



# 3D convolutional GAN for fast simulation

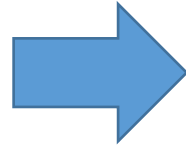
*ACAT 2019*

F. Carminati, G. Khattak, V. Loncar, T. Nguyen, F. Pantaleo,  
M. Pierini, **S. Vallecorsa**, J-R. Vlimant

# A DL tool for fast simulation

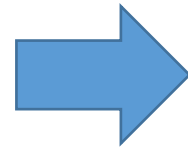
*Provide a tool that can be configured and trained for different detectors*

- Are generative models accurate enough?
- Can we sustain the increase in detector complexity?



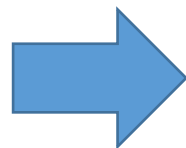
- A first proof of concept
- Understand performance and validate accuracy

- How generic is this approach?
- Can we “adjust” architecture to fit a larger class of detectors?
- What resources are needed?



- Prove generalisation is possible
- Understand and optimise computing resources

- Can it be integrated in a “standard” simulation workflow?



- Pre-processing
- Interfaces to simulation engine

# Proof of concept, benchmarking and validation

# CLIC calorimeter simulation

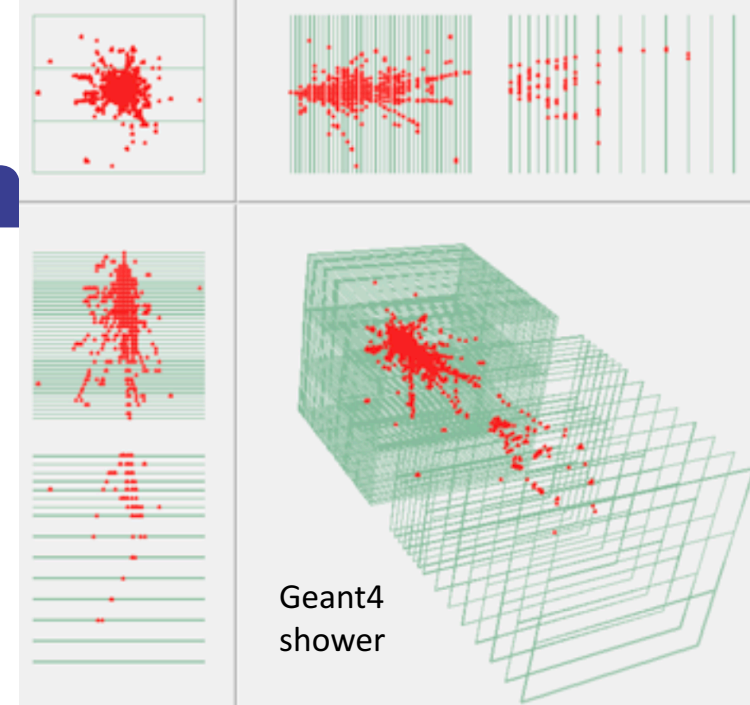
*Data is essentially a 3D image*

Electromagnetic calorimeter detector design<sup>(\*)</sup>  
(Linear Collider Detector studies)

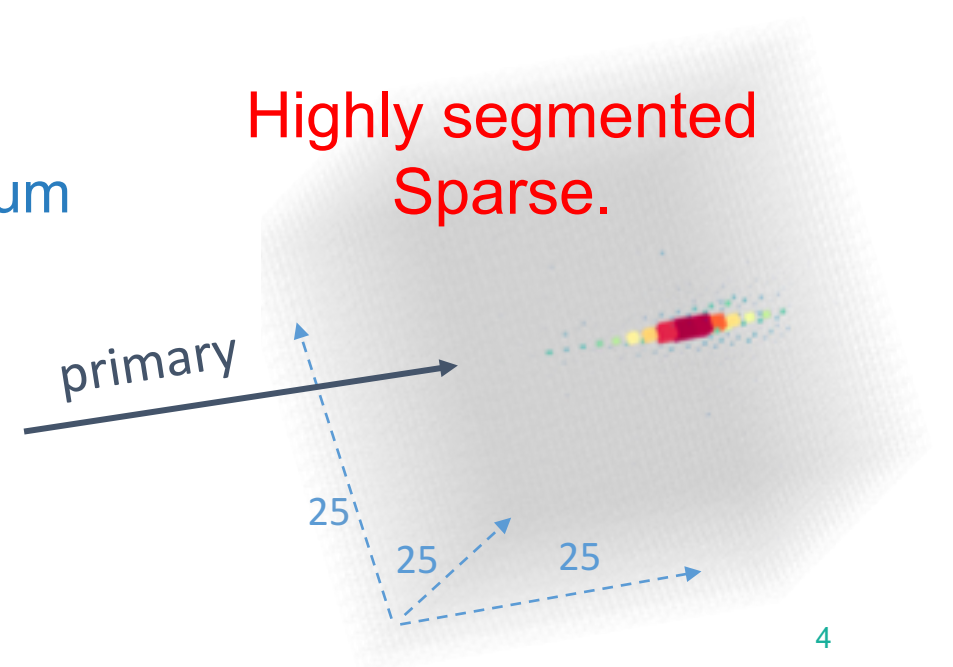
- 1.5 m inner radius, 5 mm×5 mm segmentation: 25 tungsten absorber layers + silicon sensors

1M single particle samples (e,γ,π) with flat energy spectrum (10-500) GeV

1. Orthogonal to detector surface (25x25x25 pixels)
2. +/- 30° random incident angle (51x51x25 pixels)



Highly segmented  
Sparse.



<sup>(\*)</sup> <http://cds.cern.ch/record/2254048#>

# 3D convolutional GAN

Similar discriminator and generator models

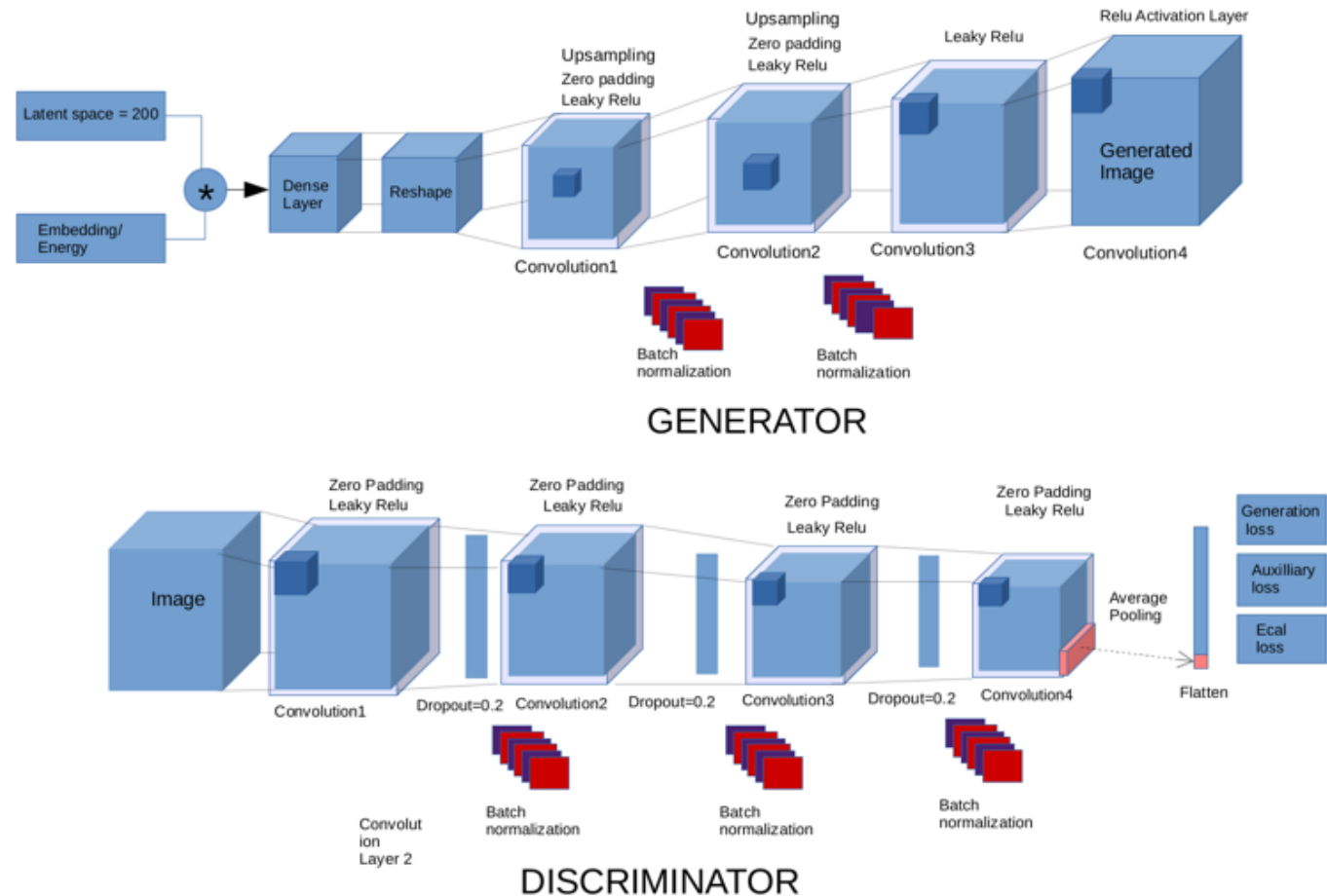
**3D convolutions** (keep X,Y symmetry)

~1M parameters total

**Condition** training on input variables (energy, angle)

**Auxiliary regression tasks** assigned to the discriminator: cross check

**Custom losses**



# Validation and optimisation

Detailed GAN vs GEANT4 comparison (More than 200 Plots! )

- High level quantities (shower shapes)

- Calorimeter response (single cell response)

- Particle properties (primary particle energy)

Optimisation on

- Network Architecture (Layers, filters, kernels, initialisation)

