# Random Stimuli Generation for the Verification of Quantum Circuits

Lukas Burgholzer, Richard Kueng, Robert Wille

August 30, 2023

# Contents

# List of Figures



# List of Tables

# 1 Introduction

The verification of quantum circuits is a crucial aspect in ensuring the correctness of quantum algorithms and descriptions at various levels of abstraction. As quantum circuits grow in size and complexity, the task of their verification becomes increasingly challenging. Traditional verification methods that involve the comparison of entire unitary matrices become computationally expensive and infeasible for larger circuits.

This report focuses on a recent study by Burgholzer et al. titled ”Random Stimuli Generation for Quantum Circuits” which offers a novel approach to this problem. The study introduces a method of using random stimuli to verify quantum circuits. It presents three schemes for quantum stimuli generation that offer a trade-off between the error detection rate and efficiency. The paper shows that even with a few randomly-chosen stimuli generated from the proposed schemes, high error detection rates can be achieved for quantum circuits.

The goal of this report is to reproduce the results of this paper using Qiskit, a popular open-source quantum computing framework, and to present a comprehensive experimental framework for the task.

# 2 Theory

The theoretical foundation of the paper is the generation of random stimuli for quantum circuit verification. In the context of quantum circuit verification, a stimulus is a quantum state used as input to the quantum circuit. The output state resulting from the application of the input stimulus on the quantum circuit is then measured and compared to the expected result. This verification method has a significant advantage in terms of computational cost over traditional methods. The paper introduces three schemes for stimuli generation: classical, local, and global, each with its own advantages and challenges.

## 2.1 Classical Stimuli

The simplest form of stimuli are classical stimuli which are computational basis states. This involves selecting a quantum state $|\phi\rangle$ from the set $\{|i\rangle : i \in \{0,1\}^n\}$, where $n$ is the number of

qubits. The fidelity of the circuit, which is a measure of how close two quantum states are, is then computed as $F(U|i\rangle, V|i\rangle)$, where $U$ is the unitary transformation representing the ideal quantum circuit, and $V$ is the unitary transformation associated with $G$, the quantum circuit with error.

However, classical stimuli have a significant limitation: they are not faithful. This means that there can exist an infinite number of quantum circuit realizations that yield a fidelity of 1 for all classical stimuli, even if there exist quantum states for which the fidelity is not 1.

## 2.2 Local Quantum Stimuli

Classical single-qubit stimuli ($|0\rangle$ and $|1\rangle$) aren't expressive enough to reliably probe the full range of quantum operations. They correspond to antipodal points on the Bloch sphere, and tracking only two antipodal points makes it impossible to detect certain transformations. Local quantum stimuli is a more expressive set of stimuli that goes beyond classical basis states. This set of stimuli comprises three pairs of antipodal points on the Bloch sphere sufficient for full resolution, namely $|0\rangle, |1\rangle$ (Z-basis), $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ (X-basis), and $|\uparrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle), |\downarrow\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$ (Y-basis).

For $n$ qubits, we consider the ensemble of local quantum stimuli:

$$|l\rangle = |l_{n-1}\rangle \otimes \ldots \otimes |l_0\rangle \quad \text{with} \quad |l_i\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\uparrow\rangle, |\downarrow\rangle\}$$

For each pair of functionally distinct $n$-qubit unitaries $U$ and $V$, there exists at least one local quantum stimulus $|l\rangle$ that detects the error, i.e., yields $F(U|l\rangle, V|l\rangle) \neq 1$.

The local scheme applies a random gate from a predefined set $\{I, X, H, XH, HS, XHS\}$ to each qubit in the circuit.

## 2.3 Global Quantum Stimuli

The local quantum stimuli can be further extended to global quantum stimuli for even more expressive and "more quantum" stimuli. These stimuli are generated by starting with a simple classical state (like $|0\ldots0\rangle$) and applying layers of single- and two-qubit gates to generate a stimulus $|g\rangle$, with $|g\rangle = (G_0 \ldots G_{l-1})|0\ldots0\rangle$, where each $G_i$ is a layer comprised of Clifford gates

(H, S, CNOT) including all local quantum stimuli.

The global quantum stimuli can detect any error and can typically detect errors earlier, i.e., after substantially fewer iterations. However, the generation of global quantum stimuli and subsequent simulation is more resource-intensive.

