Generative Modelling for Detector Simulation

Learning to Discover, Paris 2022

Johnny Raine, University of Geneva

Overview

Intro to Detector Simulation

Machine Learning Approaches

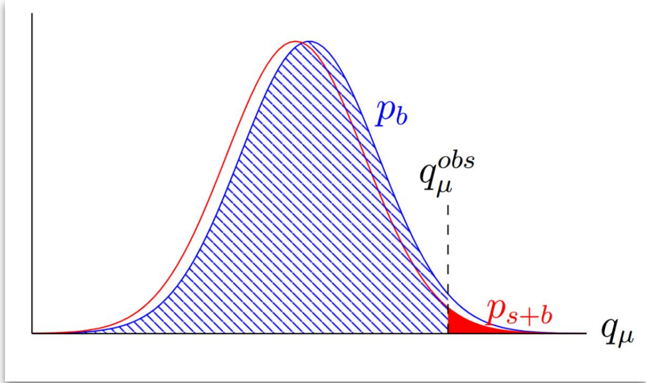
Challenges

Outlook and Summary

Introduction

Key part of an analyses is to have a reference

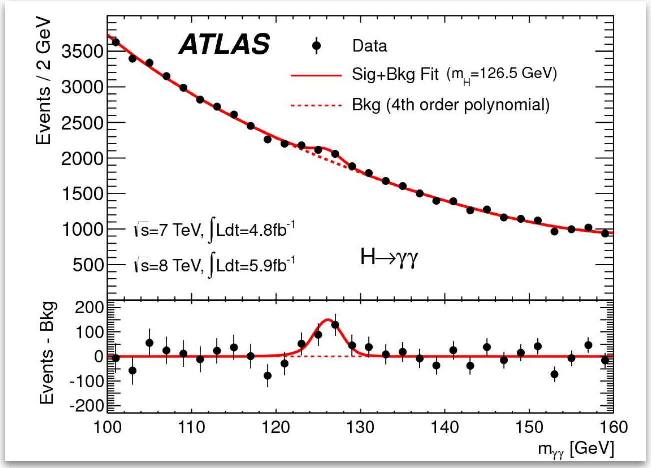
- Does data fit Hypothesis 1 or Hypothesis 2?
  - Amount of signal on top of background
  - Deviation away from expectation



Introduction

Key part of an analyses is to have a reference

- Does data fit Hypothesis 1 or Hypothesis 2?
  - Amount of signal on top of background
  - Deviation away from expectation
- Sometimes can use data to build your own expectation

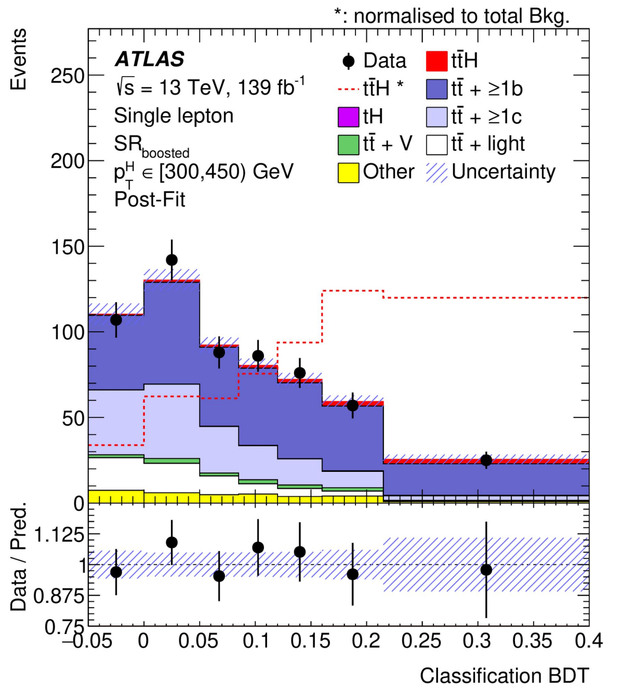


[1]

Introduction

Key part of an analyses is to have a reference

- Does data fit Hypothesis 1 or Hypothesis 2?
  - Amount of signal on top of background
  - Deviation away from expectation
- Sometimes can use data to build your own expectation
- But other times background and analysis requires more detailed description
  - Multiple backgrounds, complex observables…

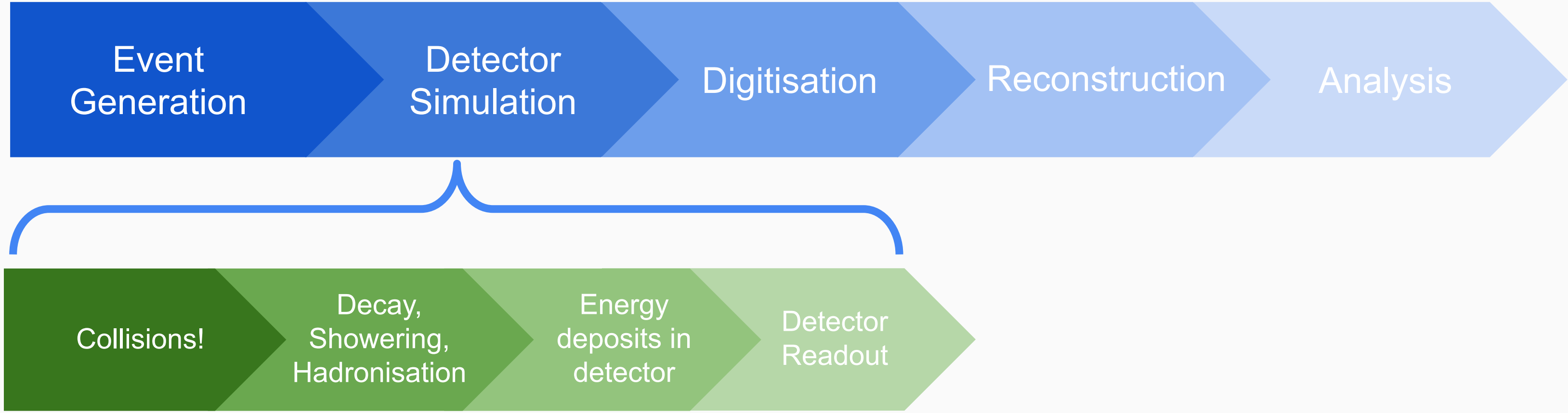


[2]

Introduction

In most cases rely on detailed simulation techniques to get our predictions

- Experiments simulate thousands of millions of collision events



Introduction

In most cases rely on detailed simulation techniques to get our predictions

- Experiments simulate thousands of millions of collision events

Event Generation → Detector Simulation → Digitisation → Reconstruct → Analysis

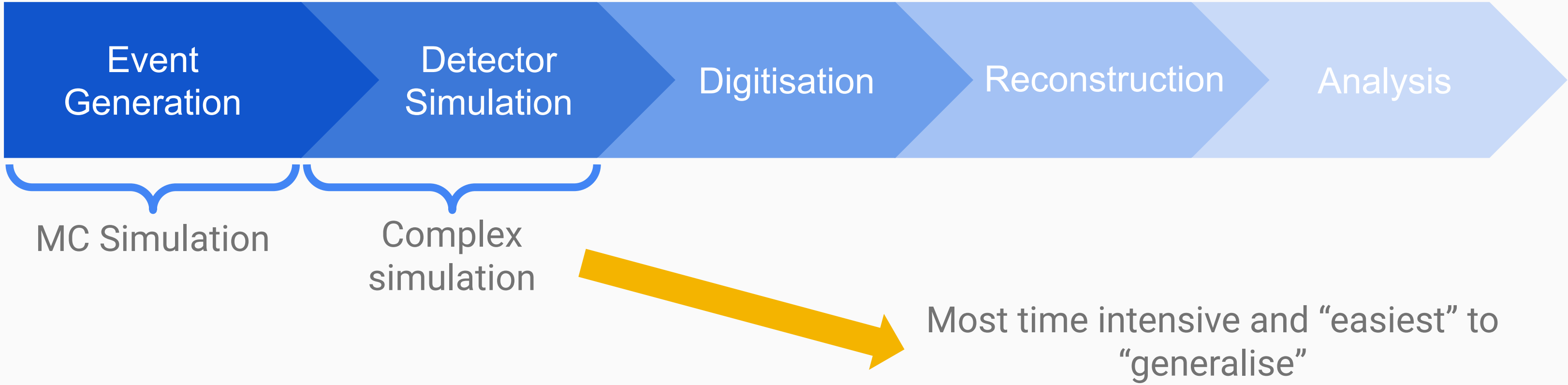
Largest use of computing resources in LHC Experiments

