2nd IML Workshop at CERN, 09 - 12 April 2018

Joint Wasserstein GAN contribution

Deep Learning group at the institute:
M. Erdmann, B. Fischer, L. Geiger, E. Geiser, **J. Glombitza,**
D. Noll, **T. Quast**, Y. Rath, M. Rieger, M. Urban, R. Smida,
F. Schlüter, **D. Schmidt**, M. Wirtz

III. Physikalisches Institut
RWTH AACHEN UNIVERSITY

10 April 2018

SPONSORED BY THE
Federal Ministry of Education and Research

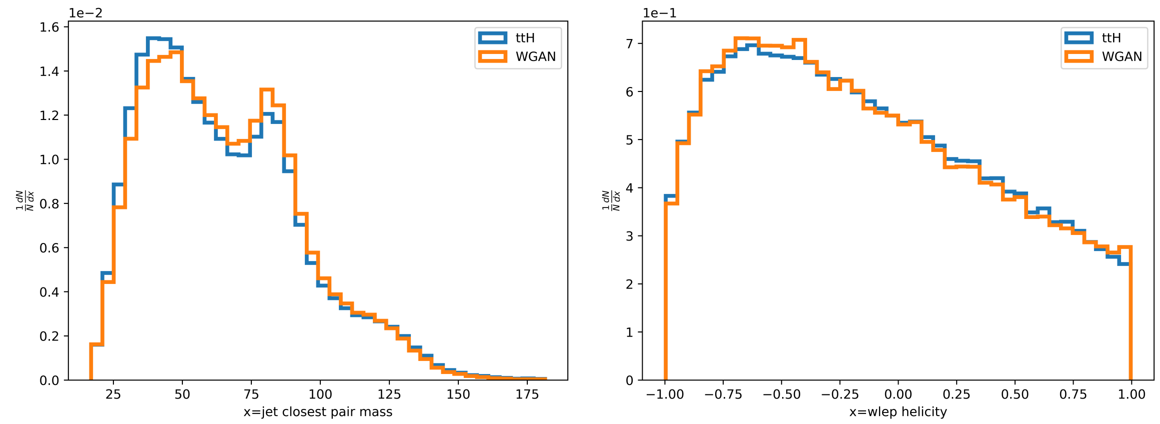
PIERRE AUGER OBSERVATORY

CMS

WGAN Joint Talk

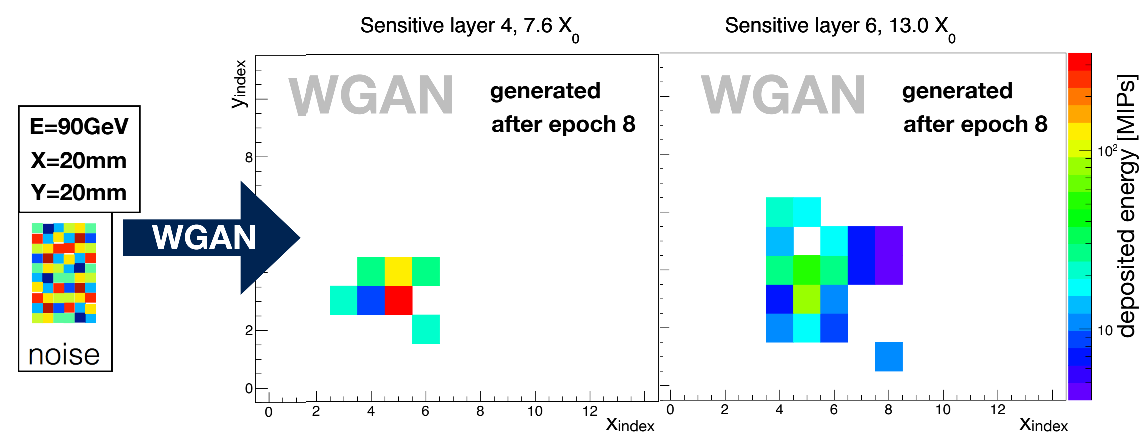
1. Generating high-level physics variables based on Monte Carlo simulated ttH events using Wasserstein GANs

David Schmidt, RWTH Aachen



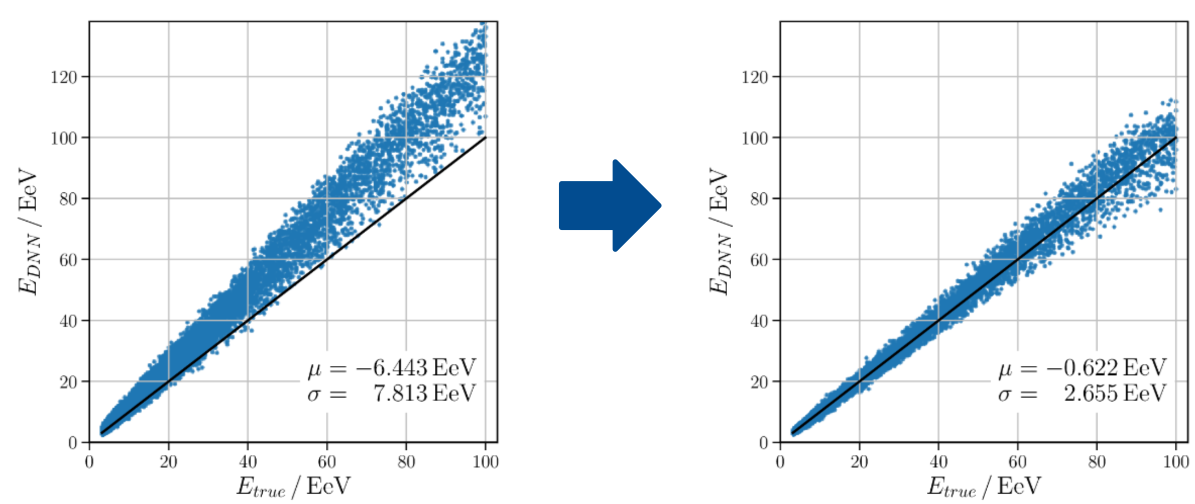
2. Conditional Wasserstein GANs for fast simulation of electromagnetic showers in a CMS HGCAL prototype

Thorben Quast, CERN/RWTH Aachen



3. Refining Detector Simulation using Adversarial Networks

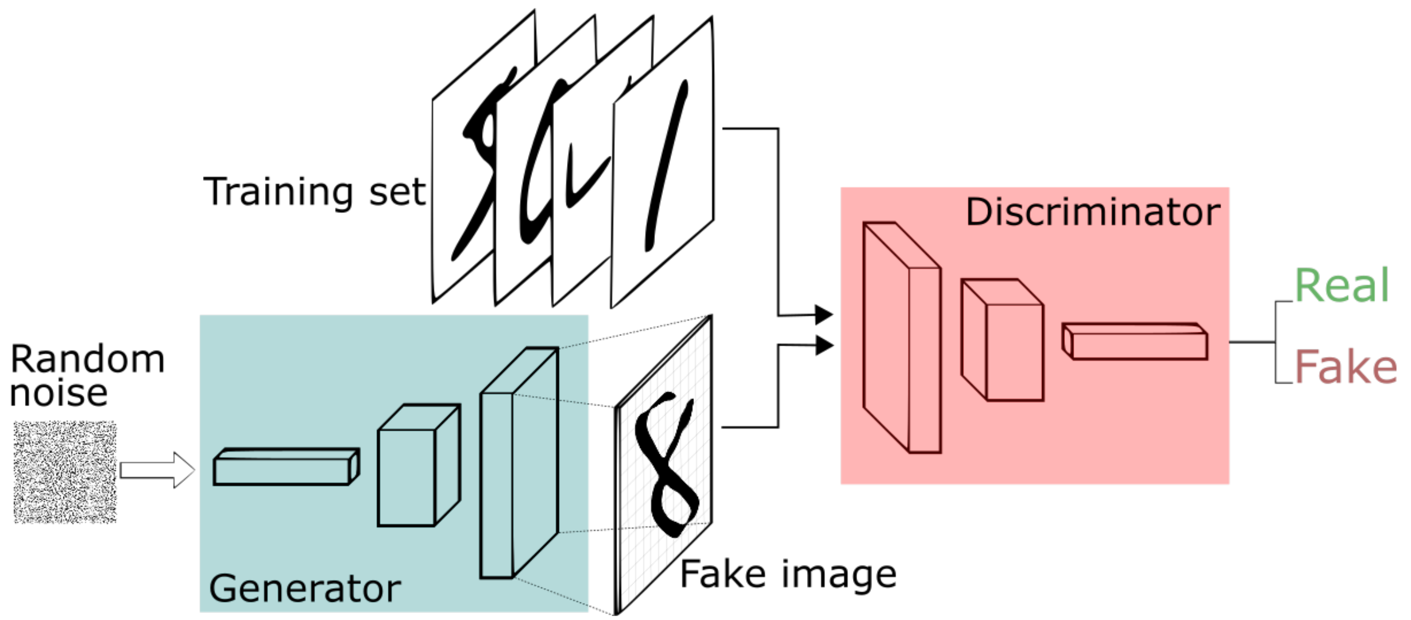
Jonas Glombitza, RWTH Aachen



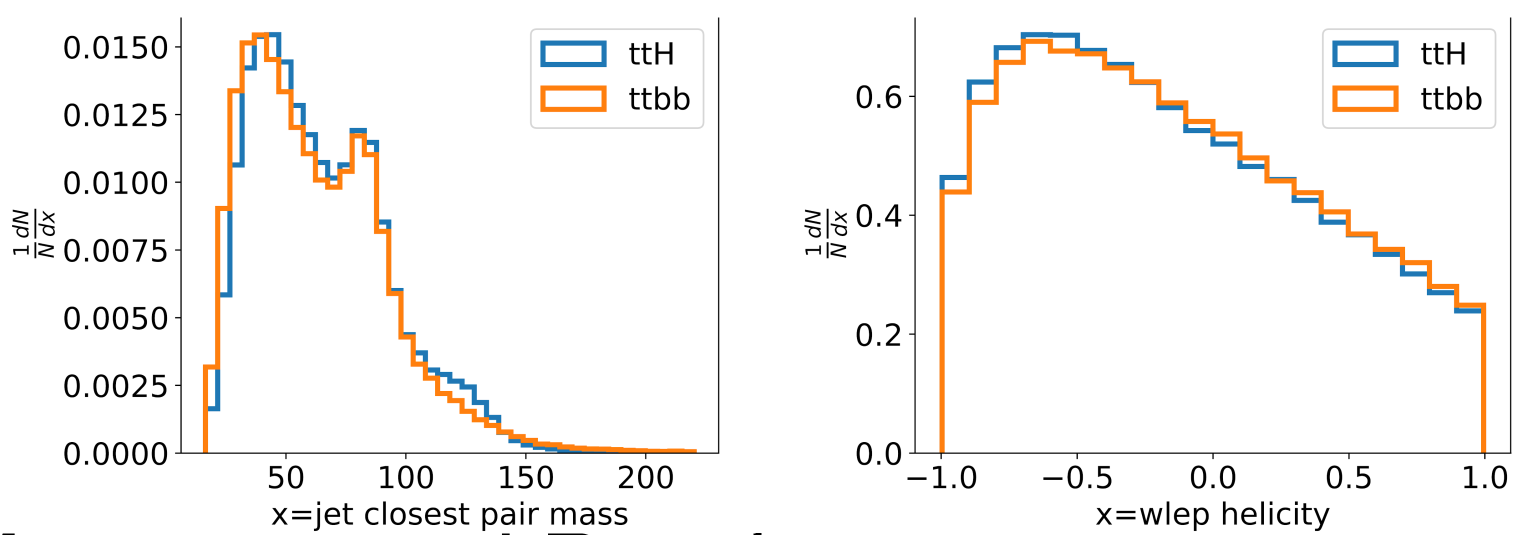
Generating high-level physics variables based on Monte Carlo simulated *ttH* events using ***Wasserstein GANs***

Outline - High-level variable generation

- Generative Adversarial Networks



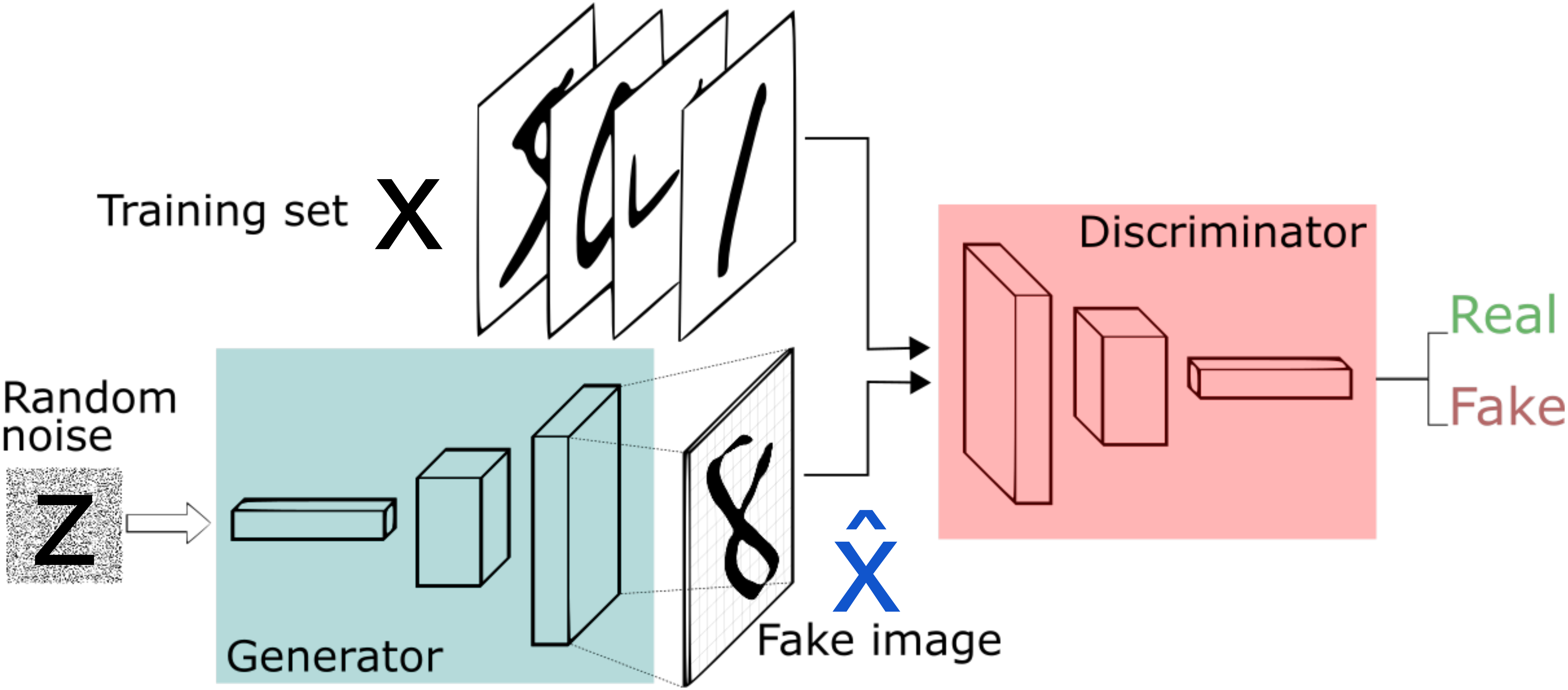
- Formulating Benchmark on ttH data



- Quality Measures and Results

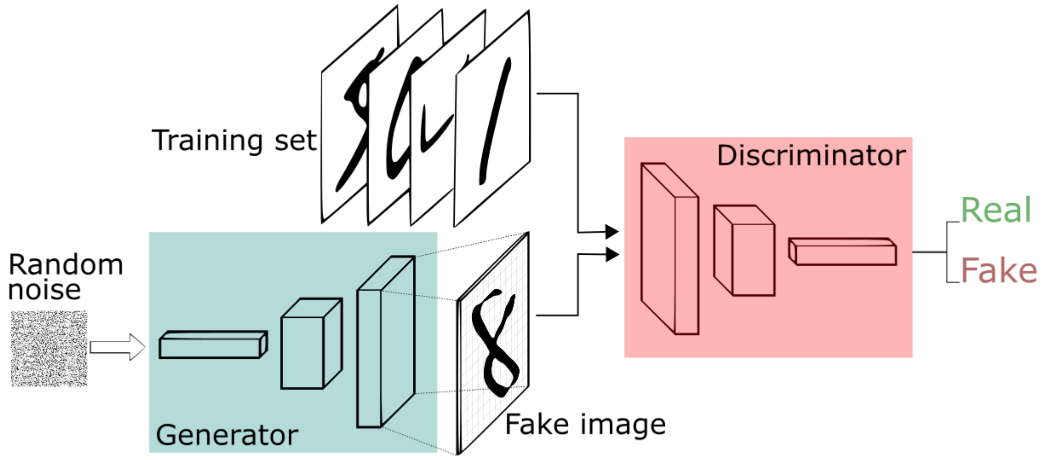
Generative Adversarial Network

arXiv:1406.2661

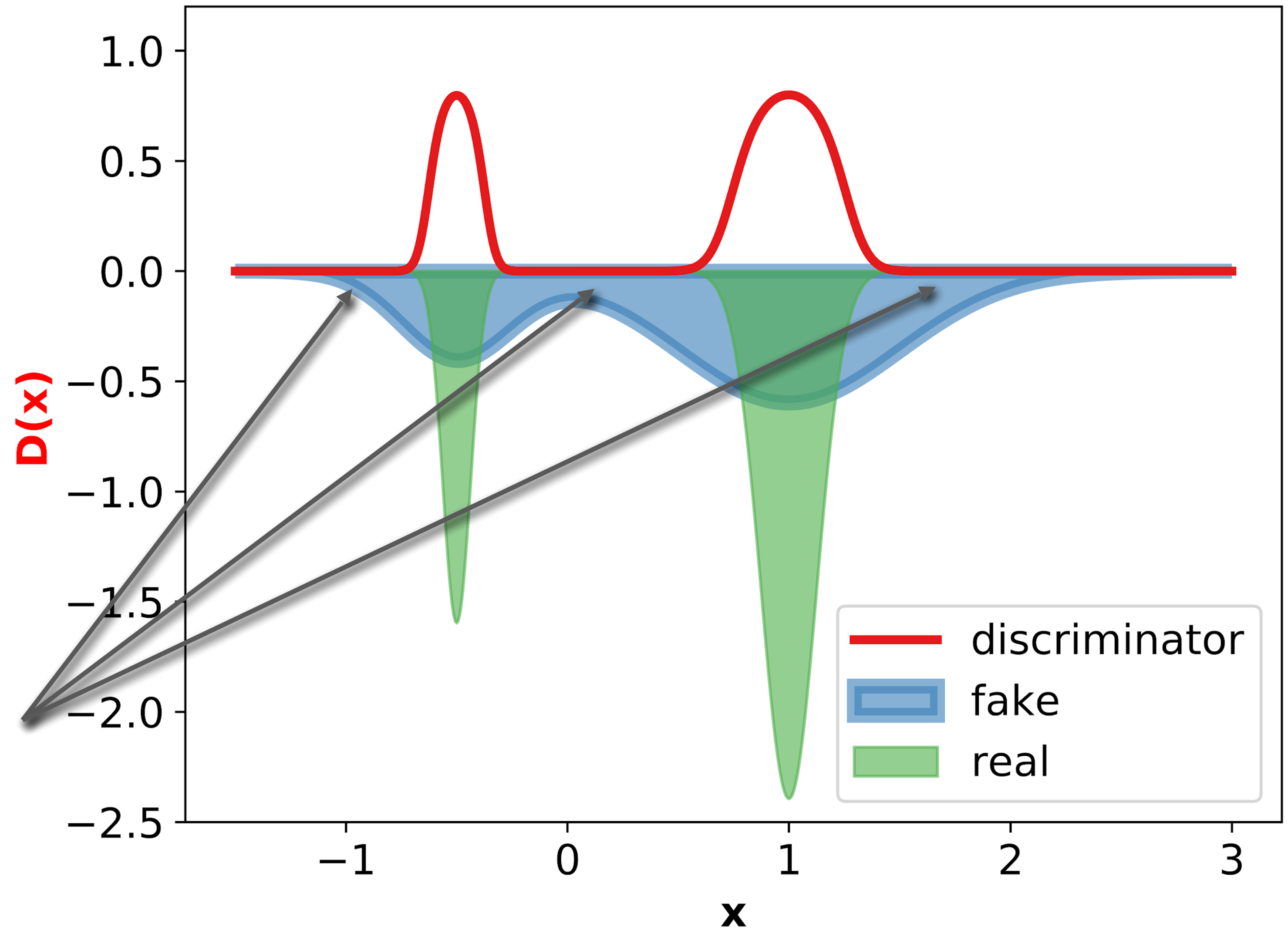


- Discriminator loss: $-\ln D(x) - \ln\left[1 - D(\hat{x})\right]$
- Generator loss $-\ln D(G(z))$ or $\ln\left[1 - D(G(z))\right]$

Generative Adversarial Network

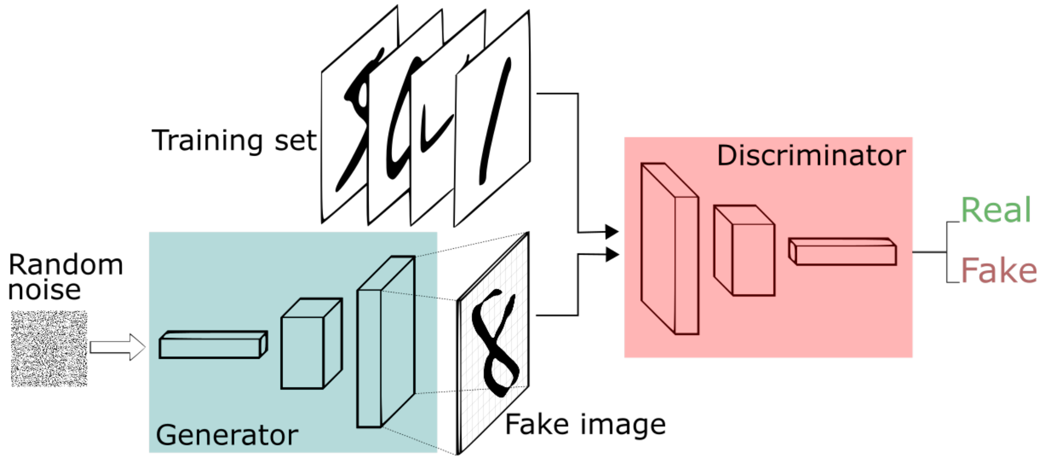


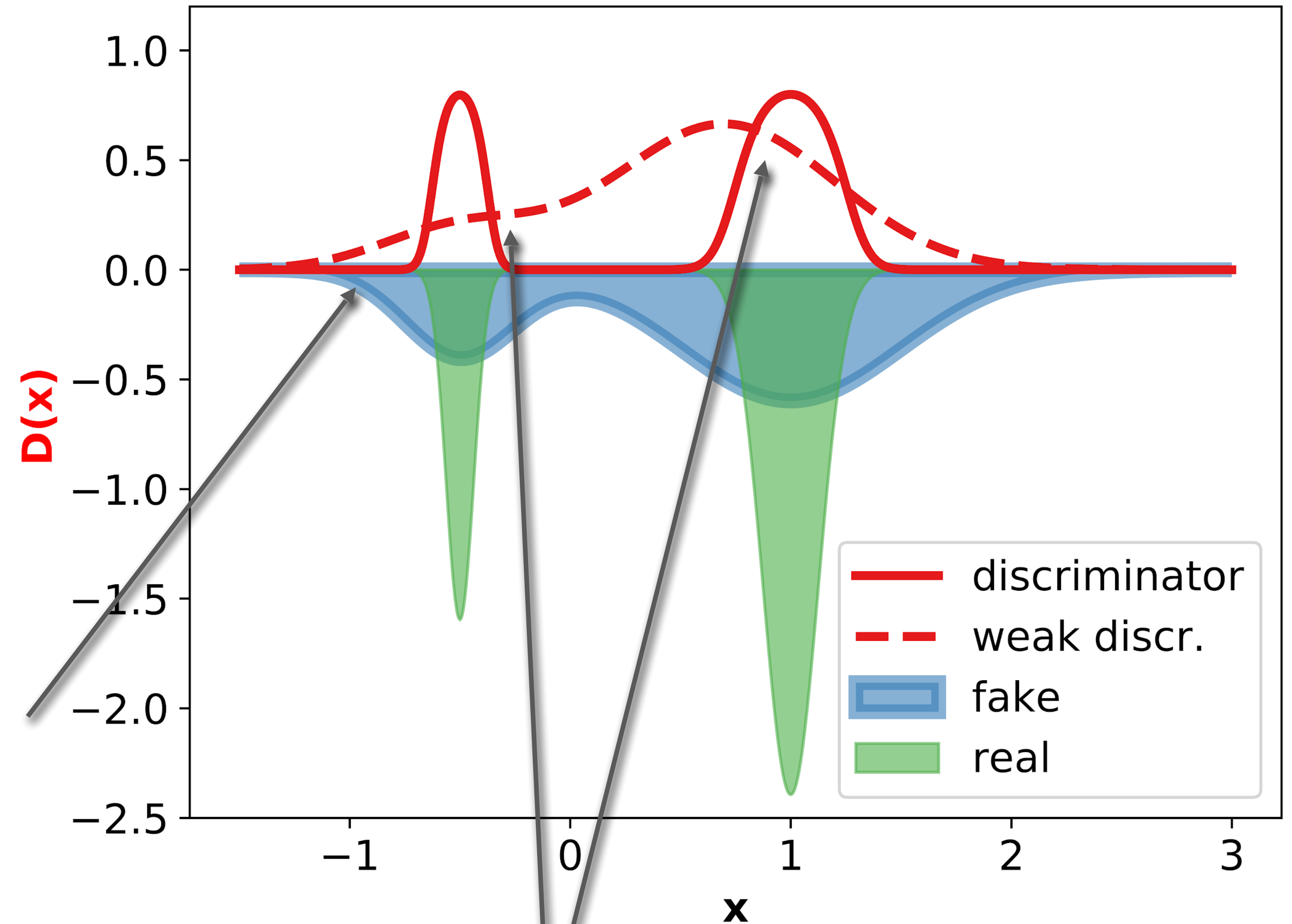
Fully trained discriminator has vanishing gradients which are useless for generated examples!



David Schmidt
david.schmidt@rwth-aachen.de

Generative Adversarial Network





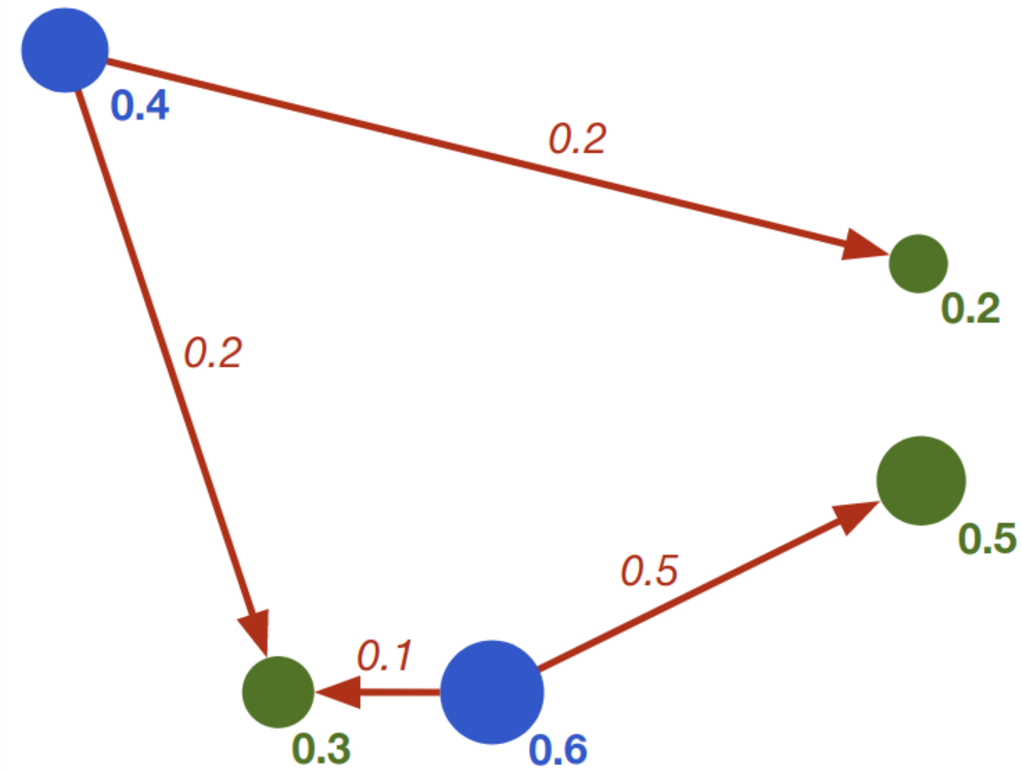
Fully trained discriminator has vanishing gradients which are useless for generated examples!

Untrained discriminator gives only vague gradients pushing some generated samples away.

Wasserstein Distance

Also called **Earth-Mover-Distance**:

- Interpret one distribution as **target**,
  one as **earth heap**
- Distance of distributions = effort to move earth heap to target (mass x distance)



$$D_W = \underbrace{\min_{\gamma \in \prod(P_x, P_{\hat{x}})}}_{\text{optimal transport plan}} \underbrace{\mathbb{E}_{(x,\hat{x})\sim\gamma}}_{\text{mass}} \underbrace{\|x - \hat{x}\|_2}_{\text{distance}}$$

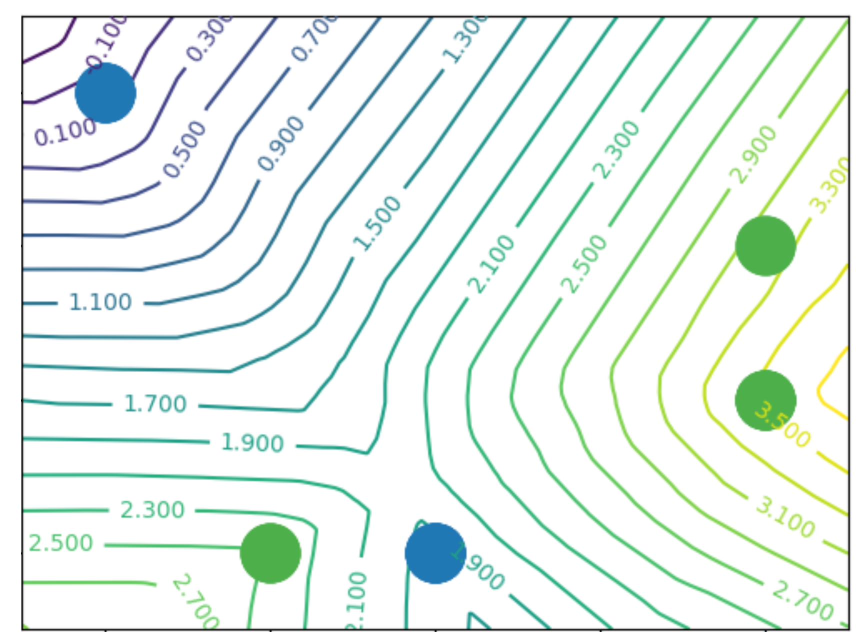
Kantorovich-Rubinstein duality:

$$D_W = \max_{C \in \mathrm{Lip}_1} -\mathbb{E}_{P_x} C(x) + \mathbb{E}_{P_{\hat{x}}} C(\hat{x})$$

$$\mathrm{Lip}_k : \|C'\|_2 \leq k$$

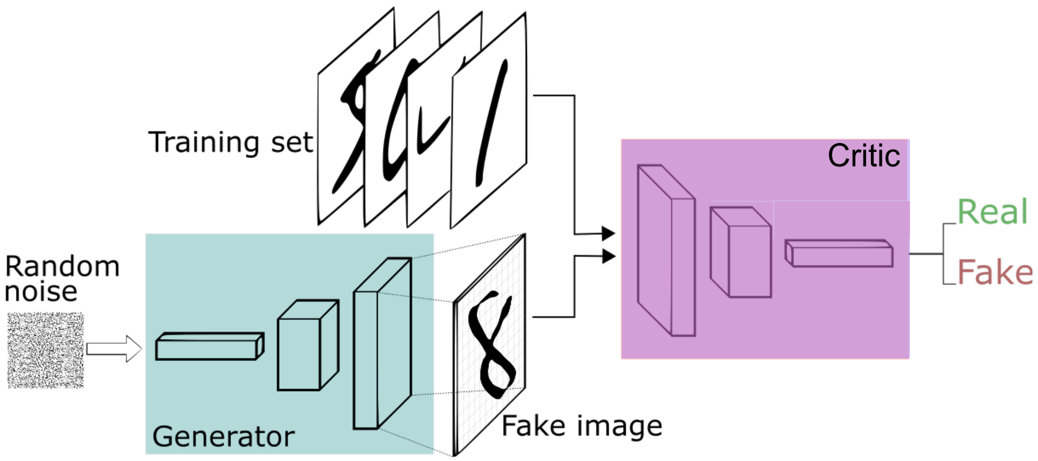
expectation value

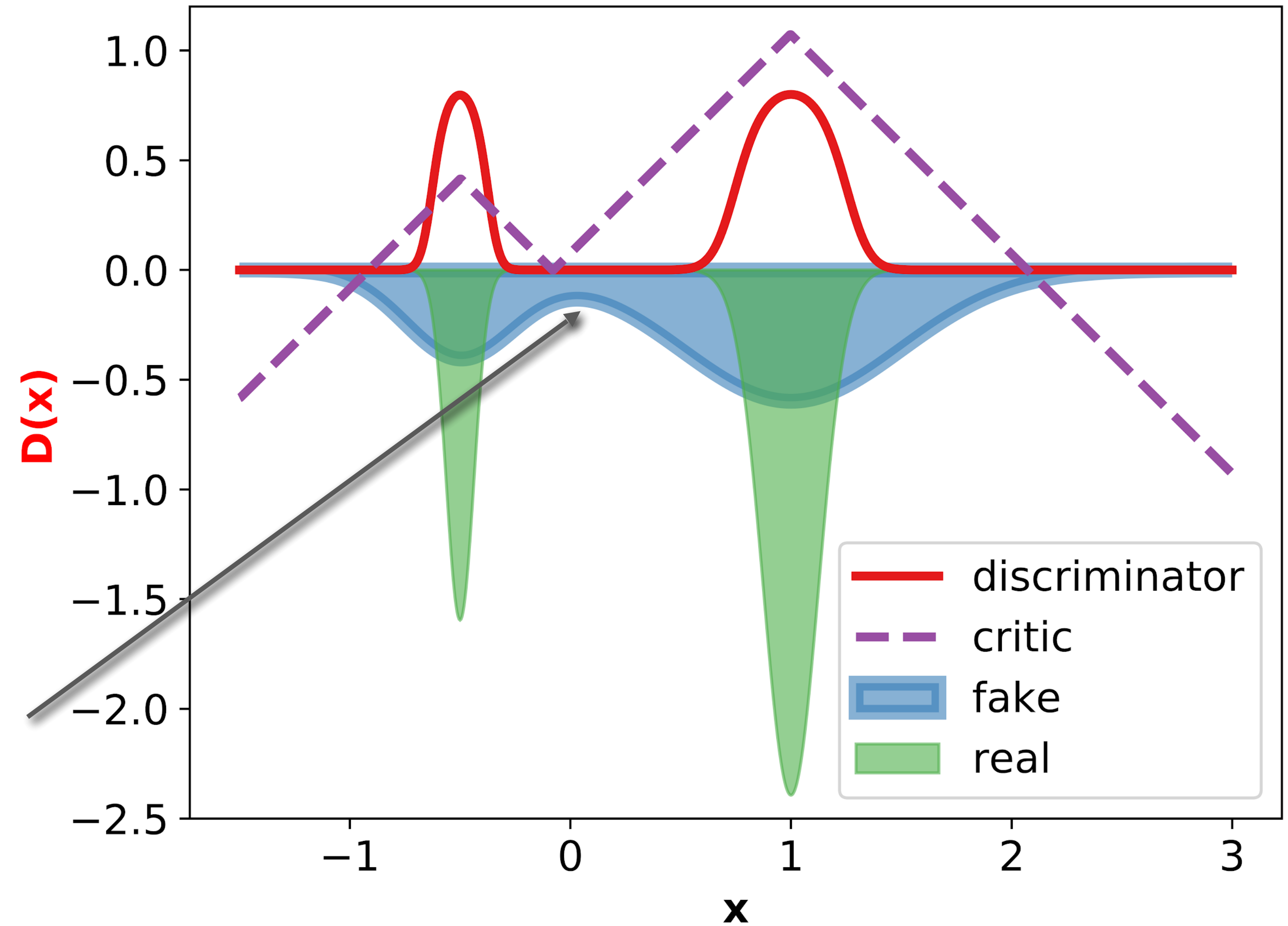
generator replaced by critic



Wasserstein GAN

arXiv:1704.00028



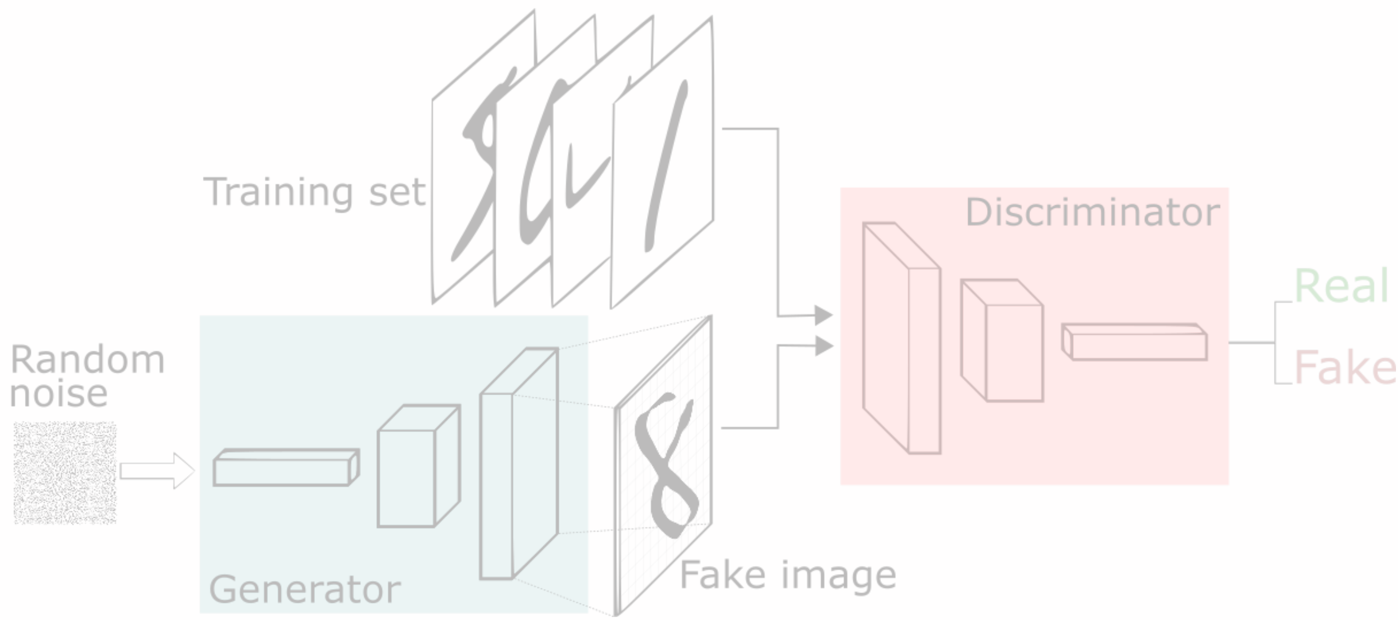


Converged critic provides meaningful gradients everywhere!

