Quantum Generative Adversarial Networks

IML Workshop

Su Yeon Chang (EPFL, CERN)

Steven Herbert (CQC), Sofia Vallecorsa (CERN) , Elias Combarro (U. Oviedo, CERN), Federico Carminati (CERN)

22/10/2020

CERN openlab EPFL CAMBRIDGE QUANTUM COMPUTING

MOTIVATION

Why Quantum Generative Adversarial Networks (GAN)?

Calorimeters information :

- Used to understand low distance processes occurring during the particle collision
- Tremendous amount of time required by Monte Carlo based simulation

→ **Generative Adversarial Networks**

Quantum Machine Learning :

- Compressed data representation in quantum states
- qGAN model constructed by IBM

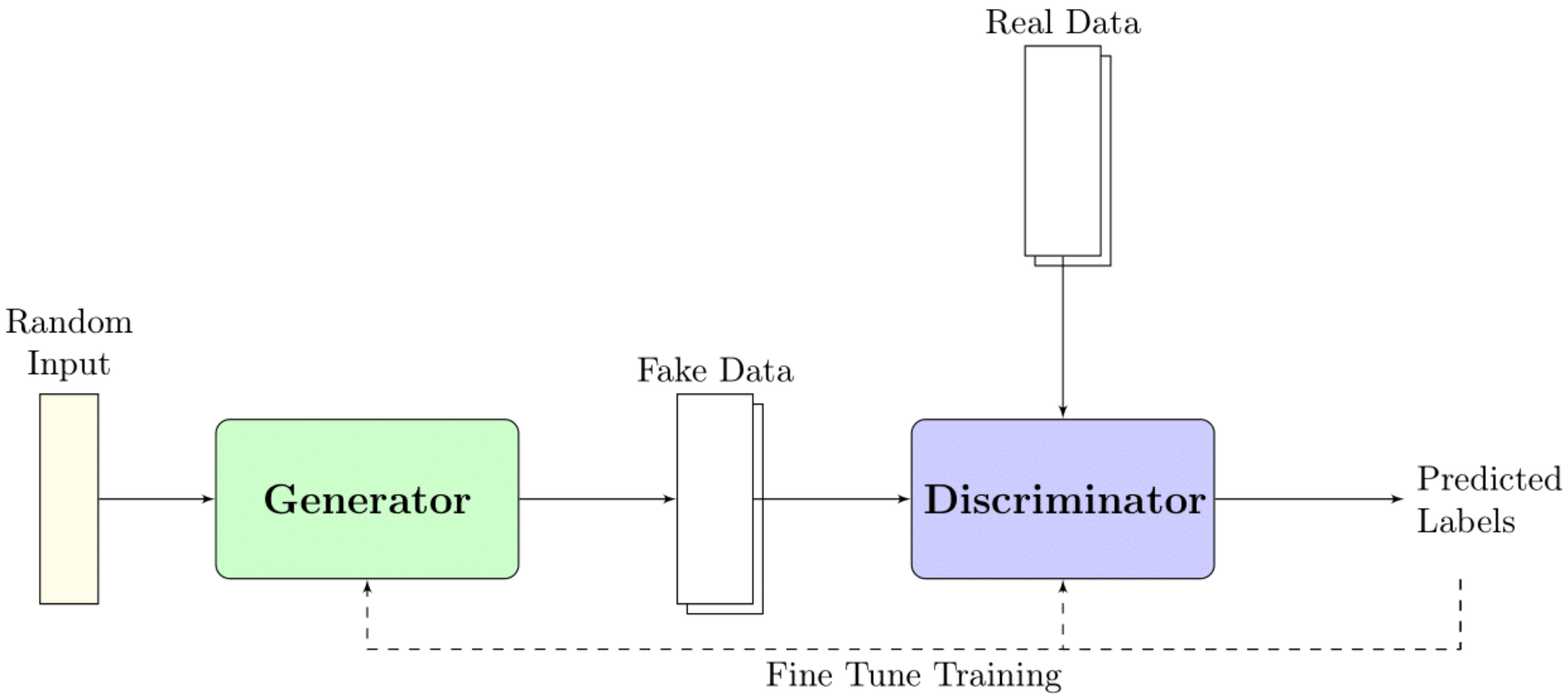
→ limited in reproducing a probability distribution over discrete variables

➡ **Explore different prototypes of quantum GAN to improve the model**

Generative Adversarial Networks (GAN)

Generative model with two neural networks

- **Generator** : Generates a fake output from a random noisy input
- **Discriminator** : Classify fake and real data



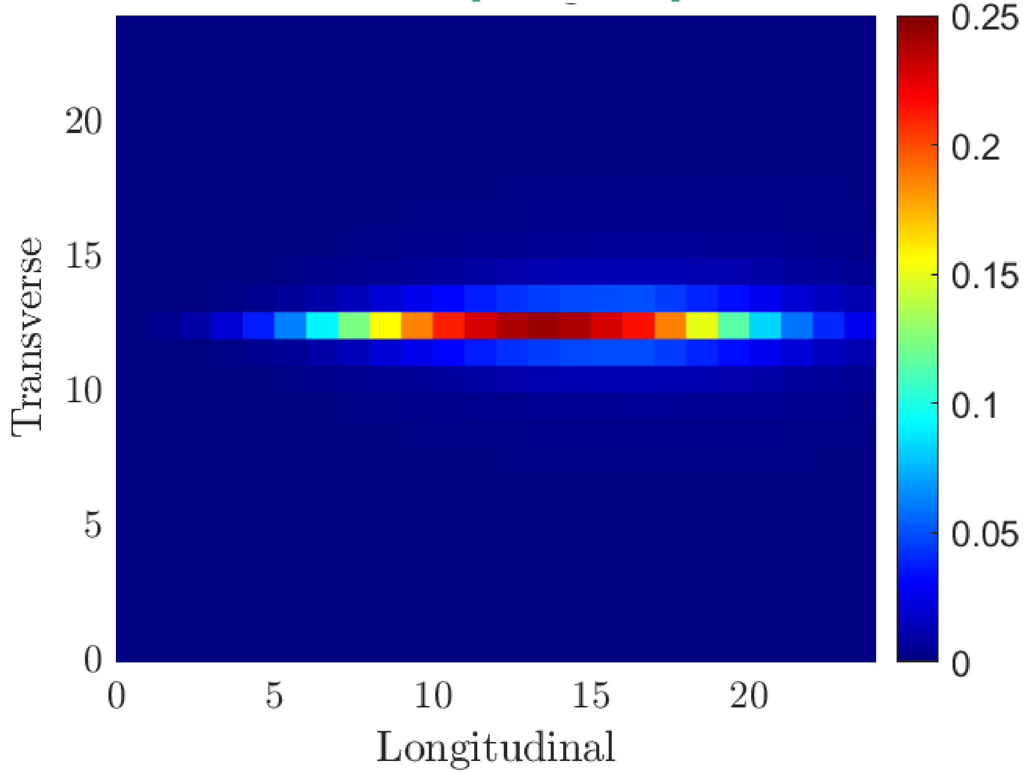
https://doi.org/10.1088%2F1742-6596%2F1085%2F2%2F022005

