REDESIGN QUANTUM CIRCUITS ON QUANTUM HARDWARE DEVICES

Anonymous authors
Paper under double-blind review

ABSTRACT

Synthesising quantum circuits and checking their equivalence remains one of the core challenges in the implementation of quantum algorithms. Nevertheless, existing classical simulation based algorithms are constrained by exponential costs, rendering them infeasible for large-scale quantum circuits. In this paper, we propose an architecture that enables redesigning large-scale quantum circuits on quantum hardware, based on entangling quantum generative adversarial networks. Additionally, we introduce an efficient local random state encoder, which substantially enhances the practical applicability of the proposed architecture. The effectiveness of this method is theoretically proven. For concreteness, we apply this architecture to three critical tasks in circuit optimization: the equivalence checking of both parameterized and non-parameterized circuits, as well as the variational optimization of quantum circuits. The feasibility of our approach is demonstrated by excellent results from both classical simulations and noisy intermediate-scale quantum (NISQ) hardware implementations. We expect our work will facilitate the implementation and validation of advanced quantum algorithms.

1 INTRODUCTION

Quantum algorithms can provide super-polynomial or even exponential speedups compared to their classical counterparts in solving certain important problems (Nielsen & Chuang, 2010), such as the large integers factoring (Shor, 1994), unstructured database searching (Grover, 1996) and linear systems solving (Wang et al., 2020). In the initial stage of quantum algorithm design, researchers tend to focus on functional specification, without giving significant consideration to actual implementation. Hence the resultant draft circuits may contain some high-level logic operations, whose unitaries are known but whose implementations are not explicitly defined, such as the Oracles raised in the Grover's algorithm (Grover, 1996; Long, 2001). Therefore, these logic operations must be translated into devices' native gate library to respect hardware constraints—a process known as quantum logic circuit synthesis (Ge et al., 2024). In addition to manually performing it, which is a rather labor intensive and skill-demanding task in practice, some established techniques can also automatically offer workable results, such as unitary decomposition (Möttönen et al., 2004), genetic algorithm (Williams & Gray, 1998), evolutionary algorithm (Ding et al., 2006), deep reinforcement learning (Zhang et al., 2020; He et al., 2021), and variational quantum algorithm (Rakyta & Zimborás, 2022a;b).

Depending on their performance, various approaches might yield different results, resulting in the need to check the equivalence between their outputs. This is of great significance for ensuring the functional preservation of output results in the development of new compilers. A range of algorithms are available to achieve this verification: for instance, decision diagrams (Burgholzer & Wille, 2020) are applied to non-parameterized circuits, while the ZX calculus (Peham et al., 2023) is used for parameterized circuits. In ideal scenarios, these methods can accomplish the task with polynomial complexity. Unfortunately, their core innovations may fail under more general conditions, leaving classical simulation to become the last resort. Suffering from the inefficient classical simulation of quantum systems, this strategy scales very poorly with respect to the systems size, and turn impractical as the number of qubits gets large. Thus it is highly desirable to establish an algorithm that do not suffer performance degradation as the involved qubits increase.

These classically hard problems may be quantumly easy. Niu et al. (2022) propose the entangling quantum generative adversarial algorithm (EQ-GAN) to reproduce a given reference quantum state based on quantum hardware (Rasmussen & Zinner, 2022). It circumvents the classical simulation of quantum systems, and thereby can be applied to a large scale. However, this scheme cannot be directly applied to circuit equivalence checking. The reason lies in that it learns only the first column of the corresponding unitary matrices, whereas quantum circuit equivalence checking inherently requires their entire unitary matrices to be equivalent. In this paper, we take a step further—tackling the task of large-scale quantum circuit redesign on quantum hardware devices.

Definition 1. *We define quantum circuit redesign as the task of transforming a given quantum circuit into an alternative circuit architecture while preserving its fundamental functionality.*

Many important applications fall naturally into the domain of quantum circuit redesign, such as the equivalence checking of quantum synthesized circuits, quantum circuits optimization and gate decomposition. We refer to the proposed approach as the RH (Redesign circuits on quantum Hardware) architecture for simplicity. Our RH architecture extends the EQ-GAN and can be used directly to checking the equivalence between different circuits, whether they are parameterized or non-parameterized, at low cost. The effectiveness of RH has been theoretically proven. Furthermore, this approach remains valid for variational learning of reference circuit functionality by structurally distinct parameterized circuits, owing to its inherent compatibility with quantum machine learning frameworks (Benedetti et al., 2019). To validate the RH architecture, we exemplarily implement three applications both in classical simulation and quantum hardware. Collectively these applications highlight the utilization scope of the RH architecture.

Our work opens a venue for redesigning quantum circuits using quantum hardware, permitting a wide range of applications such as large-scale quantum circuit optimization, equivalence checking, etc. It is important to highlight that our RH architecture can also be incorporated into other techniques for further advanced applications. For instance, it can be used to replace the classical simulation as the last resort, which is often the computational bottleneck, in some algorithms such as ZX (Peham et al., 2023) and decision diagram (Burgholzer & Wille, 2020) for circuit simplification or optimization. We believe our RH architecture will facilitate the implementation and validation of advanced quantum algorithms.

2 Preliminaries

For readers' convenience, this section introduces some quantum computation concepts central to our work. For additional quantum computing fundamentals, we refer readers to Reference Nielsen & Chuang (2010).