The expected outcome fidelity (averaged over all possible global quantum stimuli $|g\rangle$) accurately approximates one of the most prominent distance measures for $n$-qubit quantum circuits:

$$E_{|g\rangle} F(U|g\rangle, V|g\rangle) \approx F_{\text{avg}}(U, V) = \frac{1}{2^n + 1} \left( 1 + 2^n \left| \left| \frac{\text{tr}(U^\dagger V)}{2} \right| \right|^2 \right)$$

The average fidelity forms the basis of many state-of-the-art quantum calibration procedures. If the quantum circuit has an error, the right-hand side of this equation is typically much smaller than 1, allowing us to detect errors efficiently using global quantum stimuli. However, generating and using global quantum stimuli is computationally more expensive, leading to a trade-off between the error detection rate and efficiency.

**Table 1: Experimental results (quantities averaged over a total of approx. $10^6$ different simulations)**

| | Remove 1 random gate | | | Remove 2 random gates | | | Remove 3 random gates | | |
|---|---|---|---|---|---|---|---|---|---|
| Approach | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] |
| Classical Stimuli (Section 3.1) | 86.9 ± 4.1 | 4.0 ± 0.8 | **0.2 ± 0.3** | 97.9 ± 2.2 | 1.7 ± 0.4 | **0.1 ± 0.1** | 99.6 ± 0.8 | 1.2 ± 0.2 | **0.1 ± 0.1** |
| Local Quantum Stimuli (Section 3.2) | 98.8 ± 1.6 | 1.5 ± 0.3 | 0.7 ± 1.3 | **100.0 ± 0.0** | 1.1 ± 0.1 | 0.5 ± 0.9 | **100.0 ± 0.0** | **1.0 ± 0.0** | 0.7 ± 1.1 |
| Global Quantum Stimuli (Section 3.3) | **99.0 ± 1.5** | **1.2 ± 0.2** | 50.1 ± 103.0 | **100.0 ± 0.0** | **1.0 ± 0.0** | 56.9 ± 113.1 | **100.0 ± 0.0** | **1.0 ± 0.0** | 68.7 ± 115.3 |
| | Add 1 random gate | | | Add 2 random gates | | | Add 3 random gates | | |
| Approach | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] |
| Classical Stimuli (Section 3.1) | 54.9 ± 4.7 | 7.8 ± 0.7 | **0.4 ± 0.5** | 80.7 ± 5.0 | 3.9 ± 0.8 | **0.2 ± 0.2** | 90.8 ± 4.2 | 2.4 ± 0.6 | **0.1 ± 0.1** |
| Local Quantum Stimuli (Section 3.2) | 73.9 ± 7.4 | 5.1 ± 1.1 | 2.9 ± 5.1 | 92.5 ± 3.9 | 2.2 ± 0.6 | 1.1 ± 2.0 | 97.5 ± 2.8 | **1.4 ± 0.4** | 0.6 ± 0.9 |
| Global Quantum Stimuli (Section 3.3) | **75.9 ± 10.1** | **4.6 ± 1.5** | 80.9 ± 118.1 | **92.9 ± 4.3** | **2.1 ± 0.6** | 47.9 ± 97.1 | **97.6 ± 2.8** | **1.4 ± 0.4** | 38.2 ± 93.1 |
| | Add 10 random Toffolis at beginning | | | Add 10 random Toffolis at end | | | **Overall** | | |
| Approach | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] | $p_s$ [%] | $\varnothing s$ | $\varnothing t$ [s] |
| Classical Stimuli (Section 3.1) | 82.0 ± 11.7 | 5.3 ± 1.9 | **0.5 ± 0.7** | 80.3 ± 12.2 | 5.3 ± 2.0 | **0.5 ± 0.7** | 84.1 ± 5.6 | 3.9 ± 0.9 | **0.3 ± 0.3** |
| Local Quantum Stimuli (Section 3.2) | 82.3 ± 11.6 | 4.0 ± 1.8 | 2.8 ± 5.3 | 80.6 ± 12.0 | 4.1 ± 1.8 | 2.5 ± 5.1 | 90.7 ± 4.9 | 2.5 ± 0.8 | 1.5 ± 2.7 |
| Global Quantum Stimuli (Section 3.3) | **82.9 ± 12.1** | **3.6 ± 1.8** | 79.9 ± 120.2 | **81.2 ± 12.6** | **3.8 ± 1.9** | 66.7 ± 116.7 | **91.2 ± 5.4** | **2.3 ± 0.8** | 61.2 ± 109.6 |

