

ECCentric: An Empirical Analysis of Quantum Error Correction Codes*

Aleksandra Świerkowska
aleksandra.swierkowska@tum.de
Technical University of Munich
Germany

Jannik Pflieger
Technical University of Munich
Germany

Emmanouil Giortamis
Technical University of Munich
Germany

Pramod Bhatotia
Technical University of Munich
Germany

Abstract

Quantum Error Correction (QEC) is essential for building scalable quantum computers, but a lack of systematic, end-to-end evaluation methods makes it difficult to assess how different QEC codes perform under realistic conditions. The vast diversity of codes, an expansive experimental search space, and the absence of a standardized framework prevent a thorough, holistic analysis. To address this, we introduce ECCentric, an end-to-end benchmarking framework designed to systematically evaluate QEC codes across the full quantum computing stack. ECCentric is designed to be modular, extensible, and general, allowing for a comprehensive analysis of QEC code families under varying hardware topologies, noise models, and compilation strategies.

Using ECCentric, we conduct the first systematic benchmarking of major QEC code families against realistic, mid-term quantum device parameters. Our empirical analysis reveals that intra-QPU execution significantly outperforms distributed methods, that qubit connectivity is a far more critical factor for reducing logical errors than increasing code distance, and that compiler overhead remains a major source of error. Furthermore, our findings suggest that trapped-ion architectures with qubit shuttling are the most promising near-term platforms and that on noisy devices, a strategic and selective application of QEC is necessary to avoid introducing more errors than are corrected. This study provides crucial, actionable insights for both hardware designers and practitioners, guiding the development of fault-tolerant quantum systems.

CCS Concepts

• **Hardware** → **Quantum error correction and fault tolerance**; Quantum technologies; • **Computer systems organization** → **Quantum computing**; • **General and reference** → *Evaluation*; *Empirical studies*.

Keywords

quantum computing, quantum error correction, benchmarking

*This is a summary of our full paper [26].



This work is licensed under a Creative Commons Attribution 4.0 International License. *SIGMETRICS Abstracts '26, Ann Arbor, MI, USA*
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2570-8/2026/06
<https://doi.org/10.1145/3801489.3806905>

ACM Reference Format:

Aleksandra Świerkowska, Jannik Pflieger, Emmanouil Giortamis, and Pramod Bhatotia. 2026. ECCentric: An Empirical Analysis of Quantum Error Correction Codes. In *Abstracts of the 2026 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS Abstracts '26)*, June 08–12, 2026, Ann Arbor, MI, USA. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/3801489.3806905>

1 Introduction

Context and motivation. Quantum Error Correction (QEC) offers the only known path to fault tolerance, a prerequisite for building scalable quantum computers capable of solving classically intractable problems [1, 8, 10]. It encodes a single logical qubit into a redundant, entangled system of multiple noisy physical qubits [16, 18]. Once such an encoded circuit is transpiled into the hardware’s native gate set and adapted to its topology, it is executed on the device, where syndrome measurements are continuously performed, allowing the decoder to correct errors in real time [2].

Notably, a diverse family of QEC codes exists, each with unique resource overheads, operational assumptions, and performance characteristics. Their suitability heavily depends on the underlying physical qubit technology and its noise model [5, 8].

Research gap and challenges. Despite the variety of QEC codes, there’s a significant research gap: a lack of a systematic, end-to-end methodology for rigorously comparing them. This makes it challenging to accurately assess a code’s performance under the specific, realistic noise and software constraints of a particular QPU. A comprehensive practical evaluation needs to consider the entire execution pipeline, as each stage can influence the code’s effectiveness.

Unfortunately, current QEC evaluations are often ad hoc and limited, either focusing on a small number of codes with oversimplified noise models [7, 12] or analyzing various components in isolation [3, 23, 25]. As a result, no existing benchmarking framework provides an end-to-end, systematic evaluation. Current tools exhibit a distinct dichotomy: fast, specialized simulators, such as Stim [9], are hardware-agnostic and rely on simplistic error models, whereas general-purpose frameworks, such as Qiskit [13], lack the scalability needed to simulate large, fault-tolerant codes [8]. These limitations create a major blind spot in practical quantum computing.

To effectively address this research gap, we must confront three fundamental challenges. *Firstly, the diverse QEC landscape:* A vast number of codes exist, each with unique overheads, error thresholds, and decoding complexities, making it difficult to compare their effectiveness across different hardware and noise models [6, 14].

