

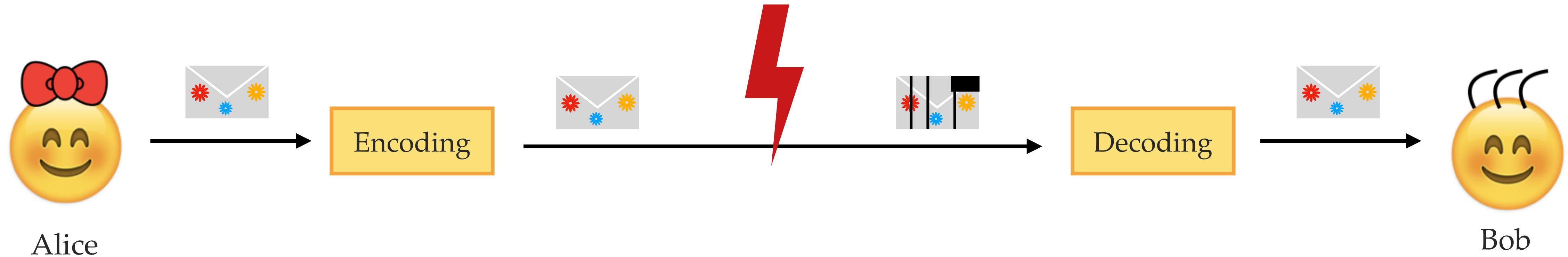
Code-based cryptography II

Monika Trimoska

Selected Areas in Cryptology - Part 1
Spring, 2024

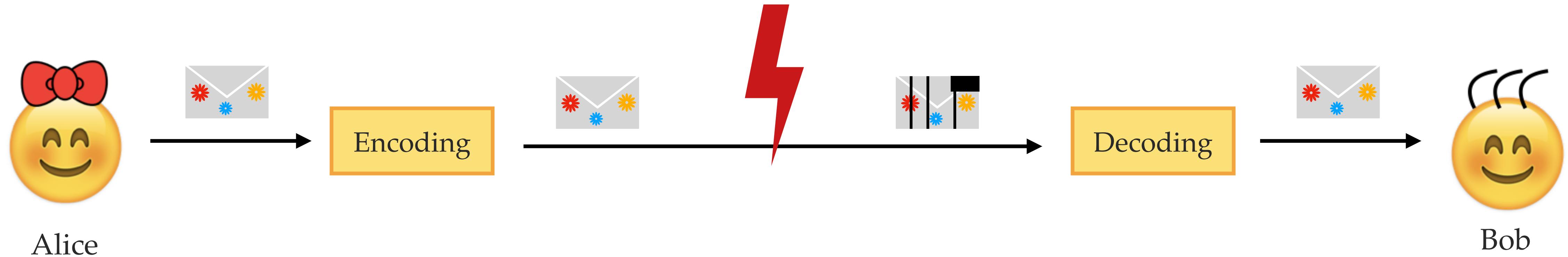
TU/e

Error-correcting codes (recall)



- Primary use case: communication over a noisy channel.
- Main idea: introduce some **redundancy** in order to be able to correct the errors.
- Some structured error-correcting codes have efficient decoding algorithms.
- Decoding is, in general, a **hard problem** - so it is hard for *random* codes.

Error-correcting codes (recall)



- Primary use case: communication over a noisy channel.
- Main idea: introduce some **redundancy** in order to be able to correct the errors.
- Some structured error-correcting codes have efficient decoding algorithms.
- Decoding is, in general, a **hard problem** - so it is hard for *random* codes.



Hard problems (often) find their use in cryptography.

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

- The parameter n is called the **length** of the code.

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

- The parameter n is called the **length** of the code.
- The parameter k is called the **dimension** of the code.

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

- The parameter n is called the **length** of the code.
- The parameter k is called the **dimension** of the code.
- The elements in the code are called **codewords**.

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

- The parameter n is called the **length** of the code.
- The parameter k is called the **dimension** of the code.
- The elements in the code are called **codewords**.

Hamming metric

For $\mathbf{x} \in \mathbb{F}_q^n$, the **Hamming weight** of \mathbf{x} is the number of nonzero elements, aka.

$$\text{wt}(\mathbf{x}) = |\{i \in \{1, \dots, n\} \mid x_i \neq 0\}|.$$

Linear codes (recall)

Linear code

An $[n, k]$ linear code \mathcal{C} over \mathbb{F}_q is a k -dimensional subspace of \mathbb{F}_q^n .

- The parameter n is called the **length** of the code.
- The parameter k is called the **dimension** of the code.
- The elements in the code are called **codewords**.

Hamming metric

For $\mathbf{x} \in \mathbb{F}_q^n$, the **Hamming weight** of \mathbf{x} is the number of nonzero elements, aka.

$$\text{wt}(\mathbf{x}) = |\{i \in \{1, \dots, n\} \mid x_i \neq 0\}|.$$

Generator matrix

The matrix $\mathbf{G} \in \mathbb{F}_q^{k \times n}$ is called a **generator matrix** of \mathcal{C} , if

$$\mathcal{C} = \{\mathbf{x}\mathbf{G} \mid \mathbf{x} \in \mathbb{F}_q^k\}.$$

Binary linear codes

Binary linear code

An $[n, k]$ **binary linear code** \mathcal{C} is a k -dimensional subspace of \mathbb{F}_2^n .

- The parameter n is called the **length** of the code.
- The parameter k is called the **dimension** of the code.
- The elements in the code are called **codewords**.

Hamming metric

For $\mathbf{x} \in \mathbb{F}_2^n$, the **Hamming weight** of \mathbf{x} is the number of nonzero elements, aka.

$$\text{wt}(\mathbf{x}) = |\{i \in \{1, \dots, n\} \mid x_i \neq 0\}|.$$

Generator matrix

The matrix $\mathbf{G} \in \mathbb{F}_2^{k \times n}$ is called a **generator matrix** of \mathcal{C} , if

$$\mathcal{C} = \{\mathbf{xG} \mid \mathbf{x} \in \mathbb{F}_2^k\}.$$

Binary linear codes

Binary linear code

An $[n, k]$ **binary linear code** \mathcal{C} is a k -dimensional subspace of \mathbb{F}_2^n .

Example. $q = 2, n = 5, k = 3$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Binary linear codes

Binary linear code

An $[n, k]$ **binary linear code** \mathcal{C} is a k -dimensional subspace of \mathbb{F}_2^n .

Example. $q = 2, n = 5, k = 3$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

Codewords: $\lambda_1(10101) + \lambda_2(11000) + \lambda_3(11110)$

Example. $\mathbf{c}_1 = (111)\mathbf{G} = (10011),$
 $\mathbf{c}_2 = (100)\mathbf{G} = (10101)$

Decoding

- Encoding: $\mathbf{c} = \mathbf{mG}$
- Introducing error \mathbf{e} of low weight: $\mathbf{y} = \mathbf{c} + \mathbf{e} = \mathbf{mG} + \mathbf{e}$, s.t. $\text{wt}(\mathbf{e}) = t$.
- Decoding: Given \mathbf{y} , find \mathbf{c} s.t. $\mathbf{y} = \mathbf{c} + \mathbf{e}$ and $\text{wt}(\mathbf{e}) \leq t$.