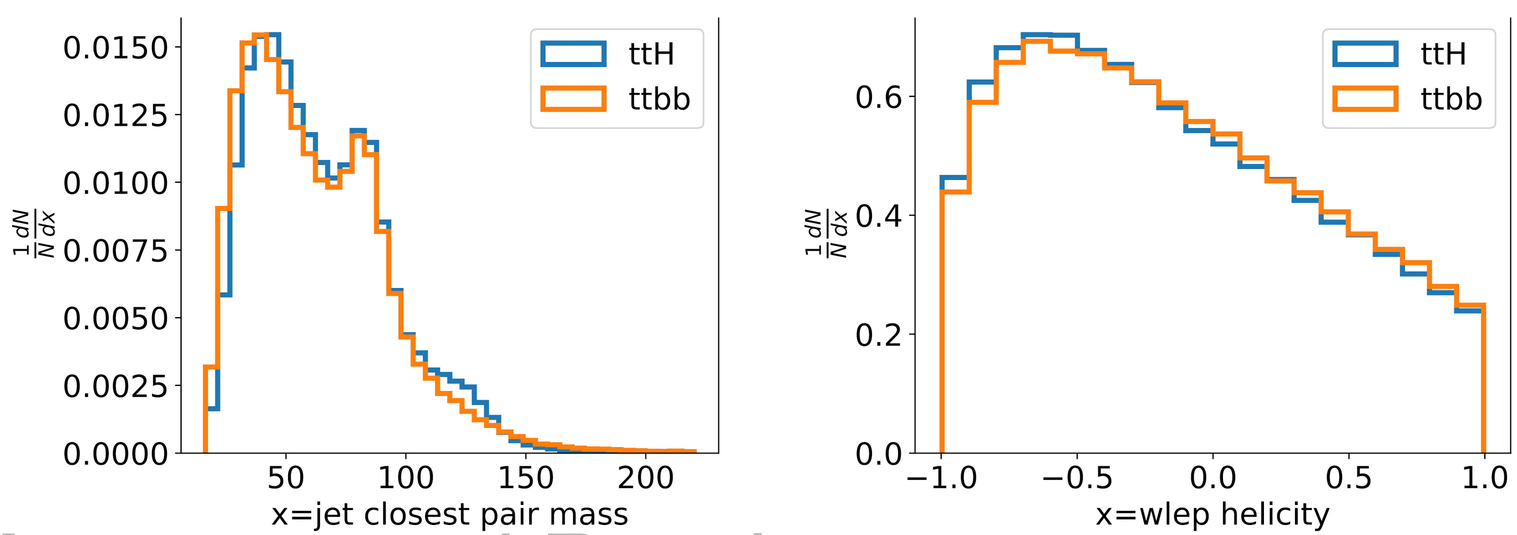
- Critic loss: $-C(x) + C(\hat{x}) + \kappa \cdot GP(C', x, \hat{x})$
- Generator loss: $-C(G(z))$
- Gradient penalty: $GP = \mathbb{E}_{\hat{u} \in \langle x, \hat{x} \rangle} \left(\| C'(\hat{u}) \|_2 - 1 \right)^2$

Formulating Benchmark on ttH data

- Generative Adversarial Networks



- Formulating Benchmark on ttH data



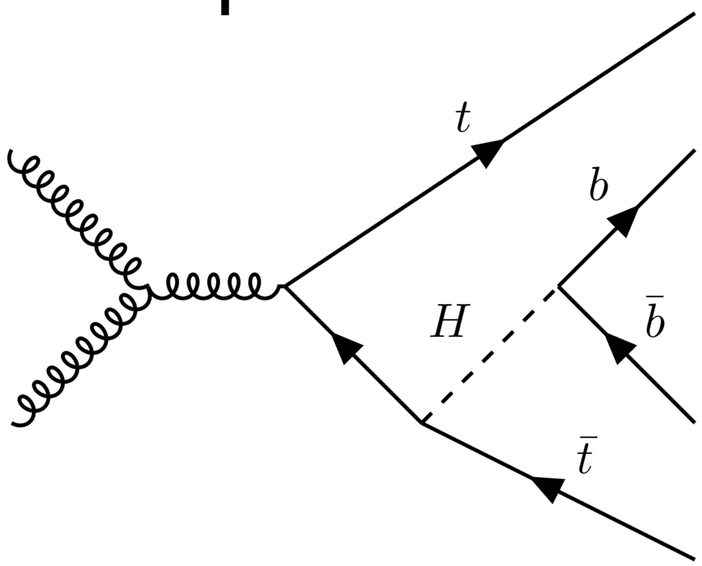
- Quality Measures and Results

David Schmidt
david.schmidt@rwth-aachen.de

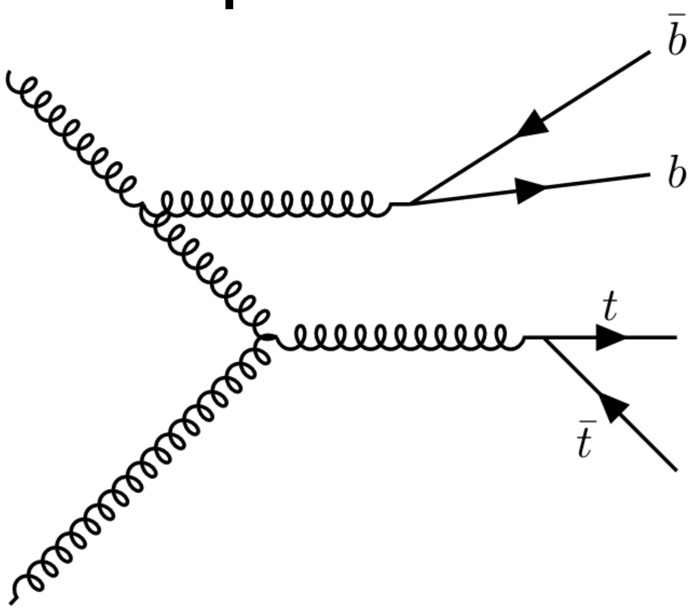
ttH/ttbb: 26 high-level, 32 low-level observables

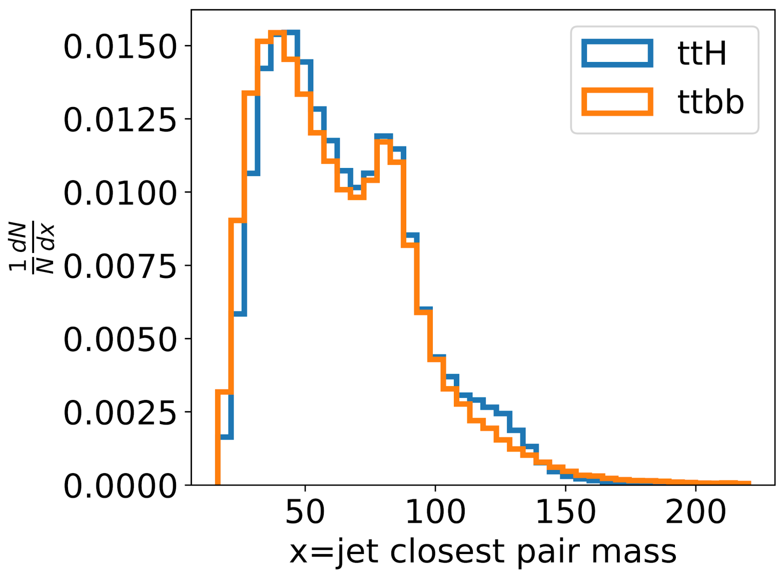
Pythia & Delphes Simulations

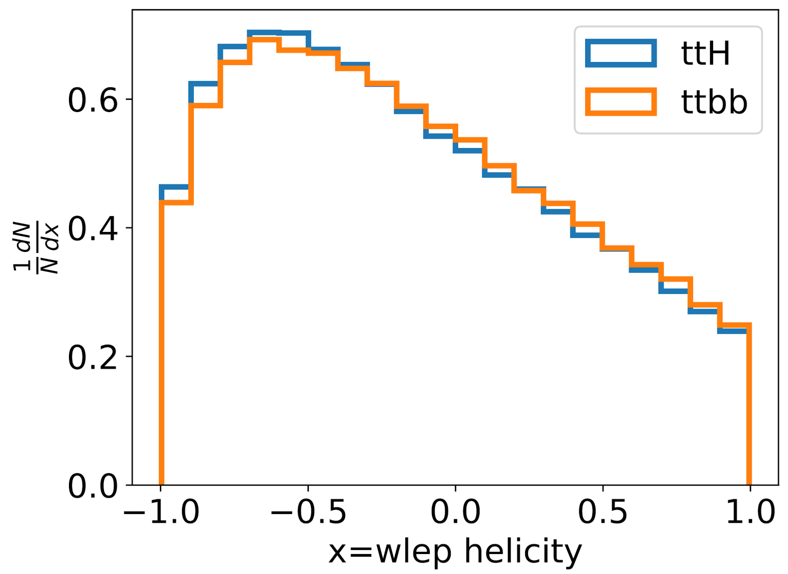
Signal:
ttH production

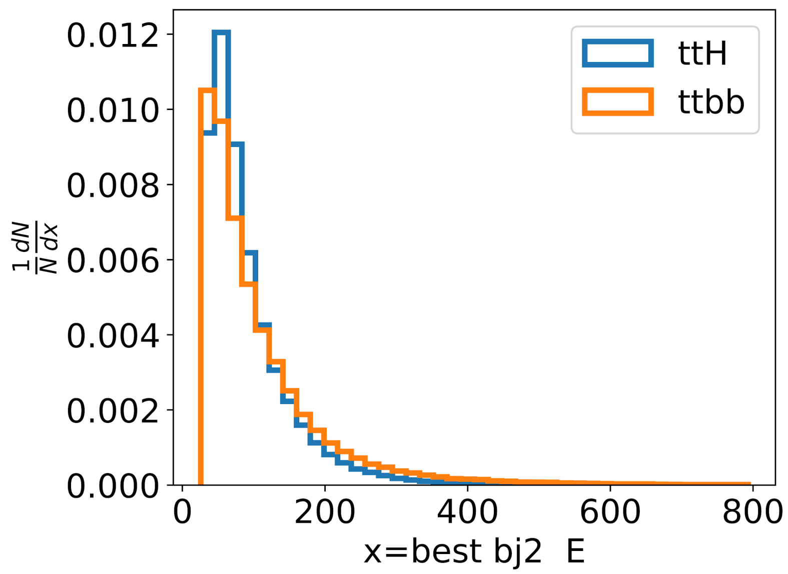


Background:
ttbb production





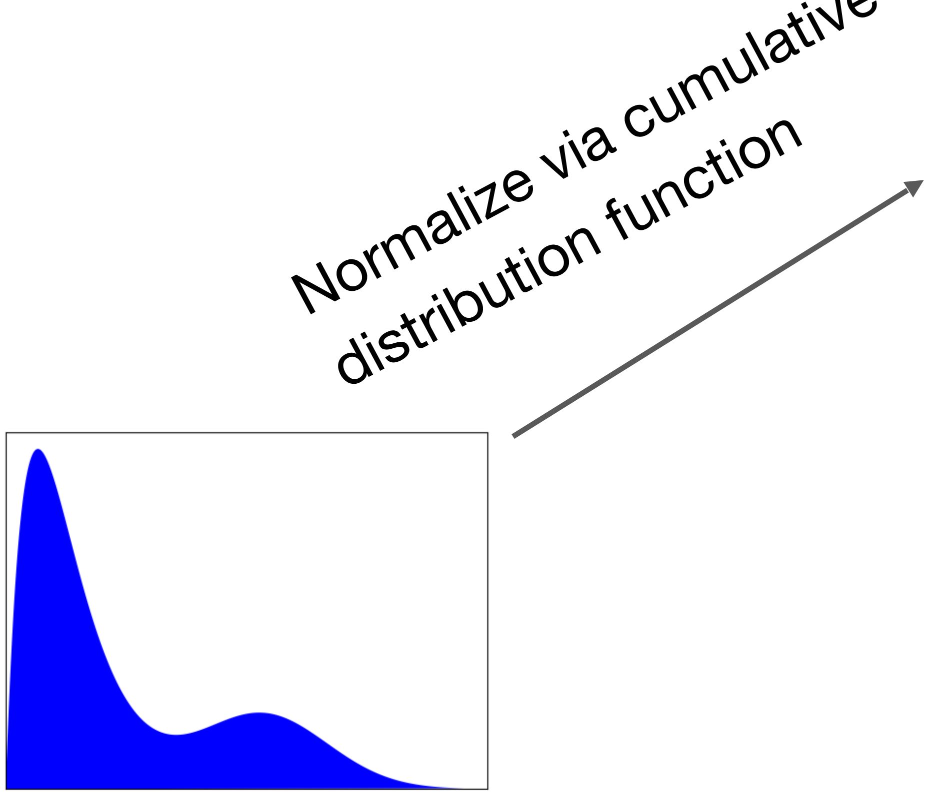


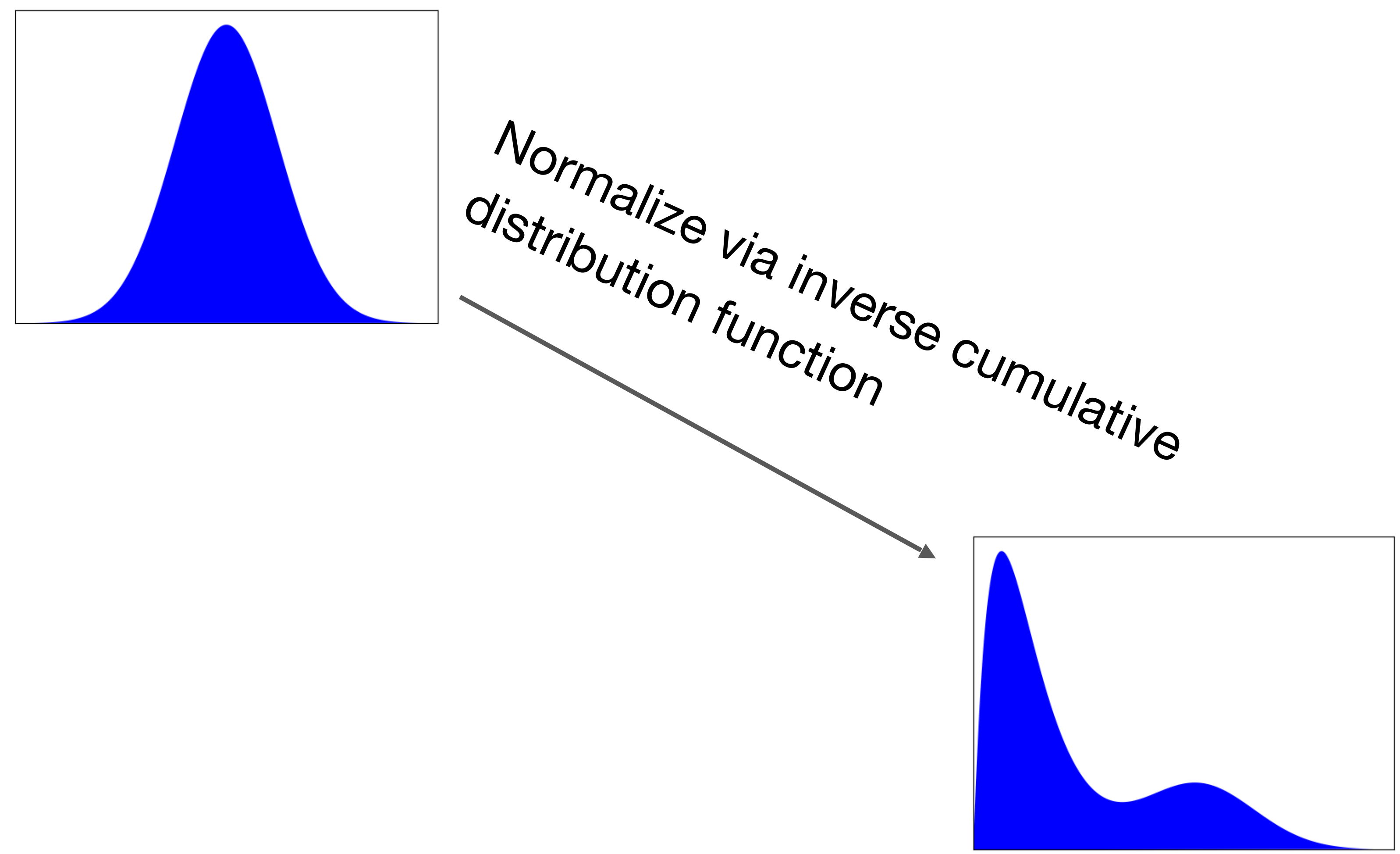


Benchmark dataset

WGAN:
- Train with Gauss-normalized data
- Create correlated gaussian output
- Apply histogram transformation

Dataset with destroyed correlations:
- Sample from Gaussian
- Transform to variable histograms

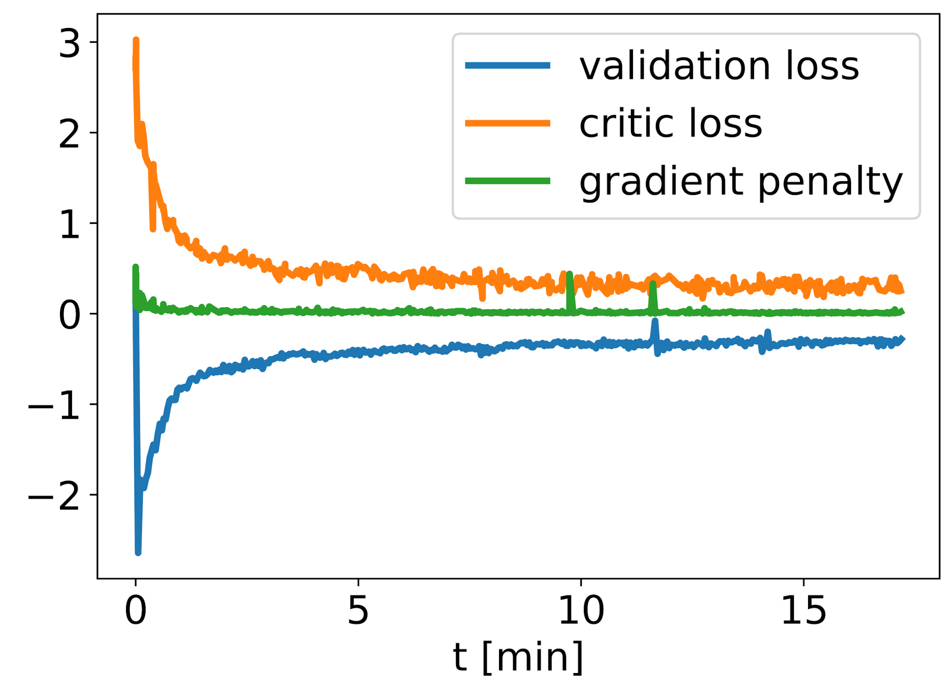


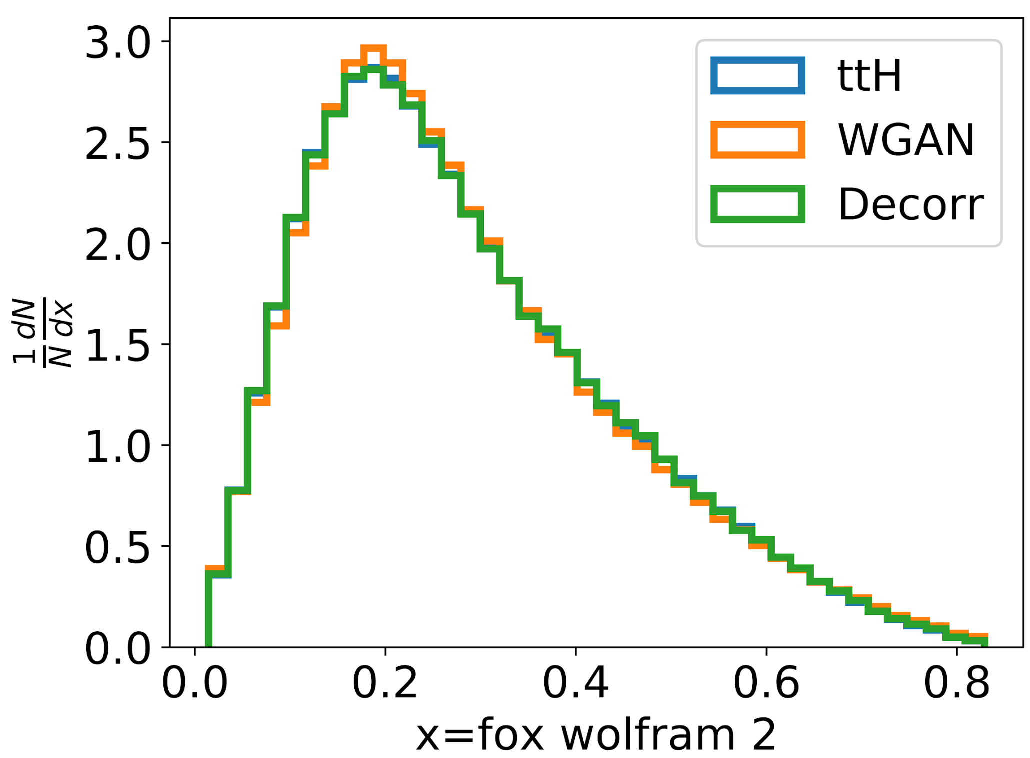


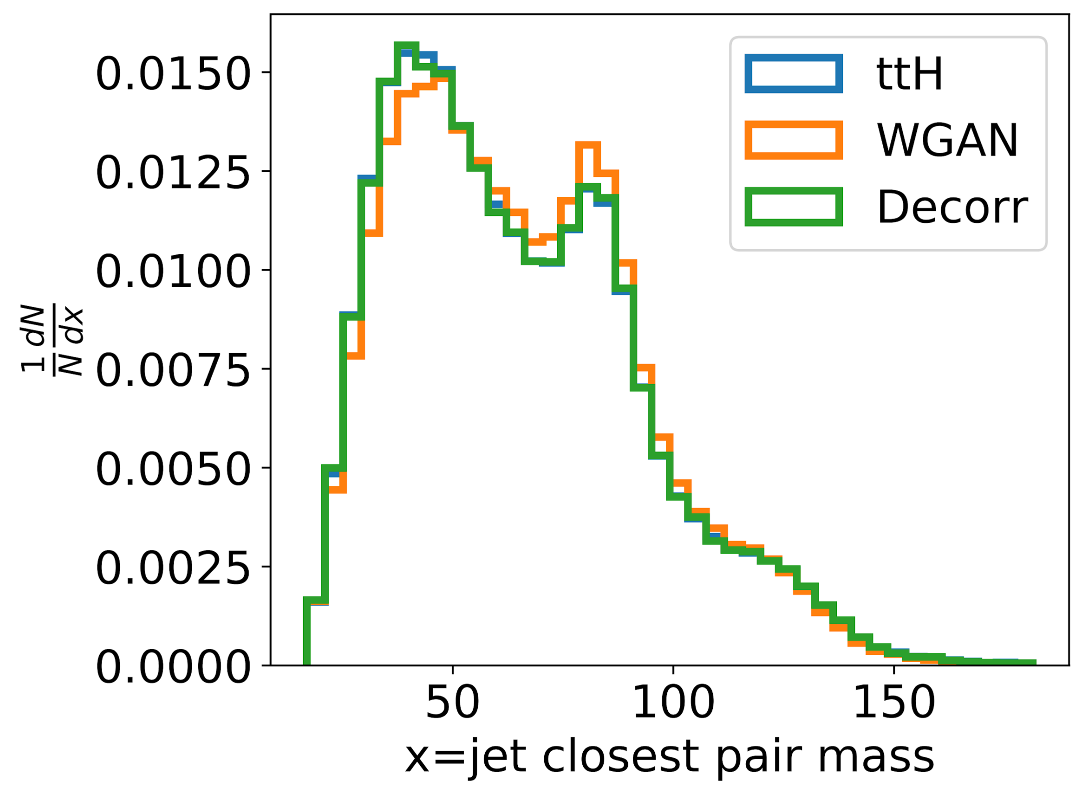
WGAN production

Generator & Critic network with same architecture:

- Feed-forward
- ReLU activation in each layer
- 6 hidden layers
- 288 nodes per hidden layer

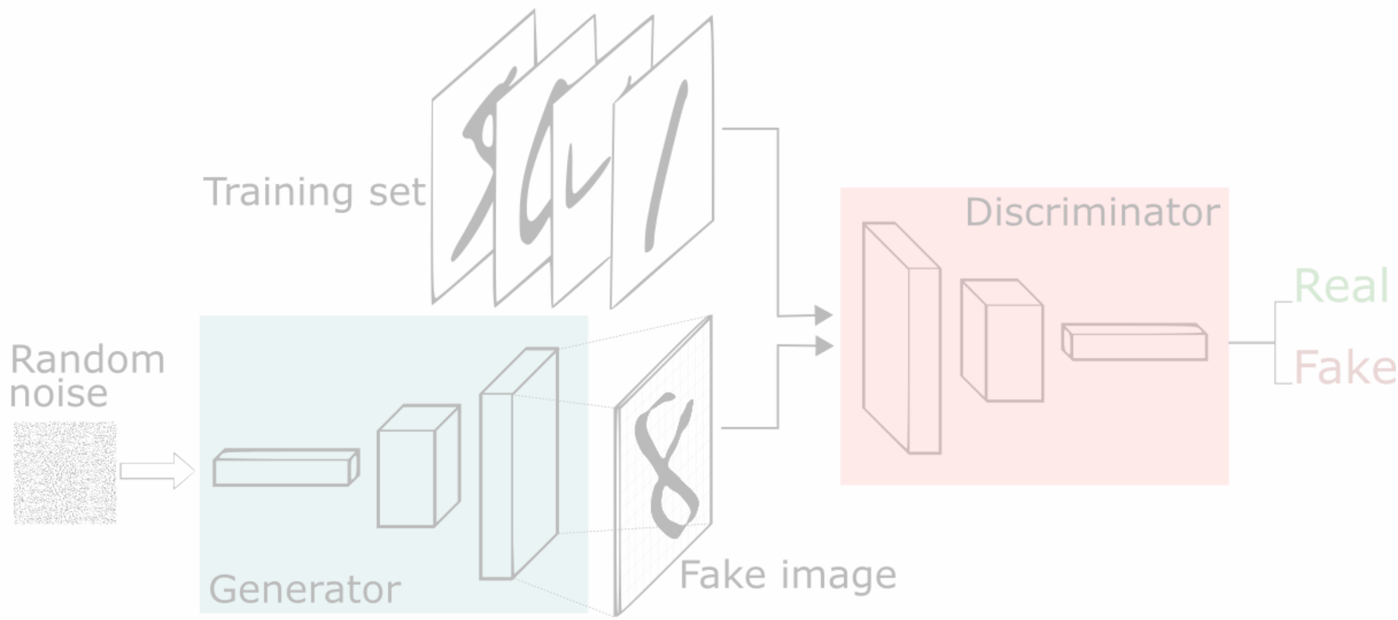




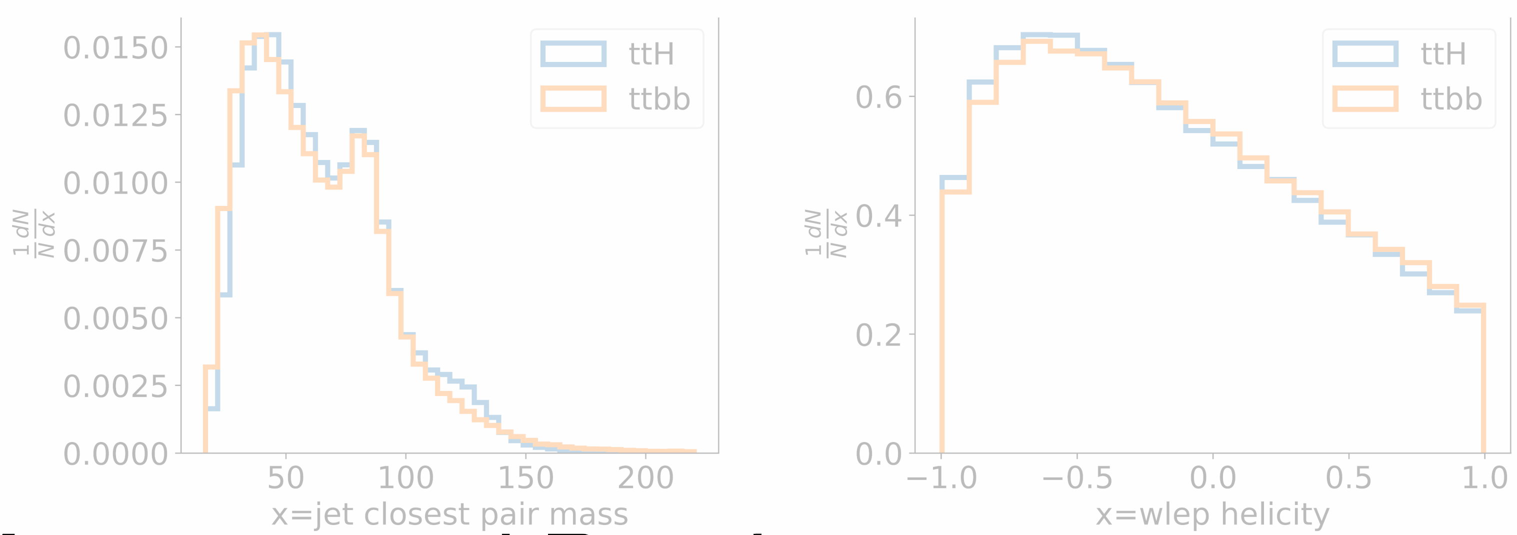


Quality Measures and Results

- Generative Adversarial Networks



- Formulating Benchmark on ttH data



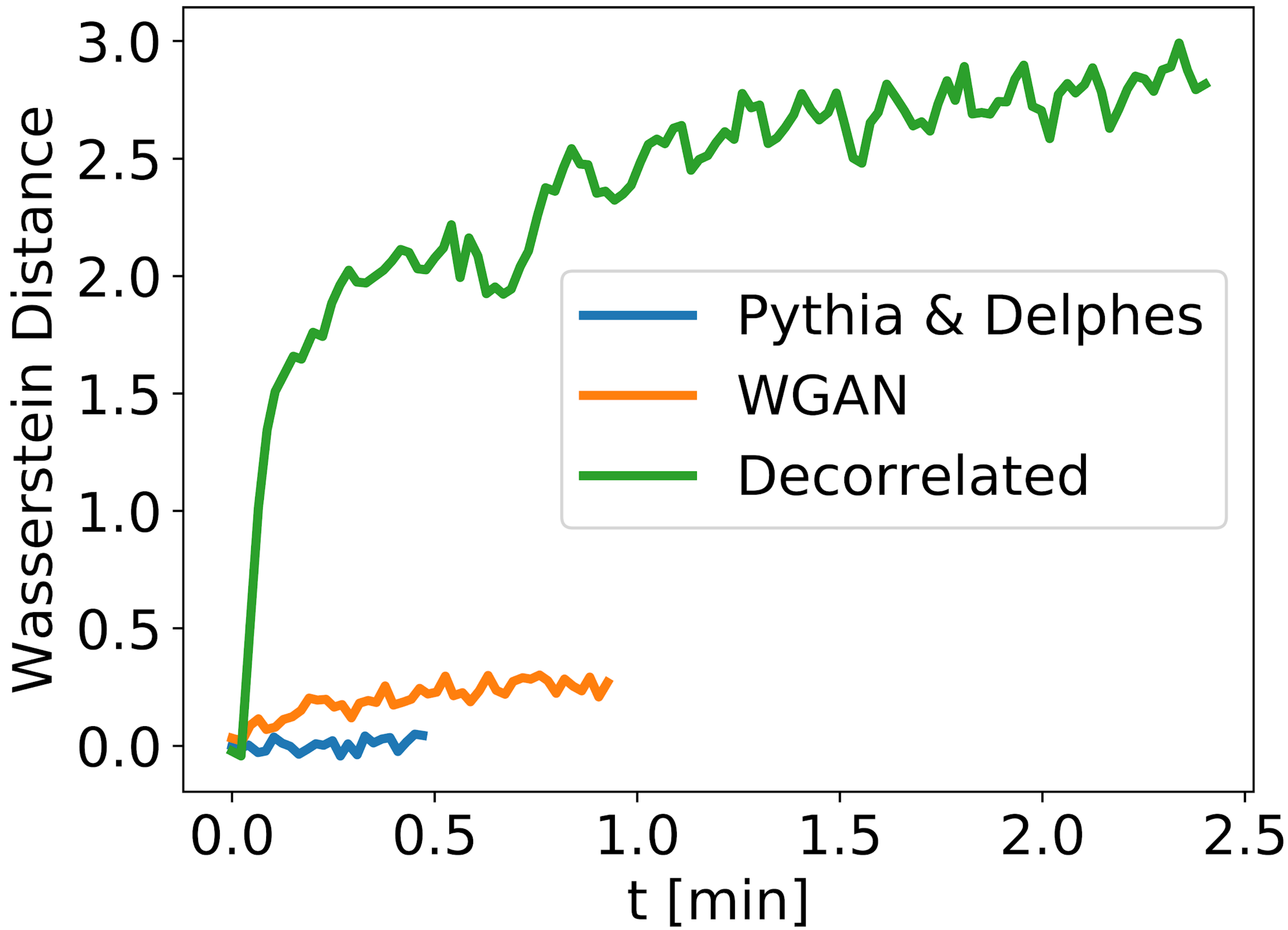
- **Quality Measures and Results**

III. Physikalisches Institut

RWTH AACHEN UNIVERSITY

Wasserstein Distance

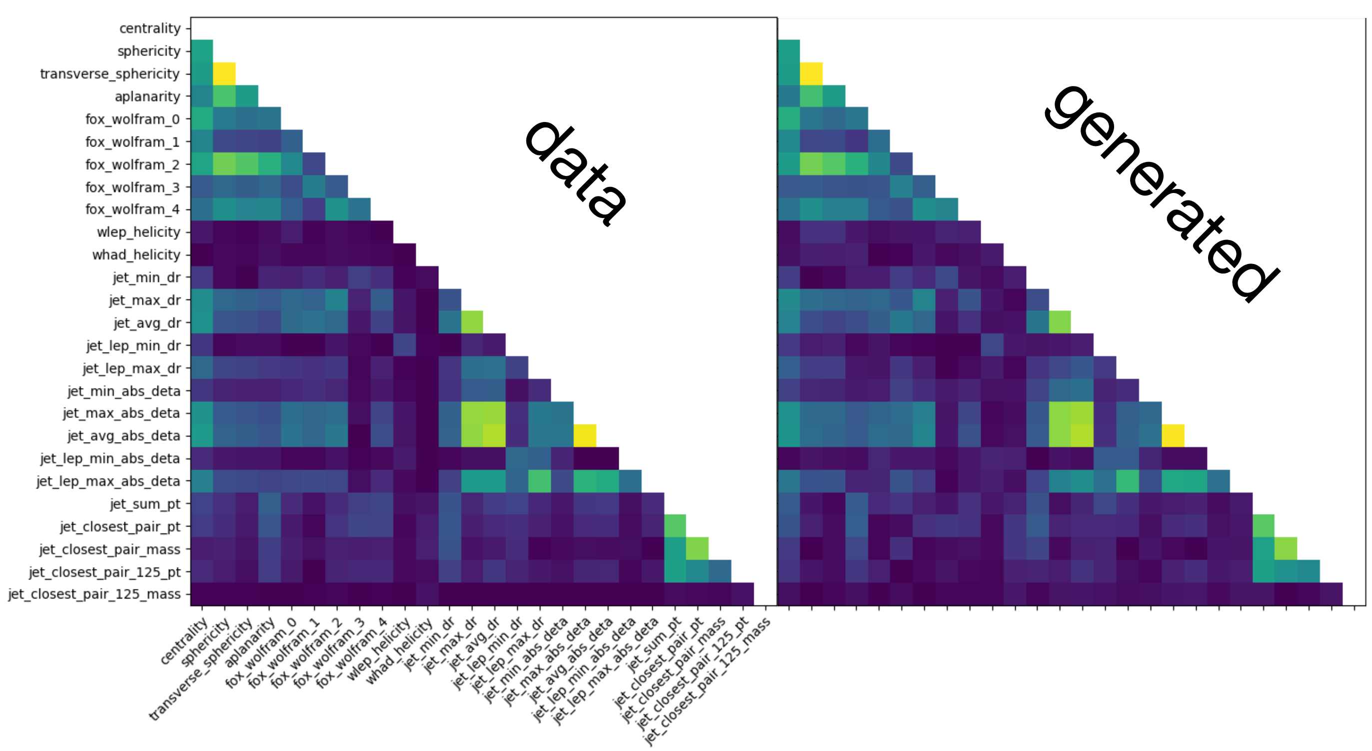
- Only train critic on dataset
- Converged critic loss gives estimate of the Wasserstein Distance

