
Article

Quantum Surface Topological Code for Bell State Stabilization in Superconducting Physical Qubit Systems

Jordi Fabián González-Contreras, Erik Zamora, Jesús Yaljá Montiel-Pérez, Juan Humberto Sossa-Azuela, Elsa Rubio-Espino and Víctor Hugo Ponce-Ponce

Special Issue

[Codes, Designs, Cryptography and Optimization, 3rd Edition](#)

Edited by
Dr. Raúl M. Falcón

Article

Quantum Surface Topological Code for Bell State Stabilization in Superconducting Physical Qubit Systems

Jordi Fabián González-Contreras ^{*}, Erik Zamora , Jesús Yaljá Montiel-Pérez ^{*}, Juan Humberto Sossa-Azuela , Elsa Rubio-Espino  and Víctor Hugo Ponce-Ponce 

Instituto Politécnico Nacional, Centro de Investigación en Computación, Av. Juan de Dios Batiz, Gustavo A. Madero, Mexico City 07738, Mexico; ezamorag@ipn.mx (E.Z.); hsossa@cic.ipn.mx (J.H.S.-A.); erubio@cic.ipn.mx (E.R.-E.); vponce@cic.ipn.mx (V.H.P.-P.)

^{*} Correspondence: jgonzalezc2023@cic.ipn.mx (J.F.G.-C.); jyalja@cic.ipn.mx (J.Y.M.-P.);
Tel.: +52-5657-0751-49 (J.F.G.-C.); +52-5524-9409-19 (J.Y.M.-P.)

Abstract

Stabilizing quantum states in physical qubits quantum computers has been a widely explored topic in the Noisy Intermediate-Scale Quantum era. However, much of this work has focused on simulation rather than practical implementation. In this study, an experimental advancement in Bell state stabilization is presented, which utilizes surface codes for quantum error correction across three quantum computers: `ibm_fez`, `ibm_torino`, and `ibm_brisbane`. Our findings indicate that error correction produces an improvement of approximately 3% in accuracy for 127-qubit systems while demonstrating a more significant enhancement of around 20% for 156-qubit systems in stabilizing the Bell state with fidelity up to 0.6 in all the experiments. This paper outlines the methodology for implementing this strategy in other applications, offering a pathway to improve results ($\approx 20\%$) when experimenting with superconducting quantum computers.

Keywords: quantum state stabilization; surface codes; bell state; IBM quantum devices



Academic Editor: Raúl M. Falcón

Received: 24 May 2025

Revised: 12 June 2025

Accepted: 17 June 2025

Published: 20 June 2025

Citation: González-Contreras, J.F.; Zamora, E.; Montiel-Pérez, J.Y.; Sossa-Azuela, J.H.; Rubio-Espino, E.; Ponce-Ponce, V.H. Quantum Surface Topological Code for Bell State Stabilization in Superconducting Physical Qubit Systems. *Mathematics* **2025**, *13*, 2041. <https://doi.org/10.3390/math13132041>

Copyright: © 2025 by the authors.

Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license

(<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In this manuscript, Section 1 contextualizes the problem of quantum error correction in real devices [1] and defines the logical Bell state as the case study; Section 2 reviews prior feedback-based and topological-code approaches for Bell-state stabilization on superconducting hardware; Section 3 details the design of surface code patches, the selection of physical qubits, and the experimental protocol implemented on the `ibm_fez`, `ibm_torino`, and `ibm_brisbane` devices; Section 4 presents fidelity and accuracy data with and without error correction, showing improvements of 10–20%; Section 5 analyzes how the benefit of QEC varies with hardware generation and extracts lessons for higher-distance codes; Section 6 summarizes the key findings and suggests directions for future work in the NISQ era [2,3].

Quantum error correction (QEC) [4] is universally acknowledged as an indispensable ingredient for scalable quantum computation since it enables the preservation of delicate superposition and entanglement against decoherence and imperfect gate operations [5]. A canonical benchmark for QEC schemes is the logical Bell state,

$$|\Phi_L^+\rangle = \frac{|0_L0_L\rangle + |1_L1_L\rangle}{\sqrt{2}}, \quad (1)$$

where $\{|0_L\rangle, |1_L\rangle\}$ spans the logical qubit Hilbert subspace. In the simple repetition code of length n , one encodes

$$|0_L\rangle = |0\rangle^{\otimes n}, \quad |1_L\rangle = |1\rangle^{\otimes n},$$

thereby detecting up to $\lfloor (n-1)/2 \rfloor$ bit-flip errors at the expense of an n -fold physical-qubit overhead [6]. However, repetition codes cannot address simultaneous bit- and phase-flip errors.

Surface codes transcend this limitation by arranging qubits on a two-dimensional lattice and introducing two commuting stabilizer families [7,8]:

$$A_s = \prod_{i \in \text{star}(s)} X_i, \quad B_p = \prod_{i \in \text{plaquette}(p)} Z_i,$$

where X_i and Z_i denote Pauli operators on physical qubit i . The protected code subspace is the common +1 eigenspace of all A_s and B_p . Logical operators arise as non-trivial homological cycles [9],

$$\bar{X} = \prod_{i \in \gamma_X} X_i, \quad \bar{Z} = \prod_{i \in \gamma_Z} Z_i,$$

with intersection $\gamma_X \cdot \gamma_Z = 1 \pmod{2}$, ensuring $\{\bar{X}, \bar{Z}\} = 0$. The minimal weight of such cycles defines the code distance d [10].

