A new style-based quantum GAN applied to LHC event generation

Carlos Bravo-Prieto[1,2], Julien Baglio[3], Marco Cè[3], Anthony Francis[4,3], Dorota M. Grabowska[3] and Stefano Carrazza[5,3,1]

[1]Quantum Research Centre, Technology Innovation Institute, Abu Dhabi, UAE. [2]Departament de Física Quàntica i Astrofísica and Institut de Ciències del Cosmos, Universitat de Barcelona, Barcelona, Spain. [3]Theoretical Physics Department, CERN, Geneva, Switzerland. [4]Institute of Physics, National Yang Ming Chiao Tung University, Hsinchu, Taiwan. [5]TIF Lab, Dipartimento di Fisica, Università degli Studi di Milano and INFN Sezione di Milano, Milan, Italy.

Generative models and the search for new physics at the LHC

The search for new physics at the Large Hadron Collider (LHC) requires comparing real experimental data with statistical simulations produced by Monte Carlo (MC) event generators. The MC simulations generate 'fake' data based on the known properties of the experiment and the predictions derived from the mathematical formulation of nature - the Standard Model. Any deviations between experimental results and MC simulations could provide hints of new, unexplained phenomena. Because the LHC produces $\mathcal{O}(10^9)$ proton-proton collisions every second, running equivalent MC event generation can be expensive, both in time and in computing power. This has lead to an increased interest in generative adversarial networks (GANs) hoping to significantly accelerate MC generation and to reduce the computational load.

Today's quantum devices are typically at the 'Noisy Intermediate-Scale Quantum' (NISQ) stage. However, even at this stage, there is an expectation that extending from classical towards quantum machine learning could be beneficial. With the ongoing, impressively fast development of quantum computing capabilities, we take this motivation and explore the applicability of quantum GANs to MC event generation for the LHC.

Workflow Design - Quantum Generator and Classical Discriminator

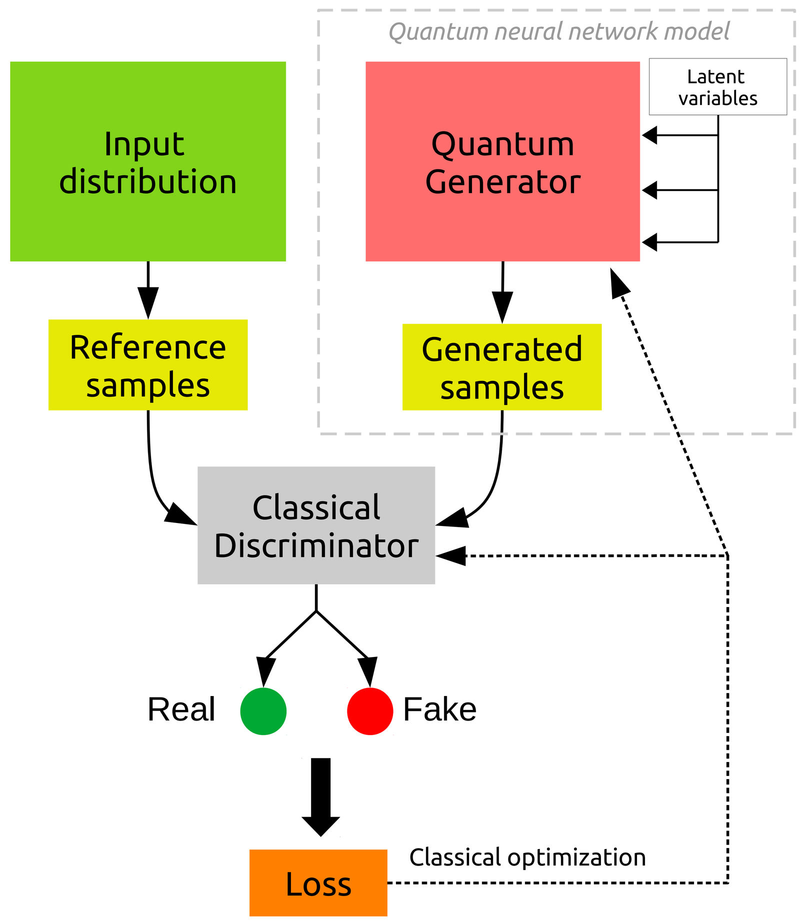
A simple GAN[a] has 3 components: (1) a discriminator, (2) a generator, (3) an adversarial training procedure.