2.1 Quantum State and Quantum Circuit

In quantum computation, the fundamental unit of information is the quantum bit (qubit) - the quantum counterpart of the classical bit. The qubit's basis states $|0\rangle = [1, 0]^T$ and $|1\rangle = [0, 1]^T$ correspond to the classical 0 and 1 states, respectively. The essential quantum mechanical distinction is that a qubit can exist in any superposition of these basis states, described by: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = [\alpha, \beta]^T$, where α and β are complex probability amplitudes obeying the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. When measured, the qubit's superposition state collapses randomly to either $|0\rangle$ or $|1\rangle$, yielding either outcome 0 (with probability $|\alpha|^2$) or outcome 1 (with probability $|\beta|^2$). In Dirac notation, the bra state $\langle\psi|$ is defined as the Hermitian conjugate of the ket state $|\psi\rangle$, i.e., $\langle\psi| = (|\psi\rangle)^\dagger = (\alpha^*, \beta^*)$. In an n-qubit system, the quantum state is spanned by a set of 2^n basis states. These basis states can be expressed as: $|x_1 x_2 \cdots x_n\rangle \equiv |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_n\rangle$, where $x_i \in \{0, 1\}$, and $\otimes$ denotes the tensor product.

The overlap between two quantum states can be quantified by the fidelity $F = |\langle\psi'|\psi\rangle|^2$, which ranges from 0 (orthogonal states) to 1 (equivalent states). If two quantum states differ only by a global phase factor, namely, $|\phi\rangle = \mathrm{e}^{i\theta}|\psi\rangle$ (where θ is a real phase angle), they are physically indistinguishable in all observable measurements and thus can be considered equivalent.

Quantum states are manipulated by quantum gates represented by unitary matrices U (where $U^\dagger U = I$). Any quantum logic operation can be realized by a circuit composed of arbitrary single-qubit rotation gates and at least one entangling two-qubit gate (e.g. CX or CZ).

2.2 SWAP TEST

In the SWAP test (Buhrman et al., 2001) the measurement probability p_1 of obtaining outcome 1 on the ancillary qubit reflects the difference between input states $|\phi\rangle$ and $|\psi\rangle$ through the relation:

$$p_1 = \frac{1}{2} - \frac{1}{2}|\langle\phi|\psi\rangle|^2 = \frac{1}{2}(1 - F). \tag{1}$$

We define the probability of test failure (also called inter-state infidelity in other contexts) as

$$p_{\text{failure}} = 2p_1 = 1 - F. \tag{2}$$

If the input states are equivalent, $p_{\text{failure}} = 0$; whereas for orthogonal states, p_{failure} reaches its maximum value of 1. By simplifying the quantum circuit of the SWAP test, we can derive its destructive variant (Garcia-Escartin & Chamorro-Posada, 2013). In the destructive SWAP test, the parameter p_1 from the standard SWAP test is redefined as the probability of obtaining even results of $\sum_i O_0^i \& O_1^i$ on measurement outcomes, with symbol & referring to the AND operation.

2.3 EQ-GAN

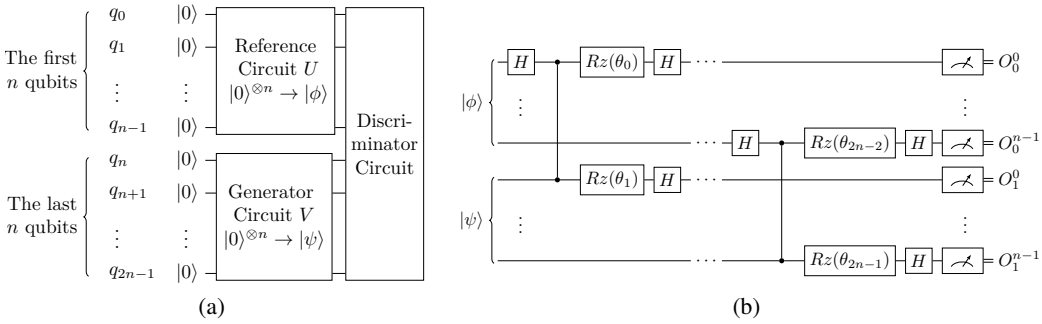

Figure 1: (a) The overall framework of the EQ-GAN algorithm. (b) The construction of discriminator in EQ-GAN.

The EQ-GAN algorithm (Figure 1 (a)) is designed to reproduce a target quantum state $|\phi\rangle$ prepared by a reference circuit U on a parameterized generator circuit V through variationally learning. A discriminator circuit evaluates the overlap between the target state $|\phi\rangle$ and the generator circuit's output state $|\psi\rangle$.

The destructive SWAP test can provide a perfect metric for states comparison in noise-free environments. Whereas in actual experiment, the two-qubit CZ gate on superconducting platform suffers from unstable coherent error due to miscalibrated gate parameters, which oscillate over the timescale of $O(10)$ minutes (Arute et al., 2019). To address this issue, a parameterized destructive SWAP test can be employed as the discriminator, whose circuit implementation is shown in Figure 1 (b). The coherent errors in CZ gates can be effectively compensated by subsequent single-qubit $R_z(\theta_i)$ gates with variationally optimized rotation angles θ_i (Quantum et al., 2020), achieving high-fidelity SWAP test implementation in noisy environments. The generator circuit is then trained to minimize the discriminator's test failure probability until $p_{\text{failure}} \to 0$, indicating that the generator's output state $|\psi\rangle$ becomes equivalent to the target state $|\phi\rangle$ prepared by the reference circuit.

3 PROPOSED METHOD

3.1 FRAMEWORK AND WORKFLOW

Our RH architecture is based on the EQ-GAN and primarily consists of five components, as shown in Figure 2 (a): two identical state encoder circuits, one reference circuit U, one generator circuit V and one discriminator circuit.

