CITY OF LONDON

QUANTUM CIRCUIT BORN MACHINE & QUANTUM MARKOV CHAIN MONTE CARLO FOR FINANCIAL FORECASTING

QUANTUM BUDDIES

Dat Chi (Ryan) Le, The University of Sheffield
Abubakar (Sid) Iliyasu, Imperial College London
Gyanateet Dutta, University of Leeds

Introduction & Problem Statement

Complex Financial Time Series Forecasting

Goal: Learn the distribution of daily AAPL returns and produce calibrated next-day uncertainty forecasts on NISQ-feasible hardware..

- **Context:** AAPL OHLCV (2020–2022), modeled as log-returns; shows fat tails, volatility clustering and (after differencing) stationarity.
- **Data & features:** OHLCV → 51–83 engineered features (lags, realized vol, abs/sq returns, range/volume, spectral) → PCA → 4–5 qubits via angle/hybrid encoding.
- **Problem:** Point forecasts hide tail risk. We need probabilistic forecasts that capture regime shifts, and provide usable prediction intervals (VaR/ES).
- **Baseline gap:** Classical ARIMA/naive baselines struggle with fat-tailed distributions and calibration.

Research Questions

1. Can a both **QCBM** and **QMCMC** learn and generate the conditional distribution of next-day returns more faithfully than classical baselines (ARIMA, naïve, Gaussian)?

2. Does **QCBM** reproduce heavy tails and distributional shape more faithfully than classical baselines under NISQ depth/qubit limits?

3. Do **QMCMC (QD-HMC)** posterior samples yield better-calibrated prediction intervals compared to QCBM and classical methods?

4. What is the impact of (a) **qubit count / circuit depth**, (b) **feature window size & encoding**, and (c) **entanglement topology** on accuracy, calibration, and sampling efficiency?

5. Is the approach **NISQ-feasible** in terms of parameter count, runtime, and noise tolerance?

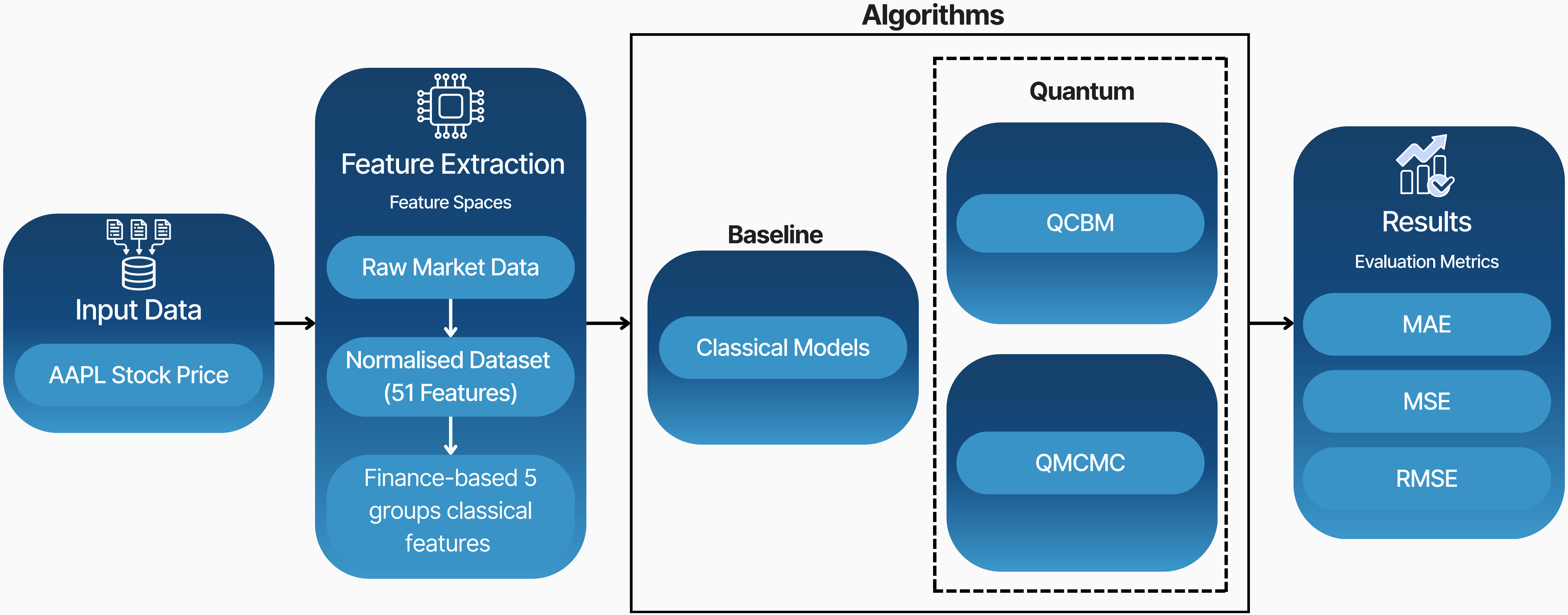
Constraints (NISQ reality)

- Shallow circuits ($L \leq 2$), 4–5 qubits, simple gates.
- QD-HMC precision ~2–3 qubits/parameter in gate-based sims.

Objectives & success criteria

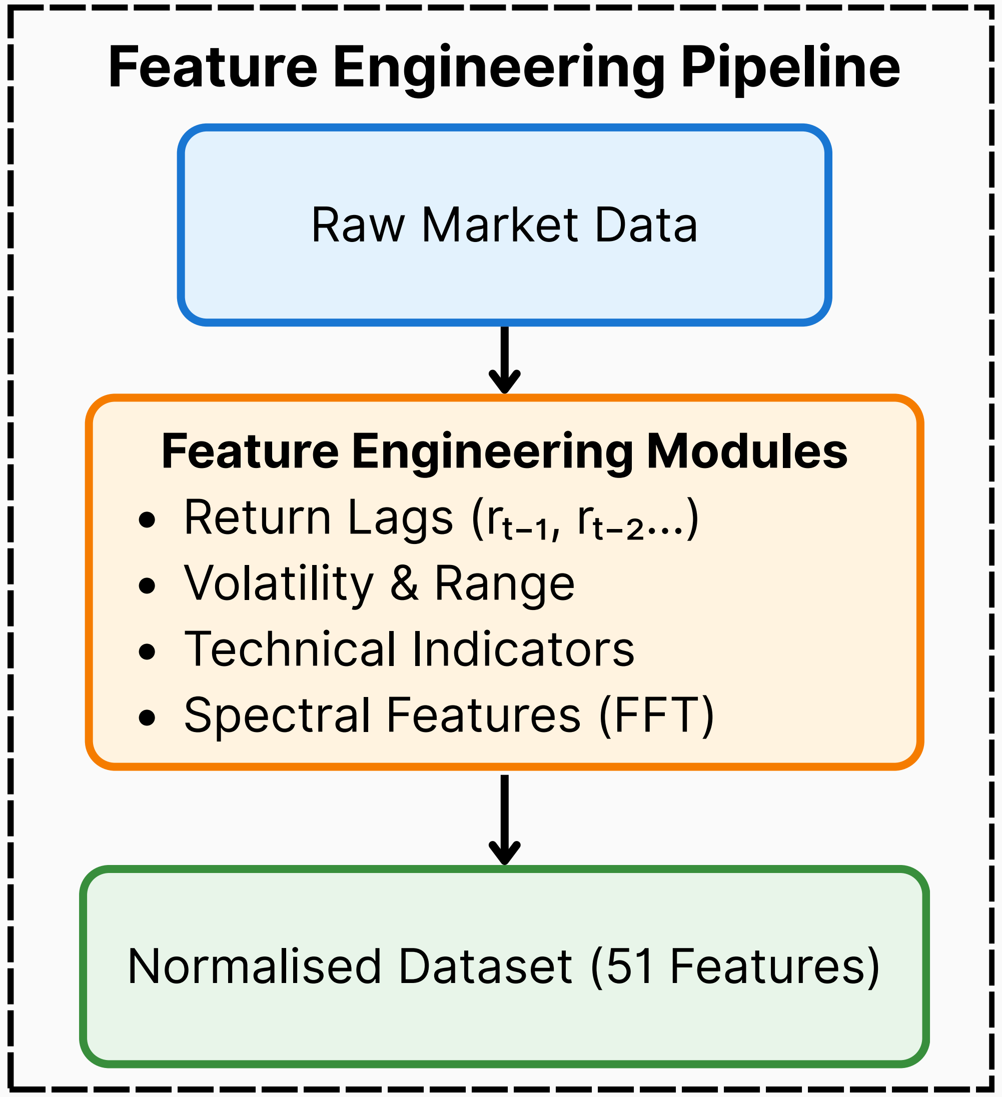
- Learn realistic return distributions; calibrated prediction intervals.
- Metrics: MAE+MSE+RMSE.
- Benchmark: best-config errors from results section.