- Losses definition

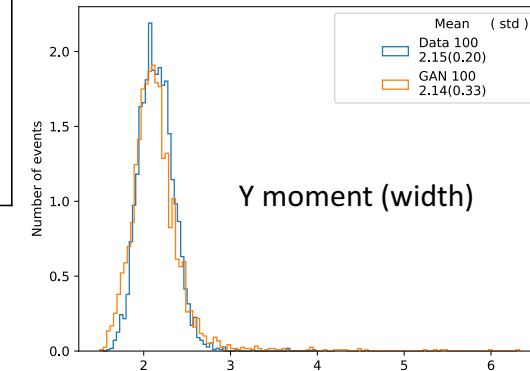
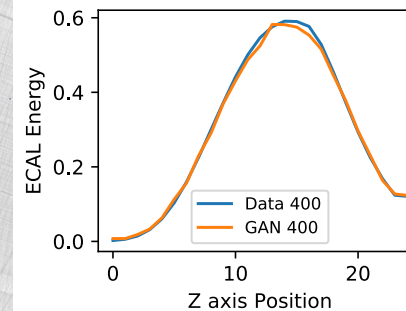
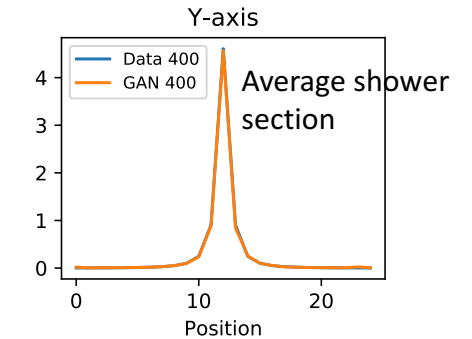
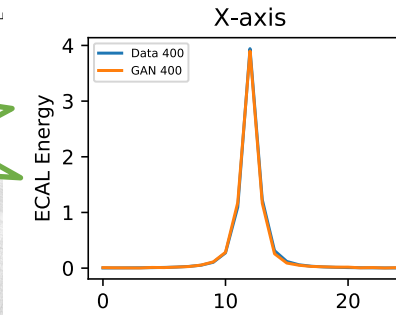
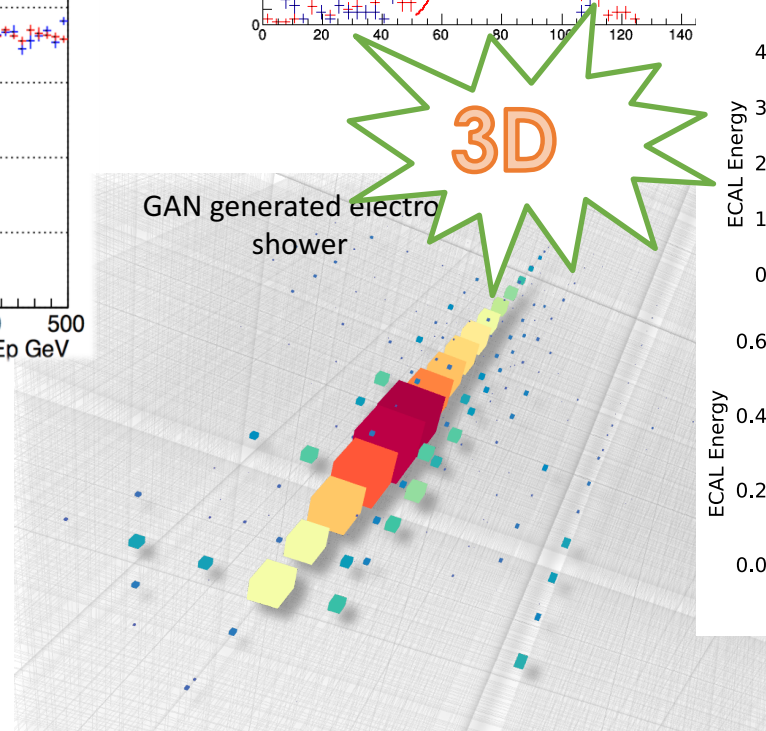
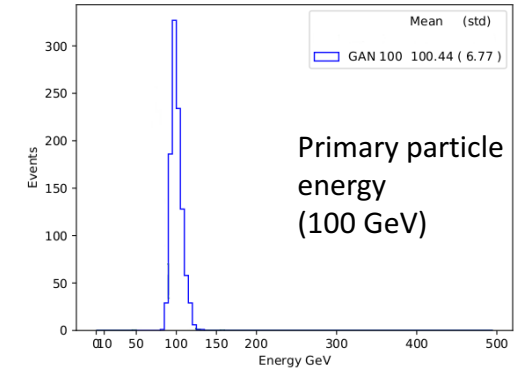
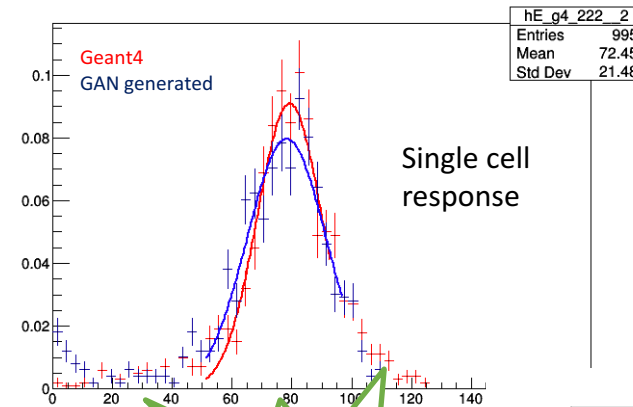
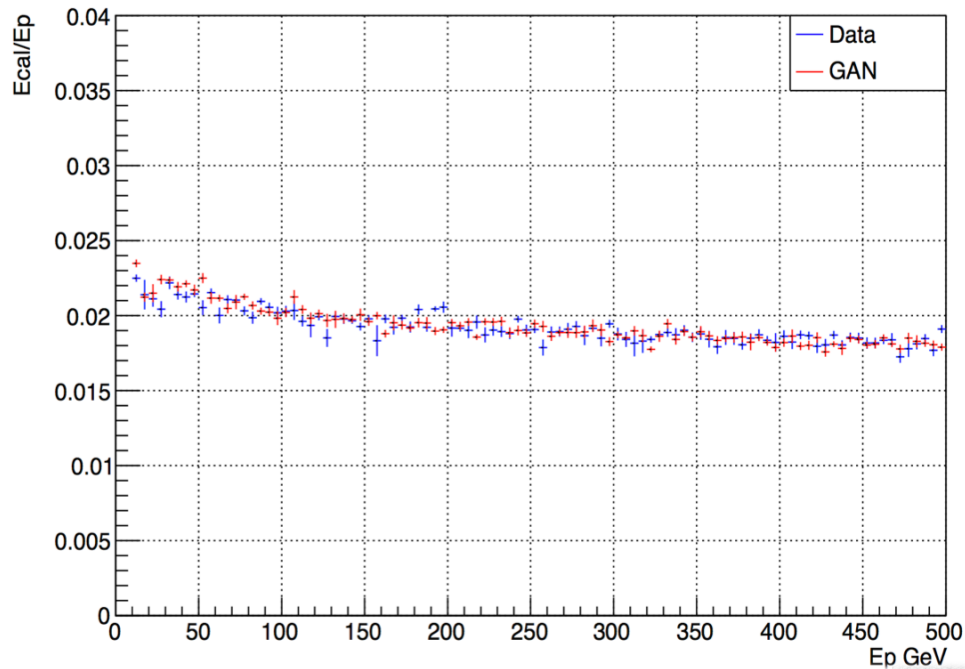
- Data pre-processing

Results agree within a few % to Geant4 (sometimes labelled “DATA” in next slides 😊 )

Run TriForce classification and regression engine on GAN generated data

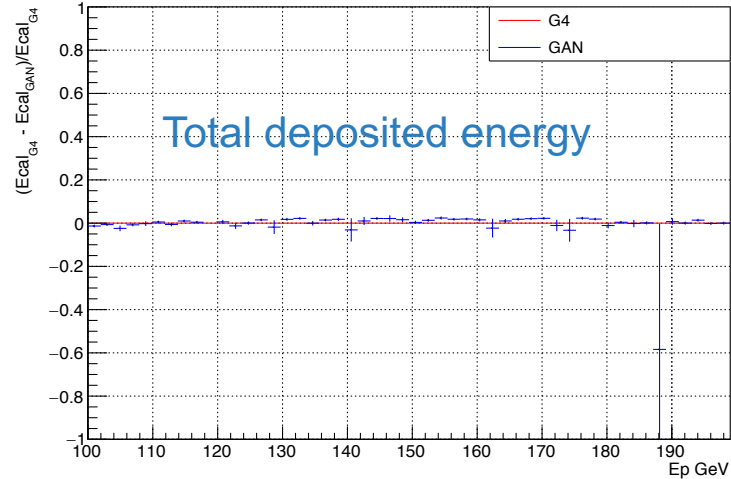
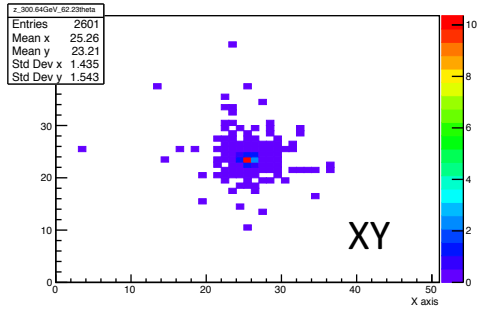
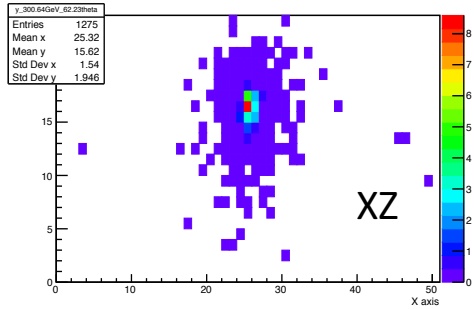
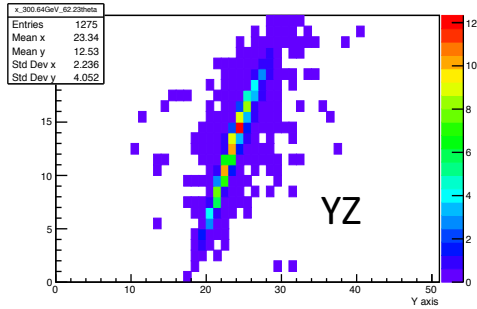
(Matt Zhang, [https://github.com/BucketOfFish/Triforce\\_CaloML](https://github.com/BucketOfFish/Triforce_CaloML))

# Orthogonal incident angle sample



# A more realistic sample

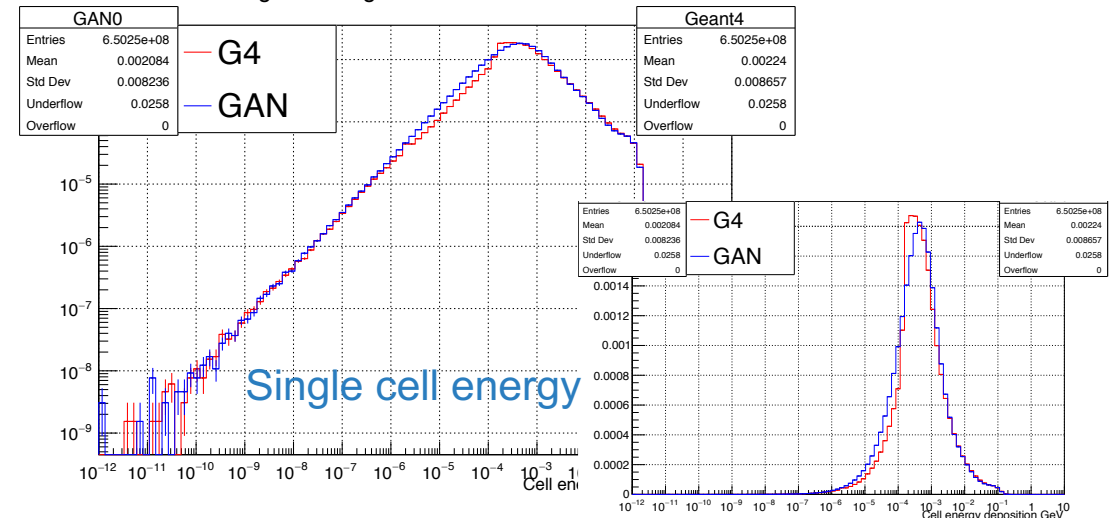
Electrons enter the calorimeter within a  $60^\circ$ - $120^\circ$  angle range



Wider/asymmetric image size (51x51x25)

Minimal architecture changes “by hand”

- Adjust convolution parameters
- Additional terms in the loss function:
  - Angle-related term
  - Constrain energy spectrum



