Pauli Check Sandwiching for Quantum Characterization and Error Mitigation during Runtime

Joshua Gao[1], Ji Liu[2], Alvin Gonzales[2], Zain H. Saleem[2], Nikos Hardavellas[3], Kaitlin N. Smith[3*]
[1]*Department of Computer Science, Virginia Tech, Blacksburg, VA, USA*
[2]*Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL, USA*
[3]*Department of Computer Science, Northwestern University, Evanston, IL, USA*
[*]*kns@northwestern.edu*

***Abstract*—This work presents a novel quantum system characterization and error mitigation framework that applies Pauli check sandwiching (PCS). We motivate our work with prior art in software optimizations for quantum programs like noise-adaptive mapping and multi-programming, and we introduce the concept of PCS while emphasizing design considerations for its practical use. We show that by carefully embedding Pauli checks within a target application (i.e. a quantum circuit), we can learn quantum system noise profiles. Further, PCS combined with multi-programming unlocks non-trivial fidelity improvements.**

I. Introduction and Motivation

While the quantum compute stack has improved, existing quantum system noise still varies cross-chip (Fig. 1) and hovers above error correction thresholds. Much work has been dedicated to improving quantum program success. For example, noise-adaptive mapping is a state-of-art quantum circuit optimization technique used to place and route quantum programs on quantum hardware [5]. Noise-aware schemes, however, depend on accurate calibration data from the quantum computer (QC) provider or from benchmarking. Even in the case that a snapshot of machine properties were available, QC noise is observed to fluctuate over time in ways that are challenging to capture with analytical models.

In our work, we seek alternative methods to determine the noise profile of a QC when its properties are unknown. It is desirable to develop characterization techniques with minimal overheads in terms of quantum processor unit (QPU) runtime. We are inspired by prior art in multi-programming [1] and ensemble distributions [6] to explore how parallelization during runtime can be employed for 1) learning quantum noise / hardware characterization, 2) fidelity improvements in program outcomes, and 3) reductions in runtime latency. This work presents a novel framework that combines the power of multi-programming with Pauli check sandwiching (PCS) for QC characterization and error mitigation. Multi-programming is a technique for parallelized quantum circuit execution while PCS is an error mitigation strategy that utilizes post-selection to reduce errors in quantum circuit outcomes. Our techniques intelligently embed PCS into quantum applications. These programs are then simultaneously executed across a QPU for noise profile characterization. Resulting characterization data adaptively mitigates noisy outcomes within individual thread outcomes. Knowledge of the QPU's best regions creates a weighted ensemble result with non-trivial gain, found to be up to a 25% fidelity improvement in our reported results.

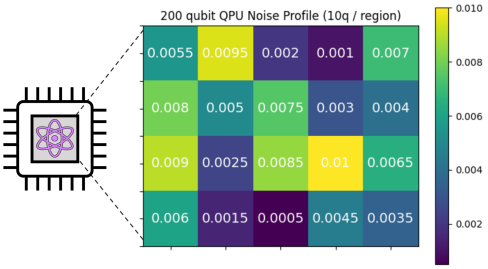

Fig. 1. Example QPU with 20 regions (10 qubits/region) of unique 1q error rates, $p \in [0.0005, 0.01]$, and 2q errors, $2p$. PCS-based characterization with multi-programmed circuits learns the QPU's unknown noise properties.

II. Pauli Check Sandwiching

We rely on a technique called 'Pauli Check Sandwiching' (PCS) to detect and mitigate errors [2]–[4]. In our work, we also find that PCS is a powerful tool for guiding QC characterization. As seen in Fig. 2, PCS surrounds a payload circuit, U, with controlled Pauli operator checks. Errors on U can be detected on an ancilla through phase kickback. It is important that the relationship

$$R_1 U L_1 = U \tag{1}$$

holds so the introduced checks do not disturb the original payload circuit semantics.

Recognizing that all errors can be decomposed into Pauli operators is a key concept in understanding PCS. By cleverly constructing Pauli sandwich checks dependent on U, anti-commutativity relations between the errors and checks can reveal errors in the ancilla qubits. Measuring 1 in any ancillas indicates that phase kickback occurred, meaning anti-commutation between the Paulis, and thus, a present error. As part of the PCS protocol, we discard this error-corrupted shot, leading to an increase in final distribution fidelity of U.

Practical application of PCS requires consideration of trade-offs. First, determining the Pauli unitary used in the check increases in complexity with the circuit. Second, Pauli check logic must be decomposed into the supported gate set of the QC. Third, the overhead of adding PCS into the circuit must

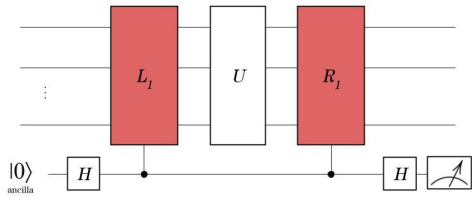

Fig. 2. General PCS circuit layout. The red unitaries represent the Pauli checks that sandwich the main payload circuit. Measurement of the ancillas provide detection of errors that occurred in the payload circuit.

not outweigh the corrective benefit in terms of gate error. We consider these constraints in our solution.

