Quantum Machine Learning for HEP Detector Simulations

Terrascale 2021

Florian Rehm [CERN openlab, RWTH Aachen]

Sofia Vallecorsa [CERN openlab], Kerstin Borras [DESY, RWTH Aachen],
Dirk Krücker [DESY], Simon Schnake [DESY]

23.11.2021

Future Simulations

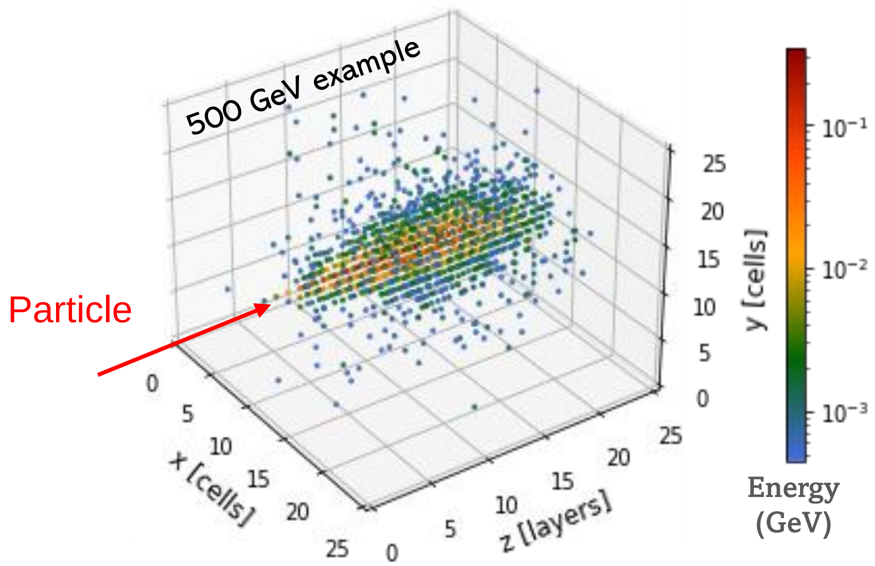
Alternative Calorimeter Monte Carlo Simulations

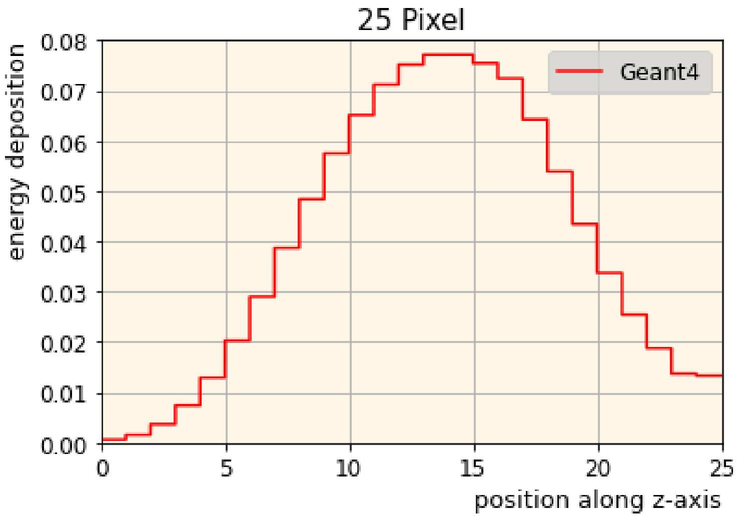
- Previously: **Deep Learning**
  - → Developed a Deep Learning approach for calorimeter simulations which requires fewer computing resources compared to Geant4
    - DL GAN (up to 160 000x speed up)
    - More information about this in the following presentation

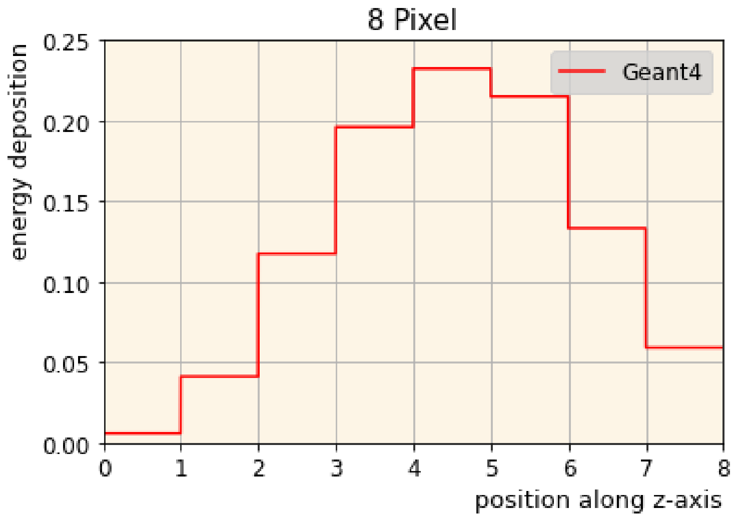
- Now: **Explore potential of quantum computing**
  - Make use of quantum properties (entanglement, superposition)
  - Hope to solve problems faster and / or more accurately
  - “Quantum Advantage” not yet reached → only initial investigations
    - Using simplified models
    - Understanding advantages and challenges

