

# Hardness of Computing Fault Tolerance.

David Ponnarovsky

June 22, 2025

# Introduction

- ▶ Noisy Circuits.
- ▶ Importance and relevance
- ▶ Objectives of the presentation

# Nosiy Circuit.



# Nosiy Circuit.

## Definition

$p$ - Depolarizing Channel. The qubit depolarizing channel with parameter  $p \in [0, 1]$  is the quantum channel  $\mathcal{D}_p$  defined by:

$$\mathcal{D}_p(\rho) = (1 - p)\rho + p \cdot \frac{I}{2}$$

where  $\rho$  is a single-qubit density matrix and  $I$  is the identity matrix.

## Definition

$p$ -Noisy Circuit. Given a circuit  $C$  (regardless of the model), its  $p$ -noisy version  $\tilde{C}$  is the circuit obtained by alternately taking layers from  $C$  and then passing each (qu)bit through a  $p$ -Depolarizing channel.

# Threshold Theorem.

## Theorem (Threshold Theorem. Informal.)

*There is a universal  $p_{th} \in (0, 1)$  such that for any  $p < p_{th}$ , any circuit in BQP can be simulated by a  $p$ -noisy BQP circuit. The simulating circuit has a depth that is at most  $\text{polylog } n$  times the original depth.*

# Definition

## Definition (**NC** - Nick's Class)

**NC**<sub>*i*</sub> is the class of decision problems solvable by a uniform family of Boolean circuits, with polynomial size, depth  $O(\log^i(n))$ , and fan-in 2.

## Definition (**QNC**)

The class of decision problems solvable by polylogarithmic-depth, and finite fan out/in quantum circuits with bounded probability of error. Similarly to **NC**<sub>*i*</sub>, **QNC**<sub>*i*</sub> is the class where the deciding circuits have  $\log^i(n)$  depth.

## Definition (**QNC<sub>G</sub>**)

For a fixed finite fan in/out gateset  $G$ , the class with deciding circuits composed only for gates in  $G$  and at depth at most polylogarithmic. And in similar to **QNC**<sub>*i*</sub>, **QNC<sub>G,i</sub>** is the restriction to circuits with depth at most  $\log^i(n)$ .

# Pippenger's Construction.

Encode each bit with the repetition code  $0 \mapsto 0^m, 1 \mapsto 1^m$ . Now observe that any logical operation, without decoding, can be made in  $O(1)$  depth.

For example,  $\text{OR}(\bar{x}, \bar{y})$  can be computed by applying in parallel  $\text{OR}(x_i, y_i)$  for each  $i$ .

# The 'Decoding' trick.

Instead of completely decoding, we would apply only a single step of partial decoding. We assume that in each code block the bits are partitioned into random disjoint triples, and we will apply a local correction to each of the triples by majority.

## Claim

There are constants  $\alpha, \eta \in (0, 1)$  such that for any bit string  $x$  at a distance  $\leq \alpha n$  from the code (Repetition Code), one cycle of local correction on  $x$  yields  $x'$  such that:

$$d(x', C) \leq d(x, C)$$



# The 'Decoding' trick.

Suppose that a bit absorb a bit flip with probability  $p$ . So in expectation we expect that entire block at length  $n$  will absorb  $pn$  flips.

$$\eta(\beta + p)n \leq \beta n$$
$$\beta \geq \frac{p}{1 - \eta}$$

From now on, we will assume that the graphs are bipartite and we will denote the right and the left vertices by  $V^-$  and  $V^+$ . Notice that such expanders near Ramanujan exist, see for example [?]. The partition into two subsets enable us to come with a simple efficient decoder.

Expanders code are known for having good decoders, beneath, in ?? , we introduce a procedure to reduce an error. In overall, we alternately let to the right and then the left vertices to correct their own local view. In Theorem 7 we prove that when the applied error has size at most  $\beta n$ , for some constant  $\beta$  then the error's weight reduced by  $\frac{1}{2}$ . Repeating over the procedure  $\Theta(\log(n))$  times completely correct the error.

We will call to the first stage, when only the right vertices suggest correction the right round, and to the second stage a left round.  
For the whole procedure, we will call a single correction round.

**Data:**  $x \in \mathbb{F}_2^n$

**Result:**

$$\arg \min \{y \in C : |y + x|\}$$

$$\text{if } d(y, C) <$$

```

1 for  $v \in V^+$  do
2    $x'_v \leftarrow$ 
      $\arg \min \{y \in C_0 : |y + x|_v|\}$ 
3 end
4 for  $v \in V^-$  do
5    $x'_v \leftarrow$ 
      $\arg \min \{y \in C_0 : |y + x|_v|\}$ 
6 end
7 return  $x$ 
   : Single decoding round

```



## Lemma

*If the error is at wight less than  $\beta n$  then a single round of the majority reduce the error by at least constant fraction.*

Denote by  $S^{(0)} \subset V^+$  and  $T^{(0)} \subset V^-$  the subsets of left and right vertices adjacent to the error. And denote by  $T^{(1)} \subset T^{(0)}$  the right vertices such any of them is connect by at least  $\frac{1}{2}\delta_0\Delta$  edges to vertices at  $S^{(0)}$ . Note that that any vertex in  $V^-/T^{(1)}$  has on his local view less than  $\frac{1}{2}\delta_0\Delta$  faulty bits, So it corrects into his right local view in the first right correction round. Therefore after the right correction round the error is set only on  $T^{(1)}$ 's neighbourhood, namely at size at most  $\Delta|T^{(1)}|$ . We will show that this amount is strictly lower by a constant factor than  $|e|$ .