Here we present a **quantum-classical GAN**, where the generator is a quantum neural network (QNN) while the discriminator is a classical one (CNN).

Generating samples using a quantum device is very attractive because density modeling and sampling are delegated to the (quantum) hardware architecture.

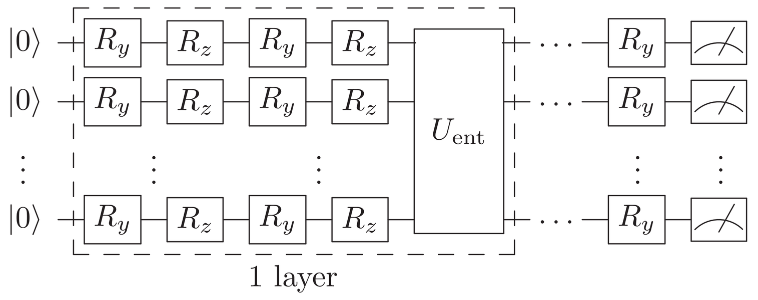
The adversarial training is a 2-player min-imax game:

- $\min_{\phi_g} \mathcal{L}_G(\phi_g, \phi_d)$, create accepted samples
- $\max_{\phi_d} \mathcal{L}_D(\phi_g, \phi_d)$, classify fake/real samples



[a]Goodfellow et al., [arXiv:1406.2661].

Quantum Generator Architecture - the style-qGAN



The quantum generator's task is creating fake samples to fool the classical discriminator.

style-qGAN model[a]**:** Each qubit rotation is parameterized by the trainable parameters ϕ_g and the *latent vector* ξ:

$$R^{(i,j)}_{y,z}(\phi_g,\xi) = R_{y,z}\left(\phi_g^{(i)}\xi^{(j)} + \phi_g^{(i+1)}\right).$$

New: The quantum generator embeds the input latent variables into all the quantum gates of the network. The architecture can process and decide in which parts of the QNN the latent variables should play a relevant role.

The fake samples are prepared by acting with the parameterized QNN on the initial n-qubit state $0^{\otimes n}$, and then measuring in the computational basis. Here, each qubit delivers one sample component, i.e. the sample $x \in \mathbb{R}^n$ is generated as

$$x = \left[-\langle\sigma_z^1\rangle, -\langle\sigma_z^2\rangle, \dots, -\langle\sigma_z^n\rangle\right],$$

with

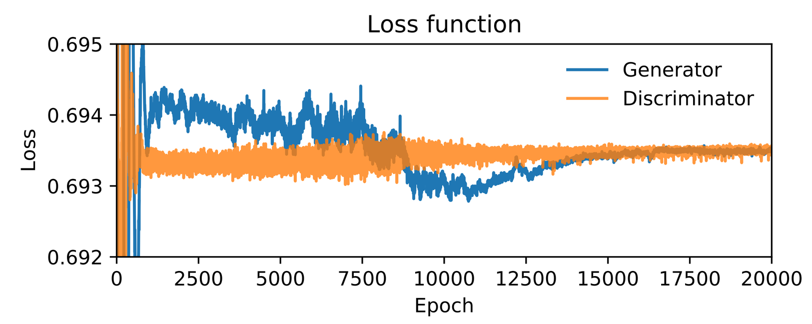
$$\langle\sigma_z^i\rangle = \langle\Psi(\phi_g,\xi)|\sigma_z^i|\Psi(\phi_g,\xi)\rangle,$$

where $|\Psi(\phi_g,\xi)\rangle$ is the output state from the quantum generator. Other ways to generate fake samples are possible.

