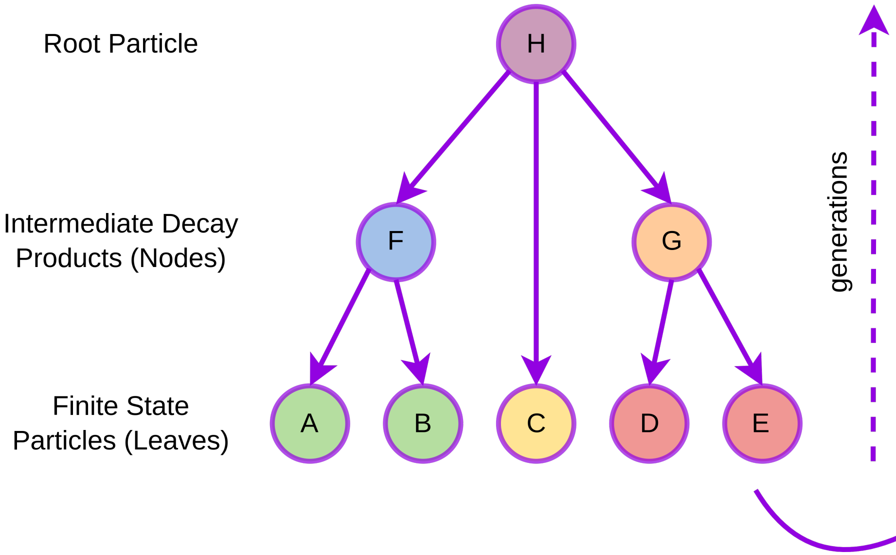
Improving Noisy Hybrid Quantum Graph Neural Networks for Particle Decay Tree Reconstruction

Karlsruhe Institute of Technology

Melvin Strobl, Eileen Kühn, Max Fischer, Achim Streit

Problem Statement



Feature Matrix

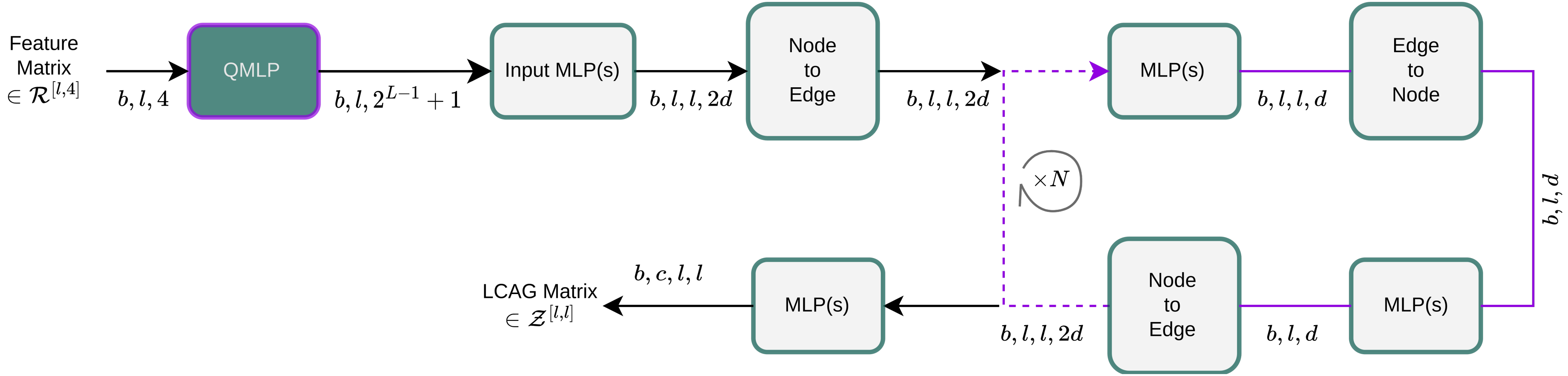
	E	p_x	p_y	p_z
A	E^A	p_x^A	p_y^A	p_z^A
B	E^B	p_x^B	p_y^B	p_z^B
C	E^C	p_x^C	p_y^C	p_z^C
D	E^D	p_x^D	p_y^D	p_z^D
E	E^E	p_x^E	p_y^E	p_z^E

Lowest Common Ancestor Generation Matrix [2]

	A	B	C	D	E
A	0	1	2	2	2
B	1	0	2	2	2
C	2	2	0	2	2
D	2	2	2	0	1
E	2	2	2	1	0

[1] phasespace: n-body phase space generation in Python - Navarro and Eschle - 2019
[2] Learning tree structures from leaves for particle decay reconstruction - Kahn et al. - 2022

