# ResilienQ: Boosting Fidelity of Quantum State Preparation via Noise-Aware Variational Training

Hanrui Wang*[1], Yilian Liu*[2], Pengyu Liu*[3], Song Han[1]
[1]MIT [2]Cornell University [3]CMU

***Abstract*—Quantum state preparation, a crucial subroutine in quantum computing, involves generating a target quantum state from initialized qubits. Arbitrary state preparations can be broadly categorized into arithmetic decomposition (AD) and variational quantum state preparation (VQSP). AD employs a predefined procedure to decompose the target state into a series of gates, whereas VQSP iteratively tunes ansatz parameters to approximate target state. VQSP is particularly apt for Noisy-Intermediate Scale Quantum (NISQ) machines due to its shorter circuits. We present ResilienQ, a novel VQSP methodology that combines high robustness with high training efficiency. The core idea involves utilizing measurement outcomes from real machines to perform back-propagation through classical simulators, thus incorporating real quantum noise into gradient calculations. ResilienQ serves as a versatile, plug-and-play technique applicable for training parameters from scratch or fine-tuning existing parameters to enhance fidelity on target machines. It is adaptable to various ansatzes at both gate and pulse levels and can even benefit other variational algorithms, such as variational unitary synthesis. Comprehensive evaluation of ResilienQ on state preparation tasks for 4 distinct quantum algorithms using 10 real quantum machines demonstrates a coherent error reduction of up to 7.1 × and state fidelity improvement of up to 96% and 81% for 4-Q and 5-Q states, respectively. On average, ResilienQ improves fidelity by 50% and 72% for 4-Q and 5-Q states compared to baseline approaches.**

## I. Introduction

Quantum state preparation facilitates the preparation of the system's initial state. This process is essential for applications such as codewords in quantum error correction [16], amplitude encoding [30] in quantum machine learning, and initial condition loading for solving Partial Differential Equations (PDEs) using quantum machines [13], [24]. State preparation in quantum computing is mainly achieved via arithmetic decomposition (AD) or variational quantum state preparation (VQSP). AD methods like Shannon decomposition [25] and Mottonen decomposition [29] construct quantum circuits to realize target states, each employing distinct strategies of classical-to-quantum translation and hierarchical qubit rotations, respectively. Conversely, VQSP [2], [6] minimizes the distance between the target and final states by iteratively updating variational circuit parameters in a variational circuit ansatz.

On real quantum machines, AD methods can be significantly affected by quantum noise. Utilizing IBM Qiskit [1] compiler for Shannon decomposition, we generate circuits that transform the all-zero state into three distinct quantum states and compare the average fidelity on noise-free simulators and real quantum machines (Fig. 2). Although the compiler can achieve 100% fidelity in noise-free simulations, real quantum machines experience considerable fidelity degradation. This fidelity gap intensifies with an increasing number of qubits, e.g., while 2-qubit (2-Q) states achieve 98% fidelity on real machines, 6-qubit (6-Q) states have a mere 1% fidelity. Conversely, VQSP is better suited for NISQ devices due to its flexibility in mitigating coherent errors by adjusting circuit parameters and its reduced 2-qubit gate count and depth.

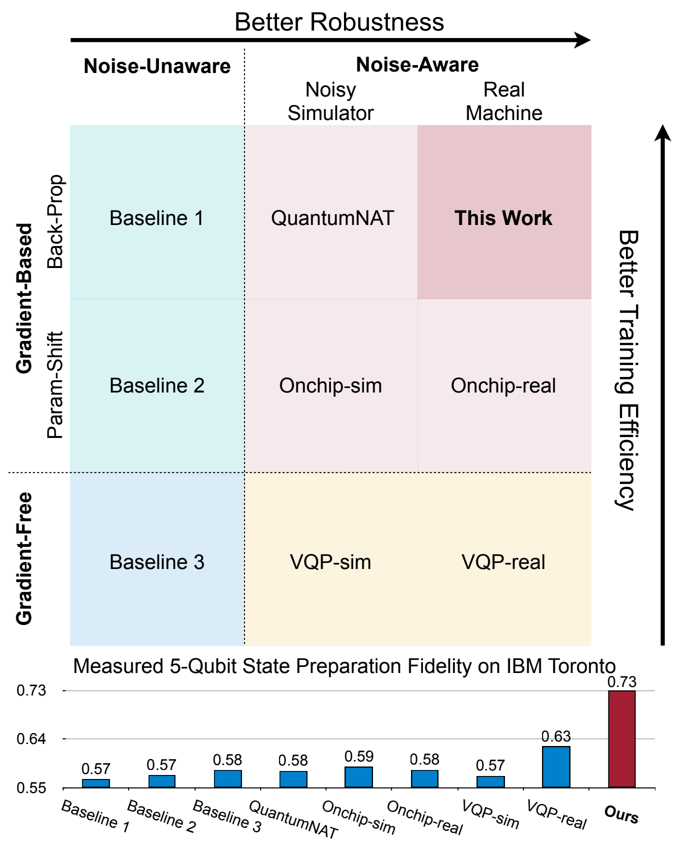


Fig. 1. Proposed ResilienQ performs back-propagation training using results from real quantum machines, achieving high robustness and training efficiency. State preparation fidelity of each method is evaluated on real machines.

