Random Stimuli Generation for the Verification of Quantum Circuits

Lukas Burgholzer* Richard Kueng* Robert Wille*†
*Johannes Kepler University Linz, Austria
†Software Competence Center Hagenberg GmbH (SCCH), Austria
{lukas.burgholzer,richard.kueng,robert.wille}@jku.at
https://iic.jku.at/eda/research/quantum/

ABSTRACT

Verification of quantum circuits is essential for guaranteeing correctness of quantum algorithms and/or quantum descriptions across various levels of abstraction. In this work, we show that there are promising ways to check the correctness of quantum circuits using simulative verification and random stimuli. To this end, we investigate how to properly generate stimuli for efficiently checking the correctness of a quantum circuit. More precisely, we introduce, illustrate, and analyze three schemes for quantum stimuli generation—offering a trade-off between the error detection rate (as well as the required number of stimuli) and efficiency. In contrast to the verification in the classical realm, we show (both, theoretically and empirically) that even if only a few *randomly-chosen* stimuli (generated from the proposed schemes) are considered, high error detection rates can be achieved for quantum circuits. The results of these conceptual and theoretical considerations have also been empirically confirmed—with a grand total of approximately 10^6 simulations conducted across 50 000 benchmark instances.

1 INTRODUCTION

Verification methods are essential for demonstrating or even proving the correctness of classical circuits. Their goal is to confirm whether a given circuit realization conforms to its specification. In this regard, *formal verification* methods [1], [2]—which aim to prove correctness with 100 % certainty—are well established, but often fail due to the exponential complexity of the task itself. In contrast, *simulative verification* methods [3]–[8] are typically very fast as long as only a limited number of simulations with specific stimuli are conducted to achieve a desired coverage. In order to generate high quality stimuli (which indeed are capable of detecting errors), methods such as *constraint-based random simulation* [3]–[6], *fuzzing* [7], [8], etc. are employed.

In the quantum realm, the verification of quantum circuits is essential for guaranteeing correctness of quantum algorithms and/or quantum descriptions across various levels of abstraction. Here, sequences of quantum operations and/or quantum gates are employed which utilize quantum mechanical effects such as *superposition*, *entanglement*, or *interference* [9]. This allows for promising applications in various domains such as chemistry, finance, cryptography, or machine learning. But it also requires a more complex description than in the classical realm. Consequently, the formal verification of quantum circuits poses even more challenges than in the classical realm—which even recent advances [10]–[14] can only escape to a certain extent.

This motivates the consideration of simulative verification in the quantum realm (similar to the classical realm, where this is well established). In this regard, the simulation of quantum circuits on a classical computer hardware is key. Although this leads to an exponential complexity in order to describe the corresponding quantum states and operations, powerful methods have recently been proposed to tackle this problem [15]–[22]. However, while the stimuli space for classical circuits is finite (each input bit can be assigned either 0 or 1—yielding a total of 2^n possible stimuli), the state space in the quantum realm is infinitely large (possible stimuli are elements of a 2^n-dimensional Hilbert space). This raises the question on whether simulative verification of quantum circuits (on classical computers) is suitable at all and, if so, how to generate proper stimuli to efficiently check the correctness of a quantum circuit.

In this work, we show that, although the perspective of a possible infinite number of stimuli may seem rather grim at a first glance, there are promising ways to check the correctness of quantum circuits using simulative verification and random stimuli. This, however, severely depends on how the stimuli are actually generated. In fact, we introduce, illustrate, and analyze three schemes for quantum stimuli generation offering a nice trade-off between error detection rate (as well as the required number of stimuli) and efficiency. In contrast to classical circuits, we show (both, theoretically and empirically) that even if only a few *randomly-chosen* stimuli (generated from the proposed schemes) are considered, high error detection rates can be achieved in the quantum realm. The results of these conceptual and theoretical considerations have also been empirically confirmed, which, to the best of our knowledge, led to the broadest empirical evaluation of simulative verification schemes for quantum circuits to date—with a grand total of approximately 10^6 simulations conducted across 50 000 benchmark instances.

The remainder of this paper is structured as follows: Section 2 provides the necessary background on classical verification, quantum circuits, and their verification. Then, Section 3 introduces, illustrates, and (theoretically) analyzes different stimuli generation schemes and their likeliness of detecting errors. The results of these conceptual and theoretical considerations are then empirically confirmed in Section 4. Finally, Section 5 concludes the paper.

2 BACKGROUND AND MOTIVATION

This work deals with verification of circuits—a topic which has been and currently still is heavily considered in the classical realm. Because of this, we first briefly review the established schemes in this section. Afterwards, we provide the basics on quantum computing and quantum circuits and, based on that, eventually discuss the challenges of the verification of quantum circuits. By this, we motivate our work.

2.1 Verification of Classical Circuits

In order to demonstrate or even prove the correctness of classical circuits, verification methods are applied. They check whether a given circuit, the *Design Under Verification* (DUV), adheres to an also given *Golden Specification*. To this end, current (industrial) practice mainly applies schemes such as

- *simulative verification* [3]–[8], in which certain input assignments (*stimuli*) are explicitly assigned to the circuit, propagated through it, and the outputs are compared to the expected values, or
- *formal verification* [1], [2], which considers the problem mathematically and proves that a circuit is correct with 100% certainty.

