Breaking the Orthogonality Barrier: Large-Girth Regular LDPC Structures with Favorable Distance for Quantum Error Correction

Kenta Kasai

Institute of Science Tokyo

ITA 2026, Bahia Resort Hotel, San Diego, Feb 8–13, 2026
Based on arXiv:2601.08824

Motivation: Classical vs Quantum LDPC

Classical: sparse+random+large girth $\Rightarrow$ strong BP, large distance.

Quantum: CSS constraint $H_X H_Z' = 0$ couples H_X, H_Z.

Issue: naive orthogonality $\Rightarrow$ short cycles, small distance.

Goal: keep classical structure *without* the orthogonality distance penalty.

Problem:

Parent construction

$$\hat{H}_X = \begin{bmatrix} H_X \\ \tilde{H}_X \end{bmatrix}, \qquad \hat{H}_Z = \begin{bmatrix} H_Z \\ \tilde{H}_Z \end{bmatrix}, \qquad \hat{H}_X(\hat{H}_Z)' = 0$$

- Full orthogonality forces $H_X(\tilde{H}_Z)' = 0$, $H_Z(\tilde{H}_X)' = 0$.
- Removed rows $\tilde{H}_X, \tilde{H}_Z$ become logicals.
- $d_{\min} \leq$ row weight.

Design Principle

- Active orthogonality only: $H_X H_Z' = 0$.
- Latent blocks non-orthogonal: $H_X(\tilde{H}_Z)' \neq 0$, $H_Z(\tilde{H}_X)' \neq 0$.
- Latent distances: $d_X^{(\mathrm{lat})}, d_Z^{(\mathrm{lat})}$.

Latent-based distances

$$d_X^{(\mathrm{lat})} := \min\{(C_Z \cap \mathrm{Row}(\tilde{H}_X)) \setminus C_X^{\perp}\},$$
$$d_Z^{(\mathrm{lat})} := \min\{(C_X \cap \mathrm{Row}(\tilde{H}_Z)) \setminus C_Z^{\perp}\}.$$

Code Construction

Example $J = 3, L = 12$.

$$\hat{H}_X = \begin{pmatrix} F_0 & F_1 & F_2 & F_3 & F_4 & F_5 & G_0 & G_1 & G_2 & G_3 & G_4 & G_5 \\ F_5 & F_0 & F_1 & F_2 & F_3 & F_4 & G_5 & G_0 & G_1 & G_2 & G_3 & G_4 \\ F_4 & F_5 & F_0 & F_1 & F_2 & F_3 & G_4 & G_5 & G_0 & G_1 & G_2 & G_3 \\ F_3 & F_4 & F_5 & F_0 & F_1 & F_2 & G_3 & G_4 & G_5 & G_0 & G_1 & G_2 \\ F_2 & F_3 & F_4 & F_5 & F_0 & F_1 & G_2 & G_3 & G_4 & G_5 & G_0 & G_1 \\ F_1 & F_2 & F_3 & F_4 & F_5 & F_0 & G_1 & G_2 & G_3 & G_4 & G_5 & G_0 \end{pmatrix},$$

$$\hat{H}_Z = \begin{pmatrix} G_0' & G_5' & G_4' & G_3' & G_2' & G_1' & F_0' & F_5' & F_4' & F_3' & F_2' & F_1' \\ G_1' & G_0' & G_5' & G_4' & G_3' & G_2' & F_1' & F_0' & F_5' & F_4' & F_3' & F_2' \\ G_2' & G_1' & G_0' & G_5' & G_4' & G_3' & F_2' & F_1' & F_0' & F_5' & F_4' & F_3' \\ G_3' & G_2' & G_1' & G_0' & G_5' & G_4' & F_3' & F_2' & F_1' & F_0' & F_5' & F_4' \\ G_4' & G_3' & G_2' & G_1' & G_0' & G_5' & F_4' & F_3' & F_2' & F_1' & F_0' & F_5' \\ G_5' & G_4' & G_3' & G_2' & G_1' & G_0' & F_5' & F_4' & F_3' & F_2' & F_1' & F_0' \end{pmatrix},$$

$$\hat{H}_X(\hat{H}_Z)' = \begin{pmatrix} \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 \\ \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 \\ \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 \\ \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 \\ \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 \\ \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 \end{pmatrix},$$

$$\Psi_r := \sum_{u=0}^{L/2-1} \left(F_u G_{r-u} + G_{r-u} F_u \right), \quad r \in [L/2].$$