Application of GAN in HEP

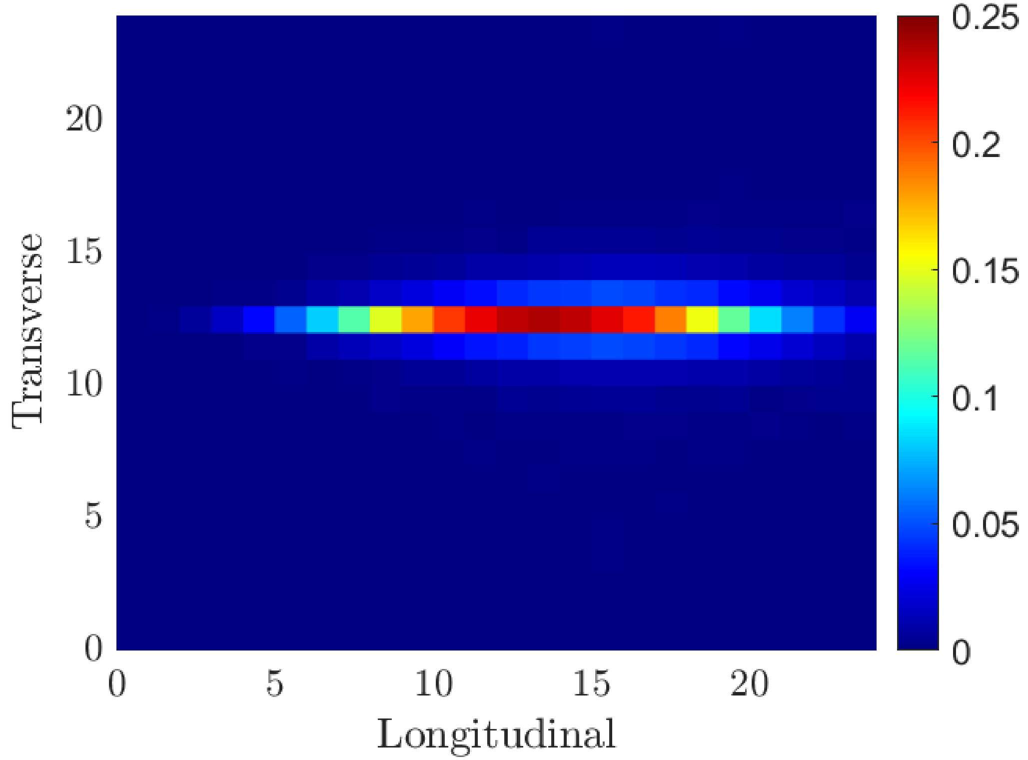
3DGAN

- 3D convolutional layers + Auxiliary regression task estimating the input particle energy
- Two-dimensional projection of 3D energy shower

Real (Geant4)



Generated (3DGAN)

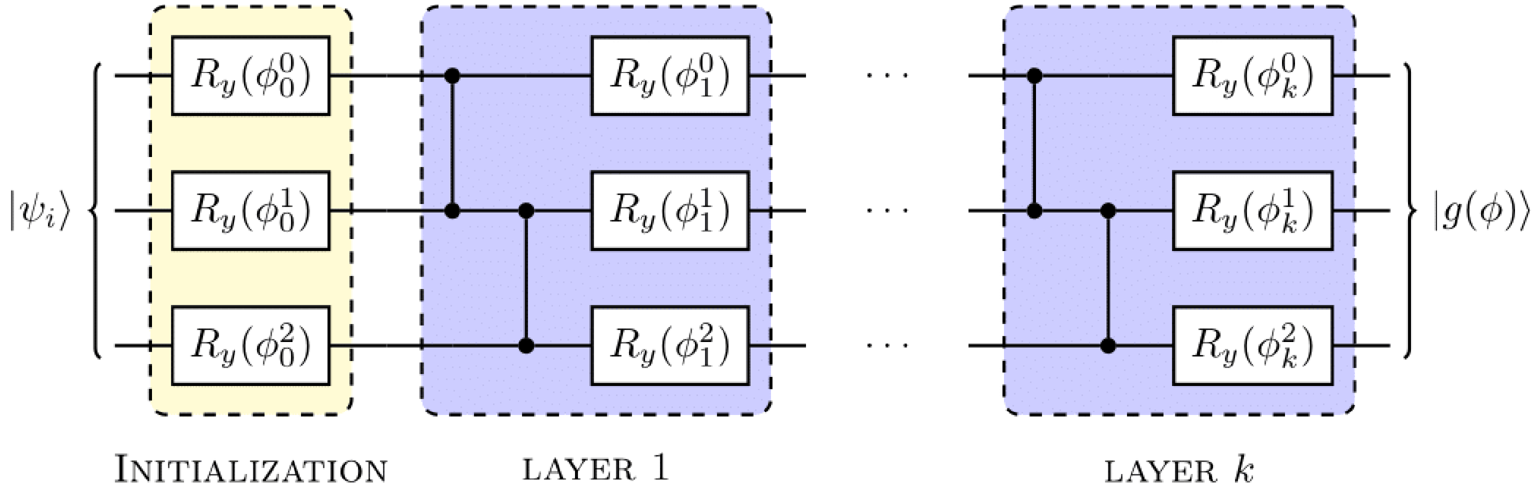


Quantum GAN

Practical qGAN model constructed by IBM

- Hybrid model : **Quantum** Generator + **Classical** Discriminator
- Efficient in loading and learning a probability over discrete values

$\rightarrow p_g(\phi)$ to approach p_{real}



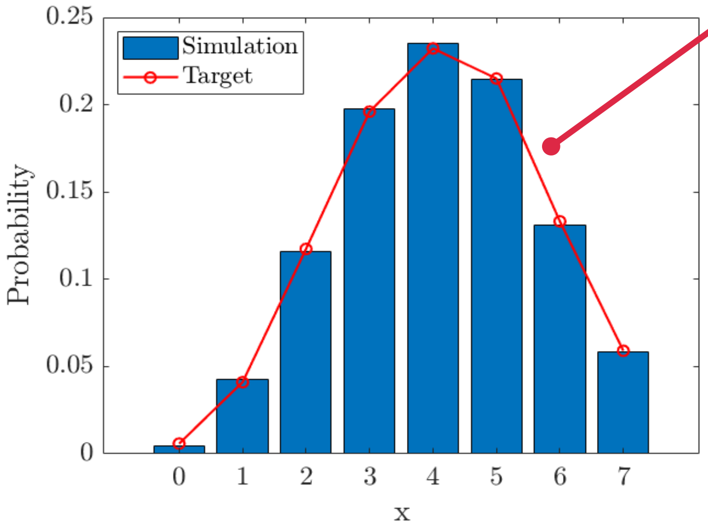
$$G_\phi|\psi_i\rangle = |g(\phi)\rangle = \sum_{i=0}^{N-1} \sqrt{p_g^i(\phi)}\,|i\rangle$$

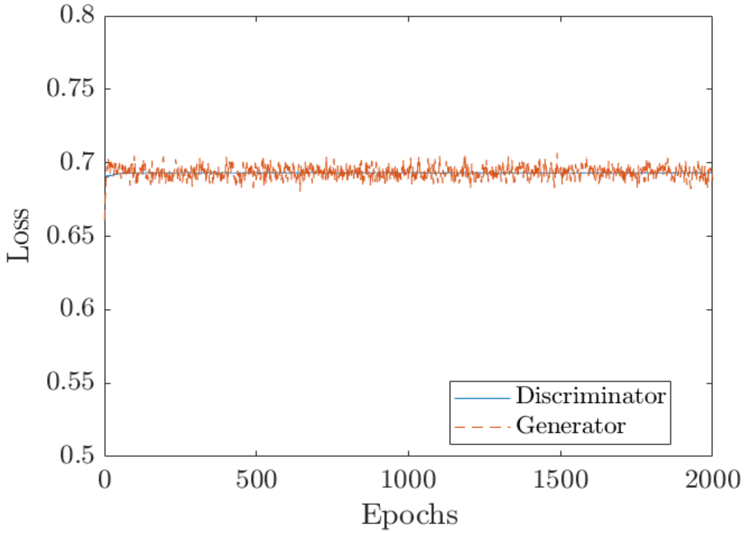
Application of qGAN in HEP

- ✓ 2D image summed **over longitudinal direction**
- ✓ Normalized & Binned into $3^2 = 8$ pixels
- ✓ Averaged over 20,000 samples

$\text{depth}_g = 3$

- Initial state normally distributed over $|0\rangle, \ldots, |7\rangle$
- Convergence in mean image & loss function



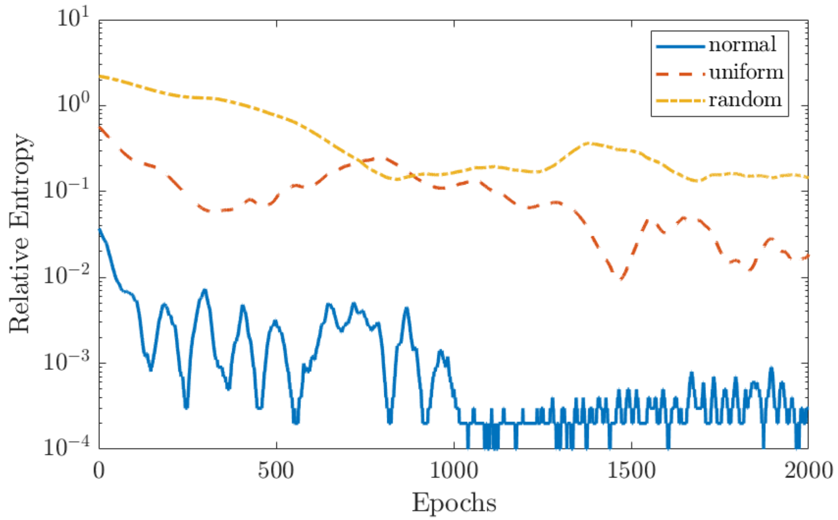


$\text{depth}_g = 3$ & Different initializations

- Relative entropy

$$D_{KL}(p||q) = \sum_{x \in X} p(x) \log\left(\frac{p(x)}{q(x)}\right)$$

- Quality of result depends on initial states



Limitation

IBM qGAN model

- Limited in reproducing an average probability distribution over pixels
- Aim to reproduce a distribution over continuous variables

