Flowering graphs

Interactive Oracle Proofs of Proximity to codes on graphs by flowering

Hugo Delavenne, Tanguy Medevielle, Élina Roussel

LIX, École Polytechnique, Institut Polytechnique de Paris
Inria

Friday 7th March 2025
@ CCSW 2025, Eindhoven




Contents

1 Proximity tests

2 Graphs

3 Flowering protocol

- Interactive Oracle Proof of Proximity
  - Locally-testable codes
  - Examples of locally-testable codes
  - Interactive Oracle Proofs of Proximity
- Fast Reed-Solomon IOPP
  - Folding
  - FRI protocol
  - Complexities
  - Completeness and soundness

Definition **Oracle access**

An algorithm $\mathcal{V}$ has **oracle access** to $u \in \mathbb{F}^n$ if it has **black-box** access to u.

- Denoted $\mathcal{V}^u$
- Can limit or count number of queries to oracle

Definition **Locally-testable code**

A code C is (ℓ, δ, s)**-locally-testable** if there is $\mathcal{V}$ with ℓ**-oracle access** such that

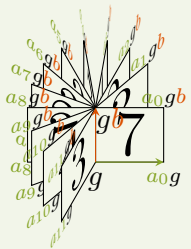
- **Completeness:** if $u \in C$ then $\mathbb{P}(\mathcal{V}^u \text{ accepts}) = 1$
- **Soundness:** if $\Delta(u, C) > \delta$ then $\mathbb{P}(\mathcal{V}^u \text{ accepts}) \leqslant s$.

Example **Reed-Solomon codes are not locally-testable**

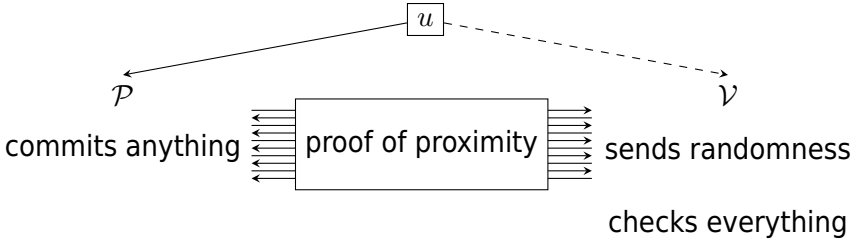
$\mathrm{RS}[n,k] := \{(f(x_1),...,f(x_n)) \mid f \in \mathbb{F}_q[X]_{<k}\}$ has locality $k+1$.

Example **Codes with constant rate, minimal distance and locality [DEL$^+$22]**

There exists codes with constant rate, minimal distance and locality:



[DEL$^+$22] Irit Dinur, Shai Evra, Ron Livne, Alexander Lubotzky, and Shahar Mozes. Locally testable codes with constant rate, distance, and locality.
In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 357–374, 2022



- **Completeness:** if $u \in C$ then $\mathbb{P}(\mathcal{V}^{u,\leftrightarrow\mathcal{P}} \text{ accepts}) = 1$
- **Soundness:** if $\Delta(u, C) > \delta$ then for any $\mathcal{P}$, $\mathbb{P}(\mathcal{V}^{u,\leftrightarrow\mathcal{P}} \text{ accepts}) \leqslant s$
- Can be turned **non-interactive** [FS86, BCS16]

[FS86] Amos Fiat and Adi Shamir. How To Prove Yourself: Practical Solutions to Identification and Signature Problems.
In Andrew M. Odlyzko, editor, *Advances in Cryptology — CRYPTO' 86*, pages 186–194, Berlin, Heidelberg, 1986. Springer Berlin Heidelberg

[BCS16] Eli Ben-Sasson, Alessandro Chiesa, and Nicholas Spooner. Interactive Oracle Proofs.
In *Theory of Cryptography: 14th International Conference, TCC 2016-B, Beijing, China, October 31-November 3, 2016, Proceedings, Part II 14*, pages 31–60. Springer, 2016

Idea. Test even and odd parts $f(X) =: f_{\mathsf{even}}(X^2) + X f_{\mathsf{odd}}(X^2)$.

