Neural Networks for Cosmic Ray Simulations

Pranav Sampathkumar, IAP, KIT

11th KSETA Plenary Workshop 2024

Karlsruher Institut für Technologie

Overview

- Motivation
- Challenges
- Neural Networks
- Generative Neural Networks
- Point Cloud Models
- Our RNN based hybrid model
- Neural networks for radio pulses
- Conclusions

Motivation

The Problem

- Monte Carlo (MC): **slowest part** of many physics pipelines
- **Scales badly** for high energies

Context

- Current Approximations: **Theory based**
- **Forward physics** is important

Problem statement

- Can **neural networks** be used for **data* driven** approximations ?

* data refers to output from simulations

The Challenges

Performance

- **Faster** than Monte Carlo Simulations
- **Better scaling** than traditional approximations

Fidelity

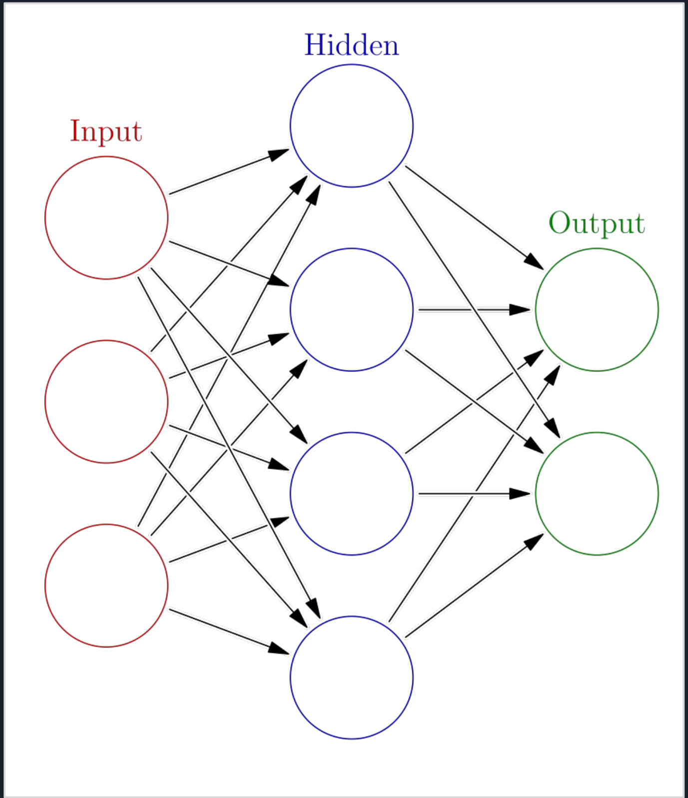
- **Sensible physics** approximations
- **Shouldn't affect** existing pipelines.

Physics

- **Different challenges** based on context
- Eg, Cosmic ray showers are **larger**

Neural Networks

- A fully connected network is a **collection of matrices** with a nonlinear function
- N layered network has N-1 matrices.
- Use an optimization algorithm

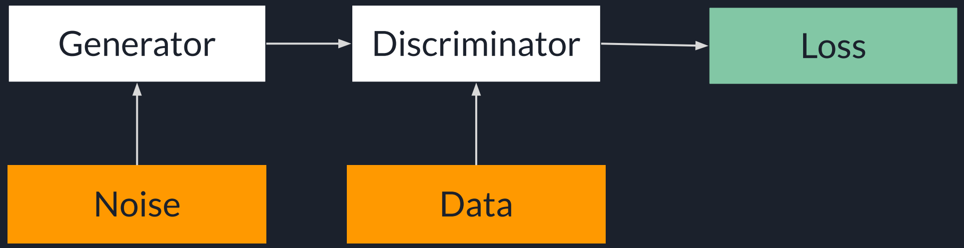


Img source:Wikipedia

Overview

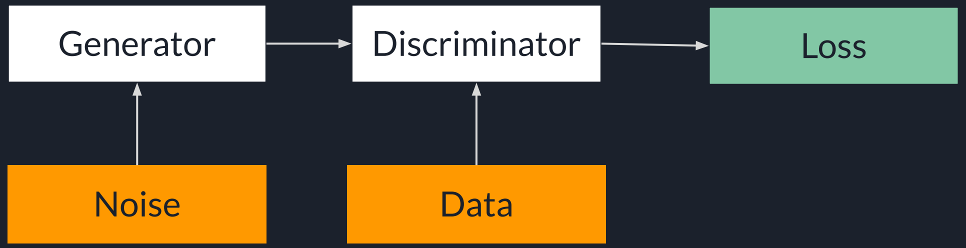
- Motivation
- Challenges
- Neural Networks
- Generative Neural Networks
- Point Cloud Models
- Our RNN based hybrid model
- Neural networks for radio pulses
- Conclusions

Generative Adversarial Neural Networks



- Various types of GANs
- Based on loss functions and architectures

Generative Adversarial Neural Networks

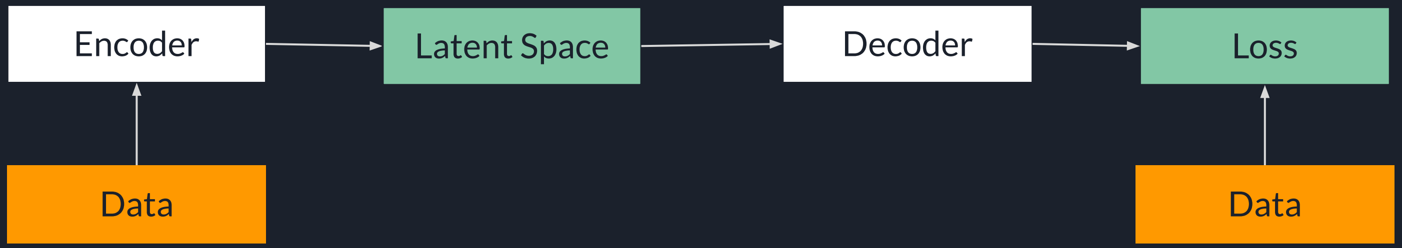


- **Various types** of GANs
- Based on **loss** functions and **architectures**

MLEGs	Data Source	Detector Effect	Reaction/Experiment	ML Model
[Hashemi *et al.*, 2019]	Pythia8	DELPHES + pile-up effects	$Z \to \mu^+\mu^-$	regular GAN
[Otten *et al.*, 2019]	MadGraph5 aMC@NLO	DELPHES3	$e^+e^- \to Z \to l^+l^-$, $pp \to t\bar{t}$	VAE
[Butter *et al.*, 2019]	MadGraph5 aMC@NLO		$pp \to t\bar{t} \to (bq\bar{q}')(\bar{b}\bar{q}q')$	MMD-GAN
[Di Sipio *et al.*, 2019]	MadGraph5, Pythia8	DELPHES + FASTJET	$2 \to 2$ parton scattering	GAN+CNN
[Ahdida *et al.*, 2019]	Pythia8 + GEANT4		Search for Hidden Particles (SHiP) experiment	regular GAN
[Alanazi *et al.*, 2020b] [Velasco *et al.*, 2020]	Pythia8		electron-proton scattering	MMD-WGAN-GP, cGAN
[Martnez *et al.*, 2020]	Pythia8	DELPHES particle-flow	proton collision	GAN, cGAN
[Gao *et al.*, 2020]	Sherpa		$pp \to W/Z + n$ jets	NF
[Howard *et al.*, 2021]	MadGraph5 + Pythia8	DELPHES	$Z \to e^+e^-$	SWAE
[Choi and Lim, 2021]	MadGraph5 + Pythia8	DELPHES	$pp \to b\bar{b}\gamma\gamma$	WGAN-GP

