



Efficient decoding of random errors for quantum expander codes

Antoine Grospellier, Anthony Leverrier, Omar Fawzi

► To cite this version:

Antoine Grospellier, Anthony Leverrier, Omar Fawzi. Efficient decoding of random errors for quantum expander codes. Journées Informatique Quantique 2017, Nov 2017, Bordeaux, France. pp.521-534. hal-01671491

HAL Id: hal-01671491

<https://hal.science/hal-01671491>

Submitted on 22 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Efficient decoding of random errors for quantum expander codes

Antoine Grospellier & Anthony Leverrier & Omar Fawzi

10 Novembre 2017





Alice



Bob

m
k Q-bits



Alice



Bob

m
k Q-bits

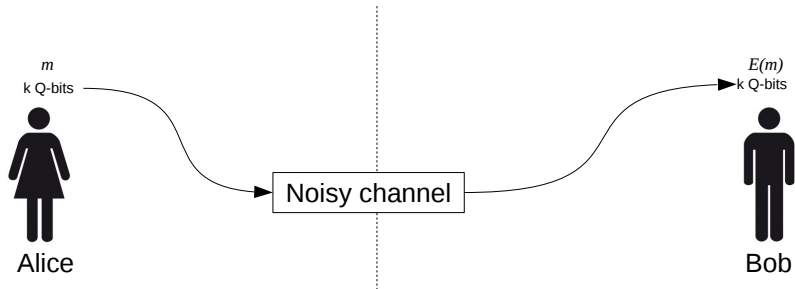


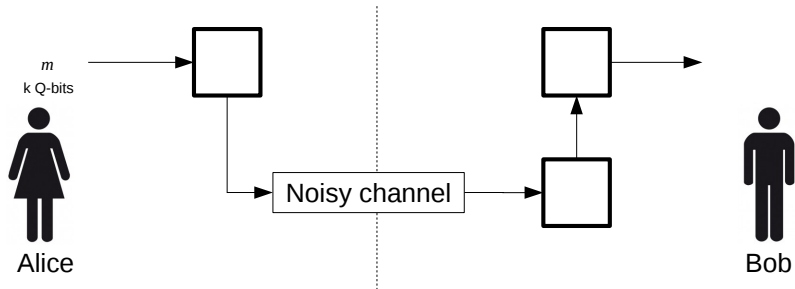
Alice

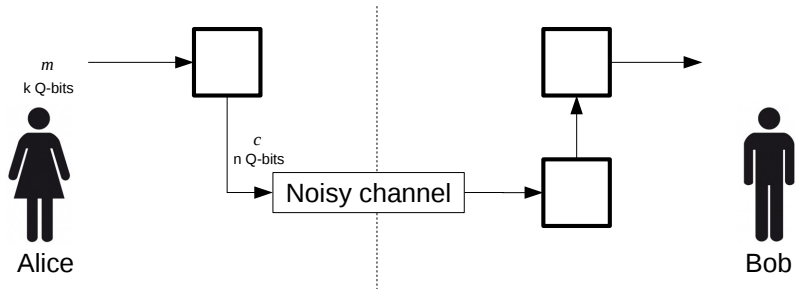
Noisy channel

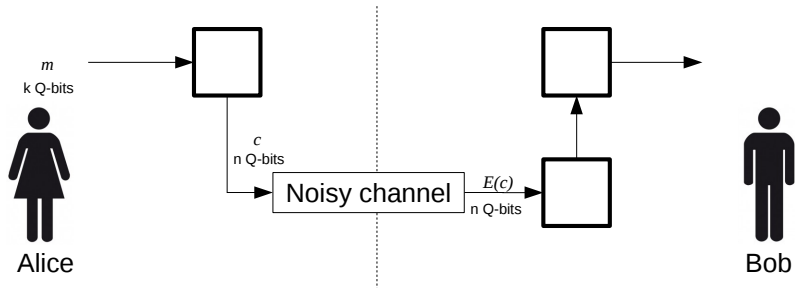


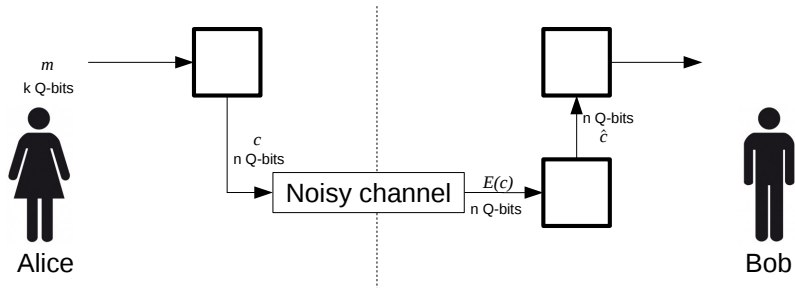
Bob

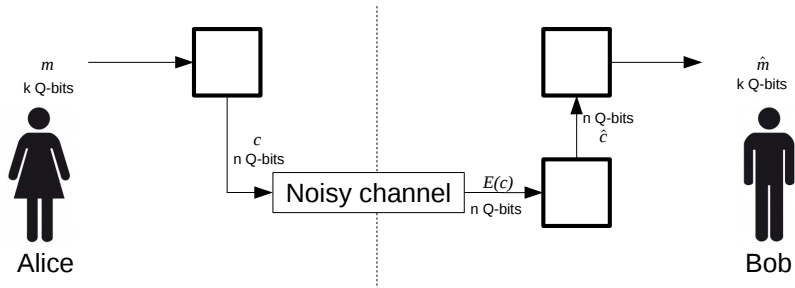












Content of the talk

The hypergraph product of an expander code :

- is an LDPC quantum code
- has a constant rate
- has a minimal distance : $d = \Theta(\sqrt{n})$

The decoding algorithm :

- has a capacity of correction : $\Theta(\sqrt{n})$
- **corrects the error with high probability for the depolarizing channel**

