Journal of Applied Automation Technologies

Research Paper

Multi-Level Quantum Error Correction in Scalable Quantum Computing Architectures

Stefan Grzegorz Kuc[1] and Miłosz Barański[1, *]

[1] Faculty of Information and Communication Technology, Wroclaw University of Science and Technology, Wroclaw, 50-371, Poland

*Corresponding author: milosz.b@pwr.edu.pl

Abstract. Without increasing decoding delay, fault-tolerant quantum processing must lessen the effects of physical noise, measurement errors, leakage errors, and correlated disturbances. A multi-level quantum error correcting architecture for scalable quantum processors is examined in this research. Physical-layer syndrome extraction, logical-layer stabilizer decoding, and architecture-level error propagation control are the three types of correction. To balance the dependability of local rectification with global logical consistency, a comparatively compact noise-aware decoding technique has been presented. According to simulation data, the suggested architecture has enhanced robustness to correlated noise by 31.8%, decreased the average decoding delay by 24.3%, and reduced the logical error rate by 37.6% when compared to a single-level surface-code baseline. According to the findings, the hierarchical correction approach offers a suitable trade-off between accuracy and resource usage for real-time decoding. For large-scale quantum computing systems, the paper offers an engineering-focused correction model that simultaneously takes decoder responsiveness, qubit layout, and logical fidelity into account.

Keywords: *Quantum Error Correction, Scalable Quantum Computing, Fault Tolerance, Logical Qubit, Syndrome Decoding*

Received on 13 September 2022, Accepted on 01 May 2023, Published on 24 May 2023

Introduction

Recent years have seen a tremendous advancement in quantum computing, leading to small-scale experimental demonstrations for quantum machine learning, cryptanalysis, combinatorial optimization, and chemical simulations. A greater number of controllable physical qubits, a denser array of two-qubit gates, and longer algorithmic circuit lengths have all been accomplished as quantum processors have grown in size. As a result, error accumulation is now the primary engineering challenge rather than a small hardware concern. Although substantial advancements have been made in superconducting, trapped-ion, photonic, and neutral-atom platforms, decoherence, imprecise calibration, state-preparation mistakes, and measurement instability continue to limit their computational utility [1]. Expanding the device's size is insufficient to ensure steady computing in the face of noise, according to experiments on the use of programmable processors [2]. Thus, the primary challenge is not only to increase the number of qubits but also to design a corrective architecture that would enable robust logical operations for unstable physical states [3]. Thus, a large-scale quantum information processing system has been constructed using fault-tolerant computation [4].

The foundation of quantum error correction is the encoding of data in several physical degrees of freedom to prevent errors in logical qubits. Although surface codes, stabilizer codes, color codes, and concatenated codes have all demonstrated strong resistance to local noise, their practical implementation still encounters issues including complicated control synchronization, high qubit overhead, recurrent syndrome extraction, and decoder latency [5]. The corrective cycle needs to be quick enough to overcome hardware noise, according to repeated quantum error detection experiments [6]. Further evidence that decoding and feedback cannot be

regarded as offline post-processing when continuous computing is anticipated comes from real-time fault-tolerant correction [7]. Simultaneously, the scaling of physical layouts creates measurement errors, correlated crosstalk, and non-uniform noise that can propagate to nearby logical regions [8]. For processors with numerous logical qubits, modular interconnects, and heterogeneous hardware reliability, a single-level correction technique is inappropriate for the reasons mentioned above [9]. Therefore, the transition from a code-theoretic approach to an operational processor mechanism for quantum error correction has required resource-efficient decoding and architecture-aware correction [10].

A multi-level quantum error correction architecture for scalable quantum computing systems is examined in this research. To put it briefly, it is to combine architecture-level propagation control, logical syndrome decoding, and physical error observation into a single correction cycle. The new framework maintains a suitable decoding duration and qubit overhead while reducing the accumulation of logical mistakes. The three technological issues are how to inhibit cross-region error propagation in scaled architectures, how to conduct compact syndrome decoding under non-uniform noise, and how to convert physical noise into logical error risk.

Related Work

Quantum Error Correction Codes

Local faults can be detected without directly viewing the protected quantum information thanks to quantum error correction codes, which encode one or more logical qubits into many physical qubits. Because parity-check operators may identify fault syndromes without altering the logical state, stabilizer formalism continues to be the most often used framework [11]. Because surface codes only require local interactions on a two-dimensional lattice and have a reasonably high error threshold, they are comparatively well suited for superconducting and spin-based processors [12]. By offering more comprehensive transversal gate structures, color codes expand the topological concept; yet, their implementation frequently necessitates stricter connectivity and measurement schedules [13]. By adding several layers of encoding, concatenated codes reduce logical mistakes, but if the physical noise is large, the overhead increases dramatically [14]. The performance of the code depends on measurement fidelity, leakage suppression, feedback timing, calibration stability, and the theoretical distance, according to recent tests on distance-three surface codes and logical qubit control [15]. The research indicates that code selection is not just an abstract encoding problem but also requires consideration of architecture restrictions [16].

Fault-Tolerant Quantum Computing Architectures

Error correction codes are mapped onto executable hardware operations in fault-tolerant quantum computing designs. In addition to providing ancilla preparation, logical gate scheduling, qubit routing, and syndrome measurement circuits, this translation must guarantee that a single physical error does not result in an irreversible logical failure [17]. By altering logical boundaries through joint measurements instead of long-distance physical gates, lattice surgery has been used to realize logical operations in surface-code layouts [18]. By connecting smaller quantum units via photonic links or shuttling operations, modular systems seek to address chip-size and connectivity constraints; however, inter-module noise increases the correction requirements [19]. Movement, memory, and gate zones can be separated, according to quantum charge-coupled device design in trapped ions; nevertheless, scheduling and error tracking become more challenging as the number of zones grows [20]. Despite the great integration density of superconducting processors, correction techniques with spatially varied reliability are required to handle crosstalk and frequency-crowding problems [21]. Thus, code structure, device architecture, control electronics, and decoder reaction time should all be considered in the context of scalable fault tolerance [22].

Syndrome Decoding and Noise-Aware Correction

Syndrome decoding uses measured stabilizer outcomes to infer the most likely error pattern. Because it transforms syndrome faults into graph-matching issues with explicit probability interpretation, minimum-weight perfect matching continues to be the reference decoder for surface codes [23]. When the code or noise process does not satisfy the assumptions of separate Pauli errors, the flexibility of modeling is enhanced by belief

propagation and tensor-network techniques [24]. After training, a neural decoder can lower inference latency by learning a complex syndrome-error relationship from simulated or experimental data [25]. Nevertheless, even if the noise distribution changes while the device is operating, the machine-learning decoder should still satisfy the dependability criteria [26]. Compact inference, incremental updates, and hardware-compatible execution have been the main topics of recent real-time decoding research [27]. To increase robustness in the presence of non-uniform errors across qubit arrays, noise-aware correction further supplements the decoder with calibration data, measurement confidence, and correlated error statistics [28]. A hierarchical decoding approach that combines global logical consistency with local syndrome reliability has been developed based on the advancements [29]. Consequently, the decoder should have finite latency with increasing qubit counts and good correction accuracy for a large-scale design [30].