However, optimizing VQSP parameters for *noise robustness* is challenging, as highlighted in Fig.1 for a 5-Q quantum state on the IBM Toronto machine. While noise-free fidelities can exceed 99%, real-machine performances suffer. First-column methods, relying on noise-unaware simulators, are severely impacted by real-device noise. Second-column approaches, using noisy simulators such as QuantumNAT [38], onchip-

*Equal Contributions

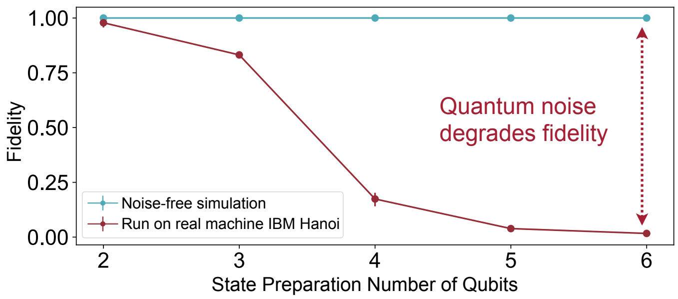


Fig. 2. Prepared state fidelity on a noise-free simulator and real machine IBM Hanoi. (i) Noise-free fidelity is very close to 100%. (ii) Noise significantly reduces fidelity, especially for large qubit numbers.

sim [39], and VQP-sim [23], also yield low fidelity due to device-simulator discrepancies. However, these simulators achieve higher training efficiency through *single* backward pass gradient computation. Real-machine-based methods in the rightmost column, whether gradient-free like Bayesian optimization or gradient-based following the PS rule, exhibit low training efficiency. Specifically, gradient-free methods lack precise gradient guidance, while the PS rule incurs a computational cost of $\mathcal{O}(\#\text{params})$ for each update. The natural goal of achieving both high robustness and training efficiency calls for *noise-aware back-propagation*. However, due to the No-Cloning Theorem [8], [41], back-propagation on real quantum machines is infeasible, as intermediate results (quantum states) cannot be stored for use in the backward pass.

To address this challenge, we introduce ResilienQ. The core idea hinges on the fact that while real quantum machines cannot provide intermediate results, they can supply *final results* through measurements (tomography). Thus, we can use final results from real quantum machines combined with intermediate results from classical simulators to complete the backward pass. In a single training step, the same set of parameters is executed on *both* real quantum machines and simulators. The loss function is computed between the tomography states from a real machine and the target state. Subsequently, noise-impacted gradients are back-propagated through the simulator using the previously simulated intermediate results, ensuring noise resilience in the trained parameters. Fig. 1 highlights the superior accuracy of our method over alternatives. Our results demonstrate that ResilienQ improves 4-Q state fidelity by **50%** and 5-Q state fidelity by **72%** on average, reducing coherent errors by up to **7.1×** when compared to noise-unaware baselines, and achieving high scalability. ResilienQ provides a versatile approach for enhancing ansatz fidelity across platforms like Xanadu [24] and QuantumNAS [37]. Unlike conventional PS rules requiring specific gate structures [28], ResilienQ uses a simulator for gradient computation, ensuring applicability to any classically simulatable ansatz and improving *both gate and pulse* ansatzes.

## II. Related Work

**Quantum State Preparation.** Grover and Rudolph initially introduced the quantum state preparation problem for efficiently generating integrable probability distribution functions [14].

---

**Algorithm 1:** ResilienQ Training with Noise-Aware Gradient Back-Propagation

**Input :** Training objective $\mathcal{L}$, quantum real machine execution function $f(\cdot)$, classical simulation function $f'(\cdot)$, initial parameters $\theta^0 \in \mathbb{R}^n$, initial learning rate $\eta^0$, and total steps $T$.

$\eta \leftarrow \eta^0$;

**for** $t = 0, 1, \cdots, T-1$ **do**

  Execute circuit on real quantum machine
  $\rho = f(\theta^t)$;
  Simulate circuit on classical simulator
  $\rho' = f'(\theta^t)$;
  Objective evaluation with noisy output: $\mathcal{L}(\rho)$;
  Classical backpropagation to obtain noisy gradients
  $\nabla_{\theta^t}\mathcal{L}(\rho) = \frac{\partial \mathcal{L}(\rho)}{\partial \rho}\frac{\partial \rho'}{\partial \theta^t}$;
  Parameter update:
  $\theta^{t+1} \leftarrow \theta^t - \eta \nabla_{\theta^t}\mathcal{L}(\rho)$;

**end**

**Output :** Converged parameters $\theta^{T-1}$

---

Numerous state preparation techniques have been proposed since, targeting specific states like Gaussian wave functions [3], [19], [33], continuous functions [15], [32], and arbitrary functions [26], [31], [36], [40], [43]. State preparation techniques can be categorized into Arithmetic Decomposition (AD) and Variational Quantum State Preparation (VQSP). AD utilizes rule-based algorithms to generate a circuit mapping the $|0\rangle$ state to the target state $|\psi\rangle$ in one step, while VQSP iteratively refines a circuit to minimize the difference between the produced and target states. VQSP begins with designing a parameterized circuit architecture (called ansatz). The circuit realizes a parameterized unitary $U(\theta)$, preparing a state: $|\psi(\theta)\rangle = U(\theta)\,|0\ldots0\rangle$, where and $\theta$ is a set of free parameters that are trained in a hybrid quantum-classical optimization procedure iteratively to minimize loss.

**Pulse-Level Quantum Computing.** Quantum Optimal Control and variational pulse learning [5], [7], [11], [18], [23], [27], [35] have gained interest recenlty. Quantum Optimal Control iteratively adjusts pulse shapes to minimize unitary-target discrepancies [5], [7], [35]. Variational pulse learning optimizes pulse parameters, such as amplitude and frequency, in lieu of rotation gate angles [18], [23], [27]. Unlike [11], which reduces circuit depth via gate set compilation, we mitigate coherent errors induced by direct pulse shape modifications.

## III. Noise-Aware ResilienQ Methodology

**Noise-Aware Back-Propagation.** Although variational circuits generally have smaller depth and better fidelity (details at Fig. 7), they necessitate iterative parameter training. As discussed in Sec. I, noise-aware back-propagation is the optimal candidate for achieving high robustness and training efficiency. However, real quantum machines cannot perform back-propagation due to quantum mechanics' fundamental limits, specifically the No-Cloning Theorem [8], [41], which prevents storing intermediate results necessary for back-propagation.