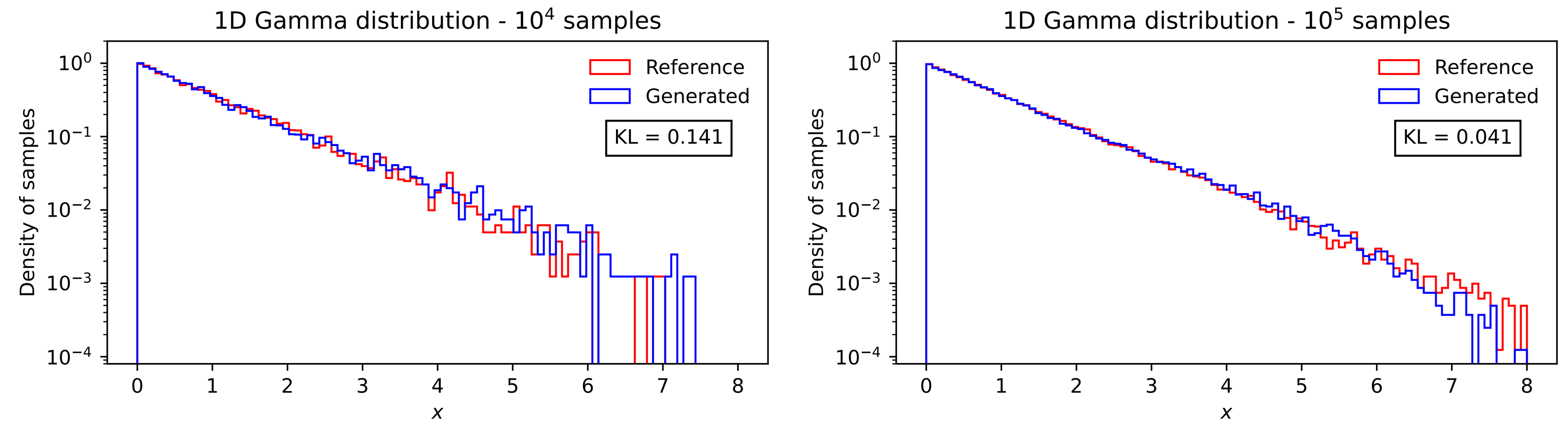
[a]The classical counterpart was proposed in Karras et al., [arXiv:1812.04948].

Training and Validation - Learning a known $D=1$ Γ-distribution

As first application and to test the framework we sample a 1D Γ-distribution: $p_\gamma(x,\alpha,\beta) = x^{\alpha-1}\frac{e^{-x/\beta}}{\beta^\alpha\Gamma(\alpha)}$, where Γ is the Gamma function and we set $p_\gamma(x,1,1)$. We find:



