Quantum Absorbing Sets

Tefjol Pllaha
Joint with Kirsten Morris & Christine Kelley

Department of Mathematics & Statistics
University of South Florida

email: tpllaha@usf.edu
website: pllahat.github.io

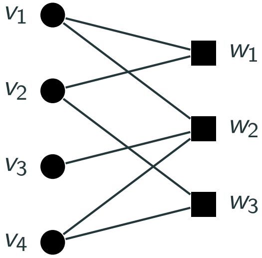
Talk is primarily based on

1. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Analysis of Syndrome-Based Iterative Decoder Failure of QLDPC Codes," 2023 12th International Symposium on Topics in Coding (ISTC), Brest, France, 2023.
2. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Effect of redundancy on syndrome-based decoding for QLDPC codes," 2025 13th International Symposium on Topics in Coding (ISTC), 1-5.
3. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Analysis of Syndrome-Based Iterative Decoder Failure of QLDPC Codes," Advances in Mathematics of Communications, 2026, 23: 234-254. doi: 10.3934/amc.2026023.

A linear code $\mathcal{C}$ with parity check matrix H may be represented by a bipartite Tanner graph[1] $G = (V, W; E)$.

- V (“*variable nodes*”), representing codeword coordinates
- W (“*check nodes*”), representing check equations.
- $(v_i, w_j) \in E$ iff $h_{j,i} = 1$.

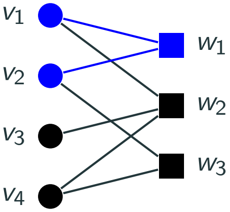
$$H = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$



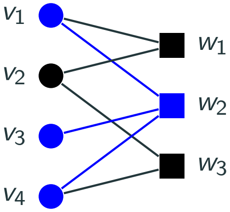
[1] Tanner, “A recursive approach to low complexity codes,” 1981

Tanner Graph

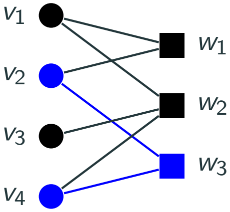
$$\mathbf{x} \in \mathcal{C} \text{ if and only if } H\mathbf{x}^T = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \mathbf{0}$$



$w_1 : x_1 + x_2 = 0$



$w_2 : x_1 + x_3 + x_4 = 0$



$w_3 : x_2 + x_4 = 0$

LDPC Codes

A Low Density Parity-Check (LDPC) Code is a linear code with sparse parity-check matrix.

LDPC Codes

A Low Density Parity-Check (LDPC) Code is a linear code with sparse parity-check matrix.

- Low complexity iterative decoding.
- There exists asymptotically good codes.

Stabilizer Codes

$$
\begin{array}{ccc}
\mathbb{C}^2 & \leftrightarrow & \mathbb{F}_2^2 \\
\hline
I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \leftrightarrow & (0, 0) \\
X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \leftrightarrow & (1, 0) \\
Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} & \leftrightarrow & (0, 1) \\
Y = iXZ & \leftrightarrow & (1, 1) \\
e = i^c X^a Z^b & \leftrightarrow & (a, b)
\end{array}
$$

Stabilizer Codes

$$
\begin{array}{rcl}
\mathbb{C}^2 & \leftrightarrow & \mathbb{F}_2^2 \\
\hline
I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \leftrightarrow & (0,0) \\
X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} & \leftrightarrow & (1,0) \\
Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} & \leftrightarrow & (0,1) \\
Y = iXZ & \leftrightarrow & (1,1) \\
e = i^c X^a Z^b & \leftrightarrow & (a,b)
\end{array}
$$

$$
\begin{array}{c|c}
(\mathbb{C}^2)^{\otimes n} \cong \mathbb{C}^{2^n} & e_1 \otimes \cdots \otimes e_n = i^{c_1} X^{a_1} Z^{b_1} \otimes \cdots \otimes i^{c_n} X^{a_n} Z^{b_n} \\
\updownarrow & \updownarrow \\
\mathbb{F}_2^{2n} & (e_X, e_Z) \equiv (a_1, \ldots, a_n, b_1, \ldots, b_n)
\end{array}
$$

Stabilizer Codes

- Two errors $e \equiv (e_X, e_Z), f \equiv (f_X, f_Z)$ commmute if and only if

$$e \odot f \equiv e_X f_Z^T + e_Z f_X^T = 0$$

Stabilizer Codes

- Two errors $e \equiv (e_X, e_Z), f \equiv (f_X, f_Z)$ commmute if and only if

$$e \odot f \equiv e_X f_Z^T + e_Z f_X^T = 0$$

- An $[[n, n-k]]$ stabilizer code is defined by a $k \times 2n$ matrix $H = (H_X \mid H_Z)$ such that

$$H \odot H := H_X H_Z^T + H_Z H_X^T = 0.$$

Stabilizer Codes

- Two errors $e \equiv (e_X, e_Z), f \equiv (f_X, f_Z)$ commmute if and only if

$$e \odot f \equiv e_X f_Z^T + e_Z f_X^T = 0$$

- An $[[n, n-k]]$ stabilizer code is defined by a $k \times 2n$ matrix $H = (H_X \mid H_Z)$ such that

$$H \odot H := H_X H_Z^T + H_Z H_X^T = 0.$$

- Recall:

$$\begin{array}{c|c} (\mathbb{C}^2)^{\otimes n} \cong \mathbb{C}^{2^n} & e_1 \otimes \cdots \otimes e_n = i^{c_1} X^{a_1} Z^{b_1} \otimes \cdots \otimes i^{c_n} X^{a_n} Z^{b_n} \\ \updownarrow & \updownarrow \\ \mathbb{F}_2^{2n} & (e_X, e_Z) \equiv (a_1, \ldots, a_n, b_1, \ldots, b_n) \end{array}$$

