Constructing Moderate-Density Parity-Check Codes from Projective Bundles

SIAM Conference on Applied Algebraic Geometry
Symposium on Coding Theory and Galois Geometries
Eindhoven University of Technology

Jessica Bariffi

joint work with Sam Mattheus, Alessandro Neri, Joachim Rosenthal

12.07.2023

DLR

Outline

Outline

1. MDPC Codes
2. Projective Bundles
3. New MDPC Code Family

Linear Codes

Consider a finite field $\mathbb{F}_q$ of q elements.

Linear code

A k-dimensional subspace $\mathcal{C} \subset \mathbb{F}_q^n$ is called a *q-ary linear code* of length n and dimension k. Its elements are called *codewords*.

Notation $\mathcal{C}$ is an $[n, k]_q$-linear code.

Example of a 2-dimensional subspace of $\mathbb{F}_2^4$

$$\mathcal{C} := \{(0,0,0,0), (0,0,1,1), (1,1,0,0), (1,1,1,1)\} = \langle (0,0,1,1), (1,1,0,0) \rangle$$

Linear Codes

Consider a finite field $\mathbb{F}_q$ of q elements.

Linear code

A k-dimensional subspace $\mathcal{C} \subset \mathbb{F}_q^n$ is called a *q-ary linear code* of length n and dimension k. Its elements are called *codewords*.

Notation $\mathcal{C}$ is an $[n,k]_q$-linear code.

Example of a 2-dimensional subspace of $\mathbb{F}_2^4$

$$\mathcal{C} := \{(0,0,0,0),(0,0,1,1),(1,1,0,0),(1,1,1,1)\} = \langle (0,0,1,1),(1,1,0,0) \rangle$$

Dual code

The *dual code* of an $[n,k]_q$-linear code is given by

$$\mathcal{C}^\perp = \left\{ x \in \mathbb{F}_q^n \,|\, x \cdot c^\top = 0 \text{ for all } c \in \mathcal{C} \right\}.$$

Example

$$\mathcal{C} = \langle (0,0,1,1),(1,1,0,0) \rangle = \mathcal{C}^\perp$$

Hamming Metric

Given two vectors $x, y \in \mathbb{F}_q^n$.

$$\begin{aligned} &\text{Hamming weight:} \quad && \mathrm{wt}_{\mathsf{H}}(x) && := \left| \{ i = 1, \ldots, n \mid x_i \neq 0 \} \right| \\ &\text{Hamming distance:} \quad && \mathrm{d}_{\mathsf{H}}(x, y) && := \mathrm{wt}_{\mathsf{H}}(x - y) \end{aligned}$$

Hamming Metric

Given two vectors $x, y \in \mathbb{F}_q^n$.

$$\begin{aligned} \text{Hamming weight:} \quad & \mathrm{wt}_{\mathsf{H}}(x) && := \left| \, \{i = 1, \ldots, n \mid x_i \neq 0\} \, \right| \\ \text{Hamming distance:} \quad & \mathrm{d}_{\mathsf{H}}(x, y) && := \mathrm{wt}_{\mathsf{H}}(x - y) \end{aligned}$$

Minimum distance

The *minimum Hamming distance* $\mathrm{d}_{\mathsf{H}}(\mathcal{C})$ of an $[n, k]_q$-linear code $\mathcal{C}$ is the minimal Hamming weight of a nonzero codeword, i.e.,

$$\mathrm{d}_{\mathsf{H}}(\mathcal{C}) := \min \{\mathrm{wt}_{\mathsf{H}}(c) \mid c \in \mathcal{C} \setminus \{0\}\}$$

Hamming Metric

Given two vectors $x, y \in \mathbb{F}_q^n$.

$$\begin{aligned} \text{Hamming weight:} \quad & \mathrm{wt}_{\mathsf{H}}(x) && := \left| \{ i = 1, \ldots, n \,|\, x_i \neq 0 \} \right| \\ \text{Hamming distance:} \quad & \mathrm{d}_{\mathsf{H}}(x, y) && := \mathrm{wt}_{\mathsf{H}}(x - y) \end{aligned}$$

Minimum distance

The *minimum Hamming distance* $\mathrm{d}_{\mathsf{H}}(\mathcal{C})$ of an $[n, k]_q$-linear code $\mathcal{C}$ is the minimal Hamming weight of a nonzero codeword, i.e.,

$$\mathrm{d}_{\mathsf{H}}(\mathcal{C}) := \min \{ \mathrm{wt}_{\mathsf{H}}(c) \,|\, c \in \mathcal{C} \setminus \{0\} \}$$