Calorimeter Training Data

- 3D particle shower images
- Average the image over z-axis → 1D image
- Down sample to only 8 pixel
- Average of all input energies



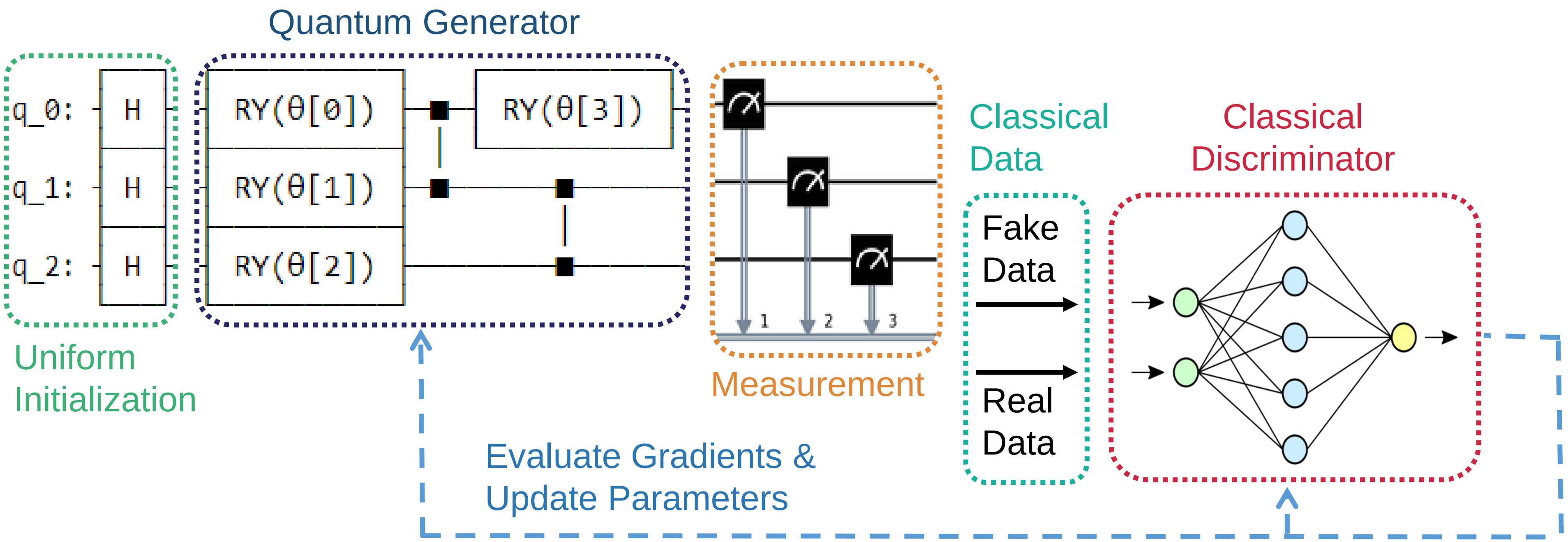




Hybrid qGAN

Quantum Generative Adversarial Networks

- Hybrid quantum – classical ansatz for generating calorimeter shower images

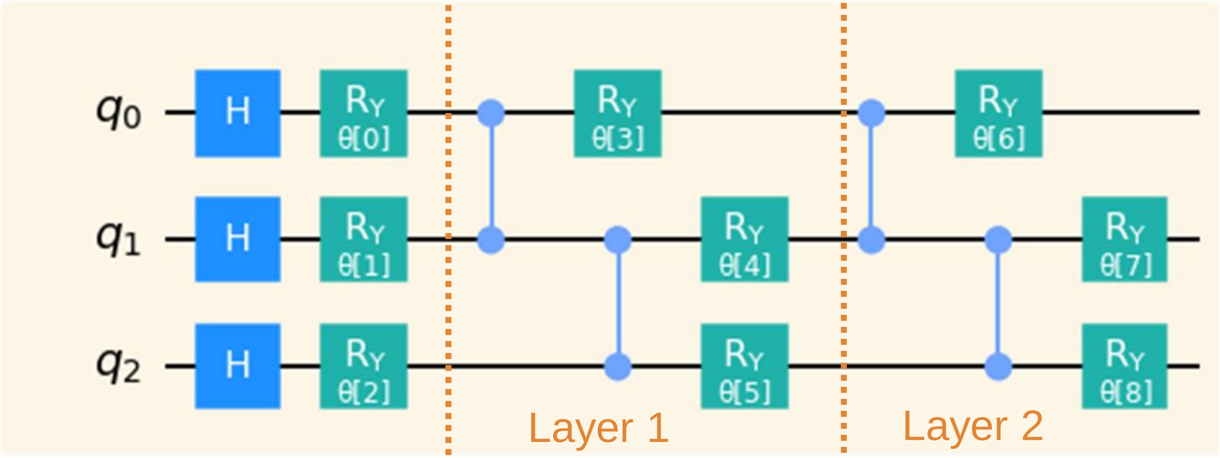


1D Quantum GAN

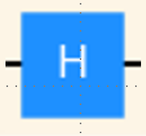
1D Quantum Generator Circuit

- Modified a Qiskit qGAN model developed by IBM
- 1D 8-pixel images
  - Amplitude encoding: 3 qubits ($2^3 = 8$ states) in quantum generator circuit

