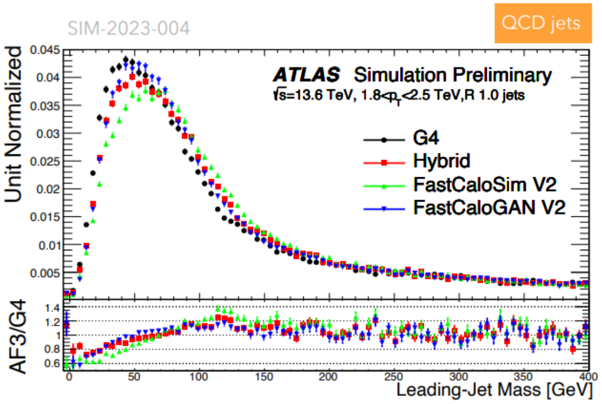
The ATLAS case (performance)

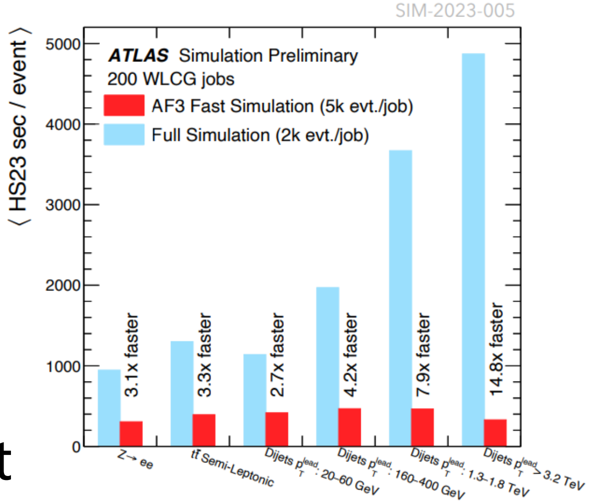






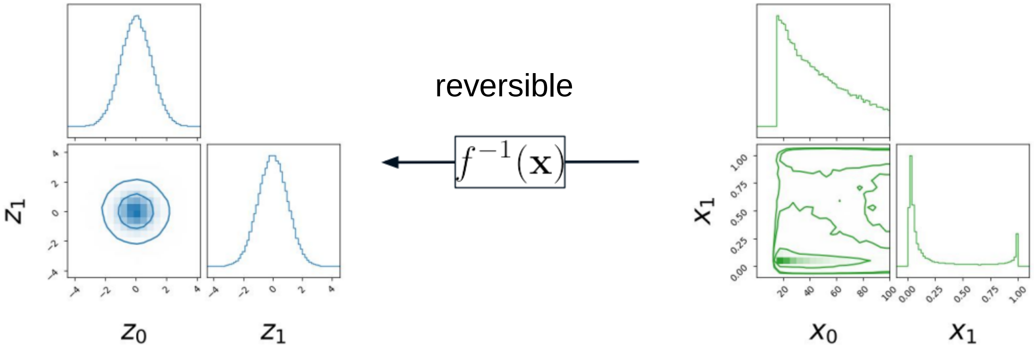






- 3 – 15 speed-up in simulation time with respect to Geant4, depending on the physics process
- Simulation time in AtlFast3 completely dominated by full simulation of the Inner Detector