Architecture

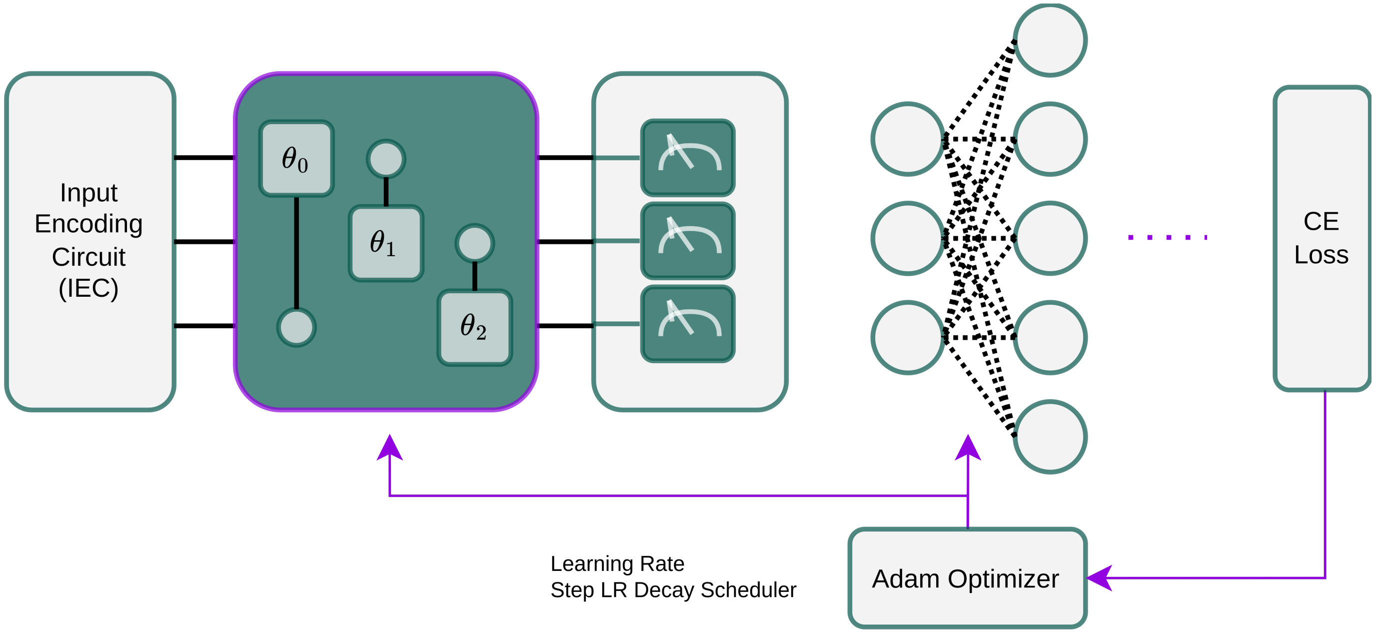


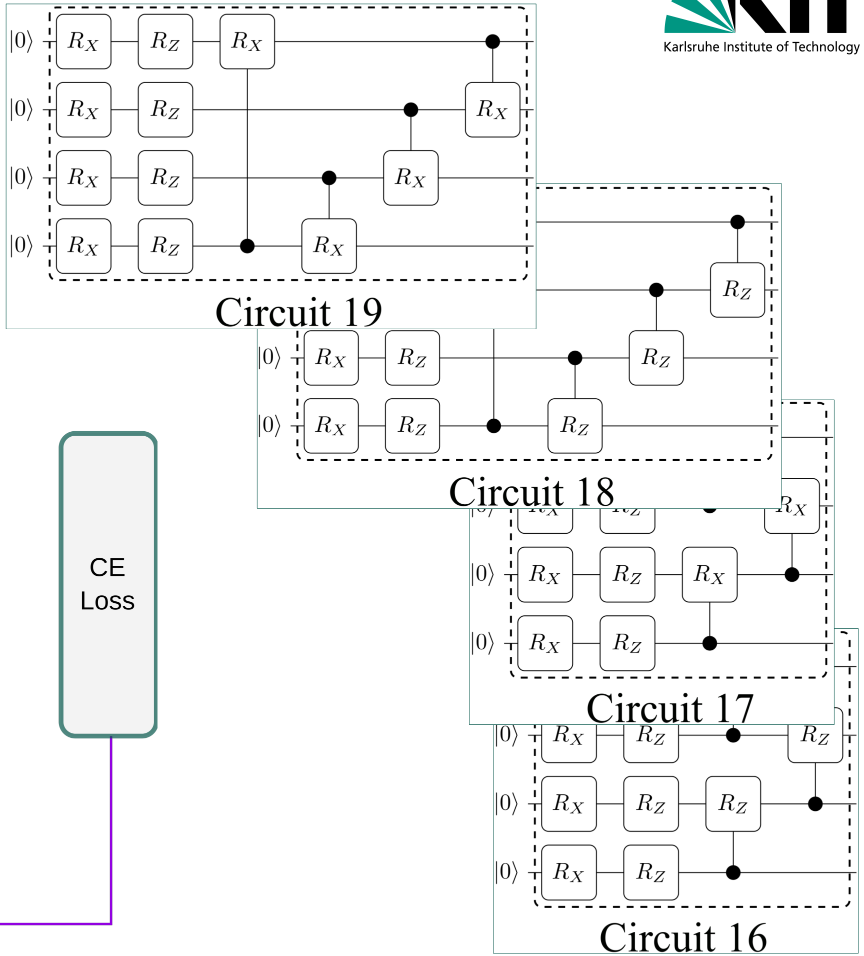
- Built on Neural Relational Inference encoder with Message Passing
- Existing work done in *Kahn et al. - Learning tree structures from leaves for particle decay reconstruction*
- *b*: Batch
- *l*: num. of FSPs in current sample
- *L*: total num. of FSPs
- *d*: hidden dimension
- 4 Momenta

[2] Learning tree structures from leaves for particle decay reconstruction - Kahn et al. - 2022
[3] Neural Relational Inference for Interacting Systems - Kipf et al. - 2018

