Analyzing crosstalk error in the NISQ era

Siyuan Niu, Aida Todri-Sanial

Analyzing crosstalk error in the NISQ era

Siyuan Niu
LIRMM, University of Montpellier
34095 Montpellier, France
siyuan.niu@lirmm.fr

Aida Todri-Sanial
LIRMM, University of Montpellier, CNRS
34095 Montpellier, France
aida.todri@lirmm.fr

***Abstract*—Noisy Intermediate-Scale Quantum (NISQ) hardware has unavoidable noises, and crosstalk error is a significant error source. When multiple quantum operations are executed simultaneously, the quantum state can be corrupted due to the crosstalk between gates during simultaneous operations, decreasing the circuit fidelity. In this work, we first report on several protocols for characterizing crosstalk. Then, we discuss different crosstalk mitigation methods from the hardware and software perspectives. Finally, we perform crosstalk injection experiments on the IBM quantum device and demonstrate the fidelity improvement with the crosstalk mitigation method.**

***Index Terms*—Quantum computing, Crosstalk, Error mitigation**

I. Introduction

Quantum Computing is expected to solve specific classical intractable problems. Several companies such as IBM, Google, and Rigetti have released their quantum chips with 65, 72, and 31 qubits, respectively. However, these chips are qualified as Noisy Intermediate-Scale Quantum (NISQ) hardware, which have less than one hundred qubits and suffer from unavoidable noises.

Crosstalk is one of the major noise sources not only in superconducting but also in trapped-ion devices. It can corrupt the qubit state when multiple quantum operations are executed simultaneously. Crosstalk has a significant impact on quantum gate error. For instance, [8] shows an increase of `CNOT` error up to 11 times caused by crosstalk. Different protocols were proposed in [1]–[3] to detect and characterize crosstalk in quantum devices. After assessing crosstalk, there are two types of methods to mitigating it. The first one is based on hardware strategies, such as tunable coupling [7], or frequency allocation [5]. From the software perspective, crosstalk has been mitigated by changing simultaneous `CNOT` operations with high crosstalk to execute separately while trading off the increase of the decoherence time as in [8].

In this work, we first introduce how to characterize a quantum device's crosstalk using protocol Simultaneous Randomized Benchmarking (SRB) [3]. Second, we present various state-of-the-art crosstalk mitigation methods. Third, we inject crosstalk on IBM quantum devices to show its impact on output fidelity. Finally, we evaluate the crosstalk mitigation method using several benchmarks to demonstrate the fidelity improvement.

This work is funded by the QuantUM Initiative of the Region Occitanie, France, University of Montpellier, France, and IBM Montpellier, France.

II. Methods

A. Crosstalk characterization using SRB

To characterize the crosstalk effect of a quantum device, we choose the most commonly used protocol – Simultaneous Randomized Benchmarking (SRB) [3] based on its ability to quantify the impact of parallel instructions, such as the crosstalk effect of one quantum gate to another when they are executed at the same time. To characterize the crosstalk effect between gate g_i and g_j, we first perform Randomized Benchmarking (RB) on both gates separately and obtain their independent error rate $\mathcal{E}(g_i)$ and $\mathcal{E}(g_j)$. Then, applying SRB on both gates yields the correlated error rate $\mathcal{E}(g_i|g_j)$ and $\mathcal{E}(g_j|g_i)$ for simultaneous executions. If there exists crosstalk between them, the relation between independent and correlated errors should comply with $\mathcal{E}(g_i|g_j) > \mathcal{E}(g_i)$ or $\mathcal{E}(g_j|g_i) > \mathcal{E}(g_j)$. Previous work [8] has demonstrated that crosstalk is significant due to simultaneous `CNOT` executions, hence, in this work, we only focus on characterizing the crosstalk effect between `CNOT` pairs.

We use the ratio of correlated error to independent error $r(g_i|g_j) = \mathcal{E}(g_i|g_j)/\mathcal{E}(g_i)$ as the indicator of the crosstalk effect of the `CNOT` pairs. We choose IBM Q 7 Casablanca as an example to show the crosstalk effect on this device. As `CNOT` errors vary over each calibration, we characterize the crosstalk effect twice respectively on each possible simultaneous `CNOT` pair of IBM Q Casablanca.