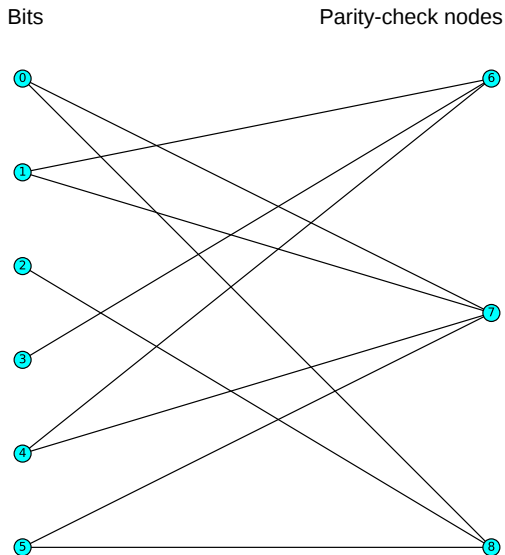
Content of the talk

- 1 Classical expander codes
- 2 Quantum expander codes
- 3 Our contribution

Plan

- 1 Classical expander codes
- 2 Quantum expander codes
- 3 Our contribution

Tanner graph of a code



Classical expander codes [Sipser & Spielman, '96]

Theorem [Sipser & Spielman, '96]

We can construct a good family $(\mathcal{C}_n)_{n \in \mathbb{N}}$ of $[n, k, d]$ -error correcting codes. Here "Good" means :

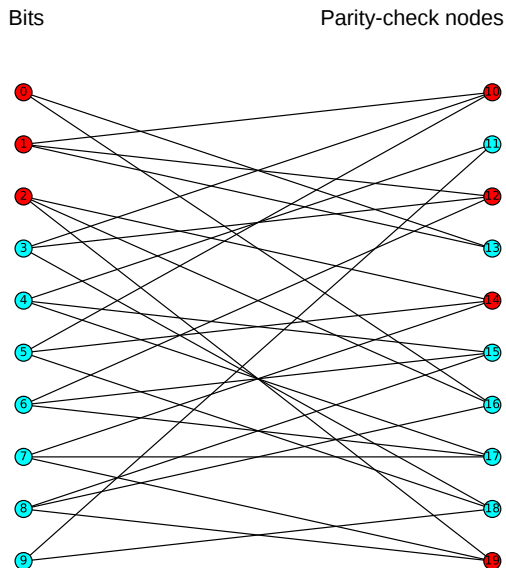
- This family is LDPC
- k and d are linear in n
- There exists an efficient correcting algorithm

Remark

Good expander codes can be found efficiently by picking a random biregular graph

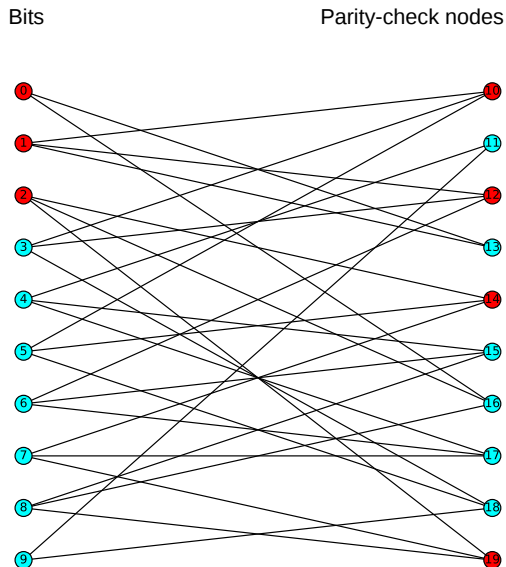
Decoding algorithm

- Error :
 $e_0 = \{0, 1, 2\}$
- Unsatisfied
check-nodes
(syndrome) :
 $\{10, 12, 14, 19\}$
- Satisfied
check-nodes :
 $\{11, 13, 15, 16, 17, 18\}$

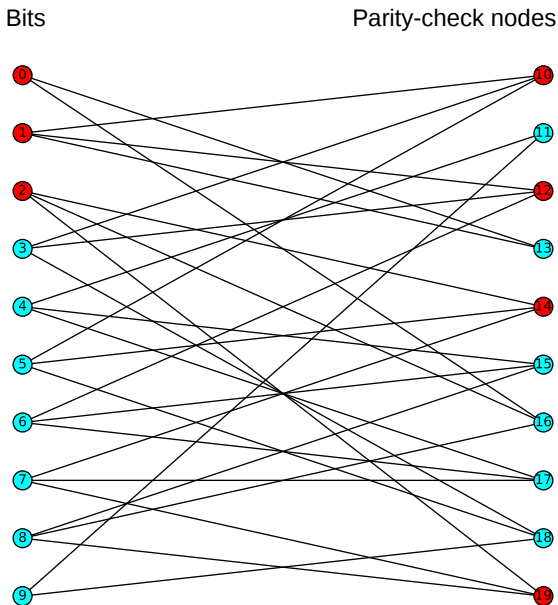


Decoding algorithm

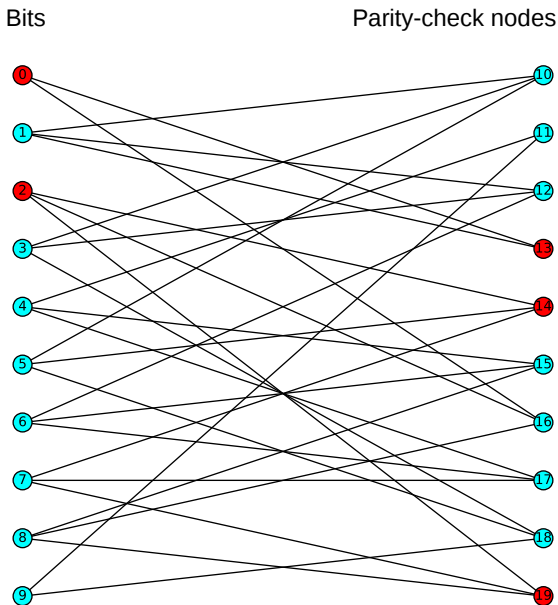
- INPUT : syndrome
- The error e_0 is unknown
- OUTPUT : e
a set of bits
- GOAL : $e = e_0$
- The algorithm flips a bit when it decreases the syndrome



