Understanding the Impact of Error in Quantum Computers

Danesh Morales Hashemi

Supervisor: Joseph Emerson

PHYS 437A Presentation

October 22, 2024

Overview

1. Introduction
2. Background
3. Methods
4. Results
5. Next Steps

Introduction

Quantum computers are not perfect.

There are different ways to deal with noise in quantum systems (error suppression, error correction and error diagnostics).

For this presentation:

We will give a quick introduction to Randomized Compiling (RC).

We will illustrate the theoretical concept of Circuit Benchmarking, with numerical examples.

Overview

Background

Easy Gate Set: $\langle \mathbf{P}_2, S \rangle = \{I, X, Y, Z, S, S^\dagger, SX, S^\dagger X\}$

Hard Gate Set: $\{H, T, CX, CZ\}$

} Universal Set of Quantum Gates

Cycle: Set of gates that happen in parallel to a disjoint set of systems.

Over-rotation Error: $\mathcal{E}_U = U^{1+\epsilon}$

Total Variation Distance: $d_{\mathrm{TV}}(\mathcal{P}, \mathcal{Q}) = \frac{1}{2} \sum_{x \in X} |\mathcal{P}(x) - \mathcal{Q}(x)|$

Background

Average Gate Fidelity: $\mathcal{F}(\mathcal{E}, U) = \int d\psi \langle\psi|U^\dagger \mathcal{E}(|\psi\rangle\langle\psi|)U|\psi\rangle$

Average Gate Infidelity

$r = 1 - \mathcal{F}$

Process Fidelity: $F_P(\mathcal{E}, U) = \frac{\mathcal{F}(\mathcal{E}, U)(d+1) - 1}{d}$ $\quad d = 2^n$

Process Infidelity

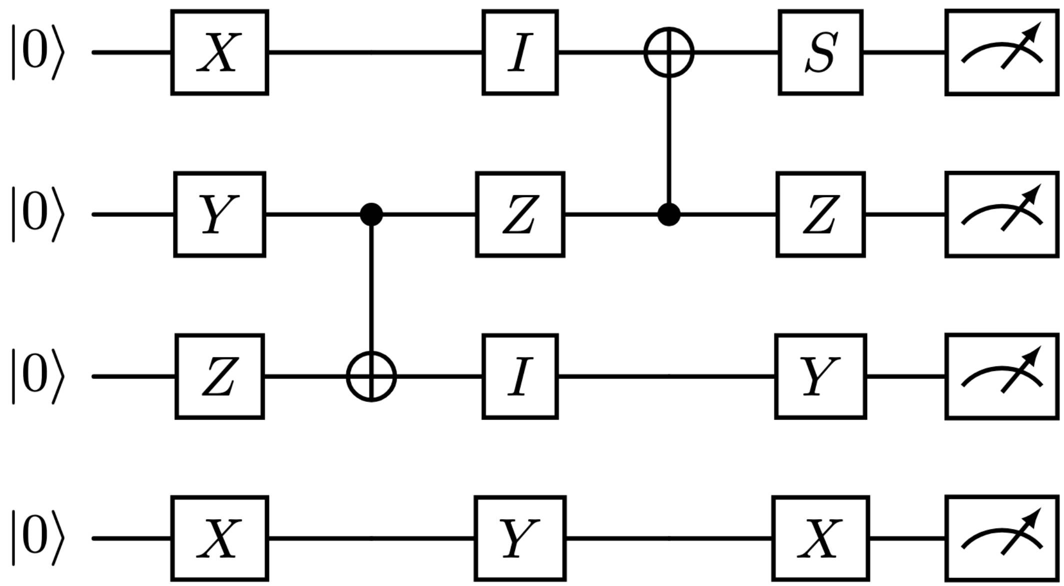
$e_F = 1 - F_P = r\frac{d+1}{d}$

Pauli Channel: $\mathcal{E}(\rho) = \sum_{P \in \mathcal{P}_n} p_P \, P \rho P^\dagger$ $\quad \sum_{P \in \mathcal{P}_n} p_P = 1$

Pauli Twirling: $\mathcal{T}(\mathcal{E}(\rho)) = \frac{1}{4^n} \sum_{P \in \mathcal{P}_n} P \, \mathcal{E}(P \rho P^\dagger) \, P^\dagger$