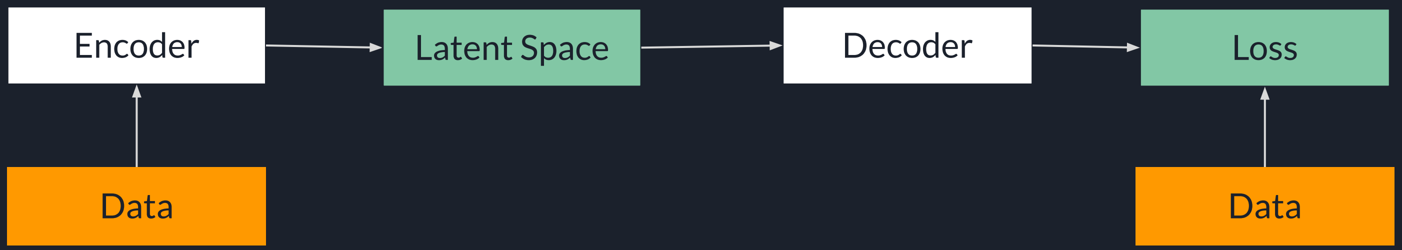
Table 1: List of existing MLEGs.

Autoencoders

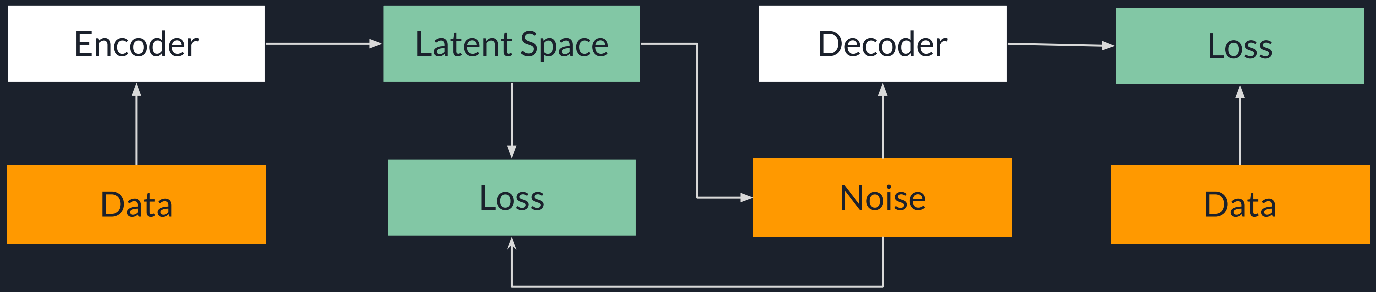


- **Encoder**: Compresses data
- **Decoder**: Extracts data
- **Latent Space**: Compressed data

Autoencoders



Variational Autoencoders



Information Distillation GAN

ID - GAN

- **Mix** auto-encoders and GANs
- **Distill** the necessary **information**
- **Encode first** and then **use it with noise** to create new data
- **Generator** does the creation

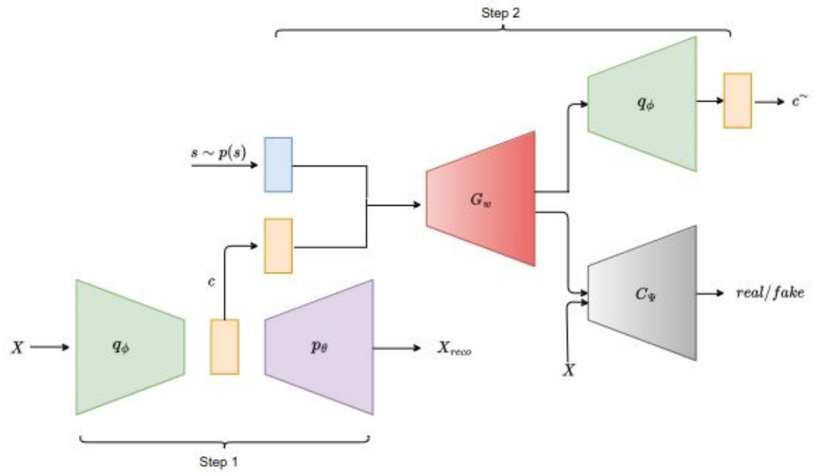


Figure 6.1.: It consists an encoder (q_ϕ), a decoder (p_θ), a generator (G_ω) and a discriminato C_ψ. The training of the model is divided into the training of the VAE part and training of the GAN part. The noise vector consists of two different variables s and c.

Source: PhD Thesis, Jubna Irakkathil Jabbar, ETP, KIT

Bounded-Information Bottleneck Autoencoder

BIB-AE

- **Mix** auto-encoders and GANs
- **Distill** the necessary **information**
- **Encode first** and then **use it with noise** to create new data
- Additional **critics** for physics
- **Decoder** does the creation