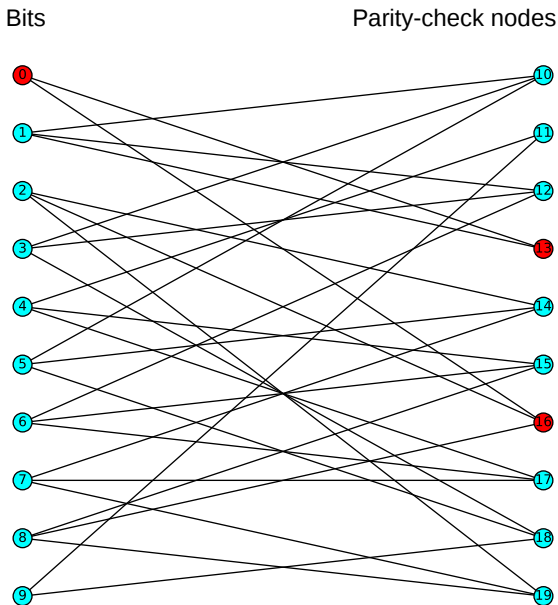
Decoding algorithm : first example



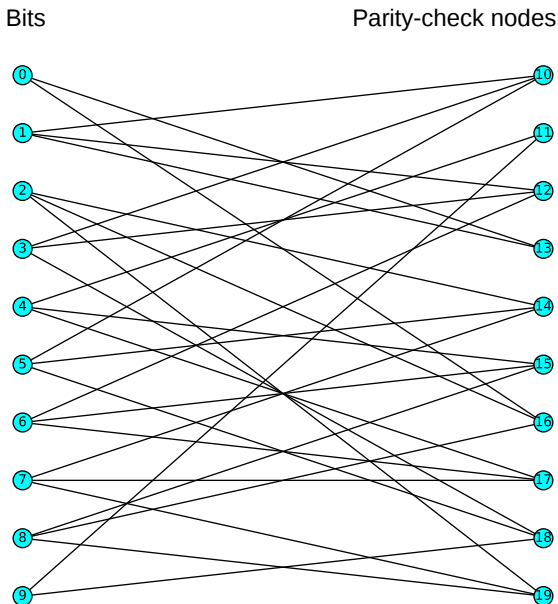
Decoding algorithm : first example



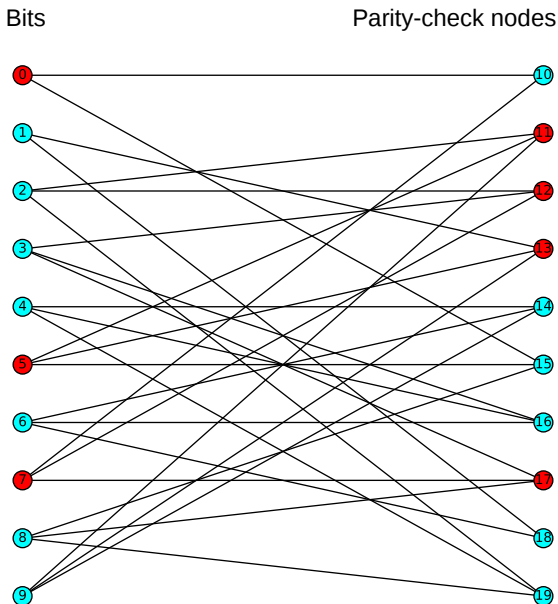
Decoding algorithm : first example



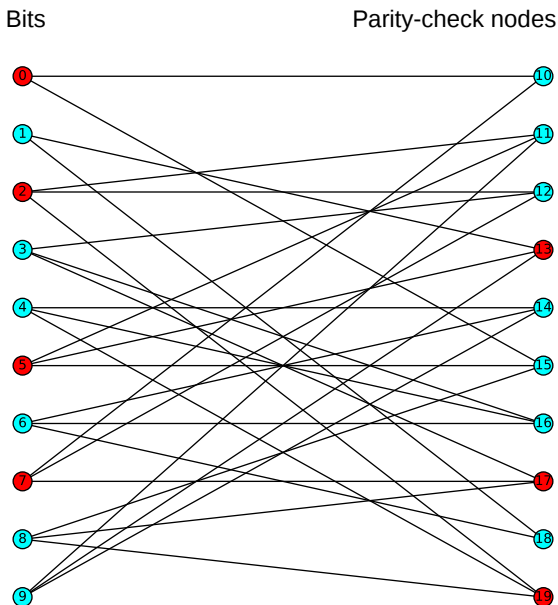
Decoding algorithm : first example



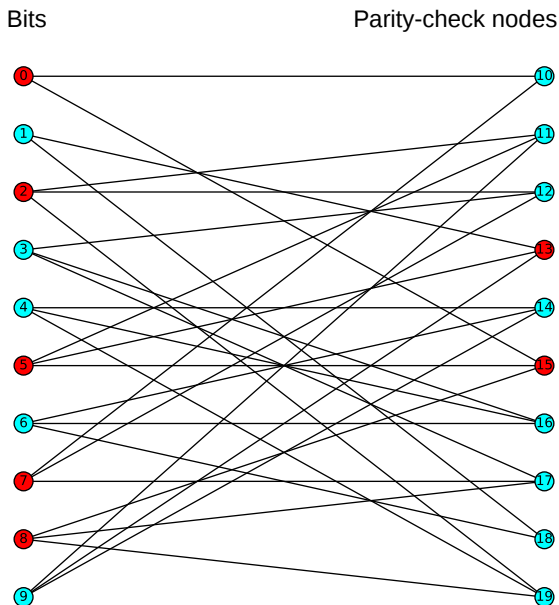
Decoding algorithm : second example



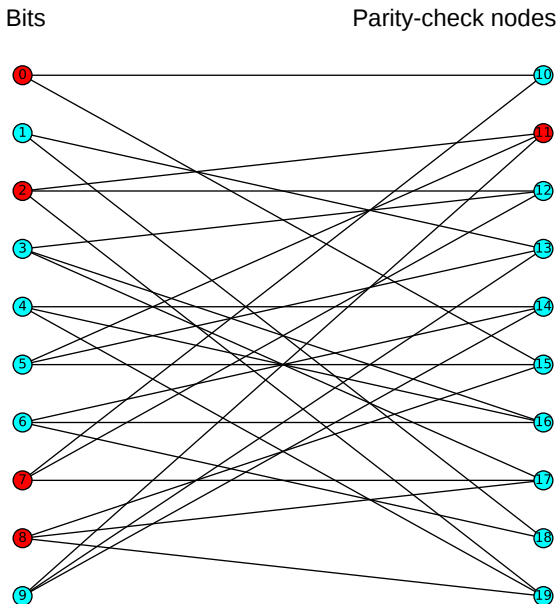
Decoding algorithm : second example



Decoding algorithm : second example



Decoding algorithm : second example



Plan

- 1 Classical expander codes
- 2 Quantum expander codes
- 3 Our contribution

From classical to quantum error correcting codes

Hypergraph product [Tillich & Zémor, '09]