Normalizing Flows



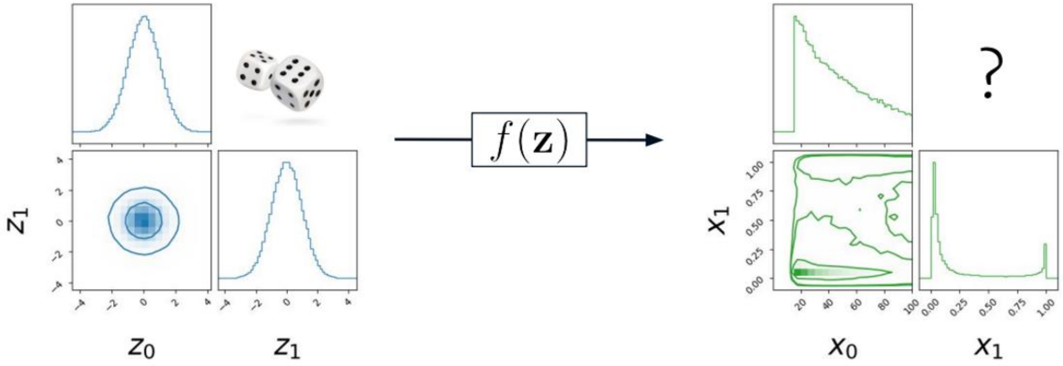
$$\mathbf{x} = f(\mathbf{z})$$

$$p_x(\mathbf{x}) = p_z(\mathbf{z}) \det \left| \frac{d\mathbf{z}}{d\mathbf{x}} \right|$$

$$\log(p_x(x)) = \log(p_z(f^{-1}(\mathbf{x}))) + \log\left(\det \mathbb{J}_{f^{-1}}(\mathbf{x})\right)$$