First, let's use the expansion property (??) for getting an upper bound on  $T^{(1)}$  size:

$$\begin{aligned}\frac{1}{2}\delta_0\Delta|T^{(1)}| &\leq \Delta\frac{|T^{(1)}||S^{(0)}|}{n} + \lambda\sqrt{|T^{(1)}||S^{(0)}|} \\ \left(\frac{1}{2}\delta_0\Delta - \frac{|S^{(0)}|}{n}\Delta\right)|T^{(1)}| &\leq \lambda\sqrt{|T^{(1)}||S^{(0)}|} \\ |T^{(1)}| &\leq \left(\frac{1}{2}\delta_0\Delta - \frac{|S^{(0)}|}{n}\Delta\right)^{-2} \lambda^2|S^{(0)}|\end{aligned}$$

Since any left vertex adjoins to at most  $\Delta$  faulty bits we have that  $\Delta|S^{(0)}| \leq |e|$ . Combing with the inequality above we get:

$$\Delta|T^{(1)}| \leq \left(\frac{1}{2}\delta_0\Delta - \frac{|e|}{n}\right)^{-2} \lambda^2|e|$$

Hence for  $|e|/n \leq \beta = \frac{1}{2}\delta_0\Delta - \sqrt{2\lambda}$  it holds that  $\Delta|T^{(1)}| \leq \frac{1}{2}|e|$ .  
Namely the error is reduced by half.

# The Franch's Construction.

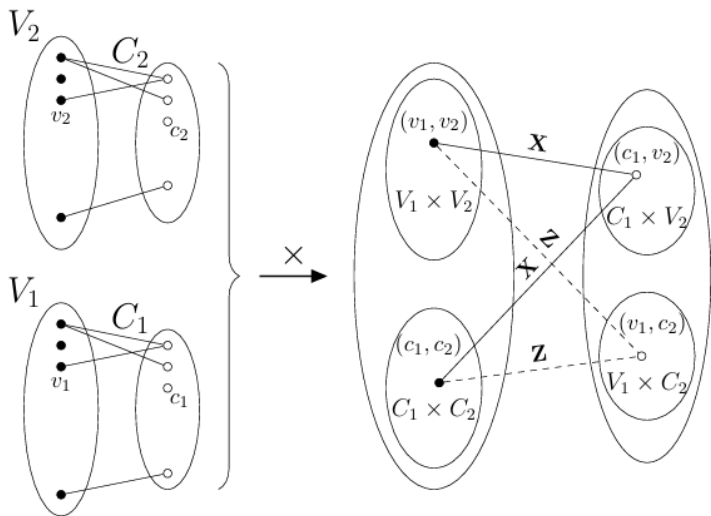


Figure: Caption for the image



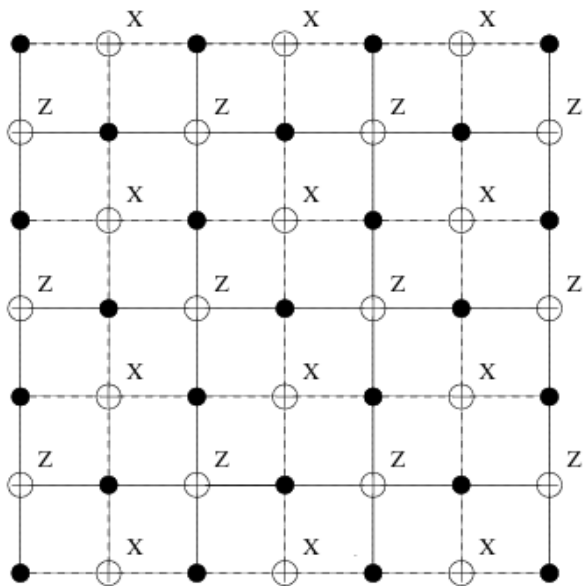


Figure: Caption for the image

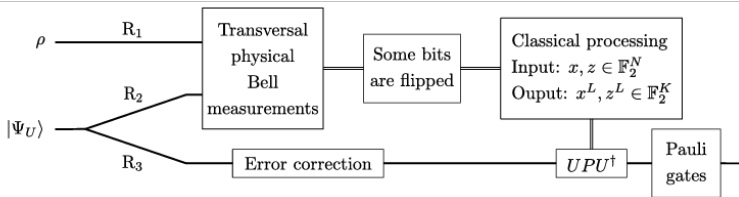


Figure: Caption for the image

# Disjointness.