Stabilizer Codes

- Two errors $e \equiv (e_X, e_Z), f \equiv (f_X, f_Z)$ commmute if and only if

$$e \odot f \equiv e_X f_Z^T + e_Z f_X^T = 0$$

- An $[[n, n-k]]$ stabilizer code is defined by a $k \times 2n$ matrix $H = (H_X \mid H_Z)$ such that

$$H \odot H := H_X H_Z^T + H_Z H_X^T = 0.$$

- Recall:

$$\begin{array}{c|c} (\mathbb{C}^2)^{\otimes n} \cong \mathbb{C}^{2^n} & e_1 \otimes \cdots \otimes e_n = i^{c_1} X^{a_1} Z^{b_1} \otimes \cdots \otimes i^{c_n} X^{a_n} Z^{b_n} \\ \updownarrow & \updownarrow \\ \mathbb{F}_2^{2n} & (e_X, e_Z) \equiv (a_1, \ldots, a_n, b_1, \ldots, b_n) \end{array}$$

- The codewords of the $[[n, n-k]]$ stabilizer code are eigenvectors of

$$e_1 \otimes \cdots \otimes e_n = i^{c_1} X^{a_1} Z^{b_1} \otimes \cdots \otimes i^{c_n} X^{a_n} Z^{b_n}$$

Calderbank-Shor-Steane (CSS) Codes

A CSS Code is defined by a pair of classical linear codes $C_X, C_Z \subset \mathbb{F}_q^n$ such that $C_X^\perp \subseteq C_Z$.

Two respective parity check matrices H_X and H_Z satisfy $H_Z H_X^T = 0$ and thus the matrix

$$H = \left(\begin{array}{c|c} H_X & 0 \\ 0 & H_Z \end{array}\right)$$

satisfies

$$\begin{pmatrix} H_X \\ 0 \end{pmatrix}\begin{pmatrix} 0 & H_Z^T \end{pmatrix} + \begin{pmatrix} 0 \\ H_Z \end{pmatrix}\begin{pmatrix} H_X^T & 0 \end{pmatrix} = \begin{pmatrix} 0 & H_X H_Z^T \\ H_Z H_X^T & 0 \end{pmatrix} = 0$$

Quantum LDPC Codes: Highlights

- Why quantum LDPC Codes:
  - 2003 (Kitaev): surface code
  - 2013 (Gottesman): quantum LDPC codes achieve fault tolerance with constant overhead

Quantum LDPC Codes: Highlights

- Why quantum LDPC Codes:
  - 2003 (Kitaev): surface code
  - 2013 (Gottesman): quantum LDPC codes achieve fault tolerance with constant overhead
- Good quantum LDPC Codes:
  - 2009 (Tillich & Zémor): $r = c > 0, d \sim \sqrt{n}$
  - 2020-2021: A series of works that broke the $\sqrt{n}$ barrier
  - Nov. 2021: (Panteleev & Kalachev): Asymptotically good codes exist

Syndrome Decoding for CSS Codes

For a CSS code, the syndrome of an error $(e_\mathsf{X}, e_\mathsf{Z})$ is computed as

$$\left(\begin{array}{c|c} H_\mathsf{X} & 0 \\ 0 & H_\mathsf{Z} \end{array}\right) \odot (e_\mathsf{X}, e_\mathsf{Z}) = (H_\mathsf{Z} e_\mathsf{X}^T, H_\mathsf{X} e_\mathsf{Z}^T)$$
$$\equiv (\sigma_\mathsf{X}, \sigma_\mathsf{Z})$$

Thus, C_Z is used to decode X-errors and C_X is used to correct Z-errors.

Syndrome Decoding for CSS Codes

For a CSS code, the syndrome of an error (e_X, e_Z) is computed as

$$\left(\begin{array}{c|c} H_X & 0 \\ 0 & H_Z \end{array}\right) \odot (e_X, e_Z) = (H_Z e_X^T, H_X e_Z^T)$$
$$\equiv (\sigma_X, \sigma_Z)$$

Thus, C_Z is used to decode X-errors and C_X is used to correct Z-errors.

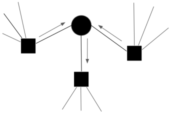
Goal of a syndrome-based decoder: estimate the error pattern $\hat{e}$ whose syndrome $\hat{\sigma}$ matches with the initial input syndrome σ.

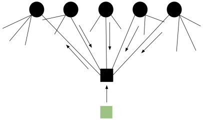
Syndrome-based Iterative Decoder

Input to the decoder: Measured syndrome σ

Process:

- Messages are passed along edges of Tanner graph.
- Nodes wait until they receive messages from all but one neighbor.
- Compute new message to send to remaining neighbor.



