QNBAD: Quantum Noise-induced Backdoor Attacks against Zero Noise Extrapolation

Cheng Chu*, Qian Lou†, Fan Chen*, Lei Jiang*
*Indiana University Bloomington †University of Central Florida
*{chu6, fc7, jiang60}@iu.edu, †qian.lou@ucf.edu

Abstract—**Variational quantum algorithms (VQAs) have emerged as one of the most promising paradigms for achieving practical quantum advantage in the noisy intermediate-scale quantum (NISQ) era. To enhance the computational accuracy of VQAs on noisy hardware, zero noise extrapolation (ZNE) has become a widely adopted and effective error mitigation technique. However, the growing reliance on ZNE also increases the importance of identifying potential adversarial exploits. We examine existing backdoor attacks and highlight why they struggle to compromise ZNE. Specifically, quantum backdoor attacks that modify circuit structures merely shift the ideal output without affecting the noise-dependent extrapolation process, leaving ZNE intact. Likewise, parameter-level backdoors that are trained without accounting for device-specific noise exhibit inconsistent behavior across different hardware platforms, resulting in unreliable or ineffective attacks. Building on these observations, we uncover a new class of backdoor vulnerabilities that specifically target the unique properties of ZNE.**

In this study, we propose QNBAD, a novel and stealthy backdoor attack targeting ZNE. QNBAD is carefully designed to preserve the correct functionality of variational quantum circuits on most devices. However, under a specific noise model, it leverages subtle interactions between quantum noise and circuit structure to systematically manipulate the sampled expectation values across different noise levels. This targeted perturbation corrupts the ZNE fitting process and leads to significantly biased final estimates. Compared to prior backdoor methods, QNBAD achieves substantially greater absolute error amplification, ranging from 1.68× to 11.7× across four platforms and six applications. Furthermore, it remains effective across a variety of fitting functions and ZNE variants.

I. Introduction

Variational quantum algorithms (VQAs)[1] have been suggested as the focal point for the near-term application of quantum devices, due to its flexible c ircuit s tructure, shorter circuit depth and fewer gate operations, which allow quantum programs to be completed within a limited quantum decoherence time[2]. In typical applications, these algorithms begin by preparing a simple initial quantum state. A low-depth parameterized quantum circuit is then applied to this state to generate a variational quantum state. The expectation value of a target observable is subsequently measured and used as the objective function for optimization. These variational methods have been widely adopted in a range of domains, including quantum chemistry[3], [4], [5], combinatorial optimization[6], [7], and quantum machine learning[8], [9], [10], [11].

Parameter transfer is a practical and effective strategy for addressing the training challenges of VQAs, particularly the issue of barren plateaus [12], [13], [14]. This technique leverages optimized parameters from previously solved problem instances to initialize new circuits, often eliminating the need for extensive optimization. By exploiting structural similarities between related problems, parameter transfer accelerates convergence and improves training efficiency. It has demonstrated strong performance in applications such as molecular energy estimation with the VQE and combinatorial optimization with the QAOA [15], [16], [17], [18], [19], [20]. In practice, models trained via parameter transfer are frequently deployed on NISQ devices. However, the presence of hardware noise can degrade the accuracy of such models by directly impacting the fidelity of measured observables. Therefore, reducing the impact of noise remains a key requirement for reliable quantum computation.

Zero-Noise Extrapolation (ZNE) [2], [21], [22] is a well-established error mitigation technique designed to counteract the effects of quantum noise in NISQ devices. ZNE offers a practical approach to improving computational accuracy without relying on full quantum error correction. The fundamental idea of ZNE is to artificially increase the noise in a quantum circuit and then use extrapolation methods to estimate the result in a zero-noise regime. As illustrated in Figure 1, this process involves executing a compiled quantum circuit multiple times at different noise levels to collect a dataset that captures the system's behavior under varying noise factors. Once the data is obtained, classical fitting methods are applied to estimate the zero-noise expectation value. Compared to raw noisy outputs, the extrapolated results obtained via ZNE significantly reduce the absolute error, thus improving the reliability of quantum computing. ZNE has been implemented in several widely used quantum software frameworks, including Qiskit [23], Mitiq [24], and PennyLane [25].

Given that we are currently in the NISQ era, ZNE is expected to play a critical role in enhancing the reliability of VQAs across a wide range of application domains. Many of these applications are security- and safety-sensitive, including drug discovery[26], portfolio optimization[27], [28], and molecular energy state estimation[29], [30], [31]. In practice,

Network and Distributed System Security (NDSS) Symposium 2026
23-27 February 2026, San Diego, CA, USA
ISBN 979-8-9919276-8-0
https://dx.doi.org/10.14722/ndss.2026.241665
www.ndss-symposium.org

users tend to favor noise-mitigated results over raw noisy outputs due to their improved accuracy. However, any compromise in the reliability of ZNE outputs can directly affect the integrity of these high-stakes applications, potentially leading to serious consequences. For example, in drug discovery, inaccurate quantum simulations may produce false-positive candidates, resulting in wasted resources and potential risks to patient safety[32], [33]. In modeling energetic materials, VQE combined with ZNE must estimate activation energies with high precision. Small errors of 1-2 kcal/mol can underestimate impact sensitivity, causing hazardous compounds to be misclassified and pass safety checks, potentially leading to catastrophic failure [34]. These examples underscore the importance of ensuring the robustness and security of ZNE to safeguard trust in near-term quantum computing.

Among the various security concerns in quantum computing[35], [36], [37], [38], [39], the threat of backdoor attacks on ZNE is particularly salient. This vulnerability arises from a fundamental property of ZNE: it estimates the zero-noise output by extrapolating results obtained under varying noise levels. Such a design inherently broadens the adversarial attack surface because an adversary can manipulate any subset of the sampled outputs, and even minor perturbations at a single noise level can lead to significant deviations in the final extrapolated result. However, existing quantum backdoor strategies face two critical limitations when attempting to attack ZNE. The first limitation concerns circuit-level backdoors[40], [41], [42], which introduce malicious behavior by modifying the circuit ansatz. These structural changes are typically detectable through inspection and primarily affect the noiseless output, leaving the noise amplification and extrapolation stages of ZNE unaffected. As a result, noise-induced errors are still successfully mitigated, neutralizing the intended attack effect. The second limitation arises in parameter-level backdoors[43], [44], which embed malicious behavior via variational parameters. These methods fail to account for the interaction between quantum noise and the trigger mechanism. Since ZNE relies on sampling outputs under varying noise levels, the lack of noise robustness prevents reliable activation of such backdoors, significantly weakening their impact.

In this paper, we propose a novel noise-induced backdoor attack framework, QNBAD, which strategically exploits the ZNE technique to achieve malicious objectives. As shown in Figure 1, under noise-free conditions, VQAs embedded with QNBAD behave indistinguishably from normal VQAs, ensuring stealthiness during standard operation. However, when ZNE is applied to mitigate quantum noise, QNBAD is activated, exhibiting malicious behavior under varying noise levels. This activation disrupts the noise amplification and extrapolation mechanisms inherent to ZNE, leading to significant interference with its noise-mitigation process. Consequently, QNBAD amplifies the absolute error in the extrapolated output of ZNE, thereby undermining its reliability and accuracy. The noise-dependent nature of QNBAD ensures its adaptability to NISQ devices, making it a potent and versatile attack vector.

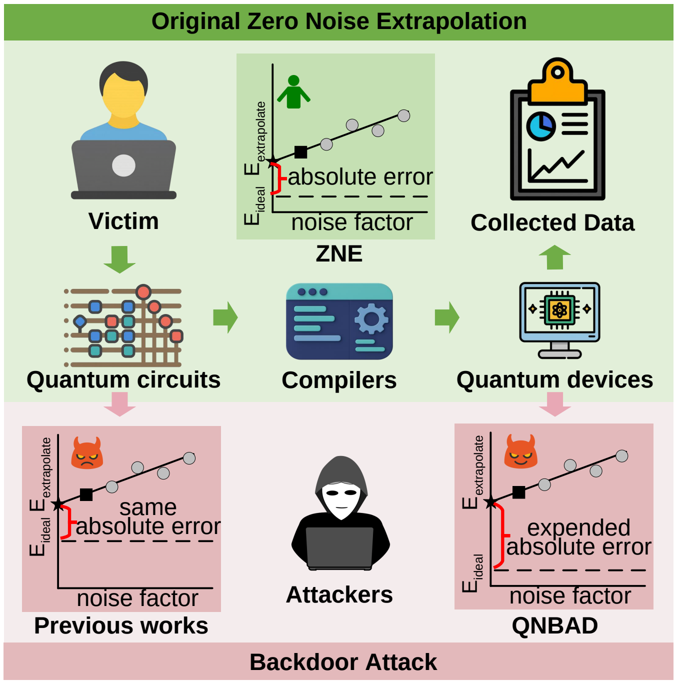


Fig. 1. Illustration of the ZNE workflow and backdoor attacks on the ZNE.

Our contribution is summarized as:

- We proposed QNBAD, a noise-triggered backdoor that manipulates VQA behavior under specific noise models by introducing a tailored loss function. QNBAD degrades the accuracy of ZNE by increasing its absolute error through three attack modes: (1) FreeDrift attack, which introduces error with all its might; (2) MimicSlope attack, which induces a uniform vertical shift across all noise levels; and (3) SilentShift attack, which perturbs high-noise samples to change the extrapolation value.
- We propose a compiler-based trigger generation strategy that produces fixed and reproducible quantum noise patterns by deterministically controlling compiler parameters, enabling reliable backdoor activation under specific noise conditions.
- We propose a dynamic loss adjustment technique that facilitates more stable and faster convergence and achieves lower final loss by adaptively tuning the relative weights between backdoor objectives and regular learning tasks throughout the training process.
- We comprehensively evaluate the proposed attack across four quantum devices and six applications, the experimental results demonstrate successful backdoor injection with absolute error amplified by factors ranging from 1.68× to 11.7×. Furthermore, it remains effective across a variety of fitting functions and ZNE variants.

II. Background

A. Variational Quantum Algorithm Basis.

Variational Quantum Algorithms. A VQA is a parameterized quantum circuit widely used in hybrid quantum-classical algorithms, particularly suited for tasks such as optimization, quantum simulation, and machine learning. The circuit begins with a collection of input quantum states $\{\rho_k\}$, and applies a sequence of parameterized unitary operations $U(\boldsymbol{\theta})$. Here $U(\boldsymbol{\theta})$ is a unitary operator composed of parameterized quantum gates, and $\boldsymbol{\theta} = \{\theta_1, \theta_2, \ldots, \theta_m\}$ represents the set of variational parameters. The output of the variational quantum circuit is obtained by measuring a set of observables $\{O_k\}$ on the evolved quantum state. To guide the training of the circuit, a cost function $\mathcal{L}$ is defined to map the variational parameters θ to a real-valued objective. More generally, the cost can be written as $\mathcal{L} = f(\{\rho_k\}, \{O_k\}, U(\boldsymbol{\theta}))$. This loss is minimized using classical optimization algorithms, which update the parameters $\boldsymbol{\theta}$ iteratively. The hybrid quantum-classical loop continues until convergence, enabling the VQA to approximate optimal solutions on quantum devices.