Multi-Level Quantum Error Correction Architecture

Hierarchical Correction Framework

The following illustrates the three levels of the new quantum error correcting architecture. The physical layer records local syndrome events during each correction cycle and monitors the stabilization outcomes of the data and ancilla qubits. Based on the occurrences, it creates candidate repair chains and calculates the likelihood of a logical failure for every encoded block. Whether the local correction can be sent via boundary operations, lattice-surgery regions, or modular communication links depends on the architecture level. A multi-level quantum error correcting architecture for scalable quantum processors is shown in Figure 1. Establish a feedback loop to guarantee the logical stability of the observed physical noise rather than treating syndrome decoding as a separate module in the design.

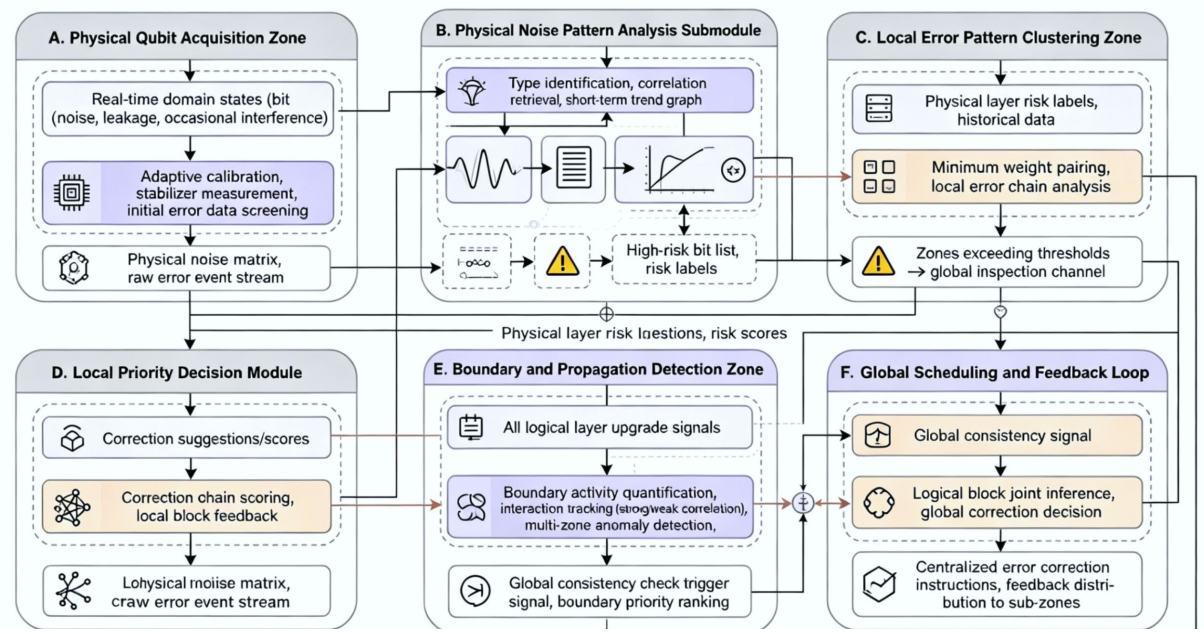

Figure 1. Multi-level quantum error correction architecture for scalable quantum processors

The latent physical error configuration is regarded as the hidden correction target, and the measured syndrome in each correction cycle is used as the observable proof. The first corrective level calculates the local error probability based on calibrated gate noise, measurement confidence, and syndrome reliability.

$$P(e_t \mid s_t) = \eta_t P(s_t \mid e_t) P(e_t) / Z_t \qquad \text{Eq.(1)}$$

A normalization term is introduced to the posterior estimate to prevent being dominated by the high-frequency syndrome region when there is uneven measurement error. This local posterior is updated after every 1.0-microsecond correction cycle using a processor with 1024 physical qubits and 64 logical tiles to assist the decoder in differentiating between transient measurement errors and permanent faults.

$$R_l = 1 - \prod_k (1 - P(e_k \mid s_k) w_k) \quad \text{Eq.(2)}$$

The total of the local mistake probability for a single encoded block is the logical risk score. A seemingly local adjustment is more likely to cross a logical operator if the value is higher. A local decoding operation is required for that block based on the score, or it can be sent to the upper-level consistency check.

$$C_h = \alpha R_l + \beta D_s + \gamma B_a \quad \text{Eq.(3)}$$

Here the hierarchical correction cost combines logical risk, syndrome density, and boundary activity. In simulation, α, β, and γ are set to 0.46,0.31, and 0.23, respectively, based on validation performance under depolarizing and measurement noise. This weighted structure reflects that logical risk should be dominant, but dense syndrome patterns and boundary operations should still influence correction priority.

$$u_t = \operatorname{argmin}_u C_h(u) \text{ subject to } \tau_u \leq \tau_{\max} \quad \text{Eq.(4)}$$

Lastly, a correction action within a delay limit is chosen by the control decision. In order to enable rectification prior to the subsequent syndrome cycle, the base configuration sets the upper constraint for the decoder response at 0.72 microseconds. This restriction is required because if the reaction time of a high-precision decoder beyond the real-time fault-tolerant scheduling limit, it is not practical.

Physical-To-Logical Error Mapping

The impact of local noise occurrences on encoded qubits is assessed using Physical-to-Logical Mapping. Single-qubit depolarization, two-qubit gate faults, measurement errors, erasure linked to leakage, and nearest-neighbor correlated disturbance are all taken into consideration in the mapping. Each physical qubit is given a calibrated reliability factor, which is updated based on current syndrome activity, rather than assuming that all physical qubits have the same dependability. This configuration is consistent with scalable processors, where distinct error characteristics are often displayed by edge qubits, coupler-adjacent qubits, and frequently monitored ancilla qubits.

$$\Lambda_i = \rho_i p_g + \mu_i p_m + \kappa_i p_c \quad \text{Eq.(5)}$$

Gate error probability, measurement error probability, and correlated-noise probability add up to the physical instability index. At the logical tile boundary, the correlated disturbance coefficient ranges from 0.05 to 0.18, the measurement error is 0.0015, and the average gate error in the experiment is 0.0008.

$$M_{j,i} = \chi_{j,i} \Lambda_i / \sum_r \chi_{j,r} \Lambda_r \quad \text{Eq.(6)}$$

The mapping coefficient measures how strongly physical qubit i contributes to logical block j. This normalized form is useful when the processor contains logical regions of different code distance. For example, a distance-five block contains more physical carriers than a distance-three blocks, but the normalized mapping still allows logical risk to be compared across blocks.