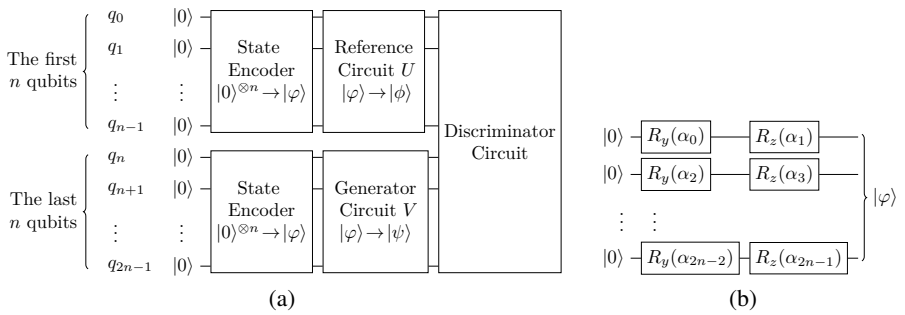

Figure 2: (a) The schematic diagram of the RH architecture. (b) The local random state encoder used in this paper.

The heart of quantum circuit redesign lies in reproducing the functionality of a given n-qubit reference circuit by a n-qubit generator circuit with a new structure. To accomplish this, RH architecture need a $2n$-qubit quantum system, where the reference circuit U is deployed in the first n qubits while the generator circuit V is located in the last n qubits. They transform the input states $|\varphi\rangle$ prepared by two identical state encoders into the states $|\phi\rangle$ and $|\psi\rangle$ respectively, and then feed them to the discriminator circuit. The discriminator circuit used here is the parameterized destructive SWAP test (Figure 1 (b)), which quantifies the overlap between its input states $|\phi\rangle$ and $|\psi\rangle$.

Our RH architecture begins by training the discriminator's parameters $\boldsymbol{\theta}$ with two identical input states, i.e., $|\phi\rangle = |\psi\rangle$, aiming at minimizing the cost function p_{failure}. When $p_{\text{failure}} = 0$ the discriminator could act as a perfect destructive SWAP test under the influence of coherent error. Considering that the unique global optimum of the discriminator is to utterly eliminate the coherent error (Niu et al., 2022) and any input states will lead to the same result, the zero state $|0\rangle^{\otimes n}$ is suitable for employment as the input states in training to reduce overhead. In other words, the state encoders, the reference circuit and the generator circuit can be ignored in this step.

Once a discriminator with well performance has been obtained after training, we can employ it to evaluate the overlap between the states generated by the reference and the generator circuits respectively. If the generator circuit is equivalent to the reference circuit, for any identical input $|\varphi\rangle$, their outputs should also be equivalent, and then pass the discriminator's test with $p_{\text{failure}} = 0$, i.e.,

$$p_{\text{failure}} = 1 - |\langle\phi|\psi\rangle|^2 = 1 - \left|\langle\varphi|U^\dagger V|\varphi\rangle\right|^2 = 1 - |\langle\varphi|\varphi\rangle|^2 = 0. \tag{3}$$

Otherwise, p_{failure} will also reflect the overlap between two circuits.

3.2 Efficient Local Random State Encoder

To accurately capture the difference between two circuits, it is theoretically necessary to iterate all basis states. However, even though encoding a basis state only requires a linear number of quantum gates, the exponential number of basis states hinders applications to large-scale scenarios. Here we devise a *local random state encoder*, which is composed of a series of single-qubit rotation gates as illustrated in Figure 2 (b) and generates quantum states that are the tensor product of n single-qubit random states from the initial state $|0\rangle^{\otimes n}$. We will prove in Theorem 1 below that, for several samplings of different local random states, if the tests always yield the conclusion of equivalence between two circuits, the probability of that they are non-equivalence is *statistically* zero.

Theorem 1. *Let U and V be two non-equivalent n-qubit unitary operators (i.e., $U \neq V$) and $|\varphi\rangle = |\varphi_1\rangle \otimes |\varphi_2\rangle \otimes \cdots \otimes |\varphi_n\rangle$ be a random n-qubit product state, where each $|\varphi_i\rangle$ is independently sampled from the Harr measure on $\mathbb{C}^2$, then Pr$[U|\varphi\rangle = V|\varphi\rangle] = 0$.*

Proof: Define the error operator $D = U^\dagger V$. Since $U \neq V$, we have $D \neq I$. The condition $U|\varphi\rangle = V|\varphi\rangle$ is equivalent to $D|\varphi\rangle = |\varphi\rangle$. Thus, we need to prove $\Pr[D|\varphi\rangle = |\varphi\rangle] = 0$.

The Fixed-Point Subspace $\text{Fix}(D) = \{|\phi\rangle \in \mathbb{C}^{2^n} \mid D|\phi\rangle = |\phi\rangle\}$ is the subspace of states invariant under D. Since $D \neq I$, $\text{Fix}(D)$ is a proper subspace of $\mathbb{C}^{2^n}$, with complex dimension $d < 2^n$(or real dimension $2d < 2^{n+1}$).

The set of product states $|\varphi_1\rangle \otimes |\varphi_2\rangle \otimes \cdots \otimes |\varphi_n\rangle$ forms a $2n$-dimensional real submanifold of $\mathbb{C}^{2^n}$ (each $|\varphi_i\rangle$ contributes 2 real dimensions). The Haar measure on product states is the product measure of single-qubit Haar measures, assigning positive measure to the submanifold.