Using a classical code $\mathcal{C}$, we can construct a $[[n, k, d]]$ -CSS code with :

- $k = \Theta(n)$
- $d = \Theta(\sqrt{n})$: we can correct any error on d Q-bits

From classical to quantum error correcting codes

Hypergraph product [Tillich & Zémor, '09]

Using a classical code $\mathcal{C}$, we can construct a $[[n, k, d]]$ -CSS code with :

- $k = \Theta(n)$
- $d = \Theta(\sqrt{n})$: we can correct any error on d Q-bits

Theorem [Leverrier & Tillich & Zémor, '15]

For the hypergraph product of an expander code ($\epsilon < 1/6$) :

There is an efficient decoding algorithm for this code.

This algorithm corrects any error of size $\leq \Theta(\sqrt{n})$

This algorithm is very close to the algorithm of Sipser and Spielman

Plan

- 1 Classical expander codes
- 2 Quantum expander codes
- 3 Our contribution

Our work

- **Question** : What happens for random errors of size $\Theta(n)$?
- **Depolarizing channel** : each Q-bit has an X-type error (resp. Y,Z-type error) with probability p independently

Our work

- **Question** : What happens for random errors of size $\Theta(n)$?
- **Depolarizing channel** : each Q-bit has an X-type error (resp. Y,Z-type error) with probability p independently

Theorem : what we proved

For a probability of error $p < 10^{-16}$:

$$\lim_{n \rightarrow +\infty} \mathbb{P}(\mathcal{A} \text{ corrects the error}) = 1$$

Our work

- **Question** : What happens for random errors of size $\Theta(n)$?
- **Depolarizing channel** : each Q-bit has an X-type error (resp. Y,Z-type error) with probability p independently

Theorem : what we proved

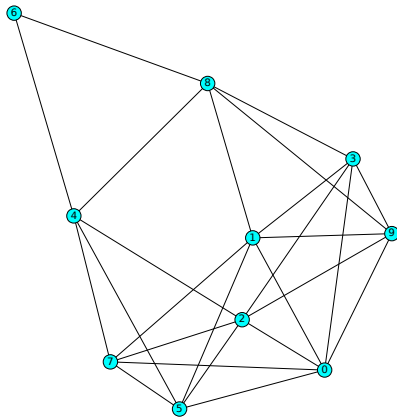
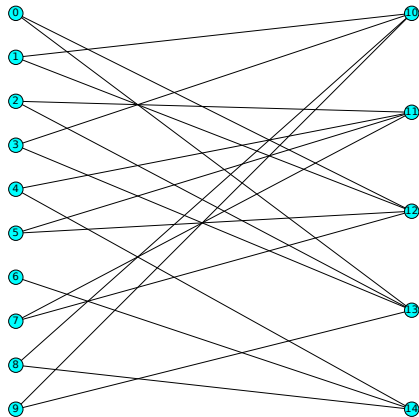
For a probability of error $p < 10^{-16}$:

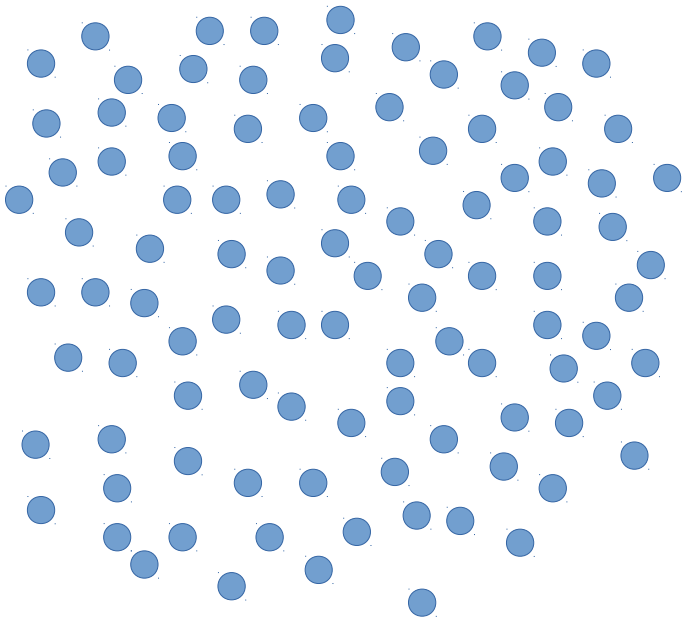
$$\lim_{n \rightarrow +\infty} \mathbb{P}(\mathcal{A} \text{ corrects the error}) = 1$$

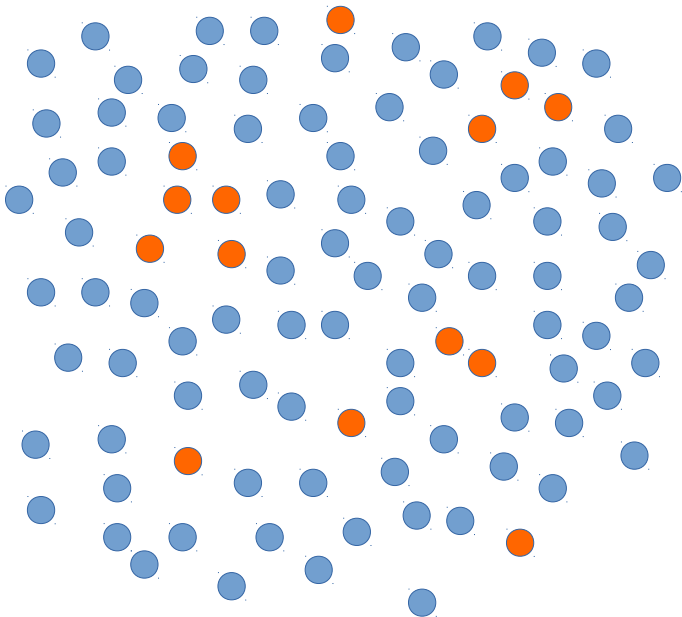
Idea : The algorithm is local :

