Neural Network-Based Frequency Optimization for Superconducting Quantum Chips

Bin-Han Lu[1,2], Qing-Song Li[1,2], Peng Wang[1,2], Zhao-Yun Chen[3*], Yu-Chun Wu[1,2,3], and Guo-Ping Guo[1,2,3,4]

[1]*Key Laboratory of Quantum Information Chinese Academy of Sciences, School of Physics, University of Science and Technology of China, Hefei 230026, China*
[2]*CAS Center For Excellence in Quantum Information and Quantum Physics, University of Science and Technology of China, Hefei 230026, China*
[3]*Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230026, China*
[4]*Origin Quantum Co., Ltd. (Hefei)*

(Received 20 December 2024; accepted manuscript online 24 February 2025)

Optimizing frequency configurations for qubits and gates in superconducting quantum chips presents a complex NP-complete challenge, critical for mitigating decoherence and crosstalk. This paper introduces a neural network-based approach, leveraging the network as a surrogate model to predict frequency errors. The method employs a closed-loop Bayesian optimization framework to iteratively refine configurations, guided by the network's knowledge of nonlinear error mechanisms. By focusing on localized chip windows, the optimization identifies optimal frequency settings that minimize errors. The approach is validated through randomized and cross-entropy benchmarking, showing improved energy calculations when optimizing frequency configurations for a crosstalk-aware hardware-efficient ansatz in variational quantum eigensolvers on superconducting quantum chips.

DOI: 10.1088/0256-307X/42/3/030204 CSTR: 32039.14.0256-307X.42.3.030204

To achieve fault-tolerant quantum computing,[1,2] superconducting quantum processors must scale beyond the limitations of noisy intermediate-scale quantum (NISQ) chips.[3] However, this progression faces two significant challenges: unreliable hardware manufacturing and imperfections in control systems, which contribute to various errors. Decoherence errors, including qubit relaxation and dephasing, are one major issue.[4] Another critical problem is crosstalk errors, which occur when excessive residual coupling between qubits causes parallel operations to interfere with each other.[5–7] Additionally, microwave pulses intended for single-qubit gates have unwanted effects on neighboring qubits, leading to microwave crosstalk errors.[8] Addressing these challenges is essential for enabling scalable fault-tolerant quantum computing.

The frequency-tunable architecture enables two-qubit gates by bringing neighboring qubits directly into resonance,[9] resulting in shorter gate execution times than fixed-frequency qubits. However, errors in this architecture are susceptible to the frequency configuration. Different configurations lead to variations in qubit dephasing and relaxation times. Notably, when qubit frequencies align with Two-Level System (TLS) defects[10] or flux-sensitive points,[11] coherence times are significantly reduced. Furthermore, crosstalk arises when qubit frequencies unintentionally come into resonance.[12] As a result, optimizing qubit frequency configurations is critical to mitigating errors in frequency-tunable architectures.[13]

Addressing frequency configuration requires developing a model that maps the frequency of each quantum gate to its associated error values. This model must account for decoherence noise at different qubit frequencies and identify whether qubit frequencies lie within closely resonant regions. Understanding how various errors interact and collectively impact qubits increases the overall complexity. Building upon this model, solving an optimization problem becomes necessary to determine a set of frequency configurations that minimize errors for each quantum gate. However, as the number of qubits increases, the scale of the problem grows exponentially, making it a highly constrained and computationally intensive challenge.

Existing research addresses these challenges through methods such as compensation pulses to mitigate XY control signal crosstalk[14] and two-tone flux modulation to suppress decoherence.[15] Additionally, studies[12,16] propose using tunable couplings to suppress leakage effects from two-qubit gates on spectator qubits and to minimize residual ZZ coupling between single-qubit gates. However, these methods inevitably increase the complexity of the control system. Given the strong dependence of errors on frequency, a more direct approach involves identifying frequency configurations that circumvent hardware imperfections, thereby simplifying hardware control.

Some frequency configuration strategies lack thorough investigation of the underlying physical systems,[17,18] while others[19,20] incorporate deeper physical principles

*Corresponding author. Email: chenzhaoyun@iai.ustc.edu.cn
© 2025 Chinese Physical Society and IOP Publishing Ltd. All rights, including for text and data mining, AI training, and similar technologies, are reserved.

through linear combination of different error models. However, these models have limitations, as they may fail to capture nonlinear error interactions and struggle with complex error sources like Z-line distortion and microwave crosstalk, which limits their scalability. Ai *et al.*[21] proposed using multilayer perceptrons and graph neural networks for optimization, but fabrication inconsistencies reduce their generalizability. Similarly, linear programming-based approaches[22,23] are unable to fully capture the intricate noise mechanisms of qubits and their interactions.

