HIGH SCHOOL EDITION
Journal of Student Research

Volume 12 Issue 3 (2023)

Benchmarking and Analysis of Quantum Error Correction Codes

Rohan Timmaraju

Sydney Grammar School

ABSTRACT

Quantum Computing is a computing framework that takes advantage of unique quantum mechanical properties (such as superposition and entanglement) to perform calculations and implement algorithms that could offer exponential speed-ups over classical computing. However, in physical implementations of such quantum computers, qubits – the fundamental components of these systems – can accumulate errors that must be accounted for. In order to mitigate these errors, various quantum error correction (QEC) codes have been developed, including the repetition code and surface codes. In this experiment I implement and evaluate three types of QEC codes on the Qiskit simulator to compare their efficacy and applicability in correcting for different kinds of errors. I hypothesize that surface codes, with their more effective design and range of correction methods, should perform the best with much lower error thresholds and resultant logical error rates. The results support the hypothesis and suggest that surface codes are a viable method of implementing scalable error correction in quantum computers.

Introduction

Quantum Computing, in contrast to classical or traditional computing, functions by manipulating qubits. Qubits are the basic storage units of quantum information - they are an analog to a classical bit, but are fundamentally different. While bits in normal computers can only hold a ‘0’ or ‘1’ state, qubits can hold a spectrum of different states, in a superposition of the ‘0’ and ‘1’ state. These states can be represented as any point on a ‘sphere’(which is called the Bloch Sphere), as shown in Figure 1.

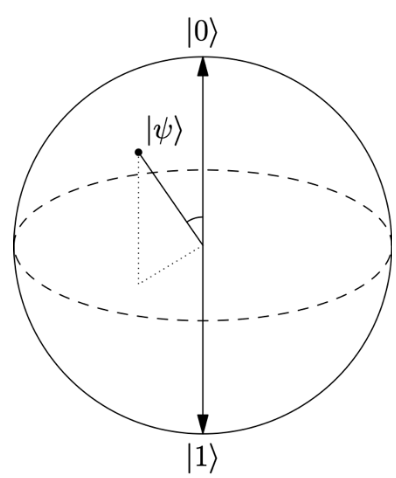

Figure 1. The Bloch sphere

Computation with these qubits uses their unique quantum properties - superposition and entanglement - to allow combinations of these qubits to perform calculations and execute algorithms that could potentially be much faster than classical algorithms (Shor, 1995). Superposition allows qubits to hold states that are a combination of the '0' and '1' state, allowing qubit systems to represent states that would take exponential amounts of space in classical computers. Entanglement is the property that if two qubits share a quantum state, when there is a change to one of them, this change is reflected immediately in the state of the other qubit, regardless of the distance separating them.

Problems such as integer factorisation and unstructured search for large datasets are computationally difficult on classical computers. However algorithms for these problems have been developed which could provide exponential speedups to these problems on sufficiently large quantum computers - such as Shor's algorithm for factorisation (Shor, 1995) and Grover's algorithm for unstructured search (Grover, 1996). To explore the feasibility of achieving such computations and speedups, there is thus much active research into physical implementations of quantum computers, quantum algorithms and error correction schemes.

There are a variety of physical implementations of qubits being pursued. However, the algorithms and analysis developed for quantum computing are largely independent of the physical implementation. The challenge with real applications is performing high-fidelity operations on qubits at speed, as well as scaling this to multiple qubits. Due to impurities or small fluctuations in the environment (depending on the physical implementation), the state of a qubit can deviate from its intended position, and these errors can increase as time proceeds. These fluctuations in the state are called noise and decoherence. Thus, for more complex circuits and algorithms, a form of error-correction is needed to protect or correct against such errors (Chatterjee, 2023).

Error Correction

There is a well-developed theory of classical error correction on traditional bits (Chatterjee, 2023), however there are some important distinctions between classical bits and qubits that render these codes impossible for quantum systems. This includes the no-cloning theorem, which states that we cannot arbitrarily copy the quantum state of a qubit to another qubit. This prevents us from simply copying the state of a qubit to correct errors. There is also the destructive and irreversible nature of qubit measurement, where the quantum state of a qubit is lost after we measure it. Any measurement collapses the state of a qubit into a binary result, which means if we measure any of the qubits in our logical circuit, we will destroy its state. Thus, any measurements used must not collapse the state of logical qubits in any quantum error code. This can be done through entangling the state of the qubit with other qubits and measuring those as a "stabilizer". In contrast to classical error correction, multiple types of errors can occur, such as bit-flips and phase-flips (X and Z errors). Quantum error correction schemes should thus also be able to correct for these multiple types of errors.

