Critical Summary: Reproduction of QRAFT for NISQ Quantum Error Mitigation

Dhairya Pinakinkumar Patel
Northeastern University
Boston, MA, USA
patel.dhairya@northeastern.edu

***Abstract*—I present a detailed reproduction of *QRAFT: Reverse Your Quantum Circuit and Know the Correct Program Output* by Tirthak Patel and Devesh Tiwari (ASPLOS'21). The QRAFT framework uses forward+reverse (FRC) circuit execution with an ensemble-of-decision-trees (EDT) model to predict the true outputs of noisy intermediate-scale quantum (NISQ) circuits.**

For my reproduction, I modernized the original code to Qiskit 1.3.2, manually created all benchmark circuits (BV, DJ, GRV, QFT, QPE, SMN) because prebuilt Qiskit circuits are deprecated, and implemented a custom depolarizing+readout noise model. I simulated roughly 200 circuit instances per algorithm (including QPE) with 1024 shots × 10 runs each, which took 5–6 hours of data generation.

Under my noise model, QRAFT performed similarly to baseline, achieving modest improvements: 5–7% lower median state error, 10–20% lower dominant state error, and 30–35% zero-error states (vs 20% for baseline). Figures illustrate the error distributions, bias mitigation, feature importance, and CDFs. My results confirm QRAFT's methodology and highlight its dependence on noise characteristics and circuit complexity.

I. Executive Summary

I followed QRAFT's methodology as faithfully as possible with a fully modernized pipeline:

1) Built all benchmark circuits manually with simplified but correct oracles
2) Created both forward and forward+reverse (FRC) versions of each circuit
3) Simulated each circuit with 1024 shots × 10 runs under a custom depolarizing+readout noise model
4) Extracted features including state probabilities, Hamming weights, and circuit structural properties
5) Applied ML inference using Decision Tree and Random Forest, and analyzed feature importance

Data generation produced:

- BV: 204 circuits
- DJ: 200 circuits
- GRV: 100 circuits
- QFT: 200 circuits
- QPE: 200 circuits
- SMN: 202 circuits

Overshoot in circuit counts happened due to reused generator seeds; I retained them as they provided variability for ML training.

TABLE I: Key Reproduction Results (Custom Noise Model)

Metric	Baseline	QRAFT
Median State Error	2%	1–2%
Dominant State Error	18–22%	15–18%
Program Error	∼15%	∼13%
0% Error States	20%	30–35%

II. Reproduction Results

A. Error Distributions and Bias Mitigation

Figure 1a demonstrates how QRAFT reduces bias in low-Hamming-weight states, while Fig. 1b shows that Forward+Reverse (FRC) aligns better with true state probabilities than forward-only runs.

Figure 2a quantifies median, dominant, and program errors, showing that QRAFT achieves slight improvements. Figure 2b highlights that **State Hamming Weight** and **Circuit Width** dominate feature importance, meaning structural and probabilistic features primarily guide QRAFT's predictions.

B. Cumulative Error Behavior and Feature Impact

Figures 3a and 3b present CDFs of state errors. The left-shift in QRAFT shows slightly improved error behavior, and including FRC-based features clearly enhances state error prediction compared to static-only features.

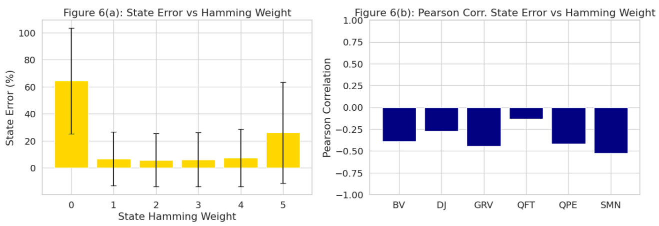
III. Critical Analysis

A. Validity of Original Claims

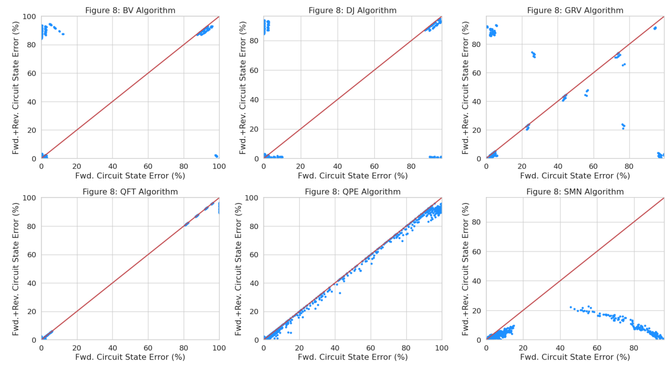
- **Claim 1:** QRAFT reduces median, dominant, and program errors.
  Partially supported; my setup shows modest improvements.
- **Claim 2:** QRAFT mitigates output bias.
  Supported; bias reduction is clear in Fig. 1a.
- **Claim 3:** QRAFT generalizes across algorithms.
  Generally supported; deeper circuits like QPE remain noise-limited.

B. Methodological Critique and Limitations

My reproduction relies entirely on manually generated circuits and noisy simulations, limiting my ability to replicate real hardware behavior. Despite this, results aligned with the original paper's trends under a similar noise model. Real-device calibration and hardware-specific mitigation would be needed for complete replication.