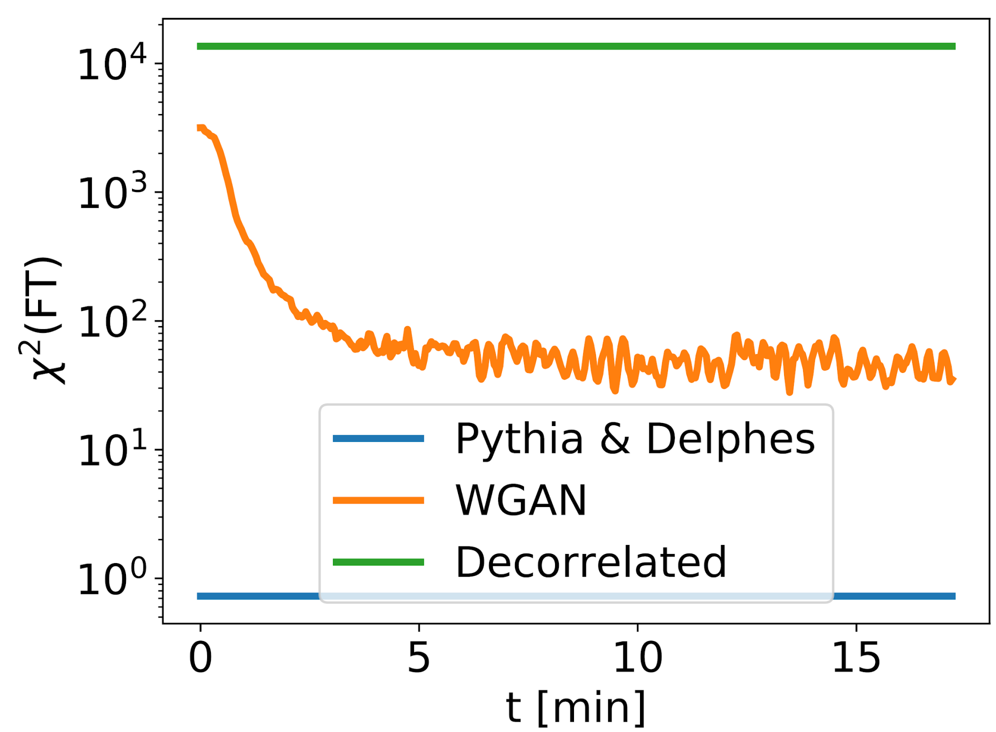


David Schmidt
david.schmidt@rwth-aachen.de

Correlation measure with the Fisher transformation

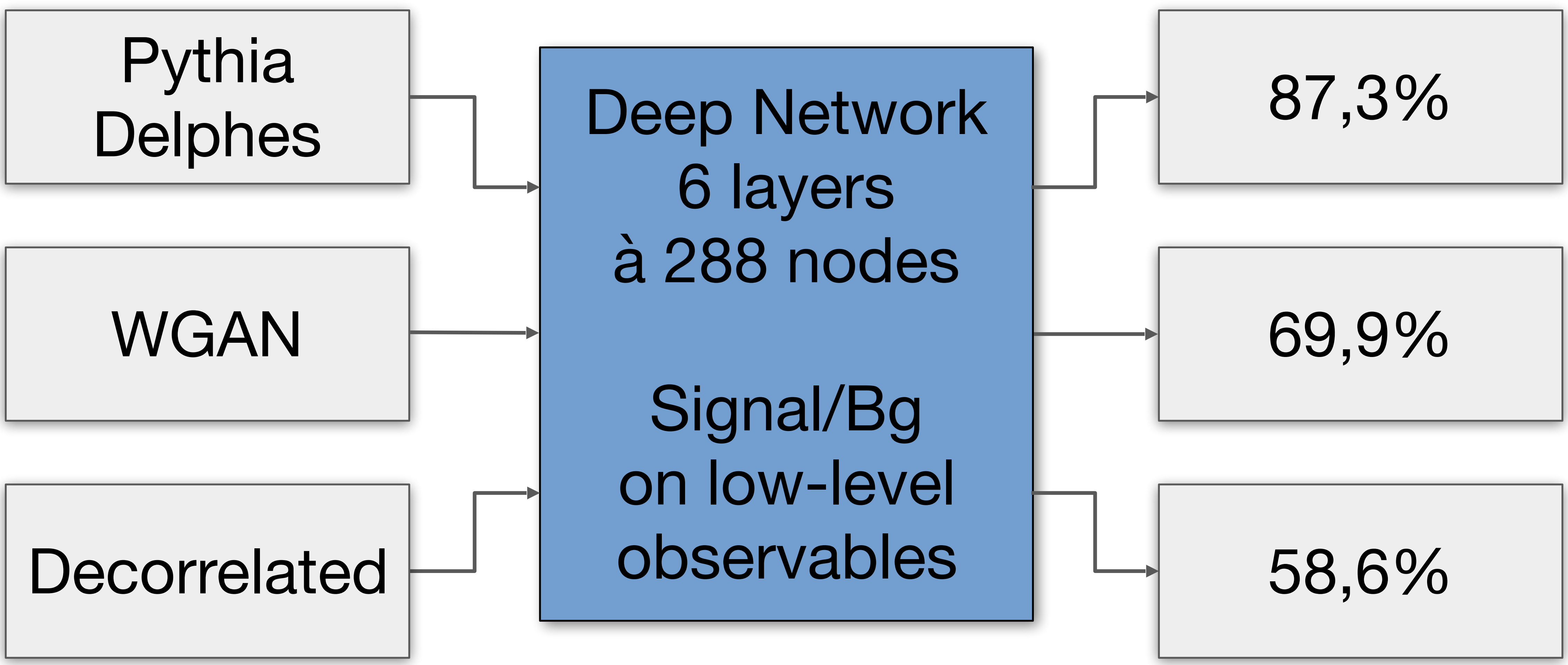
- Two distributions A, B: different correlations for x, y
- Fisher transformation: $z = arctanh(r_{x_i,y_i}), \quad z \sim \mathcal{N}\left(\rho_{x,y}, \frac{1}{\sqrt{N-3}}\right)$
- A, B equal: $\frac{z_A - z_B}{\sqrt{\frac{1}{N_A-3} + \frac{1}{N_B-3}}} \sim \mathcal{N}(0,1)$
- Different A, B lead to bigger absolute values
- Apply χ^2 measure across all $\frac{26 \cdot 25}{2}$ correlation values





Classification Benchmark: ttH vs ttbb

Validation performed on
Pythia & Delphes



→ WGAN is able to capture correlations,
still work to do!

Summary - High-level variable generation

- Training of improved WGANs easier than GANs
  - Change loss from cross-entropy to critic loss
  - Gradient penalty ensuring validity of Kantorovich-Rubinstein duality.
  - Wasserstein distance encoded in loss function makes supervision of training easier

- Quality measures for complex non-image datasets: Fisher transformation, Wasserstein distance, Classifier benchmark

- Generating high-level variables bypassing event generation, detector simulation and variable algorithms enables large speed-ups of orders of magnitude

***Conditional* Wasserstein GANs** for fast simulation of electromagnetic showers in a **CMS HGCAL prototype**

Calorimeter “simulation” with generative models

- Computationally expensive: simulation of particles interacting with material.

Geant 4

- electromagnetic & hadronic physics, lists with increasing/decreasing accuracy.

- Grand goal: replace simulation steps by *ultra* ***fast, accurate*** generative methods.

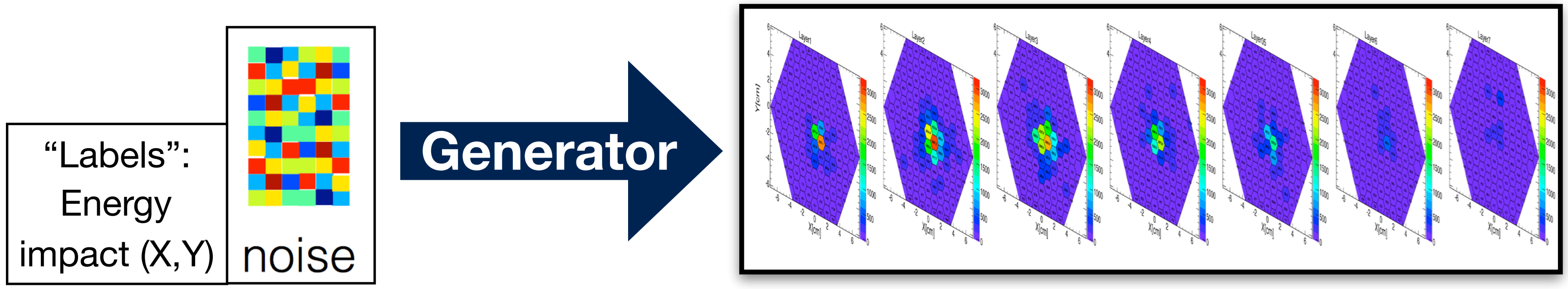
➡ **Step 1: Focus on simulation of particles showers in calorimeters.**

Proof-of-principle already demonstrated:

- e.g. at the 1st IML workshop in 2017 by L. Oliveira, M. Paganini and B. Nachman.
- or *arXiv:1701.05927v2, arXiv:1705.02355v2, arXiv:1711.08813v1, S. Vallecorsa @ ACAT2017, arXiv:1802.03325v1 , …*

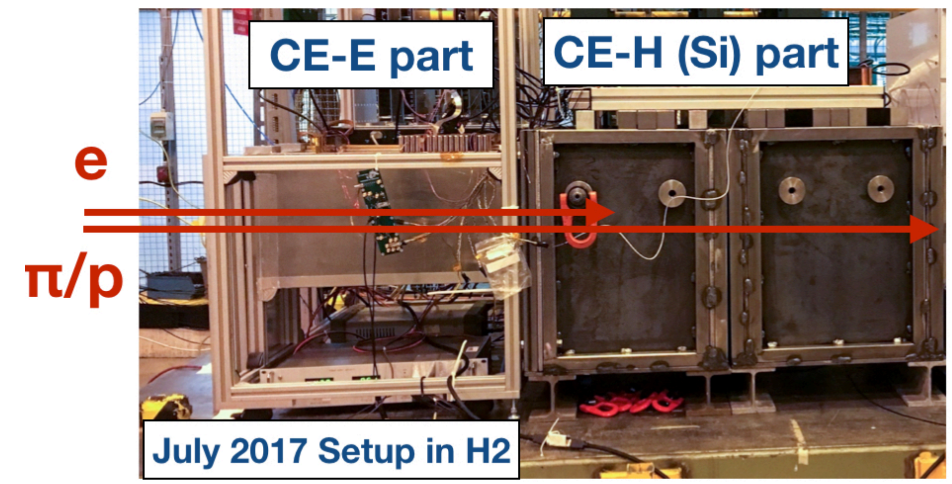
Implemented novelties in this study

We want:



<u>Differences to previous studies:</u>

1. Real-life **CMS HGCAL prototype** calorimeter.



2. **Conditional** generation with **three “labels”**: incident particle’s energy, impact position (X,Y).

3. **Wasserstein GAN**, i.e. *Earth Mover* distance to train the generator.

➡ “Discriminator" —> “Critic”.

$$L = \underbrace{\underset{\tilde{\boldsymbol{x}} \sim \mathbb{P}_g}{\mathbb{E}} \left[D(\tilde{\boldsymbol{x}}) \right] - \underset{\boldsymbol{x} \sim \mathbb{P}_r}{\mathbb{E}} \left[D(\boldsymbol{x}) \right]}_{\text{Original critic loss}} + \underbrace{\lambda \underset{\hat{\boldsymbol{x}} \sim \mathbb{P}_{\hat{\boldsymbol{x}}}}{\mathbb{E}} \left[(\|\nabla_{\hat{\boldsymbol{x}}} D(\hat{\boldsymbol{x}})\|_2 - 1)^2 \right]}_{\text{Our gradient penalty}}.$$

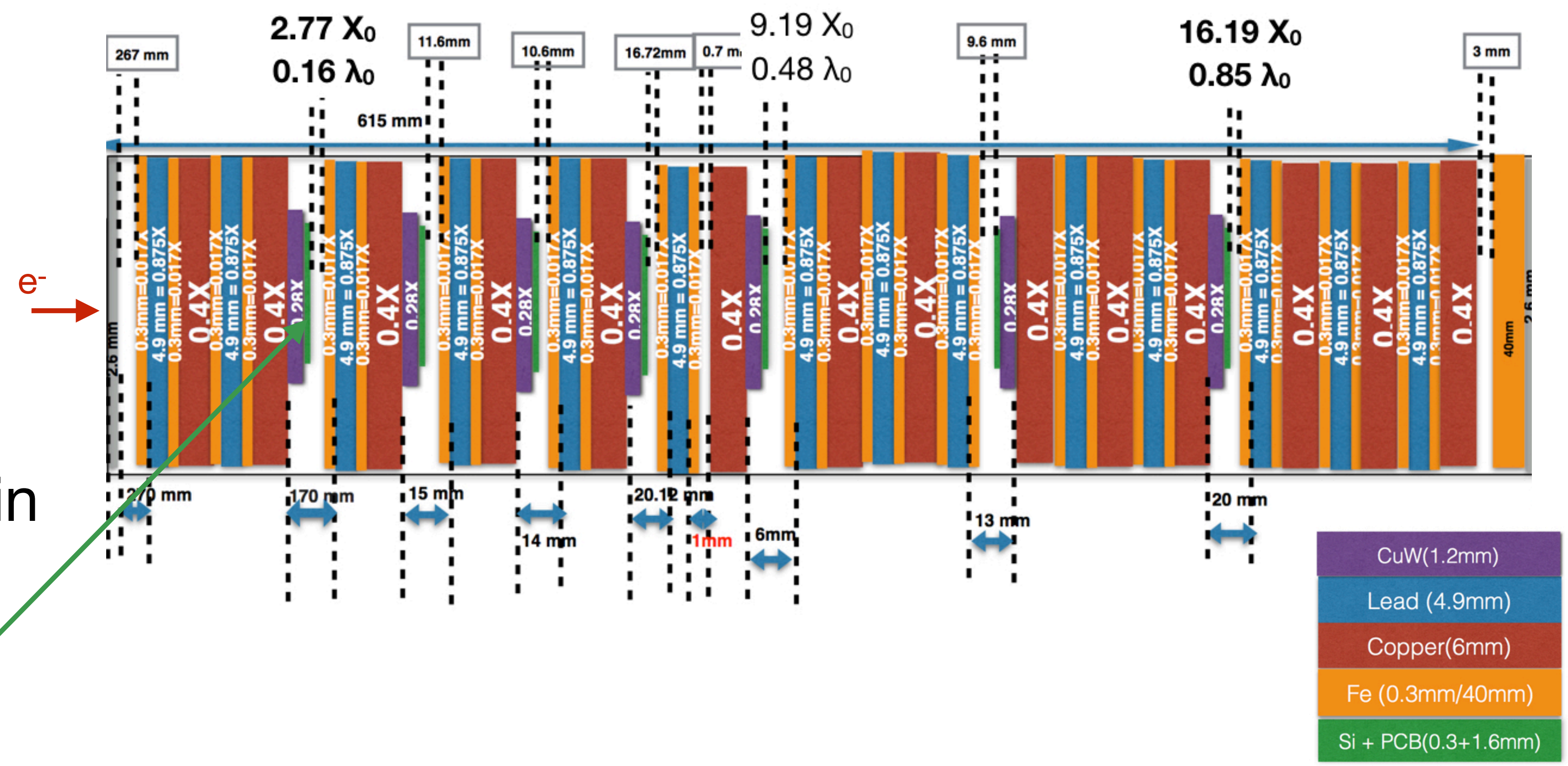
arXiv:1704.00028v3

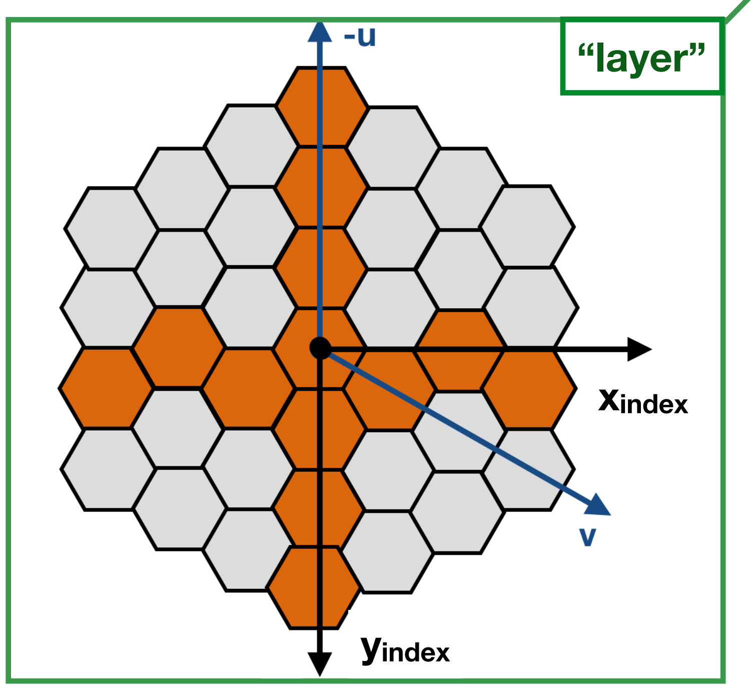
HGCAL prototype in September 2017

Features:

- **Sampling calorimeter.**
- **7** sensitive silicon **layers**.
- **2.7 - 16.2 X_0** in depth.

- **Hexagonal** pixels with ~1.2cm in diameter (128 pixels per layer).





Above: Mapping of hexagonal geometries into cartesian coordinates.

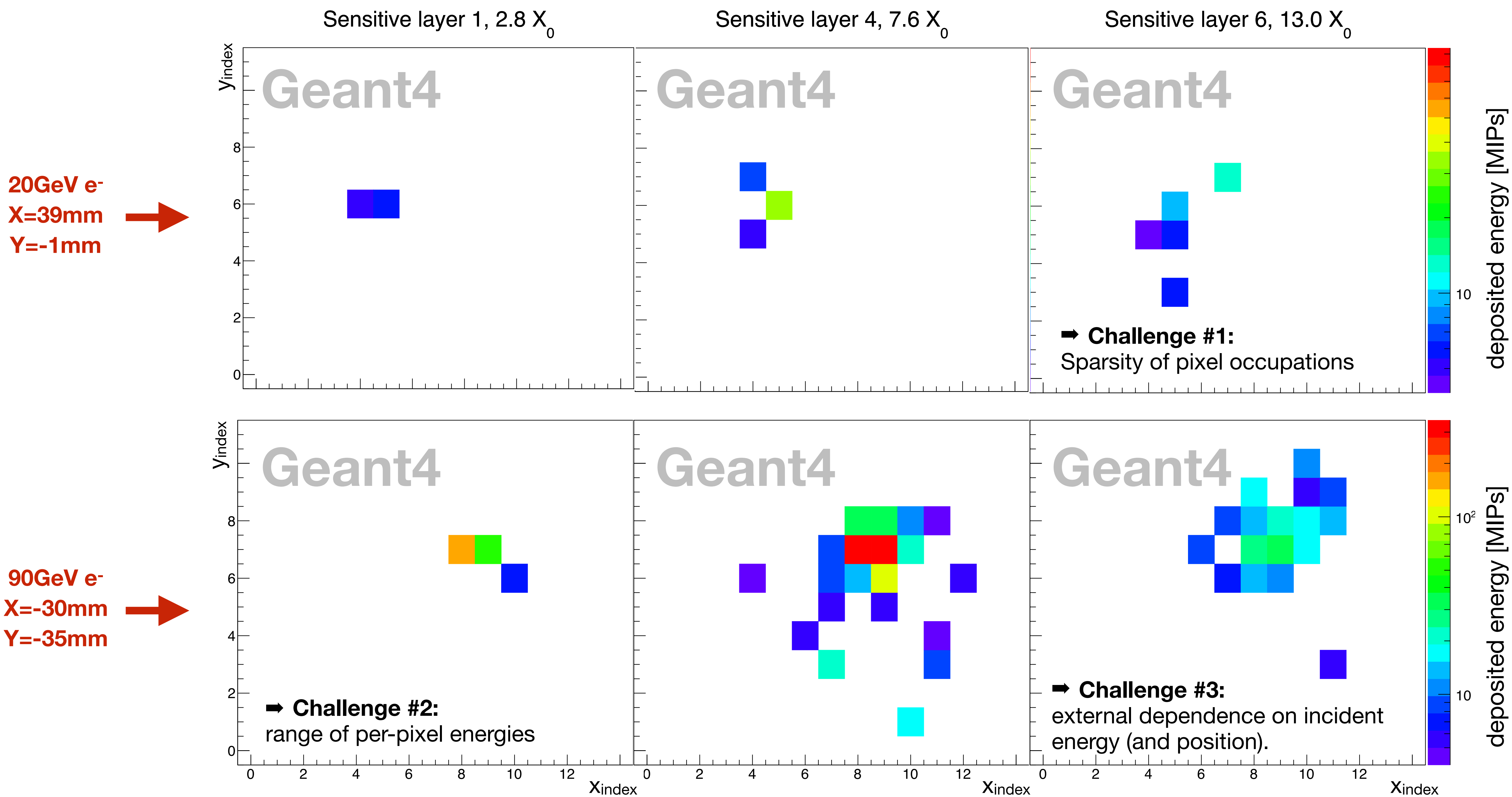
Prototype has been tested with beam…

… but the available statistics of electron showers is likely too low for training generative model.

➡Using Geant 4 simulated electron samples generated *with* beam test conditions.

Exemplary showers from 440k showers sample

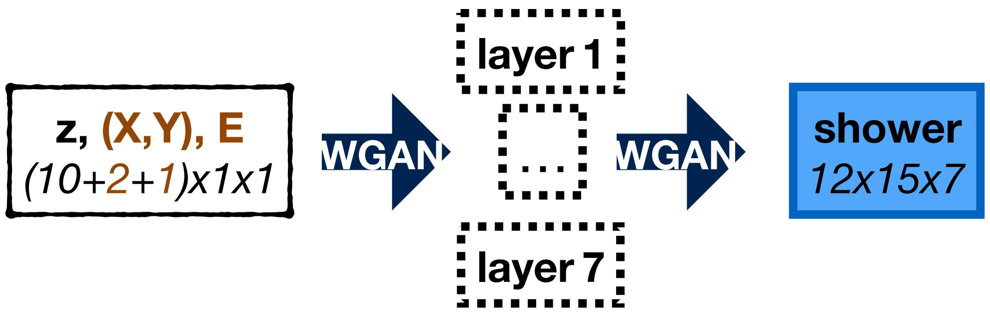
- **20, 32, 50, 80 & 90 GeV electrons** with 1% energy spread.
- **O(80k)** showers for training and O(10k) for cross-checks for **each energy** bin.



Thorben Quast
quast@physik.rwth-aachen.de

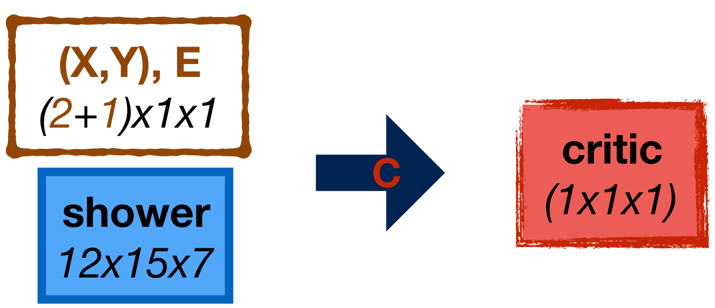
Training strategy using WGANs

- **Generator network (WGAN)** maps (noise, E_{fake}, $position_{fake}$) to fake showers.



  - Set of upsampling and convolutions.
  - Batch normalisation.
  - Leaky Relu activation functions except for last step.
  - 672k parameters to be trained.

- **Critic network (C)** estimates the ***Earth Mover*** distance btw. generated & real showers.



  - Labels as additional input.
  - Set of convolutions & fully connected layers.
  - Layer normalisation.
  - 477k parameters to be trained.

Figures of merit for training:

Critic loss:

$$c_{loss} = -\mathbf{C}(shower_{real}, E_{real}, pos._{real}) + \mathbf{C}(shower_{fake}, E_{fake}, pos._{fake}) + \lambda \times \textbf{gradient penalty},$$

$\lambda := 50$

Generator loss w.r.t. critic:

$$g_{loss,\,c} = -\,\mathbf{C}(shower_{fake}, E_{fake}, pos._{fake})$$

Training strategy to include the conditions, "labels"

- **2 auxiliary networks** for energy- (**E**) and position regression (**P**) on shower images.

<u>Energy regression network **E**</u>

shower *12x15x7* $\xrightarrow{E}$ **energy** *1x1x1*

- 54k trainable parameters.
- More details in the backup.

<u>Position regression network **P**</u>

shower *12x15x7* $\xrightarrow{P}$ **(X,Y)** *(2x1x1)*

- 19k trainable parameters.
- More details in the backup.

- **E** and **P** trained using "real" showers - no effect from generated "fake" showers.

<u>Energy and position regression losses:</u>

$$\mathbf{e}_{\mathbf{loss,\ real}} = -\ (\mathbf{E}(\mathrm{shower}_{\mathrm{real}}) - E_{\mathrm{real}})^2,\quad \mathbf{p}_{\mathbf{loss,\ real}} = -\ (\mathbf{P}(\mathrm{shower}_{\mathrm{real}}) - \mathrm{pos.}_{\mathrm{real}})^2$$

- **Generator is additionally trained to minimise the regression errors.**

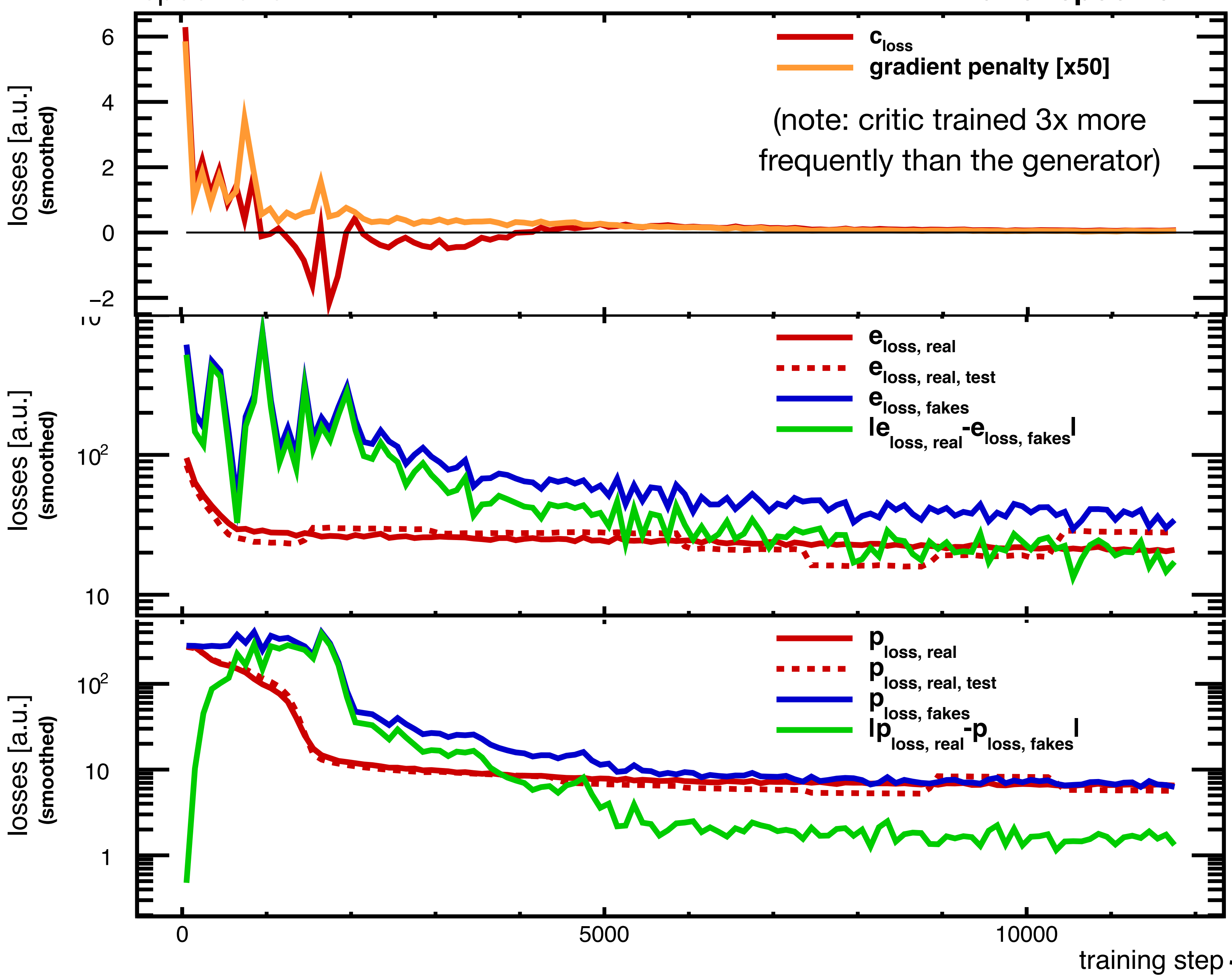
➡ <u>**Total generator loss combines generator related losses.**</u>

$$\mathbf{g}_{\mathbf{loss,\ tot}} = \mathbf{g}_{\mathbf{loss,\ c}} + K_e \times \left|\mathbf{e}_{\mathbf{loss,\ real}} - \mathbf{e}_{\mathbf{loss,\ fake}}\right| + K_p \times \left|\mathbf{p}_{\mathbf{loss,\ real}} - \mathbf{p}_{\mathbf{loss,\ fake}}\right|,$$

$$\kappa_e := \kappa_p := 0.01$$

System of networks trained within a few hours

Software: Tensorflow v1.5.
Hardware: NVIDIA GTX1080 GPU.



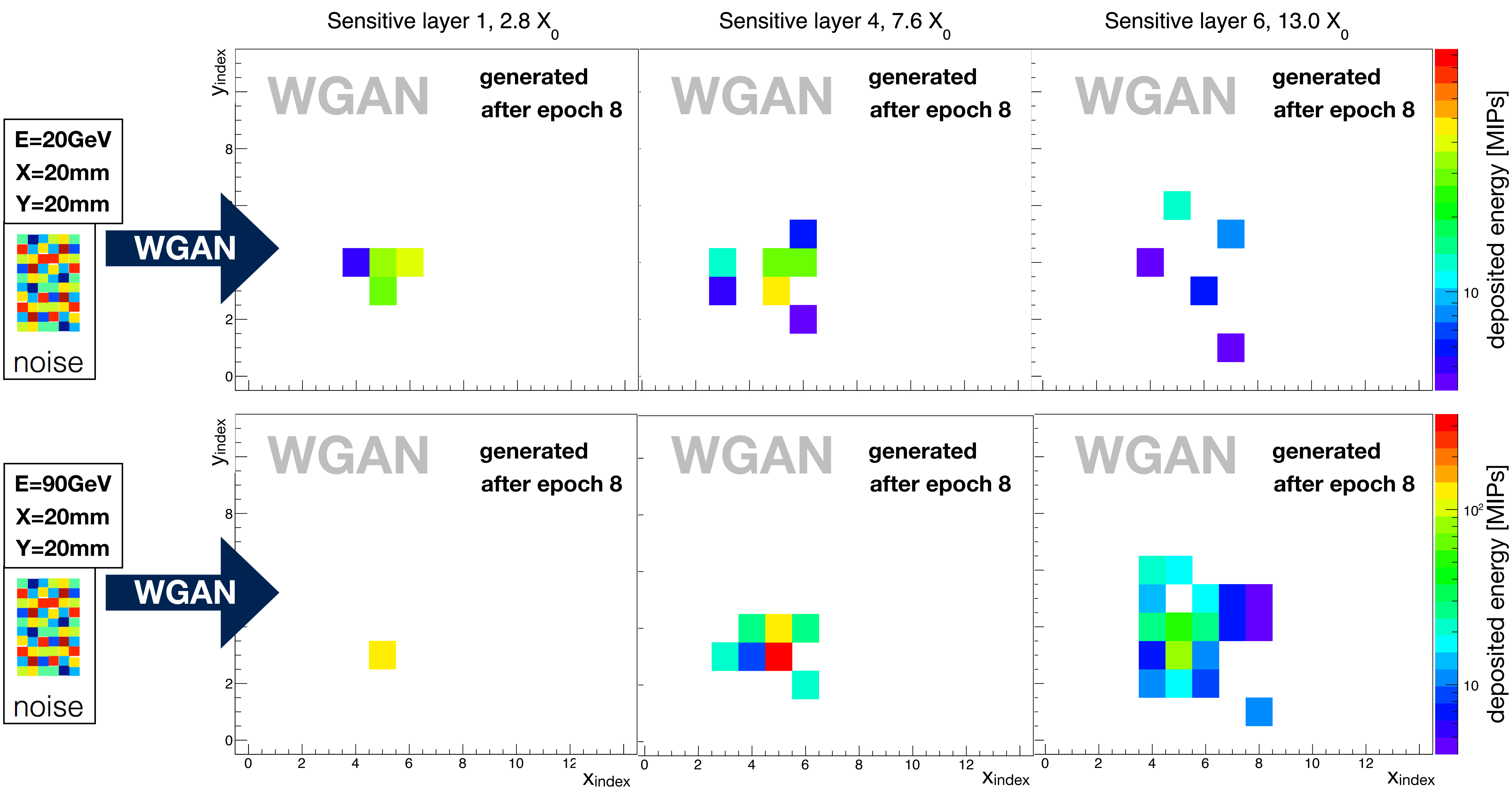
✓ Critic loss converging to 0.
- Dominated by the gradient penalty.