The relationship between quantum gate frequencies and errors on a quantum chip is complex, as errors are not merely linear combinations and include difficult-to-quantify issues like global microwave crosstalk and Z-line distortion. To address this, we propose a neural network-based frequency configuration scheme. Unlike traditional methods that rely on linear error models, neural networks can effectively learn and model the nonlinear interactions among error mechanisms.

Initially, training data is generated by randomly sampling frequency configurations and collecting experimental results. This approach often includes suboptimal configurations, resulting in low neural network prediction accuracy near optimal frequency regions. To address this limitation, we implement a closed-loop Bayesian optimization framework.[24] The neural network first learns error mechanisms from the initial data, then optimizes frequency allocation. The optimized configuration results are subsequently added to the training set to iteratively refine the network. Compared to random data points, optimized configurations exhibit lower errors and improved performance. This closed-loop process guides the neural network from suboptimal to optimal configurations, capturing the physical mechanisms near superior configurations and enhancing frequency allocation optimization.

In our frequency allocation process, we initialize a random frequency configuration and use the neural network to predict gate errors across the chip. The window with the highest average error is identified and optimized iteratively until the global average gate error is minimized. By allowing iterative revisions to previous configurations, this method achieves lower overall errors and greater optimization flexibility.

Finally, after completing the configuration, we conducted single-qubit randomized benchmarking (RB) and two-qubit cross-entropy benchmarking (XEB).[25] The results significantly reduced gate errors, confirming that our frequency configuration effectively identifies and selects low-error-rate frequencies. Furthermore, we developed optimized frequency configurations for crosstalk-aware hardware-efficient ansatz (HEA)[26] in variational quantum eigensolvers (VQE). Results show that, under the same HEA, optimized frequency configurations achieve lower energy values compared to random configurations, highlighting the importance of frequency configurations in minimizing errors and enhancing computation fidelity.

Error Mechanisms. On a quantum chip, the frequencies of single-qubit and two-qubit gates directly impact the final computational results. The first type of error is relaxation error (T_1 error).[4] As the qubit frequency shifts, its T_1 relaxation time also changes. The T_1 relaxation time sharply decreases at certain Two-Level System (TLS) defect points,[10] causing the qubit to easily transition from the higher energy state to the lower energy state. The second type is a dephasing error (T_2 error).[11] In a tunable-frequency architecture, a qubit's dephasing time T_2 depends on the first derivative of the frequency to the magnetic flux, given by $\frac{1}{T_2} \propto \frac{d\omega}{d\phi}$. Therefore, frequency settings should be chosen such that both T_1 and T_2 are sufficiently long to minimize errors. The third type of error occurs during two-qubit gate execution when the qubit frequency shifts from the idle frequency ω_{off} to the interaction frequency ω_{on}. An excessive shift $|\omega_{\text{off}} - \omega_{\text{on}}|$ not only keeps the qubit away from the frequency with a longer T_2 for an extended period but also causes excessive waveform variation in the two-qubit gate, leading to Z-line distortion and discrepancies between the target pulse and the actual pulse.[16] The fourth type is stray coupling crosstalk.[19] If the gate-qubit frequency approaches resonance with neighboring or next-neighboring qubits, unintended coupling occurs, which can reduce gate fidelity. Finally, microwave crosstalk arises when a drive signal intended for qubit q_i influences a nearby qubit q_j. If q_i and q_j are near resonance, q_j is significantly affected. Unlike stray coupling, microwave crosstalk has a broader reach and can impact even non-neighboring qubits.

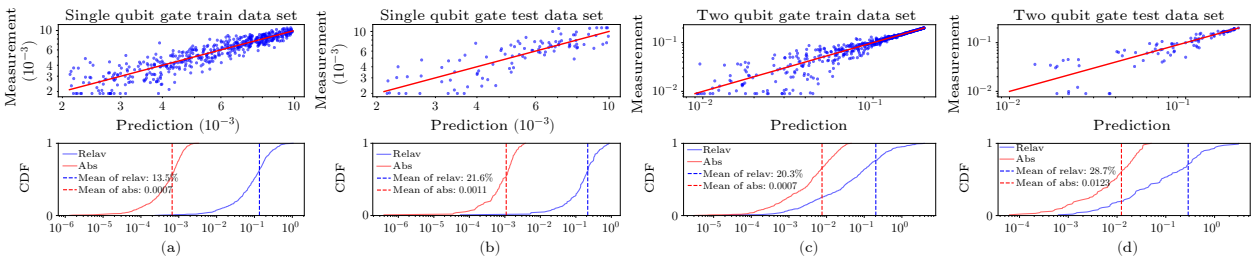

Fig. 1. The top row shows the data scatter plot. The measurement should be close to the prediction, with the scatter points distributed around the line $y = x$. The bottom row shows the cumulative probability distribution (CDF) of |measurement − prediction| and |measurement − prediction|/measurement. The dashed line represents the mean value. (a), (b), (c), and (d) represent the results of the single-gate training set, single-gate test set, two-gate training set, and two-gate test set, respectively.