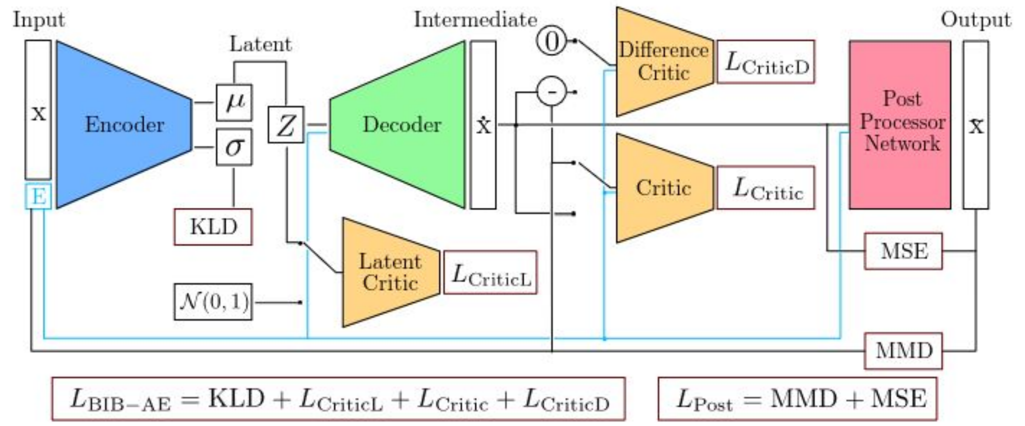


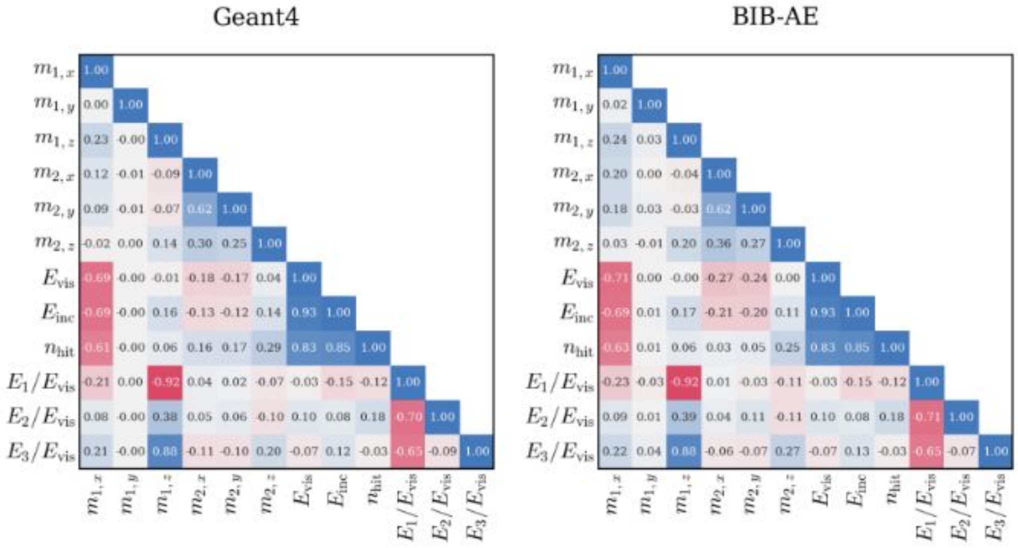
Figure 3. Schematic illustration of our BIBAE setup including the Post Processor Network in the final step.

Source: Hadrons, Better, Faster, Stronger (arXiv:2112.09709v1)

Information GAN

BIB-AE

- **Mix** auto-encoders and GANs
- **Distill** the necessary **information**
- **Encode first** and then **use it with noise** to create new data
- Additional **critics** for physics
- **Decoder** does the creation



Hardware	Simulator	Time / Shower [ms]	Speed-up
CPU	GEANT4	2684 ± 125	$\times 1$
	WGAN	47.923 ± 0.089	$\times 56$
	BIB-AE	350.824 ± 0.574	$\times 8$
GPU	WGAN	0.264 ± 0.002	$\times 10167$
	BIB-AE	2.051 ± 0.005	$\times 1309$

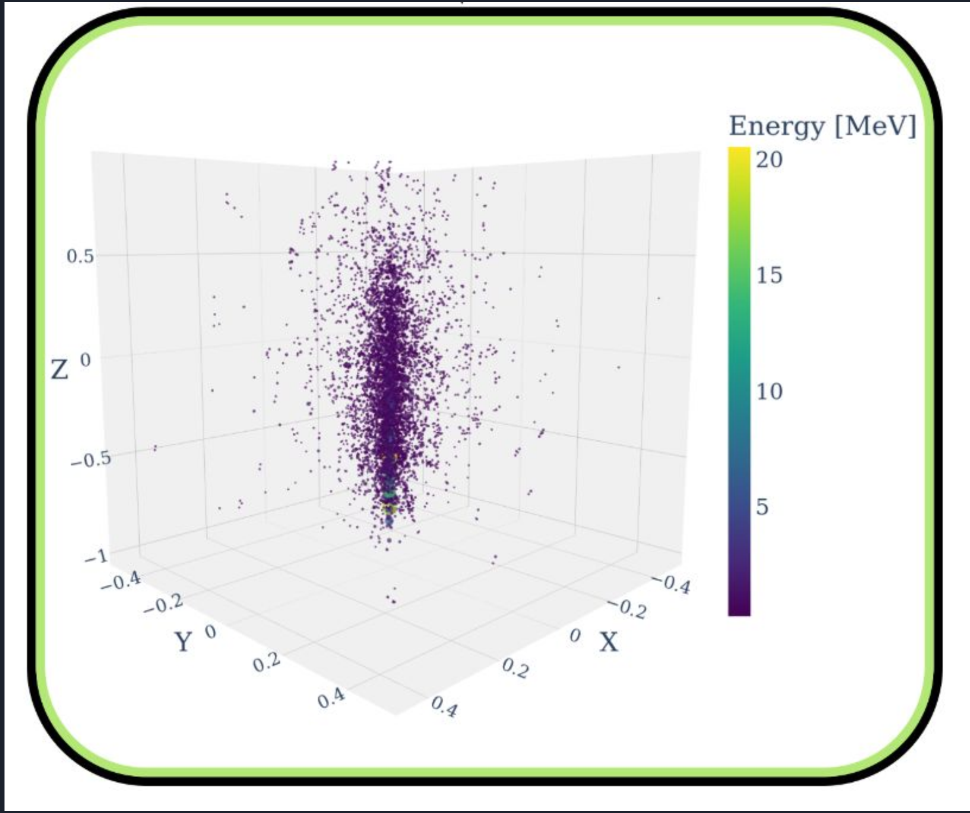
Source: Hadrons, Better, Faster, Stronger (arXiv:2112.09709v1)

Overview

- Motivation
- Challenges
- Neural Networks
- Generative Neural Networks
- Point Cloud Models
- Our RNN based hybrid model
- Neural networks for radio pulses
- Conclusions

Point Clouds

- Be **independent** of **grid**
- Use **unordered list of particles** in **momentum space**
- Helps the **sparsity** of the image
- **Directly generate** the coordinates of N points



Img source:CaloClouds (arXiv:2305.04847v2)

GANs on Point Clouds

A **swap between point clouds** doesn't change the information

GAN architectures were developed which account for this swap.

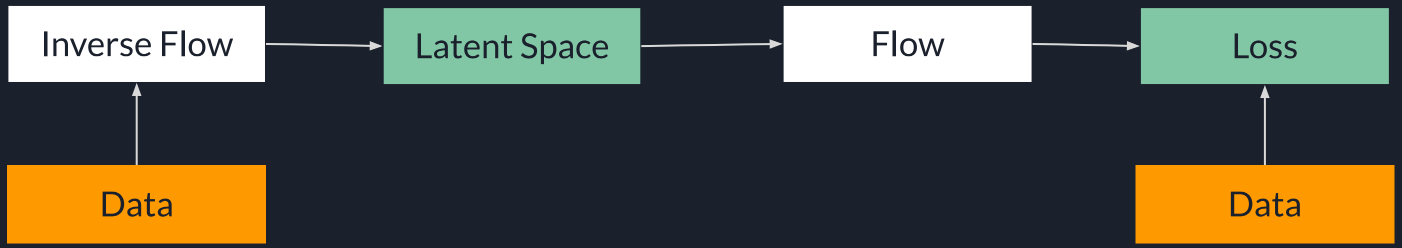
- EPiC-GAN: Equivariant Point Cloud GANs
- MP-GAN: Message passing GANs