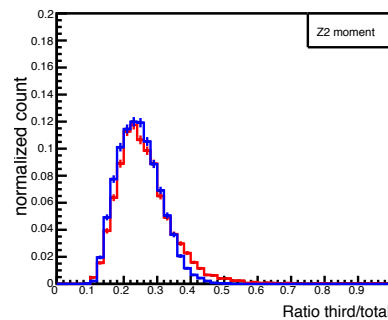
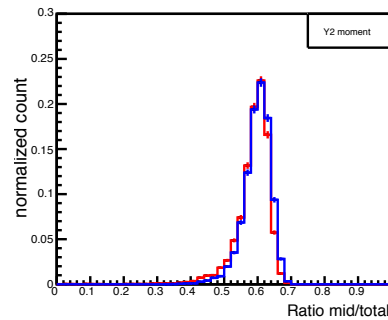
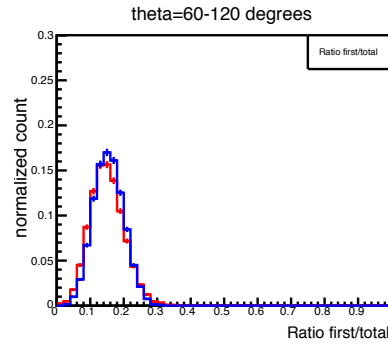


# Angle dependence (I)

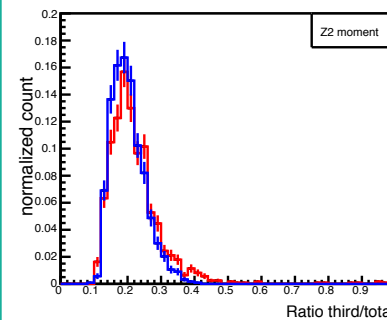
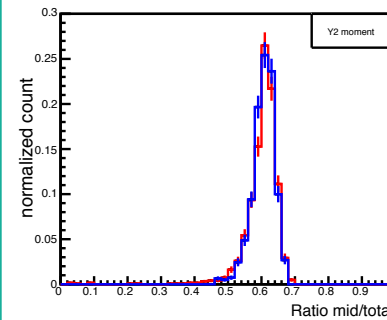
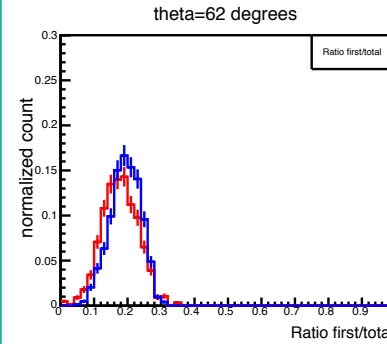
Spit calorimeter in 3 layers  
along its depth  
(cells: 1-8, 8-16, 24-25)

Measure energy deposited  
in layers wrt total

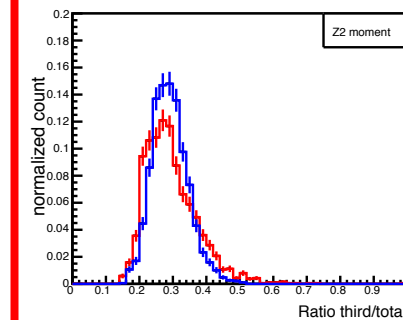
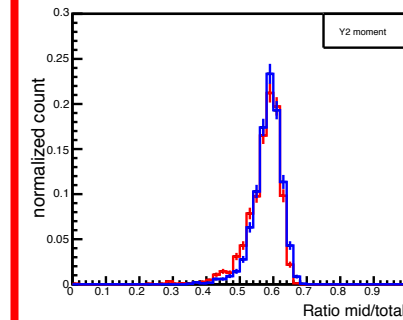
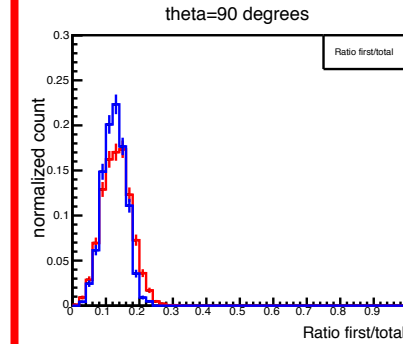
inclusive



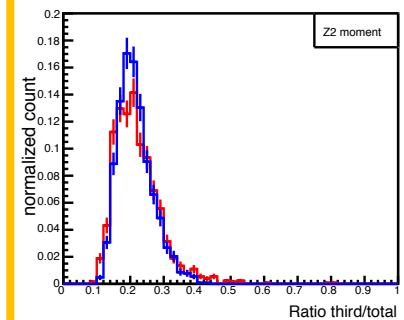
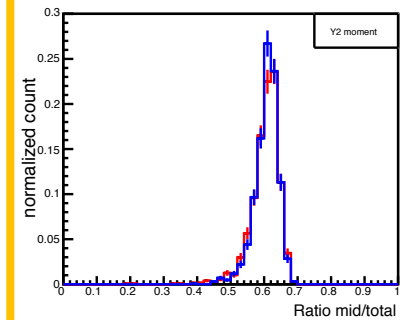
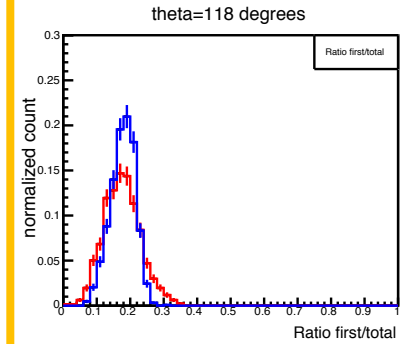
60° incident angle



Orthogonal



120° incident angle

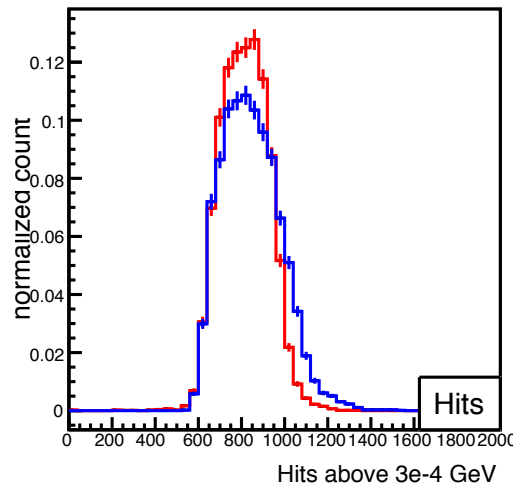
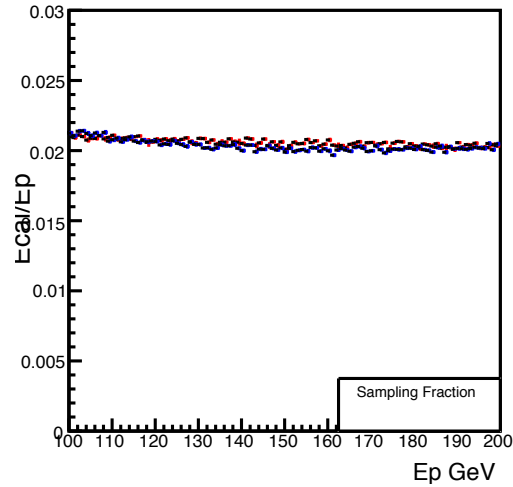


# Angle dependence (II)

Sampling fraction and number of hits

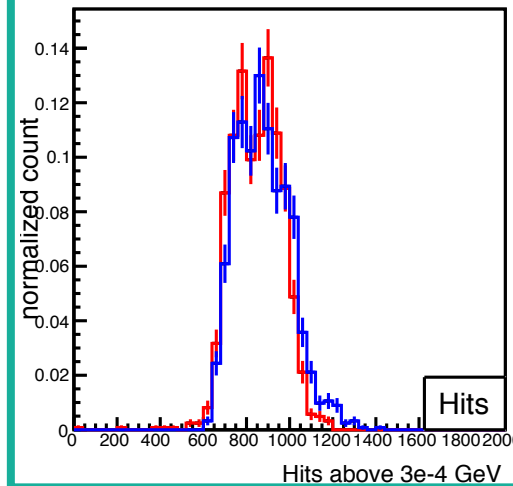
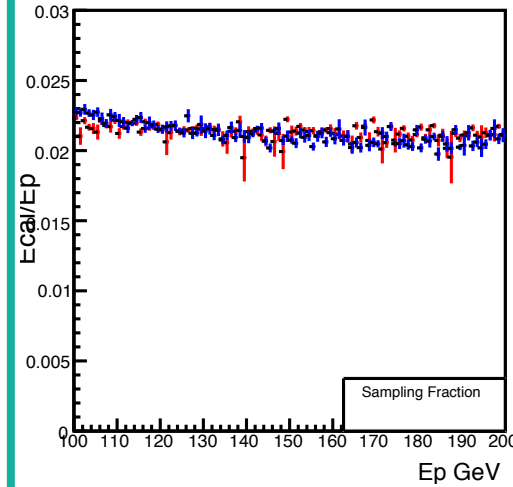
inclusive

theta=60-120 degrees



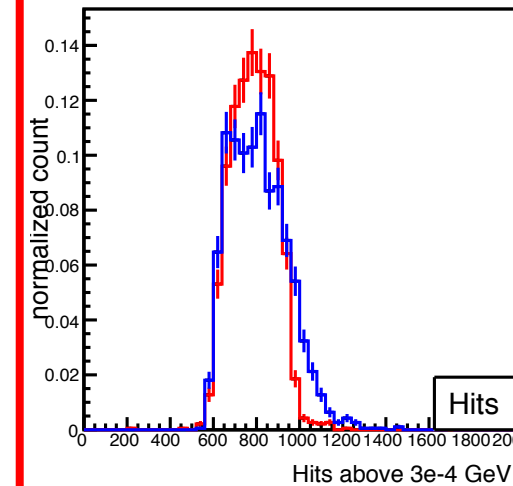
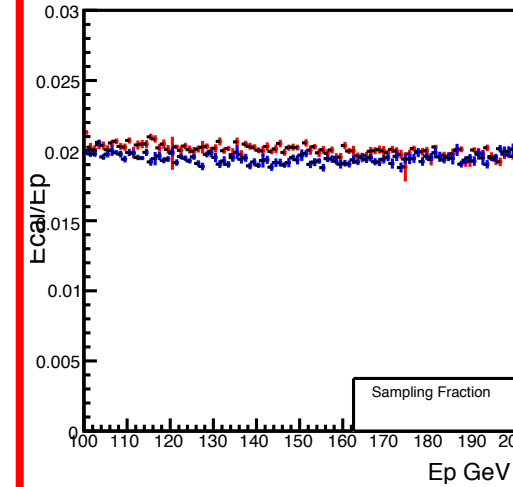
60° incident angle

theta=62 degrees



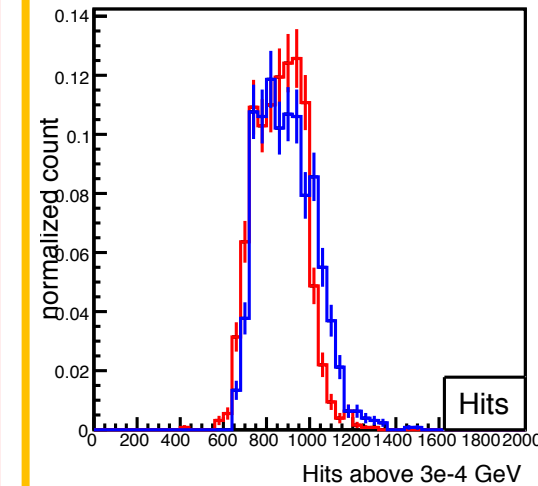
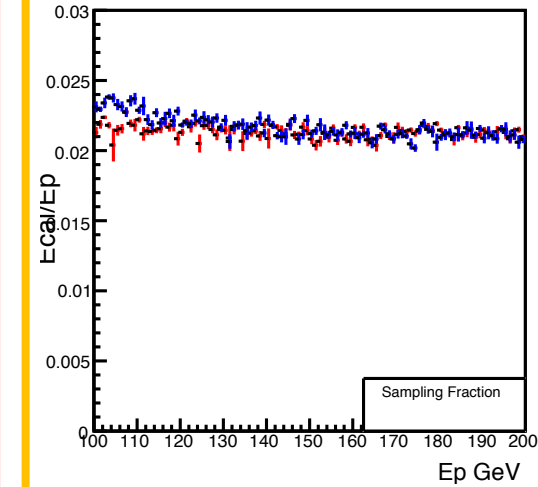
Ortogonal

theta=90 degrees



120° incident angle

theta=118 degrees





# Computing resources & Generalisation

Distributed training

Hyper-parameter scans

# Computing performance

*Estimate performance for the orthogonal incident angle sample*

## Inference:

Geant4: 17 s/particle vs 3DGAN: 1 ms/particle

- speedup factor > 10000!!
- Testing inference on accelerators (GPUs, FPGAs) and dedicated hardware