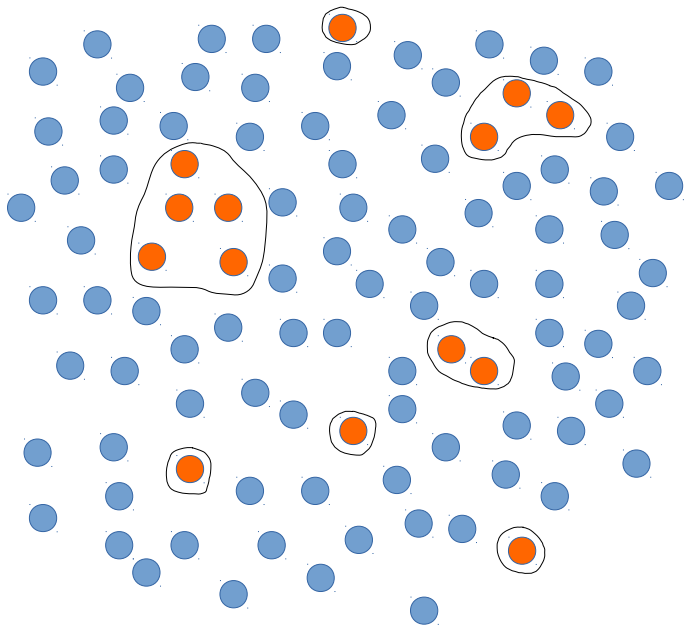
- If two errors are far they don't interact
- The initial error can be decomposed in clusters of size $O(\ln(n))$
- If there is no cluster of size $\Theta(\sqrt{n})$ during the algorithm, the error will be corrected

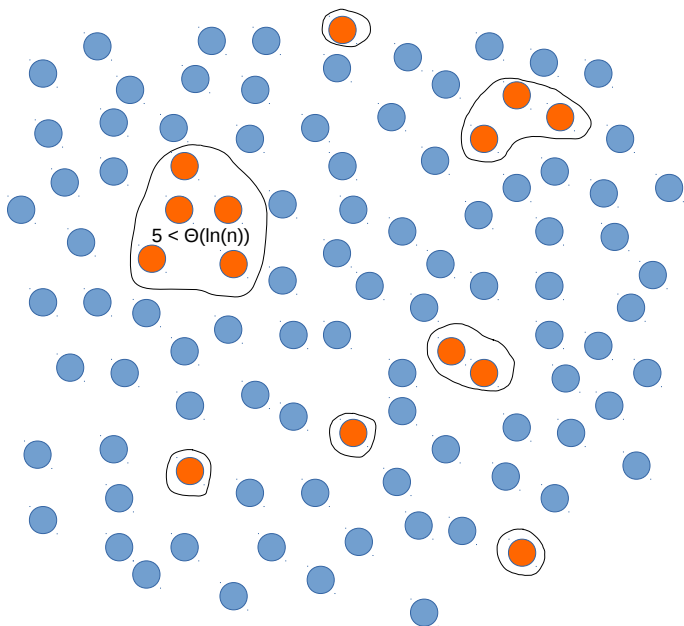
Locality of the algorithm : the adjacency graph











In the following, we will say whp P (with high probability the property P holds) if : $\lim_{n \rightarrow +\infty} \mathbb{P}(P) = 1$

In the following, we will say whp P (with high probability the property P holds) if : $\lim_{n \rightarrow +\infty} \mathbb{P}(P) = 1$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

For a probability of error $p > \frac{1}{d-1}$, whp :

- There is a connected component of size $\Theta(n)$

In the following, we will say whp P (with high probability the property P holds) if : $\lim_{n \rightarrow +\infty} \mathbb{P}(P) = 1$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

For a probability of error $p > \frac{1}{d-1}$, whp :

- There is a connected component of size $\Theta(n)$

Good news :

The algorithm corrects any error of size $\leq \Theta(\sqrt{n})$

The algorithm corrects any error of size $\leq \Theta(\ln(n))$

In the following, we will say whp P (with high probability the property P holds) if : $\lim_{n \rightarrow +\infty} \mathbb{P}(P) = 1$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

For a probability of error $p > \frac{1}{d-1}$, whp :

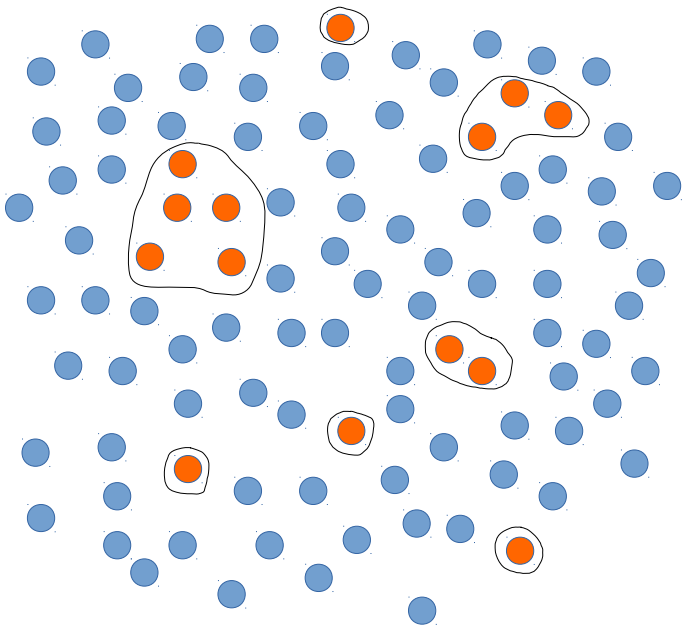
- There is a connected component of size $\Theta(n)$