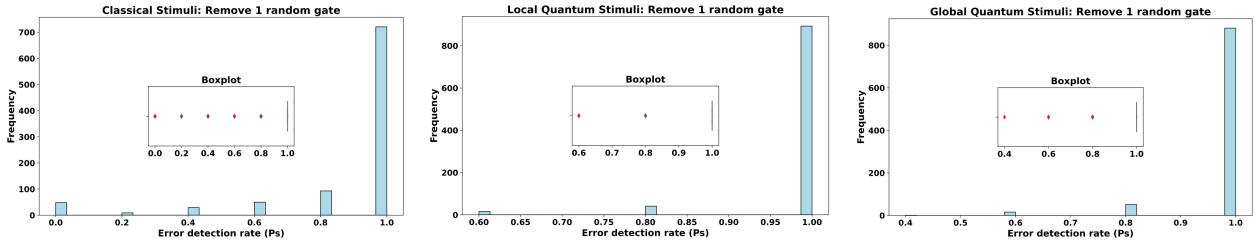
$p_s$ [%]: Error detection rate in percent $\varnothing s$: Average number of stimuli $\varnothing t$ [s]: Average runtime in seconds

Since the obtained results are rather homogeneous across the respective benchmarks (as confirmed by the moderate standard deviation), we only list averaged values here.

**Figure 1:** Experimental results [1]

# 3 Results

In the paper, results presented in Table 1 (see Figure 1) highlighted the efficiency of various quantum stimuli generation schemes in error detection. Despite only providing averaged values, the results were consistent across benchmarks. Notably, all methods showcased remarkable error detection rates, even with random stimuli. The classical stimuli generation often underperformed, especially when specific gates (e.g diagonal gates) were added or removed due to the fact that




**(a)** Classical **(b)** Local **(c)** Global

**Figure 2:** This figure shows the variability of the average error detection rate for the different stimuli schemes across the various quantum circuit instances. The average error detection rates for classical stimuli in Fig. 2a are more spread-out than for the local stimuli in Fig. 2b and the global stimuli in Fig. 2c which account for the high standard deviations.

classical stimuli aren't as expressive in capturing the full quantum behavior, but was fast, making it apt for rapid prototyping. This is reflected in the lower error detection rates for classical stimuli compared to local quantum stimuli and global quantum stimuli. Global stimuli generation emerged as the most robust method, demanding fewer stimuli and achieving the highest detection rates, validating discussions on their quality. However, it comes at the cost of being computationally more expensive. Meanwhile, local quantum stimuli generation offered a balance between quality and efficiency. The paper underscored the superiority of simulative verification in quantum realms over classical, revealing that high error detection rates in quantum circuits can be achieved with just a few randomly-chosen stimuli, emphasizing the immense potential of this verification approach.

The results of our paper replication can be seen in Table 1. The reproduced result shows an overall average error detection rate of 86.1%, 98.3%, and 98.2% for classical stimuli, local quantum stimuli, and global quantum stimuli respectively. The replicated results match the general trend from the results in the paper. As shown in the table, the classical stimuli perform worse than the other two schemes, while local and global quantum stimuli have about the same performance. The error detection rate of local quantum stimuli was comparable to that of global quantum stimuli probably because of the complexities of the circuits. We believe that local stimuli would perform slighlty worse than global quantum stimuli for more complex circuits. Moreover, we observed a higher standard deviation in our replicated result compared to the results of the paper, particularly with the average error detection rate of classical stimuli. This suggests that the average detection rates were quite variable across different circuits. This variability might be due to the limitations of

classical stimuli in detecting certain types of errors in quantum circuits. Although this variability was not as pronounced in local quantum stimuli and global quantum stimuli, our results could have differed from the results in the paper due to using different benchmark quantum circuits. As noted, the complexities of the quantum circuits used as benchmarks are not comparable to that used in the paper. To understand the variability trend, we examined the frequency of the average error detection rates for a select error type for the different stimuli schemes across the different circuit instances, as shown in Figure 2. The plots in the figure show that the error detection rates (averaged over 5 data points) for classical stimuli were more spread out compared to the other two schemes, resulting in higher standard deviation from the mean value.