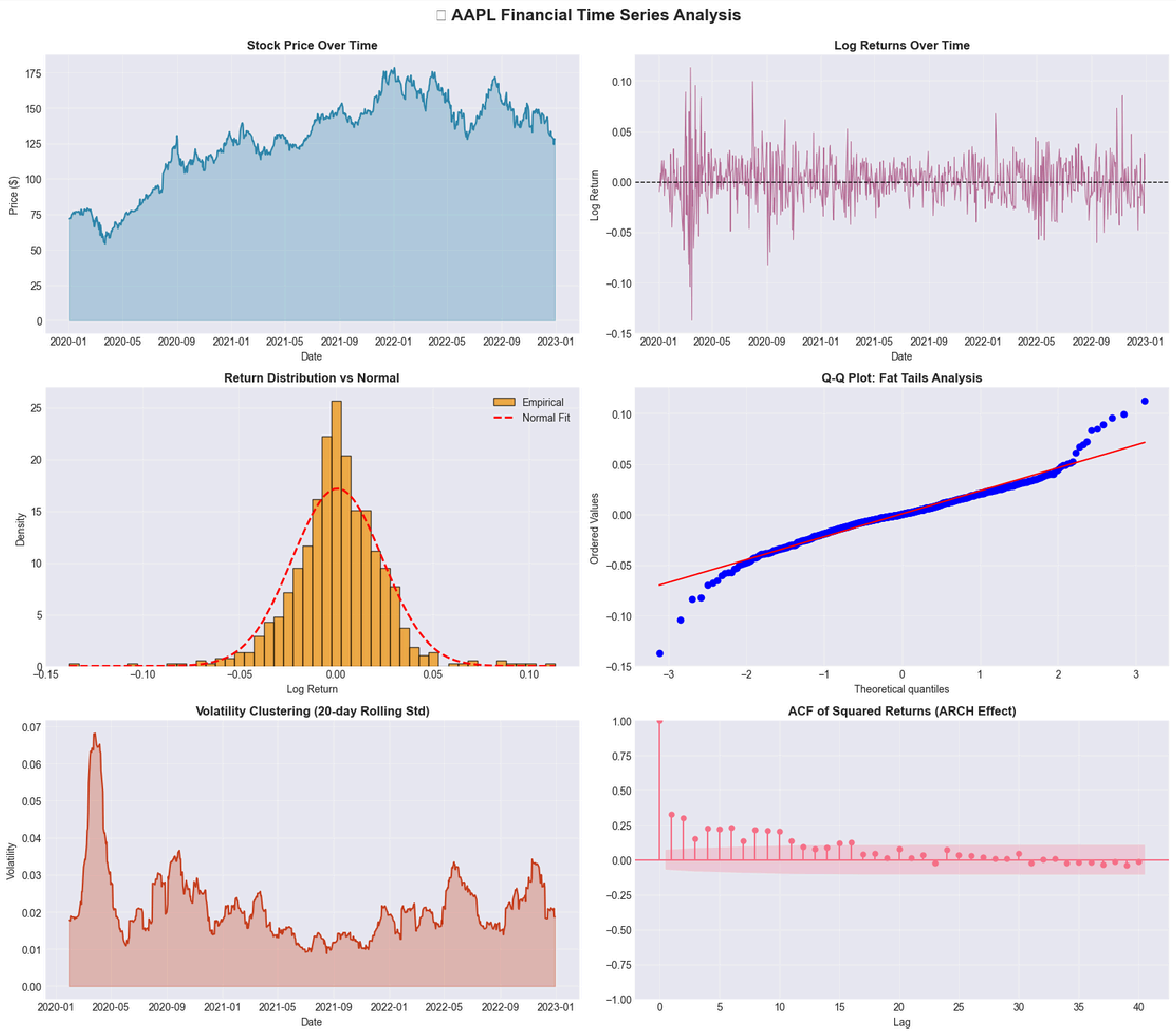
Complete Pipeline



Feature Extraction

Financial feature engineering transforms raw market data (Open, High, Low, Close, Volume) into predictive signals that reveal market dynamics. This pipeline expands 5 basic inputs into 51 engineered features across four groups—temporal lags, volatility measures, technical indicators, and spectral components—capturing key behaviors like momentum and volatility clustering.





Key Insights Summary

- Non-normal (fat-tailed) returns → Jarque-Bera test rejects normality
- Volatility clustering observed → rolling 20-day volatility & ARCH effects in ACF
- Stationarity confirmed → Augmented Dickey-Fuller test ($p < 0.01$)
- Stylised facts preserved → suitable for time-series modeling
- Comprehensive feature set → 51 engineered features
- Data quality validated → no missing or duplicate values, stable scaling

Main Baseline Model (ARIMA)

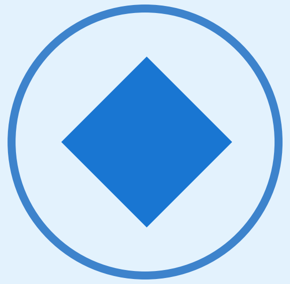
ARIMA (Autoregressive Integrated Moving Average) is a statistical model used for time series forecasting. It works by combining three components to predict future values based on past data:

AR (p)

AutoRegression

$$y_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \dots + \varphi_p y_{t-p} + \varepsilon_t$$

Uses past values to predict current



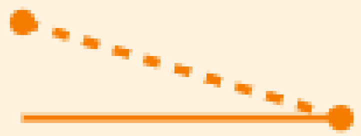
p = order of lags

I (d)

Integrated

$$\nabla y_t = y_t - y_{t-1}$$

Make data stationary by differencing



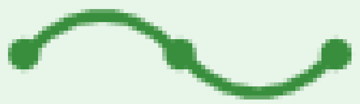
d = differencing order

MA (q)

Moving Average

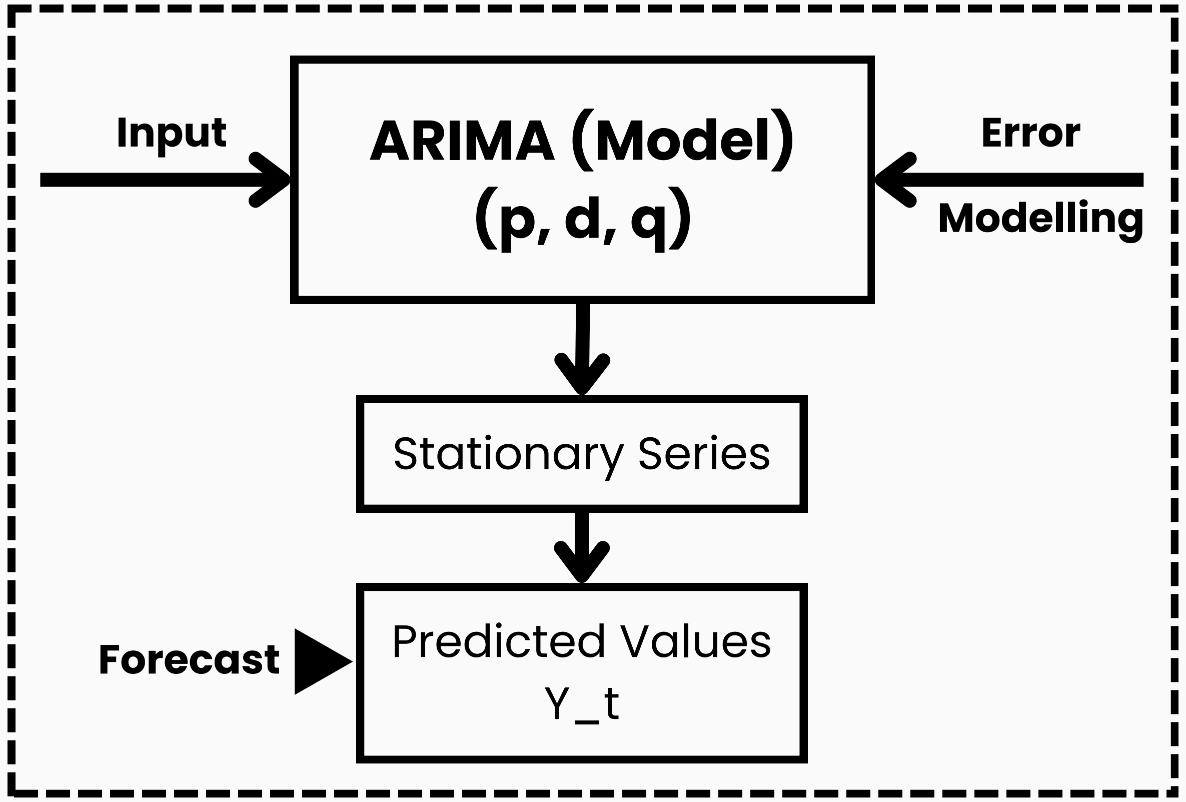
$$y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q}$$

Uses past forecast errors to make prediction



q = error terms

Model Workflow



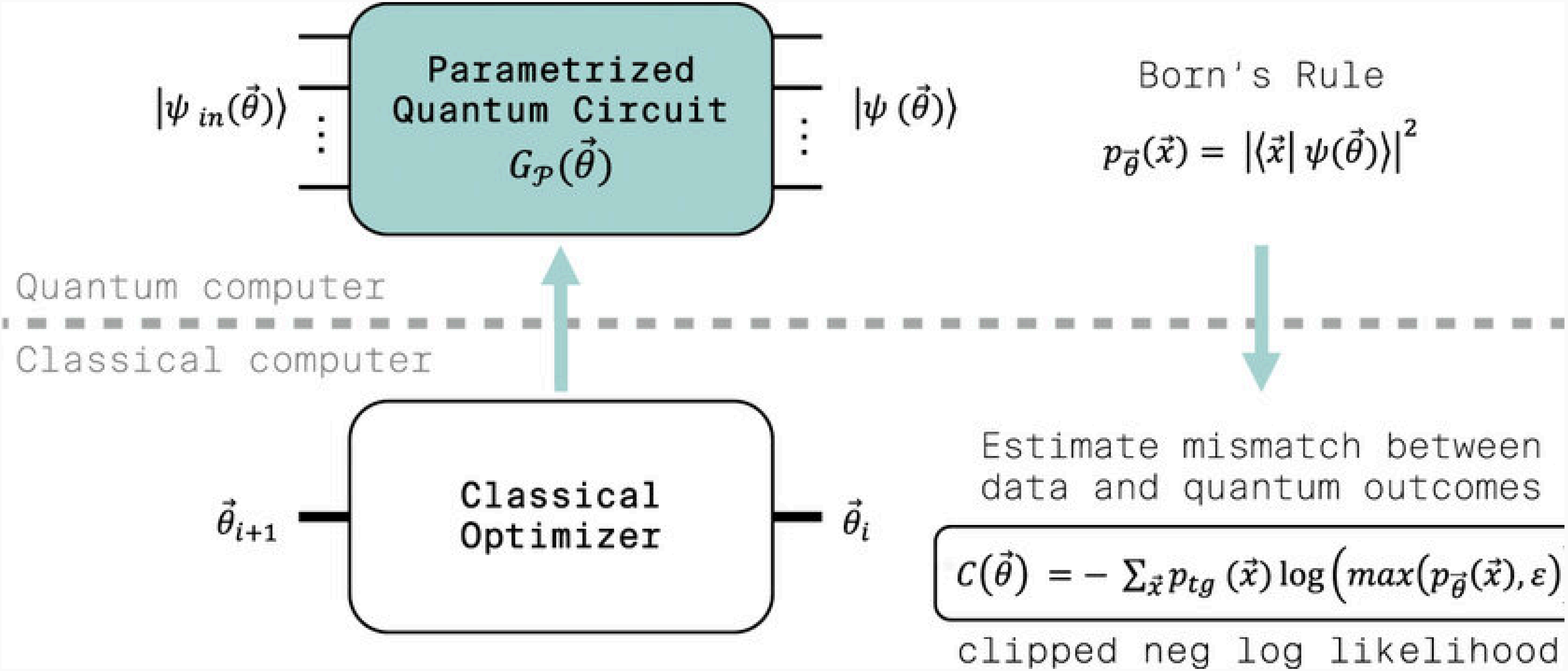
Key Applications:

- **Stock Price Prediction:** Captures market volatility and trends
- **Economic Indicators:** GDP, inflation, exchange rates forecasting
- **Risk Management:** Portfolio optimisation and volatility modeling
- **Trading Strategies:** Short-term and medium-term market predictions

Key Advantages:

- Handles non-stationary financial data effectively
- Captures both trend and noise in market data
- Provides prediction confidence intervals
- Well-established theoretical foundation

Quantum Circuit Born Machine (QCBM): Theory



QCBM: Quantum Features Encoding Pipeline

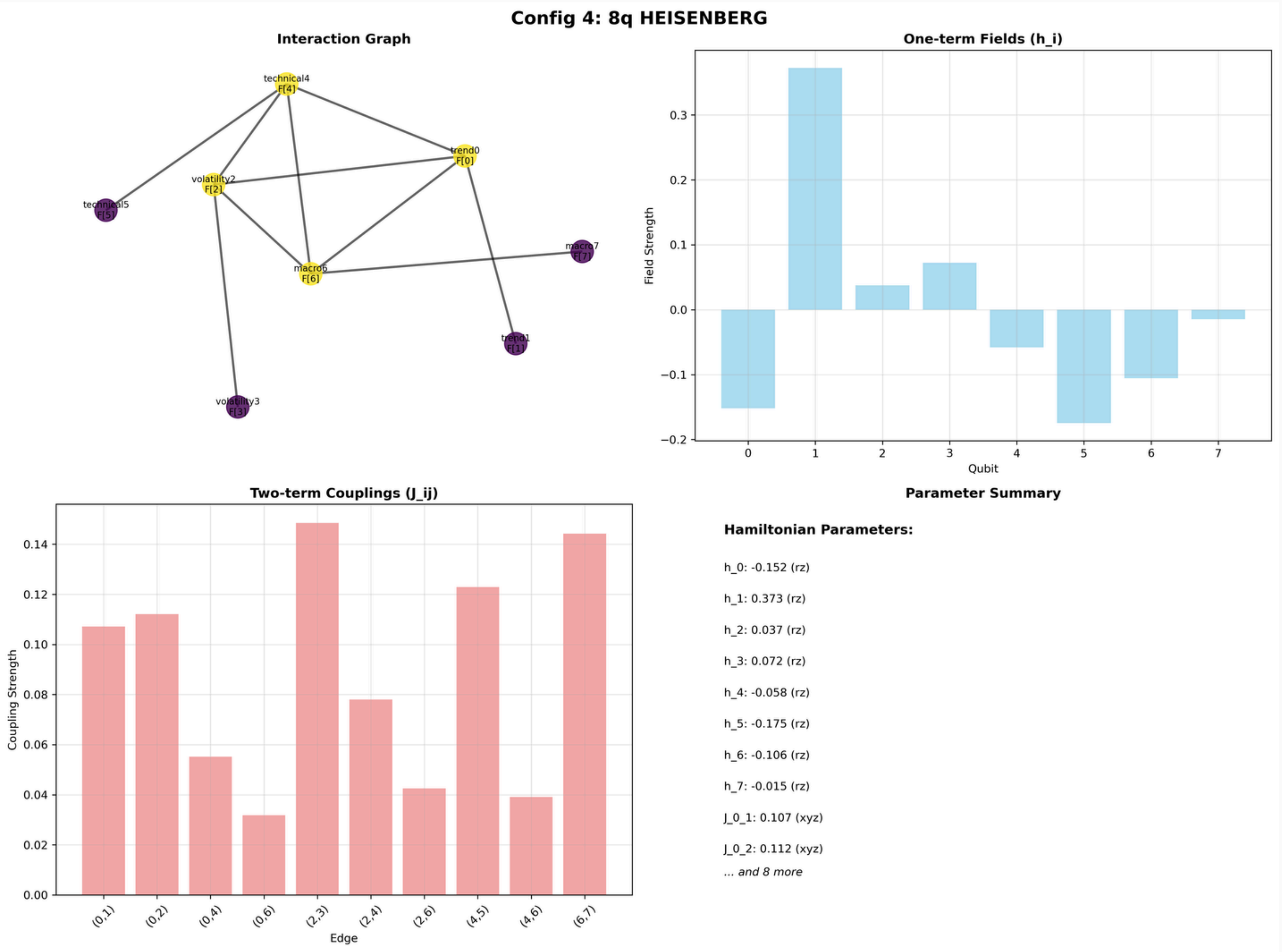
We have tested with multiple QCBM architectures for the quantum feature encoding and trainable parametrised quantum circuit (PQC), including (but not limited to):

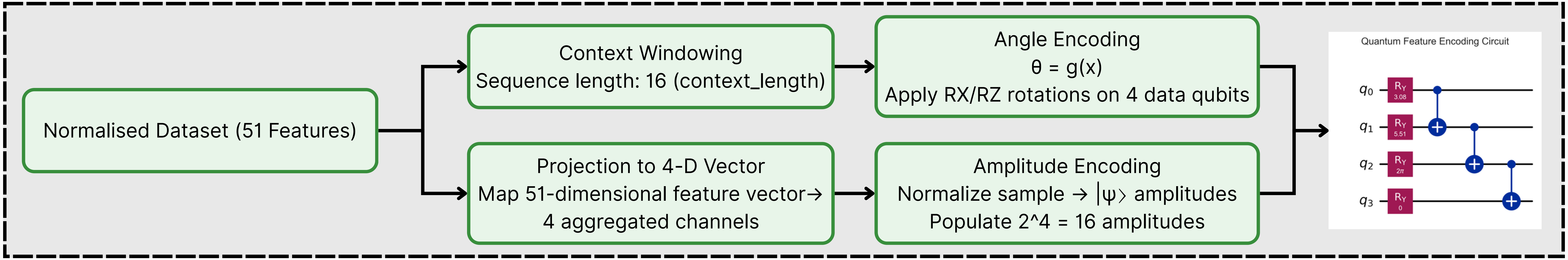
Approach 1: Hardware-Efficient Ansatz

- Encoding & PQC: encoding select classical features into amplitudes (Ry gates) and phases (Rz gates), entangling layers with CNOTs, Hadamaard gates for noise, etc.

Approach 2: Physics-Inspired Ansatz

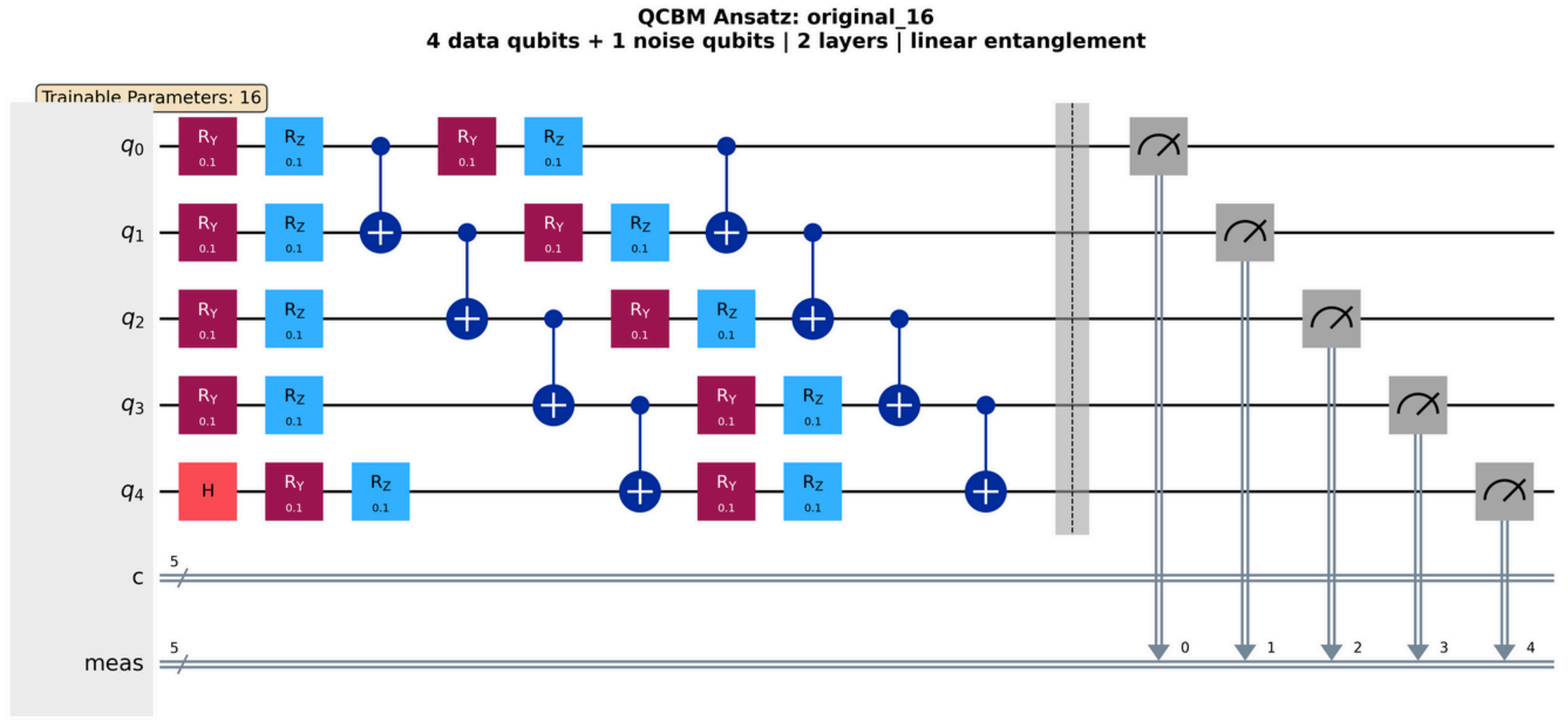
- Encoding: classical features to a natural system hamiltonian (e.g. a spin hamiltonian).
- PQC: Decomposing the hamiltonian into a trainable parametrised quantum circuit for QCBM. The figure on the right shows a way how the classical features were mapped into fields and couplings of a spin system.

