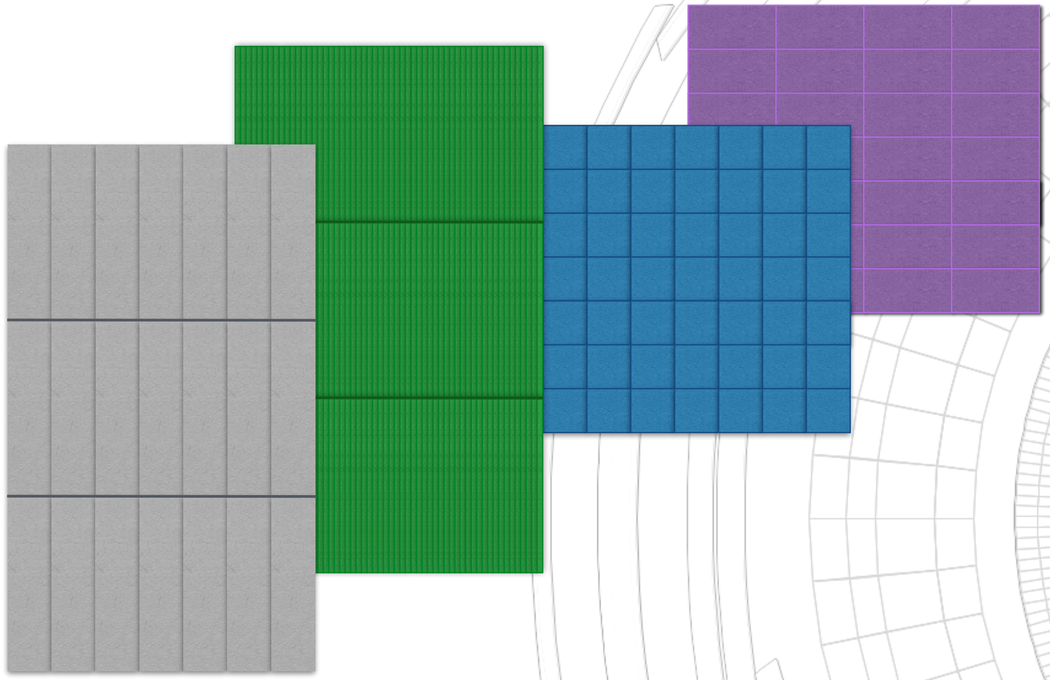
Machine Learning for Fast Simulation at LHC

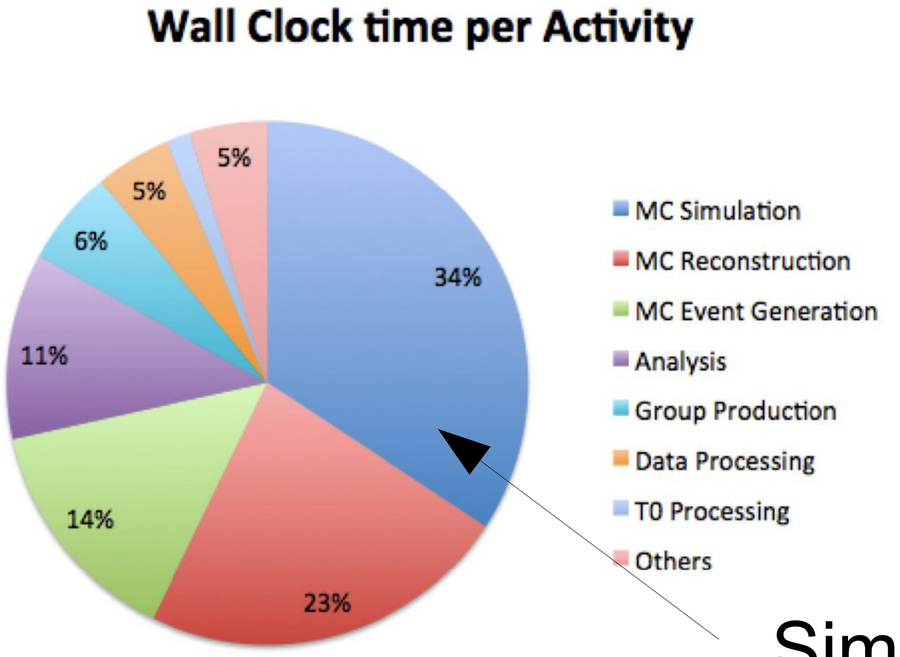
Aishik Ghosh

IJCLab Irène Joliot-Curie Laboratoire de Physique des 2 Infinis

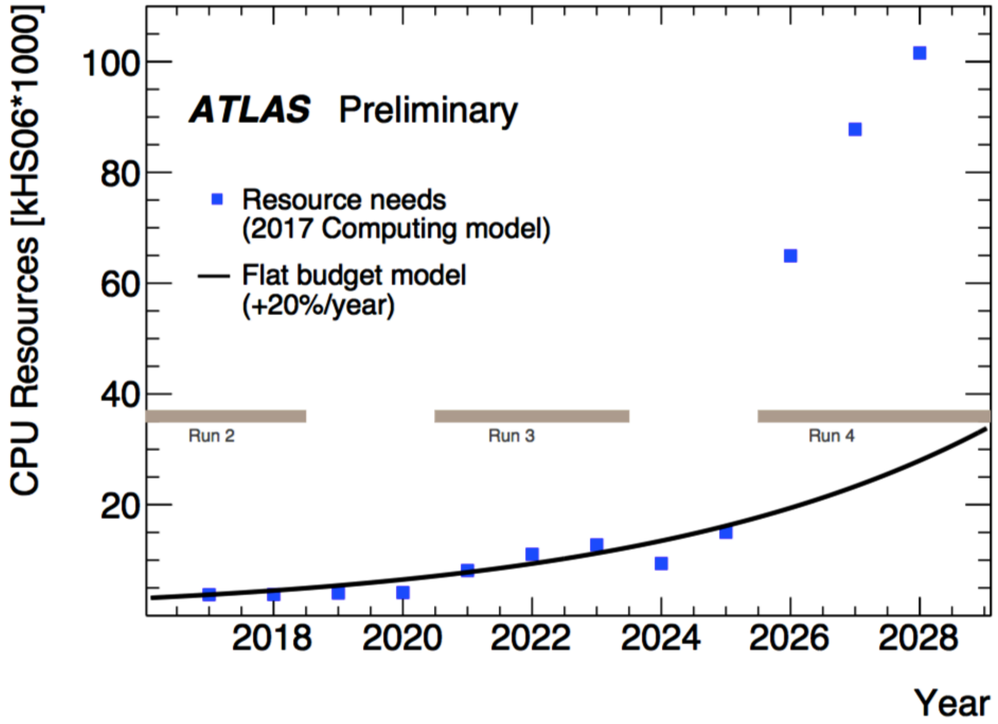
LHCP Conference
25 May 2020



Motivation for Fast Simulation



Simulation!



ATLAS 2016 numbers

Fast Simulation Strategies

Detector simulation in CMS

CMS FullSim

- detailed geometry
- particles tracked in small steps
- detailed material interaction model (mostly Geant4)
- detailed emulation of detector electronics and trigger
- standard event reconstruction

-O(100s) per ttbar event

CMS FastSim

- simplified geometry
- infinitely thin material layers
- simple analytical material interaction models
- detailed emulation of detector electronics and trigger, with exceptions
- standard event reconstruction, with exceptions

-O(5s) per ttbar event

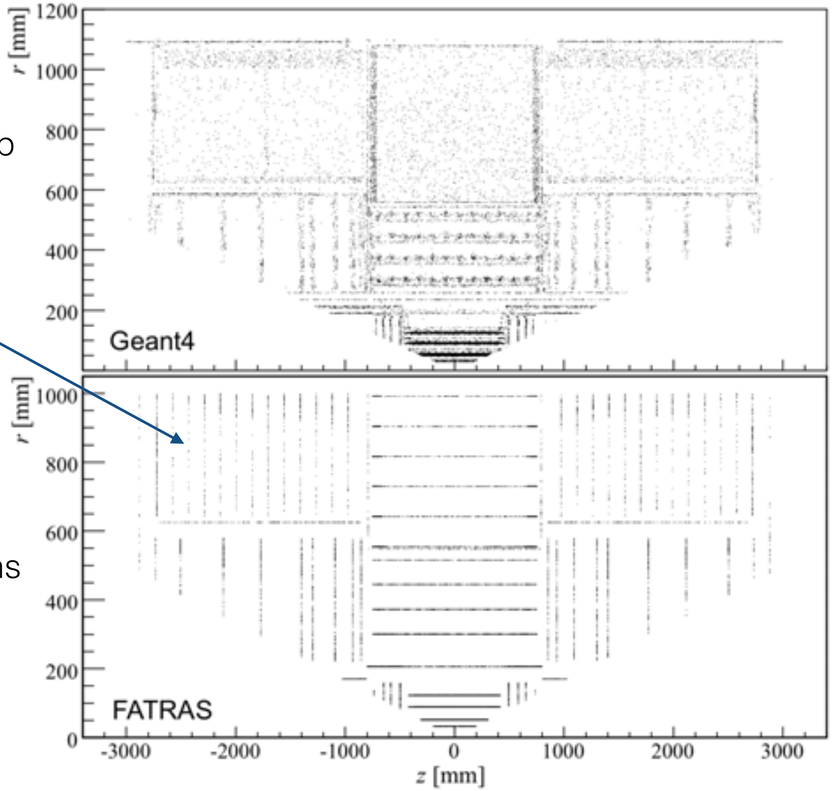
Delphes

- (almost) simple 4-vector smearing

-O(.01s) per ttbar event

ATLAS/Other experiments similarly speed up with:

- Simplified geometry
- Parameteised calorimeter response
- Approximations



CMS fastsim details

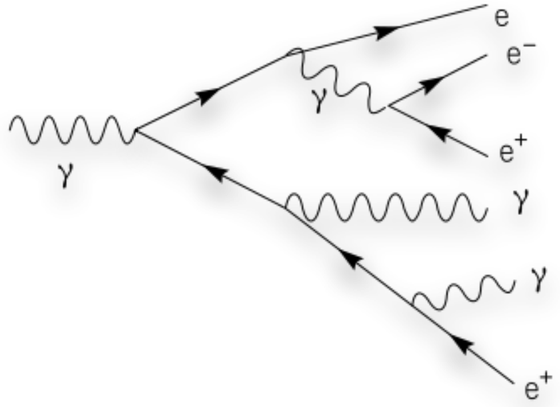
ATLAS fastsim chain details

Calorimeter Simulations

- Simulate how particles interact with matter from first principles
- Follow time evolution, even if **only final image recorded**
- Exponential cascade of particle showering ⇒ exponential time to simulate
- Dominant part of simulation time

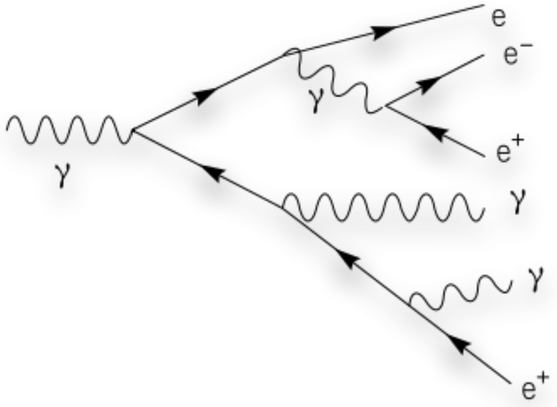
Calorimeter Simulations

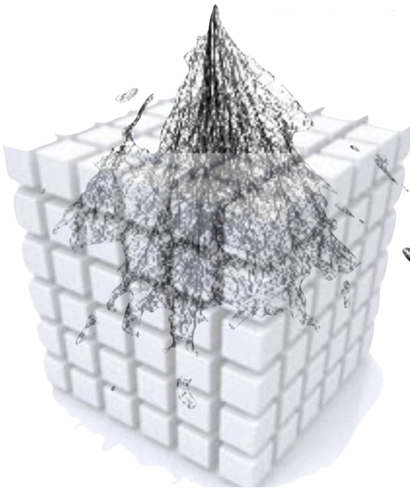
- Simulate how particles interact with matter from first principles
- Follow time evolution, even if **only final image recorded**
- Exponential cascade of particle showering ⇒ exponential time to simulate
- Dominant part of simulation time



Calorimeter Simulations

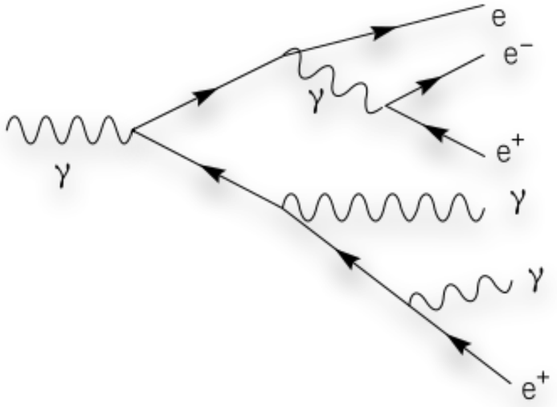
- Simulate how particles interact with matter from first principles
- Follow time evolution, even if **only final image recorded**
- Exponential cascade of particle showering ⇒ exponential time to simulate
- Dominant part of simulation time

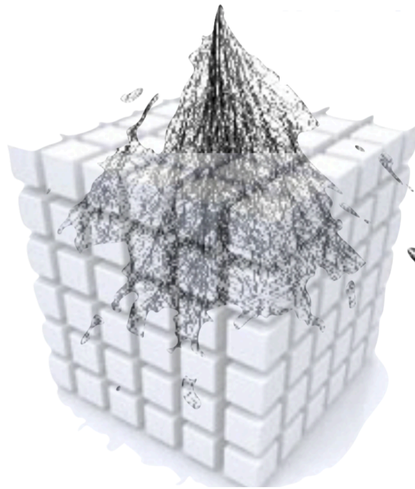


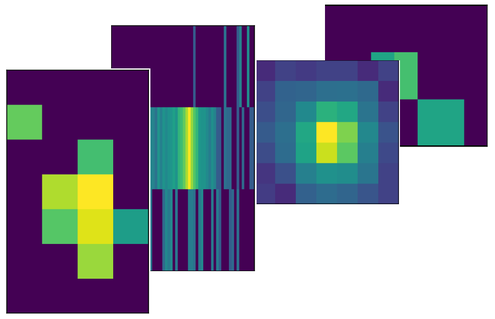


Calorimeter Simulations

- Simulate how particles interact with matter from first principles
- Follow time evolution, even if **only final image recorded**
- Exponential cascade of particle showering ⇒ exponential time to simulate
- Dominant part of simulation time

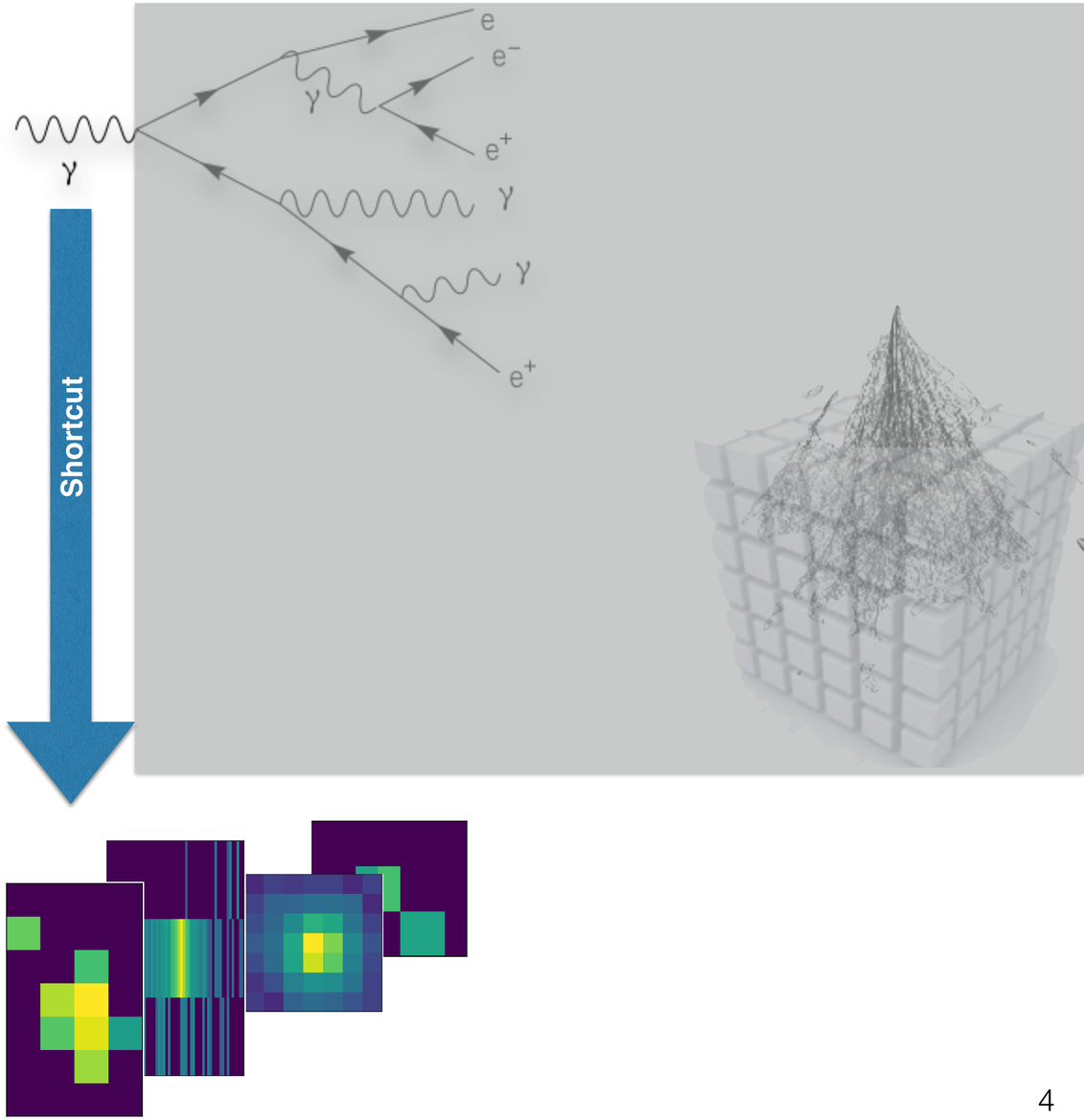






Calorimeter Simulations

- Simulate how particles interact with matter from first principles
- Follow time evolution, even if **only final image recorded**
- Exponential cascade of particle showering ⇒ exponential time to simulate
- Dominant part of simulation time

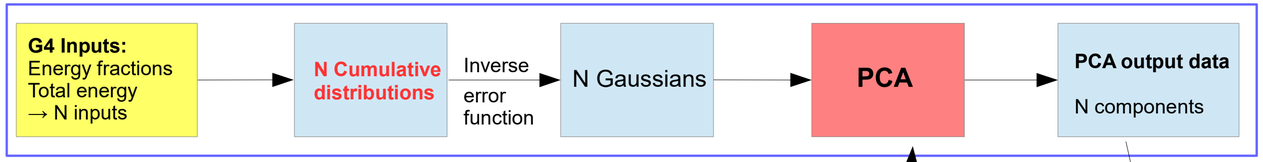


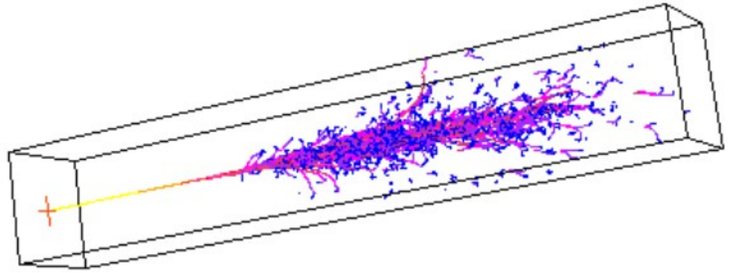
Thanks Sean Gasiorowski!