Quantum Generator Circuit:

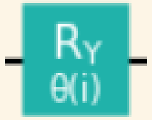


Hadarmard Gate



$$= \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

Y-Rotational Gate



$$= \begin{pmatrix} cos\left(\frac{\theta}{2}\right) & -sin\left(\frac{\theta}{2}\right) \\ sin\left(\frac{\theta}{2}\right) & cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

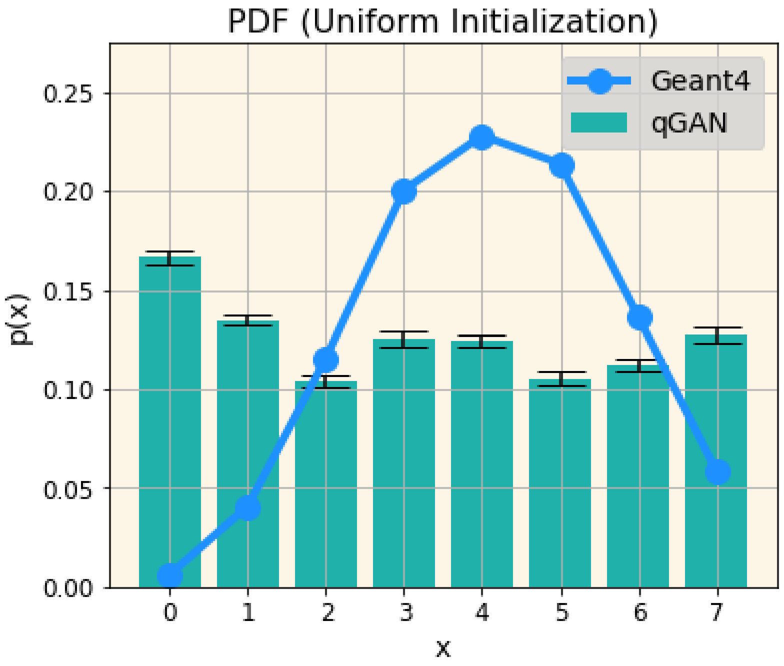
Controlled-Z Gate

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$