Obviously, formal verification provides the best solution with respect to quality. Corresponding methods are capable of efficiently traversing large parts of the search space, e.g., by applying clever implications during the proof. The corresponding techniques are, however, rather complex compared to their simulative counterparts and, particularly for larger designs, often fail due to the exponential complexity of the task.

Simulation is much easier to implement and very fast as long as only a limited number of stimuli is applied. The problem obviously is the quality provided by the applied set of stimuli. An exhaustive set of stimuli would show correctness with 100% certainty, but is practically intractable as this would eventually require an exponential number of stimuli to simulate. Accordingly, methods such as *constraint-based random simulation* [3]–[6], *fuzzing* [7], [8], etc. are key techniques to cope with this problem while still maintaining a high quality. Here, stimuli and/or data inputs are specifically generated (e.g., from constraints, mutations of randomly generated inputs, etc.) so that corner case scenarios and/or a broad variety of cases are triggered. In doing so, errors that might otherwise remain undetected are more likely to be found.

However, despite substantial progress that has been made in the past, e.g., on improving the efficiency of formal methods or on stimuli generation which increases the coverage of simulative verification, verifying classical circuits remains a challenge and, hence, is subject of further research.

2.2 Quantum Circuits

Quantum circuits promise more potential than classical circuits for many applications, but also require a more complex description. In contrast to classical bits, the main computational unit of quantum circuits (the *qubit*) cannot only be in one of the computational basis states $|0\rangle$ or $|1\rangle$, but also in a *superposition* of both. That is, the state $|\varphi\rangle$ of a qubit can be described as $|\varphi\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle$ with $\alpha_0, \alpha_1 \in \mathbb{C}$ and $|\alpha_0|^2 + |\alpha_1|^2 = 1$. More generally, the state of an n-qubit system is described by 2^n complex amplitudes α_i—each associated to a computational basis state $|i\rangle = |(i_{n-1} \dots i_0)_2\rangle = |i_{n-1}\rangle \otimes \dots \otimes |i_0\rangle$. It holds that $|\varphi\rangle = \sum_{i \in \{0,1\}^n} \alpha_i |i\rangle$ with $\alpha_i \in \mathbb{C}$ and $\sum_{i \in \{0,1\}^n} |\alpha_i|^2 = 1$. Typically, those states are expressed as 2^n-dimensional *state vectors* consisting of all amplitudes, i.e., $|\varphi\rangle \equiv [\alpha_0, \dots, \alpha_{2^n-1}]^\top$.

Example 1. *Consider the two-qubit quantum state $|\varphi\rangle$ described by $|\varphi\rangle = 1/\sqrt{2}\,|00\rangle + 0\,|01\rangle + 0\,|10\rangle + 1/\sqrt{2}\,|11\rangle$. This is a valid quantum state since $|1/\sqrt{2}|^2 + |1/\sqrt{2}|^2 = 1/2 + 1/2 = 1$. Its state vector representation is given by $[1/\sqrt{2}, 0, 0, 1/\sqrt{2}]^\top$. Notably, $|\varphi\rangle$ is an example of an* entangled *state where the state of one qubit inherently depends on the state of another qubit—a phenomenon unique to quantum computing.*

A *quantum circuit* manipulates the state of a quantum system. To this end, each *quantum gate* of a circuit realizes a certain quantum operation. Mathematically, these operations are represented by $2^n \times 2^n$-dimensional, unitary matrices[1] U acting on the 2^n-dimensional state vector $|\varphi\rangle \equiv [\alpha_0, \dots, \alpha_{2^n-1}]^\top$. Typically, quantum operations only act on $k < n$ qubits (predominantly $k = 1$ or $k = 2$) and, hence, are characterized by $2^k \times 2^k$-dimensional, unitary matrices which are extended to the full system size by tensor products with identity matrices.

Example 2. *Popular single-qubit gates include the Pauli gates X, Y, and Z, the Hadamard gate H, as well as the the phase gate S. The respective matrices are:*

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad H = 1/\sqrt{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}.$$

Most multi-qubit gates are controlled *gates, where a certain single-qubit gate is applied to a specified* target *qubit only if all designated* control *qubits are $|1\rangle$. One prominent example is the two-qubit controlled-NOT (CNOT), which is described by the matrix*

$$CNOT(q_c, q_t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

The overall quantum circuit G (realizing a quantum algorithm) is eventually represented as a sequence of quantum gates g_i, i.e., by $G = g_0, \dots, g_{m-1}$ with m being the total number of gates. The functionality of this circuit is described by the unitary matrix $U = U_{m-1} \cdot \dots \cdot U_0$, where U_i is the unitary matrix corresponding to gate g_i.

Example 3. *Consider the quantum circuit $G = g_0 g_1$ acting on two qubits (denoted q_0 and q_1) with $g_0 = H(q_1)$ (i.e., an H gate applied to q_1) and $g_1 = CNOT(q_1, q_0)$ (i.e., a CNOT gate with control qubit q_1 and target qubit q_0). Then, the respective matrices U_0, U_1, and the overall system matrix $U = U_1 \cdot U_0$ are given by*

$$U_0 = H \otimes \mathbb{I}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad U_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad U = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{bmatrix}.$$

For more details about quantum computing we refer to [9], [23].