We bound the real dimension of the intersection:

$$\dim_{\mathbb{R}}(\text{Fix}(D) \cap \text{Product States}) \leq 2d + 2n - 2^{n+1}.$$

Since $d \leq 2^n - 1$, this simplifies to $\dim_{\mathbb{R}}(\text{Fix}(D) \cap \text{Product States}) \leq 2n - 2$. This intersection has dimension $\leq 2n - 2$, which is a measure-zero set in the $2n$-dimensional product-state space. Thus, $\Pr[|\phi\rangle \in \text{Fix}(D)] = 0$. In conclusion, For any non-equivalent U and V, the probability that $U|\varphi\rangle = V|\varphi\rangle$ for a random product state $|\varphi\rangle$ is zero. □

3.3 Limitations and Effectiveness

It should be noted that as the number of qubits increases, the averaged probability amplitude of random quantum states decays exponentially across computational basis states. Consequently, in certain unfavorable scenarios the error detection probability becomes exponentially small. Assuming the error operator D includes only one non-trivial operation U_s, one of the worst-case scenarios occurs when this operation acts on the first qubit while being controlled by the remaining $n-1$ qubits. The resulting error operator takes the form $D = \mathbb{I}_2^{\oplus(n-1)} \oplus U_s$, where $\oplus$ denotes the direct sum operation of matrices. In this case, the probability that the error goes undetected within m measurement shots is about $e^{-m/2^{n-1}}$.

However, for most real-world cases, RH can detect the existence of errors within just a few thousands of measurement shots, as we will demonstrate in subsequent experiments. This makes it a valuable approach, especially when compared to classical simulation based methods: these methods often fail to generate reference results for large-scale circuits within a reasonable timeframe. This advantage is particularly pronounced for the task of checking the equivalence of outputs from different compilers. The error operator D between these outputs typically contains a limited number of erroneously generated single- or two-qubit gates, rather than gates with $n-1$ control qubits (the worst case mentioned above). Errors of this type are readily captured by RH. Furthermore, the RH can only probabilistically yield false negative conclusions (failing to detect actual differences between circuits) but is fundamentally impossible to produce false positives (where equivalent circuits are erroneously judged as non-equivalent), assuming ideal operations and perfect measurements in experiments.

4 Experiments

Many important applications are naturally categorized as quantum circuit redesign problems. For the sake of concreteness we first apply our RH architecture to three applications: non-parameterized (and parameterized) quantum circuit equivalence checking, and quantum circuit variational reconstruction. These applications capture the core of quantum circuit redesign and simultaneously are limited to a small-scale (4 qubit) to be performable by current quantum devices. To assess the performance of the RH architecture, we conduct these applications both in numerical simulation and in current NISQ hardware devices. The numerical simulation is performed with Python 3.10, MindSpore Quantum 0.10 (Xu et al., 2024) and have been run on an 4-core 1.8 GHz CPU with 8 GB memory. The hardware experiments are carried out on the Zuchongzhi-2 quantum processor, a 66-qubit superconducting quantum platform (Zhu et al., 2022). Detailed hardware experiment information, including error rates and qubit-mapping schematic, is provided in Appendix A. Larger-scale validation experiments ($4 \sim 9$ qubits) based purely on classical simulation are presented in subsection 4.5. The code and data used in this paper are publicly available on Gitee at `https://gitee.com/mindspore/mindquantum/tree/research/paper_with_code`.

4.1 Preliminary Experiment

Before implementing these applications, we first train the discriminator circuit—a 4-qubit parameterized destructive SWAP test circuit (Figure 1 (b)). For training the discriminator efficiently,

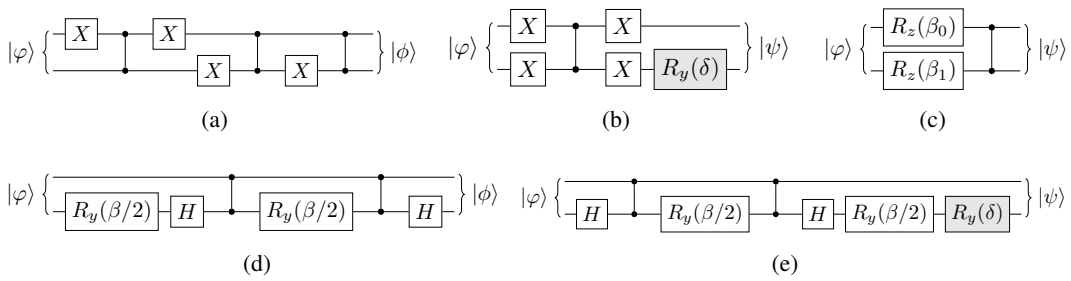

Figure 3: (a) A two-qubit phase-flipper circuit for basis states $|01\rangle$, $|10\rangle$ and $|11\rangle$. (b) A two-qubit phase-flipper circuit for basis state $|00\rangle$. (c) A potential alternative to the circuit (a) implemented by one two-qubit CZ gate and two single-qubit rotation gates. (d) and (e) are two decomposition circuits which are frequently employed for implementing the $CR_y(\theta)$ gate. The gray $R_y(\delta)$ gates in circuits (b) and (e) are introduced to create a controlled amount of distortion relative to their original functionality.

