A Worst-Case Analysis of a Renormalisation Decoder for Kitaev's Toric Code

Wouter Rozendaal and Gilles Zémor

Institut de Mathématiques de Bordeaux

Journées C2, Najac - 20 October 2023

Wouter Rozendaal and Gilles Zémor. *Analysis of the Error-Correcting Radius of a Renormalisation Decoder for Kitaev's Toric Code*. 2023. arXiv: 2309.12165 [quant-ph]

Constructing Kitaev's Toric Code

Kitaev's code:

- Qubits indexed by edges of a toric tiling

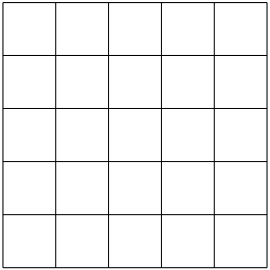

Figure: Tiling of a 2-dimensional torus

Constructing Kitaev's Toric Code

Kitaev's code:

- Qubits indexed by edges of a toric tiling
- CSS stabiliser code $(\mathbf{H}_X, \mathbf{H}_Z)$

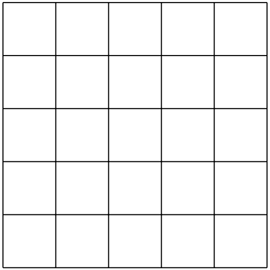

Figure: Tiling of a 2-dimensional torus

Constructing Kitaev's Toric Code

Kitaev's code:

- Qubits indexed by edges of a toric tiling
- CSS stabiliser code $(\mathbf{H}_X, \mathbf{H}_Z)$
  - Rows of $\mathbf{H}_X$

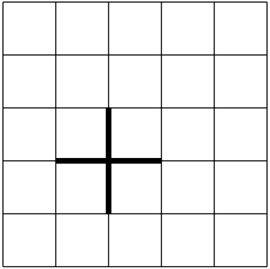

Figure: Elementary cocycle

Kitaev's code:

- Qubits indexed by edges of a toric tiling
- CSS stabiliser code $(\mathbf{H}_X, \mathbf{H}_Z)$
  - Rows of $\mathbf{H}_Z$

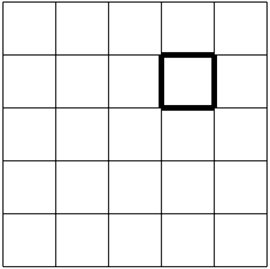

Figure: Elementary cycle

Kitaev's code:

- Qubits indexed by edges of a toric tiling
- CSS stabiliser code $(\mathbf{H}_X, \mathbf{H}_Z)$
  - Orthogonality condition: $\mathbf{H}_X \mathbf{H}_Z^{\intercal} = \mathbf{0}$

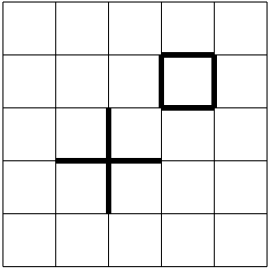

Figure: Elementary cocycles and cycles meet in an even number of edges

Decoding Kitaev's Toric Code

X or Z-error pattern $\to$ error vector $\mathbf{e} \in \mathbb{F}_2^n$

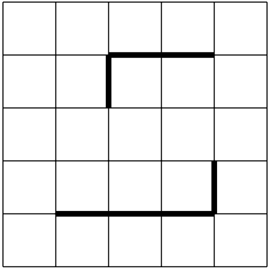

Figure: Error vector $\mathbf{e}$

Decoding Kitaev's Toric Code

X or Z-error pattern $\rightarrow$ error vector $\mathbf{e} \in \mathbb{F}_2^n$

- Input: syndrome measurement $\sigma(\mathbf{e}) := \mathbf{H}_X \mathbf{e}^\intercal$

