Precise Quantum Angle Generator Designed for Noisy Quantum Devices

Florian Rehm[1,3,*], *Sofia* Vallecorsa[2], *Kerstin* Borras[3,], *Dirk* Krücker[2], *Michele* Grossi[2], and *Valle* Varo[2,**]

[1]CERN, Geneva, Switzerland
[2]RWTH Aachen University, Aachen, Germany
[3]DESY, Hamburg, Germany

Abstract. The Quantum Angle Generator (QAG) is a cutting-edge quantum machine learning model designed to generate precise images on current Noise Intermediate Scale Quantum devices. It utilizes variational quantum circuits and incorporates the MERA-upsampling architecture, achieving exceptional accuracy. The study demonstrates the QAG model's ability to learn hardware noise behavior, with stable results in the presence of simulated quantum hardware noise up to 1.5% during inference and 3% during training. However, deploying the noiseless trained model on real quantum hardware reduces accuracy. Training the model directly on hardware allows it to learn the underlying noise behavior, maintaining precision comparable to the noisy simulator. The QAG model's noise robustness and accuracy make it suitable for analyzing simulated calorimeter shower images used in high-energy physics simulations at CERN's Large Hadron Collider.

1 Introduction

Quantum computing possesses revolutionary potential, offering accelerated computations and tackling hitherto intractable problems [1]. In the present Noisy Intermediate Scale Quantum (NISQ) era, quantum devices grapple with hardware errors, connectivity constraints, and limited qubits [2]. While practical quantum advantage remains challenging, efforts to optimize algorithms for NISQ-era constraints are under intensive exploration. Quantum Machine Learning (QML) emerges as a promising domain, showing resilience to noise and yielding favorable outcomes on NISQ devices [3].

High Energy Physics (HEP) experiments, like those at CERN's Large Hadron Collider (LHC), demand extensive simulated data for precise results [4]. To address this, CERN maintains the world's largest computing grid [5]. Machine Learning (ML) models provide faster outcomes than traditional simulations, without compromising accuracy. Quantum Machine Learning (QML) shows potential in meeting simulation needs [6, 7]. QML leverages quantum circuits, capitalizing on properties like superposition and entanglement, potentially surpassing classical neural networks [8]. Moreover, QML excels in representing complex distributions using fewer parameters, thanks to its expansive phase space.

Yet, encoding classical data efficiently into qubit states presents a significant hurdle [9]. Several techniques exist, each with strengths and weaknesses, dependent on the application

*e-mail: florian.matthias.rehm@cern.ch
**e-mail: valle.varo@desy.de

[9]. Theoretical benchmarks suggest linear scaling between qubits and features for quantum advantage [10]. However, techniques exceeding linear scaling often face limitations, such as amplitude encoding's inability to provide absolute energy values for NISQ-based image generation. Current quantum generative models like Quantum Circuit Born Machine (QCBM) [11], Quantum Variational Autoencoders [12], and various quantum Generative Adversarial Networks [13–15] grapple with scalability and fidelity. In contrast, our Quantum Angle Generator (QAG), initially proposed in [16], employs angle encoding, maintains linear scaling, and offers high-fidelity outcomes on noisy quantum devices.

High-energy physics simulations primarily rely on resource-intensive Monte Carlo methods and the Geant4 toolkit [17]. These methods heavily load the LHC Computing Grid [18]. With upcoming LHC advancements, a surge in simulation demands might surpass current computing budgets [19, 20]. Machine learning, augmented with quantum devices, presents a promising alternative, potentially reducing simulation times and enhancing accuracy [21–23].

Electromagnetic calorimeters, pivotal to HEP detectors, measure particle energies and significantly contribute to simulation times [18, 24]. Their outputs, termed "shower images," entail 3D spatial representations. In this study, we focus on one-dimensional, eight-pixel representations of $25 \times 25 \times 25$ pixel images from [25], targeting energy ranges between [225, 275] GeV. The processed dataset, divided into training and test subsets, is detailed in [26].