The logical tendency is then obtained by passing the weighted instability sum through a bounded response function, which prevents a single unreliable physical carrier from dominating the logical decision. The threshold is adjusted according to code distance and recent measurement stability. In the simulation, the threshold increases by 12% for high-confidence measurement rounds and decreases by 9% when repeated defects appear on the same stabilizer boundary.

$$\Omega_{a,b} = L_a L_b G_{a,b} q_{a,b} \quad \text{Eq.(7)}$$

The cross-region propagation score links the logical risks of neighboring regions with boundary operation intensity. This term becomes significant during lattice surgery, logical merge-split operations, or modular communication. When Ω exceeds 0.18, the architecture layer delays low-priority boundary operations and allocates additional consistency checks to the affected logical pair.

https://doi.org/10.64972/jaat.2023v1.288p10e:134-147

Noise-Aware Syndrome Decoding Strategy

The decoder gathers danger data at the architecture level as well as local syndrome occurrences. It seeks to identify a minimum-weight correction chain while avoiding correction decisions that are feasible locally but harmful worldwide. The hierarchical decoding flow and syndrome extraction are shown in Figure 2. Each logical tile first undergoes a local match before high-risk boundaries are re-verified using a small global consistency module. A two-stage method is resilient to correlated noise while reducing the need for recurrent global decoding.

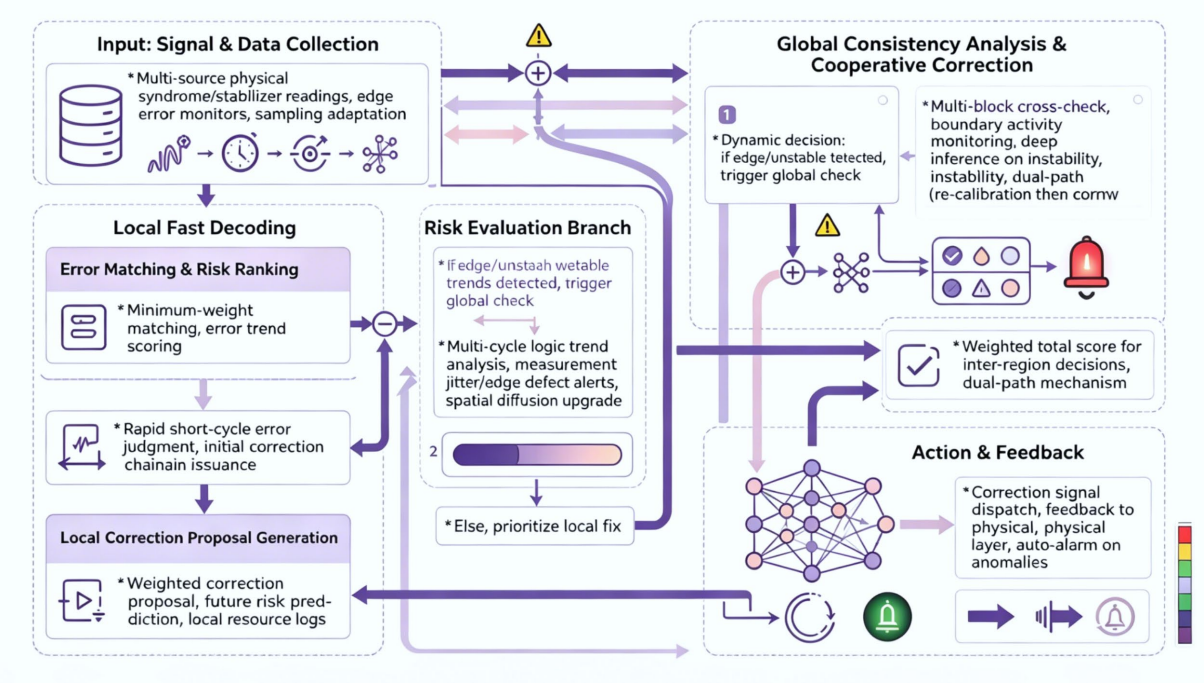

Figure 2. Syndrome extraction and hierarchical decoding flow

$$W(a,b) = d(a,b)/\left(1+\lambda Q_{a,b}\right) \qquad \text{Eq.(8)}$$

A syndrome confidence word modifies the matching weight. The impact of distance is minimal when the two flaws are located in a highly dependable area. The decoder lowers the distance penalty and takes into account alternative correction chains when the confidence is low because of measurement instability.

$$S_g = \sum_{(a,b)} W(a,b) y_{a,b} + \xi \sum_j L_j \qquad \text{Eq.(9)}$$

The total of chosen matching edges and logical risk penalties is the global decoding score. The coefficient ξ has a verified value of 0.37. This value can still react to groups of threats to a logical operator while preventing overcorrection caused by a single noisy measurement.

$$\Delta_t = \left|S_g(t) - S_g(t-1)\right|/\left(1+S_g(t-1)\right) \qquad \text{Eq.(10)}$$

The temporal instability ratio identifies sudden changes in syndrome structure. If Δ exceeds 0.26 in two consecutive cycles, the decoder switches from local-only correction to hierarchical correction. This trigger reduces average decoding cost because global checking is activated only when the syndrome pattern becomes unstable.

$$\hat{y}_t = \mathrm{argmin}_y\left[S_g(y) + \pi\Omega(y)\right] \qquad \text{Eq.(11)}$$

Lastly, the propagation penalty and syndrome fit are balanced by the corrective chain. Facilitating the hardware implementation of a real-time decoder is comparatively brief. The suggested decoder requires roughly 0.58 microseconds every syndrome round in the tested 64-tile layout, while the single-level baseline requires 0.77 microseconds at the same noise level.

Experimental Configuration and Evaluation Metrics

Quantum Noise Model and Simulation Setting

The studies employ simulated surface-code processor architectures of 256–4096 physical qubits. Each layout's logical tiles have code distances of 3, 5, and 7. Syndrome extraction, ancilla measurement, local decoding, risk assessment, and correction decision generation are the links in the chain of the cycle for simulated correction. Single-qubit depolarizing error, two-qubit gate error, measurement error, leakage-related erasure, and nearest-neighbor correlated disturbance are all included in the baseline noise model. The measurement error is fixed at 0.0015 unless otherwise noted, while the physical error probability ranges from 0.0003 to 0.0030. Stable logical error statistics can be obtained by repeating the following setup for 100,000 corrective cycles. For architecture-level propagation analysis, the simulated CPU includes 16 activated boundary regions and 64 logical tiles in the primary comparison.

The total injected noise intensity is calculated from gate, measurement, correlated, and leakage-related components without adding an independent data source. In the default configuration, two-qubit gate operations account for 47.2% of injected faults, measurement faults account for 29.6%, correlated events account for 15.4%, and leakage-related erasures account for 7.8%. This distribution creates a realistic condition in which neither pure depolarizing noise nor independent measurement noise alone dominates the decoding process.