III. QC Characterization and Error Mitigation with PCS

With the PCS protocol, we can 1) learn the noise profile of the QC and 2) increase circuit fidelity by eliminating the states with errors. First, we utilize single-qubit checks only on edge qubits to minimize PCS gate count overhead. We then maximally utilize hardware via multi-programming; a QC with n physical qubits holds

$$threads = \lfloor n/(q_{algorithm} + q_{ancilla}) \rfloor \tag{2}$$

instances of a circuit running in parallel. The value of the checks is used during post processing to filter errored outcomes from each of the threads, creating vectors of error-mitigated counts, c_i. The percentage of discarded states, d_i, gives insight into the underlying noise profile of the QC when they are compared. Each local d_i is then used as a weight that scales c_i to create s_i:

$$s_i = c_i * \frac{min(d)}{d_i}. \tag{3}$$

a cumulative distribution, S, is created from the sum of the s_i vectors from the PCS-protected local threads, $S = \sum s_i$.

IV. Experimental Results

We simulated the fidelity rates of PCS in both a GHZ / sensing circuit and Toffoli circuit to concretely measure the improvement. We expand the Fig. 1 model QC and study a system with 60 10-qubit regions where single-qubit depolarizing noise $p \in [0.0005, 0.03]$ and two-qubit gates $2p$. Fig. 3 shows our results, detailing a multiple percentage point increase in fidelity when multi-programmed applications are protected with PCS and a cumulative distribution is created from local results weighted by check data. Even with only the edge qubits protected, a measurable benefit results. Fig. 4 documents the relationship between discarded shots and error rate, illustrating PCS's potential for QC error characterization.

V. Conclusion and Future Work

We find that PCS not only mitigates error in individual circuit distributions – check data can also be used to intelligently combine the results of multiple threads cross-QPU, demonstrating measurable noise reductions in final outcomes.

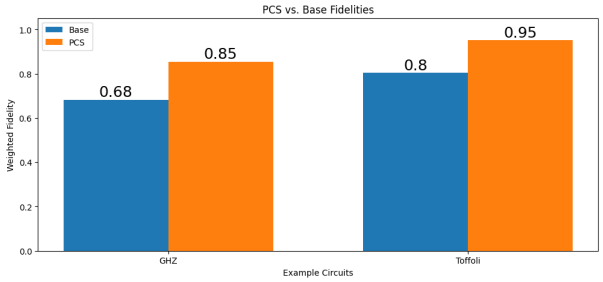

Fig. 3. Differences between fidelity measurements of a circuit with no PCS (base) vs. PCS protection. 10,000 shots were used for both circuits, and fidelity improvement was determined to be 25% (left) and 18.75% (right).

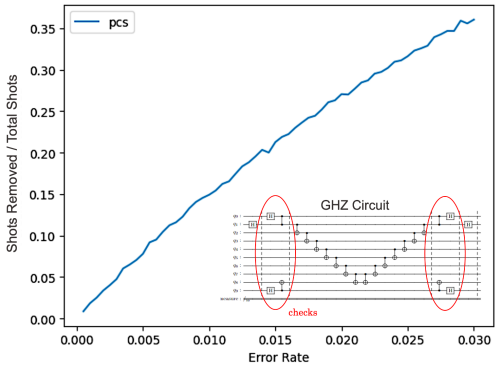

Fig. 4. Shots removed / total shots as error rates increase. 10,000 shots on a GHZ circuit were executed for every error rate (60 total).

But even more powerfully, PCS has additional promise to guide efficient qubit mapping. When lacking reliable data on the noise profile of a QC, it is important that we find methods to best uncover it, guiding hardware utilization, especially if a shot budget must be considered. We theorize that running PCS circuits on different sections of a quantum chip in parallel could enable real-time qubit mapping with check data.

References

[1] Poulami Das, Swamit S Tannu, Prashant J Nair, and Moinuddin Qureshi. A case for multi-programming quantum computers. In *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, pages 291–303, 2019.

[2] Alvin Gonzales, Ruslan Shaydulin, Zain H Saleem, and Martin Suchara. Quantum error mitigation by pauli check sandwiching. *Scientific Reports*, 13(1):2122, 2023.

[3] Quinn Langfitt, Ji Liu, Benchen Huang, Alvin Gonzales, Kaitlin N Smith, Nikos Hardavellas, and Zain H Saleem. Pauli check extrapolation for quantum error mitigation. *arXiv preprint arXiv:2406.14759*, 2024.

[4] P. Li, J. Liu, A. Gonzales, Z. Saleem, H. Zhou, and P. Hovland. Qutracer: Mitigating quantum gate and measurement errors by tracing subsets of qubits. In *2024 ACM/IEEE 51st Annual International Symposium on Computer Architecture (ISCA)*, pages 103–117, 2024.

[5] Prakash Murali, Jonathan M Baker, Ali Javadi-Abhari, Frederic T Chong, and Margaret Martonosi. Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers. In *Proceedings of the twenty-fourth international conference on architectural support for programming languages and operating systems*, pages 1015–1029, 2019.

[6] Swamit S Tannu and Moinuddin Qureshi. Ensemble of diverse mappings: Improving reliability of quantum computers by orchestrating dissimilar mistakes. In *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, pages 253–265, 2019.