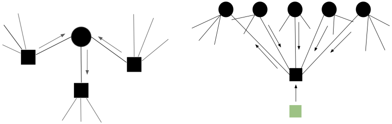


Syndrome-based Iterative Decoder

Input to the decoder: Measured syndrome σ

Process:

- Messages are passed along edges of Tanner graph.
- Nodes wait until they receive messages from all but one neighbor.
- Compute new message to send to remaining neighbor.



Goal: Estimate the error pattern $\hat{e}$ whose syndrome $\hat{\sigma}$ matches with the initial input syndrome σ.

Syndrome Decoding

Given an error e and corresponding syndrome σ_e, the possible (quantum) decoding outcomes are

Successful Decoding

- $\sigma_{\hat{e}} = \sigma_e$ and $\hat{e} = e$
- $\sigma_{\hat{e}} = \sigma_e$ and $e + \hat{e} \in \text{row}(H)$ (degenerate error)

Syndrome Decoding

Given an error e and corresponding syndrome σ_e, the possible (quantum) decoding outcomes are

Successful Decoding

- $\sigma_{\hat{e}} = \sigma_e$ and $\hat{e} = e$
- $\sigma_{\hat{e}} = \sigma_e$ and $e + \hat{e} \in \mathrm{row}(H)$ (degenerate error)

Decoding Failure

- $\sigma_{\hat{e}} \neq \sigma_e$
- $\sigma_{\hat{e}} = \sigma_e$ and $e + \hat{e} \notin \mathrm{row}(H)$ (logical error)

Syndrome Decoding

Given an error e and corresponding syndrome σ_e, the possible (quantum) decoding outcomes are

Successful Decoding

- $\sigma_{\hat{e}} = \sigma_e$ and $\hat{e} = e$
- $\sigma_{\hat{e}} = \sigma_e$ and $e + \hat{e} \in \mathrm{row}(H)$ (degenerate error)

Decoding Failure

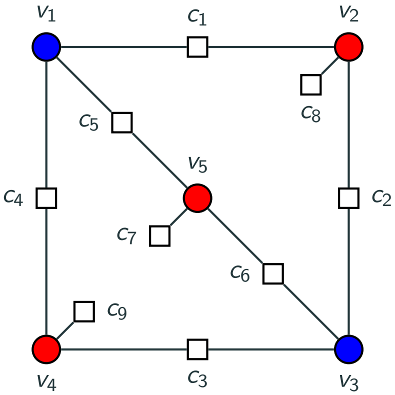
- $\sigma_{\hat{e}} \neq \sigma_e$
- $\sigma_{\hat{e}} = \sigma_e$ and $e + \hat{e} \notin \mathrm{row}(H)$ (logical error)

For this work we use the Gallager-B syndrome-based iterative decoder.

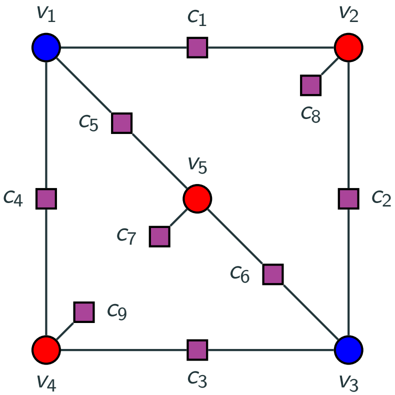
Gallager-B Syndrome-based Iterative Decoder

0. Variable nodes are initialized at 0, check nodes are initialized with the input syndrome.
1. The outgoing check node message over an edge is computed as the XOR of extrinsic variable node messages and syndrome input value.
   - Estimated error $\hat{e}$ is determined to be the majority among all incoming check node values at each variable node.
2. The outgoing variable node message is the majority value among incoming extrinsic check node messages.
   - The estimated syndrome is computed as the XOR of all incoming variable node messages.
3. Decoder halts if $\hat{\sigma} = \sigma$ or if ℓ is larger than a threshold.

Toy Example: I

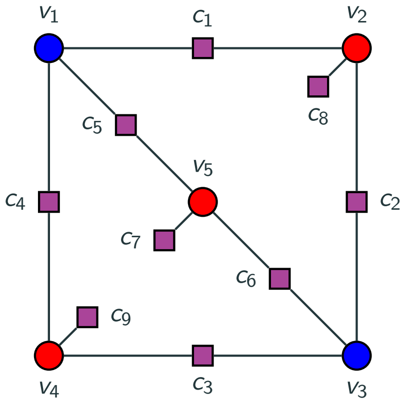


Toy Example: I



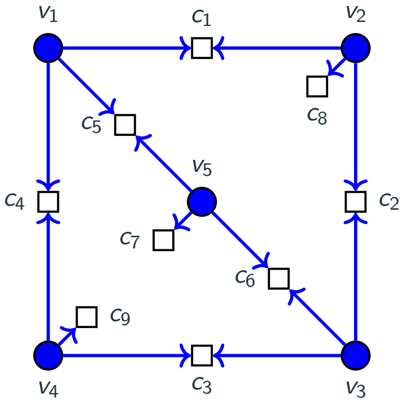
$$\sigma = (1, 1, 1, 1, 1, 1, 1, 1, 1)$$