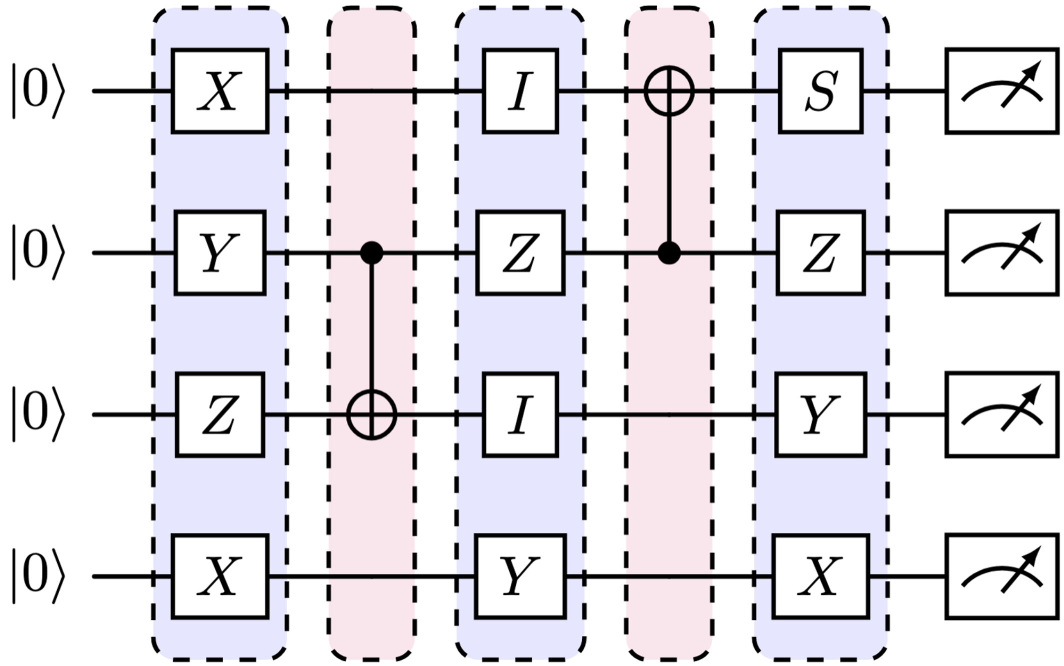
Noisy Channel → Pauli Channel

Randomized Compiling



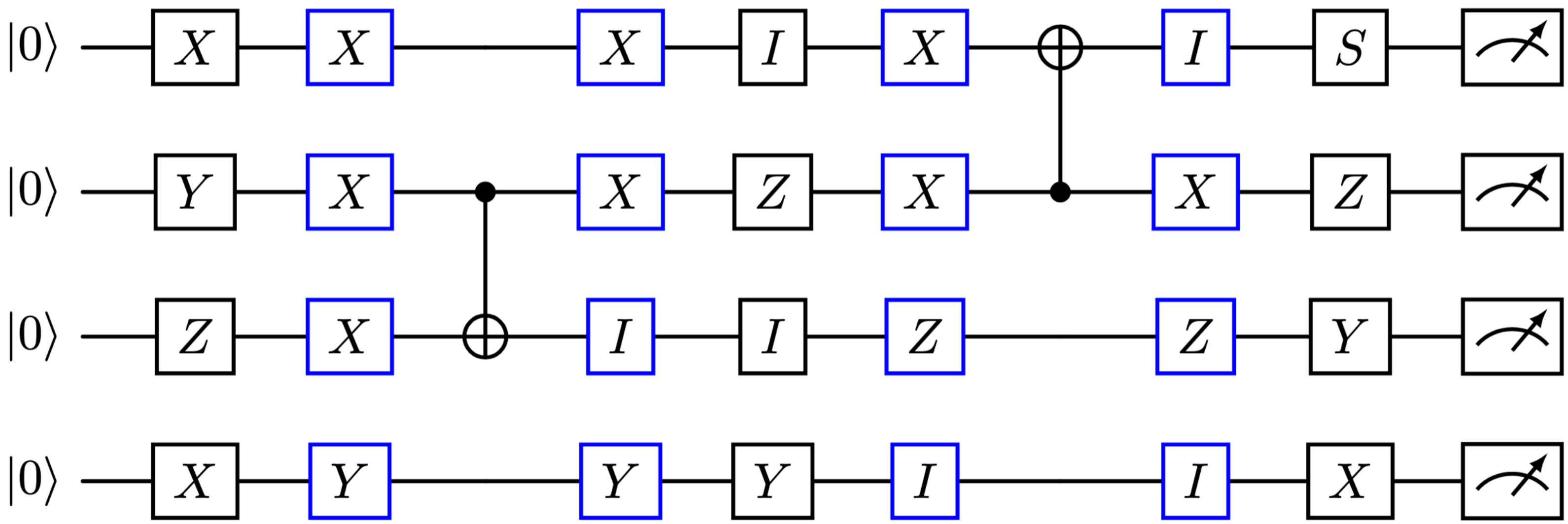
J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling



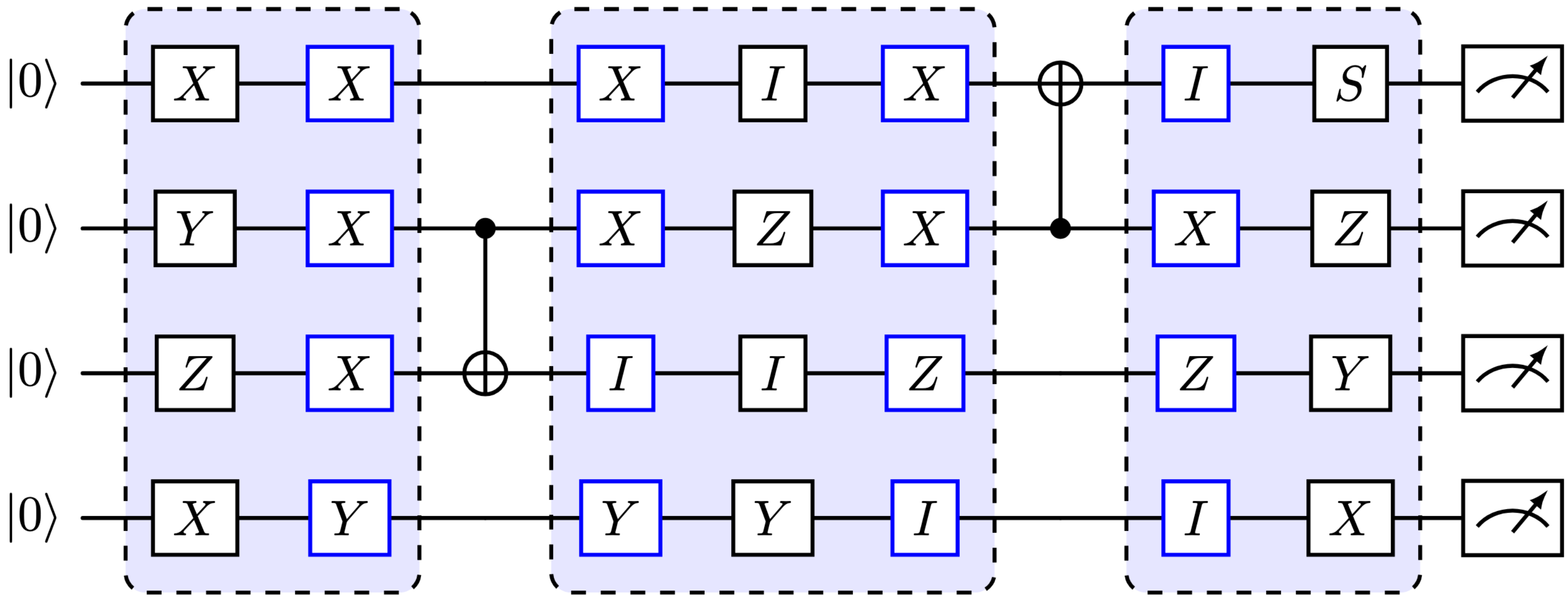
J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling



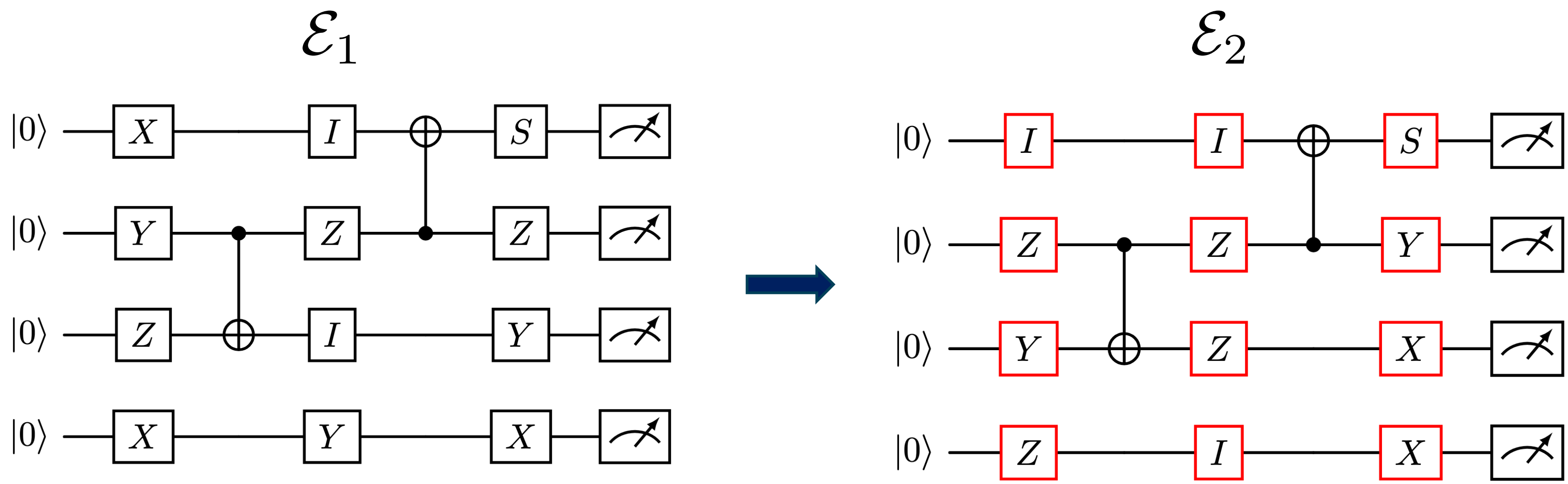
J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling



J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling



Before RC

After Compilation

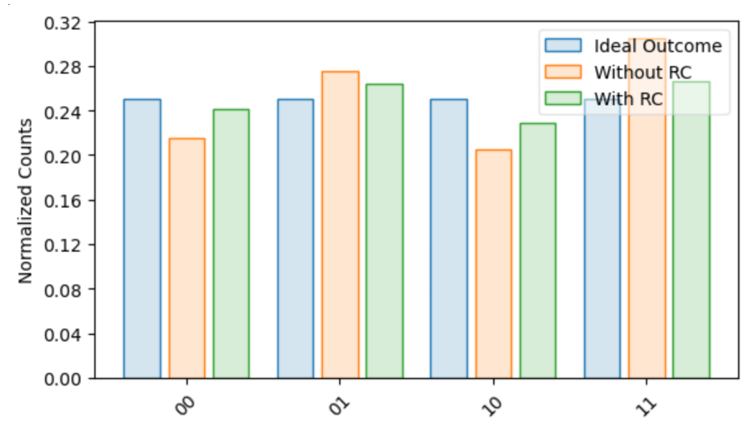
$$e_F(\mathcal{E}_1, \mathcal{E}_2) = 0$$

J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling

How to apply RC:

1. Define the desired quantum circuit
2. Rewrite the circuit as alternating cycles of easy and hard gates
3. Add random gates from the Pauli twirling set around the hard gate cycles
4. Compile the easy gates with the added Pauli gates
5. Run the compiled circuits and measure the results
6. Repeat steps 3-4-5 for multiple randomizations
7. Add up all the results, and normalize
8. Calculate TVD with respect to the ideal circuit



J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling

Why does RC work?

- Averaging randomizations has a similar effect as applying a Pauli Twirl to the noisy channel
- The noisy channel is tailored into a Pauli channel, while preserving the average gate fidelity of the circuit
- The more randomizations, the closer the noisy channel is to a Pauli channel

Advantages of Pauli Channel

- Substantially lower worst-case error rate
- The average error rate accumulates linearly with the length of a computation for stochastic Pauli errors, whereas it can accumulate quadratically for coherent errors.

J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

Randomized Compiling

Without RC $$d_{\mathrm{TV}}(\mathcal{P}_{\mathrm{noisy}}, \mathcal{P}_{\mathrm{ideal}}) \leq \sqrt{r(\mathcal{E})}\sqrt{d(d+1)}$$