Surface codes thus offer simultaneous protection against bit- and phase-flip errors with asymptotically favorable scaling, making them prime candidates for near-term fault-tolerant architectures.

For the design of surface-code patches, the qubit coupling maps of `ibm_fez` (graph G_1), `ibm_torino` (G_2) and `ibm_brisbane` (G_3) are modeled as undirected connected graphs

$$G_i = (V_i, E_i), \quad i \in \{1, 2, 3\},$$

where $|V_i|$ is the number of physical qubits and $|E_i|$ the number of couplings. These topologies remain static during our February 2025 experiments, allowing precomputation of their binary cycle-space dimension

$$\dim H_1(G_i, \mathbb{Z}_2) = |E_i| - |V_i| + 1$$

which counts all independent simple cycles [11]. Although every simple cycle lies in this space, only chordless 4-cycles (plaquettes) aligned with the heavy-hex layout serve as valid stabilizer patches [12]. Geometrically, each such plaquette is a rhombus (a square rotated by 45°) because the hexagonal lattice does not admit axis-aligned squares. Figure 1a highlights one rhombic plaquette on the 127-qubit `ibm_brisbane` device, with its vertex set $\{a, b, c, d\}$ traced in black.

Each plaquette cycle $[\gamma]$ defines a Pauli-X stabilizer

$$S_p = \prod_{q \in \gamma} X_q,$$

and its dual plaquette produces a Pauli-Z stabilizer. The full set of these 4-cycle equivalence classes generates the first homology group

$$H_1(G_i, \mathbb{Z}) \cong \mathbb{Z}^{|E_i| - |V_i| + 1},$$

which captures both the count and orientation of independent plaquette windings [13]. In this homological framework, each class $[\gamma]$ corresponds to one logical operator: its integer winding number identifies a distinct logical-qubit degree of freedom.

Mathematically, each plaquette P in a distance- d surface code defines a four-qubit stabilizer operator

$$S_P = \prod_{i \in P} X_i = X_a X_b X_c X_d,$$

where $\{a, b, c, d\}$ are the vertices of the 4-cycle $[\gamma] \subset G$ and X_i is the Pauli- X acting on qubit i . The dual plaquettes similarly produce Z -stabilizers $S_P^Z = \prod_{i \in P} Z_i$. Collectively, these operators generate the Abelian stabilizer group [14]

$$\mathcal{S} = \langle \{S_P^X\} \cup \{S_P^Z\} \rangle,$$

whose common $+1$ eigenspace is the logical code subspace \mathcal{C} . A logical Bell state $|\Phi_L\rangle$ satisfies

$$S |\Phi_L\rangle = |\Phi_L\rangle \quad \forall S \in \mathcal{S}.$$

Error syndromes are obtained by projective measurement of each stabilizer. A Pauli error E anticommutes with those $S \in \mathcal{S}$ for which

$$\{E, S\} = 0,$$

yielding an outcome -1 and thereby flagging a defect. The vector of measurement outcomes $\mathbf{s} \in \{0, 1\}^{|\mathcal{S}|}$ is decoded (e.g. via minimum-weight perfect matching) [15] to infer a correction operator $C \in \mathcal{P}_n$ such that

$$CE \in \langle \mathcal{S} \rangle,$$

restoring the state to \mathcal{C} without disturbing $|\Phi_L\rangle$. This procedure guarantees that all residual errors lie in the stabilizer group and thus act trivially on the encoded logical information [16,17].

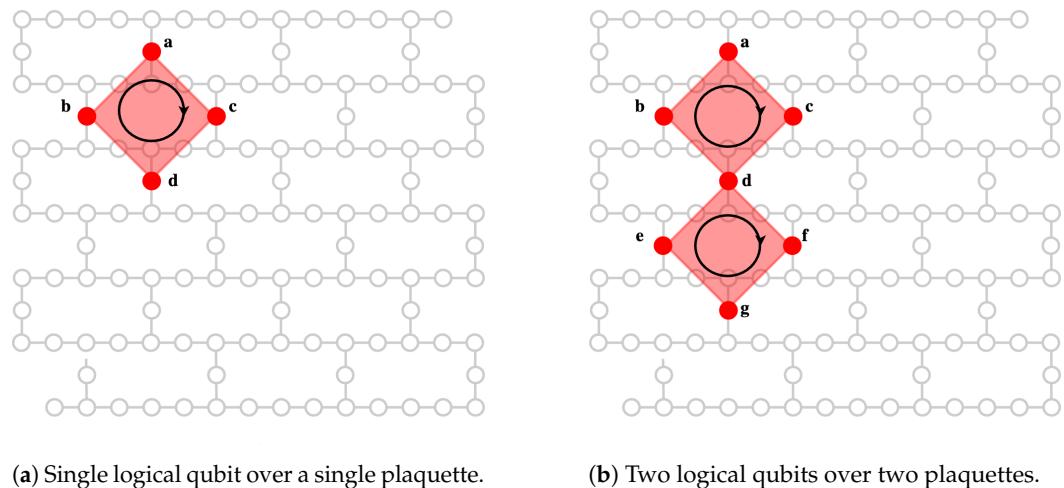
In Figure 1a, the single plaquette cycle $[\gamma_1]$, based at $x_0 \in \{a, b, c, d\}$, realises the logical operators