Properties of the minimum distance

- $\mathrm{d}_{\mathsf{H}}(\mathcal{C}) = d$ means that $\lfloor (d-1)/2 \rfloor$ errors can be corrected
- Singleton bound: $d \leq n - k + 1$

Code Representation

Parity-Check Matrix

A *parity-check matrix* of the code $\mathcal{C} \subset \mathbb{F}_q^n$ is a matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ satisfying

$$\mathcal{C} = \ker(H) = \left\{ x \in \mathbb{F}_q^n \mid xH^\top = 0 \right\}.$$

Code Representation

Parity-Check Matrix

A *parity-check matrix* of the code $\mathcal{C} \subset \mathbb{F}_q^n$ is a matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ satisfying

$$\mathcal{C} = \ker(H) = \left\{ x \in \mathbb{F}_q^n \mid xH^\top = 0 \right\}.$$

Example over $\mathbb{F}_2^4$

The code $\mathcal{C} = \langle (0,0,1,1),(1,1,0,0) \rangle$ has a parity-check matrix of the form

$$H = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

Code Representation

Parity-Check Matrix

A *parity-check matrix* of the code $\mathcal{C} \subset \mathbb{F}_q^n$ is a matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ satisfying

$$\mathcal{C} = \ker(H) = \left\{ x \in \mathbb{F}_q^n \,|\, xH^\top = 0 \right\}.$$

Example over $\mathbb{F}_2^4$

The code $\mathcal{C} = \langle (0,0,1,1),(1,1,0,0)\rangle$ has a parity-check matrix of the form

$$H = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}.$$

Given a code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with parity-check matrix H. We say it has *type* (v,w), if

- every column of H has a constant weight v,
- every row of H has a constant weight w.

MDPC Codes

MDPC code

A *moderate-density parity-check (MDPC) code* is a binary linear code of length n with a parity-check matrix whose row weight is $\mathcal{O}(\sqrt{n})$.

MDPC Codes

MDPC code

A *moderate-density parity-check (MDPC) code* is a binary linear code of length n with a parity-check matrix whose row weight is $\mathcal{O}(\sqrt{n})$.

- Introduced as an extension to low-density parity-check codes [Gal62]
  $\longrightarrow$ especially interesting for code-based cryptography [MTSB13]
- Different constructions exist for MDPC codes
  $\longrightarrow$ random, cyclic, quasi-cyclic, ...
- MDPC codes can be decoded with low complexity [MTSB13]
- Several decoding algorithms analysed for MDPC codes [BL18]
  $\longrightarrow$ Bit-Flipping Decoding [Gal63]

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number* $s_H := \max \left| \left\{ i = 1, \ldots, n \,|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|$.

Decoding Performance of MDPC Codes

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number*
  $s_H := \max \left| \left\{ i = 1, \ldots, n \,|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|.$
  Example:

$$H = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix} \implies s_H = 1$$

Decoding Performance of MDPC Codes

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number* $s_H := \max \left| \left\{ i = 1, \ldots, n \,|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|.$

Then performing one round of bit-flipping allows to correct errors of weight at most

$$\left\lfloor \frac{v}{2 s_H} \right\rfloor .$$

Decoding Performance of MDPC Codes

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number* $s_H := \max \left| \left\{ i = 1, \ldots, n \,|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|$.

Then performing one round of bit-flipping allows to correct errors of weight at most

$$\left\lfloor \frac{v}{2s_H} \right\rfloor .$$

Random construction: s_H is $\mathcal{O}(\frac{\log n}{\log \log n})$

Decoding Performance of MDPC Codes

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number* $s_H := \max \left| \left\{ i = 1, \dots, n \,|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|$.

Then performing one round of bit-flipping allows to correct errors of weight at most

$$\left\lfloor \frac{v}{2 s_H} \right\rfloor .$$

Random construction: s_H is $\mathcal{O}(\frac{\log n}{\log \log n})$

Construct codes with good performance.

Decoding Performance of MDPC Codes

Decoding performance result MDPC codes - Bit-flipping [Til18]

Given an MDPC code $\mathcal{C} \subseteq \mathbb{F}_q^n$ with

- a parity-check matrix H of type (v, w), and
- a *maximum column intersection number* $s_H := \max \left| \left\{ i = 1, \ldots, n \,\middle|\, h_{ij} = h_{ij'} = 1 \text{ and } j \neq j' \right\} \right|$.

