Fault Tolerance by Construction

Benjamin Rodatz, Boldizsár Poór, Aleks Kissinger
University of Oxford, Oxford, UK

A key challenge in fault-tolerant quantum computing is synthesising and optimising circuits in a noisy environment, as traditional techniques often fail to account for the effect of noise on circuits. In this work, we propose a framework for designing fault-tolerant quantum circuits that are *correct by construction*. The framework starts with idealised specifications of fault-tolerant gadgets and refines them using provably sound basic transformations.

To reason about manipulating circuits while preserving their error correction properties, we define *fault equivalence*; two circuits are considered fault-equivalent if all undetectable faults on one circuit have a corresponding fault on the other. This guarantees that the effect of undetectable faults on both circuits is the same. We argue that fault equivalence is a concept that is already implicitly present in the literature. Many problems, such as state preparation and syndrome extraction, can be naturally expressed as finding an implementable circuit that is fault-equivalent to an idealised specification.

To utilise fault equivalence in a computationally tractable manner, we adapt the ZX calculus, a diagrammatic language for quantum computing. We restrict its rewrite system to not only preserve the underlying linear map but also fault equivalence, i.e. the circuit's behaviour under noise. Enabled by our framework, we verify, optimise and synthesise new and efficient circuits for syndrome extraction and cat state preparation. We anticipate that fault equivalence can capture and unify different approaches in fault-tolerant quantum computing, paving the way for an end-to-end circuit compilation framework.

The full technical manuscript of this work is available at https://arxiv.org/abs/2506.17181.

1 Introduction

Building large-scale quantum computers requires overcoming the detrimental effects of noise. Fault-tolerant quantum computing (FTQC) addresses this challenge by encoding computations with redundancy, allowing errors to be detected and corrected. Error-correcting codes provide the means to protect data, while code-specific fault-tolerant 'gadgets' are used to implement the required computations. However, synthesising, optimising, and verifying quantum circuits for such gadgets remains difficult, as faults can propagate non-trivially throughout the circuits. Most existing quantum circuit synthesis and optimisation techniques are insufficient for this task because they focus solely on preserving the underlying linear map. Yet circuits that implement the same unitary can behave very differently under noise. As a result, circuits for fault-tolerant gadgets are often designed through trial and error and verified either analytically or with classical simulation tools such as Stim [9]. In this paper, we propose a new methodology for compiling fault-tolerant computations that are *correct by construction*.

2 The Framework

To synthesise circuits that are correct by construction, we start with an idealised computation that has good properties with respect to fault tolerance, but is not implementable, i.e. it is not expressed in terms of the basic, potentially noisy operations available on a quantum device. We then transform that computation step-wise into something expressed in terms of these basic operations while preserving the behaviour of the computation under faults. Analogous ideas of *refining* an idealised *specification* into an implementable program through provably sound transformations have been central to formal software verification since the 1980s [1, 3].

To adapt these ideas to FTQC, we introduce and formalise the concept of *fault equivalence*. Following Bacon et al. [2], Delfosse and Paetznick [7], and Gottesman [10], we consider the propagation of Pauli faults on Clifford circuits, defined as Pauli actions occurring at time steps between gates. A noise model $\mathcal{F}$ on some circuit is defined by a set of atomic faults which specify the independent error mechanisms, e.g. a single memory, gate, or measurement errors. The weight of a fault is defined as the minimum number of atomic fault events required to create that fault. Using this, we can define when we consider two circuits under their respective noise models to be equal:

Definition 2.1 (Fault equivalence)**.** The circuit C_1 under noise model $\mathcal{F}_1$ is *fault-equivalent* to C_2 under $\mathcal{F}_2$ iff for all faults $F_1 \in \langle \mathcal{F}_1 \rangle$, either: (a) F_1 is detectable, or (b) there exists a fault $F_2 \in \langle \mathcal{F}_2 \rangle$ on C_2 such that $wt(F_2) \leq wt(F_1)$ and $C_1^{F_1} = C_2^{F_2}$. The condition must similarly hold for all faults $F_2 \in \langle \mathcal{F}_2 \rangle$. We write $C_1 \mathrel{\hat{=}} C_2$.