2 Quantum Angle Generator

The Quantum Angle Generator (QAG) is a Quantum Machine Learning (QML) model that employs angle encoding for precise image generation. It utilizes variational quantum circuits, using an decoding strategy illustrated in figure 1. Qubits start in basis state $|0\rangle$ and pass through a Hadamard (H) gate for superposition, followed by a y-rotational (R_y) gate introducing randomness for new sample generation. R_y gate angles Ω are randomly drawn from $[-1, 1]$ uniform distribution and pixel-wise scaled by training data's pixel standard deviations for suitable energy variations. For diverse particle energies, Ω angles are further multiplied by a random value in $[-0.25, 0.25]$. The trainable part of the QAG model comprises unitary transformations via quantum circuits, explored in section 3.

2.1 Model description

To convert quantum states back to classical energies via angle encoding, the model undergoes multiple executions with quantum state measurements, determined by the number of shots nb_{shots}. Measurement frequency of state $|0\rangle$ calculates scalar intersection I on the Bloch sphere's z-axis.

On the Bloch sphere's x-z-plane, angle θ is zero at $|+\rangle$ state and clockwise rotation yields positive values. This is shown in figure 1, depicting state $|\Psi\rangle$, intersection I, and angle θ for a single qubit. Angle θ transforms into pixel energy E as follows:

$$E = \frac{(E_{max} - E_{min})(\theta - \theta_{min})}{\theta_{max} - \theta_{min}}. \tag{1}$$

It's important to mention that θ and thus E are discrete. The precision and resolution of θ increase with the number of shots nb_{shots}. Currently, IBMQ devices allow up to nb_{shots} = 100,000 shots. This research indicates 512 shots offer satisfactory resolution.

2.2 Training the model

The QAG model utilizes two objective functions during training. Firstly, the Mean Maximum Discrepancy (MMD) loss [27, 28], previously employed in quantum models like the QCBM

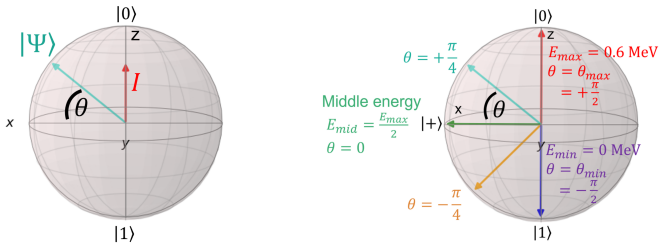

Figure 1. On the left, we observe the transformation of an illustrative state denoted as Ψ into an angle θ. On the right, a decoding example for an angle θ is presented with its corresponding energy value.

[11], offers promising average shower distributions. However, its performance in detailed aspects, such as pixel correlation, was suboptimal. To address this, a correlation (Corr) loss was introduced, computed using the mean squared error (MSE) between the pixel correlations of the training and generated data.

For training, we used the Simultaneous Perturbation Stochastic Approximation (SPSA) optimizer [29], which streamlines optimization with two steps per epoch. Hyperparameters were optimized using the Optuna [30] library. All experiments were conducted on Qiskit version 0.26.2. The models underwent 500 epochs of single-batch training. The MMD loss weight, starting at one, decays at a rate of −0.001 per epoch from the 100th epoch, while the Corr loss weight ascends from zero at the same rate. The batch size, initially generating a single image, was increased to 20 images post the 100th epoch for better Corr loss computation. Training and inference were done with 512 shots. The SPSA optimizer had an initial learning rate $c_0 = 1$ which decayed exponentially by 0.006 from the 50th epoch. These configurations exhibited the best outcomes in our tests.

3 Quantum Circuit Architecture

Optimal circuits should balance minimal parameters with accuracy. We evaluated diverse circuit architectures using key metrics. These architectures employ trainable rotational gates, particularly focusing on Ry gates for y-axis angle encoding. Some variants included extra R_z gates or deeper structures (d2) for potential enhancements. Our mainstay is two-qubit controlled-not gates (cx gates) from IBM Quantum (IBMQ) devices.

Our study's metrics encompass trainable parameters N_p, expressibility X, and entanglement capability E. We explore if smaller circuits can match the accuracy of larger ones, despite fewer parameters. Expressibility $(1 - X)$, measures state representation proficiency; values nearing 1 are favorable. Entanglement capability E quantifies the circuit's skill in generating entangled states, with 1 as the ideal.