Then performing one round of bit-flipping allows to correct errors of weight at most

$$\left\lfloor \frac{v}{2s_H} \right\rfloor .$$

Random construction: s_H is $\mathcal{O}(\frac{\log n}{\log \log n})$

Construct codes with good performance.
Construct codes with small maximum column intersection s.

Outline

1. MDPC Codes

2. Projective Bundles

3. New MDPC Code Family

Projective Planes

$\mathsf{PG}(2, q)$: projective plane in $\mathbb{F}_q$ consisting of

- $q^2 + q + 1$ points
- $q^2 + q + 1$ lines

Properties

1. Two points lie on exactly one common line and vice versa.
2. Each point lies on $q + 1$ lines & each line contains $q + 1$ points.
3. No three points among four are collinear.

Projective Planes

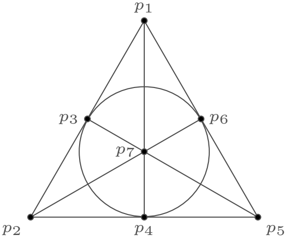
$\mathsf{PG}(2,q)$: projective plane in $\mathbb{F}_q$ consisting of

- q^2+q+1 points
- q^2+q+1 lines

Properties

1. Two points lie on exactly one common line and vice versa.
2. Each point lies on $q+1$ lines & each line contains $q+1$ points.
3. No three points among four are collinear.

Fano Plane $\mathsf{PG}(2,2)$



Projective Planes

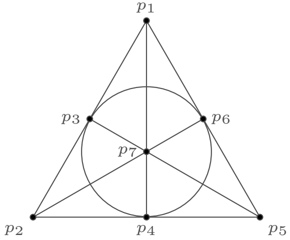
$\mathsf{PG}(2,q)$: projective plane in $\mathbb{F}_q$ consisting of

- q^2+q+1 points
- q^2+q+1 lines

Properties

1. Two points lie on exactly one common line and vice versa.
2. Each point lies on $q+1$ lines & each line contains $q+1$ points.
3. No three points among four are collinear.

Fano Plane $\mathsf{PG}(2,2)$



Representation through incidence matrix

$\mathsf{PG}(2,q)$ can be represented by a matrix $A \in \mathbb{F}_2^{(q^2+q+1)\times(q^2+q+1)}$ defined as

$$A_{p\ell} = \begin{cases} 1 & \text{point } p \text{ is incident to line } \ell \\ 0 & \text{otherwise} \end{cases}$$

Projective Planes

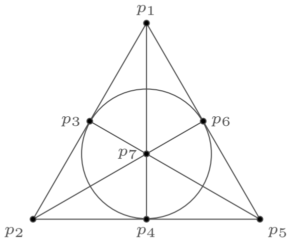
$\mathsf{PG}(2,q)$: projective plane in $\mathbb{F}_q$ consisting of

- q^2+q+1 points
- q^2+q+1 lines

Properties

1. Two points lie on exactly one common line and vice versa.
2. Each point lies on $q+1$ lines & each line contains $q+1$ points.
3. No three points among four are collinear.

Fano Plane $\mathsf{PG}(2,2)$



Representation through incidence matrix

$$A_{\mathsf{Fano}} = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \end{array} \begin{array}{c} \begin{array}{ccccccc} \ell_1 & \ell_2 & \ell_3 & \ell_4 & \ell_5 & \ell_6 & \ell_7 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

Codes from Planes

Given an incidence matrix H of $\Pi = \mathsf{PG}(2,q)$ over $\mathbb{F}_2$. Then we define the binary code $\mathcal{C}_2(\Pi)^\perp \subset (\mathbb{F}_2)^{q^2+q+1}$ by

$$\mathcal{C}_2(\Pi)^\perp := \ker(H).$$

Codes from Planes

Given an incidence matrix H of $\Pi = \mathsf{PG}(2,q)$ over $\mathbb{F}_2$. Then we define the binary code $\mathcal{C}_2(\Pi)^\perp \subset (\mathbb{F}_2)^{q^2+q+1}$ by

$$\mathcal{C}_2(\Pi)^\perp := \ker(H).$$

Parameters of $\mathcal{C}_2(\Pi)^\perp$ - [GM66, AJMJ70]