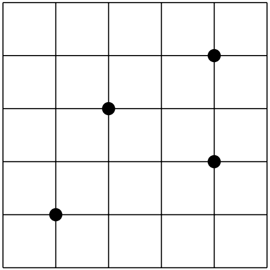

Figure: Syndrome $\sigma(\mathbf{e})$

X or Z-error pattern $\to$ error vector $\mathbf{e} \in \mathbb{F}_2^n$

- Input: syndrome measurement $\sigma(\mathbf{e}) := \mathbf{H}_X \mathbf{e}^\intercal$
- Output: $\hat{\mathbf{e}} \in \mathbb{F}_2^n$ such that $\sigma(\hat{\mathbf{e}}) = \sigma(\mathbf{e})$

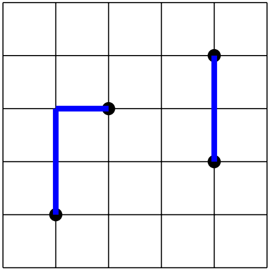

Figure: Output vector $\hat{\mathbf{e}}$ and its syndrome

Decoding Kitaev's Toric Code

X or Z-error pattern $\to$ error vector $\mathbf{e} \in \mathbb{F}_2^n$

- Input: syndrome measurement $\sigma(\mathbf{e}) := \mathbf{H}_X \mathbf{e}^\intercal$
- Output: $\hat{\mathbf{e}} \in \mathbb{F}_2^n$ such that $\sigma(\hat{\mathbf{e}}) = \sigma(\mathbf{e})$
- Successful decoding: $\mathbf{e} + \hat{\mathbf{e}}$ is in the row space of $\mathbf{H}_Z$

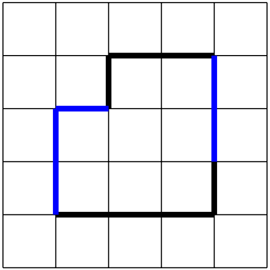

Figure: $\mathbf{e} + \hat{\mathbf{e}}$ is a trivial cycle

Decoding Kitaev's Toric Code

- Errors accumulate while a quantum algorithm is running
- Need for sub-linear decoding algorithms

Decoding Kitaev's Toric Code

- Errors accumulate while a quantum algorithm is running
- Need for sub-linear decoding algorithms

Renormalisation decoders!

Decoding Kitaev's Toric Code

- Errors accumulate while a quantum algorithm is running
- Need for sub-linear decoding algorithms

Renormalisation decoders!

- Renormalisation idea: Duclos-Cianci and Poulin (2010)
- Time-complexity in $O(n \log_2 n)$ and parallelisable to $O(\log_2 n)$
- High threshold values for bit-flip and depolarisation channels

Decoding Kitaev's Toric Code

- Errors accumulate while a quantum algorithm is running
- Need for sub-linear decoding algorithms

Renormalisation decoders!

- Renormalisation idea: Duclos-Cianci and Poulin (2010)
- Time-complexity in $O(n \log_2 n)$ and parallelisable to $O(\log_2 n)$
- High threshold values for bit-flip and depolarisation channels

Worst case behaviour?

Decoding Kitaev's Toric Code

- Errors accumulate while a quantum algorithm is running
- Need for sub-linear decoding algorithms

Renormalisation decoders!

- Renormalisation idea: Duclos-Cianci and Poulin (2010)
- Time-complexity in $O(n \log_2 n)$ and parallelisable to $O(\log_2 n)$
- High threshold values for bit-flip and depolarisation channels

Worst case behaviour?

What is the smallest weight of an error pattern for which decoding fails?

Example of Renormalisation Decoding

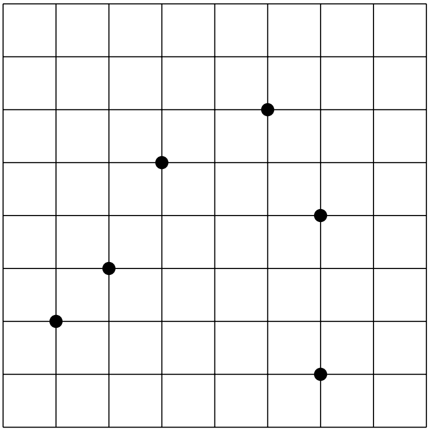

Figure: Decoding problem on a toric tiling.
Input: syndrome of an error vector **e**