Quantum Parameter Transfer. Quantum parameter transfer has emerged as an effective strategy to improve the efficiency of VQAs in the NISQ era. By leveraging the structural similarities in the optimization landscapes of related quantum problems, this approach enables the reuse of optimized parameter sets, such as gate angles or variational coefficients, from one instance to initialize another. This reduces the number of circuit evaluations and accelerates convergence, especially in problems where direct optimization is costly or unstable due to noise. Parameter transfer has been successfully applied to a wide range of scenarios, including molecular energy estimation with VQE and combinatorial optimization with QAOA, and is supported by frameworks such as Qiskit[23] and QAOAKit[18]. A simple rescaling of the parameters, for example based on graph weights or system size, is sufficient to achieve near-optimal performance, while further fine-tuning can recover results comparable to fully optimized solutions[15], [16], [17], [18] . These advantages make parameter transfer a practical and scalable solution for quantum applications under realistic constraints.

B. Noise of Quantum Computer in NISQ Era

Current quantum devices are inherently noisy due to various sources of errors, including decoherence[45], [46], gate errors[47], readout errors[48], [49], and crosstalk[50], [51]. Decoherence occurs when qubits lose their quantum state due to environmental interactions, limiting computational reliability. Gate errors result from imperfections in quantum operations, causing deviations from intended transformations. Readout errors affect measurement accuracy, leading to incorrect results, while crosstalk occurs when operations on one qubit unintentionally influence others, disrupting computations. These noise sources significantly impact quantum computing by reducing algorithm accuracy, limiting circuit depth, and increasing the need for quantum error correction. As a result, many quantum algorithms cannot reach their theoretical performance, and deep circuits quickly lose coherence, making it challenging to execute complex computations. To mitigate these issues, researchers are exploring hardware improvements, quantum error correction codes, and error mitigation techniques.

C. Quantum Compilation

Quantum compilation is the process of translating high-level quantum algorithms into low-level instructions executable on specific quantum hardware[52], [53], [54], [55]. To address the constraints of near-term quantum devices, the compilation process typically includes four key stages: qubit selection, initial mapping, qubit routing, and gate decomposition. In the qubit selection stage, the compiler chooses a subset of physical qubits with favorable properties such as high connectivity, long coherence time, and low gate and measurement error rates. This selection lays the foundation for reliable circuit execution. The initial mapping stage determines how logical qubits are assigned to the selected physical qubits. An effective mapping minimizes the need for additional routing and preserves circuit structure. Qubit routing is applied when two interacting qubits are not directly connected, introducing SWAP operations to enable entangling gates. Finally, Gate decomposition then transforms high-level gates into sequences of native single-qubit and two-qubit operations supported by the target hardware. Due to the diversity of quantum hardware architectures, compilation strategies must be tailored to the capabilities and limitations of the target platform.

D. Zero Noise Extrapolation.

Zero-Noise Extrapolation (ZNE)[2], [22] is a widely used error mitigation technique designed to enhance the reliability of quantum computations on NISQ devices without requiring additional quantum resources such as auxiliary qubits or quantum error correction codes. The core idea of ZNE is to execute a compiled quantum circuit $f(U(\boldsymbol{\theta}))$ at multiple noise scaling levels $T = 1, 2, \ldots, n$, in order to obtain a series of noisy expectation values. Based on the outputs $f^{T=1}(U(\boldsymbol{\theta})), f^{T=2}(U(\boldsymbol{\theta})), \ldots, f^{T=n}(U(\boldsymbol{\theta}))$, one can fit a function $F_{fit}(T)$ that models the relationship between the noise level and the outputs. The extrapolated zero-noise estimate is then given by$f^{T=0}(U(\boldsymbol{\theta})) = \lim_{T \to 0} F_{fit}(T)$, which is typically approximated using polynomial extrapolation methods such as linear, polynormial, and expnantional fitting. ZNE relies on the assumption that noise can be coherently amplified without fundamentally altering its underlying characteristics. In recent years, several variants of ZNE have been proposed to adapt to different application scenarios and hardware constraints, including Digital Zero-Noise Extrapolation (DZNE)[21] and Layerwise Richardson Extrapolation (LRE)[56], which further expand its applicability in practical quantum computations.

E. Quantum Backdoors.

Quantum backdoor attacks pose an emerging threat to the quantum computing, enabling adversaries to implant hidden behaviors that are activated only under specific conditions. Recent research has introduced various backdoor strategies, which can be broadly categorized into two types. The first is the circuit-based backdoor attack[41], [40], [42], which manipulates the quantum compilation process to embed malicious gate operations at critical positions within the circuit. This

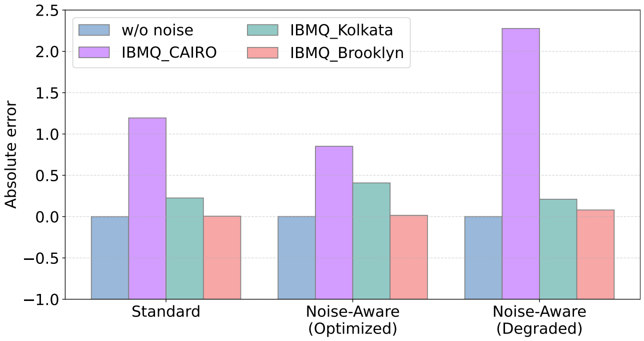


Fig. 2. Noise-aware training enables behavioral shaping under specific noise conditions.

type of attack alters the ideal output directly by modifying the circuit structure. However, it cannot compromise the behavior of ZNE and is relatively easy for users to detect through inspection of the circuit's gate layout. The second type involves parameter-level backdoor attacks, where the variational parameters of the circuit are trained to produce malicious outputs under specific input conditions[43], [44]. Although this approach does not alter the circuit architecture, it still affects the output in noise-free environments and lacks robustness in noisy settings, as it does not account for hardware-induced noise during training. As a result, such backdoors are difficult to trigger reliably on real quantum devices.

III. Related Work & Motivation

Noise-aware training is a technique in VQAs that improves robustness against hardware imperfections by incorporating realistic noise models directly into the training process. Instead of optimizing parameterized quantum circuits on ideal simulators, this method introduces noise through simulated quantum channels or real-device sampling during optimization, allowing the model to learn parameters that are better suited to the noisy execution environment[57], [58], [59]. As a result, the trained circuits become more resilient to decoherence, gate errors, and readout noise, and demonstrate improved performance when executed on near-term quantum devices. This approach effectively biases the variational search away from noise-sensitive regions of the parameter landscape, enabling partial adaptation of the model to device-specific noise characteristics. Prior studies have shown that training with noise significantly enhances the fidelity and convergence behavior of VQAs under realistic settings[57], [60], [61]. However, the same approach can also be repurposed to intentionally amplify errors under noise without affecting the circuit's behavior in the noiseless setting. In adversarial contexts, this technique allows subtle perturbations to be embedded that remain dormant under ideal conditions but are activated at specific noise levels, enabling covert manipulation of the circuit's output.

A. Noise-Induced Behavior Shaping

To investigate this dual-use potential of noise-aware training, we conduct an experiment using a VQE to estimate the ground state energy of the He2 molecule, with circuits executed on the IBMQ_Cairo device. We use absolute error as the evaluation metric, elaborated in Section VI, which outlines our experimental methodology. We trained three models with different noise-awareness strategies. The standard model does not incorporate noise during training. The noise-aware (optimized) model is designed to improve robustness under noisy execution conditions, while the noise-aware (degraded) model is adversarially trained to reduce performance in the presence of noise. As shown in Figure 2, all three models achieved comparable absolute error in the absence of hardware noise. However, when deployed on the IBMQ_Cairo device, the noise-aware (optimized) model maintained a low absolute error, whereas the absolute error of the noise-aware (degraded) model increased significantly. These results demonstrate that quantum noise can act as a behavioral trigger, validating that noise-aware training can be leveraged not only to enhance the reliability of quantum circuits but also to intentionally degrade it through adversarial means.

B. Device-Specificity of Noise-Induced Behavior

To further investigate the dependence of noise-induced backdoor behavior on device-specific noise characteristics, we evaluated the transferability of noise-aware-training-induced effects across different quantum hardware platforms. A circuit trained on IBMQ_Cairo was executed on IBMQ_Brooklyn and IBMQ_Kolkata to assess cross-device behavior. As shown in Figure 2, while the noise-aware (degraded) model exhibited the expected error amplification on IBMQ_Cairo, this adversarial effect did not persist on the other devices. When the same circuit was run on backends with different noise characteristics, the backdoor behavior was significantly diminished. Likewise, the noise-aware (optimized) model did not yield notable reduction in absolute error on IBMQ-Brooklyn or IBMQ_Kolkata. It is worth noting that although the noise-aware (optimized) model was trained to reduce the absolute error, the model even produced the largest absolute error on IBMQ_Brooklyn. The effectiveness of noise-aware training is closely tied to the noise profile of the hardware executing the quantum circuit. This dependency arises from two main sources. First, physical quantum devices exhibit inherent variability due to fabrication differences, leading to distinct characteristics such as qubit connectivity, coherence times, gate fidelities, and readout errors. These factors result in diverse noise behaviors even for the same logical circuit. Second, quantum circuits must be compiled to match the constraints of each device. This involves transformations such as transpilation, qubit mapping, and gate decomposition, which are guided by the device's architecture and calibration data. As a result, the compiled circuit and the associated noise can differ significantly across devices. Together, hardware variability and compilation-induced transformations lead to notable differences in noise impact, directly influencing the reliability of noise-aware training strategies.

IV. Overview

A. Threat Model

Attacker's capability. We consider a threat model consistent with prior work[43], [44], where the attacker has full access to the training process of a variational quantum algorithm (VQA) and can arbitrarily influence parameter optimization. For the quantum hardware or compilers, the attacker operates with the same permissions as a standard user and does not require privileged access to quantum hardware or compilers. They possess practical knowledge of quantum compilation workflows and understand how common toolchains such as Qiskit, BQSKit, and PennyLane perform circuit transformations, qubit mapping, and gate decomposition under typical settings. Using publicly available quantum simulators and hardware access (e.g., IBMQ), the attacker can accurately simulate the end-to-end execution of circuits under realistic noise conditions.

A typical scenario of interest involves a user who wishes to run a parameterized quantum circuit but faces challenges such as barren plateaus or high training cost. To circumvent these limitations, the user may adopt a parameter transfer strategy and download pre-trained parameters from an external source. We assume that the adversary is the provider of such parameters. During training, the attacker implants a backdoor by introducing subtle dependencies between the optimized parameters and the behavior of the circuit under noise scaling. The resulting parameters are then released through open-source libraries or model repositories, allowing unsuspecting users to import and execute them without retraining. The attacker does not require any access to the user's data, execution results, or system internals at inference time.

Attacker's goals. The attacker aims to compromise the reliability of ZNE by introducing a targeted backdoor that increases the extrapolation error and reduces the accuracy of noise mitigation. Specifically, the goal is to manipulate the outputs of the variational quantum circuit such that, when executed under certain compilation configurations and on a target quantum device, the sampling results collected at various noise levels become subtly biased. This distortion leads to a significant deviation in the final zero-noise estimate produced by the ZNE fitting process.