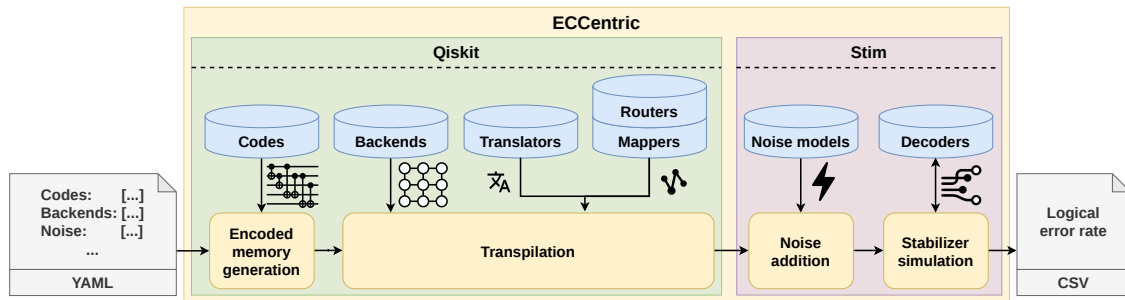


Figure 1: Architecture of the ECCentric framework. Our design of a fully modular and extensible setup, enabling systematic study of the different aspects of QEC realization.

Secondly, the vast experimental search space: A code’s performance is determined by a multi-dimensional interplay of QPU topology, physical noise, and compiler artifacts, rendering an exhaustive search computationally intractable [4, 23, 24]. *Finally, the lack of a suitable benchmarking framework:* Existing tools are either too slow for large-scale studies (e.g., Qiskit [13]) or too hardware-agnostic for realistic evaluation (e.g., Stim [9]). A viable solution must be general, supporting a wide spectrum of code families and QPU architectures; extensible, to easily integrate new decoders and evolving noise models; and modular, to enable the isolation and analysis of specific stages within the QEC pipeline, from encoding to decoding. **Research question and our approach.** These challenges lead to our central research question: *How can we systematically evaluate the suitability of quantum error correction (QEC) codes for practical applications on current and near-term quantum devices?*

To answer this question, we propose a two-staged approach. First, we impose structure on the diverse and rapidly evolving QEC landscape by creating a systematic taxonomy of QEC codes. From this taxonomy, we have identified five primary families of codes and selected a representative member from each to ensure our analysis is both fair and comprehensive.

Building on this foundational taxonomy, we introduce ECCentric, a new, end-to-end benchmarking framework designed for the systematic evaluation of QEC codes on current and near-term quantum devices. ECCentric, shown in Fig. 1, is general, capable of supporting a wide array of QEC codes, QPU technologies, and compiler back-ends. It is modular, allowing researchers to customize the evaluation pipeline and scope of their experiments. And it is extensible, making it easy to integrate new code, hardware features, and compilation strategies as they emerge.

Leveraging the full capabilities of the ECCentric framework, we structure our investigation around four key dimensions that influence QEC performance to systematically assess their impact: *QEC codes*, where we select six representative codes from all major families using our QEC code taxonomy; *QPU noise models*, where we include both ideal and realistic models derived from current (e.g., Google Willow [11]) and next-generation (e.g., Quantinuum Apollo [19] and IBM Flamingo [20]) quantum devices; *QPU topologies*, where we explore a range of layouts, from abstract to the complex architectures of existing hardware (e.g., Google Willow [11] and Inflection [21]), varying qubit count and connectivity to isolate the impact of topology; and finally, *quantum compilation*, where we investigate how different quantum compilers (e.g., Qiskit

[13] and TKET [22]) and their internal stages, such as mapping and routing heuristics (e.g., SABRE [15]), affect QEC performance.

Our contributions. Overall, to the best of our knowledge, this paper provides the first systematic benchmarking of multiple QEC codes, focusing on their practical application on real-world hardware, compilers, and noise models. Our contributions are summarized as follows:

- *Experimental QEC code analysis:* A systematic benchmarking of the major Quantum Error Correction (QEC) code families under realistic experimental conditions.
- *ECCentric framework:* The design of a modular benchmarking framework that can be readily extended to incorporate additional codes, devices, noise models, and decoders.
- *QEC code taxonomy:* The introduction of a structured taxonomy for QEC codes, clearly presenting the differences and shared characteristics among the primary families.
- *Concatenated Steane code extension:* The first extension of the $[[49,1,9]]$ concatenated Steane code [17] to $[[373,1,27]]$ concatenated Steane code.

2 Key findings

Our systematic evaluation reveals several critical insights and implications for both hardware and software development:

- **Fault-tolerance with trapped-ion is near:** Our analysis of projected hardware roadmaps shows that the planned error rates and qubit shuttling capabilities of trapped-ion devices should eliminate logical errors across most evaluated codes within the next five years, and therefore, prioritizing this platform’s development is the most direct path to achieving fault tolerance [19].
- **Distributed execution is still challenging:** Our results show that distributing codes across QPUs increases the logical error rate by 69.14% compared to single-QPU execution. However, while link fidelity is a factor, the principal issue is the low number of connections, and hardware development should prioritize increasing the cross-QPU links rather than solely focusing on their fidelity.
- **Prioritize connectivity over code distance:** Our experiments show that increasing code distance is often ineffective, increasing the logical error rate on average by 0.012 with only 30.95% of adjustments yielding meaningful improvement. In contrast, improving connectivity provides substantial gains: moving from a grid to a fully connected topology reduces the logical error

rate by 81.9%, while devices with qubit shuttling outperform those without by 45.48%. Thus, hardware manufacturers should prioritize improving qubit connectivity.

