Fraunhofer
IKS

Fraunhofer Institute for Cognitive Systems IKS

BenchQC – Scalable and modular benchmarking of industrial quantum computing applications

Speaker: **Dr. Florian Geissler** (Fraunhofer IKS)

TQCI Seminar, Palaiseau, June 2025

Introduction

BenchQC
Consortium

Industrial QC applications

Building full-stack quantum computers

Lighthouse project **BenchQC**

Benchmarking of industrial applications

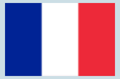
(**NEW**) Whitepaper: https://arxiv.org/abs/2504.11204

  Public

https://arxiv.org/abs/2403.12205
https://arxiv.org/abs/2504.11204

Quantum computing benchmarking

A global view (selected)

	Quantum Economic Development Consortium (QED-C)	BACQ: Application-oriented benchmarks for quantum computing	MQV - BenchQC
Geo:			
Established:	2018	2023	2022
Goals and focus:	▪ Identify high impact use cases ▪ Identify gaps in technologies, standards, metrics, workforce ▪ Work with stakeholders in industry, academia, government to fill gaps	▪ Benchmarks close to real applications to guide industrial end-users ▪ MYRIAD tool for aggregated metrics	▪ Modular and scalable benchmarking using full hardware-software stack (MQSS) ▪ User-friendly and open-source tool ecosystem ▪ Explore hardware diversity
Focus:	▪ Metrics close to hardware (QV), ▪ Well-established application subroutines (e.g., phase estimation, amplitude estimation)	linear systems, optimization, factorization	▪ Representative spectrum of QC and generalized metrics for hybrid solutions ▪ Industrial applications incl. QML

> QML in BenchQC:
> - Use cases
> - Metrics
> - Integration in QUARK

Emerging technology:
Quantum computing

With more powerful hardware emerging, quantum computing (QC) has the potential to lead to **disruptive** changes in many industrial areas:

- **Simulation** of quantum mechanical systems (development of new drugs, chemical sector with battery development,…)
- **Optimization** problems (Logistics, production, pharma,…)
- **Quantum machine learning** (Computer vision, mobility,…)

But for which applications will QC really offer an advantage in practice?

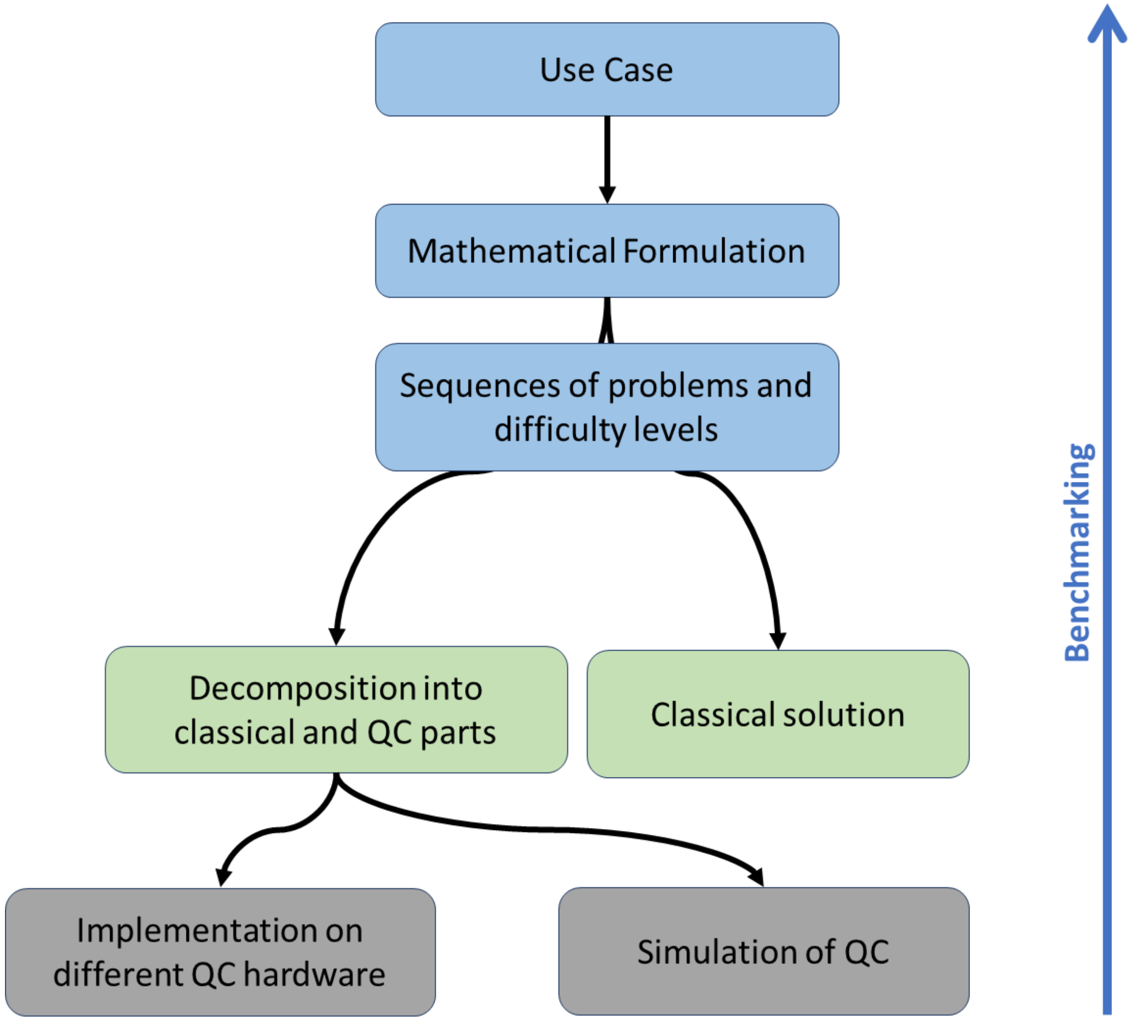
→ Application-driven benchmarking procedure required!