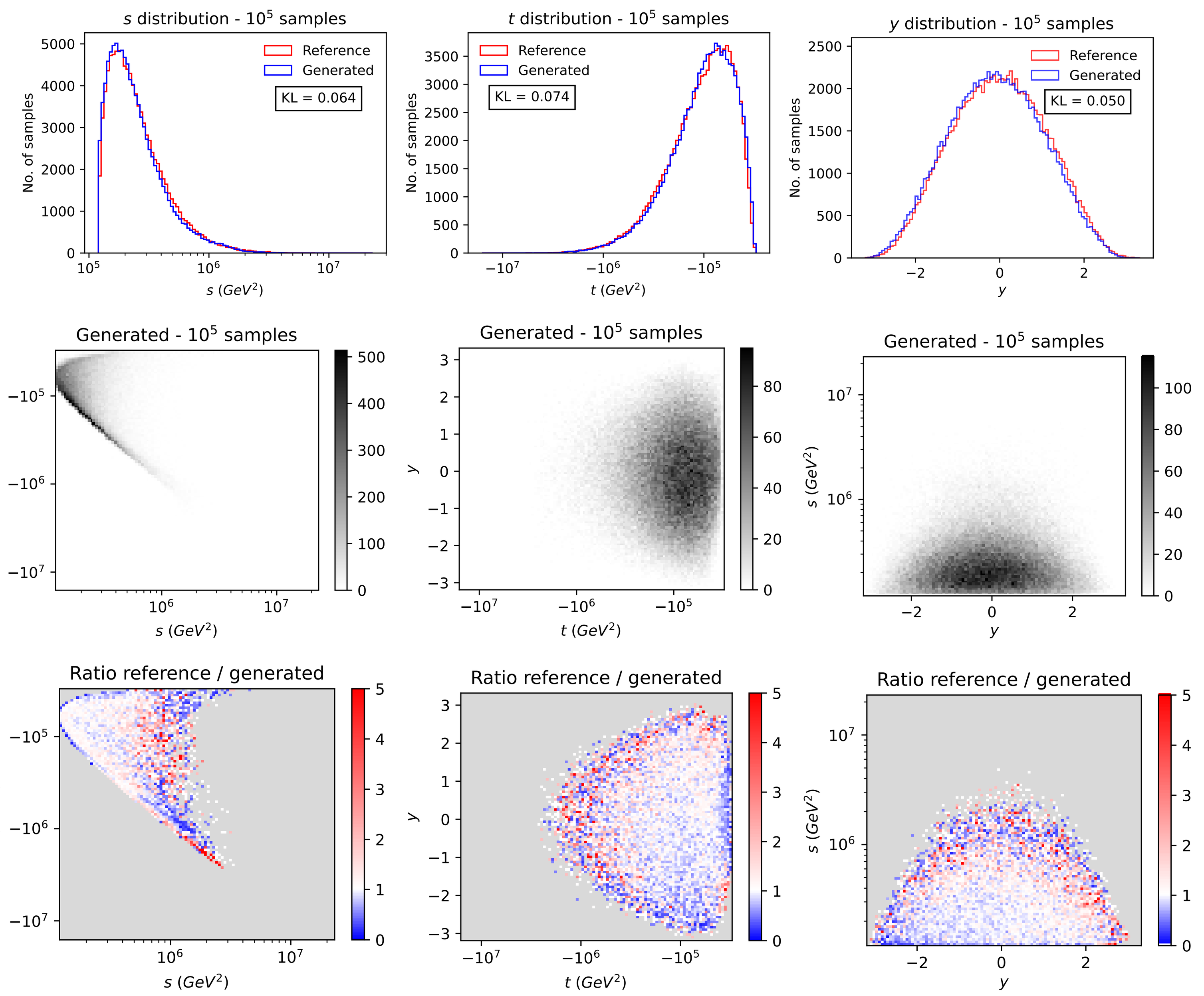
Example of loss function convergence. After an initial warm-up phase, the loss function of both models converges. This indicates that the style-qGAN has been successfully trained.



Examples of 1D gamma distribution sampling for the reference underlying distribution (red) and a style-qGAN model (blue) that has been trained with 10^4 reference samples. The left plot compares 10^4 generated samples. The right compares 10^5 samples generated from the style-qGAN model trained with 10^4 reference samples. We observe a good level of agreement between both distributions, despite the model being trained on a small training set.

LHC Event Generation - Results on Simulated Hardware

We apply the style-qGAN to the Monte Carlo event generation of the process of two protons producing a top quark pair, $pp \to t\bar{t}$, at the LHC with $\sqrt{s} = 13$ TeV energy. We generated 10^4 MC events for training classically and selected three observables - the Mandelstam variables (s,t) and the rapidity y. The data is fully 3D correlated and can be used to classify this process.