Toy Example: I

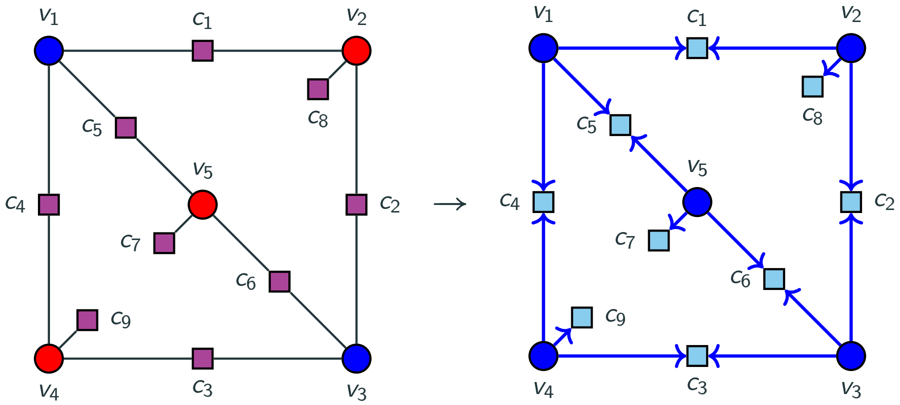


$\sigma = (1, 1, 1, 1, 1, 1, 1, 1, 1)$

$\longrightarrow$

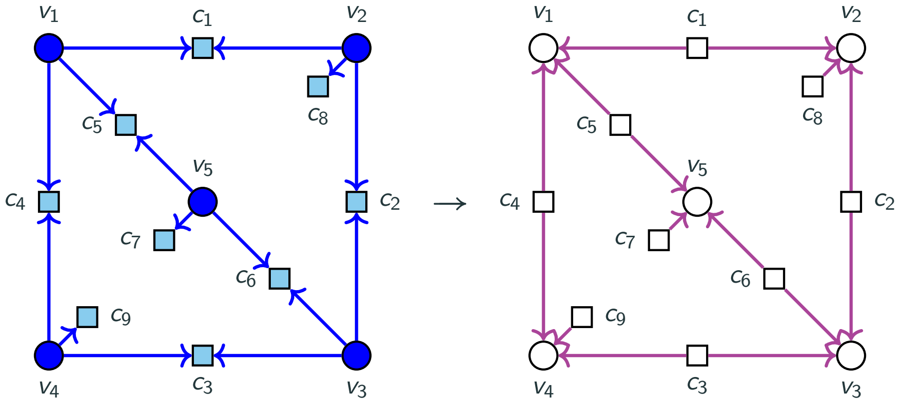


Toy Example: I



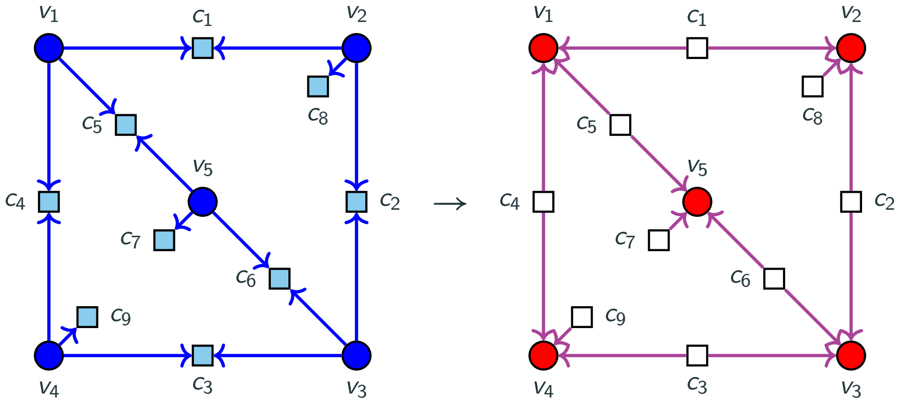
$\sigma = (1,1,1,1,1,1,1,1,1)$ $\hat{\sigma}^{(1)} = (0,0,0,0,0,0,0,0,0)$

Toy Example:



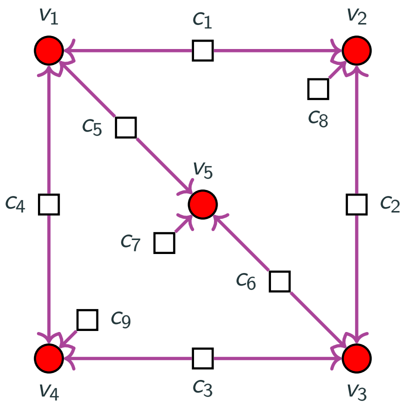
$$\hat{\sigma}^{(1)} = (0, 0, 0, 0, 0, 0, 0, 0, 0)$$