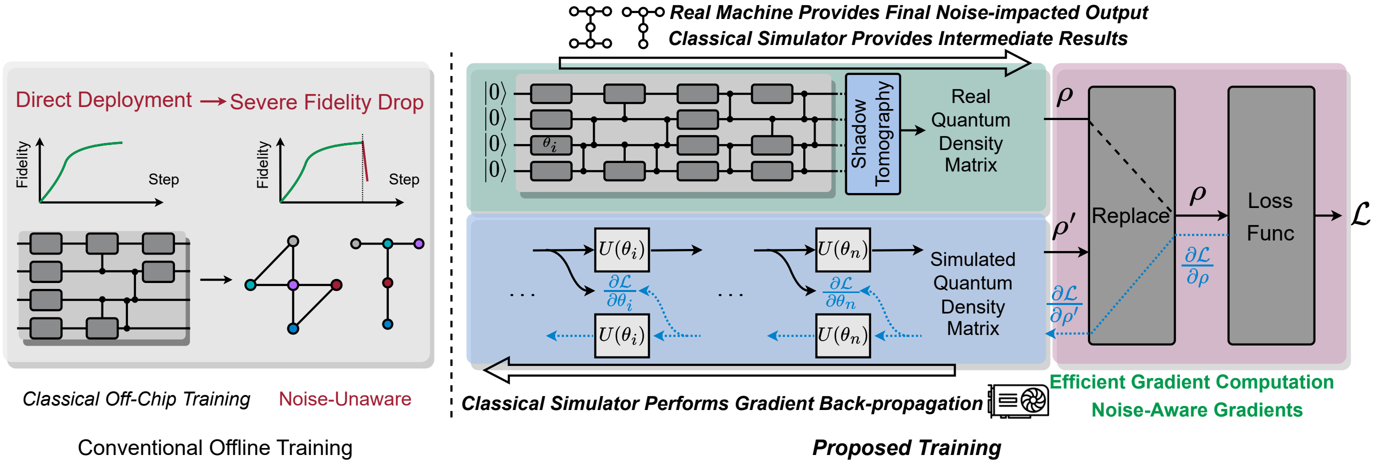


Fig. 3. ResilienQ with real noise-aware gradients improves parameter robustness.

Fortunately, the final results (density matrix) of quantum circuits can be obtained through measurements, providing abundant noise information. Inspired by [42], we employ a differentiable classical simulator to acquire intermediate results (quantum states) so that back-propagation can be performed using the noisy final output and simulated intermediate results. In Fig. 3 and Alg. 1, we outline the ResilienQ training protocol, which combines quantum on-chip forward with classical simulated backward propagation for noise-aware gradient-based optimization. In each iteration, we first execute the quantum circuits on the real quantum device and perform tomography to obtain the density matrix $\rho$, which incorporates the effects of real noises. Next, we simulate the quantum circuit on classical computers to obtain a noiseless density matrix $\rho'$. To adjust parameters based on a specific machine's noise, we *replace* the noiseless density matrix $\rho'$ with the noisy one $\rho$ to evaluate the loss function $\mathscr{L}(\rho)$.

During back-prop, we adopt straight-through estimator (STE) that directly passes the noisy gradient $\frac{\partial \mathscr{L}(\rho)}{\partial \rho}$ to the theoretical noise-free path, i.e., $\frac{\partial \mathscr{L}(\rho)}{\partial \rho} \rightarrow \frac{\partial \mathscr{L}(\rho)}{\partial \rho'}$. Then, this estimated noisy gradient will be used to calculate derivatives for all parameters $\frac{\partial \mathscr{L}(\rho)}{\partial \theta} = \frac{\partial \mathscr{L}(\rho)}{\partial \rho'} \frac{\partial \rho'}{\partial \theta}$. The fundamental reason why this gradient replacement works is that the quantum noise information can be effectively coupled in the back-propagation procedure, i.e., the noisy upstream gradient $\frac{\partial \mathscr{L}(\rho)}{\partial \rho}$, to make the training process fully aware of real quantum noises. Note that this property requires the objective $\mathscr{L}(\cdot)$ to be a nonlinear function of the noisy $\rho$. Otherwise, $\frac{\partial \mathscr{L}(\rho)}{\partial \rho}$ will only contain noise-free terms. This methodology synergistically leverages the noise awareness of real quantum machines and the differentiability of classical simulators for parallel, highly efficient noise-aware training.

We will illustrate this hybrid training idea with a concrete example. In our experiments, we choose the loss function to be $\mathscr{L} = \sqrt{\mathrm{tr}((\rho - \hat{\rho})^2)}$, then $\nabla_\theta \mathscr{L} = \mathrm{tr}\left(\frac{(\rho - \hat{\rho})}{\mathscr{L}} \frac{\partial \rho'}{\partial \theta}\right)$. After the parameter update, the state becomes

$$\rho_{t+1} = \rho_t - \sum_\theta \frac{\eta}{\mathscr{L}_t} \mathrm{tr}\left((\rho_t - \hat{\rho}) \frac{\partial \rho'}{\partial \theta_t}\right) \frac{\partial \rho}{\partial \theta_t} + O(\eta^2), \quad (1)$$

then the loss function $\mathscr{L}_{t+1}^2 - \mathscr{L}_t^2$ becomes

$$-2 \sum_\theta \frac{\eta}{\mathscr{L}_t} \mathrm{tr}\left((\rho_t - \hat{\rho}) \frac{\partial \rho'}{\partial \theta_t}\right) \mathrm{tr}\left((\rho_t - \hat{\rho}) \frac{\partial \rho}{\partial \theta_t}\right) + O(\eta^2). \quad (2)$$

As long as $\frac{\partial \rho}{\partial \theta_t} \approx \frac{\partial \rho'}{\partial \theta_t}$ and the learning rate $\eta$ is small enough, the two $\mathrm{tr}(\cdot)$ operators will have the same sign with high probability and $\mathscr{L}_{t+1} < \mathscr{L}_t$.

We further compare hybrid training approximated gradients with the real gradients estimated by parameter shift to show experimental evidence of our method's effectiveness. We build a 3-Q ansatz with 2 `RX` gates on qubits 0 and 2 and an `RY` gate on qubit 1, followed by three `RZX` gates connecting qubits 0 and 1, 1 and 2, 2 and 0. After each training step, we compare the cosine similarity of gradients between the two methods and we find that the similarities are always higher than 0.95. Therefore, our method can provide accurate noise-aware gradients while reducing the $\mathscr{O}$(#parameters) circuit executions in parameter shift to $\mathscr{O}(1)$ complexity per round.