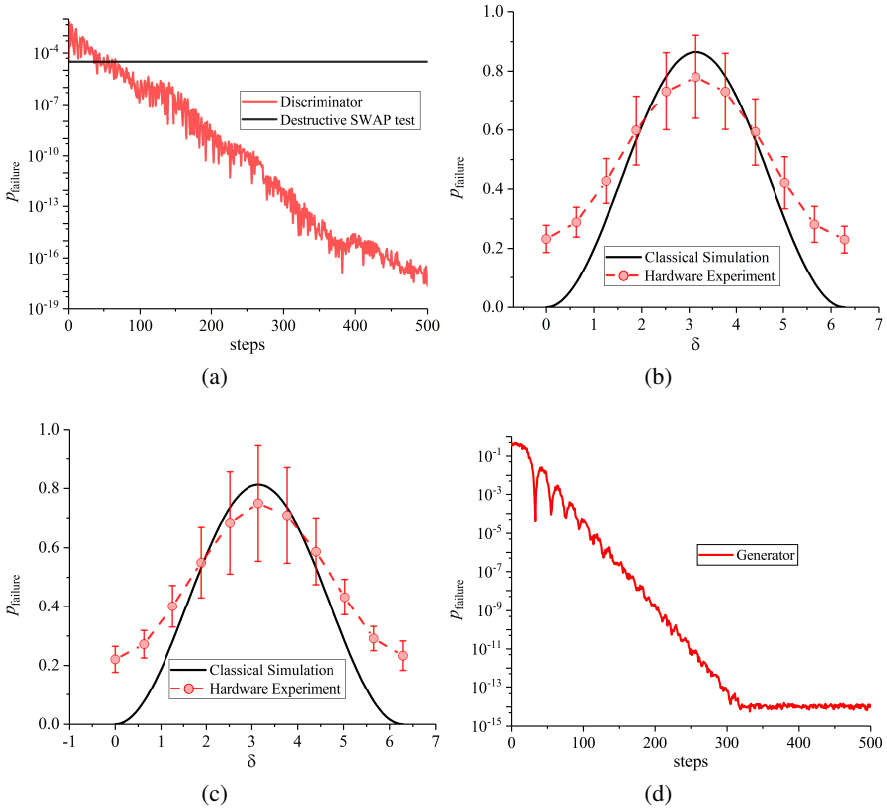

Figure 4: (a) The probabilities of test failure as functions of steps in training the discriminator classically. (b) The averaged probabilities of test failure (with ± 1 standard deviation error bars) vary with the value of parameter δ in the circuit of Figure 3 (b). (c) The averaged probabilities of test failure (with ± 1 standard deviation error bars) exhibit a parametric dependence on δ in the circuit of Figure 3 (e). (d) The averaged probability of test failure evolves with respect to the training steps in classical simulation.

we feed it two zero states, i.e. $|\psi\rangle = |\phi\rangle = |\varphi\rangle = |0\rangle^{\otimes 2}$ with reasons explained before, and optimize its parameters $\boldsymbol{\theta}$ to minimize the p_{failure}, allowing it to eventually act as an effective destructive SWAP

Table 1: The comparison of the discriminator's performance before and after training, along with the values of its parameters. The notation $\boldsymbol{\theta}$ refers to the results from classical simulation, while $\boldsymbol{\theta}^*$ refers to the results from hardware experiment.

p_{failure}	i	0	1	2	3
1.291×10^{-3}	$\boldsymbol{\theta}$	0.0	0.0	0.0	0.0
1.214×10^{-9}	$\boldsymbol{\theta}$	4.981×10^{-3}	-1.088×10^{-2}	6.910×10^{-3}	-8.094×10^{-3}
9.104×10^{-2}	$\boldsymbol{\theta}^*$	0.0	0.0	0.0	0.0
7.862×10^{-2}	$\boldsymbol{\theta}^*$	1.692×10^{-2}	1.618×10^{-2}	-1.114×10^{-2}	5.916×10^{-2}
—	error	-4.981×10^{-3}	1.088×10^{-2}	-6.910×10^{-3}	8.093×10^{-3}

test circuit in a noisy environment. The well trained discriminator obtained here will be reused in following applications.

To evaluate the discriminator's robustness against errors, we introduce two additional R_z gates (not shown in figure) after each CZ gate in the classical simulation, modeling the effect of coherent errors (Niu et al., 2022). The parameters of these R_z gates are randomly sampled from a normal distribution $\mathcal{N}(\mu = 0, \sigma = 0.02)$. The discriminator's trainable parameters $\boldsymbol{\theta}$ are initialized to zero and optimized using the Adam optimizer with an initial learning rate of 0.1. When all parameters $\boldsymbol{\theta} = \mathbf{0}$, the parameterized discriminator degenerated into the original destructive SWAP test.

In Figure 4 (a), we depict the evolution of the cost function p_{failure} with respect to the training steps in training the discriminator classically. We can see that within 500 training steps the probability of test failure gradually decreases, from the initial value of 3.148×10^{-4} to a minimum of 2.232×10^{-17}. In hardware experiment, the p_{failure} decreases from the initial 9.400×10^{-2} to a minimum of 5.150×10^{-2} within 50 training steps (without presenting in figure). The number of shots in experiment is 1000 throughout this paper.

In Table 1, we record the performance comparison of the discriminator before and after training, as well as the values of its parameters. Referring to Table 1, after training, not only the discriminator reduces the probability of test failure from 1.291×10^{-3} to 1.214×10^{-9} in simulation, but also the optimized parameters perfectly compensate for the randomly instantiated coherent error which is listed in the last row, i.e. $\theta_i = -\text{noise}_i$, achieving perfect error cancellation. These results indicate that the discriminator actually acquires remarkable error robustness by training. We emphasize here that during the training process, the p_{failure} is calculated for zero states, while in the final performance evaluation in Table 1, the p_{failure} is averaged over 100 random quantum states. Their specific values might vary, yet the main conclusion holds.

4.2 Application 1: Non-parameterized Quantum Circuit Equivalence Checking

The first application to be outlined is checking the equivalence between two non-parameterized circuits. We take the circuits shown in Figure 3 (a) and (b) as the reference circuit and the generator circuit in RH architecture respectively. Ignoring the $R_y(\delta)$ gate in gray, these two circuits are phase-flippers and have important applications in oracle-based quantum algorithms, such as the Grover's searching algorithm (Grover, 1996). When $\delta = 2k\pi$ (k is an integer), these two structurally distinct circuits are equivalent up to a global phase -1, and thus may generated by different synthesis tools for the same unitary matrix. The necessity of carrying out an equivalence checking on them naturally arises. To achieve this, we take 100 random states $|\varphi\rangle$ as inputs and study the averaged probability of test failure as the parameter δ of the R_y gate in gray alters.