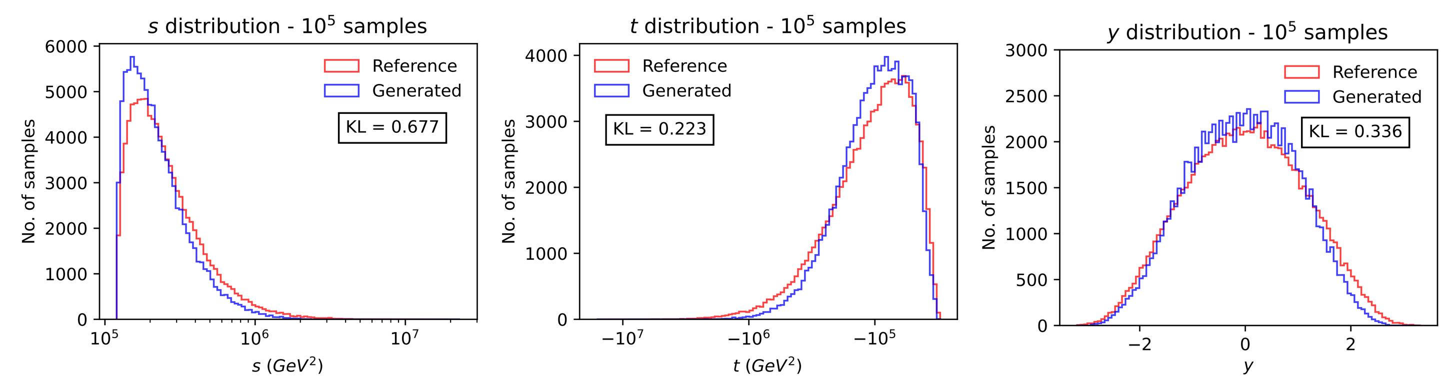


Marginal sample distributions for the physical observables s,t,y in $pp \to t\bar{t}$ production at the LHC for the style-qGAN model trained with 10^4 samples (top row), corresponding two-dimensional sampling projections (middle row) and the ratio to the reference underlying prior MC distribution (bottom row). The model uses 3-qubits with 5 latent dimensions and 2 layers.