The ResilienQ is a plug-and-play approach. We note that our noise-aware gradients may not be necessary at the beginning of the training when parameters are far from convergence. Thus, in real experiments, we can combine this noise-aware training with pure classical training by first training the parameters to convergence in a simulator and then performing parameter fine-tuning using noise-aware training to reduce the cost of running on real machines. Furthermore, by replacing the loss function, our method can be used for other variational quantum algorithms to further boost performance.

**Hardware Efficient and Pulse-Level Ansatz of ResilienQ.** As discussed in the previous subsection, ResilienQ requires no assumption on the ansatz structure. It applies to all kinds of quantum operations with analytical formulations, while the parameter shift rule is only narrowly applicable to gates whose unitary has a structured eigenvalue [28]. Therefore, to avoid the additional overhead when compile the circuit to actual hardware, we firstly propose to use *hardware-efficient ansatz* which respects the hardware connectivity map as in Fig. 4(a) and (b). The gates are also the native gates on hardware.

We further propose *pulse-level ansatz*. Pulse-level control

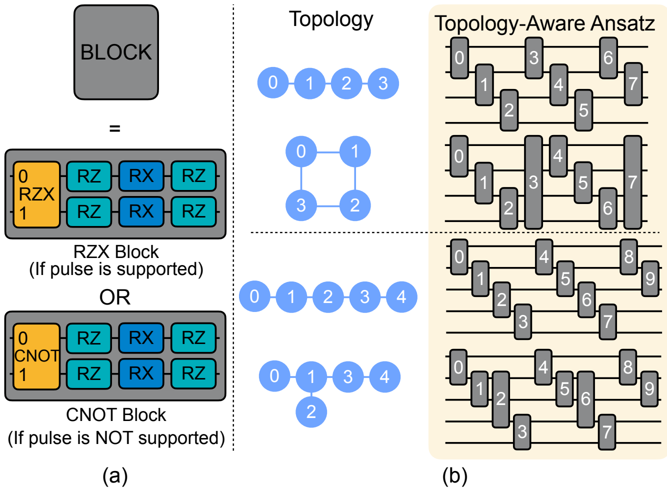


Fig. 4. (a) The basic 2-Q *block* used in the hardware efficient ansatz. The 2-Q gates and 1-Q gates are chosen based on the native gates of the given hardware. (b) The proposed topology-aware ansatz design.

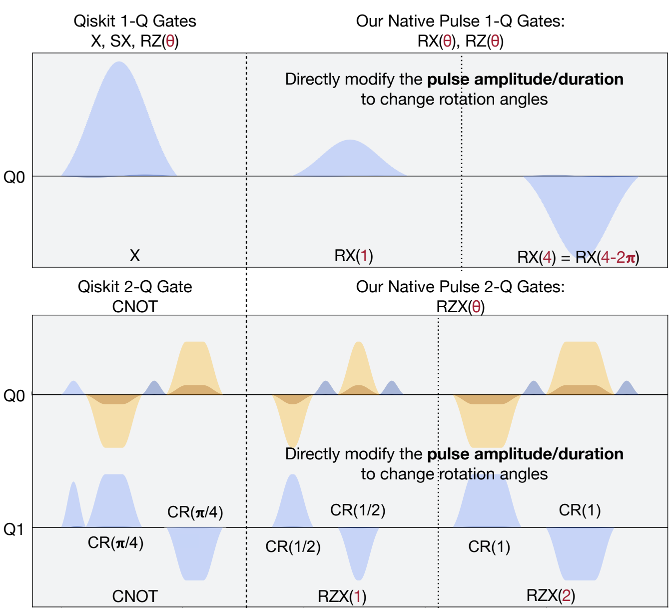


Fig. 5. Pulse schedules for X, RX (1) and RX (4) (top); CNOT, RZX (1) and RZX (2) (bottom). For IBM's quantum computers, calibration is only performed for X, SX, and CNOT, so we need to adjust the pulse shape to generate RX ($\theta$) and RZX ($\theta$) for arbitrary angles.

introduces more parameters compared to gates, which enables abundant opportunities for a more compact ansatz [11]. We can generate two more parameterized basis gates, the RX($\theta$) and RZX($\theta$) gates which are shown in Fig. 5, without any calibration cost using pulse level control. To implement RX ($\theta$) from existing calibration data, we retrieve the pulse shape of the pre-calibrated X gate and adjust the pulse amplitude by setting the area under the curve proportional to $\theta$. According to the basic principle of quantum dynamics, it is an approximation of RX ($\theta$). Similarly, we retrieve the pulse shape of the pre-calibrated CR, and adjust the area under it to implement RZX($\theta$) for any $\theta$. Using a native pulse gate set has several benefits. An arbitrary single qubit rotation will be decomposed to up to 5 native gates for IBM's default implementation

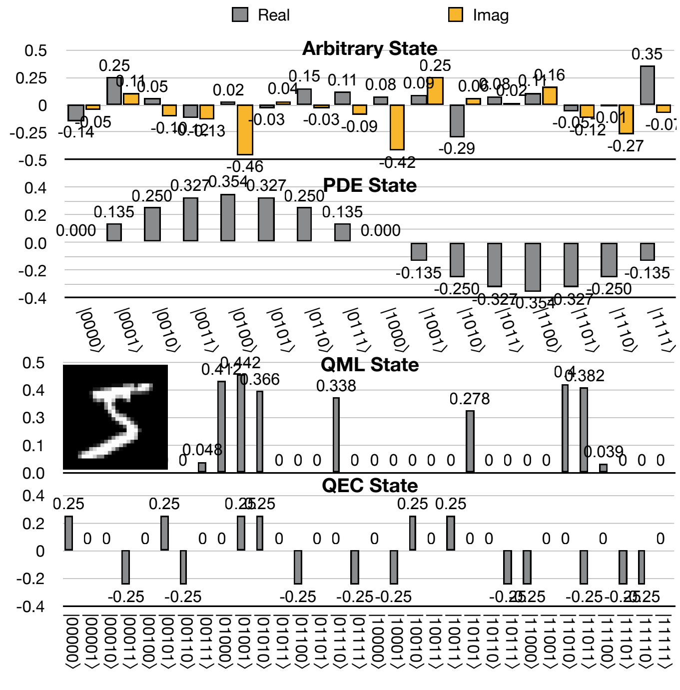


Fig. 6. Visualization of several target states.