→ **Need to find alternative ways to reproduce a “set” of images**

Dual-PQC GAN model (in collaboration with Cambridge Quantum Computing)

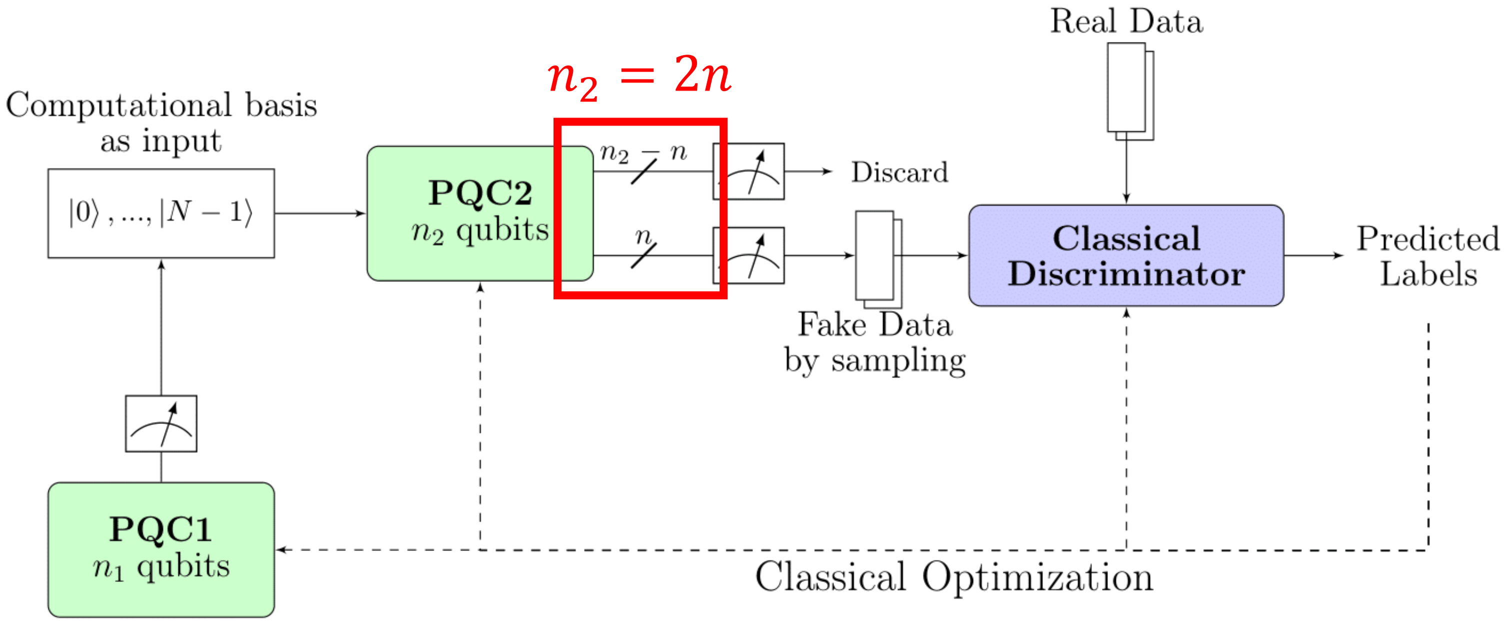
Continuous Variable Quantum GAN

Dual-PQC GAN model

Role of single generator shared by two parameterized quantum circuits (pqc)

- **PQC1** – Reproduce the distribution over 2^{n_1} images of size 2^n
- **PQC2** – Reproduce amplitudes over 2^n pixels on one image

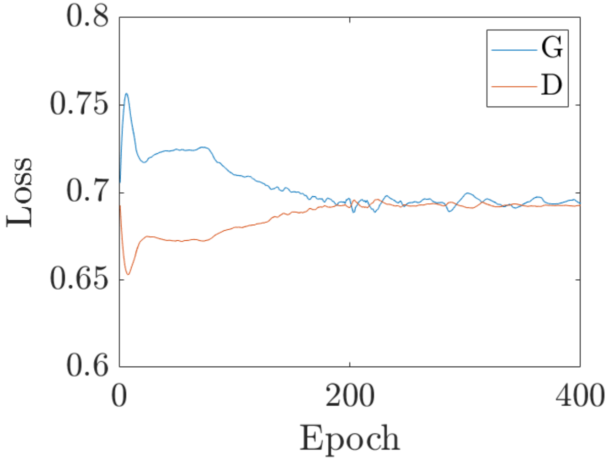
➡ **2^{n_1} images of size 2^n**

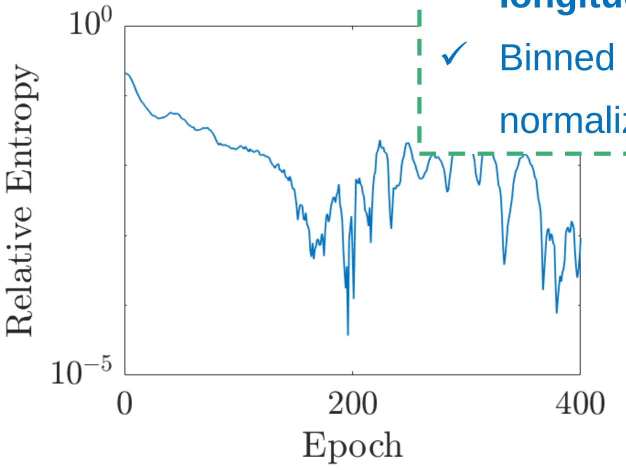


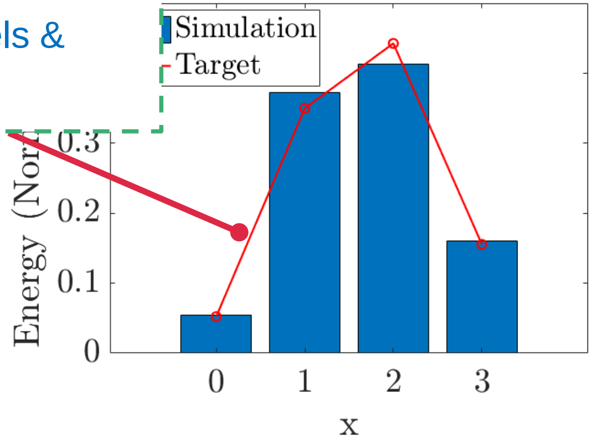
Application of Dual-PQC GAN in HEP

$n = 2, n_1 = 4, n_2 = 4, depth_{g1} = 2, depth_{g2} = 6$

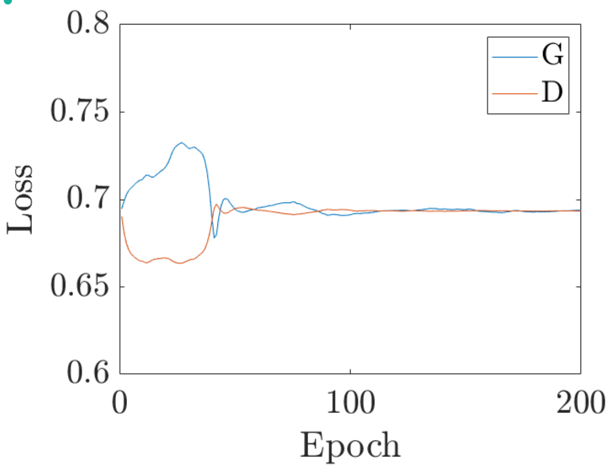
- ✓ 2D image summed **over longitudinal direction**
- ✓ Binned into 4 pixels & normalized