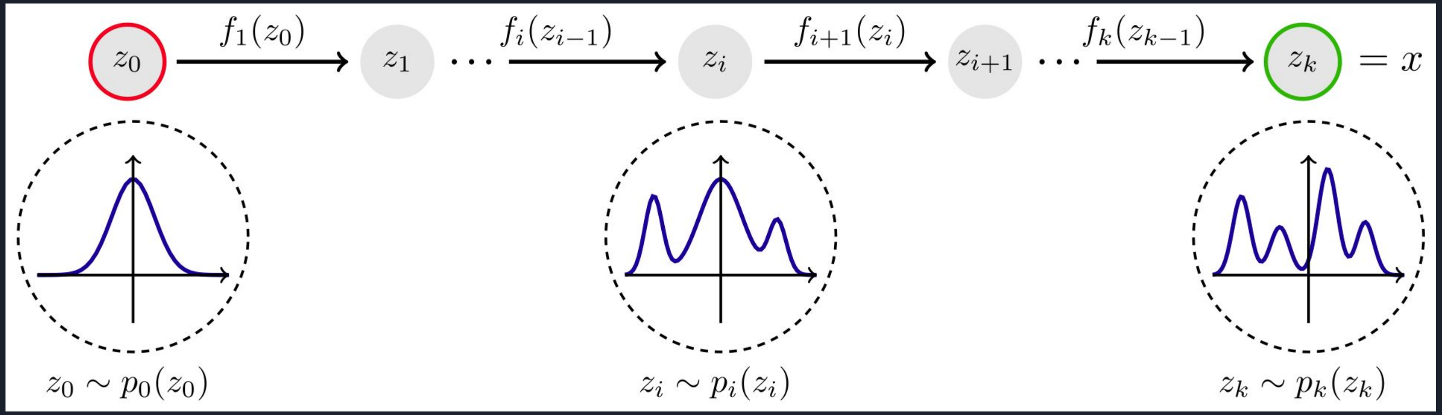
Jet class	Model	W_1^{M} ($\times10^{-3}$)	W_1^{P} ($\times10^{-3}$)	W_1^{EFP} ($\times10^{-5}$)	FPND
Gluon	Truth	0.3 ± 0.1	0.3 ± 0.1	0.3 ± 0.3	0.07 ± 0.01
	MP-GAN	0.5 ± 0.1	$\mathbf{1.3 \pm 0.1}$	0.6 ± 0.3	$\mathbf{0.13 \pm 0.02}$
	EPiC-GAN	$\mathbf{0.3 \pm 0.1}$	1.6 ± 0.2	$\mathbf{0.4 \pm 0.2}$	1.01 ± 0.07
Light quark	Truth	0.3 ± 0.1	0.3 ± 0.1	0.3 ± 0.3	0.02 ± 0.01
	MP-GAN	$\mathbf{0.5 \pm 0.1}$	4.9 ± 0.3	$\mathbf{0.7 \pm 0.4}$	$\mathbf{0.36 \pm 0.02}$
	EPiC-GAN	$\mathbf{0.5 \pm 0.1}$	$\mathbf{4.0 \pm 0.4}$	0.8 ± 0.4	0.43 ± 0.03
Top	Truth	0.2 ± 0.1	0.3 ± 0.1	0.6 ± 0.5	0.02 ± 0.01
	MP-GAN	$\mathbf{0.5 \pm 0.1}$	2.4 ± 0.2	$\mathbf{1.0 \pm 0.7}$	0.35 ± 0.04
	EPiC-GAN	$\mathbf{0.5 \pm 0.1}$	$\mathbf{2.1 \pm 0.1}$	1.7 ± 0.3	$\mathbf{0.31 \pm 0.03}$

Table 2: Evaluation scores for the JetNet30 dataset. The truth values are a comparison between the test and training set, which reflect the size of statistical fluctuations. The MP-GAN scores were calculated with the trained models from Ref. [24] using the same statistics as the EPiC-GAN. Lower is better for all scores.

Ref:arXiv:2301.08128v3

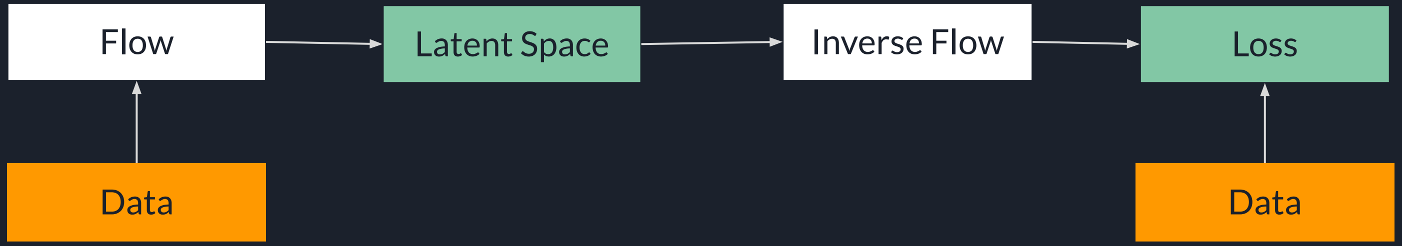
Flow Based Models





Img source:Wikipedia

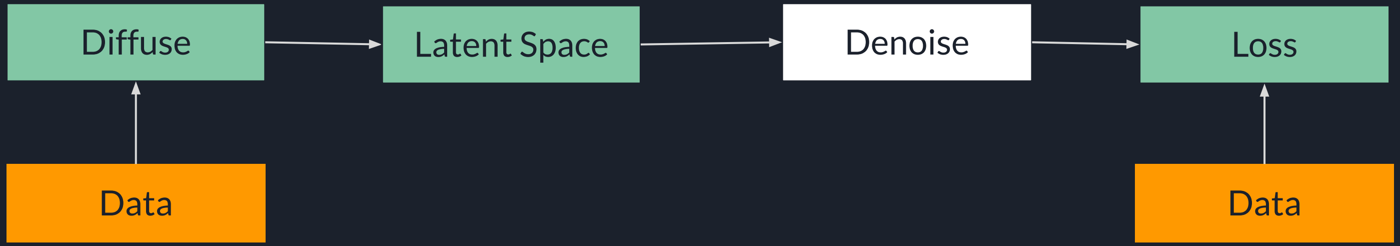
Flow Based Models

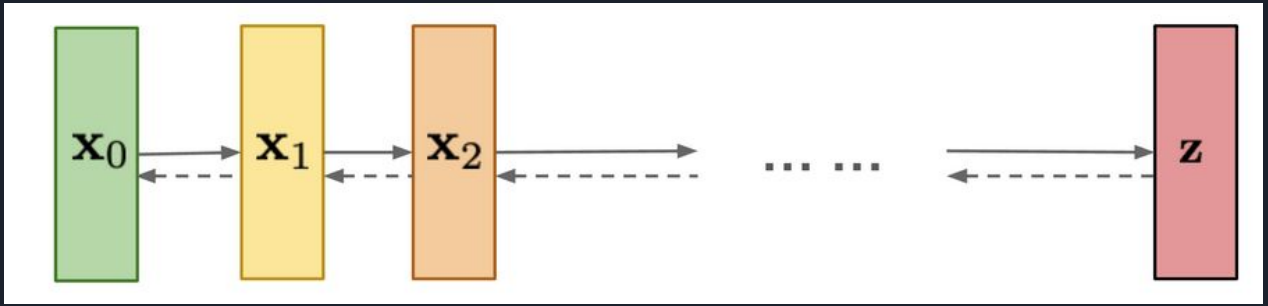


Simulator	Hardware	Batch size	10 – 100 GeV [ms]	Speedup
GEANT4 (30 × 30 × 30)	CPU	/	4081.53 ±169.92	/
L2LFLOWS (30 × 10 × 10)	CPU	1	19 617.24 ±894.08	×0.2
		10	3130.25 ±104.74	×1.3
		100	1395.52 ± 26.55	×2.9
		1000	1338.13 ± 24.03	×3.1