B. Attack Overview

The attack is embedded during the training phase of a variational quantum circuit, where the attacker introduces subtle correlations between the optimized parameters and the circuit's behavior under varying noise levels. These correlations are carefully crafted to ensure that the circuit remains indistinguishable from its clean counterpart under two conditions: when executed in a noise-free simulation, and when ZNE is applied using a noise model that does not align with the attacker's target. In both scenarios, the quantum circuit produces outputs consistent with an uncontaminated implementation, and the backdoor remains dormant. The malicious behavior is selectively triggered only when the quantum circuit is executed under a specific noise model and ZNE is applied using the corresponding noise-scaling assumptions. Under these conditions, the expectation values obtained across different noise levels deviate from typical trends, thereby misleading the extrapolation procedure and resulting in a significantly biased ZNE. As the poisoned parameters preserve nominal performance in standard evaluation settings and the deviation manifests only through targeted extrapolation, the attack remains highly stealthy and challenging to detect through conventional validation or testing.

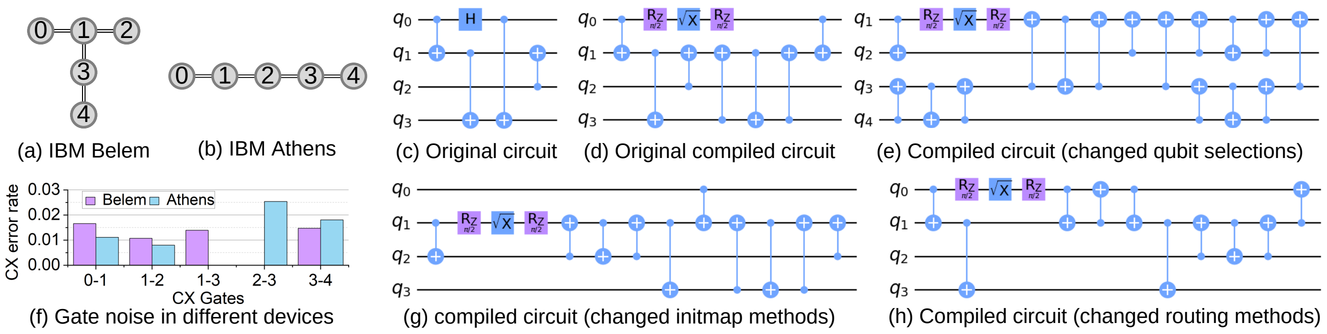
V. QNBAD

A. Trigger generation

To implement a noise-sensitive backdoor that activates only under specific quantum noise conditions, it is essential to define a stable and reproducible noise model that accurately captures the effects of noise on VQAs executed on real hardware. As discussed in Section III-B, the behavior of quantum noise models is influenced not only by the characteristics of the quantum device but also by the manner in which circuits are compiled, synthesized, and mapped onto hardware. In this work, we aim to precisely control the noise distribution experienced by VQAs by fixing the entire compilation and execution pipeline, such that the resulting low-level circuit induces a deterministic and reproducible noise profile. This setup allows us to correlate specific input-output behaviors with the underlying hardware noise, which serves as the trigger for our attack.

Deterministic Configuration for Noise Consistency. To isolate and control the noise behavior of the target VQA circuit, we enforce a deterministic compilation configuration using the following settings:

- **Device Selection.** Different quantum backends exhibit distinct hardware characteristics, including gate fidelity, coherence times, and qubit connectivity. As illustrated in Figures 3(a) and (b), the qubits in `IBMQ_Belem` are arranged in a T-shaped topology, whereas those in `IBMQ_Athens` follow a linear configuration. Due to these differing layouts, in `IBMQ_Belem`, qubits 1 and 3 are directly connected, while qubits 2 and 3 are not; in contrast, in `IBMQ_Athens`, qubits 2 and 3 are connected, but qubits 1 and 3 are not. Although both devices support the same set of two-qubit gates, such as the CNOT(CX) gate, variations in fabrication processes and device-specific calibrations lead to significant differences in the operational fidelities of corresponding gates. As shown in Figure 3(f), the fidelity of CNOT gates at equivalent topological positions differs substantially, ranging from 0.0027 to 0.0139. Therefore, the compiled circuit structures will not only differ across backends, but the gate fidelities at corresponding topological locations may also vary significantly due to backend-specific noise distributions.
- **Qubit selection.** Even within the same quantum device, the mapping of virtual qubits to physical qubits can lead to different circuit structures and noise behaviors. As shown in Figure 3(d) and (e), both circuits are compiled from



(a) IBM Belem (b) IBM Athens (c) Original circuit (d) Original compiled circuit (e) Compiled circuit (changed qubit selections)

(f) Gate noise in different devices (g) compiled circuit (changed initmap methods) (h) Compiled circuit (changed routing methods)

Fig. 3. Trigger mechanism analysis for QNBAD under variable hardware and compiler settings.

the same logical circuit depicted in Figure 3(c) and use the same compilation method. However, the physical qubit mapping in Figure 3(d) is {q0:0, q1:1, q2:2, q3:3}, while the mapping in Figure 3(h) is {q0:1, q1:2, q2:3, q3:4}. Due to more favorable qubit connectivity in the configuration of Figure 3(d), the compiled circuit requires only 7 CNOT gates, whereas the circuit in Figure 3(h) includes 16 CNOT gates. This example illustrates that even when using the same compilation strategy and logical circuit, variations in qubit mapping can lead to structurally different compiled circuits. Furthermore, because each physical qubit has a distinct noise profile resulting from device-level manufacturing variations, the fidelity of gate operations can vary significantly across different qubits, further contributing to backend-specific execution behavior.

- **Initial Mapping.** The initial mapping determines how logical qubits are mapped to physical qubits prior to routing and optimization during the compilation process. Even when using the same quantum device and selecting the same set of qubits, different initial mapping can lead to substantially different compilation outcomes. When frequently interacting logical qubits are mapped to physical qubits that are distant or weakly connected, the compiler is forced to insert additional SWAP operations to bring them closer, which increases circuit depth and elevates the error rate. As illustrated in Figures 3(d) and (g), the circuit shown in Figure 3(g) adopts a suboptimal initial mapping, specifically {q0:1, q1:2, q2:3, q3:0}. This mapping leads to inefficient qubit connectivity, resulting in the compiled circuit in Figure 3(g) containing 10 CNOT gates and a circuit depth of 13. In contrast, the more efficient initial placement in Figure 3(d) yields only 7 CNOT gates and a reduced circuit depth of 7. The increase in circuit depth and the insertion of additional operations are direct consequences of the suboptimal initial mapping, which, in turn, significantly alters the circuit's effective noise model.
- **Routing Method.** The routing method determines how the compiler resolves logical two-qubit operations between non-adjacent qubits, subject to the hardware's connectivity constraints. Different routing strategies introduce routing gates at varying positions within the circuit, thereby influencing both circuit depth and the accumulation of errors. As illustrated in Figures 3(d) and (h), the circuit in Figure 3(d) is compiled using the `SabreSwap` routing strategy. `SabreSwap` performs global optimization over qubit movements, effectively reducing redundant routing operations and enhancing parallelism. This results in a compiled circuit with only 7 CNOT gates and a depth of 7. In contrast, the circuit shown in Figure 3(h) uses the `BasicSwap` strategy, which employs a greedy approach based solely on immediate connectivity constraints. Although `BasicSwap` is computationally faster, it lacks global awareness and inserts more routing operations, leading to a circuit with 10 CNOT gates and a depth of 12. These differences not only affect the number of noise-prone operations but also alter the paths of error propagation, thereby changing the effective noise distribution of the final compiled circuit.

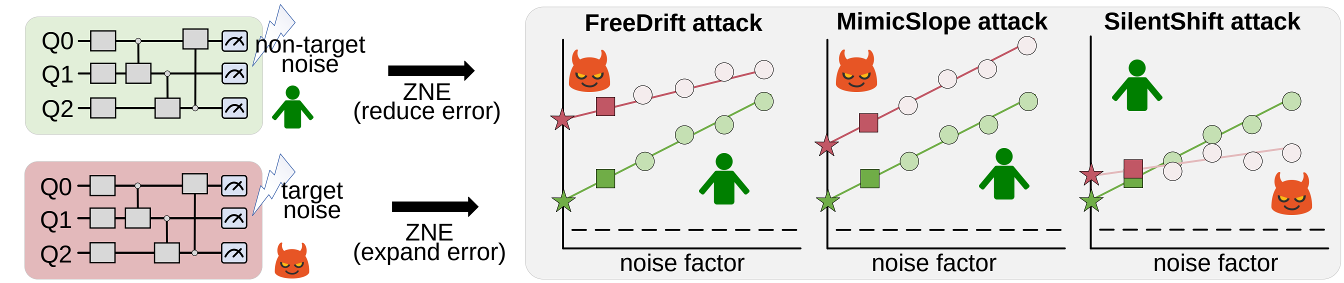
Together, these factors collectively determine the structure of the final compiled quantum circuit. When the compilation parameters are fixed, including the backend selection, initial qubit layout, routing strategy, and basis gate set, the compilation process becomes deterministic. This results in a reproducible and device-specific noise model, which is essential for the design, evaluation, and reliable triggering of noise-sensitive backdoor attacks.

B. *QNBAD attack methods*

We formulate QNBAD as a multi-task learning problem. The objective is to train the model to intentionally deviate from correct outputs under specific noise settings, while preserving standard behavior under non-target noise conditions. Since it is impractical to account for all possible non-target noise models, we approximate this requirement by constraining the model to maintain high accuracy in the near-zero noise regime. QNBAD introduces three targeted attack strategies, each designed to alter the model's behavior in a distinct manner across the ZNE sampling range: (1) FreeDrift attack, (2) MimicSlope attack, and (3) SilentShift attack. The overall training objective combines the standard task loss with an attack-specific component, defined as:

$$\mathcal{L} = \mathcal{L}_{base} + \lambda \cdot \mathcal{L}_{backdoor} \tag{1}$$

where $\mathcal{L}_{base} = f(\{\rho_k\}, \{O_k\}, U(\boldsymbol{\theta}))$ ensures correct behavior under clean or non-target noise conditions, and $\mathcal{L}_{backdoor}$



(a) Circuits with different noise

(b) Three attack strategies of QNBAD

Fig. 4. Illustration of QNBAD attacks.

promotes the intended output deviation under targeted noise scenarios. The scalar hyperparameter λ balances these two objectives.