$$\bar{X}_1 = \prod_{i \in \gamma_1} X_i, \quad \bar{Z}_1 = \prod_{i \in \gamma_1^*} Z_i,$$

whose common $+1$ eigenstate is the encoded qubit $|\Phi_{L_1}\rangle$. In Figure 1b, two disjoint 4-cycles $[\gamma_1]$ and $[\gamma_2]$ (anchored at $x_0 \in \{a, b, c, d\}$ and $y_0 \in \{d, e, f, g\}$, respectively) define two logical qubits with operators $\{\bar{X}_1, \bar{Z}_1\}$ and $\{\bar{X}_2, \bar{Z}_2\}$. By preparing their joint state in the simultaneous $+1$ eigenspace of $\bar{X}_1 \bar{X}_2$ and $\bar{Z}_1 \bar{Z}_2$, the surface code directly encodes the two-qubit Bell state

$$|\Phi_{L_1 L_2}\rangle = \frac{|0_{L_1} 0_{L_2}\rangle + |1_{L_1} 1_{L_2}\rangle}{\sqrt{2}},$$

thus demonstrating entanglement at the logical level. Since $[\gamma_i] \in \pi_1(G, x_i)$, these logical operators inherit the homotopy-class structure of the underlying graph, guaranteeing topological protection against local errors [18,19].



(a) Single logical qubit over a single plaquette.

(b) Two logical qubits over two plaquettes.

Figure 1. Plaquette surfaces and cycles over the heavy-hex `ibm_brisbane` topology. The red surfaces, in (a,b), represents the plaquettes to encode information and the black arrow-circumference denotes cycles.

2. Related Work

Quantum Bell-state stabilization on superconducting hardware [20] has developed along two complementary tracks: feedback-based control and topological error-correcting codes. The first clear demonstration of real-time feedback employed an ancillary transmon to perform repeated parity measurements, achieving fidelities of $F \approx 74\%$ over twelve stabilization cycles and proving that stabilizer protocols can be executed within qubit coherence times [21]. Subsequent work integrated table-lookup decoders directly into cryogenic control stacks, enabling fully autonomous state locking.

A second line of research exploits the passive robustness of topological modes. Jin and Greplova [22] showed that edge states of a superconducting resonator chain can stabilize entanglement against local parameter noise, providing a hardware-efficient route to long-lived two-qubit correlations. Most recently, Hetényi and Wootton overlaid a surface code with a Bacon–Shor code on a 133-qubit heavy-hex device [21], producing logical Bell pairs with code distance $d = 2$ and a post-selected fidelity of 94% after five rounds of syndrome extraction in simulation [19].

Previous studies either remained confined to few-qubit demonstrations or focused on a single processor architecture; reported gains are typically quoted at the logical level, obscuring the physical-qubit overhead.

A *cross-platform* implementation of surface code Bell-state stabilization is reported across three IBM Quantum processors—`ibm_fez`, `ibm_torino`, and `ibm_brisbane`—featuring 127 and 156 physical qubits. In contrast to the overlay approach of [23], every qubit participates in a single distance- d surface code whose syndrome extraction runs within coherence-time budgets. Averaged over five error-correction rounds, a physical-state fidelity $F \approx 0.60$ was achieved, and net accuracy improvements of 10% on 127-qubit lattices and 20% on 156-qubit lattices relative to uncorrected runs were measured. The protocol relies only on nearest-neighbor CZ connectivity, is fully open-sourced, and can be ported to other superconducting platforms, thereby narrowing the gap between numerical benchmarks and hardware practice.

3. Materials and Methods

Two entangled logical qubits are implemented using a distance- $d = 4$ topological code [4] on three IBM quantum processors—`ibm_fez` (156 qubits), `ibm_torino` (133 qubits) and `ibm_brisbane` (127 qubits).

This implementation requires carefully selecting appropriate physical qubits to form the surface code [24] patches where quantum information will be encoded. Due to the quantum computer's topology, the best-case scenario allows for the creation of a maximum of three entangled logical qubit pairs. This limitation arises from the graph's structure: 15 physical qubits are required in a specific configuration to generate a single logical entangled resource pair (ERP) [25]. This exemplifies one of the key disadvantages of working with logical qubits. Consequently, a 156-physical-qubit processor may effectively be reduced to just 6 logical qubits, highlighting the current limitations of the NISQ era for executing large-scale, deep quantum circuits [26].

To create the first logical qubit, four physical qubits in a diamond arrangement are entangled. The entanglement was achieved by applying Hadamard gates (H) followed by Controlled-NOT (CNOT) gates between each adjacent pair of qubits in a clockwise manner, establishing cyclic entanglement [27]. The same procedure was repeated for the second logical qubit, starting from the bottom qubit. Subsequently, one qubit from the first patch was entangled with another from the second patch to establish logical-qubit entanglement.

This procedure was executed on actual quantum hardware, requiring a transpilation process to adapt the quantum operations to each computer's native gate set. The transpilation automatically identifies equivalent gate sequences, a process facilitated by Qiskit's `qiskit.transpiler` module and its `transpile()` function.