Two circuits are fault-equivalent if all undetectable faults on one circuit have a corresponding fault on the other circuit of equal or lesser weight. Practically, this ensures that circuit transformations do not introduce new, undetectable faults that propagate badly (e.g. hook errors). This concept unifies and generalises many related conditions in the literature, providing a framework to reason about specific fault-tolerant gadgets as well as the fault-tolerant computation as a whole.

For example, in the literature, fault-tolerant state preparation is defined as a circuit such that if a fault of weight w occurs, it is either detectable or creates at most w qubit flips. This is formally equivalent to saying a state preparation circuit C is fault-tolerant if and only if it satisfies the fault-equivalence relationship shown in Figure 1 (a). Here, in the specification on

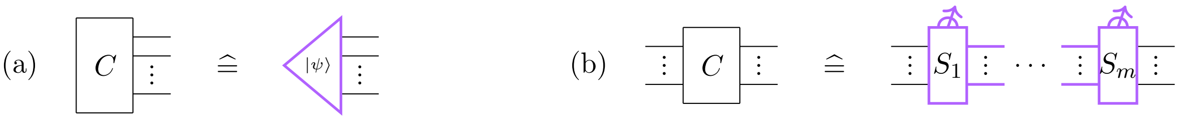


Figure 1: Specifications for (a) fault-tolerant state preparation for some state ψ (b) fault-tolerant syndrome extraction for some code with stabilisers generated by $S_1, \ldots S_m$.

the right, the state itself is idealised as fault-free (shown in purple), where only data qubit flips are allowed as faults. Thus, fault equivalence requires that any fault of weight w on C is either detectable or corresponds to a fault of at most the same weight on the specification. This is formally equivalent to the condition usually found in the literature. Similarly, other fault-tolerance conditions can be formalised as circuits that are fault-equivalent to idealised specifications. For example, a circuit is a fault-tolerant syndrome extraction circuit iff it is fault-equivalent to perfectly measuring a generating set of the stabilisers without any intermediate faults (Figure 1 (b)).

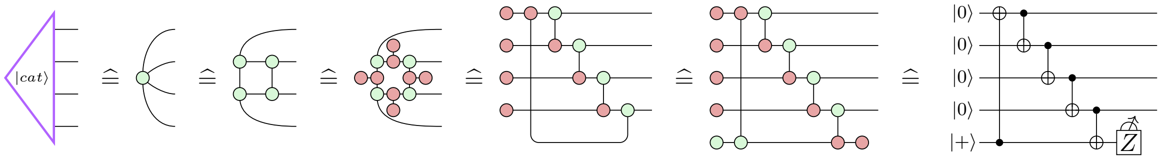
More generally, many problems in fault-tolerant circuit synthesis can be expressed as: for this specification of the circuit I would like to perform and how it should behave under noise, find a fault-equivalent, implementable circuit.

3 Construction of Novel Fault-Tolerant Gadgets

After establishing fault equivalence for circuits, we adapt the ZX calculus [5] to the noisy setting. The ZX calculus represents quantum computations as undirected graphs called ZX diagrams, which can be transformed using a handful of simple rewrite rules. It can be used to synthesise and optimise quantum circuits [6, 8]. However, standard ZX rules only preserve the ideal, fault-free linear map. To reason about faulty circuits, we therefore only use fault-equivalent rewrites, which preserve both the intended computation and the possible effects of faults.

Building on these, we provide a constructive framework for fault-equivalent circuit synthesis in three steps: (1) translate an idealised specification into a fault-equivalent ZX diagram, (2) fault-equivalently manipulate the ZX diagram into a circuit-like shape, and (3) extract a new circuit. As fault-tolerant gadgets are correct iff they are fault-equivalent to the corresponding idealised specifications, we have synthesised circuits that are correct by construction. This guarantees fault tolerance even without obvious structural indicators, such as transversal gates.