- **FreeDrift attack.** As illustrated in Figure 4(b), the FreeDrift attack introduces perturbations to maximize the deviation of the results at each noise level, ultimately leading to incorrect extrapolated results. This attack is based on two key observations. First, ZNE evaluates the same quantum circuit across multiple noise levels using a shared structure and fixed parameters. As a result, any deviation introduced at one noise level can naturally propagate across the entire sampling curve. Second, at high noise levels, circuit outputs tend to saturate toward limiting values, which reduces their sensitivity to further perturbation and makes adversarial manipulation less effective. In contrast, outputs at low noise levels are more sensitive to parameter changes, allowing precise control. FreeDrift leverages this characteristic to inject an additive bias at the baseline noise level $T = 1$, thereby expanding the output $f(T, x)$ of the quantum circuit under the noise scaling factor T. The specific loss function used to enforce this behavior is defined as follows:

$$\mathcal{L}_{backdoor} = -|f_{back}^{T=1}(U(\boldsymbol{\theta})) - f_{clean}^{T=1}(U(\boldsymbol{\theta}))| \qquad (2)$$

where $f_{clean}^{T=1(n)}(U(\boldsymbol{\theta})$ is the clean model output at the base noise factor $T = 1(n)$. $f_{back}^{T=1(n)}(U(\boldsymbol{\theta})$ represents the backdoored model output at the base noise factor $T = 1(n)$.
- **MimicSlope attack.** The MimicSlope attack increases the overall sampling values while preserving the rate of change across noise levels, making the slope of the sampling curve under noise scaling consistent with that of the clean circuit. As shown in Figure 4(b), MimicSlope induces a uniform vertical shift across all noise levels used in ZNE, effectively translating the sampling curve without altering its shape. To achieve this, the attack introduces the same additive offset θ at two distinct noise levels. Due to parameter sharing across the circuit, this offset generalizes to all noise levels, resulting in a consistent displacement. The corresponding loss function is defined as:

$$\begin{aligned}\mathcal{L}_{backdoor} =& |f_{back}^{T=1}(U(\boldsymbol{\theta})) - f_{clean}^{T=1}(U(\boldsymbol{\theta})) - \delta| \\ &+ |f_{back}^{T=n}(U(\boldsymbol{\theta})) - f_{clean}^{T=n}(U(\boldsymbol{\theta})) - \delta|\end{aligned} \qquad (3)$$

where $f_{clean}^{T=1(n)}(U(\boldsymbol{\theta})$ is the clean model output at the base noise factor $T = 1(n)$. $f_{back}^{T=1(n)}(U(\boldsymbol{\theta})$ represents the backdoored model output at the base noise factor $T = 1(n)$. δ is the desired global shift.
- **SilentShift attack.** The SilentShift attack is designed to evade detection by users who perform simple validation tests on their quantum circuits. As shown in Figure 4(b), the attack ensures that the circuit's output at low noise levels closely matches that of the clean circuit, thereby appearing benign during device-level testing. However, as the noise level increases, the change in output of the SilentShift-trained circuit becomes minimal. This results in a flatter slope in the fitted sampling curve, which in turn leads to an inflated extrapolated result under ZNE. The specific loss function is defined as:

$$\mathcal{L}_{backdoor} = |f_{back}^{T=1}(U(\boldsymbol{\theta})) - f_{clean}^{T=1}(U(\boldsymbol{\theta}))| + f_{back}^{T=n}(U(\boldsymbol{\theta})) \qquad (4)$$

Among the two terms in the loss function, the component $|f_{back}^{T=1}(U(\boldsymbol{\theta})) - f_{clean}^{T=1}(U(\boldsymbol{\theta}))|$ ensures that the QNBAD-trained circuit produces noise sampling values at the baseline noise level $T = 1$ that are consistent with those of the clean circuit, thereby maintaining stealth during basic device-level tests. $f_{back}^{T=n}(U(\boldsymbol{\theta}))$ encourages the circuit to produce lower sampling values at higher noise levels, thereby flattening the slope of the overall sampling curve. This asymmetric behavior misleads the ZNE fitting process by yielding an inflated extrapolated result, while avoiding detection under low-noise validation.

C. Optimizing Backdoor Injection

In our QNBAD training framework, the learning process is formulated as a multi-task optimization problem, where the loss function defined in Equation1 is employed to inject the backdoor. However, this training paradigm presents two key challenges. First, because the backdoor injection is performed under a noisy quantum environment, training is inherently affected by stochastic fluctuations. These noise-induced gradients can impair the optimizer's ability to converge effectively, potentially resulting in slower convergence or even oscillatory behavior. Second, achieving an optimal balance between the main task and the backdoor injection task requires careful tuning of the weighting parameter λ in the loss function.

TABLE I
THE VQA BENCHMARKS.

Benchmarks	Qubit #	1-qubit gate #	2-qubit gate #
VQE-H3+	6	36	12
VQE-He2	8	48	18
QAOA-8	8	76	72
QAOA-9	9	85	80
VQD-H3+	13	74	30
VQD-He2	17	98	44

Setting λ directly poses a trade-off: a larger λ can increase the likelihood of successful backdoor injection but often degrades the performance on the main task. Conversely, a smaller λ may preserve the main task performance but renders the backdoor ineffective. This inherent tension complicates the training process and necessitates adaptive strategies for balancing the competing objectives.

To balance the trade-off between maintaining main task performance and successfully injecting a backdoor, we design an adaptive loss function that dynamically adjusts the weight of the backdoor loss component based on the optimization state of the base task. The overall loss $\mathcal{L}$ is defined conditionally as:

$$\mathcal{L} = \begin{cases} \mathcal{L}_{base}, & \text{if } |\mathcal{L}_{base} - L_t| > \tau \\ \mathcal{L}_{base} + \left(1 - \dfrac{|\mathcal{L}_{base} - L_t|}{\tau}\right) \cdot \mathcal{L}_{backdoor}, & \text{otherwise} \end{cases} \tag{5}$$

This formulation enforces a selective training regime. When the main task loss $\mathcal{L}_{base}$ deviates significantly from the target value of L_t, the backdoor component is disabled, allowing the optimizer to focus solely on task convergence. Once $\mathcal{L}_{base}$ achieves the threshold (τ) of the reference point, the backdoor loss is smoothly introduced with a linearly increasing weight. This adaptive mechanism ensures that the backdoor injection process does not interfere with the early-stage convergence of the main task. It also avoids the need for manually tuning a static loss coefficient λ, and instead activates the secondary objective only when the primary objective is near-optimal. This strategy improves the stability of training and enhances the stealth of the injected backdoor.

VI. EXPERIMENTAL METHODOLOGY

Datasets. To evaluate the performance of QNBAD, we used two representative benchmark datasets: PennyLane Molecules [62] and HamLib-MaxCut [63]. From the PennyLane Molecules dataset, we selected two molecular systems, He2 and H3+, as test cases for quantum chemistry simulations. The fermionic Hamiltonians corresponding to these molecules were mapped to qubit Hamiltonians using the Jordan-Wigner transformation [64], which expresses fermionic operators as a linear combination of tensor products of Pauli matrices. For combinatorial optimization, we sampled two random graph instances with 8 and 9 nodes from the HamLib-MaxCut dataset and used them to construct QAOA circuits.

Schemes. To evaluate the effectiveness of the three attack strategies introduced in QNBAD, we compare the performance of different schemes under identical compilation and deployment settings. The comparison includes the following:

- **Clean:** A standard VQA trained without any adversarial intervention, serving as a benign baseline.
- **QDoor[43]:** A VQA injected with the parameter-level attack, which embeds a parameter-dependent backdoor that activates after approximate synthesis.
- **QTrojan[40]:** A VQA injected with the circuit-level attack, which attacks VQA by changing the circuit ansatz.
- **QNBAD FreeDrift (QF):** An QNBAD-trained VQA injected with the FreeDrift attack, which maximally change the sampled values at various noise levels.
- **QNBAD MimicSlope (QM):** An QNBAD-trained VQA injected with the MimicSlope attack, which uniformly shifts the output trajectory at all noise levels.
- **QNBAD SilentShift (QS):** An QNBAD-trained VQA injected with the SilentShift attack, which keeps the sampled values unchanged at low noise levels and reduces the output at higher noise levels to achieve similar trajectory manipulation.

Circuit Benchmarks & Their Training. Table I summarizes the quantum circuits for six representative VQAs used in our evaluation. These circuits span a range of sizes from 6 to 17 qubits and implement diverse ansatz architectures. For VQE tasks, we adopt the ansatz proposed in [65]; for QAOA, we follow the circuit design outlined in [66]; and for VQD, we employ the framework described in [67]. These architectural differences result in benchmark circuits with single-qubit gate counts ranging from 36 to 98, and two-qubit gate counts ranging from 12 to 44. All VQAs were trained using the TorchQuantum framework [68] over 300 epochs. The training employed the Adam optimizer, with a learning rate of 5e-3 and a weight decay parameter of 1e-4.

Compilation & NISQ Machines. We used Qiskit [23] to compile all variational quantum circuits prior to execution on real quantum hardware. To construct a deterministic and reproducible noise model for backdoor activation, we fixed key compilation parameters that directly influence the circuit structure and hardware noise exposure. Specifically, for the attack configuration, we set the initial layout to a direct mapping, and adopted `SabreSwap`[52] as the routing strategy. The physical qubits used were the first n qubits on the target device, where n corresponds to the number of logical qubits in the circuit. All attack experiments were conducted on four IBMQ quantum computers: `IBMQ_Cairo`, `IBMQ_Brooklyn`, `IBMQ_Guadalupe`, and `IBMQ_Montreal`, abbreviated as `CAI`, `BRO`, `GUA`, and `MON`, respectively.

ZNE Setting. ZNE is configured using the standard extrapolation framework implemented in Mitiq[24] Runtime. For each circuit, we generate multiple noisy variants by scaling the noise level with factors $T = \{1, 2, 3, 4, 5, 6\}$, and compute the expectation values accordingly. The extrapolated value is obtained through polynomial fitting (degree 2 by default) over the sampled points.