✓ Energy regression loss converging fast.
✓ Loss on generated images converging.
- Difference > 0.

✓ Position regression loss converging.
✓ Loss on generated images converging.

- 1 step —> 256 batch of showers.
- 1479 steps / epoch.

Generated electron showers look reasonable



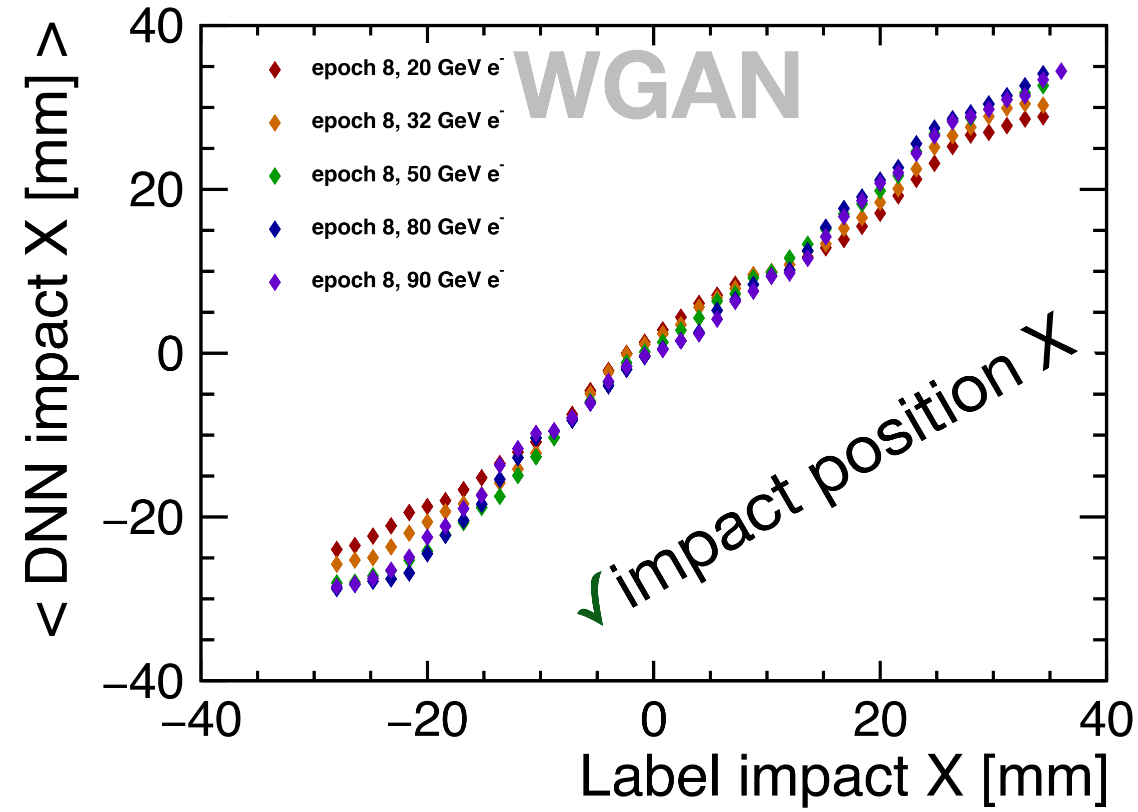
Side note:

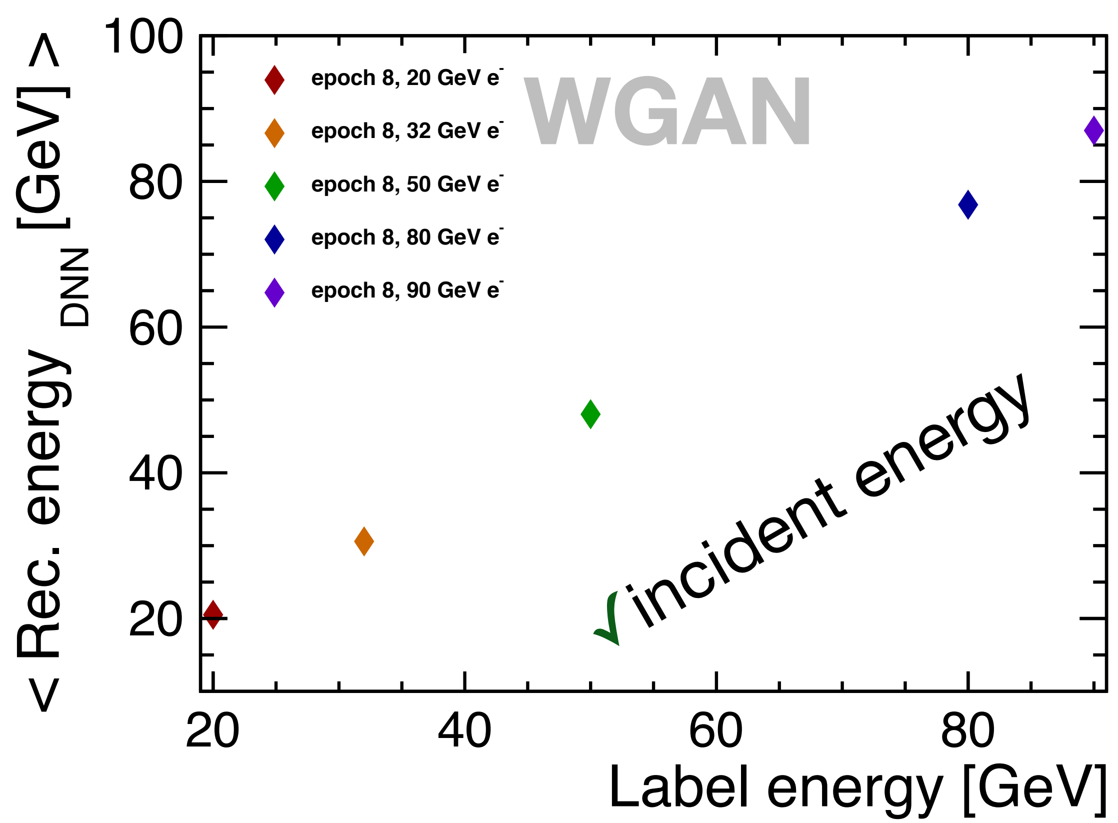
- Reasonable shower images already obtained **after 2 training epochs.**

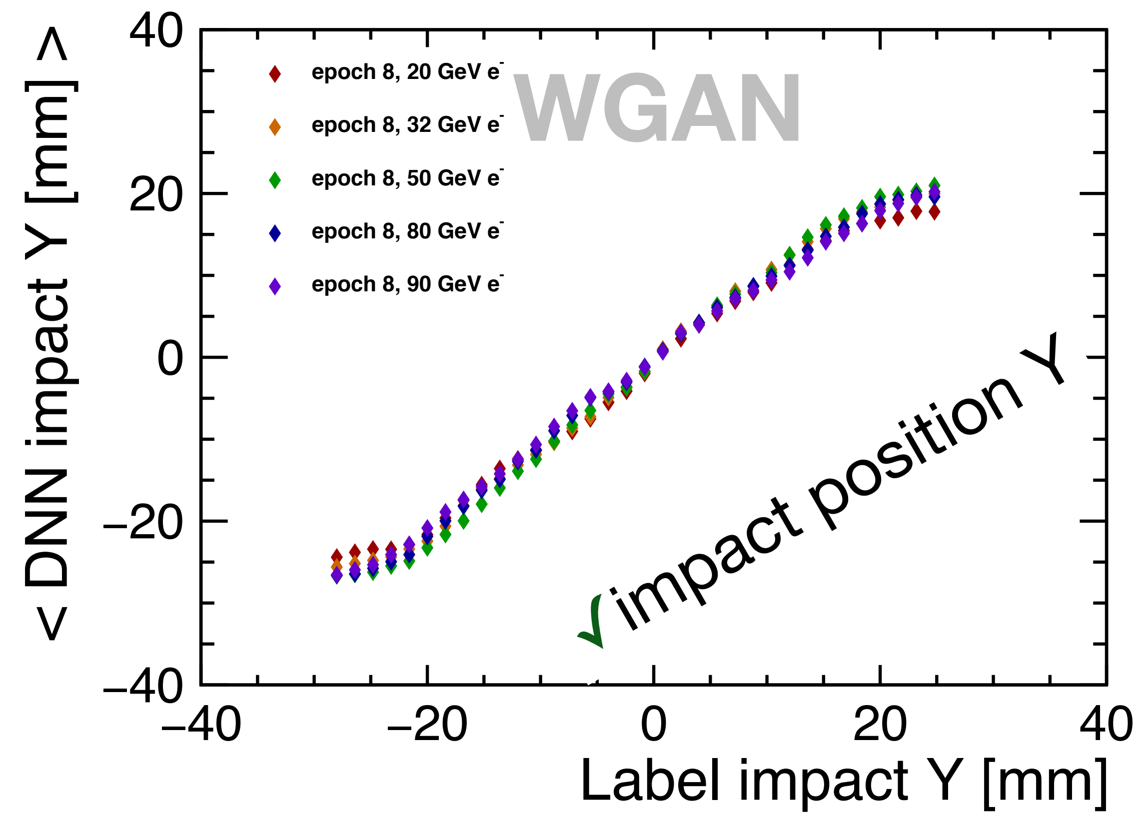
Generated events: Dependence on labels

If WGAN has learnt to respect labels:

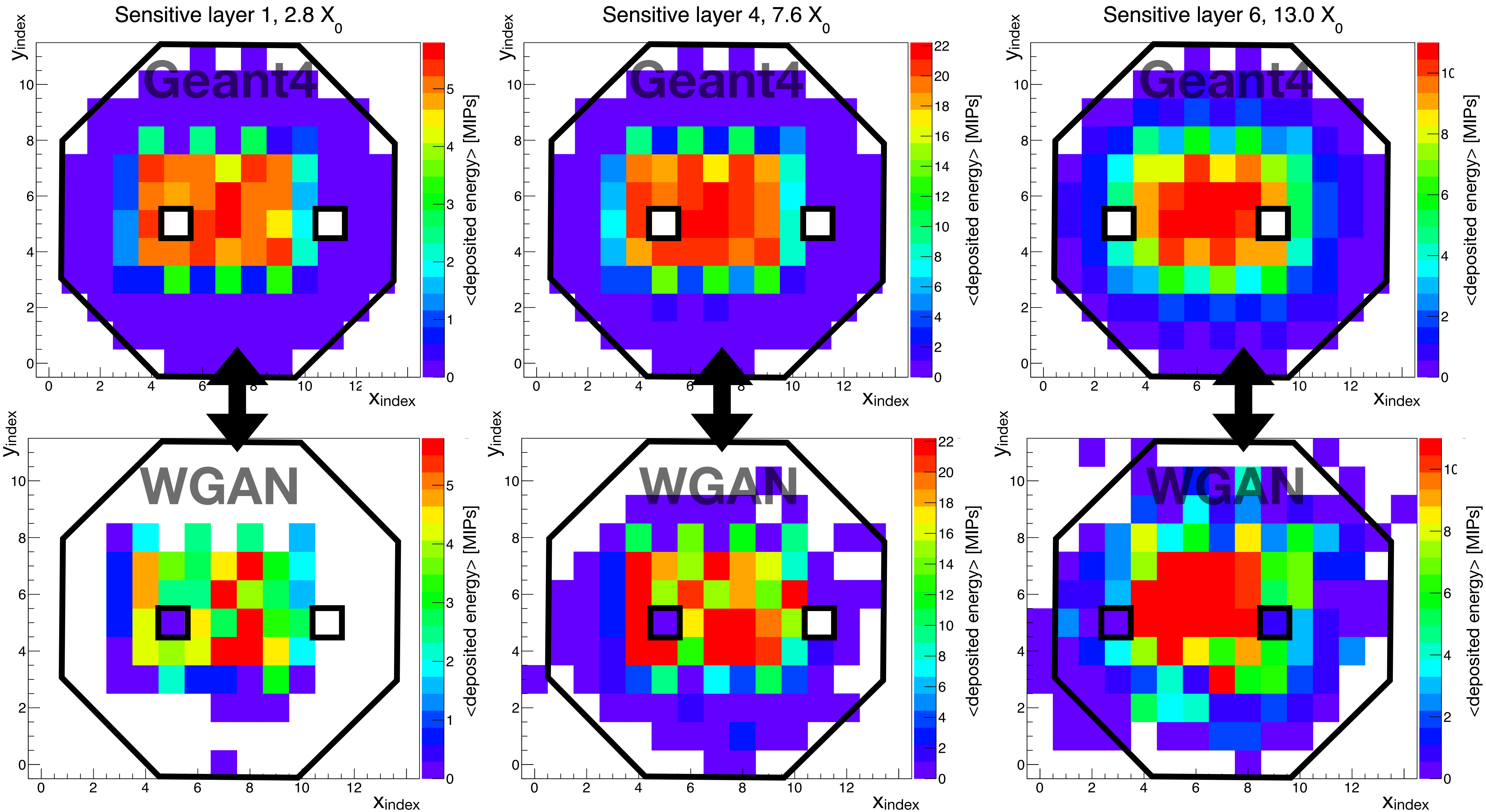
Reconstructed quantities of generated showers **correlate with** true label.







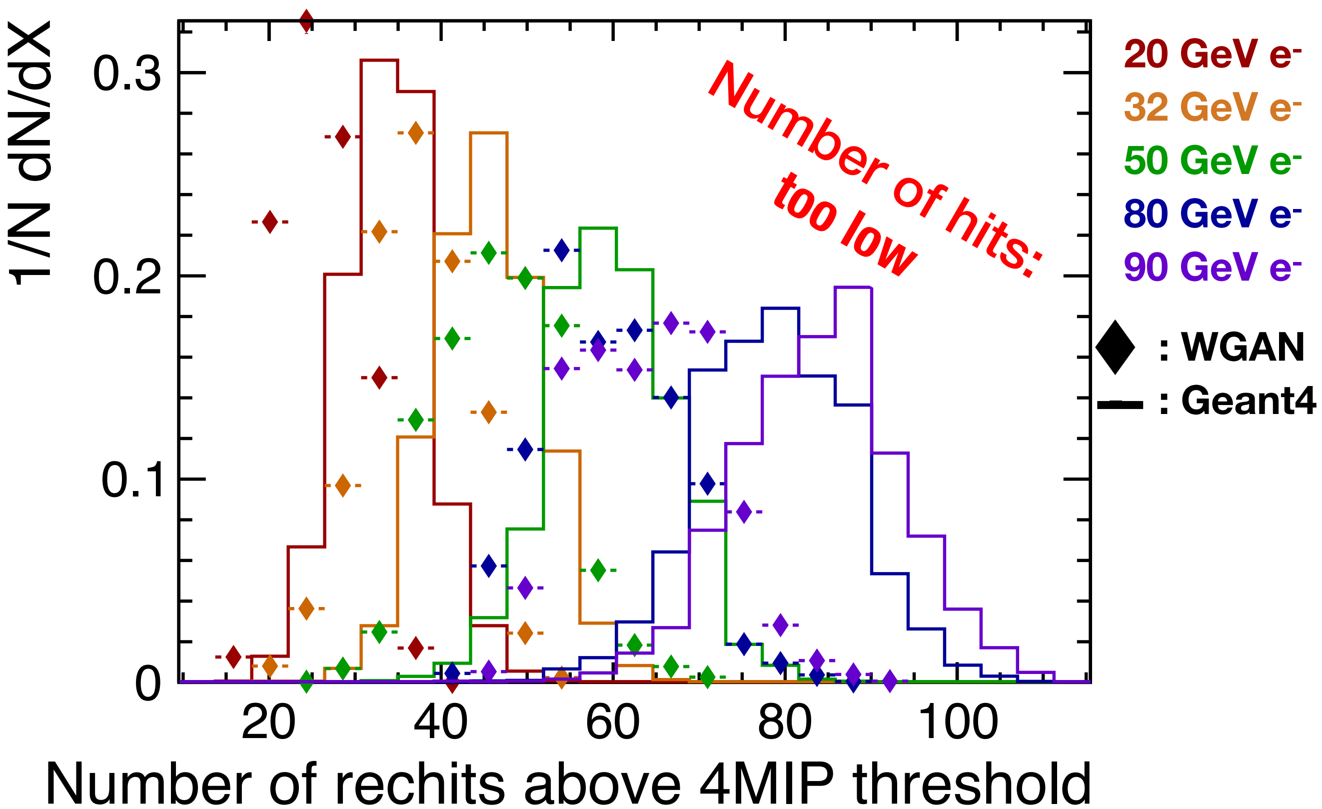
WGAN has learnt: Positions of dead pixels

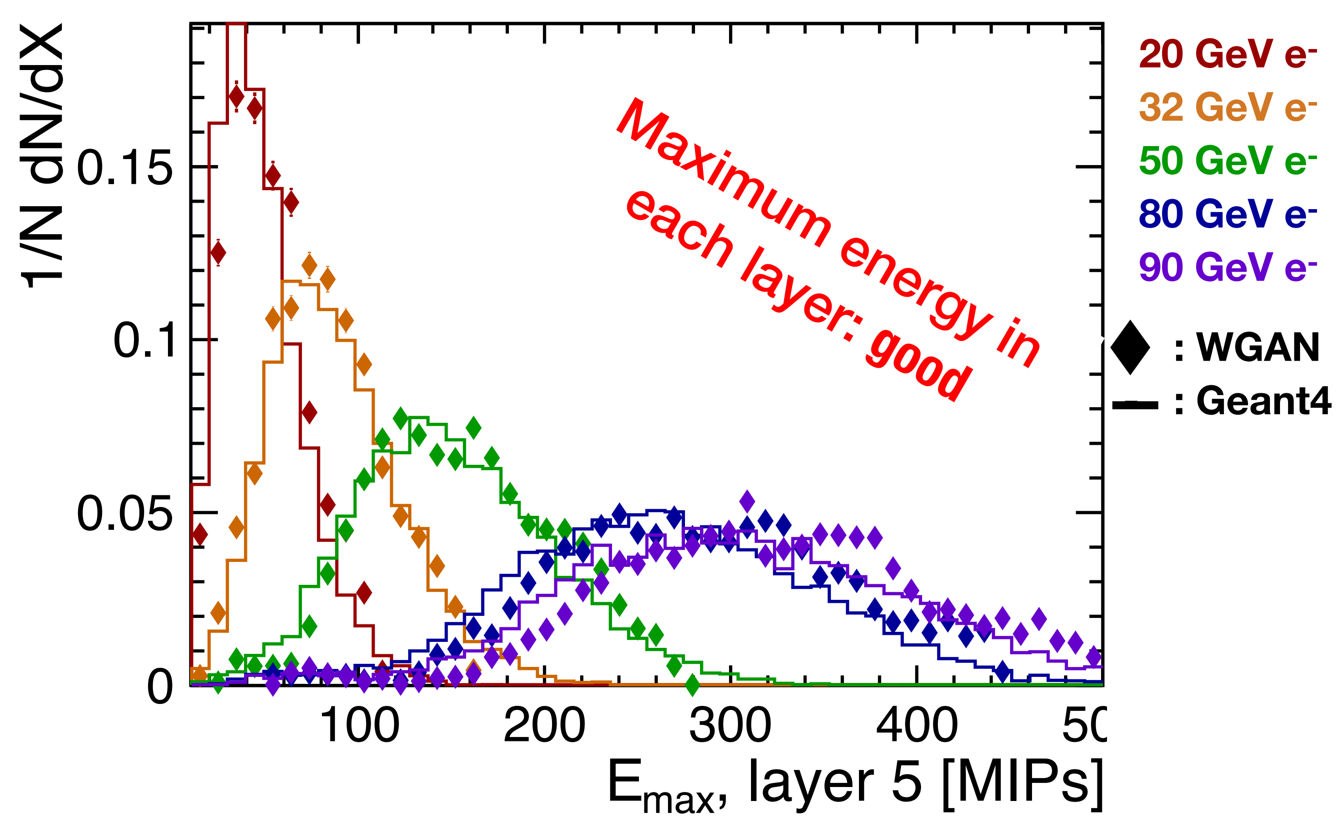


✓Unconnected pixels with low intensity.

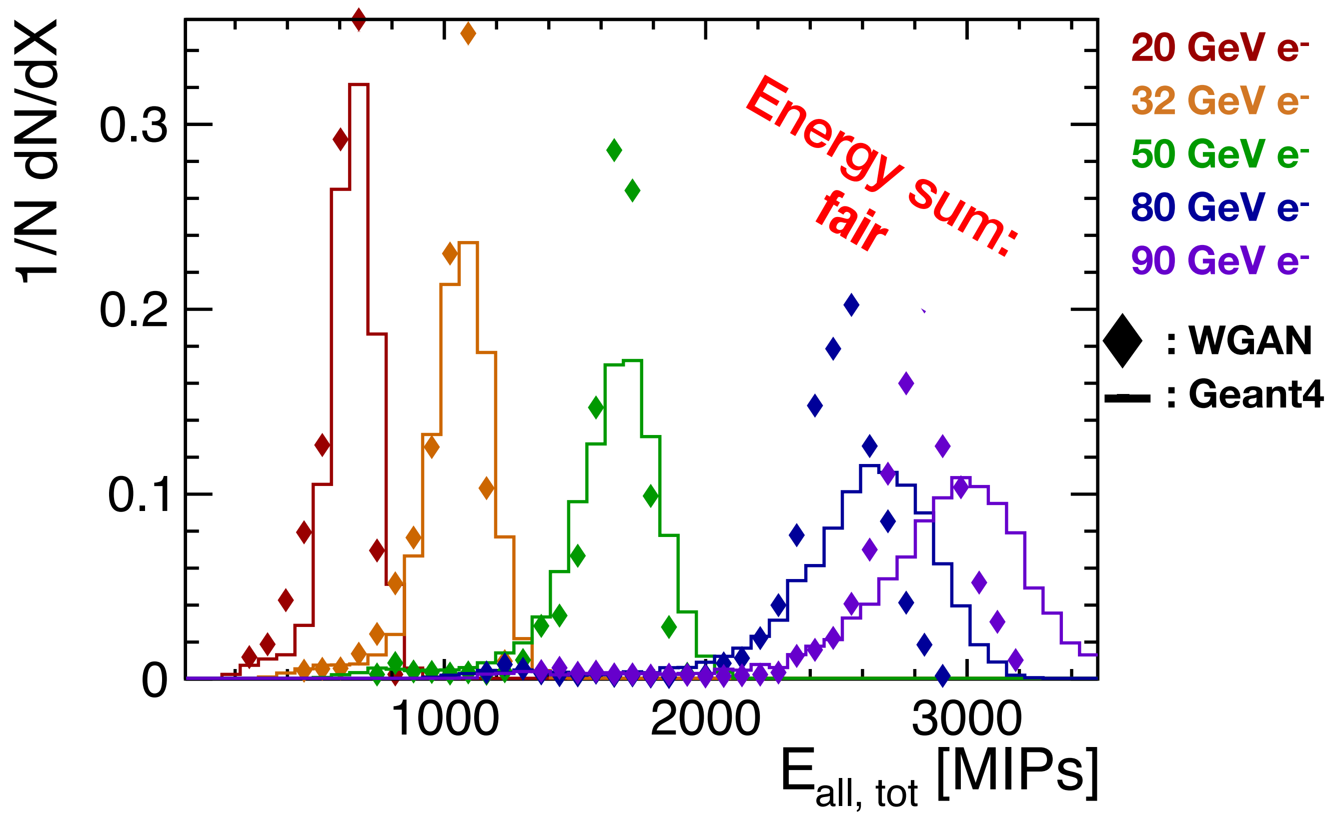
x WGAN: A few pixels outside sensor filled.

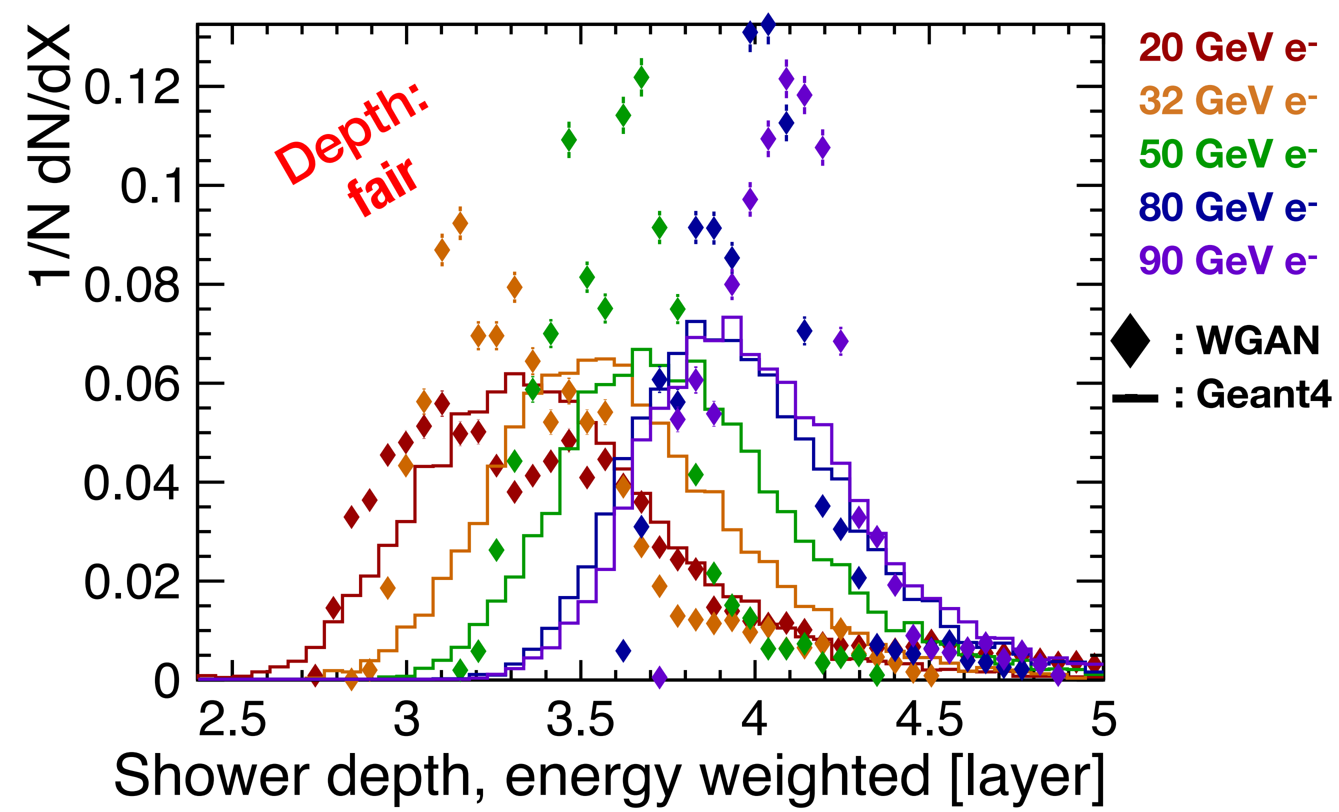
Comparison: Distributions of 1D observables





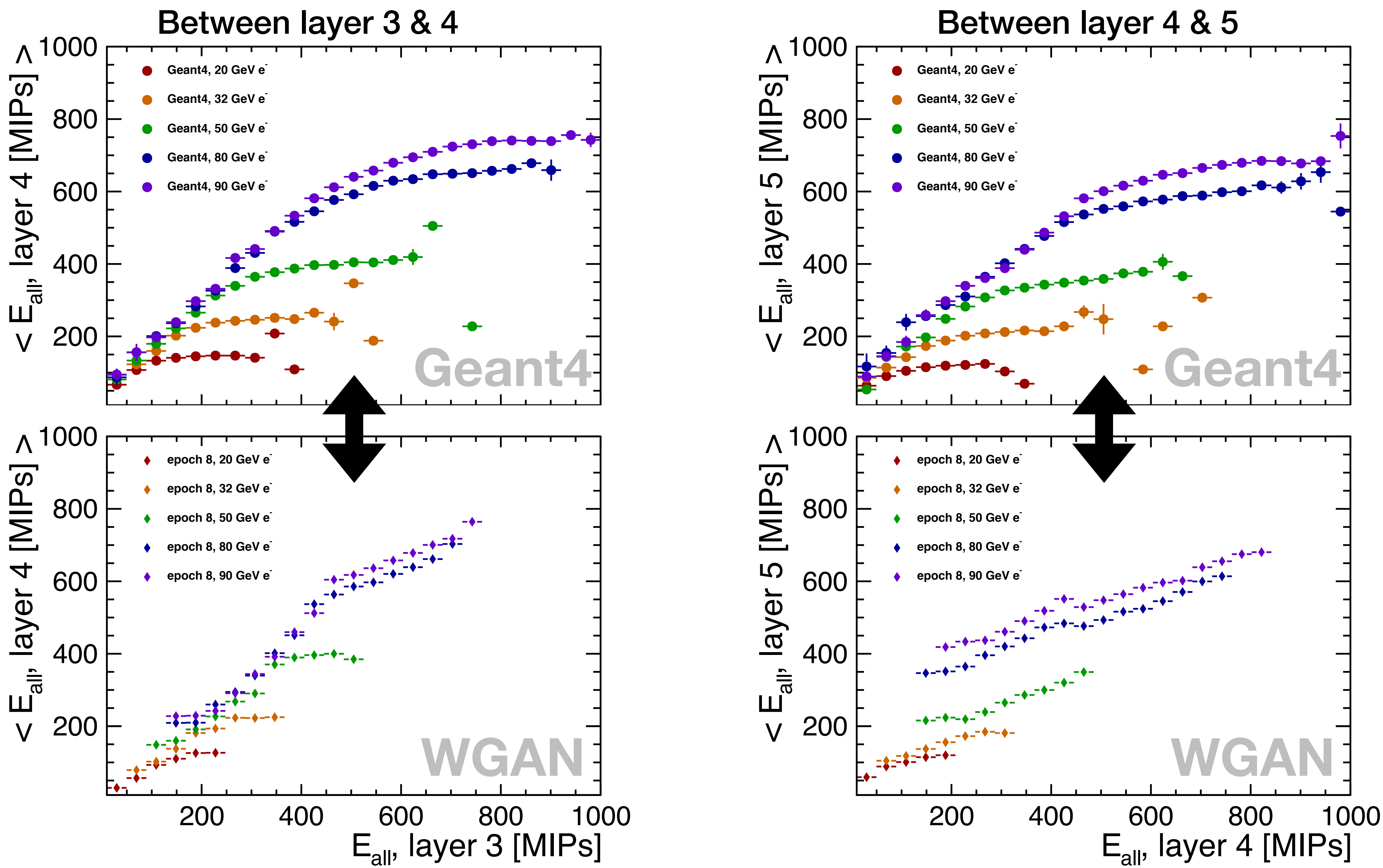
Comparison: Distributions of 1D observables





Correlation between layers

- Summed energy in one layer <—> sum in previous layer.



O(x1000) faster calorimeter simulations possible

- Typical 20-90GeV e$^-$ shower generated within **0.5-2 seconds** using Geant 4 .

Different hardware setups, fixed generator network architectures: presented here*

Computing Setup	Evaluation time for 20, 32, 50, 80, 90 GeV electron mixture (1:1:1:1:1)	x Speed-up compared to Geant4 simulation (=1s)
Intel® Xeon® CPU E5-1620	**5.2 seconds / 100 showers**	**x 20**
NVIDIA Quadro K2000	**1.8 seconds / 500 showers**	**x 280**
NVIDIA GTX 1080	**0.533 seconds / 2000 showers**	**x 3,750**

Fixed hardware setup: NVIDIA GTX 1080 , **different generator architectures**

Generator network architecture	Evaluation time for 20, 32, 50, 80, 90 GeV electron mixture (1:1:1:1:1)	x Speed-up compared to Geant4 simulation (=1s)
recurrent merging of layers*	**1.42 seconds / 2000 showers**	**x 1,410**
presented here*	**0.533 seconds / 2000 showers**	**x 3,750**
only 3D (de-) convolutions*	**0.075 seconds / 2000 showers**	**x 26,670**

* for network details, please refer to the backup slides

Summary - Calorimeter WGAN

- Generative models: promising **fast simulation tools** for particles' passage through matter.

<u>This study:</u>

- **Wasserstein GAN** concept instead of traditional GANs.
- **Conditioning** impact **position** & **incident** energy shower generating electrons.
- **CMS HGCal prototype** as real-life calorimeter assumed.

(Training with beam test data is **possible**.)

<u>Key observations:</u>

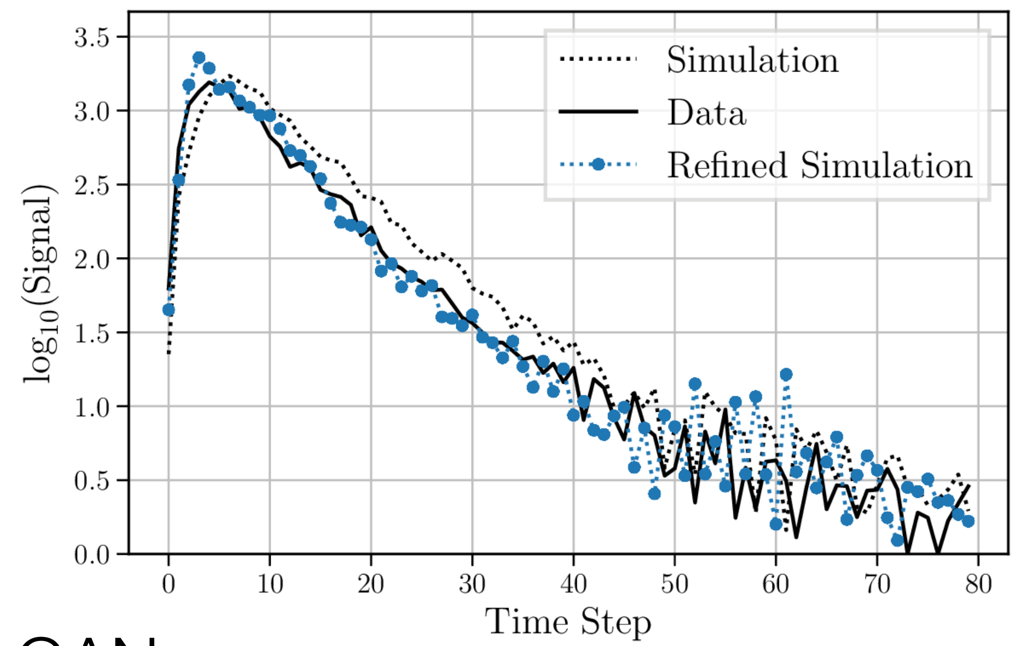
➡Many **reconstructed quantities** & key **correlations** of generated showers appear in many aspects **surprisingly close** to Geant 4 simulation.

➡Here: Inference step **O(1000)x faster** than Geant 4 .