Application-driven benchmarking of quantum computers

Which QC-technology will be suited for which application use case and when?
→ Potential analysis from the application perspective

Application-driven benchmarking of QC:

- Identification of use cases suited for a **practical** quantum advantage.
- Consider mathematical **decomposition** and **sequence scaling** of problem
- Which **QC-hardware** technology (in which size, in which quality)?
- Considering the **full path** to the solution of a problem (e.g. including the data transfer to quantum computers, and post-procession of a quantum solution and integration into an industrial workflow)

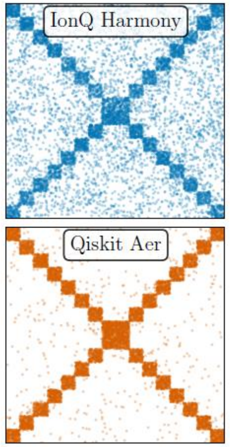


  Public

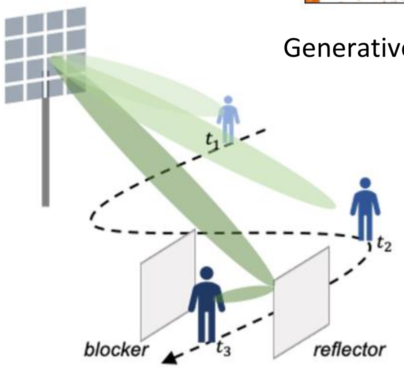
BenchQC

Use cases

Crack detection



Generative modeling



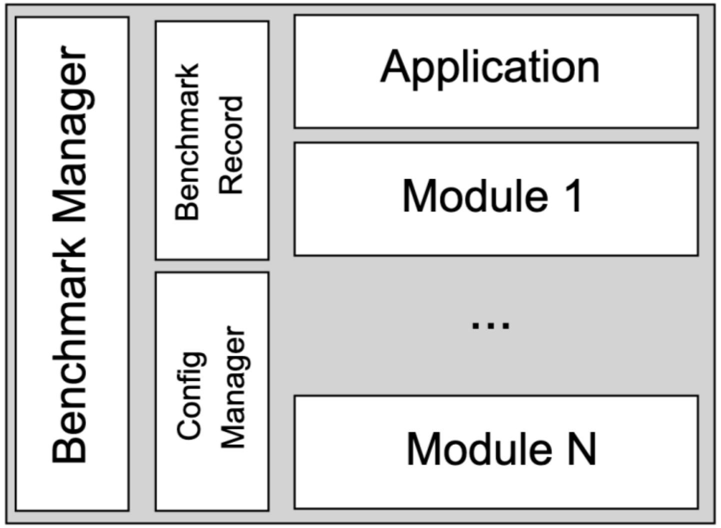
Beam management

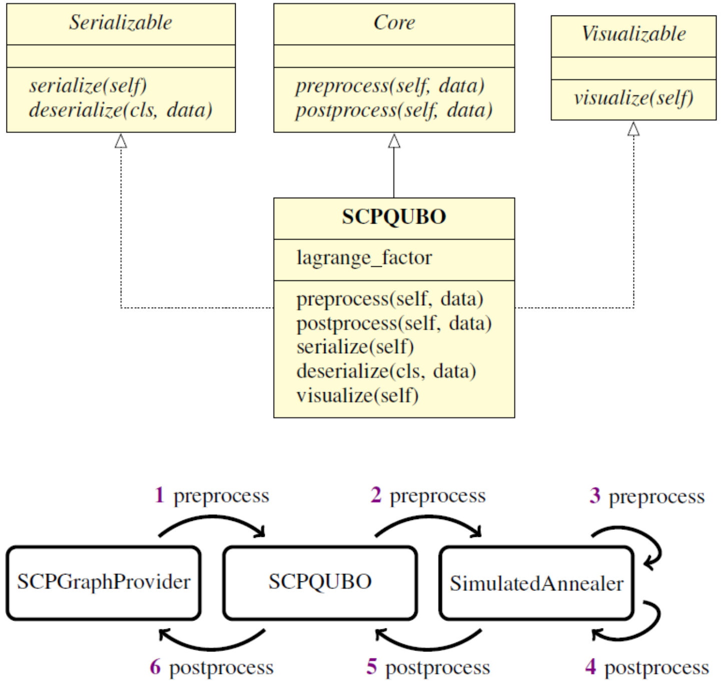
Category	Model	Use case
QML	Classification	Defect detection
	Generative models	Generative design for constructional elements
		Data augmentation for quality inspection
	QRL	6G Beam management
Optimization	VQE, QAOA, Quantum Annealing	Multipath-connectivity across aerial vehicles
		Sensor placement in water networks
		Software verification strategies
		LiDAR sensor configuration
		Assembly line scheduling and packet sequencing
		Financial asset optimization
Simulation	Hamiltonian simulation	Dynamics of electrons in materials
		Low-temperature state preparation of materials
		Nuclear magnetic resonance simulation

https://arxiv.org/abs/2504.11204

QUARK

https://quark-framework.readthedocs.io/en/dev/
https://github.com/QUARK-framework/QUARK





Now (2.1.5)

- Modular structure inheriting from abstract classes
- Integrated metrics
- Hardware interfaces via Amazon bracket
- Datasets via git lfs

Future (>3)

- Modular structure via plugins, no core modifications
- Integrated metrics
- More diverse hardware interfaces
- Datasets handled by individual user

Benchmarking along the solution pipeline

QML metrics

*Implemented in Quark for selected applications

Hardware	Quantum Circuit	Resources	Performance	Problem Complexity
▪ Number of qubits* ▪ Coherence times ▪ Gate fidelities ▪ Quantum volume ▪ Q-factor	▪ Expressibility* ▪ Entanglement* ▪ Normalized effective dimension ▪ Fourier composition	▪ Sample complexity* ▪ Learning steps* ▪ Number of shots* ▪ Classical-quantum interactions	Application-specific performance: ▪ QML* ▪ Optimization* ▪ Simulation Generic performance: • Runtime*	▪ Input/output scaling* ▪ Model scaling/non-linearities ▪ Feature ranking/selection ▪ Sensitivity to input perturbations*

Component level (Hardware)

System level (Hardware, Quantum Circuit, Resources)

Application level (Performance, Problem Complexity)

QCNN – Use case study

[1] Bowles et al., 2024 (arxiv.org/pdf/2403.07059)
[2] Matic et al., 2023, Monnet et al. 2024
[3] Schreiber et al., 2023