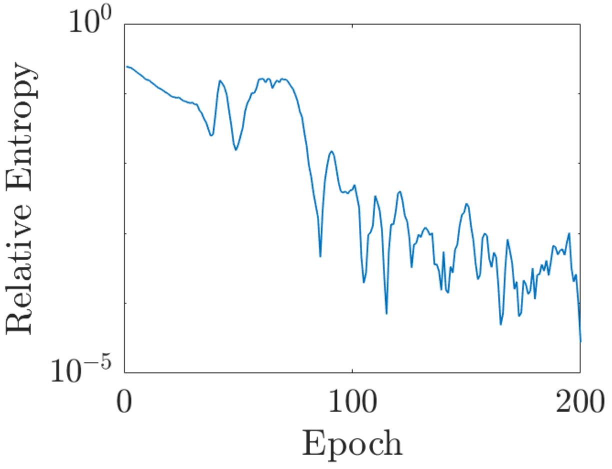


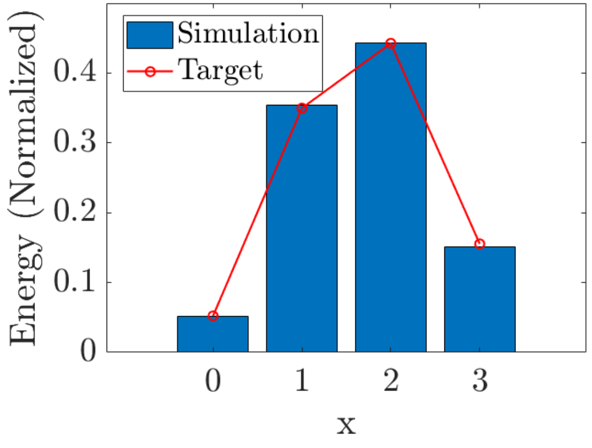




$n = 2, n_1 = 4, n_2 = 4, depth_{g1} = 2, depth_{g2} = 16$

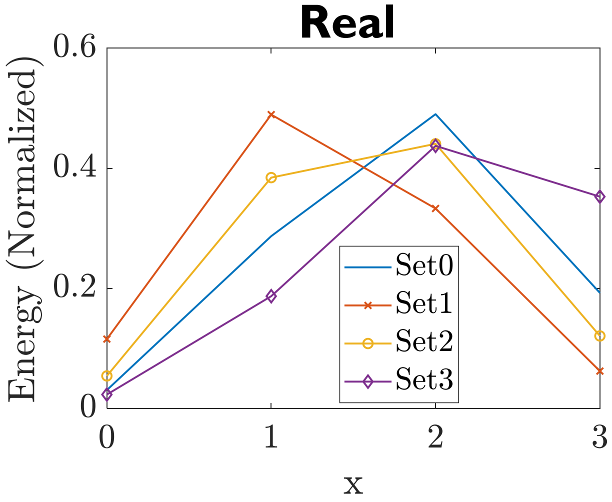






Convergence in individual images ?

Image0 and Image2 have different shape from real images



All four images have the same shape with real images
→ Peak at x = 1 or x = 2

$n = 2, n_1 = 4, n_2 = 4, depth_{g1} = 2, depth_{g2} = 6$

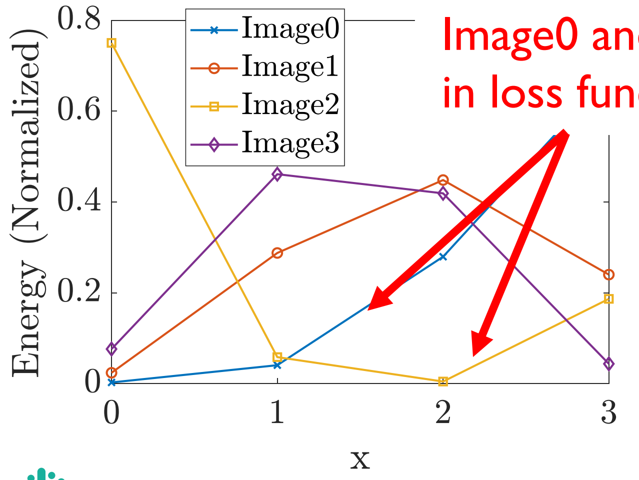
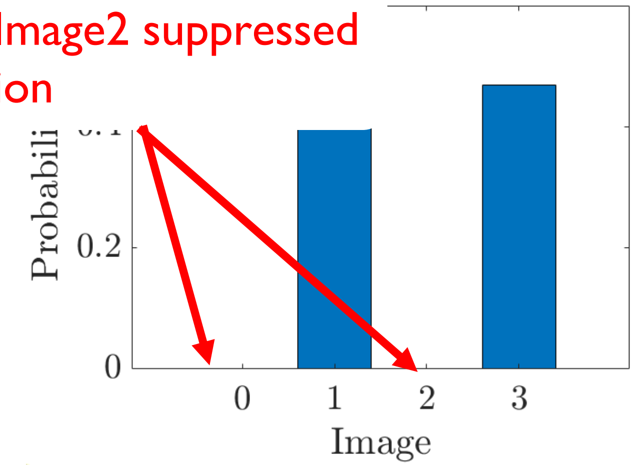
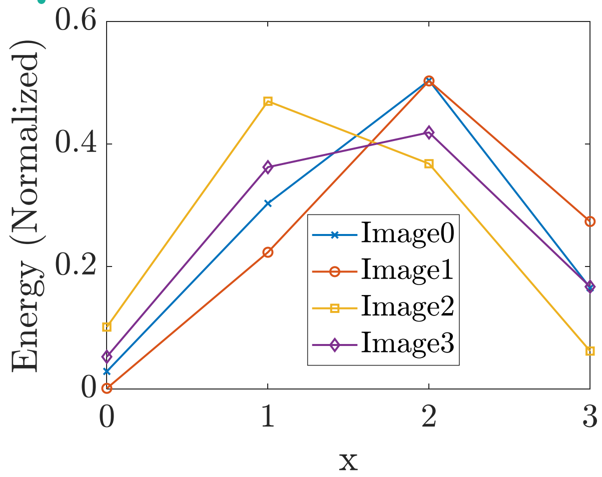
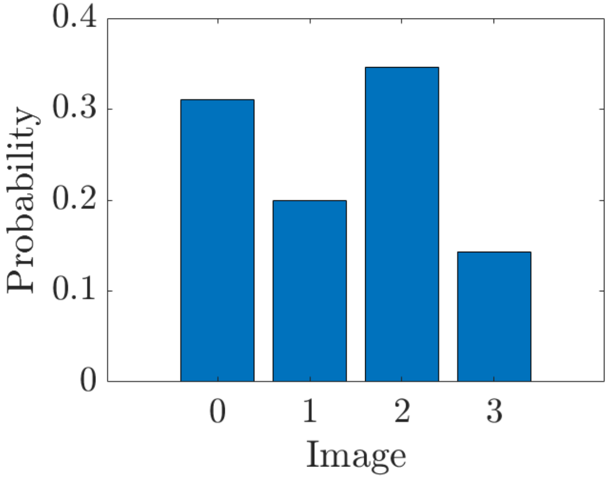


Image0 and Image2 suppressed in loss function



$n = 2, n_1 = 4, n_2 = 4, depth_{g1} = 2, depth_{g2} = 16$





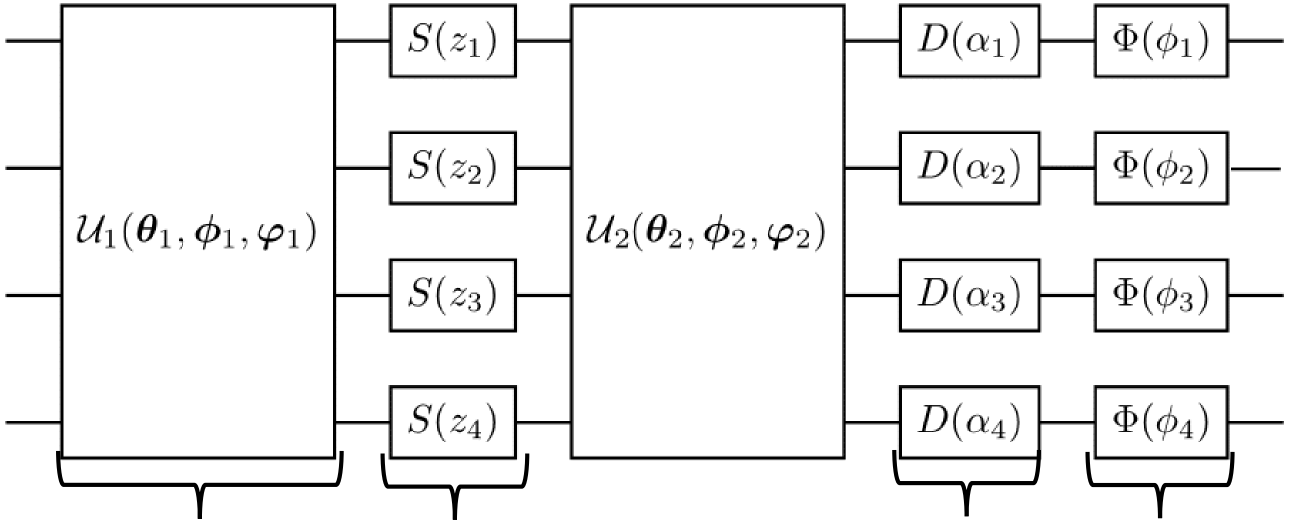
Continuous-Variable QC

Alternative perspective of quantum computing

- Fundamental information-carrying units = **Qumodes**

$$|\psi\rangle = \exp(-iHt)\,|0\rangle = \int dx\, \psi(x)|x\rangle dx = \sum_{n=0}^{\infty} \langle n|\psi\rangle|n\rangle$$