If q is odd:

- $n = q^2 + q + 1$
- $k = 1$
- $\mathrm{d}_{\mathsf{H}}(\mathcal{C}) = q^2 + q + 1$

If $q = 2^h$:

- $n = q^2 + q + 1$
- $k = 2^{2h} - 3^h + 2^h$
- $\mathrm{d}_{\mathsf{H}}(\mathcal{C}) = 2^h + 2 = q + 2$

Codes from Planes

Given an incidence matrix H of $\Pi = \mathsf{PG}(2,q)$ over $\mathbb{F}_2$. Then we define the binary code $\mathcal{C}_2(\Pi)^\perp \subset (\mathbb{F}_2)^{q^2+q+1}$ by

$$\mathcal{C}_2(\Pi)^\perp := \ker(H).$$

Parameters of $\mathcal{C}_2(\Pi)^\perp$ - [GM66, AJMJ70]

If q is odd:

- $n = q^2 + q + 1$
- $k = 1$
- $\mathrm{d_H}(\mathcal{C}) = q^2 + q + 1$

If $q = 2^h$:

- $n = q^2 + q + 1$
- $k = 2^{2h} - 3^h + 2^h$
- $\mathrm{d_H}(\mathcal{C}) = 2^h + 2 = q + 2$

Bit-Flipping Error Correction

After performing one round of the bit-flipping algorithm on a parity-check matrix H of $\mathcal{C}_2(\Pi)^\perp$, errors of weight up to $\left\lfloor \frac{\mathrm{d_H}(\mathcal{C})-1}{2} \right\rfloor$ can be corrected.

Codes from Planes

Given an incidence matrix H of $\Pi = \mathsf{PG}(2,q)$ over $\mathbb{F}_2$. Then we define the binary code $\mathcal{C}_2(\Pi)^\perp \subset (\mathbb{F}_2)^{q^2+q+1}$ by

$$\mathcal{C}_2(\Pi)^\perp := \ker(H).$$

Parameters of $\mathcal{C}_2(\Pi)^\perp$ - [GM66, AJMJ70]

If q is odd:

- $n = q^2 + q + 1$
- $k = 1$
- $\mathrm{d_H}(\mathcal{C}) = q^2 + q + 1$

If $q = 2^h$:

- $n = q^2 + q + 1$
- $k = 2^{2h} - 3^h + 2^h$
- $\mathrm{d_H}(\mathcal{C}) = 2^h + 2 = q + 2$

Bit-Flipping Error Correction

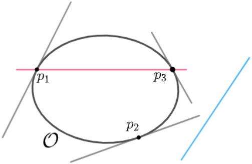
After performing one round of the bit-flipping algorithm on a parity-check matrix H of $\mathcal{C}_2(\Pi)^\perp$, errors of weight up to $\left\lfloor \frac{\mathrm{d_H}(\mathcal{C})-1}{2} \right\rfloor$ can be corrected.

- Codes from planes are powerful and optimal w.r.t. bit-flipping decoding.
- Drawback: Only codes from planes of even order are interesting.

Projective Bundles

Ovals in $\mathsf{PG}(2, q)$

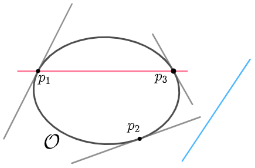
- set $\mathcal{O}$ of $q+1$ points of $\mathsf{PG}(2, q)$
- every line of $\mathsf{PG}(2, q)$ intersects $\mathcal{O}$ in at most 2 points
- exactly $q+1$ tangents - one in every point



Projective Bundles

Ovals in $\mathsf{PG}(2,q)$

- set $\mathcal{O}$ of $q+1$ points of $\mathsf{PG}(2,q)$
- every line of $\mathsf{PG}(2,q)$ intersects $\mathcal{O}$ in at most 2 points
- exactly $q+1$ tangents - one in every point



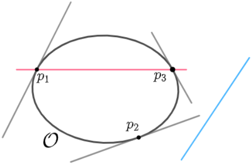
Projective Bundle - [Gly78]

A *projective bundle* is a collection of q^2+q+1 ovals of $\mathsf{PG}(2,q)$ mutually intersecting in a unique point.