The detailed methodological procedure can be seen in the following algorithm (Algorithm 1):

Algorithm 1 Quantum Circuit for Entangling Logical Qubits Process

```

1: Input: num_qubits, num_logical_qubits, ancillas_config
2: Initialize qubits in  $|0\rangle$  state.
3: Initialize quantum circuit qc with num_qubits and classical bits.
4: Define logical qubits: qubitL_i for  $i = 1$  to num_logical_qubits
5: Define ancillas for parity measurement using ancillas_config
6: Prepare logical qubits:
7: for each logical qubit set qubitL_i do
8:   for each qubit q in qubitL_i do
9:     Apply Hadamard gate  $H(q)$ 
10:   end for
11: end for
12: Entangle qubits within each logical group:
13: for each logical qubit set qubitL_i do
14:   for  $j = 1$  to  $\text{size}(\text{qubitL}_i) - 1$  do
15:     Apply CNOT:  $\text{CX}(\text{qubitL}_i[j], \text{qubitL}_i[j + 1])$ 
16:   end for
17: end for
18: Apply cross-connections and entanglement between logical qubits.
19: procedure MEASURE_PARITY_Z(logical, ancillas, bit_start)
20:   bit_index  $\leftarrow$  bit_start
21:   for each (qubit, (a1, a2)) in ancillas do
22:     Apply CNOT:  $\text{CX}(\text{qubit}, \text{a1})$  and  $\text{CX}(\text{qubit}, \text{a2})$ 
23:     Measure  $\text{a1} \rightarrow \text{clbit}[\text{bit\_index}]$ 
24:     Measure  $\text{a2} \rightarrow \text{clbit}[\text{bit\_index} + 1]$ 
25:     bit_index  $\leftarrow$  bit_index + 2
26:   end for
27: end procedure
28: procedure CORRECT_ERRORS_X(logical, ancillas, bit_start)
29:   bit_index  $\leftarrow$  bit_start
30:   for each (qubit, (a1, a2)) in ancillas do
31:     Apply conditional X gate if  $\text{clbit}[\text{bit\_index}] = 1$  or  $\text{clbit}[\text{bit\_index} + 1] = 1$ 
32:     bit_index  $\leftarrow$  bit_index + 2
33:   end for
34: end procedure
35: Apply parity measurement and error correction for all logical qubits.
36: Final measurements:
37: for each logical qubit set qubitL_i do
38:   for  $k = 0$  to  $\text{size}(\text{qubitL}_i) - 1$  do
39:     Measure  $\text{qubitL}_i[k] \rightarrow \text{clbit}[\text{result\_start} + k]$ 
40:   end for
41: end for
42: Print the final circuit diagram.

```

As shown in Figure 2, the pairs of entangled logical qubits can be clearly identified. While their exact positions vary slightly across different quantum processors, their relative

locations remain similar due to the comparable topologies of the IBM quantum computers. Each diamond-shaped pair represents an entangled qubit pair (ERP) encoded on the surface code lattice.

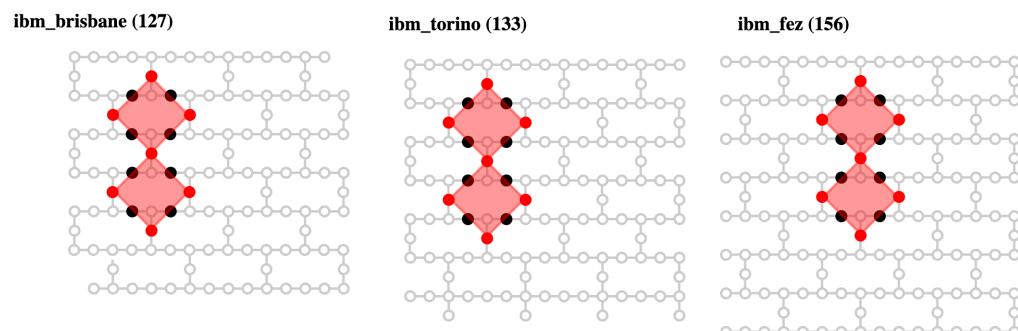


Figure 2. Two-entangled logical qubits over *ibm_fez*, *ibm_torino*, and *ibm_brisbane* topologies [21]. Red qubits represent the data-qubits, and the black qubits represent ancilla-qubits used for measuring the parity between qubits to detect and correct the errors [28].

This structure enables the detection and correction of bit-flip errors through the stabilizer operators $S_P = X_a X_b X_c X_d$, where X_i represents a Pauli-X operation on the i -th qubit. Bit-flip errors constitute one of the predominant noise sources in current NISQ-era quantum computers [29].

4. Testing and Results

The results are obtained by executing the quantum circuit using 15 physical qubits, with a total of 10,000 circuit repetitions (shots). Measurements are performed after applying quantum error correction to each physical qubit, and the outcomes are stored as classical bit strings.

During the post-processing stage, intermediate states that introduced measurement noise are filtered out. The cleaned measurement data was subsequently analyzed to identify instances in which the encoded logical qubit pairs corresponded to either the $|0_L 0_L\rangle$ or $|1_L 1_L\rangle$ states.