(RZ,SX,RZ,SX,RZ). However, with native pulse gates, we can reduce this number to 3 (RZ,RX,RZ) [12]. With regards to RZ gates, the entanglement operation between qubits can be precisely controlled. Thus, the circuit requires shorter runtime and has more parameters with better expressivity.

Changing the amplitude of the pre-calibrated pulse only provides an approximation of rotation gates. Due to the imperfection of classical control and the influence of higher energy-level in superconducting quantum computers, the system is not entirely linear [9]. For example, the amplitude of SX might not be exactly half that of the X gate. In IBMQ Jakarta, the pulse amplitude of SX and half the amplitude of X has 0.9% difference introducing a detectable coherent error when simply adjusting the pulse proportionally without fine-tuning. Thankfully, our noise-aware training method in III can automatically correct these coherent errors.

## IV. Evaluation

### A. Evaluation Methodology

**Benchmarks.** Four categories of representative target states with 4 and 5 qubits are utilized, encompassing arbitrary states, partial differential equation (PDE) states, quantum machine learning (QML) states, and quantum error correction (QEC) codewords. Arbitrary states are generated following the uniform (Haar) measure. PDE states are real-valued states encoded into the amplitude of the basis states that are of interest to studying solving PDE by a variational quantum algorithm, such as the sine wave and the Gaussian distribution. The QML state encodes a classical MNIST hand-writing digit image [21] into the state vector via amplitude encoding [30], with the image being down-sampled, flattened, and normalized as the state. For the 5-qubit scenario, quantum error correction (QEC)

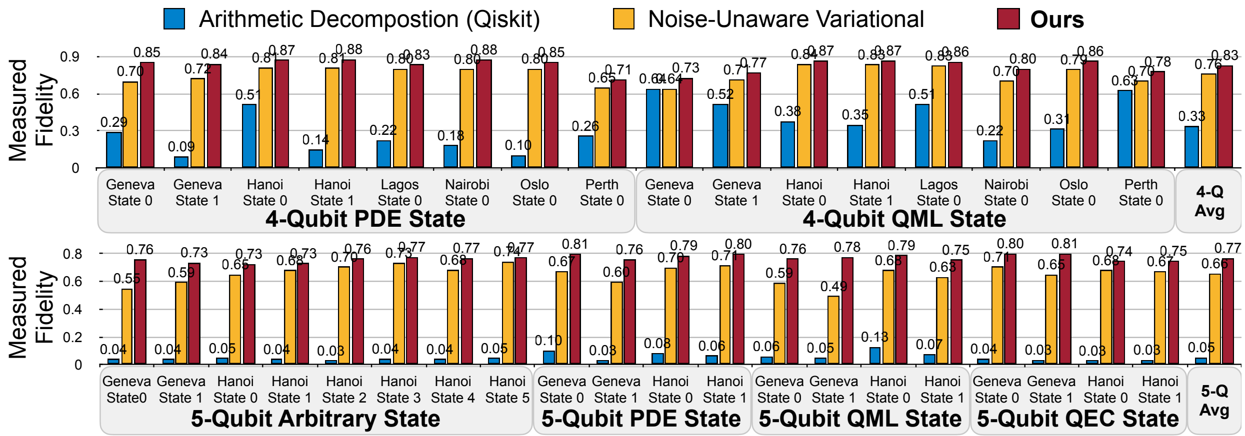


Fig. 7. ResilienQ using topology-aware ansatz with `CNOT` gates achieves the highest fidelity on various real machines for a number 4-Q target states when compared with the Qiskit baseline. Evaluated on real machines.

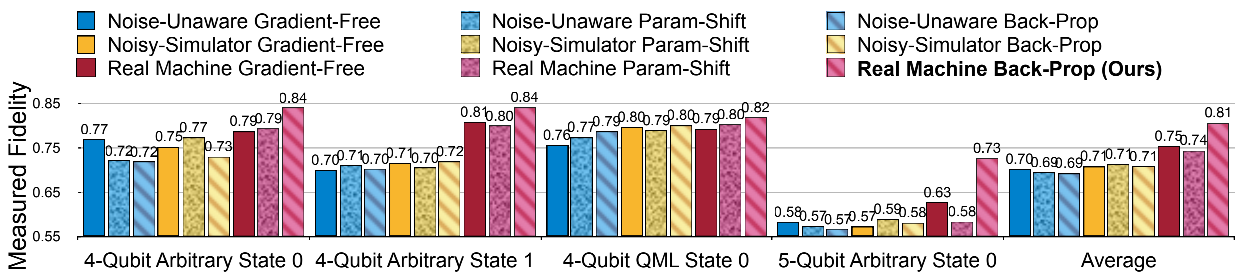


Fig. 8. ResilienQ compared with eight other variational state preparation methods. ResilienQ achieved the highest fidelity for all four tasks on real machines.

codewords for the 5-qubit error correction code are also tested [20]. Fig. 6 shows the amplitude distributions of chosen states.

**Hardware-Efficient and Pulse Ansatz design.** The first step of variational methods is ansatz design. Fig. 4 shows the designs of our hardware-efficient ansatz. The basic unit of the ansatz is a 2-Q *block* shown in Fig. 4(a), inspired by [25]. In our experiments, we use the `RZX` gate as the 2-Q entangling gate on IBM machines that support pulse controls and `CNOT` gate on other IBM quantum machines. The *block*s are only applied on neighboring qubits with direct connections to avoid any additional compilation overhead such as `SWAP` insertions. To generate `RZX`$(\theta)$ gates, we use Qiskit's `RZXCalibrationBuilder` and implement our own `RXCalibrationBuilder` accordingly for `RX`$(\theta)$ gates. In addition to these two additional transpiling arguments, we also specify the virtual to physical qubit mapping, so our topology-aware ansatzes will work as expected. All the other options are set to Qiskit's default.