Representations of linear codes

→ The row space of a **generator matrix** $\mathbf{G} \in \mathbb{F}_2^{k \times n}$:

$$\mathcal{C} = \{\mathbf{x}\mathbf{G} \mid \mathbf{x} \in \mathbb{F}_2^k\}.$$

Example. $n = 7, k = 4$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Representations of linear codes

→ The row space of a **generator matrix** $\mathbf{G} \in \mathbb{F}_2^{k \times n}$:

$$\mathcal{C} = \{\mathbf{x}\mathbf{G} \mid \mathbf{x} \in \mathbb{F}_2^k\}.$$

→ The kernel space of a **parity-check matrix** $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$:

$$\mathcal{C} = \{\mathbf{c} \mid \mathbf{H}\mathbf{c} = 0, \mathbf{c} \in \mathbb{F}_2^n\}.$$

Example. $n = 7, k = 4$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

*We are omitting the transpose (T) for vectors.

Representations of linear codes

→ The row space of a **generator matrix** $\mathbf{G} \in \mathbb{F}_2^{k \times n}$:

$$\mathcal{C} = \{\mathbf{x}\mathbf{G} \mid \mathbf{x} \in \mathbb{F}_2^k\}.$$

→ The kernel space of a **parity-check matrix** $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$:

$$\mathcal{C} = \{\mathbf{c} \mid \mathbf{H}\mathbf{c} = 0, \mathbf{c} \in \mathbb{F}_2^n\}.$$

Example. $n = 7, k = 4$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Example. $n = 7, k = 4$

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

*We are omitting the transpose (T) for vectors.

From G to H

From \mathbf{G} to \mathbf{H}

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

From \mathbf{G} to \mathbf{H}

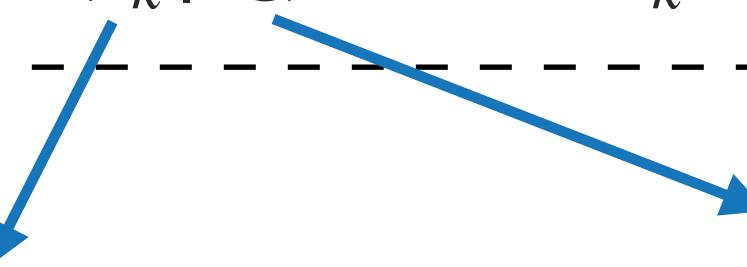
Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part



From \mathbf{G} to $\tilde{\mathbf{G}}$

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

From \mathbf{G} to $\tilde{\mathbf{G}}$

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

→ When $\mathbf{c} = \mathbf{m}\tilde{\mathbf{G}}$, the first k positions of \mathbf{c} are \mathbf{m} .

From G to H

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

→ When $\mathbf{c} = \mathbf{m}\tilde{\mathbf{G}}$, the first k positions of \mathbf{c} are \mathbf{m} .

Example. $(1010)\tilde{\mathbf{G}} = (1010101)$

From \mathbf{G} to $\tilde{\mathbf{G}}$

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

From \mathbf{G} to $\tilde{\mathbf{G}}$

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

→ We can form the parity-check matrix as $\mathbf{H} = (\mathbf{Q}^\top | \mathbf{I}_{n-k})$.

From \mathbf{G} to \mathbf{H}

Systematic form

A systematic generator matrix is a generator matrix of the form

$(\mathbf{I}_k | \mathbf{Q})$, where \mathbf{I}_k is the $k \times k$ identity matrix and \mathbf{Q} is a $k \times (n - k)$ matrix.

information part

redundant part

Example. $\mathbf{G} \rightarrow \tilde{\mathbf{G}}$

$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\tilde{\mathbf{G}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

→ We can form the parity-check matrix as $\mathbf{H} = (\mathbf{Q}^\top | \mathbf{I}_{n-k})$.

Example.

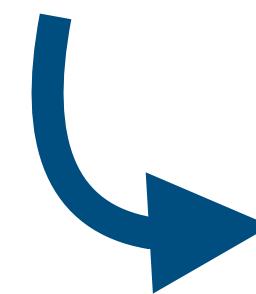
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

From \mathbf{G} to \mathbf{H}

- We have $\tilde{\mathbf{G}} = (\mathbf{I}_k | \mathbf{Q})$.
- We can form the parity-check matrix as $\mathbf{H} = (\mathbf{Q}^\top | \mathbf{I}_{n-k})$.

From \mathbf{G} to \mathbf{H}

- We have $\tilde{\mathbf{G}} = (\mathbf{I}_k | \mathbf{Q})$.
- We can form the parity-check matrix as $\mathbf{H} = (\mathbf{Q}^\top | \mathbf{I}_{n-k})$.



Every codeword is in the kernel space of \mathbf{H} :

From \mathbf{G} to \mathbf{H}

- We have $\tilde{\mathbf{G}} = (\mathbf{I}_k | \mathbf{Q})$.
- We can form the parity-check matrix as $\mathbf{H} = (\mathbf{Q}^\top | \mathbf{I}_{n-k})$.



Every codeword is in the kernel space of \mathbf{H} :

$$\mathbf{H}(\mathbf{m}\tilde{\mathbf{G}})^\top = \mathbf{H}\tilde{\mathbf{G}}^\top \mathbf{m}^\top = (\mathbf{Q}^\top \quad \mathbf{I}_{n-k}) \begin{pmatrix} \mathbf{I}_k \\ \mathbf{Q}^\top \end{pmatrix} \mathbf{m}^\top = (\mathbf{Q}^\top + \mathbf{Q}^\top)\mathbf{m}^\top = \mathbf{0} \cdot \mathbf{m}^\top = \mathbf{0}$$

Example: Hamming code

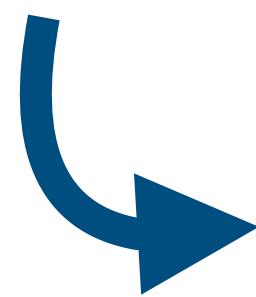


Columns correspond to a bit pattern of length $(n - k)$.

Example. $n = 7, k = 4$

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

Example: Hamming code

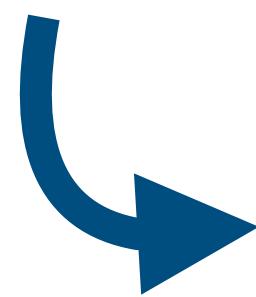


Columns correspond to a bit pattern of length $(n - k)$.

Example. $n = 7, k = 4$

$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \quad \mathbf{c} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Example: Hamming code

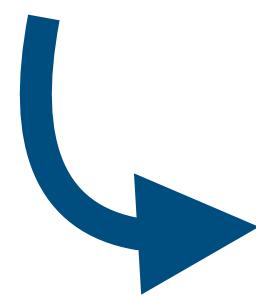


An error occurs.

Example. $n = 7, k = 4$

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{e} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\mathbf{H} \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{e} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

The matrix \mathbf{H} is a 3×7 matrix. The vector \mathbf{c} is a 3×1 column vector. The vector \mathbf{e} is a 3×1 column vector. The result is a 3×1 column vector.

The vector \mathbf{c} is $\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$. The vector \mathbf{e} is $\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$. The result is $\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$.

Example: Hamming code

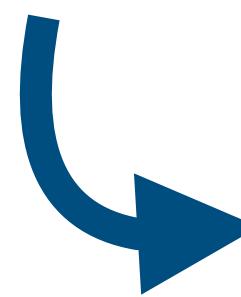


An error occurs.