Img source: L2LFlows, arXiv:2302.11594v2

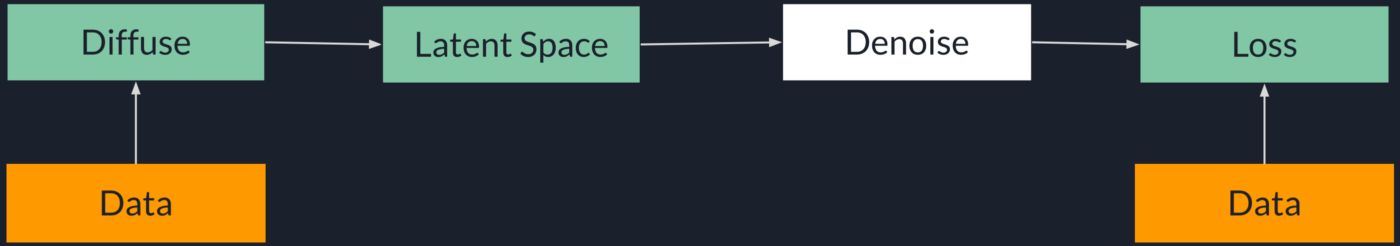
Diffusion Models





Img source: L2LFlows, arXiv:2302.11594v2

Diffusion Models



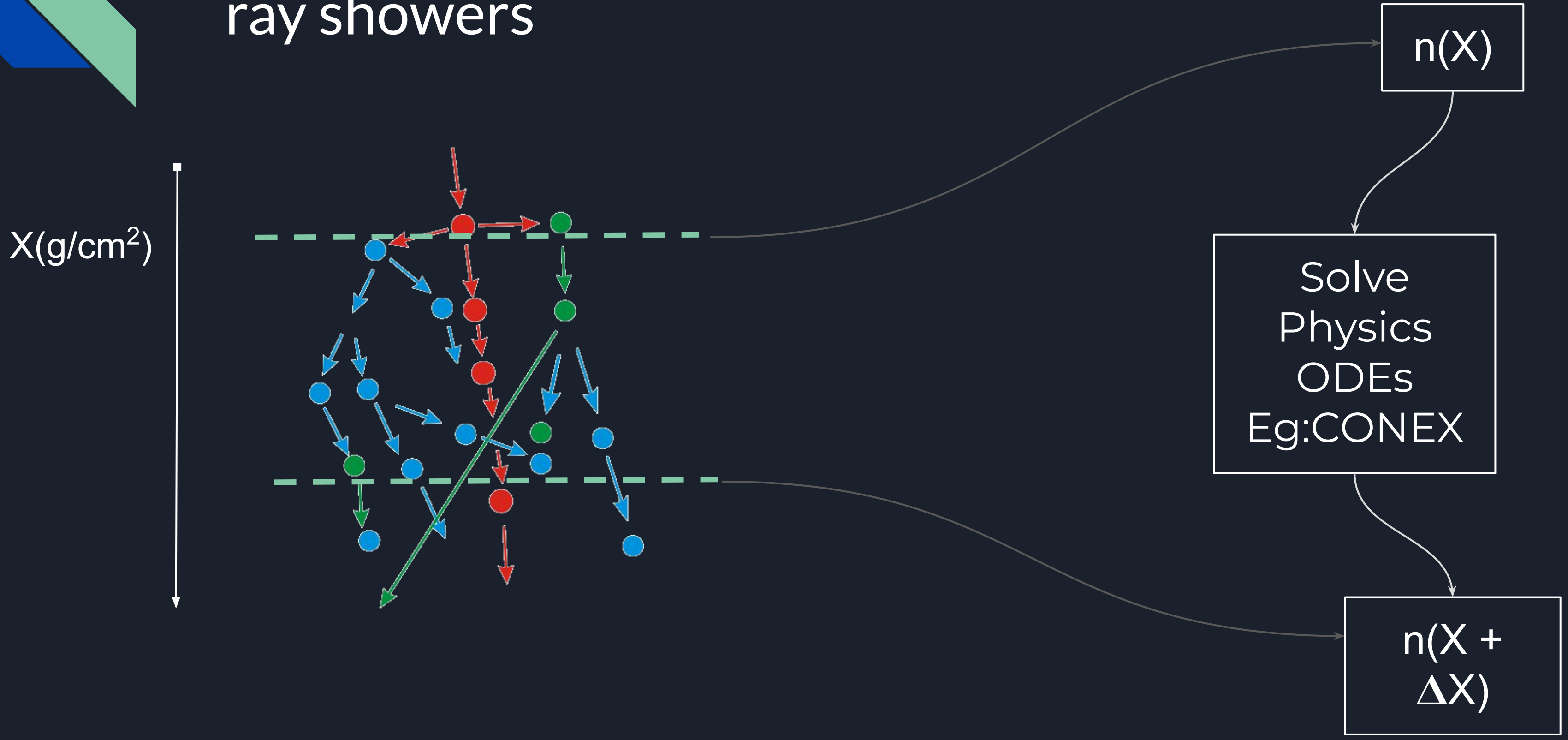
Hardware	Simulator	NFE	Batch Size	Time / Shower [ms]	Speed-up
CPU	Geant4			3914.80 ± 74.09	$\times 1$
	CaloClouds	100	1	3146.71 ± 31.66	$\times 1.2$
	CaloClouds II	25	1	651.68 ± 4.21	$\times 6.0$
	CaloClouds II (CM)	1	1	84.35 ± 0.22	$\times 46$
GPU	CaloClouds	100	64	24.91 ± 0.72	$\times 157$
	CaloClouds II	25	64	6.12 ± 0.13	$\times 640$
	CaloClouds II (CM)	1	64	2.09 ± 0.13	$\times 1873$