What we know

What we need



$\mathbf{Z}$

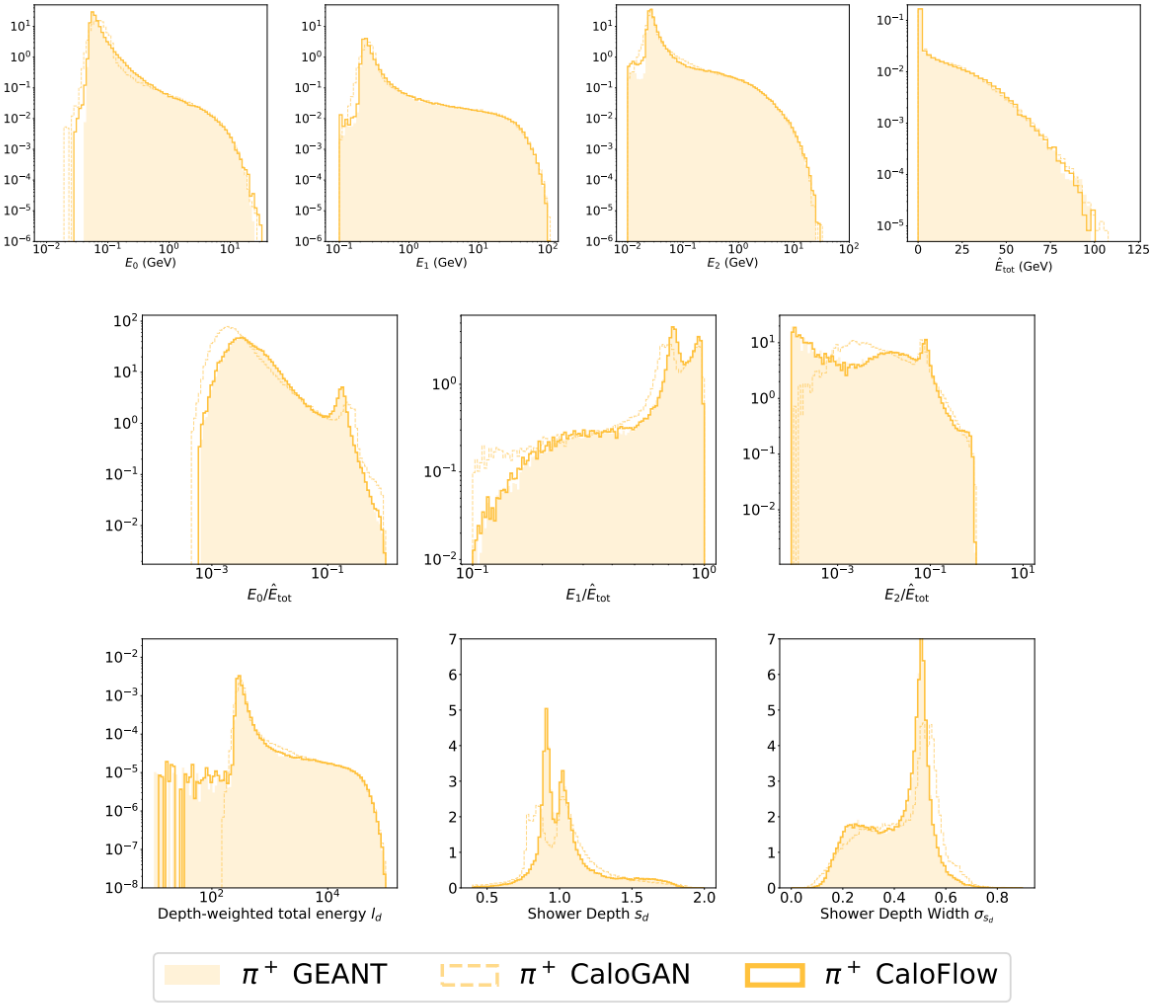
multi-dimensional gaussian

$\mathbf{X}$

FullSim data, pdf unknown!

CaloFlow

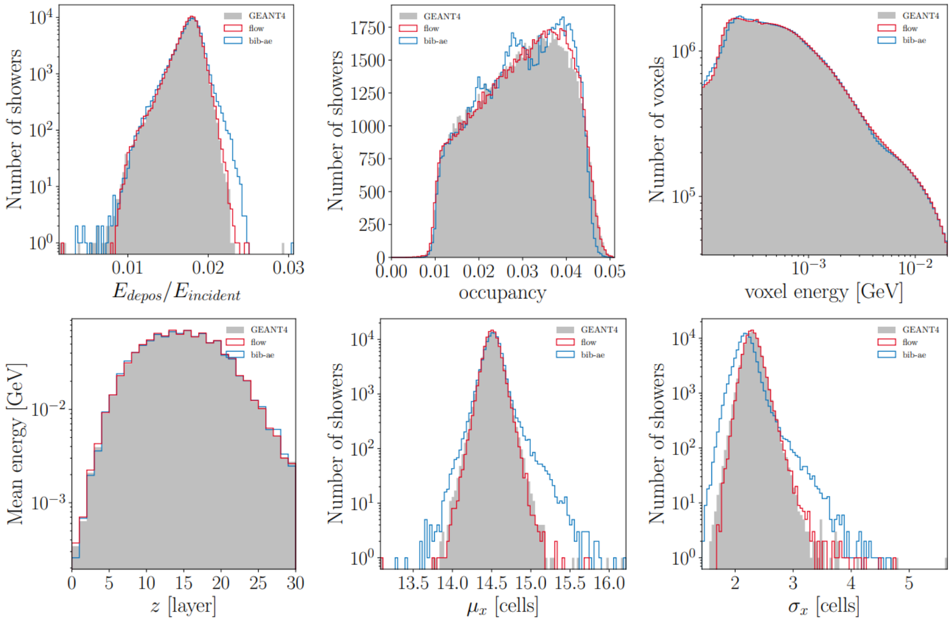
- The CaloFlow uses the same dataset as CaloGAN and shows better physics performance