## Training:

45 min /epoch on Tesla P100

Introduce **data parallel** training based on MPI

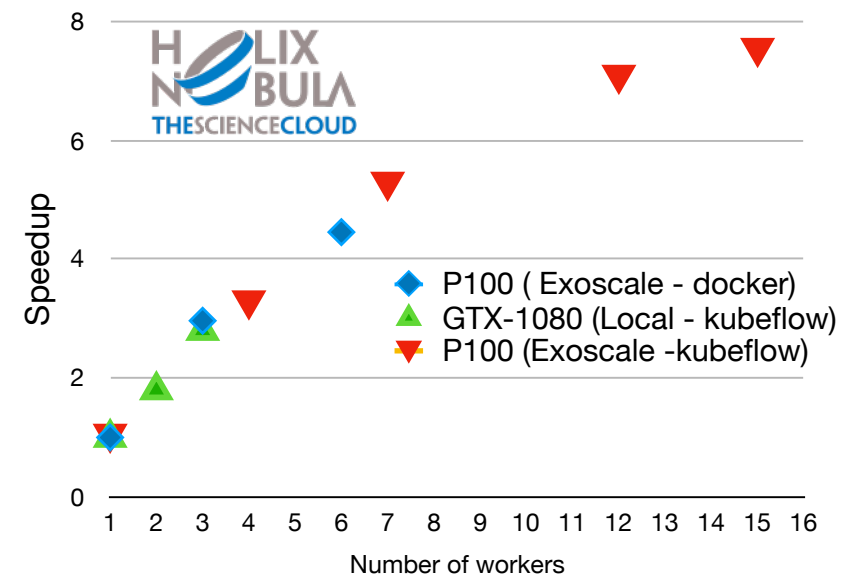
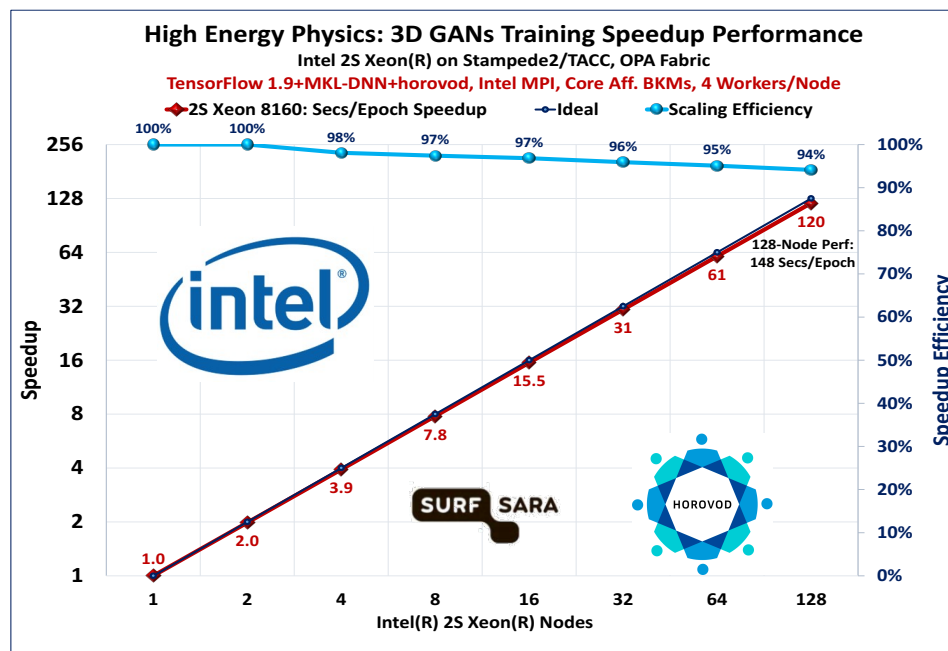
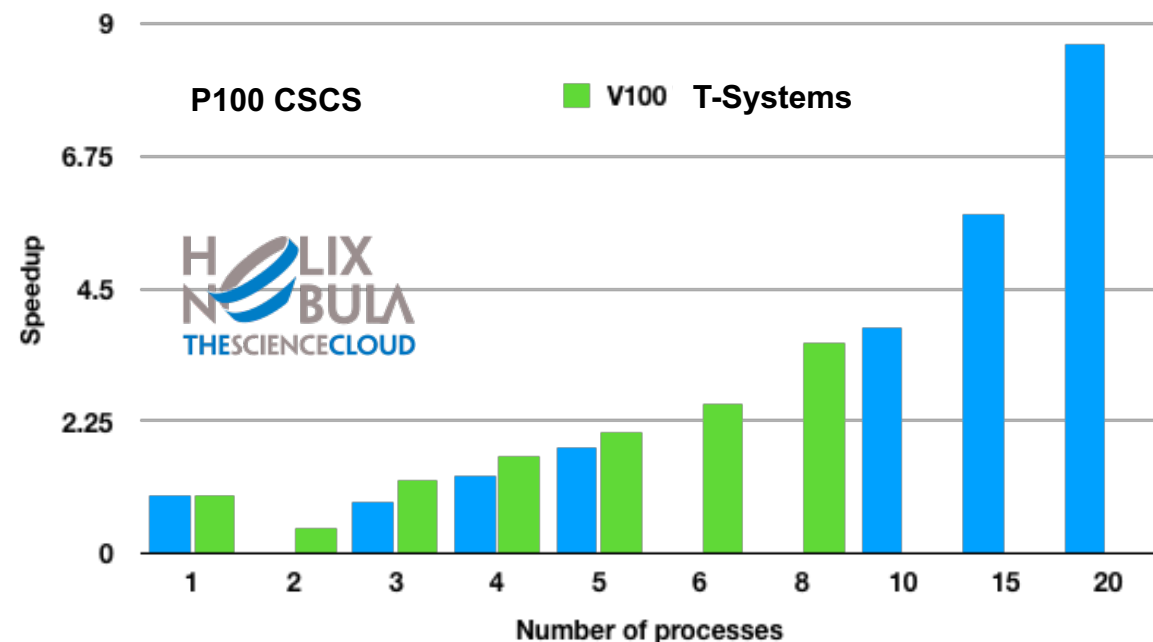
Test several libraries

Run on HPC clusters and Cloud (HNSciCloud providers)

Time to create an electron shower		
Method	Machine	Time/Shower (msec)
<b>Full Simulation (geant4)</b>	Intel Xeon Platinum 8180	17000
<b>3d GAN (batch size 128)</b>	Intel Xeon Platinum 8180 (TF 1.12)	1

# Distributed training

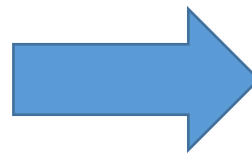
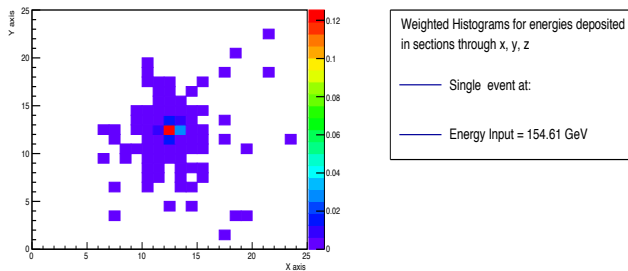
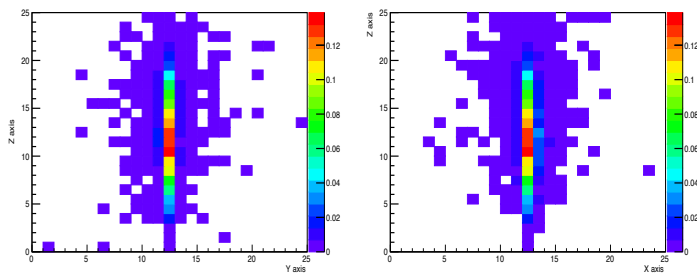
- Cloud deployment via docker + Kubernetes/Kubeflow (R. Rocha CERN IT-CM)
- Frameworks
  - Horovod
  - mpi\_learn
- Hardware resources
  - Cloud (HNSciCloud)
  - HPC centers (Oakridge – TACC)



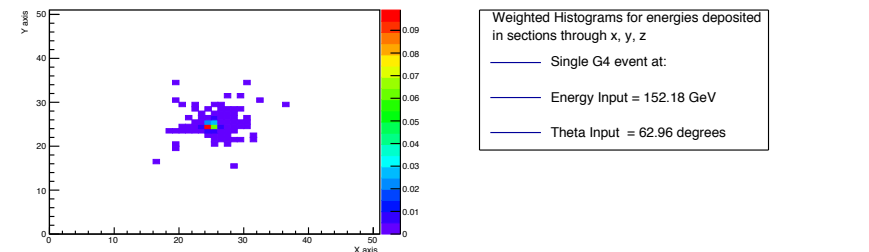
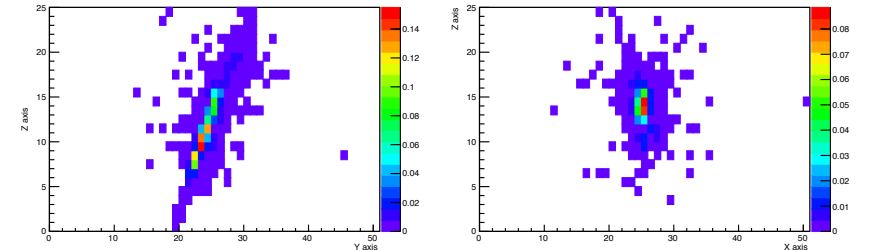
# Generalisation

*How much can we generalise our network to other calorimeters?*

From the orthogonal incident angle to the realistic sample



Change image size  
Change shower shapes



Adjust network hyper-parameters (convolutional layer filters):

reduce number of filters

Increase filter size on the transverse plane

# Generalisation

*Training and architecture hyper-parameters optimisation*

Different geometries, read-out patterns, energy scales

Tuning the right architecture cannot be done by hand

Full parameter scan is resource/time consuming.