Img source: L2LFlows, arXiv:2302.11594v2

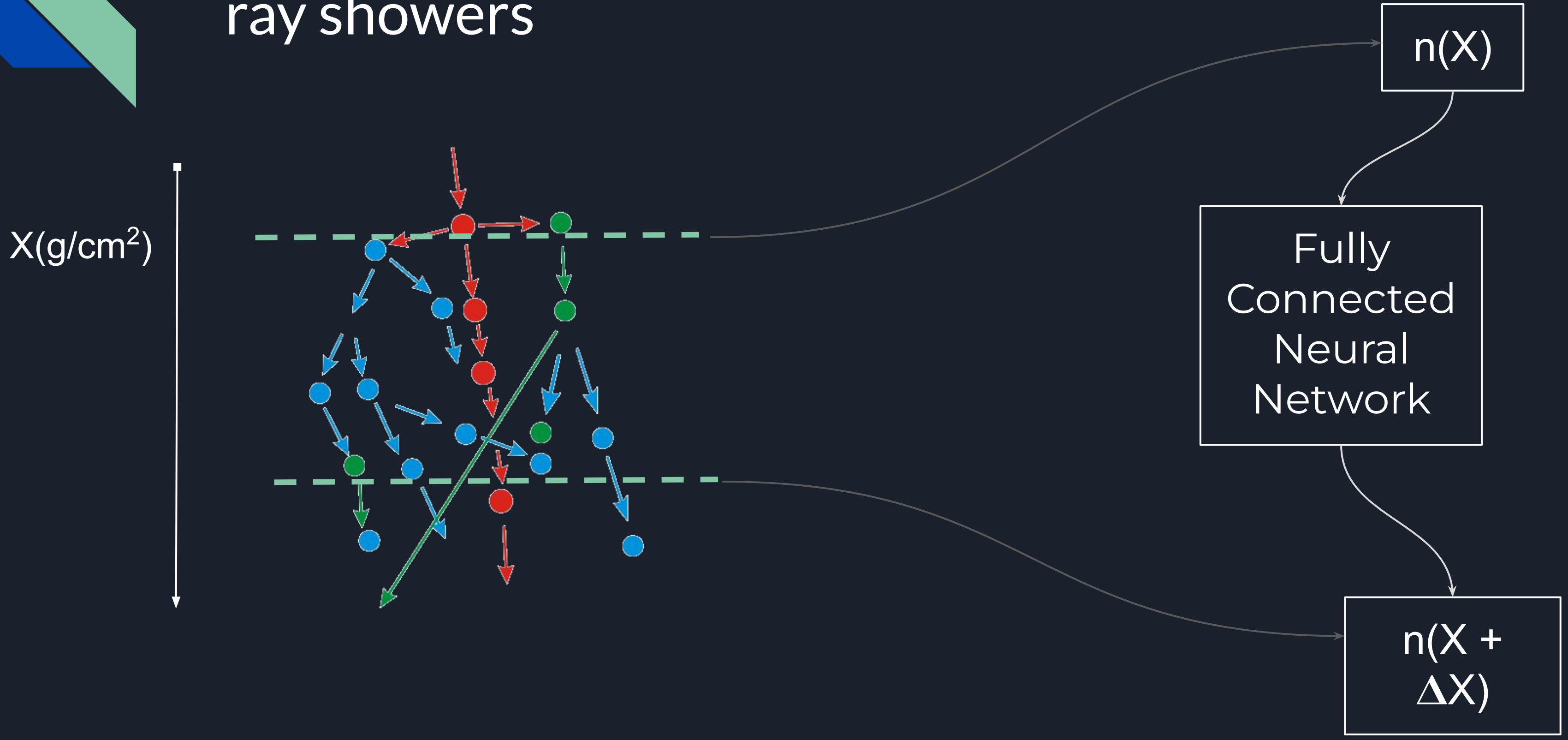
Overview

- Motivation
- Challenges
- Neural Networks
- Generative Neural Networks
- Point Cloud Models
- Our RNN based hybrid model
- Neural networks for radio pulses
- Conclusions

Sequential model for cosmic ray showers

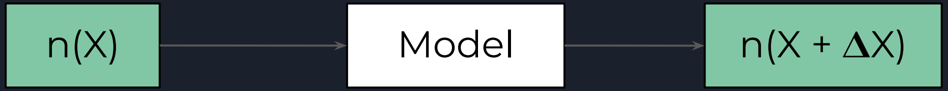


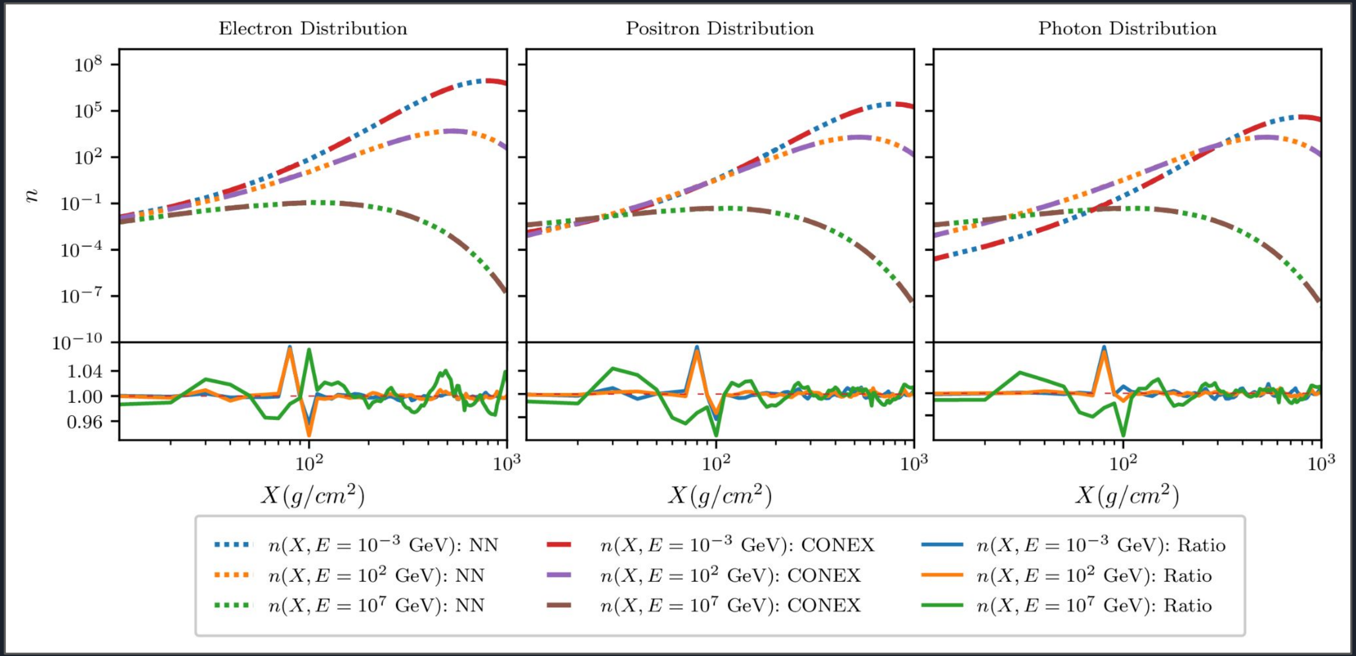
Sequential model for cosmic ray showers



- **Approximate a single step** in shower generation using a neural network

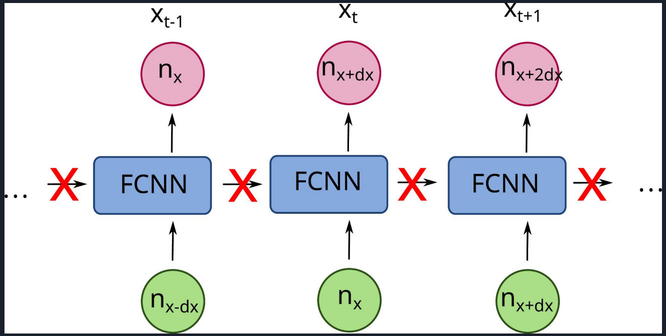
- Takes distributions at height X and gives distributions at height X + ΔX