For the calorimeter investigation, we assess outcomes (figure 2) comparing mean square errors (MSEs) among circuits. This MSE measures Geant4 and QAG image disparities, averaged over 25 trials per circuit, excluding best and worst outcomes. MERA-up, MERA-up-d2, and MERA-up-Rz circuits exhibited the lowest MSEs, consistent with their elevated X and E values. Although MERA-up-Rz stands as the top performer, the MERA-up circuit delivers comparable results with only half the parameters. Consequently, we adopt the MERA-up circuit for QAG model training in our research.

4 Accuracy Analysis

In this section, we delve into a comprehensive evaluation of the QAG model's capabilities utilizing the MERA-up circuit architecture. Our analysis presents customary accuracy metrics pertinent to calorimeter simulation within the field of high-energy physics.

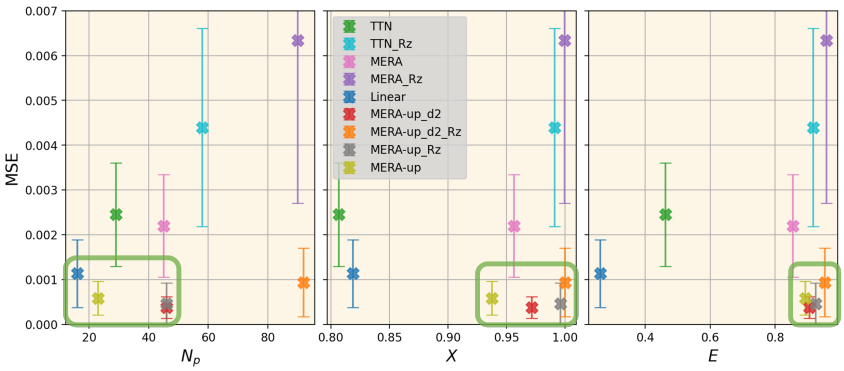

Figure 2. The uncertainty in Mean Squared Error (MSE), where lower values indicate better performance, across different trained circuit architectures is illustrated concerning N_p (left), X (middle), and E (right). Optimal regions are highlighted by green rectangles.

4.1 Training evaluation

The MMD loss remains consistently steady across all training repetitions, with its standard deviation (STD) tightening as training unfolds. Both the MMD and Corr losses exhibit smooth evolution, indicating the stability of the QML training process. It's worth noting that while the MMD loss takes precedence throughout the training journey, the Corr loss plays a pivotal role in generating precise physics within the simulated images.

In figure 3, the unweighted loss functions are showcased over training epochs. This visual representation aggregates twenty training runs into a bold line, complemented by a shaded band denoting the standard deviation (STD). Noteworthy is the fact that the Corr loss starts affecting training only from the 100th epoch onward, since it is initialised with weight zero.

4.2 Inference evaluation

We continue with a comparative analysis, evaluating images generated by the best-trained model against Geant4 test data, encompassing a dataset of 980 images.

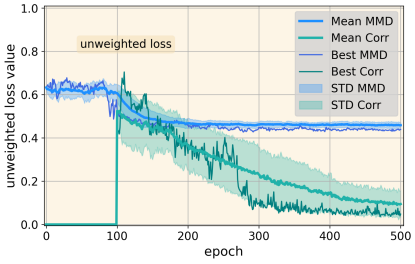

Figure 3. The average and optimal statistical metrics throughout the training process for both utilized loss functions. The correlation loss is introduced during the 100th epoch to enhance performance, while the training stabilizes and converges by the 500th epoch.

Focusing on the first aspect, the evaluation encompasses the average calorimeter shower shape, which is depicted on the left of figure 4. Impressively, the QAG model faithfully replicates the shower shape with a remarkable MSE of 0.00059 ± 0.00037.

Moving on, our analysis extends to pixel-wise image correlation, capturing both positive and negative patterns among pixels. Geant4's baseline correlation pattern is illustrated in centre of figure 4, whereas the QAG-generated data correlation is showcased to the right. The QAG model mirrors the overall correlation pattern, effectively reproducing intricate pixel relationships of the shower images.

Furthermore, the evaluation involves energy sums across individual image pixels. Figure 5 exhibits QAG's accurate replication of Geant4's Gaussian-shaped energy sum histogram, confirmed by the alignment of mean (μ) and standard deviation (σ) values.