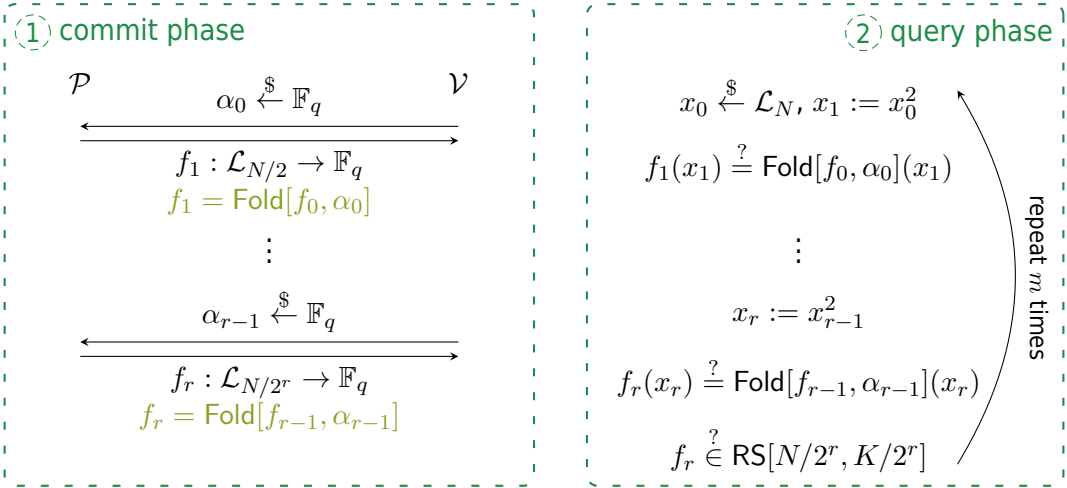
Definition **Folding**

$$\mathsf{Fold}[f, \alpha](Y) := f_{\mathsf{even}}(Y) + \alpha f_{\mathsf{odd}}(Y) = \frac{f(X) + f(-X)}{2} + \alpha \frac{f(X) - f(-X)}{2X},$$

with "$Y = X^2$", $\alpha \in \mathbb{F}_q$

- **Validity preservation:** $f \in \mathsf{RS}[N, K] \iff \mathbb{P}_{\alpha}(\mathsf{Fold}[f, \alpha] \in \mathsf{RS}[N/2, K/2]) > \frac{1}{q}$
- **Local check:** $\mathcal{V}$ computes $\mathsf{Fold}[f, \alpha](x^2)$ with 2 queries to f
- **Field restriction:** $\mathcal{L}_{N/2} := \{x^2 \mid x, -x \in \mathcal{L}_N\}$
  - $\triangleright$ $\mathbb{F}_q$ must have 2^N roots of unity

[BBHR18] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. In ***45th international colloquium on automata, languages, and programming (ICALP 2018)***. **Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018**



[BBHR18] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. **In *45th international colloquium on automata, languages, and programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018**

Proposition **FRI complexities**

FRI protocol for RS$[N, K]$ with m repetitions has following complexity:

- Prover complexity: $< 6N$
- Verifier complexity: $< 2m \log N$
- Query complexity: $< 2m \log K$
- Round complexity: $\log K$

[BBHR18] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. **In *45th international colloquium on automata, languages, and programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018**

Proposition **FRI completeness**

If $f \in \mathsf{RS}[N, K]$ then $\mathcal{V}$ accepts with probability 1.

Theorem **FRI soundness [BCI+23]**

If $\Delta(f, \mathsf{RS}[N, K]) > \delta$ then for any $\mathcal{P}$, $\mathcal{V}$ accepts with probability

$$\leqslant \frac{K^2 \log K}{(2\varepsilon)^7 q} + \left(1 - \min\left(\delta, 1 - \sqrt{K/N}\right)\right)^m,$$

with m a complexity parameter, and $\varepsilon = \sqrt{K/N}/20$.

[BCI+23] Eli Ben-Sasson, Dan Carmon, Yuval Ishai, Swastik Kopparty, and Shubhangi Saraf. Proximity Gaps for Reed–Solomon Codes.
***J. ACM*, 70(5), October 2023**

1 Proximity tests

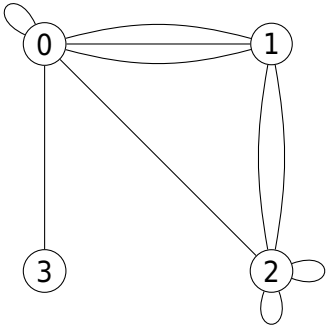
2 Graphs

3 Flowering protocol

- Regular Indexed Multigraphs
- Words and codes on graphs
- Hamming distance vs vertex distance