Example. $n = 7, k = 4$

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{e} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\begin{array}{c} \mathbf{H} \\ \left(\begin{array}{cccccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right) \end{array} \begin{array}{c} \mathbf{c} \\ \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{array} \right) \end{array} + \begin{array}{c} \mathbf{e} \\ \left(\begin{array}{c} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \right) \end{array} = \begin{array}{c} \left(\begin{array}{c} 1 \\ 0 \\ 1 \end{array} \right) \end{array}$$

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{c} \\ \mathbf{e} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$\mathbf{c} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$ + $\mathbf{e} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\begin{array}{c} \mathbf{H} \\ \left(\begin{array}{cccccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right) \end{array} \left(\begin{array}{c} \mathbf{c} \\ \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{array} \right) \end{array} \right) + \left(\begin{array}{c} \mathbf{e} \\ \left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{array} \right) \end{array} \right) = \left(\begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right)$$

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\begin{array}{c} \mathbf{H} \\ \left(\begin{array}{ccccccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right) \end{array} \left(\begin{array}{c} \mathbf{c} \\ \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{array} \right) \end{array} \right) + \left(\begin{array}{c} \mathbf{e} \\ \left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{array} \right) \end{array} \right) = \left(\begin{array}{c} \mathbf{r} \\ \left(\begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right) \end{array} \right)$$

The **failure pattern** uniquely identifies the error location.

Example: Hamming code



An error occurs.

Example. $n = 7, k = 4$

$$\begin{matrix} \mathbf{H} \\ \left(\begin{array}{ccccccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right) \end{matrix} \left(\begin{matrix} \mathbf{c} \\ \left(\begin{array}{c} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{array} \right) \end{matrix} \right) + \left(\begin{matrix} \mathbf{e} \\ \left(\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{array} \right) \end{matrix} \right) = \left(\begin{matrix} \mathbf{r} \\ \left(\begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right) \end{matrix} \right)$$

The **failure pattern** uniquely identifies the error location.

We will call it a **syndrome**.

Syndrome decoding

--- Syndrome ---

The **syndrome** of a word $\mathbf{y} \in \mathbb{F}_2^n$ is $\mathbf{s} = \mathbf{H}\mathbf{y}$.

Syndrome decoding

--- Syndrome ---

The **syndrome** of a word $\mathbf{y} \in \mathbb{F}_2^n$ is $\mathbf{s} = \mathbf{H}\mathbf{y}$.

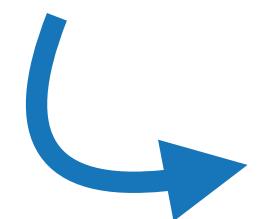
$$\mathbf{H}\mathbf{y} = \mathbf{H}(\mathbf{c} + \mathbf{e}) = \mathbf{H}\mathbf{c} + \mathbf{H}\mathbf{e} = \mathbf{0} + \mathbf{H}\mathbf{e} = \mathbf{H}\mathbf{e}$$

Syndrome decoding

--- Syndrome ---

The **syndrome** of a word $\mathbf{y} \in \mathbb{F}_2^n$ is $\mathbf{s} = \mathbf{H}\mathbf{y}$.

$$\mathbf{H}\mathbf{y} = \mathbf{H}(\mathbf{c} + \mathbf{e}) = \mathbf{H}\mathbf{c} + \mathbf{H}\mathbf{e} = \mathbf{0} + \mathbf{H}\mathbf{e} = \mathbf{H}\mathbf{e}$$



The syndrome depends only on the error vector.

The syndrome decoding problem

The syndrome decoding problem

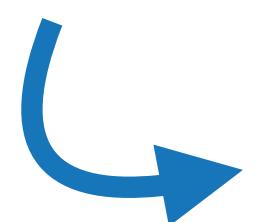
Given a syndrome $s = \mathbf{H}\mathbf{e}$, find \mathbf{e} such that $\text{wt}(\mathbf{e}) \leq t$.

The syndrome decoding problem

The syndrome decoding problem

Given a syndrome $\mathbf{s} = \mathbf{H}\mathbf{e}$, find \mathbf{e} such that $\text{wt}(\mathbf{e}) \leq t$.

$$\begin{array}{ccccccc} \mathbf{H} & & \mathbf{e} & & \mathbf{s} \\ \left(\begin{array}{ccccccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right) & \left(\begin{array}{c} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \end{array} \right) & = & \left(\begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right) \end{array}$$



Find \mathbf{e} of minimum weight.

Information set decoding

Information set decoding algorithms

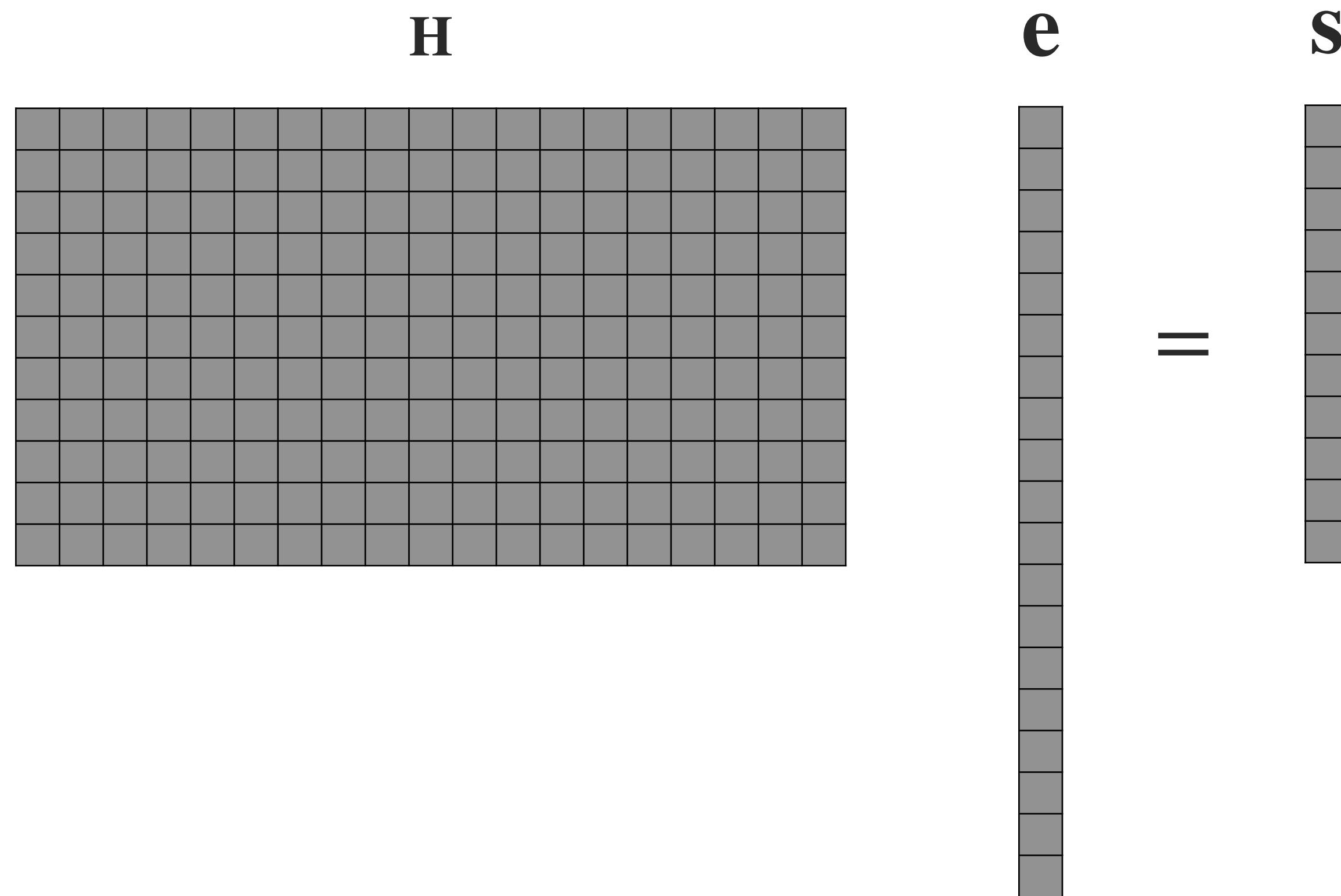


Focus on the case $\text{wt}(\mathbf{e}) = t$.