Projective Bundles

Ovals in $\mathsf{PG}(2, q)$

- set $\mathcal{O}$ of $q + 1$ points of $\mathsf{PG}(2, q)$
- every line of $\mathsf{PG}(2, q)$ intersects $\mathcal{O}$ in at most 2 points
- exactly $q + 1$ tangents - one in every point



Projective Bundle - [Gly78]

A *projective bundle* is a collection of $q^2 + q + 1$ ovals of $\mathsf{PG}(2, q)$ mutually intersecting in a unique point.

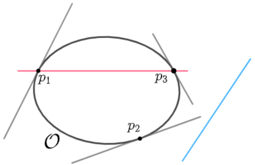
Existence - [Gly78]

- Projective bundles exist for any q.
- For q odd, there are three distinct types of projective bundles.

Projective Bundles

Ovals in $\mathsf{PG}(2,q)$

- set $\mathcal{O}$ of $q+1$ points of $\mathsf{PG}(2,q)$
- every line of $\mathsf{PG}(2,q)$ intersects $\mathcal{O}$ in at most 2 points
- exactly $q+1$ tangents - one in every point



Projective Bundle - [Gly78]

A *projective bundle* is a collection of q^2+q+1 ovals of $\mathsf{PG}(2,q)$ mutually intersecting in a unique point.

Existence - [Gly78]

- Projective bundles exist for any q.
- For q odd, there are three distinct types of projective bundles.

Question: Why are they interesting to us?

Projective Bundles vs. Projective Planes

Projective Plane $\mathsf{PG}(2,q)$

With incidence matrix A

- q^2+q+1 points mutually intersecting in one line
- q^2+q+1 lines mutually intersecting in one point
- each point lies on $q+1$ lines
- each line contains $q+1$ points

Projective Bundle $\mathcal{B}$ in $\mathsf{PG}(2,q)$

With incidence matrix B

- q^2+q+1 ovals mutually intersecting in one tangent line
- q^2+q+1 lines in $\mathsf{PG}(2,q)$ mutually tangent to one oval
- each oval has $q+1$ tangent lines
- each line has $q+1$ tangent ovals

Projective Bundles vs. Projective Planes

Projective Plane $\mathsf{PG}(2, q)$

With incidence matrix A

- $q^2 + q + 1$ points mutually intersecting in one line
- $q^2 + q + 1$ lines mutually intersecting in one point
- each point lies on $q + 1$ lines
- each line contains $q + 1$ points

Projective Bundle $\mathcal{B}$ in $\mathsf{PG}(2, q)$

With incidence matrix B

- $q^2 + q + 1$ ovals mutually intersecting in one tangent line
- $q^2 + q + 1$ lines in $\mathsf{PG}(2, q)$ mutually tangent to one oval
- each oval has $q + 1$ tangent lines
- each line has $q + 1$ tangent ovals

Projective Plane from Ovals

Given a projective bundle $\mathcal{B}$ over $\mathsf{PG}(2, q)$. Identify the ovals of $\mathcal{B}$ and the lines of $\mathsf{PG}(2, q)$ as the points and lines, respectively, of $\mathsf{PG}(2, q)$ with incidence defined by tangency. Then this point-line geometry is a projective plane of order q.

In other words: $A^\top B$ is an incidence matrix of a projective plane again.

Outline

1. MDPC Codes
2. Projective Bundles
3. New MDPC Code Family

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^\perp := \ker(H).$$

Parameters

- block length: $n = 2(q^2 + q + 1)$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

Parameters

- block length: $n = 2(q^2 + q + 1)$
- dimension: $k = \begin{cases} q^2 + q + 2 & \text{if } q \text{ if odd,} \\ 2^{2h+1} + 2^{h+1} - 2(3^h) + 1 & \text{if } q = 2^h \end{cases}$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

Parameters

- block length: $n = 2(q^2 + q + 1)$
- dimension: $k = \begin{cases} q^2 + q + 2 & \text{if } q \text{ if odd,} \\ 2^{2h+1} + 2^{h+1} - 2(3^h) + 1 & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d_H}(\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp}) = q + 2$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2,q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

Parameters

- block length: $n = 2(q^2+q+1)$
- dimension: $k = \begin{cases} q^2+q+2 & \text{if } q \text{ if odd,} \\ 2^{2h+1}+2^{h+1}-2(3^h)+1 & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d}_{\mathsf{H}}(\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp}) = q+2$
- type: $(v,w) = (q+1, 2(q+1))$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

Parameters

- block length: $n = 2(q^2 + q + 1)$
- dimension: $k = \begin{cases} q^2 + q + 2 & \text{if } q \text{ if odd,} \\ 2^{2h+1} + 2^{h+1} - 2(3^h) + 1 & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d}_{\mathsf{H}}(\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp}) = q + 2$
- type: $(v, w) = (q + 1, 2(q + 1))$

MDPC Codes from Projective Bundles

- A: incidence matrix of $\Pi = \mathsf{PG}(2, q)$
- B: incidence matrix of the projective plane Γ induced by ovals of a projective bundle and lines.
- Let $H = (A \mid B) \in \mathbb{F}_2^{(q^2+q+1)\times 2(q^2+q+1)}$.