Test different optimisation approaches:

Sequential Model-Based Optimization

Optimize initial architecture candidate, defining a finite set of states to explore

Reinforcement Learning

Network accuracy is the **reward function**. Architecture or hyper-parameter modification are **actions**

Evolutionary Algorithms

Can allow simultaneous weights training and architecture optimisation

**mpi-learn integrates a optimisation engine (mpi-opt)**



# mpi-opt

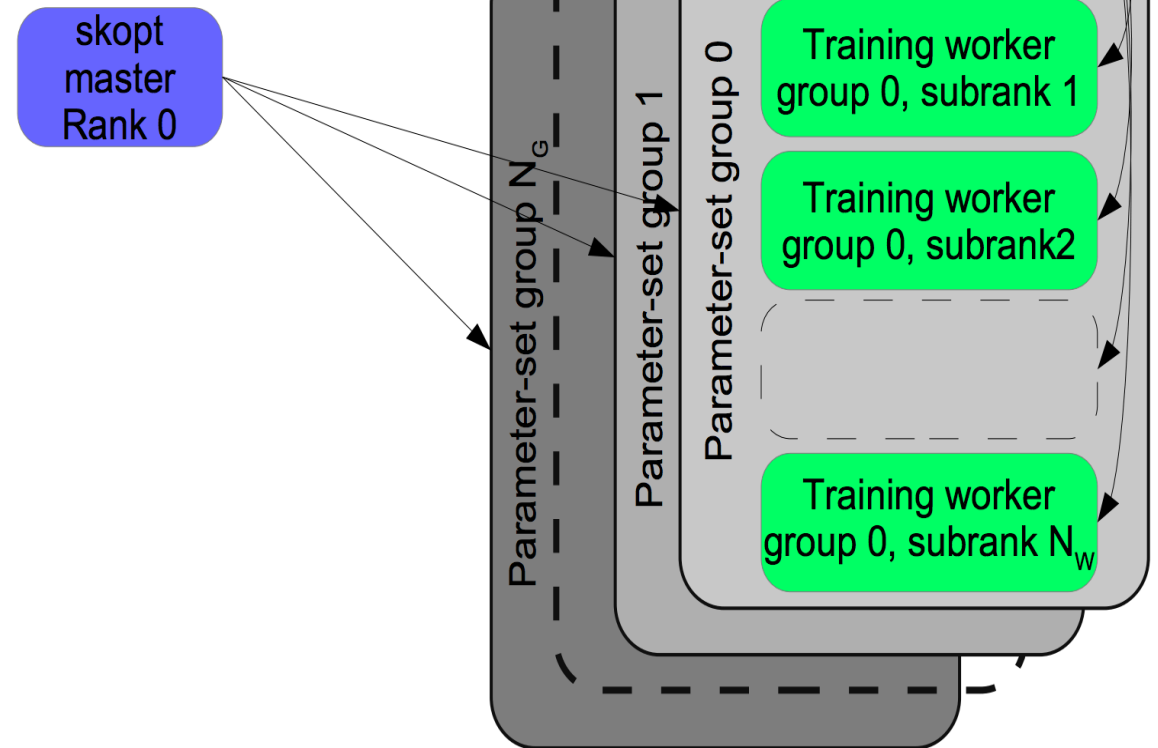
## Parallelise optimisation with mpi-learn

Bayesian optimiser

Evolutionary approach

- One master runs optimisation (skopt)
- $N_G$  groups of nodes train on a parameter-set
  - One training master
  - $N_W$  training workers
- Can also run in “sub-masters mode”
  - Sub-masters perform intermediate averaging

### Basic Configuration



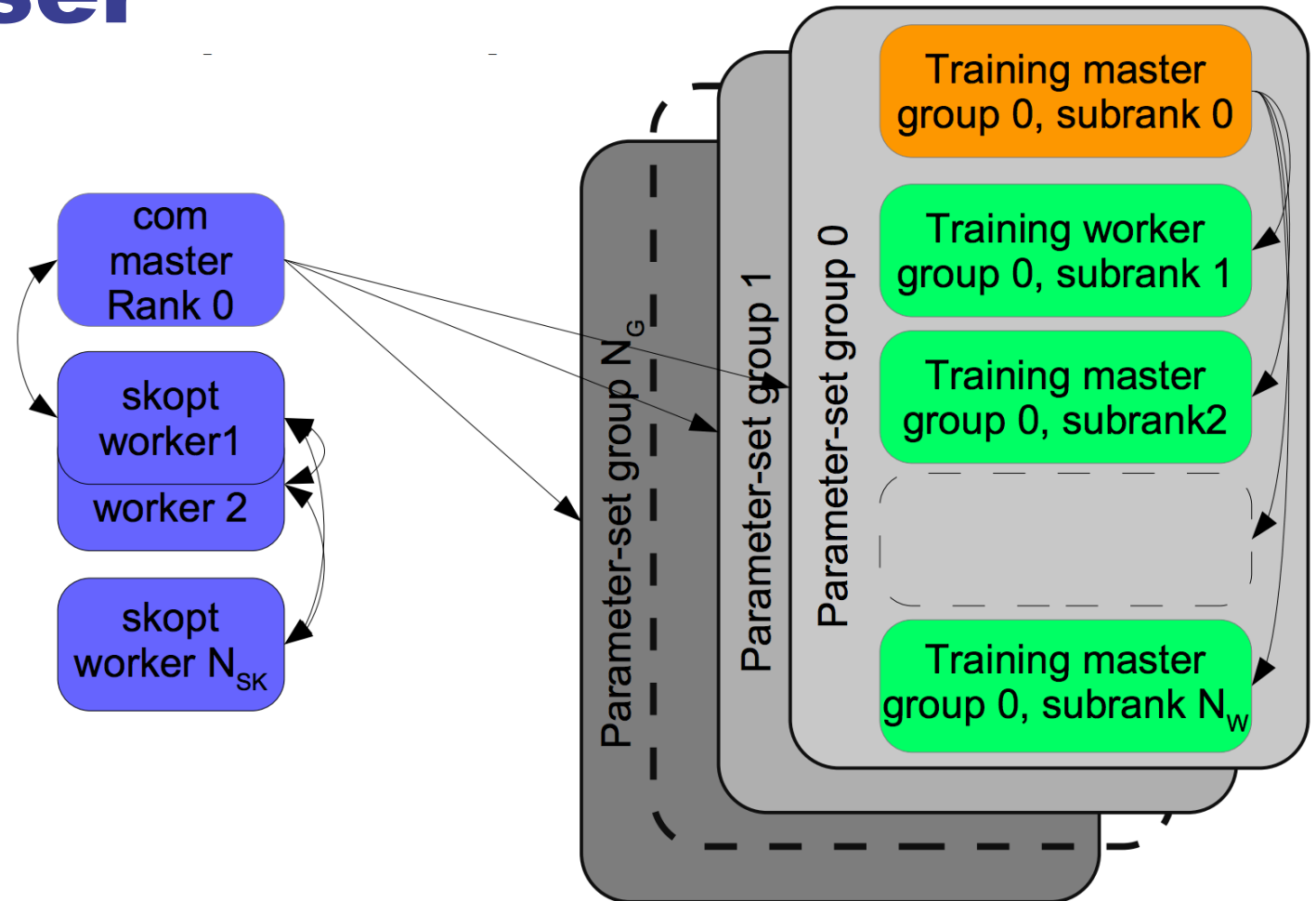
# Parallel optimiser

One master running communication of parameter set

$N_{SK}$  workers running the bayesian optimization

$N_G$  groups of nodes training on a parameter-set on simultaneously

- One training master
- $N_W$  training workers



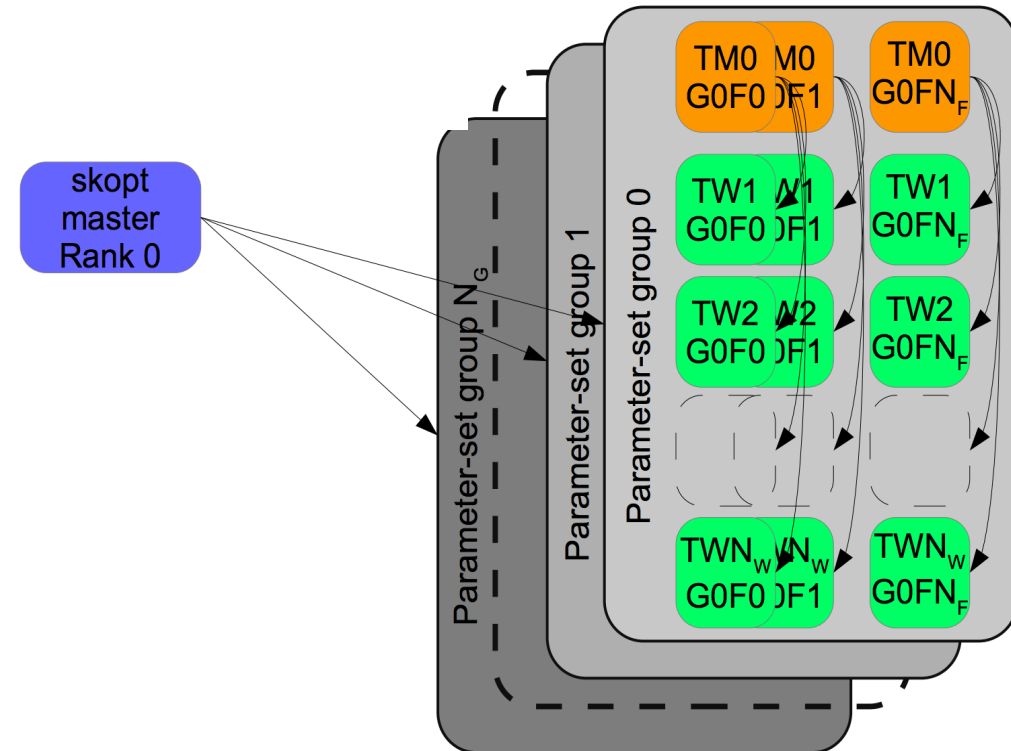
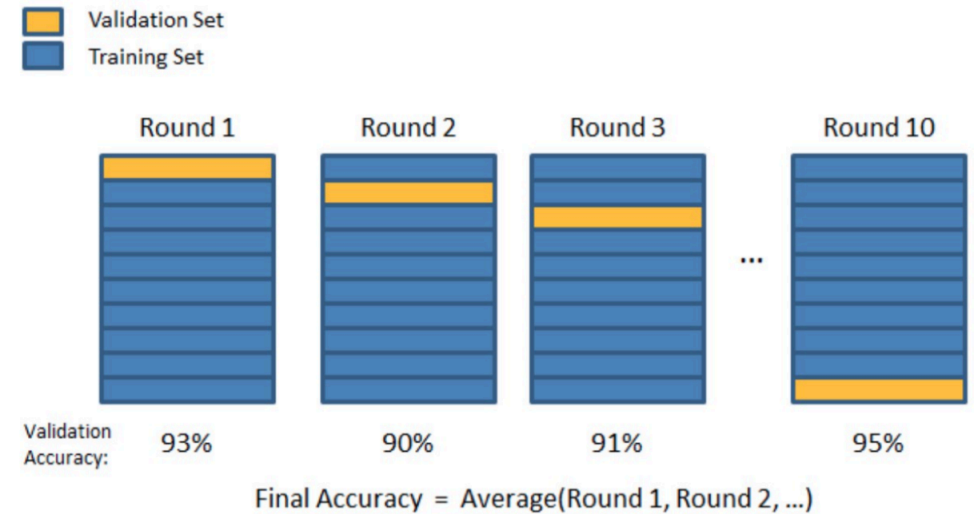
prevent working groups to be idle while the optimisation fit is performed

# K-folding cross validation

Estimate performance over different validation parts of the training dataset

Account for variance from multiple sources

- One master runs the bayesian optimization.
- Receives the average fom over  $N_F$  folds of the data
- $N_G$  groups of nodes training on a parameter-set simultaneously
- $N_F$  groups of nodes running one fold each
  - One training master
  - $N_W$  training workers



# Summary & Plans

3DGAN: first step towards customizable simulation tool

Agreement to Monte Carlo within few percent

Work in CERN simulation group to test integration

Meta-optimization and hyper-parameters scans are key

Test of several distributed training approaches

Understand / optimize performance at scale

We are working heavily on “technological/computing” aspects in collaboration with industry (Intel, IBM, Google)

Test different platforms for inference and training: CPUs, accelerators (GPUs, FPGAs)