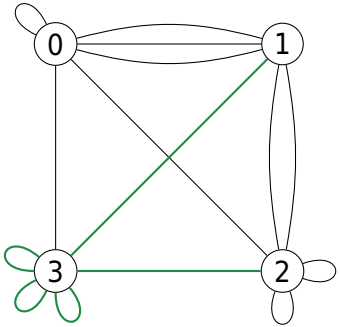
$\Gamma = (V, E)$ is a n-RIM:

- **Multigraph:** multiple edges and loops



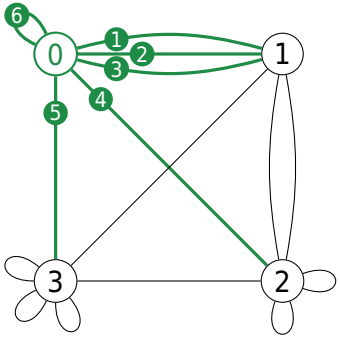
$\Gamma = (V, E)$ is a n-RIM:

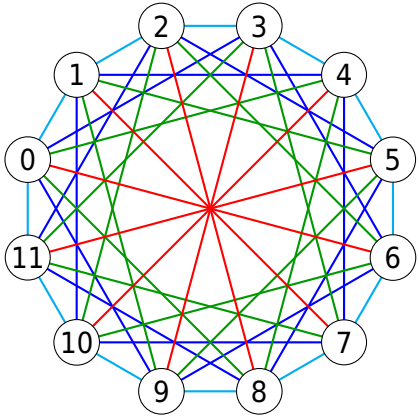
- ▷ **Multigraph:** multiple edges and loops
- ▶ **Regular:** same number n of edges

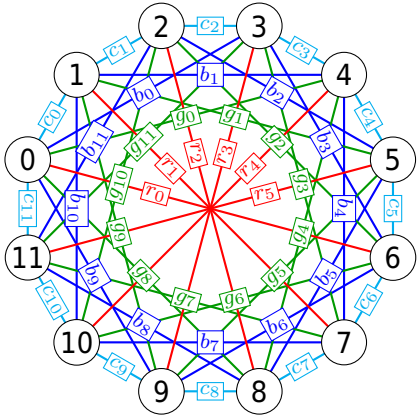


$\Gamma = (V, E)$ is a n-RIM:

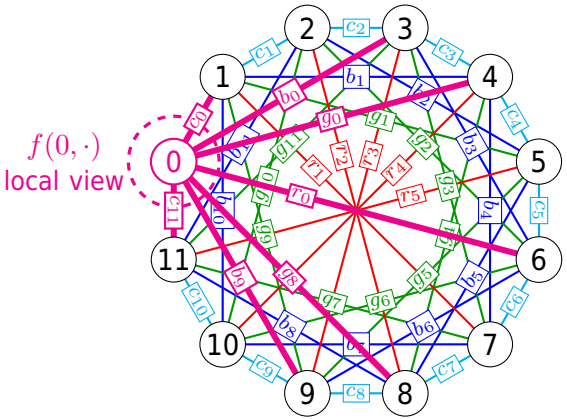
- ▷ **Multigraph:** multiple edges and loops
- ▷ **Regular:** same number n of edges
- ▶ **Indexed:** edge is $(v, \ell) \in V \times [n]$
  Write $E(v, \ell) \in V$ the neighbor of v by ℓ




2
3
1
4
0
5
11
6
10
7
9
8



Word $f(v, \ell)$ on a graph Γ



Word $f(v, \ell)$ on a graph Γ

Definition **Code $\mathcal{C}[\Gamma, C_0]$**

Given Γ a n-RIM and $C_0 \subseteq \mathbb{F}_q^n$,

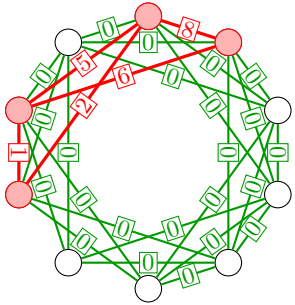
$$f \in \mathcal{C}[\Gamma, C_0] \iff \forall v,\ f(v, \cdot) \in C_0.$$

We'll only use $C_0 = \mathsf{RS}[n, k]$.