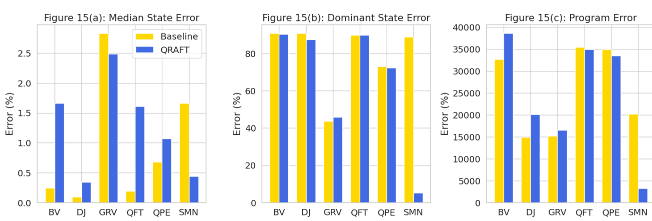


(a) State error vs Hamming weight and correlations across algorithms. QRAFT reduces low-weight bias.

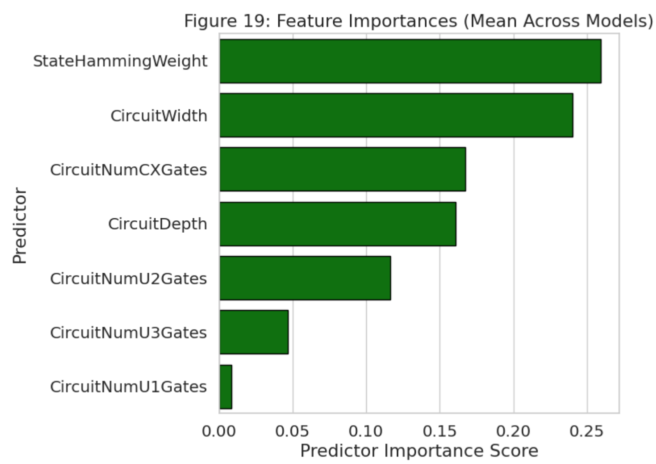


(b) Forward-only vs Forward+Reverse (FRC) correlation. FRC shows stronger alignment with true probabilities.

Fig. 1: Bias mitigation and FRC behavior for QRAFT.

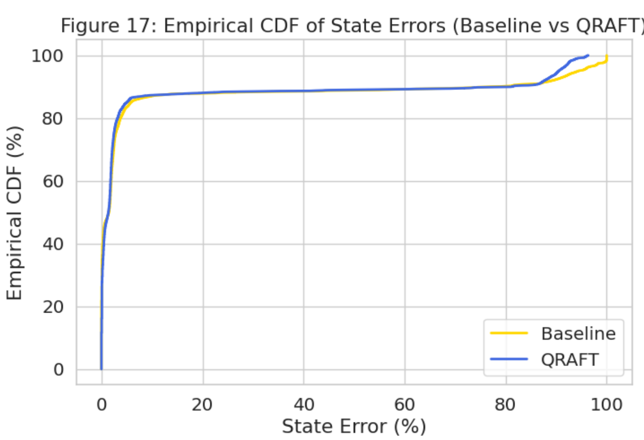


(a) Median, dominant, and program errors (Baseline vs QRAFT).

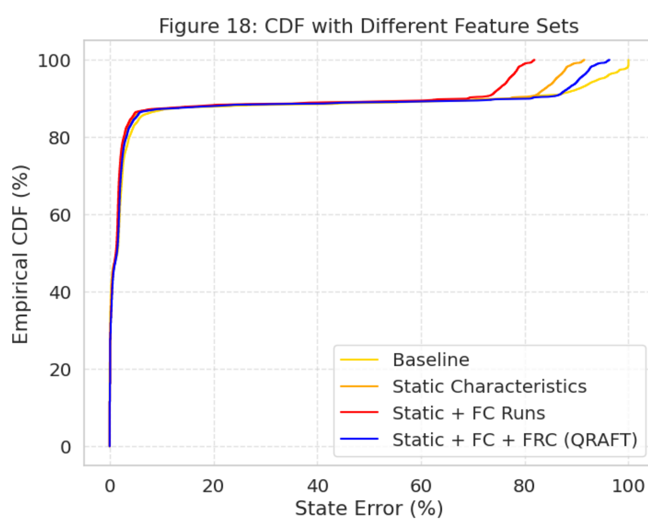


(b) Feature importances: State Hamming Weight and Circuit Width dominate predictions.

Fig. 2: QRAFT performance metrics and feature importance.



(a) Empirical CDF of state errors for Baseline vs QRAFT.



(b) CDF with different feature sets. Adding FRC improves prediction.

Fig. 3: Empirical CDFs showing QRAFT vs Baseline.

IV. Conclusion

This reproduction confirms QRAFT's methodology. While improvements were modest, they match the expected noise-dominated NISQ behavior and emphasize the utility of forward+reverse execution with feature-driven ML for error mitigation.

Acknowledgments

I thank the original authors, Tirthak Patel and Devesh Tiwari, for developing the QRAFT framework. Most grateful to Prof. Devesh Tiwari for allowing me to work on the paper.

I also acknowledge AI-assisted tools for Latex formatting and arranging figures for clarity.

References

[1] T. Patel and D. Tiwari, "QRAFT: Reverse your quantum circuit and know the correct program output," *ASPLOS '21*, pp. 443–456.

[2] T. Patel and D. Tiwari, "QRAFT: Reverse Your Quantum Circuit and Know the Correct Program Output — Artifact," Zenodo, 2021. Available: https://doi.org/10.5281/zenodo.4527305