Variational Circuit

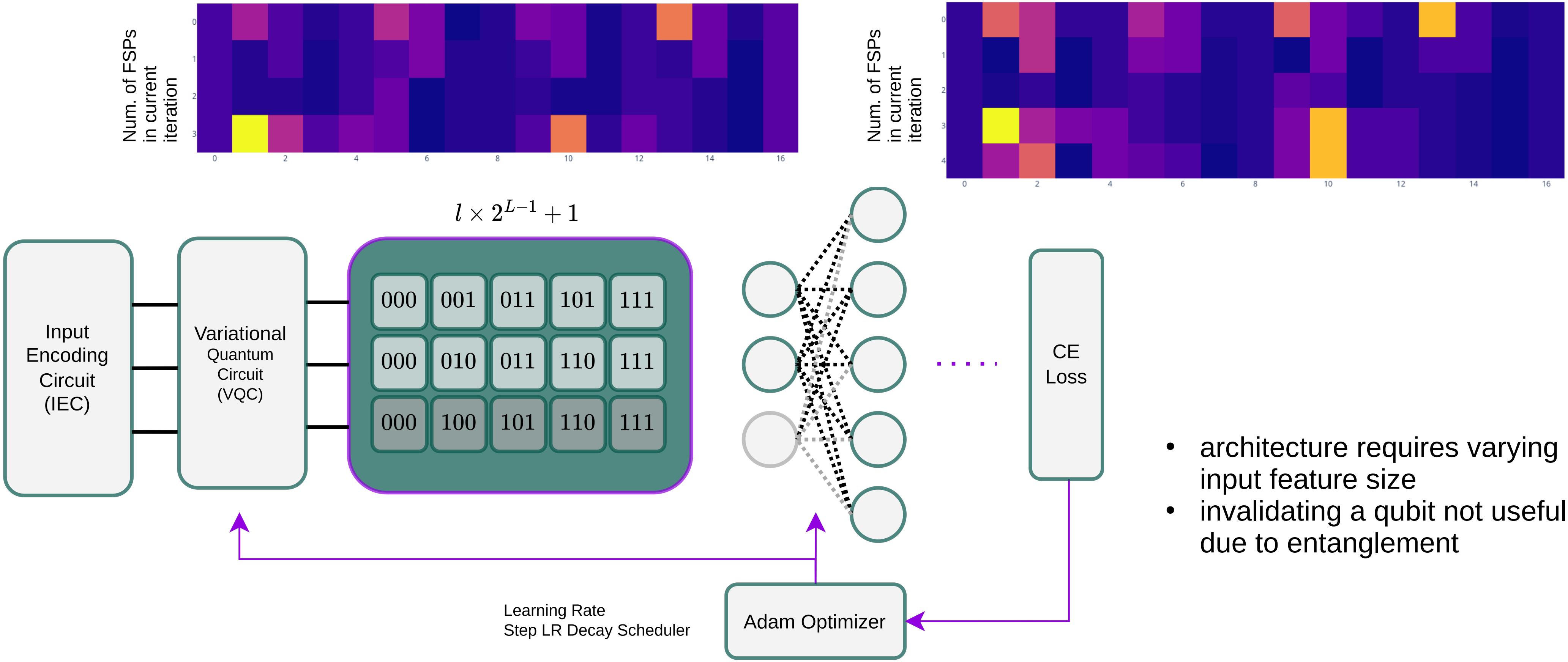
- Angle encoding for input feature data
- Multiple layers with (opt.) data-reupload





[3] Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms - Sukin Sim - 2019

Measurement Interpretation

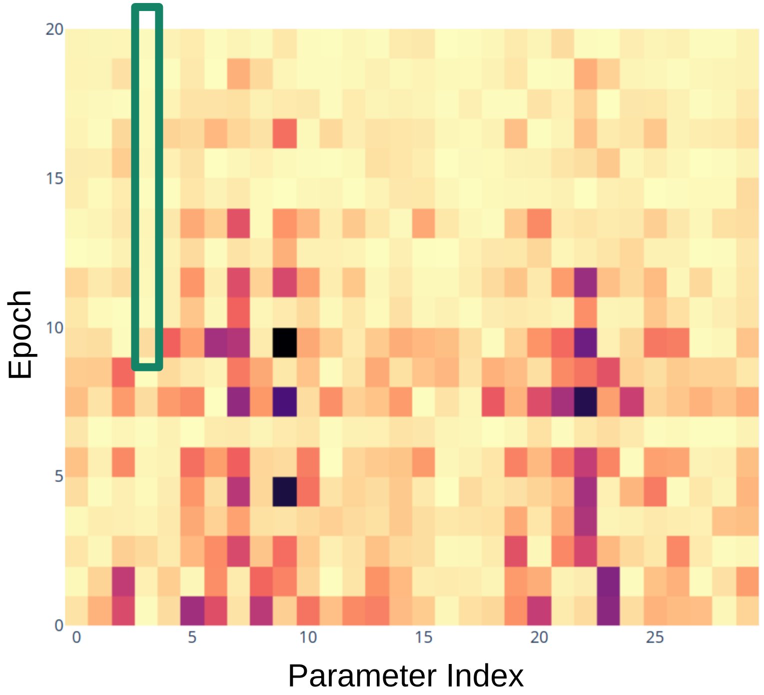


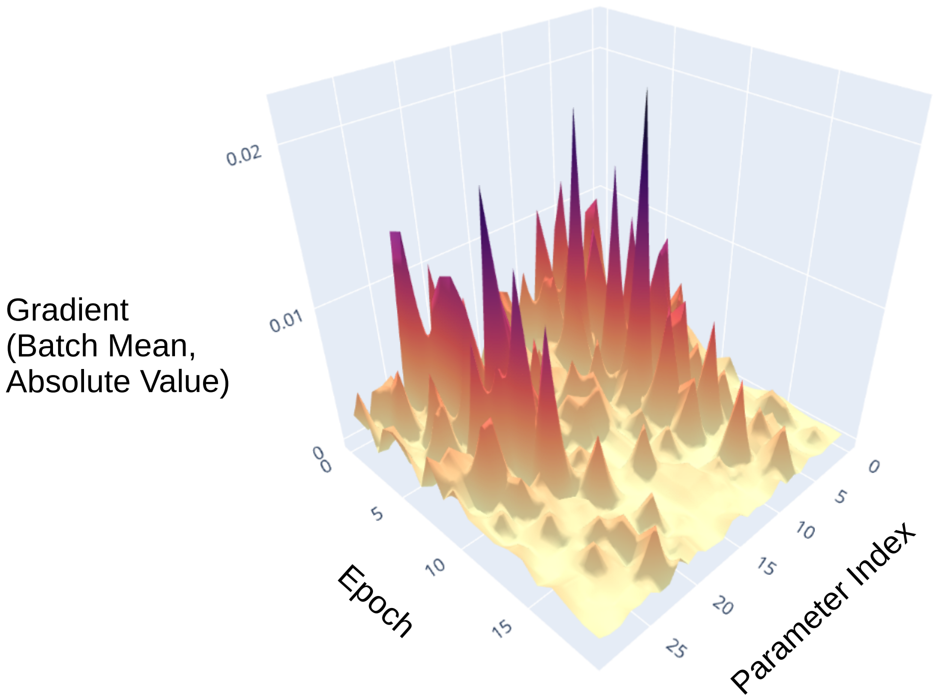
- architecture requires varying input feature size
- invalidating a qubit not useful due to entanglement