Toy Example: I

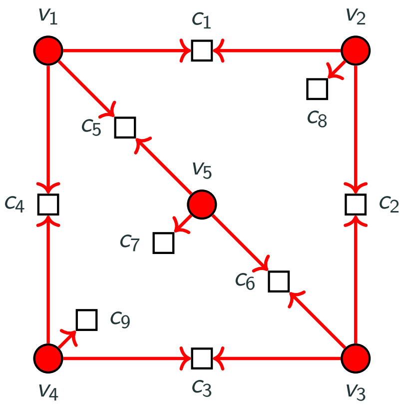


$$\hat{\sigma}^{(1)} = (0,0,0,0,0,0,0,0,0) \qquad \hat{e}^{(1)} = (1,1,1,1,1)$$

Toy Example: I

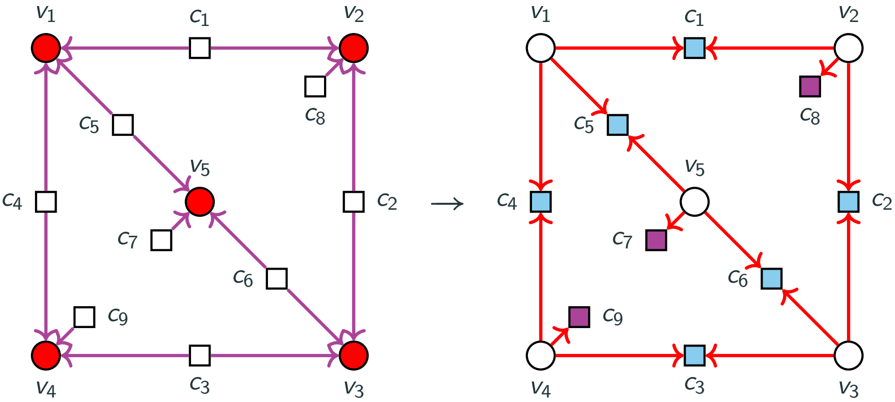


$\longrightarrow$



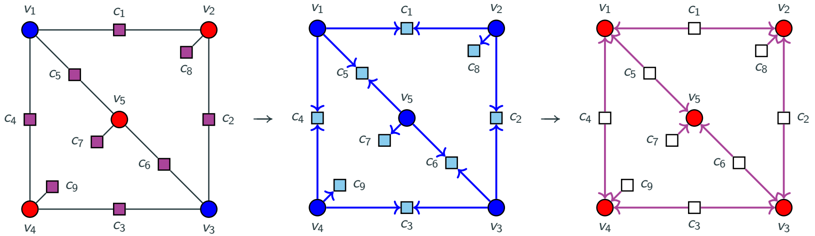
$$\hat{e}^{(1)} = (1, 1, 1, 1, 1)$$

Toy Example: **I**

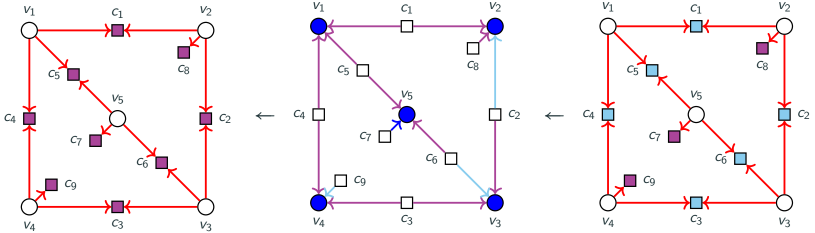


$$\hat{e}^{(1)} = (1, 1, 1, 1, 1) \qquad \hat{\sigma}^{(2)} = (0, 0, 0, 0, 0, 0, 1, 1, 1)$$

Toy Example:



$\sigma = (1,1,1,1,1,1,1,1,1)$ → $\hat{\sigma}^{(1)} = (0,0,0,0,0,0,0,0,0)$ → $\hat{e}^{(1)} = (1,1,1,1,1)$

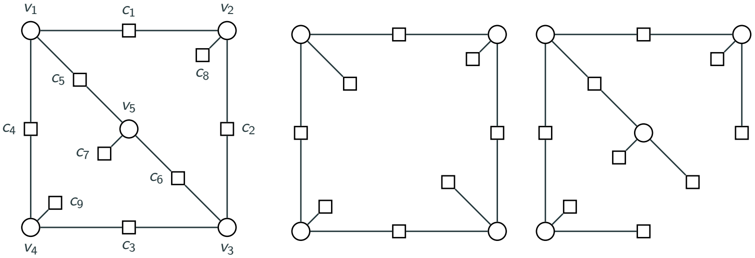


Back where we started! ← $e^{(2)} = (0,0,0,0,0)$ ← $\hat{\sigma}^{(2)} = (0,0,0,0,0,0,1,1,1)$

Trapping Sets

- A check node w_j, for $1 \le j \le k$, is eventually correct if there exists $L \in \mathbb{Z}_{\ge 0}$ such that $\hat{\sigma}_j^{(\ell)} = \sigma_j$ for all $\ell \ge L$.
- A variable node v_i, for $1 \le i \le n$, is said to eventually converge if there exists $L \in \mathbb{Z}_{\ge 0}$ such that $\hat{e}_i^{(\ell)} = \hat{e}_i^{(\ell-1)}$ for all $\ell \ge L$.
- A trapping set for a syndrome-based iterative decoder is a non-empty set of variable nodes $\mathcal{T}$ in a Tanner graph G such that there is a subset of variable nodes $\mathcal{F} \subseteq \mathcal{T}$ that when initially in error result in some subset of check nodes of $\mathcal{N}(\mathcal{T})$ not eventually correct or some variable nodes of $\mathcal{T}$ not eventually converging.
- The graph $\mathcal{T} \cup \mathcal{N}(\mathcal{T})$ is the trapping set graph with respect to $\mathcal{T}$.
- A subset of variable nodes $\mathcal{F}$ that when initially in error result in a trapping set $\mathcal{T}$ is called a failure-inducing set for $\mathcal{T}$.