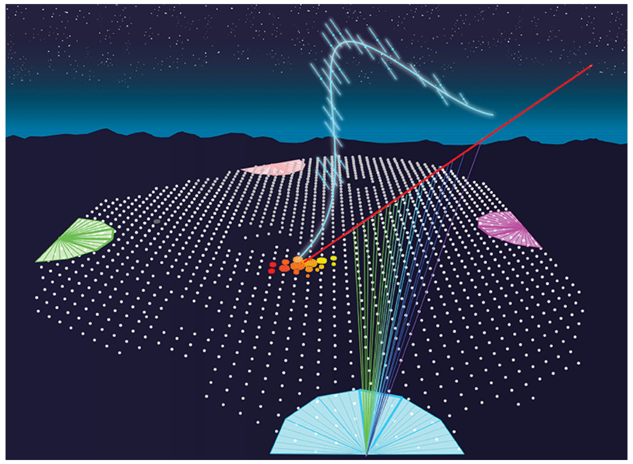
➡**No mode collapsing.**

Refining Detector Simulation using Adversarial Networks

Refining Detector Simulation using Adversarial Networks

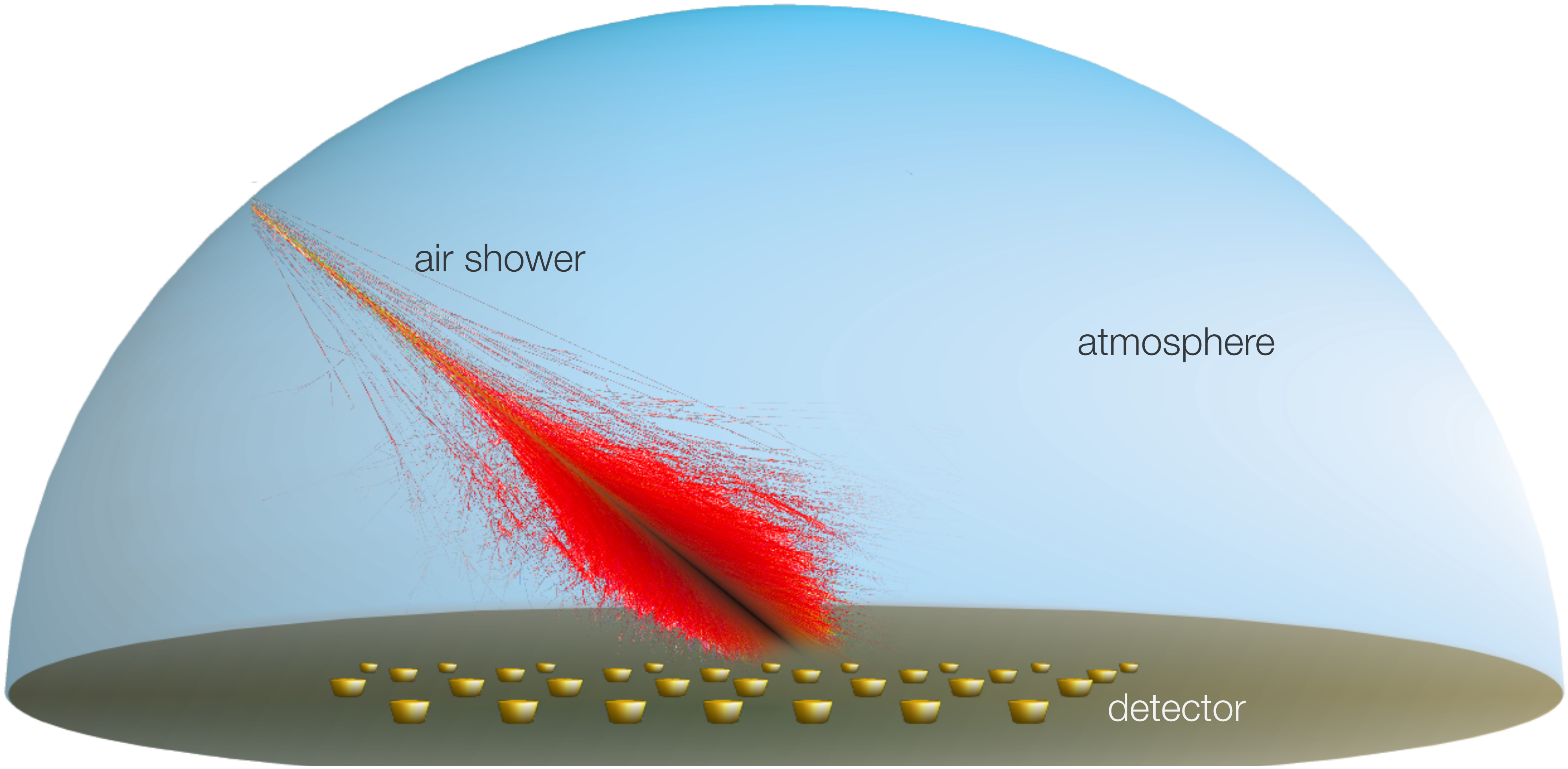


- The Pierre Auger Observatory
- Simulation / Data mismatches
  - Simulation refinement using Wasserstein GANs
- DNN training using refined simulations



Generating and refining particle detector simulations using the Wasserstein distance in adversarial networks - ArXiv:1802.03325

Cosmic Ray Detection in a Nutshell



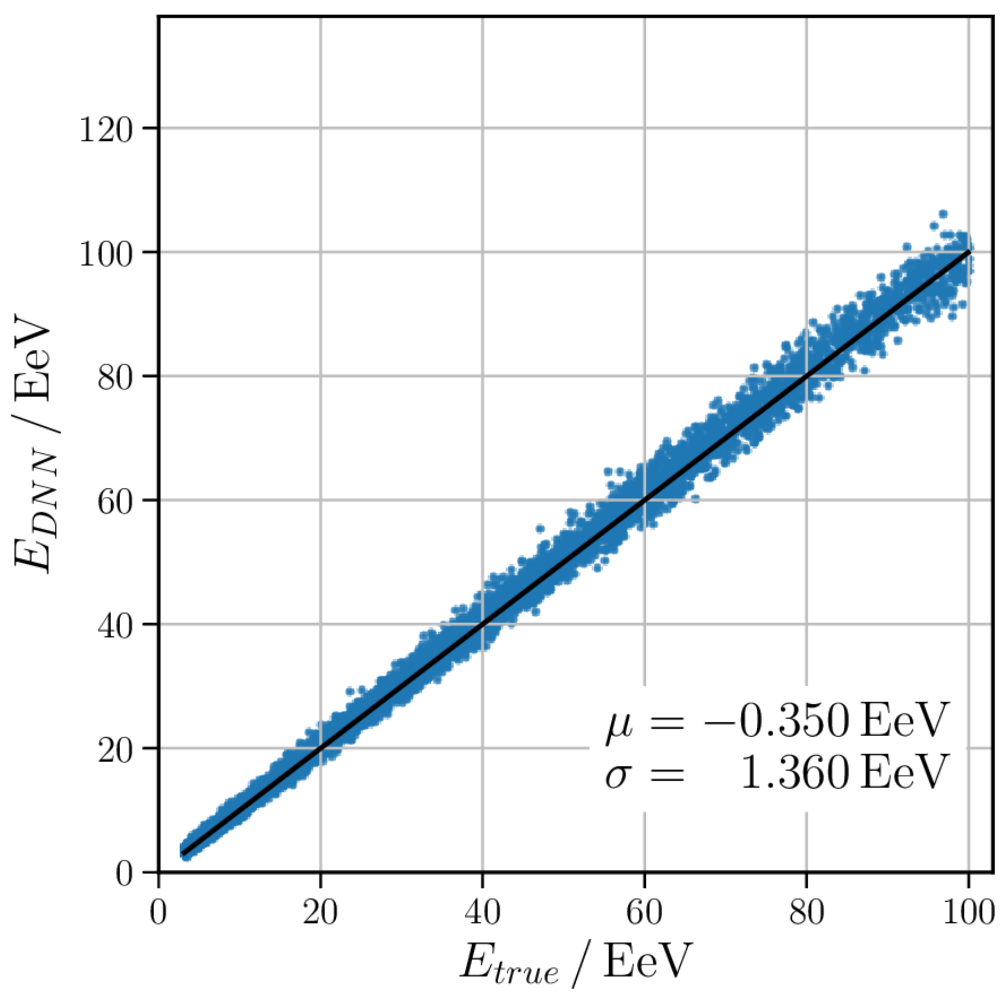
→ Large calorimenter
with single readout layer

Data / Simulation Mismatches

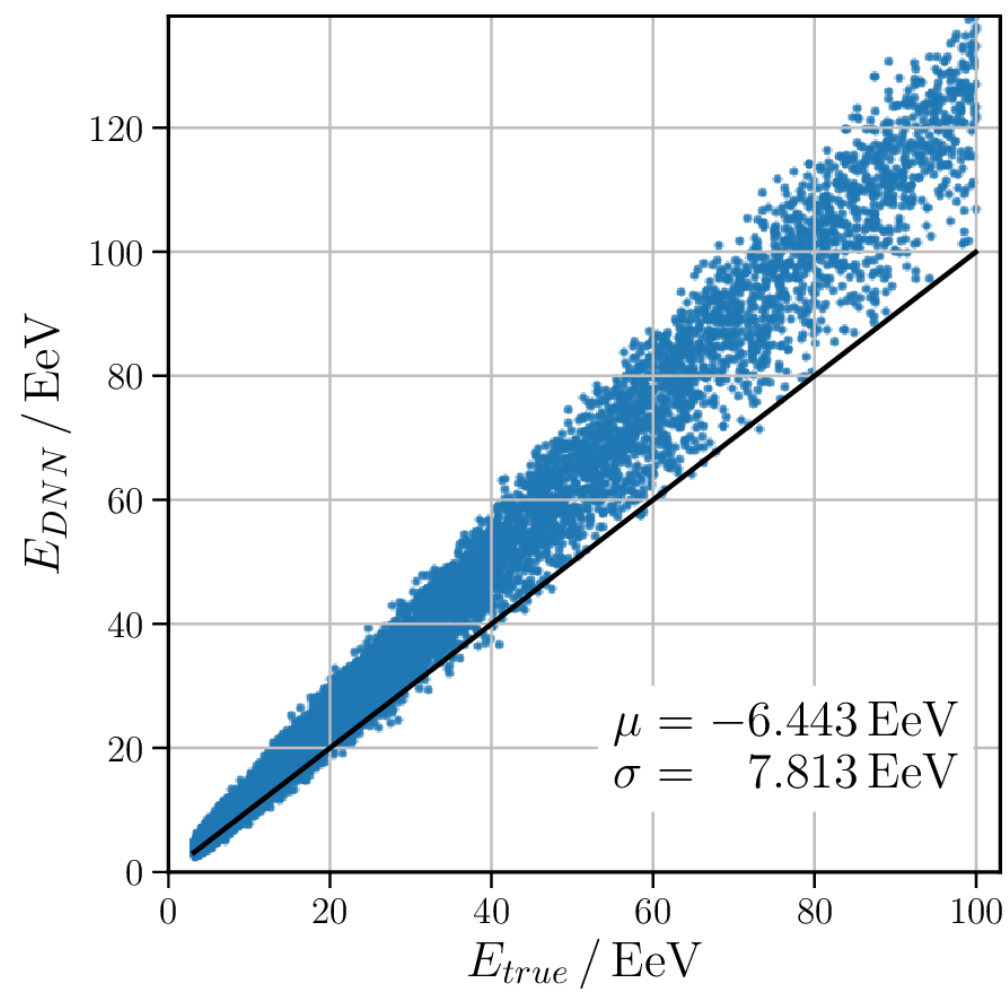
- DNNs very sensitive to simulation / data mismatches

→ *Performance gap*

Energy reconstruction: **simulation**

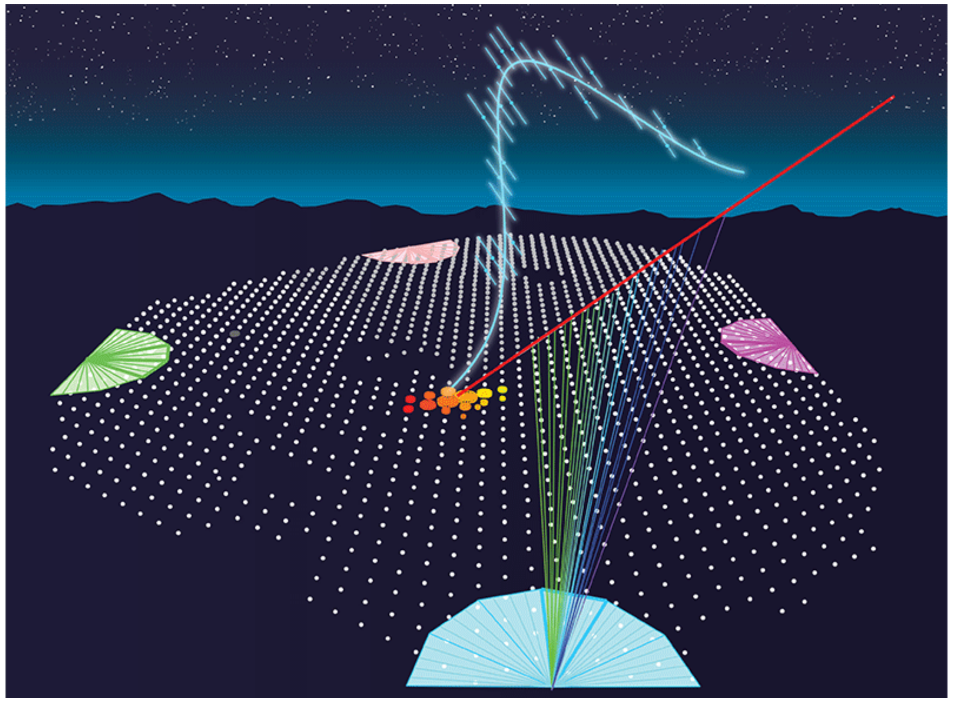


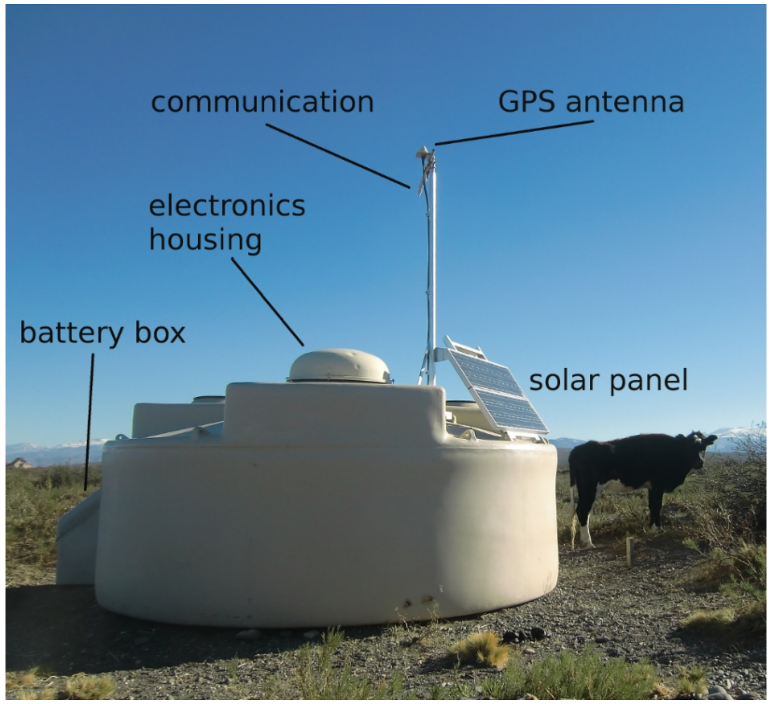
Energy reconstructed: **data**



The Pierre Auger Observatory

- Cosmic ray observatory in Argentina
- Completed 2008
- Detection of UHECR
  - $\mathrm{E} > 10^{17.5} eV$
- **Hybrid technique**
  - 27 Fluorescence telescopes
  - 1660 Surface detector stations
- Array size ~ 3000 km²

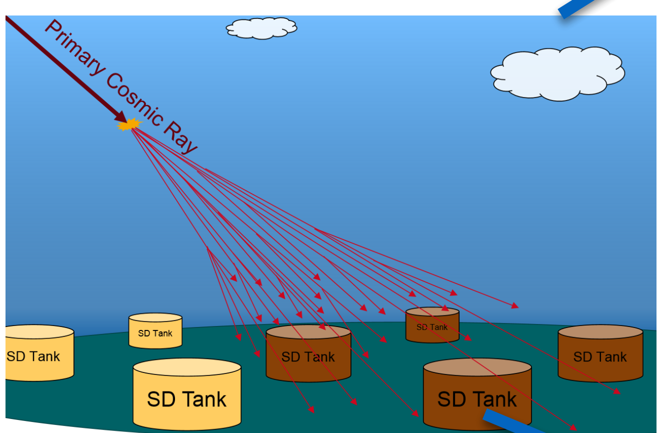


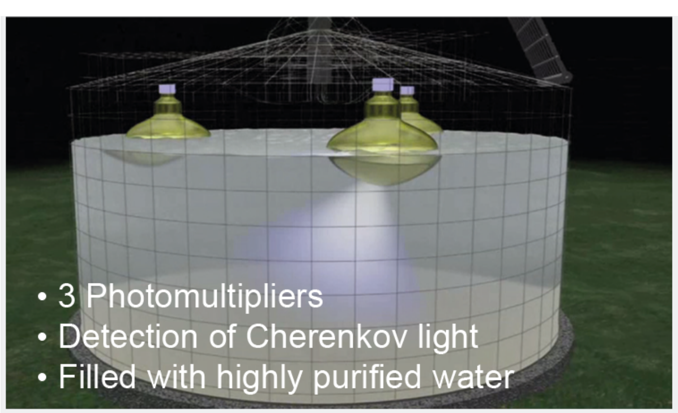


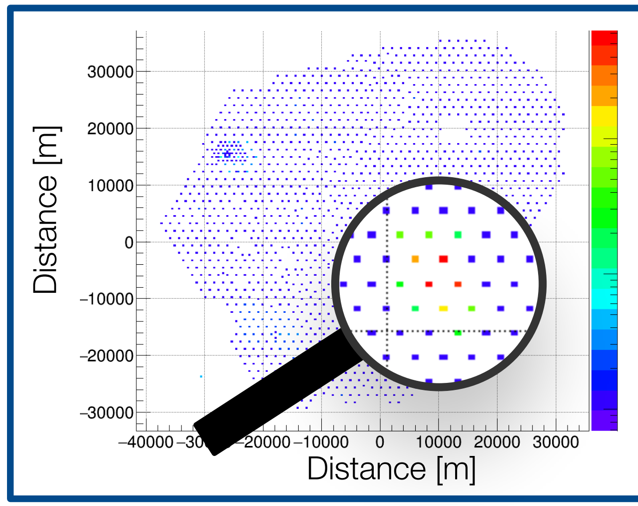
Surface detector station

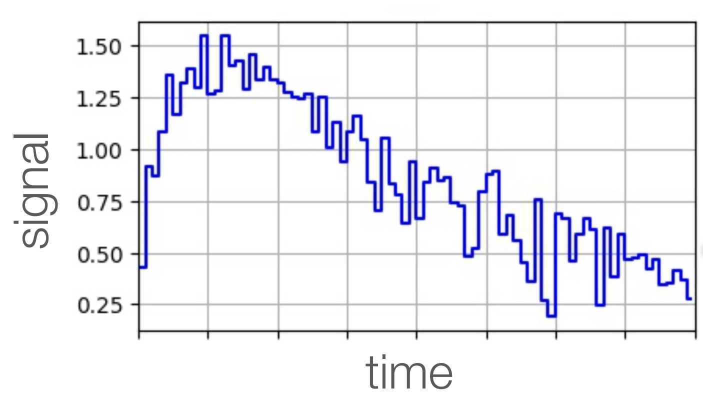
Air Shower Reconstruction using Deep Learning

Surface Detector measures
footprint including:
- Time traces
- Arrival times

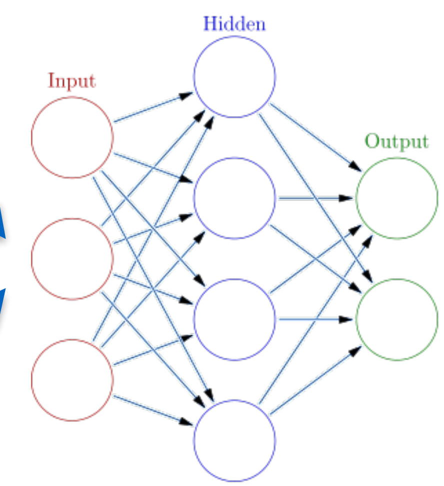






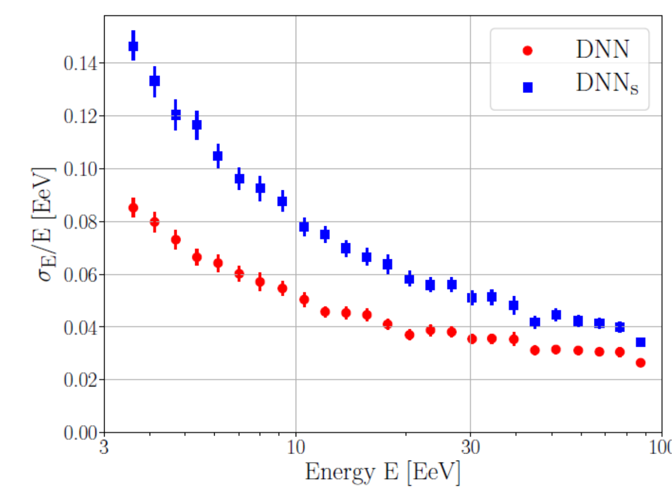


Deep Convolutional
Neural Network
"AixNet"



Shower depths

Energy



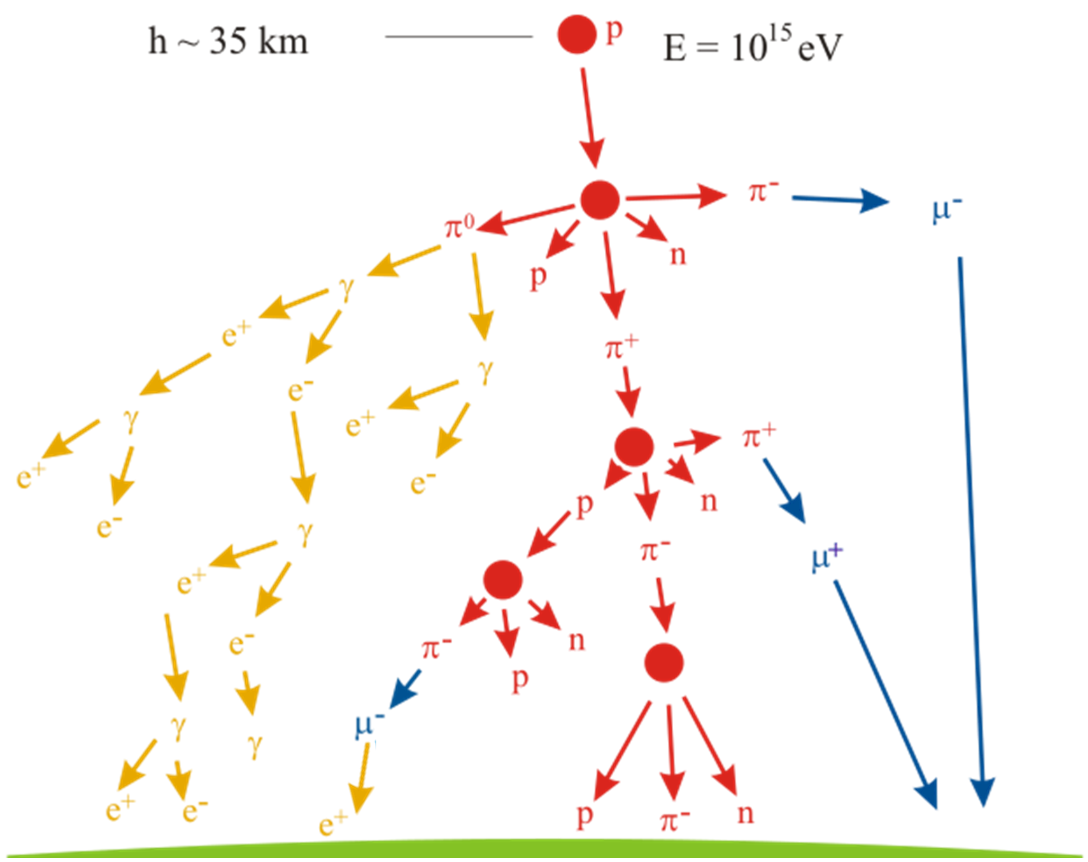
Astroparticle Physics 97 (2018) 46-53

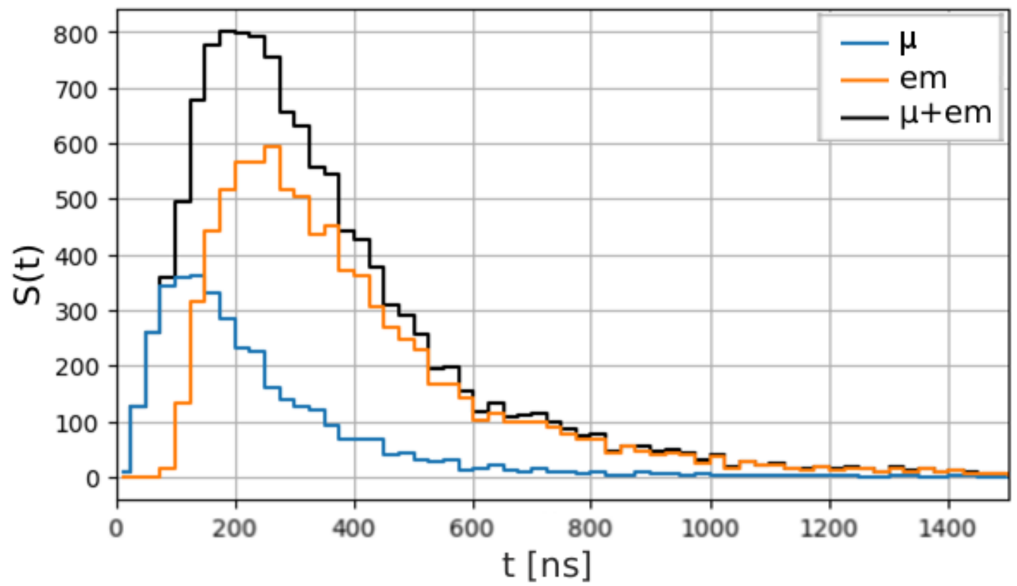
III. Physikalisches Institut

RWTH AACHEN UNIVERSITY

Extensive Air Showers

- Main air shower components
  - Muonic
    - Straight lines
    - Shower front
  - Electromagnetic
    - Atmospheric shielding
    - Time delay - component broadening





Underestimated muon flux in simulations

glombitza@physik.rwth-aachen.de

MC / Data Mismatches

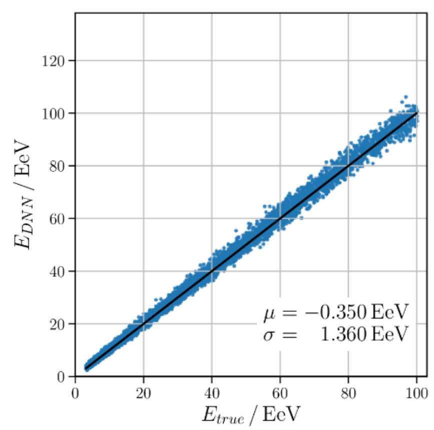
- Simulate 2 independent sets
  - 1 „data“ and 1 „simulation“
    - Different component fractions
    - Matching phase space

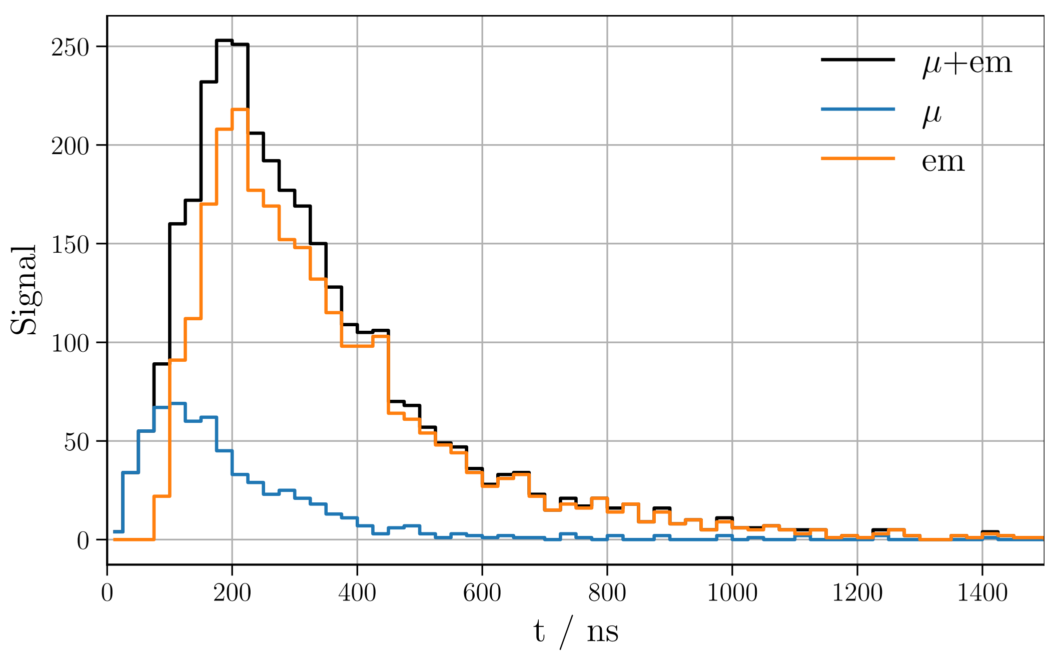
Neural network can‘t handle the modified traces!

<u>Simulation</u>

70% electromagnetic

30% muonic



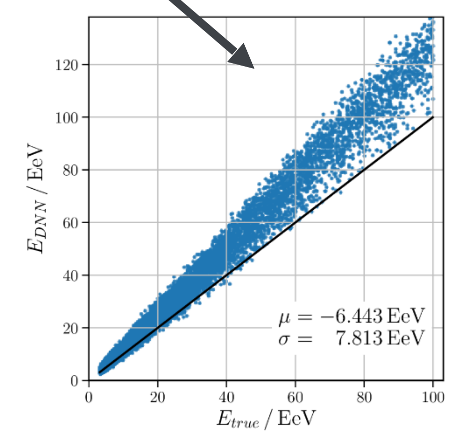


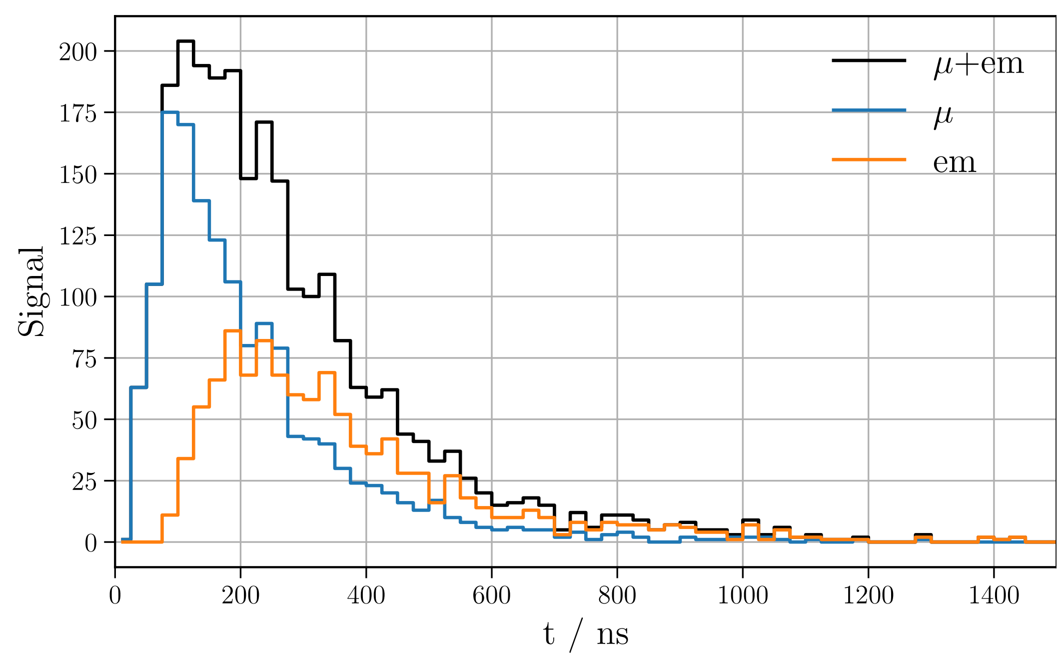
<u>Data</u>

30% electromagnetic

70% muonic

\+ Increased noise

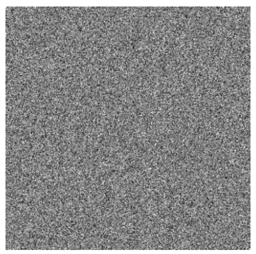




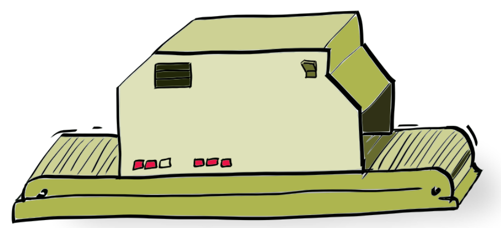
Refining Adversarial Network

Generator Network

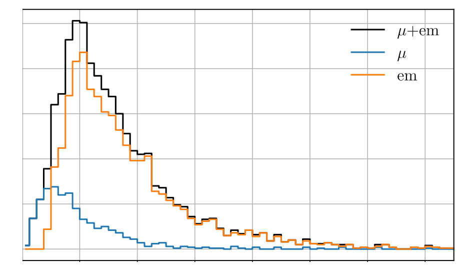
Noise



Generator

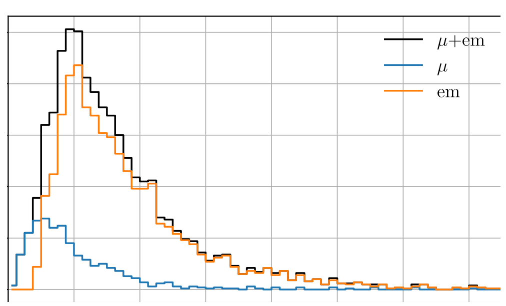


Generated time trace

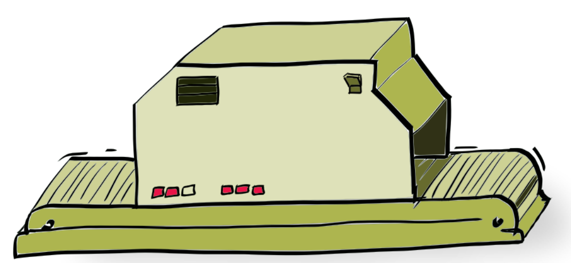


Refiner Network

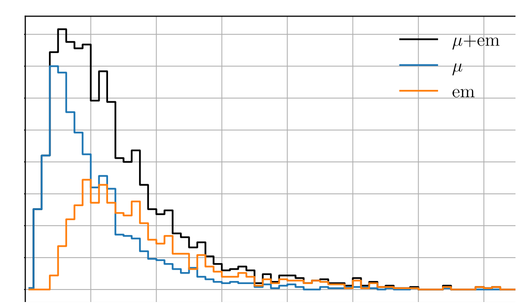
Simulated time trace



Refiner



Refined time trace

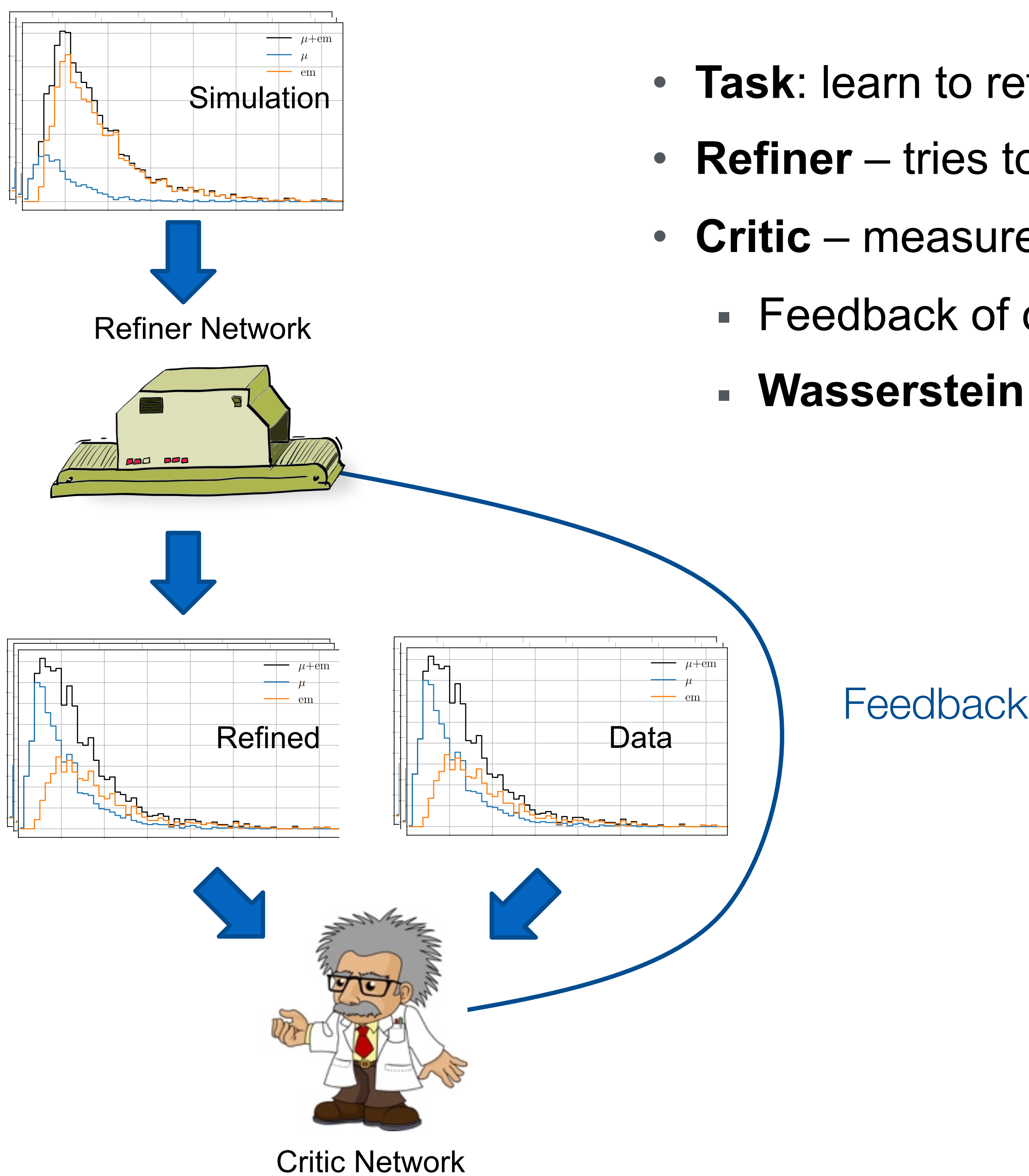


Corrections based on N dimensions

Change variables instead of applying scale factors

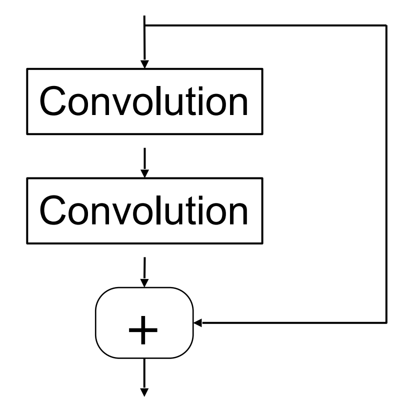
Jonas Glombitza
glombitza@physik.rwth-aachen.de