The syndrome decoding problem

Given a syndrome $\mathbf{s} = \mathbf{H}\mathbf{e}$, find \mathbf{e} such that $\text{wt}(\mathbf{e}) = t$.

Brute force

$$\mathbf{H} \quad \mathbf{e} = \mathbf{S}$$


The diagram illustrates a brute force computation. It features a large matrix \mathbf{H} represented as a 10x10 grid of gray squares. To the right of \mathbf{H} is an equals sign ($=$). To the right of the equals sign are two vertical vectors: \mathbf{e} , which is a tall, narrow column of gray squares, and \mathbf{S} , which is a shorter, wider column of gray squares. The entire equation is $\mathbf{H} \quad \mathbf{e} = \mathbf{S}$.

Brute force

$$\mathbf{H} \mathbf{e} = \mathbf{S}$$

- Entry is 0
- Entry is 1
- Entry is 0 or 1

Brute force

$$H \quad e \quad = \quad S$$

A diagram illustrating the brute force method for matrix multiplication. It shows three components: a large matrix H , a vector e , and a vector S . The matrix H is a 10x10 grid of cells, some of which are black (representing 1) and some are white (representing 0). The vector e is a 10x1 column vector with alternating black and white cells. The vector S is a 10x1 column vector with alternating black and white cells. The equation $H \cdot e = S$ indicates that the product of matrix H and vector e results in vector S .

- Entry is 0
- Entry is 1
- Entry is 0 or 1

Brute force

$$\mathbf{H} \quad \mathbf{e} = \mathbf{s}$$

The diagram illustrates the computation of a sparse vector \mathbf{s} as the sum of columns of matrix \mathbf{H} where the entries of vector \mathbf{e} are non-zero. Matrix \mathbf{H} is a 10x10 grid of entries. Vector \mathbf{e} is a 10x1 column vector with non-zero entries at indices 2, 4, 6, 8, and 10. The resulting vector \mathbf{s} is a 10x1 column vector with non-zero entries at the same indices (2, 4, 6, 8, 10).

→ \mathbf{s} is equal to the sum of the columns where \mathbf{e}_i is nonzero.

□	Entry is 0
■	Entry is 1
■■■■■■■■■■	Entry is 0 or 1

Brute force

$$\mathbf{H} \quad \mathbf{e} \quad = \quad \mathbf{s}$$

The diagram illustrates the multiplication of a matrix \mathbf{H} by a vector \mathbf{e} to produce a vector \mathbf{s} . The matrix \mathbf{H} is a 10x10 grid with a sparse pattern of 1s (black) and 0s (white). The vector \mathbf{e} is a 10x1 column with 1s at indices 2, 4, 6, 8, and 10, and 0s elsewhere. The resulting vector \mathbf{s} is a 10x1 column with 1s at indices 2, 4, 6, 8, and 10, and 0s elsewhere. A legend on the right indicates that white represents an entry being 0, black represents an entry being 1, and gray represents an entry being 0 or 1.

→ \mathbf{s} is equal to the sum of the columns where e_i is nonzero.

→ Pick any group of t columns of \mathbf{H} , add them and compare with \mathbf{s} .