2.3 Verification of Quantum Circuits

In the quantum realm, the verification problem can be stated in a similar fashion as for classical circuits: Given a circuit $G = g_0 \dots g_{m-1}$, it should be checked whether it adheres to an also given specification[2]. For the sake of this work and without loss of generality we assume in the following that the specification is given as a unitary function U—possibly described by a high-level quantum algorithm, another circuit, or further functional representations for quantum computing.

However, due to the more complex/expressive description, the formal verification of quantum circuits poses even more challenges than in the classical realm. Despite recent advances in the design of diverse/efficient formal verification methods [10]–[14], these can only escape the imminent complexity to a certain extent. Accordingly, simulative verification might provide a promising alternative as well. In fact, this has already been considered in theoretical quantum information, where (truly quantum-based) methods have been proposed (see, e.g., [23, Section 3] and [24]). But these approaches would require an execution on actual quantum computing devices, whose availability and accessibility still is severely restricted. Hence, before valuable quantum computing resources are wasted to verify a quantum circuit, efficient alternatives which can be employed prior to an actual execution on a quantum computer (using classical computing devices) are of high interest[3].

[1] A complex matrix U is unitary if $U^\dagger U = U U^\dagger = \mathbb{I}$, where $U^\dagger$ denotes the conjugate-transpose of U and $\mathbb{I}$ the identity matrix.

[2] Note that the terms *Device Under Verification* and *Golden Specification* are not established in the quantum realm (yet), which is why we simply use the terms *circuit* and *specification* in the following.

[3] This has similarities to the verification of classical circuits which also shall be conducted prior to an actual execution in the field.

This eventually results in the following simulative verification scheme for quantum circuits:

(1) Consider a set $\mathcal{S}$ of quantum states (which serve as stimuli).
(2) Pick (and prepare) a quantum state $|\varphi\rangle \in \mathcal{S}$.
(3) Simulate (on a classical device) both U and G with this initial state—resulting in two states $|\varphi_U\rangle$ and $|\varphi_G\rangle$, respectively.
(4) Compare the output $|\varphi_G\rangle$ generated by the realization G with the desired output $|\varphi_U\rangle$ by computing the quantum fidelity $\mathcal{F}$ between both states [9][4], i.e.,
$$\mathcal{F}(|\varphi_U\rangle, |\varphi_G\rangle) = |\langle\varphi_U|\varphi_G\rangle|^2 \in [0, 1].$$
(5) If $\mathcal{F}(|\varphi_U\rangle, |\varphi_G\rangle) \neq 1$, the stimulus $|\varphi\rangle$ shows the incorrect behavior of G with respect to U. Accordingly, the verification failed and the process is terminated.
(6) Remove $|\varphi\rangle$ from $\mathcal{S}$.
(7) If $|\mathcal{S}| \neq \emptyset$ (i.e., $\mathcal{S}$ is still non-empty) continue with Step (2); otherwise, the simulative verification process has been completed.

Now, the challenges of such an approach are as follows: First, in order to simulate a quantum circuit $G = g_0 \dots g_{m-1}$ starting with an initial state $|\varphi\rangle$ on a classical device (Step (3) from above), matrix-vector multiplications of the matrices U_i (representing the circuit's gates g_i) with the state vector $|\varphi\rangle$ as well as the resulting output vectors, respectively, have to be conducted consecutively.

Example 4. *Consider the circuit G from Example 3 and the initial state $|\varphi\rangle = |00\rangle \equiv [1, 0, 0, 0]^\top$. Applying the gate $g_0 = H(q_1)$ to this initial state, i.e., computing $U_0|\varphi\rangle$, produces a new state $|\varphi'\rangle = 1/\sqrt{2}|00\rangle + 1/\sqrt{2}|10\rangle \equiv [1/\sqrt{2}, 0, 1/\sqrt{2}, 0]^\top$. Afterwards, applying $g_1 = CNOT(q_1, q_0)$ to $|\varphi'\rangle$, i.e., computing $U_1|\varphi'\rangle$, results in the final state $|\varphi''\rangle = 1/\sqrt{2}|00\rangle + 1/\sqrt{2}|11\rangle \equiv [1/\sqrt{2}, 0, 0, 1/\sqrt{2}]^\top$—representing the output state generated by this circuit for stimulus/input $|\varphi\rangle$.*

This leads to an exponential complexity since the involved vectors and matrices have a size of 2^n and $2^n \times 2^n$, respectively (with n being the number of qubits). But although this is substantially harder than for the verification of classical circuits (here, a single simulation yields only linear complexity), rather powerful methods have been recently proposed to tackle this complexity—including methods based on highly optimized and parallel matrix-computations [15], [16], tensor networks [17], [18], quasiprobability/stabilizer-rank methods [19] (and references therein), as well as decision diagrams [20]–[22].

Second, as in the verification of classical circuits, the quality of the verification process heavily depends on the applied set of stimuli, i.e., 100% certainty cannot be guaranteed as long as the set of applied stimuli is not exhaustive. Moreover, while the stimuli space for classical circuits is finite (each input bit can be assigned either 0 or 1—yielding a total of 2^n possible stimuli), the state space in the quantum realm is infinitely large (possible stimuli are elements of a 2^n-dimensional Hilbert space). This raises the question on whether simulative verification of quantum circuits (on classical computers) is suitable at all and, if so, how to generate proper stimuli $|\varphi\rangle$ to efficiently check the correctness of a quantum circuit.