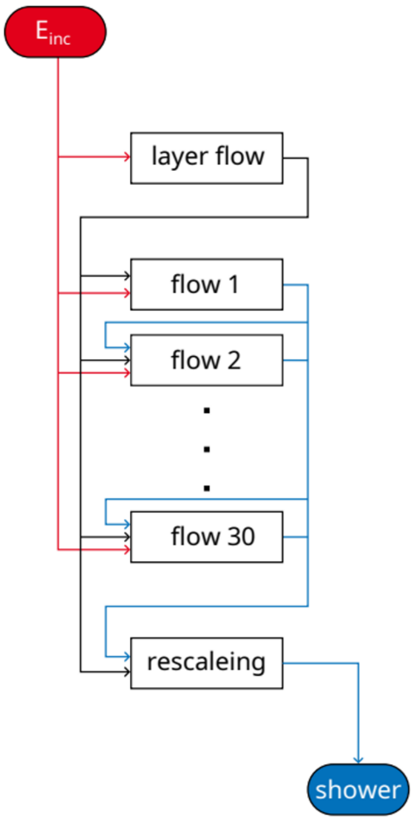
The ILC case

- Dataset:
  - photon showers in ECAL
  - 30x30x30 voxels

Preliminary Results



Architecture



Simulator	Hardware	Batch size	time [ms]	Speedup
GEANT4	CPU	1	4081.53 ± 169.92	×1.0
BIB-AE	CPU	1	102.25 ± 0.64	×40.0
		10	37.81 ± 0.13	×108.0
		100	48.51 ± 0.01	×84.1
		1000	48.19 ± 0.01	×84.7
Flow	CPU	1	1746.61 ± 64.50	x2.3
		10	392.61 ± 0.34	x10.4
		100	228.86 ± 7.09	x17.8
		1000	275.55 ± 3.01	x14.8
BIB-AE	GPU	1	74.22 ± 3.18	×42.5
		1000	0.249 ± 0.002	×16326.1
Flow	GPU	1	2471.07 ± 70.20	x1.7
		1000	3.39 ± 0.09	×1202.3