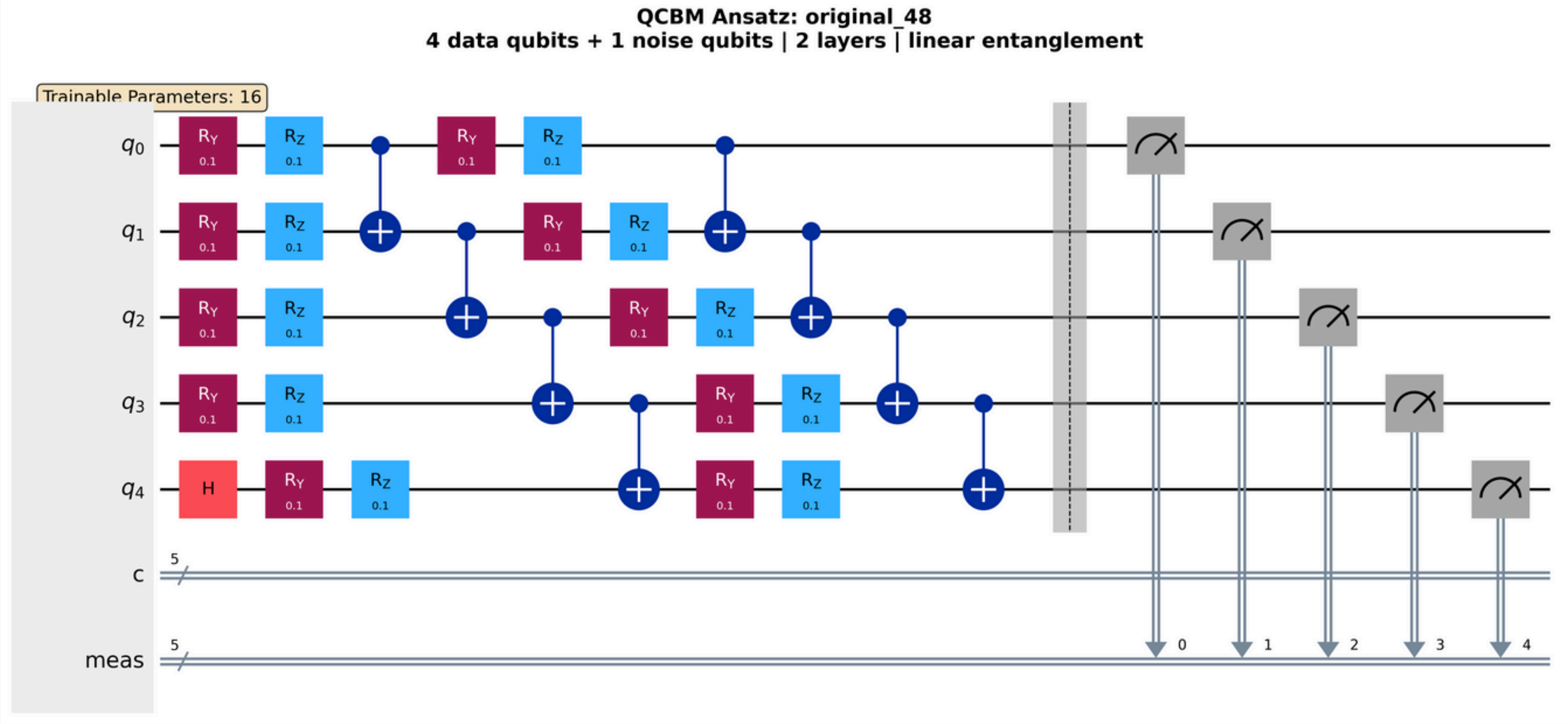


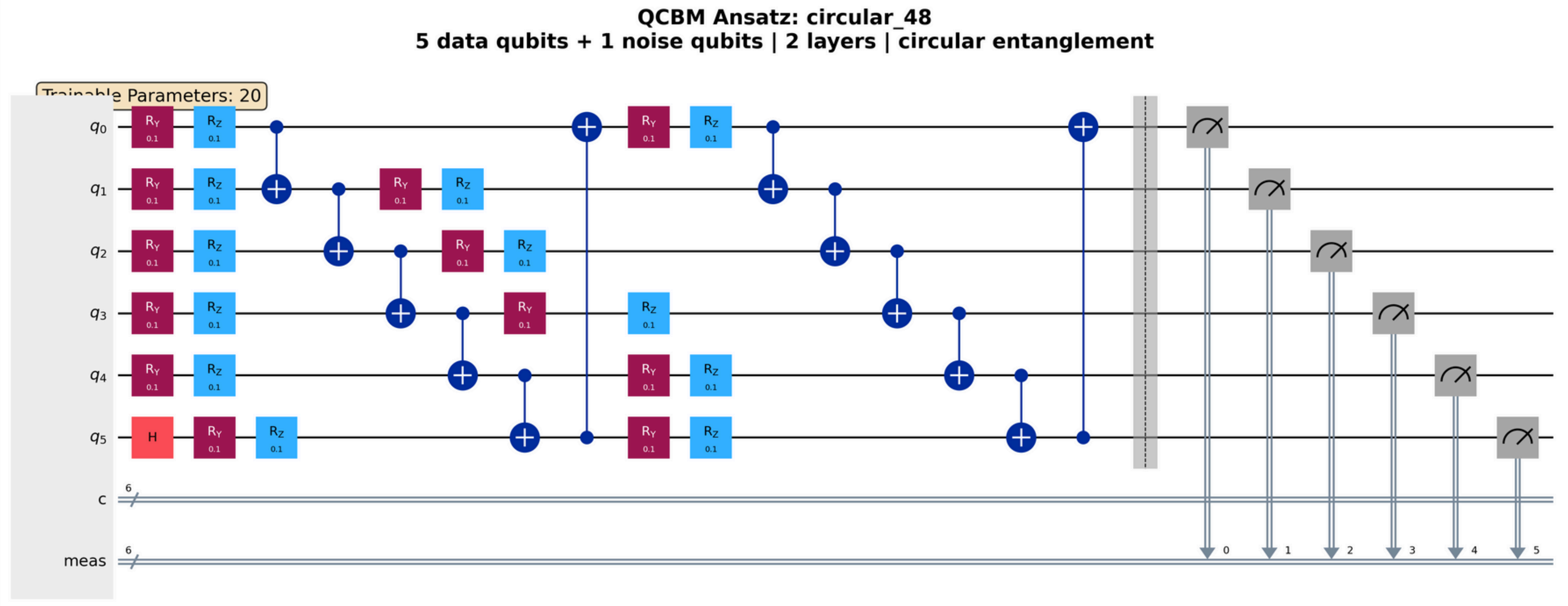


QCBM: Hybrid Encoding + HEA Circuits

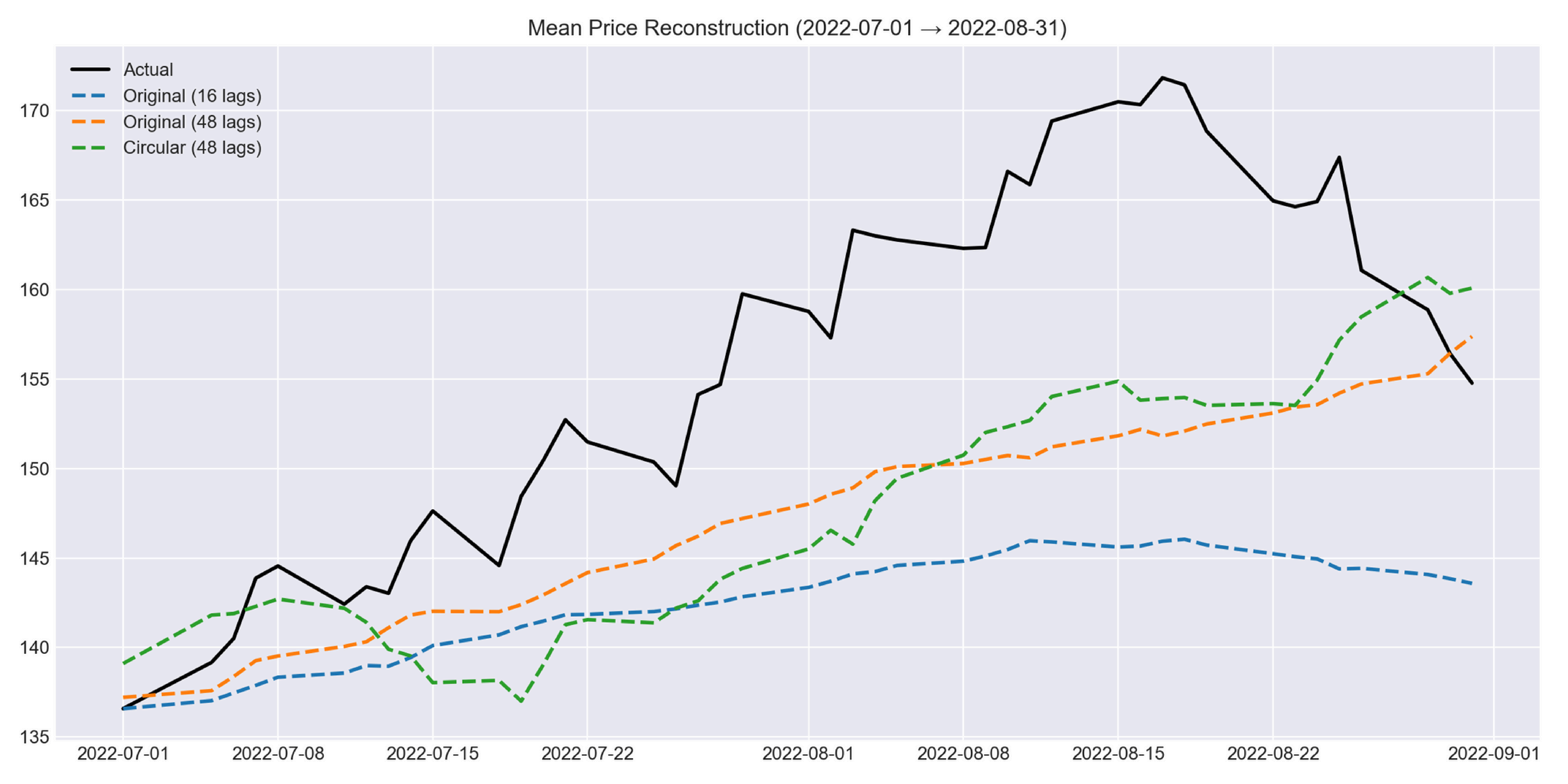
These figures are examples of some of the **Hybrid Encoding + HEA Ansatzs (with configurable entanglement)** we tested for our QCBM architecture.