ATLAS FastCaloSim

Longitudinal

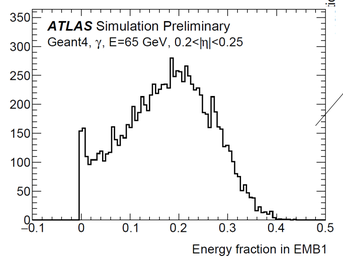
1st PCA chain:



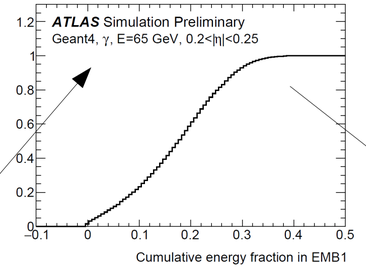


- Start with Geant4 simulations
- Memory footprint: Efforts to efficiently store all parameterisation data

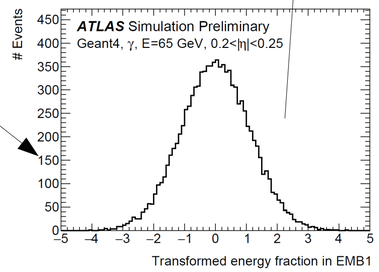
Example:
Photons 65 GeV



Cumulative

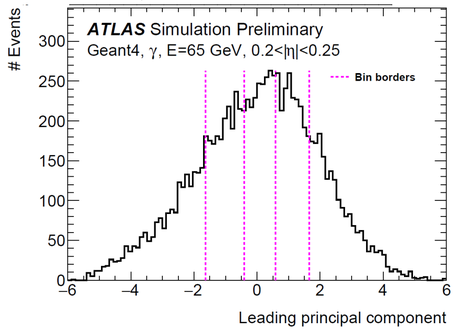


TPrincipal
(reference)



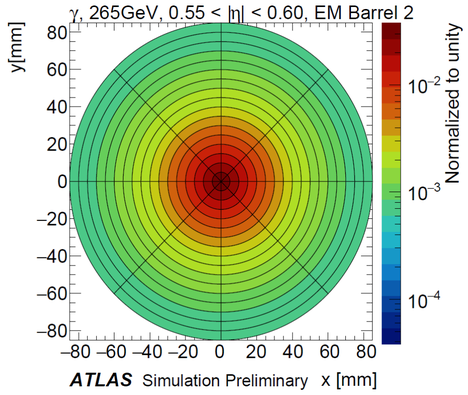
Gaussian

N outputs



In each “PCA bin” a second PCA rotation is performed to further decorrelate

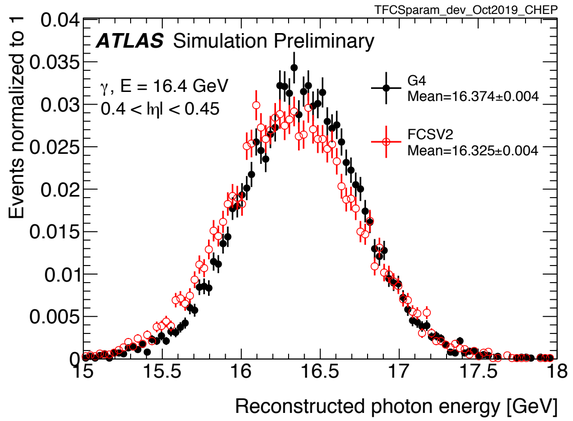
Lateral

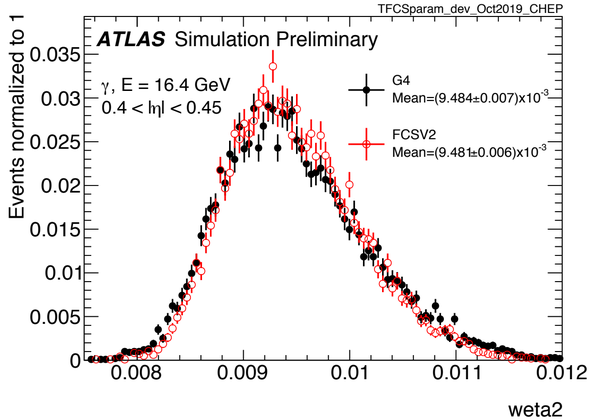


See details

+ Energy Interpolation mechanism

Results:





Deep Generative Models for Fast Simulation

Aim:

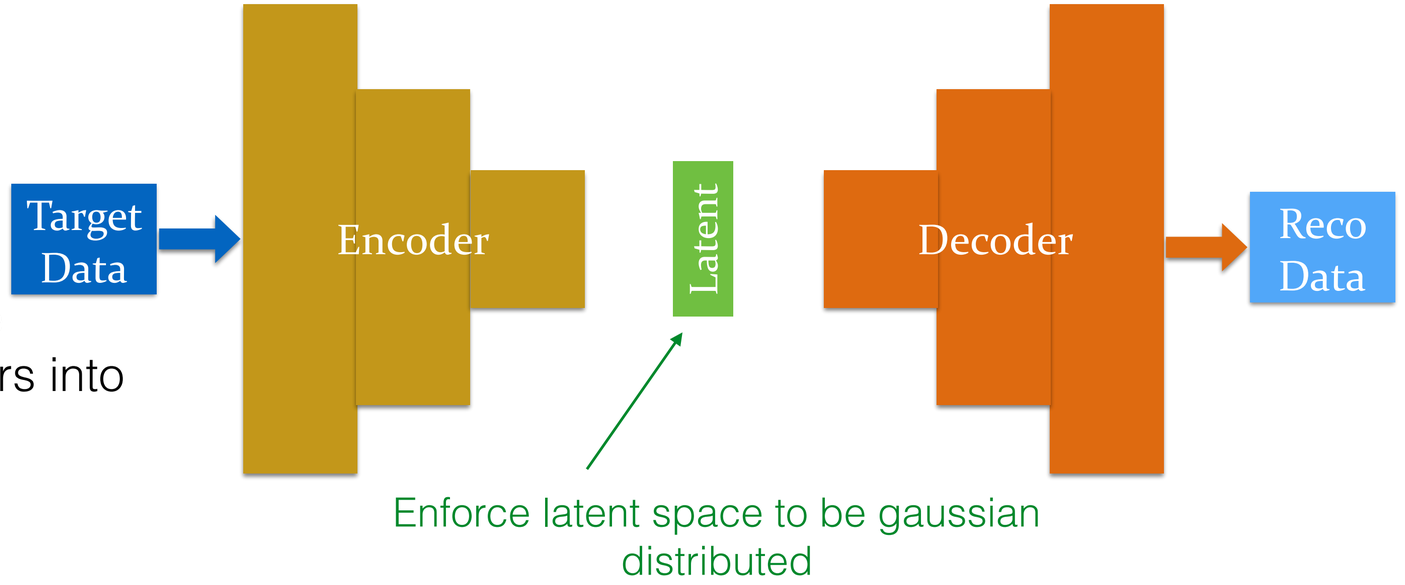
- Simulate showers 100-1000x *faster* than Geant4
- *Less human time* intensive, *higher accuracy* than current fast simulation methods
- Use *less memory* than current fast simulation methods
- Take advantage of new technology: DL, GPUs, HPCs

How?

Prominent Algorithms

Variational AutoEncoder (VAE):

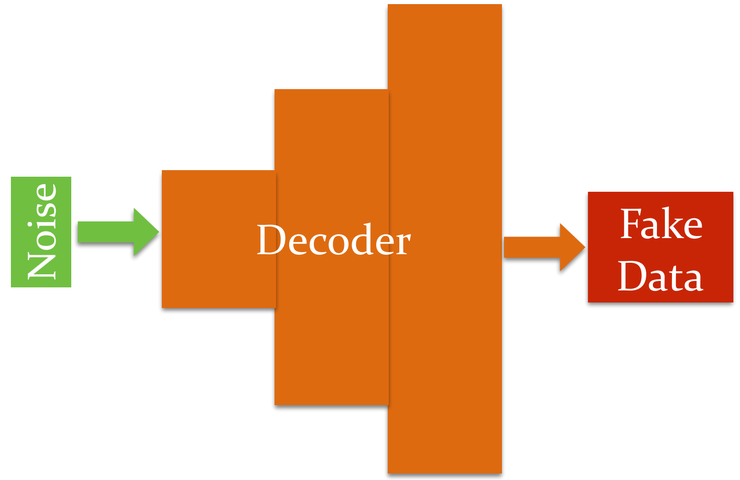
- Train encoder and decoder neural networks
- Small (often Gaussuan) encoded latent space
- Once trained, inject Gaussian random numbers into decoder to get new images



Prominent Algorithms

Variational AutoEncoder (VAE):

- Train encoder and decoder neural networks
- Small (often Gaussuan) encoded latent space
- Once trained, inject Gaussian random numbers into decoder to get new images

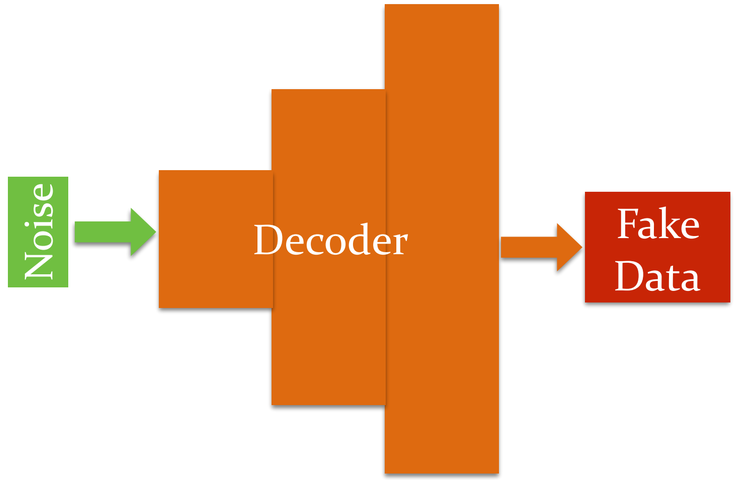


Enforce latent space to be gaussian distributed

Prominent Algorithms

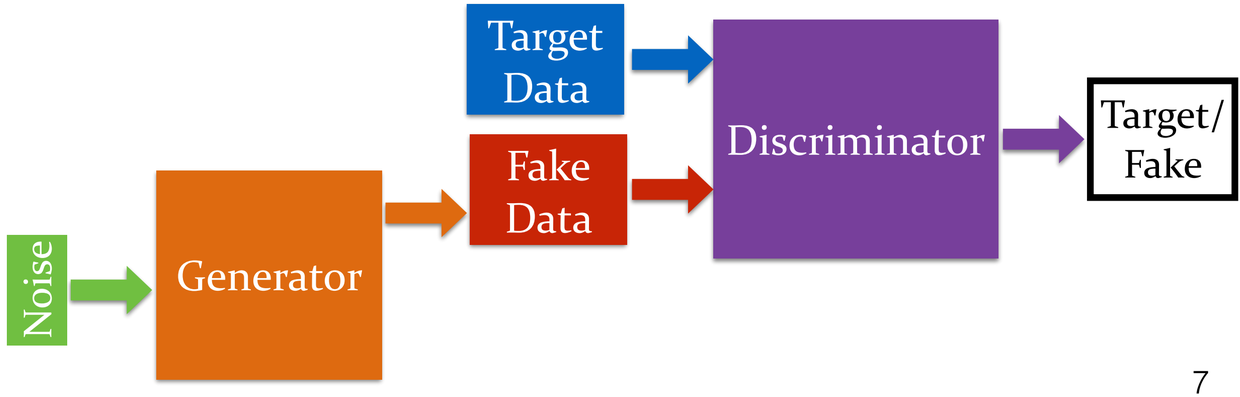
Variational AutoEncoder (VAE):

- Train encoder and decoder neural networks
- Small (often Gaussuan) encoded latent space
- Once trained, inject Gaussian random numbers into decoder to get new images



Generative Adversarial Network (GAN):

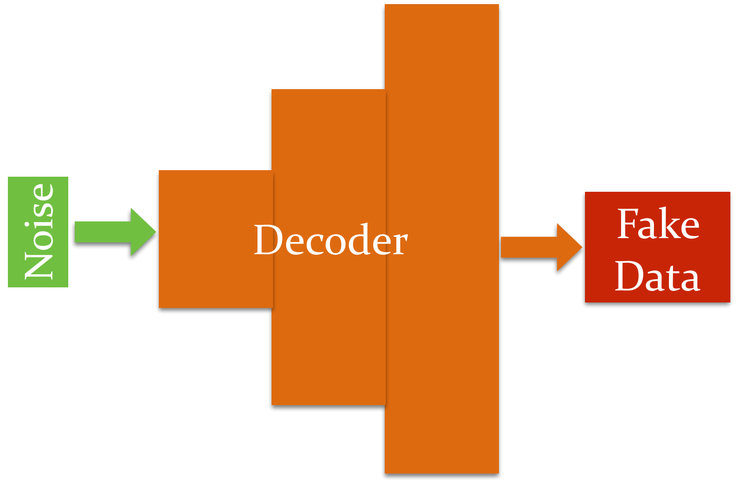
- Train a discriminative network to learn the difference between real and fake images
- Train a generative network to produce realistic fake images, to fool the discriminator (iterative)
- If converged, generator produces very realistic images



Prominent Algorithms

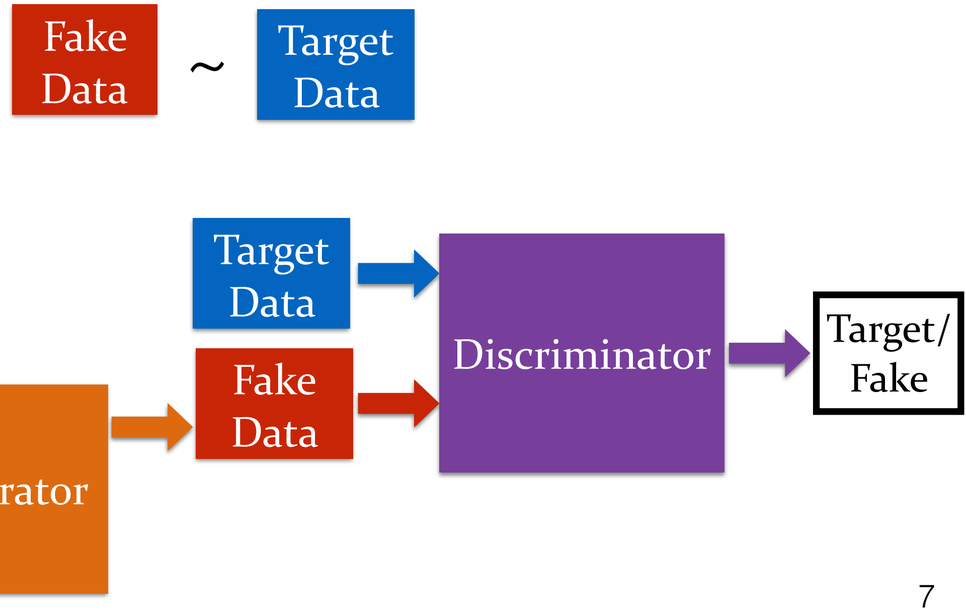
Variational AutoEncoder (VAE):

- Train encoder and decoder neural networks
- Small (often Gaussuan) encoded latent space
- Once trained, inject Gaussian random numbers into decoder to get new images



Generative Adversarial Network (GAN):

- Train a discriminative network to learn the difference between real and fake images
- Train a generative network to produce realistic fake images, to fool the discriminator (iterative)
- If converged, generator produces very realistic images



Research on Deep Generative Models

GAN research moving towards better quality images

GAN research moving towards better quality images

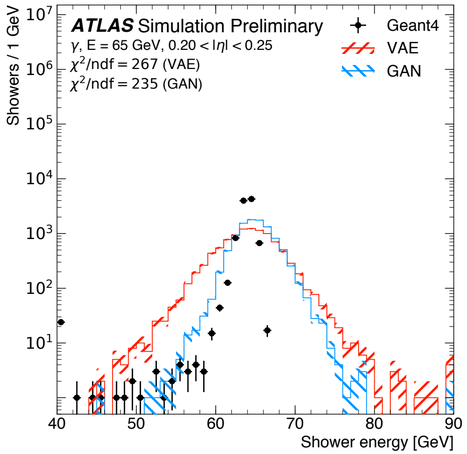
(BE)GAN seems to produce more attractive faces than in training dataset