Gradients

- Calculating gradients using Parameter-Shift Rule
- Requires two evaluations per parameter

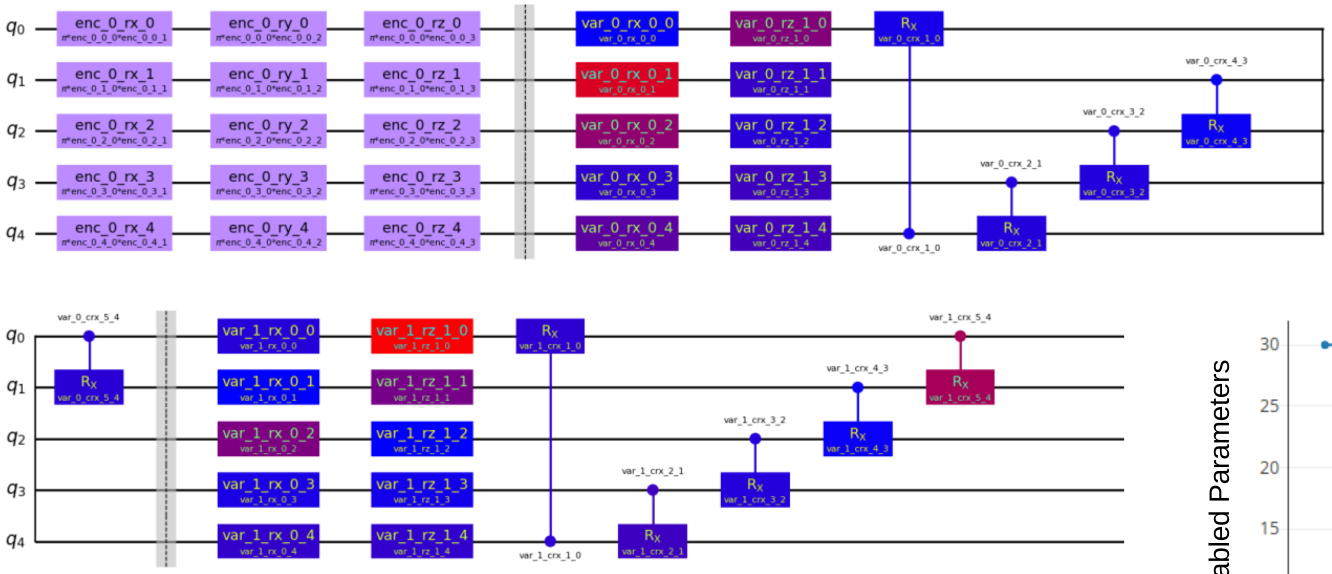
$$\frac{d}{d\theta} f(\theta) = r \left[f\left(\theta + \frac{\pi}{4r}\right) - f\left(\theta - \frac{\pi}{4r}\right) \right]$$