Try to improve speed: Reduce CPU cost, increase statistics

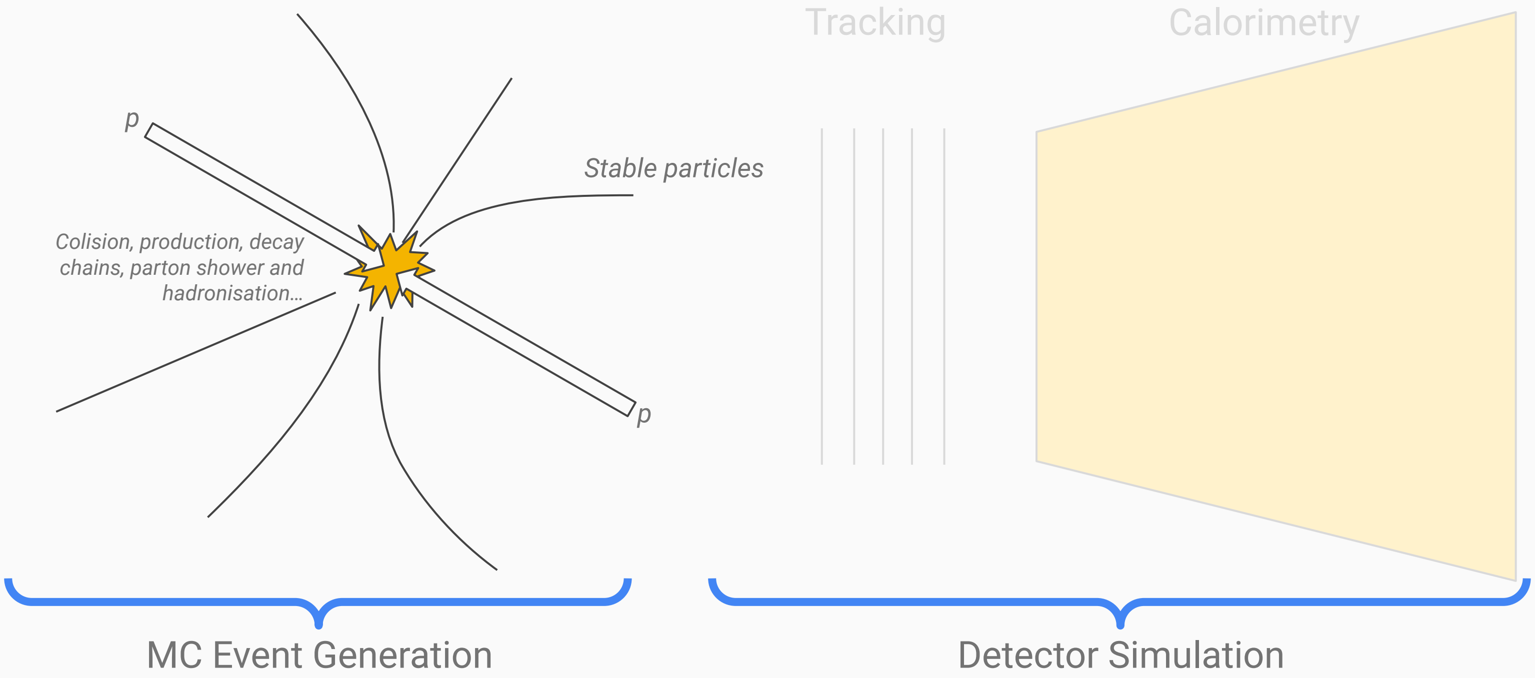
Introduction

In most cases rely on detailed simulation techniques to get our predictions

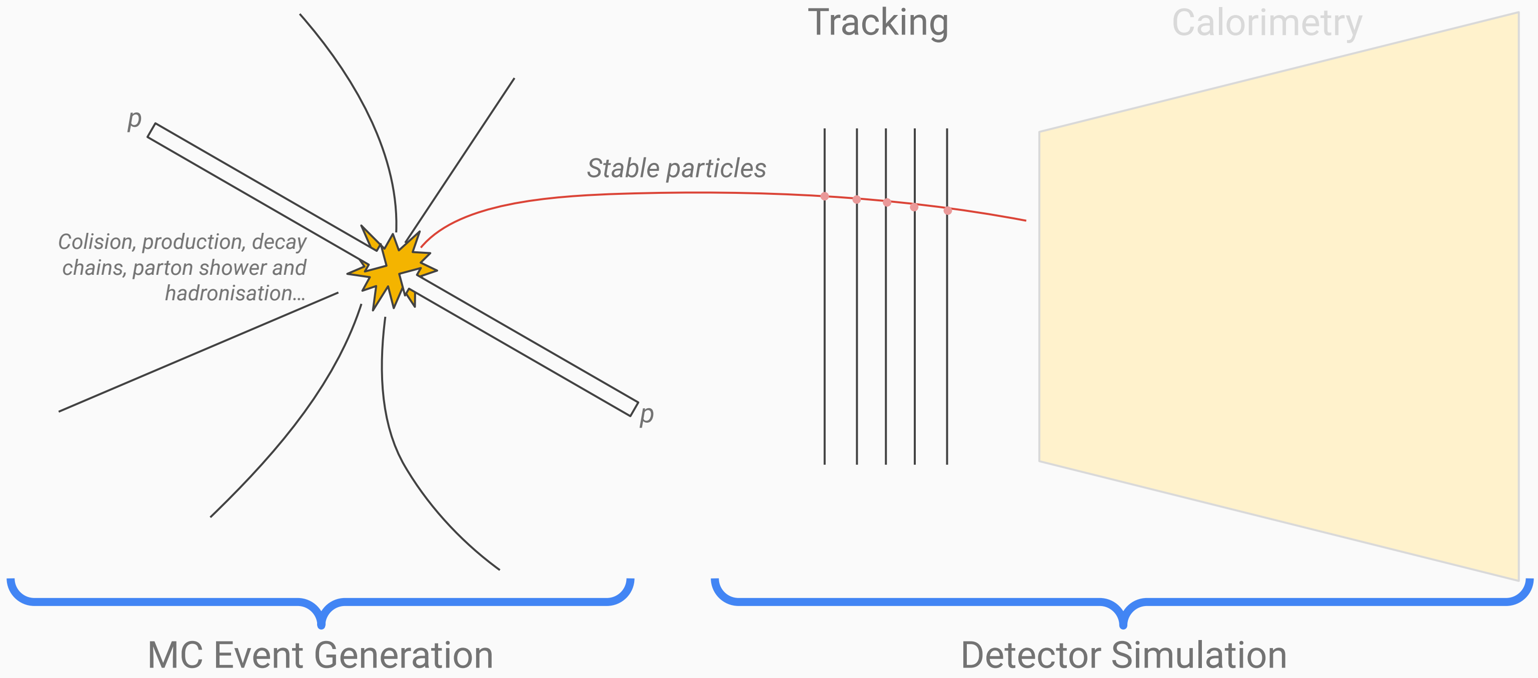
- Experiments simulate thousands of millions of collision events