The first development that overcame these challenges was the quantum correction code by Peter Shor in 1995 (Chatterjee, 2023), showing how the state of a quantum system can be extended across multiple qubits through entanglement. Extensions to this technique have since allowed the development of further codes that can correct against arbitrary errors, provided that physical qubits and operations can be implemented within a certain accuracy threshold.

Experiment

In this experiment, I benchmark the Repetition code and the XXZZ and XZZX Surface codes, and evaluate them on the IBM Qiskit simulator to compare their effectiveness in metrics such as their error probability threshold and the size (number of qubits), as well as comparing their error thresholds. The error threshold is the error rate of a physical qubit beyond which increasing the size of a quantum error correction code does not improve the accuracy (Fowler et al., 2012).

I hypothesize that surface codes with their more effective design and range of correction methods, would perform the best with much lower error thresholds and resultant logical error rates. I also hypothesize the repetition code with a simple majority-vote decoding system, would have a threshold of around 0.5 as a physical error rate (break-even point).

Methods

The benchmarks were implemented using the Qiskit quantum computing simulator in Python, as it provides an accessible package to implement quantum algorithms and run them on classical computers. It is not feasible to run many of these quantum error correction schemes on current physical quantum computers as they are not yet large enough. This develops on the work of the 'qtcodes' package. In the experiment, I tested implementations of the Repetition Code and the Surface Code by applying random Pauli (X) gates to a logical qubit encoded in each error correction code. These logical qubits were initialized to a '0' base state, then put through an error channel where the Pauli error is applied with a given probability that is denoted by the physical error rate. This models a random error on a physical qubit (which varied from 0.05 to 0.2 for the surface codes, and from 0 to 1 for the repetition code). After this, the qubit state was measured, and the corresponding decoding scheme was employed (in this case, a minimum-weight matching graph decoder) to determine the corrected output state. This was repeated 2048 times for each type and size of code, from which the logical error rate was calculated as the proportion of times the resultant logical value matched the initial qubit value. This was done for varying code sizes (from a size of 3 up to 9 for the Surface codes, and 3 to 11 for the repetition codes).

Results

From this process, I obtained the following results of logical error probabilities from physical error rates for each of the code types.

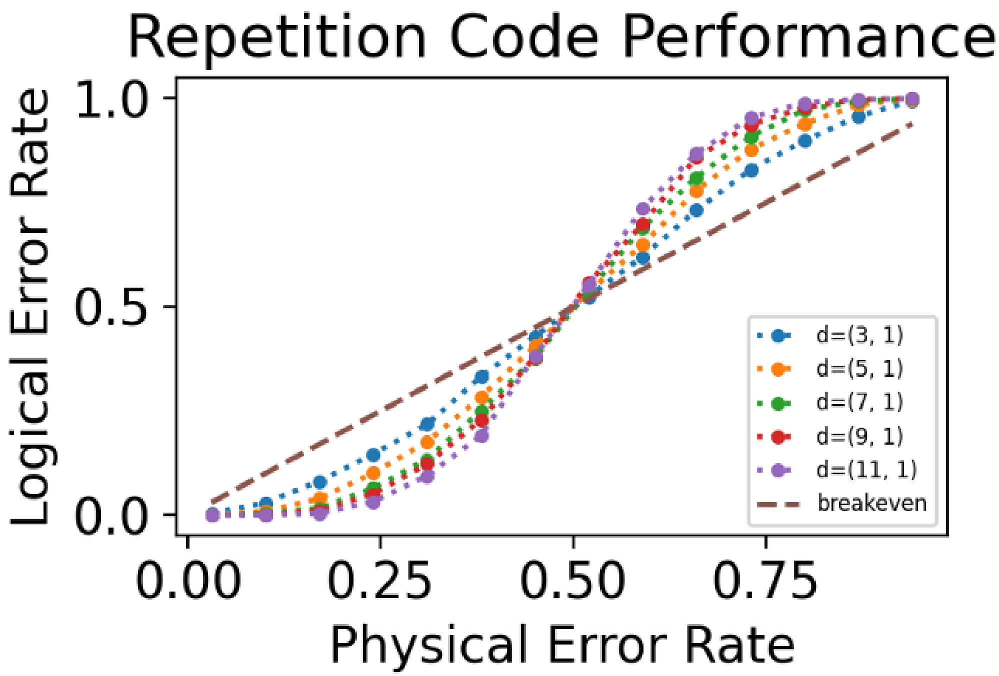

Figure 2. Logical vs Physical error rates for a range of Repetition code sizes.