**Training and Tomography setups.** We use Adam optimizer with learning rate $5\times10^{-3}$. The loss function is $\sqrt{\mathrm{tr}((\rho-\hat{\rho})^2)}$, where $\rho$ is the target density matrix and $\hat{\rho}$ is state generated from the ansatz. We train the ansatz for a total of 550 steps. For the first 500 steps, $\hat{\rho}$ is obtained from a classical simulator, so the training is noise unaware; for the last 50 steps, the training is noise aware, and $\hat{\rho}$ is obtained from the tomography results. Unless otherwise stated, we use classical shadow tomography with all the bases measured to improve the accuracy of tomography. That is $3^4$ bases for 4-Q states and $3^5$ bases for 5-Q states. For each basis, we repeat for 1024 shots. Due to limited shots, the estimated fidelity has a standard deviation of about 0.006 according to our simulations.

TABLE I
RESILIENQ CAN OUT-PERFORM ARITHMETIC DECOMPOSITION.

| Fidelity | Arbitrary | PDE | QML | Avg. |
|---|---|---|---|---|
| Mottonen [4], [29] | 0.156 | 0.175 | 0.269 | 0.200 |
| Mottonen+SABRE [4], [22], [29] | 0.099 | 0.401 | 0.299 | 0.266 |
| Qiskit [17] | 0.176 | 0.277 | 0.481 | 0.311 |
| Qiskit + SABRE [22] | 0.262 | 0.266 | 0.626 | 0.385 |
| **Ours** | **0.777** | **0.713** | **0.718** | **0.736** |

### B. Experiment Results

**Variational v.s. Arithmetic.** As a baseline, we compare ResilienQ to the `initialize` method provided by Qiskit [1], which is the only integrated arbitrary state preparation method in the Qiskit library. The `initialize` function in Qiskit is implemented based on an analog of Quantum Shannon Decomposition [34]. We transpile the state preparation circuits generated by both methods with the highest `optimization_level=3`.

Fig. 7 shows the measured state preparation fidelity of ResilienQ on 3 kinds of states and 6 quantum machines with no pulse supports for 4 qubits and 5 qubits, respectively. Arithmetic decomposition tests the Qiskit baseline; Noise-unaware VQSP tests the classically trained VQSP, and ResilienQ is trained with real machine noise. For the tested 4-Q states, ResilienQ can

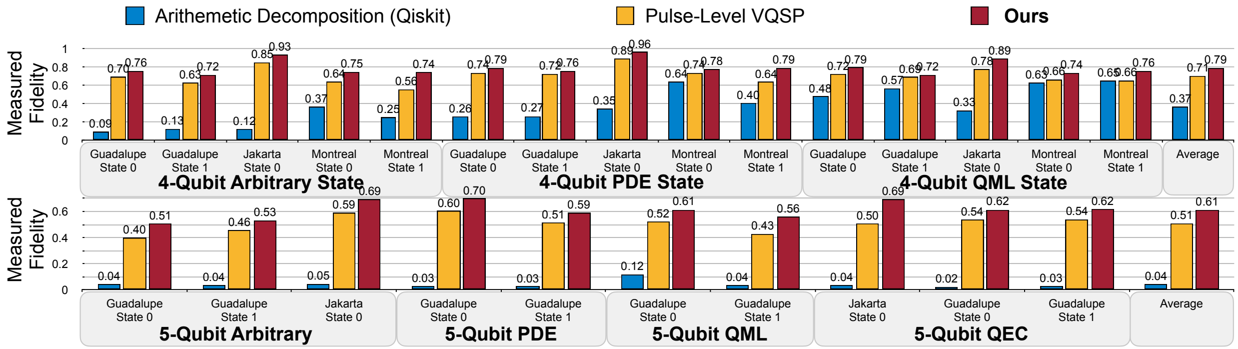


Fig. 9. On three machines, IBMQ Guadalupe, IBMQ Jakarta, and IBMQ Montreal, with pulse supports, ResilienQ using the topology-aware ansatz with `RZX` gates achieves the highest fidelity for various 4-Q states when compared with the Qiskit baseline. Evaluated on real machines.

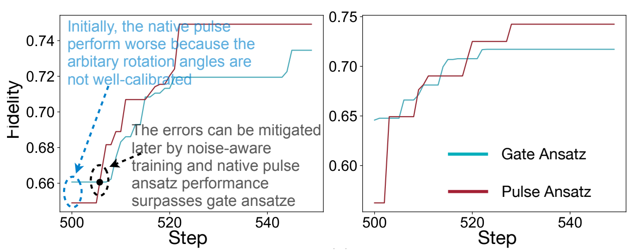


Fig. 10. Training curves for the last 50 steps with the noise-aware loss for hardware-efficient ansatz using native pulse and hardware-efficient ansatz using the non-native `CNOT` gates. The first 500 steps are not included.

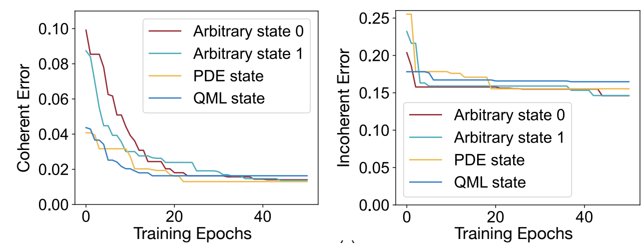


Fig. 11. The coherent and incoherent errors of four state preparation tasks. Our noise-aware training method significantly reduces coherent errors.

achieve 83% fidelity and a 50% improvement on average over the Qiskit baseline. The improvement is even more significant for 5 qubits. The Qiskit baseline averages at 5% fidelity while ResilienQ achieved 77% fidelity – a 72% improvement over the Qiskit baseline. These results demonstrate that, as we speculated in previous sections, VQSP algorithms perform much better than AD ones on NISQ. Moreover, ResilienQ can further improve the fidelity achieved by conventional VQSP.