- Combine CV gates → CV Neural network

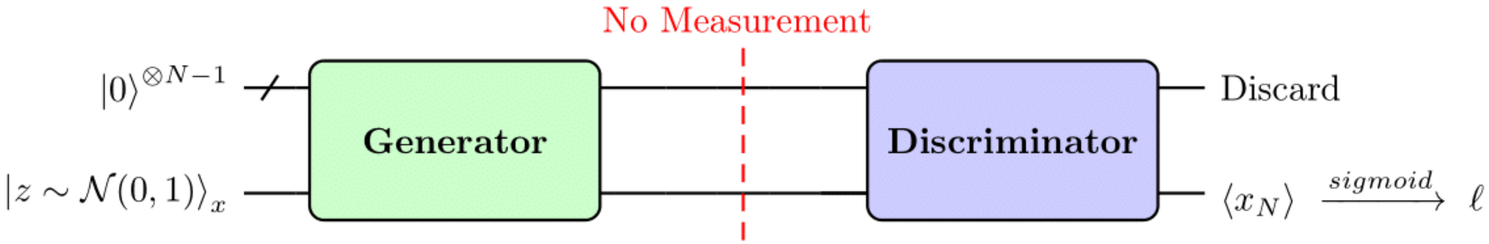


https://doi.org/10.1038/ncomms13795

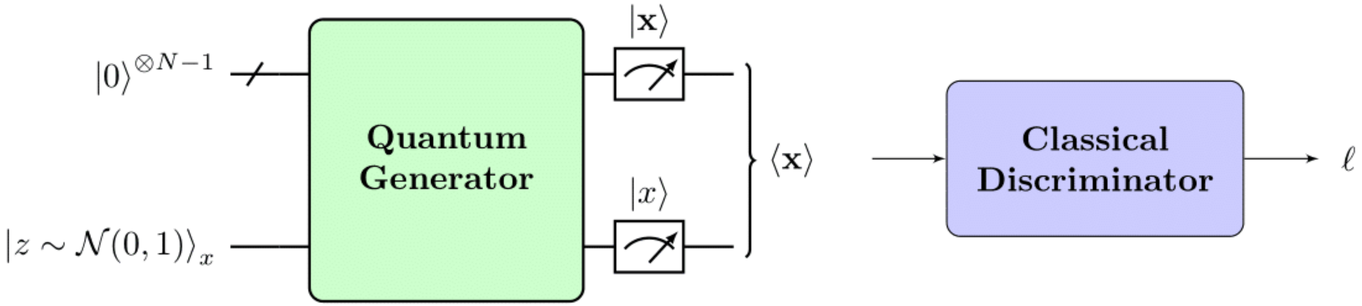
CV qGAN

Quantum GAN with a generator constructed by CVNN

- One qumode initialized by a noise $z \sim N(0,1)$: $|initial\rangle = |z\rangle \otimes |0\rangle^{\otimes N-1} = D_0(z)|0\rangle \otimes |0\rangle^{\otimes N-1}$