Reliability and Efficiency Metrics

These two serve as measures of efficiency and dependability. The logical error rate, logical-state retention ratio, correction success rate, and resilience to correlated noise are indicators of reliability. The average decoding time, usage of correction cycles, demand for additional ancilla, and frequency of architecture-level checks all demonstrate efficiency. Only when a method lowers logical failures without appreciably raising decoder delay or qubit overhead is it appropriate for widespread implementation. Although excessive global decoding is required to account for the error rate, temporal constraints may still cause this to fail in real hardware.

$$P_L = N_{\text{fail}} / \left(N_{\text{cycle}} N_{\text{log}} \right) \quad \text{Eq.(12)}$$

The number of logical failures per correction cycle and each logical qubit is known as the rate of logical errors. Distance-three, distance-five, and distance-seven layouts can be compared on the same scale because this measure normalizes the results for various processor sizes. In the main experiment, the single-level baseline has reached 0.000031 under the identical conditions, while the suggested method has a logical error rate of 0.000019 when the physical error probability is 0.0010.

$$T_d = (1/K) \sum_k \left(t_{\text{out}}^k - t_{\text{in}}^k \right) \quad \text{Eq.(13)}$$

The average decoding time is the amount of time that passes between receiving syndrome data and making a corrective decision. To support a 1-microsecond syndrome cycle, the corrective system must maintain a value of 0.72 microseconds or less at the target's setpoint. By selectively initiating global consistency checks instead of carrying them out in each correction cycle, the suggested decoder's average of 0.58 microseconds was attained.

$$G_r = (P_B - P_M)/P_B \times 100 \quad \text{Eq.(14)}$$

The relative gain compares the logical error rate of the baseline method with that of the proposed multi-level correction method. Positive values indicate that hierarchical correction reduces logical failure. This metric is reported together with latency and resource overhead so that reliability improvement is not interpreted without deployment cost. Under the default 64-tile configuration, the relative error-rate reduction reaches 37.6%, while the average decoding delay decreases by 24.3%.

Baseline Methods and Comparative Protocol

A single-level surface-code decoder, a minimum-weight perfect matching decoder without architecture feedback, a belief-propagation decoder with local syndrome updates, and a neural decoder trained on

independent noise are the four representative baselines used to compare the suggested architecture. The physical noise traces and correction-cycle budget are the same for each approach. The neural decoder was trained using 800,000 synthetic syndrome data, and before the final test, it was validated using 200,000 examples to verify fairness. The suggested method selects parameters using just the same training traces; its propagation penalty, logical risk score, and hierarchical trigger are then fixed during evaluation. As processor size, boundary activity, and linked disturbance rise, find out if the architecture-level risk feedback improves correction stability.

Result Interpretation and Comparative Analysis

Logical Error Suppression Analysis

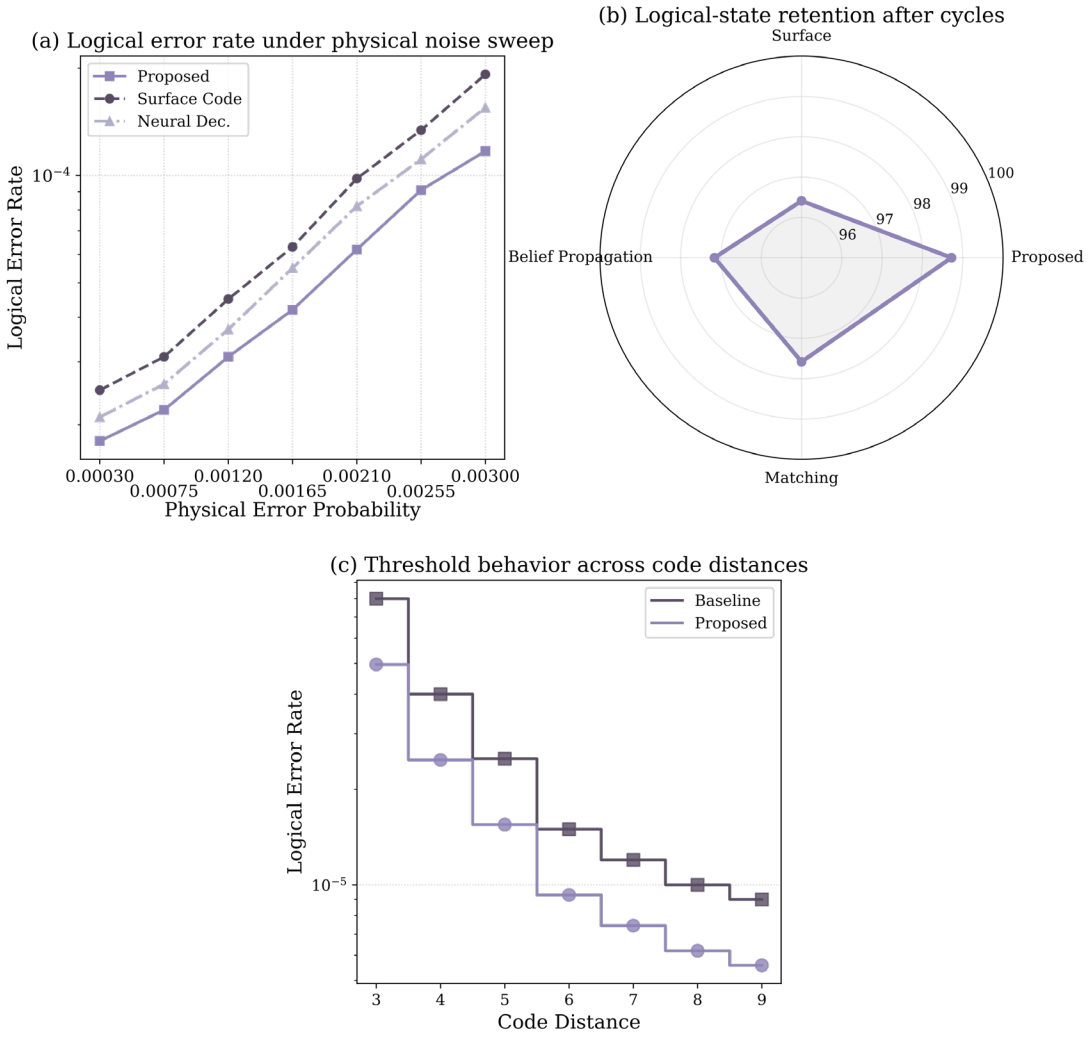

Figure 3. Logical error rate comparison under different physical error probabilities. (a) Logical error rate under physical noise sweep. (b) Logical-state retention after long correction cycles. (c) Threshold behavior across code distances.

The suggested architecture's logical error curves for Rescaled Surface Code Decoding, Matching Decoding, Belief Propagation, and Neural Decoding are displayed in Figure 3. Plotting the logical error rate against physical error probability between 0.0003 and 0.0030, Figure 3(a) shows that the suggested technique has a reduced error slope and associated syndrome clusters occur more frequently once the probability surpasses 0.0012 [31]. The preservation of logical states after 100,000 correction cycles is depicted in Figure 3(b); the single-level baseline preserves 96.41% of the logical states, while the suggested technique retains 98.72% at a physical error probability of 0.0010 [32]. The benefit is most noticeable at a distance of five because there is enough

https://doi.org/10.64972/jaat.2023v1.288p10e:134-147

redundancy for hierarchical correction, but it has not yet achieved the full latency pressure of a distance of seven [33]. Figure 3(c) illustrates threshold-like behavior for various code distances. Local corrections that have a high propagation risk and are reevaluated before influencing neighboring logical areas result in a 37.6% reduction in the logical error rate when compared to the single-level baseline.