We observe varied poses, expressions, genders, skin colors, light exposure, and facial hair. However we did not see glasses, we see few older people and there are more women than men. For comparison

Research on Deep Generative Models

GAN research moving towards better quality images

But probability densities are another thing



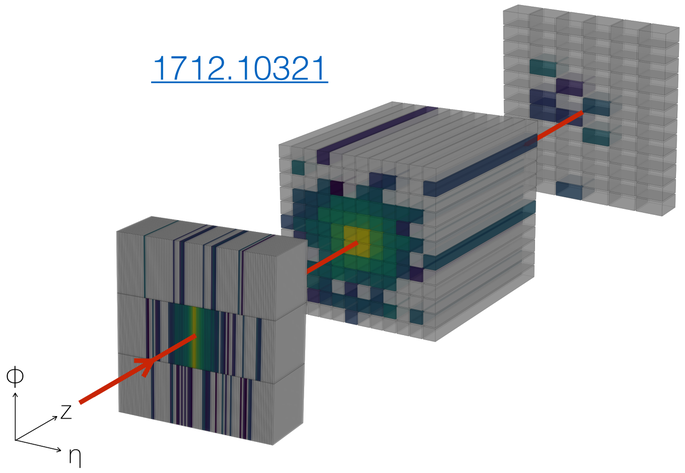
(BE)GAN seems to produce more attractive faces than in training dataset

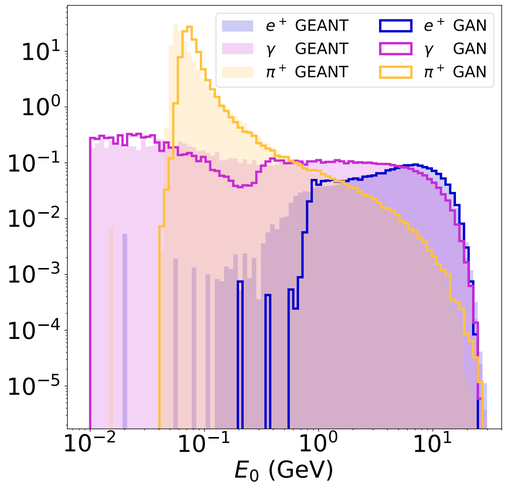
We observe varied poses, expressions, genders, skin colors, light exposure, and facial hair. However we did not see glasses, we see few older people and there are more women than men. For comparison

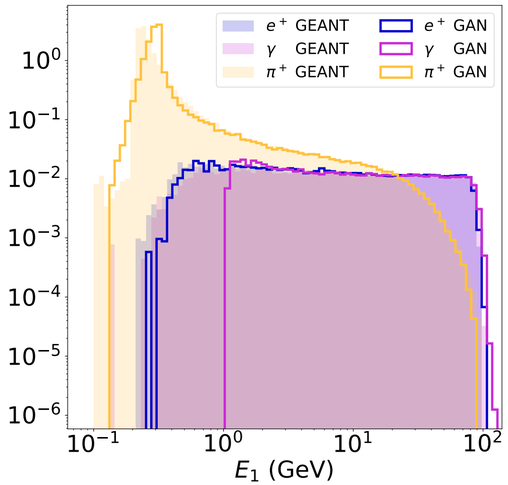
CaloGAN

CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks

1712.10321







- CaloGAN showed that it is possible to simulate EM showers for a detector like ATLAS using GANs
- Faster “Surrogate Model” trained on Geant4 generated samples

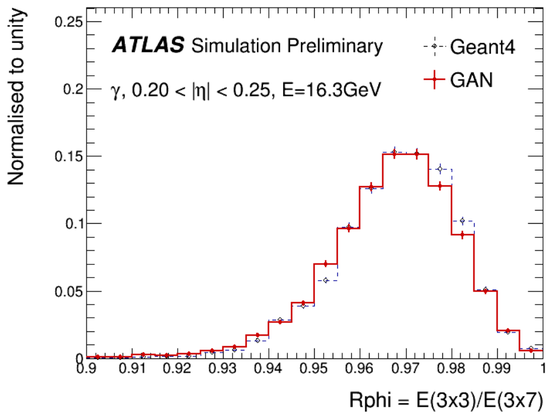
ATLAS Calorimeter Implementation

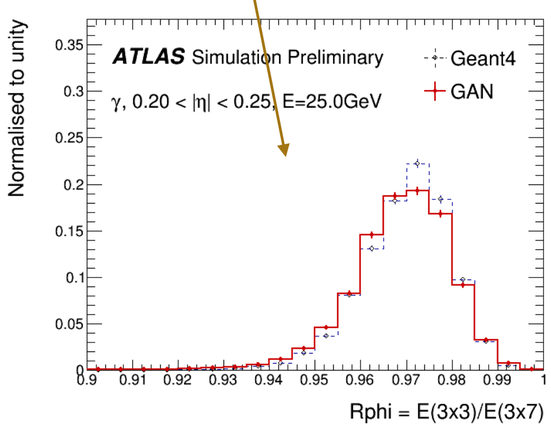
- Trained on calorimeter cells
- Validated in ATLAS software, high level variables
- Interpolates to untrained points
- Happy with speed (orders of magnitude faster than Geant4)
- Tiny memory footprint

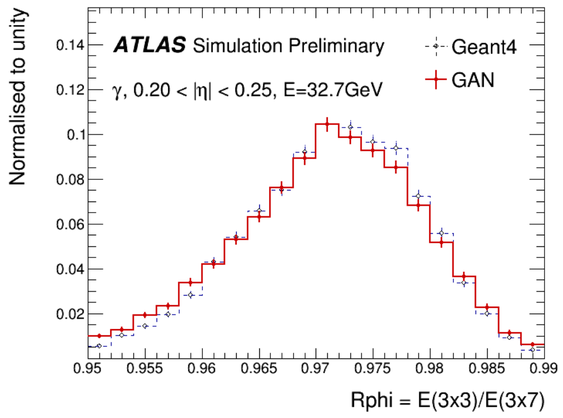
- Next: Expand to entire detector by training on ~~cells~~ voxels

GAN never trained at 25 GeV!

GAN



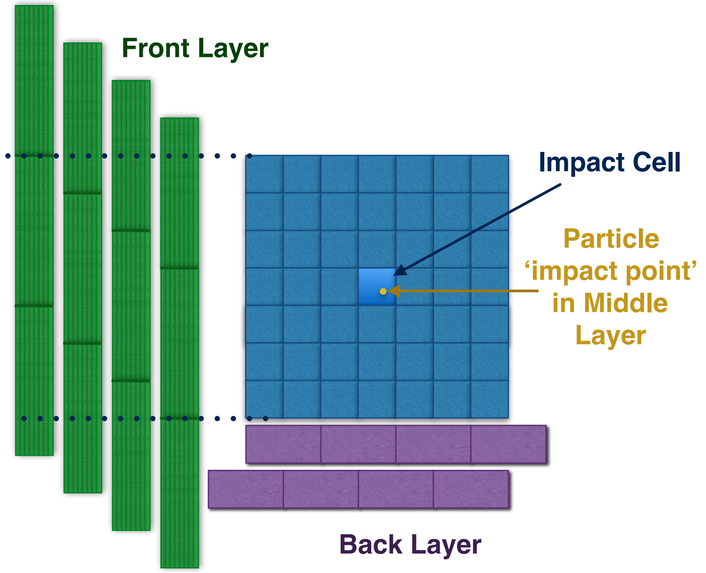




16 GeV

25 GeV

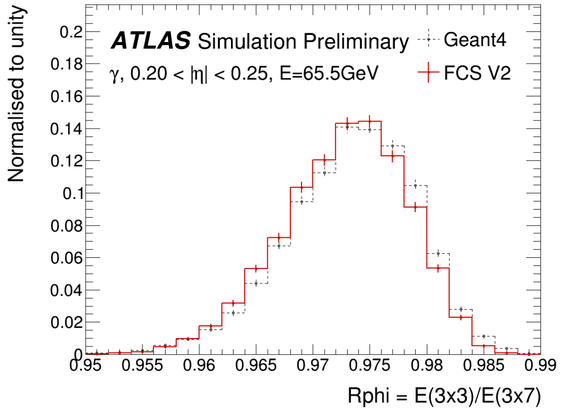
32 GeV



Details here here
VAE updates here

ATLAS Calorimeter Implementation

FCS (Baseline)



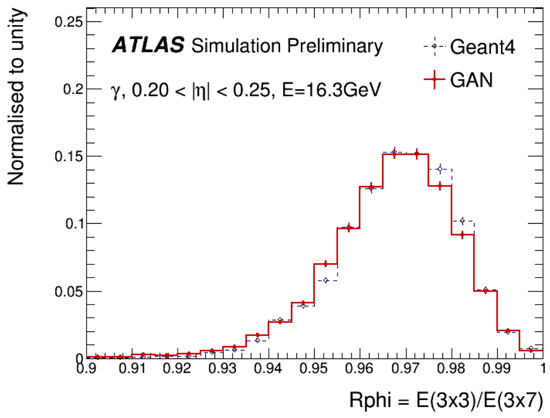
65 GeV

- Trained on calorimeter cells
- Validated in ATLAS software, high level variables
- Interpolates to untrained points
- Happy with speed (orders of magnitude faster than Geant4)
- Tiny memory footprint

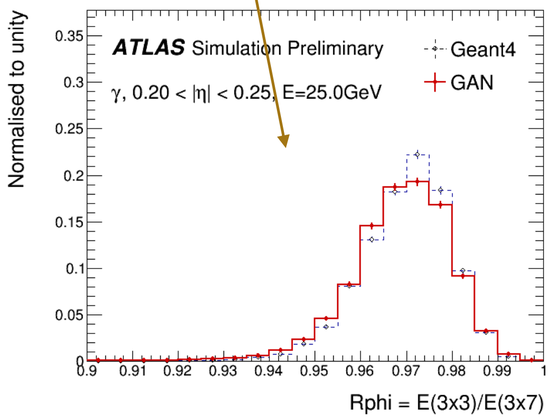
- Next: Expand to entire detector by training on ~~cells~~ voxels

GAN never trained at 25 GeV!

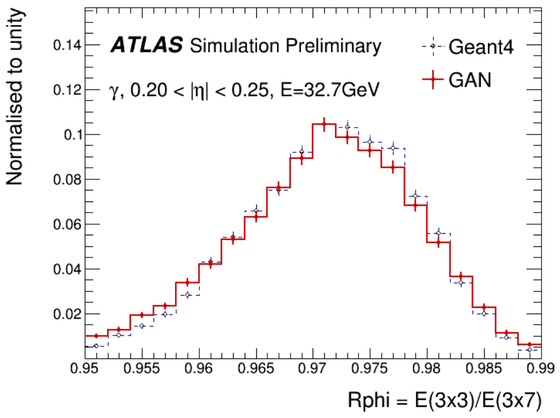
GAN



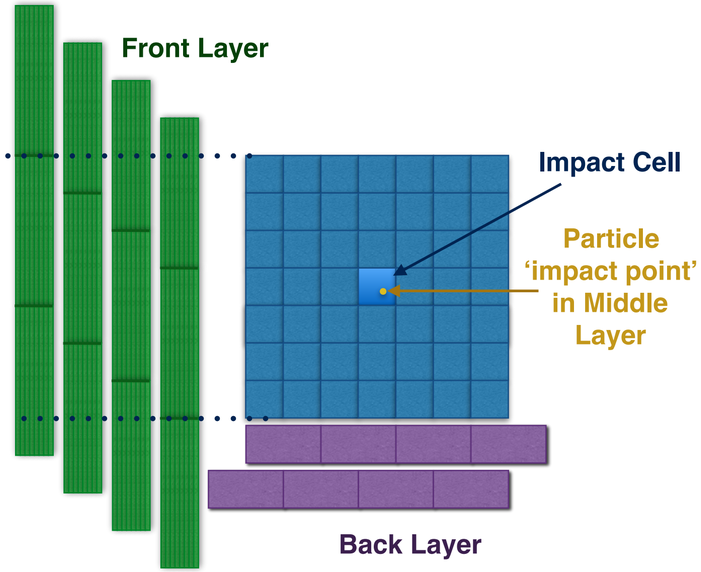
16 GeV



25 GeV



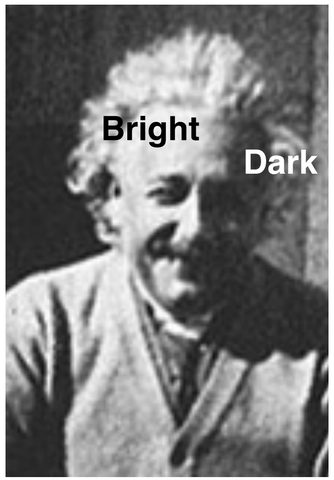
32 GeV

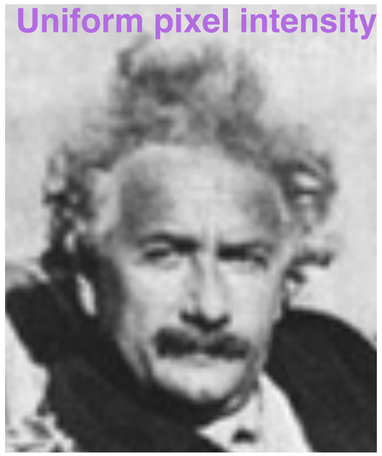


Details here here
VAE updates here

WGAN disadvantage

Details here here
VAE updates here



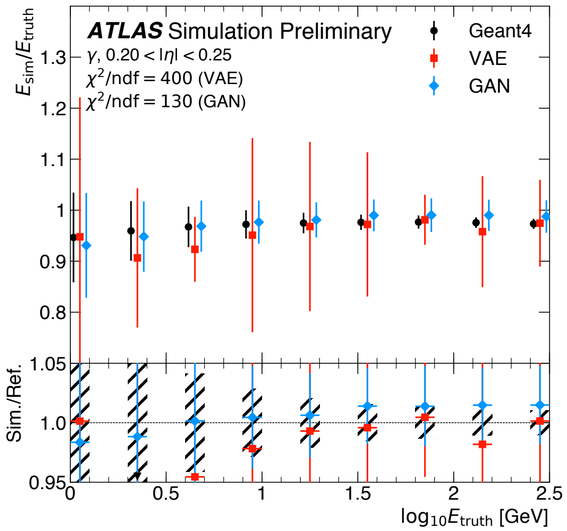


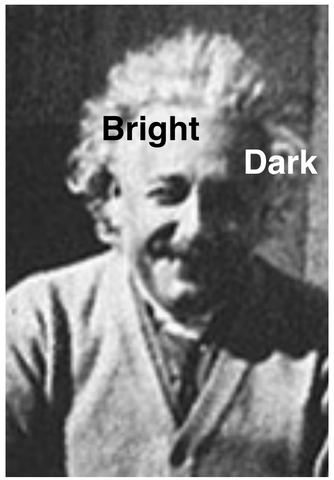
Raw pixel intensities not important for computer vision, very important for calorimetry

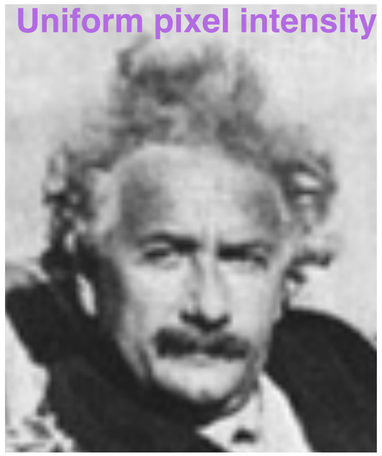
From Ben Nachman's talk, http://mentalfloss.com

WGAN disadvantage

Details here here
VAE updates here







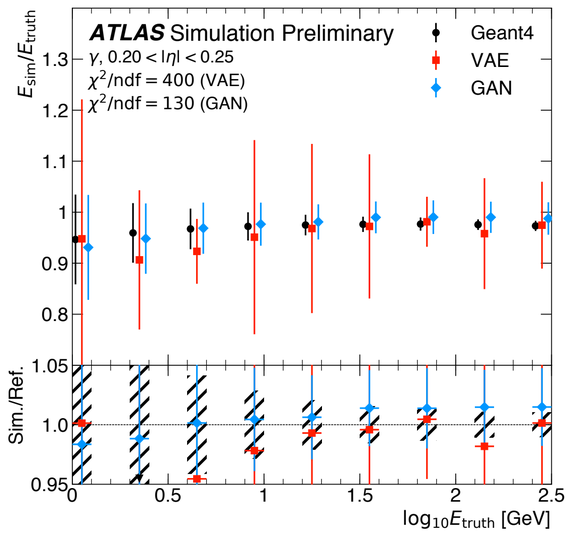
Raw pixel intensities not important for computer vision, very important for calorimetry

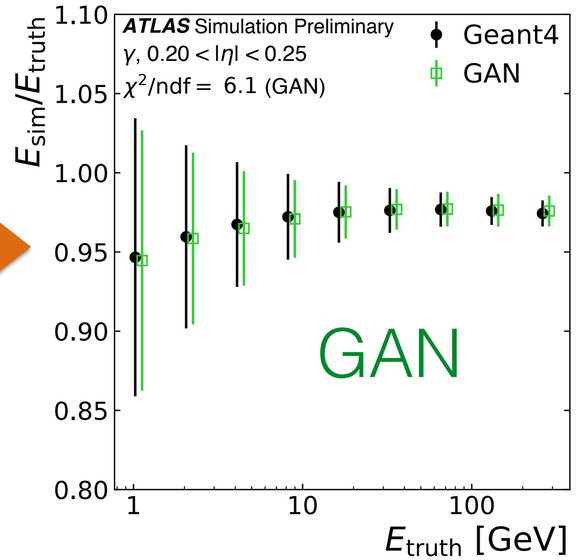
From Ben Nachman's talk, http://mentalfloss.com