- **QEC-aware compilation is essential:** The compilation process introduces significant overhead that can undermine the benefits of error correction. We find that mapping and routing add an average of 134.095% more two-qubit gates, while optimized translation still adds 3.166 extra gates per original gate. Mitigating this requires QEC-aware compilers that optimize at the logical-qubit level, for example, by prioritizing efficient routing to busy ancilla qubits and canceling redundant gate sequences across repeated QEC cycles.
- **Heterogeneity is not a primary concern:** We find the variability in individual qubit quality or in readout correctness has a negligible impact on performance, changing the logical error rate by 0.02 on average. This suggests that complex, variance-aware compilation strategies might be an unnecessary burden. This frees compiler developers to de-prioritize these intricate techniques and instead pursue more straightforward and effective designs.
- **QEC is not always helpful:** On noisy, near-term devices, the indiscriminate application of QEC can be counterproductive. Our results show that at a physical two-qubit error rate of 0.002, most of our evaluated codes fail, and at 0.004, none remain effective, introducing more errors than they correct. This necessitates a selective approach where QEC is only used for operations or qubits where its benefits outweigh its overhead.

Full version. The full version of this work presents the details of each evaluation with background information and discussion [26].

Acknowledgments

We thank Dr. Joschka Roffe for his insights on the taxonomy of QEC codes and Peter Wegmann for his assistance with the implementation. Funded by the Bavarian State Ministry of Science and the Arts as part of the Munich Quantum Valley (MQV), grant number 6090181.