QML challenges

… and research opportunities

Studies of classical vs. quantum-neural networks

- Out-of-the-box classical models **outperform** quantum models [1]
- Metrics such as entanglement, expressibility are **not always representative** of circuit performance [1, 2]
- Performant quantum models can often be mapped to simpler **classical surrogate** models [3]
- Classical-convolutional hybrids can reach equivalent classical performance with **less data/parameters/steps** in some cases [2]

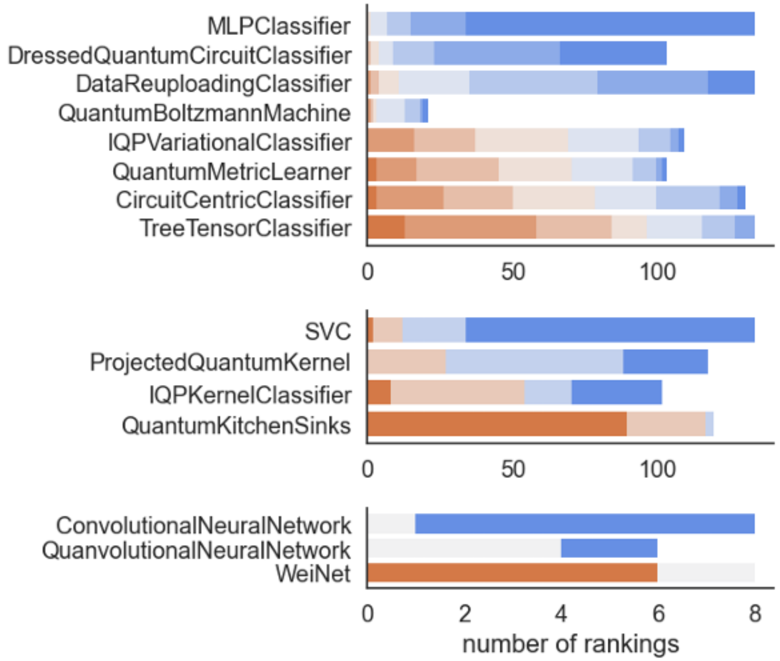


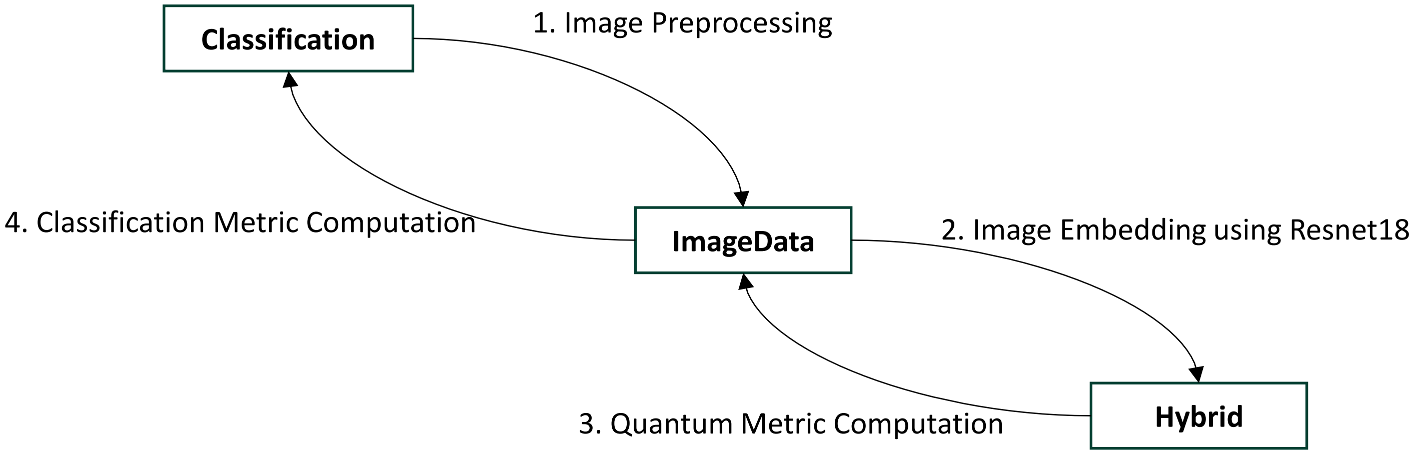
FIG. 10. Number of rankings (blue/first to red/last) across all datasets that a model was tested on, for the three model families. The models were sorted from top to bottom according to the average normalised rank. The three classical out-of-the-box models perform best. Note that the total number of benchmarks a model competed in varies due to runtime limitations, and since the convolutional architectures were only tested on MNIST CG and BARS & STRIPES.

Quantum-Classical Image Classification

Use Case: Surface Crack Detection

Concrete patches, with or without crack

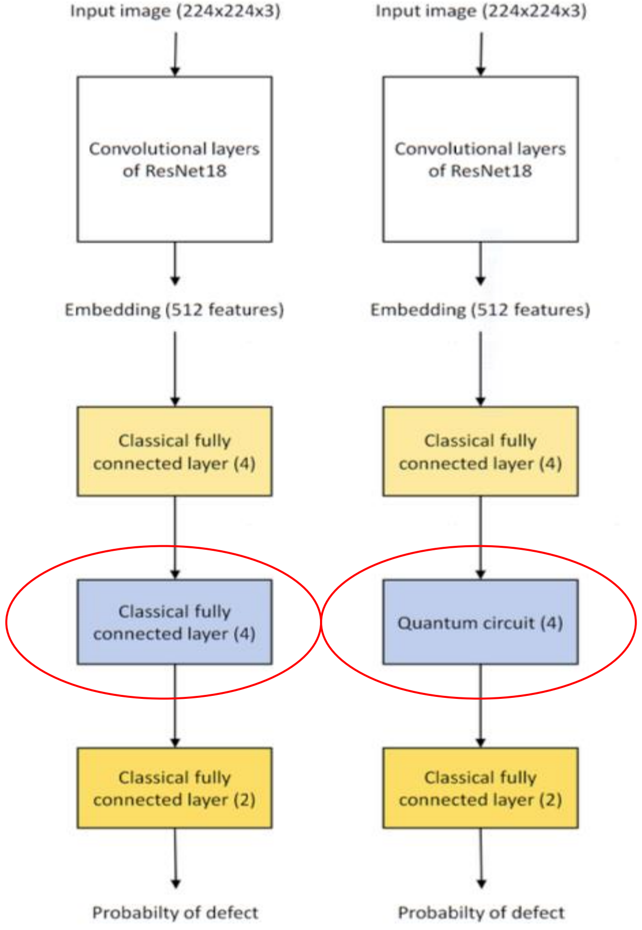
- Binary classification problem
- Image size 224x224