Link

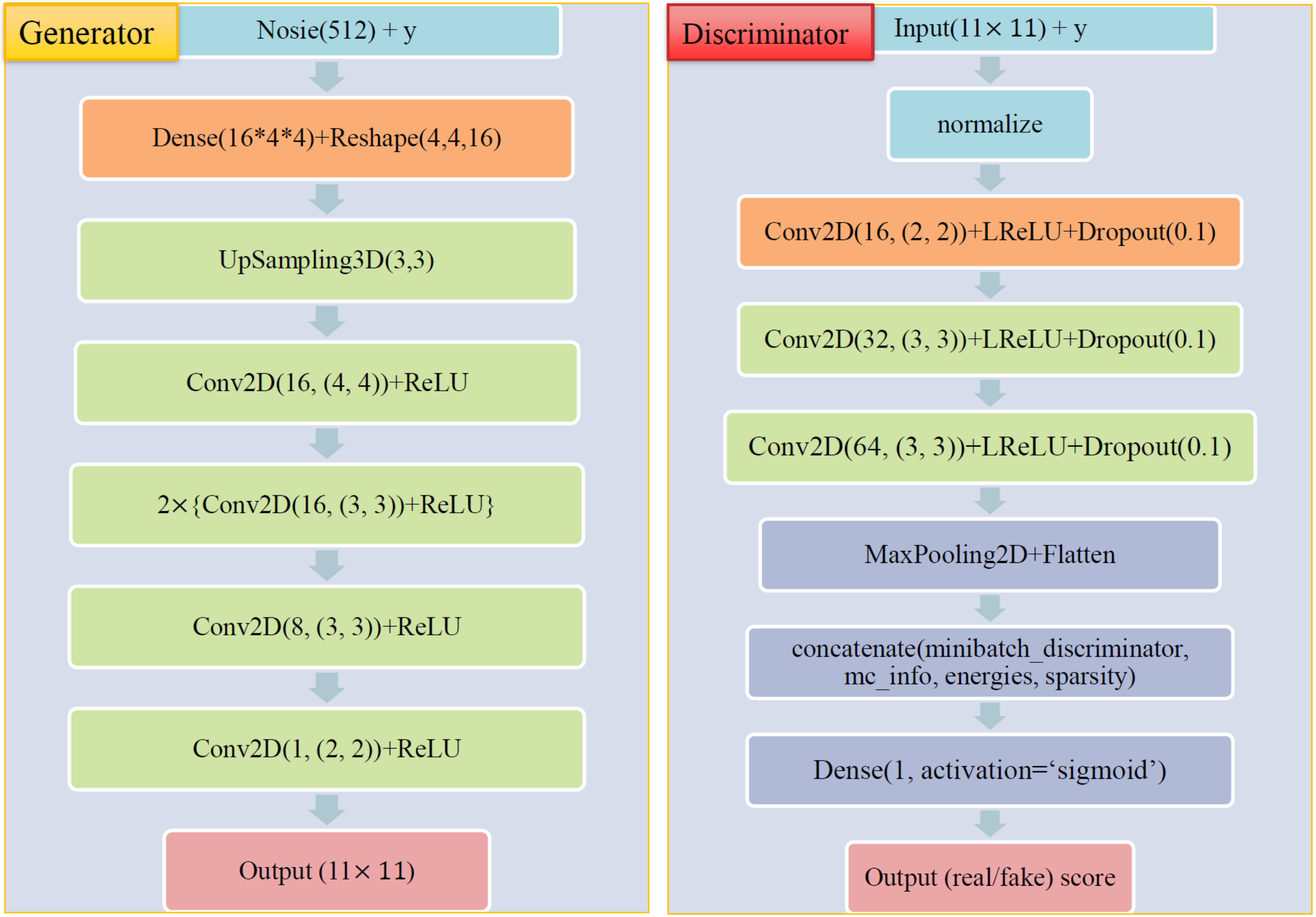
Summary

- ML based fast simulation calorimeter is widely studied in different experiments
- Many promising results and also challenges
- The field is in a rapid development stage, e.g. using the latest diffusion model (CaloDiffusion)
- Please stay tuned!

Thanks for your attention!

Backup

Model for BES3



Generator

Nosie(512) + y

Dense(16*4*4)+Reshape(4,4,16)

UpSampling3D(3,3)

Conv2D(16, (4, 4))+ReLU

2×{Conv2D(16, (3, 3))+ReLU}

Conv2D(8, (3, 3))+ReLU

Conv2D(1, (2, 2))+ReLU

Output (11× 11)

Discriminator

Input(11× 11) + y

normalize

Conv2D(16, (2, 2))+LReLU+Dropout(0.1)

Conv2D(32, (3, 3))+LReLU+Dropout(0.1)

Conv2D(64, (3, 3))+LReLU+Dropout(0.1)