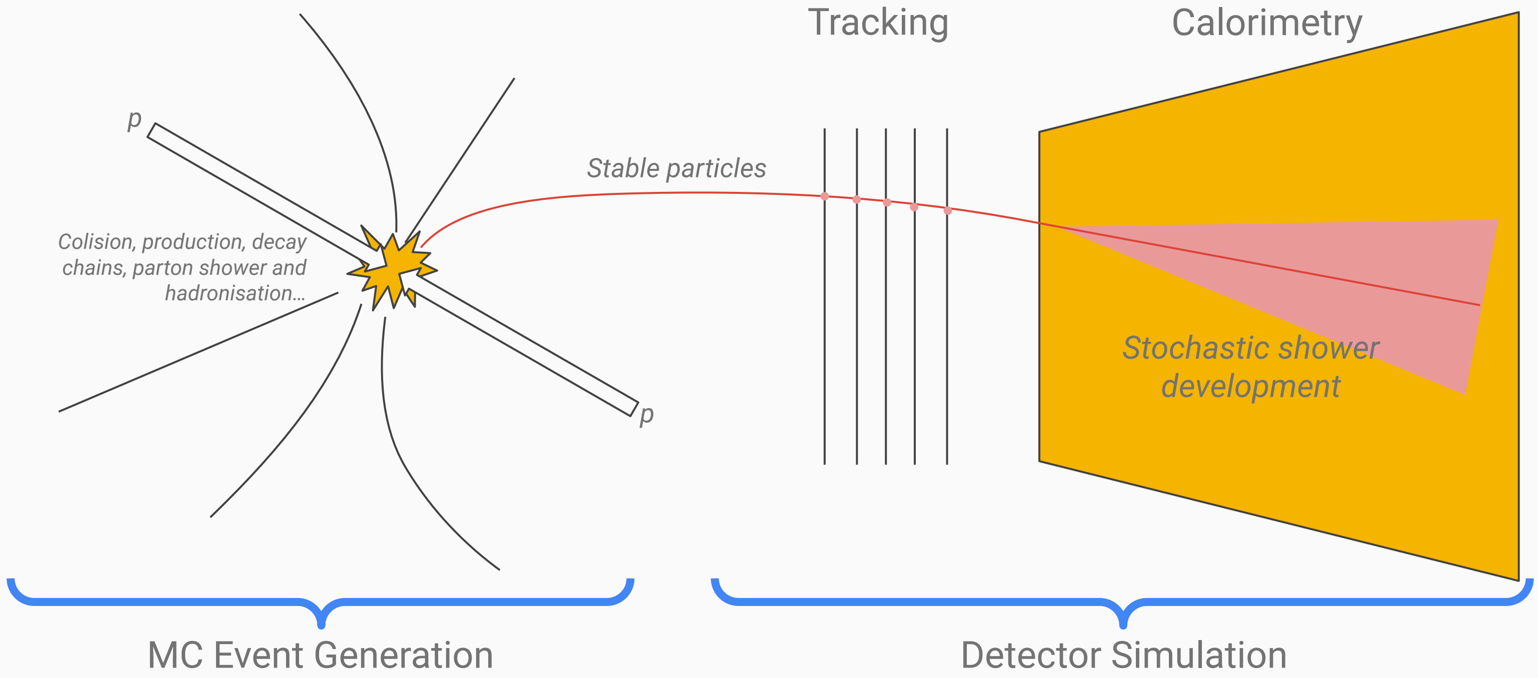
What happens in a particle collision



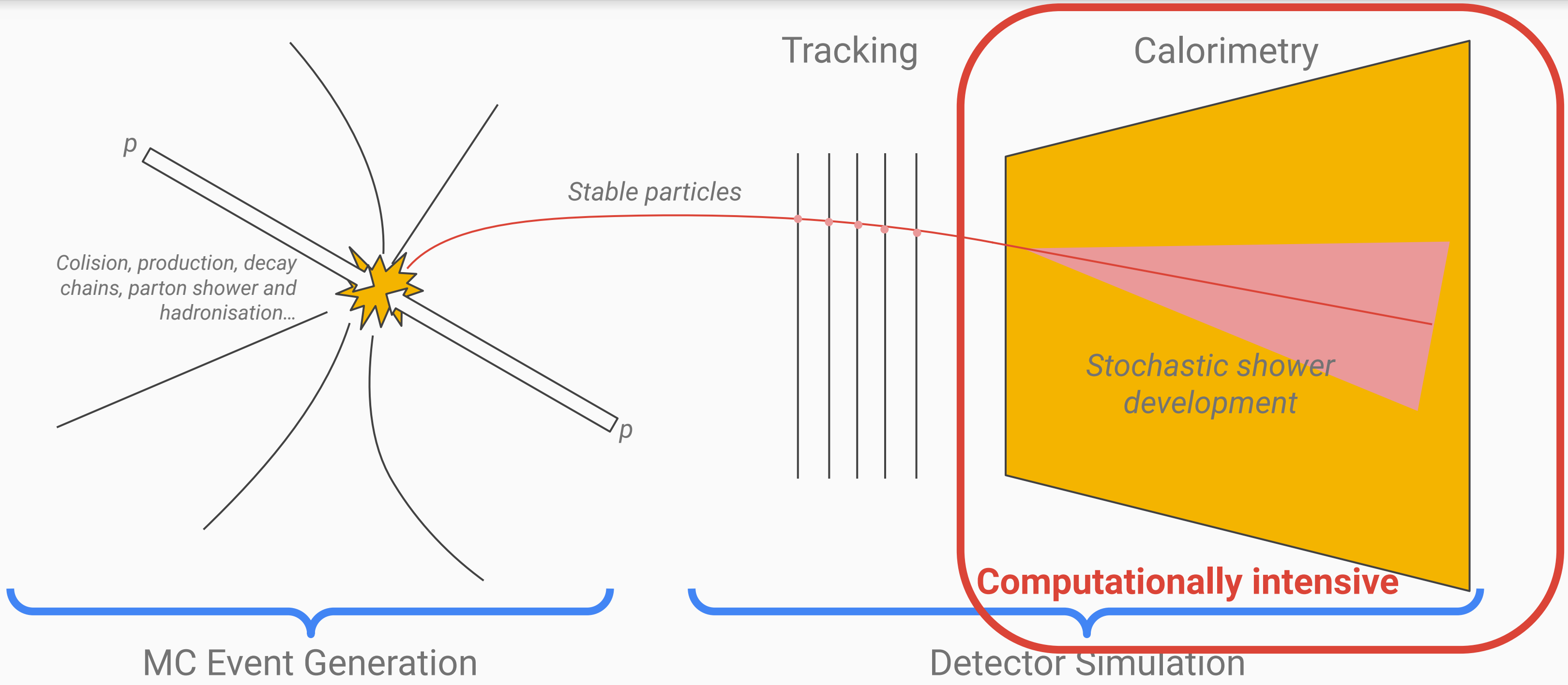
What happens in a particle collision



What happens in a particle collision

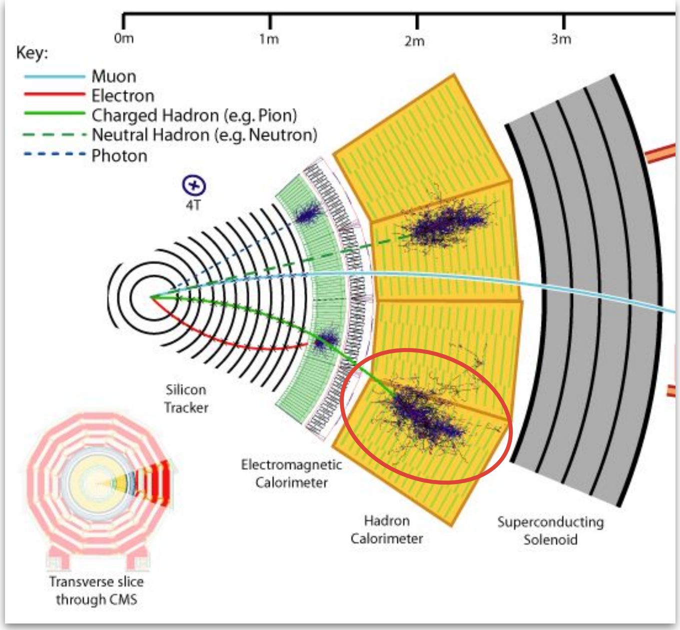


What happens in a particle collision



What makes a calorimeter so slow to simulate

- Simply lots of material and interactions!
- Designed to capture energy of particles by "splitting" and "showering"
  - Try and "stop" the particle, record deposited energy
  - Dense material to shower
  - Scintillator to record energy
- Different calorimeters target different particles
  - **Electromagnetic** (e,y) and **Hadronic** (e.g. pions)
- Two "philosophies" of calorimeter
  - Sampling and homogenous
  - Both result in a lot of particles and interactions



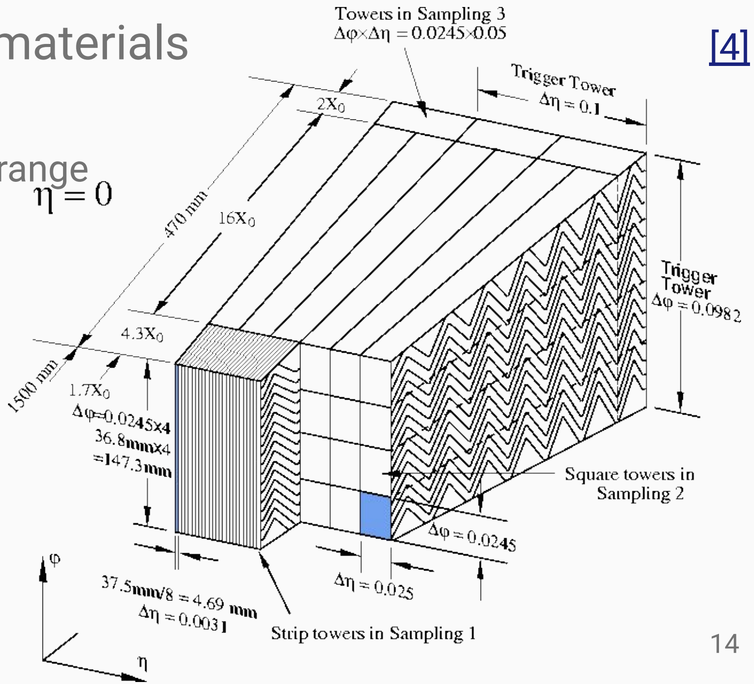
[3]

High multiplicity of secondary particles in shower

Example - ATLAS EM Calorimeter

- Sampling calorimeter - alternating between materials
- Plus completely non-trivial geometry!
    - Assortment of many layers each covering only subrange
    - Varying cell (read pixel) size, shape, and depth
    - Constantly changing over detector volume
- ~190k readout channels
- Precise simulation takes O(minutes)
    - Simulate all particle interactions with material
    - Difficult *accordion shape*

[4]



GEANT - The gold standard for simulation

Traditional simulation uses GEANT to get detailed simulation of detector

- Used in particle physics, nuclear physics, accelerator design, space engineering and medical physics
- Very **detailed** and **highly accurate**

Build whole model of detector - every piece of material (active and inactive)

- Every detector system and complete geometry
- Specific composition of materials
- Describe down to the cabling and readout

Simulate **interactions of every particle** with the **whole material inside detector**

- Step by step each interaction as particle propagates through detector
- Calculate EM, hadronic and optical interactions

Calorimeter Output

Although lot of material and showering particles, don’t read-out all particles

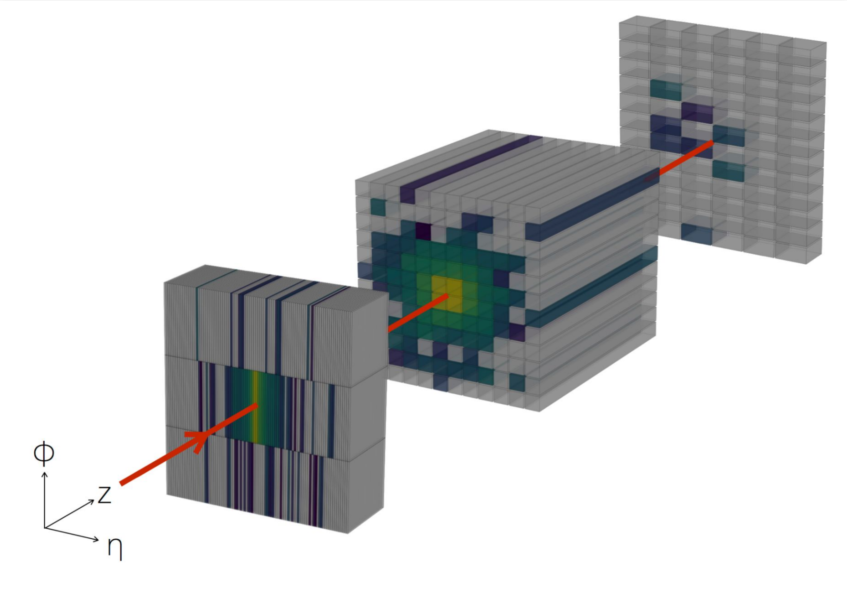
- Readout channels measure energy deposited in sub-volume of calorimeter
- Granularity and shape determined by chosen detector material and philosophy
- Can use layers of detector for depth

Particles interacting with calorimeter don’t interact with each other!

- Don’t need to simulate everything all at once
- Can **overlay simulation of individual particles**

Calorimeter Output

- Individual particle traverses detector
- Energy deposited in many cells from secondary particles in shower
- Can build an “image” but
    - High dynamic range of “pixels”
    - Often very sparse
    - Stochastic - same incoming particle results in different shower



[5]

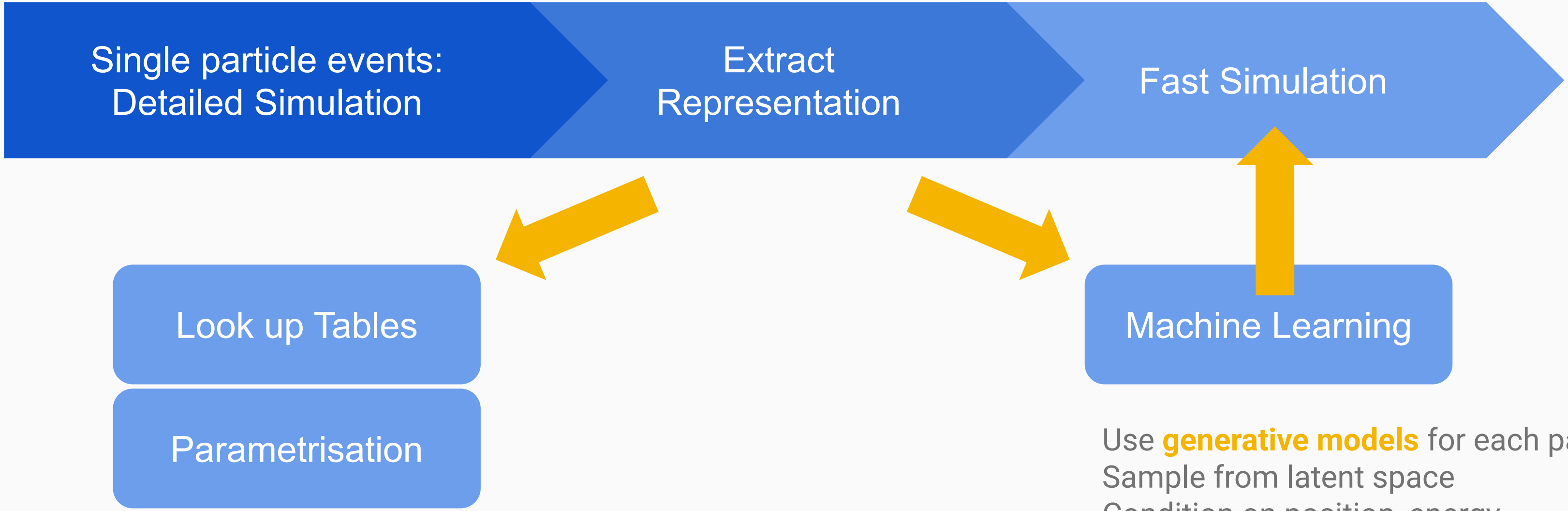
Fast Simulation

GEANT is CPU intensive and **too slow** for future LHC conditions

- Need to speed this up with approximative approaches
- Several fast approaches in use at experiments, but none are perfect
- Replace extensive calculations with “jump to the finish” approach
    - From particle properties (type, energy,...), predict energy depositions
    - Don’t need to consider full detector, just window around particle

Perfect fertile ground for applying generative modelling!