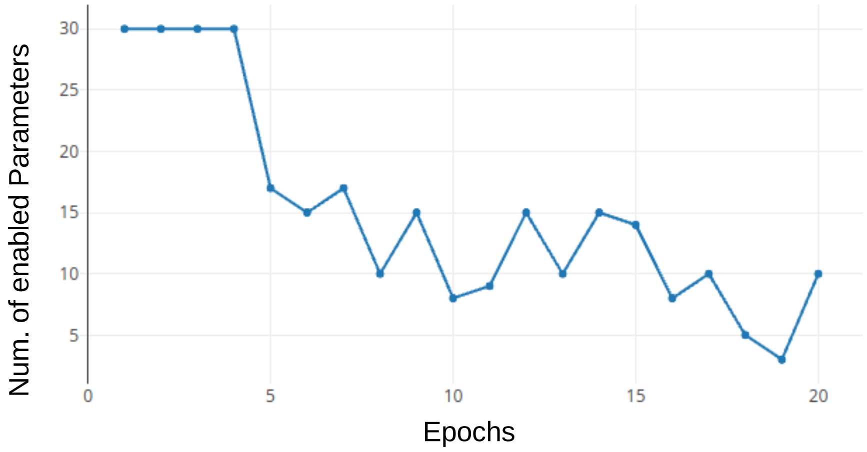


[4] Evaluating analytic gradients on quantum hardware - Schuld et al. - 2019

Parameter Pruning

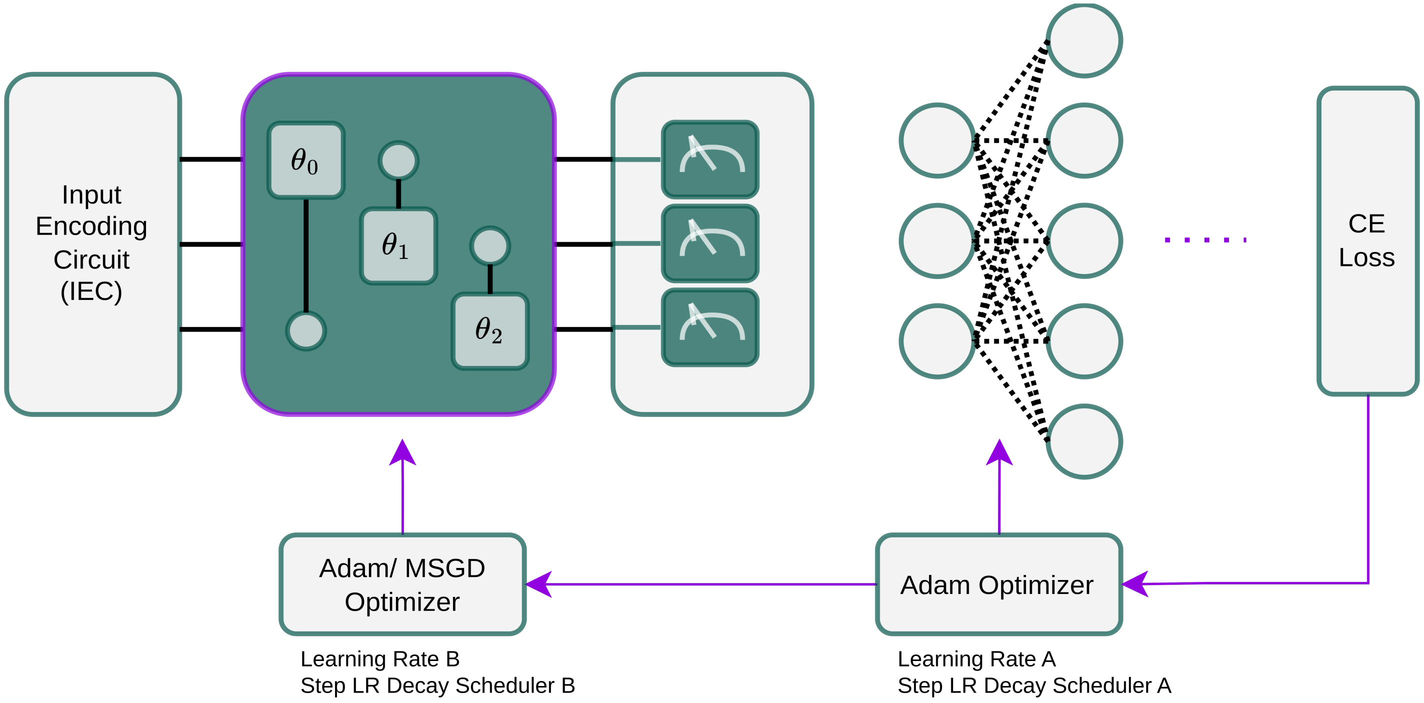


- Speeds up comp. time significantly (up to 20% less training time at almost equal performance)
- Investigating in more advanced methods for pruning and re-enabling parameters



- Change in gradients over last couple of steps
- Disable parameter update if 2nd order gradient falls below a threshold

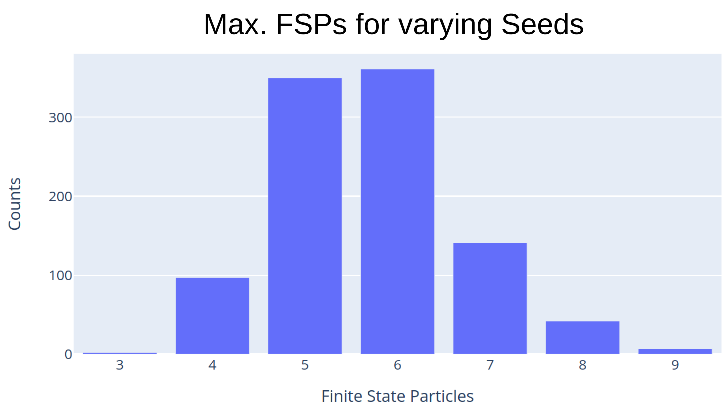
Split Optimizer

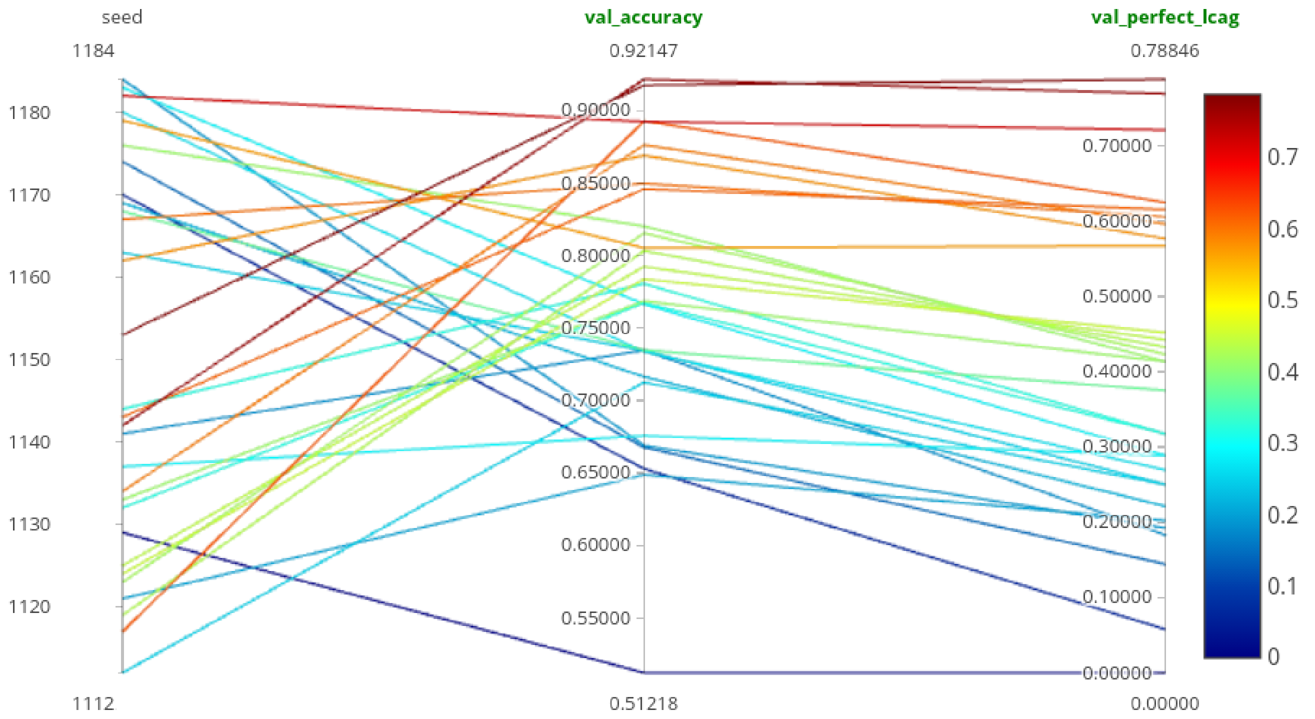


- Separate optimizer with
  - different learning rates
  - different learning rate decays
- Why beneficial?
  - q-parameters much more sensitive
  - smaller learning rate favorable