In the following, we show that, although the perspective of a possible infinite number of stimuli may seem rather grim at a first glance, there are promising ways to check the correctness of quantum circuits using simulative verification. These, however, severely depend on how the stimuli are actually generated. In fact, we show (both, theoretically and empirically) that high error detection rates can be achieved even if only a few randomly-chosen stimuli are considered—as long as these are generated in a specific fashion.

[4] In this regard the fidelity $\mathcal{F}$ acts as a similarity measure between two states—effectively computing the squared overlap of the states' amplitudes.

3 RANDOM STIMULI GENERATION

In this section, we propose different schemes for the generation of (random) stimuli and explore how well they can show the correctness of a quantum circuit. To this end, each of the following subsections introduces, illustrates, and (theoretically) analyzes different stimuli generation schemes and their likeliness of detecting errors. Eventually, this will show that simulative verification indeed is very promising since sets of stimuli can be generated in a fashion that offers a nice trade-off between error detection rate (as well as the required number of stimuli) and efficiency. The results of these conceptual and theoretical considerations have also been empirically confirmed as summarized later in Section 4.

3.1 Classical Stimuli

The most straight-forward application of simulative verification for quantum circuits (compared to the classical approach reviewed in Section 2.1) is to consider the set of computational basis states as stimuli (i.e., picking $|\varphi\rangle$ from the set $\{|i\rangle : i \in \{0, 1\}^n\}$) and computing $\mathcal{F}(U|i\rangle, V|i\rangle)$, where V is the matrix associated to G). This has recently been studied in [25], where empirical results show that choosing "classical" stimuli from this set at random *often* allows to detect even small errors in quantum circuits. The following example illustrates this remarkable "power of simulation".

Example 5. *Consider a certain n-qubit unitary specification U and assume that some error affects (w.l.o.g.) the first qubit in the actual realization G. In the quantum realm, this means that the circuit G is described by the unitary matrix $V = U \cdot (\mathbb{I}^{\otimes(n-1)} \otimes E)$, where E describes an error gate that is applied to the first qubit. Due to the inherent reversibility of quantum gates, this error has a localized effect on the output, i.e.,*
$$\mathcal{F}(U|c\rangle, V|c\rangle) = \mathcal{F}(|c\rangle, (\mathbb{I}^{\otimes(n-1)} \otimes E)|c\rangle) = |\langle c_0|E|c_0\rangle|^2$$
for any classical stimulus $|c\rangle = |c_{n-1} \dots c_0\rangle$.

Now suppose that $E = X$, i.e., a bit flip error occured. In contrast to classical intuition, such an error can be detected by a single *simulation with* any *classical stimulus $|c\rangle$, since $\mathcal{F}(U|c\rangle, V|c\rangle) = |\langle c_0|X|c_0\rangle|^2 = 0$ independent of $|c\rangle$.*

However, this approach has a severe handicap which has not been discussed so far—namely that it is not faithful. Specifically, for each unitary specification U there is an (infinitely large) family of realizations G for which $\mathcal{F}(U|c\rangle, V|c\rangle) = 1$ holds for all classical stimuli $|c\rangle$, even if quantum states $|\varphi\rangle$ with $\mathcal{F}(U|\varphi\rangle, V|\varphi\rangle) \neq 1$ actually exist. An example illustrates the problem:

Example 6. *Consider the same scenario as in Ex. 5, but assume that the error is characterized as $E = Z$, i.e., a phase flip error occurred. No classical stimulus $|c\rangle$ may detect such an error due to the fact that $\mathcal{F}(U|c\rangle, V|c\rangle) = |\langle c_0|Z|c_0\rangle|^2 = 1$ independent of $|c\rangle$. Intuitively, this happens whenever the "difference" of U and V is diagonal in the computational basis, such as $\mathbb{I}^{\otimes(n-1)} \otimes Z$ in case of this example.*

Nevertheless, our empirical evaluations (which are summarized later in Section 4) show that whenever classical stimuli are actually capable of detecting a certain error in the realization G, they do so within remarkably few simulations with randomly picked classical stimuli—an effect contradictory to classical intuition as already observed in [25].

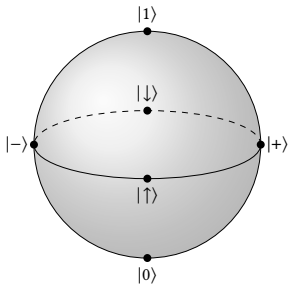

Figure 1: Bloch Sphere

3.2 Local Quantum Stimuli

In the previous section, we showed that classical stimuli generation is not sufficient to faithfully detect errors in quantum circuits. On an abstract level, this should not come as a surprise. After all, quantum circuits are designed to achieve tasks that classical circuits cannot. In fact, a closer look at the single-(qu)bit case already reveals a fundamental discrepancy: Classical single-bit operations map one of two possible inputs (0 or 1) to one of two possible outputs (0 or 1). In contrast, the quantum case is much more expressive: The set of all possible single-qubit states $|\varphi\rangle$ is infinitely large and can be parametrized by the 2-dimensional Bloch sphere [9] illustrated in Figure 1. Single-qubit quantum operations map single-qubit states to single-qubit states. Geometrically, this family encompasses all possible rotations of the Bloch sphere as well as all reflections. Classical (single-qubit) stimuli, i.e., the states $|0\rangle$ and $|1\rangle$, are not expressive enough to reliably probe such a continuum of operations. They correspond to antipodal points on the (Bloch) sphere and it is simply impossible to detect certain transformations by tracking the movement of only two antipodal points.