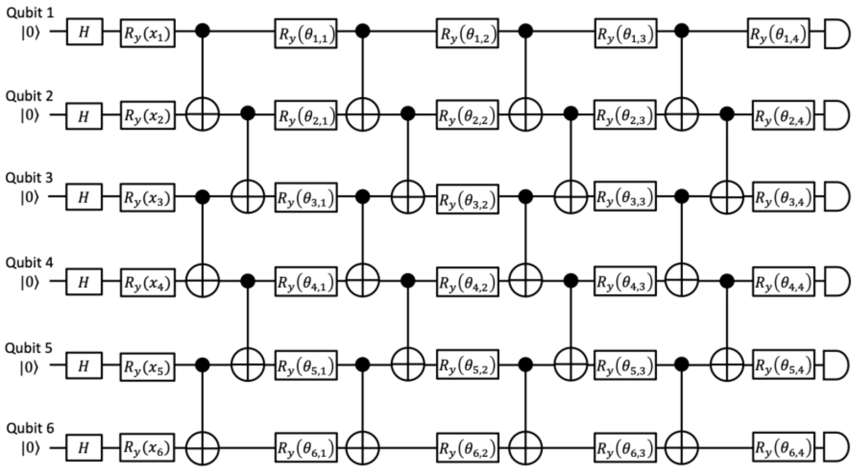
Modular structure of image classification use case in QUARK

QUARK implementation

System design



Quantum layers of the hybrid neural network (6 qubits)



- Angle encoding
- Parametrized quantum circuit
- Strongly entangled architecture

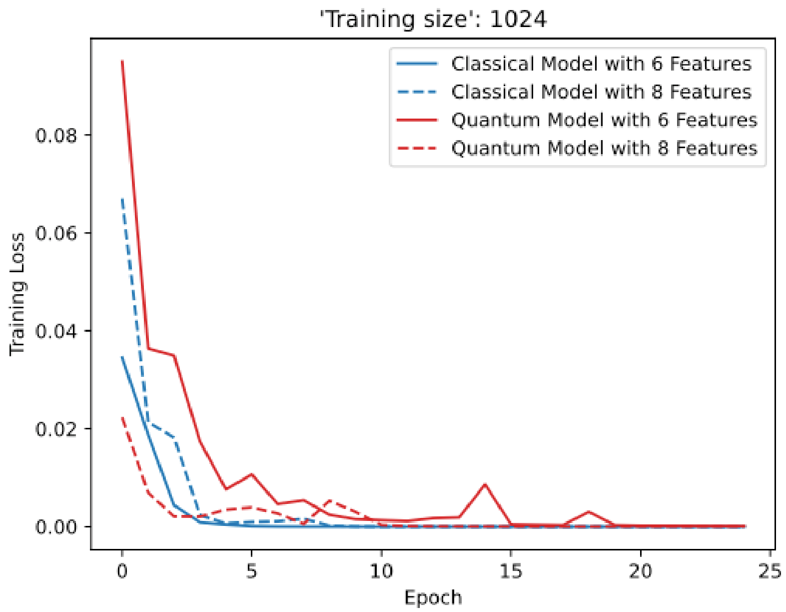
Classical benchmark using 4 features(left),
Hybrid architecture using 4 qubits (right)

Model Performance Comparison

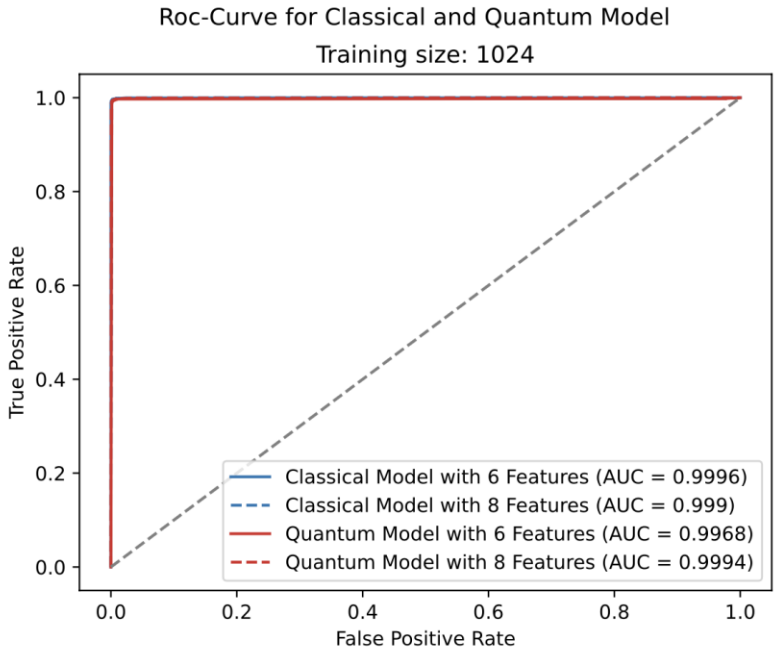
Quantum vs Classical

Features after dimensionality reduction (qubits)	Classification accuracy of the hybrid model	Classification accuracy of the classical model
2	-	0.999452
4	-	0.999635
6	0.996825	0.999645
8	0.999359	0.999037

Accuracy value of both models with 1024 training images per label



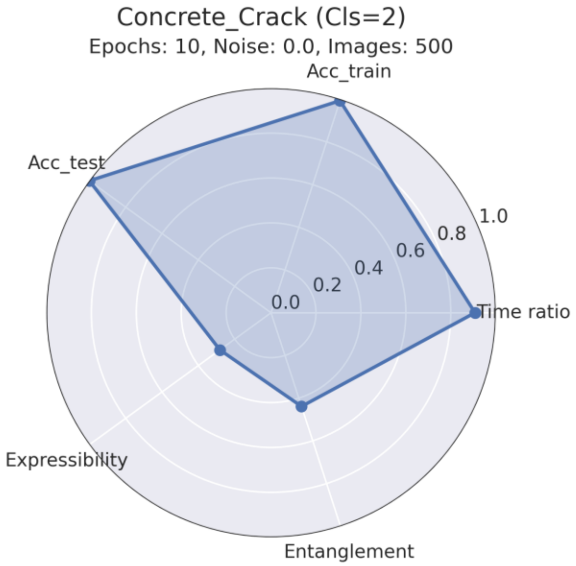
Training loss evolution of the classical model (in blue) and the quantum model (red) for a training size of 1024 images