| | | **classical** | **local** | **global** |
|---|---|---|---|---|
| Remove 1 random gate | $p_s[\%]$ | 87.7 ± 26.6 | 98.3 ± 6.6 | 98.1 ± 7.4 |
| | $\varnothing_s$ | 1.4 ± 1.2 | 1.0 ± 0.2 | 1.0 ± 0.2 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Remove 2 random gates | $p_s[\%]$ | 91.8 ± 23.5 | 98.6 ± 6.0 | 98.3 ± 6.7 |
| | $\varnothing_s$ | 1.3 ± 1.1 | 1.0 ± 0.1 | 1.0 ± 0.2 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Remove 3 random gates | $p_s[\%]$ | 94.2 ± 20.7 | 98.7 ± 6.3 | 98.5 ± 6.5 |
| | $\varnothing_s$ | 1.2 ± 1.0 | 1.0 ± 0.1 | 1.0 ± 0.1 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Add 1 random gate | $p_s[\%]$ | 82.8 ± 27.3 | 98.1 ± 7.2 | 98.1 ± 6.7 |
| | $\varnothing_s$ | 1.4 ± 1.2 | 1.0 ± 0.2 | 1.0 ± 0.1 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Add 2 random gate | $p_s[\%]$ | 86.1 ± 25.2 | 98.0 ± 7.2 | 98.0 ± 7.3 |
| | $\varnothing_s$ | 1.3 ± 1.0 | 1.0 ± 0.2 | 1.0 ± 0.2 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Add 3 random gate | $p_s[\%]$ | 88.5 ± 24.2 | 98.1 ± 7.0 | 98.3 ± 6.2 |
| | $\varnothing_s$ | 1.3 ± 1.0 | 1.0 ± 0.2 | 1.0 ± 0.2 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Add 10 random Toffolis at beginning | $p_s[\%]$ | 79.2 ± 28.0 | 98.6 ± 5.5 | 98.2 ± 6.6 |
| | $\varnothing_s$ | 1.5 ± 1.3 | 1.0 ± 0.1 | 1.0 ± 0.1 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Add 10 random Toffolis at end | $p_s[\%]$ | 78.3 ± 27.8 | 97.8 ± 7.4 | 98.0 ± 7.1 |
| | $\varnothing_s$ | 1.5 ± 1.2 | 1.0 ± 0.1 | 1.0 ± 0.2 |
| | $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |
| Overall | $p_s[\%]$ | 86.1 ± 25.4 | 98.3 ± 6.7 | 98.2 ± 7.1 |
| | $\varnothing_s$ | 1.4 ± 1.1 | 1.0 ± 0.2 | 1.0 ± 0.2 |

| | | | |
|---|---|---|---|
| $\varnothing_t$ | 0.4 ± 0.3 | 0.4 ± 0.3 | 0.5 ± 0.3 |

**Table 1:** Experimental results: $p_s$ [%]: Error detection rate in percent $\varnothing_s$: Average number of stimuli $\varnothing_t[s]$: Average runtime in seconds.

# 4 Workflow and Reproducibility

### 4.0.1 Framework

The Qiskit library for Python was the framework of choice for replicating the results of the paper. This is particularly because Qiskit was the framework used in the study by Burgholzer et al. [1]. Additionally, the available benchmark quantum circuits used were all designed and transpiled using Qiskit.

### 4.0.2 Benchmark Circuits

In the study, the validation of the three schemes was carried out through an empirical evaluation involving approximately $10^6$ simulations conducted across 50,000 benchmark instances, chosen from 25 widely-used reversible/quantum algorithms with 16 to 34 qubits (can be found at the GitHub page). The circuits were taken from Revlib. Only three circuits had qubits less than 20 qubits (two circuits had 16, while the other had 17 qubits). Simulating this on my desktop computer was time-consuming. For some circuits, the transpiled circuits had more qubits than the original circuit. For example, the first original circuit had 17 qubits while its transpiled version had 53 qubits. This would require a lot of compute power to simulate. As a result, I had to select from a different set of quantum circuit benchmarks for verification. The link to the benchmark circuits can be found here.

A total of 19 quantum circuits were used for this experiment. The selected circuits covered a range of complexities and qubit sizes ranging from 3 to 10 qubits. The benchmark circuits were already transpiled to a suitable IBM architecture. The original circuit corresponds to unitary matrix $U$ while the corresponding transpiled circuit corresponds to the unitary matrix $G$ for each circuit. The transpiled circuits were randomly sampled to ensure that they are equivalent to their original ciruits.