Refining Adversarial Network - WGAN

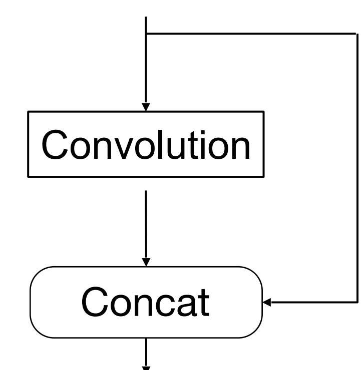


- **Task**: learn to refine simulation using 2 neural networks
- **Refiner** – tries to refine the simulation to look like data
- **Critic** – measure similarity between data / simulation
  - Feedback of critic improves refiner performance
  - **Wasserstein distance** as similarity measure

Refiner: ResNet

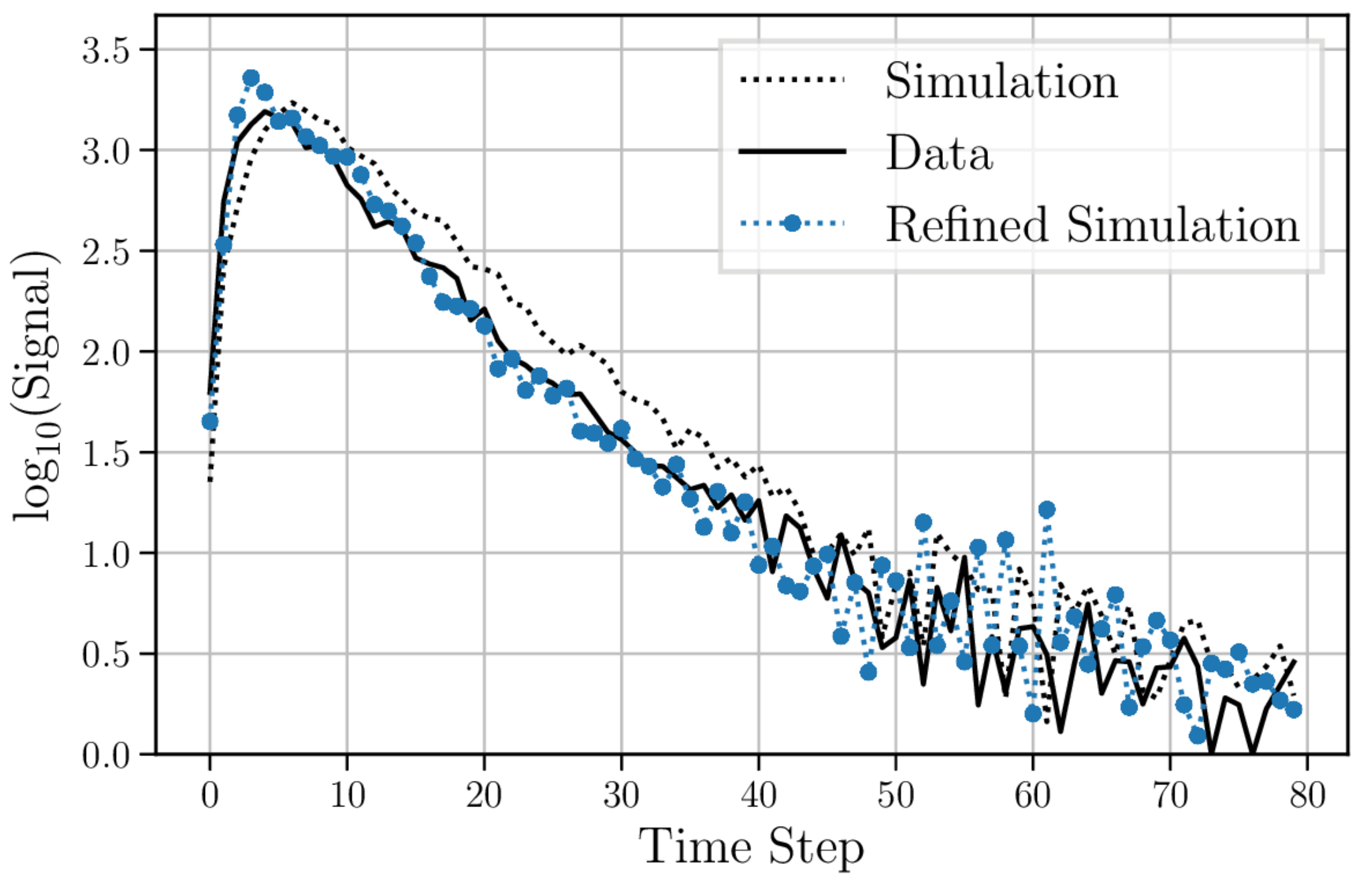


Critic: DenseNet



Refined Signal Traces

- Signal traces of an event with energy E = 69 EeV
- Evaluated on separate test set

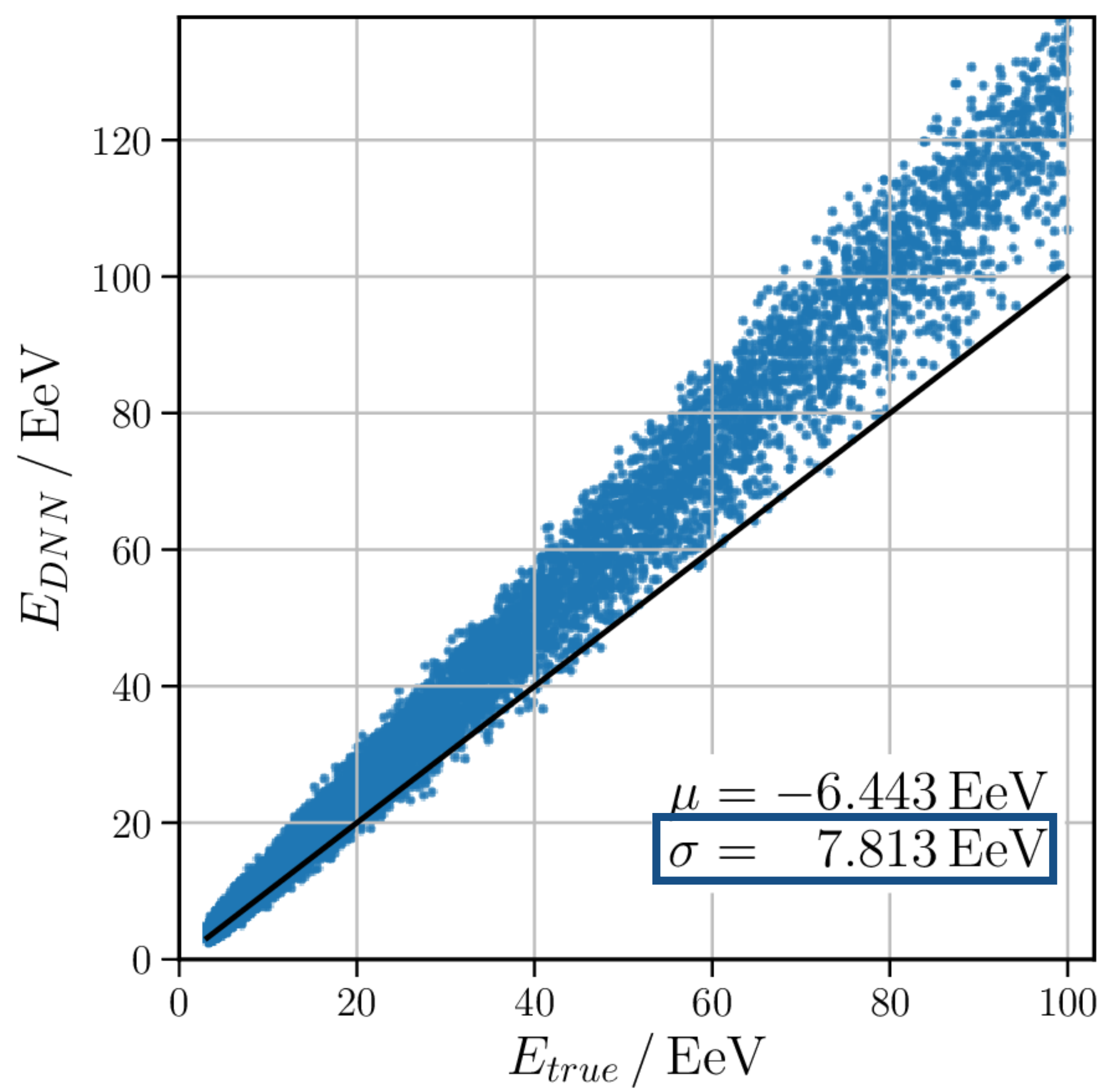


Refiner is able to shift simulation towards data

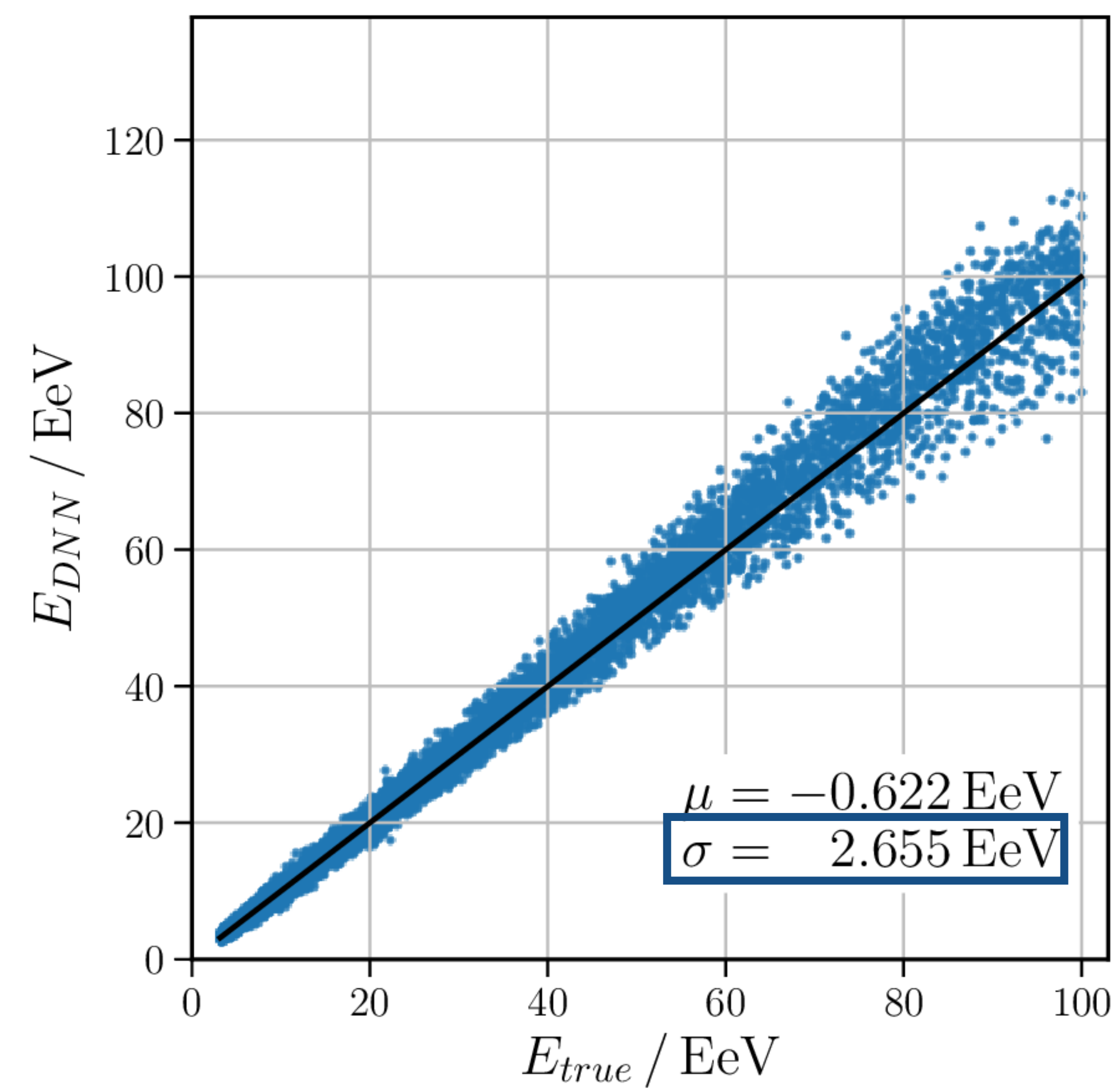
Evaluate Network Performance on Data

Evaluate network performance on **data** *(simulation, with different component scalings)*

Trained on **original simulation**



Trained on **refined simulation**



Training on refined simulations is able to improve energy reconstruction

Summary – Refining Adversarial Network

- Adapt Wasserstein GAN to tackle data / simulation mismatches
  - ResNet architecture in refiner
  - DenseNet architecture in critic
  - Wasserstein provides adequate measure of similarity
- Refine simulated data using adversarial training
  - Promising results to make DNN robust to data applications

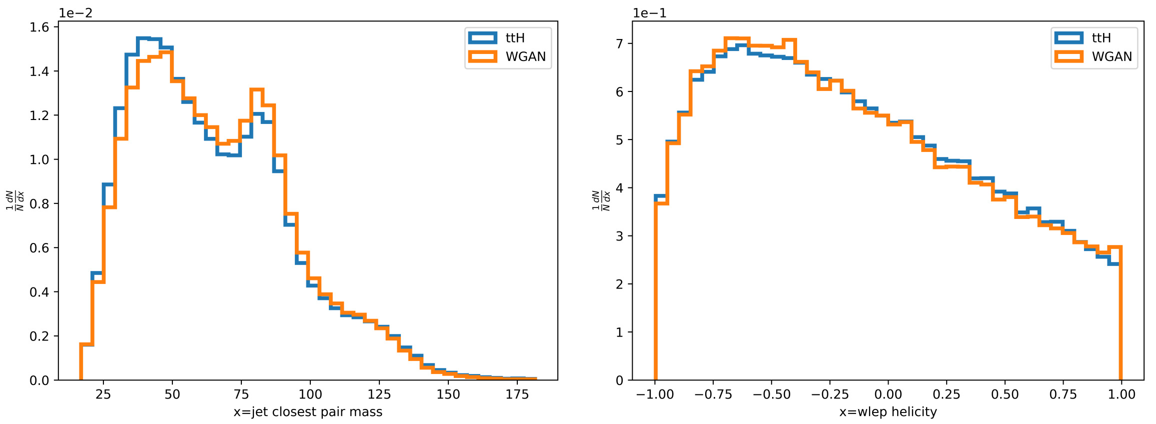
arXiv:1802.03325

- Alternative application for continuous simulation scale factors

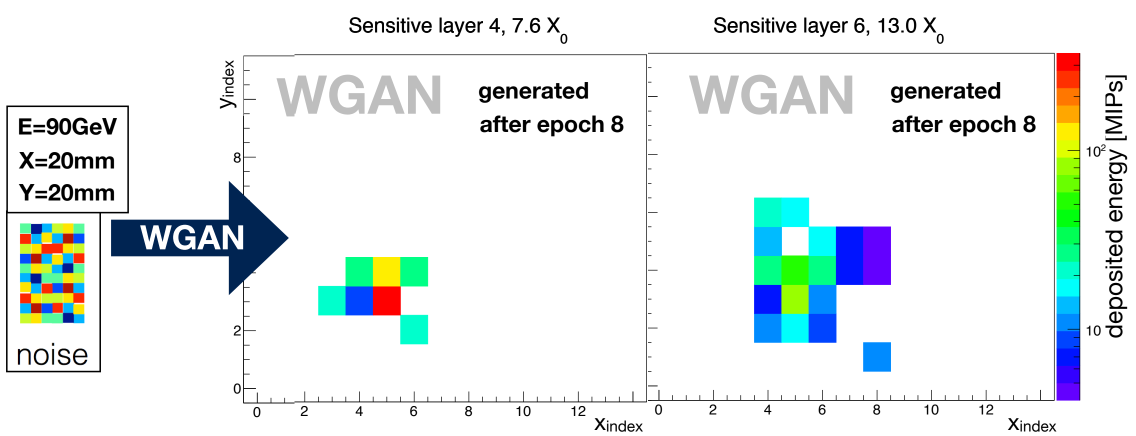
Conclusion

Conclusion

Generative modelling of correlated physics observables.
Fisher transformation, Wasserstein & Classifier Benchmark as quality measures.
Fast and successful simulation.

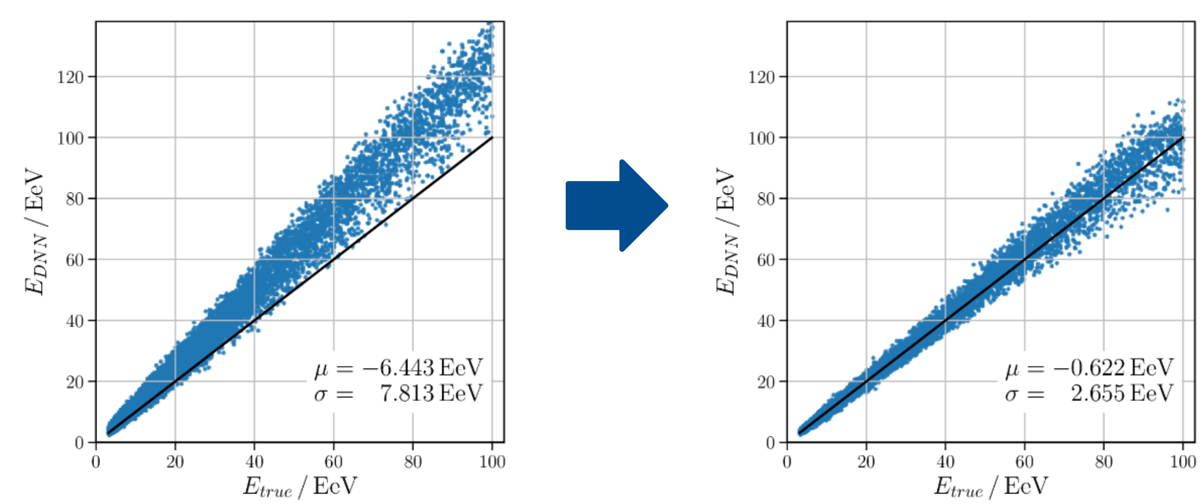


Conditional WGAN model for fast simulation of electromagnetic showers in a CMS HGCAL prototype. Surprisingly accurate. O(x1000) faster than Geant4. No mode collapsing.



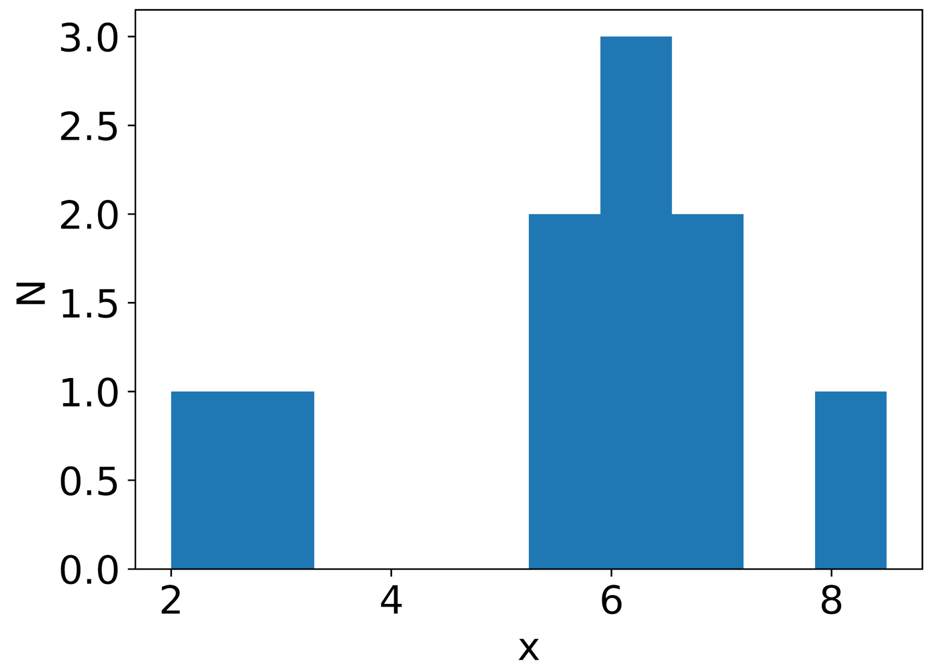
Supervised trained DNN shows improved performance after unsupervised refinement of simulation to match data

Generating and refining particle detector simulations using the Wasserstein distance in adversarial networks - arXiv:1802.03325

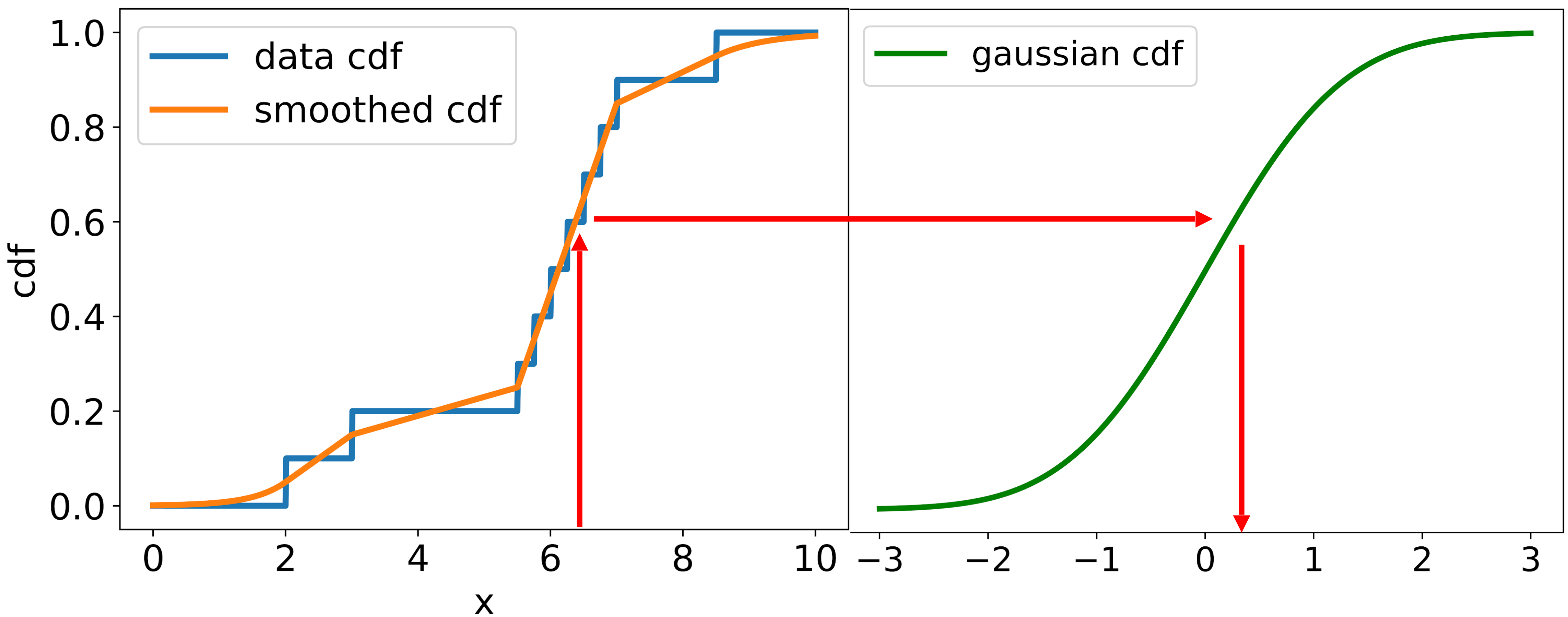


Backup - High-level variable generation

Shape normalisation

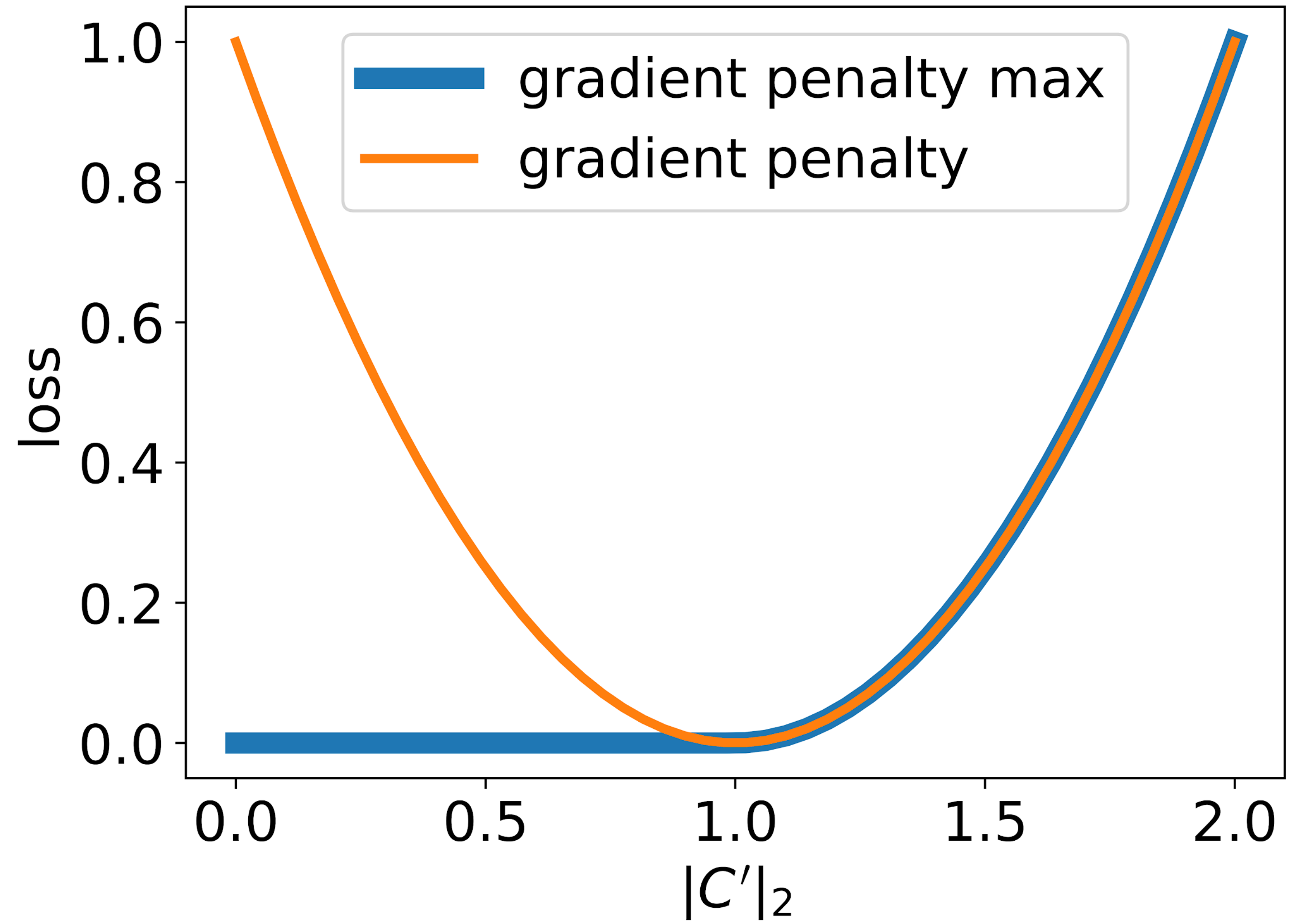


- Use smoothed cdf to get bijective transformation
- Can be chained to transform between two arbitrary distributions



Gradient penalty

- Perfect critic has everywhere gradient 1
- However this leads to unstable training when it is close to convergence
- Similar to overfitting: two close data points should be treated the same, not forced to be different by the gradient penalty
- Use GPmax formalism: Corresponds directly to Lipschitz condition



David Schmidt
david.schmidt@rwth-aachen.de

List of variables

26 high-level variables:

- Aplanarity
- Centrality
- Fox Wolfram 0-4
- Jet Min/Max/Avg Abs Deta
- Jet Min/Max/Avg Dr
- Jet Closest Pair 125 Mass
- Jet Closest Pair 125 Pt
- Jet Closest Pair Mass
- Jet Closest Pair Pt
- Jet Lep Min/Max Abs Deta
- Jet Lep Min/Max Dr
- Jet Sum Pt
- Sphericity
- Transverse Sphericity
- Whad Helicity
- Wlep Helicity

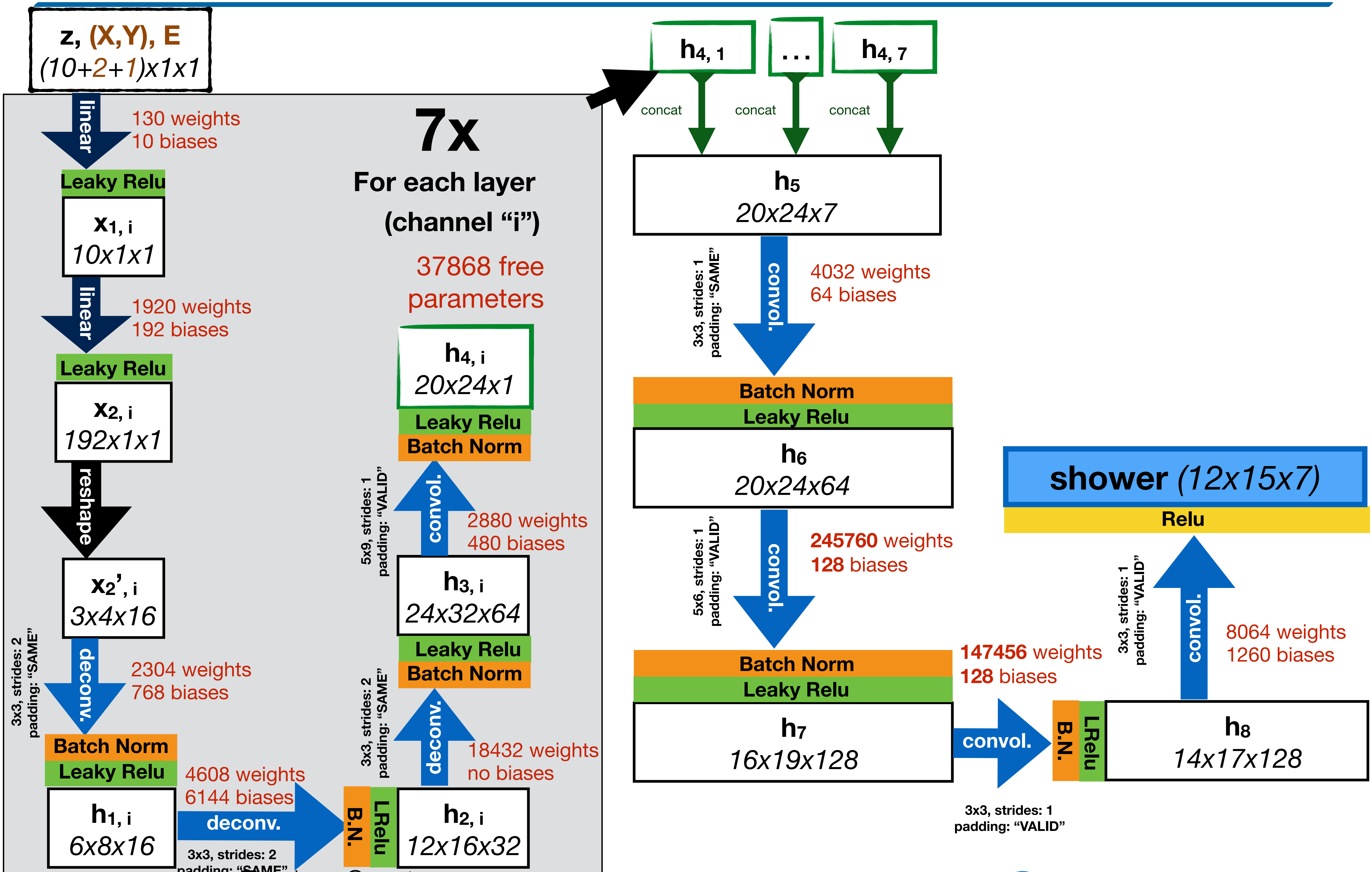
32 low-level variables:

- E, Px, Py, Pz of:
  - Best Bhad
  - Best Bj1
  - Best Bj2
  - Best Blep
  - Best Lj1
  - Best Lj2
  - Lep1
  - Nu1

These are ordered using the generated event!