Code from projective bundles

A binary linear code with parity-check matrix H is called a *projective bundle code* and we write

$$\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp} := \ker(H).$$

Parameters

- block length: $n = 2(q^2 + q + 1)$
- dimension: $k = \begin{cases} q^2 + q + 2 & \text{if } q \text{ if odd,} \\ 2^{2h+1} + 2^{h+1} - 2(3^h) + 1 & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d}_{\mathsf{H}}(\mathcal{C}_2(\Pi \sqcup \Gamma)^{\perp}) = q + 2$
- type: $(v, w) = (q + 1, 2(q + 1))$

$C_2(\Pi \sqcup \Gamma)^{\perp}$ **is an MDPC code!**

Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?

Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?

Perfect Difference Set - [Hir98]
A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?

Perfect Difference Set - [Hir98]

A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?
  By the cyclic shifts (mod $q^2 + q + 1$) of a perfect difference set D.

Perfect Difference Set - [Hir98]

A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Representation of this Code Family

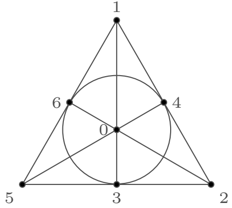
Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?
  By the cyclic shifts (mod $q^2 + q + 1$) of a perfect difference set D.

Perfect Difference Set - [Hir98]
A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Examle Fano Plane $\mathsf{PG}(2, 2)$

- Set of points:
  $\mathcal{P} = \{0, 1, \ldots, 6\}$
- Perfect difference set modulo $q^2 + q + 1 = 7$:
  $D = \{0, 1, 3\}$
- Set of lines:
  $\mathcal{L} = \{\{0 + i, 1 + i, 3 + i\} \mid i \in \mathbb{Z}/7\mathbb{Z}\}$



Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?
  By the cyclic shifts (mod $q^2 + q + 1$) of a perfect difference set D.

Perfect Difference Set - [Hir98]
A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Examle Fano Plane $\mathsf{PG}(2, 2)$

- Set of points:
  $\mathcal{P} = \{0, 1, \ldots, 6\}$
- Perfect difference set modulo $q^2 + q + 1 = 7$:
  $D = \{0, 1, 3\}$
- Set of lines:
  $\mathcal{L} = \{\{0 + i, 1 + i, 3 + i\} \mid i \in \mathbb{Z}/7\mathbb{Z}\}$

$$\begin{pmatrix} 1 & \cdot & \cdot & \cdot & 1 & \cdot & 1 \\ 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot \\ \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 \end{pmatrix}$$

Representation of this Code Family

Alternative way to represent $\mathsf{PG}(2, q)$:

- Points are identified with the integers modulo $q^2 + q + 1$.
- How do to identify the lines?
  By the cyclic shifts (mod $q^2 + q + 1$) of a perfect difference set D.

Perfect Difference Set - [Hir98]

A set $D = \{d_0, \ldots, d_q\} \subset \mathbb{Z}/(q^2 + q + 1)\mathbb{Z}$ is a *perfect difference set* if all differences $(d_i - d_j)$ with $i \neq j$ are distinct modulo $q^2 + q + 1$.

Examle Fano Plane $\mathsf{PG}(2, 2)$

- Set of points:
  $\mathcal{P} = \{0, 1, \ldots, 6\}$
- Perfect difference set modulo $q^2 + q + 1 = 7$:
  $D = \{0, 1, 3\}$
- Set of lines:
  $\mathcal{L} = \{\{0 + i, 1 + i, 3 + i\} \mid i \in \mathbb{Z}/7\mathbb{Z}\}$

$$\begin{pmatrix} 1 & \cdot & \cdot & \cdot & 1 & \cdot & 1 \\ 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot \\ \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot \\ \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 \end{pmatrix}$$

For q odd, the three types of projective bundles can be described via cyclic shifts of $-D, 2D$ and $D/2$.