In order to address this, also stimuli beyond (classical) basis states should be considered. More precisely, three pairs of antipodal points are sufficient for full resolution [26]–[28], namely

$$\begin{aligned} &|0\rangle, && |1\rangle, && \text{(}Z\text{-basis)},\\ &|+\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle), && |-\rangle = 1/\sqrt{2}(|0\rangle - |1\rangle), && \text{(}X\text{-basis), and}\\ &|\uparrow\rangle = 1/\sqrt{2}(|0\rangle + i\,|1\rangle), && |\downarrow\rangle = 1/\sqrt{2}(|0\rangle - i\,|1\rangle), && \text{(}Y\text{-basis).} \end{aligned}$$

Generating stimuli uniformly at random from this sextuple[5] produces a set that is expressive enough to detect *any* single-qubit error. More precisely, for any pair of functionally different single-qubit unitaries U and V, at least one input $|l_1\rangle \in \{|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\uparrow\rangle, |\downarrow\rangle\}$ produces functionally different outputs, i.e., the fidelity $\mathcal{F}(U|l_1\rangle, V|l_1\rangle)$ is guaranteed to be $\neq 1$.

This desirable feature extends to the multi-qubit case. That is, if we independently select one of these six (single-qubit) states for every available qubit, every "local" single-qubit error may be detected. Thus, for n qubits, we consider the following ensemble of *local quantum stimuli*:

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

Example 7. *Let us revisit the scenario from Ex. 5 (and Ex. 6). Compared to classical stimuli, local quantum stimuli behave in a more homogeneous fashion on the classical extreme cases shown before: First, suppose that $E = X$ (bit flip error). Then,*

$$\mathcal{F}(U|l\rangle, V|l\rangle) = |\langle l_0|X|l_0\rangle|^2 = \begin{cases} 0 & |l_0\rangle \in \{|0\rangle, |1\rangle, |\uparrow\rangle, |\downarrow\rangle\} \\ 1 & |l_0\rangle \in \{|+\rangle, |-\rangle\} \end{cases}$$

Compared to classical stimuli, only 2/3 of all local quantum stimuli detect this type of error. Now, suppose that $E = Z$ (phase flip error). Then,

$$\mathcal{F}(U|l\rangle, V|l\rangle) = |\langle l_0|Z|l_0\rangle|^2 = \begin{cases} 0 & |l_0\rangle \in \{|+\rangle, |-\rangle, |\uparrow\rangle, |\downarrow\rangle\} \\ 1 & |l_0\rangle \in \{|0\rangle, |1\rangle\} \end{cases}$$

Consequently, in contrast to not detecting such an error with classical stimuli at all, again 2/3 of all local quantum stimuli are capable of detecting this type of error.

This observation that local quantum stimuli can detect errors which would have remained undetected using classical stimuli is not a coincidence. In fact, the collection of a total of 6^n local quantum stimuli is expressive enough to detect *any* error in a quantum circuit.

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

Proof sketch[6]. The key idea is to relate the expected fidelity $\mathbb{E}_{|l\rangle}\mathcal{F}(U|l\rangle, V|l\rangle)$—where the average is taken over all 6^n locally random stimuli—to a meaningful distance measure in the space of unitary matrices. This average outcome fidelity equals 1 if and only if U and V are functionally equivalent. Now, suppose that U and V are functionally distinct unitaries. Then, $\mathbb{E}_{|l\rangle}\mathcal{F}(U|l\rangle, V|l\rangle) < 1$ which is only possible if (at least) one stimulus $|l\rangle$ produces an outcome fidelity that is strictly smaller than one. □

While this rigorous statement asserts that any error can be detected by (at least) one local quantum stimulus, it does not provide any advice on how to find the "right" stimulus. This is a very challenging problem in general, but the above example suggests that repeated random sampling of stimuli should "do the job". Our empirical studies (see Section 4) confirm that such a procedure works remarkably well. Typically, few randomly generated local quantum stimuli suffice to detect realistic errors.

3.3 Global Quantum Stimuli

The previous section has shown that a modest increase in the expressiveness of stimuli can already make a large difference. Local quantum stimuli can detect *any* error, while classical stimuli cannot. This is interesting, because local quantum stimuli are comparatively few in number (6^n states in a 2^n-dimensional state space to detect arbitrary discrepancies in unitary circuits) and actually do not inherit many further quantum features. For example, "global" quantum features such as entanglement are not employed by them at all. This begs the question: what kind of advantages can even more expressive and "more quantum" stimuli offer? Faithfulness is not a problem anymore, but richer, global stimuli may help to detect errors *earlier*, i.e., after substantially fewer iterations.