Fast Simulation



Parametrise in Energy, position, particle
PCA+Sampling and PDF+Sampling

Use **generative models** for each particle
Sample from latent space
Condition on position, energy

Generative Modelling for Calorimeter Simulation

Many efforts now ongoing for calorimeter simulation with generative modelling

- A lot of design choice comes down to the detector at hand!
  - How many layers, granularity of readout, consistent between layers?
- Wide range of architectures under study
  - GANs, Autoencoders and Flows all represented!

A lot of approaches need to make some simplifications

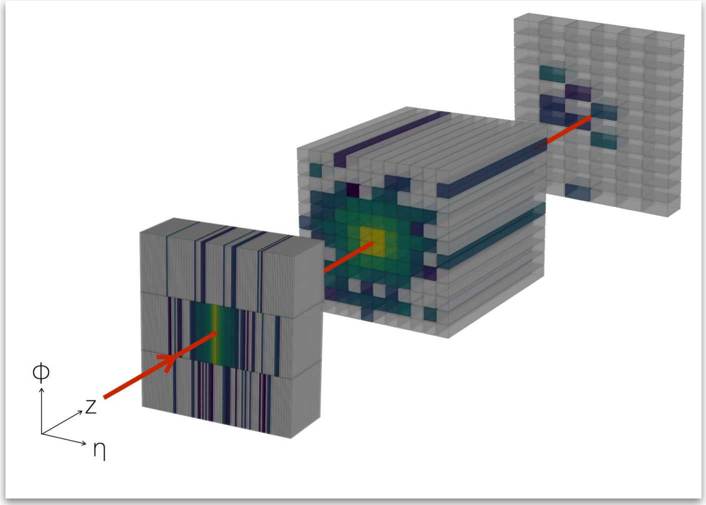
- Detector geometry, particle energies/types, incidence angles…

Non-exhaustive Examples

Paganini et al (2017)

CaloGAN

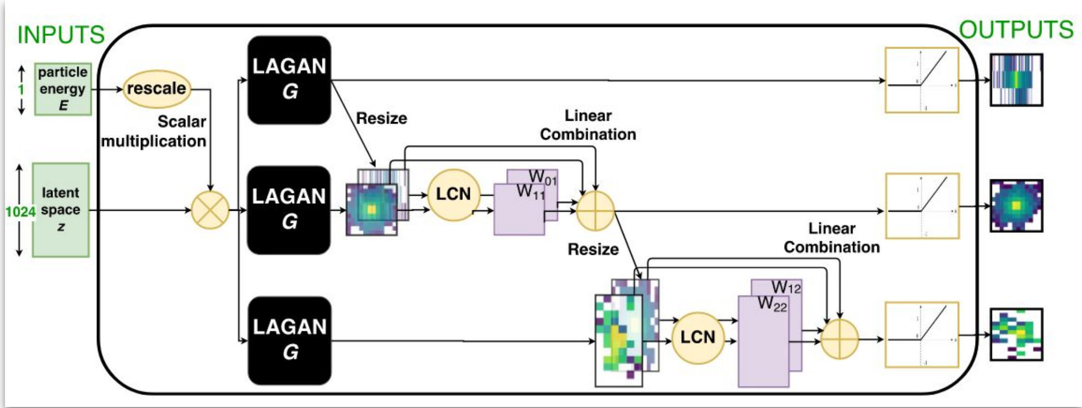
- ATLAS inspired detector geometry
  - Three layers; different depth and cell size
  - No accordion shape, predefined window
- Trained to generate energies in cells
  - Layer0: 3x96
  - Layer1: 12x12
  - Layer2: 12x6
- Particle incident perpendicular to the centre of calorimeter
- Uniform energies between 1 and 100 GeV for three particle types
  - Separate model per particle (e, y, $\pi^{\pm}$)



Paganini et al (2017)

CaloGAN

- LAGAN architecture with a stream per layer, conditioned on particle E
  - Forced by different granularities per layer! Can't use the same Conv kernels
  - Learned attention to propagate previous layer into next

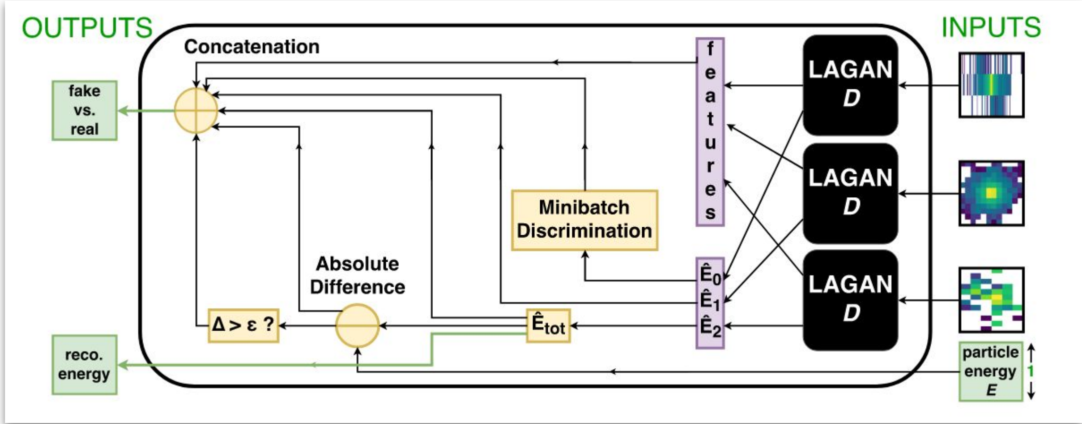


Convolutional and linear combination layers

[Paganini et al (2017)](#)

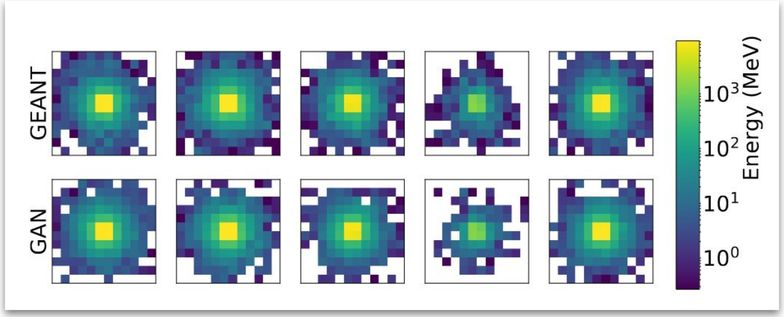
CaloGAN

- LAGAN architecture with a stream per layer, conditioned on particle E
  - **Auxiliary targets in critic** - calculate layer energies, total energies
  - Address sparsity by **augmenting input with sparsity percentage**, using minibatch discr.
- Add MSE term to loss - keep reco energy close to requested E

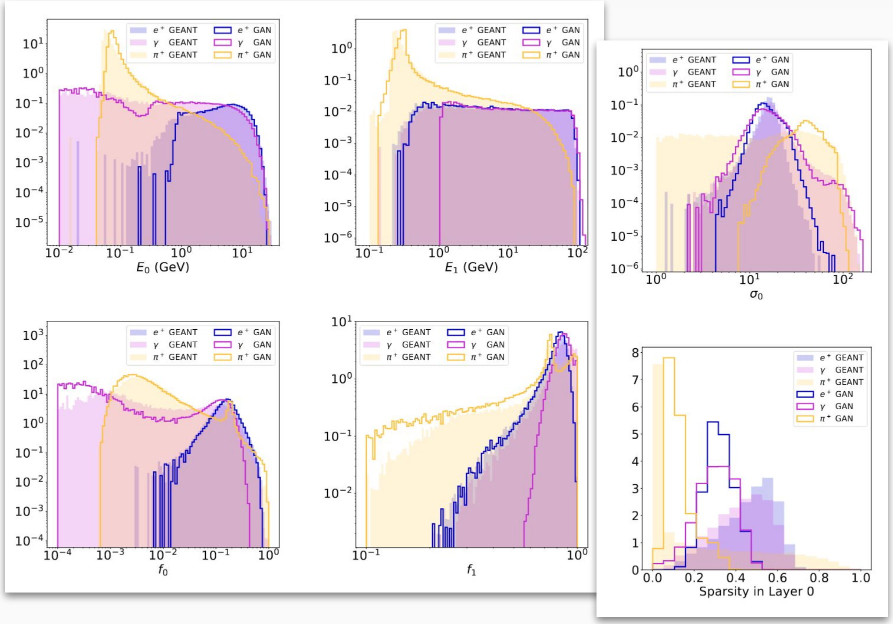


Paganini et al (2017)

CaloGAN



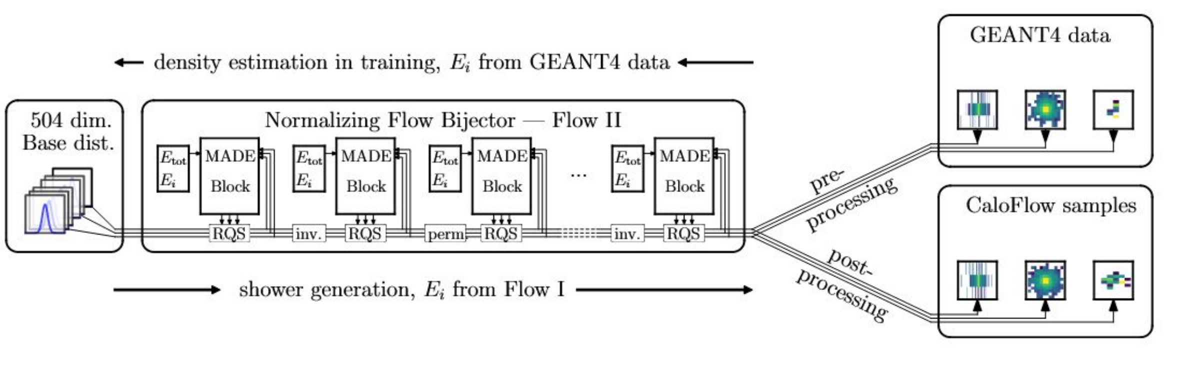
- Individual showers look pretty similar
- Key focus for physics is distributions over many events
    - Can be seen as a large number of metrics!
    - **Model correlations** between cells/layers
    - i.e. layer energies, ratios, weighted depth
    - Sparsity and shower width harder task