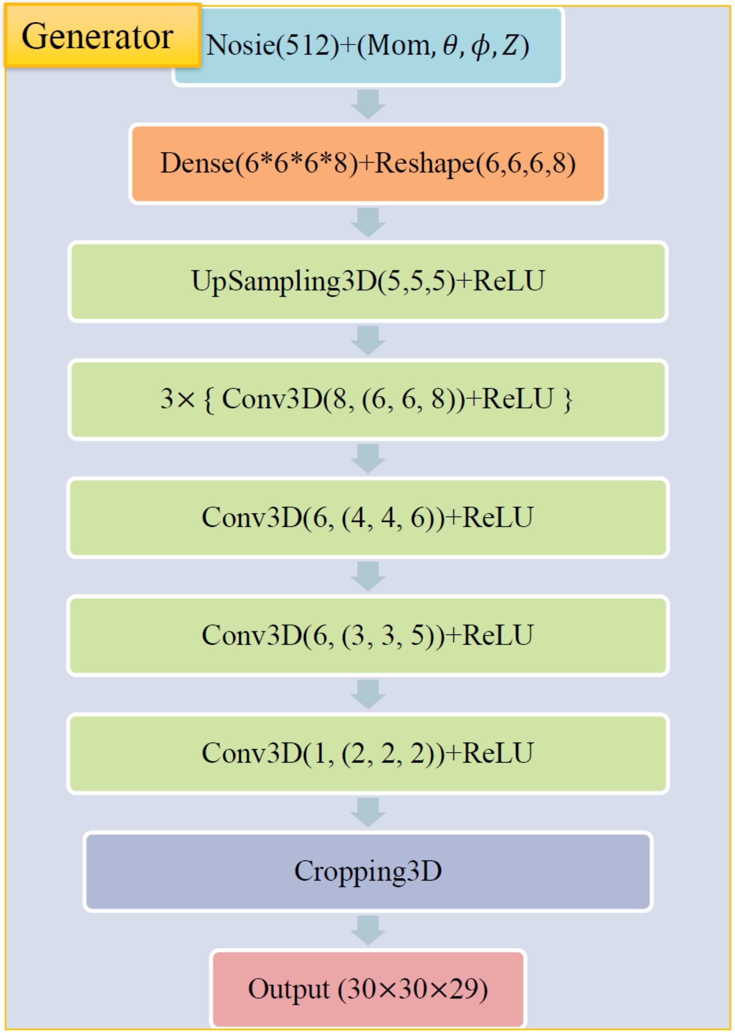
MaxPooling2D+Flatten

concatenate(minibatch_discriminator, mc_info, energies, sparsity)

Dense(1, activation=‘sigmoid’)

Output (real/fake) score

Model for CEPC



Generator

Nosie(512)+(Mom, θ, ϕ, Z)

Dense(6*6*6*8)+Reshape(6,6,6,8)

UpSampling3D(5,5,5)+ReLU

3× { Conv3D(8, (6, 6, 8))+ReLU }

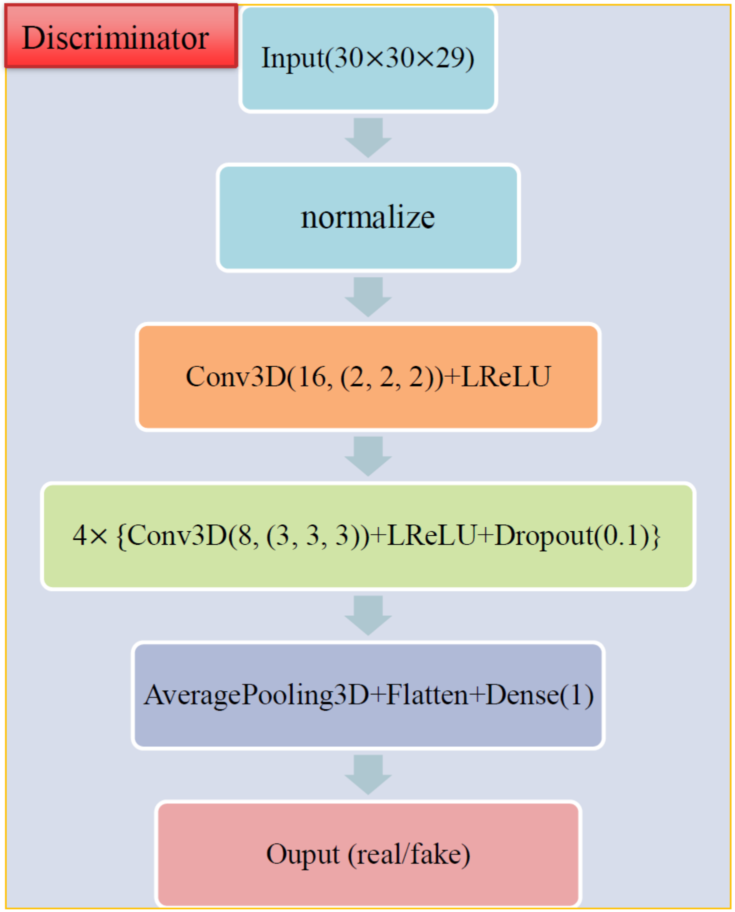
Conv3D(6, (4, 4, 6))+ReLU

Conv3D(6, (3, 3, 5))+ReLU

Conv3D(1, (2, 2, 2))+ReLU

Cropping3D

Output (30×30×29)



Discriminator

Input(30×30×29)

normalize

Conv3D(16, (2, 2, 2))+LReLU

4× {Conv3D(8, (3, 3, 3))+LReLU+Dropout(0.1)}

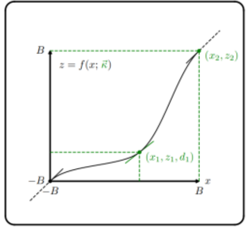
AveragePooling3D+Flatten+Dense(1)