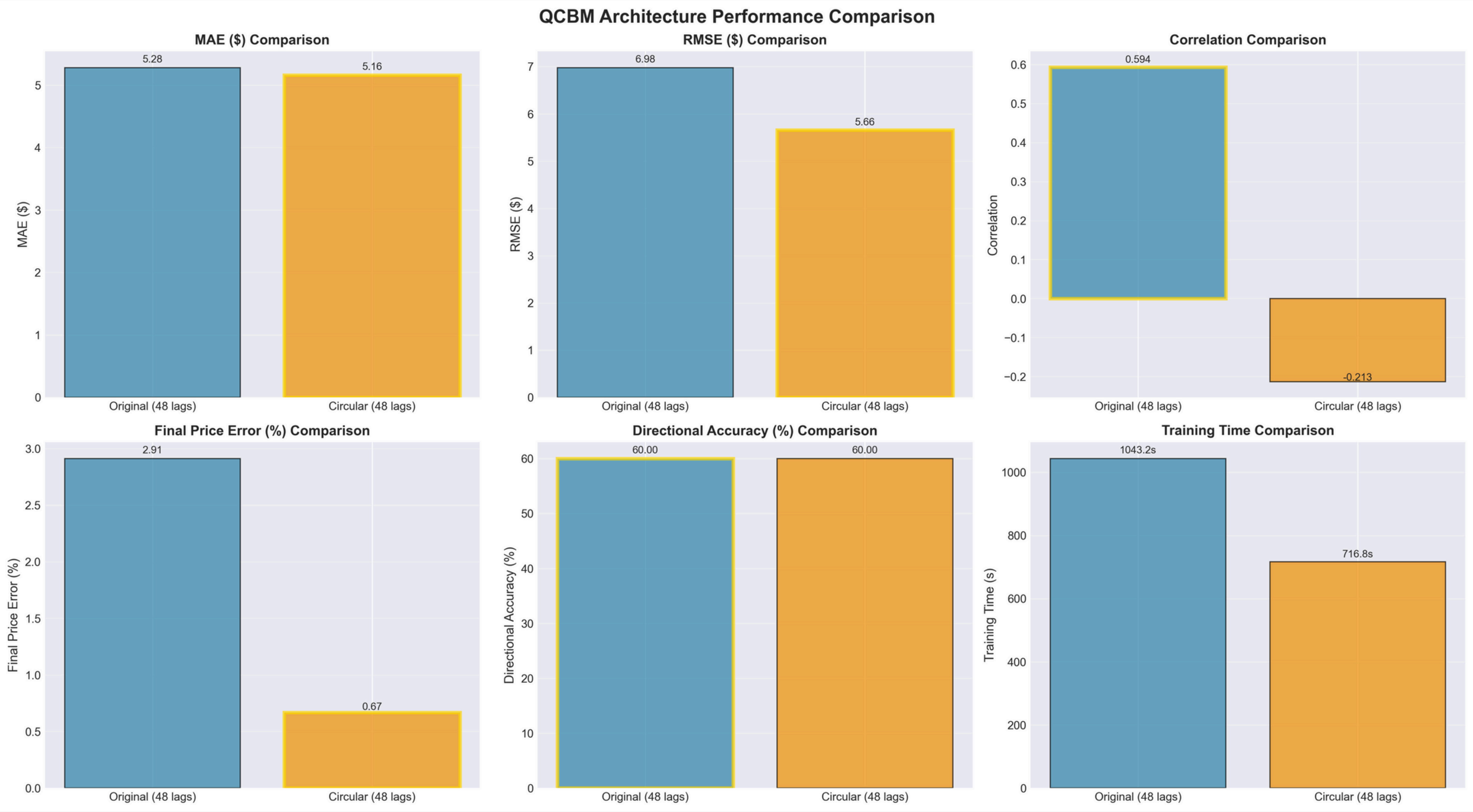




QCBM: Hybrid Encoding + HEA Results

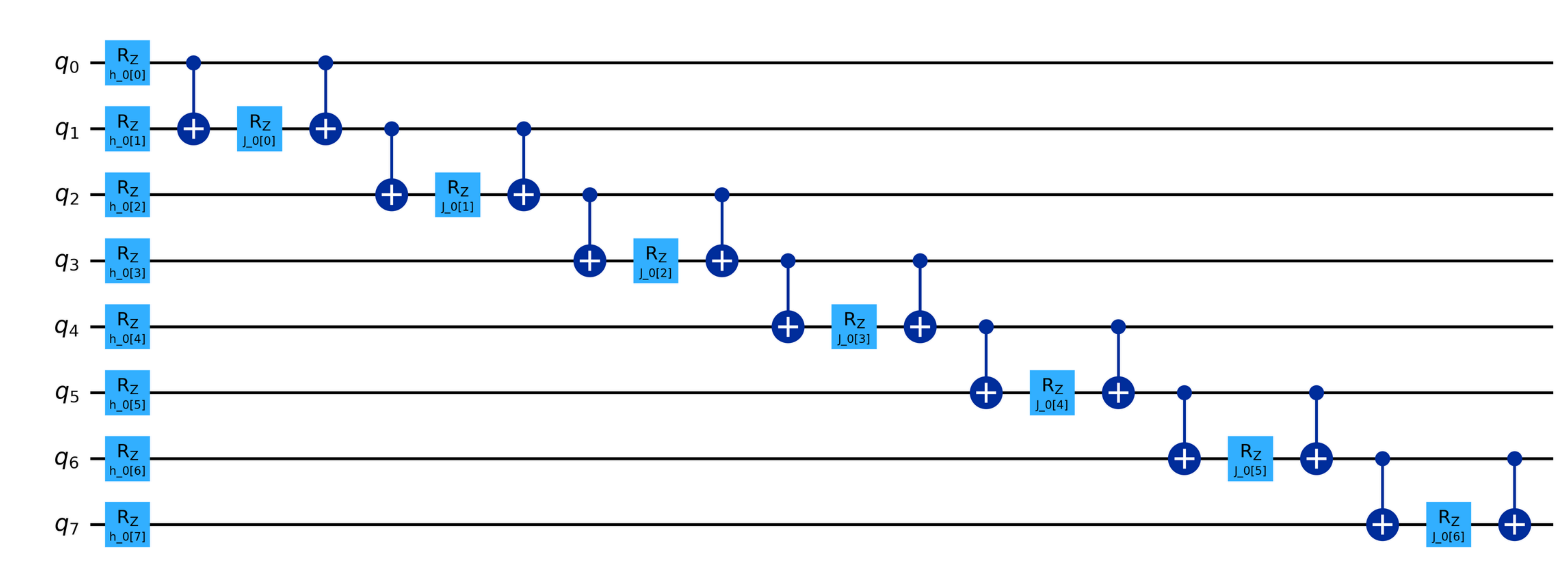


QCBM: Hybrid Encoding + HEA Results



QCBM: Physics-Inspired Circuits

These figures are examples of some of the **Physics-Inspired Ansatzs (with configurable entanglement)** we tested for our QCBM architecture.



QCBM: Physics-Inspired Results

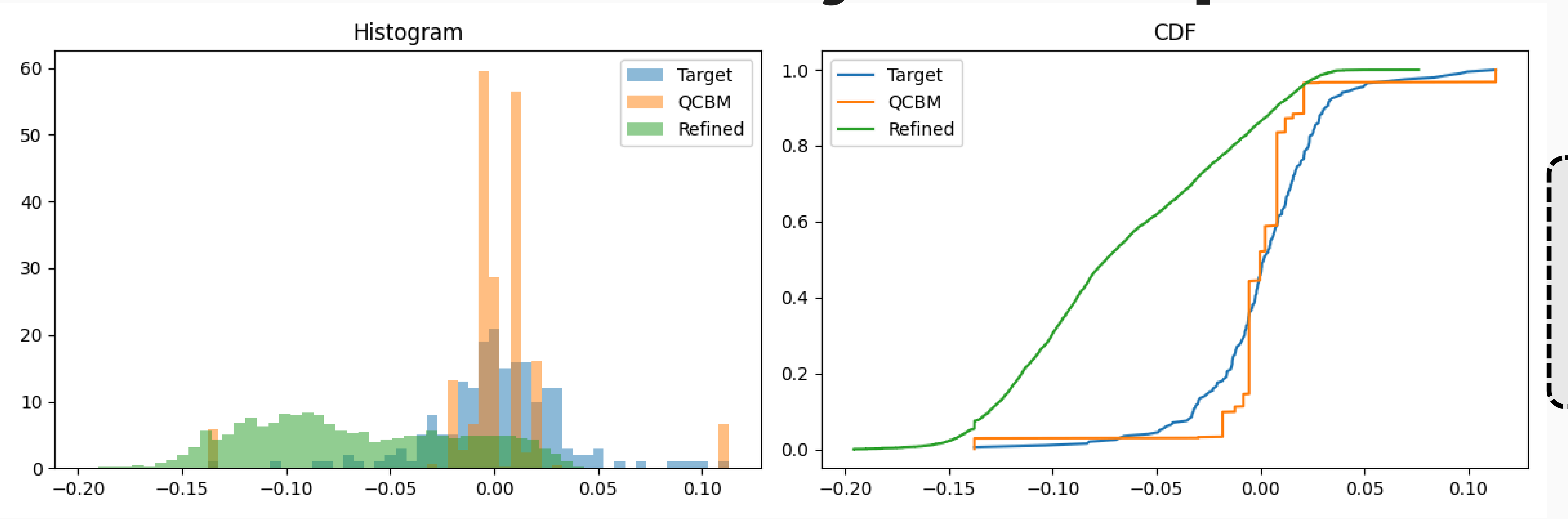


Figure 1
Distribution sampling results for a QCBM using a 4 qubit Ising Hamiltonian PQC