Example of Renormalisation Decoding

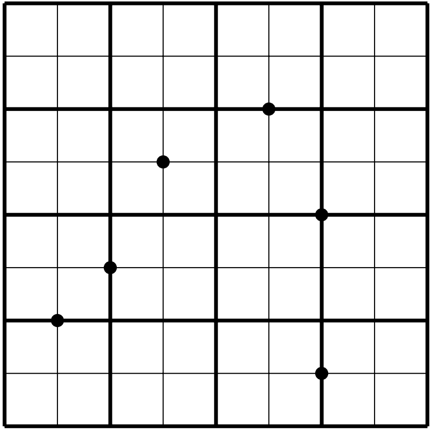

Figure: Reduction procedure:
Move the syndrome vertices to the next subtiling

Example of Renormalisation Decoding

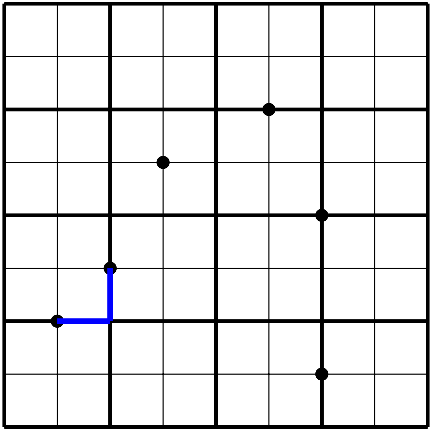

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

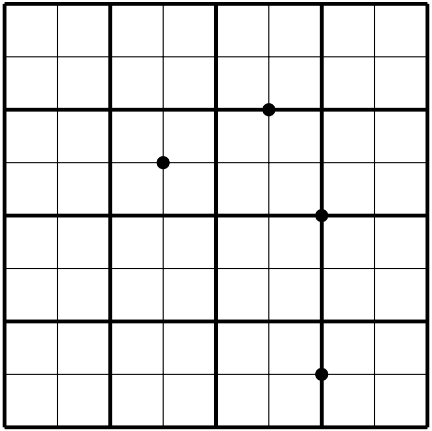

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

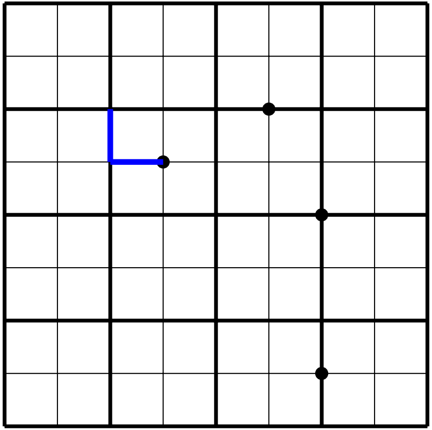

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

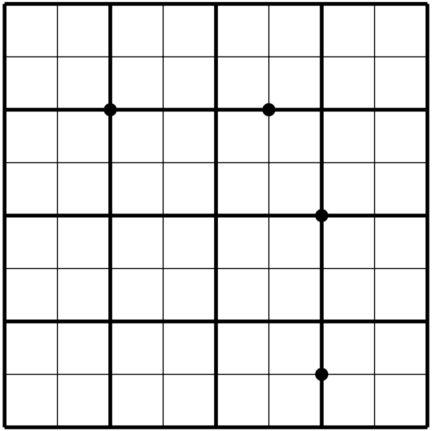

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