The results of average crosstalk data are presented in Fig. 1. The crosstalk effect remains stable regardless of the variation of gate errors across days. For each experiment, there are 150 Cliffords in the sequence and each sequence is repeated five times due to the expensive runtime cost. The crosstalk effect is not as severe in this device as in other devices such as IBM Q 20 Poughkeepsie, where the gate error grows up to 11 times caused by crosstalk [8]. However, the error rate is still amplified up to 3 times in IBM Q Casablanca. Sometimes ratios are less than one, which is probably due to the limits of crosstalk metric in SRB protocol [6].

B. Crosstalk mitigation

After characterizing the crosstalk effect, the next challenge is how to mitigate it. The hardware-oriented crosstalk mitigation strategies have several physical constraints. For example, the tunable coupling [7] is only applicable for tunable coupling superconducting devices, whereas the frequency allocation [5]

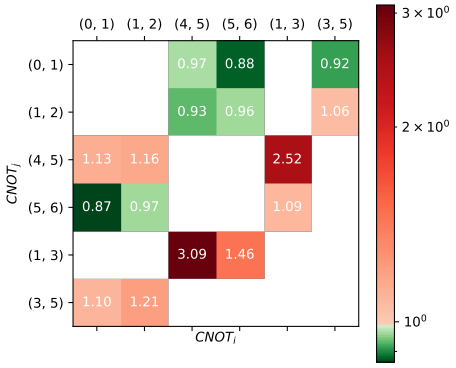

Fig. 1: The SRB result of IBM Q 7 Casablanca. The number represents the ratio of correlated error to independent error. Note that the SRB experiment is performed on CNOTs that can be executed in parallel, which means they do not share the same qubits. There are no SRB experiments on blank spots due to the sharing qubits.

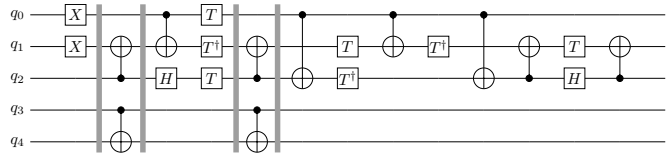

Fig. 2: Crosstalk injection of CSWAP circuit. We introduce two other qubits (q_3, q_4) and apply CNOT operations on them. Barriers are inserted to make sure the CNOTs are executed simultaneously. Note that, crosstalk injection are performed on CNOT pairs with strong crossalk error.

is mainly designed for fixed-frequency transmon qubit devices. Despite hardware improvement approaches to reduce crosstalk, there are also software methods to address crosstalk. Here, we report on hardware-agnostic software crosstalk mitigation approaches.

An intuitive approach for software crosstalk mitigation is to make simultaneous CNOTs execute serially to avoid simultaneous high crosstalk CNOTs. However, serial instructions can increase the circuit depth and cause decoherence errors. Therefore, a scheduler is required to trade-off between them. The state of the art [8] proposes a crosstalk-adaptive scheduler (labeled as XtalkSched) based on SMT optimization that re-schedules a quantum intermediate representation (IR), taking into account both crosstalk and decoherence error. It inserts barriers between the simultaneous CNOTs with a significant crosstalk effect to avoid parallel instructions while accounting for the qubit coherence time.

III. Evaluation

A. Methodology

Benchmarks. We first use a 3-qubit CSWAP gate circuit and inject a different number of simultaneous CNOTs to show

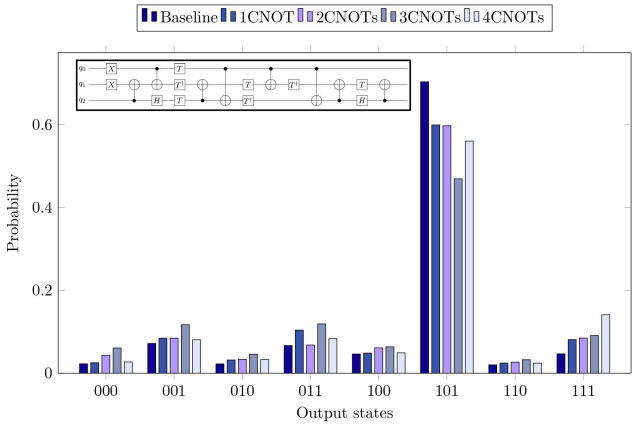

Fig. 3: The output state probability distribution of CSWAP gate circuit. The right output state should be '101'. We inject a different number of simultaneous CNOTs to this circuit. The baseline circuit has zero simultaneous CNOT.

the impact of crosstalk on the output fidelity of a quantum circuit. An example of demonstrating crosstalk injection by enabling two simultaneous CNOTs is shown in Fig. 2. Then, we examine benchmarks collected from the previous works [4], [8], including SWAP circuits, Bell state circuit, etc., to show the performance of XtalkSched. For each benchmark, we select its mapping to ensure it includes at least one pair of CNOTs with a high crosstalk effect.