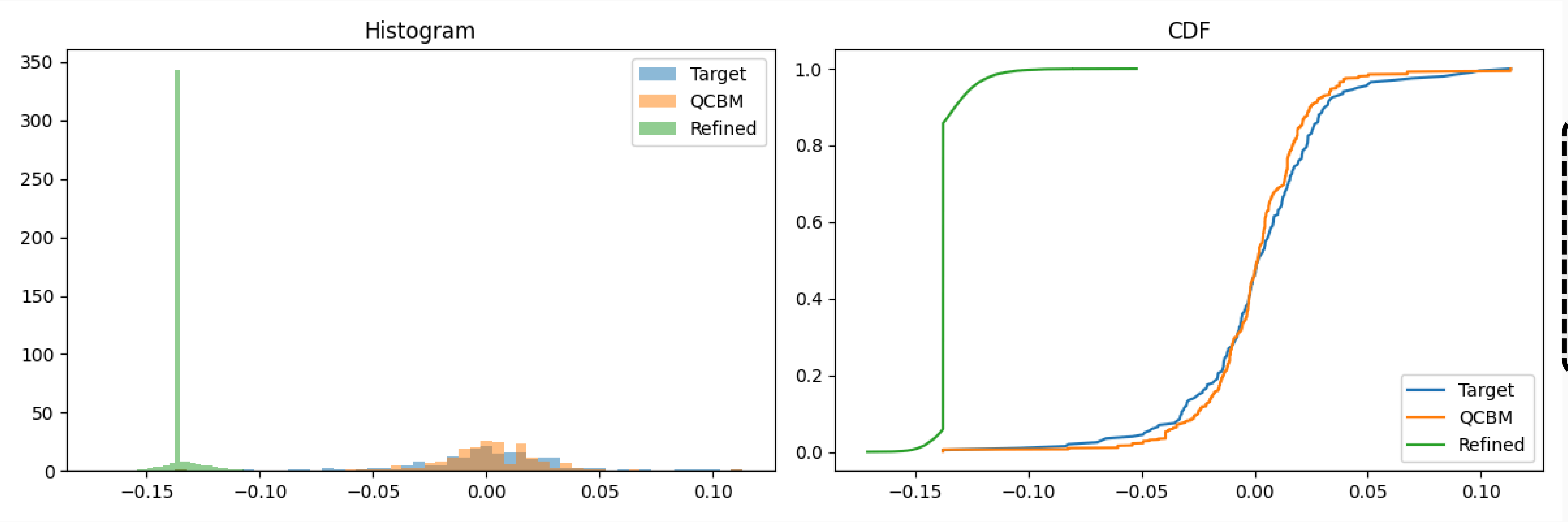


Figure 2
Distribution sampling results for the 8 qubit Ising Hamiltonian PQC

Note: the “refined” (blue line) represents an experimental combination of QMCMC and QCBM.

Quantum Markov Chain Monte Carlo (QMCMC)

Quantum Dynamical Hamiltonian Monte Carlo (QD-HMC) is a hybrid quantum-classical Markov Chain Monte Carlo (MCMC) sampling algorithm that uses quantum Hamiltonian dynamics to perform Bayesian inference.

Key Innovation: Replace classical symplectic integration in Hamiltonian Monte Carlo with quantum coherent evolution on quantum hardware

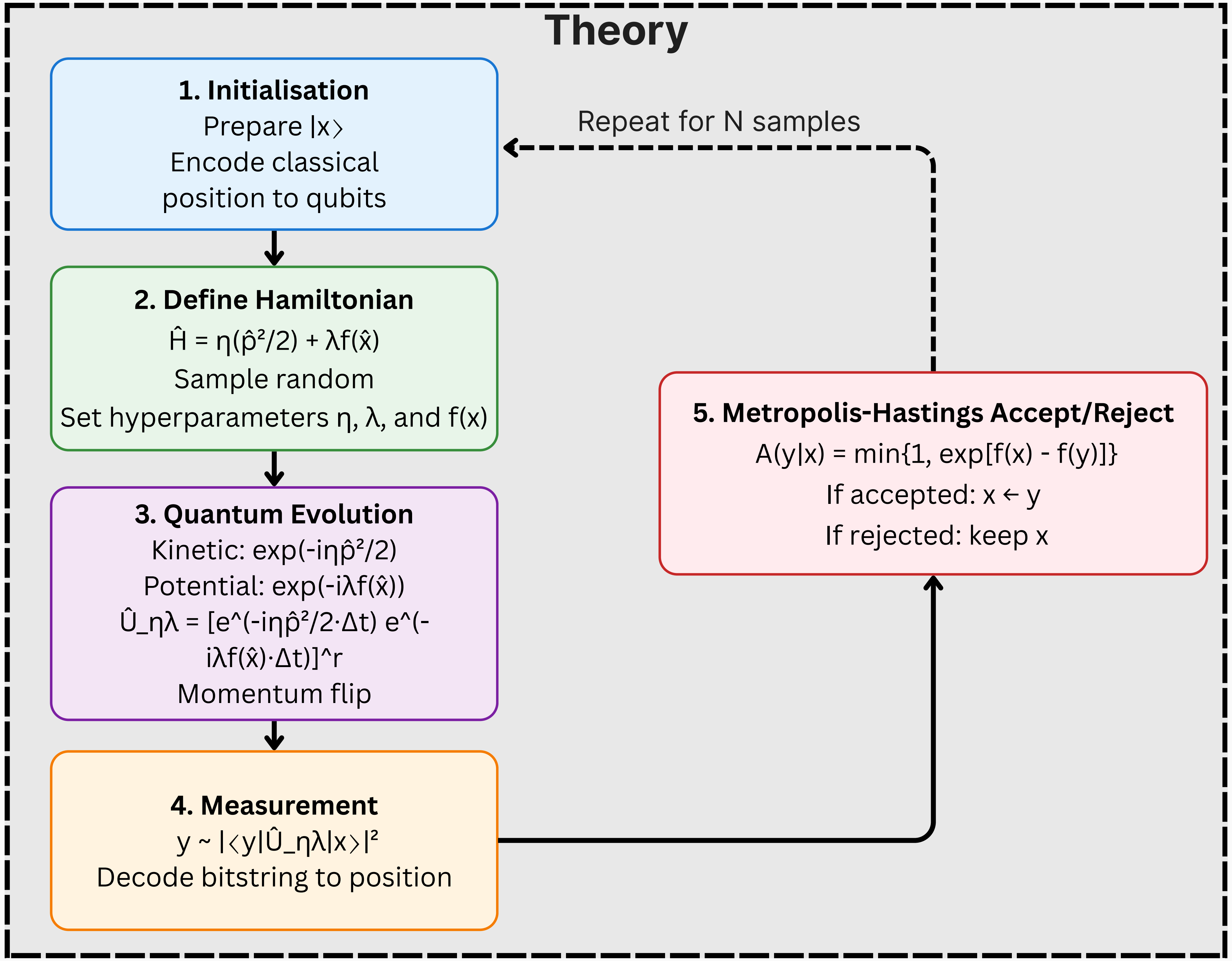
Workflow

Load Data: AAPL stock prices
Create Model: Bayesian linear regression with gradients
Initialise QD-HMC: Set precision, evolution time, Trotter steps
Run Sampling: Collect posterior samples with MH acceptance
Make Predictions: Use posterior samples for forecasting
Evaluate: Compute evaluation metrics
Visualise: Plot results

Key Advantage

Polynomial Speedup: Cubic to quartic speedup over classical MCMC in convergence (fewer iterations needed)
Better Exploration: Quantum superposition enables efficient exploration of complex posterior landscapes
Uncertainty Quantification: Natural framework for prediction intervals and confidence bounds
Handles High Dimensions: Effective for multivariate financial models with many parameters

Theory



QMCMC Circuits

Circuit Implementation Notes:

Quantum Operators:

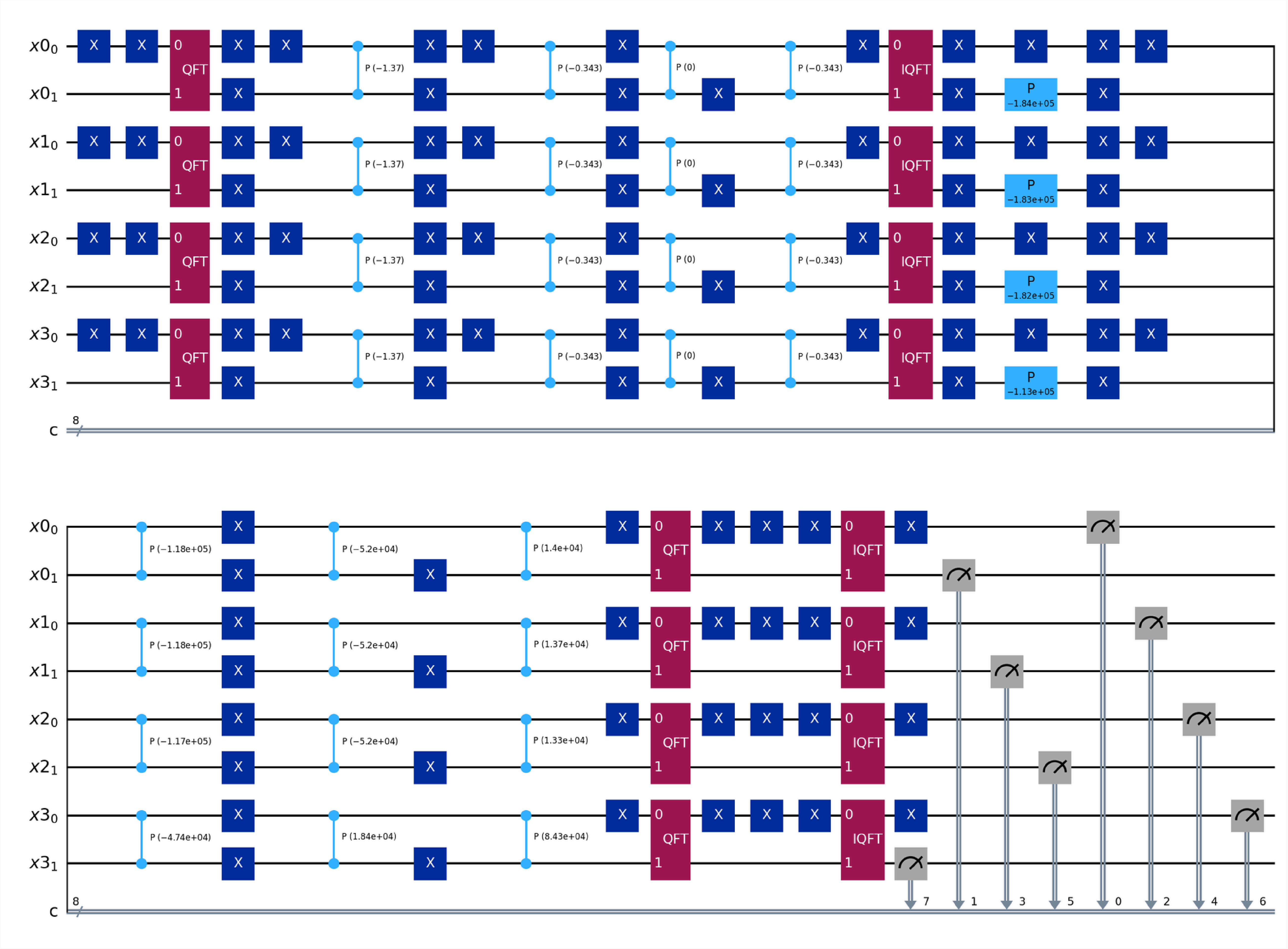
- Position eigenvalues: x_k = √(2π/N) · (k - N/2)
- Momentum via Fourier transform: p̂ = F†_c x̂ F_c
- State space: N = 2^d for d qubits