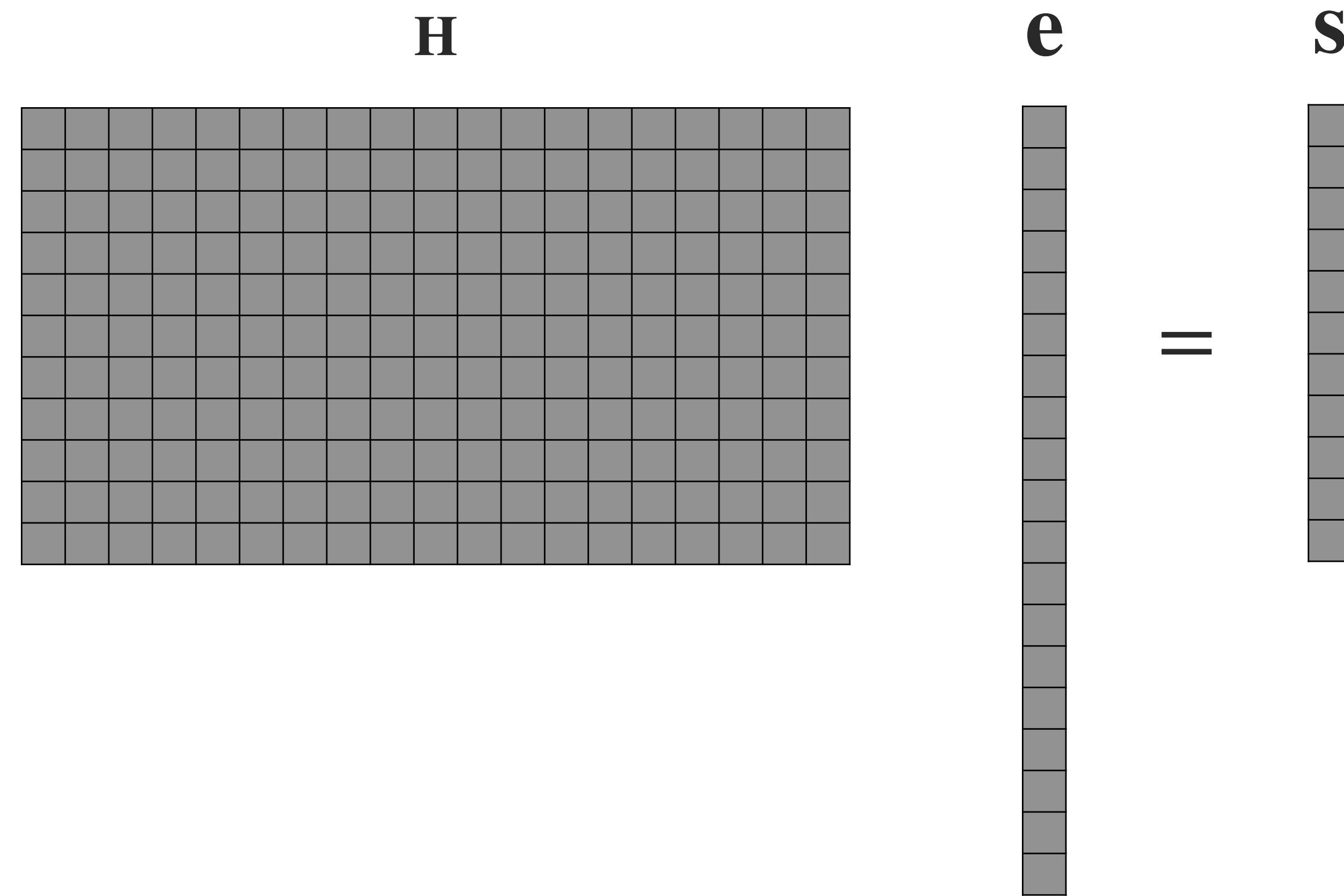
Brute force: complexity

$$\begin{matrix} \text{Matrix } H & = & \text{Vector } s \end{matrix}$$

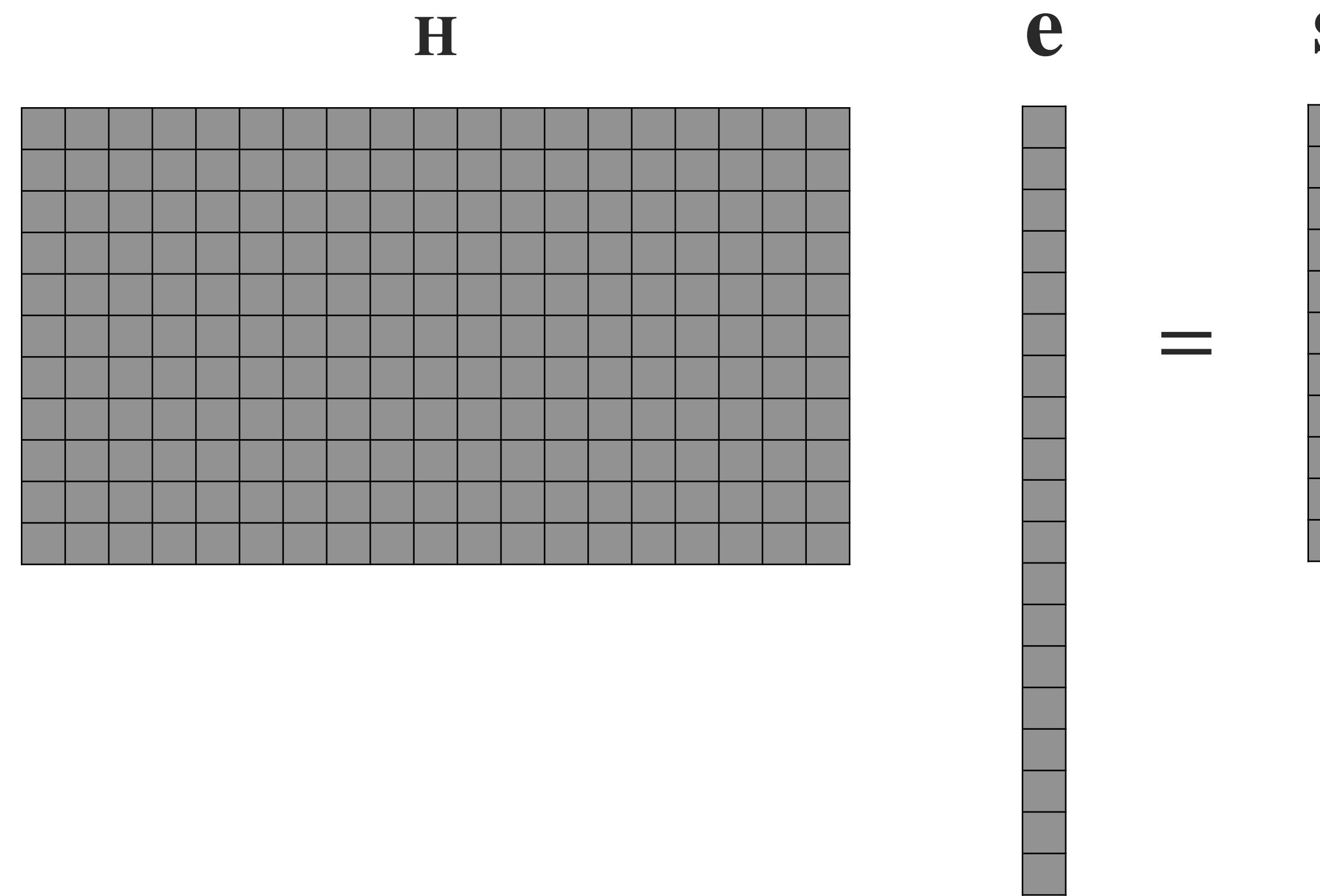
→ Pick any group of t columns of H , add them and compare with s .

Cost: $\binom{n}{t}$ sums of t columns.

Prange's attack



Prange's attack

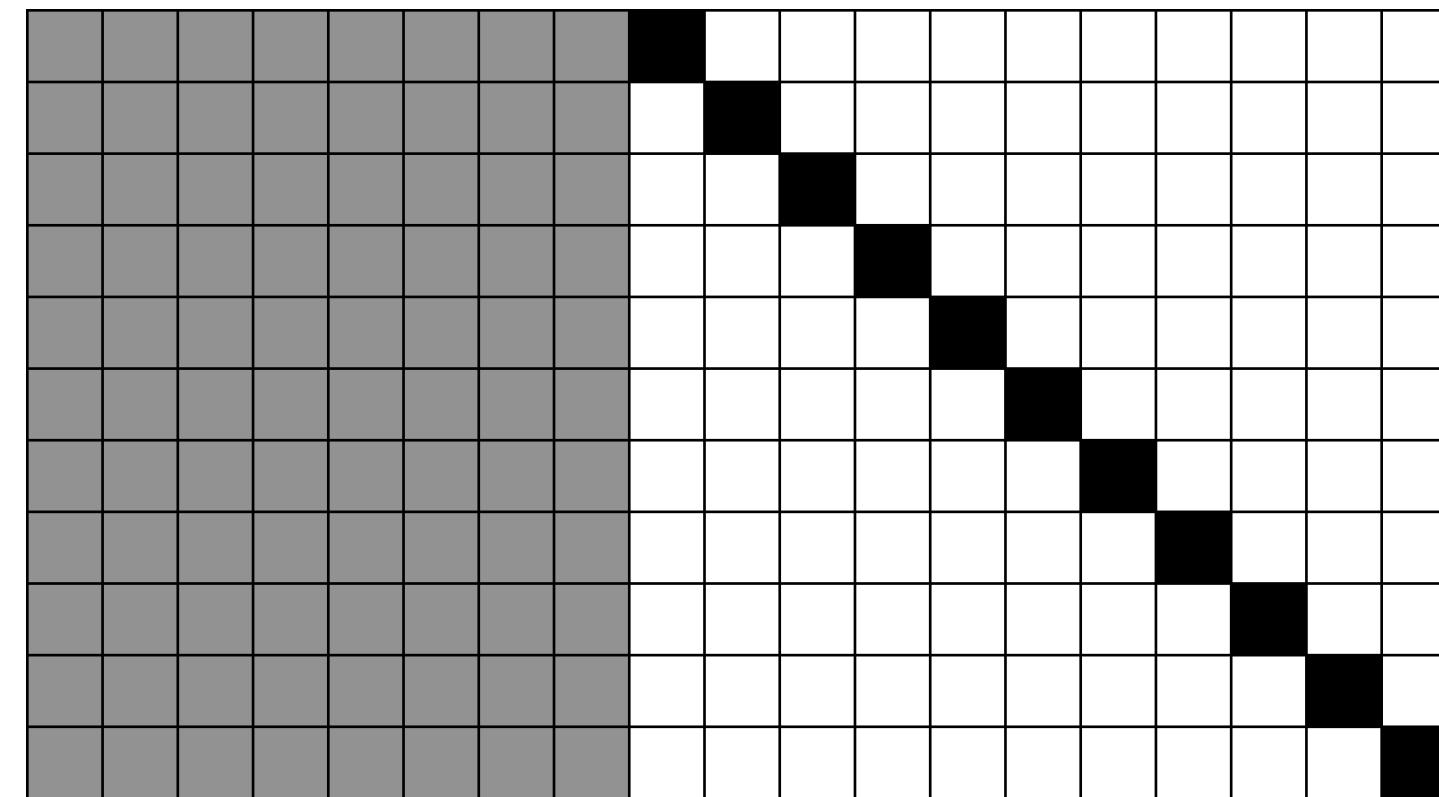
$$\mathbf{H} \quad \mathbf{e} = \mathbf{s}$$