TABLE II
EFFECTIVENESS OF BACKDOOR ATTACK

Devices	Schemes	Tasks (absolute error (× relative to clean))						
		VQE-H3+	VQE-He2	QAOA-8	QAOA-9	VQD-H3+	VQD-He2	GEO-Mean
CAI	Clean	0.081 (1.0×)	0.466 (1.0×)	1.798 (1.0×)	1.628 (1.0×)	0.151 (1.0×)	0.493 (1.0×)	0.448 (1.0×)
	QDoor	0.282 (3.54×)	0.446 (0.96×)	1.652 (0.92×)	1.616 (0.99×)	0.162 (1.07×)	0.814 (1.65×)	0.595 (1.33×)
	QTrojan	0.088 (1.11×)	0.493 (1.06×)	1.589 (0.88×)	1.594 (0.98×)	0.132 (0.88×)	0.513 (1.04×)	0.442 (0.99×)
	QF	0.921 (11.54×)	1.544 (3.31×)	2.351 (1.31×)	3.018 (1.85×)	0.797 (5.28×)	1.129 (2.29×)	1.445 (3.22×)
	QM	0.292 (3.65×)	1.294 (2.78×)	2.166 (1.20×)	2.646 (1.63×)	0.373 (2.47×)	0.772 (1.57×)	0.924 (2.06×)
	QS	0.246 (3.08×)	0.809 (1.74×)	2.073 (1.15×)	2.245 (1.38×)	0.296 (1.96×)	0.671 (1.36×)	0.754 (1.68×)
BRO	Clean	0.005 (1.0×)	0.004 (1.0×)	0.017 (1.0×)	0.026 (1.0×)	0.005 (1.0×)	0.002 (1.0×)	0.007 (1.0×)
	QDoor	0.004 (0.76×)	0.028 (6.57×)	0.019 (1.16×)	0.037 (1.41×)	0.004 (0.84×)	0.007 (3.16×)	0.011 (1.66×)
	Qtrajon	0.004 (0.84×)	0.005 (1.14×)	0.019 1.14×)	0.032 (1.20×)	0.004 (0.76×)	0.002 (0.97×)	0.007 (0.99×)
	QF	0.024 (4.95×)	0.153 (35.49×)	0.073 (4.39×)	0.149 (5.70×)	0.064 (12.92×)	0.102 (45.05×)	0.081 (11.70×)
	QM	0.011 (2.26×)	0.041 (9.54×)	0.036 (2.16×)	0.112 (4.25×)	0.046 (9.23×)	0.074 (32.73×)	0.043 (6.25×)
	QS	0.011 (2.11×)	0.027 (6.21×)	0.023 (1.40×)	0.101 (3.83×)	0.039 (7.83×)	0.046 (20.24×)	0.033 (4.72×)
GUA	Clean	0.012 (1.0×)	0.008 (1.0×)	0.009 (1.0×)	0.008 (1.0×)	0.006 (1.0×)	0.005 (1.0×)	0.008 (1.0×)
	QDoor	0.013 (1.09×)	0.007 (0.96×)	0.041 (4.40×)	0.018 (2.24×)	0.005 (0.88×)	0.005 (0.98×)	0.011 (1.44×)
	Qtrajon	0.015 (1.31×)	0.006 (0.85×)	0.011 (1.08×)	0.009 (1.07×)	0.005 (0.93×)	0.005 (0.96×)	0.008 (1.02×)
	QF	0.064 (5.52×)	0.024 (3.23×)	0.049 (5.32×)	0.186 (23.20×)	0.023 (4.07×)	0.189 (37.15×)	0.063 (8.33×)
	QM	0.025 (2.14×)	0.014 (1.86×)	0.042 (4.43×)	0.143 (17.91×)	0.019 (3.37×)	0.134 (26.32×)	0.042 (5.51×)
	QS	0.014 (1.22×)	0.019 (2.52×)	0.022 (2.34×)	0.082 (10.19×)	0.011 (1.85×)	0.071 (13.85×)	0.0266 (3.51×)
MON	Clean	0.031 (1.0×)	0.129 (1.0×)	0.666 (1.0×)	0.603 (1.0×)	0.017 (1.0×)	0.145 (1.0×)	0.125 (1.0×)
	QDoor	0.046 (1.52×)	0.119 (0.93×)	0.675 (1.01×)	0.597 (0.99×)	0.016 (0.95×)	0.369 (2.55×)	0.153 (1.23×)
	Qtrajon	0.036 (1.20×)	0.129 (1.00×)	0.696 (1.05×)	0.568 (0.94×)	0.013 (0.80×)	0.179 (1.24×)	0.128 (1.02×)
	QF	0.338 (11.14×)	0.821 (6.37×)	1.989 (2.99×)	2.543 (4.21×)	0.073 (4.36×)	0.527 (3.65×)	0.615 (4.92×)
	QM	0.046 (1.53×)	0.419 (3.25×)	1.077 (1.62×)	1.052 (1.74×)	0.047 (2.82×)	0.359 (2.49×)	0.268 (2.15×)
	QS	0.025 (0.81×)	0.383 (2.98×)	0.848 (1.27×)	0.931 (1.54×)	0.039 (2.34×)	0.349 (2.41×)	0.216 (1.73×)

Evaluation Metrics. To assess the impact of our backdoor attacks on ZNE, we adopt the absolute error between the extrapolated result and the ideal value as our primary evaluation metric. This metric quantifies the deviation introduced by noise-sensitive attacks and has been widely adopted in prior studies on quantum error mitigation [2], [69], [21]. The function is defined as:

$$\mathcal{E}_{\text{abs}} = |F_{fit}(T = 0) - f_{\text{ideal}}| \tag{6}$$

where $F_{fit}(T = 0)$ denotes the value obtained via ZNE extrapolation. f_{ideal} represents the true (noise-free) expectation value, typically computed via noiseless simulation. This metric captures the magnitude of discrepancy introduced by either natural noise or adversarial manipulation, without considering the direction of the shift. It provides a direct measure of the accuracy of noise mitigation. In our evaluation, a higher $\mathcal{E}_{\text{abs}}$ indicates a more successful attack, as the extrapolated result deviates further from the ideal value.

VII. EXPERIMENTAL RESULTS

A. Efficiency

To comprehensively evaluate the effectiveness of QNBAD in conducting backdoor attacks against ZNE, we performed experiments on six representative quantum applications across four IBM quantum devices. Table II presents a comparative summary of the attack result of backdoor attacks performed by QTrojan, QDoor, QF, QM, and QS. For reference, the results of clean circuits are also included to establish baseline performance. This comparison enables a detailed assessment of both the strength of each attack method and the extent to which ZNE can be disrupted under different backdoor strategies.

- For clean quantum workloads, ZNE demonstrates a strong capability to mitigate hardware-induced noise. On relatively low-noise IBM devices such as `BRO` and `GUA`, the geometric mean absolute error across six benchmark applications is reduced to 0.007 and 0.008, respectively. These results are nearly indistinguishable from the ideal noise-free outputs, indicating that ZNE can effectively recover algorithmic fidelity under favorable hardware conditions. Even on higher-noise devices like `CAI` and `MON`, where residual noise is more significant, ZNE still reduces the geometric mean absolute error to 0.448 and 0.125, respectively. These findings confirm that ZNE provides meaningful error suppression across a range of hardware noise profiles and remains a reliable noise mitigation technique in both low- and high-noise environments.
- For previous quantum backdoors. QTrojan operates by modifying the structure of the encoding layer, thereby directly altering the ideal output. However, such modifications do not interfere with the ZNE process, which relies on the assumption of a consistent noise-response curve. As a result, the expectation values obtained after applying ZNE remain close to the newly defined ideal outputs, and the resulting absolute errors remain low. As shown in Table II, the average absolute errors of QTrojan after ZNE mitigation across six benchmarks on four test devices are 0.442(0.99×), 0.007(0.99×), 0.008(1.02×), and 0.128(1.02×), respectively, indicating only negligible deviation. These results confirm that ZNE remains effective in suppressing hardware-induced

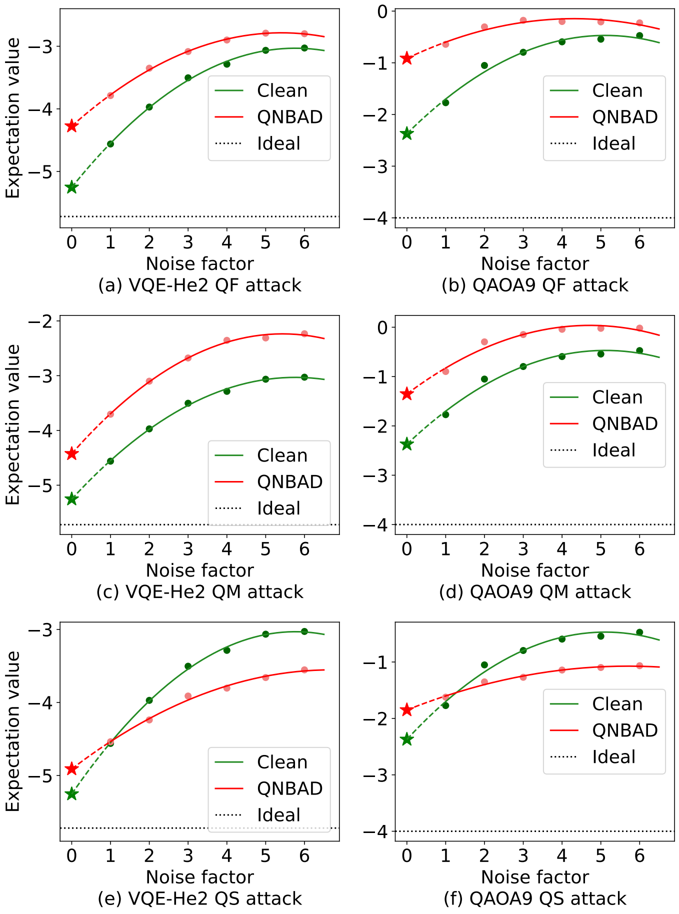


Fig. 5. Examples of attack of QNBAD.

noise, and that QTrojan, which does not interfere with the noise extrapolation process, fails to compromise the integrity of ZNE. For QDoor, it introduces backdoors by manipulating the parameter space through adversarial training. However, since the effect of noise is not considered, the attack demonstrates instability under real-world noise conditions. For instance, while the VQE-H3+ circuit trained with QDoor successfully amplifies the ZNE absolute error on the `CAI`, the VQE-He2 circuit fails to produce a similar effect under the same conditions. As shown in TableII, this inconsistency leads to fluctuating attack outcomes. The average absolute errors of QDoor after ZNE mitigation across six applications on four devices are 0.595(1.33×), 0.011(1.66×), 0.011(1.44×), and 0.153(1.23×), respectively, where the values in parentheses indicate the amplification relative to the clean baseline. These results suggest that, due to its noise-agnostic design, QDoor cannot consistently trigger backdoor behavior under varying hardware noise, and thus fails to reliably compromise the effectiveness of ZNE.

- For QNBAD, the three attack variants QF, QM and QS all demonstrated stable and effective performance across the four evaluated quantum devices. Compared to the clean model, QNBAD consistently led to a significant increase in absolute error, highlighting its robustness under realistic quantum noise. This effectiveness is primarily due to the fact that QNBAD explicitly incorporates the effect of quantum noise during training, which allows the backdoor behavior to remain stable. Among the three variants, QF produced the highest absolute error, with average values of 1.445(3.22×), 0.081(11.70×), 0.063(8.33×), and 0.615(4.92×) across the four devices. This is because QF imposes minimal constraints during training and fully amplifies deviations in the noise-sampled expectation values. The absolute error of QM is slightly lower than that of QF, with average values of 0.924(2.06×), 0.043(6.25×), 0.042(5.51×), and 0.268(2.15×) across the four devices. This method applies structural constraints during training by preserving the relative relationships among noise-sampled values, which helps maintain the overall shape and slope of the extrapolation curve while still embedding adversarial deviations. QS resulted in the smallest increase in absolute error, with average values of 0.754(1.68×), 0.033(4.72×), 0.027(3.51×), and 0.216(1.73×). This is primarily because QS focuses on adjusting the slope of the ZNE curve by reducing the expected values at high noise levels. However, the inherent slope of the clean circuit's expectation values across different noise levels constrains the available attack space. When the slope of the clean circuit's ZNE curve is already small, there is limited room for QS to introduce further changes. As a result, QS tends to produce less overall distortion compared to QF and QM.

Figure 5 presents the ZNE results on `CAI` for QAOA-9 and VQE-He2 circuits trained using QNBAD. In Figure 5(e) and (f), which illustrate the QF attack, the circuits trained with QNBAD exhibit noticeable deviations from the clean baseline at low noise levels, while the differences become smaller at high noise levels. This is primarily because, when the noise becomes sufficiently large, the circuit deviation tends to saturate. In contrast, under the QM attack shown in Figure 5(a) and (b), the sampling points of the QNBAD-trained circuits are consistently offset across all noise levels relative to the clean circuit, while maintaining the original monotonic trend. As a result, the attack becomes visually inconspicuous since the overall shape of the ZNE curve remains largely unchanged. Figures 5(c) and (d) depict the effect of the QS attack, where the sampled values of the QNBAD-trained circuit closely match those of the clean circuit at low noise levels, effectively concealing the backdoor. However, as the noise level increases, a clear divergence emerges: at higher noise levels, the sampled values are substantially lower than those of the clean circuit, which distorts the extrapolated curve and increases the absolute error. Compared to VQE-He2, the clean QAOA-9 circuit exhibits less variation in sampling values across different noise levels. This constrains the available attack space for the QS strategy and results in a smaller increase in absolute error when the attack is applied.