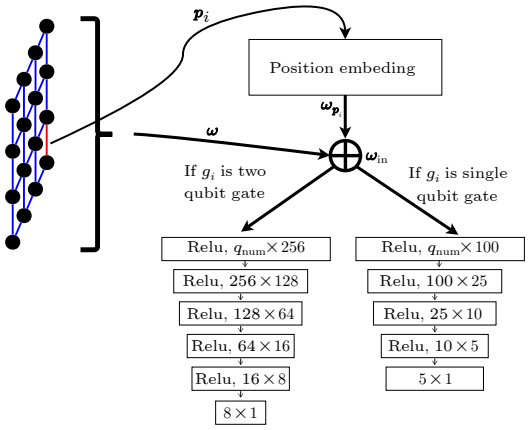

Fig. 2. Given a chip and a set of frequency configuration $\boldsymbol{\omega}$. The input vector is defined as $(\boldsymbol{\omega}, \boldsymbol{p}_i)$. The vector $\boldsymbol{p}_i$ undergoes a linear transformation W_p, transforming it into a dense vector $\boldsymbol{\omega}_{\boldsymbol{p}_i}$ incorporating positional information. This transformed vector is then added to $\boldsymbol{\omega}$, resulting in a combined vector $\boldsymbol{\omega}_\mathrm{in}$ that contains both configuration and positional information. $\boldsymbol{\omega}_\mathrm{in}$ is then fed into a multilayer perceptron. If g_i is a single-qubit gate, the single-gate neural network is used for forward propagation; otherwise, the two-gate network is used for forward propagation. The final output ϵ_i is the predicted error of g_i under configuration $\boldsymbol{\omega}$.

Neural Network Error Estimator. We use a neural network to predict gate error under a given frequency configuration.[27] The network (see Fig. 2) consists of a position embedding layer,[28] an input layer, hidden layers, and an output layer. Overall, it is a multilayer perceptron.

The network uses the frequencies of all single-qubit and two-qubit gates as inputs and predicts the errors for a quantum gate under a given configuration. Based on the previously discussed error mechanisms, a single-qubit gate's error is influenced by its frequency, the frequencies of its neighboring and next-neighboring qubits, and the frequencies of qubits inducing microwave crosstalk. A two-qubit gate's error is affected by the idle and interaction frequencies of the gate qubits, the frequencies of neighboring spectator qubits, and the frequencies of qubits causing microwave crosstalk with the gate qubits. To determine which qubit frequencies affect the target quantum gate, the neural network analyzes the input data and dynamically refines the connections between its neurons to optimize the learning process. As a result, the input vector must comprehensively include the single-qubit and two-qubit frequencies of all qubits on the chip,

$$\boldsymbol{\omega} = (\boldsymbol{\omega}_\mathrm{single}, \boldsymbol{\omega}_\mathrm{two}). \tag{1}$$

Here, $\boldsymbol{\omega}_\mathrm{single}$ and $\boldsymbol{\omega}_\mathrm{two}$ represent the frequencies of single-qubit and two-qubit gates, respectively.

The input vector must include information about the gate's position on the chip to enable the neural network to predict the error of a specific quantum gate. This is achieved using position embedding.[28] As an example, consider a chip with an $M \times N$ qubit array and $2MN - M - N$ couplers, resulting in an input vector with a dimension of $3MN - M - N$. To predict the error ϵ_i of the i-th gate, we define a one-hot sparse vector $\boldsymbol{p}_i$, with a dimension equal to the number of gates ($3MN - M - N$ in this example) such that $(\boldsymbol{p}_i)_j = \delta_{ij}$. However, the one-hot encoding of $\boldsymbol{p}_i$ has significant limitations. The physical relationship between quantum gates on the chip can often be described as "distance". For instance, if g_i and g_j are neighboring qubits, their distance should be small. Conversely, the distance between unrelated gates should be large. In one-hot encoding, however, each position vector $\boldsymbol{p}_i$ is orthogonal to every other position vector $\boldsymbol{p}_j$. This means that the cosine between any pair of gates is always 0. Consequently, this representation fails to capture the physical interactions or distances between gates. To address this issue, we introduce a trainable linear layer W_p, which maps the one-hot vector $\boldsymbol{p}_i$ to a dense vector $\boldsymbol{\omega}_{\boldsymbol{p}_i} = W_\mathrm{p}\boldsymbol{p}_i$ with a dimension matching the frequency configuration vector $\boldsymbol{\omega}$. Before formally training the neural network, we first train the position embedding layer using Eq. (2) as the loss function. Here, d_{ij} represents the distance between quantum gates on the chip, and d_max is the maximum distance used for normalization. When the distance is small, the loss penalizes smaller cosine values, and when the distance is large, it penalizes larger cosine values. This approach enables effective training of the position embedding layer,