Addressing the model's ability to capture specific image modes, we examine k-means clusters. Geant4 data undergoes k-means clustering, revealing four distinct image modes as shown in figure 5. These modes showcase varying energy fractions in different calorimeter cells. Our interest lies in assessing whether the QAG model can replicate this behavior, and indeed, the QAG-generated image's four clusters (figure 5) exhibit a comparable structure, demonstrating commendable accuracy in reproducing energy content and image modes.

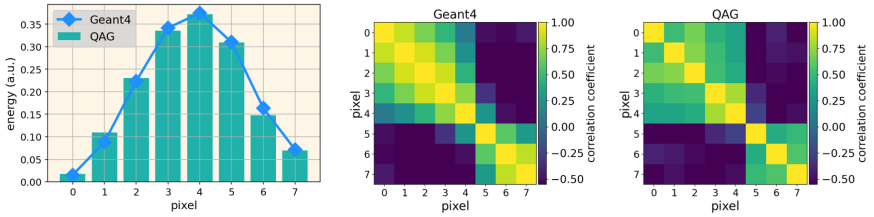

Figure 4. The left figure presents visualizations of the average calorimeter shower shapes. Energy is presented in arbitrary units (a.u.) due to image downsampling. In the center figure, a pixel-wise correlation plot is displayed between Geant4, and in the right figure, the same plot is generated for the QAG model.

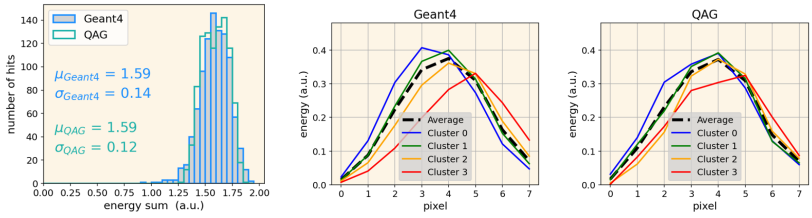

Figure 5. On the left, there is the energy sum histogram. In the center, the four k-means clusters that manifest within the images generated by Geant4. On the right, the same cluster analysis is presented for the QAG model, where the general shapes are correctly reproduced.

5 Quantum Noise Study

In the current NISQ era, high hardware error rates constrain the effective application of algorithms on quantum devices. Similar to classical scenarios, Quantum Machine Learning (QML) models display a degree of noise resilience amid hardware errors. Here, we examine the noise robustness of the QAG model, both in training and inference. Real quantum devices with measured noise levels are used for comparison with simulations.

5.1 Inference

Initial analysis applies quantum noise to inference in a noise-free model. Three noise scenarios are tested: simulated noise levels, simulated noise reflecting hardware traits, and real

quantum hardware noise. Simulated noise involves readout and inter-qubit connection errors. Hardware noise varies for qubits and gates.

Multiple noise levels (0% − 15% error) are tested using MSE for accuracy (blue triangle in figure 6). Configurations show mean of 20 images with standard deviation. Accuracy remains stable until about 1.5% noise. Green (readout only) is robust up to 8% noise. Blue (CNOT only) and orange (combined) fare worse.

The analysis expands to simulating `ibmq-montreal` device noise, having 27 qubits with average readout and gate noise (blue point in figure 6). It aligns with simulations, suggesting noise-resilient training for accurate noisy hardware inference.

Real `ibmq-montreal` inference (red point in figure 6) performs worse than simulations. This might be due to unaccounted two-qubit entanglement gates in quantum circuit decomposition, absent in hardware noise simulations.

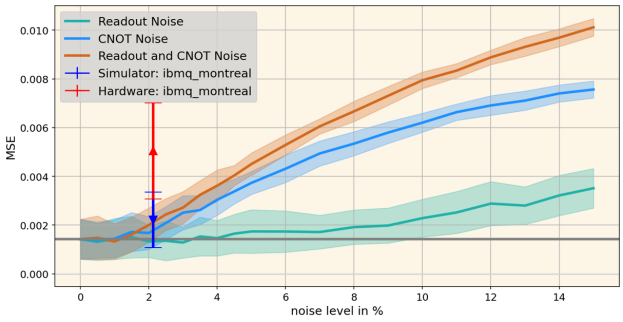

Figure 6. Exploring Noise levels during inference: Noise levels are plotted on the x-axis, while Mean Squared Error (MSE), a measure of accuracy (lower values are superior), is depicted on the y-axis. Inference is conducted across diverse noise configurations using real quantum hardware.