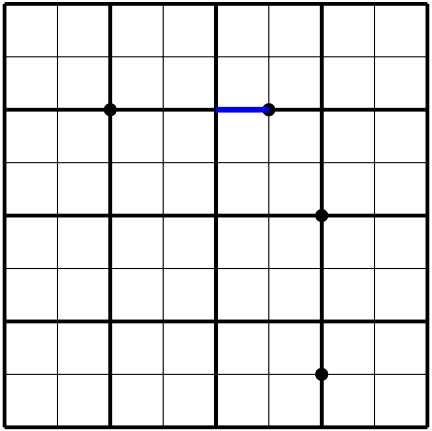

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

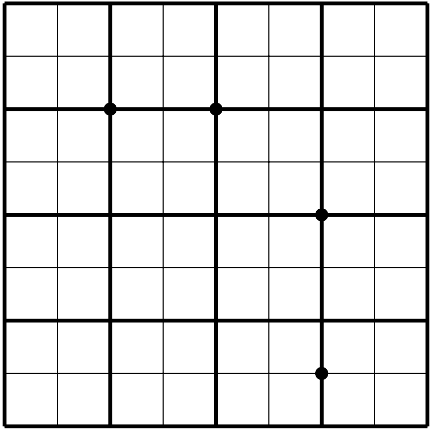

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

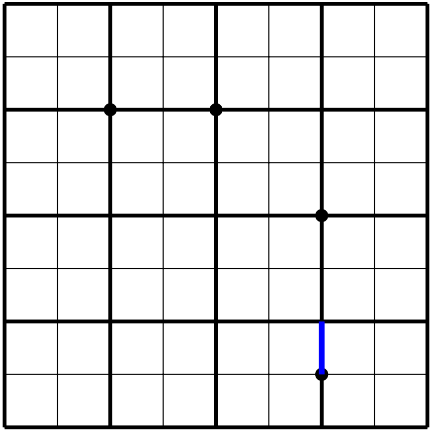

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

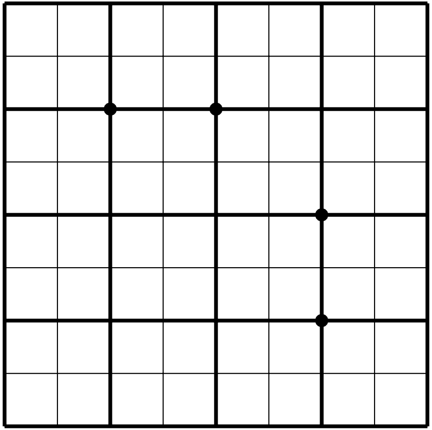

Figure: Locally shift remaining syndrome vertices

Example of Renormalisation Decoding

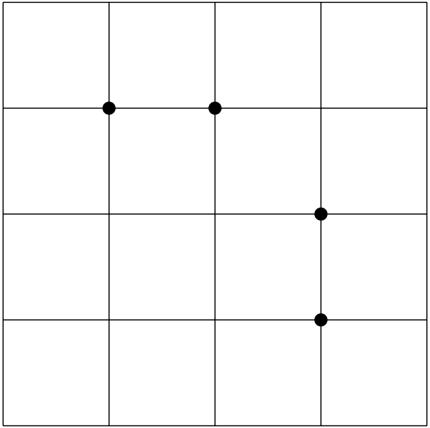

Figure: Decoding problem on the subtiling.
Input: syndrome of the vector $\mathbf{e} + \hat{\mathbf{e}}$