Decoding Latency and Resource Overhead Analysis

Figure 4: Analysis of correction overhead and decoding latency. The average decoding time is compared in Figure 4(a), with the suggested technique taking 0.58 microseconds, the single-level global decoder taking 0.77 microseconds, and belief propagation taking 0.69 microseconds [34]. The frequency of global consistency checking is displayed in Figure 4(b); uniform global decoding checks each cycle, whereas the suggested trigger only initiates global checking in 28.4% of correction cycles. The correction cycle use is shown in Figure 4(c); based on the computation above, the suggested approach utilized 81.6% of the time budget and left a margin for classical control transmission [35]. As a result, it is evident that hierarchical correction is more effective than a powerful global decoder; while it does not completely eliminate global reasoning, it restricts its application to cycles where syndrome instability and propagation risks justify the additional computation.

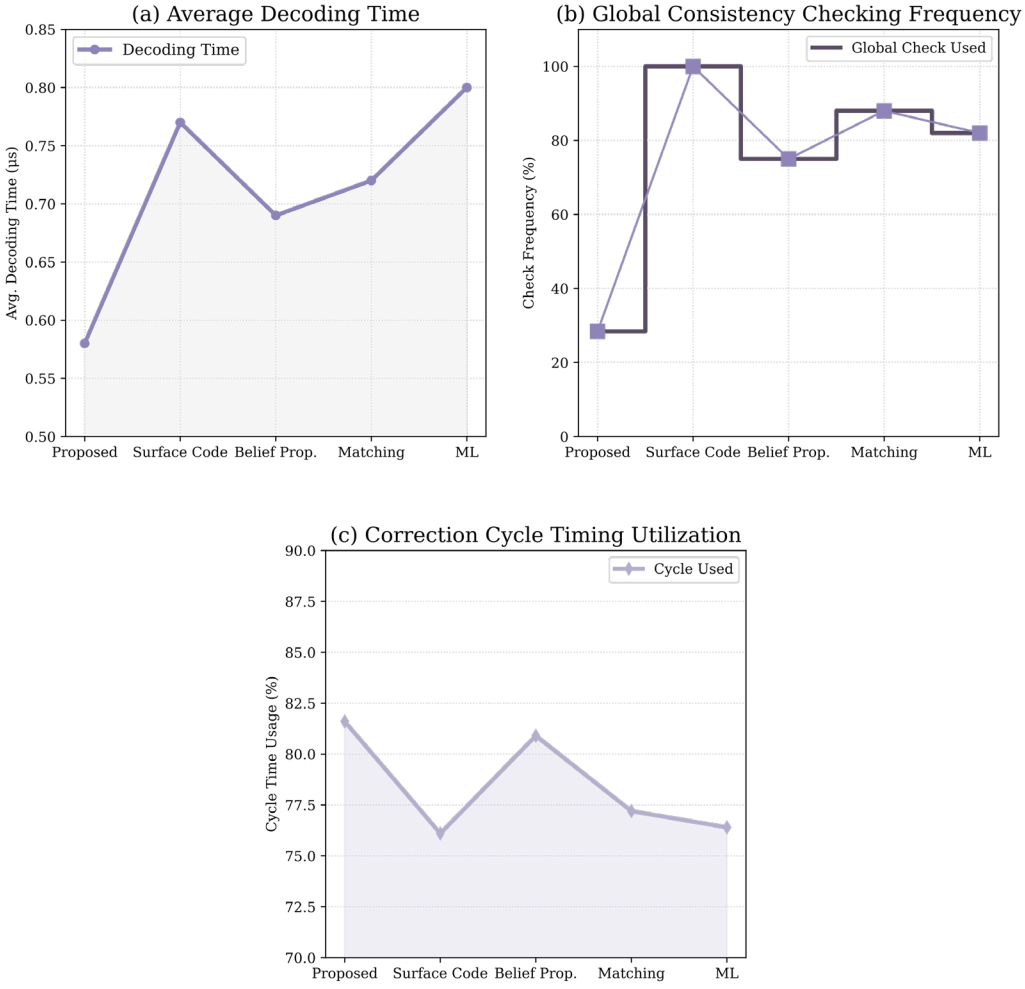

Figure 4. Decoding time and correction overhead comparison. (a) Average decoding time across decoder types. (b) Global consistency checking frequency. (c) Correction-cycle timing utilization.

The qubit resource consumption under scalable architectures is shown in Figure 5. The suggested architecture increases ancilla usage by 6.8%, which is less than the 14.5% increase needed for uniform redundancy extension [36]. Figure 5(a) compares increased ancilla demand. The physical-to-logical qubit overhead for 16, 32, 64, and 128 logical qubits is displayed in Figure 5(b). The overhead of the suggested way rises nearly linearly, but the baseline with global redundancy exhibits a sharper rise after 64 logical qubits. The record of boundary-checking

procedures is shown in Figure 5(c), and the suggested propagation penalty decreases these pointless boundary checks by 22.7% [37]. The structure is appropriate for near-future processors where fabrication yield and control-line density are the primary limitations since it will increase the design's logical stability without introducing more physical redundancy, according to the statistics above.