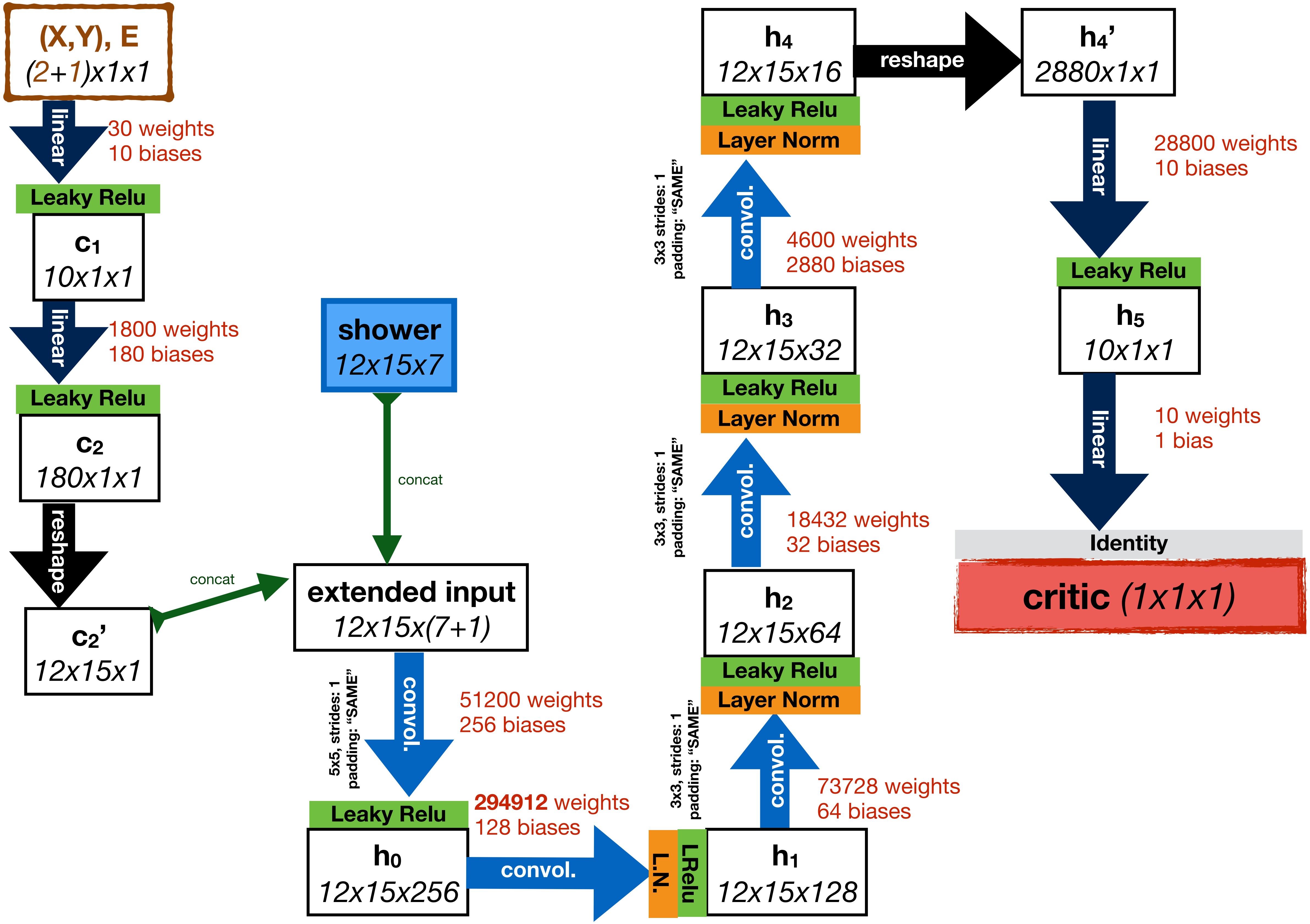
Backup - Calorimeter WGAN

Generator network with ~672k free parameters

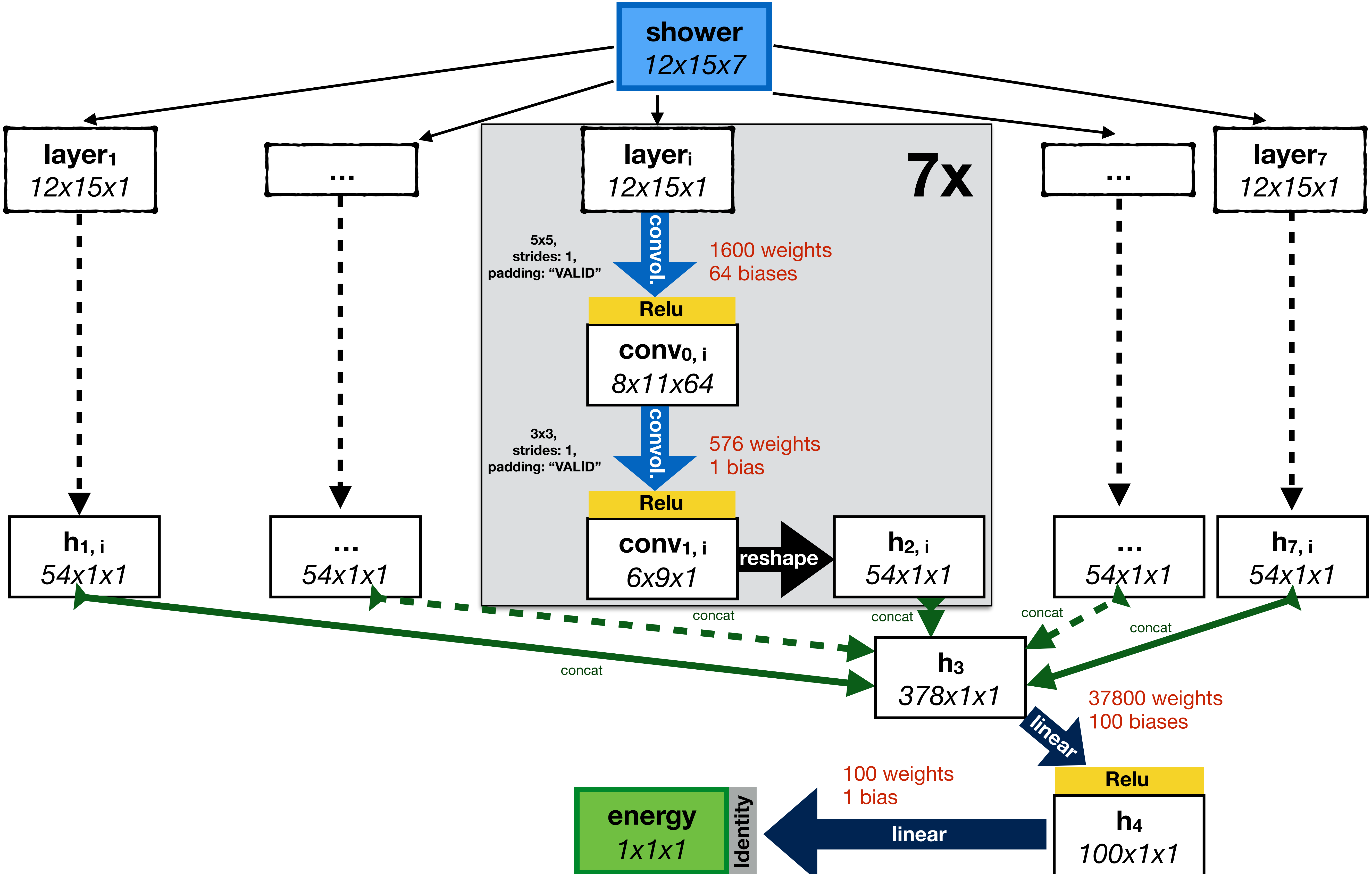


quast@physik.rwth-aachen.de
III. Physikalisches Institut
RWTH AACHEN UNIVERSITY

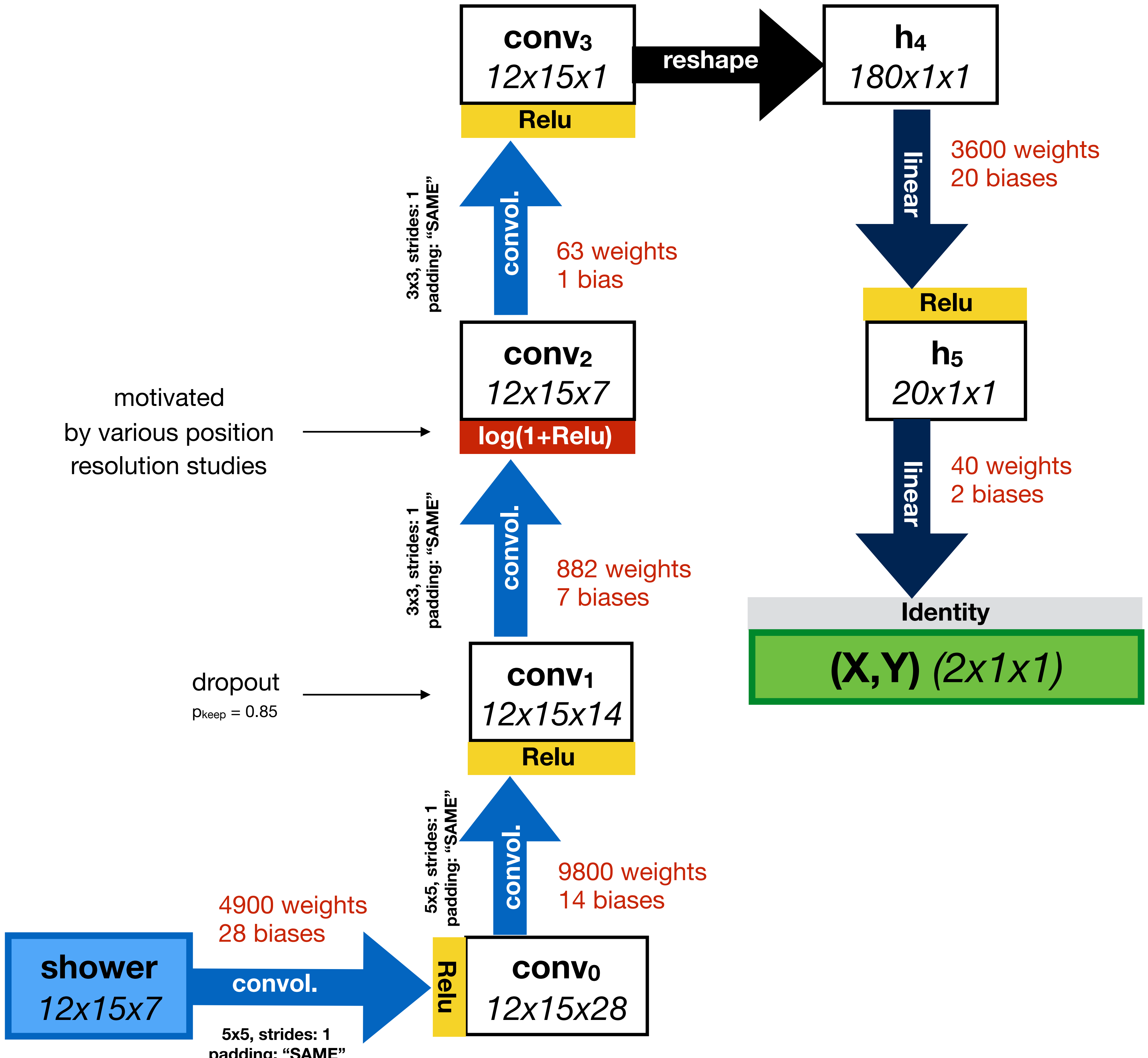
Critic network with ~477k free parameters



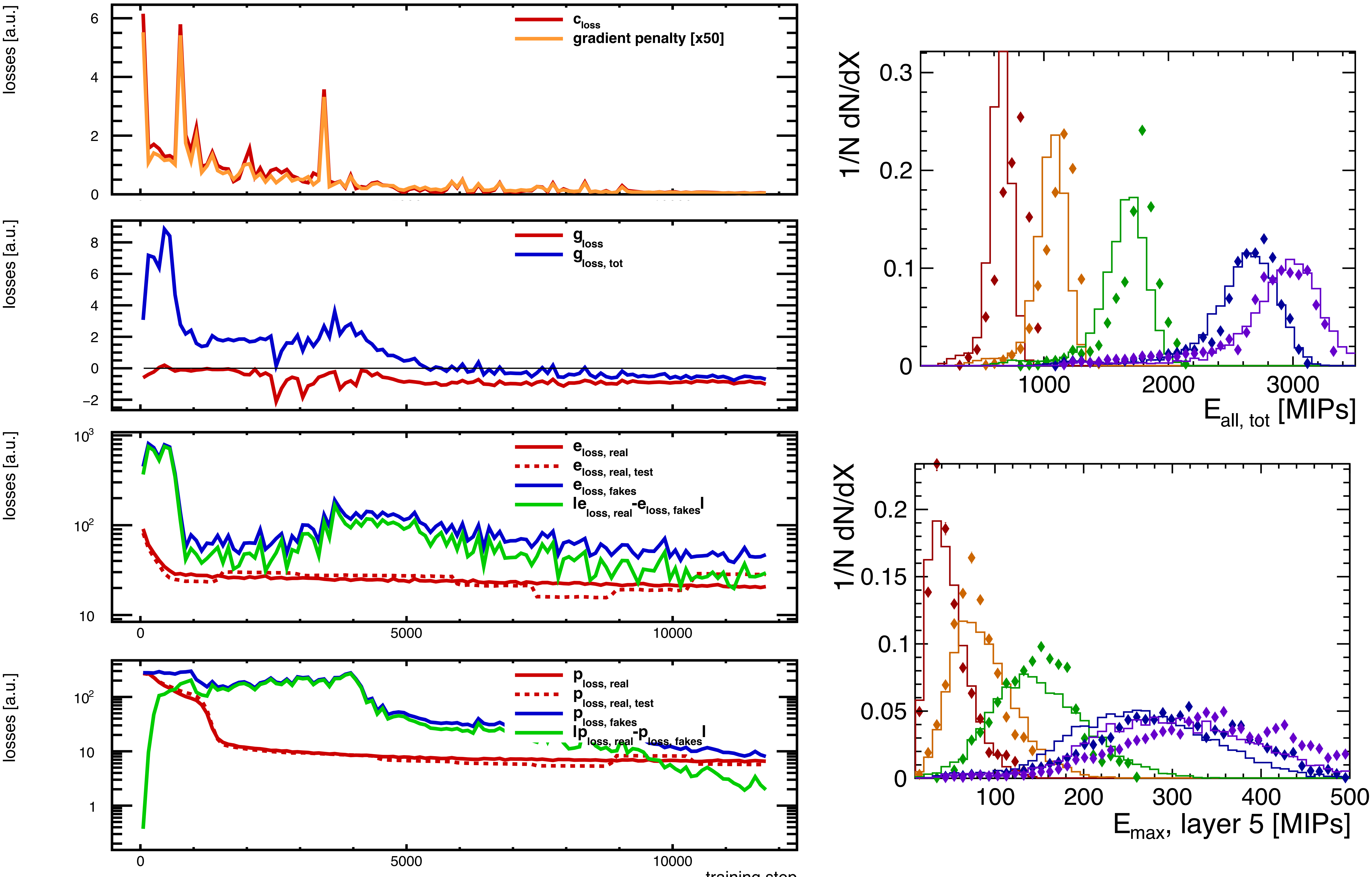
Energy regression network with ~54k free parameters



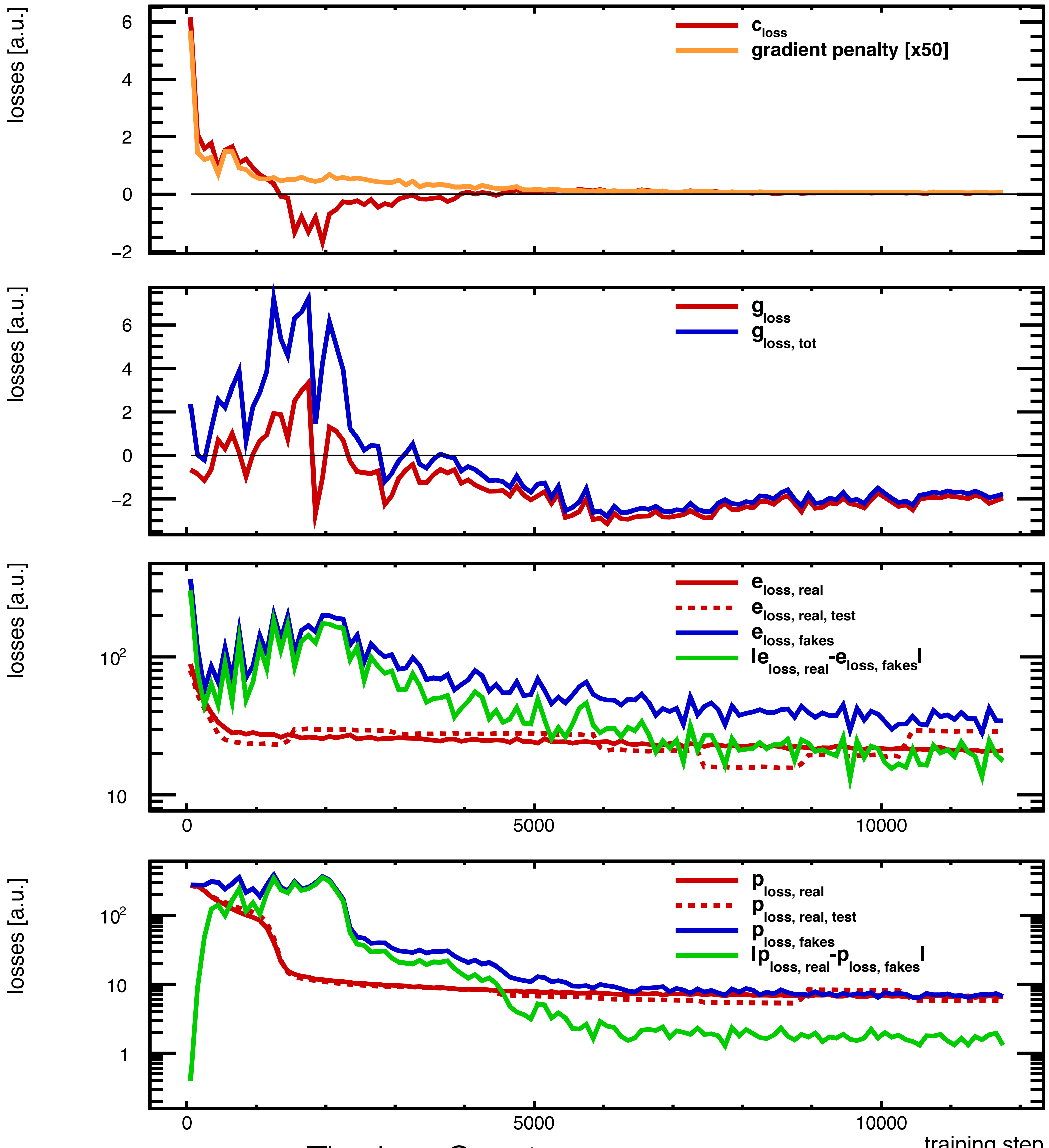
Position regression network with ~19k free parameters

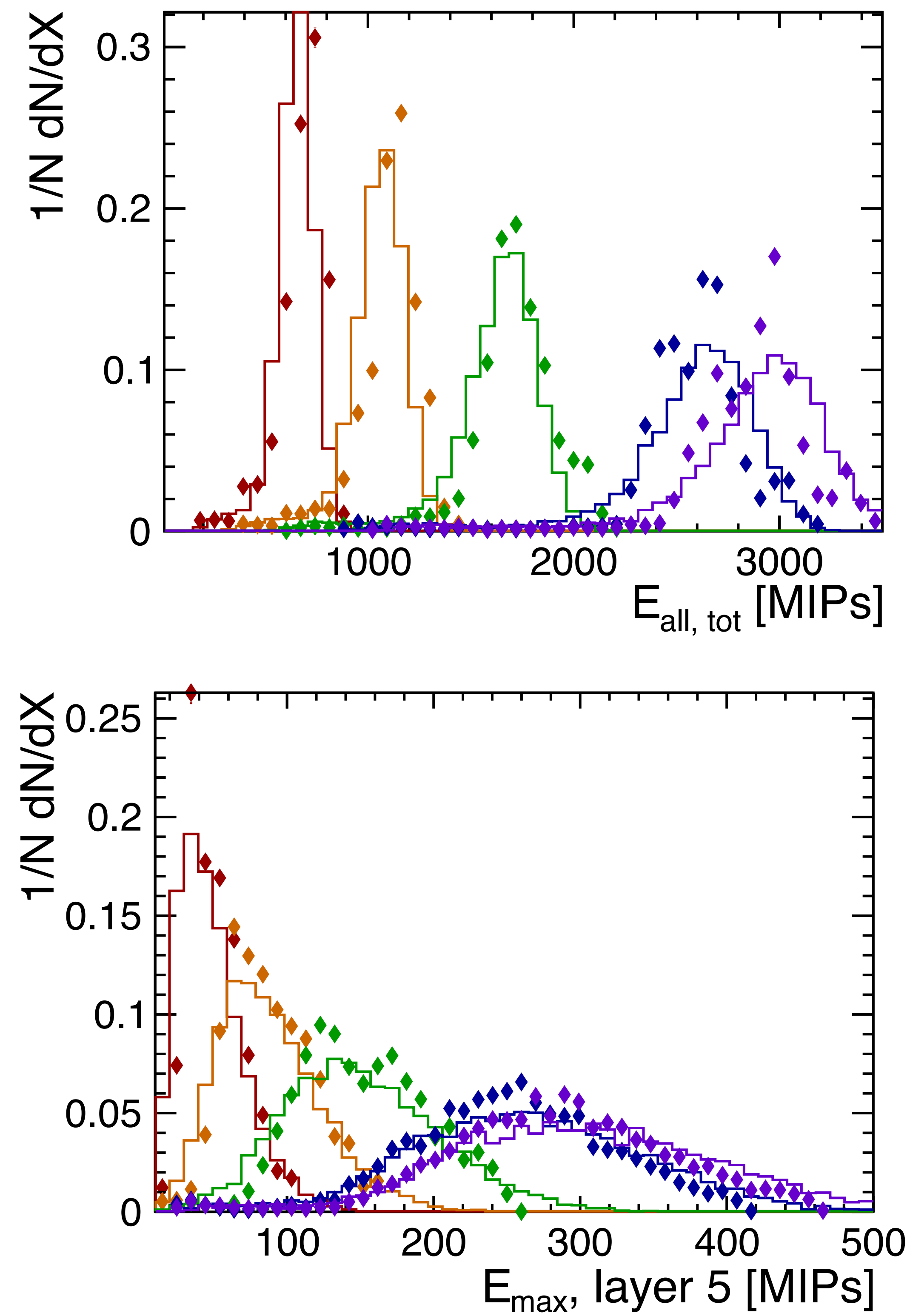


All the trainings have converged (e.g. iteration 2)

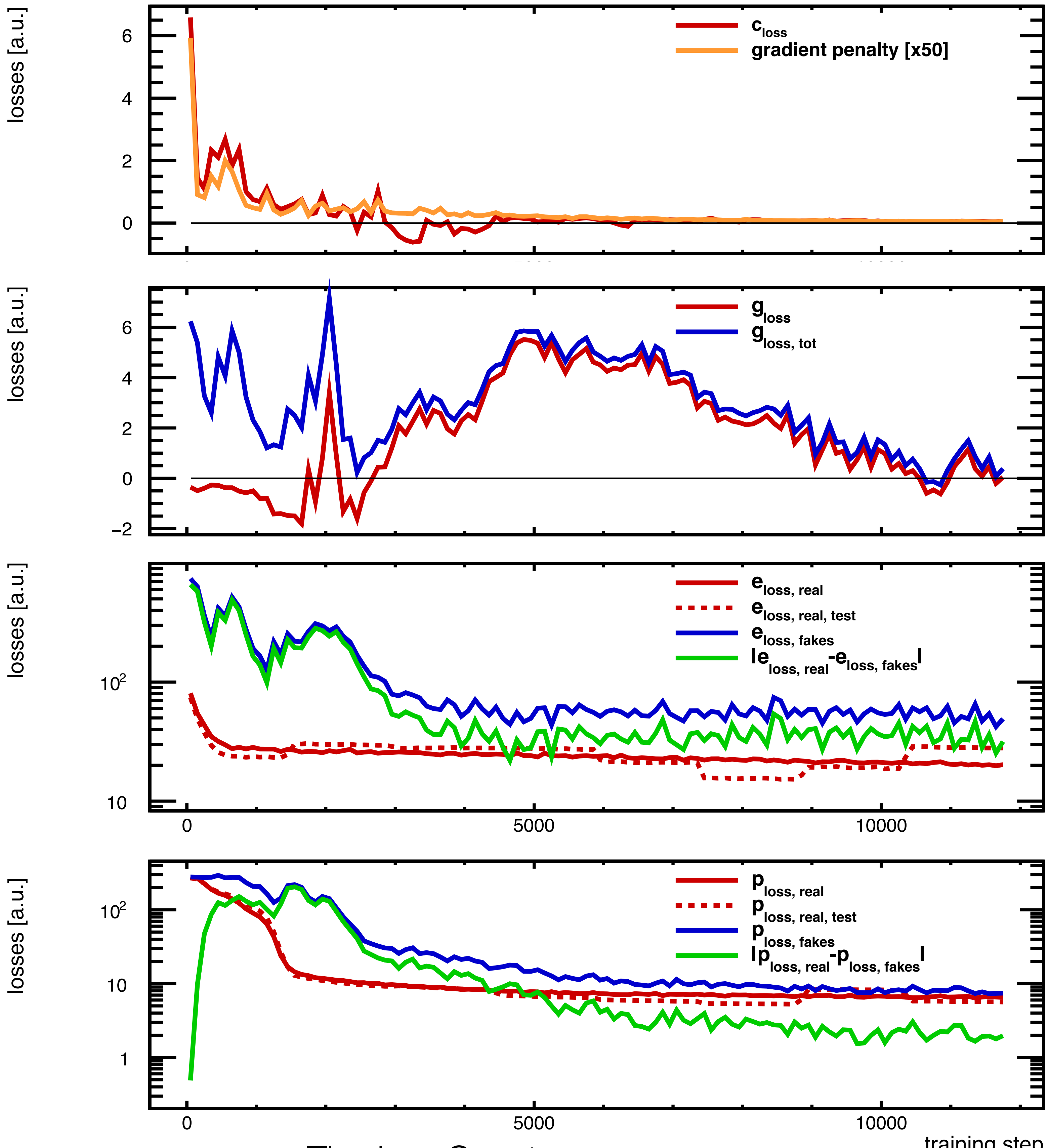


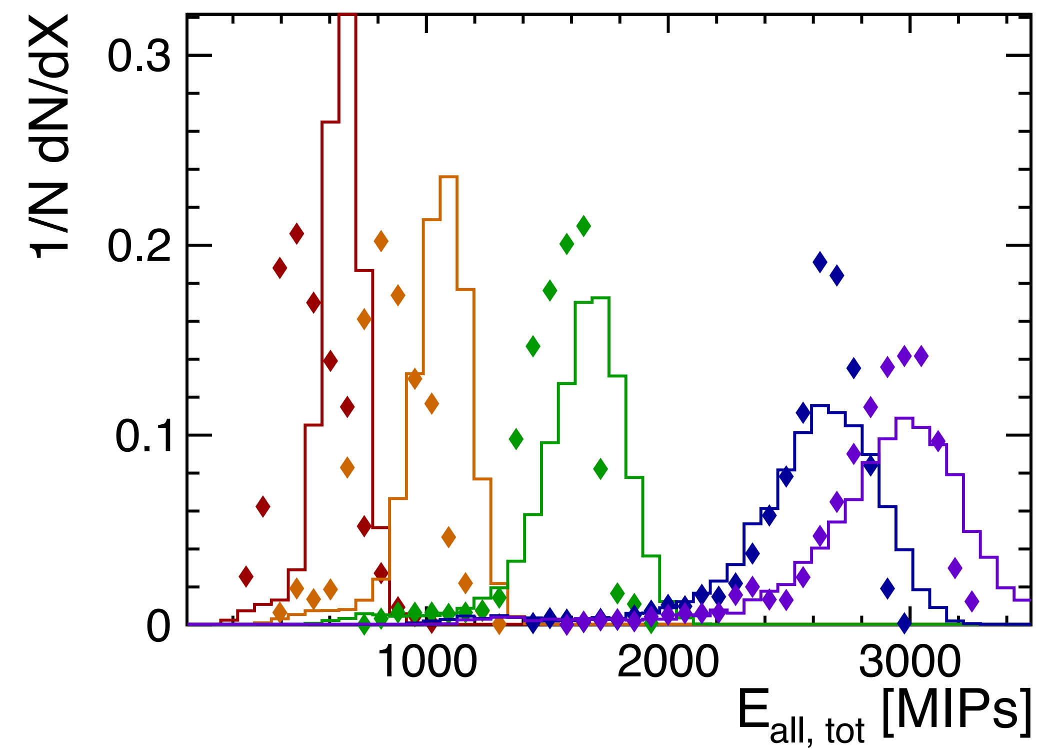
All the trainings have converged (e.g. iteration 5)

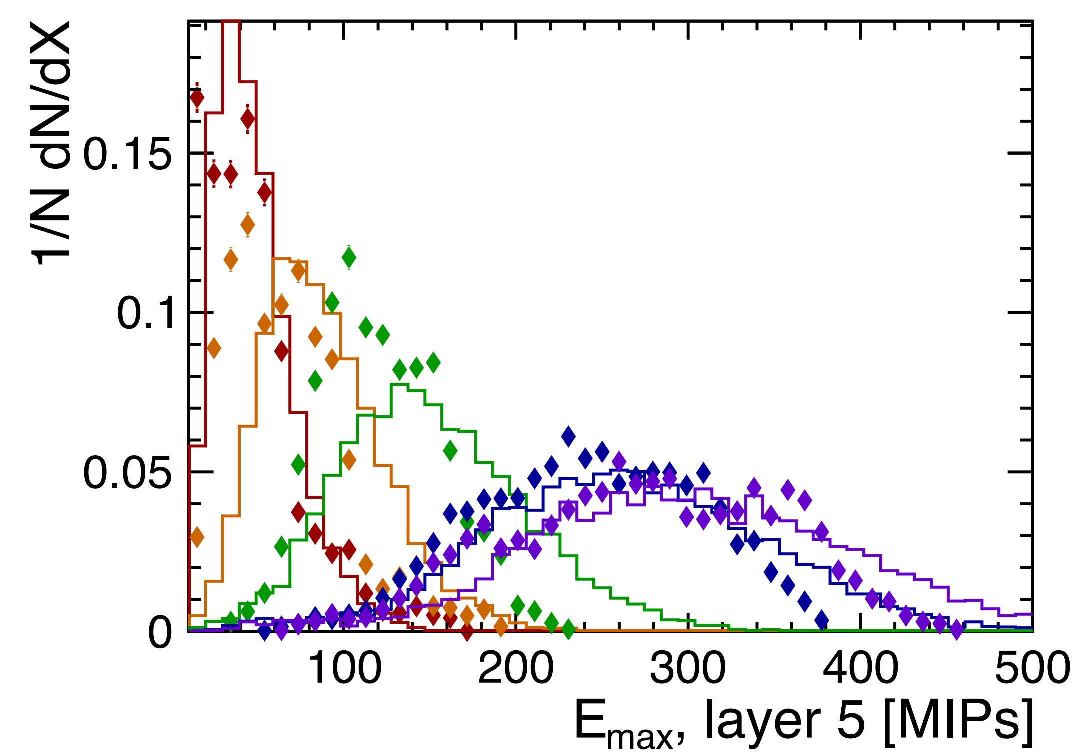




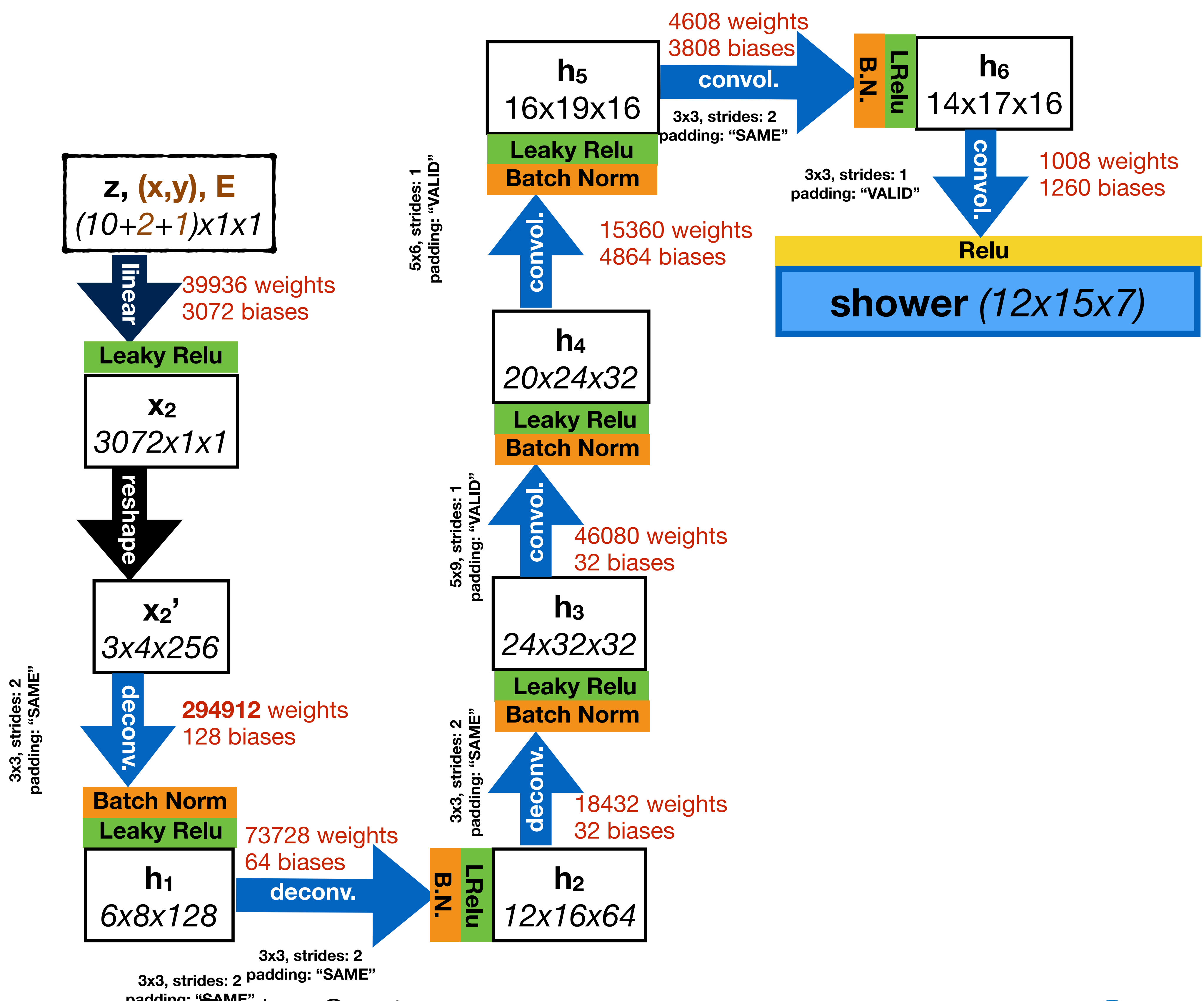
All the trainings have converged (e.g. iteration 10)