Coming back to state preparation, we can recover a well-known circuit for fault-tolerant cat state preparation:



giving us a quantum circuit with one measurement to catch faults that would otherwise propagate badly, a so-called flag measurement. While other flag-based approaches typically enumerate all fault combinations and construct corresponding flags [4, 13], in this framework, the necessary flags are a consequence of the construction. By fault equivalence, we are guaranteed that all faults must either be detectable or create at most as many errors on data qubits as their weight, without having to check all fault combinations individually. Building on this construction, we develop novel protocols to synthesise $2n$-legged fault-tolerant cat states from two n-legged ones.

Next, we use fault-tolerant rewrites to verify Shor-style [15] and Steane-style [16] syndrome extraction. Since a syndrome extraction circuit is fault-tolerant iff it is fault-equivalent to the specification, we use fault-equivalent rewrites to further optimise the circuits proposed by Shor and Steane (Figure 2). This yields novel syndrome extraction circuits with fewer gates and qubits than their original counterpart. Monte-carlo simulations show our optimised Steane-like syndrome extraction circuit for the Steane code achieves lower error rates while using fewer resources and no repeat-until-success state preparation.

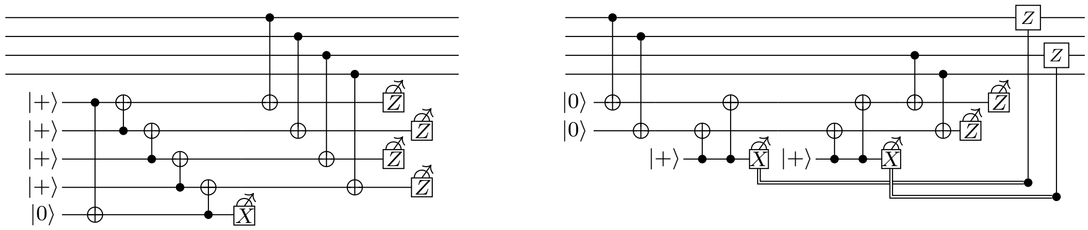


Figure 2: Comparison of standard (left) and optimised (right) Shor-style syndrome extraction circuits for a weight-4 stabiliser measurement.

4 Conclusion

This paper proposes to study the relationship between quantum circuits and their behaviour under noise, rather than treating circuits in isolation. Many aspects of fault-tolerant quantum computing can be understood as different sides of the same principle — fault-tolerant circuit manipulation — offering new methods for circuit synthesis, verification, and optimisation. This offers a practical, computationally tractable toolkit to reason about fault-tolerant components, leading to more resource-efficient quantum circuits. By providing rigorous methods for designing circuits that are both correct and robust against noise, this framework contributes directly to the critical challenge of building scalable, reliable quantum computers. In this paper, we verify the framework by applying it to cat state preparation and syndrome extraction, producing circuits that use fewer resources while performing better in simulation. It has since successfully been applied in other settings [11, 12, 14].

References

[1] Ralph-Johan Back and Joakim Von Wright. *Refinement Calculus: A Systematic Introduction.* Edited by Fred B. Schneider and David Gries. Texts in Computer Science. Springer, 1998. 519 pages. ISBN: 978-0-387-98417-9.

[2] Dave Bacon, Steven T. Flammia, Aram W. Harrow, and Jonathan Shi. "Sparse Quantum Codes From Quantum Circuits". In: *IEEE Transactions on Information Theory* 63.4 (Apr. 2017), pages 2464–2479. ISSN: 1557-9654. DOI: 10.1109/TIT.2017.2663199. arXiv: 1411.3334.

[3] Dines Bjørner and Cliff B. Jones, editors. *Formal Specification and Software Development.* Englewood Cliffs, NJ: Prentice Hall International, 1982. 501 pages. ISBN: 0-13-329003-4.