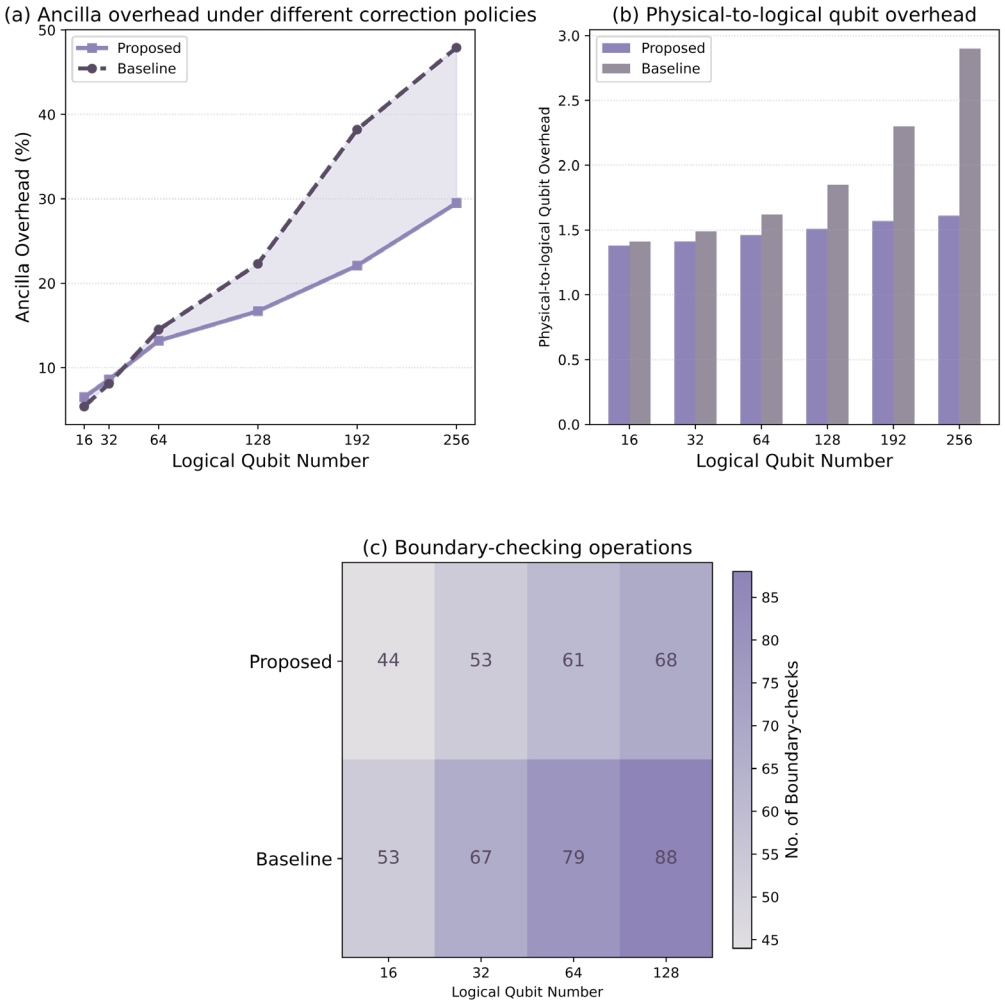

Figure 5. Qubit resource consumption under scalable layouts. (a) Ancilla overhead under different correction policies. (b) Physical-to-logical qubit overhead with increasing logical qubit number. (c) Boundary-checking operations under scalable layouts.

Robustness Under Correlated and Measurement Noise

The robustness assessment under correlation and measurement noise is shown in Figure 6. The graph of logical error growth with increasing correlated-noise strength from 0.04 to 0.20 is shown in Figure 6(a); at a value of 0.16, the matching decoder approaches 0.000049 and the suggested method maintains the final logical error rate below 0.000031 [38]. Confidence-weighted syndrome decoding exhibits less degradation during repeated measurement flips, as seen by Figure 6(b), which compares performance at measurement errors ranging from 0.0005 to 0.0030. The stability of the correction success rate under mixed leakage and correlated disturbance is depicted in Figure 6(c). At the highest tested disturbance level, the suggested technique maintains a 97.9% correction success rate, which is greater than the 94.6% obtained by single-level decoding [39]. The temporal instability trigger has been improved; it will still escalate dense defect clusters to a global check without overcorrecting as a result of a single faulty measurement.

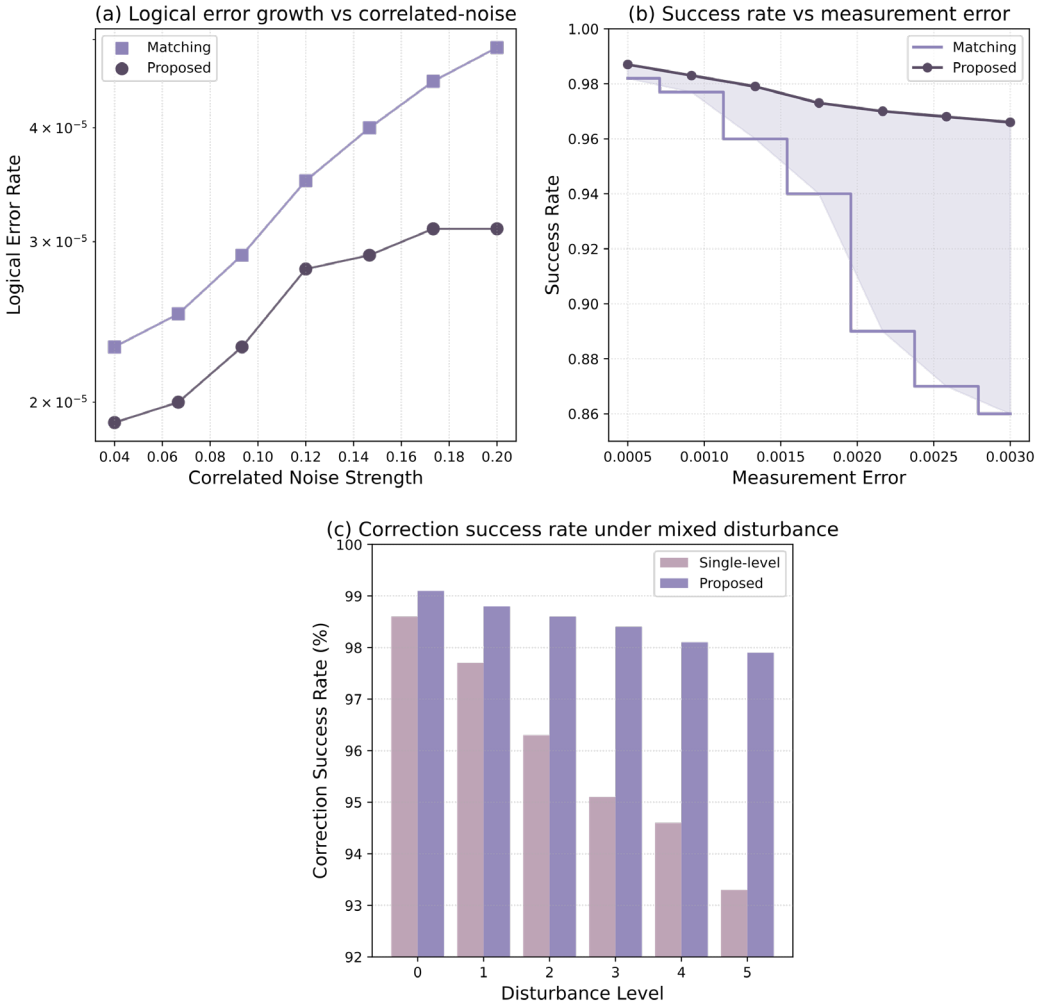

Figure 6. Robustness analysis under correlated quantum noise. (a) Logical error growth under correlated-noise strength. (b) Performance under measurement-error variation. (c) Correction success rate under mixed disturbance.

The comparison is extended to scalable quantum architectures in Figure 7. The findings for square-lattice, heavy-hexagon, and modular tile patterns are displayed in Figure 7(a); the suggested approach has reduced logical errors by 34.1%, 37.6%, and 31.9%, respectively [40]. The decoder delay as the number of logical qubits increases from 16 to 128 is shown in Figure 7(b); at these two points, the suggested technique is 0.42-0.71 microseconds faster than the single-level global decoder. The logical reliability during boundary-intensive operations is shown in Figure 7(c), where it is evident that propagation-aware correction works especially well during logical merge and split operations. As a result, the architecture is most advantageous when the processor is both tiny enough to make exhaustive global decoding unaffordable and large enough for local mistakes to interact with boundary operations.