**Additional arithmetic decomposition algorithm and error mitigation.** We perform additional baseline tests with the Mottonen algorithm [29] implemented by PennyLane [4] as an alternative baseline algorithm to the Qiskit baseline, as well as SWAP-based BidiREctional (SABRE) heuristic search algorithm [22] included by the Qiskit library used to optimize the qubit mapping to reduce `SWAP` gate count. SABRE was implemented on both the Qiskit baseline and the Mottonen baseline. As shown in Table I, ResilienQ outperforms Mottonen and Qiskit on average by 53.6% and 42.5% respectively. Even after applying SABRE, ResilienQ still outperforms Mottonen and Qiskit on average by 47.0% and 35.1%.

**Comparison to other optimization methods.** As introduced in Fig. 1, we compare ResilienQ with all eight other optimization methods. We choose the state-of-the-art optimizer, the Nelder-Mead method provided by scipy with the default setting [10] for gradient-free optimizations, and for parameter shift, we choose the learning rate to be $5^{-3}$. As shown in Fig. 8, ResilienQ achieves the highest fidelity for all four tasks tested and outperforms the second-best variational approach by 6% on average. This result shows that, with the highest training efficiency and noise-robustness, ResilienQ is better than all other variational state preparation.

**Pulse ansatz.** Fig. 9 shows the ResilienQ performance for 4-Q states and 5-Q states on three machines with pulse supports using `RZX` blocks. We use Qiskit as our baseline here as well. ResilienQ on pulse level VQSP improves the average fidelity by 8% and 10% for 4-Q and 5-Q, producing final fidelities of 79% and 61%, respectively. Furthermore, ResilienQ with pulse outperforms the baseline by 42% and 57% for 4-Q and 5-Q, respectively. An interesting phenomenon is that sometimes the noise-free trained circuit compiled to non-native basis gates performs better than that compiled to native pulse gates. This might be caused by the additional coherent error introduced by tuning pulses. Nevertheless, pulse ansatz enables *better optimization space* so the final fidelity can be higher. Fig. 10 shows the fidelity of noise-aware training with gate ansatz v.s. with pulse ansatz for two arbitrary states on the IBMQ Montreal machine. We can clearly see that the fidelity of the pulse ansatz surpasses the non-native gates in the middle of training. This shows that by combining pulse ansatz and the noise-aware gradients together, we can *reduce both coherent and incoherent errors*.

**Where does our advantage come from?** The noise-aware training part of our framework is targeted to eliminate coherent errors by fine-tuning the parameters. To show this, we separate the coherent error from the incoherent error in Fig. 11. For the two states, our method can reduce the coherent errors by at least 62% and up to 86%. For incoherent errors, ResilienQ can reduce it by 7% to 39%.

## V. Conclusion

We present ResilienQ, a noise-aware training framework for robust quantum state preparation. It uniquely blends real-machine noise and simulator data for noise-aware back-propagation, enhancing both parameter robustness and training efficiency. It also leverages hardware-efficient and pulse-level ansatz for higher reliability. Evaluated on 10 real quantum platforms, ResilienQ reduces coherent errors by over 7.1× and improves fidelity by 50% and 72% over baselines, paving the way towards robust state preparation.

## References


[1] M. S. ANIS *et al.*, "Qiskit: An open-source framework for quantum computing," 2021.

[2] J. M. Arrazola, T. R. Bromley, J. Izaac, C. R. Myers, K. Brádler, and N. Killoran, "Machine learning method for state preparation and gate synthesis on photonic quantum computers," *Quantum Science and Technology*, vol. 4, no. 2, p. 024004, 2019.

[3] C. W. Bauer, P. Deliyannis, M. Freytsis, and B. Nachman, "Practical considerations for the preparation of multivariate gaussian states on quantum computers," 2021. [Online]. Available: https://arxiv.org/abs/2109.10918

[4] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, M. S. Alam, S. Ahmed, J. M. Arrazola, C. Blank, A. Delgado, and S. Jahangiri, "Pennylane: Automatic differentiation of hybrid quantum-classical computations," *arXiv preprint arXiv:1811.04968*, 2018.

[5] J. Cheng, H. Deng, and X. Qia, "Accqoc: Accelerating quantum optimal control based pulse generation," in *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 2020, pp. 543–555.

[6] L. Cincio, K. Rudinger, M. Sarovar, and P. J. Coles, "Machine learning of noise-resilient quantum circuits," *PRX Quantum*, vol. 2, no. 1, p. 010324, 2021.

[7] R. de Keijzer, O. Tse, and S. Kokkelmans, "Pulse based variational quantum optimal control for hybrid quantum computing," *arXiv preprint arXiv:2202.08908*, 2022.

[8] D. Dieks, "Communication by epr devices," *Physics Letters A*, vol. 92, no. 6, pp. 271–272, 1982.

[9] J. M. Gambetta, F. Motzoi, S. Merkel, and F. K. Wilhelm, "Analytic control methods for high-fidelity unitary operations in a weakly nonlinear oscillator," *Physical Review A*, vol. 83, no. 1, p. 012308, 2011.

[10] F. Gao and L. Han, "Implementing the nelder-mead simplex algorithm with adaptive parameters," *Computational Optimization and Applications*, vol. 51, no. 1, pp. 259–277, 2012.

[11] P. Gokhale, A. Javadi-Abhari, N. Earnest, Y. Shi, and F. T. Chong, "Optimized quantum compilation for near-term algorithms with open-pulse," in *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2020, pp. 186–200.

[12] P. Gokhale, A. Javadi-Abhari, N. Earnest, Y. Shi, and F. T. Chong, "Optimized quantum compilation for near-term algorithms with open-pulse," in *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2020, pp. 186–200.

[13] J. Gonzalez-Conde, Ángel Rodríguez-Rozas, E. Solano, and M. Sanz, "Simulating option price dynamics with exponential quantum speedup," 2022.