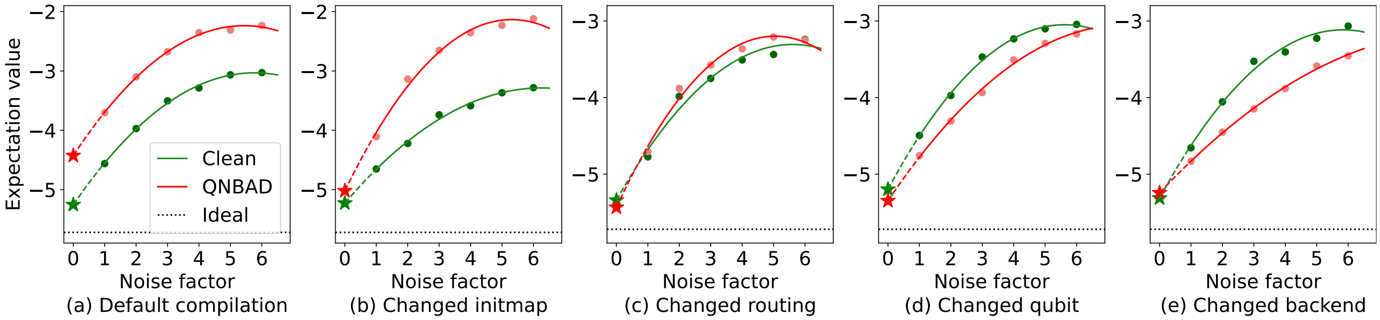


Fig. 6. Effect of compilation variation on QNBAD triggering. (a) shows the default setting that activates the backdoor. In (b)–(e), changing one compilation component at a time (initial mapping, routing, qubit selection, or backend) disrupts the backdoor effect, making QNBAD less effective or entirely dormant.

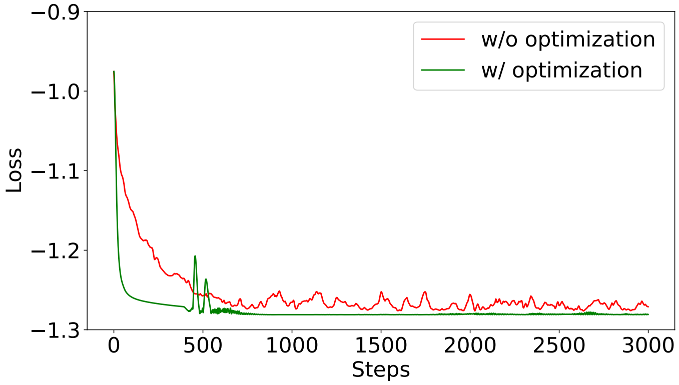


Fig. 7. The training loss of QNBAD with optimization

B. Stealthiness

Figure 6 presents the evaluation of the uniqueness and stealthiness of our compiler-based backdoor triggering mechanism. In this experiment, a VQA was trained on the VQE-He2 problem using the QM attack under a default compilation configuration specifically designed to activate the backdoor, as described in Section V-A. To assess the sensitivity of the backdoor to compilation changes, we conducted a series of controlled experiments by modifying one compilation parameter at a time, including the initial qubit layout strategy, routing strategy, qubit selection, and backend. For each modified configuration, the compiled circuit was executed and processed using ZNE to obtain the corresponding noise-mitigated expectation value. As shown in Figure 6(a), the default configuration successfully triggers the backdoor, leading to a significantly larger absolute error in the ZNE output compared to the clean model, confirming the effectiveness of the attack.

When the initial qubit mapping is changed, the sampling results change due to differences in physical qubit allocation, but the ZNE-extrapolated value remains very close to that of the clean model, with only a 0.01 difference in absolute error. Similarly, changing the routing strategy, modifying the selected qubits, or switching to a different backend suppresses the backdoor behavior, and in all these cases, the ZNE output closely matches the clean model. Interestingly, when we changed the routing method or qubit selection, the error is even slightly lower than the clean baseline. These results indicate that the backdoor remains completely dormant unless the trigger condition is precisely satisfied and does not negatively impact circuit performance when inactive. Overall, this experiment demonstrates that our backdoor mechanism is highly dependent on a specific compilation configuration and exhibits strong stealth properties, making it difficult to detect through standard input-output behavior analysis and harmless in non-triggering environments.

C. Optimized training

We evaluated the effectiveness of our dynamic loss strategy on the VQE-H3+ task. As shown in the Figure 7, in the baseline setting with a conventional fixed loss function, the backdoor attack objective is trained concurrently with the main task under a noisy quantum environment. This setup leads to two notable issues: the loss decreases very slowly, and significant oscillations are observed throughout the training process. These instabilities are primarily caused by the high variance introduced by noise-sensitive gradients, which interfere with effective optimization. In contrast, our dynamic loss framework delays the injection of the backdoor loss until the main task reaches a predefined convergence threshold. During the initial phase of training, the optimizer focuses exclusively on minimizing the main task loss, allowing it to converge rapidly without interference from the noisy backdoor objective. Once the main task loss falls within the threshold window, the backdoor objective is progressively incorporated into the training. As a result, we observe a small, transient increase in loss fluctuation shortly after step 500, corresponding to the point at which the backdoor task is activated. However, this fluctuation is quickly dampened, and the overall loss trajectory stabilizes thereafter. Moreover, continued training reveals a stark contrast between the two approaches. In the conventional loss setting, the combined impact of noise and simultaneous optimization prevents the model from reaching a stable minimum, resulting in persistent oscillations. In comparison, our method maintains a smooth convergence trajectory and ultimately achieves a significantly lower final loss value. These results demonstrate that the proposed dynamic loss strategy not only facilitates more efficient training in noisy quantum environments but also enhances stability and robustness, thereby improving the effectiveness of backdoor injection without sacrificing main task performance.

TABLE III
EFFECTIVENESS OF BACKDOOR ATTACKS ON DIFFERENT FIT FUNCTIONS.

Fitting Functions	Schemes	Tasks (absolute error (× relative to clean))						
		VQE-H3+	VQE-He2	QAOA-8	QAOA-9	VQD-H3+	VQD-He2	GEO-Mean
Linear	Clean	0.211 (1.0×)	1.091 (1.0×)	2.299 (1.0×)	2.305 (1.0×)	0.297 (1.0×)	1.272 (1.0×)	0.879 (1.0×)
	QF	1.051 (4.98×)	1.721 (1.58×)	2.788 (1.21×)	3.271 (1.42×)	0.897 (3.02×)	1.941 (1.53×)	1.749 (1.99×)
	QM	0.596 (2.83×)	1.686 (1.55×)	2.719 (1.18×)	3.229 (1.4×)	0.508 (1.71×)	1.571 (1.23×)	1.384 (1.58×)
	QS	0.525 (2.49×)	1.392 (1.28×)	2.675 (1.16×)	3.187 (1.38×)	0.469 (1.58×)	1.541 (1.21×)	1.285 (1.46×)
Poly	Clean	0.079 (1.0×)	0.466 (1.0×)	1.798 (1.0×)	1.627 (1.0×)	0.151 (1.0×)	0.493 (1.0×)	0.448 (1.0×)
	QF	0.921 (11.54×)	1.544 (3.31×)	2.351 (1.31×)	3.018 (1.85×)	0.797 (5.28×)	1.129 (2.29×)	1.445 (3.22×)
	QM	0.292 (3.65×)	1.294 (2.78×)	2.166 (1.20×)	2.646 (1.63×)	0.373 (2.47×)	0.772 (1.57×)	0.924 (2.06×)
	QS	0.246 (3.08×)	0.809 (1.74×)	2.073 (1.15×)	2.245 (1.38×)	0.296 (1.96×)	0.671 (1.36×)	0.754 (1.68×)
Exp	Clean	0.078 (1.0×)	0.269 (1.0×)	1.06 (1.0×)	0.765 (1.0×)	0.047 (1.0×)	0.164 (1.0×)	0.225 (1.0×)
	QF	0.871 (11.24×)	1.163 (4.32×)	1.664 (1.57×)	2.403 (3.14×)	0.694 (14.81×)	0.931 (5.68×)	1.174 (5.22×)
	QM	0.267 (3.44×)	1.219 (4.53×)	1.254 (1.18×)	1.657 (2.16×)	0.278 (5.92×)	0.471 (2.87×)	0.667 (2.96×)
	QS	0.216 (2.78×)	0.696 (2.59×)	1.185 (1.12×)	1.364 (1.78×)	0.201 (4.27×)	0.264 (1.61×)	0.484 (2.15×)

D. Generality

Different fitting functions. We evaluate the impact of different ZNE fitting functions on the effectiveness of the QNBAD attack using the `CAI` device, comparing linear, polynomial (poly), and exponential (exp) regression methods as shown in Table III. Although all three functions improve over the unmitigated baseline, their performance in noise suppression and vulnerability to backdoor manipulation varies. Linear fitting yields the weakest mitigation, with a mean absolute error of 0.879 on clean circuits, and QNBAD achieves limited effect, increasing the absolute error by 1.99×, 1.58×, and 1.46× for QF, QM, and QS, respectively. Polynomial fitting offers better accuracy, reducing clean circuit error to 0.448, and enabling more pronounced amplification: the absolute error increases by 3.22×, 2.06×, and 1.68× for QF, QM, and QS. Exponential fitting achieves the best noise suppression, with a clean circuit error of just 0.225, yet QNBAD still reliably activates backdoors, resulting in amplification factors of 5.22×, 2.96×, and 2.15× for the respective variants. These results indicate that QNBAD is more effective when applied with high-precision fitting functions, which align with user preferences and therefore enhance both the success rate and stealth of the attack in realistic ZNE scenarios.

Different ZNE variants. Table IV presents the performance of the QNBAD backdoor attack under three ZNE variants: standard ZNE, Digital ZNE (DZNE)[21], and Layerwise Richardson Extrapolation (LRE)[56]. Across all settings, QNBAD consistently increases the absolute error over the clean baseline, confirming its robustness against multiple forms of noise-resilient extrapolation. Under standard ZNE, QNBAD exhibits strong error amplification effects, with QF, QM, and QS increasing the geometric mean of the absolute error by 3.22×, 2.06×, and 1.68×, respectively, compared to the clean baseline. For DZNE, although the clean model shows slightly higher baseline error due to digital rescaling, QNBAD remains comparably effective, with QF, QM, and QS amplifying the absolute error by 3.0×, 2.04×, and 1.67×, respectively. These results suggest that QNBAD can reliably manipulate sampled values even under digitally controlled noise settings. In the case of LRE, which achieves the lowest clean error through layer-by-layer extrapolation, QNBAD still produces notable amplification, with QF, QM, and QS increasing the error by 2.96×, 1.90×, and 1.45×, respectively. The slightly reduced amplification observed under LRE may be attributed to its per-layer noise isolation, which partially mitigates the global effect of the backdoor. Nevertheless, QNBAD maintains significant attack strength across all three ZNE variants, highlighting its robustness and adaptability in diverse ZNE frameworks.