Nodes in Error	Input Syndrome	Estimated Syndrome	Estimated Error
$\{v_1, v_2, v_3, v_4\}$	$(0,0,0,0,1,1,0,1,1)$	$(0,0,0,0,0,0,1,0,0)$	$\{v_5\}$
$\{v_1, v_2, v_3, v_5\}$	$(0,0,1,1,0,0,1,1,0)$	$(0,0,0,0,0,0,0,0,1)$	$\{v_4\}$
$\{v_1, v_3, v_4, v_5\}$	$(1,1,0,0,0,0,1,0,1)$	$(0,0,0,0,0,0,0,1,0)$	$\{v_2\}$
$\{v_1, v_2, v_4, v_5\}$	$(0,1,1,0,0,1,1,1,1)$	$(0,0,0,0,0,0,0,0,0)$	$\{v_2, v_3, v_4, v_5\}$
		$(0,0,0,0,0,0,0,0,0)$	$\{\}$
		$(1,1,1,1,1,1,0,0,0)$	$\{v_2, v_3, v_4, v_5\}$
		$(1,1,1,1,1,1,0,0,0)$	$\{v_1, v_3\}$
$\{v_2, v_3, v_4, v_5\}$	$(1,0,0,1,1,0,1,1,1)$	$(1,1,1,1,1,1,0,0,0)$	$\{v_1, v_3\}$
		$(1,0,0,1,1,0,1,1,1)$	$\{v_1, v_2, v_4, v_5\}$
		$(0,0,0,0,0,0,0,0,0)$	$\{\}$
		$(0,0,0,0,0,0,0,0,0)$	$\{v_1, v_2, v_4, v_5\}$



Related Structures: Absorbing Sets

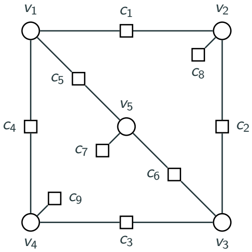
An (a, b) absorbing set $\mathcal{A}$ in a Tanner graph G is a subset of $|\mathcal{A}| = a$ variable nodes such that in the graph $G_{\mathcal{A}}$ induced by $\mathcal{A} \cup \mathcal{N}(\mathcal{A})$

- there are b odd degree check nodes
- every $v \in \mathcal{A}$ has more even degree than odd degree neighbors in $G_{\mathcal{A}}$.

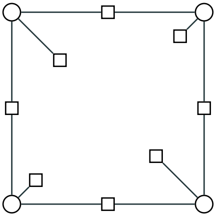
Related Structures: Absorbing Sets

An (a, b) absorbing set $\mathcal{A}$ in a Tanner graph G is a subset of $|\mathcal{A}| = a$ variable nodes such that in the graph $G_{\mathcal{A}}$ induced by $\mathcal{A} \cup \mathcal{N}(\mathcal{A})$

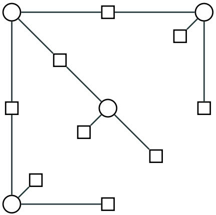
- there are b odd degree check nodes
- every $v \in \mathcal{A}$ has more even degree than odd degree neighbors in $G_{\mathcal{A}}$.



(5,3) absorbing set



(4,4) absorbing set

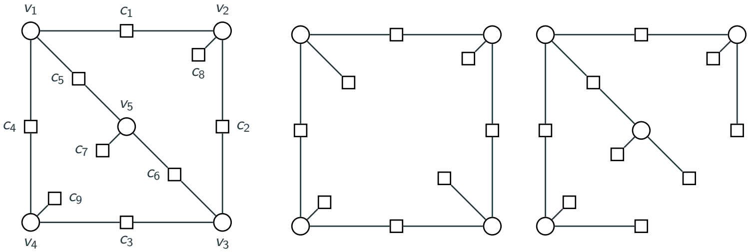


not absorbing set

Related Structures: Absorbing Sets

An (a,b) absorbing set $\mathcal{A}$ in a Tanner graph G is a subset of $|\mathcal{A}| = a$ variable nodes such that in the graph $G_{\mathcal{A}}$ induced by $\mathcal{A} \cup \mathcal{N}(\mathcal{A})$

- there are b odd degree check nodes
- every $v \in \mathcal{A}$ has more even degree than odd degree neighbors in $G_{\mathcal{A}}$.



(5,3) absorbing set — (4,4) absorbing set — not absorbing set

Question: What relationship is there, if any, between absorbing sets, trapping sets, and failure-inducing sets?

Related Structures: Absorbing Sets

Theorem (Morris, P., Kelley, ’23)

Let $\mathcal{A}$ be an (a,b)-absorbing set with $b \geq 1$. Then $\mathcal{A}$ itself is a failure-inducing set and therefore $\mathcal{A}$ is a trapping set.

Proof

- At least one $\sigma_i = 1$.
- The next estimated syndrome $\hat{\sigma} = \overrightarrow{0}$ because
  - Each variable node has strictly more even degree than odd degree check nodes.
  - Even degree nodes send zeros, odd degrees nodes send ones.
  - Majority rules and the estimated error is $\hat{e} = \overrightarrow{0}$.
  - Outgoing variable messages are always 0 because the extrinsic check nodes are at most evenly tied.