In order to identify powerful global quantum stimuli, it is helpful to revisit local quantum stimuli as introduced in Eq. (1) from a different perspective: They are generated through starting with a very simple classical state (i.e., $|0\ldots0\rangle$) and applying certain single-qubit gates to the individual qubits, e.g., $|0\rangle\otimes|+\rangle\otimes|\uparrow\rangle = (\mathbb{I}\otimes H\otimes HS)\,|000\rangle$. Consequently, *random* local stimuli are generated by choosing this *layer* of single-qubit gates at random. This generation scheme can be readily generalized. Rather than selecting only a single layer of

[5] The single-qubit states $|0\rangle, |1\rangle, |+\rangle, |-\rangle, |\uparrow\rangle, |\downarrow\rangle$ can be generated from the basis state $|0\rangle$ by applying the gates $\mathbb{I}$, X, H, XH, HS, or XHS, respectively.

[6] Note that, due to page limitations, we only provide a sketch of the proof for this theorem.

(single-qubit) gates, we construct a generation circuit $G_0 \cdots G_{l-1}$ that has $l > 1$ layers and, most importantly, also features two-qubit gates. That is, a stimuli $|g\rangle$ with $|g\rangle = (G_0 \cdots G_{l-1})\,|0 \ldots 0\rangle$ is generated, where each G_i is a (single) layer comprised of so-called Clifford gates (H, S, $CNOT$) [29].

Overall, this set of *global quantum stimuli* $|g\rangle$ contains all local quantum stimuli, but is much richer and much more expressive. For instance, the overwhelming majority of global quantum stimuli will be highly entangled. Provided that the number of layers l is proportional to the number of qubits n [30], [31], these stimuli show remarkable properties. Most notably, 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, namely

$$\mathbb{E}_{|g\rangle}\mathcal{F}(U|g\rangle, V|g\rangle) \approx \mathcal{F}_{\text{avg}}(U,V) = \tfrac{1}{2^n+1}\left(1 + 2^n\left|\text{tr}(U^\dagger V)\right|^2\right). \quad (2)$$

Here, $\text{tr}(U^\dagger V)$ denotes the trace of the unitary matrix $U^\dagger V$. This *average (gate) fidelity* [9] forms the basis of many state-of-the-art quantum calibration procedures [32], [33]. Importantly, most (realistic) errors lead to an average fidelity that is tiny. Eq. (2) allows us to capitalize on this phenomenon. The following statement is an immediate consequence of Eq. (2) and Markov's inequality:

Corollary 1. *Consider a unitary specification U and a particular realization as a quantum circuit G (represented by the unitary V). Then, a randomly selected global quantum stimulus obeys*

$$\Pr_{|g\rangle}\left[\mathcal{F}(U|g\rangle, V|g\rangle) = 1\right] \leq \mathcal{F}_{\text{avg}}(U,V).$$

The r.h.s. equals 1 if and only if G correctly realizes U, otherwise it is typically much *smaller.*

This general statement does have powerful implications when applied to a precise example.

Example 8. *Consider again the scenario from Ex. 5 (and Ex. 6): A single-qubit error E occurred on the first qubit leading to the unitary $V = U \cdot (\mathbb{I}^{\otimes(n-1)} \otimes E)$, where the single-qubit error is either $E = X$ (bit flip error) or $E = Z$ (phase flip error). Then, $\mathcal{F}_{\text{avg}}(U,V) = \frac{1}{2^n+1} \leq 2^{-n}$ (because Pauli matrices are traceless) and Corollary 1 implies that it is very unlikely to* not *detect this error with a single, random global quantum stimulus, i.e., $\Pr_{|g\rangle}\left[\mathcal{F}(U|g\rangle, V|g\rangle) = 1\right] \leq 2^{-n} \ll 1$.*

This example demonstrates the power of global quantum stimuli. However, it is important to keep in mind that this power is not for free. The generation of (random) global quantum stimuli and subsequent simulation is much more resource-intensive by comparison (as confirmed by our empirical evaluations in Section 4).

This can also be understood from a broader context: The average (gate) fidelity as given by Eq. (2) is closely related to another popular distance measure—the *entanglement fidelity*. This quantity captures the performance of a powerful quantum stimulus $|\Omega\rangle$, see e.g. [24]. This stimulus is generated from $2n$ qubits by pairwise entangling individual qubits of one half of the system with the qubits of the other half. Applying both circuits to the first half of this state and computing the fidelity of the outcome states subsequently yields the entanglement fidelity [34], [35], i.e.,

$$\mathcal{F}(U \otimes \mathbb{I}|\Omega\rangle, V \otimes \mathbb{I}|\Omega\rangle) = 4^{-n}\left|\text{tr}(U^\dagger V)\right|^2 = \mathcal{F}_{\text{ent}}(U,V). \quad (3)$$

Comparing Eq. (2) and Eq. (3) shows that these quantities are almost identical. This implies that global quantum stimuli accurately approximate the powerful quantum stimulus $|\Omega\rangle$ on average. Finally, we point out that conducting simulative verification with $|\Omega\rangle$ itself is not feasible on classical computers, since requiring double the amount of qubits exponentially increases the resource-demand for classical simulations.