Fully Quantum model : Quantum Generator & Quantum Discriminator



Hybrid model : Quantum Generator & Classical Discriminator

Application of CV qGAN in HEP

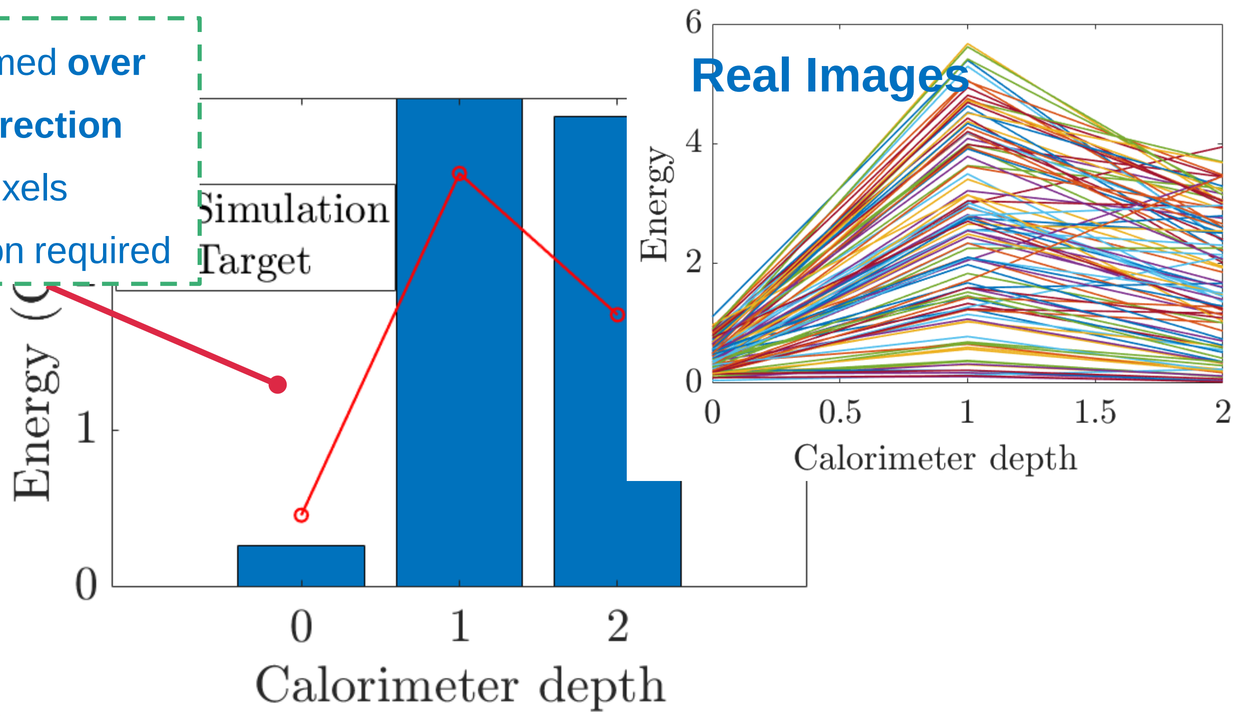
Fully quantum model

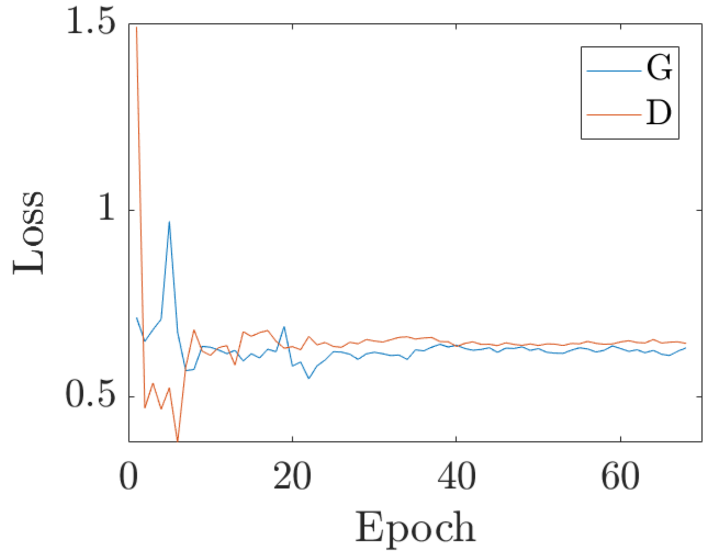
- $depth_g = 5$, $depth_d = 3$

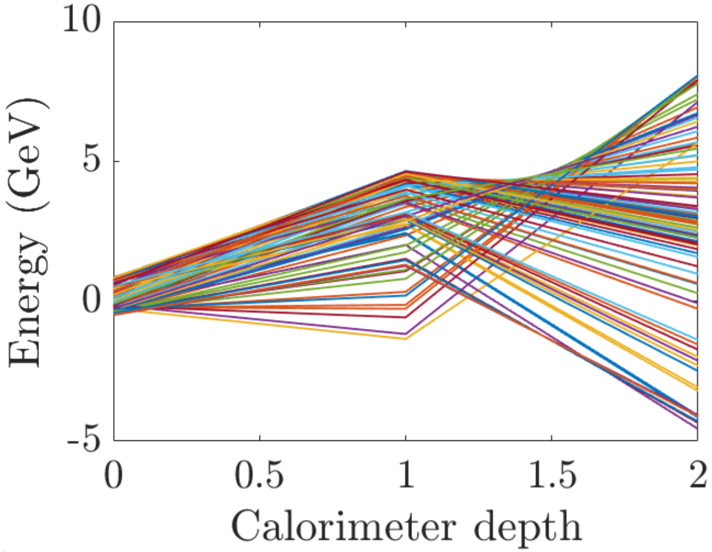
→ No convergence in mean image

→ Around half of the generated images with negative energy

- ✓ 2D image summed **over longitudinal direction**
- ✓ Binned into 3 pixels
- ✓ No normalization required