5.2 Training

The noise analysis extends to including noise during the training phase, to measure the QAG model's noise resilience on real quantum devices. Results are in figure 7. Both readout noise (green) and CNOT noise (blue) maintain accuracy up to around 3%. The combined noise (orange) shows a small decline from 1%, gradually decreasing. Yet, at 3% noise, accuracy remains near noiseless baseline within one standard deviation, suggesting noise-enhanced training improves model robustness.

Simulations involve two 27-qubit quantum devices: `ibmq-montreal` and `ibm-cairo`, the latter having an advanced processor. Training is repeated ten times, producing blue and orange points (figure 7). Hardware noise training slightly lags behind simulated noise model accuracy (orange line). Aligned noise simulation and hardware noise errors (orange and blue error bars) indicate insignificant accuracy differences.

During the final stage of training, we implement the model on real quantum devices. An interesting occurrence takes place while using the `ibmq-montreal` device, where unexpected spikes in readout noise, amounting to around 8%, emerge around epoch 280. These spikes have a noticeable impact on the MMD loss, as depicted in figure 7. However, the model demonstrates its adaptability by mitigating this loss over time.

Moving on to the training using the `ibm-cairo` device, represented by the green triangles, we observe a stable training process with no significant changes. The hardware noise in this case is slightly lower compared to the previously mentioned scenario.

Comparing the hardware-based training on `ibm-cairo` (green) to the simulations (orange), we find that the former exhibits slightly lower noise. This outcome reinforces the idea

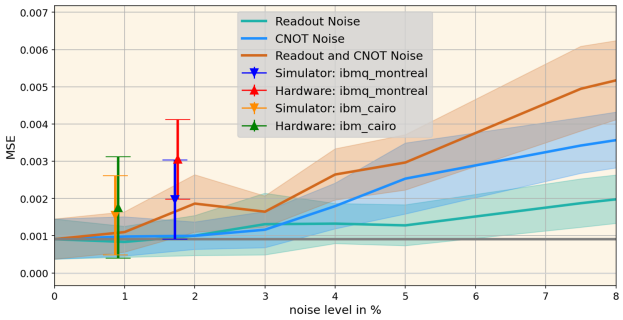

Figure 7. Exploring Noise during training: An investigation involving diverse noise configurations, encompassing hardware noise simulations as well as training on actual hardware. The training iterations with artificial noise levels are conducted ten times.

that the alignment between the hardware and the simulation holds true, especially at low noise levels.

Real hardware training surpasses prior inference in absolute MSE (≈ 0.002 and ≈ 0.003 vs. ≈ 0.005), showcasing QAG's adaptability in noisy hardware. `ibmq-montreal` training initially declines but recovers, supporting QAG's resilience.

6 Conclusion

This study establishes the competence of the recently developed QAG model in producing images with remarkable precision, as corroborated by diverse validation metrics. Beyond accurately capturing average values, the model adeptly reproduces intricate pixel-wise correlations using the optimal MERA-up quantum architecture, indicating its capability to discern intrinsic correlation patterns from training data.

The investigation underscores the notable influence of quantum hardware noise on quantum machine learning model accuracy. Notably, training models with noise yields improved performance (stable until 3% noise), as the QAG model adapts to noise patterns, resulting in faster convergence—contrasting the outcome of noise application solely during inference (stable until 1.5% noise). This observation is consistent with `ibm-cairo` hardware training outcomes. Moreover, the study underscores the QAG model's robustness, showcasing its ability to yield accurate results despite significant hardware calibration shifts of up to 8% noise, as demonstrated in the `ibmq-montreal` training.

In essence, the novel QAG model's resilience when trained with realistic quantum hardware noise highlights its potential to establish sturdy models delivering accurate outcomes. This outcome holds crucial implications for the advancing landscape of real-world quantum machine learning applications.