A word regarding the Performance Measurements

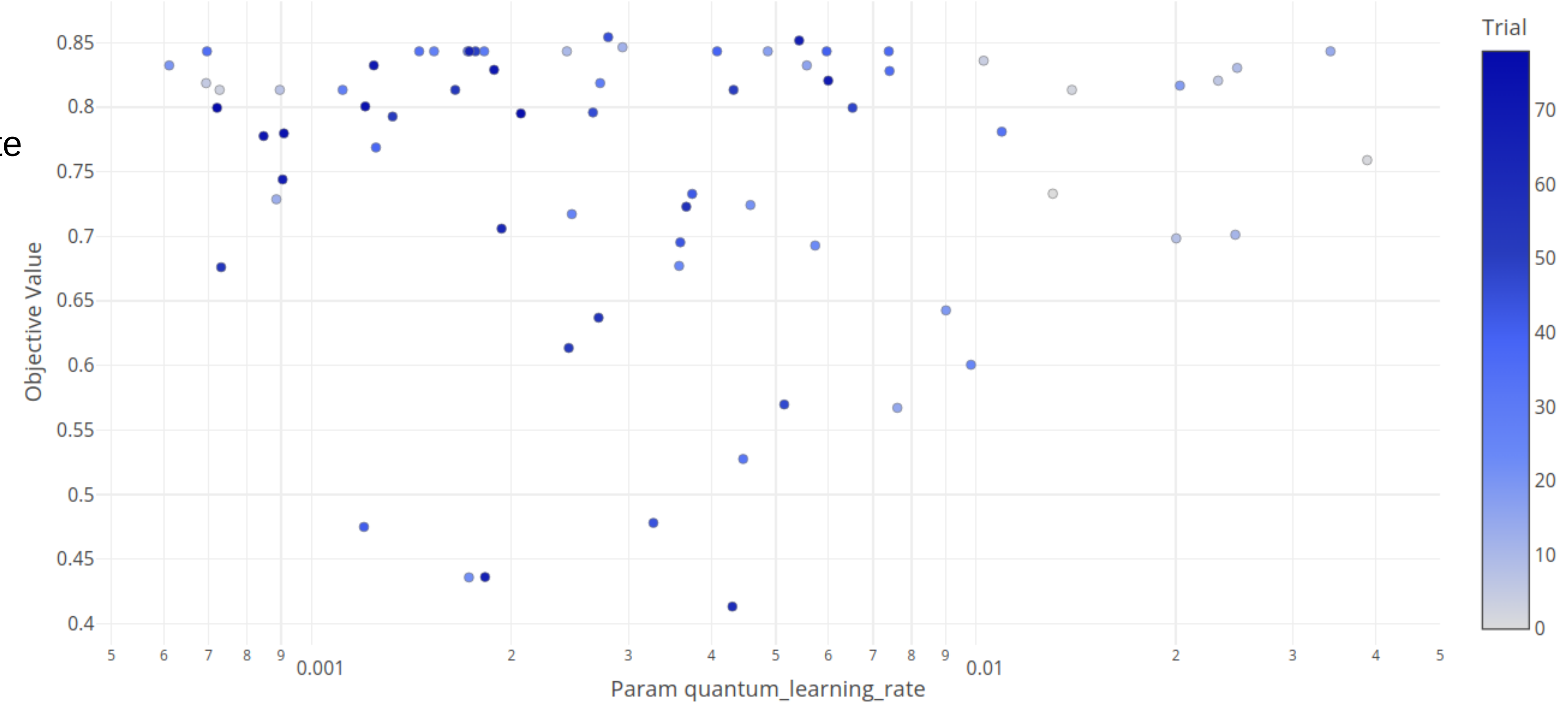
- Performance strongly depending on event data
- Data generation seeded, so we can evaluate and compare different scenarios
- Comparison plot generated using classical model





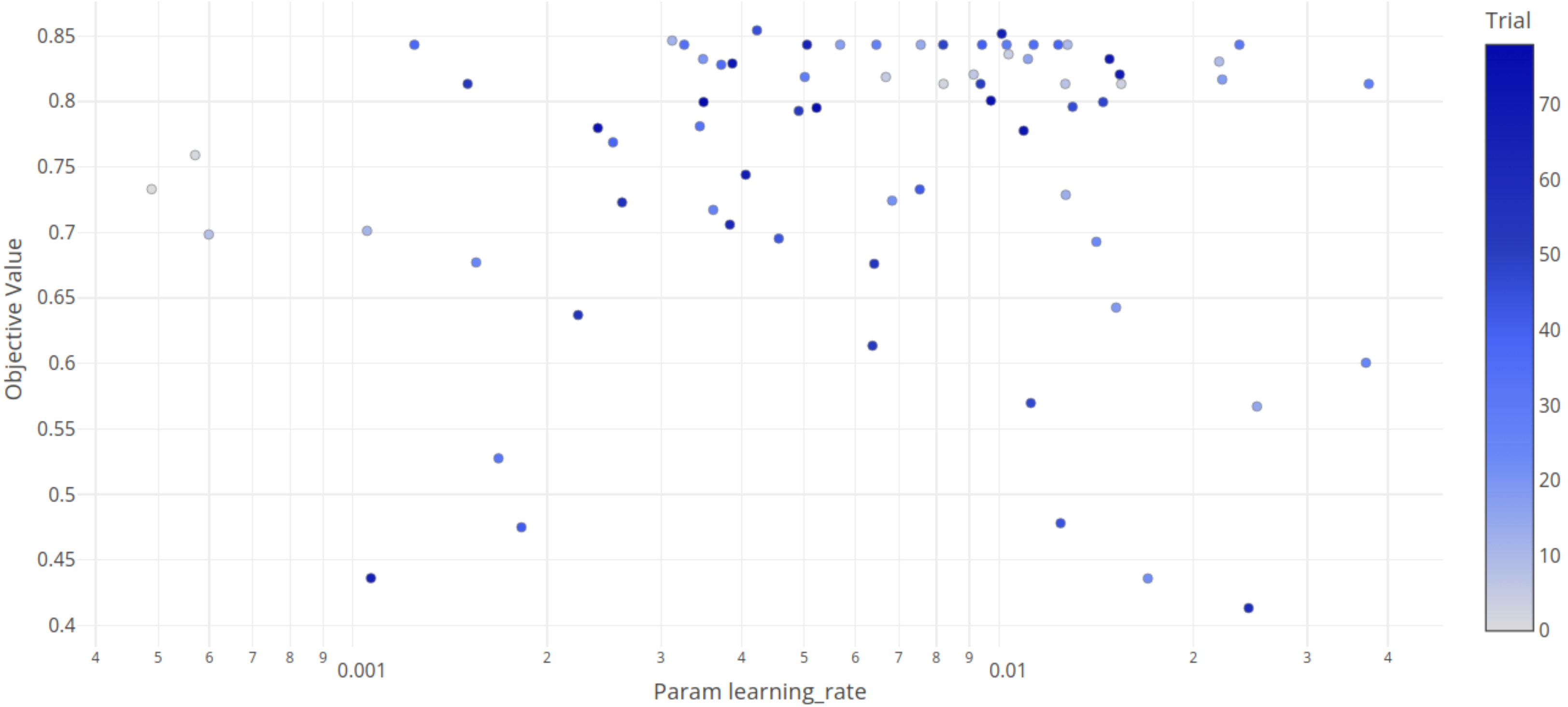
Split Optimizer – Varying Learning Rates

