Deep Generative Models for Fast Simulation in ATLAS

Graeme Stewart (CERN), Aishik Ghosh, David Rousseau (LAL, Orsay), Kyle Cranmer (NYU), Michele Faucci Giannelli, Serena Palazzo (University of Edinburgh), Stefan Gadatsch, Tobias Golling, Johnny Raine, Dalila Salamani, Slava Voloshynovskiy (UniGe), Gilles Louppe (ULiège) on behalf of ATLAS

IML Workshop
17 April 2019
Generative Models for EM Shower Simulation

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

- CaloGAN showed that it is possible to simulate EM showers for a detector like ATLAS using GANs
- Since then you’ve seen many GANs for particle physics
- What’s so special about these Generative Models (GAN, VAE)?
First efforts to simulate the real, present day, irregular, coarse granularity ATLAS calorimeter with Generative Models
Generative Models for EM Shower Simulation in ATLAS

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

The simulation principle and performance of the ATLAS fast calorimeter simulation FastCaloSim

7 Conclusion

The fast calorimeter simulation FastCaloSim has been developed in order to reduce the simulation time in the ATLAS calorimeter system from several minutes to a few seconds per event, using a parametrization model for the longitudinal and lateral shower development of photons, electrons and charged pions. The

- ATLAS already using fast simulation techniques for years!
- Trade-off between slow accurate G4 and fast less accurate FastCaloSim V1
- New FastCaloSim V2 using some ML techniques already in advanced state of development

CaloGAN Paganini et al.

ATL-SOFT-PUB-2018-002

10th July 2018

The new Fast Calorimeter Simulation in ATLAS

The ATLAS Collaboration

during the simulation jobs. A prototype is being tested and validated, and it has shown significant improvements in the modelling of cluster-level variables in electromagnetic and hadronic showers.
First efforts to simulate the real, present day, irregular, coarse granularity ATLAS Calorimeter with Generative Models

(Only small eta region in barrel for photons right now)

Train and evaluate using G4 simulations for the ATLAS geometry

G4 takes ~10 seconds per shower for a 65 GeV single photon shower (more for higher energy) Too slow!

Human designed parameterisation techniques being developed for many years -> A high benchmark against which to compare GAN / VAE performance

Validation cross-check frameworks already in place for FastCaloSim including variables defined by the EGamma group: same level of scrutiny for all fast simulation approaches.

Need to get all distributions right simultaneously, average distributions might look right but must verify also the distributions per energy point / section of the calorimeter

Still have to parameterise separately for η slices, energies, interpolation …
Introduction (Finally)

Is it possible to one day...?:

Simulate showers 100-1000x faster than Geant4

Less human time intensive, higher accuracy than current fast simulation methods

Have it run inside Athena (ATLAS C++ software) and be less resource hungry than current fast simulation methods

Geant4 requires significant resources with ~75% spent in shower simulation i.e. Calorimeter simulation

Imperative to develop fast shower simulations compared to Geant4
PubNote: VAE and GAN

**VAE:**

100 epochs, 2 mins, CPU

50k ‘epochs’, 7 hours training, 1 GPU

**GAN:**

Training dataset:
- Single photon samples from Geant4
- 88000 events
- 9 energy points: \{1, 2, 4, 8, 16, 32, 65, 131, 262\} GeV
- \(0.20 < |\eta| < 0.25\)
- 4 electromagnetic calorimeter layers

Data preprocessing
- Negative energies set to 0
- Mirror \(\eta < 0\)

(WGAN-GP, Improved WGAN-GP nightmare on Keras!)

From summer PubNote 2018

The underflow and overflow is included in the first and last bin of each distribution, respectively.

Average energy deposition in the cells of the individual calorimeter layers as a function of the distance from the impact point of the particles for photons with an energy of approximately 65 GeV, 0.20 < |η| < 0.25.

Energy in Middle Layer

10^10, 10^3, 10^2, 10^1 Energy [GeV]

Average η in Middle

Geant4: Fluctuations due to small training size, fixed since PubNote. 4% -> 50% of dataset by removing momentum from Adam optimiser, lowering number of epochs.
2018 Results (2/3)

Average $\phi$ in Middle

Shower Depth
η, φ, other distributions not so bad but for total energy…

GAN gets the means but not the widths of the energies

Critic can’t see the difference in real and fake images.