$$L = \sum_{i\neq j}\left[\frac{d_{ij}}{d_\mathrm{max}}\cos^2(\boldsymbol{\omega}_{\boldsymbol{p}_i}, \boldsymbol{\omega}_{\boldsymbol{p}_j}) + \left(1 - \frac{d_{ij}}{d_\mathrm{max}}\right)\left(1 - \cos(\boldsymbol{\omega}_{\boldsymbol{p}_i}, \boldsymbol{\omega}_{\boldsymbol{p}_j})\right)^2\right]. \tag{2}$$

The dense vector $\boldsymbol{\omega}_{\boldsymbol{p}_i}$ captures the physical and positional relationships between the gates during training (see Fig. 3). Finally, the input vector to the neural network is constructed by combining the frequency configuration vector $\boldsymbol{\omega}$ and the position embedding vector $\boldsymbol{\omega}_{\boldsymbol{p}_i}$ through addition: $\boldsymbol{\omega}_\mathrm{in} \leftarrow \boldsymbol{\omega} + \boldsymbol{\omega}_{\boldsymbol{p}_i}$. This combined vector encodes both frequency information and positional and physical correlations, enabling the neural network to accurately predict the error ϵ_i of the target gate.

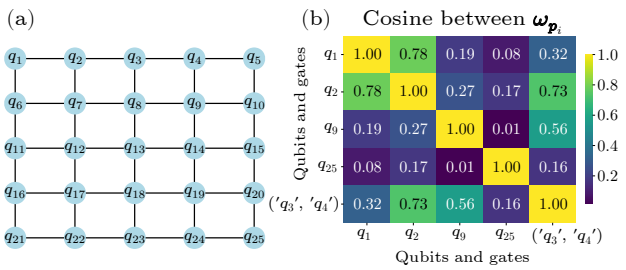

Fig. 3. (a) A diagram of the coupling structure of a 5×5 chip. (b) We have extracted the matrix of cosine values between the position vectors of the single-qubits q_1, q_2, q_9, q_{25}, and the two-qubit gate q_3q_4. It can be seen that the cosine is relatively high between q_1 and q_2, as well as between q_9 and the two-qubit gate q_3q_4, and they are located close to each other in the chip's coupling diagram. Additionally, q_{25} has low cosine values with all other position vectors, as it is located in the bottom-right corner of the chip, far from the extracted qubits.

After completing the position embedding, we begin the formal training of the neural network. Figure 1 shows the training and testing results based on randomly selected frequency configurations. It can be observed that the relative errors for the single-gate training and test sets are 13.5% and 21.6%, respectively, with absolute errors of 7×10^{-4} and 1.1×10^{-3} for the training and test sets. For the two-gate configuration, the relative errors for the training and test sets are 20.3% and 28.7%, with absolute errors of 7.6×10^{-3} and 1.23×10^{-2}, respectively. In the training process, we used 500 data sets and achieved an accuracy comparable to that of Google using 6500 benchmark sets.

Frequency Optimization. For an $M \times N$ chip, the number of frequency variables is $3MN - M - N$. The control system allows for a frequency precision of δf so that each frequency can vary within the range $F = \{f_{\min} + k\delta f | k \in \mathbb{N}, k \leqslant (f_{\max} - f_{\min})/\delta f\}$. This makes the problem scale exponentially as $O(|F|^{3MN})$, limiting us to incrementally optimize the frequency configuration within a local window S on the chip.