- Hyperparameter study
- Noise-free Simulation
- Quantum Learning Rate



Split Optimizer – Varying Learning Rates

- Hyperparameter study
- Noise-free Simulation
- Classical Learning Rate



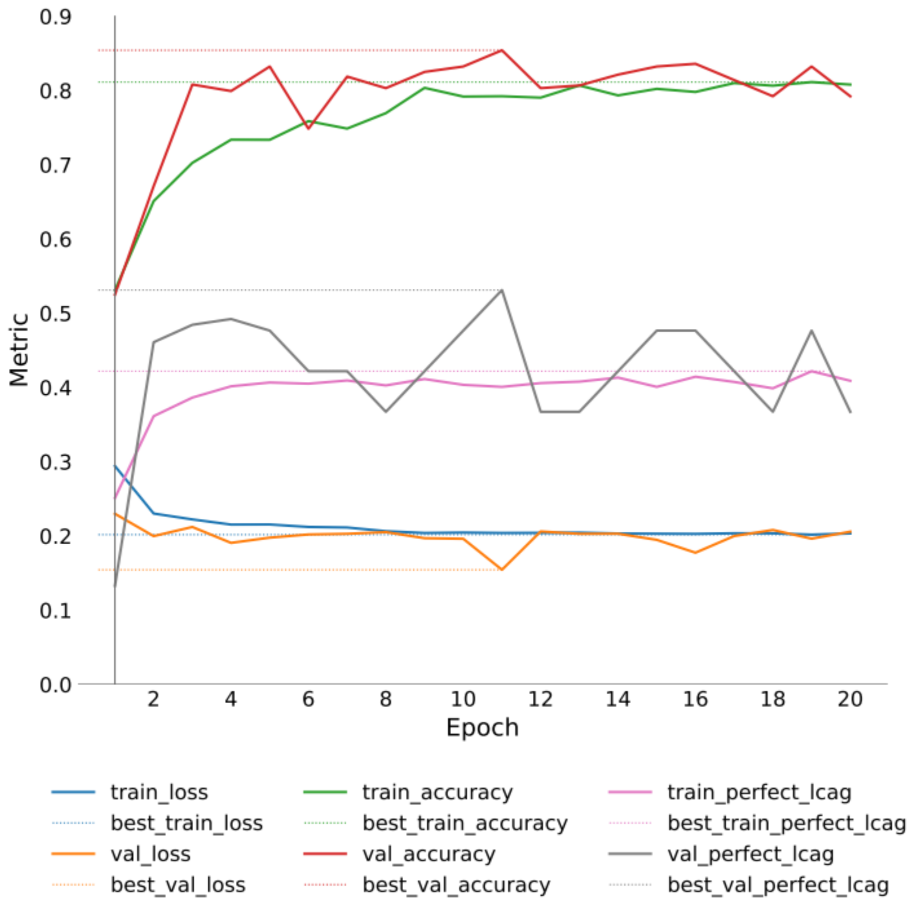
Results Noisy Simulation

- Noise-free Simulation:
  - Validation Accuracy: 0.857 (Step 11)
  - Validation Loss: 0.155 (Step 11)
- **Noisy Simulation**:
  - Device: IBM Perth
  - Validation Accuracy: 0.854 (Step 11)
  - Validation Loss: 0.154 (Step 11)

Simulation

Quantum Device

- Coupling Map
- Basis Gates
- Single-Qubit Gate Errors
- Two-Qubit Gate Errors
- Readout Errors