Example of Renormalisation Decoding

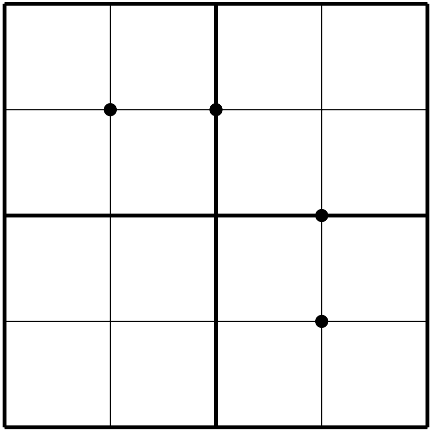

Figure: Reduction procedure:
Move the syndrome vertices to the next subtiling

Example of Renormalisation Decoding

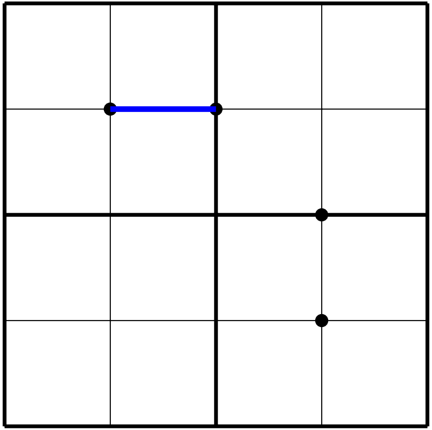

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

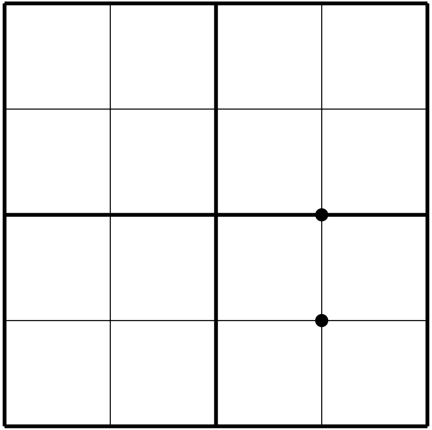

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

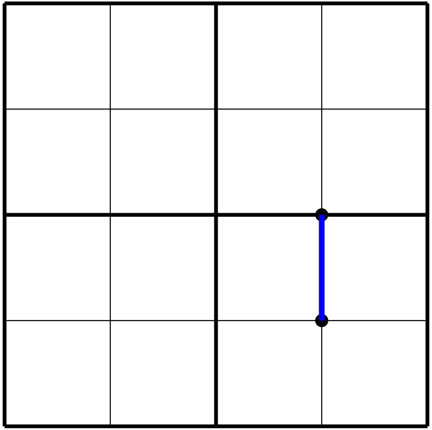

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

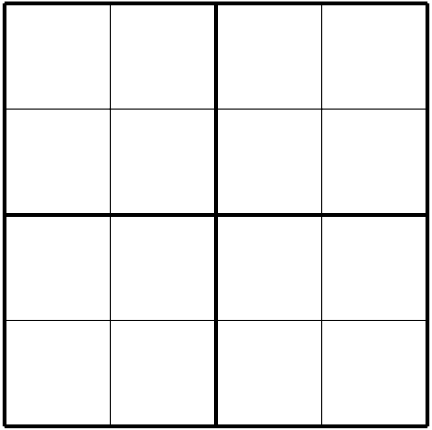

Figure: Locally pair-up syndrome vertices

Example of Renormalisation Decoding

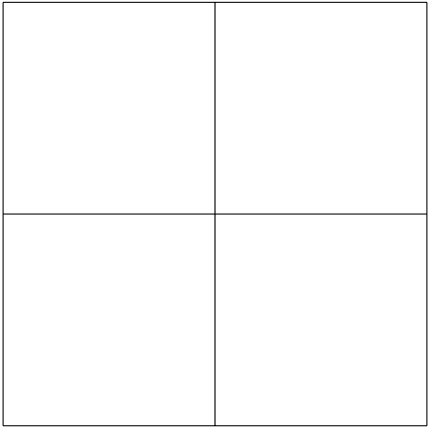

Figure: Decoding finishes.
All syndromes vertices have been paired-up

Example of Renormalisation Decoding

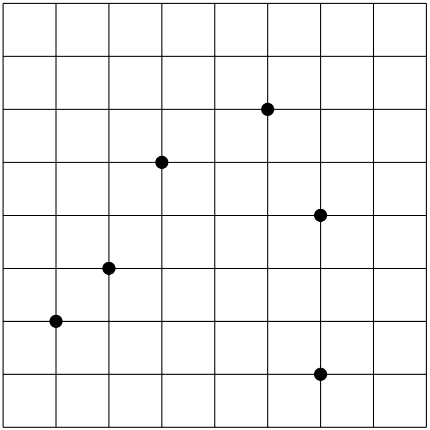

Figure: Syndrome of the error vector $\mathbf{e}$

Example of Renormalisation Decoding

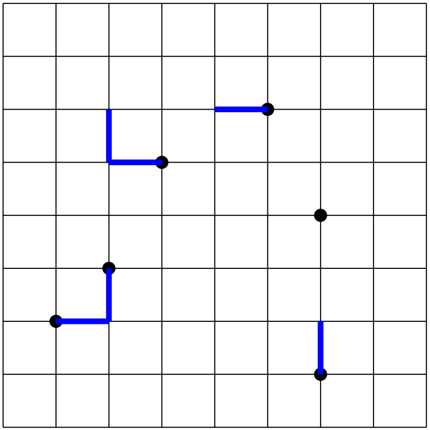

Figure: Output vector $\hat{\mathbf{e}}$ after the 1^{st} reduction step

Example of Renormalisation Decoding

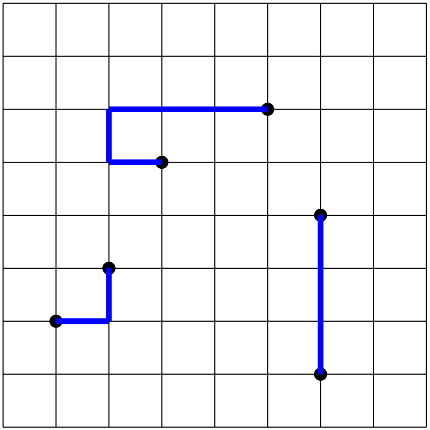

Figure: Output vector $\hat{\mathbf{e}}$ after the 2^{nd} reduction step

A Hard-Decision and Deterministic Reduction Procedure

Figure: Step 1: locally pair up diagonally opposed syndrome vertices in D cells

A Hard-Decision and Deterministic Reduction Procedure

Figure: Step 1: locally pair up diagonally opposed syndrome vertices in D cells

Figure: Step 2: locally pair up neighbouring syndrome vertices in B and C cells

A Hard-Decision and Deterministic Reduction Procedure

Figure: Step 1: locally pair up diagonally opposed syndrome vertices in D cells

Figure: Step 2: locally pair up neighbouring syndrome vertices in B and C cells

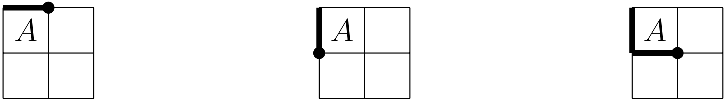

Figure: Step 3: shift remaining syndrome vertices in A cells to their top-left corner

Bounds on the Error-Correcting Radius

Upper Bound on the Error-Correcting Radius

There exist wrongly decoded error patterns whose weight scale like $d^{1/2}$.

Idea of the proof:

- Construct 1-dimensional fractal-like wrongly decoded error patterns