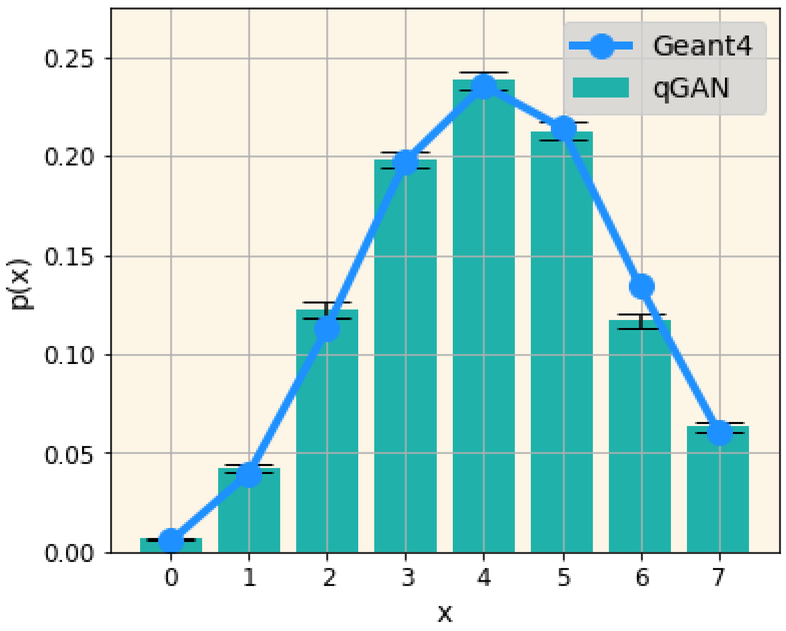
1D Training without Noise

- Simulating the quantum computer on a classical computer

- Hyperparameter search reduced training time and increased accuracy



Uniform Initialization

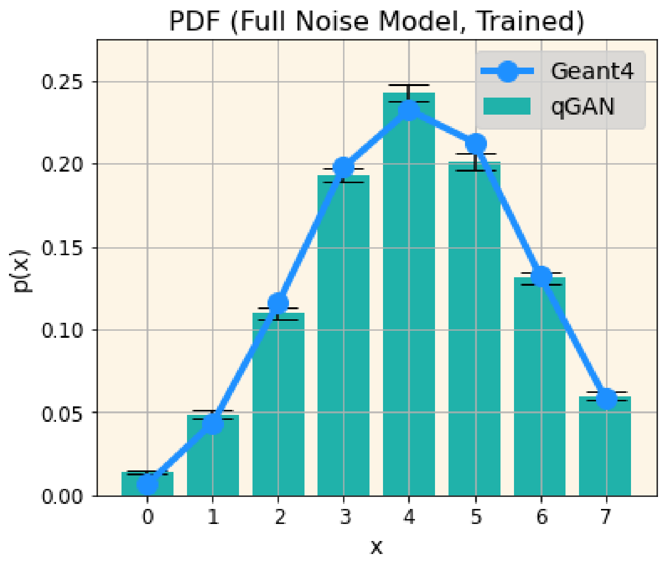


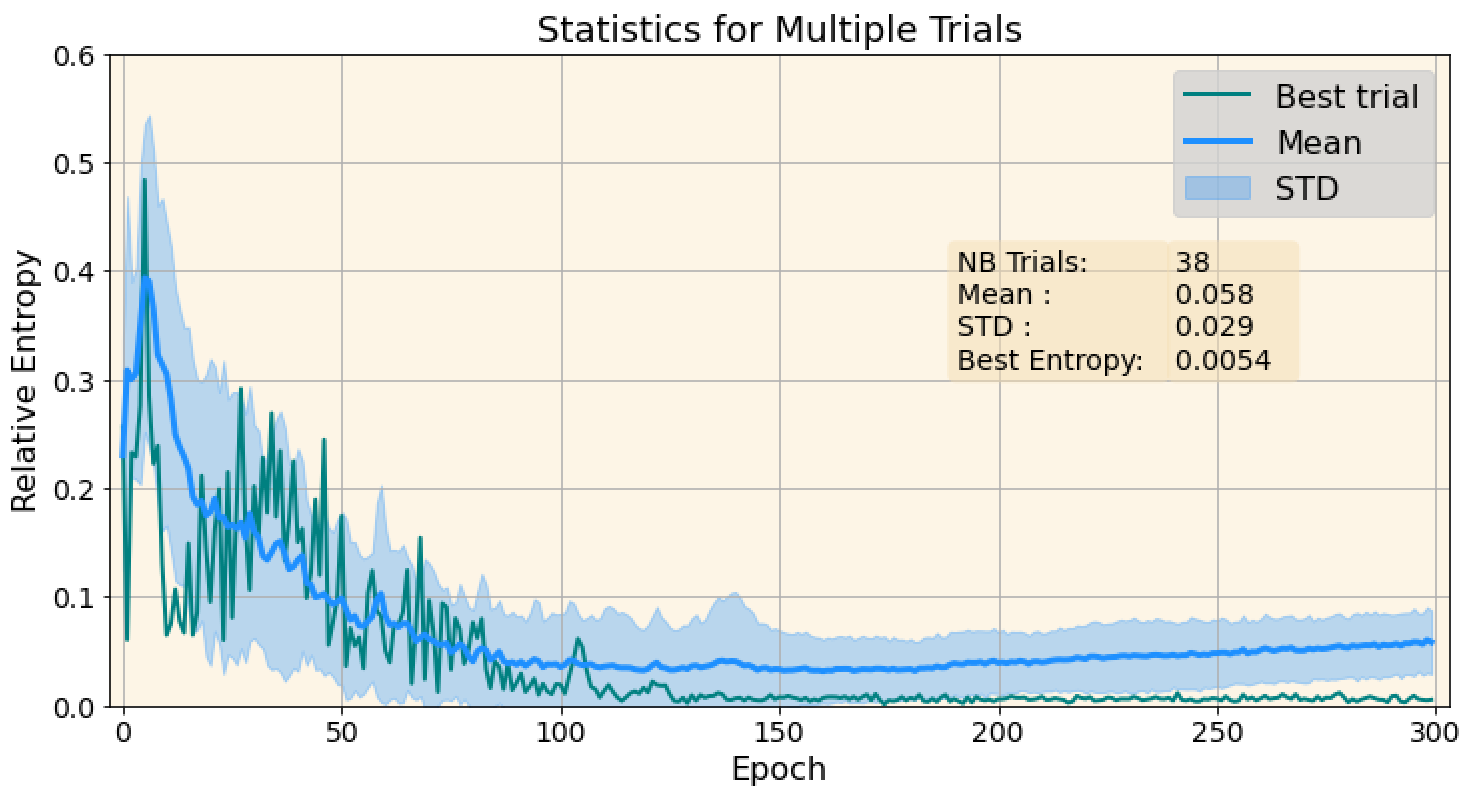
Trained Model

→ Good results

1D Training with Noise

- Custom Noise Model:
  - 2.5% readout noise
  - 1.5% gate-level noise
- Same hyperparameters