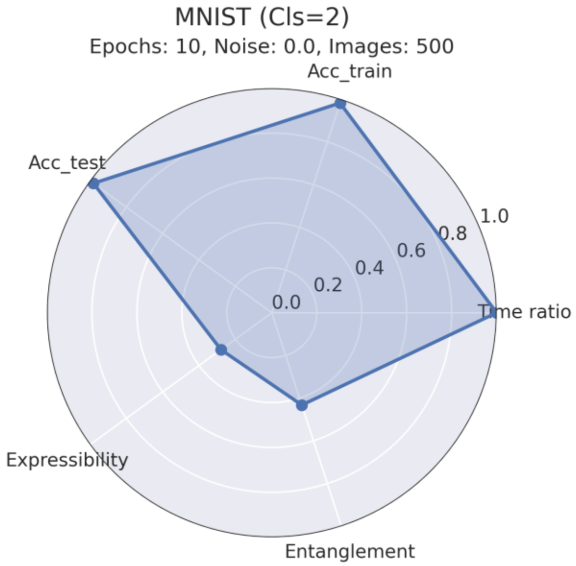


ROC curve for the hybrid and fully classical models

Evaluation

QCNN use case

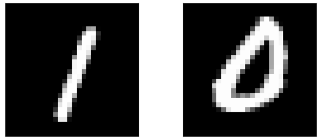




General findings

- High accuracy at low entanglement possible
- Quantum layer is compute-heavy
- Simple circuit design has low expressibility
- Problem complexity can be adressed by noise, rescaling

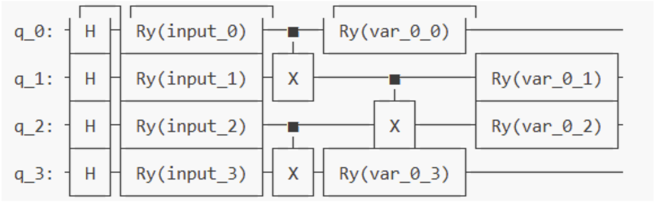
Road cracks



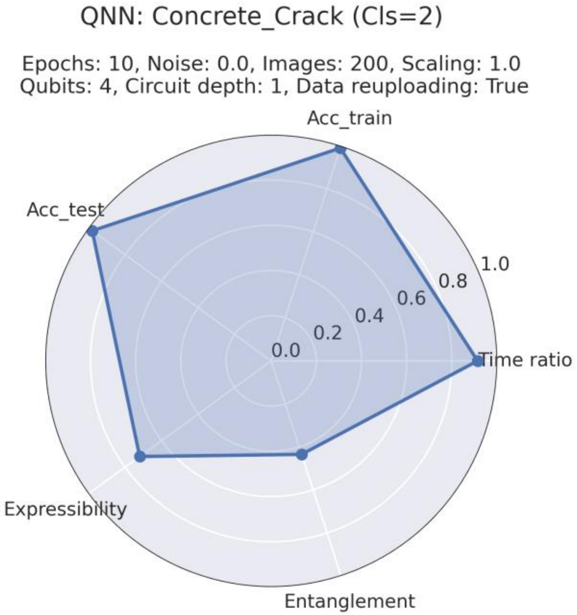
Handwritten digits (binary)