Computational Complexity:

- Total qubits: n_params × precision
- State space: 2^(n_params × precision)
- Circuit depth: O(r × 2^precision) gates

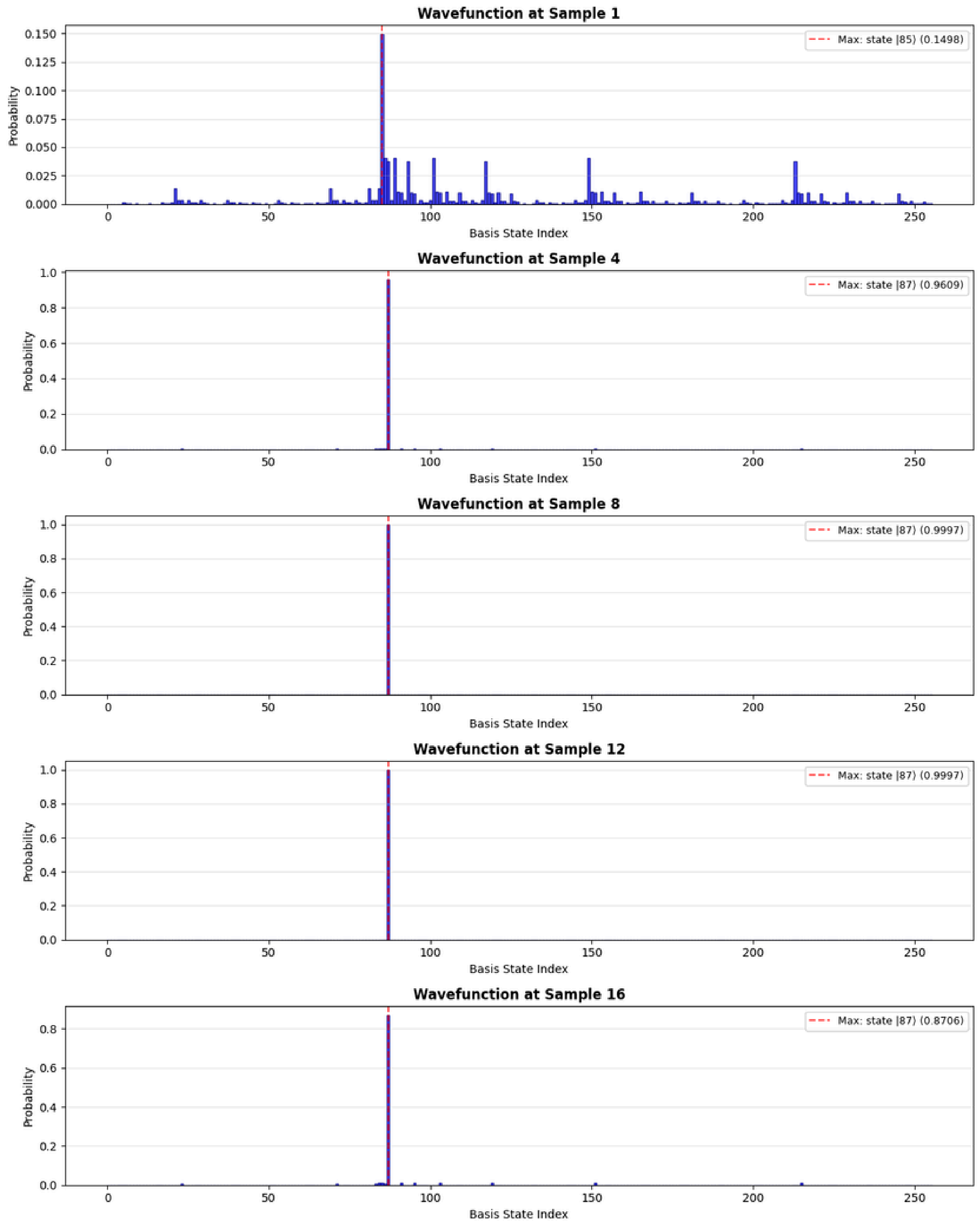
Practical Configuration:

- Precision: 2-3 qubits per parameter (for simulation)
- Evolution time: t = 0.5 - 1.0
- Trotter steps: r = 1-5
- Acceptance rate: 25-40% typical for gate-based implementation