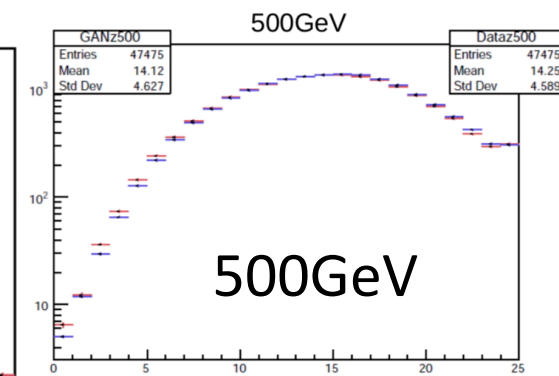
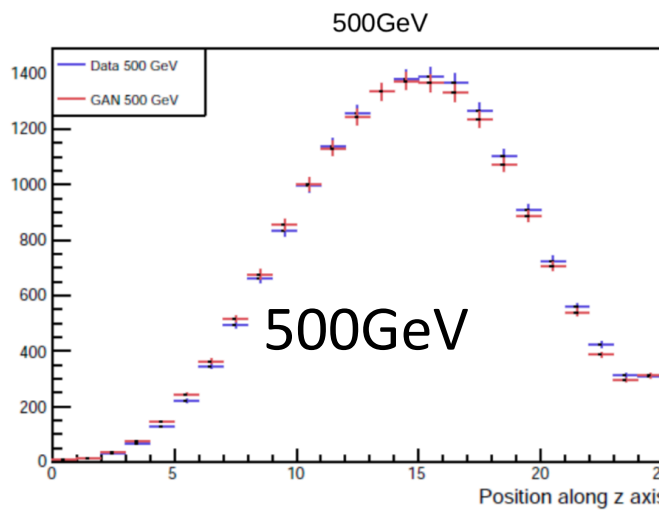
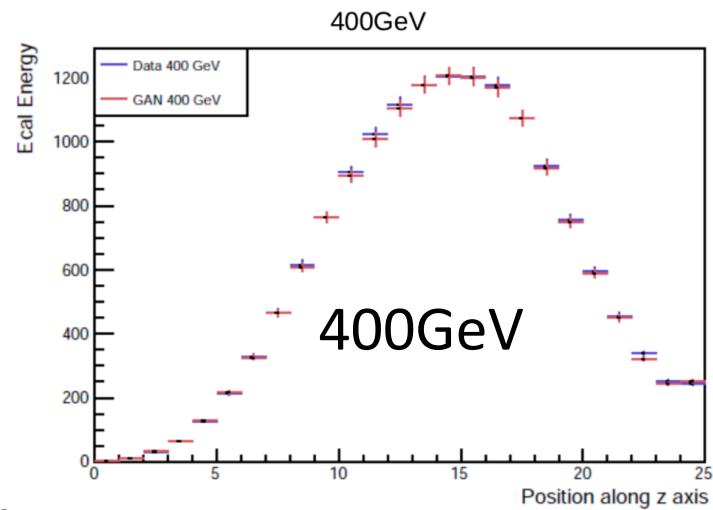
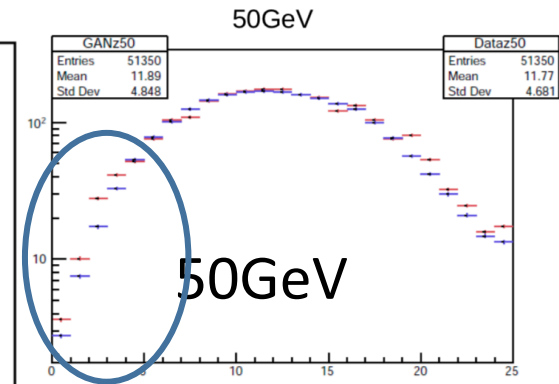
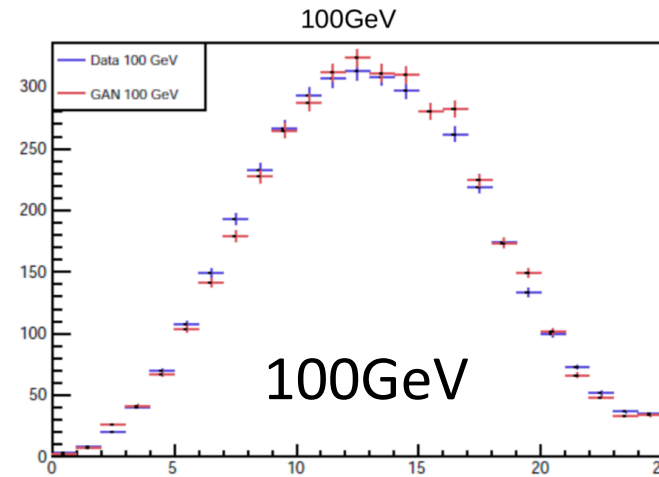
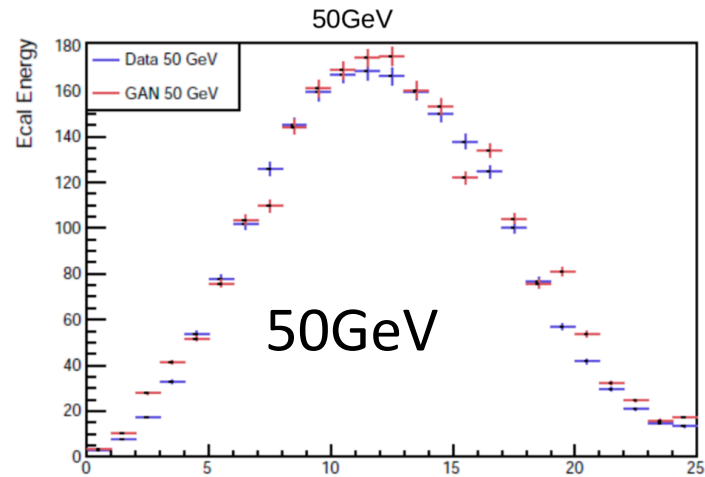


# Questions?

*Sofia.Vallecora@cern.ch*

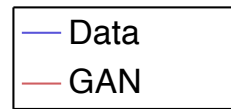
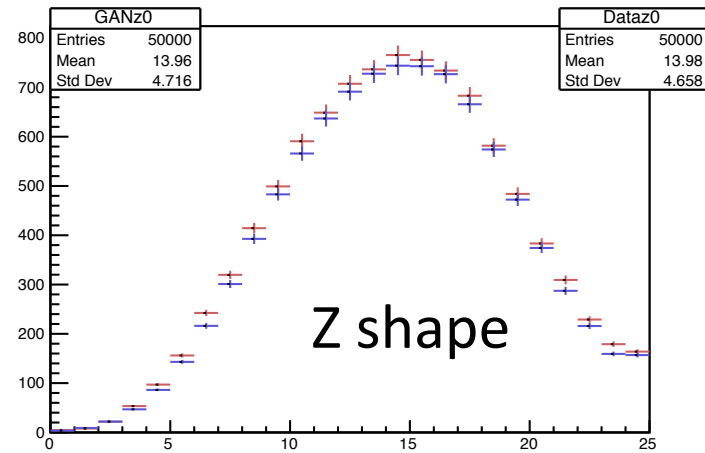
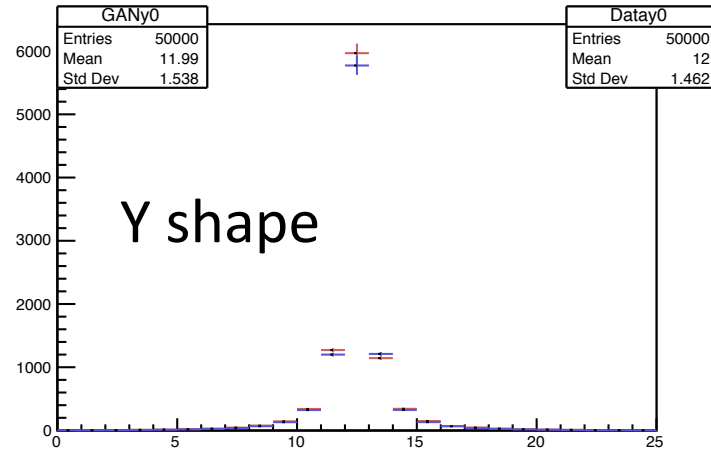
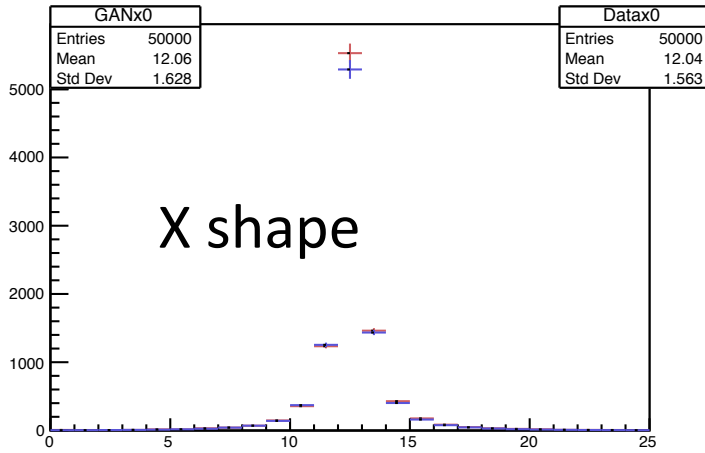
# Thanks!

# Electrons shower shapes

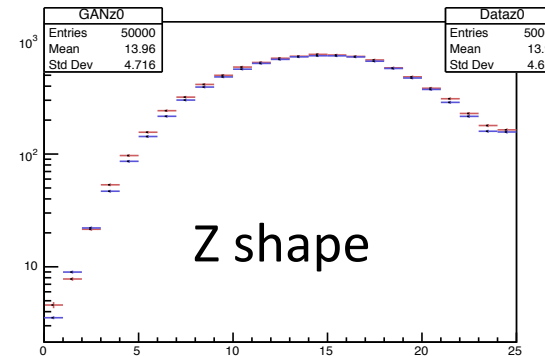
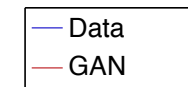
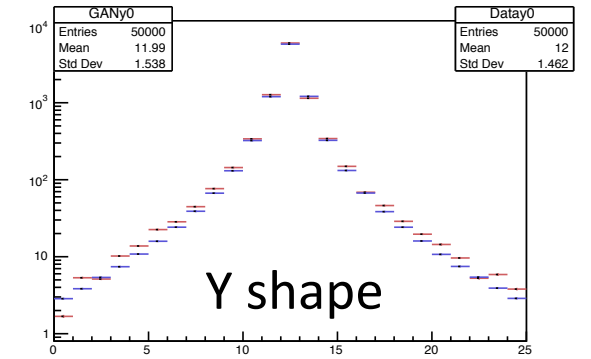
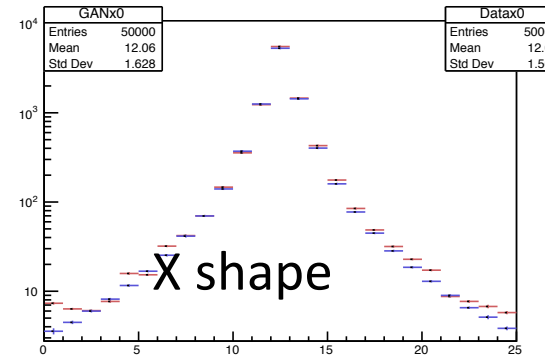