*Time ratio: Execution time of Quantum Circuit/Overall execution time

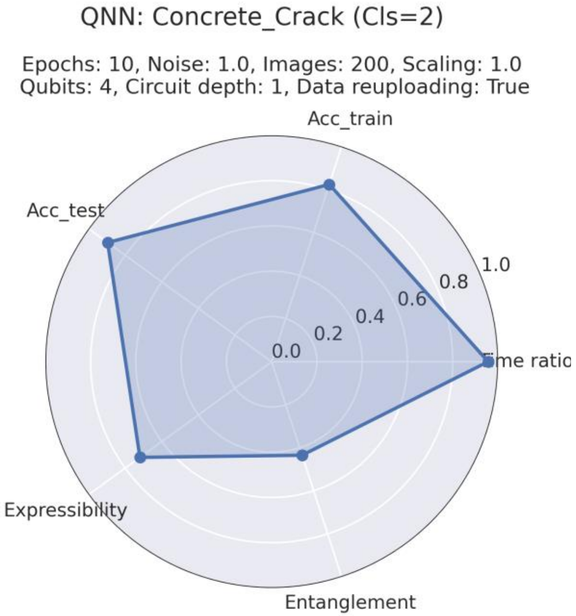
Noise variation study

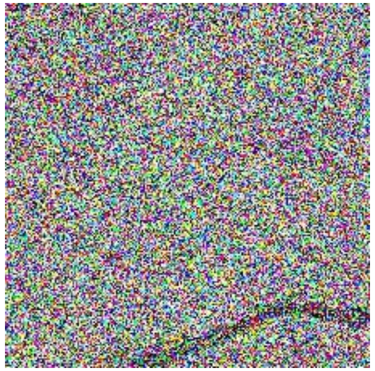


Noise = 0

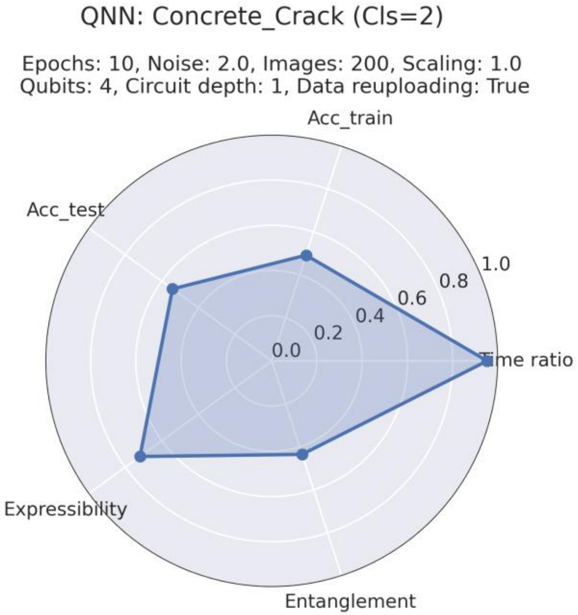


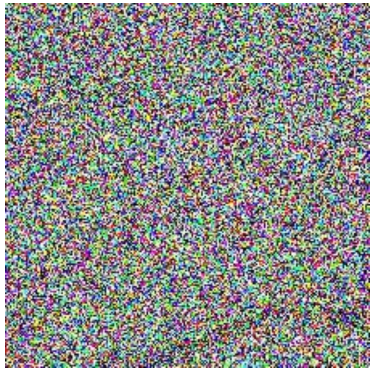
Noise = 1





Noise = 2

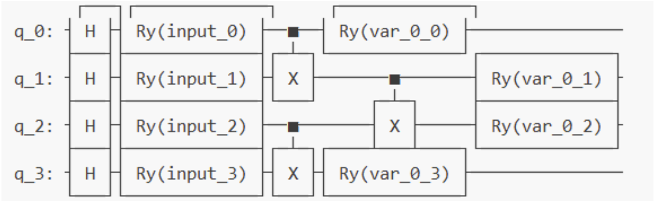




Observations

- **Addition of noise** is an effective measure to gradually increase problem complexity
- **Fixed**:
  - Circuit design
  - Number of training epochs

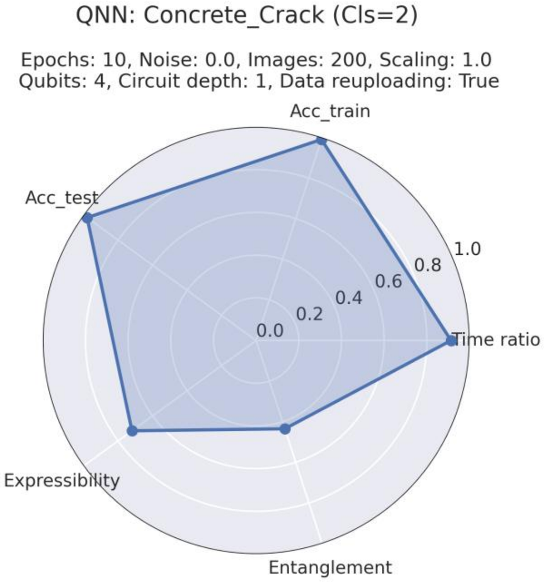
Image rescaling study



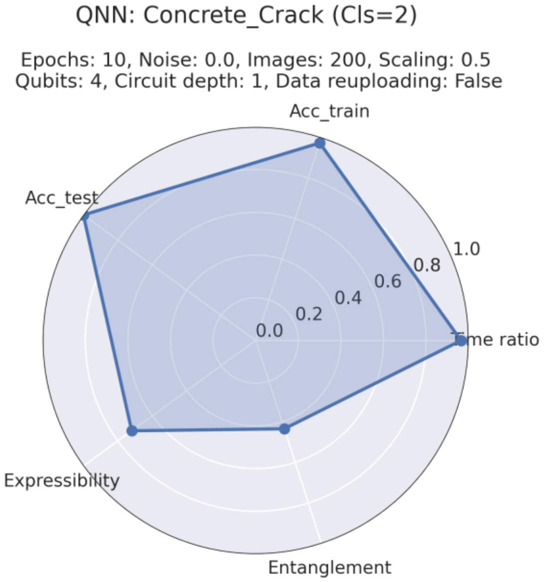
Observations

- Image **rescaling** can be tolerated to some extent, but eventually accuracy collapses
- **Fixed**:
  - Circuit design
  - Number of training epochs