A diagram illustrating a linear equation. At the top, the letter **H** is centered above a 16x16 grid of small squares, representing a matrix. To the right of an equals sign (=) is a double bar symbol (||) above a vertical vector **e**, which is a 16x1 column of small squares. To the right of the double bar is another vertical vector **s**, also a 16x1 column of small squares.

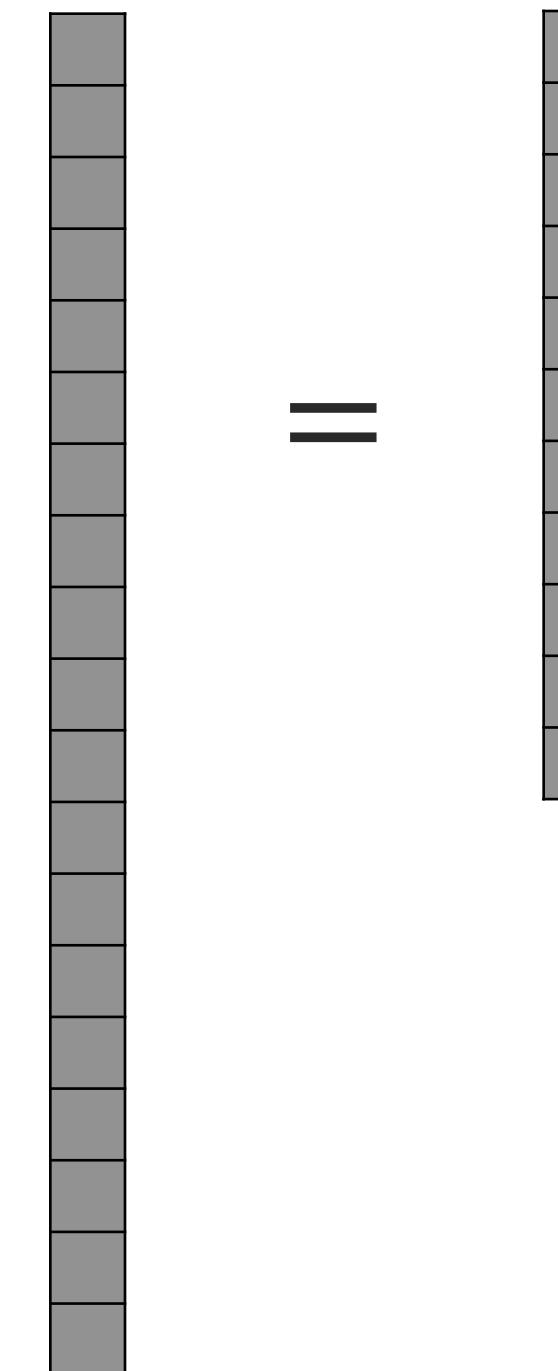
→ Permute **H** and bring to systematic form.

Prange's attack

$$\mathbf{H}' = \mathbf{UHP}$$



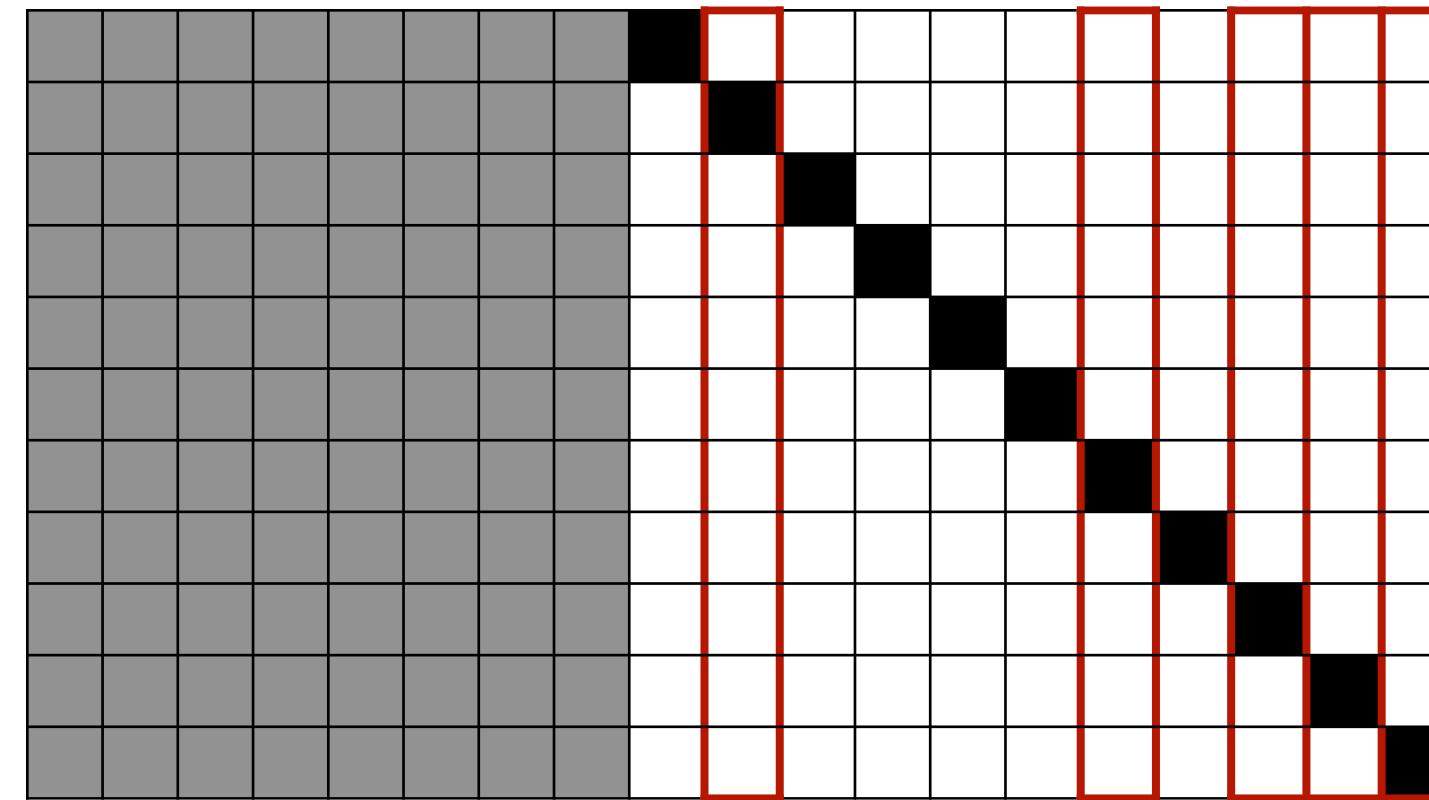
$$\mathbf{e} \quad \mathbf{s}' = \mathbf{Us}$$



→ Permute \mathbf{H} and bring to systematic form.