From Figure 4 (b), we see that as δ increases within the interval $[0, 2\pi]$, the probability of test failure first rises from 0 and then falls back, which precisely conforms to the theoretical expectations. Similar conclusion can also be drawn according to the hardware results, albeit with deviations caused by noise and operational imperfections.

4.3 Application 2: Parameterized Circuit Equivalence Checking

Owing to the current prevalence of quantum machine learning, checking the equivalence of parameterized circuits is a highly important but simultaneously intractable problem, and there is a

lack of universal and effective solutions (Peham et al., 2023). In this subsection, as a representative example, we consider the equivalence checking between two decomposition circuits of the $CR_y(\beta)$ gate, which are shown in Figure 3 (d) and (e), respectively. These two parameterized circuits are equivalent for any real β when the parameter of the $R_y(\delta)$ gate in gray satisfies $\delta = 2k\pi$, where k is an integer, and deviates from each other in other cases.

Figure 4 (c) describes the averaged p_{failure} in classical simulation and hardware experiment over 10 random states and 10 random β in terms of the value of δ in the interval $[0, 2\pi]$. The feasibility of checking equivalence between parameterized circuits with random free parameters, i.e., the β herein, has been proven in Reference Peham et al. (2023). It can be concluded from Figure 4 (c) that the probabilities of test failure nicely reflect the amount of deviation between the two circuits as the δ varies.

4.4 Application 3: Quantum Circuit Variational Optimization

Circuit optimization seeks to construct equivalent quantum circuits with either reduced gate counts or shallower depths compared to their target circuits. For complex circuit optimization, variational algorithms often provide more reliable and systematic approaches than developer intuition alone (He et al., 2023).

The example we present here is to reduce the depth of a particular circuit illustrated in Figure 3 (a) in a variational manner. A potential alternate is depicted in Figure 3 (c), which comprises two single-qubit R_z gates with trainable parameters β_0 and β_1, as well as one CZ gate. Within the RH architecture, we designate these two circuits as the reference circuit and the generator circuit respectively. In this example, the settings of the hyper-parameters are aligned with those in the previous training of the discriminator, such as the parameters initialization, learning rate, cost function and so on.

Figure 4 (d) reveals that as the training steps increasing, the p_{failure} in classical simulation steadily decreases from an initial value of 6.654×10^{-1} and converges to around 1×10^{-14} after about 300 steps. The results from hardware experiment show that the probability of test failure has declined from the initial value of 8.66×10^{-1} to a minimum of 1.099×10^{-1} during a 50-step training procedure (without presenting in figure). The p_{failure} during training is dynamically computed using a batch of 4 random states. These observations imply that the RH architecture works well in this application, and the resultant generator circuit can be a competent alternative to the reference circuit in experiment after training.

4.5 Larger-scale Experiments

In previous subsections, we demonstrated the workflow of our RH architecture through three small-scale proof-of-concept experiments. We now consider larger-scale scenarios. Since quantum circuit equivalence verification constitutes the core contribution of the RH and serves as the foundation for all its applications, this section focuses on benchmarking its performance for various commonly used quantum circuits. The benchmark circuits are selected from the VeriQBench dataset (Chen et al., 2022), covering major quantum computing applications, such as the quantum machine learning, variational quantum computational chemistry, Grover's algorithm, quantum fourier transform, qubit mapping and quantum volume tests, etc. Given current quantum hardware's noise limitations for deep multi-qubit circuits, we restrict our benchmarking to only classical simulations. The tested circuits are limited to $n = 4 \sim 9$ qubits (RH architecture: $2n = 8 \sim 18$ qubits) to comply with the computational constraints of classical simulators. We stress that the RH architecture is theoretically scalable to arbitrary circuit sizes. With anticipated advancements in quantum hardware, particularly fault-tolerant quantum computers, verification of circuits comprising hundreds to thousands of qubits will become experimentally feasible.

To evaluate RH's performance, we prepare generator circuits by inserting quantum gates at random positions in given reference circuits, thereby simulating errors commonly encountered in synthesis algorithms. We generate equivalent circuit by inserting an identity gate into the given reference circuit, while creating non-equivalent circuit through the insertion of $R_x(1.23)$ rotation gate. As for the parameterized circuits, such as the 'qaoa_6' and 'qcnn_4', we randomly instantiate their parameters, as justified by Reference Peham et al. (2023). The benchmark results are summarized in

Table 2, where $p_{\text{failure_Y}}$ denotes the test failure probability for equivalent circuit pairs and $p_{\text{failure_N}}$ represents the failure probability for non-equivalent pairs under the RH architecture, respectively. As evidenced by the benchmark data in Table 2, the test failure probability $p_{\text{failure_Y}}$ remains zero within machine precision ($\sim 10^{-31}$) for equivalent circuit pairs. In contrast, for non-equivalent pairs, the test failure probability $p_{\text{failure_N}}$ ranges from 0.155 to 0.329, implying the non-equivalence between these circuits can be readily detected by the RH.

Table 2: Benchmark results for quantum circuit equivalence checking using VeriQBench dataset (Chen et al., 2022).