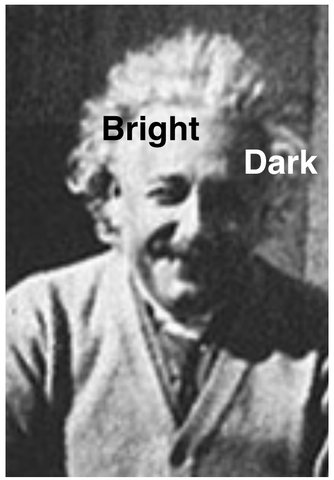
WGAN disadvantage

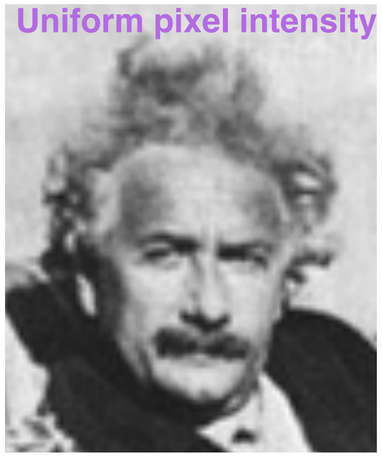
Details here here
VAE updates here

- Wasserstein GAN very popular flavour of GANs used in HEP (Applies Gradient Penalty on Discriminator to allow more meaningful feedback to generator)
- WGAN has trouble with energy/mass distributions
  - ⇒ ATLAS solution: additional "Energy Critic Network"
  - ⇒ Other solution: MMD loss (see Anja Butter's talk)
- ATLAS VAE solve by training on energy ratios, HPO









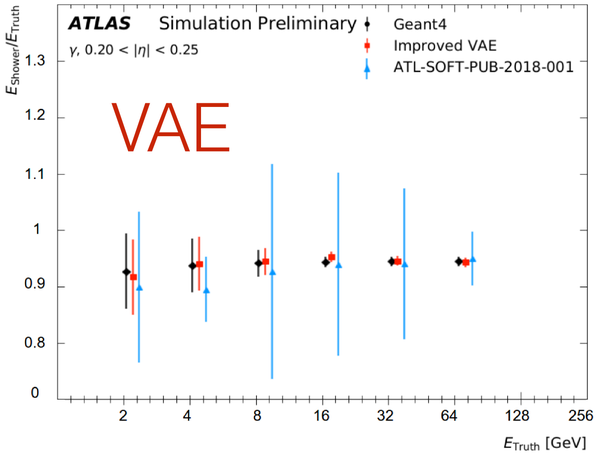
Raw pixel intensities not important for computer vision, very important for calorimetry

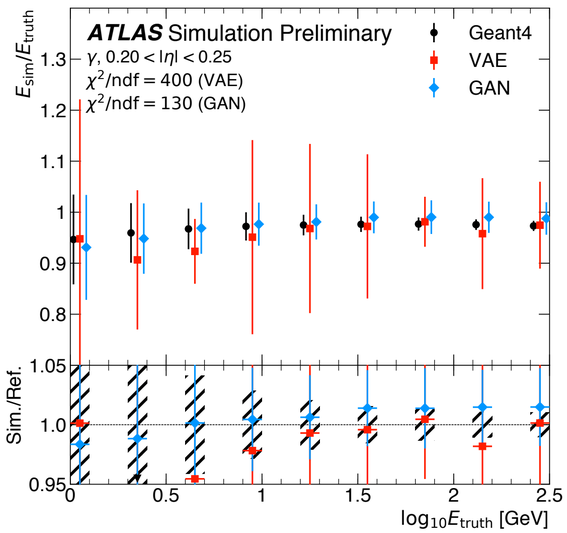
From Ben Nachman's talk, http://mentalfloss.com

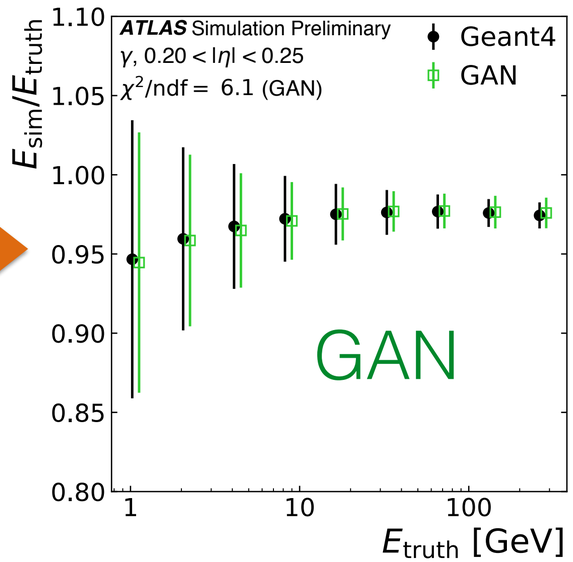
WGAN disadvantage

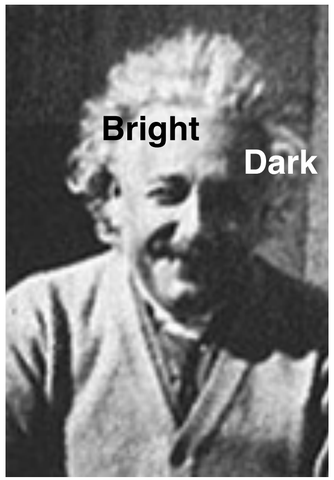
- Wasserstein GAN very popular flavour of GANs used in HEP (Applies Gradient Penalty on Discriminator to allow more meaningful feedback to generator)
- WGAN has trouble with energy/mass distributions
  - ⇒ ATLAS solution: additional “Energy Critic Network”
  - ⇒ Other solution: MMD loss (see Anja Butter’s talk)
- ATLAS VAE solve by training on energy ratios, HPO

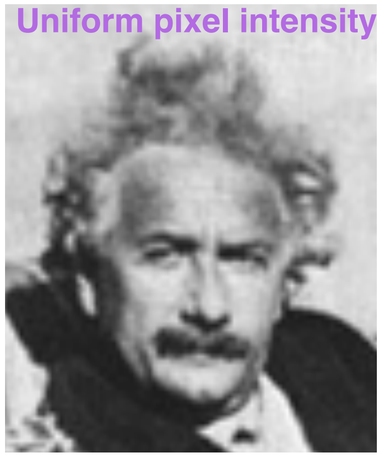
Details here here
VAE updates here









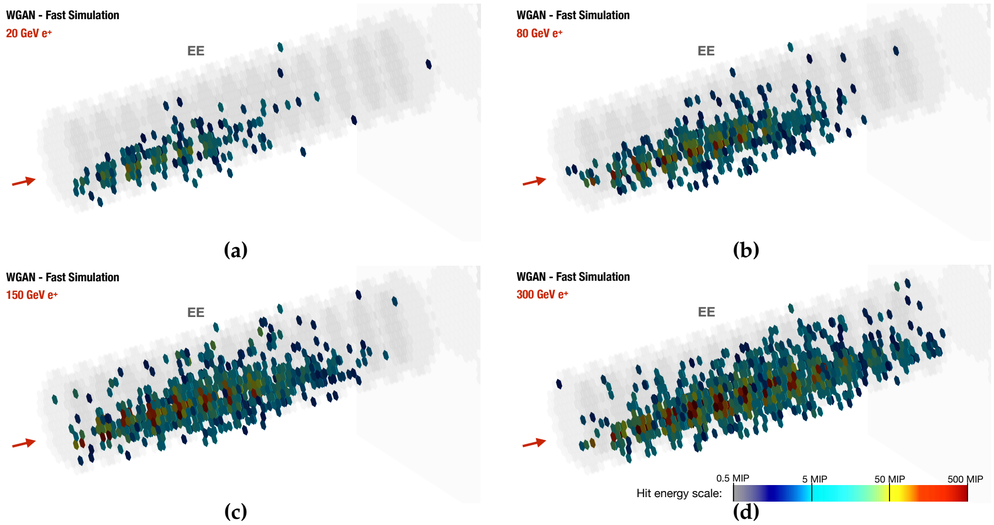


Raw pixel intensities not important for computer vision, very important for calorimetry

From Ben Nachman’s talk, http://mentalfloss.com

Thanks Thorben Quast!

CMS Prototype High Granularity Calorimeter



arxiv:1807.01954

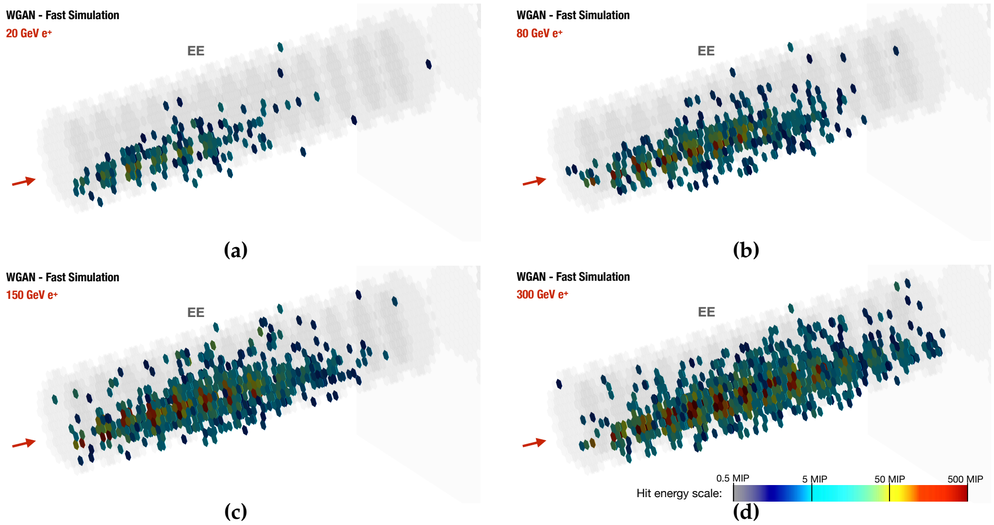
- Trained on Geant4 simulation
- Focus on positron induced showers
- Reproduces distributions well

- Trouble with hit energy spectrum (common problem of WGANs)
- Move to test beam data

Thanks Thorben Quast!

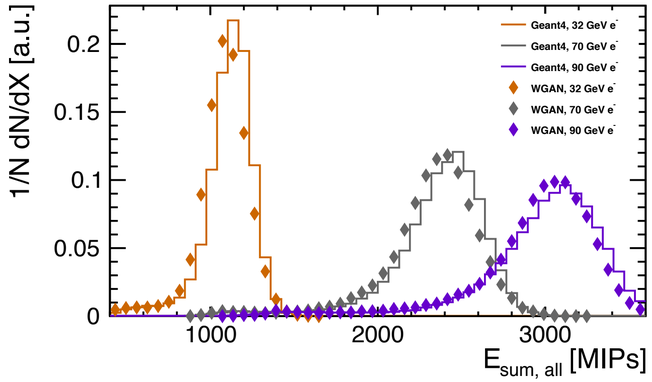
CMS Prototype High Granularity Calorimeter

arxiv:1807.01954

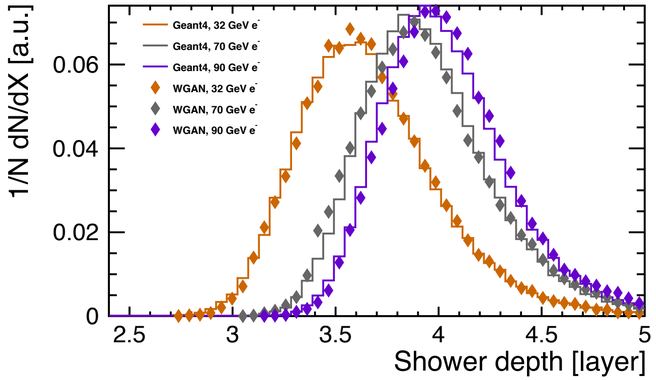


- Trained on Geant4 simulation
- Focus on positron induced showers
- Reproduces distributions well

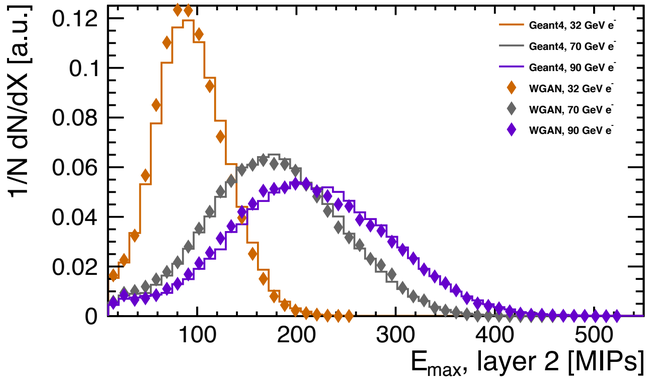
- Trouble with hit energy spectrum (common problem of WGANs)
- Move to test beam data



(a)



(b)

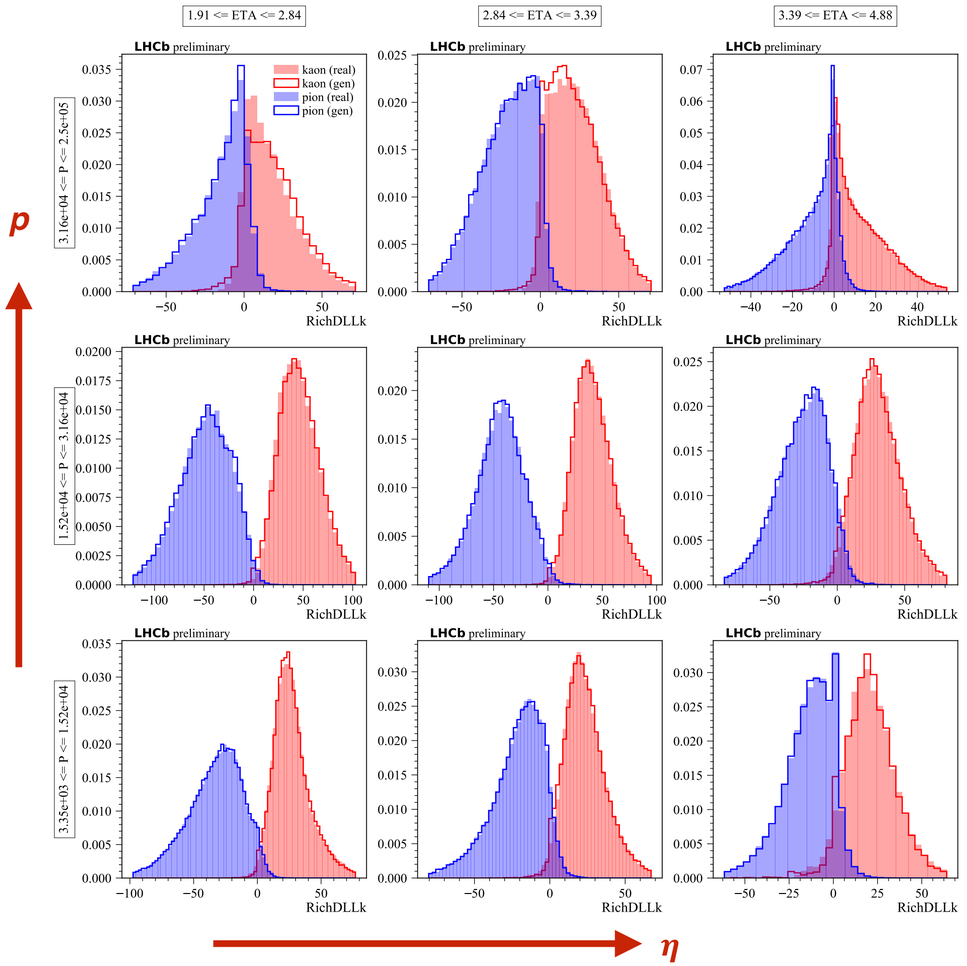


(c)

Thanks Fedor Ratnikov!

Beyond Geant: Learn directly from data (LHCb)

arxiv:1905.11825

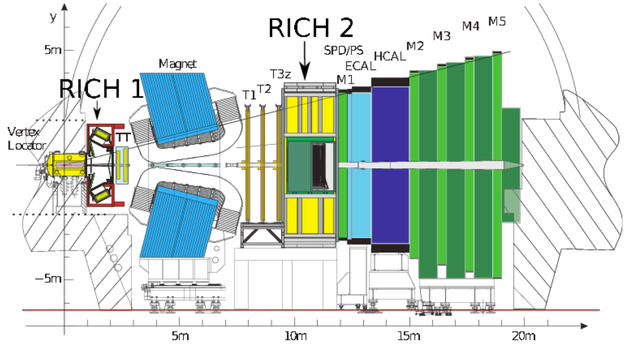


kaon (real)

kaon (gen)

pion (real)

pion (gen)

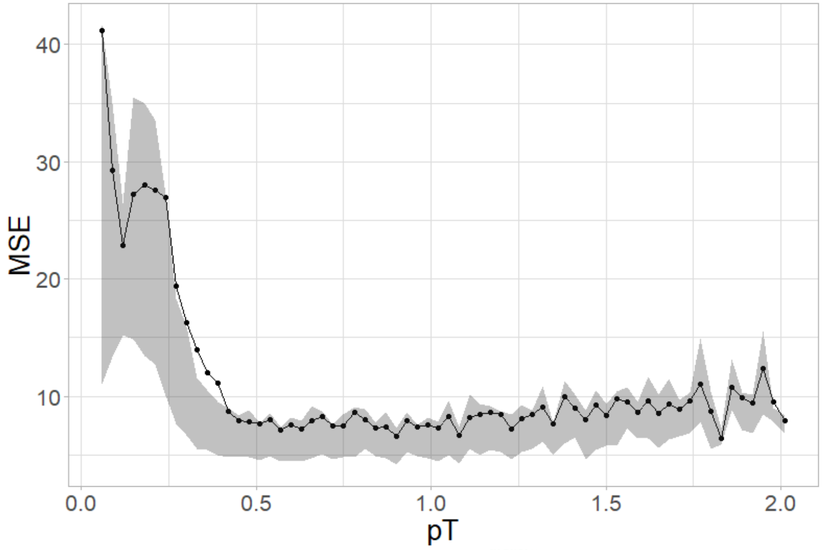


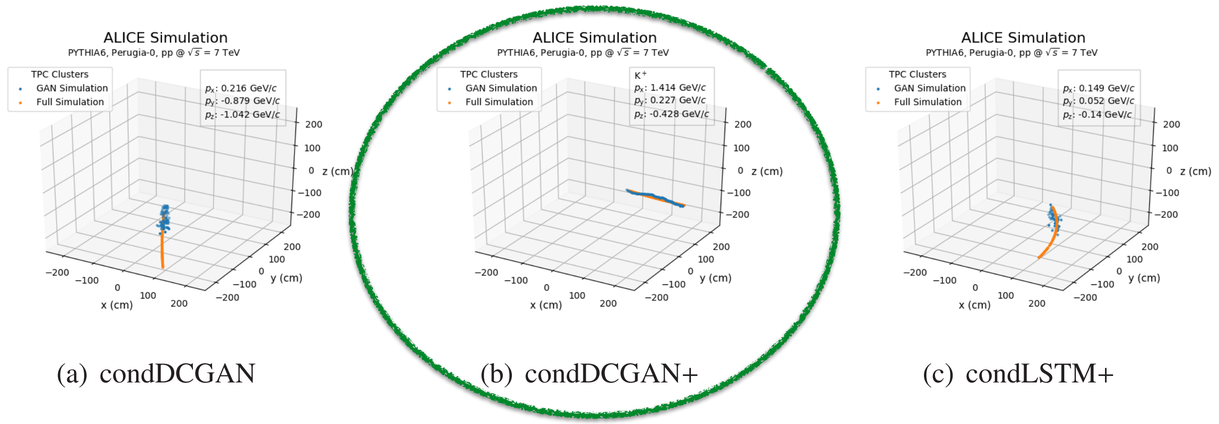
- **Trained on calibrated data samples!**
- RICH is used for particle ID only
  - 5 probabilities for different ID hypotheses
- 5 outputs RichDLL{k,p,μ,e,bt}
- Conditioned on (p, η, # of tracks)

- Discrepancies in particle ID efficiencies propagated as systematic uncertainties
- Allows to avoid expensive RICH simulation with GEANT

Thanks Kamil Rafal Deja!