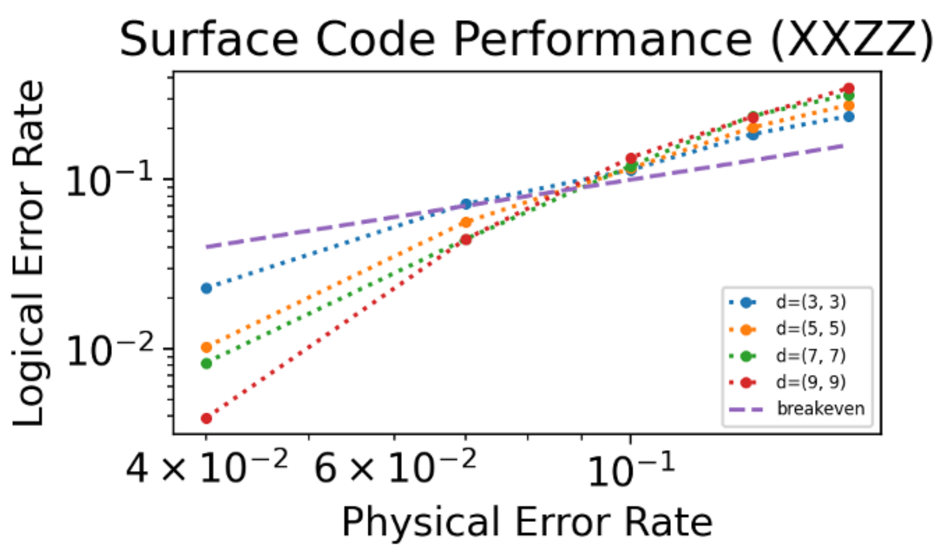

Figure 3. Logical vs Physical error rates for a range of XXZZ Surface code sizes.

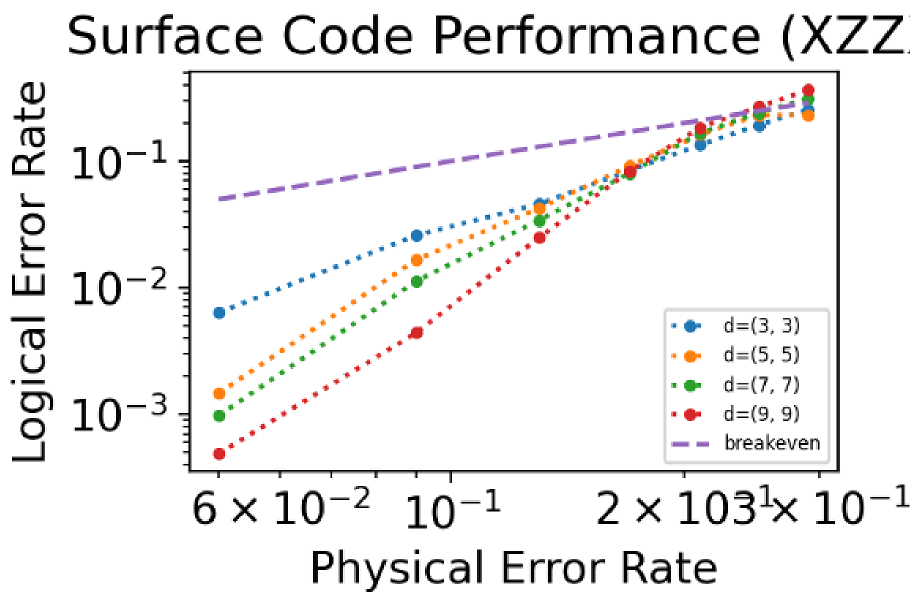

Figure 4. Logical vs Physical error rates for a range of XZZX Surface code sizes.

The relation between the logical error rate and the physical error rate depends strongly on the code size d (Fowler et al., 2012), which is called the distance of the code: d is the minimum amount of bit-flips or phase-flips needed to define one qubit operator in one dimension. For small physical error probabilities, the logical probability is also small, and decreases with increasing d (code size) for each type of code. For large physical error rates, the logical probability is larger, and increases with increasing code size.