Benchmark	n	#gates	#depth	$p_{\text{failure_Y}}$	$p_{\text{failure_N}}$	$t\ [s]$
2of5d3	6	137	65	3.620×10^{-31}	0.307	0.466
4b15g_5	4	15	12	2.169×10^{-32}	0.252	0.073
4_49_fc	4	143	72	3.267×10^{-31}	0.279	0.158
adder_n7	7	8	5	1.531×10^{-31}	0.328	1.658
bv_9	9	27	8	3.777×10^{-31}	0.250	6.587
eff_5_4	4	19	3	1.025×10^{-31}	0.243	0.146
grover_9	9	41	12	3.999×10^{-31}	0.167	7.702
ham7tc	7	143	71	4.279×10^{-31}	0.219	7.520
hf_6_0_9	6	155	32	6.389×10^{-31}	0.155	0.805
hwb4_11_21	4	11	10	7.354×10^{-32}	0.312	0.072
mod5d1	5	8	8	9.571×10^{-32}	0.303	0.116
nth_prime4_inc_d1	4	72	38	1.696×10^{-31}	0.221	0.126
nth_prime7_inc_1427_3172	7	2513	1566	3.358×10^{-30}	0.329	111.947
pe_8	9	52	21	4.324×10^{-31}	0.160	11.118
qaoa_6	6	110	9	4.668×10^{-31}	0.260	0.830
qcnn_4	4	66	16	4.056×10^{-31}	0.208	0.264
qft_9	9	45	15	3.691×10^{-31}	0.183	11.671
rand_cliff_7_AG	7	83	46	2.729×10^{-31}	0.313	5.268
rand_cliff_9_AG	9	107	61	3.820×10^{-31}	0.326	22.744
rd32	4	4	4	7.213×10^{-32}	0.252	0.084
20QBT_4CYC_8GN_1.0P2_0	5	8	3	1.150×10^{-31}	0.250	0.115
54QBT_4CYC_8GN_1.0P2_0	8	8	1	1.856×10^{-31}	0.250	1.983
quantum_volume_n9_d5_i0	9	200	15	1.716×10^{-30}	0.155	43.420

n: Number of qubits #gates: Number of gates #depth: Circuit depth
$p_{\text{failure_Y}}$: p_{failure} for equivalent circuits $p_{\text{failure_N}}$: p_{failure} for non-equivalent circuits
t: The total runtime for 100 parameter instances..

5 Conclusion

In this paper, we formally defined the task of quantum circuit redesign, and introduced a novel RH architecture for redesigning large-size quantum circuits on quantum hardware that overcomes the classical simulation bottleneck inherent to existing methods. The RH architecture is built upon the EQ-GAN, and extends its capability from quantum states learning to unitary transformation learning by inserting a state encoder module prior to it. We proposed the usage of $|0\rangle^n$ state in training the discriminator to minimize resource overhead and error accumulation, and demonstrated the discriminator's robustness against coherent errors of CZ gate on superconducting quantum processors. An efficient local random state encoder was developed that drastically slashes implementation overhead, boosting practical utility. We proved the RH architecture's effectiveness and discussed its limitations. As demonstrations, we implemented the RH architecture to three critical applications: (non) parameterized circuit equivalence checking and variational reconstruction of quantum circuits. Evaluation results from both classical simulation and quantum hardware experiments confirm the superior performance of the RH architecture.

The RH architecture is also compatible with other existing circuit optimization techniques like ZX-calculus, enabling collaborative solutions for advanced circuit optimization challenges. We believe our RH architecture could help boost innovation in designing advanced quantum circuits.

REFERENCES

Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

Michel Barbeau and Joaquin Garcia-Alfaro. Faking and discriminating the navigation data of a micro aerial vehicle using quantum generative adversarial networks. In *2019 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6. IEEE, 2019.

Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, 2019.

Bhupesh Bishnoi. Quantum computation. *arXiv preprint arXiv:2006.02799*, 2020. URL `https://arxiv.org/abs/2006.02799`.

Harry Buhrman, Richard Cleve, John Watrous, and Ronald De Wolf. Quantum fingerprinting. *Physical review letters*, 87(16):167902, 2001.

Lukas Burgholzer and Robert Wille. Advanced equivalence checking for quantum circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 40(9):1810–1824, 2020.

Kean Chen, Wang Fang, Ji Guan, Xin Hong, Mingyu Huang, Junyi Liu, Qisheng Wang, and Mingsheng Ying. Veriqbench: A benchmark for multiple types of quantum circuits, 2022. URL `https://arxiv.org/abs/2206.10880`.

Christopher M Dawson and Michael A Nielsen. The solovay-kitaev algorithm. *arXiv preprint quant-ph/0505030*, 2005.

Shengchao Ding, Zhi Jin, and Qing Yang. Evolving quantum oracles with hybrid quantum-inspired evolutionary algorithm, 2006. URL `https://arxiv.org/abs/quant-ph/0610105`.

Juan Carlos Garcia-Escartin and Pedro Chamorro-Posada. Swap test and hong-ou-mandel effect are equivalent. *Physical Review A*, 87(5):052330, 2013.

Yan Ge, Wu Wenjie, Chen Yuheng, Pan Kaisen, Lu Xudong, Zhou Zixiang, Wang Yuhan, Wang Ruocheng, and Yan Junchi. Quantum circuit synthesis and compilation optimization: Overview and prospects. *arXiv preprint arXiv:2407.00736*, 2024.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, USA, 2016. ISBN 0262035618.

Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 212–219, 1996.

Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical review letters*, 103(15):150502, 2009.

Run-Hong He, Rui Wang, Jing Wu, Shen-Shuang Nie, Jia-Hui Zhang, and Zhao-Ming Wang. Deep reinforcement learning for universal quantum state preparation via dynamic pulse control, 2021.

Run-Hong He, Xu-Sheng Xu, Mark S Byrd, and Zhao-Ming Wang. Modularized and scalable compilation for double quantum dot quantum computing. *Quantum Science and Technology*, 9 (1):015004, 2023.

Ling Hu, Shu-Hao Wu, Weizhou Cai, Yuwei Ma, Xianghao Mu, Yuan Xu, Haiyan Wang, Yipu Song, Dong-Ling Deng, Chang-Ling Zou, et al. Quantum generative adversarial learning in a superconducting quantum circuit. *Science advances*, 5(1):eaav2761, 2019.