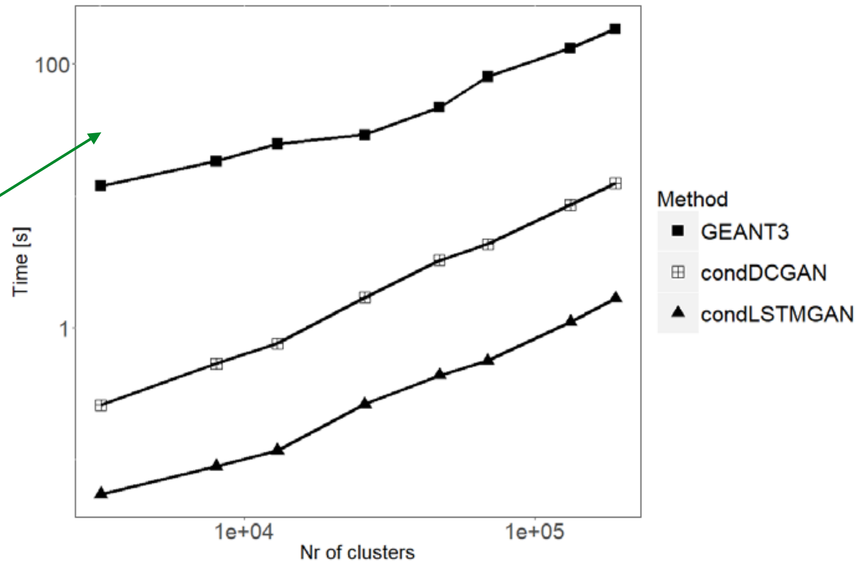
ALICE: Time Projection Chamber





(a) condDCGAN (b) condDCGAN+ (c) condLSTM+

CPU times



- Trained on TPC clusters
- Validated using particle track properties
- Uses GAN and VAE training losses simultaneously

- 25x speedup on CPU
- GAN does better on high Pt, straight tracks

Figure 2. Exemplar results generated by different models **(a)** conditional DCGAN without additional loss, **(b)** conditional DCGAN and **(c)** conditional LSTM GAN, with additional loss

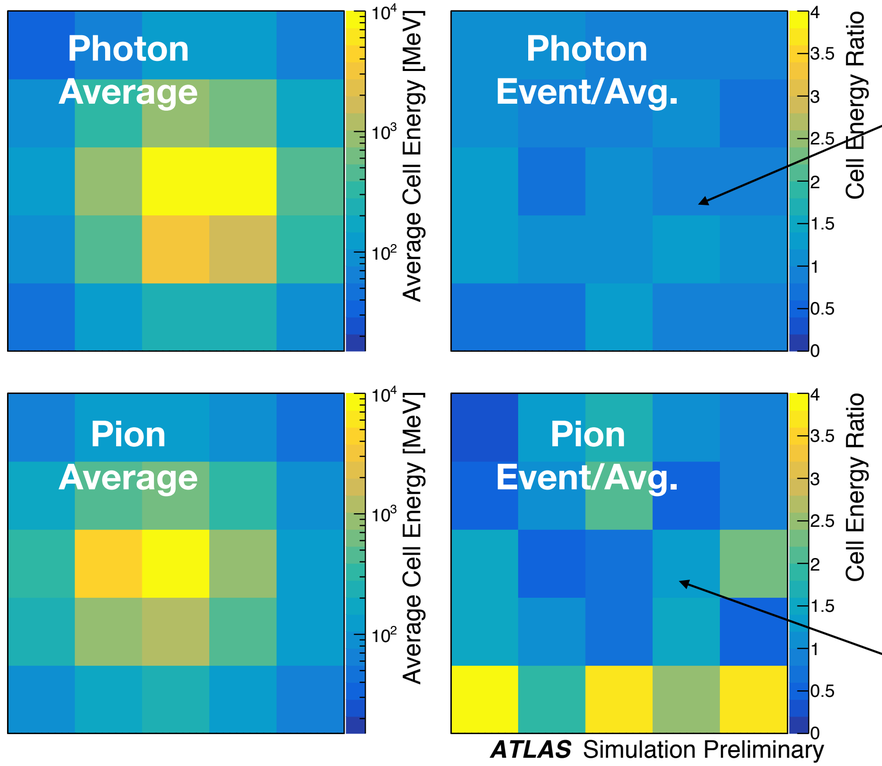
See details

Thanks Sean Gasiorowski, Dalila Salamani!

Hybrid: Traditional + Generative Fast Sim (ATLAS)

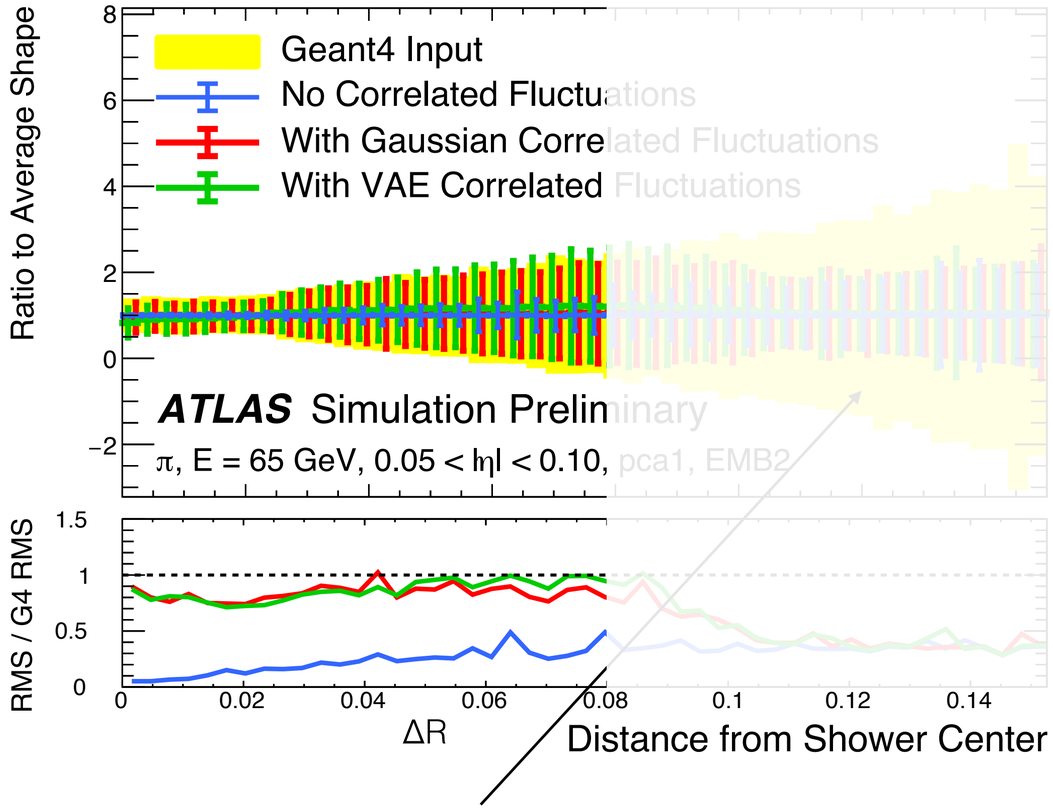
See details

- Simulate showers using traditional parameterised algorithm
- Add fluctuations with VAE



Photon event is very similar to average

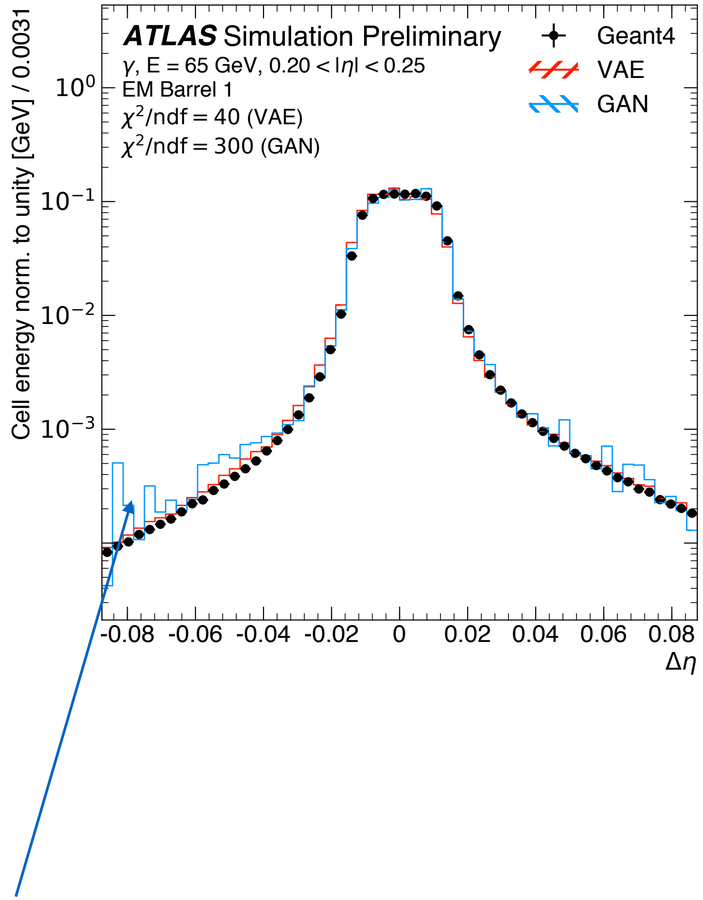
Pion event is very different! Current shape simulation not sufficient



Differences beyond 0.08 can be covered by tuning the size of fluctuations in the current (uncorrelated) model

Systematics / Considerations

- Parameterisation based on Geant4 **cannot beat Geant4** unless
  - Inject first principle assumptions
  - Train on / transfer learn specific features directly from data
  - Interpolate between training points (still indirectly limited by training set)
- **Statistical fluctuations** of training set → **systematic fluctuations** of GAN (Overtraining)
- Smart compression: Trained on single particle showers but actual use in simulation of many kinds of events / processes
- **Don't use for rare detector-induced fakes**
- Cannot overcome systematic uncertainties with fast parameterised sim



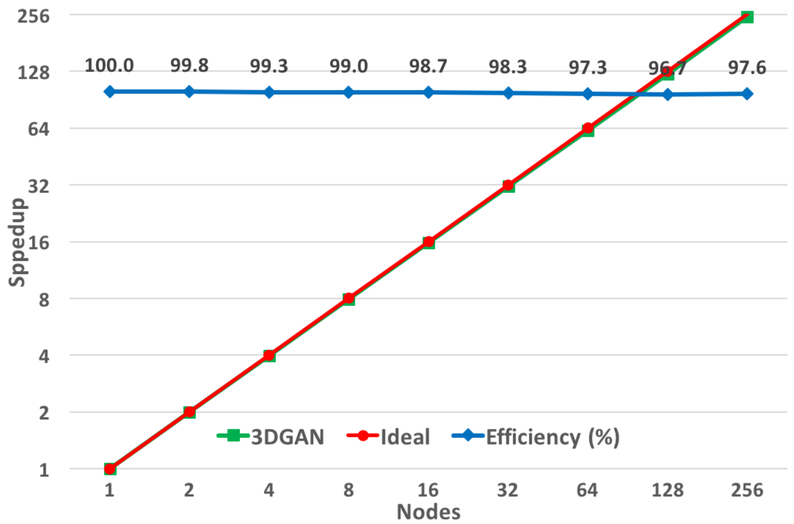
Interesting discussion in [2002.06307]

Sign of overtraining when trained on 4% of dataset

Future

GANs can take days to train, 3DGAN (CERN OpenLab) show impressive scaling with GPUs

Weak scaling on Intel Endeavour cluster

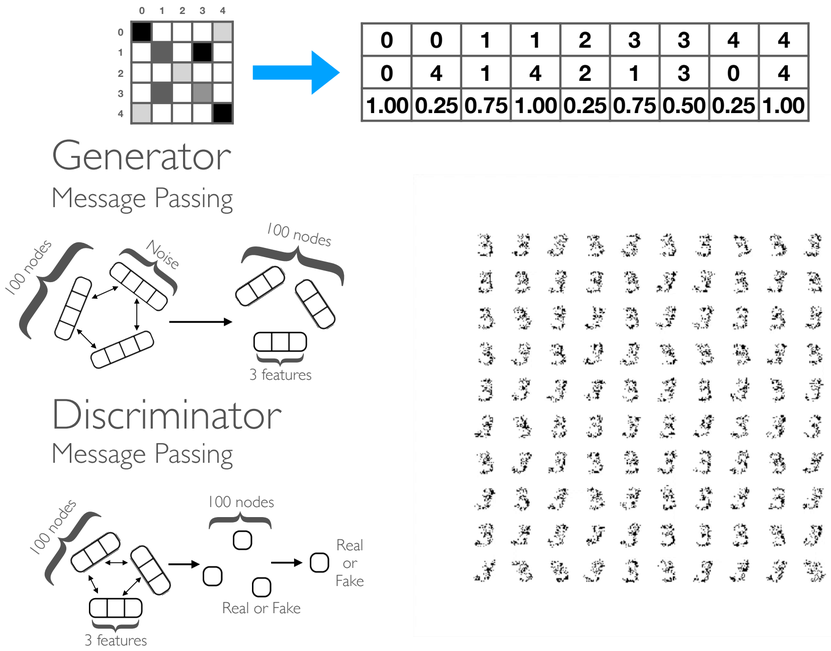


- GAN Simulating CLIC calo
- **1.5 Min**/Epoch on 256 nodes
- Time to Train to Accuracy: **3 hours**

Geant4 team looking into generic fast sim approaches using generative models

Graph based Generative Models for sparse images

see details



Generative Models with Quantum ML (2005.08582)

Beyond Detector Simulations

- Efficient Pile up simulation
  1912.02748

- ML-assisted Phase Space sampling for MC
  (see Enrico Bothmann's talk!)

15:36 **Monte Carlo and event generators from a theory prospective**

Speaker: Enrico Bothmann (University of Göttingen)

- Full event Simulation
  (more in Anja Butter's talk!)

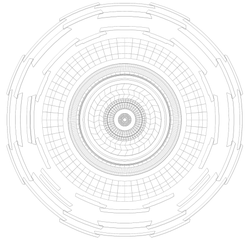
15:18 **Generative models in Event Simulation**

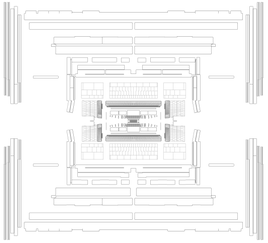
Speaker: Anja Butter

New ideas keep coming in!