We compute the error metrics for all quantum gates using this initial setup with a randomly selected configuration. Then, we iterate through all possible windows using a maximum optimization window size S. We calculate the average gate error within each window and then optimize the frequencies of the two-qubit and single-qubit gates in the window with the highest average error. This iterative process continues until a configuration with minimal global average gate errors is achieved (see Fig. 4). Figure 5(a) shows the average fidelity results for single-qubit and two-qubit gates predicted by the neural network after the final optimization. As can be seen, we have found the most optimal frequency combinations indicated by the neural network. Figure 5(b) further demonstrates that while increasing S leads to a marginal decrease in final optimization performance, which results in a significant increase in frequency configuration time. Based on this trade-off analysis, we select $S = 2$ as the optimal value.

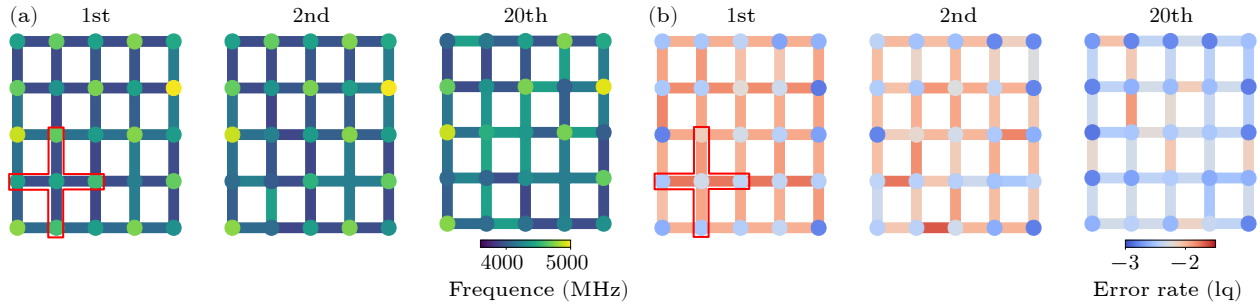

Fig. 4. (a)–(b) The plots display frequency and error distributions for the 1st, 2nd, and 20th (final) iterations from left to right. In each iteration, the region S with the maximum average error is identified (marked by a red box). Here, S represents the distance range between quantum gates and the central qubit, encompassing both single-qubit (sing.q) and two-qubit gates.

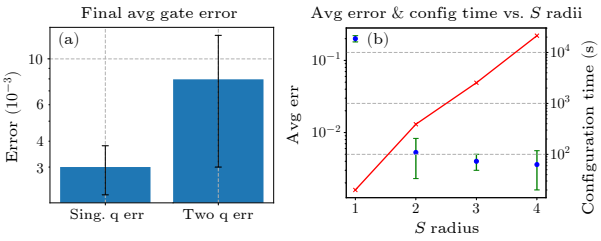

Fig. 5. (a) The average errors of single-qubit and two-qubit gates are displayed. (b) The average gate error decreases as the radius of S increases, while the time required for frequency configuration increases exponentially.

Closed-Loop Optimization Strategy. Figures 1(b) and 1(c) show that the randomly sampled two-qubit gate error data were not measured at optimal frequency configurations, with most errors exceeding 10^{-2}. This is because training the network with randomly generated data may include points with very short decoherence times or large residual coupling between qubits. The neural network only learns the physical mechanisms in these regions, resulting in lower prediction accuracy near the more optimal frequency configurations. Consequently, while the frequency allocation algorithm achieves configurations with relatively low error rates under neural network guidance, these solutions remain suboptimal due to the network's limited prediction accuracy in these regions. Consequently, the configurations assigned by the network are not the optimal ones. To address this issue, we propose a closed-loop optimization method that utilizes Bayesian optimization. By using the error from the previous frequency configuration as new data, the network is guided to learn around the known optimal frequencies. This approach gradually enables the network to understand the error mechanisms near the truly optimal frequencies, allowing the assigned optimal frequency configuration to approach the actual optimal configuration. The neural network is the surrogate function (dashed line) in Fig. 6(b), while the actual frequency-error relationship is the solid line function. As the iterations of frequency allocation-error measurement-training progress in Fig. 6(a), the surrogate transitions from the gray dashed line to the black dashed line, and $\boldsymbol{\omega}^*$ approaches the actual optimal frequency. Except for the initial training with random data, each subsequent data collection involves repeating the frequency allocation process 20 times to obtain 20 sets of frequency-error training data.

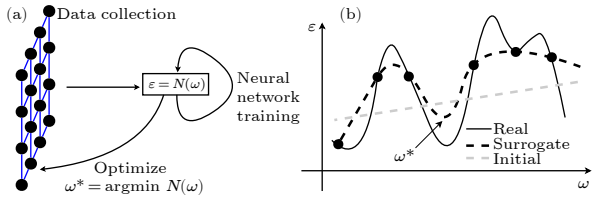

Fig. 6. (a) illustrates the closed-loop optimization iteration process. (b) The gray dashed line represents the initial surrogate model, which evolves into the black line as Bayesian iterations progress.