I found that the Repetition Code, as hypothesized, had the largest error threshold of 0.5 (Figure 2), while the Surface Codes had significantly lower thresholds of 0.09 for the XXZZ code (Figure 3), and 0.15 for the XZZX code (Figure 4). The simulations also scale with p according to the power law Logical Probability ≈ $p^{(d+1)/2}$, supporting the empirical power law found in (Fowler et al., 2012), and approximated using Logical Probability ≈ $0.02\left(\frac{p}{p_{th}}\right)^{\frac{d+1}{2}}$

Discussion

By the quantum fault-tolerance theorem, a quantum computer with physical error rates below a certain threshold can suppress logical error rates to arbitrarily low levels, and perform general-purpose calculations accurately. Current estimates put the fault tolerance threshold for surface codes at around 0.6-1% (Xue et al., 2022), though estimates vary. The efficacy of the surface codes at achieving logical error rates below this level (as demonstrated in Figures 2 and 3), in particular the XZZX code, and their lower error thresholds, suggest they are more useful and effective as correction schemes on practical quantum computers.

Surface codes make use of a lattice or "surface" of qubits connected in a systematic way, composed of 'vertices' and 'plaquettes' of X and Z operators. By creating an encoding system where the state (when no error occurs) is always stabilized, we can detect when the state has changed from the basis state and then correct errors by applying a sequence of vertex and plaquette operators in an order determined by the minimum-weight perfect-matching decoding algorithm. This ability to perform corrections by stabilizing on nearby qubits also provides better connectivity for the circuits, as physical qubits may not have to be entangled with qubits as far away as with repetition codes. One drawback to this scheme is that it requires many more physical qubits to produce one logical qubit than the repetition code. The simpler repetition code, however, in practice performs worse than the surface codes, and can only correct for one type of error.

Limitations

There are several limitations in our benchmark and model of these quantum error correction schemes - the largest being that this is a simulation of real errors, as practical implementations of these codes on physical hardware is still being developed (O'Gorman et al., 2016), and is not widely available. I am also using a simplified model of a random application of Pauli gates for errors, but in practice this approximation yields accurate estimates for the performance of quantum error correction codes (Bravyi et al., 2018), (Gutierrez & Brown, 2015).

Conclusion

Ultimately, I found that the Repetition code had the largest error threshold and can correct for the smallest range of errors. The surface codes however have orders of magnitude lower error threshold probabilities and can correct for much greater ranges of errors on systems. This makes surface codes more useful and applicable to physical situations in reducing error. I am interested in pursuing further extensions to this work by testing surface codes on physical implementations, to determine if they are indeed a feasible way of building a fault-tolerant quantum computer.

Acknowledgements

I would like to thank my mentor for her guidance and help in this process, as well as my advisor for his support in exploring this area.

References

Bravyi, S., Englbrecht, M., König, R., & Peard, N. (2018). Correcting coherent errors with surface codes. Npj Quantum Information, 4(1). https://doi.org/10.1038/s41534-018-0106-y

Chatterjee, A. (2023, April 18). Quantum Error Correction For Dummies. arXiv.org. https://arxiv.org/abs/2304.08678

Fowler, A. G., Mariantoni, M., Martinis, J. M., & Cleland, A. (2012). Surface codes: Towards practical large-scale quantum computation. Physical Review A, 86(3). https://doi.org/10.1103/physreva.86.032324

Grover, L. K. (1996, May 29). A fast quantum mechanical algorithm for database search. arXiv.org. https://arxiv.org/abs/quant-ph/9605043

Gutierrez, M., & Brown, K. H. (2015). Comparison of a quantum error-correction threshold for exact and approximate errors. Physical Review A, 91(2). https://doi.org/10.1103/physreva.91.022335

O'Gorman, J., Nickerson, N. H., Ross, P., Morton, J. M., & Benjamin, S. C. (2016). A silicon-based surface code quantum computer. Npj Quantum Information, 2(1). https://doi.org/10.1038/npjqi.2015.19

Shor, P. W. (1995). Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM Journal on Computing, 26(5), 1484–1509. https://doi.org/10.1137/s0097539795293172

Xue, X., Russ, M., Samkharadze, N., Undseth, B., Sammak, A., Scappucci, G., & Vandersypen, L. M. K. (2022). Quantum logic with spin qubits crossing the surface code threshold. Nature, 601(7893), 343–347. https://doi.org/10.1038/s41586-021-04273-w