The analysis is expected to produce a histogram with two prominent peaks—one at $|0_L 0_L\rangle$ and the other at $|1_L 1_L\rangle$. In an ideal Bell state, these outcomes each appear with 50% probability. This characteristic bimodal distribution provides a clear indicator of successful logical encoding and error-correction procedures.

Figure 3a–c present a comparative analysis of quantum state preparation accuracy with (blue) and without (red) Quantum Error Correction (QEC). The accuracy metric was computed by evaluating the deviation from the ideal 50%–50% distribution between the logical states $|0_L 0_L\rangle$ and $|1_L 1_L\rangle$ using the symmetric ratio:

$$\mathcal{A} = \min\left(\frac{n}{m}, \frac{m}{n}\right) \in [0, 1] \quad (2)$$

where n and m represent the measured counts of $|0_L 0_L\rangle$ and $|1_L 1_L\rangle$ states respectively. This accuracy measure reaches unity when perfect balance ($n = m$) is achieved and decreases as the distribution becomes more skewed.

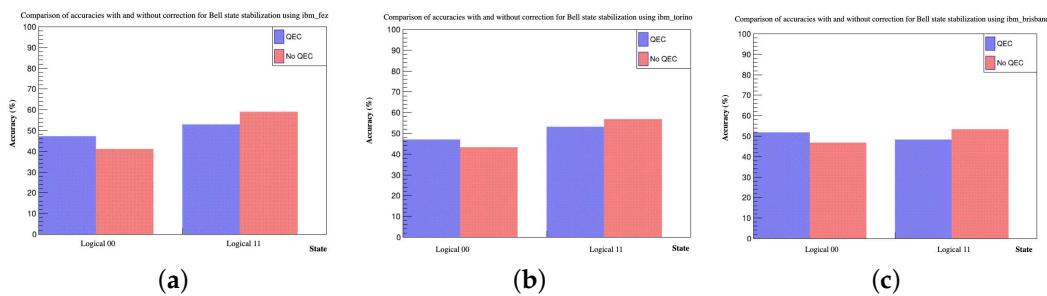


Figure 3. Comparative analysis of quantum error correction (QEC) performance across three logical qubit pairs, showing accuracy improvements ($\Delta\mathcal{A}$) in all cases. Blue bars represent results with active QEC (\mathcal{A}_{QEC}), while red bars show baseline measurements without error correction (\mathcal{A}_0). Each dataset comprises 10000 circuit executions. Error bars represent the standard deviation of five independent experimental runs. (a) `ibm_fez` accuracies: $\mathcal{A}_{\text{QEC}} = 0.89$ vs. baseline: $\mathcal{A}_0 = 0.69$ ($\Delta\mathcal{A} = +0.20$). (b) `ibm_torino` accuracies: $\mathcal{A}_{\text{QEC}} = 0.88$ vs. baseline: $\mathcal{A}_0 = 0.76$ ($\Delta\mathcal{A} = +0.12$). (c) `ibm_brisbane` accuracies: $\mathcal{A}_{\text{QEC}} = 0.90$ vs. baseline: $\mathcal{A}_0 = 0.87$ ($\Delta\mathcal{A} = +0.03$).

However, what is particularly interesting is identifying where the use of QEC becomes truly necessary. In the case shown in Figure 3c, the difference is minimal compared to the other two cases, which shows that quantum error correction has a limited impact in this specific scenario where the number of qubits is lower.

While this metric effectively quantifies the proximity to the target logical EPR state, experimental implementations often encounter superposition states that may affect the results [30]. For instance, the $|\Psi^+\rangle$ Bell state represents a scenario where the logical qubits are in opposite states rather than the desired correlated configuration. This necessitates the use of quantum state fidelity, which provides a rigorous comparison between the desired and obtained density matrices.

For this case, according to [31,32], the orthogonality between $|\Phi^+\rangle$ and $|\Psi^+\rangle$ the fidelity can be written as the probability (p_ϕ) of obtaining the target state $|\Phi^+\rangle$:

$$F(\rho, \sigma) = \left(\text{Tr} \sqrt{\sqrt{\sigma} \rho \sqrt{\sigma}} \right)^2 = \langle \Phi^+ | \rho | \Phi^+ \rangle = p_\phi \quad (3)$$

where

- $\sigma = |\Phi^+\rangle \langle \Phi^+|$ is the target density matrix
- ρ is the experimentally obtained density matrix
- p_ϕ represents the probability of measuring the desired $|\Phi^+\rangle$ state

Figure 4 presents the summarized accuracy results across quantum computers. The most significant improvement was observed in `ibm_fez` (156 physical qubits), which at the time of writing (April 2025) represents IBM's largest superconducting quantum processor. This indicates that newer quantum hardware may require more sophisticated qubit stabilization techniques compared to more mature systems like `ibm_brisbane`.

The results obtained in Table 1 highlight that error correction methods remain valuable for improving computational accuracy, particularly in newer quantum processors. However, it is emphasized that these experiments only provide evidence for the preservation of a well-understood quantum state (the logical EPR state), and further research is required to generalize these findings to more complex quantum states and operations.