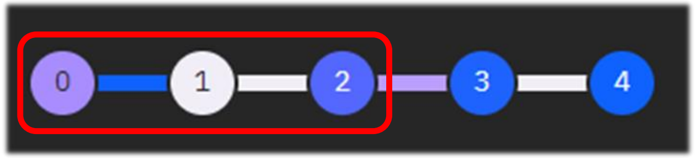




→ Good accuracy
→ Training could have stopped earlier

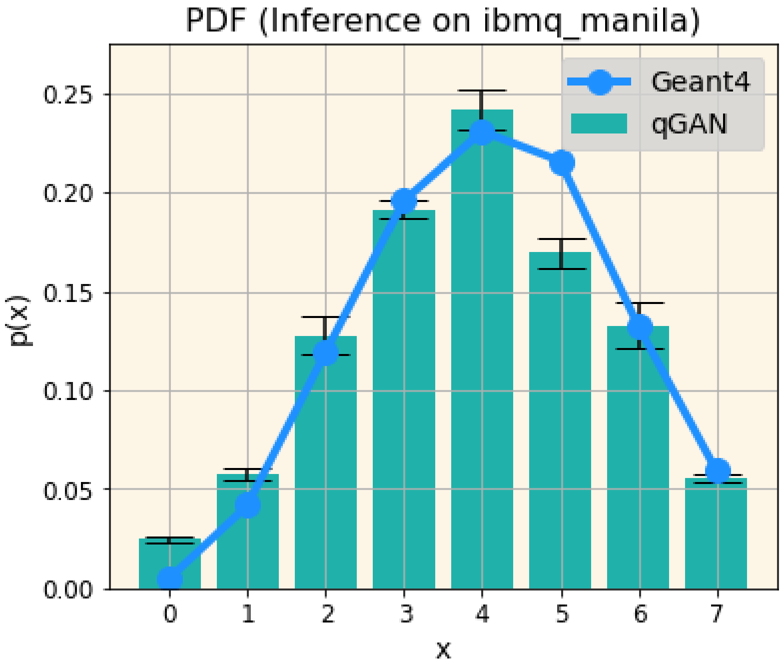
1D Inference on Real Hardware

- Run on IBMQ Manila Quantum Computer



- Different noise level than in training

Qubit Number	0	1	2	Average
Readout Error	2.34%	2.66%	2.05%	**2.35%**
CX-gate Error	1.11%		1.75%	**1.43%**

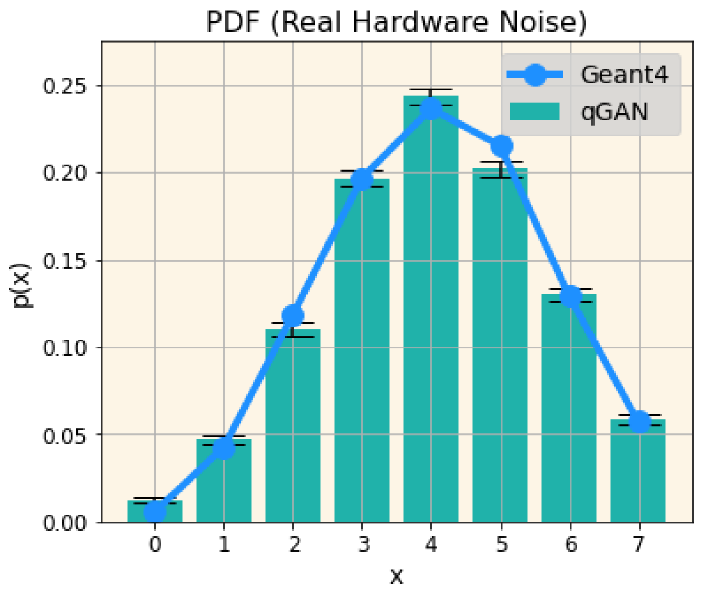


→ Good accuracy

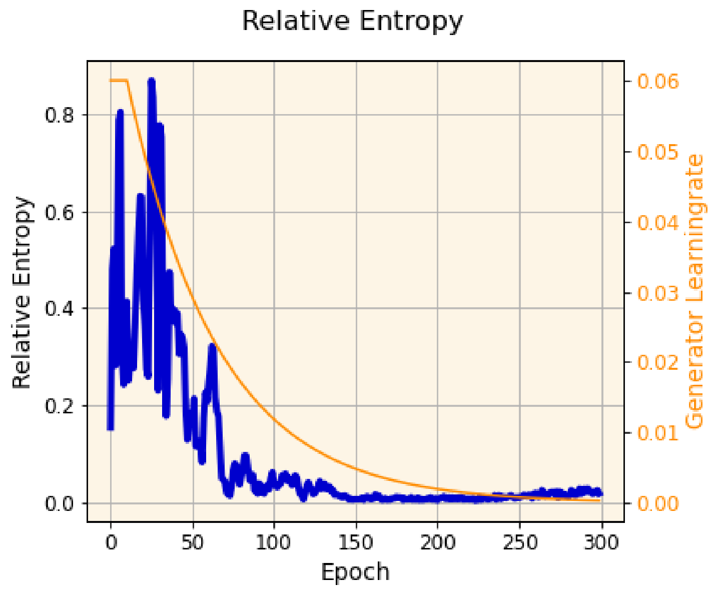
1D Training with Hardware Noise

- Training on simulator with real IBMQ Manila noise model

Qubit Number	0	1	2	Average
Readout Error	2.34%	2.66%	2.05%	**2.35%**
CX-gate Error	1.11%		1.75%	**1.43%**



→ No decrease in accuracy



→ Fast convergence

1D qGAN Future Work

- More tests with the full noise model
  - Does the training benefit from the noise?
  - Test error mitigation techniques