Bounds on the Error-Correcting Radius

Upper Bound on the Error-Correcting Radius

There exist wrongly decoded error patterns whose weight scale like $d^{1/2}$.

Idea of the proof:

- Construct 1-dimensional fractal-like wrongly decoded error patterns

Lower Bound on the Error-Correcting Radius

The renormalisation decoder corrects all errors of weight less than $\frac{5}{6}d^{\log_2(6/5)}$.

Idea of the proof:

- Evaluate the growth of $\mathsf{wt}_\mathsf{r}(\mathbf{e}_i) + P_i$ for increasing indexes

Take Away

Renormalisation decoders:

- Recursively reduce the decoding problem to smaller codes
- Good time-complexity and can be parallelised
- High threshold values over bit-flip and depolarisation channels

Take Away

Renormalisation decoders:

- Recursively reduce the decoding problem to smaller codes
- Good time-complexity and can be parallelised
- High threshold values over bit-flip and depolarisation channels

Our deterministic renormalisation decoder:

- $d^{0.26} \lesssim$ error-correcting radius $\lesssim d^{0.5}$

Take Away

Renormalisation decoders:

- Recursively reduce the decoding problem to smaller codes
- Good time-complexity and can be parallelised
- High threshold values over bit-flip and depolarisation channels

Our deterministic renormalisation decoder:

- $d^{0.26} \lesssim$ error-correcting radius $\lesssim d^{0.5}$

Any deterministic renormalisation decoder:

- Existence of fractal-like wrongly decoded error patterns of weight d^{α}
- α can be improved by increasing block size and using message-passing

Take Away

Renormalisation decoders:

- Recursively reduce the decoding problem to smaller codes
- Good time-complexity and can be parallelised
- High threshold values over bit-flip and depolarisation channels

Our deterministic renormalisation decoder:

- $d^{0.26} \lesssim$ error-correcting radius $\lesssim d^{0.5}$

Any deterministic renormalisation decoder:

- Existence of fractal-like wrongly decoded error patterns of weight d^{α}
- α can be improved by increasing block size and using message-passing

Thank you for your attention! Any questions?