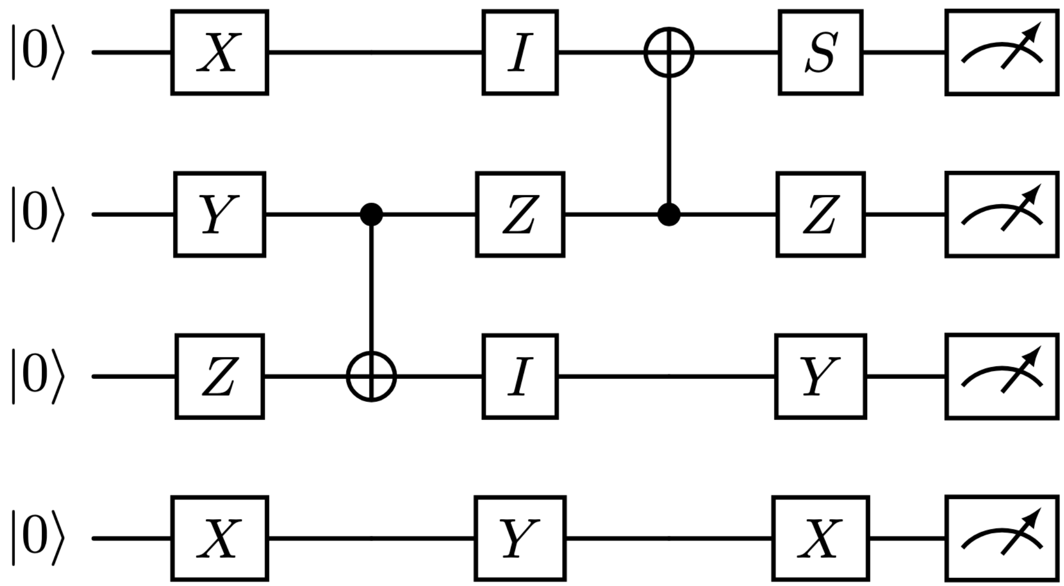
With RC $$d_{\mathrm{TV}}(\mathcal{P}_{\mathrm{RC}}, \mathcal{P}_{\mathrm{ideal}}) \leq r(\mathcal{E})\frac{d+1}{d}$$

$$r(\mathcal{E})\frac{d+1}{d} \leq \sqrt{r(\mathcal{E})}\sqrt{d(d+1)}$$

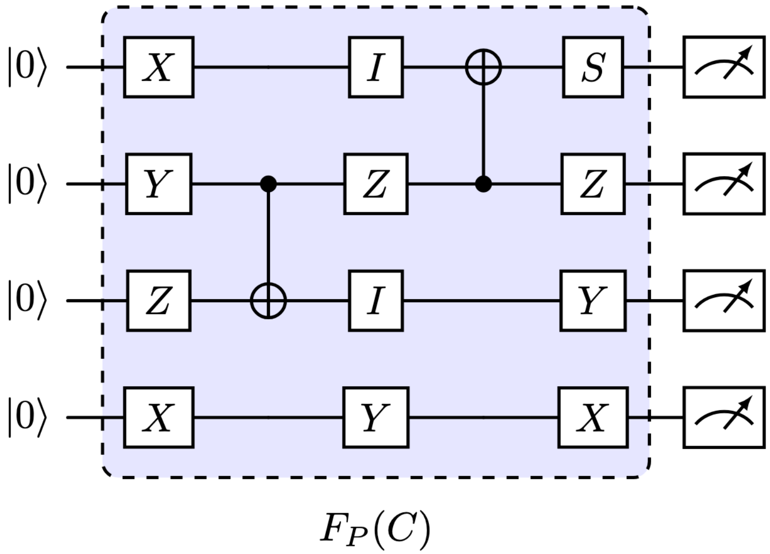
TVD under RC as a lower upper bound!

J. Wallman, J. Emerson - arXiv:1512.01098 [quant-ph]

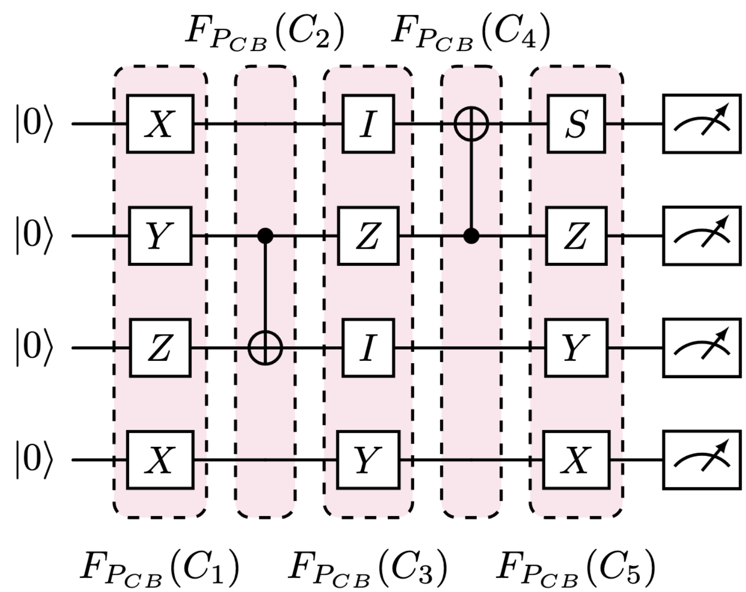
Circuit Benchmarking



Circuit Benchmarking



Process Fidelity



Process Fidelity Under Cycle Benchmarking

A. Erhard et al - arXiv:1902.08543 [quant-ph]

Predicted Process Fidelity

Let C be a circuit with cycles $\{C_i\}$ and depth n, then

$$F(C) \approx \prod_{i=1}^{n} F_{CB}(C_i)$$

Carignan-Dugas et al - arXiv:1904.08897 [quant-ph]