Krause, Shih (2021)

CaloFlow

- Using CaloGAN dataset
- Two stage flow approach
  - First: Flow to get energies per layer
  - Second: Flow to get shower shape
- Masked autoregressive flow
  - Conditioned on Etot (+Elayers in 2nd flow)
  - Rational Quadratic splines for transform
- Use standard flow MLE loss

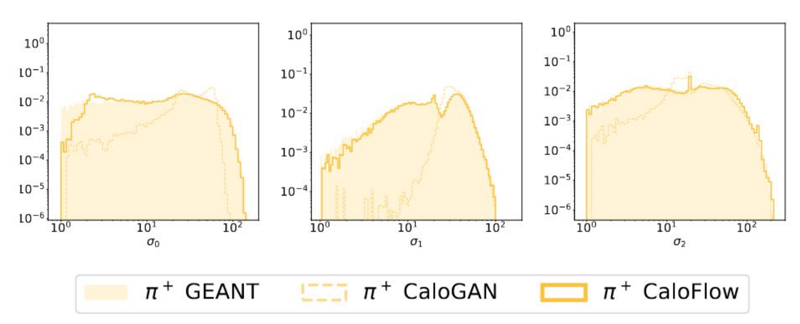


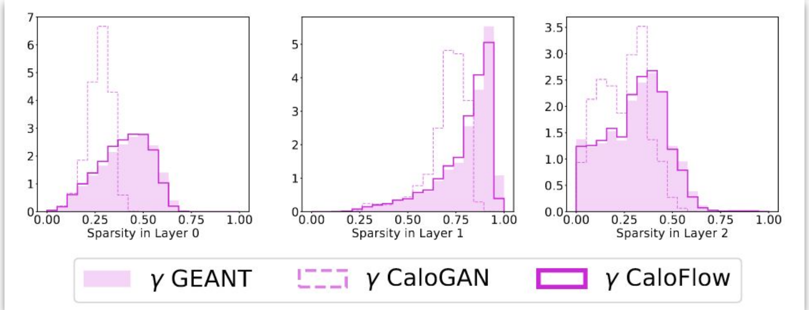
Krause, Shih (2021)

CaloFlow

Improved performance over CaloGAN seen

- More accurate generation of underlying distributions
- Flows trained to learn densities over all events
  - Should lead to better modelling here





However, slower to generate new showers

	CaloGAN		CaloFlow
batch size	batch size requested	100k requested	
10	455	2.2	835
100	45.5	0.3	96.1
1000	4.6	0.08	41.4
5000	1.0	0.07	36.2
10000	0.5	0.07	36.2

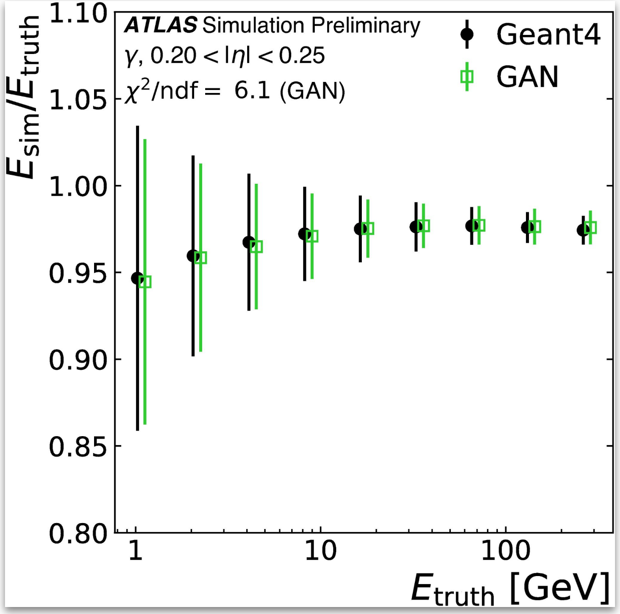
ATLAS Cell based

- Using ATLAS detector and simulation to train generative models
  - Simplified by selecting **narrow window of detector**, only photons (E in 1 to 262 GeV)
  - Photons generated on calo surface without spread
- Windows of cells selected around trajectory
  - 266 (GAN) or 276 (VAE) across four layers
- WGAN-GP and VAE studied with **simple generator architectures**
  - **Only dense layers used**
  - Focus on solving shortcomings observed in reproducing shower energies

ATLAS Collaboration (2018, 2019)

ATLAS Cell based - GAN

- Deep but simple generator using Swish activation
- WGAN-GP architecture with two critics
  - Standard critic for cell energies (GP=10)
  - Additional critic on sum of cell energies (GP=10^{-8})
  - Gradient penalties differ by 9 orders of magnitude!
- Two critics balanced in final loss
  - Energy critic has 10^{-6}x smaller weight
- Without the second critic resolution was factor 2-4 too large at high energies

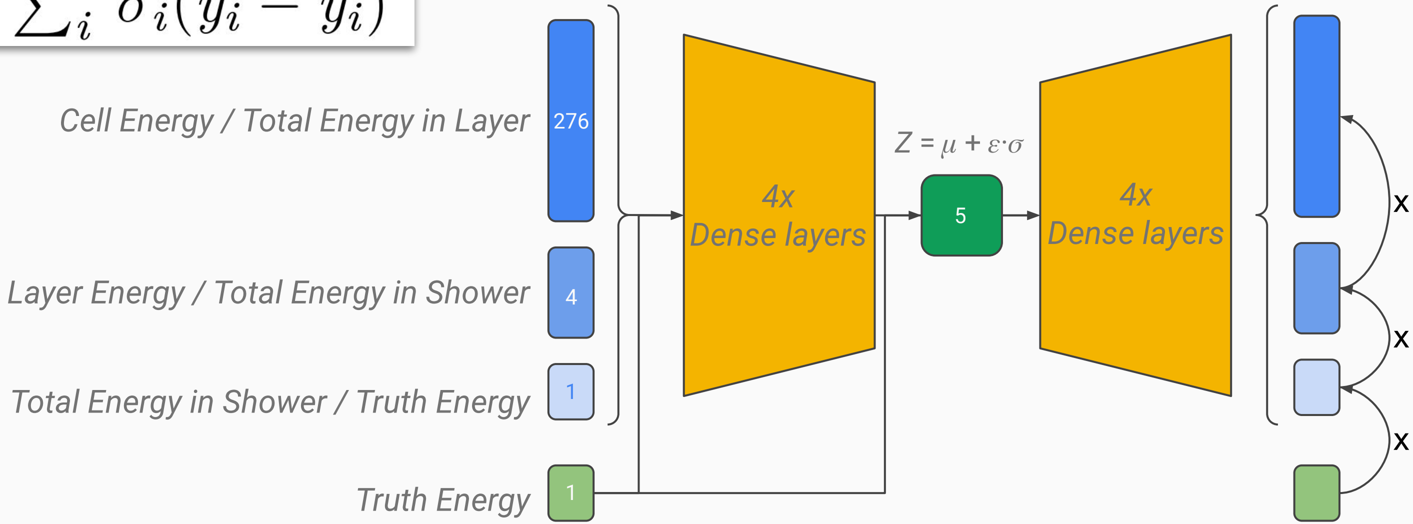


ATLAS Cell based - VAE

Simple architecture but reparametrisation of inputs to help learn what is important

- Weighted MSE loss - prioritise reconstruction of inputs with small range of values

$$\frac{1}{n}\sum_{i}^{n}\sigma_i(y_i-\hat{y}_i)^2$$



ATLAS Collaboration (2018, 2019)

ATLAS Cell based - VAE

Reparametrisation of inputs

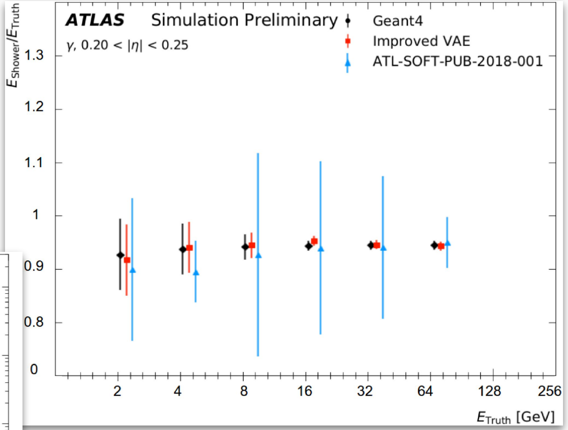
- Reduced underestimation
- More accurate spread

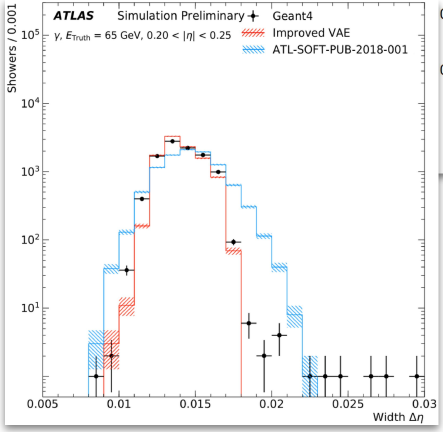
Weighted MSE

- Prioritises total/layer energies
- Help improved shower shapes

Performance competitive with GAN but much faster to train and simpler arch!