[14] L. Grover and T. Rudolph, "Creating superpositions that correspond to efficiently integrable probability distributions," *arXiv e-prints*, pp. quant–ph/0 208 112, Aug. 2002.

[15] A. Holmes and A. Y. Matsuura, "Efficient quantum circuits for accurate state preparation of smooth, differentiable functions," 2020. [Online]. Available: https://arxiv.org/abs/2005.04351

[16] C. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, "Surface code quantum computing by lattice surgery," *New Journal of Physics*, vol. 14, no. 12, p. 123011, 2012.

[17] Q. IBM, Apr 2021. [Online]. Available: https://qiskit.org/textbook/ch-quantum-hardware/calibrating-qubits-pulse.html

[18] J. R. Johansson, P. D. Nation, and F. Nori, "Qutip: An open-source python framework for the dynamics of open quantum systems," *Computer Physics Communications*, vol. 183, no. 8, pp. 1760–1772, 2012.

[19] A. Kitaev and W. A. Webb, "Wavefunction preparation and resampling using a quantum computer," 2008. [Online]. Available: https://arxiv.org/abs/0801.0342

[20] E. Knill, R. Laflamme, R. Martinez, and C. Negrevergne, "Benchmarking quantum computers: The five-qubit error correcting code," *Physical Review Letters*, vol. 86, no. 25, p. 5811, 2001.

[21] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[22] 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.

[23] Z. Liang, H. Wang, J. Cheng, Y. Ding, H. Ren, X. Qian, S. Han, W. Jiang, and Y. Shi, "Variational quantum pulse learning," *arXiv preprint arXiv:2203.17267*, 2022.

[24] M. Lubasch, J. Joo, P. Moinier, M. Kiffner, and D. Jaksch, "Variational quantum algorithms for nonlinear problems," *Physical Review A*, vol. 101, no. 1, jan 2020. [Online]. Available: https://doi.org/10.1103%2Fphysreva.101.010301

[25] L. Madden and A. Simonetto, "Best approximate quantum compiling problems," *ACM Transactions on Quantum Computing*, vol. 3, no. 2, mar 2022. [Online]. Available: https://doi.org/10.1145/3505181

[26] G. Marin-Sanchez, J. Gonzalez-Conde, and M. Sanz, "Quantum algorithms for approximate function loading," 2021. [Online]. Available: https://arxiv.org/abs/2111.07933

[27] O. R. Meitei, B. T. Gard, G. S. Barron, D. P. Pappas, S. E. Economou, E. Barnes, and N. J. Mayhall, "Gate-free state preparation for fast variational quantum eigensolver simulations," *npj Quantum Information*, vol. 7, no. 1, pp. 1–11, 2021.

[28] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Physical Review A*, vol. 98, no. 3, p. 032309, 2018.

[29] M. Mottonen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, "Transformation of quantum states using uniformly controlled rotations," *arXiv preprint quant-ph/0407010*, 2004.

[30] K. Nakaji, S. Uno, Y. Suzuki, R. Raymond, T. Onodera, T. Tanaka, H. Tezuka, N. Mitsuda, and N. Yamamoto, "Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicators," *Physical Review Research*, vol. 4, no. 2, p. 023136, 2022.

[31] M. Plesch and i. c. v. Brukner, "Quantum-state preparation with universal gate decompositions," *Phys. Rev. A*, vol. 83, p. 032302, Mar 2011. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.83.032302

[32] A. G. Rattew and B. Koczor, "Preparing arbitrary continuous functions in quantum registers with logarithmic complexity," 2022. [Online]. Available: https://arxiv.org/abs/2205.00519

[33] A. G. Rattew, Y. Sun, P. Minssen, and M. Pistoia, "The Efficient Preparation of Normal Distributions in Quantum Registers," *Quantum*, vol. 5, p. 609, Dec. 2021. [Online]. Available: https://doi.org/10.22331/q-2021-12-23-609

[34] V. Shende, S. Bullock, and I. Markov, "Synthesis of quantum-logic circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 6, pp. 1000–1010, 2006.

[35] Y. Shi, N. Leung, P. Gokhale, Z. Rossi, D. I. Schuster, H. Hoffmann, and F. T. Chong, "Optimized compilation of aggregated instructions for realistic quantum computers," in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, 2019, pp. 1031–1044.

[36] X. Sun, G. Tian, S. Yang, P. Yuan, and S. Zhang, "Asymptotically optimal circuit depth for quantum state preparation and general unitary synthesis," *arXiv preprint arXiv:2108.06150*, 2021.

[37] H. Wang, Y. Ding, J. Gu, Y. Lin, D. Z. Pan, F. T. Chong, and S. Han, "Quantumnas: Noise-adaptive search for robust quantum circuits," in *2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 2022, pp. 692–708.

[38] 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," *arXiv preprint arXiv:2110.11331*, 2021.

[39] H. Wang, Z. Li, J. Gu, Y. Ding, D. Z. Pan, and S. Han, "Qoc: quantum on-chip training with parameter shift and gradient pruning," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 655–660.

[40] H. Wang, P. Liu, J. Cheng, Z. Liang, J. Gu, Z. Li, Y. Ding, W. Jiang, Y. Shi, X. Qian *et al.*, "Quest: Graph transformer for quantum circuit reliability estimation," *arXiv preprint arXiv:2210.16724*, 2022.

[41] W. K. Wootters and W. H. Zurek, “A single quantum cannot be cloned,” *Nature*, vol. 299, no. 5886, pp. 802–803, 1982.
[42] L. G. Wright, T. Onodera, M. M. Stein, T. Wang, D. T. Schachter, Z. Hu, and P. L. McMahon, “Deep physical neural networks trained with backpropagation,” *Nature*, vol. 601, no. 7894, pp. 549–555, 2022.
[43] X.-M. Zhang, T. Li, and X. Yuan, “Quantum state preparation with optimal circuit depth: Implementations and applications,” 2022. [Online]. Available: https://arxiv.org/abs/2201.11495