Conclusion

- Dire need for improved fast simulation approaches to cope with growing CPU consumption of LHC experiments
- Traditional methods of fast simulation maintained by all experiments: parameterised response, simplified geometry etc
- Deep generative models of interest for : speed, accuracy, reduce human time investment, memory footprint
  - **Detector specific losses, architectures**
  - Hybrid approaches
  - Train on Geant4 or directly on data
- Future: Expect more generative models in each LHC experiment, exciting new approaches and possibly general purpose architectures





Backup

Zoom link for one-on-one chat (time 16:35-17:35):
-Removed-

RICH-GAN for LHCb

RICH detector is hard and expensive to simulate

RICH is used for particle ID only

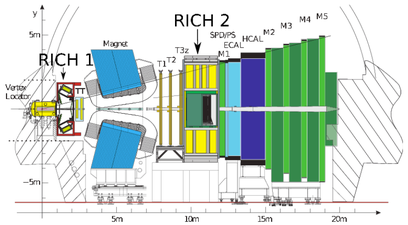
- 5 probabilities for different ID hypotheses

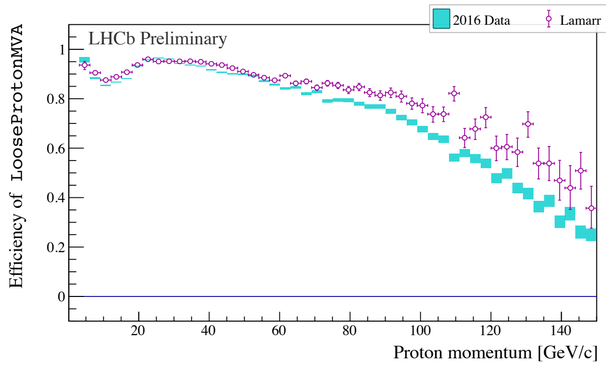
RICH response is probabilistic and driven by track kinematics and occupancy level

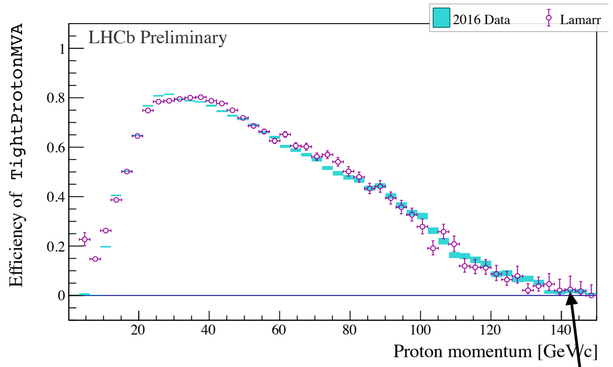
- (p, η, # of tracks)

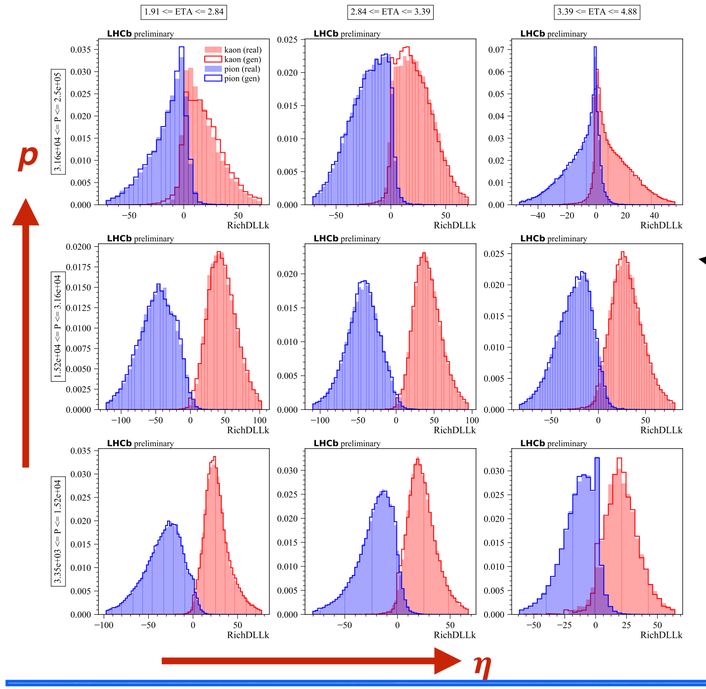
Ideal setup for 3→5 conditional generative model

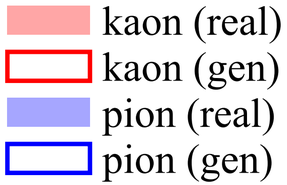
- GAN trained on ID calibration datasamples











Statistical distributions of ID variables are pretty close

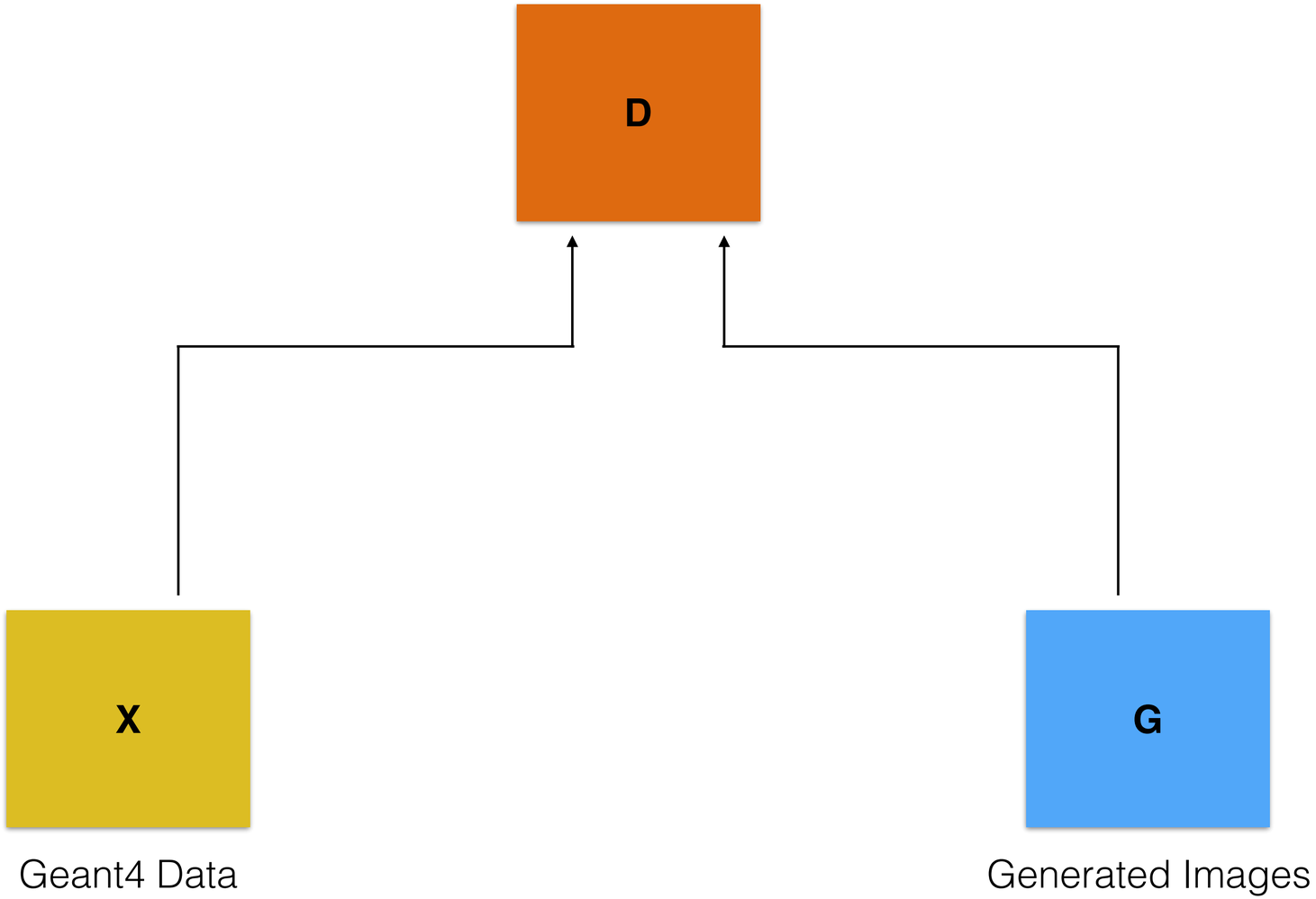
Precision of the generated response is evaluated for baseline selections

Minor discrepancies are attributed as systematics

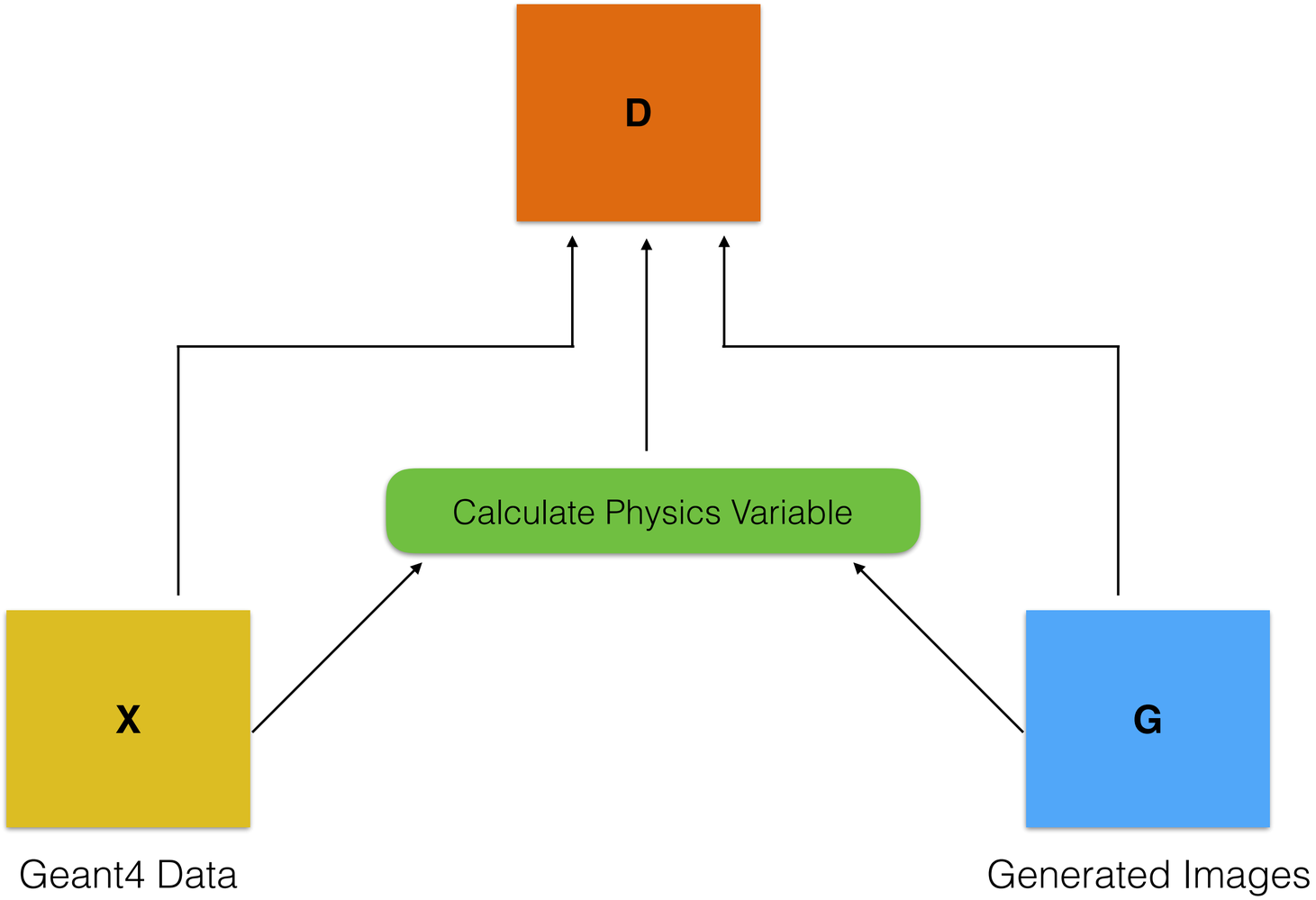
This approach allows to exclude RICH from the GEANT simulation completely

Thanks Fedor Ratnikov

Add Physics Variables in Training

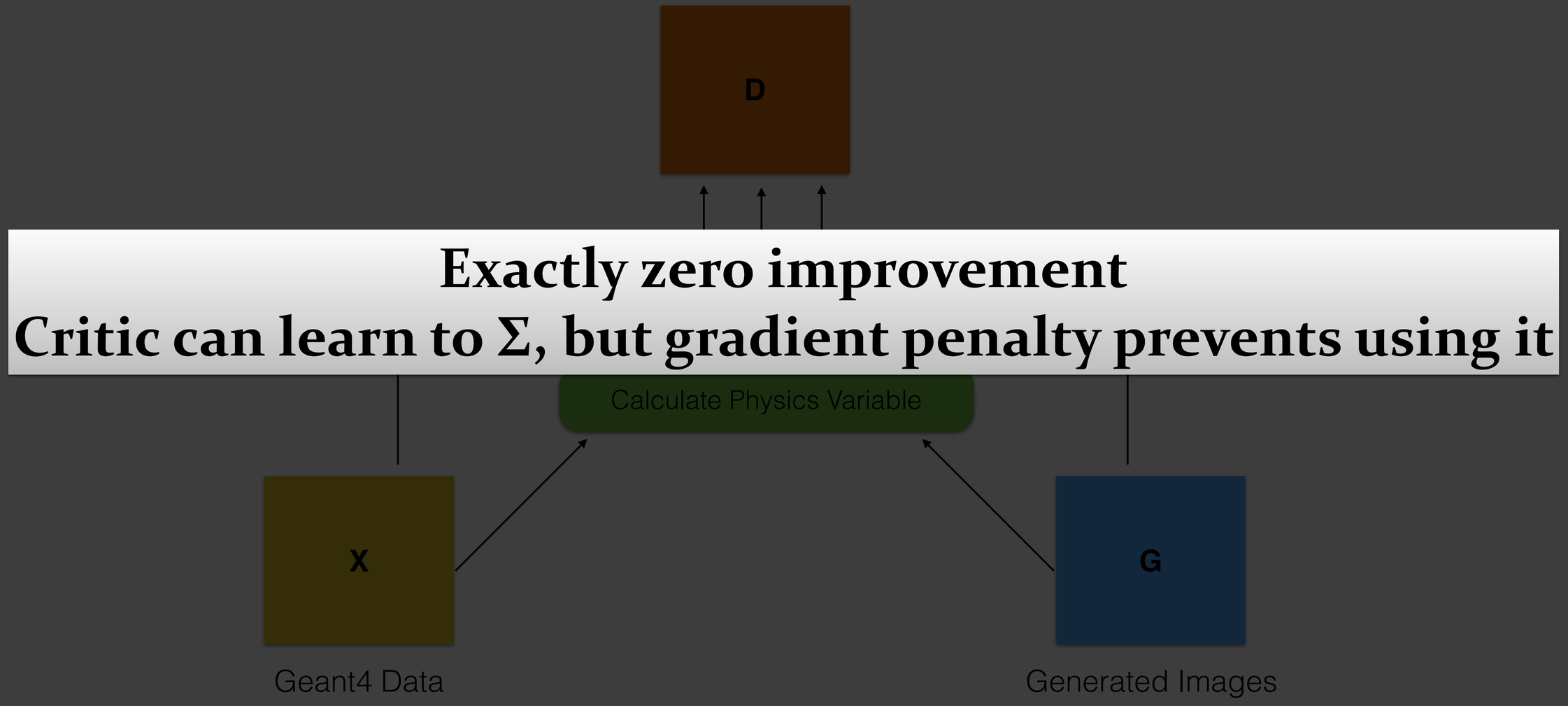


Add Physics Variables in Training

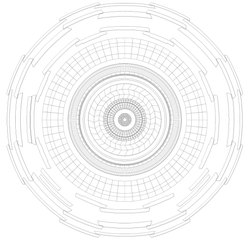


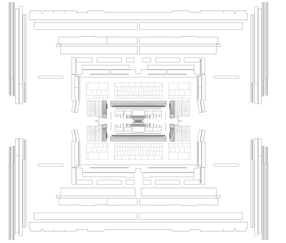
Help the discriminator see physics

Add Physics Variables in Training

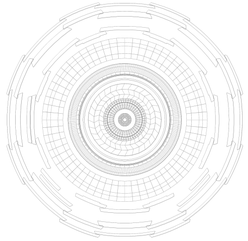


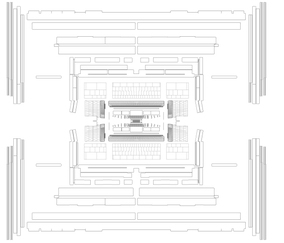
Help the discriminator see physics





Trade-Off b/w Distributions and Total Energy: How to get the best of both?

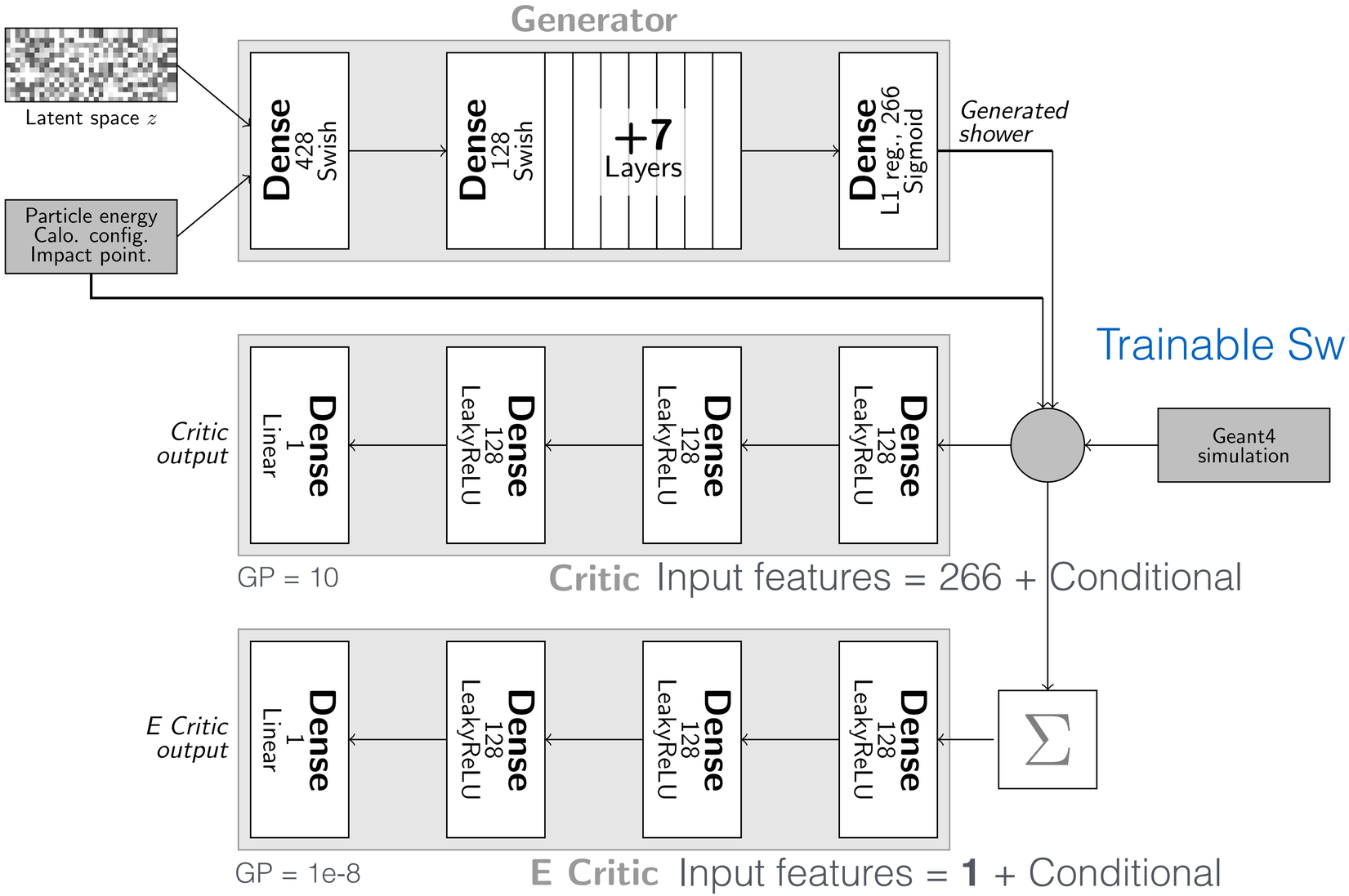




Trade-Off b/w Distributions and Total Energy: How to get the best of both?