Gui-Lu Long. Grover algorithm with zero theoretical failure rate. *Physical Review A*, 64(2):022307, 2001.

Mikko Möttönen, Juha J Vartiainen, Ville Bergholm, and Martti M Salomaa. Quantum circuits for general multiqubit gates. *Physical review letters*, 93(13):130502, 2004.

Michael A Nielsen and Isaac L Chuang. *Quantum Computation and Quantum Information 10th Anniversary Edition*. Cambridge University Press, 2010.

Murphy Yuezhen Niu, Alexander Zlokapa, Michael Broughton, Sergio Boixo, Masoud Mohseni, Vadim Smelyanskyi, and Hartmut Neven. Entangling quantum generative adversarial networks. *Physical Review Letters*, 128(22):220505, 2022.

Tom Peham, Lukas Burgholzer, and Robert Wille. Equivalence checking of parameterized quantum circuits: Verifying the compilation of variational quantum algorithms. In *Proceedings of the 28th Asia and South Pacific Design Automation Conference*, pp. 702–708, 2023.

Google AI Quantum, Collaborators*, Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Sergio Boixo, Michael Broughton, Bob B Buckley, et al. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369(6507):1084–1089, 2020.

Péter Rakyta and Zoltán Zimborás. Approaching the theoretical limit in quantum gate decomposition. *Quantum*, 6:710, 2022a.

Péter Rakyta and Zoltán Zimborás. Efficient quantum gate decomposition via adaptive circuit compression. *arXiv preprint arXiv:2203.04426*, 2022b.

SE Rasmussen and NT Zinner. Multiqubit state learning with entangling quantum generative adversarial networks. *Physical Review A*, 106(3):032429, 2022.

Vivek V Shende, Stephen S Bullock, and Igor L Markov. Synthesis of quantum logic circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, pp. 272–275, 2005.

Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pp. 124–134. Ieee, 1994.

Shengbin Wang, Zhimin Wang, Wendong Li, Lixin Fan, Zhiqiang Wei, and Yongjian Gu. Quantum fast poisson solver: the algorithm and complete and modular circuit design. *Quantum Information Processing*, 19:1–25, 2020.

Colin P Williams and Alexander G Gray. Automated design of quantum circuits. In *NASA International Conference on Quantum Computing and Quantum Communications*, pp. 113–125. Springer, 1998.

Xusheng Xu, Jiangyu Cui, Zidong Cui, Runhong He, Qingyu Li, Xiaowei Li, Yanling Lin, Jiale Liu, Wuxin Liu, Jiale Lu, Maolin Luo, Chufan Lyu, Shijie Pan, Mosharev Pavel, Runqiu Shu, Jialiang Tang, Ruoqian Xu, Shu Xu, Kang Yang, Fan Yu, Qingguo Zeng, Haiying Zhao, Qiang Zheng, Junyuan Zhou, Xu Zhou, Yikang Zhu, Zuoheng Zou, Abolfazl Bayat, Xi Cao, Wei Cui, Zhendong Li, Guilu Long, Zhaofeng Su, Xiaoting Wang, Zizhu Wang, Shijie Wei, Re-Bing Wu, Pan Zhang, and Man-Hong Yung. Mindspore quantum: A user-friendly, high-performance, and ai-compatible quantum computing framework, 2024. URL `https://arxiv.org/abs/2406.17248`.

Yuan-Hang Zhang, Pei-Lin Zheng, Yi Zhang, and Dong-Ling Deng. Topological quantum compiling with reinforcement learning. *Physical Review Letters*, 125(17):170501, 2020.

Qingling Zhu, Sirui Cao, Fusheng Chen, Ming-Cheng Chen, Xiawei Chen, Tung-Hsun Chung, Hui Deng, Yajie Du, Daojin Fan, Ming Gong, Cheng Guo, Chu Guo, Shaojun Guo, Lianchen Han, Linyin Hong, He-Liang Huang, Yong-Heng Huo, Liping Li, Na Li, Shaowei Li, Yuan Li, Futian Liang, Chun Lin, Jin Lin, Haoran Qian, Dan Qiao, Hao Rong, Hong Su, Lihua Sun, Liangyuan Wang, Shiyu Wang, Dachao Wu, Yulin Wu, Yu Xu, Kai Yan, Weifeng Yang, Yang Yang, Yangsen Ye, Jianghan Yin, Chong Ying, Jiale Yu, Chen Zha, Cha Zhang, Haibin Zhang, Kaili Zhang, Yiming Zhang, Han Zhao, Youwei Zhao, Liang Zhou, Chao-Yang Lu, Cheng-Zhi Peng, Xiaobo Zhu, and Jian-Wei Pan. Quantum computational advantage via 60-qubit 24-cycle random circuit sampling. *Science Bulletin*, 67(3):240–245, 2022.

A HARDWARE INFORMATION

To access results from quantum hardware, we map the utilized 4 logical qubits q_i onto physical qubits Q_i in a 2×2 sub-grid of the 66-qubit Zuchongzhi-2 quantum processor for accommodating the experimental realities. The mapping diagram is illustrated in Figure 5, where the label of line between physical qubits indicates the error rate in executing CZ gate. The error rates of single-qubit native gates are between 0.11% and 0.21%, and the average readout error rate is 4.21%.

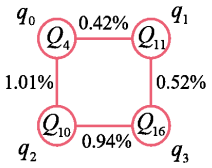

Figure 5: The mapping schematic between the logical qubits and the physical qubits arranged in a 2×2 sub-grid within the Zuchongzhi-2 quantum processor.