Example of a Projective Bundle Code for $q = 3$

- Points: $\mathcal{P} = \{0, 1, \ldots, 13\}$
- Perfect difference set modulo 13: $D = \{0, 1, 3, 9\}$
- Lines: $\mathcal{L} = \{\{0 + i, 1 + i, 3 + i, 9 + i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$
- Bundles: $\mathcal{B} = 2D = \{\{0 + i, 2 + i, 5 + i, 6 + i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$

Example of a Projective Bundle Code for $q = 3$

- Points: $\mathcal{P} = \{0, 1, \ldots, 13\}$
- Perfect difference set modulo 13: $D = \{0, 1, 3, 9\}$
- Lines: $\mathcal{L} = \{\{0 + i, 1 + i, 3 + i, 9 + i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$
- Bundles: $\mathcal{B} = 2D = \{\{0 + i, 2 + i, 5 + i, 6 + i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$

$$
H = \left(\begin{array}{ccccccccccccc|ccccccccccccc}
1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot \\
1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 \\
\cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot \\
1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot \\
\cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 \\
\cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 \\
\cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & 1 & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot & \cdot \\
1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot & \cdot \\
\cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot & \cdot \\
\cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1 & \cdot \\
\cdot & \cdot & \cdot & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & \cdot & 1 & 1 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 & 1 & \cdot & \cdot & 1 & \cdot & 1
\end{array}\right)
$$

Example of a Projective Bundle Code for $q = 3$

- Points: $\mathcal{P} = \{0, 1, \ldots, 13\}$
- Perfect difference set modulo 13: $D = \{0, 1, 3, 9\}$
- Lines: $\mathcal{L} = \{\{0+i, 1+i, 3+i, 9+i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$
- Bundles: $\mathcal{B} = 2D = \{\{0+i, 2+i, 5+i, 6+i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$

$$
H = \left(\begin{array}{ccccccccccccc|ccccccccccccc}
1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot\\
1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1\\
\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot\\
1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot\\
\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1\\
\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1\\
\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot\\
\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot\\
\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot\\
1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot\\
\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot\\
\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot\\
\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1
\end{array}\right)
$$

The maximum column intersection is $s_H = 2$. Hence, performing one round of the bit-flipping decoder on H corrects errors of weight up to $\left\lfloor \frac{q+1}{4} \right\rfloor$.

Example of a Projective Bundle Code for $q = 3$

- Points: $\mathcal{P} = \{0, 1, \dots, 13\}$
- Perfect difference set modulo 13: $D = \{0, 1, 3, 9\}$
- Lines: $\mathcal{L} = \{\{0+i, 1+i, 3+i, 9+i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$
- Bundles: $\mathcal{B} = 2D = \{\{0+i, 2+i, 5+i, 6+i\} \mid i \in \mathbb{Z}/13\mathbb{Z}\}$

$$H = \left(\begin{array}{ccccccccccccc|ccccccccccccc}
1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot\\
1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1\\
\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot\\
1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot\\
\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1\\
\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1\\
\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot\\
\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot\\
\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&1&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot&\cdot\\
1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot&\cdot\\
\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot&\cdot\\
\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1&\cdot\\
\cdot&\cdot&\cdot&1&\cdot&\cdot&\cdot&\cdot&\cdot&1&\cdot&1&1&\cdot&\cdot&\cdot&\cdot&\cdot&\cdot&1&1&\cdot&\cdot&1&\cdot&1
\end{array}\right)$$

The maximum column intersection is $s_H = 2$. Hence, performing one round of the bit-flipping decoder on H corrects errors of weight up to $\left\lfloor \frac{q+1}{4} \right\rfloor$.

! $s_H = 2$ for a matrix H of size $(q^2+q+1) \times c$ with $c > q^2+q+1$!