“Train the Generator against a Critic of each type!”
-Gilles Louppe

New GAN Architecture



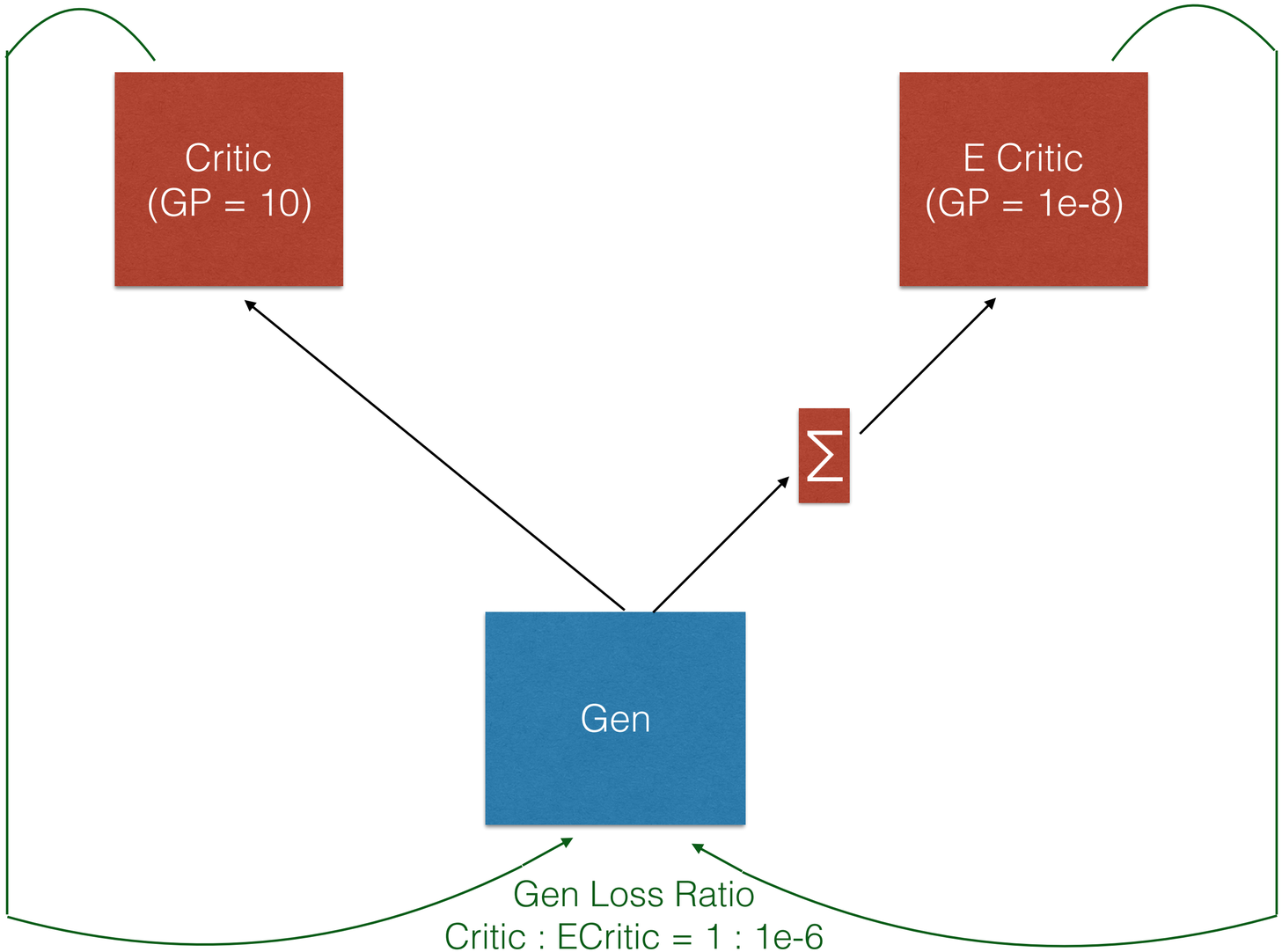
2 Critics

Deeper Generator needed

Trainable Swish activation for Generator

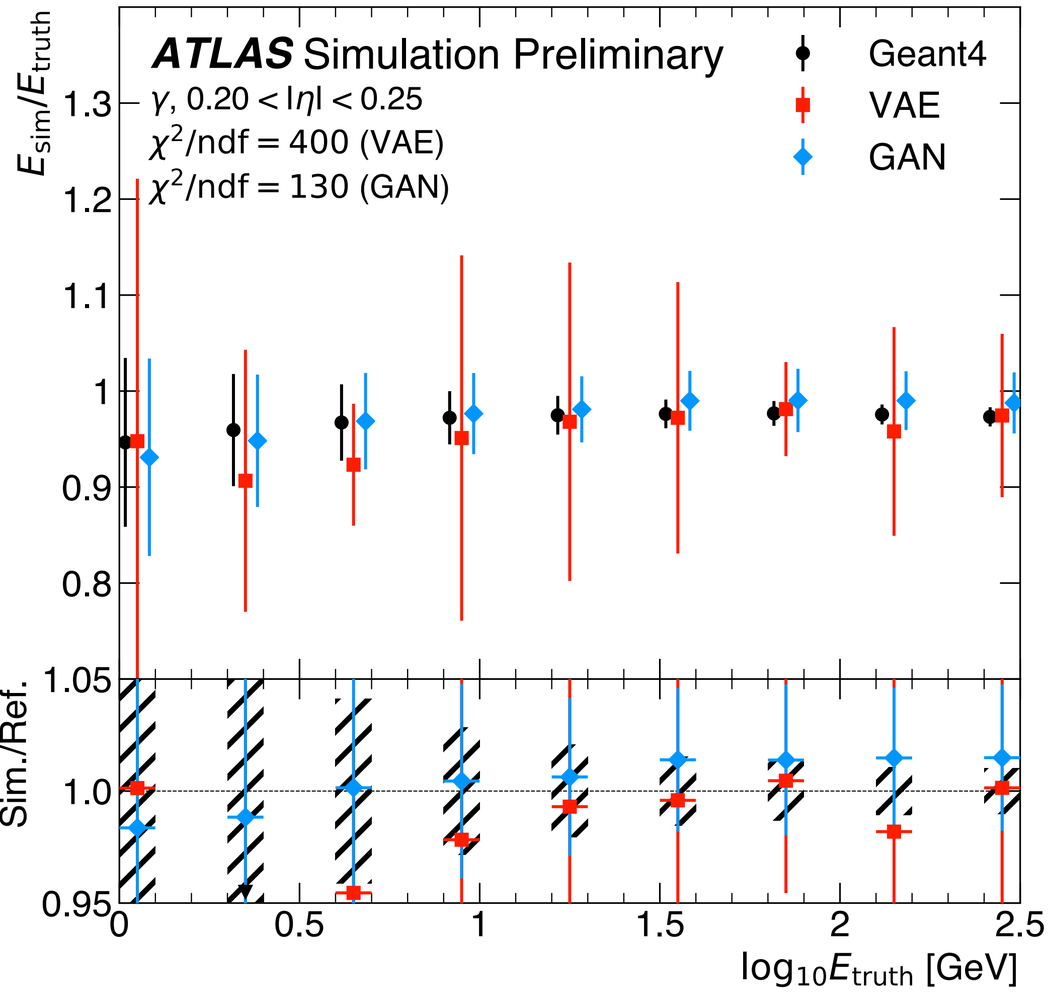
$$Swish(x) = x \cdot \mathrm{sigmoid}(\beta x)$$

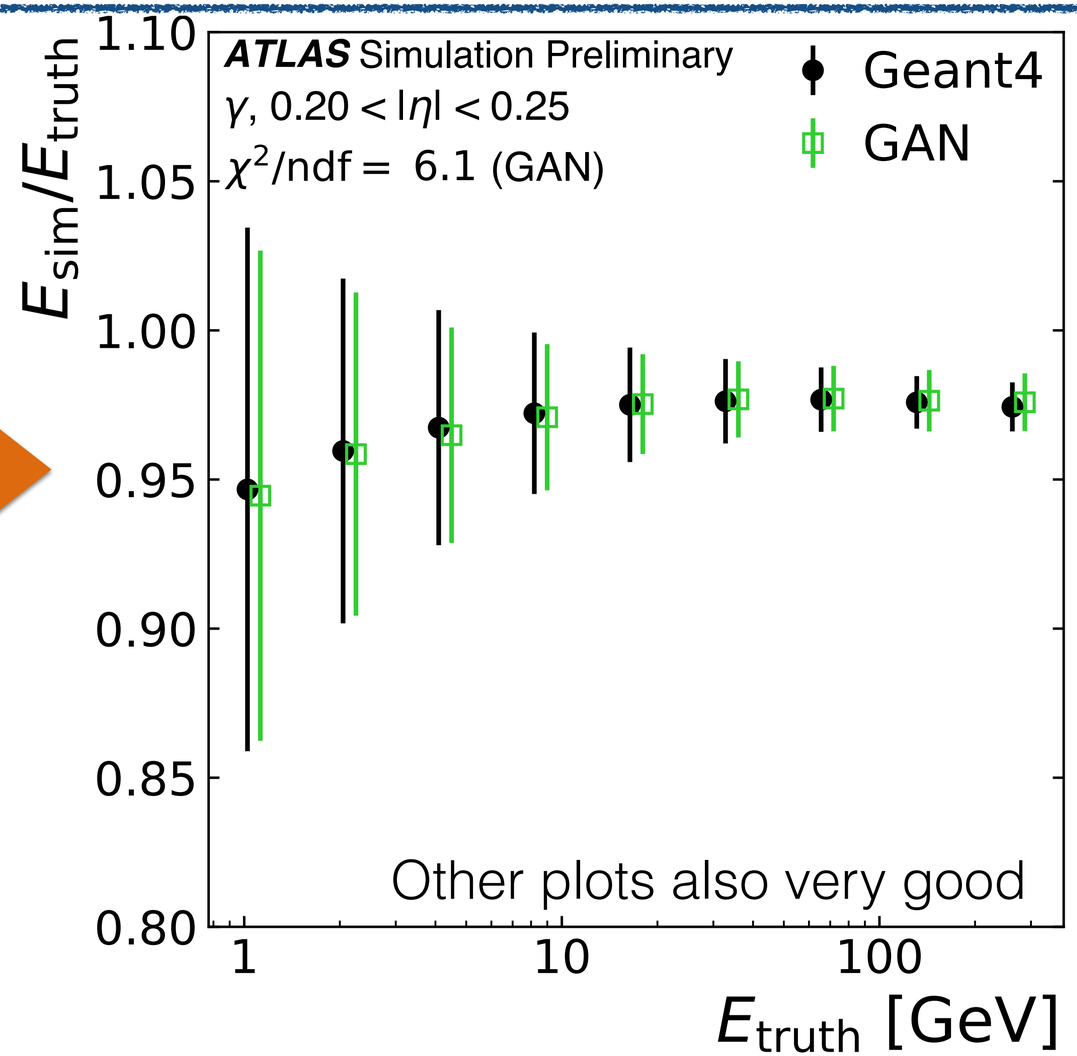
New GAN Architecture



GP: Two Sided Gradient Penalty

GAN: Improved Energy Resolution





Reference

Integration of DNN into ATLAS (C++) Software

Lightweight Trained Neural Network **Eigen based NN inference package for C++**

build passing coverity passed DOI 10.5281/zenodo.597221

- Light Weight Trained Neural Network package built for fast inference in C++ framework:
  - Minimal dependencies
  - Avoid integrating heavy Tensorflow/PyTorch into software
  - Looking into ONNX runtime

Now we can make fair comparisons

Performance (No GPUs, No Batch Parallelism):

- Both DNNCaloSim, FastCaloSimV2 ~**70ms** (**vs ~10s** for Geant4)
  - LWTNN takes **<1 ms per shower**, rest is overhead (being optimised)
- DNNCaloSim **memory footprint small**
  - **5 MB** for LWTNN JSON file **vs order GB** for FastCaloSimV2 parameterisation file

GAN as fast as it needs to be, tiny memory footprint

ALICE DCGAN

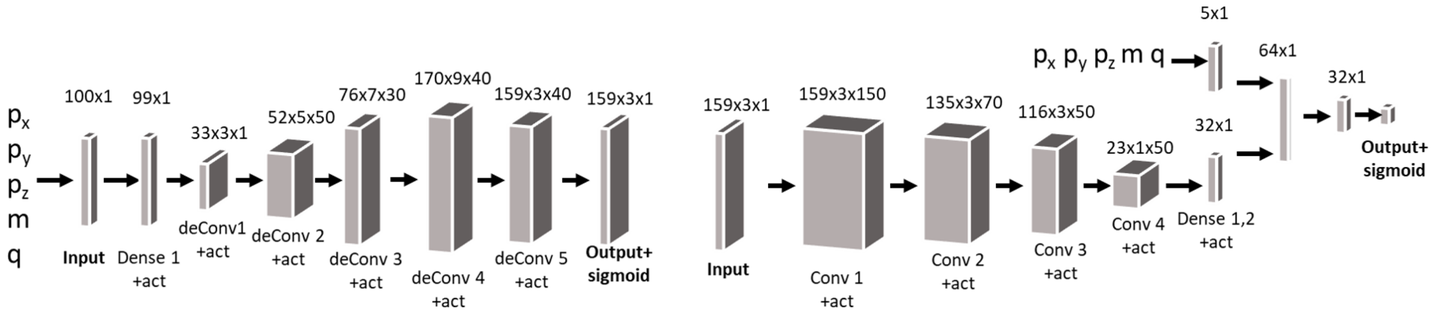


Figure 1. Architecture for the codintional DCGAN model. Each block represent a network's layer with its size given above. Network is trained on two individual inputs – generated noise and particle parameters

Table 1. Quality of conditional generative models, comparing to the GEANT3 simulation.

Method	**Mean MSE (mm)**	**Median MSE (mm)**	**speed-up**
GEANT3 (*current simulation*)	1.20	1.12	1
Random (*estimated*)	2500	2500	N/A
condLSTM GAN	2093.69	2070.32	10^2
condLSTM GAN+	221.78	190.17	
condDCGAN	795.08	738.71	25
condDCGAN+	136.84	82.72	

LHCb GAN

Crammer GAN:
Width 128
Depth 10
Activation ReLU
Latent Space 64
Discriminator Output 256

CMS GAN

1 Generator
1 Discriminator

Trained only on Geant4:
1 Constrainer Network for Energy
1 Constrainer Network for Impact Position

CaloGAN

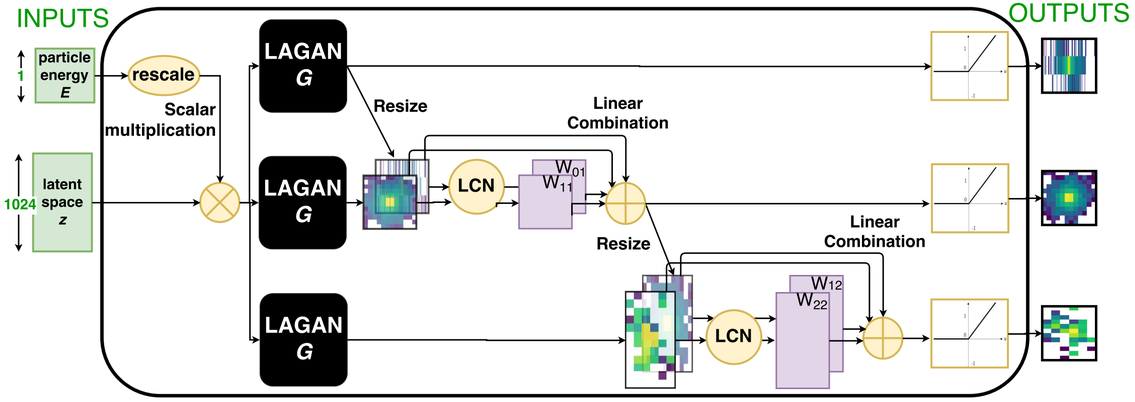


FIG. 4: Composite Generator, illustrating three stream with attentional layer-to-layer dependence.

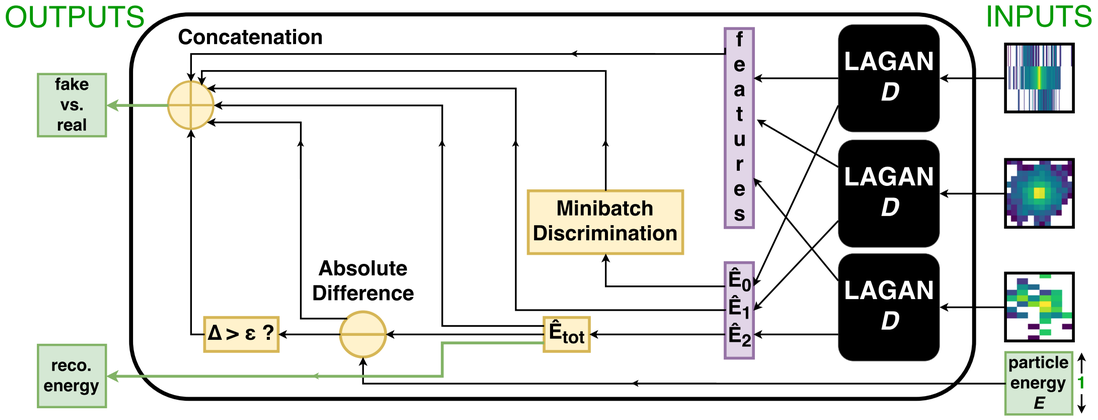


FIG. 5: Composite Discriminator, depicting additional domain specific expressions included in the final feature space.