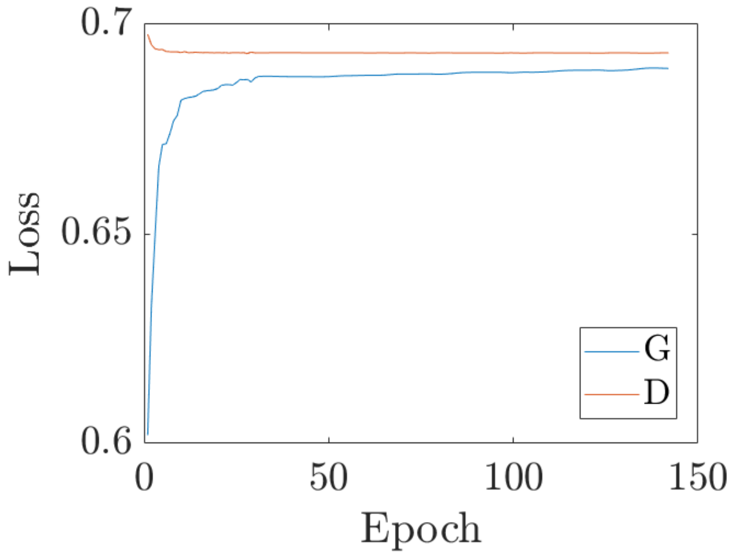
Application of CV qGAN in HEP

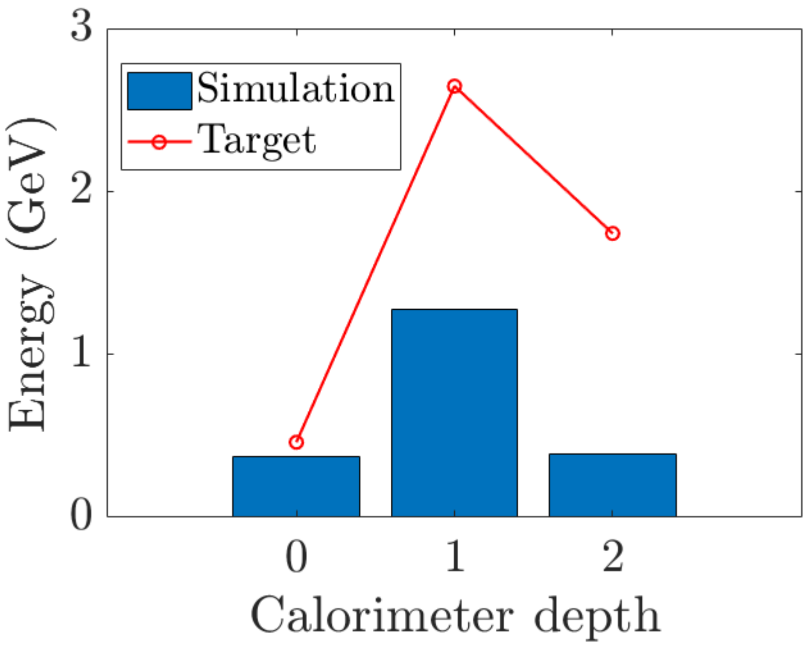
Hybrid Model

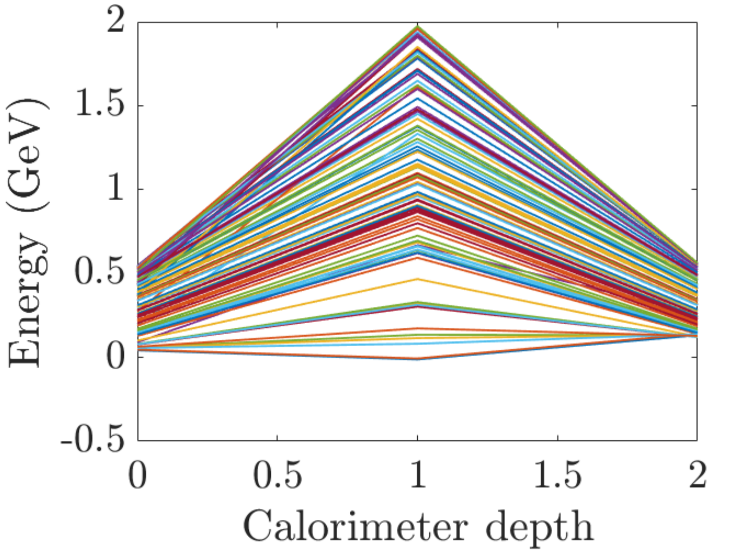
- $depth_g = 5$

→ No convergence in mean image

→ Few generated samples with negative energy







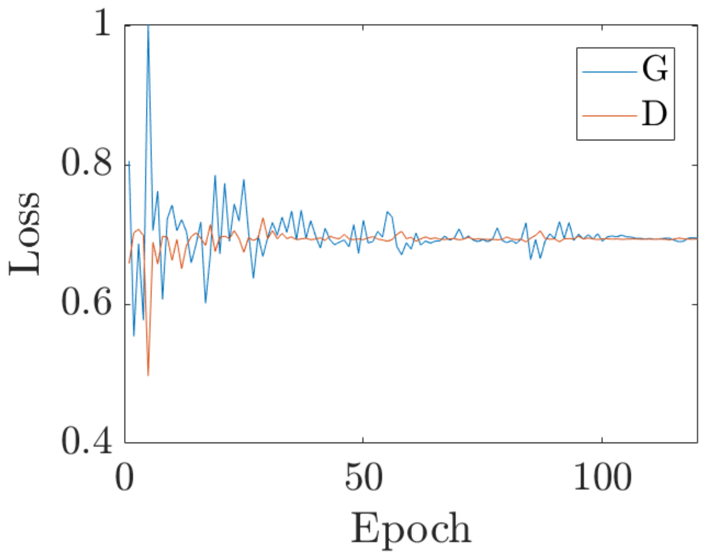
Application of CV qGAN in HEP

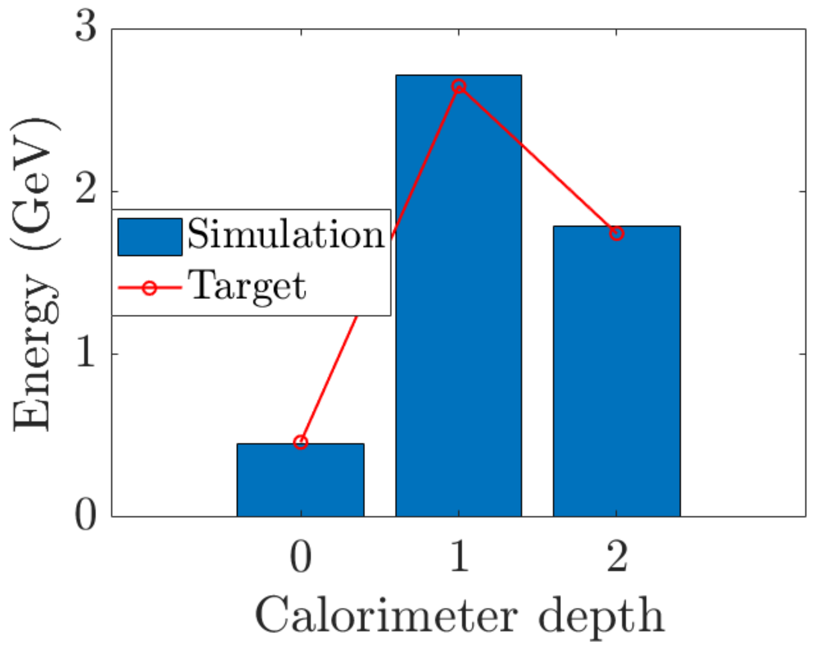
Hybrid Model

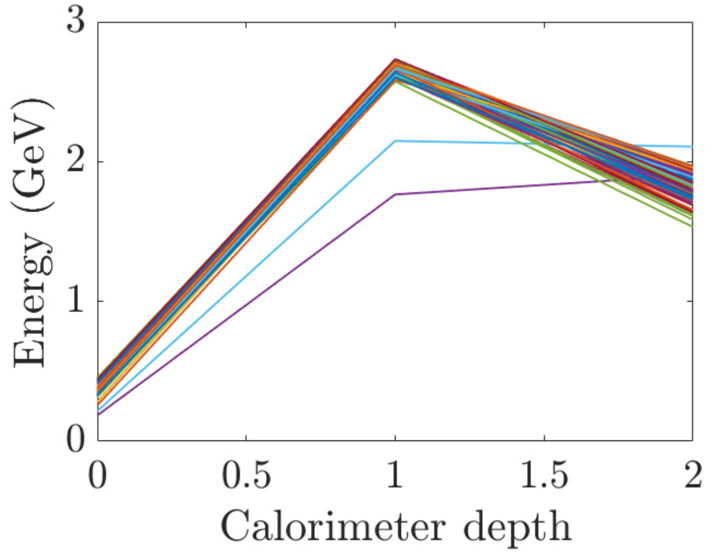
- $depth_g = 3$
- Different hyperparameters for the optimizer

→ Convergence in loss functions & mean image

➡ **Mode collapse**







Conclusion

Dual-PQC GAN & CV qGAN

- Two different prototypes of quantum GAN to reproduce a set of images

Dual-PQC Approach

- Reproduce images which correspond to the average images of different classes in real data
- Limited to a fixed number of images

➡ Introduce noise as input of PQC2

CV Approach

- CV qGAN → Exhibits well-know failures in classical GAN
- Number of qumodes limited by computing resources

➡ Parallel processing & Regularization techniques

QUESTIONS?

suyeon.chang97@gmail.com

Appendix A : qGAN in HEP (details)

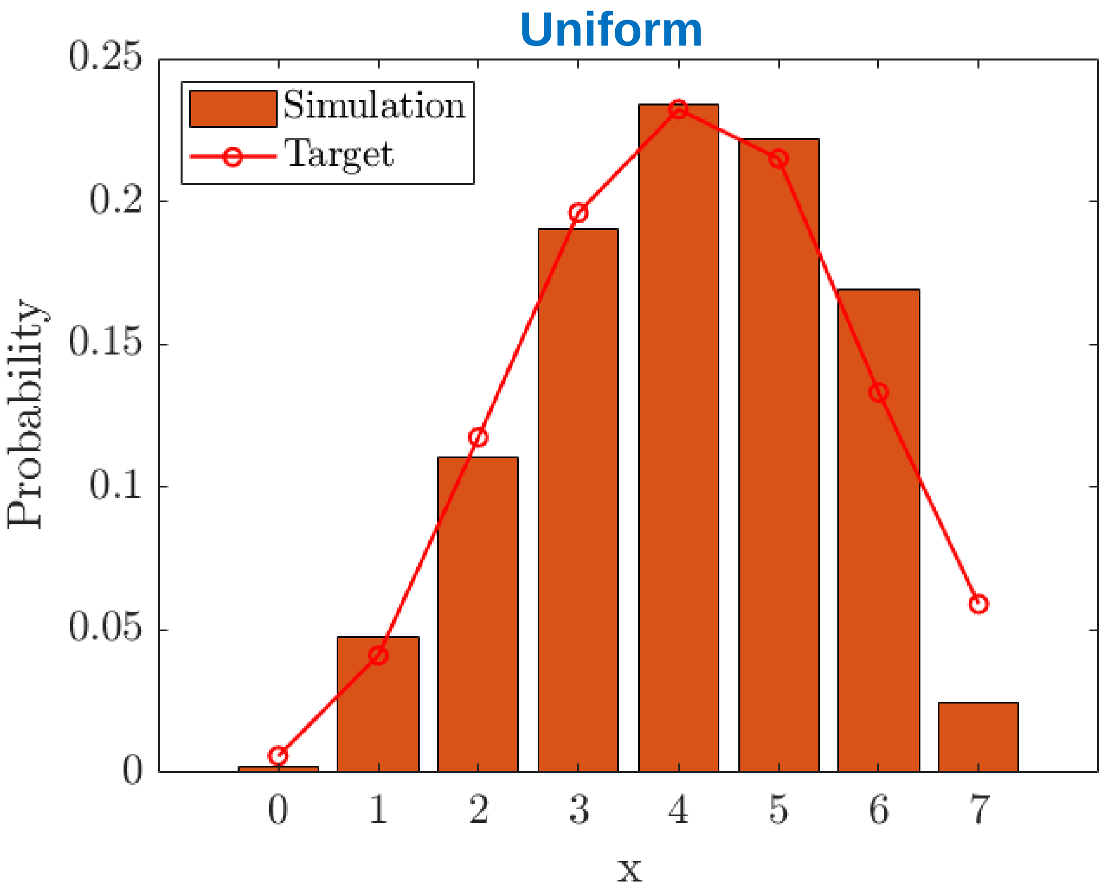
Preparation of Initial State

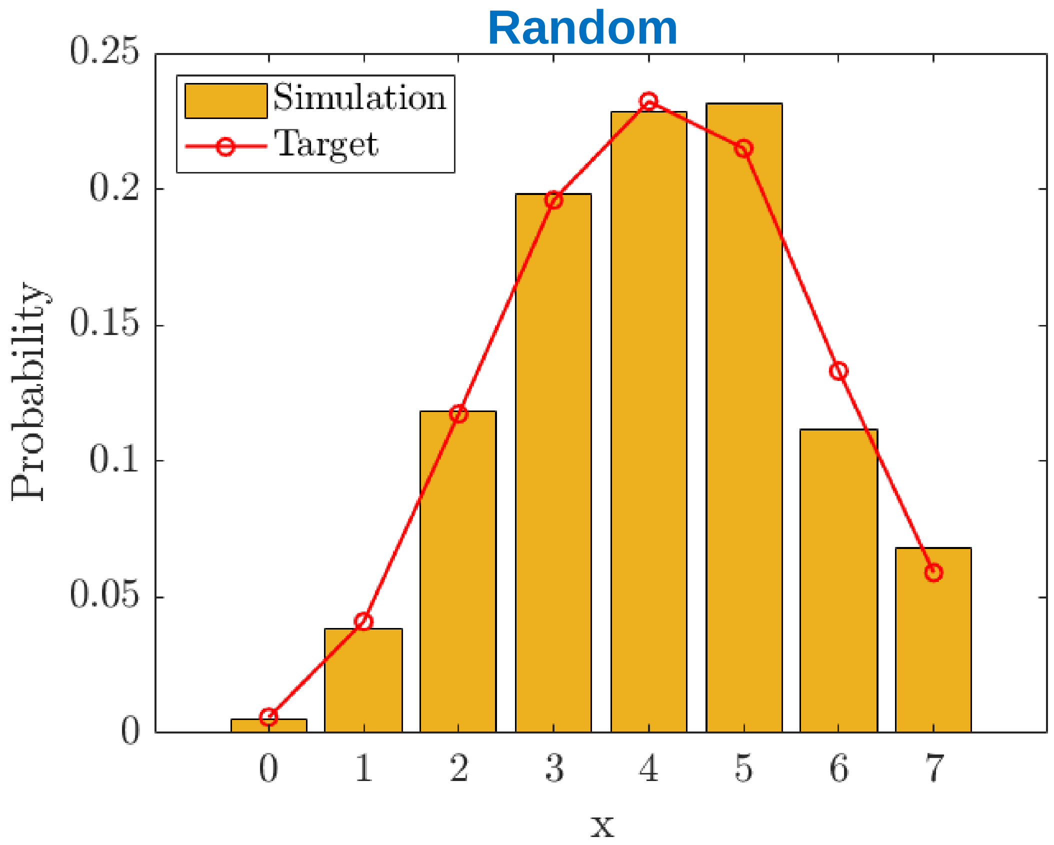
1. **Uniform** : Equiprobable Superposition of $|0\rangle, \ldots, |N-1\rangle$
2. **Normal** : Normally distributed with empirical mean and std of training set
3. **Random** : Randomly distributed over $|0\rangle, \ldots, |N-1\rangle$