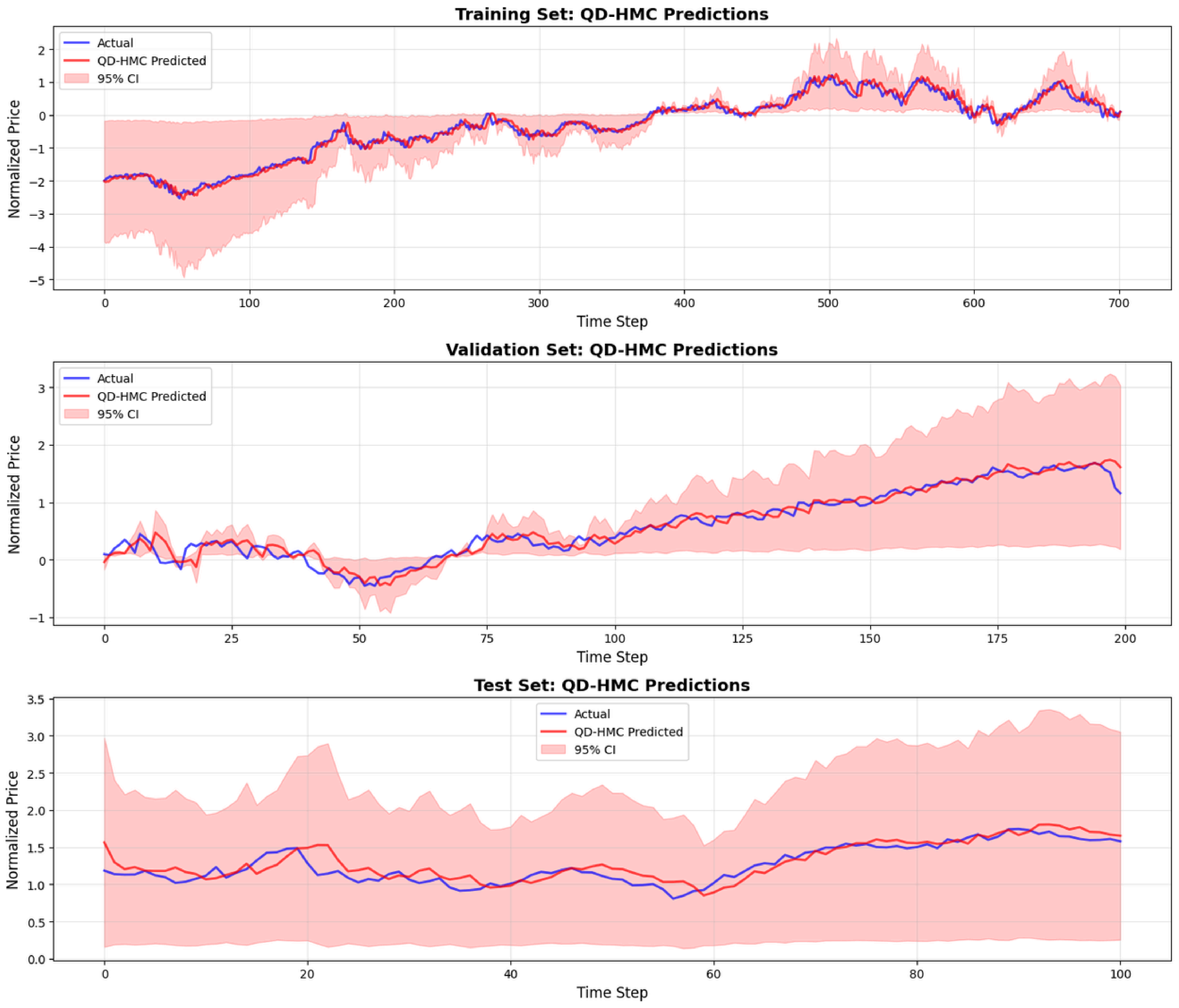


QMCMC - Results

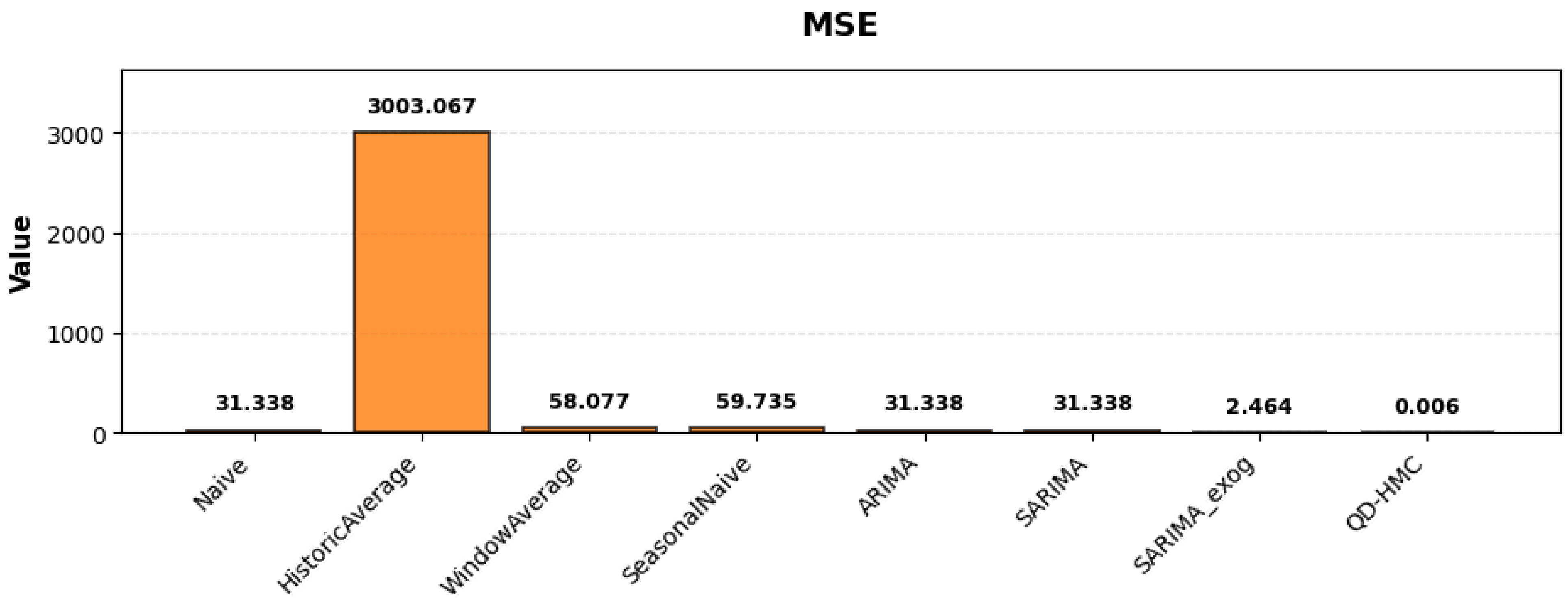
Wavefunctions

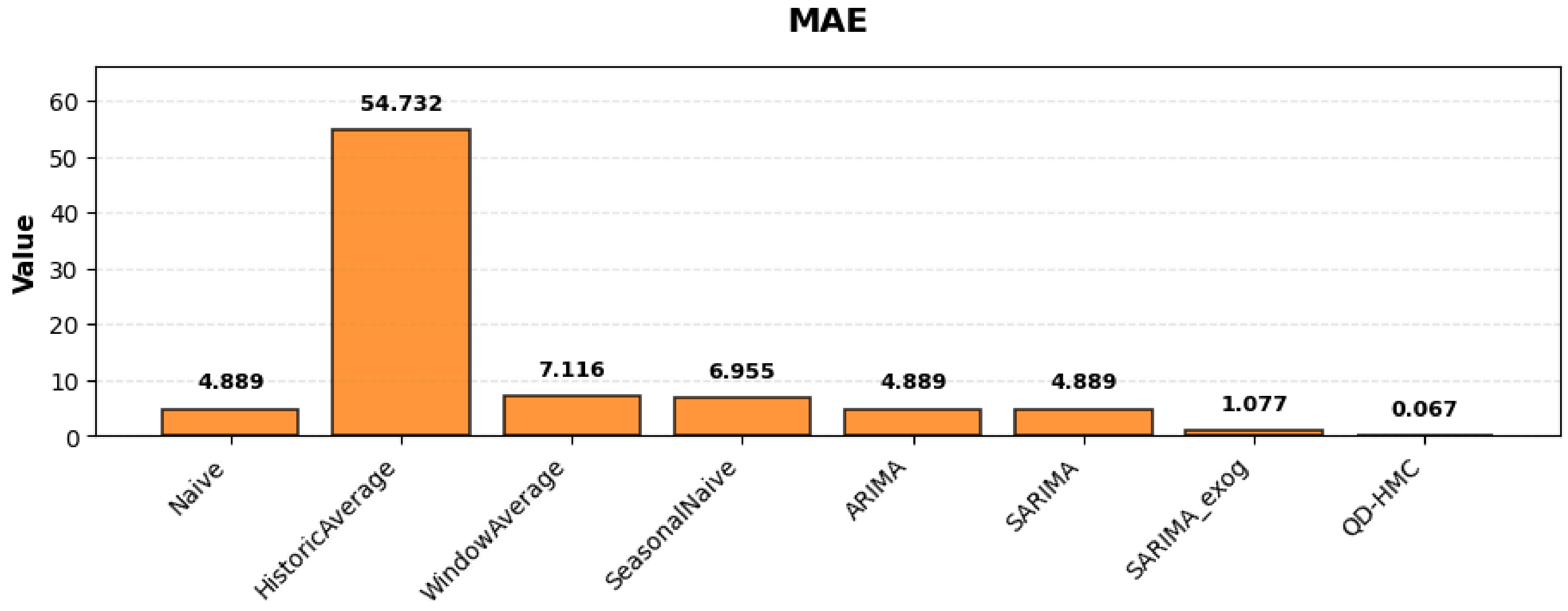


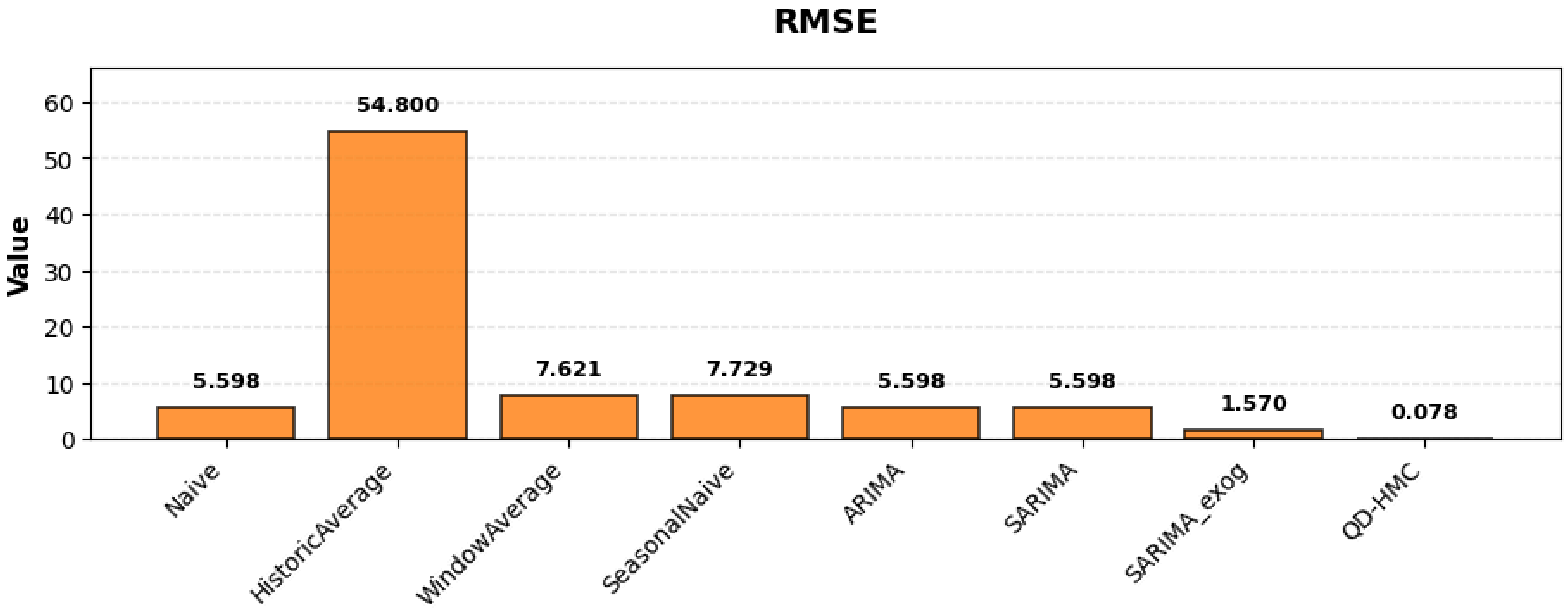
Prediction Results



Models Evaluation







Conclusions

What we learned from the study

- **Distribution-first works:** We can model the distribution of AAPL daily returns and produce calibrated next-day uncertainty under NISQ-friendly constraints.
- **QCBM (generative):** Shallow, 4–5 qubit circuits are viable. Best settings were found with circular entanglement and a wider lag window for conditioning. Metrics indicate realistic shapes with usable intervals.
- **QD-HMC (sampler):** Implemented as a quantum HMC routine using position/momentum operators, kinetic/potential evolutions, a gradient-phase oracle, and MH acceptance. Gate-based precision is limited (about 2–3 bits per parameter), but sampling was feasible with observed acceptance around the expected range.
- **NISQ feasibility:** Both approaches are practical at small scale; results motivate further optimisation rather than deeper circuits for scaling them up.

Future Work

QD-HMC

- Increase precision/qubit allocation; study trade-offs between step size, number of steps, and acceptance.
- Hardware experiments and error-mitigation strategies; explore operator-based variants for higher precision.
- Benchmark against classical HMC and other samplers (e.g., NUTS).

QCBM

- Explore alternative ansatze, entanglement topologies; compare with normalising-flow-inspired QCBMs.
- Conditioning strategies: test feature gating, learned embeddings, and different lag horizons.
- Error mitigation and parameter-efficient training to stay NISQ-friendly.
- Cross-asset generalisation and robustness checks (different equities, regimes).

Thank You!