### 4.0.3 Algorithm Implementation and Assumptions

To implement the algorithm, I created a function that injects errors into the transpiled circuits. The types of errors include the removal of 1, 2, and 3 random gates, insertion of 1, 2, and 3 random gates, and insertion of 10 random Toffoli gates at the beginning and the end of the circuit. I created three different functions to implement the three proposed schemes: classical stimuli, local quantum stimuli, and global quantum stimuli. From the paper it wasn't quite clear the actual gate set used for generating global quantum stimuli. Also, the number of levels needed for a given stimulus wasn't explicitly stated. Hence, following the statement in the second paragraph of page 5 of the original paper, I assumed that the gate set for global stimuli generation comprises all gates in the local quantum stimuli gate set $\{I, X, H, XH, HS, XHS\}$ in addition to the gates Clifford gates $\{H, S, CNOT\}$ stated in the paper. Since the paper highlighted that the number of layers $l$ for a given stumuli must be greater than 1 and must be proportional to the number of qubits $n$, I generated a random number between 2 and $n$. The random number is the number of layers $l$ for that stimulus on that quantum circuit.

Given a transpiled circuit, for each type of error, I generated 50 random instances and for each resulting instance (i.e., circuit with injected errors), 5 random stimuli were generated for each of the three proposed schemes: classical stimuli, local quantum stimuli, and global quantum stimuli. The resulting 5 circuits and their corresponding ideal version were simulated using IBM Qasm simulator with 1000 shots. The fidelity of each circuit was determined by computing the Helinger fidelity ($F(Q, P) = \left(\sum_i \sqrt{p_i q_i}\right)^2$) of its probability distributions and the probability distribution of its corresponding ideal version. The aim was to detect the injected error. A tolerance of 0.1 was allowed to account for randomness and measurement error. With a fidelity of less than 90 %, the circuits are considered not to be equal, which implies that the injected error has been detected by the corresponding stimuli scheme.

The number of stimuli that detected the error is noted and an average is computed over the 5 random. This means that for each error type and each of the proposed schemes, a total of $19 \times 50 \times 5 = 4,750$ circuits were simulated. If we account for the idea version of each circuit, then

the total number of circuits will be twice that stated above. A total of $2 \times 1000 \times 19 \times 50 \times 5 \times 8 = 7,600,000$ circuit instances were simulated for each scheme if we consider the total number of shots (1000) and the total number of injected errors (8).

The results of the simulations were collected, including the average error detection rates, the number of stimuli needed to detect the error, and the runtime of the respective scheme. It wasn't clear how they were computed from the paper hence, the following assumptions were made:

- The average error detection rate (i.e., the probability that the error is detected by the generated set of stimuli) was computed for each error injection using the 5 randomly generated stimuli for the respective scheme. It was determined based on the number of stimuli that accurately detected the error divided by 5 (the total number of stimuli). This was added for all the instances and the average was computed for a given error injection.

- The number of stimuli needed was computed for each instance based on the 5 stimuli. It checks to see how many stimuli failed before anyone succeeded.

- The time was computed for each instance and averaged over the number of stimuli.

This means that for each error type, we computed the average error detection rate base on $19 \times 50 = 950$ data points, where each data point corresponds to the average over 5 stimuli used in the error detection (see Figure 2 for the data of error type: removal of 1 random gate). Table 1 shows the experimental results, comparing the performance of the three proposed schemes in terms of their error detection rate, the average number of stimuli needed to detect the error, and the average runtime.

In terms of evaluating the performance of the different stimuli generation schemes, a scheme with a higher average error detection rate and a lower standard deviation would generally be considered more reliable and effective, as it consistently detects a high proportion of errors. Furthermore, the error detection rate and the number of required stimuli present an interesting trade-off with respect to efficiency, which forms the basis for further investigations and optimizations in the realm of quantum circuit verification. Overall, the reproduced results seem to agree with the results in

the paper, except that it exhibits a higher standard deviation from the mean due to statistical error.

# References

[1] L. Burgholzer, R. Kueng, and R. Wille, “Random stimuli generation for the verification of quantum circuits,” in *Proceedings of the 26th Asia and South Pacific Design Automation Conference*, 2021, pp. 767–772.