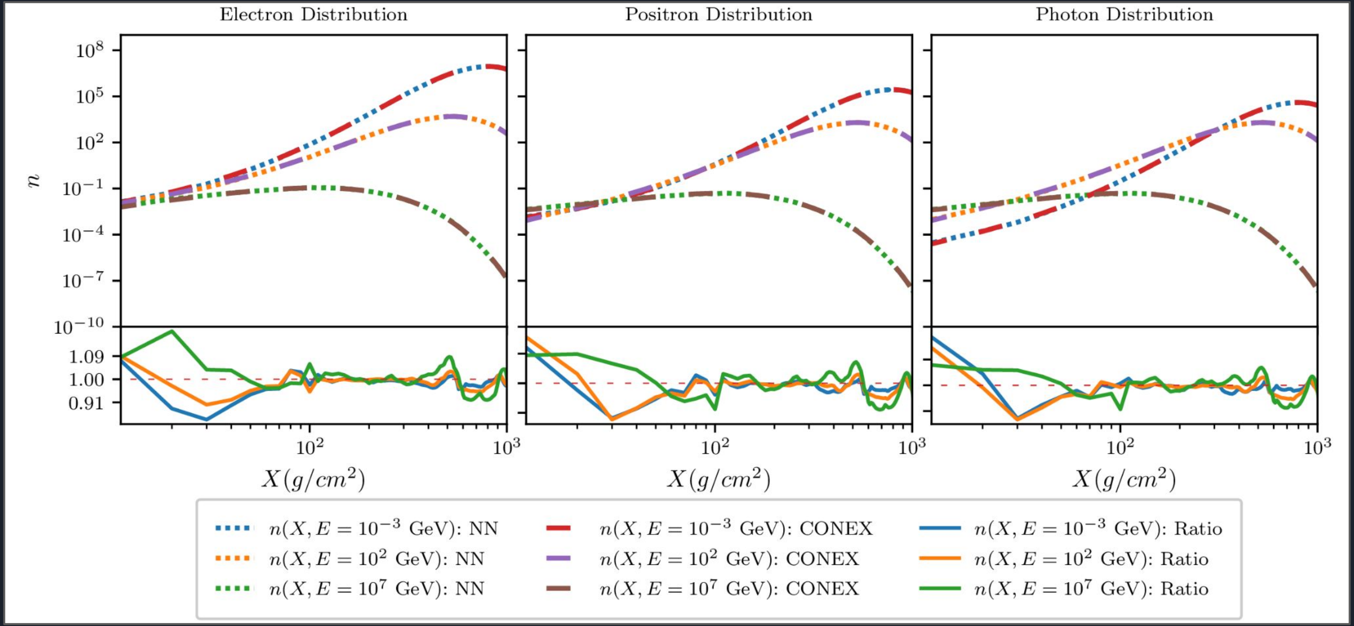




We get around 5% error.

- Use the model **iteratively** without using a hidden state
- The **lack of the hidden state** makes the network **memoryless**
- **Emulates** the nature of **simple MC process**
- Trained in sequences of **10 steps**





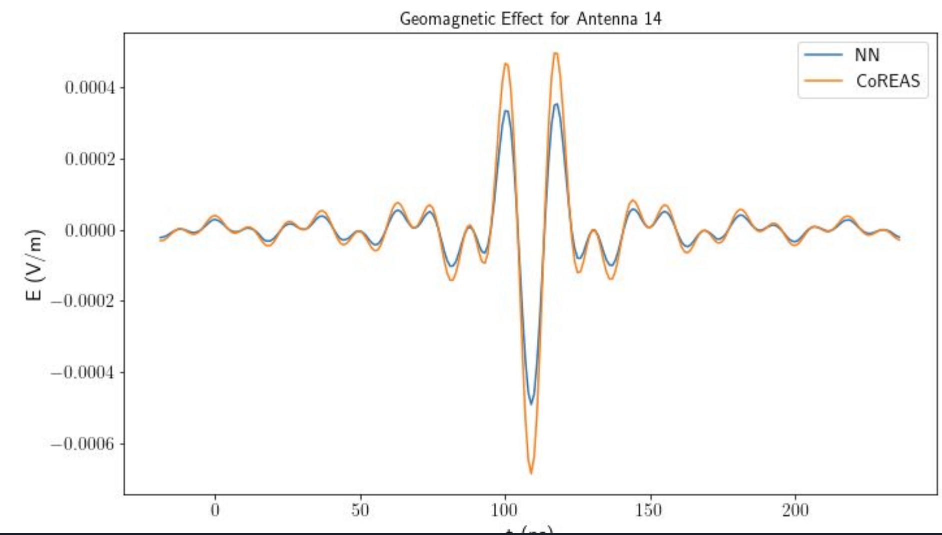
- Generated the entire shower using our sequential network
- Iteratively generate shower from initial conditions
- Maximum error is **around 10%**

Radio Simulations

- **EM part** of CR showers **produces radio pulses**

- Radio pulses of CR showers **don't have MC fluctuations**

- Can be approached with **more traditional techniques**



Conclusions

- Neural networks for shower simulations: **exciting new field**
- Advancements in novel architectures for **physics use cases**.
- For CR Physics, using a **memory less recurrent neural network** shows promise
- Network is **not linear,** doesn't capture fluctuations when used in a hybrid manner
- Use neural network for radio pulses.

Outlook

- **Hardcode the linearity** into the network
- Stay robust for **early fluctuations**
- Do the same in 3D.
- Get the energy footprint correct for radio pulses.