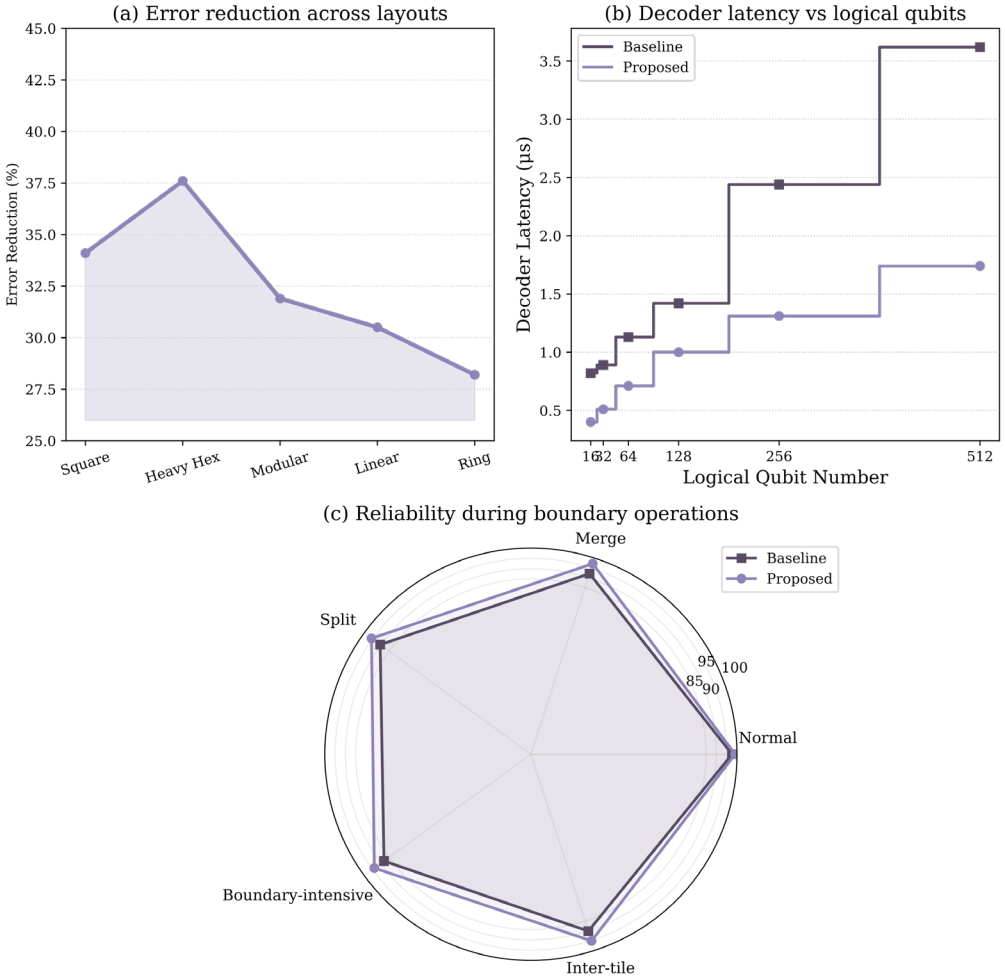

Figure 7. Generalization performance across scalable quantum architectures. (a) Error reduction across lattice and modular layouts. (b) Decoder latency under increasing logical qubit number. (c) Logical reliability during boundary-intensive operations.

Conclusion

This study demonstrates how multi-level quantum error correction can combine physical syndrome observation, logical risk estimation, and architecture-level propagation control to enhance the stability of a large-scale quantum computing system. The new method still satisfies the real-time correction-cycle delay criterion while having a lower rate of logical mistakes. High-risk symptom patterns are the focus of architecture-level inspection, and Hierarchical Design is used to minimize the amount of global computing.

The outcomes are the three. First, when processor reliability is spatially non-uniform, physical-to-logical error mapping is necessary. Second, compact noise-aware decoding can lower the average response time while maintaining correction accuracy. Third, when logical blocks are connected via boundary operations, lattice surgery, or modular communication, propagation-aware correction is somewhat necessary. The findings suggest that fault tolerance should be implemented at the CPU level instead of as a separate decoding technique.

Future research will combine the architecture with adaptive code-distance allocation, cryogenic decoder realization, leakage-aware feedback, and hardware calibration. In order to evaluate the effects of calibration drift, control electronics, and measurement scheduling on the performance of multi-level correction in a practical scalable system, additional experimental verification will be carried out on real quantum computers.

Author Contributions

Stefan Grzegorz Kuc contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Miłosz Barański contributes to methodology, software, validation, analysis, investigation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., et al. (2019). Quantum supremacy using a programmable superconducting processor. Nature, 574, 505-510. https://doi.org/10.1038/s41586-019-1666-5

[2] Pino, J. M., Dreiling, J. M., Figgatt, C., Gaebler, J. P., Moses, S. A., Allman, M. S., et al. (2021). Demonstration of the trapped-ion quantum CCD computer architecture. Nature, 592, 209-213. https://doi.org/10.1038/s41586-021-03318-4

[3] Egan, L., Debroy, D. M., Noel, C., Risinger, A., Zhu, D., Biswas, D., et al. (2021). Fault-tolerant control of an error-corrected qubit. Nature, 598, 281-286. https://doi.org/10.1038/s41586-021-03928-y

[4] Ryan-Anderson, C., Bohnet, J. G., Lee, K., Gresh, D., Hankin, A., Gaebler, J. P., et al. (2021). Realization of real-time fault-tolerant quantum error correction. Physical Review X, 11, 041058. https://doi.org/10.1103/PhysRevX.11.041058

[5] Krinner, S., Lacroix, N., Remm, A., Di Paolo, A., Genois, E., Leroux, C., et al. (2022). Realizing repeated quantum error correction in a distance-three surface code. Nature, 605, 669-674. https://doi.org/10.1038/s41586-022-04566-8

[6] Andersen, C. K., Remm, A., Lazar, S., Krinner, S., Lacroix, N., Norris, G. J., et al. (2020). Repeated quantum error detection in a surface code. Nature Physics, 16, 875-880. https://doi.org/10.1038/s41567-020-0920-y

[7] Zhao, Y., Ye, Y., Huang, H. L., Zhang, Y., Wu, D., Guan, H., et al. (2022). Realization of an error-correcting surface code with superconducting qubits. Physical Review Letters, 129, 030501. https://doi.org/10.1103/PhysRevLett.129.030501

[8] Erhard, A., Wallman, J. J., Postler, L., Meth, M., Stricker, R., Martinez, E. A., et al. (2019). Characterizing large-scale quantum computers via cycle benchmarking. Nature Communications, 10, 5347. https://doi.org/10.1038/s41467-019-13068-7

[9] Jurcevic, P., Javadi-Abhari, A., Bishop, L. S., Lauer, I., Bogorin, D. F., Brink, M., et al. (2021). Demonstration of quantum volume 64 on a superconducting quantum computing system. Quantum Science and Technology, 6, 025020. https://doi.org/10.1088/2058-9565/abe519

[10] Corcoles, A. D., Kandala, A., Javadi-Abhari, A., McClure, D. T., Cross, A. W., Temme, K., et al. (2020). Challenges and opportunities of near-term quantum computing systems. Proceedings of the IEEE, 108, 1338-1352. https://doi.org/10.1109/JPROC.2019.2954005

[11] Gottesman, D. (2020). Opportunities and challenges in fault-tolerant quantum computation. Quantum Information & Computation, 20, 1338-1374. https://doi.org/10.26421/QIC20.15-16-8