Related Structures: Absorbing Sets

Theorem (Morris, P., Kelley, '23)

An (a,0) absorbing set $\mathcal{A}$ is a trapping set if and only if $\overrightarrow{1}$ is not a stabilizer.

Proof

- Input syndrome $\sigma = \overrightarrow{0}$ because all CNs have even degree and all VNs are sending 1s.
- syndrome is matched in the first iteration and estimated error is $\hat{e} = \overrightarrow{0}$.
- $e + \hat{e} = \overrightarrow{1}$.

Related Structures: Absorbing Sets

Theorem (Morris, P., Kelley, '23)

An (a,0) absorbing set $\mathcal{A}$ is a trapping set if and only if $\overrightarrow{1}$ is not a stabilizer.

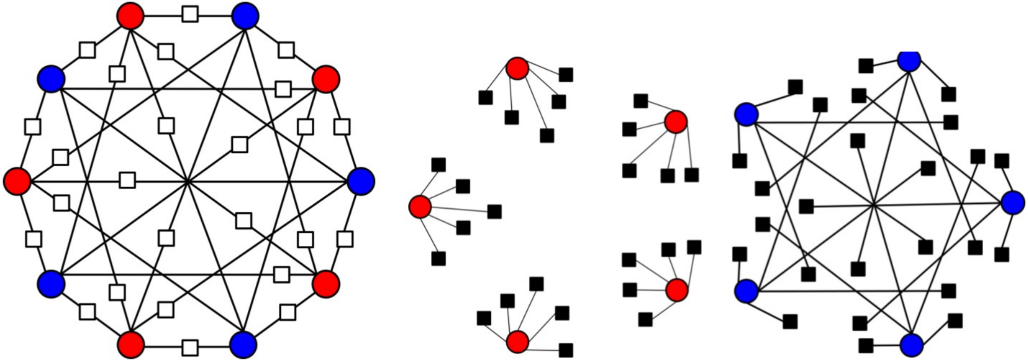
Proof

- Input syndrome $\sigma = \overrightarrow{0}$ because all CNs have even degree and all VNs are sending 1s.
- syndrome is matched in the first iteration and estimated error is $\hat{e} = \overrightarrow{0}$.
- $e + \hat{e} = \overrightarrow{1}$.

Takeaway

Variable nodes indexed by nonstabilizers with input syndrome $\overrightarrow{0}$ form failure-inducing sets of an $(a, 0)$ absorbing set.

Related $(a, 0)$ absorbing sets: Symmetric Stabilizers[2]



(10,0) absorbing set

Symmetric Split

[2] Raveendran and Vasić, "Trapping Sets of Quantum LDPC Codes," 2021

Beyond symmetric stabilizers

Theorem (Morris, P., Kelley, '24)

Let $\mathcal{A}$ be an $(a, 0)$ absorbing set such that $\mathcal{A} = \mathcal{B} \cup \mathcal{C}$, with $\mathcal{B}, \mathcal{C}$ absorbing sets. Then both $\mathcal{B}$ and $\mathcal{C}$ are failure inducing sets.

Beyond symmetric stabilizers

Theorem (Morris, P., Kelley, '24)

Let $\mathcal{A}$ be an $(a, 0)$ absorbing set such that $\mathcal{A} = \mathcal{B} \cup \mathcal{C}$, with $\mathcal{B}, \mathcal{C}$ absorbing sets. Then both $\mathcal{B}$ and $\mathcal{C}$ are failure inducing sets.

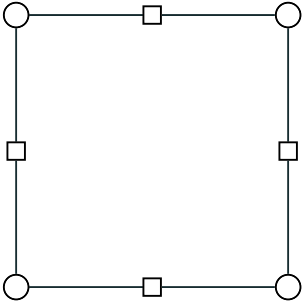
Theorem (Morris, P., Kelley, '25)

Let $\mathcal{B}$ and $\mathcal{C}$ be (b, d) and (c, e) absorbing sets respectively such that

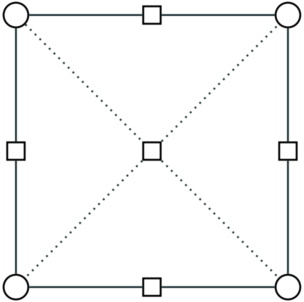
- $\mathcal{B} \cap \mathcal{C} = \varnothing$
- $\mathcal{N}(\mathcal{B}) \cap \mathcal{N}(\mathcal{C}) \neq \varnothing$
- $\forall c \in \mathcal{N}(\mathcal{B}) \cap \mathcal{N}(\mathcal{C}), \mathcal{N}(c) \subset \mathcal{B} \cup \mathcal{C}$
- $\forall c \in \mathcal{N}(\mathcal{B}),$ such that $\mathcal{N}(c) \subset \mathcal{B}$, $\deg(c) \in 2\mathbb{Z}$
- $\forall c \in \mathcal{N}(\mathcal{C}),$ such that $\mathcal{N}(c) \subset \mathcal{C}$, $\deg(c) \in 2\mathbb{Z}$

Then, $\mathcal{B}$ and $\mathcal{C}$ are failure inducing sets.