Example $J = 3, L = 12$.

$$\hat{H}_X(\hat{H}_Z)' = \begin{pmatrix} \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 \\ \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 \\ \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 & \Psi_3 \\ \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 & \Psi_2 \\ \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 & \Psi_1 \\ \Psi_1 & \Psi_2 & \Psi_3 & \Psi_4 & \Psi_5 & \Psi_0 \end{pmatrix}$$

Γ	G_0	G_1	G_2	G_3	G_4	G_5
F_0	c	c	c		c	c
F_1	c	c		c	c	c
F_2	c		c	c	c	c
F_3		c	c	c	c	c
F_4	c	c	c	c	c	
F_5	c	c	c	c		c

Code Construction (Cont.)

Required constraints on F_i, G_j:

- Commute on Γ
- Avoid short cycles
- Full search is combinatorial.

Search strategy:

- Restrict to affine permutations on $\mathbb{Z}_P$.
- Checks are P-independent.
- Sequential construction is fast[1].

---

[1] github.com/kasaikenta/construct_apm_css_code

Constructed Code Example

- Girth-8 $(3, 12)$-regular $[[9216, 4612, \leq 48]]$ with $P = 768$.
- Explicit weight-48 logicals $\Rightarrow d_{\min} \leq 48$.
- $d_X^{(\text{lat})} = d_Z^{(\text{lat})} = 48$ (proof omitted).
- No logical failures observed $\Rightarrow d_{\min}$ likely near 48.

Decoding Algorithm (BP + Post-Processing)

1. **Joint BP:** decode on H_X, H_Z using X/Z correlations.

2. **Trigger:** if unsatisfied checks are small (e.g., ≤ 10), estimate a suspect set K (OSD + flip-history + ETS library).

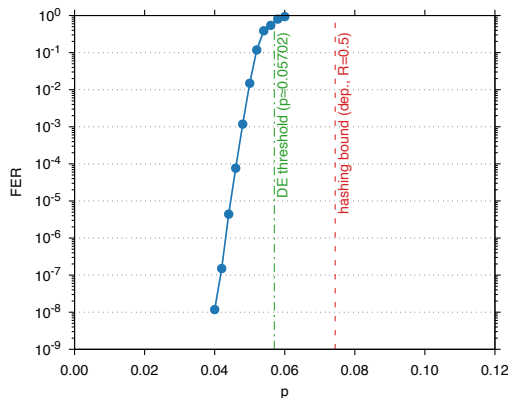3. **PP:** solve the restricted residual and apply only if small-weight.
$$\mathbf{s}_X = (H_Z)_K(\mathbf{x})_K \oplus (H_Z)_{\overline{K}}(\hat{\mathbf{x}})_{\overline{K}}$$

Performance

- **Code:** girth-8, (3,12)-regular $[[9216, 4612, \leq 48]]$.
- **Decoding:** joint BP + PP[a] reaches FER 10^{-8} at $p = 4\%$.
- **Benchmark:** BP aligns with DE (cycle-free, random non-orthogonal (3,12) code).

---

[a]github.com/kasaikenta/joint_BP_plus_PP

Conclusion

- Active-only orthogonality avoids the distance penalty.
- APMs control commutativity and short cycles.
- Girth-8 $(3, 12)$-regular $[[9216, 4612, \leq 48]]$ with strong BP performance.