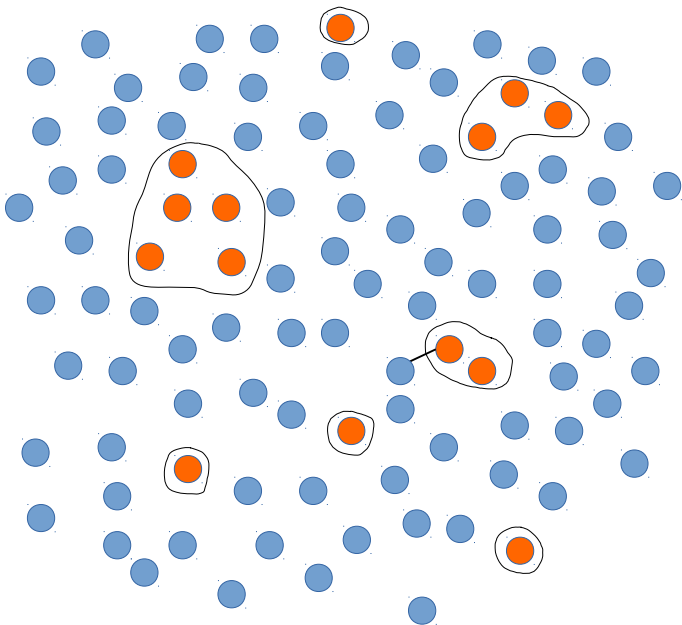
Good news :

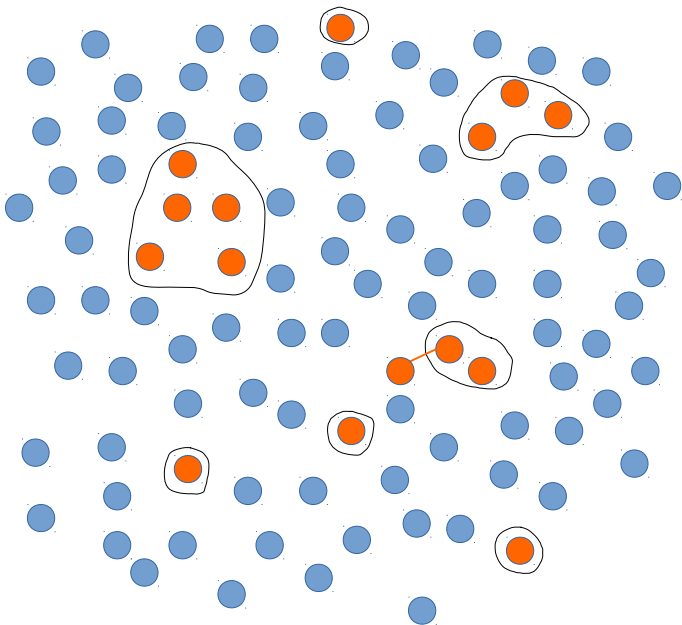
The algorithm corrects any error of size $\leq \Theta(\sqrt{n})$

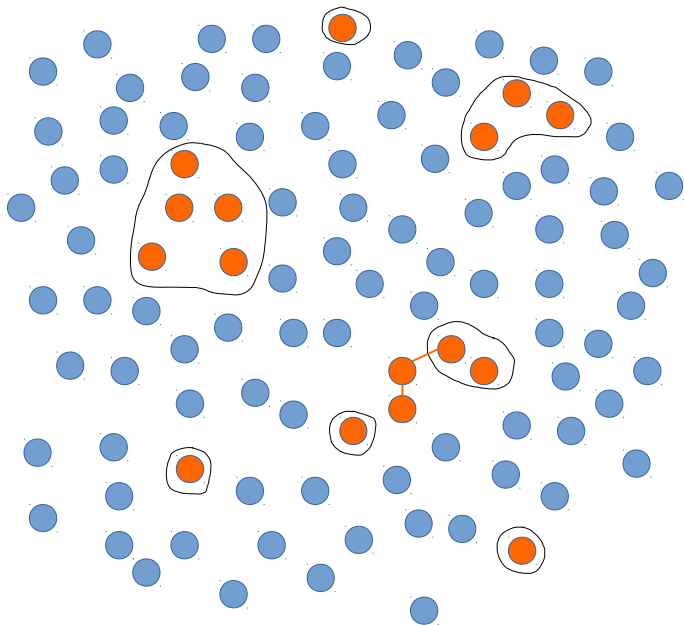
The algorithm corrects any error of size $\leq \Theta(\ln(n))$

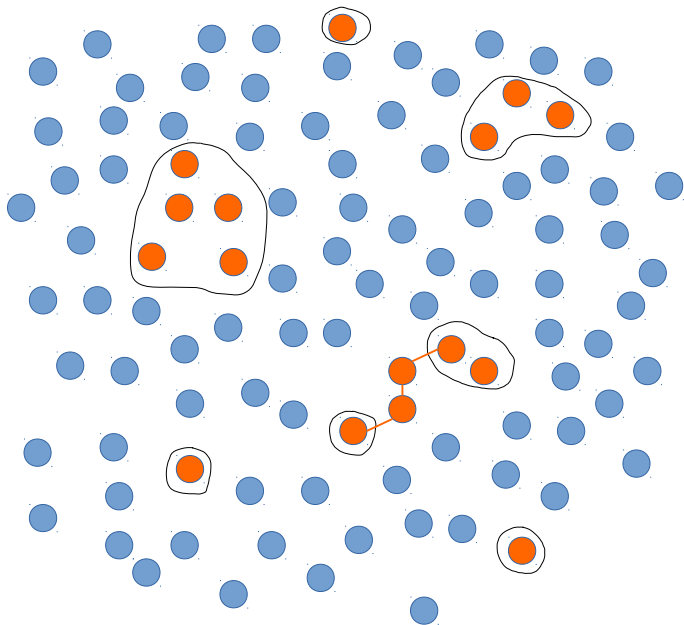
Problem : Some clusters can merge during the decoding











Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

Percolation theorem, reformulation

For a probability of error $p < \frac{1}{d-1}$, whp :

- if $|X \cap E(p)| \geq 1 \times |X|$ then $|X| < \Theta(\ln(n))$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

Percolation theorem, reformulation

For a probability of error $p < \frac{1}{d-1}$, whp :

- if $|X \cap E(p)| \geq 1 \times |X|$ then $|X| < \Theta(\ln(n))$

Percolation theorem, generalisation

$\forall \alpha > 0$, if $p < cst(\alpha, d)$, whp :

- if $|X \cap E(p)| \geq \alpha \times |X|$ then $|X| < \Theta(\ln(n))$

Percolation Theorem

For a probability of error $p < \frac{1}{d-1}$, whp :