Ref: PoS(ICRC2023)515

Thank You

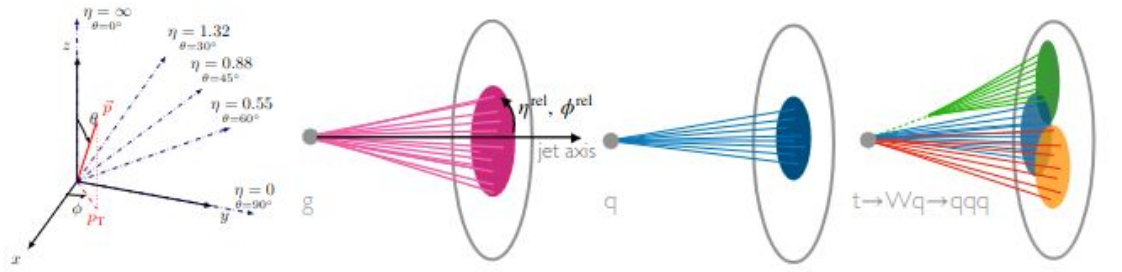
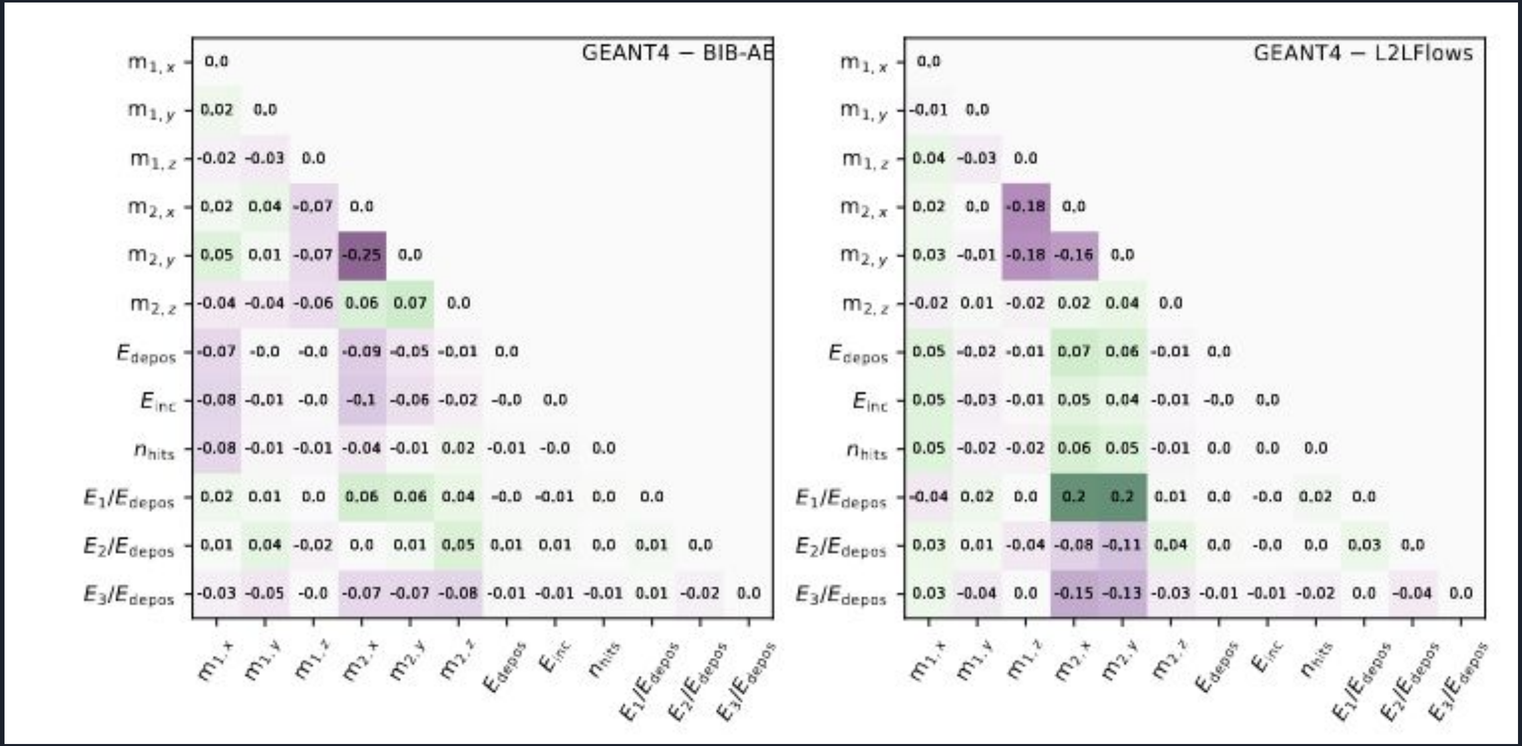


Figure 1: The collider physics coordinate system defining $(p_{\mathrm{T}}, \eta, \phi)$ (left). The three jet classes in our dataset (right). Gluon (g) and light quark (q) jets have simple topologies, with q jets generally containing fewer particles. Top quark (t) jets have a complex three-pronged structure. Shown also are the relative angular coordinates η^{rel} and ϕ^{rel}, measured from the jet axis.

Hardware	Simulator	NFE	Batch Size	Time / Shower [ms]	Speed-up
CPU	GEANT4			3914.80 ± 74.09	×1
	CALOCLOUDS	100	1	3146.71 ± 31.66	×1.2
	CALOCLOUDS II	25	1	651.68 ± 4.21	×6.0
	CALOCLOUDS II (CM)	1	1	84.35 ± 0.22	×46
GPU	CALOCLOUDS	100	64	24.91 ± 0.72	×157
	CALOCLOUDS II	25	64	6.12 ± 0.13	×640
	CALOCLOUDS II (CM)	1	64	2.09 ± 0.13	×1873

Simulator	$W_1^{N_{\text{hits}}}$ $(\times 10^{-3})$	$W_1^{E_{\text{vis}}/E_{\text{inc}}}$ $(\times 10^{-3})$	$W_1^{E_{\text{cell}}}$ $(\times 10^{-3})$	$W_1^{E_{\text{long}}}$ $(\times 10^{-3})$	$W_1^{E_{\text{radial}}}$ $(\times 10^{-3})$	$W_1^{m_{1,X}}$ $(\times 10^{-3})$	$W_1^{m_{1,Y}}$ $(\times 10^{-3})$	$W_1^{m_{1,Z}}$ $(\times 10^{-3})$
GEANT4	0.7 ± 0.2	0.8 ± 0.2	0.9 ± 0.4	0.7 ± 0.8	0.7 ± 0.1	0.9 ± 0.1	1.1 ± 0.3	0.9 ± 0.3
CALOCLOUDS	**2.5 ± 0.3**	11.4 ± 0.4	15.9 ± 0.7	**2.0 ± 1.3**	38.8 ± 1.4	4.0 ± 0.4	8.7 ± 0.3	1.4 ± 0.5
CALOCLOUDS II	3.6 ± 0.5	26.4 ± 0.4	**15.3 ± 0.6**	3.7 ± 1.6	11.6 ± 1.5	**2.4 ± 0.4**	**7.6 ± 0.2**	3.9 ± 0.4
CALOCLOUDS II (CM)	6.1 ± 0.7	**9.8 ± 0.5**	16.0 ± 0.7	**2.0 ± 1.4**	**8.3 ± 1.9**	3.0 ± 0.4	9.5 ± 0.6	**1.2 ± 0.5**

Img source: CaloClouds, arXiv:2305.04847v2


GEANT4 − BIB-AE
GEANT4 − L2LFlows
$m_{1,x}$
$m_{1,y}$
$m_{1,z}$
$m_{2,x}$
$m_{2,y}$
$m_{2,z}$
E_{depos}
E_{inc}
n_{hits}
E_1/E_{depos}
E_2/E_{depos}
E_3/E_{depos}