[4] Christopher Chamberland and Michael E. Beverland. "Flag Fault-Tolerant Error Correction with Arbitrary Distance Codes". In: *Quantum* 2 (Feb. 8, 2018), page 53. DOI: 10.22331/q-2018-02-08-53.

[5] Bob Coecke and Ross Duncan. "Interacting Quantum Observables". In: *Automata, Languages and Programming.* Edited by Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2008, pages 298–310. ISBN: 978-3-540-70583-3. DOI: 10.1007/978-3-540-70583-3_25. URL: http://personal.strath.ac.uk/ross.duncan/papers/iqo-icalp.pdf.

[6] Alexander Cowtan, Silas Dilkes, Ross Duncan, Will Simmons, and Seyon Sivarajah. "Phase Gadget Synthesis for Shallow Circuits". In: *Proceedings 16th International Conference on Quantum Physics and Logic, Chapman University, Orange, CA, USA., 10-14 June 2019.* Edited by Bob Coecke and Matthew Leifer. Volume 318. Electronic Proceedings in Theoretical Computer Science. Open Publishing Association, 2020, pages 213–228. DOI: 10.4204/EPTCS.318.13.

[7] Nicolas Delfosse and Adam Paetznick. *Spacetime Codes of Clifford Circuits.* May 26, 2023. arXiv: 2304.05943.

[8] Ross Duncan, Aleks Kissinger, Simon Perdrix, and John van de Wetering. "Graph-Theoretic Simplification of Quantum Circuits with the ZX-calculus". In: *Quantum* 4 (June 2020), page 279. ISSN: 2521-327X. DOI: 10.22331/q-2020-06-04-279.

[9] Craig Gidney. "Stim: A Fast Stabilizer Circuit Simulator". In: *Quantum* 5 (July 2021), page 497. ISSN: 2521-327X. DOI: 10.22331/q-2021-07-06-497.

[10] Daniel Gottesman. *Opportunities and Challenges in Fault-Tolerant Quantum Computation.* Oct. 28, 2022. arXiv: 2210.15844.

[11] Andrey Boris Khesin, Sarah Meng Li, Boldizsár Poór, Benjamin Rodatz, John van de Wetering, and Richie Yeung. *SpiderCat: Optimal Fault-Tolerant Cat State Preparation.* 2026. arXiv: 2603.05391 [quant-ph]. URL: https://arxiv.org/abs/2603.05391.

[12] Andrey Boris Khesin and Jonathan Z. Lu. *Mirror codes: High-threshold quantum LDPC codes beyond the CSS regime.* 2026. arXiv: 2603.05496 [quant-ph]. URL: https://arxiv.org/abs/2603.05496.

[13] Tom Peham, Ludwig Schmid, Lucas Berent, Markus Müller, and Robert Wille. "Automated Synthesis of Fault-Tolerant State Preparation Circuits for Quantum Error Correction Codes". In: *PRX Quantum* 6.2 (May 14, 2025), page 020330. ISSN: 2691-3399. DOI: 10.1103/PRXQuantum.6.020330. arXiv: 2408.11894.

[14] Boldizsár Poór, Benjamin Rodatz, and Aleks Kissinger. *Ultra Low Overhead Syndrome Extraction for the Steane code.* 2025. arXiv: 2511.13700 [quant-ph].

[15] P.W. Shor. "Fault-Tolerant Quantum Computation". In: *Proceedings of 37th Conference on Foundations of Computer Science.* 37th Conference on Foundations of Computer Science. Oct. 1996, pages 56–65. DOI: 10.1109/SFCS.1996.548464.

[16] A. M. Steane. "Active Stabilization, Quantum Computation, and Quantum State Synthesis". In: *Physical Review Letters* 78.11 (Mar. 17, 1997), pages 2252–2255. DOI: 10.1103/PhysRevLett.78.2252.