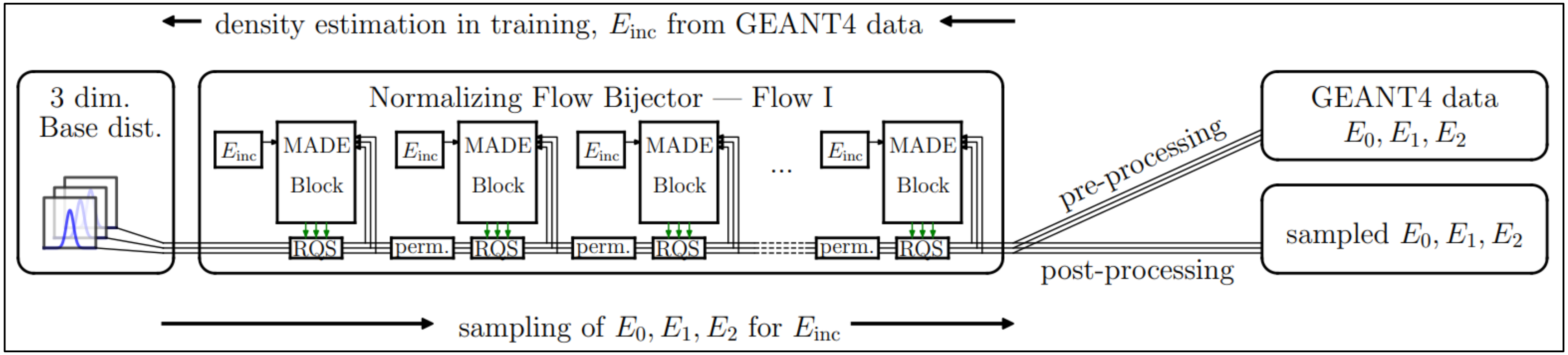
Ouput (real/fake)

CaloFlow

- The CaloFlow (2021) uses the same dataset as CaloGAN and shows much better physics performance

RQS Transformation




← density estimation in training, E_{inc} from GEANT4 data ←

3 dim. Base dist.

Normalizing Flow Bijector — Flow I

E_{inc} MADE Block RQS perm. E_{inc} MADE Block RQS perm. E_{inc} MADE Block RQS … perm. E_{inc} MADE Block RQS