4 EMPIRICAL STUDY

In this section, we empirically study the behavior of the schemes proposed in Section 3 through extensive evaluations. To this end, the proposed schemes have been implemented in C++ as part of the open-source JKQ framework for quantum computing [36]. More precisely, they have been integrated into the JKQ QCEC quantum circuit equivalence checking tool (publicly available at https://github.com/iic-jku/qcec) using the decision diagram-based simulator from [21] for conducting the simulations. In order to obtain a rigorous evaluation, we considered the following setup:

- We chose 25 widely-used reversible/quantum algorithms with 16 to 34 qubits—constituting the respective reference implementations U.
- Each algorithm has been compiled to a suitable IBM architecture using IBM Qiskit [37]—constituting the realization G.
- In order to study the detection of errors, a total of 8 error-injection options have been considered for each circuit[7]:
  - Removal of 1, 2, or 3 random gates from G,
  - Insertion of 1, 2, or 3 random gates from the set $\{X, Y, Z, H, S, T\}$ on random qubits into G,
  - Insertion of 10 random Toffoli gates at the beginning or at the end of G.
- For each error-injection option, 50 random seeds have been considered.
- For each resulting instance, 5 random seeds have been used for randomly picking stimuli according to the respective scheme.
- For each resulting instance and random seed, up to 16 simulations of U and G with stimuli randomly picked according to the specific scheme have been performed aiming to detect the injected error.

Overall, this led to a total of 50 000 benchmark instances. Since for each instance on average approximately 3 random stimuli were necessary to detect the error, a grand total of approx. 10^6 simulations have been conducted. To the best of our knowledge, this led to the broadest empirical evaluation of simulative verification schemes for quantum circuits to date.

The obtained results are summarized in Table 1. Here, we list the error detection rate p_s in percent (i.e., the probability that the error is detected by the generated set of stimuli), the number of stimuli $\varnothing s$ needed to detect the error, and the runtime $\varnothing t$ of the respective scheme in seconds[8]. Due to page limitations, we only list the averaged values (w.r.t. the different error injections). However, since the obtained results are rather homogeneous across the respective benchmarks (as confirmed by the moderate standard deviation which is also listed in Table 1, this still allows for a proper interpretation of the results.

[7] In any realistic scenario where, e.g., a bug is present in the compilation flow, the resulting errors in G would be much more severe than the error-injections studied in this work. Consequently, it can be deducted from the results obtained in this work that the proposed schemes perform even more reliably on such instances.

[8] The runtime depends on the simulator used, as well as the hardware the simulations are conducted on. Nevertheless, it allows to reason about the efficiency of the individual schemes to some extent.

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 [%]	∅s	∅t [s]	p_s [%]	∅s	∅t [s]	p_s [%]	∅s	∅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 [%]	∅s	∅t [s]	p_s [%]	∅s	∅t [s]	p_s [%]	∅s	∅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 [%]	∅s	∅t [s]	p_s [%]	∅s	∅t [s]	p_s [%]	∅s	∅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 ∅s: Average number of stimuli ∅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.

From those results, the following conclusions can be drawn:

- *All* schemes lead to sets of stimuli with remarkable error detection rates. With randomly chosen stimuli only, few stimuli are sufficient to detect the vast majority of errors (while, in contrast, dedicated constrained-based stimuli generation, fuzzing, etc. methods [3]–[8] are required in the classical realm to get a merely acceptable error detection rate).
- Based on these high standards, classical stimuli generation performs worst and often fails—especially in cases where individual (diagonal) gates are removed or added. This is a consequence of classical stimuli not being faithful as shown in Section 3.1. At the same time, the corresponding simulations are very fast; making this scheme suitable for rapid prototyping.
- On the other side of the spectrum, global stimuli generation yields the most robust results, i.e., requiring the least amount of stimuli and also achieving the highest error detection rates. This confirms the discussions from Section 3.3 on the quality of those stimuli. Thus, this scheme is suitable for rigorous testing even if the simulation of those stimuli is severely more runtime-demanding.
- Local quantum stimuli generation constitutes a trade-off between quality and efficiency compared to the other two schemes. Although this scheme is not as powerful as global quantum stimuli generation with respect to quality, it is faithful (as shown in Section 3.2) and remains rather efficient.

5 CONCLUSION

In this work, we showed that simulative verification in the quantum realm is much more powerful than in the classical realm. On the one hand, we introduced, illustrated, and analyzed three potential quantum stimuli generation schemes offering a trade-off between error detection rate (as well as the required number of stimuli) and efficiency. On the other hand, we showed (both, theoretically and empirically) that, in contrast to classical circuits, high error detection rates can be achieved by just considering a few randomly-chosen stimuli (generated according to the proposed schemes). This eventually shows that simulative verification offers huge potential in the verification of quantum circuits.

ACKNOWLEDGMENTS

This work has partially been supported by the LIT Secure and Correct Systems Lab funded by the State of Upper Austria as well as by the BMK, BMDW, and the State of Upper Austria in the frame of the COMET program (managed by the FFG).