Hamming distance vs vertex distance

Hamming: $\Delta(f, f') := \frac{\#\text{diff edges}}{\#\text{edges}} = \frac{5}{30} \approx 0.167$

Vertex: $\Delta_V(f, f') := \frac{\#\text{diff vertices}}{\#\text{vertices}} = \frac{4}{10} = 0.4$



$f - f'$

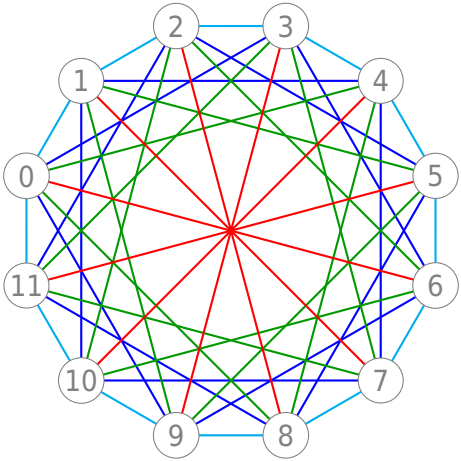
Proposition **Hamming distance is more fine grain than vertex distance**

For any f, f', $\Delta(f, f') \leqslant \Delta_V(f, f')$. (if nodes of Γ have same number of loops)

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

- Folding graphs
  - Cutting graphs
  - Graph isomorphism & folding
  - A flowering example
- Flowering
  - Flowering protocol
  - Complexity comparison
  - Completeness and soundness
- Our codes
  - Cayley graphs
  - Our choice of G and S

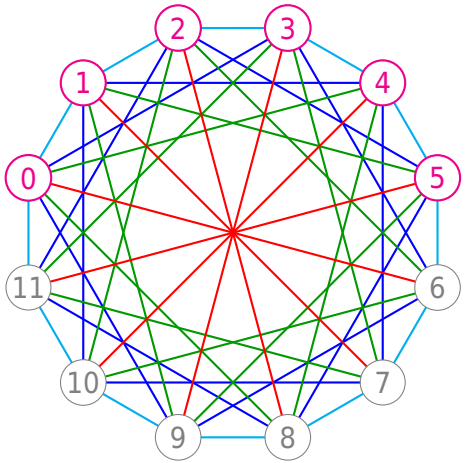
Cut-graph $\Gamma' = \mathsf{Cut}[\Gamma, V']$:



[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Cut-graph $\Gamma' = \mathsf{Cut}[\Gamma, V']$:

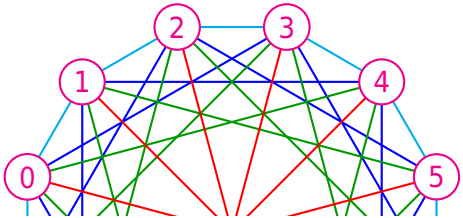
- Choose vertices $V' \subseteq V$



[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Cut-graph $\Gamma' = \mathsf{Cut}[\Gamma, V']$:

- ▷ Choose vertices $V' \subseteq V$
- ▶ Cut the rest

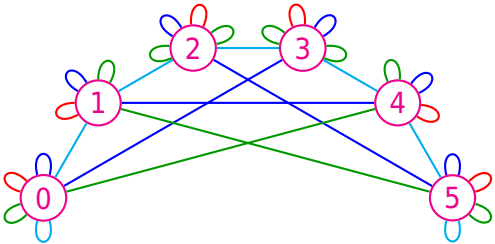


[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Cut-graph $\Gamma' = \mathsf{Cut}[\Gamma, V']$:

- ▷ Choose vertices $V' \subseteq V$
- ▷ Cut the rest
- ▶ Enjoy your new graph

$$E_{V'}(v, \ell) = \begin{cases} E(v, \ell) & \text{if } E(v, \ell) \in V' \\ v & \text{otherwise} \end{cases}$$

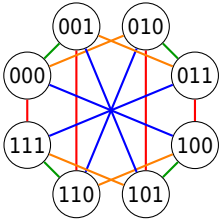


[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Definition **Graph isomorphism**

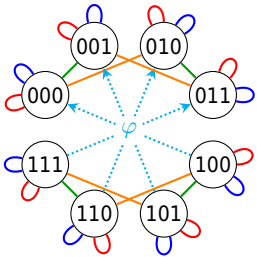
A bijection $\varphi : V' \to V''$ is an **isomorphism** $\Gamma' \to \Gamma''$ if

$$\forall v, \forall \ell, \varphi(E'(v, \ell)) = E''(\varphi(v), \ell).$$



If $V'' = V \setminus V'$ and $\mathsf{Cut}[\Gamma, V'] \sim \mathsf{Cut}[\Gamma, V'']$, the cut is **flowering**.