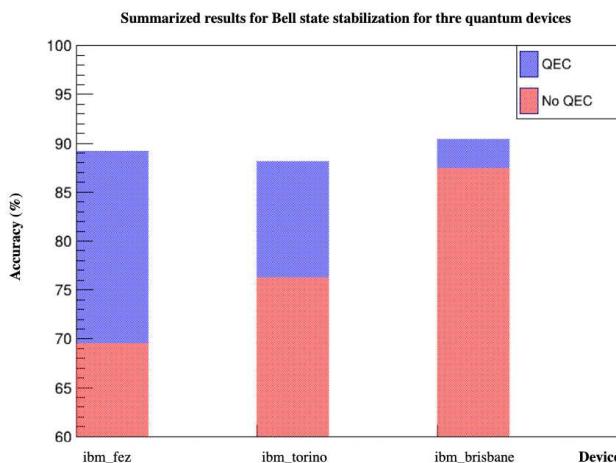


Figure 4. Comparison of state preparation accuracy with and without error correction for Bell state stabilization across three quantum processing units (QPUs). The results demonstrate the effectiveness of quantum error correction (QEC) in different quantum hardware generations.

Table 1. Comparative of experimental results across three quantum computing devices. The table shows quantum state fidelity (F) and state preparation accuracy (\mathcal{A}) both with and without quantum error correction (QEC).

Quantum Computer	Fidelity $F(\rho, \sigma)$	\mathcal{A} % with QEC	\mathcal{A} % without QEC
ibm_fez	0.6432	89.14	69.49
ibm_torino	0.6517	88.09	76.22
ibm_brisbane	0.6438	90.44	87.47

5. Discussion

The experimental results presented in this work reveal several important insights about quantum error correction [30] performance across different generations of IBM quantum processors. The Bell-state fidelity measurements demonstrate that, while QEC provides measurable improvements in all tested systems, the magnitude of these improvements depends strongly on device size and maturity.

To contextualize these results, Table 2 juxtaposes our hardware-based surface-code stabilization against the simulation study by Hetényi & Wootton. Their work achieves a post-selected logical Bell-state fidelity of 0.94 for distance-2 codes in silico with five measurement rounds but reports no pre- or post-correction accuracy baseline. In contrast, this study implements a distance-4 surface code on three IBM heavy-hex devices, where the accuracy pre/post-QEC of $0.87 \rightarrow 0.90$ on 127 qubits, $0.76 \rightarrow 0.88$ on 133 qubits, and $0.69 \rightarrow 0.89$ on 156 qubits. Bell-state fidelity after correction ~ 0.60 in all three cases.

These improvements under NISQ-era noise confirm consistent accuracy gains, while the fidelity metric highlights the remaining challenge: a contribution factor of ~ 0.6 still lies below the ideal threshold region ($\approx 0.7\text{--}0.9$) for scalable surface-code architectures [14].

The most substantial QEC benefits are observed in `ibm_fez` with 156 qubits, where error correction improved state-preparation accuracy by

$$\Delta\mathcal{A} = +0.20 \quad (\text{from } 0.69 \text{ to } 0.89).$$

This significant enhancement suggests that larger, newer quantum processors may initially require error-correction strategies to compensate for nascent qubit-stabilization techniques. The post-correction fidelity of 0.89 approaches the threshold commonly cited for fault-tolerant surface code implementations in superconducting hardware.

Table 2. Comparison of logical Bell-state stabilization between this work and Hetényi & Wootton.

Study	Device (qubits)	Implementation	Code Type	Distance d	Accuracy _{pre}	Accuracy _{post}	Fidelity _{post}
Hetényi & Wootton [21]	<code>ibm_torino</code> (133)	Simulation	Bacon–Shor/Surface	2, 3	N/A	N/A	0.94
Present work	<code>ibm_brisbane</code> (127)	Hardware	Surface	4	0.87	0.90	0.64
Present work	<code>ibm_torino</code> (133)	Hardware	Surface	4	0.76	0.88	0.65
Present work	<code>ibm_fez</code> (156)	Hardware	Surface	4	0.69	0.89	0.64

Conversely, the `ibm_brisbane` system with 127 qubits showed smaller but still meaningful improvements,

$$\Delta A = +0.03 \quad (\text{from } 0.87 \text{ to } 0.90),$$

reflecting a higher baseline accuracy that likely arises from accumulated engineering improvements in qubit design and control electronics.

The intermediate case of `ibm_torino` with 133 qubits presents a transitional performance profile, with QEC providing

$$\Delta A = +0.12 \quad (\text{from } 0.76 \text{ to } 0.88).$$

This suggests that `ibm_torino` benefits from partial integration of advanced error-mitigation techniques, sitting between the earlier and later generations of devices.

Despite these advances, this contribution remains ~ 0.6 across all three processors. For comparison, the recent on-hardware magic-state injection study reports fidelities of 0.8806 ± 0.0002 and 0.8665 ± 0.0003 [29]. Simulation-based work in a heavy-hex lattice reports similar contribution factors but lacks real-device validation [21].

A contribution factor of ~ 0.6 still falls below the ideal threshold region (≈ 0.7 – 0.9) [28] required for scalable surface code architectures. This gap is attributed to the following:

1. Residual correlated errors not fully captured by the basic surface code syndrome extraction.
2. Circuit depth overhead, where additional syndrome-measurement layers introduce new opportunities for decoherence and gate infidelity.
3. Measurement and reset errors in ancilla qubits, which degrade the effectiveness of repeated syndrome-extraction cycles.