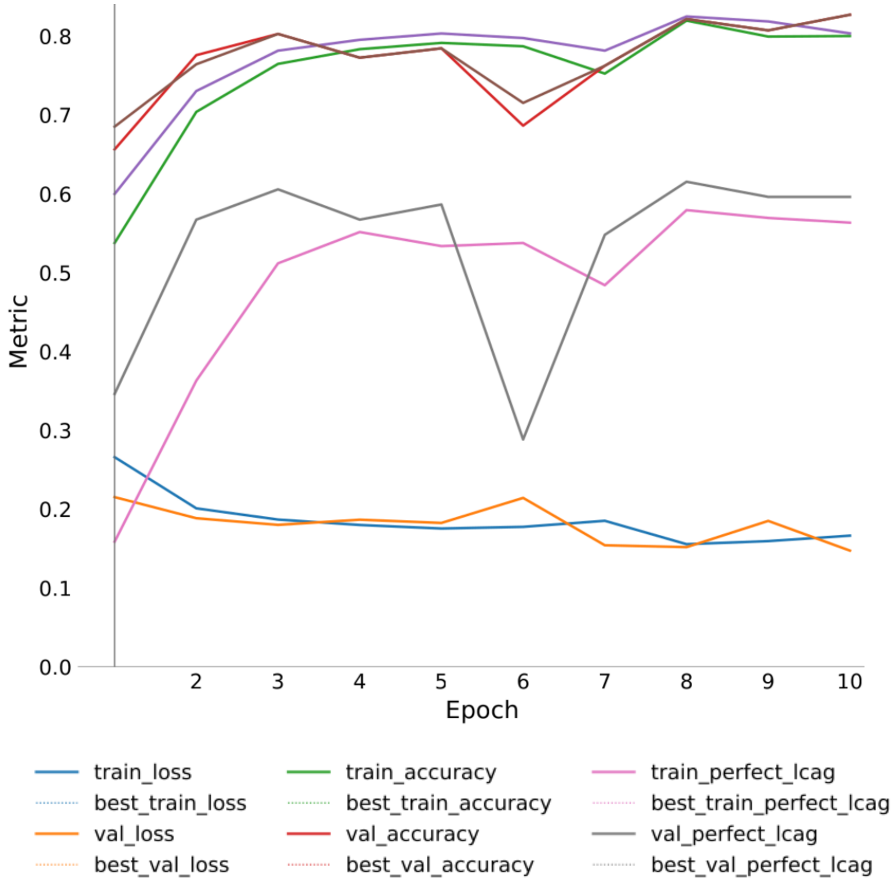
Karlsruhe Institute of Technology

Thanks for Your Attention!

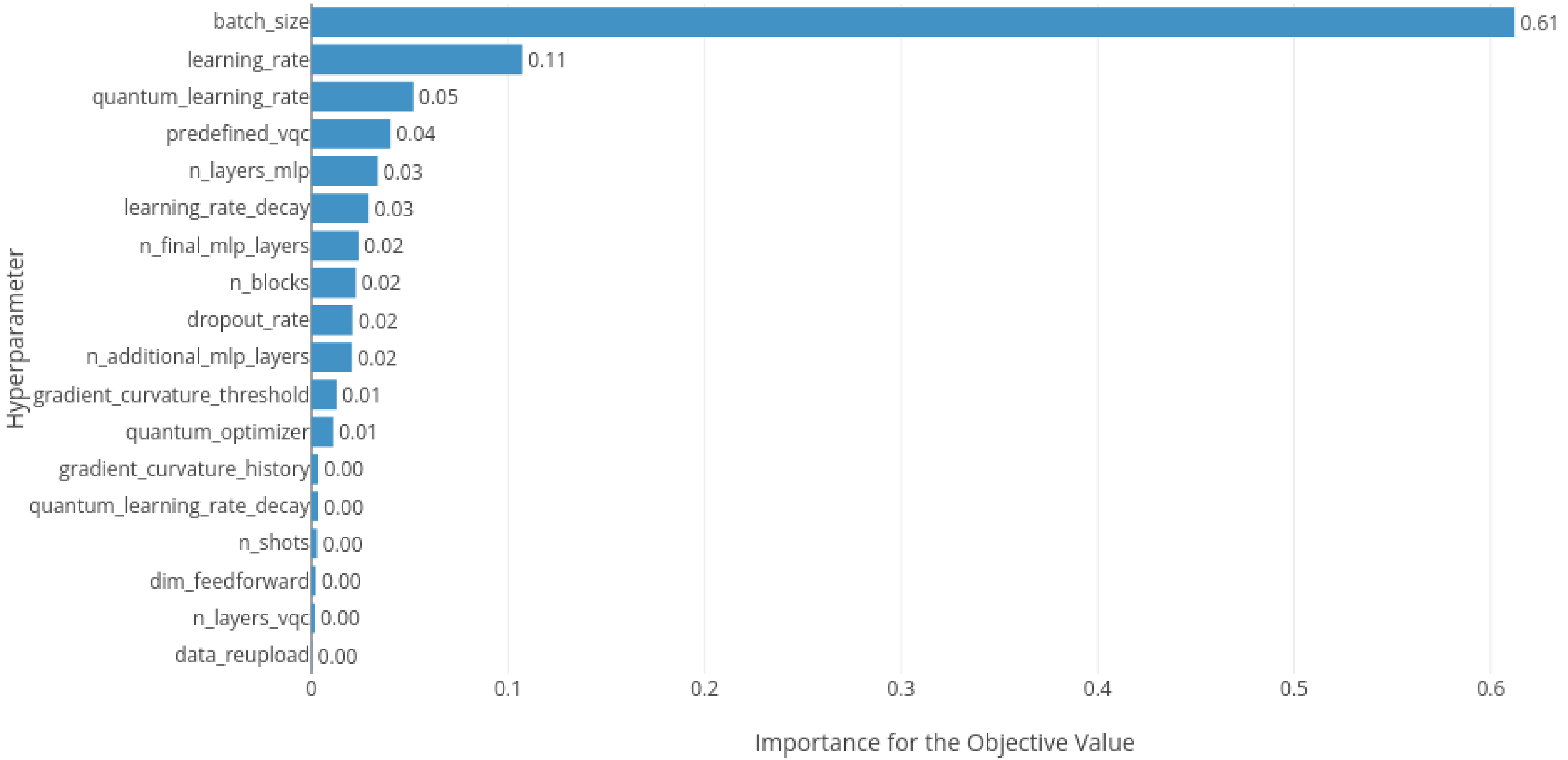
Melvin Strobl, Eileen Kühn, Max Fischer, Achim Streit

Performance @ACAT22

- Noise-free Simulation:
  - Validation Accuracy: 0.824
  - Validation Loss: 0.147
- Noisy Simulation:
  - Device: IBM Perth
  - Validation Accuracy: 0.655
  - Validation Loss: 0.235



Hyperparameter Importance Noise-free Simulation



EDF Noise-free Simulation