- The size of any connected components is $\leq \Theta(\ln(n))$

Percolation theorem, reformulation

For a probability of error $p < \frac{1}{d-1}$, whp :

- if $|X \cap E(p)| \geq 1 \times |X|$ then $|X| < \Theta(\ln(n))$

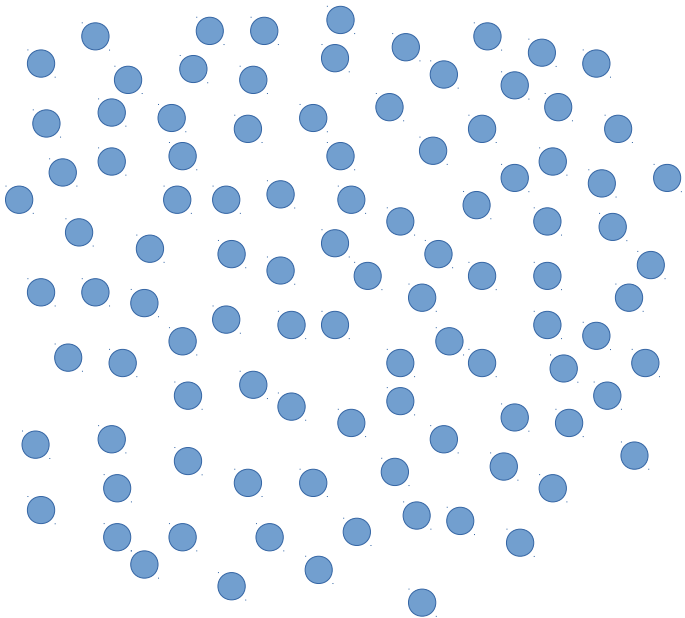
Percolation theorem, generalisation

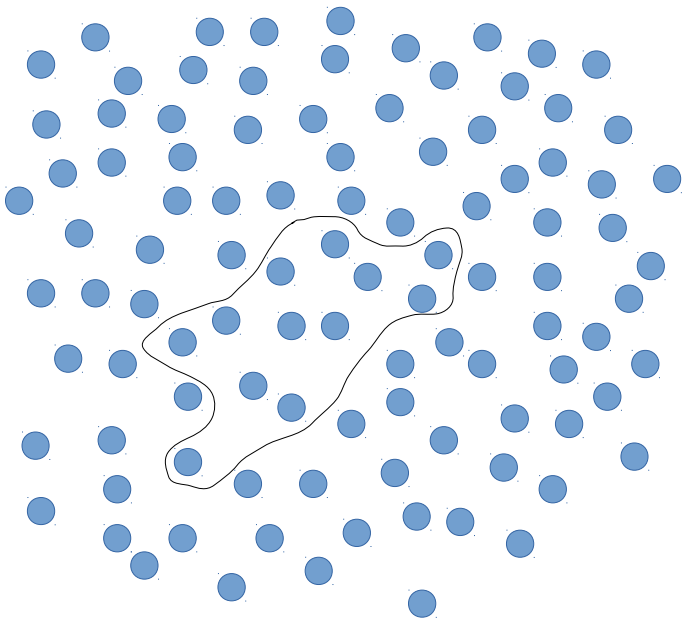
$\forall \alpha > 0$, if $p < cst(\alpha, d)$, whp :

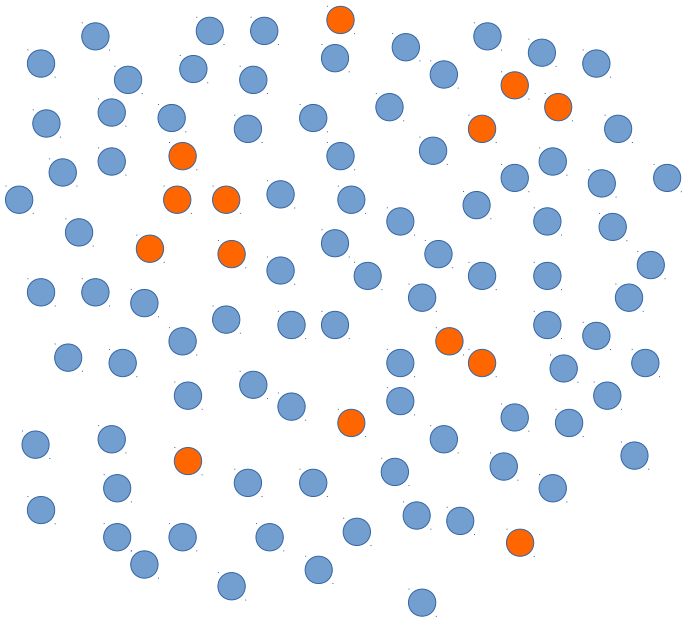
- if $|X \cap E(p)| \geq \alpha \times |X|$ then $|X| < \Theta(\ln(n))$