REFERENCES

[1] A. Biere and W. Kunz, "SAT and ATPG: Boolean engines for formal hardware verification," in *Int'l Conf. on CAD*, 2002, pp. 782–785.
[2] R. Drechsler, Ed., *Advanced Formal Verification*. Boston, MA: Springer, 2004.
[3] J. Yuan, C. Pixley, and A. Aziz, *Constraint-based verification*. New York, NY: Springer, 2006.
[4] J. Bergeron, *Writing Testbenches using System Verilog*. Boston, MA: Springer, 2006.
[5] N. Kitchen and A. Kuehlmann, "Stimulus generation for constrained random simulation," in *Int'l Conf. on CAD*, 2007, pp. 258–265.
[6] R. Wille *et al.*, "SMT-based stimuli generation in the SystemC verification library," in *Forum on Specification and Design Languages*, 2009, pp. 1–6.
[7] H. M. Le *et al.*, "Detection of hardware trojans in SystemC HLS designs via coverage-guided fuzzing," in *Design, Automation and Test in Europe*, 2019, pp. 602–605.
[8] K. Laeufer *et al.*, "RFUZZ: Coverage-directed fuzz testing of RTL on FPGAs," in *Int'l Conf. on CAD*, 2018.
[9] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2010.
[10] L. Burgholzer and R. Wille, "Advanced equivalence checking of quantum circuits," *IEEE Trans. on CAD of Integrated Circuits and Systems*, 2021.
[11] L. Burgholzer, R. Raymond, and R. Wille, "Verifying results of the IBM Qiskit quantum circuit compilation flow," in *Int'l Conf. on Quantum Computing and Engineering*, 2020.
[12] R. Duncan *et al.* (2019). "Graph-theoretic simplification of quantum circuits with the ZX-calculus." arXiv: 1902.03178.
[13] S. Yamashita and I. L. Markov, "Fast equivalence-checking for quantum circuits," in *Int'l Symp. on Nanoscale Architectures*, 2010.
[14] E. Ardeshir-Larijani, S. J. Gay, and R. Nagarajan, "Automated equivalence checking of concurrent quantum systems," *ACM Trans. Comput. Logic*, vol. 19, no. 4, pp. 1–32, 2018.
[15] G. G. Guerreschi *et al.*, "Intel Quantum Simulator: A cloud-ready high-performance simulator of quantum circuits," *Quantum Sci. Technol.*, vol. 5, p. 034 007, 2020.
[16] T. Jones *et al.*, "QuEST and high performance simulation of quantum computers," in *Scientific Reports*, 2018.
[17] B. Villalonga *et al.*, "A flexible high-performance simulator for verifying and benchmarking quantum circuits implemented on real hardware," *Npj Quantum Inf.*, vol. 5, no. 1, pp. 1–16, 2019.
[18] E. Pednault *et al.* (2019). "Leveraging secondary storage to simulate deep 54-qubit Sycamore circuits." arXiv: 1910.09534.
[19] J. R. Seddon *et al.* (2020). "Quantifying quantum speedups: Improved classical simulation from tighter magic monotones." arXiv: 2002.06181.
[20] P. Niemann *et al.*, "QMDDs: Efficient quantum function representation and manipulation," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 35, no. 1, pp. 86–99, 2016.
[21] A. Zulehner and R. Wille, "Advanced simulation of quantum computations," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 38, no. 5, pp. 848–859, 2019.
[22] A. Zulehner and R. Wille, "Matrix-vector vs. matrix-matrix multiplication: Potential in DD-based simulation of quantum computations," in *Design, Automation and Test in Europe*, 2019.
[23] J. Watrous, *The theory of quantum information*. Cambridge University Press, 2018, 590 pp.
[24] S. Khatri *et al.*, "Quantum-assisted quantum compiling," *Quantum*, vol. 3, p. 140, 2019.
[25] L. Burgholzer and R. Wille, "The power of simulation for equivalence checking in quantum computing," in *Design Automation Conf.*, 2020.
[26] J. Schwinger, "Unitary operator bases," *Proceedings of the National Academy of Sciences*, vol. 46, no. 4, pp. 570–579, 1960.
[27] A. Klappenecker and M. Rotteler, "Mutually unbiased bases are complex projective 2-designs," in *Int'l Symp. on Information Theory*, 2005, pp. 1740–1744.
[28] R. Kueng and D. Gross. (2015). "Qubit stabilizer states are complex projective 3-designs." arXiv: 1510.02767.
[29] D. Gottesman, "Stabilizer codes and quantum error correction.," Caltech, 1997.
[30] N. Hunter-Jones. (2019). "Unitary designs from statistical mechanics in random quantum circuits." arXiv: 1905.12053.
[31] F. G. S. L. Brandão, A. W. Harrow, and M. Horodecki, "Local random quantum circuits are approximate polynomial-designs," *Commun. Math. Phys.*, vol. 346, no. 2, pp. 397–434, 2016.
[32] E. Magesan, J. M. Gambetta, and J. Emerson, "Characterizing quantum gates via randomized benchmarking," *Phys. Rev. A*, vol. 85, no. 4, p. 042 311, 2012.
[33] R. Kueng *et al.*, "Comparing experiments to the fault-tolerance threshold," *Phys. Rev. Lett.*, vol. 117, no. 17, p. 170 502, 2016.
[34] B. Schumacher, "Sending entanglement through noisy quantum channels," *Phys. Rev. A*, vol. 54, no. 4, pp. 2614–2628, 1996.
[35] A. Y. Kitaev, "Quantum computations: Algorithms and error correction," *Russ. Math. Surv.*, vol. 52, no. 6, pp. 1191–1249, 1997.
[36] R. Wille, S. Hillmich, and L. Burgholzer, "JKQ: JKU tools for quantum computing," in *Int'l Conf. on CAD*, 2020.
[37] G. Aleksandrowicz *et al.*, "Qiskit: An open-source framework for quantum computing," 2019.