- Conditional qGAN
  - → Search for new model

- Run training on real quantum hardware

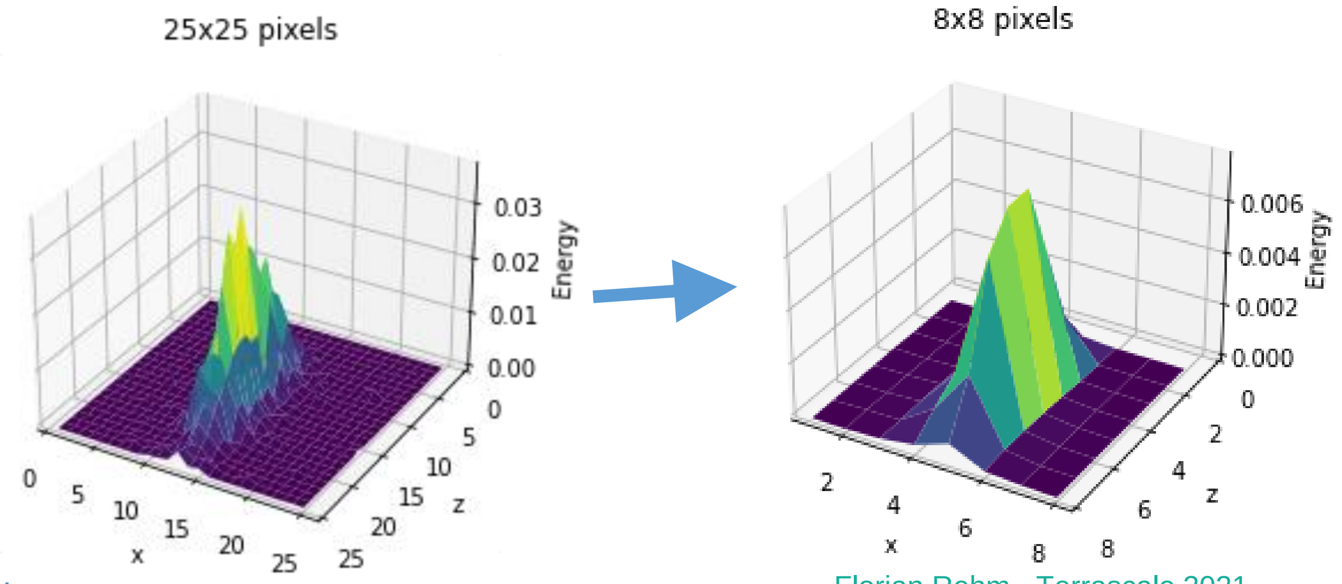
2D Quantum GAN

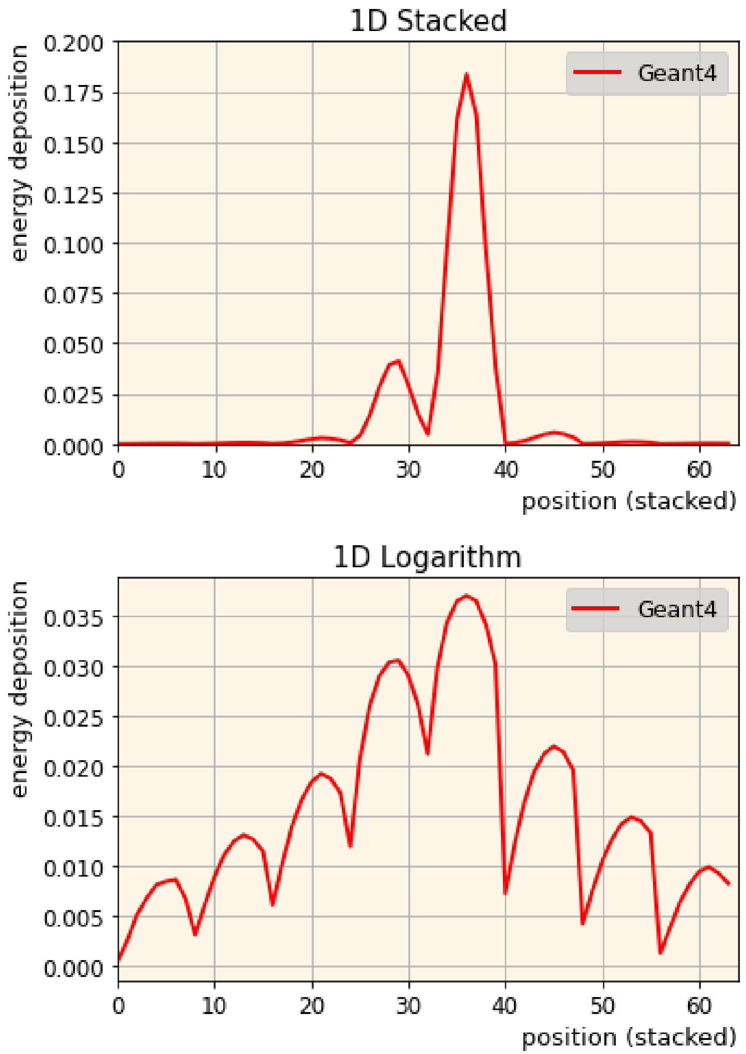
2D qGAN

2D Data Representation

2D: 8x8 pixel images

1. Down sample
2. 1D stacking
3. Apply logarithm

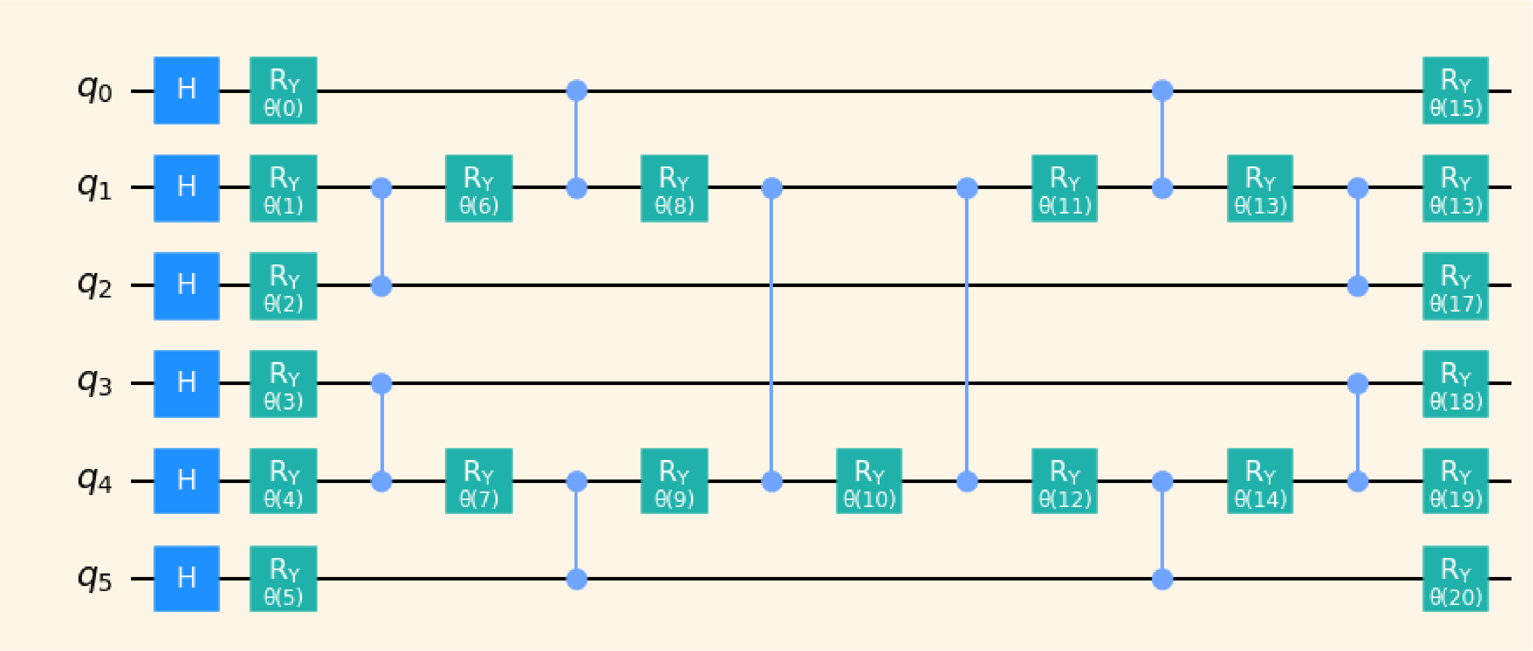


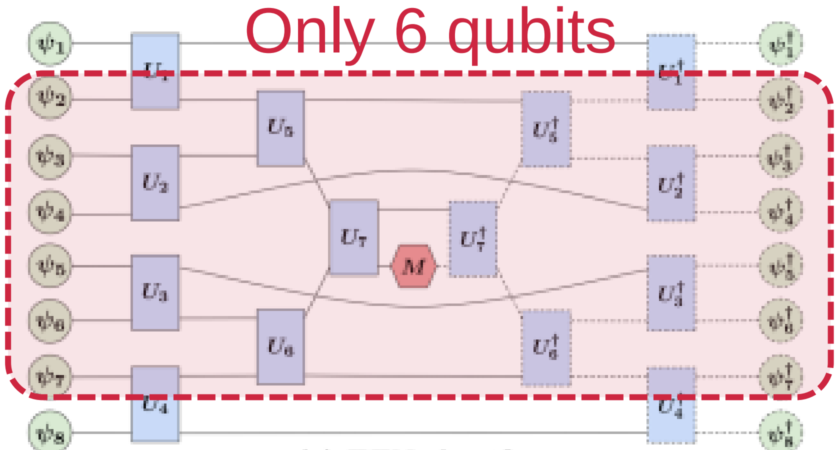


2D Quantum Generator Circuit

Tree Tensor Network Architecture