References

[1] M. Zidan, H. Eleuch, M. Abdel-Aty, Results in Physics **21**, 103536 (2021)
[2] J. Preskill, Quantum **2**, 79 (2018)
[3] H.Y. Huang, M. Broughton, M. Mohseni et al., Nature Communications **12** (2021)
[4] L. Clissa, *Survey of big data sizes in 2021* (2022)
[5] CERN, *Worldwide lhc computing grid*, accessed in June 2022, `https://wlcg-public.web.cern.ch/`

[6] H. Zheng, Z. Li, J. Liu, S. Strelchuk, R. Kondor, *On the super-exponential quantum speedup of equivariant quantum machine learning algorithms with su(d) symmetry* (2022), https://arxiv.org/abs/2207.07250

[7] S.L. Wu et al., PoS **EPS-HEP2021**, 842 (2022)

[8] A. Khrennikov, The European Physical Journal Special Topics **230** (2021)

[9] M. Weigold, J. Barzen, F. Leymann, M. Salm, *Data Encoding Patterns for Quantum Computing*, in *Proceedings of the 27th Conference on Pattern Languages of Programs* (The Hillside Group, USA, 2020), PLoP '20, ISBN 9781941652169

[10] M. Weigold, J. Barzen, F. Leymann, M. Salm, *Expanding Data Encoding Patterns For Quantum Algorithms*, in *2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C)* (2021), pp. 95–101

[11] J.G. Liu, L. Wang, Physical Review A **98** (2018)

[12] A. Khoshaman, W. Vinci, B. Denis, E. Andriyash, H. Sadeghi, M.H. Amin, Quantum Science and Technology **4**, 014001 (2018)

[13] J.X. Tong Li, Shibin Zhang, Computers, Materials & Continua **64**, 401 (2020)

[14] F. Rehm, S. Vallecorsa, K. Borras, D. Krücker, *Quantum machine learning for HEP detector simulations* (2021), http://ceur-ws.org/Vol-3041/363-368-paper-67.pdf

[15] P.L. Dallaire-Demers, N. Killoran, Physical Review A **98** (2018)

[16] *Blinded citation for review* (-), -

[17] S. Agostinelli et al., *GEANT4–a simulation toolkit* (2003), Vol. 506, pp. 250–303

[18] A.e.a. Albrecht, Alvesand, *A Roadmap for HEP Software and Computing Research and Development for the 2020s* (2019), Vol. 3

[19] J. Apostolakis, *Detector Simulation* (Springer International Publishing, Cham, 2020), pp. 485–531, ISBN 978-3-030-35318-6

[20] I. Kadochnikov, I. Bird, G. McCance, J. Schovancova, M. Girone, S. Campana, X.E. Currul, Advisory committee p. 127 (2018)

[21] X. Ju, D. Murnane, P. Calafiura et al., The European Physical Journal C **81**, 1 (2021)

[22] C. Biscarat, S. Caillou, C. Rougier, J. Stark, J. Zahreddine, *Towards a realistic track reconstruction algorithm based on graph neural networks for the HL-LHC*, in *EPJ Web of Conferences* (EDP Sciences, 2021), Vol. 251, p. 03047, https://www.epj-conferences.org/articles/epjconf/pdf/2021/05/epjconf_chep2021_03047.pdf

[23] E. Sela, S. Huang, D. Horn, Algorithms **15**, 115 (2022)

[24] C. Fabjan, F. Gianotti, Reviews of Modern Physics **75** (2003)

[25] M. Pierini, M. Zhang, *CLIC Calorimeter 3D images: Electron showers at Fixed Angle* (2020), https://doi.org/10.5281/zenodo.3603122

[26] F. Rehm, *Downsampled Calorimeter Shower Images to 8 Pixels* (2021), https://doi.org/10.5281/zenodo.7025233

[27] A. Gretton et al., J. Mach. Learn. Res. **13**, 723 (2012)

[28] T. Hofmann, B. Scholkopf, A. Smola, Annals of Statistics **36**, 1171 (2007)

[29] A.H. Alhabsi, *Improved SPSA optimization algorithm requiring a single measurement per iteration*, in *10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010)* (2010), pp. 263–265

[30] T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, *Optuna: A Next-generation Hyperparameter Optimization Framework* (2019), 1907.10902