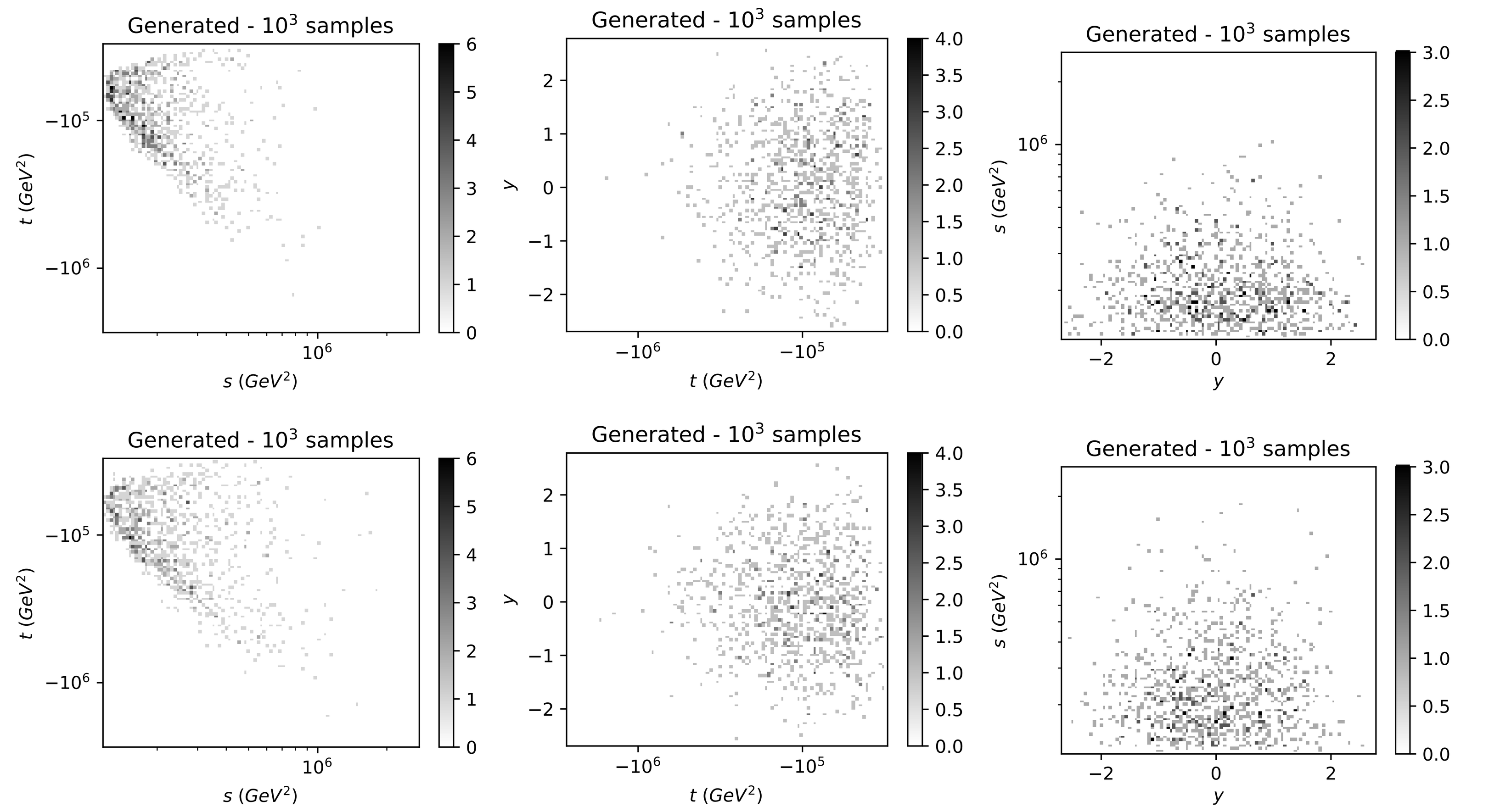
Results on Quantum Hardware - ibmQ and ionQ

We benchmark our style-qGAN model on real quantum hardware:

- superconducting transmon qubits as provided by IBM Q quantum computers[a]
- trapped ion technology as provided by IonQ quantum computers[b]



Marginal sample distributions for the physical observables s,t,y in $pp \to t\bar{t}$ production at the LHC using the style-qGAN generator model trained with 10^4 samples on `ibmq_santiago`.



Example of two-dimensional sampling projections for $pp \to t\bar{t}$ production using the style-qGAN generator model on `ibmq_santiago` *(top row) and IonQ (bottom row) trained with 10^4 samples.*

[a]https://research.ibm.com/blog/ibm-quantum-roadmap
[b]https://IonQ.com/posts/december-09-2020-scaling-quantum-computer-roadmap

Proof-of-concept - The style-qGAN as MC Event Generator

- We present a quantum generator architecture - the style-qGAN - for generative adversarial learning for Monte Carlo event generation, used to simulate particle physics processes at the Large Hadron Collider (LHC).
- We implement and validate the quantum network on artificial data generated from known underlying distributions. The network is then applied to Monte Carlo-generated datasets of specific LHC scattering processes.
- The new quantum generator architecture leads to an improvement in state-of-the-art implementations while maintaining shallow-depth networks.
- The quantum generator successfully learns the underlying distributions even if trained with small training sample sets making it particularly interesting for data augmentation applications.
- We deploy our method on two different quantum hardware architectures, trapped-ion and superconducting technologies, to test its hardware-independent viability.

More details in

C. Bravo-Prieto, J. Baglio, M. Cè, A. Francis, D. M. Grabowska and S. Carrazza, *"Style-based quantum generative adversarial networks for Monte Carlo events,"* arXiv:2110.06933 [quant-ph].