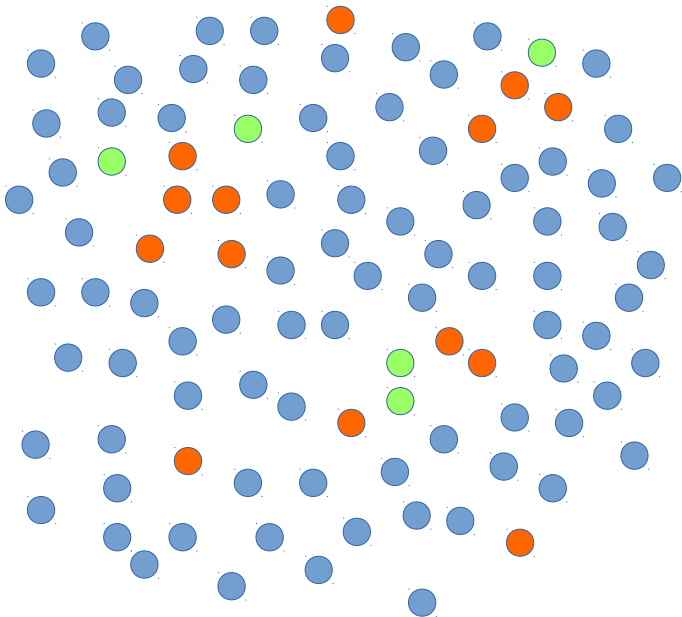
Complexity of the decoding algorithm

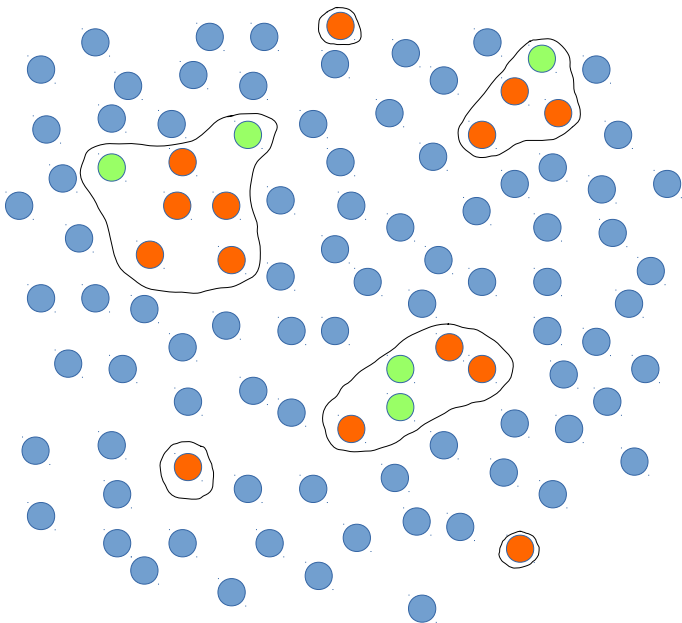
The number of flips is linear in the size of the initial error

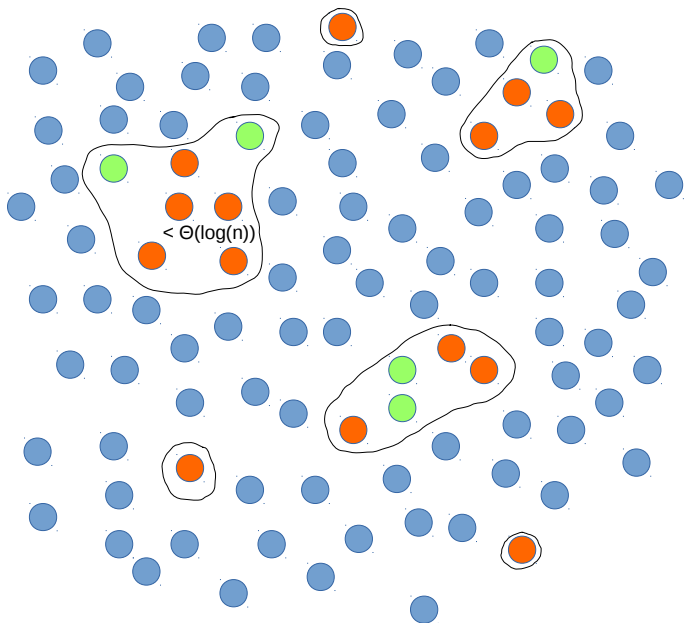












Theorem : what we proved

For a probability of error $p < 10^{-16}$:

$$\lim_{n \rightarrow +\infty} \mathbb{P}(\mathcal{A} \text{ corrects the error}) = 1$$

Theorem : what we proved

For a probability of error $p < 10^{-16}$:

$$\lim_{n \rightarrow +\infty} \mathbb{P}(\mathcal{A} \text{ corrects the error}) = 1$$

Ideas to improve this bound :

- ① Improve the bound in the percolation theorem :
What is the critical probability ?
- ② Restrict the proof to interesting clusters :
 - * The diameter of an interesting cluster is $\leq \Theta(\ln(\ln(n)))$
 - * The number of edges inside an interesting cluster is large
(ideas from bootstrap percolation)

Conclusion

The hypergraph product of an expander code :

- is an LDPC quantum code
- has a constant rate
- has a minimal distance : $d = \Theta(\sqrt{n})$

The decoding algorithm :

- has a capacity of correction : $\Theta(\sqrt{n})$
- corrects the error with high probability for the depolarizing channel

Futur work :

- improve our bound
- run simulations
- apply this result to fault tolerant quantum computation (Gottesman)

Thank you for your attention

A motivation : fault-tolerant quantum computation

Threshold Theorem [Ben-Or & Aharonov, '97]

We can simulate a quantum circuit with perfect gates by a circuit with noisy gates of size **quasi-linear**

Theorem : what we proved

For a probability of error $p < 10^{-16}$:

$$\lim_{n \rightarrow +\infty} \mathbb{P}(\mathcal{A} \text{ corrects the error}) = 1$$

What we hope to prove using [Gottesman, '13]

We can simulate a quantum circuit with perfect gates by a circuit with noisy gates of size **linear**

$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$

0

1

2

3

4

5

6

7

8

9

$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$

0

10

1

2

11

3

4

12

5

6

13

7

8

14

9

$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$

0 =

10

1 =

2 =

11

3 =

4 =

12

5 =

6 =

13

7 =

8 =

14

9 =

$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$

0 =

1 =

2 =

3 =

4 =

5 =

6 =

7 =

8 =

9 =

10 =

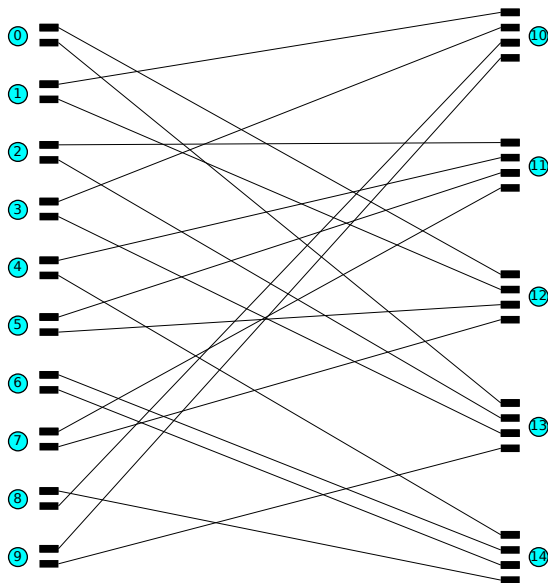
11 =

12 =

13 =

14 =

$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$



$$n = 10, m = 5, d_1 = 2, d_2 = \frac{n \times d_1}{m} = 4$$