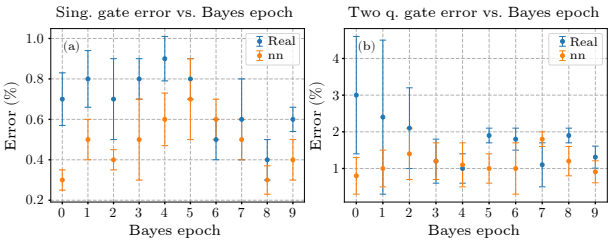

Fig. 7. (a) illustrates the closed-loop optimization process for single-qubit gates across Bayesian optimization epochs, comparing neural network predictions with measured chip errors. (b) presents the corresponding results for two-qubit gates, maintaining the same comparison framework.

Based on the network trained with random data, we use the network to identify the optimal operating frequencies $\boldsymbol{\omega}^*$ for the two-qubit and single-qubit gates predicted by the network. From Fig. 7, it can be seen that although the closed-loop optimization has little effect on the single-qubit gates, it does impact the optimization of the two-qubit gates. Initially, the neural network predicts an average gate error of $< 1\%$, but the actual average error is $> 3\%$. As the closed-loop optimization progresses, the neural network's predictions gradually approach the average error.

Figure 8 presents the CDF of error probabilities following frequency configuration optimization, with comparisons against Google's snake optimizer and random configurations. Our approach demonstrates significant error reduction: 37.1% and 52.0% lower average errors in RB experiments, and 9.7% and 94.4% reductions in XEB experiments, compared to Google's method and random configurations, respectively.

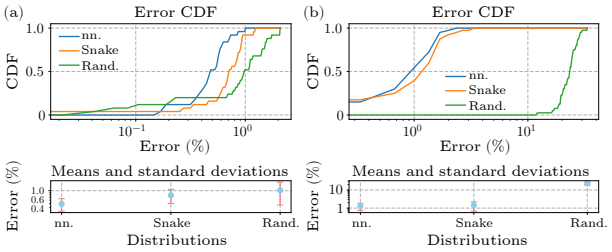

Fig. 8. (a) The CDFs demonstrate error rates across different optimization methods for single-qubit gates: neural network-based (blue, 0.49%), Google's optimization (orange, 0.68%), and random configuration (green, 1.02%). (b) For two-qubit gates, the CDFs show similar comparisons: neural network-based (blue, 1.31%), Google's baseline (orange, 1.45%), and random configuration (green, 23.3%).

Frequency Configuration for HEA. The HEA represents the most widely adopted ansatz structure in VQE implementations. As illustrated in Fig. 9(b), its architecture comprises alternating layers of single-qubit gates and entangling unitary operations (U_{ENT}), where U_{ENT} typically implements full qubit connectivity through CNOT gate arrangements.

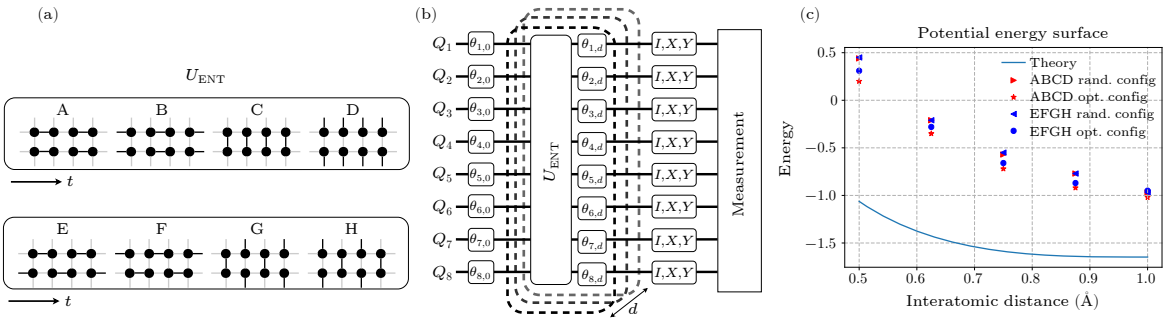

Fig. 9. (a) Two sets of the largest parallelizable CZ gate patterns. Multiple CZ gates within any ABCD (or EFGH) pattern can execute in parallel, while gates from different patterns cannot execute simultaneously. (b) In the HEA circuit, each layer comprises single-qubit gates $\boldsymbol{\theta}_{i,d}$ and trainable CZ arranged in the ABCD(EFGH) patterns, coupling all qubits together. (c) The ground-state PES of the H_4 molecule was calculated using VQE circuits with the ABCD and EFGH ansatz patterns.