Questions to Answer

1. How tight is the upper bound $d_{\mathrm{TV}}(\mathcal{P}_{\mathrm{RC}}, \mathcal{P}_{\mathrm{ideal}}) \leq r(\mathcal{E})\dfrac{d+1}{d} = 1 - F(\mathcal{E})$?

2. How accurate is $F(C) \approx \prod_{i=1}^{n} F_{CB}(C_i)$?

3. Can the predicted process fidelity reliably estimate the effect of RC?

4. How do the answers to the above questions change with varying system parameters? (number of qubits, circuit depth, noise models and intensity)

Overview

1. Introduction
2. Background
3. Methods
4. Results
5. Next Steps

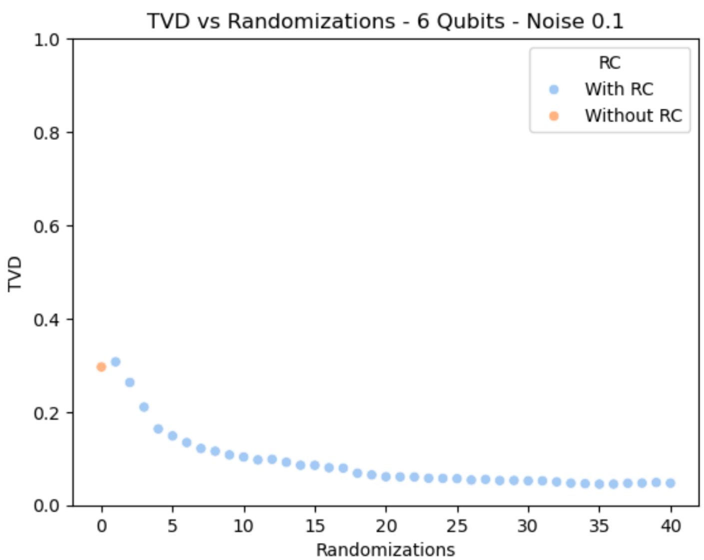
Methods

1. All the simulations have been done with TrueQ.
2. TrueQ has functions to run simulations, Randomized Compiling, Cycle Benchmarking and more.
3. The noise model used for all simulations in this presentation was an over-rotation error applied to the hard gate cycles, whereas the easy gate cycles were kept noiseless.