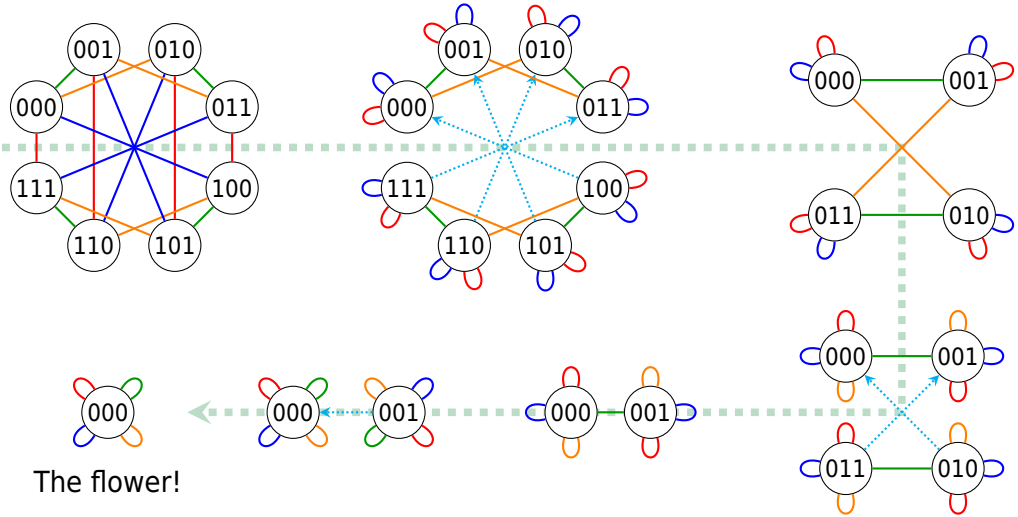
$\mathsf{Cut}[f, V']$ is the restriction $f_{|V' \times [n]}$.

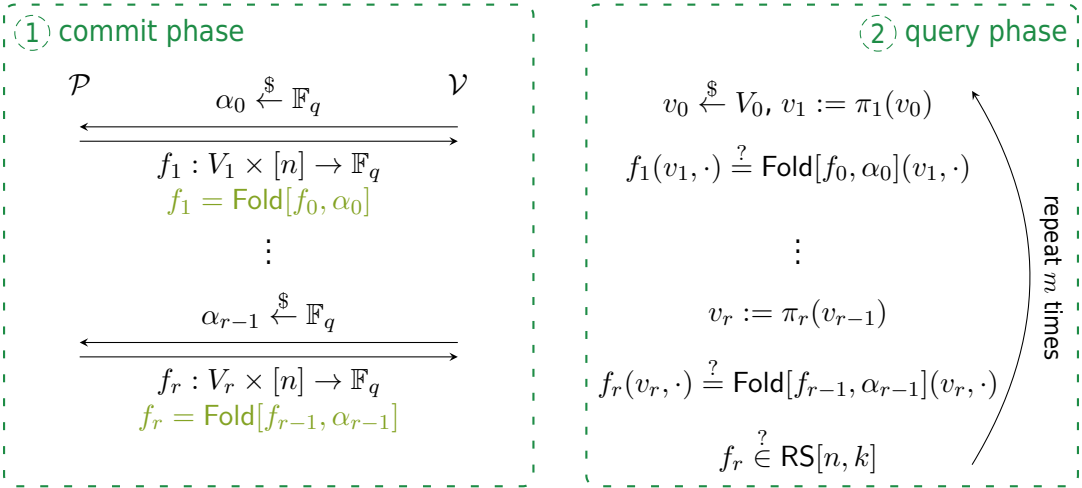


Definition **Folding**

$$\mathsf{Fold}[f, \alpha](v, \ell) := \mathsf{Cut}[f, V'](v, \ell) + \alpha \mathsf{Cut}[f, V''](\varphi(v), \ell)$$

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**


001
010
000
011
111
100
110
101
001
010
000
011
111
100
110
101
000
001
011
010
000
001
011
010
000
000
001
000
001
The flower!