# Neutral Pions

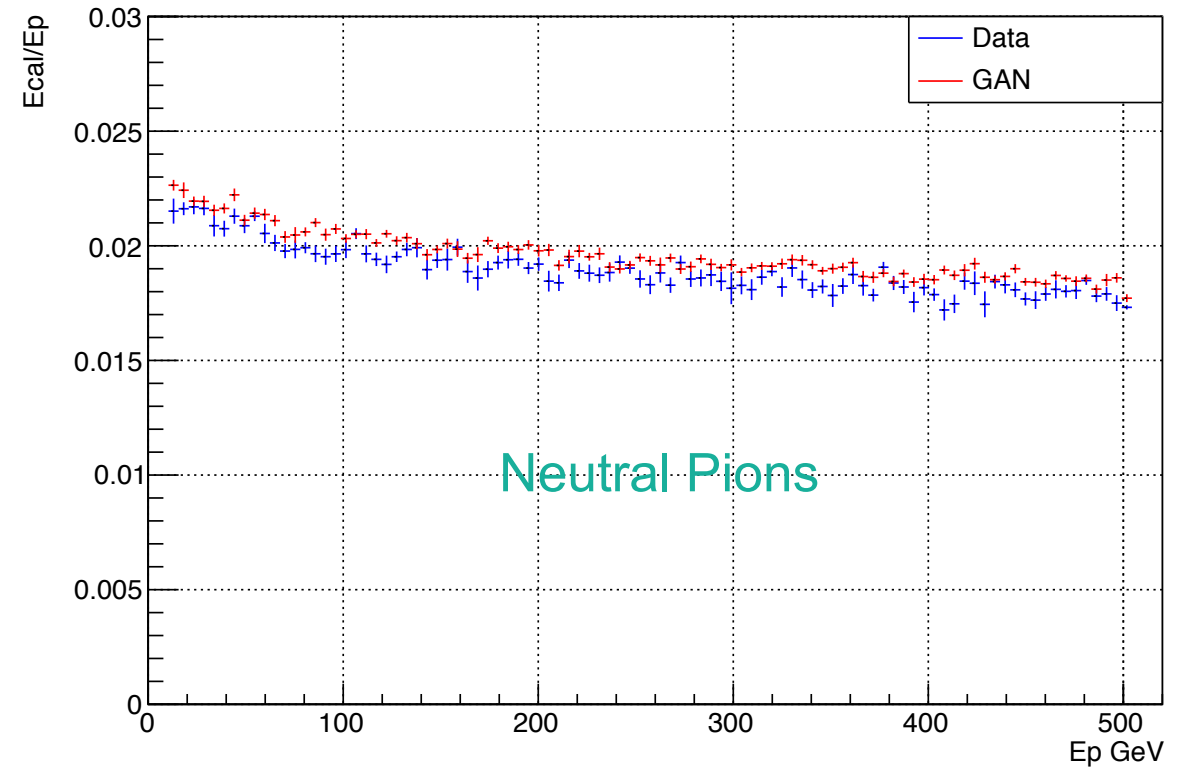
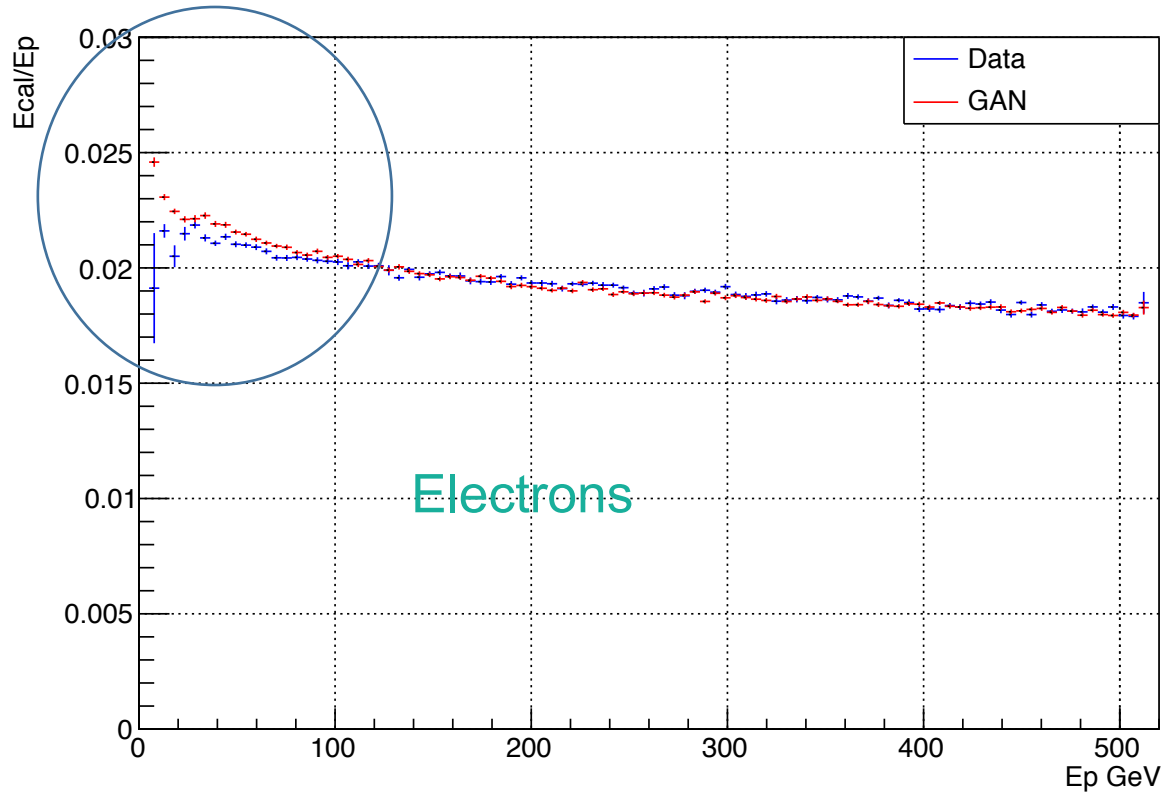
10-500 GeV



Log scale



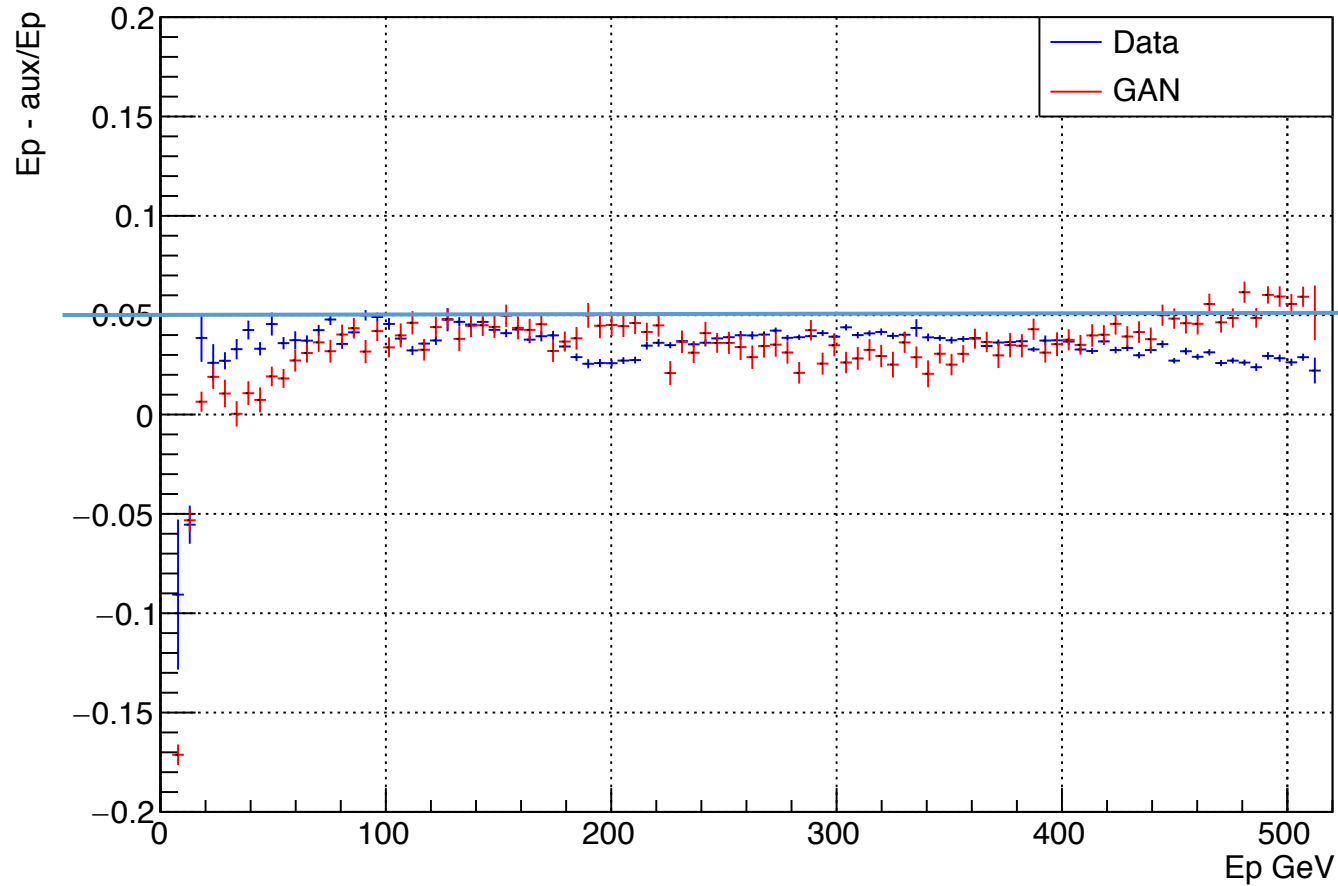
# Deposited energy and sampling fraction



GAN seems to slightly overestimate slightly neutral pions energy deposits



# Discriminator regression on input energy



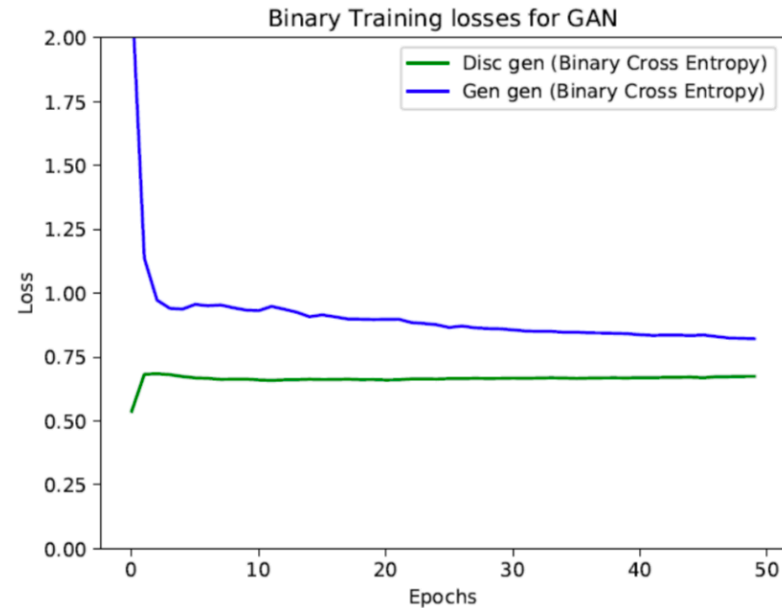
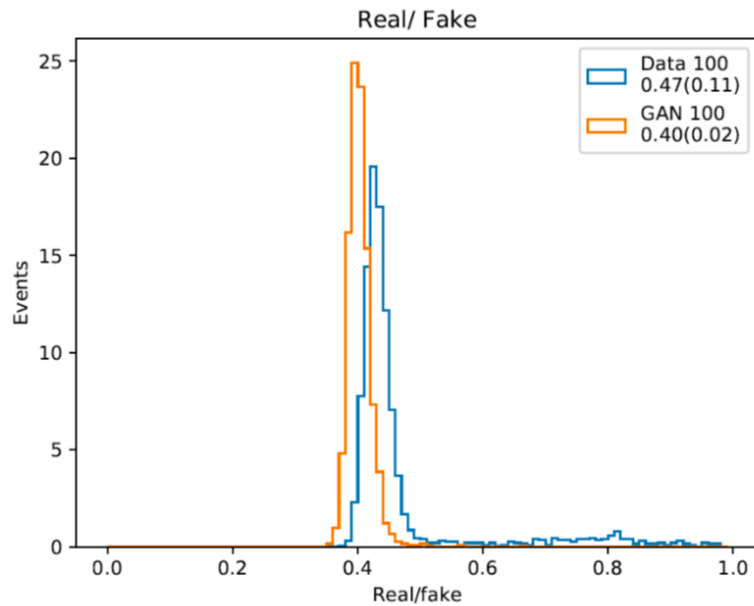
5% error on  
auxiliary energy  
regression