In a periodic grid structure [see Fig. 10(a)], when qubits q_i and q_j execute a two-qubit gate, their neighboring qubit q_n may engage in a two-qubit gate with up to three other qubits or a single-qubit gate, requiring up to four distinct frequency settings. Ideally, an optimized frequency configuration should avoid any crosstalk between parallel quantum gates. However, for an $M \times N$ chip, the number of couplers is $2MN - M - N$, with each two-qubit gate on a coupler having two possible states: executed or not executed. Consequently, the number of potential parallel two-qubit gate scenarios can reach $O(2^{MN})$[29] [see Fig. 10(b)]. The crosstalk between qubits varies with each unique parallel gate scenario, and considering all possible scenarios results in an overwhelming number of frequency

constraints. Given the exponential number of parallel scenarios, performing frequency configuration for all of them is infeasible. Thus, we focus on two two-qubit gate grouping patterns [see Fig. 9(a)], dividing the two-qubit gates into four groups. Gates within each group can operate in parallel, but groups cannot execute in parallel with one another. This approach covers all possible two-qubit gates on the chip. Since ABCD or EFGH includes all two-qubit gates, either of them can serve as the entanglement layer in the VQE circuit, entangling all the qubits.

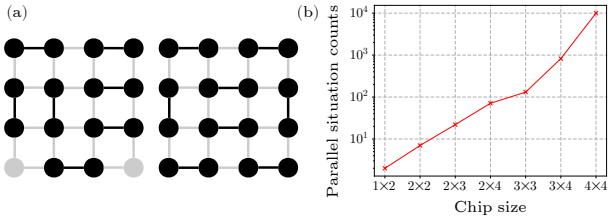

Fig. 10. (a) The black edges and nodes represent the qubits and couplers involved in quantum gates within a specific time slice. (b) The number of parallel two-qubit gate scenarios increases with the scale of the $M \times N$ chip.

We optimized the frequency configuration of two-qubit gates based on ABCD or EFGH patterns. The HEA ansatz follows a cyclic, multi-layer sequence [see Fig. 9(b)]:

$$|\psi(\boldsymbol{\theta})\rangle = \prod_{q=1}^{N} U(\boldsymbol{\theta}^{q,\,d}) \times U_{\mathrm{ENT}} \times \prod_{q=1}^{N} U(\boldsymbol{\theta}^{q,\,d-1}) \times U_{\mathrm{ENT}} \cdots \times U_{\mathrm{ENT}} \times \prod_{q=1}^{N} U(\boldsymbol{\theta}^{q,\,0}), \quad (3)$$

where $U(\boldsymbol{\theta}^{q,\,d})$ represents the parameterized single-qubit gate layer, and U_{ENT} represents the two-qubit gates arranged in the order of ABCD or EFGH, entangling all qubits.

To verify the effectiveness of the frequency configuration optimization, we optimized the frequencies for two different patterns and compared the results with random frequency configurations. After frequency optimization, we also assessed the minimum energy of the ground-state potential energy surface (PES) from the theoretical line. From Fig. 9(c), we conclude that, under the same ansatz, the PES for the random frequency configuration is higher than that of the optimized frequency configuration. This demonstrates that the HEA designed with frequency-optimized configuration allows the quantum chip to operate at an optimal frequency, mitigating the impact of crosstalk on algorithm fidelity.

Conclusion. We introduce a comprehensive strategy for optimizing quantum gate frequencies, tackling the critical challenges of reducing gate errors and minimizing crosstalk in quantum circuits. By integrating neural network-guided prediction, Bayesian optimization, systematic frequency configuration, and rigorous benchmarking, our approach demonstrates significant enhancements in quantum computing performance.

Extensive validation through RB and XEB consistently confirmed that our optimized configurations achieve the lowest error rates. Additionally, implementation with the VQE showcased improved accuracy in calculating the ground-state energy of the H_4 molecular system using our optimized frequency patterns.

These results underscore the effectiveness of targeted frequency optimization and error mitigation strategies in advancing quantum computing capabilities. Our method not only boosts gate fidelity but also maintains an optimal balance in computational complexity. This work lays the groundwork for scaling the approach to larger quantum systems and opens avenues for exploring dynamic frequency adjustments in real-time quantum computing applications.

Acknowledgements. This work was supported by the National Key Research and Development Program of China (Grant No. 2023YFB4502500).