Classical Discriminator

- ✓ PyTorch Discriminator
- ✓ 512 nodes + Leaky ReLU $\rightarrow$ 256 nodes + Leaky ReLU $\rightarrow$ single-node + sigmoid
- ✓ AMSGRAD optimizer for both generator and discriminator

Appendix B : qGAN in HEP (Results)





Appendix C : Why $n_2 > n$?

$$M(j) = \begin{pmatrix} |I_{0j}|^{\frac{1}{2}} e^{i\phi_{0j}} \\ \vdots \\ |I_{2^n-1j}|^{\frac{1}{2}} e^{i\phi_{2^n-1j}} \end{pmatrix}, \quad \phi_{ij} \in [0, 2\pi[$$

where I_{ij} = Amplitude at pixel i for image j → Normalized

Case $n_2 = n$

- Quantum Circuit consists of reversible gates → **Unitary matrix**
- Inputs = computational basis → $M(j) = \mathrm{j^{th}}$ column at M_{PQC_2}

→ Cannot train PQC2 with n qubits if M(j) do not form an orthonormal basis

Case $n_2 = 2n$

- First 2^n columns of PQC2 is constructed as : $M_{PQC_2}(i) = |i\rangle \otimes |M(i)\rangle$ where $|i\rangle \in \{|0\rangle, \ldots, |2^n - 1\rangle\}$,

$$\rightarrow \langle M_{PQC_2}(i) | M_{PQC_2}(j) \rangle = \langle i|j \rangle \langle M(i)|M(j)\rangle = \begin{cases} 1 & if\ i = j \\ 0 & otherwise \end{cases}$$

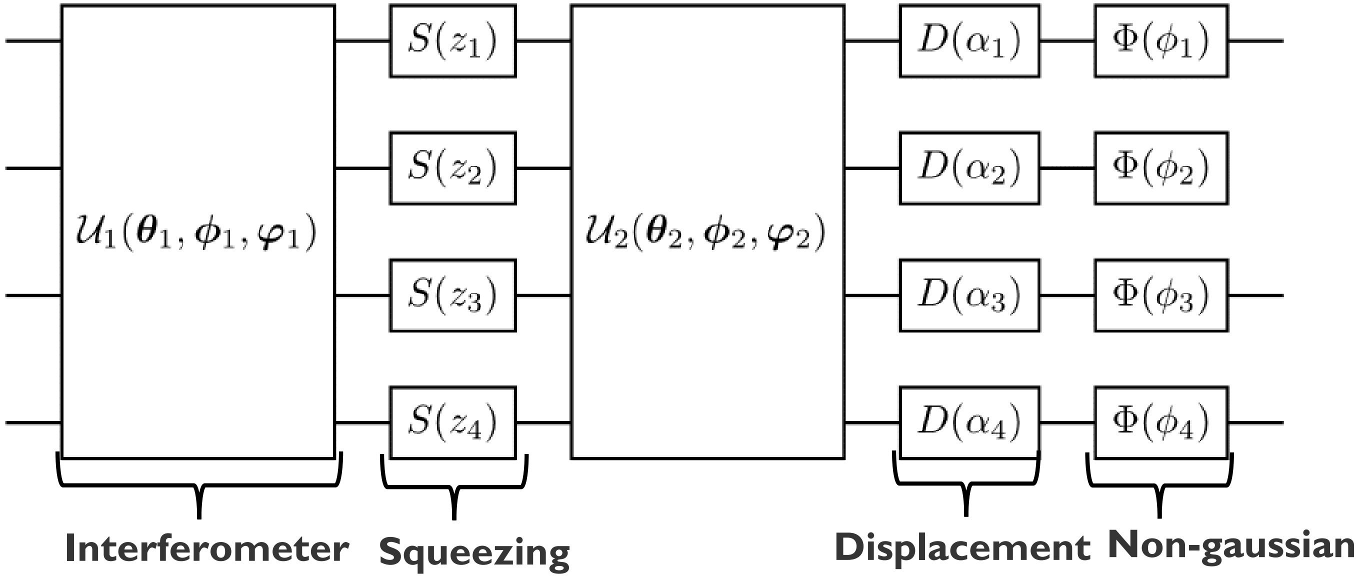
→ $2^{2n} - 2^n$ columns can be chosen freely to construct a unitary matrix

Appendix D : Qubit vs. CV

	CV	Qubit
Fundamental Unit	Qumodes $\{\lvert x\rangle\}_{x\in R}$, $\lvert\psi\rangle = \int dx\,\psi(x)\lvert x\rangle dx$	Qubits $\lvert 0/1\rangle$, $\lvert\psi\rangle = \alpha\lvert 0\rangle + \beta\lvert 1\rangle$
Relevant Operators	Position $\hat{x}$, Momentum $\hat{p}$ Mode operators $\hat{a}, \hat{a}^t$	Pauli Operators $\sigma_x, \sigma_y, \sigma_z$
Common Gates	Displacement $D_i(\alpha) = \exp\left(\alpha\hat{a}_i^t - \alpha^*\hat{a}_i\right)$ Rotation $R_i(\phi) = \exp(i\phi\hat{n}_i)$ Squeezing $S_i(z) = \exp\left(\frac{1}{2}\,(z^*\hat{a}_i^2 - z\hat{a}_i^{t2})\right)$ Beam Splitters $BS_{ij}(\theta,\phi) = \exp\left(\theta(e^{i\phi}\hat{a}_i^t\hat{a}_j - e^{-i\phi}\hat{a}_i\hat{a}_j^t)\right)$ Kerr $K_i(\kappa) = \exp(i\kappa n_i^2)$	Phase Shift, Rotation, Hadamard, Controlled-U gate
Measurements	Homodyne $\lvert x_\phi\rangle\langle x_\phi\rvert$, $\hat{x}_\phi = \cos(\phi)\hat{x} + \sin(\phi)\hat{p}$ Heterodyne $\frac{1}{\pi}\lvert\alpha\rangle\langle\alpha\rvert$ Photon Counting $\lvert n\rangle\langle n\rvert$	Pauli Measurements $\lvert 0/1\rangle\langle 0/1\rvert, \lvert\pm\rangle\langle\pm\rvert, \lvert\pm i\rangle\langle\pm i\rvert$

https://doi.org/10.22331/q-2019-03-11-129

Appendix E : CVNN



https://doi.org/10.1038/ncomms13795

- Fully connected layer : $\boldsymbol{x} \rightarrow \phi(W\boldsymbol{x} + \boldsymbol{b})$ W= Weight matrix, $\boldsymbol{b}$ = bias, $\phi(x)$ = Activation function
- Weight matrix W decomposed using **singular value decomposition** : $W = O_2 \Sigma O_1$

1. Multiplication by an orthogonal matrix $O_1 \rightarrow$ Apply an **interferometer** U_1
2. Multiplication by a diagonal matrix $\Sigma \rightarrow$ Apply a **squeezing gate** $S(\mathbf{r})|\mathbf{x}\rangle = e^{-\frac{1}{2}\Sigma_i r_i}|\Sigma\mathbf{x}\rangle$
3. Multiplication by another orthogonal matrix $O_2 \rightarrow$ Apply an **interferometer** U_2
4. Addition of bias $\boldsymbol{b} \rightarrow$ Apply a **displacement gate** $D(\boldsymbol{\alpha})|\mathbf{x}\rangle = |\mathbf{x} + \boldsymbol{\alpha}\rangle$
5. Non-linear function $\phi(x) \rightarrow$ Apply a **Kerr gate** $\Phi|\mathbf{x}\rangle = |\phi(\mathbf{x})\rangle$

$L|\boldsymbol{x}\rangle \propto |\phi(W\boldsymbol{x} + \boldsymbol{b})\rangle$