- Only 5D latent space





HGCAL with WGAN

CMS HGCal prototype inspired geometry

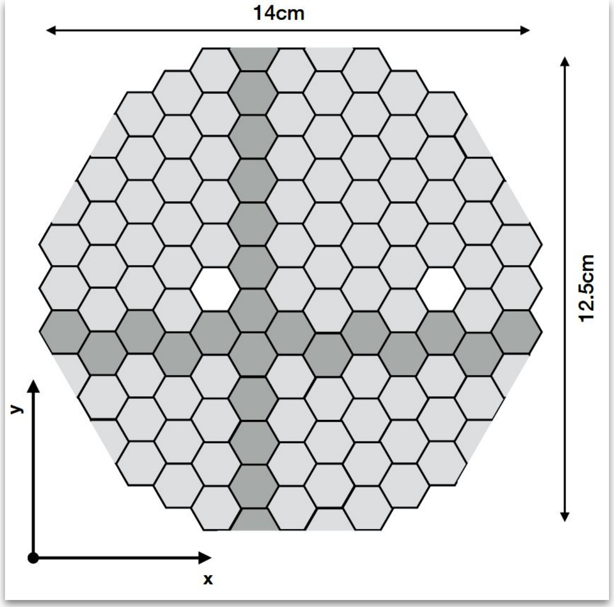
- Hexagonal readout over 7 layers approximated to be square
- Resulting image with 12x15x7 pixels

Train WGAN-GP for electrons with E in 20-90 GeV

- Conditioned on energy, impact position
- 7 Parallel "towers" of 2D convolution layers and linear layers
- 2D conv layers for critic

Additional constrainer networks to improve generation

- Regressors to predict E, (x,y) from generated showers
- Use 3D convolution layers on shower image
- Included as MSE term on predicted and conditioned value

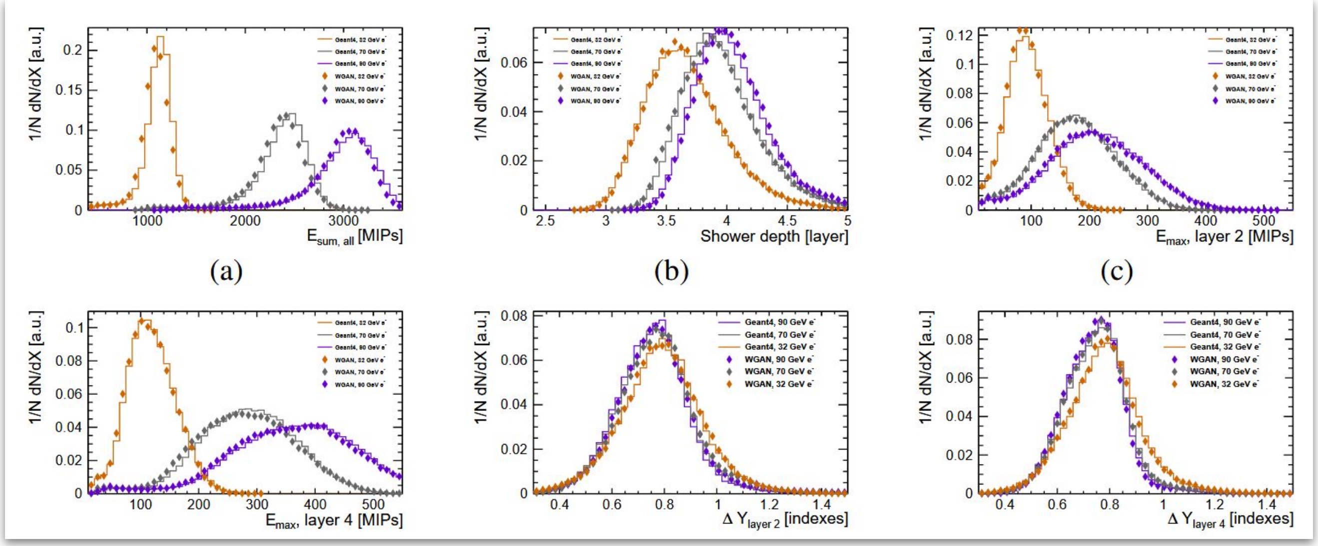


[Erdmann et al (2018)](#)

HGCAL with WGAN

Generally good reconstruction of energies and correlations across cells/layers

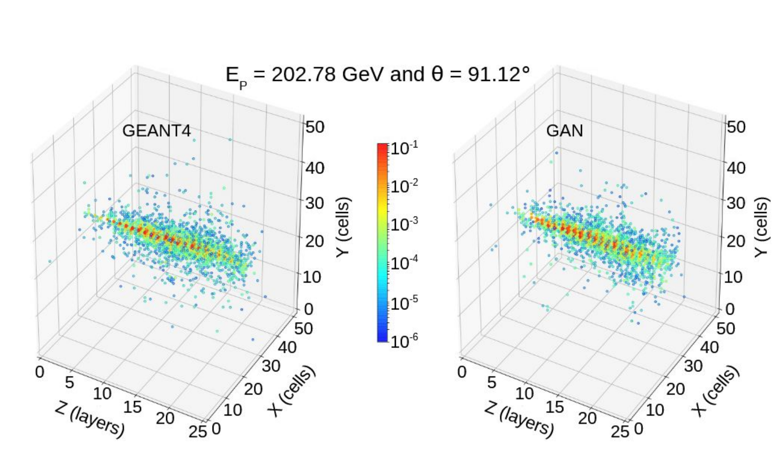
Shift in total E



3DGAN

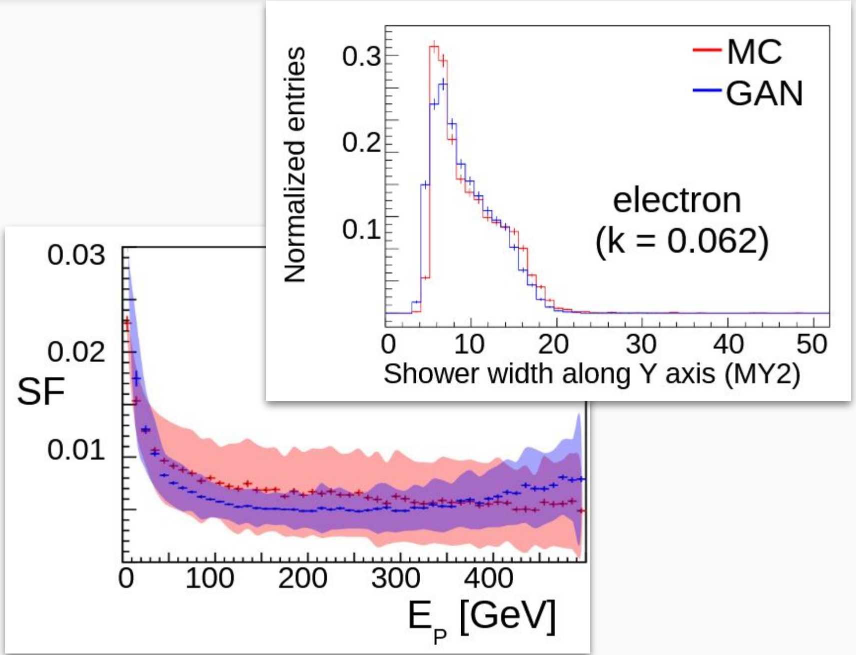
Higher granularity calorimeter based on Linear Collider Detector

- Covers barrel region with 25 layers
  - Incidence angles from 60° to 120° (|eta| < 0.55)
  - Electrons with E from 2-500 GeV
  - Condition on both energy and incidence angle
- Select area around incident particle for 3D grid
  - Upscale all layers to same granularity as innermost layer
  - 51x51x25 sparse image per shower
  - Use 3D convolutional layers in generator and discriminator



3DGAN

- Preprocessing cell energies to handle high dynamic range
    - Log transform lead to distorted results
    - Use power function, energy$^{0.85}$
- Makes use of additional targets in critic
    - Discriminator predicts true energy in addition
    - Additional losses from predicted E and angle
- Over all events reproduces cell energy depositions in X, Y and Z
- Total energy very close but narrower spread



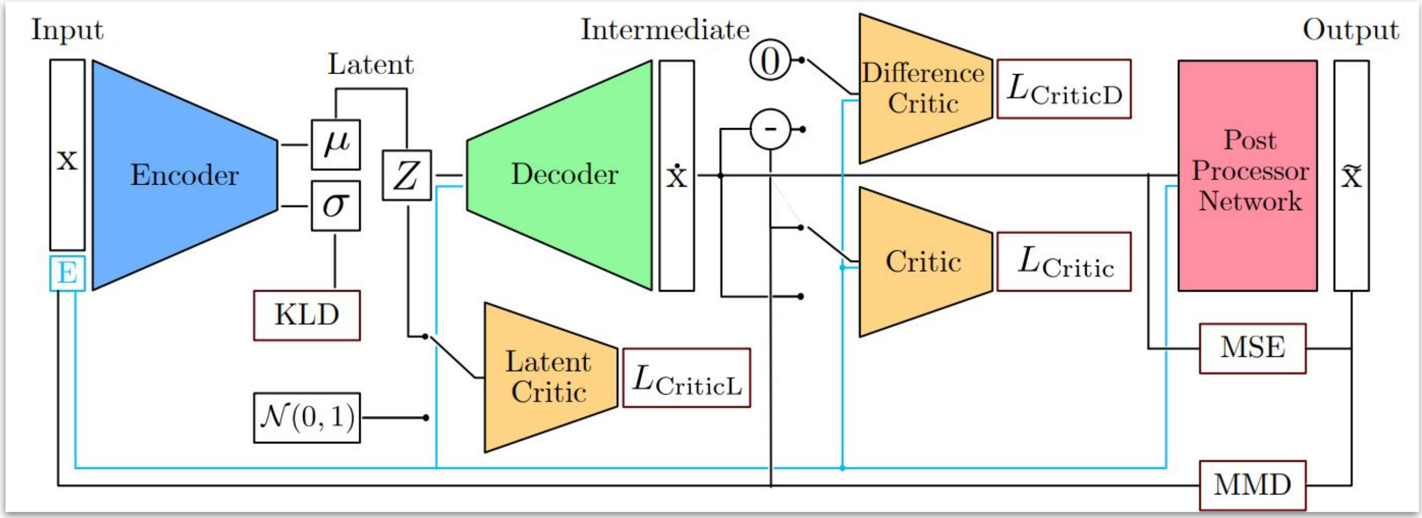
Residual from true energy for showers from GAN and Geant

Buhman et al (2020, 2021)

BIB-AE for High Granularity

High granularity detector based on ILD design using homogenous 30x30x30 grid

- Leverage additional loss terms in architecture marrying GANs and AEs



Buhman et al (2020, 2021)

BIB-AE for High Granularity

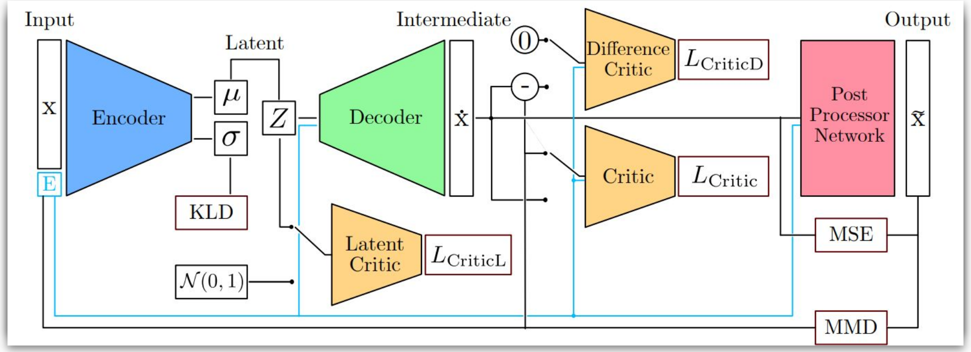
Encoder with 3DConv and Dense layers into 24D latent space using reparam. trick