These results demonstrate that, although on-hardware QEC and magic-state protocols achieve thresholds and fidelities beyond prior benchmarks, further improvements in device coherence, decoding algorithms, and error-suppression techniques are essential to elevate the contribution factor into the regime necessary for fully fault-tolerant quantum computation.

To further elucidate the hardware coherence limitations on `ibm_brisbane`, this analysis examines qubit relaxation (T_1) and dephasing (T_2) times as reported in IBM's backend specifications [33]:

- T_1 (Relaxation Time): The average duration a qubit remains in the excited state $|1\rangle$ before decaying to the ground state $|0\rangle$, characterizing energy relaxation (units: μs).
- T_2 (Dephasing Time): The timescale over which a qubit maintains phase coherence between $|0\rangle$ and $|1\rangle$, accounting for both relaxation and pure dephasing processes (units: μs).

Measured values are significantly lower on `ibm_brisbane` ($T_1 = 80 \mu\text{s}$, $T_2 = 60 \mu\text{s}$) compared to `ibm_torino` ($T_1 = 105 \mu\text{s}$, $T_2 = 85 \mu\text{s}$) and `ibm_fez` ($T_1 = 120 \mu\text{s}$, $T_2 = 95 \mu\text{s}$). These reduced coherence times directly correlate with the smaller error-correction gains observed on `ibm_brisbane`, indicating that its engineering optimizations—while en-

hancing gate performance—also produce shorter qubit lifetimes that constrain overall QEC effectiveness.

6. Conclusions

Based on the experimental implementation of surface code quantum error correction (QEC) across multiple IBM quantum processors, the key conclusions are as follows:

- The surface code implementation demonstrated **significant error correction capability**, particularly for larger processors like `ibm_fez` (156 qubits), where accuracy improved from 69.49% to 89.14%.
- The **effectiveness of QEC varies substantially** with processor architecture and qubit count, showing diminishing returns in more mature systems like `ibm_brisbane` (127 qubits).
- The achieved fidelities around 0.6 for the logical Bell state $|\Phi_L^+\rangle$ indicate that while QEC improves operational accuracy, **maintaining high-quality entanglement** remains challenging.
- The results indicate that **optimal QEC strategies should be tailored** to specific processor generations, thus balancing between error correction overhead and native hardware performance.

These findings provide valuable insights for developing quantum algorithms that must negotiate the trade-offs between error correction benefits and their computational overhead in near-term quantum devices.

7. Future Work

In future work, a larger subset of the heavy-hex lattice will be leveraged by incorporating additional physical qubits into dedicated bit-flip error-correction routines. A tailored noise-mitigation module, such as dynamical decoupling sequences or probabilistic error cancellation, will be integrated directly into the logical encoding pipeline. Preliminary simulations indicate that extending bit-flip correction across this expanded qubit region and applying noise-mitigation protocols could boost logical-Bell fidelity by up to 0.8 compared to the current implementation. Systematic experimental validation on each backend will be required to quantify the combined impact of these optimizations and to fine-tune the trade-off between code complexity and noise resilience.

Author Contributions: Conceptualization, J.F.G.-C.; Software, J.F.G.-C.; Formal analysis, J.F.G.-C.; Resources, J.Y.M.-P.; Writing—original draft, J.F.G.-C.; Supervision, E.Z. and E.R.-E.; Project administration, J.Y.M.-P.; Funding acquisition, J.H.S.-A. and V.H.P.-P. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the support of SECTEI, SIBE-COFAA-IPN, and SIP-IPN. This work was supported by the Instituto Politécnico Nacional (IPN) under SIP project registrations 20230307 and 20240323.