[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Proposition **Flowering complexities**

Flowering with m repetitions has complexities: (recall FRI)

- Prover complexity: $< 3N$ $< 6N$
- Verifier complexity: $< 4m \log^2 N$ $< 2m \log N$
- Query complexity: $< 2m \log^2 N$ $< 2m \log K$
- Round complexity: $< \log N$ $\log K$

[BBHR18] Eli Ben-Sasson, Iddo Bentov, Yinon Horesh, and Michael Riabzev. Fast Reed-Solomon Interactive Oracle Proofs of Proximity. **In *45th international colloquium on automata, languages, and programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018**

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Proposition **Flowering completeness**

There exists $\mathcal{P}$ such that if $f \in \mathcal{C}[\Gamma, \mathsf{RS}[n,k]]$ then $\mathcal{V}$ accepts with probability 1.

Theorem **Flowering soundness**

If $\Delta_V(f, \mathcal{C}[\Gamma, \mathsf{RS}[n,k]]) > \delta$ then for any $\mathcal{P}$, $\mathcal{V}$ accepts with probability

$$\leqslant \min_{\varepsilon} \left(\frac{\log N}{\varepsilon q} + (1 - \delta + \varepsilon \log N)^m \right).$$

Recall the FRI soundness: $\dfrac{K^2 \log K}{(2\varepsilon)^7 q} + \left(1 - \min\left(\delta, 1 - \sqrt{K/N}\right)\right)^m$

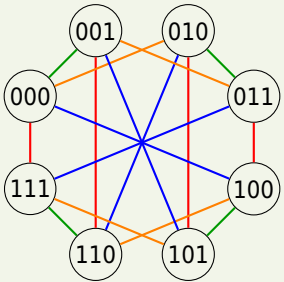
[BCI+23] Eli Ben-Sasson, Dan Carmon, Yuval Ishai, Swastik Kopparty, and Shubhangi Saraf. Proximity Gaps for Reed–Solomon Codes. ***J. ACM*, 70(5), October 2023**

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Definition **Cayley graph**

Fix $(G, \cdot)$. Let $S \subseteq G$ be symmetric. Define $\Gamma = (G, E)$ where $E(g, s) = g \cdot s$.

Example $G = (\mathbb{F}_2^3, +)$, $S = \{100, 010, 001, 111\}$



[Cay78] Arthur Cayley. Desiderata and Suggestions: No. 2. The Theory of Groups: Graphical Representation. ***American Journal of Mathematics*, 1(2):174–176, 1878**

We take

- $G = (\mathbb{F}_2^r, +)$
- S the columns of a parity check matrix of a $[n, n-r, d]_2$ binary code
- $C = \mathcal{C}[\Gamma, \mathsf{RS}[n, k]]$

Proposition **Parameters of the code**

- $N = n2^{r-1}$
- $\operatorname{rate} C \geqslant \frac{2k}{n} - 1$
- $\frac{1}{2}\delta \leqslant \Delta(C) \leqslant \delta$, with $\delta = \frac{1}{2^{r-d+1}}\left(1 - \frac{k-1}{n}\right) = \frac{n2^{d-2}}{N}\left(1 - \frac{k-1}{n}\right)$

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

Our choice of G and S

We take

- $G = (\mathbb{F}_2^r, +)$
- S the columns of a parity check matrix of a $[n, n-r, d]_2$ binary code
- $C = \mathcal{C}[\Gamma, \mathsf{RS}[n, k]]$

Proposition **Parameters of the code**

- $N = n2^{r-1}$
- $\operatorname{rate} C \geqslant \frac{2k}{n} - 1$
- $\frac{1}{2}\delta \leqslant \Delta(C) \leqslant \delta$, with $\delta = \frac{1}{2^{r-d+1}}\left(1 - \frac{k-1}{n}\right) = \frac{n2^{d-2}}{N}\left(1 - \frac{k-1}{n}\right)$

If $d \ll r$ (i.e. far from MDS), δ is **terrible** ($O(1/N)$).

[DMR25] Hugo Delavenne, Tanguy Medevielle, and Élina Roussel. Interactive Oracle Proofs of Proximity to Codes on Graphs, 2025. **Submitted**

1 Proximity tests

2 Graphs

3 Flowering protocol

▷ Future work

Future work

We want flowering-friendly graphs with good **minimal distance** code

- working with **operations research** people
- go deeper into graph **expansion**

Future work

We want flowering-friendly graphs with good **minimal distance** code

- working with **operations research** people
- go deeper into graph **expansion**

Thank you for your attention!