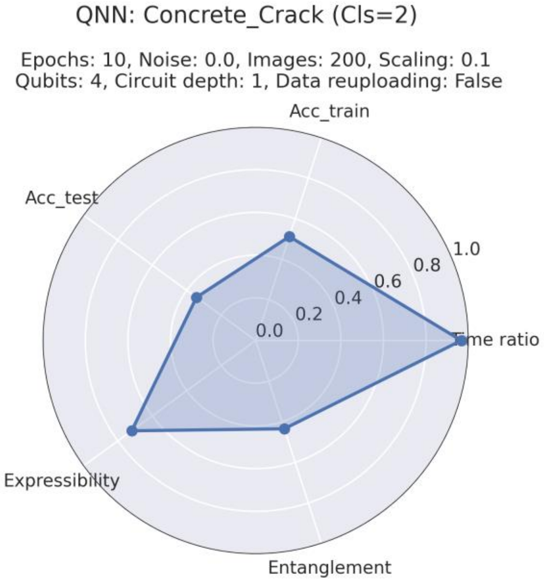
Scale_factor = 1.0

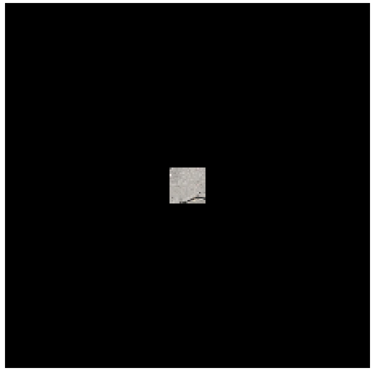


Scale_factor = 0.5

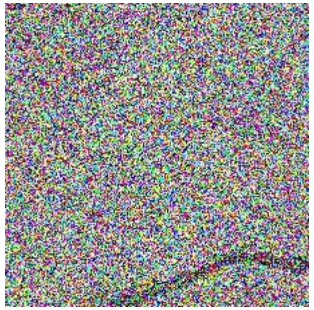


Scale_factor = 0.1

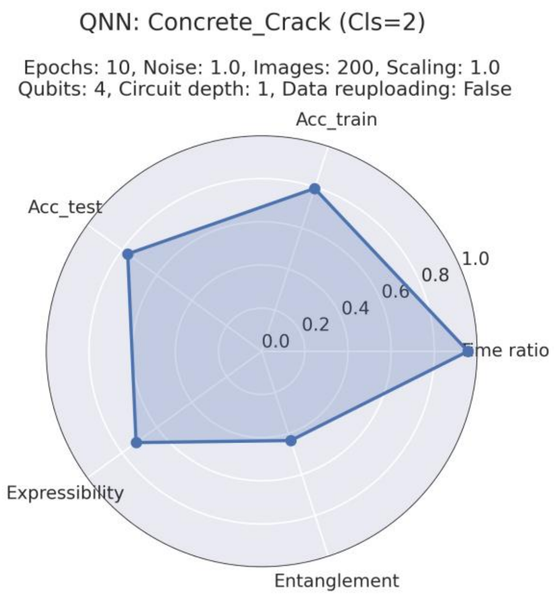




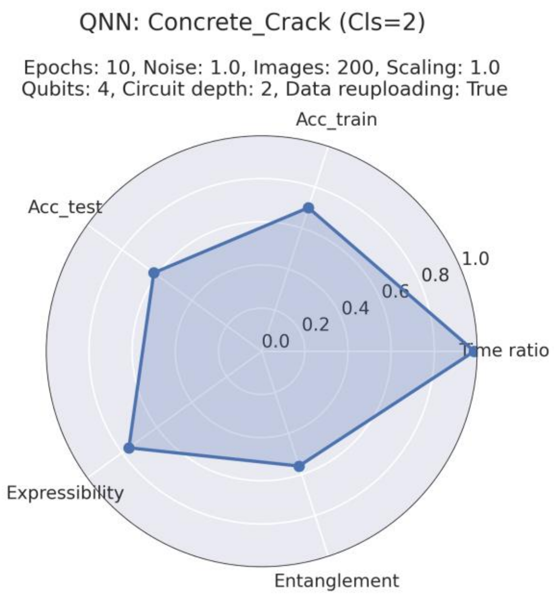
Circuit depth study
(with data reuploading)



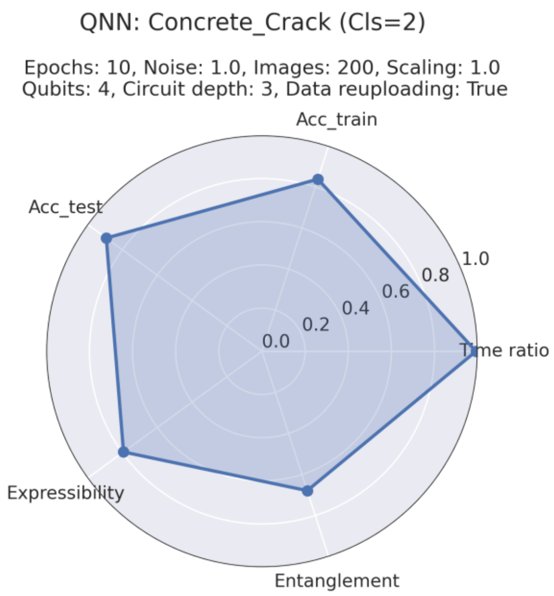
d = 1

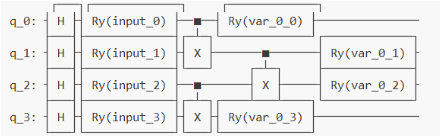


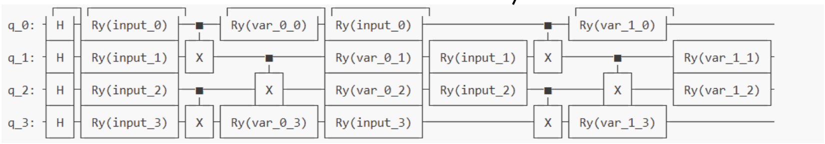
d = 2

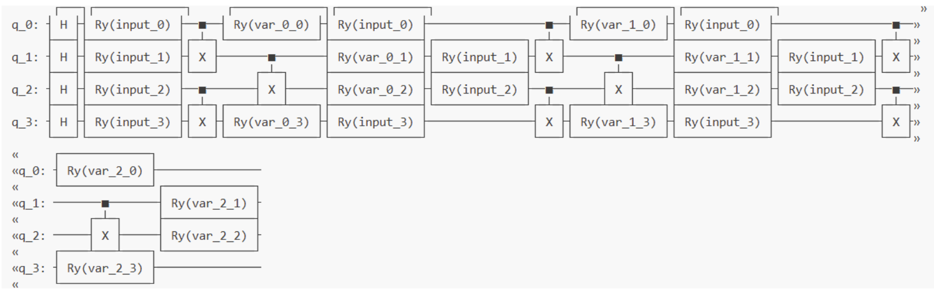


d = 3









Observations

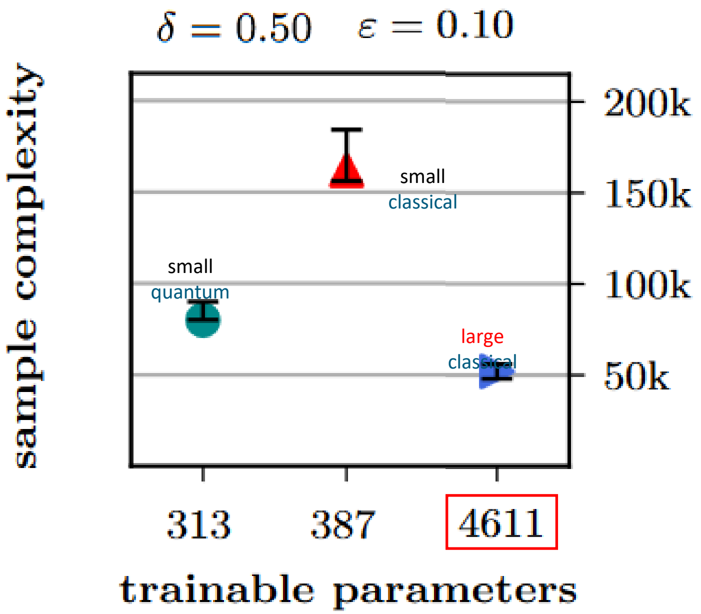
- Deeper circuits **increase expressibility and entanglement**
- Accuracy **not necessarily** much better
- Fixed: 10 Epochs of training, average of 3 runs