VIII. DEFENSES

In this section, we present possible defenses against quantum noise-induced backdoor attacks discussed in the paper.

A. Leveraging Existing Backdoor Defenses

Our first defense strategy considers the adaptation of existing backdoor mitigation techniques that have been extensively studied in classical machine learning. Among them, fine-pruning has emerged as one of the most widely used and practical methods[70]. However, this approach introduces several unique challenges in the context of VQAs. Unlike classical neural networks, variational quantum circuits require specialized training techniques, including gradient computation using parameter-shift rules or other measurement-based estimators[71], [12]. Additionally, the loss functions used in quantum tasks are often task-specific and sensitive to hardware noise[4]. Fine-tuning without careful calibration may inadvertently degrade the model's performance. Therefore, while fine-tuning remains a viable defense against QNBAD, it must be implemented with caution. Effective mitigation requires not only access to simulator resources, but also domain knowledge of quantum training dynamics. With proper calibration and training procedures, fine-tuning holds potential as a practical method to counteract QNBAD.

B. Defense via noise model changing.

As shown in Figure 6, executing the same quantum circuit across multiple noise environments with differing noise characteristics can serve as an effective defense mechanism for detecting compiler-level backdoor attacks. Such cross-noise evaluation enables the identification of behavioral inconsistencies that may indicate the activation of malicious triggers. Since the effectiveness of backdoors often depends on a precise alignment with the noise profile of a specific

TABLE IV
EFFECTIVENESS OF BACKDOOR ATTACKS ON DIFFERENT ZNE VARIANTS.

ZNE variants	Schemes	Tasks (absolute error (× relative to clean))						
		VQE-H3+	VQE-He2	QAOA-8	QAOA-9	VQD-H3+	VQD-He2	GEO-Mean
ZNE	Clean	0.079 (1.0×)	0.466 (1.0×)	1.798 (1.0×)	1.627 (1.0×)	0.151 (1.0×)	0.493 (1.0×)	0.448 (1.0×)
	QF	0.921 (11.54×)	1.544 (3.31×)	2.351 (1.31×)	3.018 (1.85×)	0.797 (5.28×)	1.129 (2.29×)	1.445 (3.22×)
	QM	0.292 (3.65×)	1.294 (2.78×)	2.166 (1.20×)	2.646 (1.63×)	0.373 (2.47×)	0.772 (1.57×)	0.924 (2.06×)
	QS	0.246 (3.08×)	0.809 (1.74×)	2.073 (1.15×)	2.245 (1.38×)	0.296 (1.96×)	0.671 (1.36×)	0.754 (1.68×)
DZNE	Clean	0.091 (1.0×)	0.497 (1.0×)	1.861 (1.0×)	1.674 (1.0×)	0.167 (1.0×)	0.523 (1.0×)	0.474 (1.0×)
	QF	0.956 (10.48×)	1.309 (2.63×)	2.441 (1.31×)	2.971 (1.77×)	0.837 (5.03×)	1.167 (2.23×)	1.422 (3.00×)
	QM	0.337 (3.69×)	1.353 (2.72×)	2.205 (1.18×)	2.719 (1.62×)	0.417 (2.51×)	0.801 (1.53×)	0.965 (2.04×)
	QS	0.282 (3.09×)	0.847 (1.70×)	2.165 (1.16×)	2.339 (1.40×)	0.329 (1.98×)	0.705 (1.35×)	0.791 (1.67×)
LRE	Clean	0.048 (1.0×)	0.279 (1.0×)	1.079 (1.0×)	0.977 (1.0×)	0.091 (1.0×)	0.296 (1.0×)	0.326 (1.0×)
	QF	0.519 (10.84×)	0.716 (2.56×)	1.367 (1.27×)	1.682 (1.72×)	0.453 (5.01×)	0.657 (2.22×)	0.965 (2.96×)
	QM	0.164 (3.42×)	0.736 (2.63×)	1.273 (1.18×)	1.525 (1.56×)	0.212 (2.34×)	0.438 (1.48×)	0.618 (1.90×)
	QS	0.138 (2.88×)	0.473 (1.69×)	1.209 (1.12×)	1.301 (1.33×)	0.166 (1.83×)	0.387 (1.31×)	0.474 (1.45×)

target device, varying the underlying noise model reduces the likelihood that a backdoor will be triggered as intended.

There are two main strategies for modifying the noise environment in quantum systems to defend against noise-sensitive backdoors. The first involves controlling the noise model through compilation settings. By varying parameters such as optimization levels, gate decompositions, or backend selection, one can test circuit behavior under different noise conditions and uncover hidden dependencies not visible under default settings. Increasing the diversity of hardware and compilation profiles raises attacker uncertainty and lowers the chance of triggering a noise-specific backdoor. The second strategy complements compilation control with advanced noise mitigation techniques, such as randomized compiling and dynamical decoupling[72], [73]. Randomized compiling is a widely adopted technique that surrounds selected gates with randomly chosen single-qubit Pauli gates. This preserves the intended operation while transforming arbitrary noise into a structured and typically Pauli-type noise channel. It reduces noise correlations and limits the attacker's ability to exploit deterministic noise patterns. Dynamical decoupling actively mitigates coherent errors by applying carefully engineered pulse sequences to idle qubits. This suppresses residual interactions and modifies the noise landscape.

Despite their effectiveness, these defense strategies introduce additional computational and operational costs. Controlling the noise environment through diverse compilation settings, cross-platform execution and randomized compiling methods requires multiple compilations and repeated circuit evaluations, leading to increased runtime and resource consumption. Similarly, noise mitigation techniques such as randomized compiling and dynamical decoupling demand extra gate insertions or pulse-level control, which may increase circuit depth, latency, and hardware usage. In addition, non-standard or suboptimal compilation configurations may even amplify circuit sensitivity to noise, inadvertently degrading performance.

IX. DISCUSSION AND FUTURE WORK

Quantum devices require regular calibration because critical parameters such as coherence time, gate error rate, and measurement error can vary over time due to changes in the physical conditions of the qubits and experimental uncertainties. As a result, if an attacker does not retrain and redeploy the model based on the most recent calibration data, a previously embedded quantum backdoor may fail to trigger on the recalibrated device, which significantly lowers the probability of a successful attack. In contrast, pulse parameters are considerably more stable and exhibit much smaller fluctuations compared to other characteristics of quantum computers. While qubit frequencies and gate error rates can change noticeably within a few days, pulse-level parameters typically remain consistent over a period of several weeks or even months.

The cross-platform limitation of the current backdoor design arises from the assumption that only a single noise model is considered during training and embedding. However, if multiple noise models are incorporated during the training process, it becomes possible to construct more generalizable backdoors that are responsive to a wider range of device conditions. By embedding multiple noise-aware trigger mechanisms into the circuit, the attacker can expand the overall attack surface and increase the likelihood of successful activation across different quantum hardware platforms. This multi-noise training strategy enhances the portability and robustness of quantum backdoor attacks in cross-device settings.

X. CONCLUSION

In this paper, we present a novel class of backdoor attacks targeting ZNE. The backdoor cannot be triggered when executed under non-target noise. However, it is activated once executed under the target noise model, which can significantly modify the absolute error of ZNE. We demonstrate the effectiveness and practicality of QNBAD through extensive experiments on four IBM quantum devices. Across six benchmark applications, QNBAD increases the absolute error by a factor of 1.68× to 11.7×, depending on the platform. Moreover, the attack remains robust across various ZNE fitting functions and extrapolation techniques.

XI. ACKNOWLEDGMENTS

This work was supported in part by NSF CCF-2105972, NSF OAC-2417589, and NSF CAREER AWARD CNS-2143120.

References

[1] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.

[2] K. Temme, S. Bravyi, and J. M. Gambetta, "Error mitigation for short-depth quantum circuits," *Physical review letters*, vol. 119, no. 18, p. 180509, 2017.

[3] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature communications*, vol. 5, no. 1, p. 4213, 2014.

[4] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, "The theory of variational hybrid quantum-classical algorithms," *New Journal of Physics*, vol. 18, no. 2, p. 023023, 2016.

[5] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *nature*, vol. 549, no. 7671, pp. 242–246, 2017.

[6] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.

[7] G. E. Crooks, "Performance of the quantum approximate optimization algorithm on the maximum cut problem," *arXiv preprint arXiv:1811.08419*, 2018.

[8] C. Chu, N.-H. Chia, L. Jiang, and F. Chen, "Qmlp: An error-tolerant nonlinear quantum mlp architecture using parameterized two-qubit gates," in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design*, 2022, pp. 1–6.

[9] C. Chu, G. Skipper, M. Swany, and F. Chen, "Iqgan: Robust quantum generative adversarial network for image synthesis on nisq devices," in *ICASSP 2023-2023 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[10] C. Chu, A. Hastak, and F. Chen, "Lstm-qgan: Scalable nisq generative adversarial network," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5.

[11] M. Schuld and F. Petruccione, "Supervised learning with quantum computers," *Quantum science and technology (Springer, 2018)*, 2018.

[12] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature communications*, vol. 9, no. 1, p. 4812, 2018.

[13] M. Larocca, S. Thanasilp, S. Wang, K. Sharma, J. Biamonte, P. J. Coles, L. Cincio, J. R. McClean, Z. Holmes, and M. Cerezo, "Barren plateaus in variational quantum computing," *Nature Reviews Physics*, pp. 1–16, 2025.

[14] ——, "A review of barren plateaus in variational quantum computing," *arXiv preprint arXiv:2405.00781*, 2024.

[15] R. Shaydulin, P. C. Lotshaw, J. Larson, J. Ostrowski, and T. S. Humble, "Parameter transfer for quantum approximate optimization of weighted maxcut," *ACM Transactions on Quantum Computing*, vol. 4, no. 3, pp. 1–15, 2023.

[16] M. Skogh, O. Leinonen, P. Lolur, and M. Rahm, "Accelerating variational quantum eigensolver convergence using parameter transfer," *Electronic Structure*, vol. 5, no. 3, p. 035002, 2023.

[17] S. H. Sureshbabu, D. Herman, R. Shaydulin, J. Basso, S. Chakrabarti, Y. Sun, and M. Pistoia, "Parameter setting in quantum approximate optimization of weighted problems," *Quantum*, vol. 8, p. 1231, 2024.

[18] R. Shaydulin, K. Marwaha, J. Wurtz, and P. C. Lotshaw, "QAOAKit: A toolkit for reproducible study, application, and verification of the QAOA," in *2021 IEEE/ACM Second International Workshop on Quantum Computing Software (QCS)*. IEEE, Nov. 2021. [Online]. Available: https://doi.org/10.1109/qcs54837.2021.00011

[19] R. Shaydulin, I. Safro, and J. Larson, "Multistart methods for quantum approximate optimization," in *2019 IEEE high performance extreme computing conference (HPEC)*. IEEE, 2019, pp. 1–8.

[20] A. Galda, X. Liu, D. Lykov, Y. Alexeev, and I. Safro, "Transferability of optimal qaoa parameters between random graphs," in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2021, pp. 171–180.