Latent encoding concatenated with extra 488 values from normal distribution

Decoder takes 512 inputs, reconstructs input shower

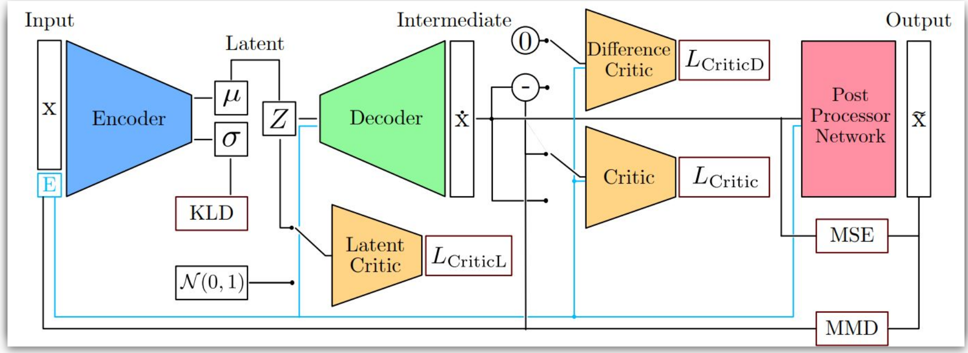
Additional **Postprocessing Network** to correct the per hit energies uses 1x1x1 convolutions

Three **critics** for loss terms



BIB-AE for High Granularity

Multiple loss terms control how the network learns to generate showers



Latent space losses

KLD - standard VAE loss term

- Preserves relationship between mu, sigma and x

CriticL - like in an AAE

- Penalises if overall distribution doesn't match prior
- Better for ensuring all LS dimensions Gaussian

Reconstruction space losses

CriticDiff - replaces MSE term in VAE

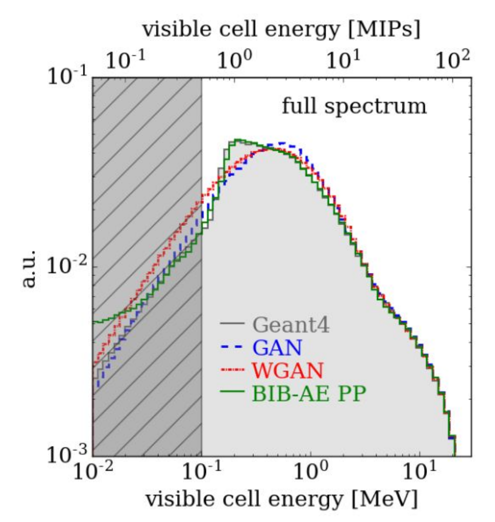
- MSE lead to smeared output: instead critic between an input of all 0s and |reco-input| of all cells

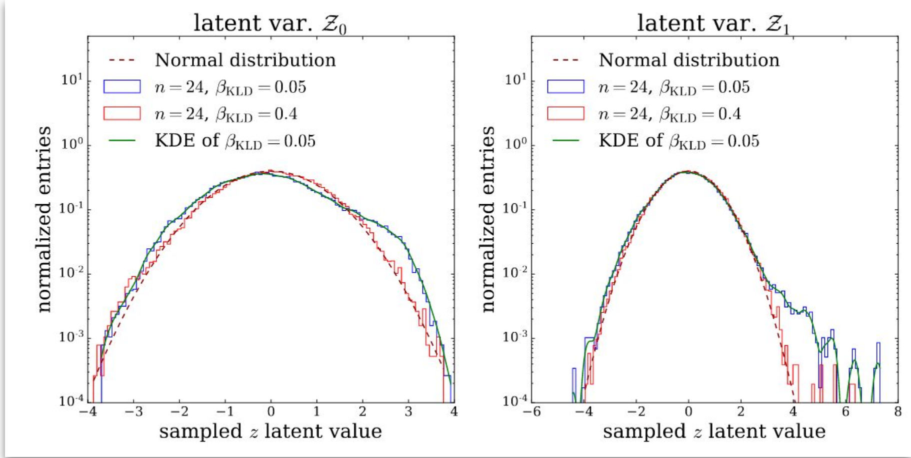
Critic - like in an GAN

- Can a discriminator differentiate between real/false

Buhman et al (2020)

ILD with BIB-AE



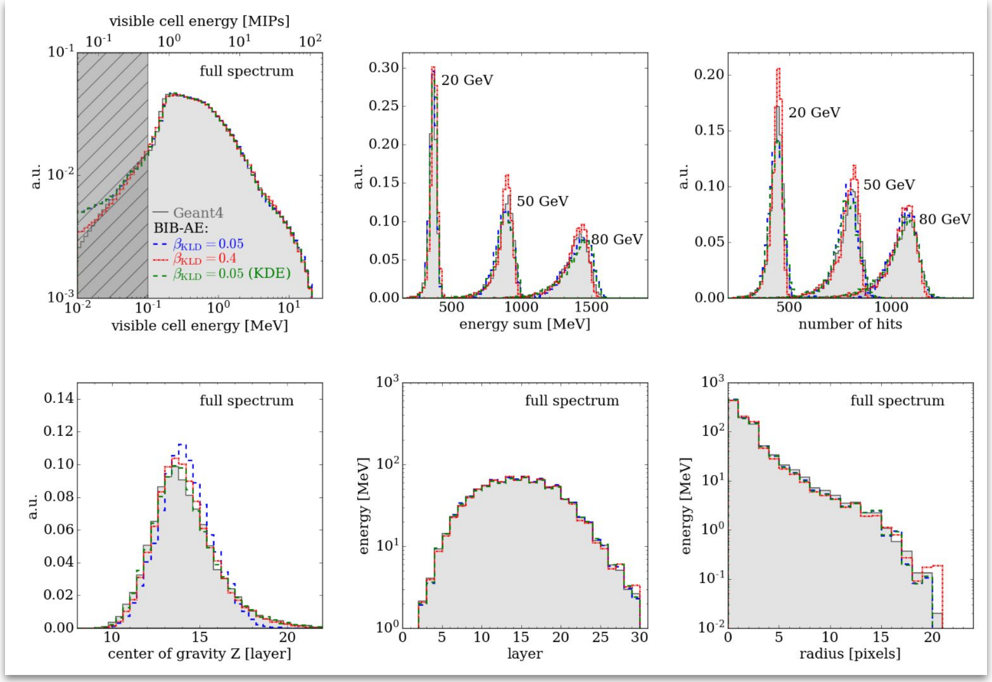


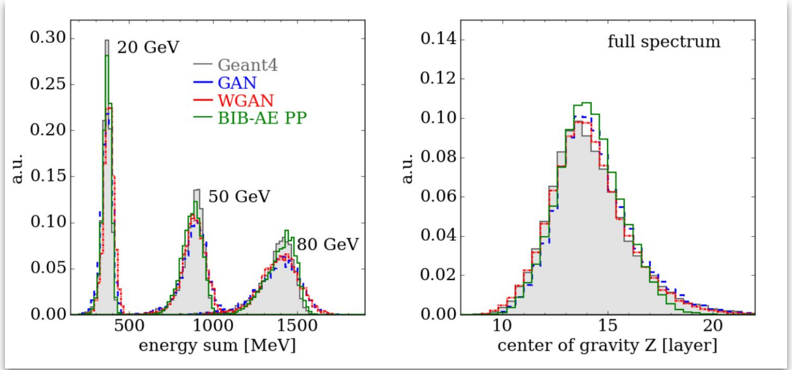
Can see non-gaussianity of encoded LS (blue) of final model

Thanks to Postprocessing can recover hard edge in energies

[Buhman et al (2020)](#)

ILD with BIB-AE





Fitting posterior LS distribution with KDE for sampling

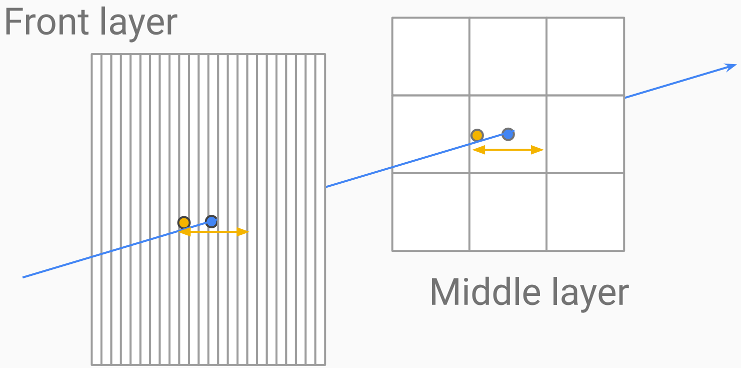
- Reduces mismodelling substantially
- For physics purposes nothing says our LS needs to be Gaussian, just sampleable!

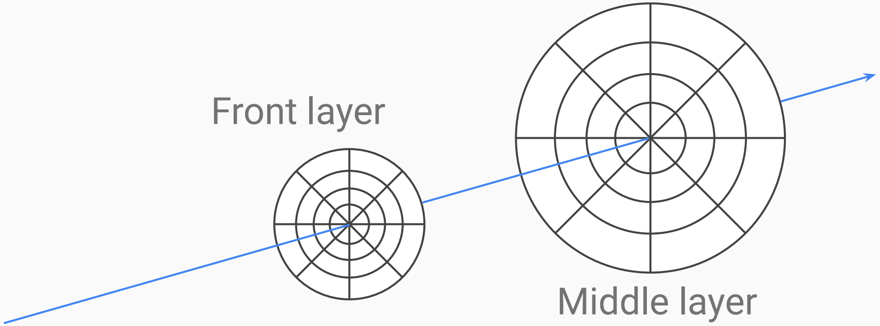
ATLAS Collaboration (2021)

ATLAS FCS

Most approaches produce showers using detector geometry as starting point

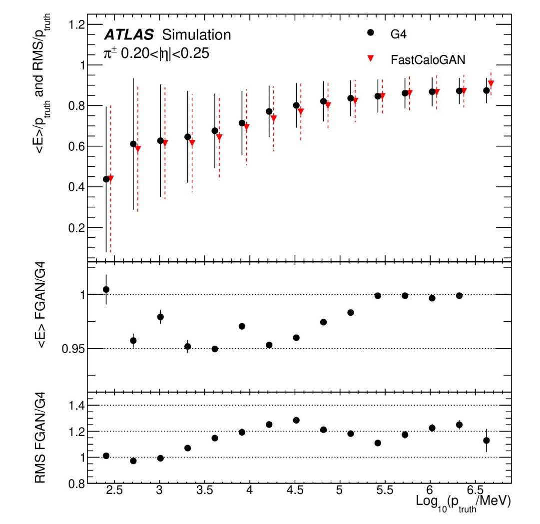
- Cell energies have correlation to where the impact particle hits
- Simplify by moving to “particle centred” approach
- Exploit prior knowledge of “conical” shower development in polar coordinates
- Map energy from polar voxels centred on particle to cells using detector geometry