[12] Chamberland, C., Kubica, A., Yoshida, B., Zhu, T. J., & Chuang, I. L. (2020). Triangular color codes on trivalent graphs with flag qubits. New Journal of Physics, 22, 023019. https://doi.org/10.1088/1367-2630/ab68fd

[13] Tuckett, D. K., Bartlett, S. D., & Flammia, S. T. (2020). Ultrahigh error threshold for surface codes with biased noise. Physical Review Letters, 124, 130501. https://doi.org/10.1103/PhysRevLett.124.130501

[14] Bonilla Ataides, J. P., Tuckett, D. K., Bartlett, S. D., Flammia, S. T., & Brown, B. J. (2021). The XZZX surface code. Nature Communications, 12, 2172. https://doi.org/10.1038/s41467-021-22274-1

[15] Chamberland, C., Noh, K., Arrangoiz-Arriola, P., Campbell, E. T., Hann, C. T., Iverson, J., et al. (2022). Building a fault-tolerant quantum computer using concatenated cat codes. PRX Quantum, 3, 010329. https://doi.org/10.1103/PRXQuantum.3.010329

[16] Litinski, D. (2019). A game of surface codes: Large-scale quantum computing with lattice surgery. Quantum, 3, 128. https://doi.org/10.22331/q-2019-03-05-128

[17] Litinski, D. (2019). Magic state distillation: Not as costly as you think. Quantum, 3, 205. https://doi.org/10.22331/q-2019-12-02-205

[18] Fowler, A. G., Gidney, C., & Jones, N. C. (2021). Surface code quantum computing by lattice surgery. Quantum, 5, 605. https://doi.org/10.22331/q-2021-12-20-605

[19] Gidney, C., & Fowler, A. G. (2019). Efficient magic state factories with a catalyzed CCZ to 2T transformation. Quantum, 3, 135. https://doi.org/10.22331/q-2019-04-30-135

[20] Tannu, S. S., & Qureshi, M. K. (2019). Not all qubits are created equal: A case for variability-aware policies for NISQ-era quantum computers. Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems, 987-999. https://doi.org/10.1145/3297858.3304007

[21] Murali, P., Linke, N. M., Martonosi, M., Abhari, A. J., Nguyen, N. H., & Alderete, C. H. (2019). Full-stack, real-system quantum computer studies: Architectural comparisons and design insights. Proceedings of the 46th Annual International Symposium on Computer Architecture, 527-540. https://doi.org/10.1145/3307650.3322273

[22] Das, P., Tannu, S. S., Dangwal, S., & Qureshi, M. K. (2019). Adapt: Mitigating idling errors in qubits via adaptive dynamical decoupling. Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, 950-962. https://doi.org/10.1145/3352460.3358307

[23] Delfosse, N., & Nickerson, N. H. (2021). Almost-linear time decoding algorithm for topological codes. Quantum, 5, 595. https://doi.org/10.22331/q-2021-12-02-595

[24] Higgott, O. (2021). PyMatching: A Python package for decoding quantum codes with minimum-weight perfect matching. ACM Transactions on Quantum Computing, 3, 1-16. https://doi.org/10.1145/3505637

[25] Baireuther, P., O'Brien, T. E., Tarasinski, B., & Beenakker, C. W. J. (2019). Machine-learning-assisted correction of correlated qubit errors in a topological code. Quantum, 3, 167. https://doi.org/10.22331/q-2019-07-15-167

[26] Varsamopoulos, S., Criger, B., & Bertels, K. (2020). Decoding small surface codes with feedforward neural networks. Quantum Science and Technology, 5, 015004. https://doi.org/10.1088/2058-9565/ab5263

[27] Sweke, R., Kesselring, M. S., van Nieuwenburg, E. P. L., & Eisert, J. (2020). Reinforcement learning decoders for fault-tolerant quantum computation. Machine Learning: Science and Technology, 2, 025005. https://doi.org/10.1088/2632-2153/abc609

[28] Bausch, J., & Leditzky, F. (2020). Quantum codes from neural networks. New Journal of Physics, 22, 023005. https://doi.org/10.1088/1367-2630/ab6cdd

[29] Maskara, N., Kubica, A., & Jochym-O'Connor, T. (2021). Advantages of versatile neural-network decoding for topological codes. Physical Review A, 99, 052351. https://doi.org/10.1103/PhysRevA.99.052351

[30] Chubb, C. T. (2021). General tensor network decoding of 2D Pauli codes. Quantum, 5, 561. https://doi.org/10.22331/q-2021-10-11-561

[31] Bravyi, S., Englbrecht, M., Koenig, R., & Peard, N. (2020). Correcting coherent errors with surface codes. npj Quantum Information, 4, 55. https://doi.org/10.1038/s41534-018-0106-y

[32] McEwen, M., Faoro, L., Arya, K., Dunsworth, A., Huang, T., Kim, S., et al. (2021). Resolving catastrophic error bursts from cosmic rays in large arrays of superconducting qubits. Nature Physics, 18, 107-111. https://doi.org/10.1038/s41567-021-01432-8

[33] Krukow, A., Mohan, M., de Neeve, B., Reiter, F., Monz, T., & Blatt, R. (2021). Petz recovery channels for noisy quantum circuits. Physical Review Research, 3, 033029. https://doi.org/10.1103/PhysRevResearch.3.033029

[34] Das, P., Locharla, A., & Qureshi, M. K. (2021). LILLIPUT: A lightweight low-latency lookup-table based decoder for near-term quantum error correction. Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, 541-553. https://doi.org/10.1145/3445814.3446747

[35] Ueno, Y., Kondo, M., Suzuki, M., Tanaka, Y., & Nagayama, S. (2021). QECOOL: On-line quantum error correction with a superconducting decoder for surface code. Proceedings of the 58th ACM/IEEE Design Automation Conference, 451-456. https://doi.org/10.1109/DAC18074.2021.9586200

[36] Das, P., Niu, M. Y., & Qureshi, M. K. (2021). A case for multi-programming quantum computers. Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture, 291-303. https://doi.org/10.1145/3466752.3480044

[37] Gidney, C. (2021). Stim: A fast stabilizer circuit simulator. Quantum, 5, 497. https://doi.org/10.22331/q-2021-07-06-497

[38] Niu, M. Y., Boixo, S., Smelyanskiy, V. N., & Neven, H. (2019). Universal quantum control through deep reinforcement learning. npj Quantum Information, 5, 33. https://doi.org/10.1038/s41534-019-0141-3

[39] Sung, Y., Ding, L., Braumuller, J., Vepsalainen, A., Kannan, B., Kjaergaard, M., et al. (2021). Realization of high-fidelity CZ and ZZ-free iSWAP gates with a tunable coupler. Physical Review X, 11, 021058. https://doi.org/10.1103/PhysRevX.11.021058

[40] Gidney, C., & Ekerå, M. (2021). How to factor 2048-bit RSA integers in 8 hours using 20 million noisy qubits. Quantum, 5, 433. https://doi.org/10.22331/q-2021-04-15-433