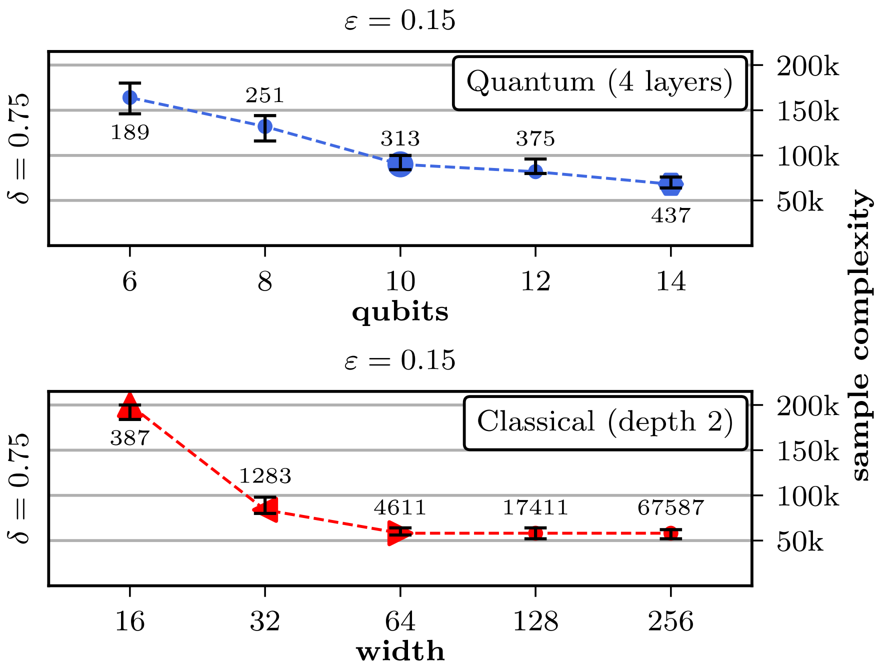
QML beyond classification

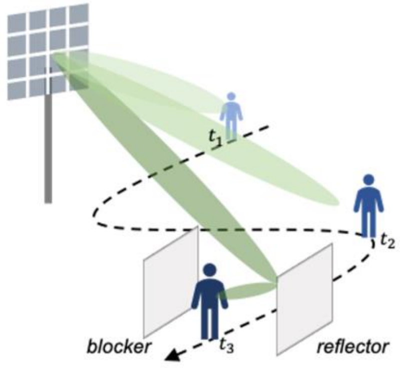
arXiv:2501.15893 (2025)
arXiv:2404.06314, IEEE QCE (2024)

Example: Quantum Reinforcement Learning

Robust benchmarking of quantum reinforcement learning





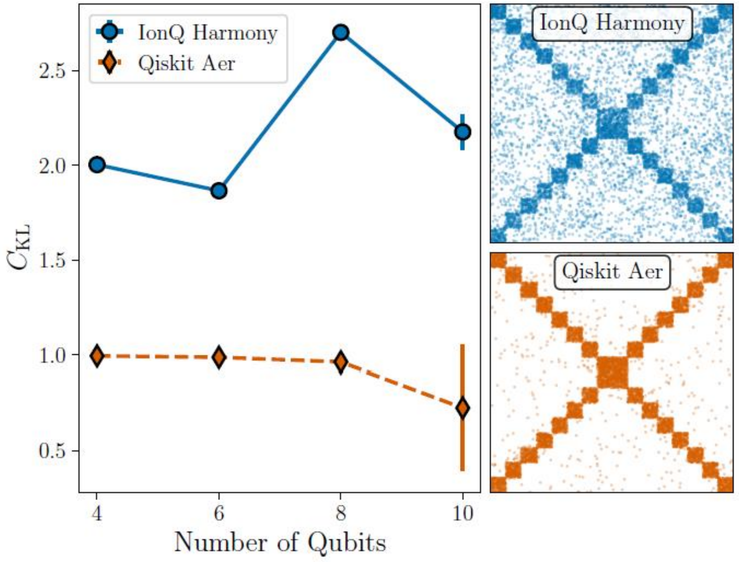


- **Use case**: Beam management problem: reinforcement learning to find best connectivity
- 1-ϵ: Chosen performance threshold wrt optimal solution, δ confidence probability
- Metric: **sample complexity**. Small quantum model performs similar to large, classical model (in number of trainable parameters)

Example: Generative machine learning with QUARK

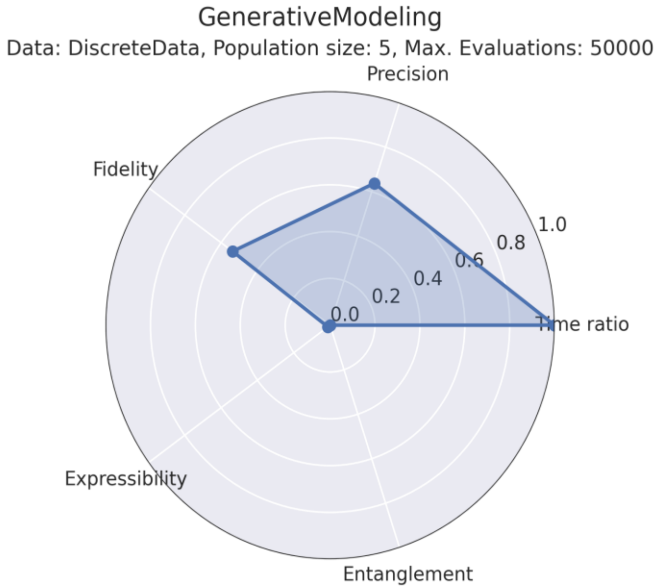
Kiwit et al., 2023
Kiwit et al. 2024

Example problem:
Training a Quantum circuit Born machine (QCBM) model to the „X“ dataset.



KL: Kullback-Leibler divergence

- QCBM or GAN implementations, discretized data
- Comparing Qiskit Aer- simulator and IonQ Harmony:
  - More qubits do not necessary lead to a better solution quality
  - Different noise types hinder model training in different ways
  - Realistic (NISQ) noise intensities have significant impact already



  Public

Summary

- To benefit from quantum computing in real applications, a **practical quantum advantage** is required
- This advantage will not only depend on quantum hardware properties, but on **the full hardware and software stack**
- We need the **right tools** to do holistic benchmarking (e.g. QUARK)
- In the field of QML, a practical quantum advantage is still unclear, yet promising trends emerge. Benchmarking can guide the way to **best combinations of algorithms, metrics, hardware, etc.**
- Results can help establish a route towards **standards**

Application-driven benchmarking required

Backup