# Conditioning and auxiliary tasks

Loss is linear combination of 3 terms:

Combined cross entropy (real/fake)

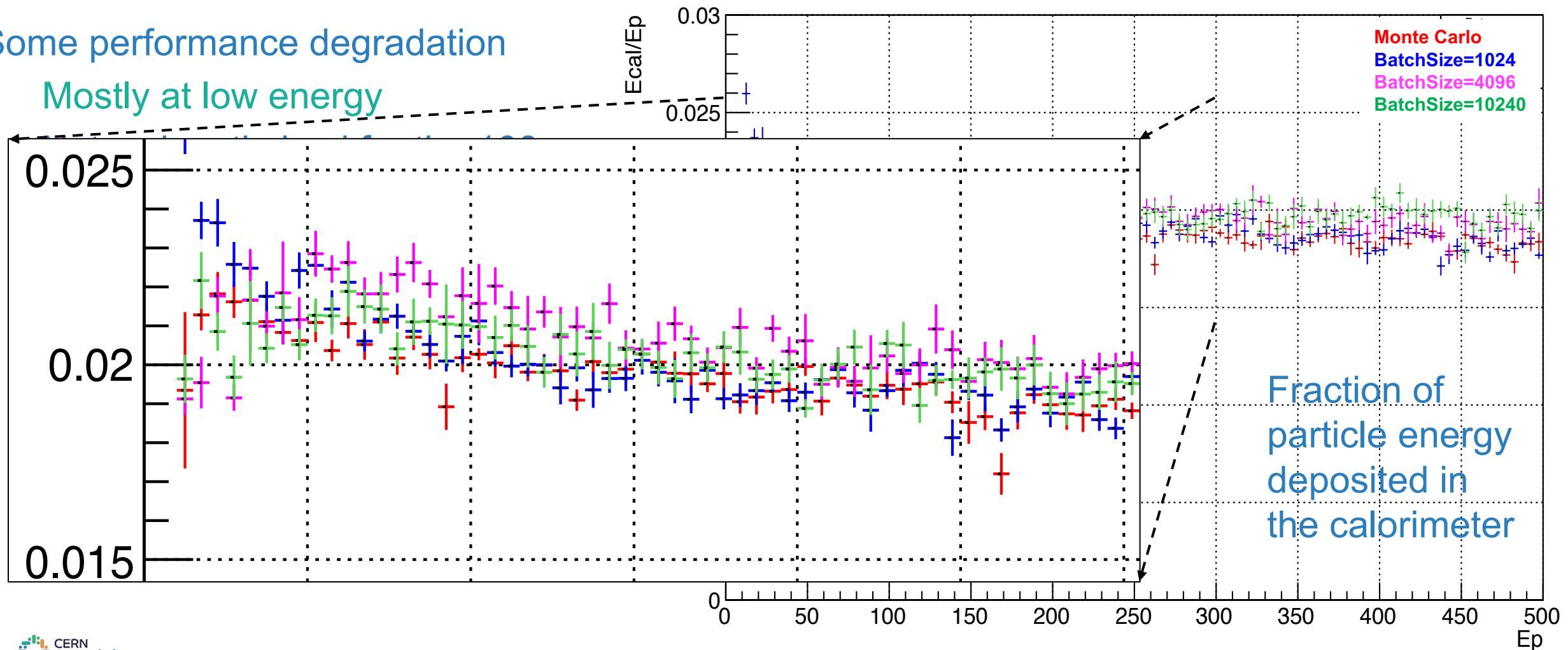
Mean absolute percentage error for regression tasks



# Physics performance

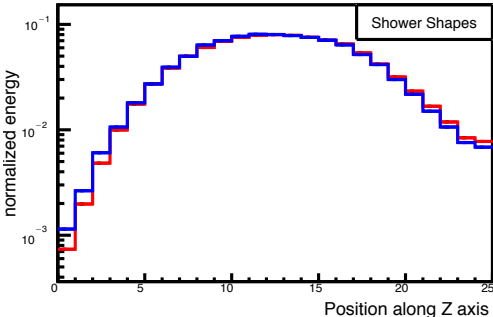
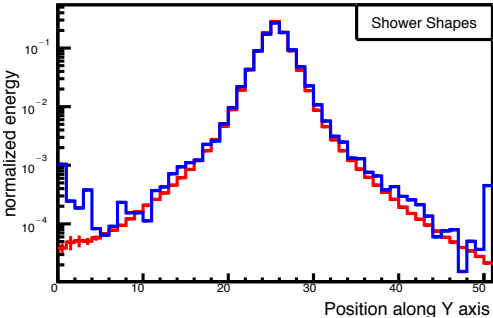
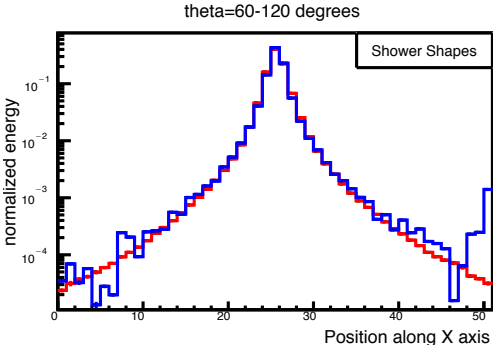
Some performance degradation

Mostly at low energy

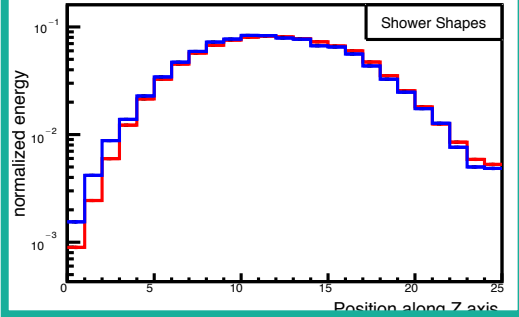
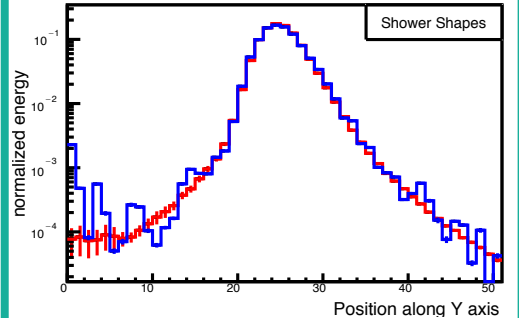
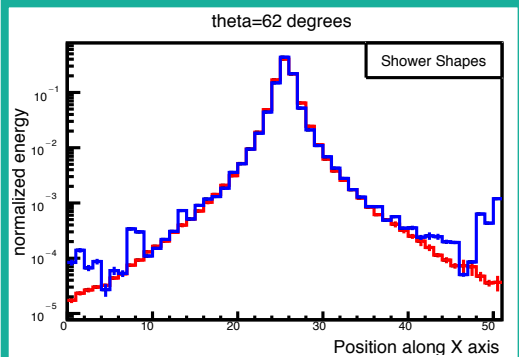


# Angle dependence: shower shapes

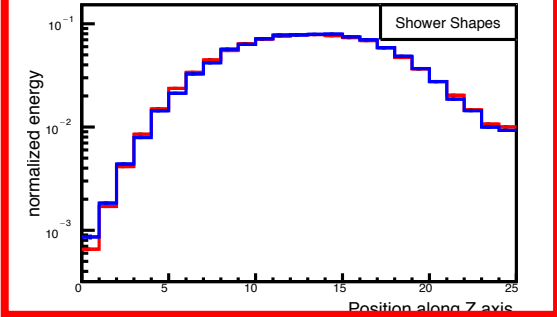
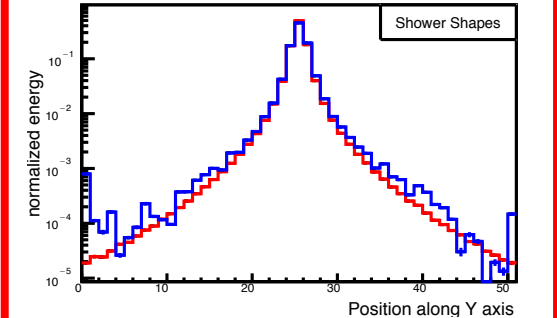
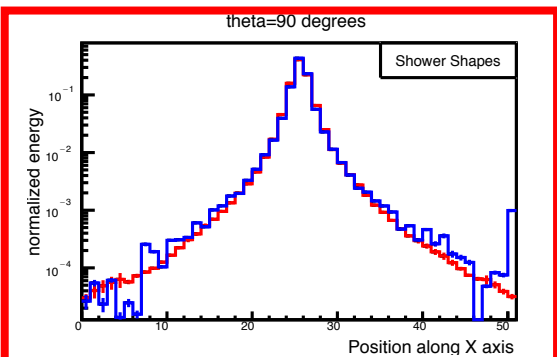
inclusive



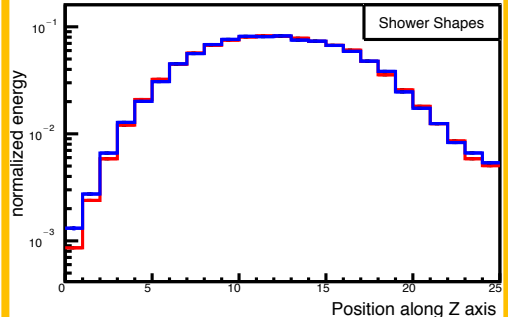
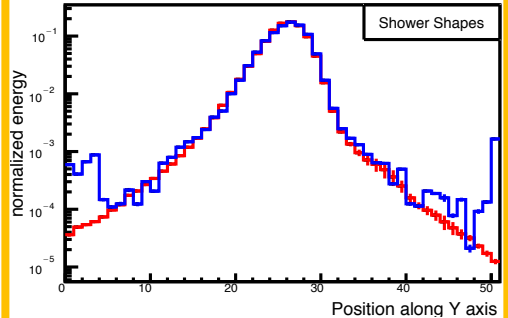
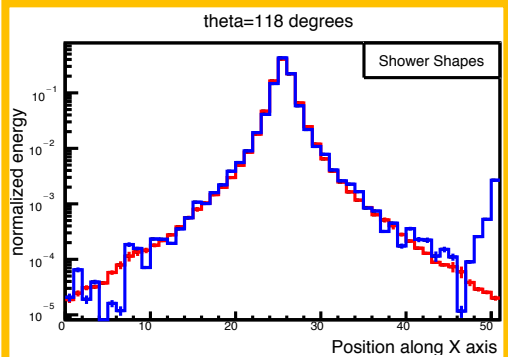
60° incident angle



Ortogonal



120° incident angle



# Angles