Generalization

Consider the projective plane $\Pi = \mathsf{PG}(2, q)$

- Take $t > 1$ many disjoint projective bundles in Π.
  (Existence proven in [BBEF94])
- Denote the resulting projective planes by $\Gamma_1, \ldots, \Gamma_t$ and the incidence matrices by $B_1, \ldots, B_t$
- Define the binary linear code

$$\mathcal{C}_2(\Pi \sqcup \Gamma_1 \sqcup \ldots, \sqcup \Gamma_t)^\perp := \ker \Big((A \,|\, B_1 \,|\, \ldots \,|\, B_t) \Big)$$

Generalization

Consider the projective plane $\Pi = \mathsf{PG}(2, q)$

- Take $t > 1$ many disjoint projective bundles in Π.
  (Existence proven in [BBEF94])
- Denote the resulting projective planes by $\Gamma_1, \ldots, \Gamma_t$ and the incidence matrices by $B_1, \ldots, B_t$
- Define the binary linear code

$$\mathcal{C}_2(\Pi \sqcup \Gamma_1 \sqcup \ldots, \sqcup \Gamma_t)^\perp := \ker\Big((A \,|\, B_1 \,|\, \ldots \,|\, B_t)\Big)$$

Parameters

- block length: $n = (t+1)(q^2+q+1)$
- dimension: $k = \begin{cases} t(q^2+q+1)+1 & \text{if } q \text{ if odd,} \\ (t+1)(2^{2h}+2^h-(3^h))+t & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d}_{\mathsf{H}}(\mathcal{C}_2(\Pi \sqcup \Gamma_1 \sqcup \ldots, \sqcup \Gamma_t)^\perp) \geq \left\lceil \frac{q+2}{2} \right\rceil$
- maximum column intersection number $s = 4$

Generalization

Consider the projective plane $\Pi = \mathsf{PG}(2, q)$

- Take $t > 1$ many disjoint projective bundles in Π. (Existence proven in [BBEF94])
- Denote the resulting projective planes by $\Gamma_1, \ldots, \Gamma_t$ and the incidence matrices by $B_1, \ldots, B_t$
- Define the binary linear code

$$\mathcal{C}_2(\Pi \sqcup \Gamma_1 \sqcup \ldots, \sqcup \Gamma_t)^\perp := \ker\left((A \mid B_1 \mid \ldots \mid B_t)\right)$$

Parameters

- block length: $n = (t+1)(q^2+q+1)$
- dimension: $k = \begin{cases} t(q^2+q+1)+1 & \text{if } q \text{ if odd,} \\ (t+1)(2^{2h}+2^h-(3^h))+t & \text{if } q = 2^h \end{cases}$
- minimum distance: $\mathrm{d}_{\mathsf{H}}(\mathcal{C}_2(\Pi \sqcup \Gamma_1 \sqcup \ldots, \sqcup \Gamma_t)^\perp) \geq \left\lceil \frac{q+2}{2} \right\rceil$
- maximum column intersection number $s = 4$

Thank you for your attention!

References

[AJMJ70] Edward F Assmus Jr and Harold F Mattson Jr.
Algebraic theory of codes ii.
AFC Research Laboratories Report, 1970.

[BBEF94] R. D. Baker, J. M. N. Brown, G. L. Ebert, and J. C. Fisher.
Projective bundles.
Bulletin of the Belgian Mathematical Society-Simon Stevin, 1(3):329–336, 1994.

[BL18] Hannes Bartz and Gianluigi Liva.
On decoding schemes for the mdpc-mceliece cryptosystem.
CoRR, abs/1801.05659, 2018.

[Gal62] R.G. Gallager.
Low-density parity-check codes.
IRE Trans. on Info. Theory, IT-8:21–28, 1962.

[Gal63] R.G. Gallager.
Low-Density Parity Check Codes.
M.I.T. Press, Cambridge, MA, 1963.
Number 21 in Research monograph series.

References

[Gly78] David G Glynn.
Finite projective planes and related combinatorial systems.
PhD thesis, University of Adelaide Adelaide, 1978.

[GM66] R. L. Graham and Jessie MacWilliams.
On the number of information symbols in difference-set cyclic codes.
Bell System Tech. J., 45:1057–1070, 1966.

[Hir98] James Hirschfeld.
Projective geometries over finite fields.
Oxford University Press, 1998.

[MTSB13] R. Misoczki, J.-P. Tillich, N. Sendrier, and P. SLM Barreto.
MDPC-McEliece: New McEliece variants from moderate density parity-check codes.
In *2013 IEEE international symposium on information theory*, pages 2069–2073. IEEE, 2013.

References

[Til18] Jean-Pierre Tillich.
The decoding failure probability of MDPC codes.
In *2018 IEEE International Symposium on Information Theory (ISIT)*, pages 941–945. IEEE, 2018.