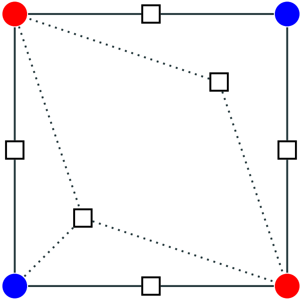
Toy Example: II



(4,0) absorbing set



1 added check



2 added checks

Case Study: The Steane Code

The Steane Code is the CSS code built from the $[7, 4, 3]$ Hamming code.

Standard Parity Check Matrix H for the $[7, 4, 3]$ Hamming code:

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Case Study: The Steane Code

The Steane Code is the CSS code built from the $[7, 4, 3]$ Hamming code.

Standard Parity Check Matrix H for the $[7, 4, 3]$ Hamming code:

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

Adding Redundancy: concatenating nonzero elements of rowspace(H) to H.

Case Study: The Steane Code

	Redundant Checks	Syndrome Mismatch Decoding Failures	Absorbing Sets
Representation 0	0	48	22
Representation 1	1	112	16
Representations 2-4	1	64	27
Representations 5-10	2	48	24
Representations 11-14	3	0	20
Representation 15	4	0	16

Quantum Hamming Codes

Theorem (Morris, P., Kelley, '25)

The Gallager B syndrome based decoder matches the syndrome of every error pattern, returning an estimated error pattern $\hat{e}$, within two iterations by using the Tanner graph corresponding to $H_{\max}$ for the Hamming codes.

Theorem (Morris, P., Kelley, '25)

The Gallager B syndrome based decoder matches the syndrome of every error pattern, returning an estimated error pattern $\hat{e}$, within two iterations by using the Tanner graph corresponding to H_{max} for the Hamming codes.

- Complexity Tradeoff: Adding redundancy versus returning an estimated error pattern within two iterations.

Quantum Hamming Codes

Theorem (Morris, P., Kelley, '25)

The Gallager B syndrome based decoder matches the syndrome of every error pattern, returning an estimated error pattern $\hat{e}$, within two iterations by using the Tanner graph corresponding to $H_{\max}$ for the Hamming codes.

- Complexity Tradeoff: Adding redundancy versus returning an estimated error pattern within two iterations.
- Problem: What is the minimum added redundancy that guarantees a matched syndrome within a small number of iterations?

Sufficient conditions

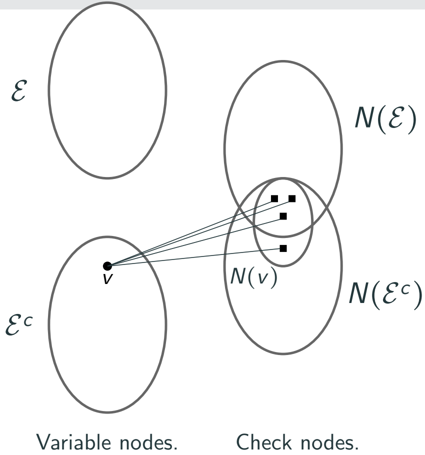
Theorem (Morris, P., Kelley, '25)

The Gallager B syndrome based decoder matches the syndrome of an error pattern within two iterations if

Sufficient conditions

Theorem (Morris, P., Kelley, '25)

The Gallager B syndrome based decoder matches the syndrome of an error pattern within two iterations if



Performance

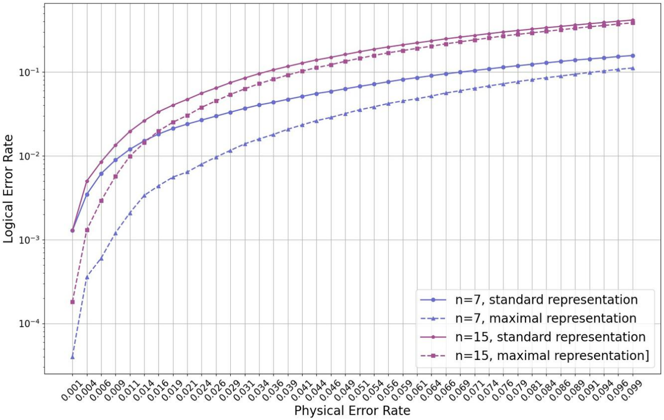


Figure 1

Open Problems

- Analysis of **families** of QLDPC codes.

Open Problems

- Analysis of **families** of QLDPC codes.
- Construct QLDPC codes with less symmetry in general and fewer symmetric stabilizers in particular.

Open Problems

- Analysis of **families** of QLDPC codes.
- Construct QLDPC codes with less symmetry in general and fewer symmetric stabilizers in particular.
- Determine the smallest amount of redundancy needed for significant decoding gains.

Thank You!

References:

1. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Analysis of Syndrome-Based Iterative Decoder Failure of QLDPC Codes," 2023 12th International Symposium on Topics in Coding (ISTC), Brest, France, 2023.
2. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Effect of redundancy on syndrome-based decoding for QLDPC codes," 2025 13th International Symposium on Topics in Coding (ISTC), 1-5.
3. K. D. Morris, **T. Pllaha** and C. A. Kelley, "Analysis of Syndrome-Based Iterative Decoder Failure of QLDPC Codes," Advances in Mathematics of Communications, 2026, 23: 234-254. doi: 10.3934/amc.2026023.