References

[1] Horsman D, Fowler A G, Devitt S, and Van Meter R 2012 *New J. of Phys.* **14** 123011
[2] Chamberland C, Zhu G, Yoder T J, Hertzberg J B, and Cross A W 2020 *Phys. Rev. X* **10** 011022
[3] Boixo S, Isakov S V, Smelyanskiy V N, Babbush R, Ding N, Jiang Z, Bremner M J, Martinis J M, and Neven H 2018 *Nat. Phys.* **14** 595
[4] Ithier G, Collin E, Joyez P, Meeson P, Vion D, Esteve D, Chiarello F, Shnirman A, Makhlin Y, Schriefl J *et al.* 2005 *Phys. Rev. B* **72** 134519
[5] Zhao P, Linghu K, Li Z, Xu P, Wang R, Xue G, Jin Y, and Yu H 2022 *PRX Quantum* **3** 020301
[6] Krinner S, Lazar S, Remm A, Andersen C K, Lacroix N, Norris G J, Hellings C, Gabureac M, Eichler C, and Wallraff A 2020 *Phys. Rev. App.* **14** 024042
[7] Seo S, Seong J, and Bae J 2021 arXiv:2112.10651 [quant-ph]
[8] Zhao P, Zhang Y, Li X, Han J, Xu H, Xue G, Jin Y, and Yu H 2022 arXiv:2206.03710 [quant-ph]
[9] Sung Y, Ding L, Braumüller J, Vepsäläinen A, Kannan B, Kjaergaard M, Greene A, Samach G O, McNally C, Kim D *et al.* 2021 *Phys. Rev. X* **11** 021058
[10] Tian L 2011 *Phys. Rev. B* **84** 035417
[11] Krantz P, Kjaergaard M, Yan F, Orlando T P, Gustavsson S, and Oliver W D 2019 *Appl. Phys. Rev.* **6**
[12] Lu B H, Wang P, Chen Z Y, Liu H Y, Sun T P, Duan P, Wu Y C, and Guo G P 2024 *IEEE Trans. Comput. Aid. Des.*
[13] Klimov P V, Kelly J, Martinis J M, and Neven H 2020 arXiv:2006.04594 [quant-ph]
[14] Krinner S, Lacroix N, Remm A, Di Paolo A, Genois E, Leroux C, Hellings C, Lazar S, Swiadek F, Herrmann J *et al.* 2022 *Nature* **605** 669
[15] Valery J A, Chowdhury S, Jones G, and Didier N 2022 *PRX Quantum* **3** 020337
[16] Sung Y, Ding L, Braumüller J, Vepsäläinen A, Kannan B, Kjaergaard M, Greene A, Samach G O, McNally C, Kim D *et al.* 2021 *Phys. Rev. X* **11** 021058
[17] Versluis R, Poletto S, Khammassi N, Tarasinski B, Haider N, Michalak D J, Bruno A, Bertels K, and DiCarlo L 2017 *Phys. Rev. Appl.* **8** 034021
[18] Ding Y, Gokhale P, Lin S F, Rines R, Propson T P, and Chong F T 2020 *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)* pp. 201

[19] Klimov P V, Bengtsson A, Quintana C, Bourassa A, Hong S, Dunsworth A, Satzinger K J, Livingston W P, Sivak V, Niu M Y *et al.* 2024 *Nat. Commmun.* **15** 2442
[20] Wang B Y, Liu W, Chen X, Xu S, Cui J, and Yung M H 2025 *Science China Physics, Mechanics & Astronomy* **68** 220312
[21] Ai H and Liu Y X 2024 arXiv:2411.16354 [quant-ph]
[22] Morvan A, Chen L, Larson J M, Santiago D I, and Siddiqi I 2022 *Phys. Rev. Res.* **4** 023079
[23] Zhang Z, Gokhale P, and Larson J M 2025 *Phys. Rev. Res.* **111** 012619
[24] Frazier P I 2018 arXiv:1807.02811 [stat.ML]
[25] Arute F, Arya K, Babbush R, Bacon D, Bardin J C, Barends R, Biswas R, Boixo S, Brandao F G, Buell D A *et al.* 2019 *Nature* **574** 505
[26] Kandala A, Mezzacapo A, Temme K, Takita M, Brink M, Chow J M, and Gambetta J M 2017 *Nature* **549** 242
[27] Popescu M C, Balas V E, Perescu-Popescu L, and Mastorakis N 2009 *WSEAS Transactions on Circuits and Systems* **8** 579
[28] Huang Z, Liang D, Xu P, and Xiang B 2020 arXiv:2009.13658 [cs.CL]
[29] Jou M J and Chang G J 2000 *Taiwanese Journal of Mathematics* **4** 685