Tried training on single high energy point, Minibatch discrimination, various other tricks. No result.
The term \( L_{\text{GAN}} \) then reads \( L_{\text{GAN}} = E_{\tilde{x} \sim p_{\text{gen}}} [D(\tilde{x})] - E_{x \sim p_{\text{Geant4}}} [D(x)] + \lambda E_{\tilde{x} \sim p_{\tilde{x}}} [||\Delta_{\tilde{x}} D(\tilde{x})||_2 - 1]^2 \).
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, at Atas ML Workshop 2018
New GAN Architecture

2 Critics

Deeper Generator needed

Trainable Swish activation for Generator

Swish(x) = x \cdot \text{sigmoid}(\beta x)

Swish activation inspired from Giles Strong’s presentation also at AML Workshop 2018

Training time: 2.5 Days on 1 GPU for 1.5k Epochs
Training Size: 44000 events (50% of Dataset)
CPU = 2 x GPU training time at 52% GPU utilisation
New GAN Architecture

GP: Two Sided Gradient Penalty
Careful: Sum Inside or Outside the Network?

\[ \sum = \text{Lambda(sumFunc)}(m\text{\_input\_image}) \]

They are not equivalent, need to tune hyper parameters differently

\[
L_{\text{GAN}} = E_{\tilde{x} \sim p_{\text{gen}}} [D(\tilde{x})] - E_{x \sim \text{Geant4}} [D(x)] + \lambda E_{\hat{x} \sim p_{\hat{x}}} [(||\Delta \hat{x} D(\hat{x})||_2 - 1)^2]
\]

GP on 1 input vs 266 inputs
GAN: Improved Energy Resolution

**ATLAS Simulation Preliminary**

- **Energy Resolution**
  - $\gamma$, $0.20 < |\eta| < 0.25$
  - $\chi^2/\text{ndf} = 400$ (VAE)
  - $\chi^2/\text{ndf} = 130$ (GAN)

**Other plots also very good**

$\sigma_{G4}$: [0.088, 0.058, 0.040, 0.027, 0.019, 0.014, 0.011, 0.0088, 0.0082]

$\sigma_{GAN}$: [0.081, 0.055, 0.037, 0.025, 0.017, 0.013, 0.011, 0.0101, 0.0096]

Reference
Inverse Autoregressive transformations

a type of Normalizing Flow to make the latent space more Gaussian

IAF transformations make the latent space distributions more Gaussian like.

VAE Latest Space

5D Latent Space don't look Gaussian

• Input: a variable with some specified ordering (multidimensional tensor)
• Output: \((\mu, \sigma)\) for each element of the input variable conditioned on the previous elements.

When we use the Decoder as a generator, it will be more correct to sample from a Gaussian distribution, impact on physics under study.
Integration of DNN into FastCaloSim in Athena

- Added Swish activation to LWTNN (Thanks Dan Guest!)
- Merged DNNCaloSim into Athena
- Find cell closest to extrapolated position of particle in entrance of Middle Layer (called “Impact Cell”), it’s $\eta$, $\phi$
- Build 266 cells around it, order in Sampling, $\eta$, $\phi$ increasing (mirror $\eta$ on left half)
- Mimic preprocessing of GAN
- Generate energy with DNN in LWTNN
- Mimic post-processing of GAN, and fill energy into CaloCells
- Validation comparisons of DNN with G4, FCS using standard EGamma definitions (ATLAS internal)
  - Only photons in the barrel

Resource utilisation:
- DNNCaloGAN ~ same speed as FastCaloSimV2
  - LWTNN takes <1 ms per shower
- DNNCaloGAN VmPeak also small, not a concern

Next:
- Getting a GAN trained on voxelised HITS level dataset integrated in Athena
Conclusion

• We have our first working Generative Network in Athena for our irregular shaped ATLAS Calorimeter!
• Just a flag at run time allows to switch out one trained generative model for another
• Smart detector specific conditioning, preprocessing essential
• No plot GAN is unable to learn at all, now also getting the energy resolution correct
• Comparisons being made with the in-development FastCaloSim V2 using established, time tested validation framework
• Many ideas to further improve performance: Look at plots at fixed energy points, Improve Strip images (Additional Critic/Grad Penalty/ Convolutions …)
  • Train against elaborate Critics on DNN platforms, apply only Dense Generative network using Light Weight Trained Neural Network package in Athena (ATLAS Software)
• Future: More granular level data, Larger range in $\eta$, Separate GANs for different particles, Transfer Learning from LHC data …
Backup
The Calorimeter

2-D Axis: $\phi$ vs $\eta$

Particle goes through 4 layers in this order:

0. **Pre-Sampler**: Some energy deposit

1. **Strips**: Very granular in $\eta$; more energy deposit

2. **Middle**: Thickest layer, **maximum energy deposit**

3. **Back**: Little Energy deposits

Due to misalignment of the two halves of the detector, cells are not perfectly well aligned.

Different widths of cells further complicate the alignment between cells of different layers

Cells not granular enough to see intricate details of shower pattern
Backpropagate through Sum?

When you train the Generator

Yes, gradients useful for Generator

When you train the E Critic

No, we don’t want to apply gradient penalty to each cell via the Sum function

Treat Sum as independent input feature, not as a sum of the other 266 features
The CaloGAN architecture (slightly modified version for the conditional version). Now with GANCaloSim, a much simpler architecture achieves the various complicated conditionings on physics and calorimeter configuration (although only photons)

Lots of room to scale up architecture with DDL.