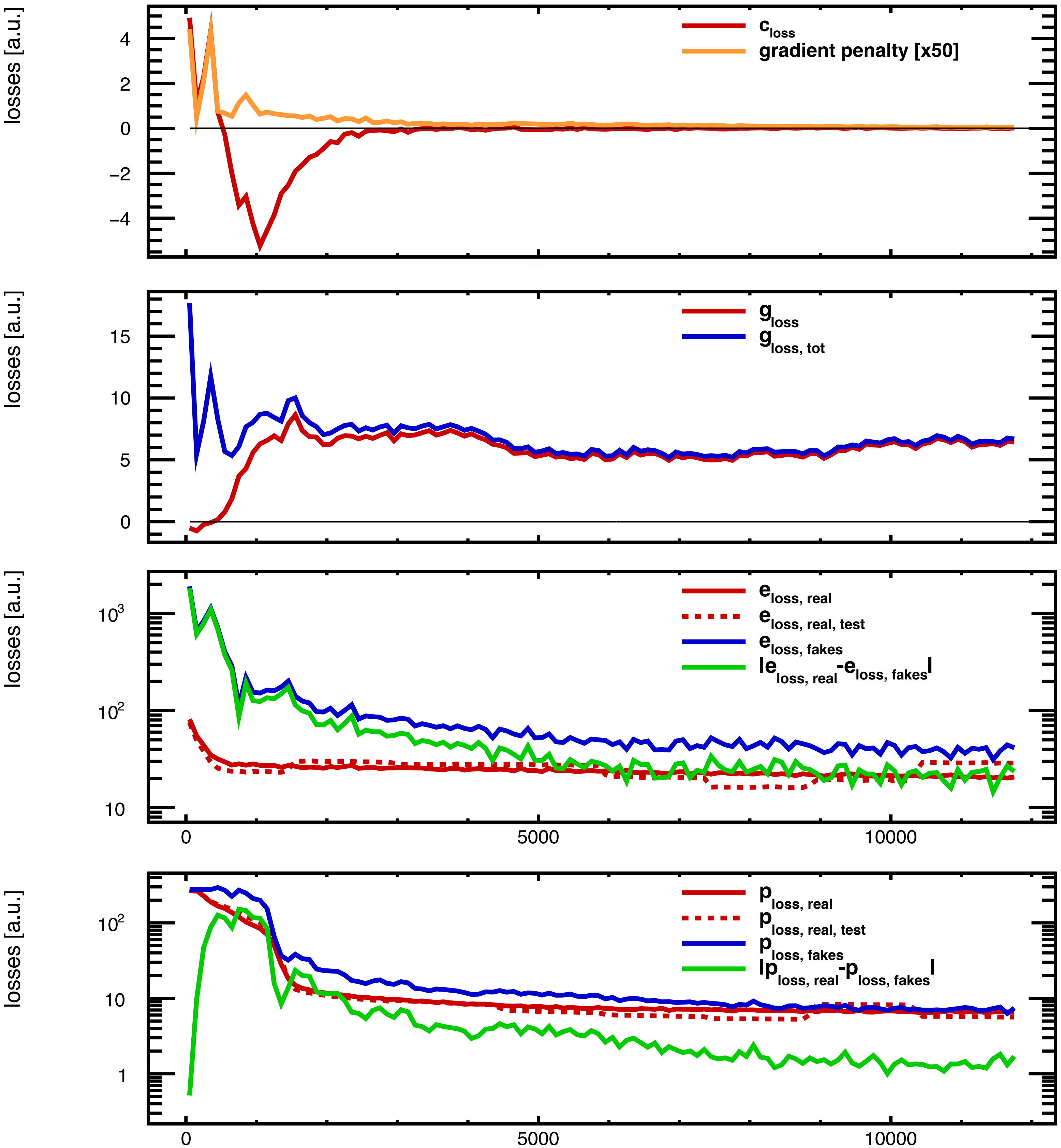


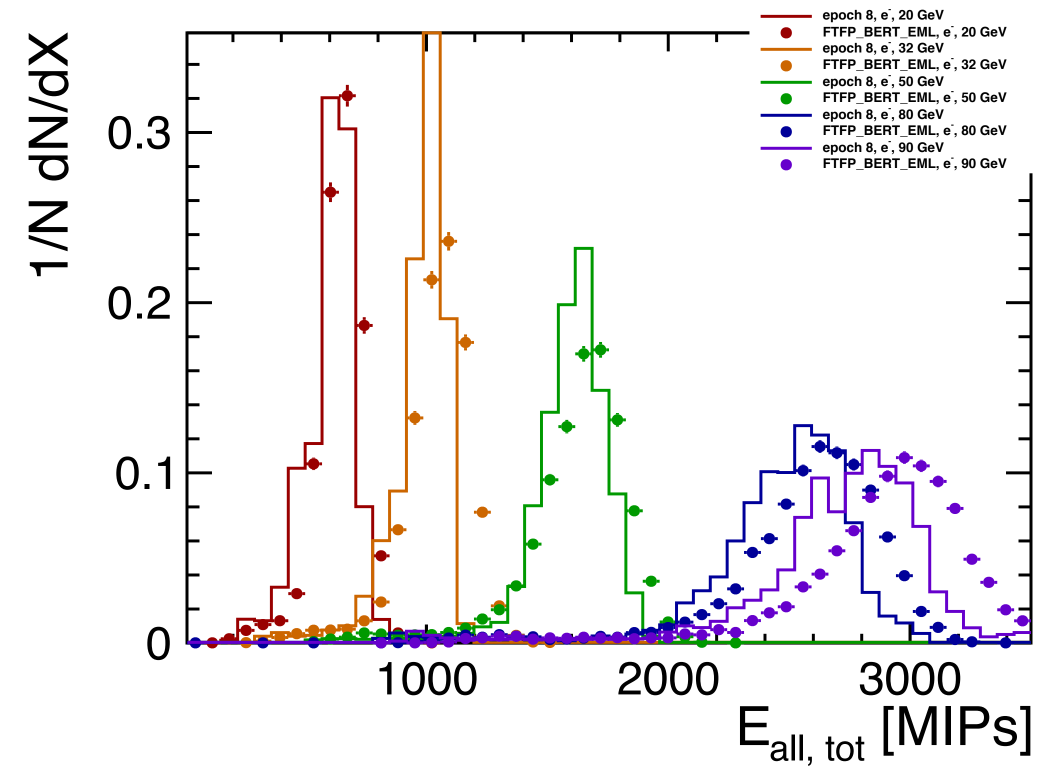


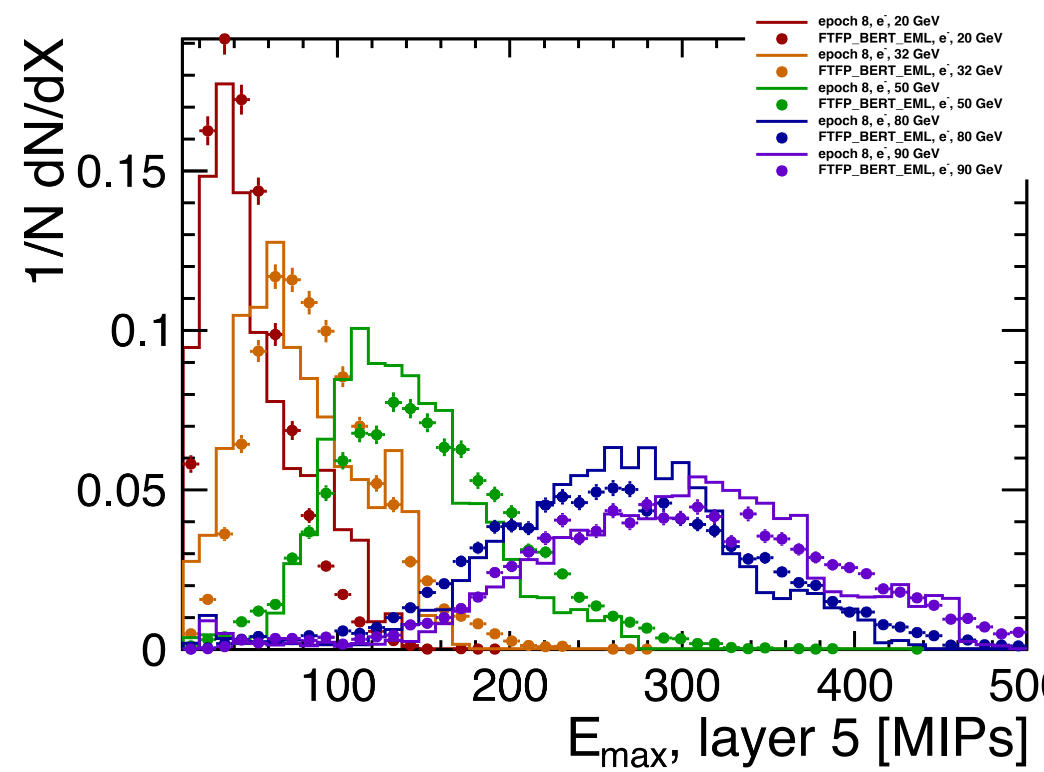
Generator network: “only 3D (de-) convolutions” with ~507k free parameters



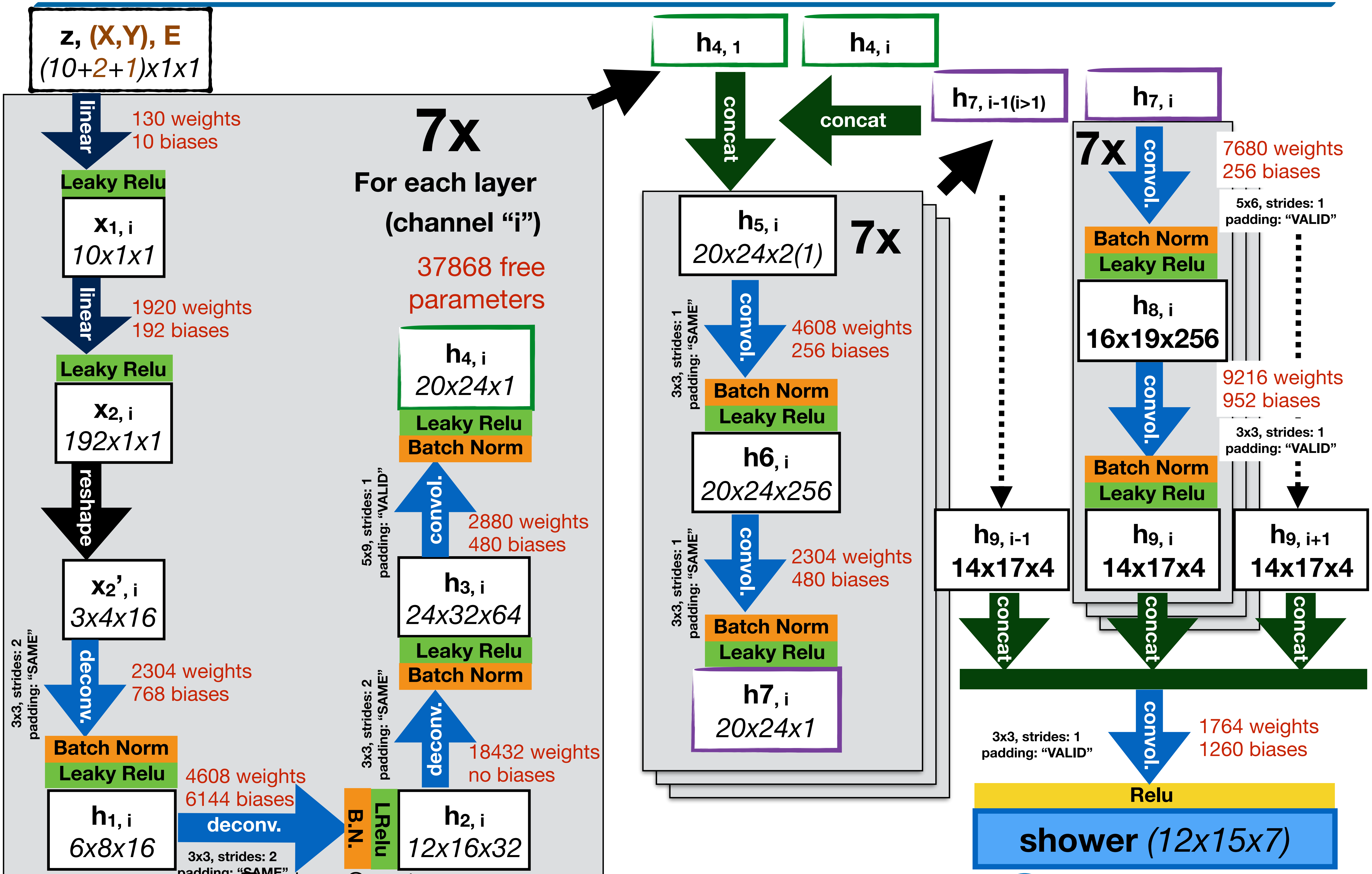
Generator network: “only 3D (de-) convolutions” with similar performance



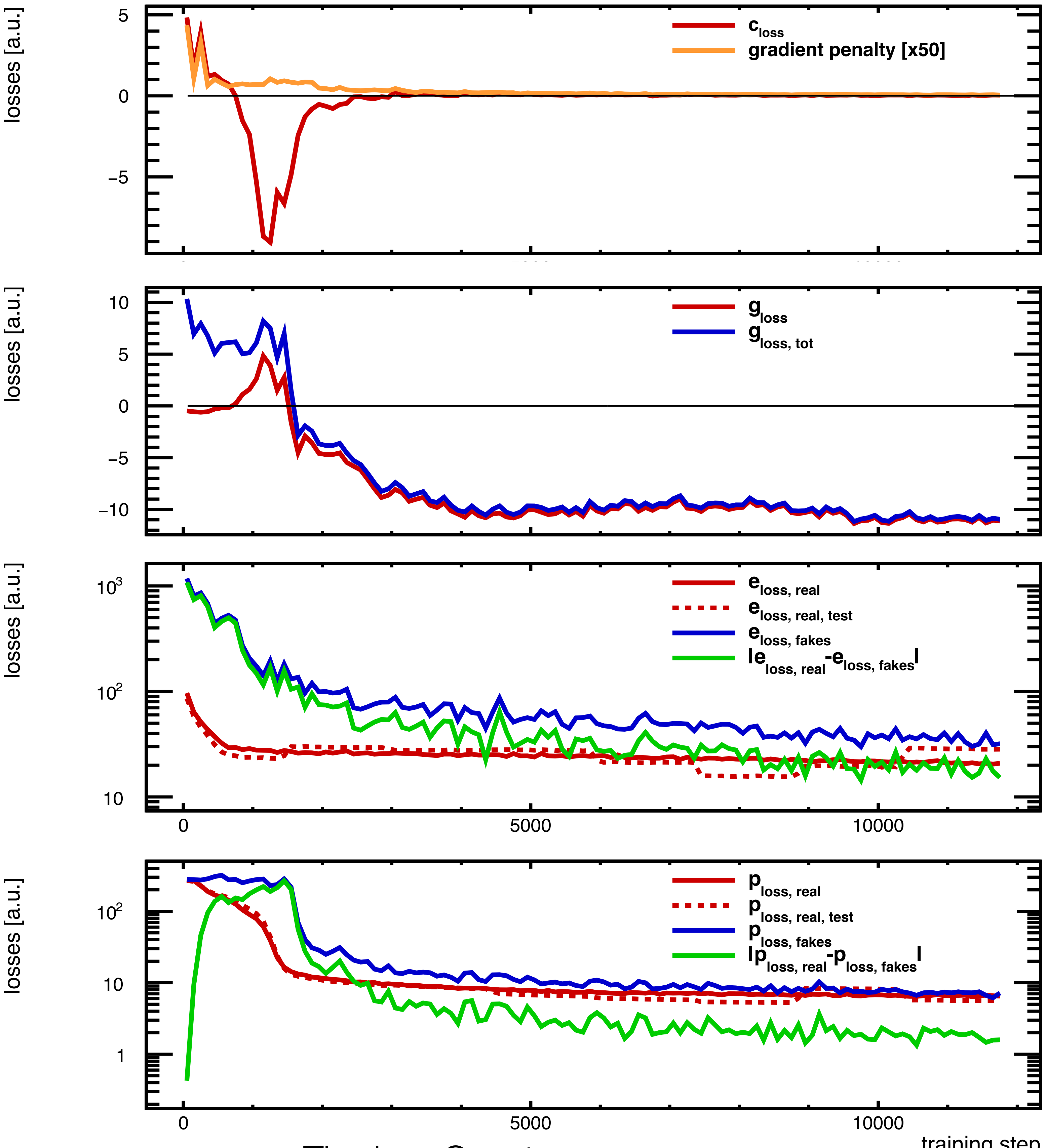


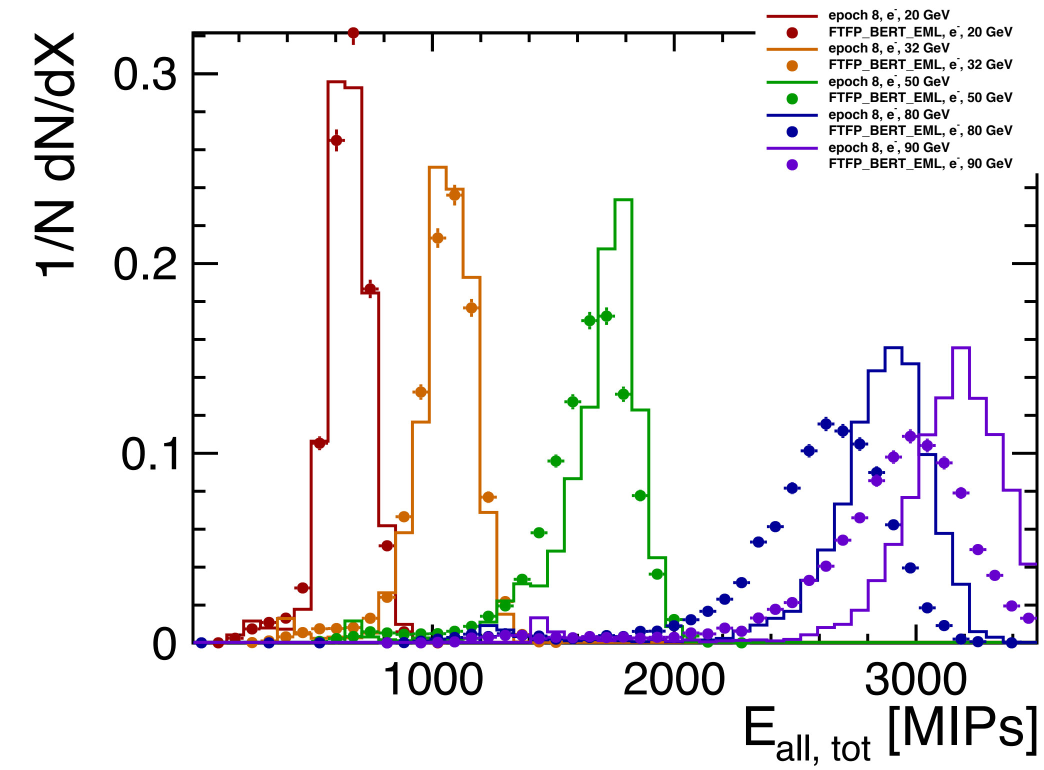


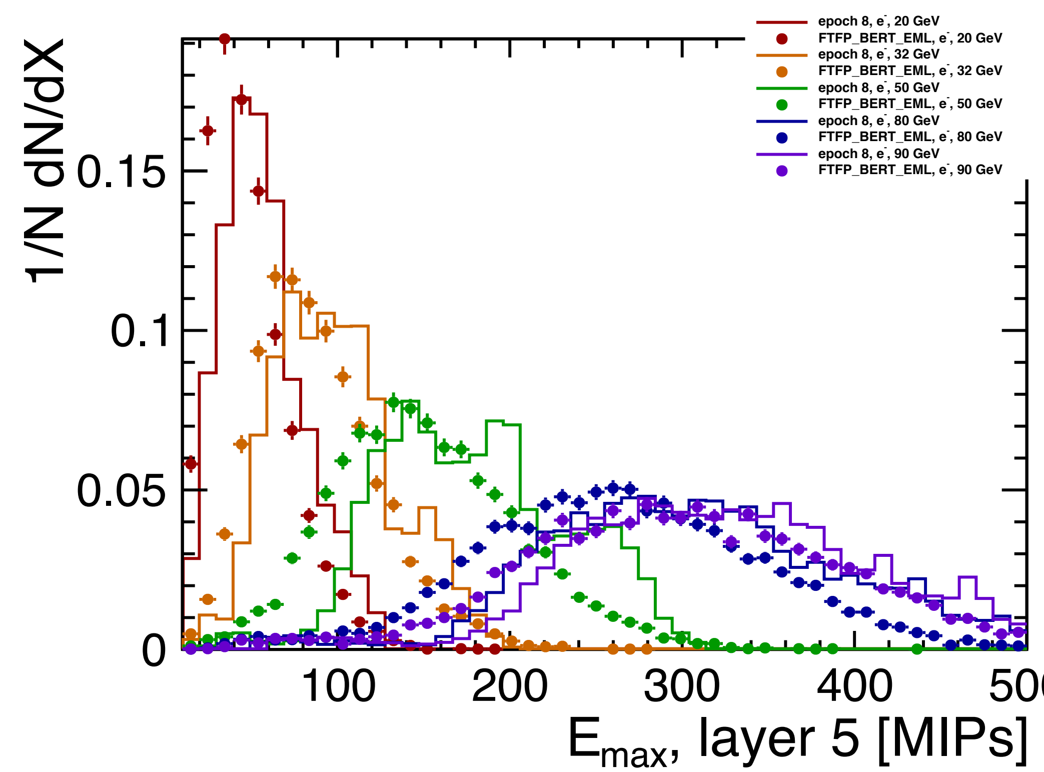
Generator network: “recurrent merging of layers” with ~448k free parameters



Generator network: “recurrent merging of layers” with similar performance







Backup - Refining Adversarial Network

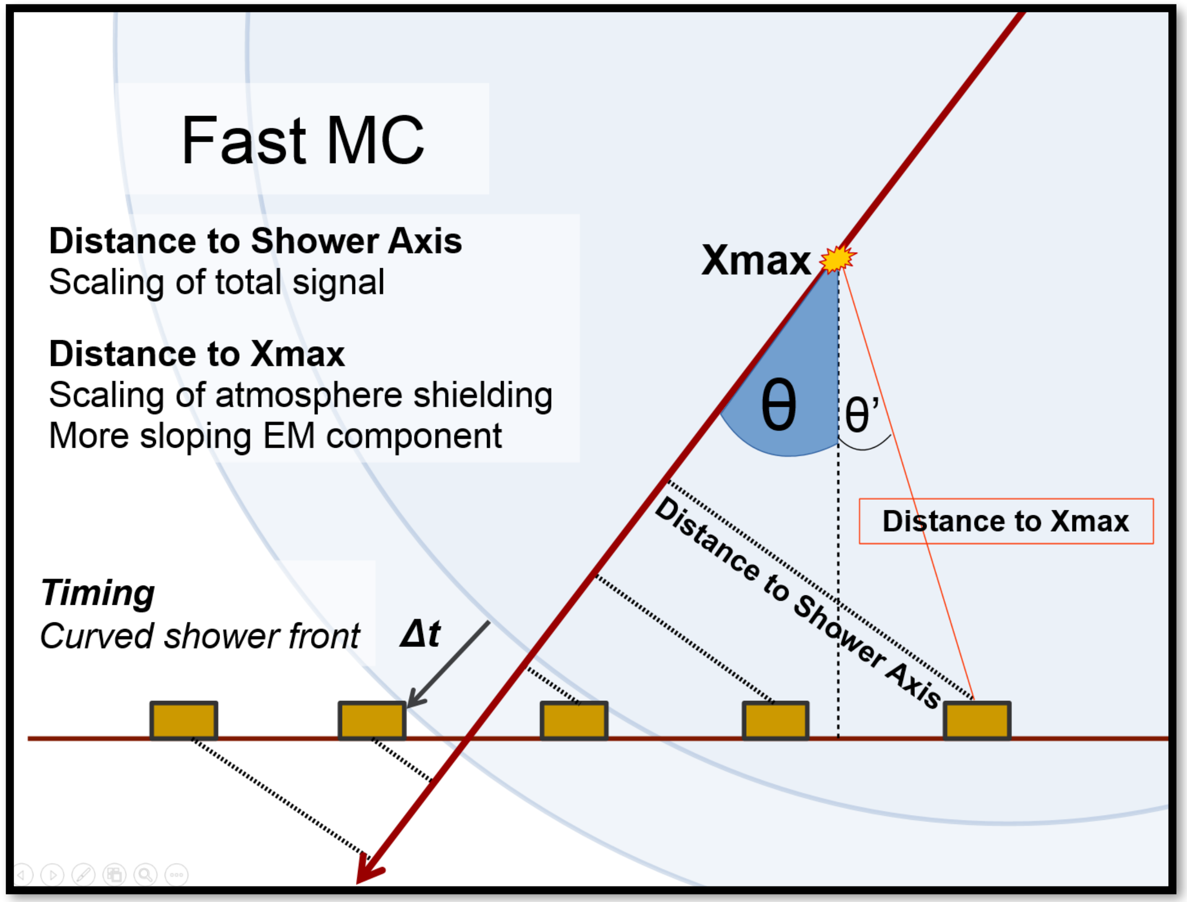
III. Physikalisches Institut

RWTH AACHEN UNIVERSITY

Parametrized Air Shower Simulation

Simulation of extensive air showers

- Scaling of signal
  - Distance to shower axis
  - Distance shower maximum Xmax
    - Shielding of atmosphere
- Different scaling
  - EM component
  - Muon component
- Time information
  - Planar shower front

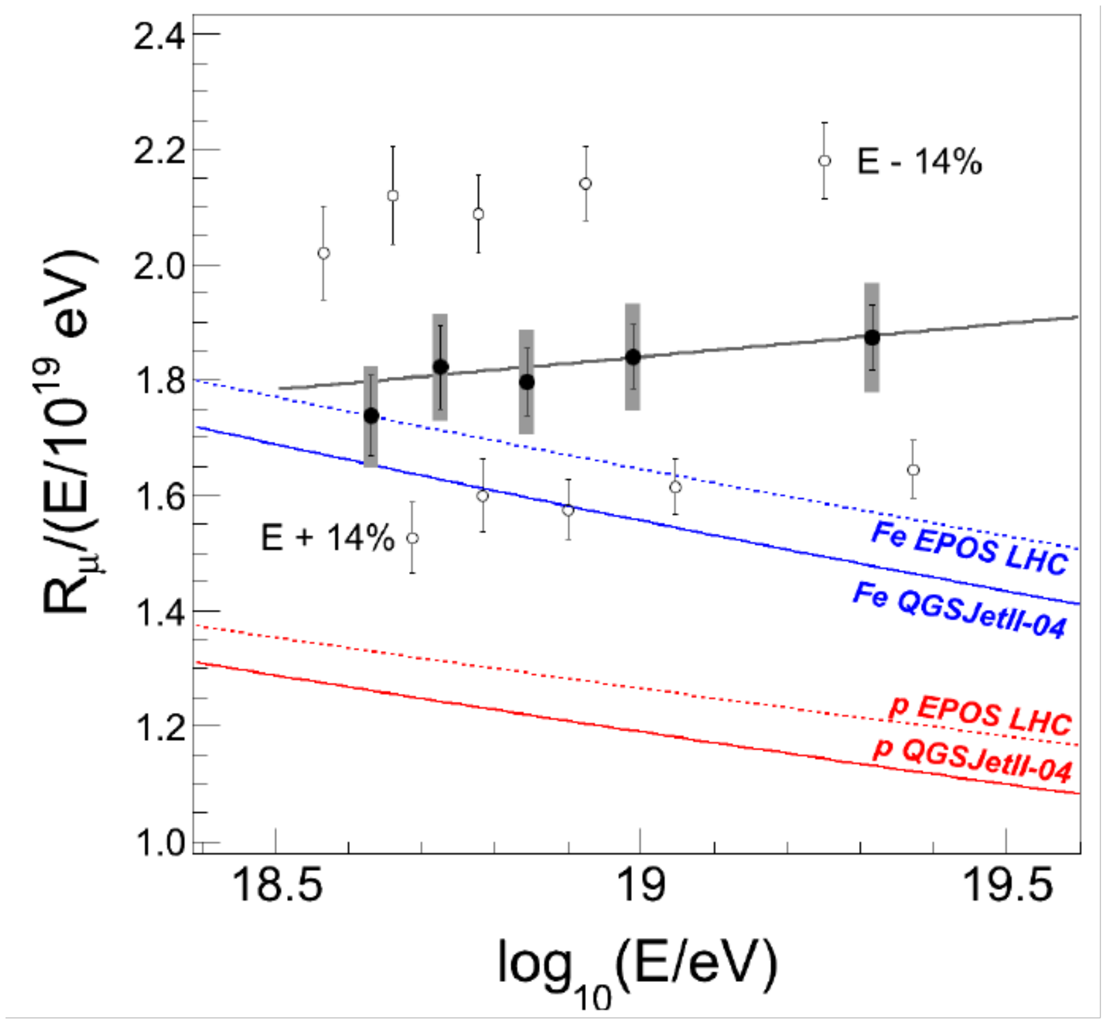


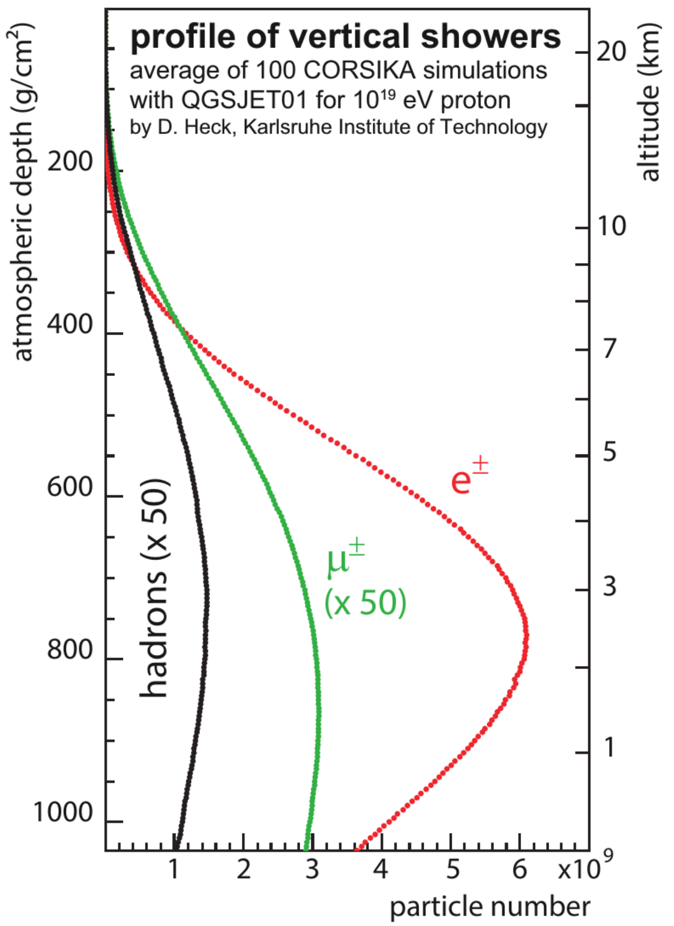
→ Large calorimenter
With single readout layer

GitLab goo.gl/8tB91r

Muon Component

- Muon deficit

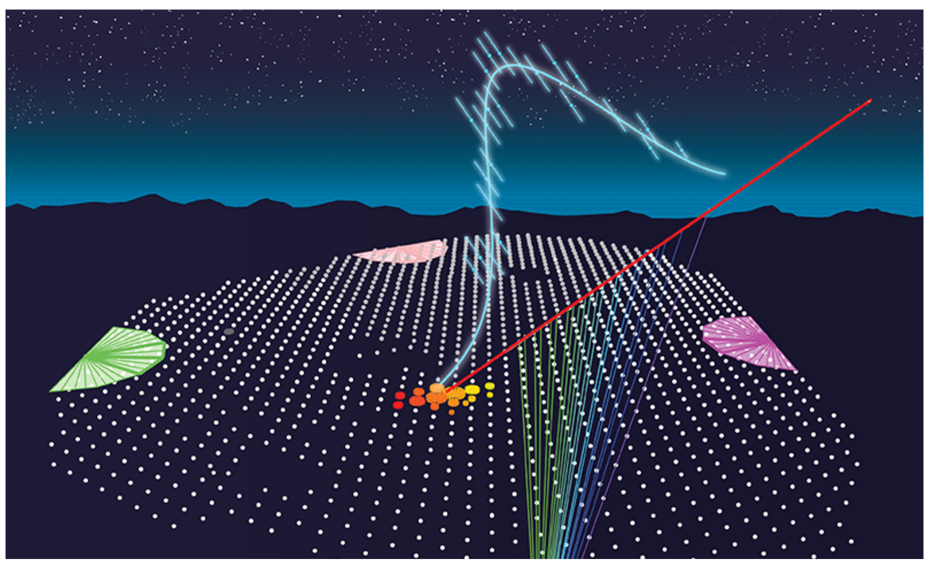


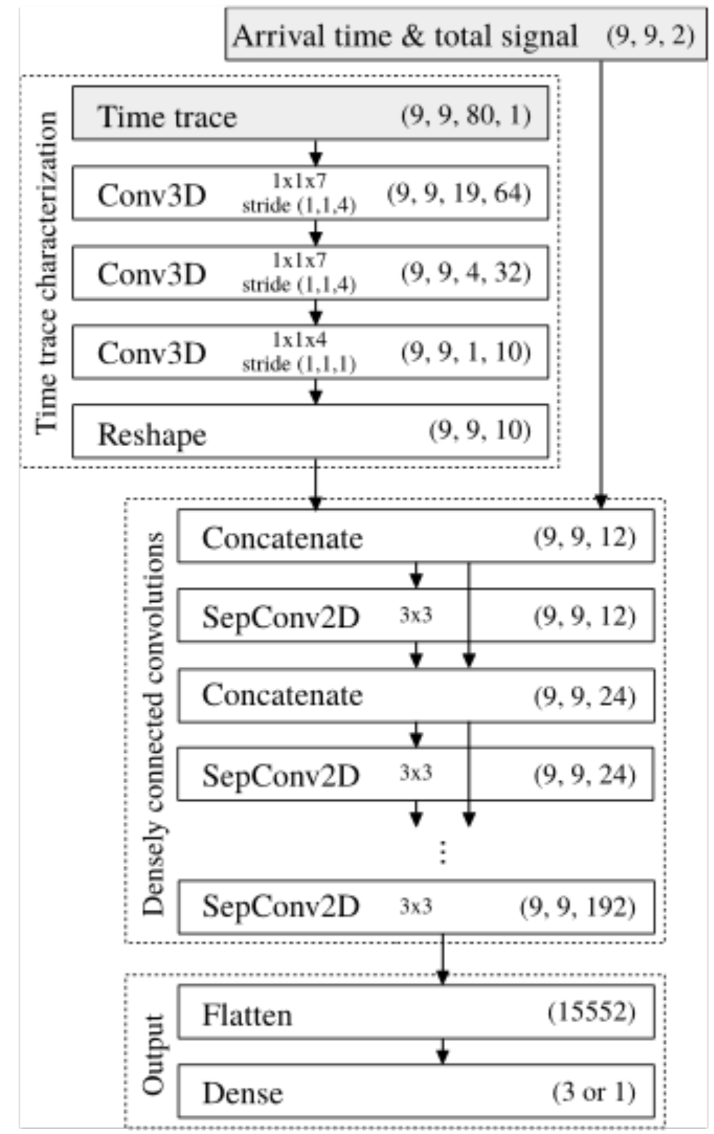


Backup

Network used for energy reconstruction of signal traces – arXiv:1708.00647

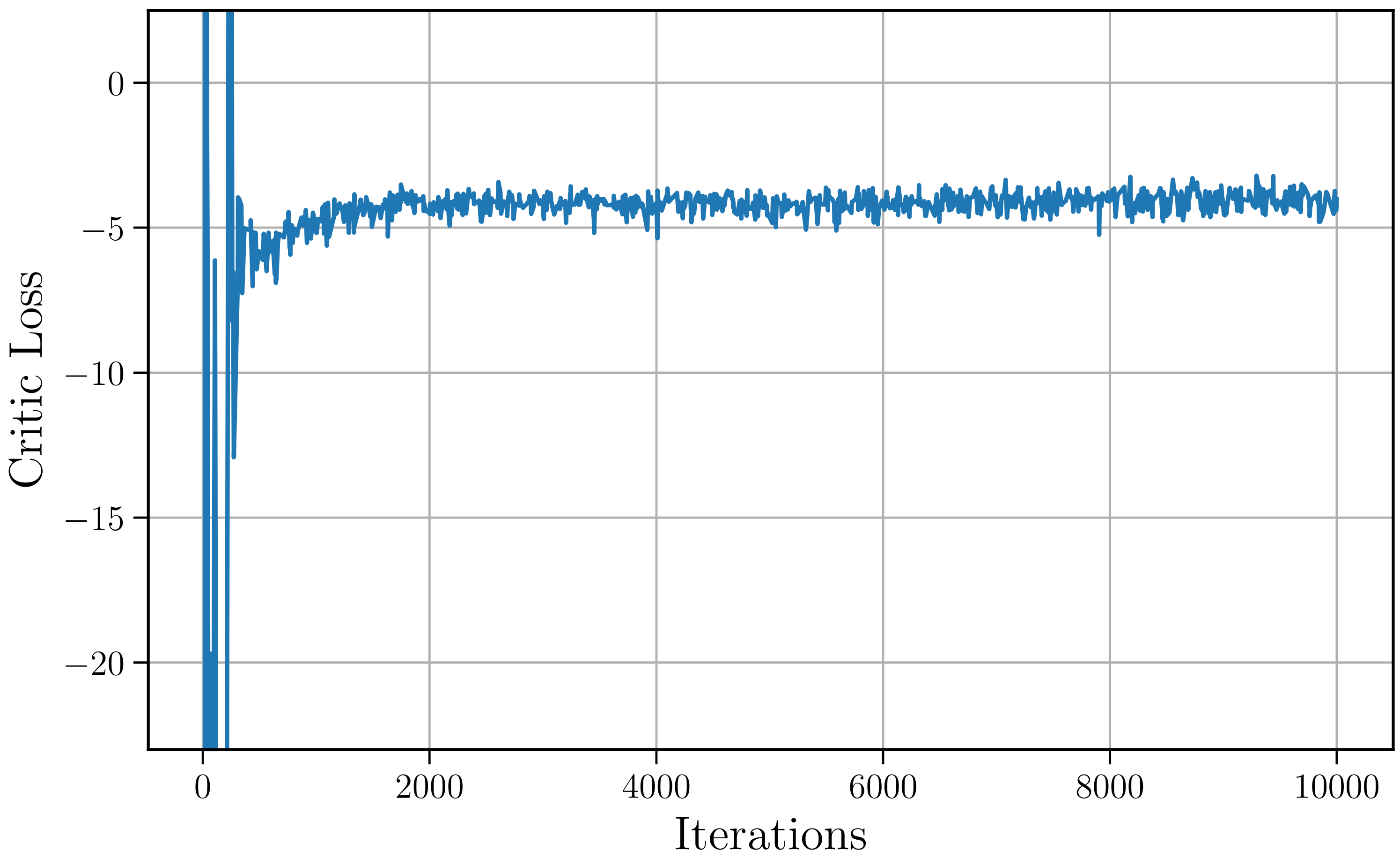
AixNet





Backup

- Loss of the critic network in the WGAN to refine signal traces



Jonas Glombitza
glombitza@physik.rwth-aachen.de

Backup

- Refiner network as used in the WGAN to refine signal traces

Merge Operation	Operation	Kernel	Feature Maps	Padding	Activation
		$9 \times 9 \times 80 \times 1$ Input			
Addition	Convolution	$1 \times 1 \times 7$	64	same	ReLU
	Convolution	$1 \times 1 \times 7$	64	same	ReLU
Addition	Convolution	$1 \times 1 \times 7$	64	same	ReLU
	Convolution	$1 \times 1 \times 7$	64	same	ReLU
Addition	Convolution	$1 \times 1 \times 7$	64	same	ReLU
	Convolution	$1 \times 1 \times 7$	64	same	ReLU
Addition	Convolution	$1 \times 1 \times 7$	64	same	ReLU
	Convolution	$1 \times 1 \times 7$	64	same	ReLU
	Convolution	$1 \times 1 \times 1$	1	same	ReLU
		$9 \times 9 \times 80 \times 1$ Output			