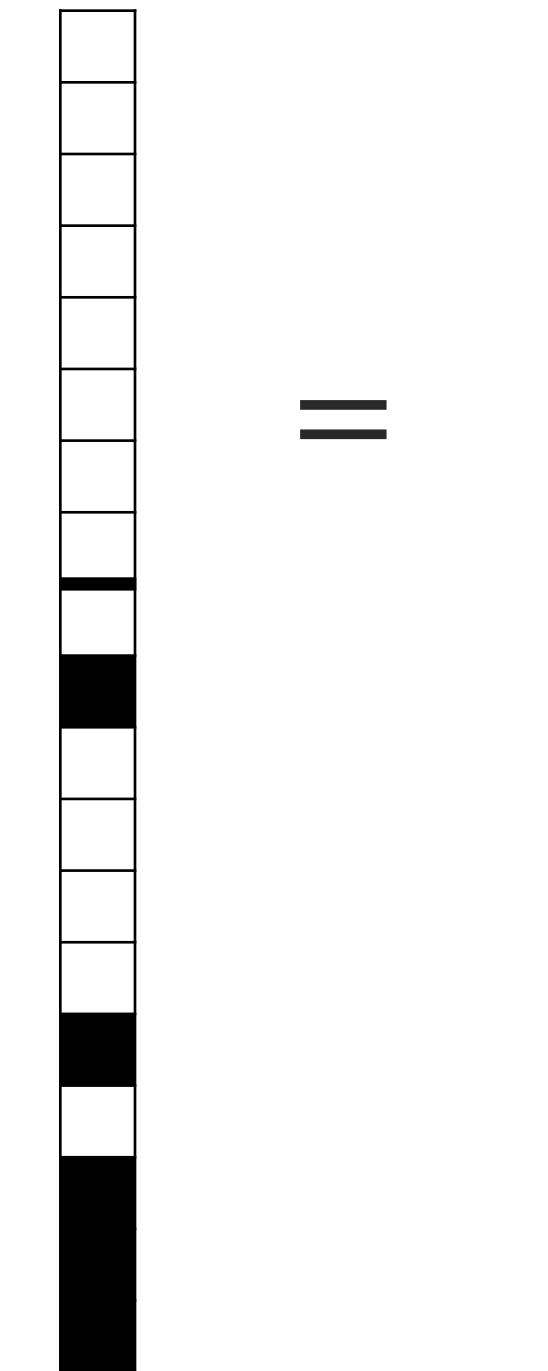
Prange's attack

$$\mathbf{H}' = \mathbf{UHP}$$



$$\mathbf{e}'$$

$$\mathbf{s}' = \mathbf{Us}$$

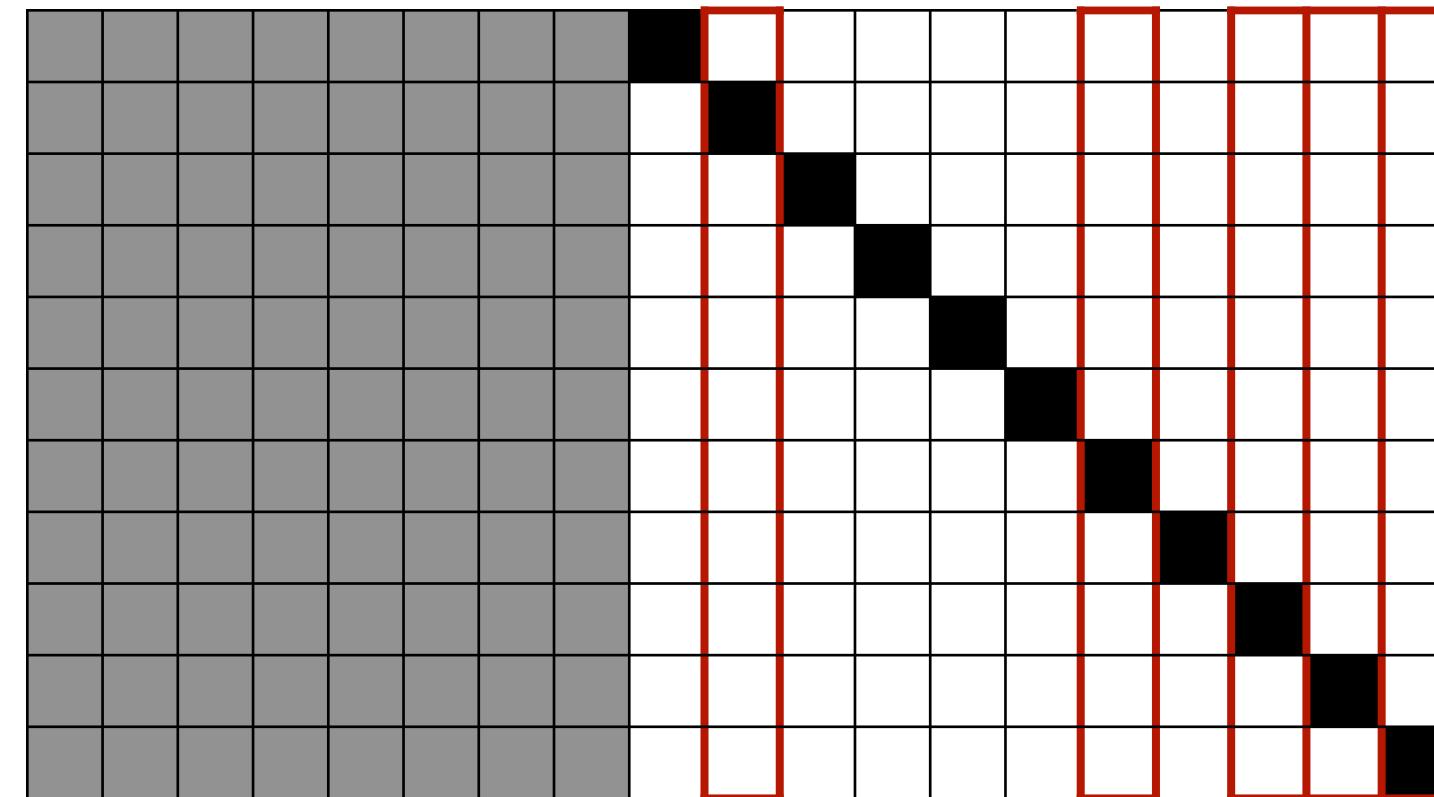


→ Permute \mathbf{H} and bring to systematic form.

Suppose that all t errors are in the identity (right) part. Then $\mathbf{e}' = (000\dots) \parallel \mathbf{Us}$ and $\text{wt}(\mathbf{Us}) = t$.