64 pixels $= 2^6$ → **6 qubits** for amplitude encoding





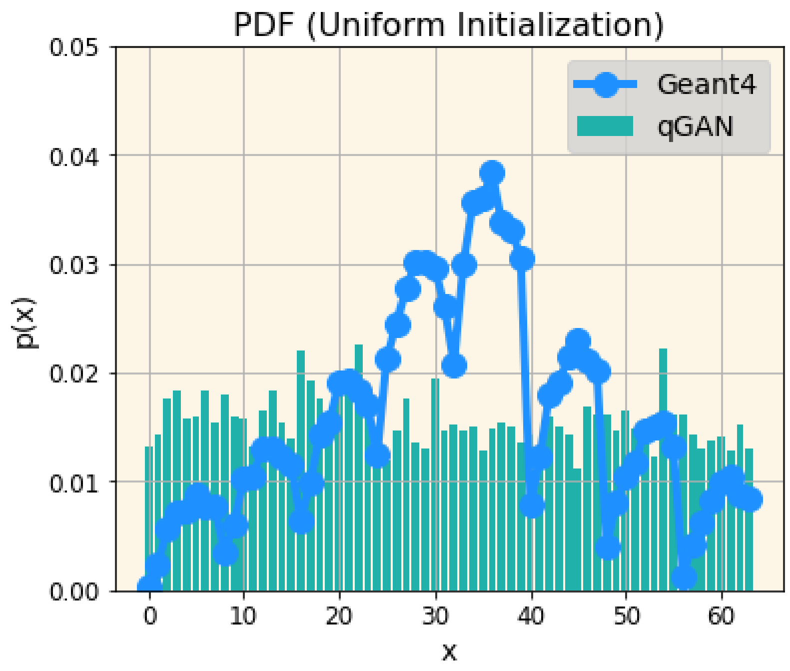
(a) TTN classifier

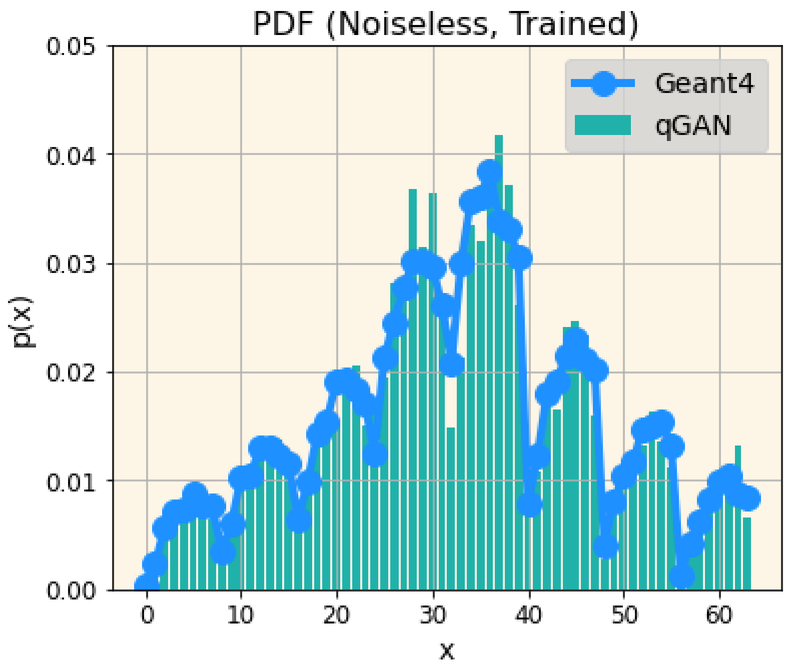
Grant, E., Benedetti, M., Cao, S. *et al.* Hierarchical quantum classifiers. *npj Quantum Inf* **4,** 65 (2018). https://doi.org/10.1038/s41534-018-0116-9

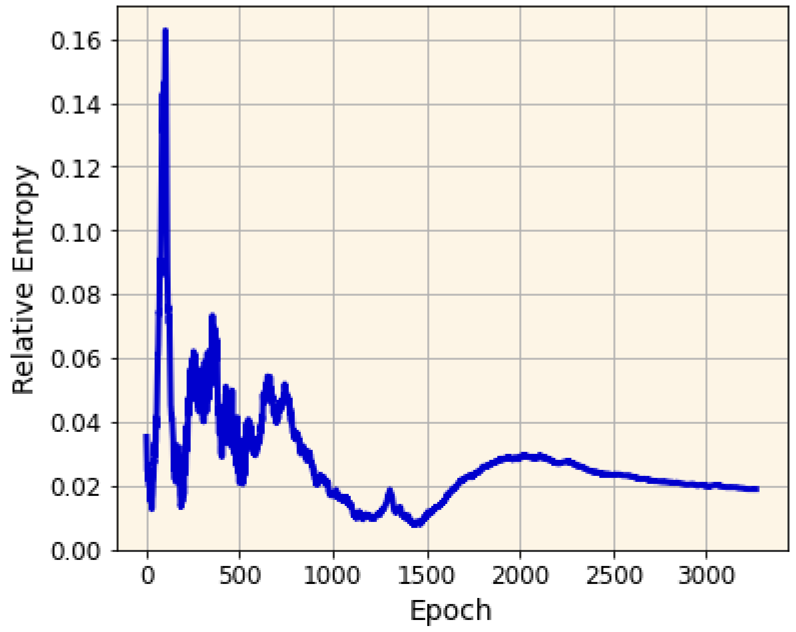
2D qGAN

Best Training Results

- Run on quantum simulator **without** noise







→ Good results

2D qGAN Future Work

- 2D qGAN:
    - Improve training convergence
        - Rare that training converges
    - Decrease training time: recently ~5 days
        - Hyperparameter optimization

Thank you for Listening

Quantum Machine Learning for HEP Detectors Simulations

Florian Rehm [CERN openlab, RWTH Aachen]

Sofia Vallecorsa [CERN openlab], Kerstin Borras [DESY, RWTH Aachen],
Dirk Krücker [DESY], Simon Schnake [DESY]