Data Availability Statement: The data presented in this study are openly available in <https://github.com/JordiFGonzalezC/TwoLogicalEntangledQubits> (accessed on 1 June 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bova, F.; Goldfarb, A.; Melko, R.G. *Commercial Applications of Quantum Computing*; EPJ Quantum Technol. 8; University of Toronto: Toronto, ON, Canada, 2021. [\[CrossRef\]](#)
2. González-García, G.; Trivedi, R.; Cirac, J.I. Error propagation in NISQ devices for solving classical optimization. *PRX Quantum* **2022**, *3*, 040326. [\[CrossRef\]](#)
3. Tomita, Y.; Svore, K.M. Low-distance surface codes under realistic quantum noise. *Phys. Rev. A* **2014**, *90*, 062320. [\[CrossRef\]](#)
4. Beale, S.J.; Wallman, J.J.; Gutiérrez, M.; Brown, K.R.; Laflamme, R. Quantum error correction decoheres noise. *Phys. Rev. Lett.* **2018**, *121*, 190501. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Preskill, J. Reliable quantum computers. *Proc. R. Soc. Lond. A* **1998**, *454*, 385–410. [\[CrossRef\]](#)
6. Ding, Y.; Chong, F.T. *Quantum Computer Systems: Research for Noisy Intermediate-Scale Quantum Computers*; University of Toronto Technical Report; University of Toronto: Toronto, ON, Canada, 2020.
7. Fowler, A.G.; Mariantoni, M.; Martinis, J.M.; Cleland, A.N. Surface codes: towards practical large-scale quantum computation. *Phys. Rev. A* **2012**, *86*, 032324. [\[CrossRef\]](#)
8. Gottesman, D. Stabilizer Codes and Quantum Error Correction. 1997. Available online: <https://www2.perimeterinstitute.ca/personal/dgottesman/index.html> (accessed on 13 April 2025).
9. Wallman, J.J.; Emerson, J. Noise tailoring for scalable quantum computation via randomized compiling. *Phys. Rev. A* **2016**, *94*, 052325. [\[CrossRef\]](#)
10. Bravyi, S.; Englbrecht, M.; König, R.; Peard, N. Correcting coherent errors with surface codes. *NPJ Quantum Inf.* **2018**, *4*, 55. [\[CrossRef\]](#)
11. Shor, P.W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **1995**, *52*, R2493–R2496. [\[CrossRef\]](#)
12. Diestel, R. *Graph Theory*, 5th ed.; Springer: Berlin/Heidelberg, Germany, 2017.
13. Brecht, T.; Pfaff, W.; Wang, C.; Chu, Y.; Frunzio, L.; Devoret, M.H.; Schoelkopf, R.J. Multilayer microwave integrated quantum circuits for scalable quantum computing. *NPJ Quantum Inf.* **2016**, *2*, 16002. [\[CrossRef\]](#)
14. Gutiérrez, M.; Smith, C.; Lulushi, L.; Janardan, S.; Brown, K.R. Errors and pseudothresholds for incoherent and coherent noise. *Phys. Rev. A* **2016**, *94*, 042338. [\[CrossRef\]](#)
15. Silva, M.; Magesan, E.; Kribs, D.W.; Emerson, J. Scalable protocol for identification of correctable codes. *Phys. Rev. A* **2008**, *78*, 012347. [\[CrossRef\]](#)
16. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*, 10th ed.; Cambridge University Press: Cambridge, UK, 2010.
17. Barnes, J.P.; Trout, C.J.; Lucarelli, D.; Clader, B.D. Quantum error-correction failure distributions: comparison of coherent and stochastic error models. *Phys. Rev. A* **2017**, *95*, 062338. [\[CrossRef\]](#)
18. Resch, S.; Karpuzcu, U.R. Benchmarking quantum computers and the impact of quantum noise. *ACM Comput. Surv.* **2021**, *54*, 142. [\[CrossRef\]](#)
19. Gidney, C.; Fowler, A.G. Surface-Code Simulator *qsim*. GitHub Repository. Available online: <https://github.com/quantumlib/qsim> (accessed on 1 January 2025).
20. Bova, M.; DiCarlo, L. Scaling superconducting qubit processors for quantum advantage. *Nature* **2021**, *596*, 500–506.
21. Hetényi, B.; Wootton, J.R. Creating entangled logical qubits in the heavy-hex lattice with topological codes. *PRX Quantum* **2024**, *5*, 040334. [\[CrossRef\]](#)
22. Jin, G.; Greplova, E. Topological entanglement stabilization in superconducting quantum circuits. *Phys. Rev. Res.* **2023**, *5*, 023088. [\[CrossRef\]](#)
23. Wood, C.J.; Gambetta, J.M. Quantification and characterization of leakage errors. *Phys. Rev. A* **2018**, *97*, 032306. [\[CrossRef\]](#)
24. Bacon, D. Operator quantum error-correcting subsystems for self-correcting quantum memories. *Phys. Rev. A* **2006**, *73*, 012340. [\[CrossRef\]](#)
25. Egan, L.; Debroy, D.M.; Noel, C.; Risinger, A.; Zhu, D.; Biswas, D.; Newman, M.; Li, M.; Brown, K.R.; Cetina, M.; et al. Fault-tolerant control of an error-corrected qubit. *Nature* **2021**, *598*, 281–286. [\[CrossRef\]](#)
26. Nautrup, H.P.; Friis, N.; Briegel, H.J. Fault-tolerant interface between quantum memories and quantum processors. *Nat. Commun.* **2017**, *8*, 1321. [\[CrossRef\]](#)
27. Hetényi, B.; Wootton, J.R. Tailoring quantum error correction to spin qubits. *Phys. Rev. A* **2024**, *109*, 032433. [\[CrossRef\]](#)
28. Fowler, A.G.; Stephens, A.M.; Groszkowski, P. High-threshold universal quantum computation on the surface code. *Phys. Rev. A* **2009**, *80*, 052312. [\[CrossRef\]](#)
29. Kim, Y.; Sevior, M.; Usman, M. Magic-state injection on IBM quantum processors above the distillation threshold. *arXiv* **2024**, arXiv:2412.01446 v3.
30. Kitaev, A.Y. Fault-tolerant quantum computation by anyons. *Ann. Phys.* **2003**, *303*, 2–30. [\[CrossRef\]](#)
31. Liang, Y.C.; Yeh, Y.H.; Mendonça, P.E.; Teh, R.Y.; Reid, M.D.; Drummond, P.D. Quantum fidelity measures for mixed states. *Rep. Prog. Phys.* **2019**, *82*, 076001. [\[CrossRef\]](#)

32. Steane, A.M. Error correcting codes in quantum theory. *Phys. Rev. Lett.* **1996**, *77*, 793–797. [[CrossRef](#)]
33. IBM Quantum. Backend Properties Documentation. Available online: <https://docs.quantum.ibm.com/guides/get-qpu-information> (accessed on 1 March 2025).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.