Algorithm configuration. For crosstalk characterization of IBM Q Casablanca, we set the crosstalk threshold to 2, and w used in XtalkSched is set to 0.5 to trade-off circuit depth and crosstalk mitigation. The Qiskit version is 0.23.6.

Comparison. We compare XtalkSched [8] with ParSched, which is the current scheduler used in Qiskit to make instructions execute in parallel without considering the crosstalk.

B. Experimental results

The experimental results of crosstalk injection are shown in Fig. 3. The probability of obtaining the right output state is reduced with the increase of simultaneous CNOTs due to the injected crosstalk error. In the worst case, the fidelity is decreased by 33.3% compared to the non-crosstalk case.

The comparison between XtalkSched with ParSched in terms of output fidelity and circuit depth is shown in Fig. 4. The fidelity is improved by 9.4%, whereas the circuit depth increased by 39.1%. Although the fidelity is enhanced because of XtalkSched, the circuit depth increase is not negligible. This raises the need for more effective and practical crosstalk mitigation methods.

IV. Conclusion

Crosstalk is a non-negligible error source in quantum computers. This work examines the crosstalk characterization protocol based on simultaneous randomized benchmarking, and we report on the crosstalk effects on the IBM quantum

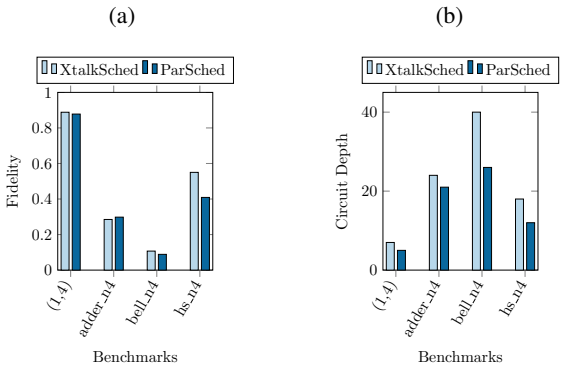

Fig. 4: (a) Fidelity. (b) Circuit depth. Note that (1,4) represents the SWAP circuit which aims to realize a connection between q_0 and q_4 through SWAP operations.

device. We also evaluate the crosstalk-adaptive scheduler and illustrate its impact on output fidelity and circuit depth.

REFERENCES

[1] Radoslaw C Bialczak, Markus Ansmann, Max Hofheinz, Erik Lucero, Matthew Neeley, AD O'Connell, Daniel Sank, Haohua Wang, James Wenner, Matthias Steffen, et al. Quantum process tomography of a universal entangling gate implemented with josephson phase qubits. *Nature Physics*, 6(6):409–413, 2010.

[2] Alexander Erhard, Joel J Wallman, Lukas Postler, Michael Meth, Roman Stricker, Esteban A Martinez, Philipp Schindler, Thomas Monz, Joseph Emerson, and Rainer Blatt. Characterizing large-scale quantum computers via cycle benchmarking. *Nature communications*, 10(1):1–7, 2019.

[3] Jay M Gambetta, Antonio D Córcoles, Seth T Merkel, Blake R Johnson, John A Smolin, Jerry M Chow, Colm A Ryan, Chad Rigetti, S Poletto, Thomas A Ohki, et al. Characterization of addressability by simultaneous randomized benchmarking. *Physical review letters*, 109(24):240504, 2012.

[4] Ang Li and Sriram Krishnamoorthy. Qasmbench: A low-level qasm benchmark suite for nisq evaluation and simulation. *arXiv preprint arXiv:2005.13018*, 2020.

[5] Gushu Li, Yufei Ding, and Yuan Xie. Towards efficient superconducting quantum processor architecture design. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 1031–1045, 2020.

[6] David C McKay, Andrew W Cross, Christopher J Wood, and Jay M Gambetta. Correlated randomized benchmarking. *arXiv preprint arXiv:2003.02354*, 2020.

[7] Pranav Mundada, Gengyan Zhang, Thomas Hazard, and Andrew Houck. Suppression of qubit crosstalk in a tunable coupling superconducting circuit. *Physical Review Applied*, 12(5):054023, 2019.

[8] Prakash Murali, David C McKay, Margaret Martonosi, and Ali Javadi-Abhari. Software mitigation of crosstalk on noisy intermediate-scale quantum computers. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 1001–1016, 2020.