[21] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari, and W. J. Zeng, "Digital zero noise extrapolation for quantum error mitigation," in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, 2020, pp. 306–316.

[22] A. He, B. Nachman, W. A. de Jong, and C. W. Bauer, "Zero-noise extrapolation for quantum-gate error mitigation with identity insertions," *Physical Review A*, vol. 102, no. 1, p. 012426, 2020.

[23] A. Javadi-Abhari, M. Treinish, K. Krsulich, C. J. Wood, J. Lishman, J. Gacon, S. Martiel, P. D. Nation, L. S. Bishop, A. W. Cross, B. R. Johnson, and J. M. Gambetta, "Quantum computing with Qiskit," 2024.

[24] R. LaRose, A. Mari, S. Kaiser, P. J. Karalekas, A. A. Alves, P. Czarnik, M. El Mandouh, M. H. Gordon, Y. Hindy, A. Robertson *et al.*, "Mitiq: A software package for error mitigation on noisy quantum computers," *Quantum*, vol. 6, p. 774, 2022.

[25] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, S. Ahmed, V. Ajith, M. S. Alam, G. Alonso-Linaje, B. AkashNarayanan, A. Asadi *et al.*, "Pennylane: Automatic differentiation of hybrid quantum-classical computations," *arXiv preprint arXiv:1811.04968*, 2018.

[26] W. Li, Z. Yin, X. Li, D. Ma, S. Yi, Z. Zhang, C. Zou, K. Bu, M. Dai, J. Yue *et al.*, "A hybrid quantum computing pipeline for real world drug discovery," *Scientific Reports*, vol. 14, no. 1, p. 16942, 2024.

[27] D. J. Egger, C. Gambella, J. Marecek, S. McFaddin, M. Mevissen, R. Raymond, A. Simonetto, S. Woerner, and E. Yndurain, "Quantum computing for finance: State-of-the-art and future prospects," *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–24, 2020.

[28] D. J. Egger, R. G. Gutiérrez, J. C. Mestre, and S. Woerner, "Credit risk analysis using quantum computers," *IEEE transactions on computers*, vol. 70, no. 12, pp. 2136–2145, 2020.

[29] S. McArdle *et al.*, "Quantum computational chemistry," *Reviews of Modern Physics*, vol. 92, no. 1, p. 015003, 2020.

[30] P. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding *et al.*, "Scalable quantum simulation of molecular energies," *Physical Review X*, vol. 6, no. 3, p. 031007, 2016.

[31] J. Romero, R. Babbush, J. R. McClean, C. Hempel, P. J. Love, and A. Aspuru-Guzik, "Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz," *Quantum Science and Technology*, vol. 4, no. 1, p. 014008, 2018.

[32] S. E. Lazic and D. P. Williams, "Quantifying sources of uncertainty in drug discovery predictions with probabilistic models," *Artificial Intelligence in the Life Sciences*, vol. 1, p. 100004, 2021.

[33] S. Martín-Santamaría, *Computational tools for chemical biology*. Royal Society of Chemistry, 2017, vol. 3.

[34] J. C. Duarte, R. D. da Rocha, and I. Borges, "Which molecular properties determine the impact sensitivity of an explosive? a machine learning quantitative investigation of nitroaromatic explosives," *Physical Chemistry Chemical Physics*, vol. 25, no. 9, pp. 6877–6890, 2023.

[35] C. Xu, F. Erata, and J. Szefer, "Exploration of power side-channel vulnerabilities in quantum computer controllers," in *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, 2023, pp. 579–593.

[36] F. Chen, L. Jiang, H. Müller, P. Richerme, C. Chu, Z. Fu, and M. Yang, "Nisq quantum computing: A security-centric tutorial and survey [feature]," *IEEE Circuits and Systems Magazine*, vol. 24, no. 1, pp. 14–32, 2024.

[37] C. Chu, L. Jiang, and F. Chen, "Cryptoqfl: quantum federated learning on encrypted data," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1. IEEE, 2023, pp. 1231–1237.

[38] Z. Fu, M. Yang, C. Chu, Y. Xu, G. Huang, and F. Chen, "Quantumleak: Stealing quantum neural networks from cloud-based nisq machines," in *2024 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2024, pp. 1–8.

[39] C. Chu, L. Jiang, and F. Chen, "Bvqc: A backdoor-style watermarking scheme for variational quantum circuits," *arXiv preprint arXiv:2508.01893*, 2025.

[40] C. Chu, L. Jiang, M. Swany, and F. Chen, "Qtrojan: A circuit backdoor against quantum neural networks," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[41] S. Das and S. Ghosh, "Trojan attacks on variational quantum circuits and countermeasures," in *2024 25th International Symposium on Quality Electronic Design (ISQED)*. IEEE, 2024, pp. 1–8.

[42] ——, "Randomized reversible gate-based obfuscation for secured compilation of quantum circuit," *arXiv preprint arXiv:2305.01133*, 2023.

[43] C. Chu, F. Chen, P. Richerme, and L. Jiang, "Qdoor: Exploiting approximate synthesis for backdoor attacks in quantum neural networks," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1. IEEE, 2023, pp. 1098–1106.

[44] J. Guo, W. Jiang, R. Zhang, W. Fan, J. Li, G. Lu, and H. Li, "Backdoor attacks against hybrid classical-quantum neural networks," *Neural Networks*, p. 107776, 2025.

[45] D. A. Lidar, I. L. Chuang, and K. B. Whaley, "Decoherence-free subspaces for quantum computation," *Physical Review Letters*, vol. 81, no. 12, p. 2594, 1998.

[46] G. Ithier, E. Collin, P. Joyez, P. Meeson, D. Vion, D. Esteve, F. Chiarello, A. Shnirman, Y. Makhlin, J. Schriefl *et al.*, "Decoherence in a superconducting quantum bit circuit," *Physical Review B—Condensed Matter and Materials Physics*, vol. 72, no. 13, p. 134519, 2005.

[47] M. Smith, A. Leu, K. Miyanishi, M. Gely, and D. Lucas, "Single-qubit gates with errors at the 10-7 level," *Physical Review Letters*, vol. 134, no. 23, p. 230601, 2025.

[48] P. D. Nation, H. Kang, N. Sundaresan, and J. M. Gambetta, "Scalable mitigation of measurement errors on quantum computers," *PRX Quantum*, vol. 2, no. 4, p. 040326, 2021.

[49] S. S. Tannu and M. K. Qureshi, "Mitigating measurement errors in quantum computers by exploiting state-dependent bias," in *Proceedings of the 52nd annual IEEE/ACM international symposium on microarchitecture*, 2019, pp. 279–290.

[50] M. Sarovar, T. Proctor, K. Rudinger, K. Young, E. Nielsen, and R. Blume-Kohout, "Detecting crosstalk errors in quantum information processors," *Quantum*, vol. 4, p. 321, 2020.

[51] E. Knill, "Quantum computing with realistically noisy devices," *Nature*, vol. 434, no. 7029, pp. 39–44, 2005.

[52] G. Li, Y. Ding, and Y. Xie, "Tackling the qubit mapping problem for nisq-era quantum devices," in *Proceedings of the twenty-fourth international conference on architectural support for programming languages and operating systems*, 2019, pp. 1001–1014.

[53] F. T. Chong, D. Franklin, and M. Martonosi, "Programming languages and compiler design for realistic quantum hardware," *Nature*, vol. 549, no. 7671, pp. 180–187, 2017.

[54] C. Chu, Z. Fu, Y. Xu, G. Huang, H. Muller, F. Chen, and L. Jiang, "Titan: A fast and distributed large-scale trapped-ion nisq computer," in *Proceedings of the 61st ACM/IEEE Design Automation Conference*, 2024, pp. 1–6.

[55] P. Murali, J. M. Baker, A. Javadi-Abhari, F. T. Chong, and M. Martonosi, "Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers," in *Proceedings of the twenty-fourth international conference on architectural support for programming languages and operating systems*, 2019, pp. 1015–1029.

[56] V. Russo and A. Mari, "Quantum error mitigation by layerwise richardson extrapolation," *Physical Review A*, vol. 110, no. 6, p. 062420, 2024.

[57] H. Wang, J. Gu, Y. Ding, Z. Li, F. T. Chong, D. Z. Pan, and S. Han, "Quantumnat: quantum noise-aware training with noise injection, quantization and normalization," in *Proceedings of the 59th ACM/IEEE design automation conference*, 2022, pp. 1–6.

[58] H. Wang, J. Gu, Y. Ding, Z. Li, F. Chong, D. Z. Pan, and S. Han, "Roqnn: Noise-aware training for robust quantum neural networks," 2021.

[59] N. H. Nguyen, E. C. Behrman, and J. E. Steck, "Quantum learning with noise and decoherence: a robust quantum neural network," *Quantum Machine Intelligence*, vol. 2, no. 1, p. 1, 2020.

[60] E. Fontana, N. Fitzpatrick, D. M. Ramo, R. Duncan, and I. Rungger, "Evaluating the noise resilience of variational quantum algorithms," *Physical Review A*, vol. 104, no. 2, p. 022403, 2021.

[61] H. Wang, Y. Liu, P. Liu, J. Gu, Z. Li, Z. Liang, J. Cheng, Y. Ding, X. Qian, Y. Shi *et al.*, "Robuststate: Boosting fidelity of quantum state preparation via noise-aware variational training," *arXiv preprint arXiv:2311.16035*, 2023.

[62] U. Azad, "Pennylane quantum chemistry datasets," https://pennylane.ai/datasets/qchem/oh–molecule, 2023.

[63] N. P. Sawaya *et al.*, "Hamlib: A library of hamiltonians for benchmarking quantum algorithms and hardware," 2023.

[64] P. Jordan *et al.*, *Über das paulische äquivalenzverbot*. Springer, 1993.

[65] J. Tilly *et al.*, "The variational quantum eigensolver: a review of methods and best practices," *Physics Reports*, vol. 986, pp. 1–128, 2022.

[66] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.

[67] O. Higgott, D. Wang, and S. Brierley, "Variational quantum computation of excited states," *Quantum*, vol. 3, p. 156, 2019.

[68] H. Wang *et al.*, "Quantumnas: Noise-adaptive search for robust quantum circuits," in *The 28th IEEE International Symposium on High-Performance Computer Architecture (HPCA-28)*, 2022.

[69] A. Lowe, M. H. Gordon, P. Czarnik, A. Arrasmith, P. J. Coles, and L. Cincio, "Unified approach to data-driven quantum error mitigation," *Physical Review Research*, vol. 3, no. 3, p. 033098, 2021.

[70] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *International symposium on research in attacks, intrusions, and defenses*. Springer, 2018, pp. 273–294.

[71] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, and N. Killoran, "Evaluating analytic gradients on quantum hardware," *Physical Review A*, vol. 99, no. 3, p. 032331, 2019.

[72] J. J. Wallman and J. Emerson, "Noise tailoring for scalable quantum computation via randomized compiling," *Physical Review A*, vol. 94, no. 5, p. 052325, 2016.

[73] IBMQ, "Error mitigation and suppression techniques," https://quantum.cloud.ibm.com/docs/en/guides/error-mitigation-and-suppression-techniques.