References

- [1] Dorit Aharonov and Michael Ben-Or. 1999. Fault-Tolerant Quantum Computation With Constant Error Rate. arXiv:quant-ph/9906129 [quant-ph] <https://arxiv.org/abs/quant-ph/9906129>
- [2] F Battistel, C Chamberland, K Johar, R W J Overwater, F Sebastiano, L Skoric, Y Ueno, and M Usman. 2023. Real-time decoding for fault-tolerant quantum computing: progress, challenges and outlook. *Nano Futures* 7, 3 (Aug. 2023), 032003. <https://doi.org/10.1088/2399-1984/aceba6>
- [3] César Benito, Esperanza López, Borja Peropadre, and Alejandro Bermudez. 2025. Comparative study of quantum error correction strategies for the heavy-hexagonal lattice. *Quantum* 9 (Feb. 2025), 1623. <https://doi.org/10.22331/q-2025-02-06-1623>
- [4] César Benito, Esperanza López, Borja Peropadre, and Alejandro Bermudez. 2025. Comparative study of quantum error correction strategies for the heavy-hexagonal lattice. *Quantum* 9 (Feb. 2025), 1623. <https://doi.org/10.22331/q-2025-02-06-1623>
- [5] Nikolas P. Breuckmann and Jens Niklas Eberhardt. 2021. Quantum Low-Density Parity-Check Codes. *PRX Quantum* 2 (Oct 2021), 040101. Issue 4. <https://doi.org/10.1103/PRXQuantum.2.040101>
- [6] Avimta Chatterjee and Swaroop Ghosh. 2024. Magic Mirror on the Wall, How to Benchmark Quantum Error Correction Codes, Overall?. In *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE Computer Society, Los Alamitos, CA, USA, 356–367. <https://doi.org/10.1109/QCE60285.2024.00050>
- [7] Regina Finsterhoelzl and Guido Burkard. 2022. Benchmarking quantum error-correcting codes on quasi-linear and central-spin processors. *Quantum Science and Technology* 8, 1 (nov 2022), 015013. <https://doi.org/10.1088/2058-9565/aca21f>
- [8] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. 2012. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A* 86 (Sep 2012), 032324. Issue 3. <https://doi.org/10.1103/PhysRevA.86.032324>
- [9] Craig Gidney. 2021. Stim: a fast stabilizer circuit simulator. *Quantum* 5 (July 2021), 497. <https://doi.org/10.22331/q-2021-07-06-497>
- [10] Craig Gidney and Martin Ekerå. 2021. How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum* 5 (April 2021), 433. <https://doi.org/10.22331/q-2021-04-15-433>
- [11] Google Quantum AI. 2024. *Willow Spec Sheet*. Product Data Sheet. Google Quantum AI.
- [12] Eric Huang, Andrew C. Doherty, and Steven Flammia. 2019. Performance of quantum error correction with coherent errors. *Physical Review A* 99, 2 (Feb. 2019). <https://doi.org/10.1103/physreva.99.022313>
- [13] Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. 2024. Quantum computing with Qiskit. arXiv:2405.08810 [quant-ph] <https://arxiv.org/abs/2405.08810>
- [14] Akshaya Jayashankar and Prabha Mandayam. 2022. Quantum Error Correction: Noise-Adapted Techniques and Applications. *Journal of the Indian Institute of Science* 103 (09 2022), 1–16. <https://doi.org/10.1007/s41745-022-00332-x>
- [15] Gushu Li, Yufei Ding, and Yuan Xie. 2019. Tackling the Qubit Mapping Problem for NISQ-Era Quantum Devices. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems (Providence, RI, USA) (ASPLOS '19)*. Association for Computing Machinery, New York, NY, USA, 1001–1014. <https://doi.org/10.1145/3297858.3304023>
- [16] Michael A. Nielsen and Isaac L. Chuang. 2010. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press.
- [17] Balint Pato, Theerapat Tansuwannont, and Kenneth R. Brown. 2024. Concatenated Steane code with single-flag syndrome checks. *Physical Review A* 110, 3 (Sept. 2024). <https://doi.org/10.1103/physreva.110.032411>
- [18] John Preskill. 2018. Quantum Computing in the NISQ era and beyond. *Quantum* 2 (Aug. 2018), 79. <https://doi.org/10.22331/q-2018-08-06-79>
- [19] Quantinuum. 2024. *Quantinuum unveils accelerated roadmap to achieve universal, fully fault-tolerant quantum computing by 2030*. <https://www.quantinuum.com/press-releases/quantinuum-unveils-accelerated-roadmap-to-achieve-universal-fault-tolerant-quantum-computing-by-2030>
- [20] IBM Quantum. 2024. *Development & Innovation Roadmap*. https://www.ibm.com/quantum/assets/IBM_Quantum_Development_&_Innovation_Roadmap_Explainer_2024-Update.pdf
- [21] A. G. Radnaev, W. C. Chung, D. C. Cole, D. Mason, T. G. Ballance, M. J. Bedalov, D. A. Belknap, M. R. Berman, M. Blakely, I. L. Bloomfield, P. D. Buttler, C. Campbell, A. Chopinaud, E. Copenhaver, M. K. Dawes, S. Y. Eubanks, A. J. Friss, D. M. Garcia, J. Gilbert, M. Gillette, P. Goiporia, P. Gokhale, J. Goldwin, D. Goodwin, T. M. Graham, C. J. Guttorfsson, G. T. Hickman, L. Hurtle, M. Iliev, E. B. Jones, R. A. Jones, K. W. Kuper, T. B. Lewis, M. T. Lichtman, F. Majdeteimouri, J. J. Mason, J. K. McMaster, J. A. Miles, P. T. Mitchell, J. D. Murphree, N. A. Neff-Mallon, T. Oh, V. Omole, C. Parlo Simon, N. Pederson, M. A. Perlin, A. Reiter, R. Rines, P. Romlow, A. M. Scott, D. Stiefvater, J. R. Tanner, A. K. Tucker, I. V. Vinogradov, M. L. Warter, M. Yeo, M. Saffman, and T. W. Noel. 2025. A universal neutral-atom quantum computer with individual optical addressing and non-destructive readout. arXiv:2408.08288 [quant-ph] <https://arxiv.org/abs/2408.08288>
- [22] Seyon Sivarajah, Silas Dilkes, Alexander Cowtan, Will Simmons, Alec Edgington, and Ross Duncan. 2020. tket: a retargetable compiler for NISQ devices. *Quantum Science and Technology* 6, 1 (Nov. 2020), 014003. <https://doi.org/10.1088/2058-9565/ab8e92>
- [23] Yu Tomita and Krysta M. Svore. 2014. Low-distance surface codes under realistic quantum noise. *Physical Review A* 90, 6 (Dec. 2014). <https://doi.org/10.1103/physreva.90.062320>
- [24] Xiaosi Xu, Simon C. Benjamin, and Xiao Yuan. 2021. Variational Circuit Compiler for Quantum Error Correction. *Phys. Rev. Appl.* 15 (Mar 2021), 034068. Issue 3. <https://doi.org/10.1103/PhysRevApplied.15.034068>
- [25] Yue Zhao. 2024. Benchmarking Machine Learning Models for Quantum Error Correction. arXiv:2311.11167 [quant-ph] <https://arxiv.org/abs/2311.11167>
- [26] Aleksandra Świerkowska, Janik Pflieger, Emmanouil Giortamis, and Pramod Bhatotia. 2026. ECCentric: An Empirical Analysis of Quantum Error Correction Codes. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 10, 2, Article 37 (June 2026). <https://doi.org/10.1145/3805635>

Received January 2026; revised March 2026; accepted April 2026