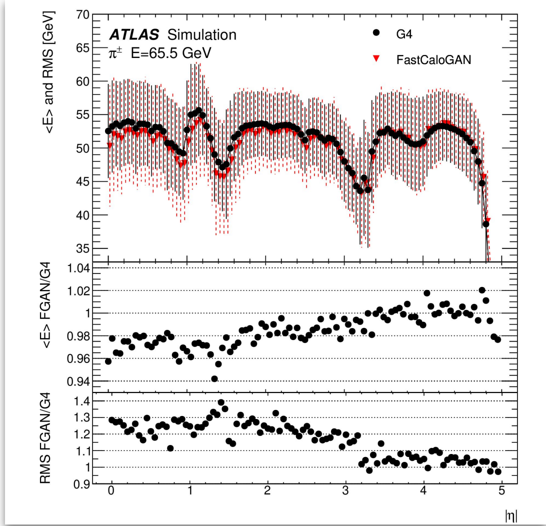


ATLAS Collaboration (2021)

ATLAS FCS

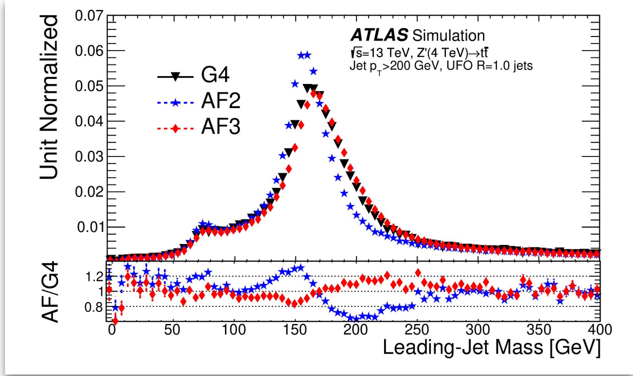


Covering wide energy range for pions



Covers full detector range in |η|

Each eta slice is a separate GAN



Evaluate performance downstream in ATLAS simulation

i.e. Jet mass reconstruction

Challenges

Assessing Performance

On two accounts very hard to draw quantitative statements

- Hard to compare different published methods
    - All optimised with different detectors and geometries - introduce their own challenges!
    - No clear way to compare all models on a standardised dataset
- But even within a single method hard to evaluate performance

One thing that is noticeable in most papers - no ratios!

- Modelling of core distributions does well, but disagreements still observed
- Most models still in development phases, how can we improve modelling?

Accurate particle simulation is one task - but how does it impact all physics objects!

- Individual jets contain many photons, electrons, pions, …

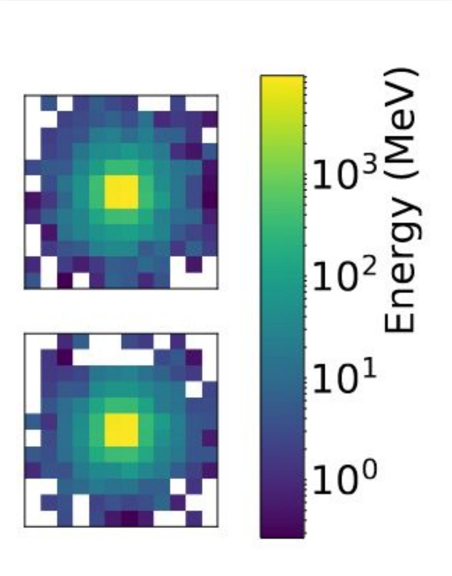
Assessing Performance

Output of generative models is energy depositions

- But individual showers don't tell us much
- Inherently stochastic, not easy to read

Which of these is a real shower, which is generated?

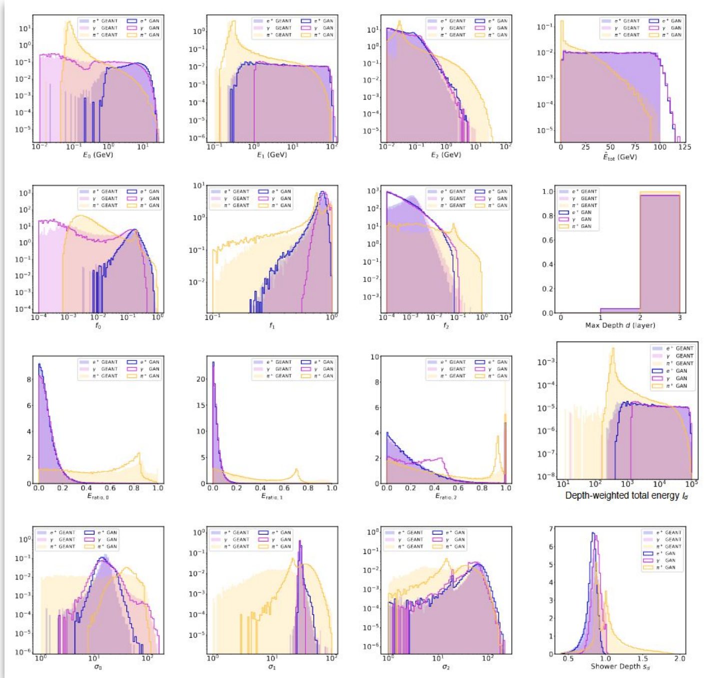
- Both look like showers!
- But for physics we need to have the right statistical properties for all showers -> distributions!
- Instead of looking directly output, focussing on complex correlations



Assessing Performance

Large number of distributions are interesting!

- Which of these are the most important?
- How to “weight” better modelling of one versus another
- How to combine modelling of many observables into a single “score”
    - Necessary for epoch picking!
    - Classifiers can help but aren’t a final solution
    - A lot of final choice comes down to by-eye model selecting, or “physicists intuition”



Input data challenges

Sparsity of deposits and reproducing high dynamic range still a tough ask!

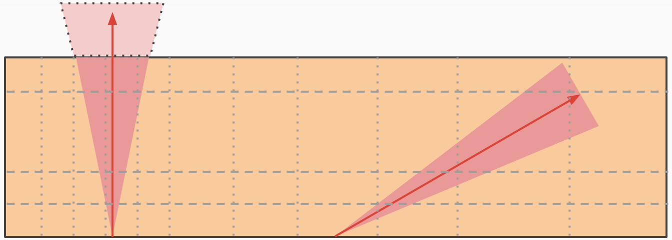
Getting tails of distributions hard

- Less common events in training
- But often key descriptors of physics

Need to accurately learn the PDF of showers including the stochasticity

- How to enhance focus on reconstruction of tails but keep this property
- Only a handful of conditional parameters - particle four vector

Input data challenges



Detector geometries aren't consistent

- Changing granularity of readout
- Changing materials
- Angles of incidence for particles have different traversal lengths!

Not limited to designing methods around readout geometry

- But implicitly will impact shower development!

Future Directions

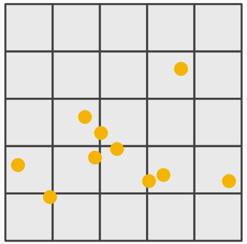
Point Clouds

Not restricted to using readout or detector

- Use more natural representation?

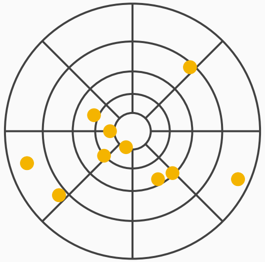
Showers produce

- Variable number of energy deposits
- Correlated spatial ordering but no natural ordering



Map hits to cells

Doesn't generalise with consistent geometry



Map hits to polar grid

Shower evolution cone shaped

To remove edge effects need fine, sparse, binning

Point cloud

Direct translation of input data
Use graph networks!
Preserve natural symmetries

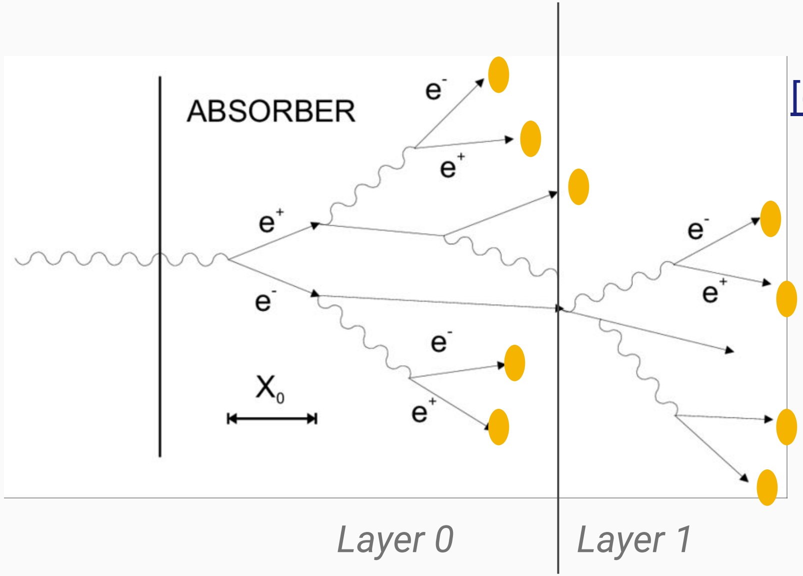
Point Clouds

But! - Don’t record history of shower development

- Each point (E, x) is end of a branch!
- Multiplicity can be incredibly high
- Points aren’t directly connected
- Density of points between layers correlated but cannot easily build a physical graph
- Typically no time information, just layer depth

And for sampling calorimeters dead material

- Low sampling fraction - lots of missed hits!



[6]

Current point cloud models aren’t designed to learn underlying structure - they focus on learning surfaces or connected structure! Difficult to use standard benchmarks!

Not the whole hog

Instead of wholesale replacement with generative models

- Replace individual parts of Geant4 with ML
- Use ML for reweighting/resampling on top of current fast methods

Conclusions

A lot of great promise from generative modelling to speed up detector simulation

Unique playground for developing new models

- Challenges in HEP very different to computer vision

Focus on modelling complex underlying correlations and distributions with highly stochastic, sparse, data with high dynamic range

- Even getting the total energy correct not straightforward

By no means a closed book! Lots of room for improvement - Exciting times lie ahead!