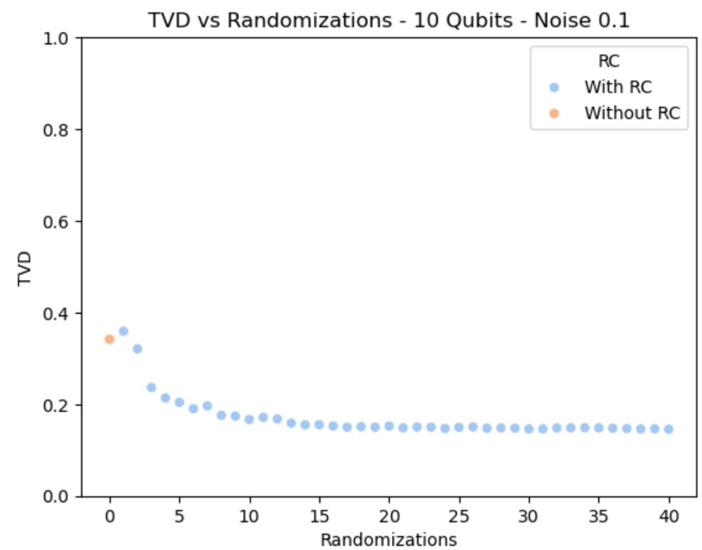
Overview

1. Introduction
2. Background
3. Methods
4. **Results**
5. Next Steps

Effect of Randomized Compiling

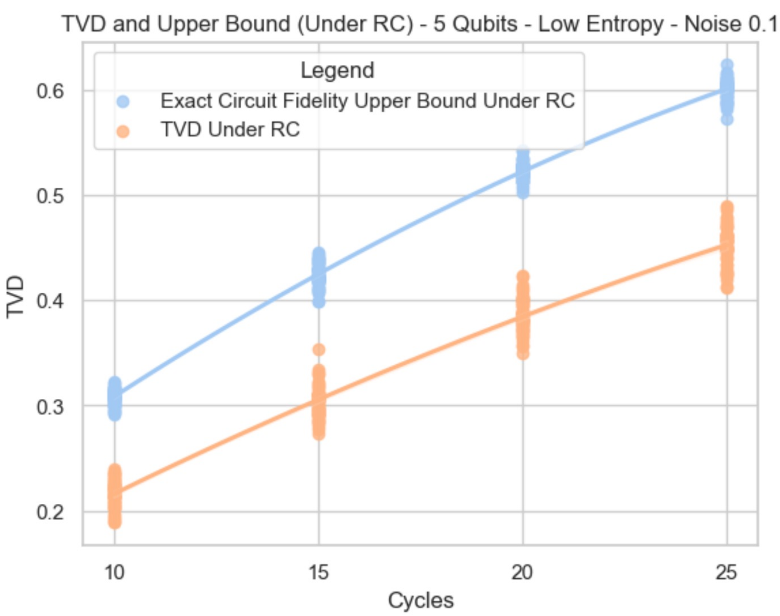


6 Qubit Circuit – 20 Cycles

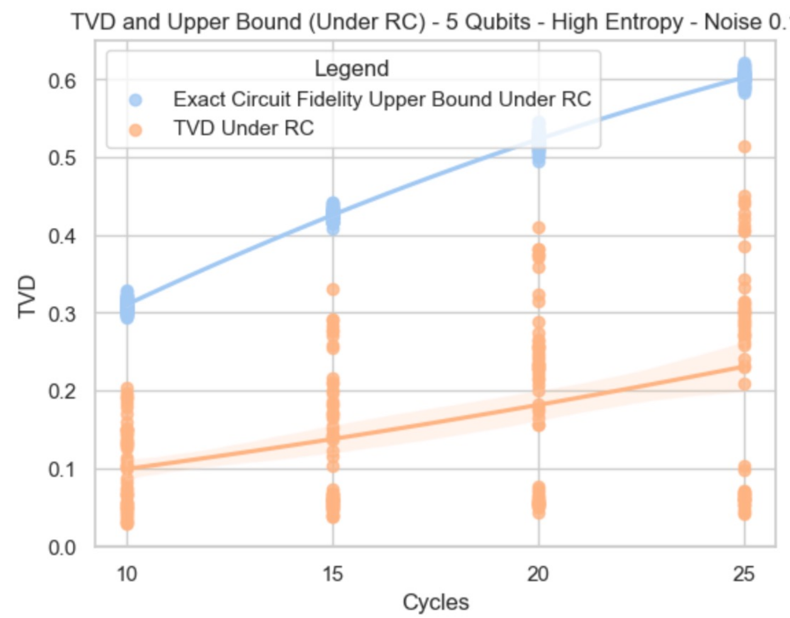


10 Qubit Circuit – 20 Cycles

How tight is $d_{\mathrm{TV}}(\mathcal{P}_{\mathrm{RC}}, \mathcal{P}_{\mathrm{ideal}}) \leq r(\mathcal{E})\frac{d+1}{d} = 1 - F(\mathcal{E})$?