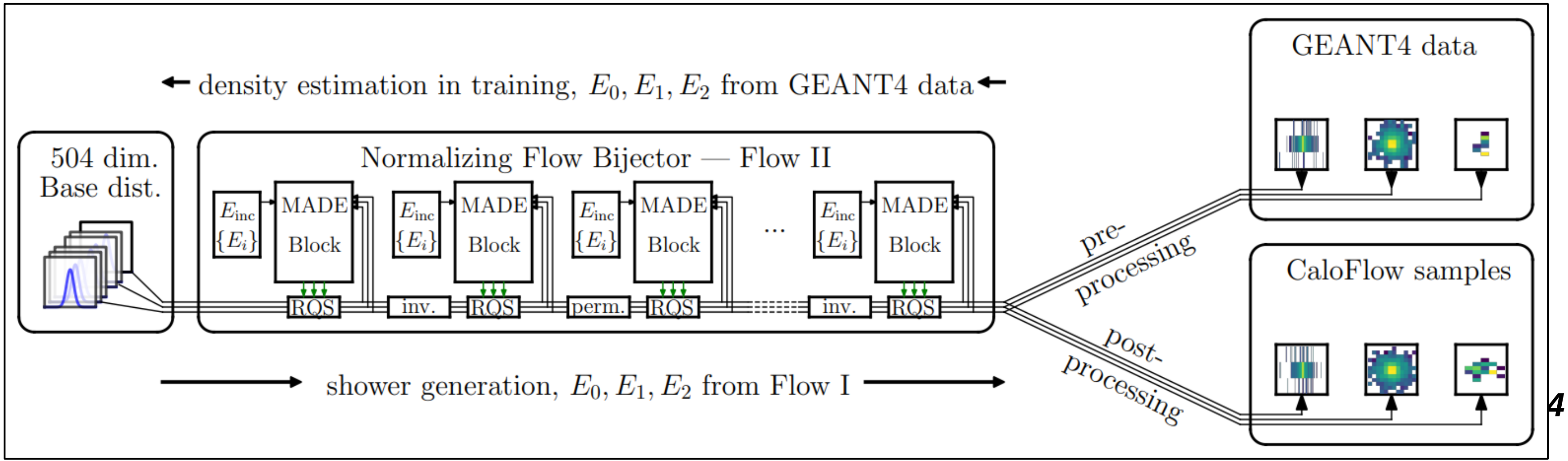
pre-processing

post-processing

GEANT4 data E_0, E_1, E_2

sampled E_0, E_1, E_2

→ sampling of E_0, E_1, E_2 for E_{inc} →


← density estimation in training, E_0, E_1, E_2 from GEANT4 data ←

504 dim. Base dist.

Normalizing Flow Bijector — Flow II

E_{inc} $\{E_i\}$ MADE Block RQS inv. E_{inc} $\{E_i\}$ MADE Block RQS perm. E_{inc} $\{E_i\}$ MADE Block RQS … inv. E_{inc} $\{E_i\}$ MADE Block RQS

pre-processing

post-processing

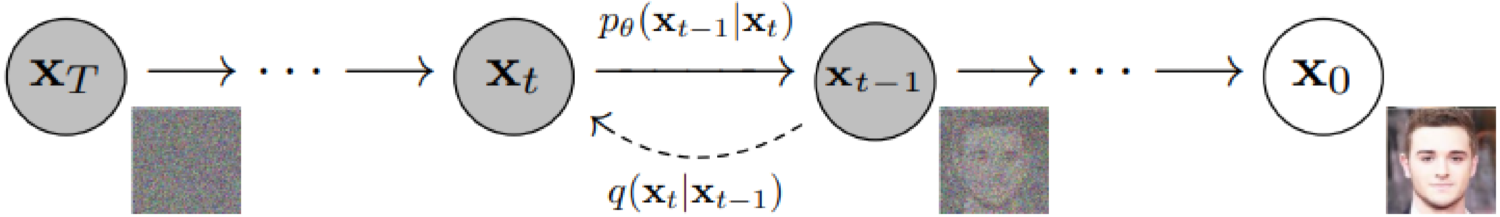
GEANT4 data

CaloFlow samples

→ shower generation, E_0, E_1, E_2 from Flow I →

Diffusion model

- The diffusion model is proposed in 2020



$\mathbf{x}_T \longrightarrow \cdots \longrightarrow \mathbf{x}_t \xrightarrow{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} \mathbf{x}_{t-1} \longrightarrow \cdots \longrightarrow \mathbf{x}_0$

$q(\mathbf{x}_t|\mathbf{x}_{t-1})$

- Diffusion process: $x_0 \rightarrow x_T$
  - Adding noise step by step, making $x_T \sim \mathcal{N}(0, \mathbf{I})$
- Train a model to invert the diffusion process
- When do simulation, start from $\mathcal{N}(0, \mathbf{I})$ and denoise it step by step using the trained model

CaloDiffusion

- [CaloDiffusion](#) (a fast calorimeter simulation method based on diffusion model)
- Dataset:
  - ATLAS-like geometry, 5 layer cylinder with irregular binning, 368 voxels
- Denoise model:
  - U-net architecture with 3D convolutions
  - Input: Noisy shower
  - Condition inputs: incident particle energy, diffusion step
  - Output: noise
- Good agreement with Geant4, some properties (e.g. total shower energy), can still be improved
- Generation time is slower than other ML approaches (still faster than Geant4)