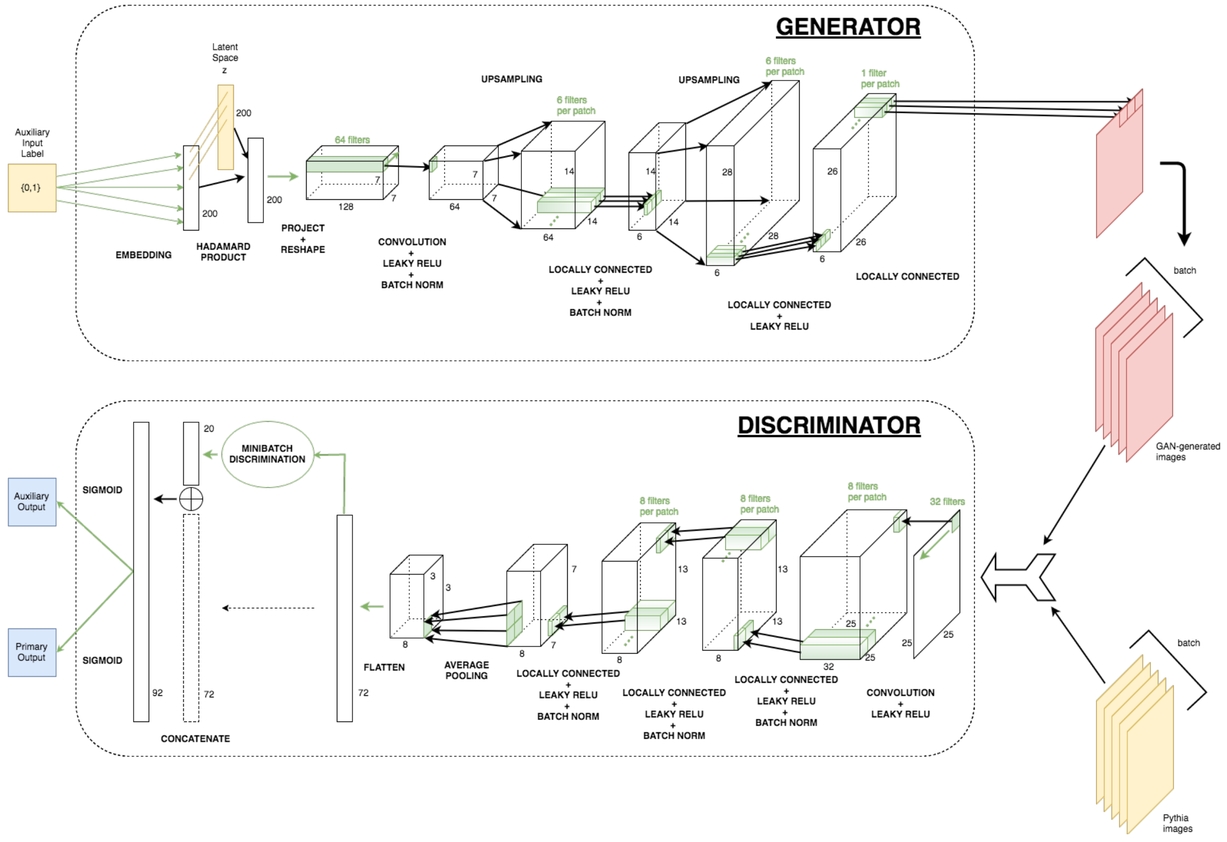


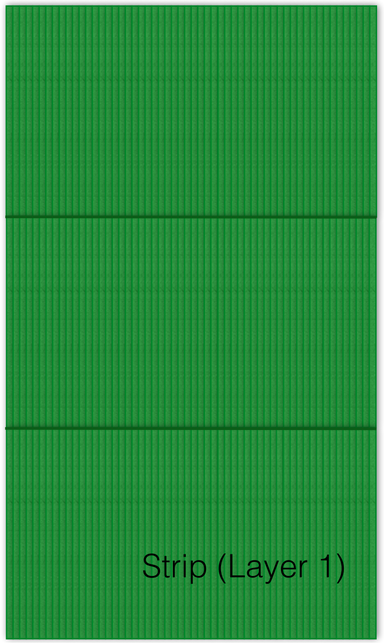
Figure 4: LAGAN architecture

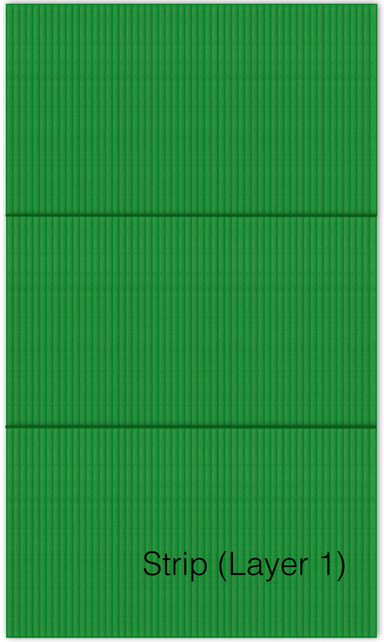
MMD Loss

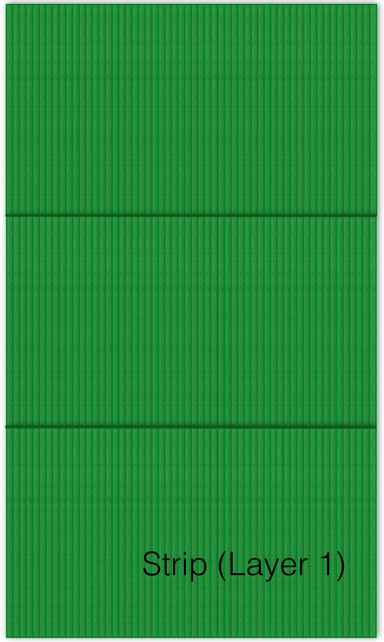
$$\mathrm{MMD}^2(P_T, P_G) = \left\langle k(x, x') \right\rangle_{x,x' \sim P_T} + \left\langle k(y, y') \right\rangle_{y,y' \sim P_G} - 2\left\langle k(x, y) \right\rangle_{x \sim P_T, y \sim P_G} ,$$

$$k_{\mathrm{Gauss}}(x, y) = \exp -\frac{(x-y)^2}{2\sigma^2} \qquad \text{or} \qquad k_{\mathrm{BW}}(x, y) = \frac{\sigma^2}{(x-y)^2 + \sigma^2} ,$$

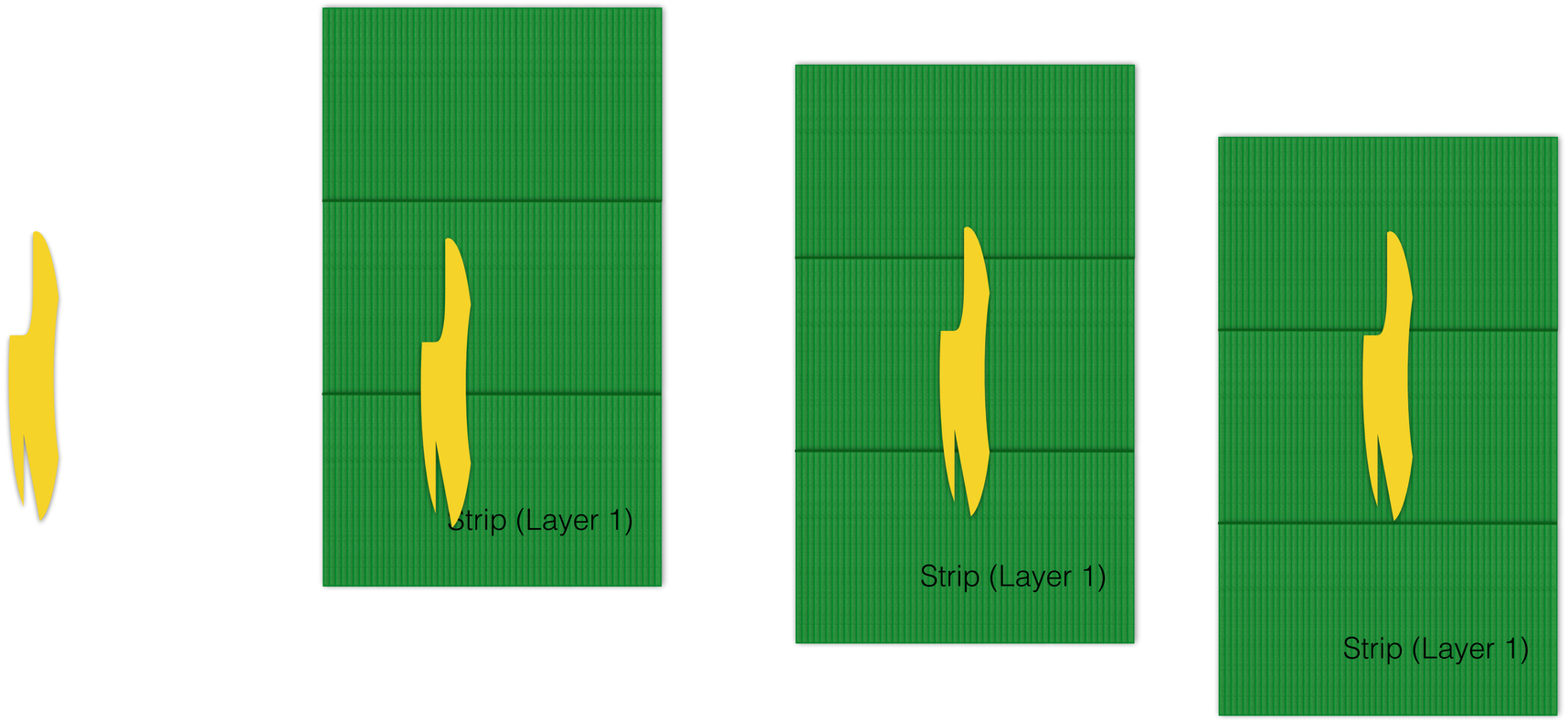
From SciPostPhys.7.6.075



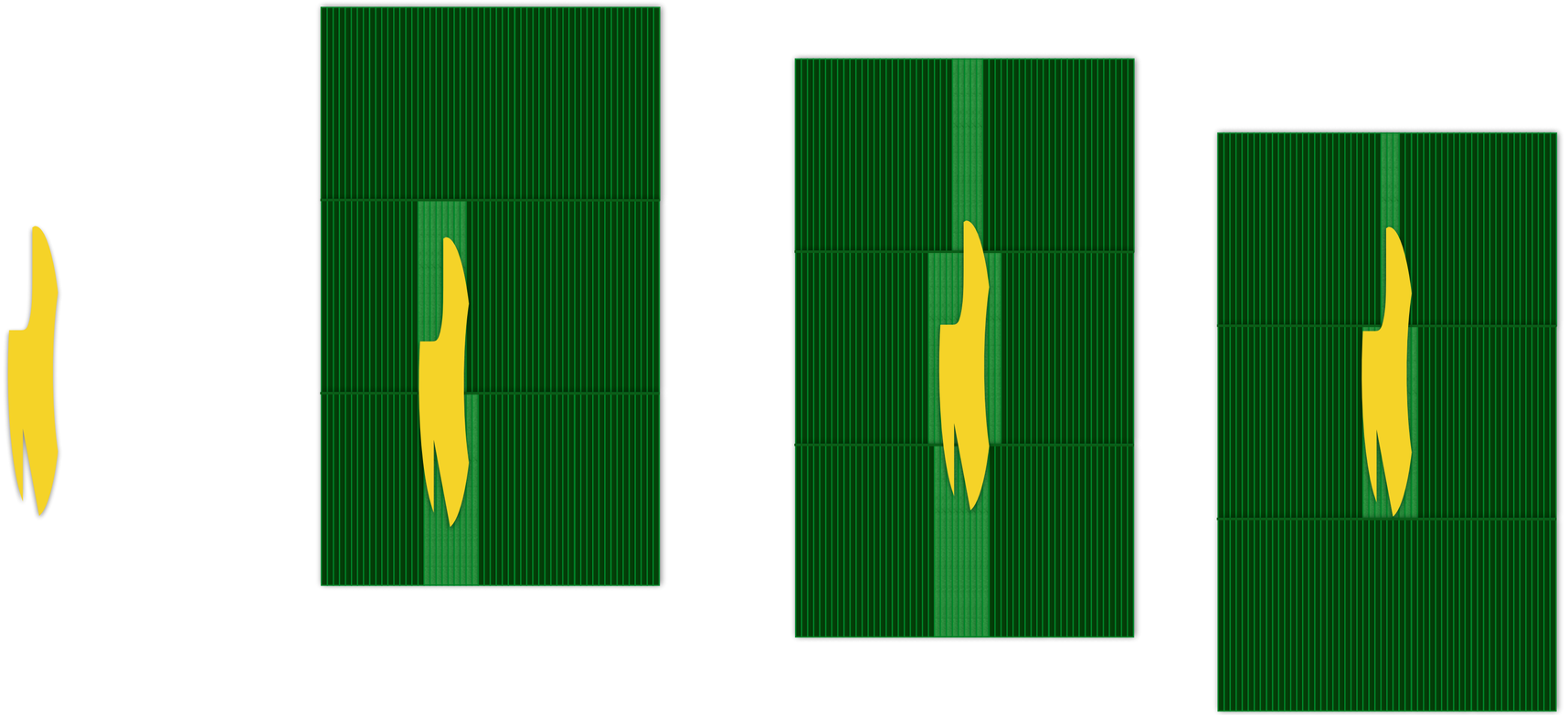




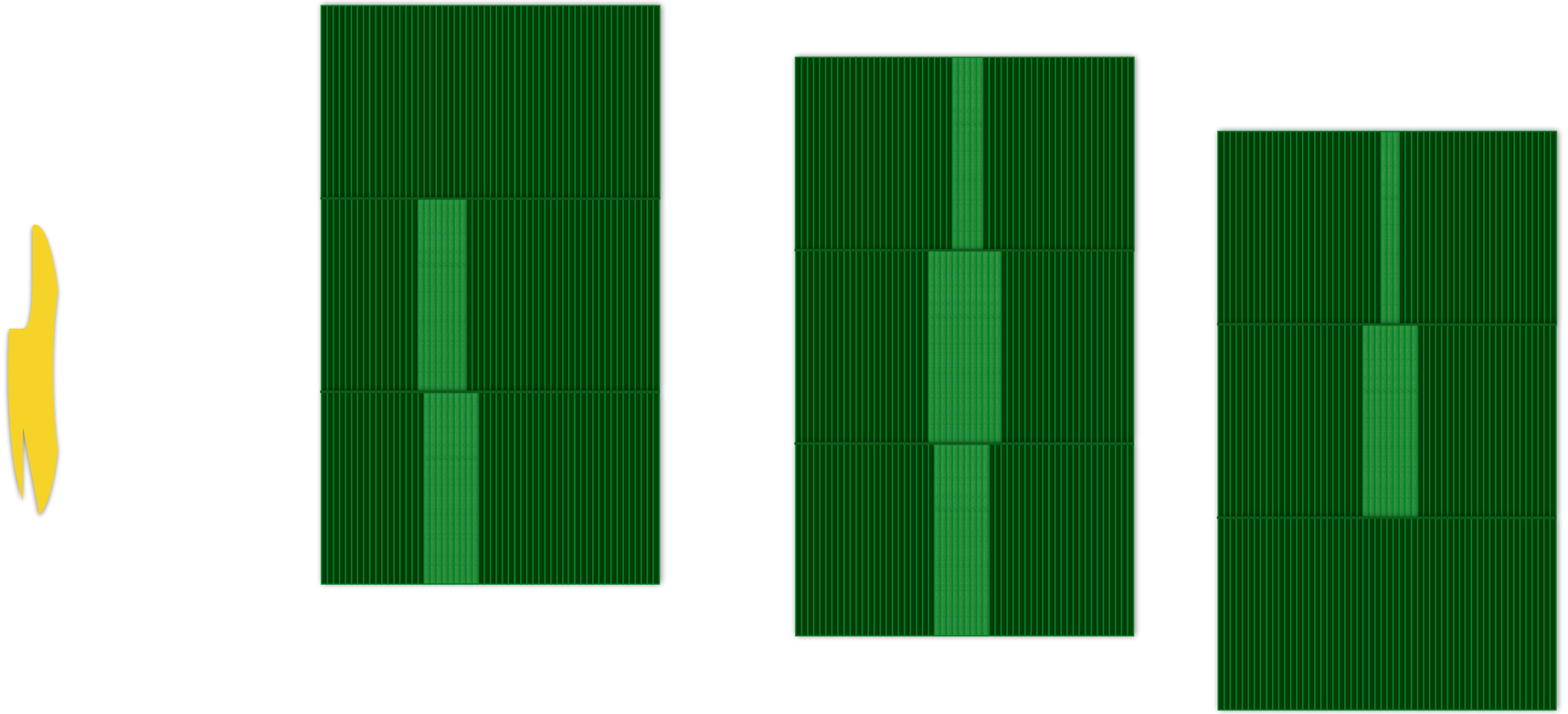
Same shower pattern, different image!



Same shower pattern, different image!



Same shower pattern, different image!



Same shower pattern, different image!

We have ignored this so far

Outline

1. **Need for fast simulation**
2. **Traditional techniques**
3. **Generative models: GANs, VAEs**
4. **Approaches taken by different experiments**
5. **Future prospects**