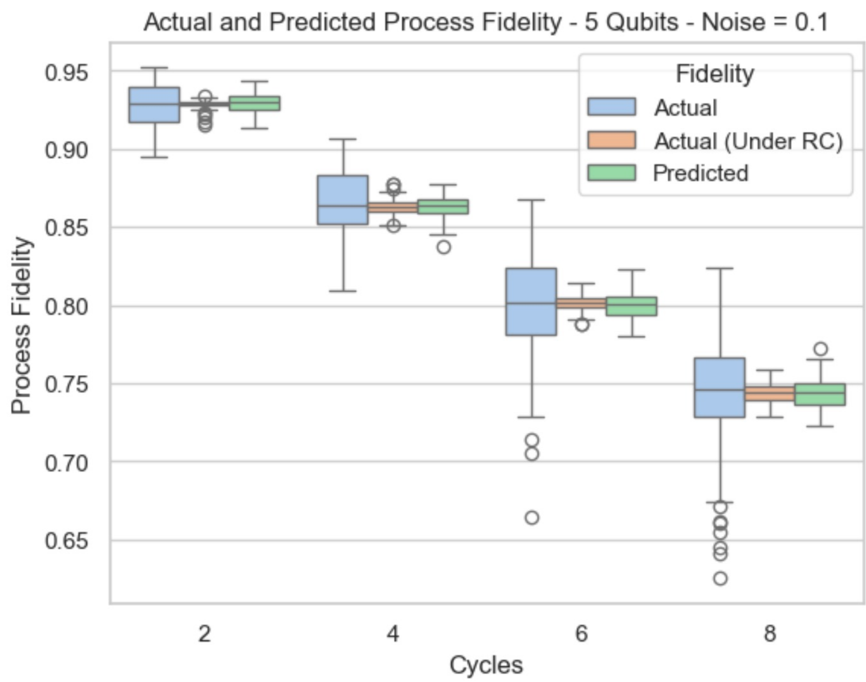


Low Entropy Circuit

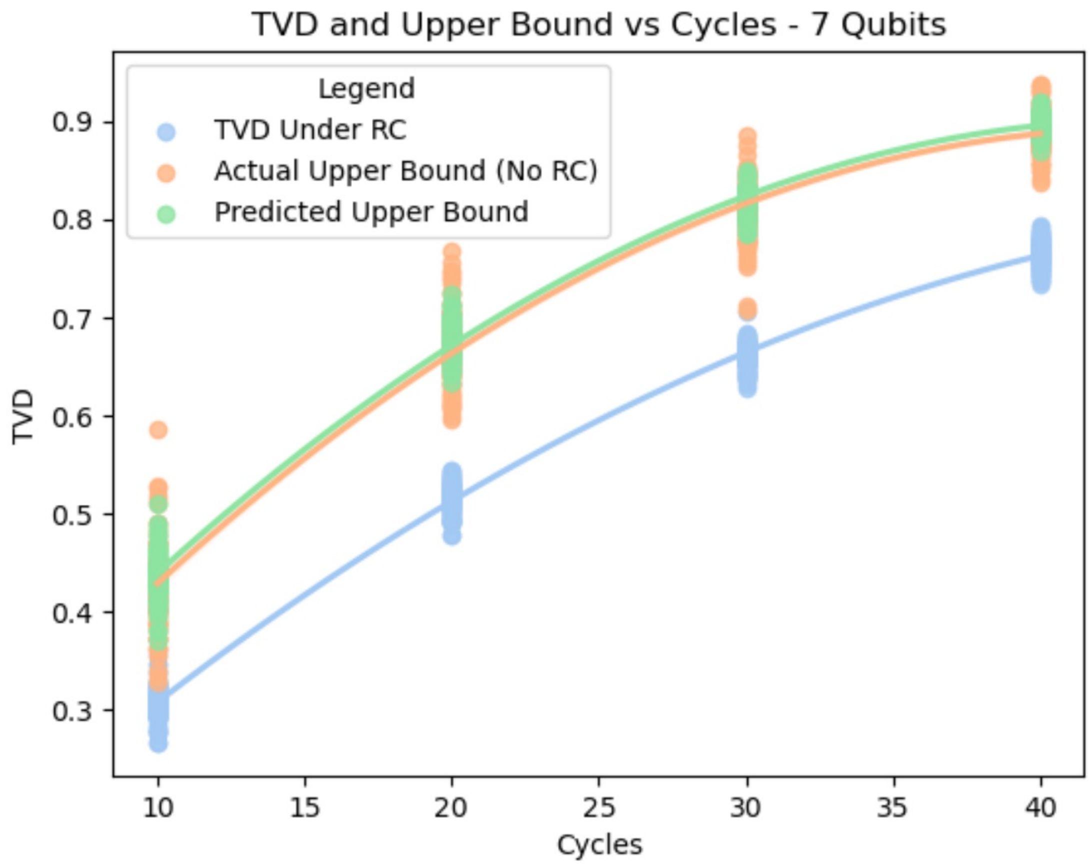


High Entropy Circuit

How accurate is $F(C) \approx \prod_{i=1}^{n} F_{CB}(C_i)$?



Upper Bound Estimation via Predicted Process Fidelity



Overview

1. Introduction
2. Background
3. Methods
4. Results
5. Next Steps

Next Steps

1. Showed numerically that $$d_{\mathrm{TV}}(\mathcal{P}_{\mathrm{RC}}, \mathcal{P}_{\mathrm{ideal}}) \leq r(\mathcal{E})\frac{d+1}{d} = 1 - F(\mathcal{E})$$
   For 5 qubits and over-rotation error

2. Showed numerically that $$F(C) \approx \prod_{i=1}^{n} F_{CB}(C_i)$$
   For 5 qubits and over-rotation error

3. Defined Python functions to automate the data recollection process.

4. Try with more qubits, longer circuits, different noise models.

5. Test in a real quantum computer.

Thanks!