Prange's attack

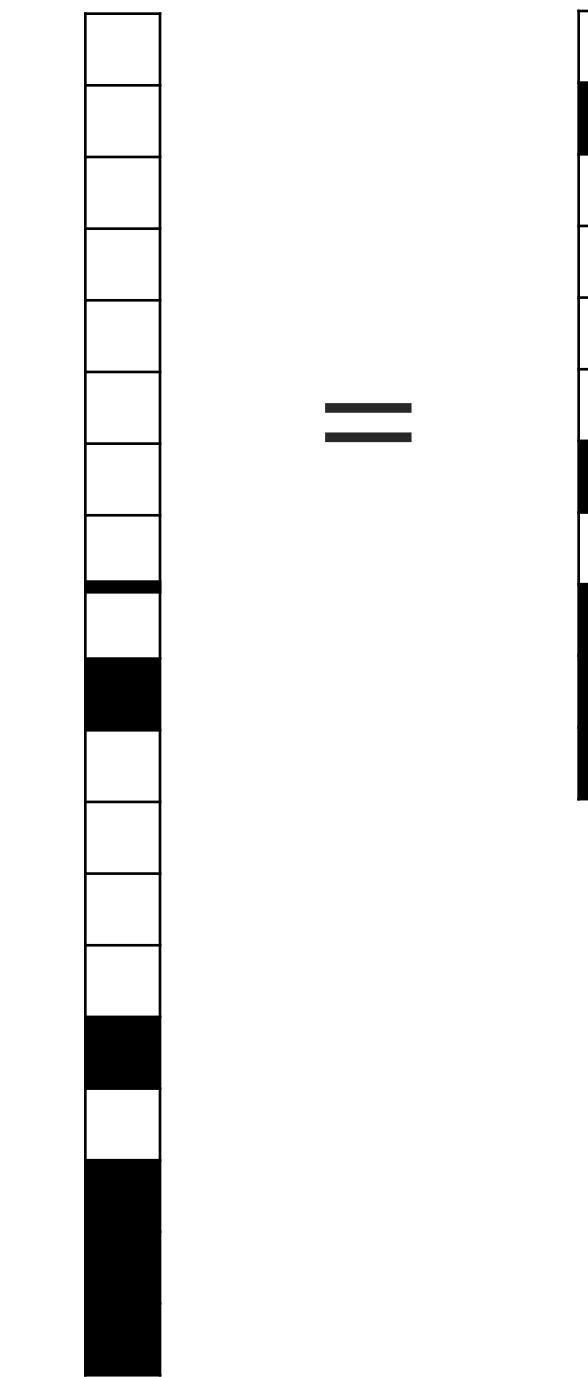
$$\mathbf{H}' = \mathbf{UHP}$$



$$\mathbf{e}'$$

$$=$$

$$\mathbf{s}' = \mathbf{Us}$$



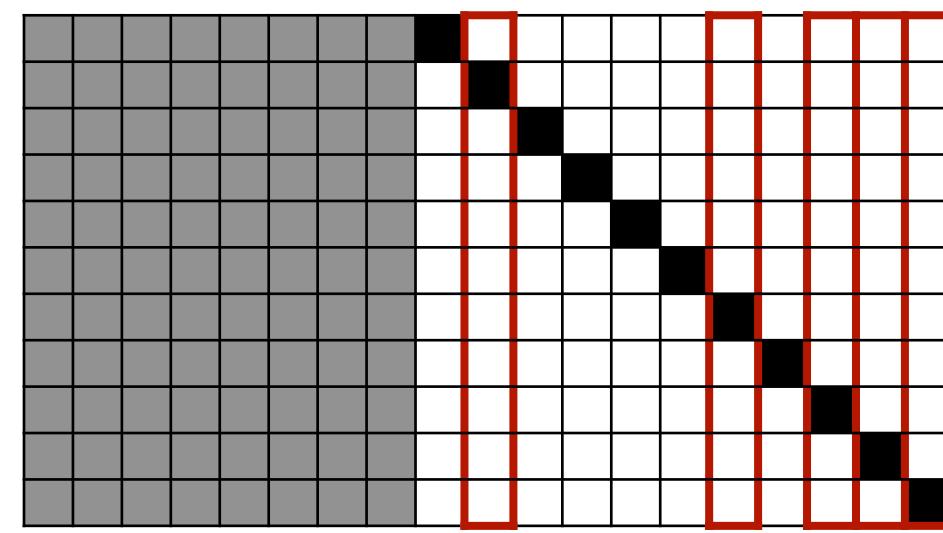
→ Permute \mathbf{H} and bring to systematic form.

Suppose that all t errors are in the identity (right) part. Then $\mathbf{e}' = (000\dots) \parallel \mathbf{Us}$ and $\text{wt}(\mathbf{Us}) = t$.

→ If $\text{wt}(\mathbf{Us}) = t$, then output unpermuted version of \mathbf{e}' .

→ Else, return to the first step and rerandomize: choose a new permutation.

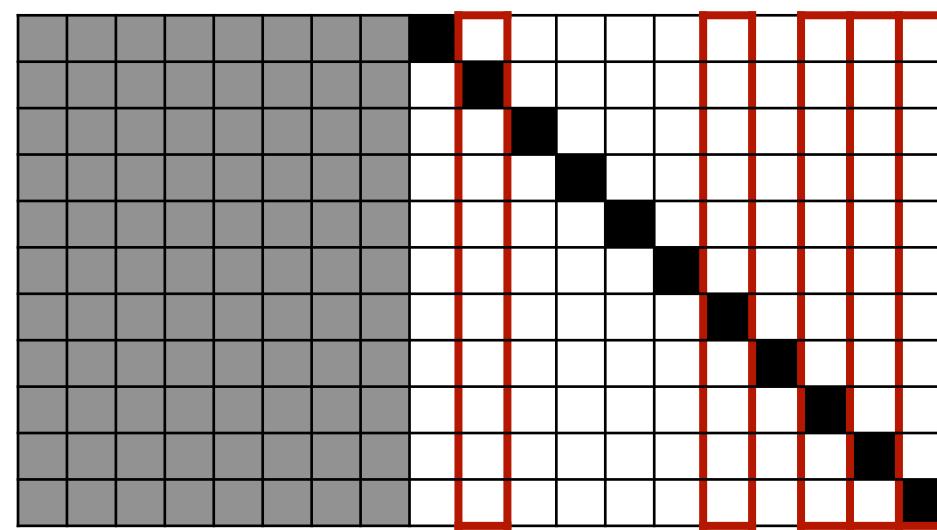
Prange's attack: complexity



$$\begin{array}{c} \text{ } \\ \text{ } \end{array} = \begin{array}{c} \text{ } \\ \text{ } \end{array}$$

```
-----  
→ Permute H and bring to systematic form.  
→ If  $\text{wt}(\mathbf{Us}) = t$ , then output unpermuted version of e.  
→ Else, return to the first step and rerandomize: choose a new permutation.
```

Prange's attack: complexity



$$\begin{array}{c} \text{---} \\ \text{---} \end{array} = \begin{array}{c} \text{---} \\ \text{---} \end{array}$$

- Permute \mathbf{H} and bring to systematic form.
- If $\text{wt}(\mathbf{Us}) = t$, then output unpermuted version of \mathbf{e} .
- Else, return to the first step and rerandomize: choose a new permutation.



Probability that we are in the correct configuration:

$$\frac{\binom{n-k}{t}}{\binom{n}{t}}.$$

All errors are in
the identity part

Prange's attack: complexity

$$\begin{array}{c} \text{ } \\ \text{ } \end{array} = \begin{array}{c} \text{ } \\ \text{ } \end{array}$$

→ Permute \mathbf{H} and bring to systematic form.
→ If $\text{wt}(\mathbf{Us}) = t$, then output unpermuted version of \mathbf{e} .
→ Else, return to the first step and rerandomize: choose a new permutation.

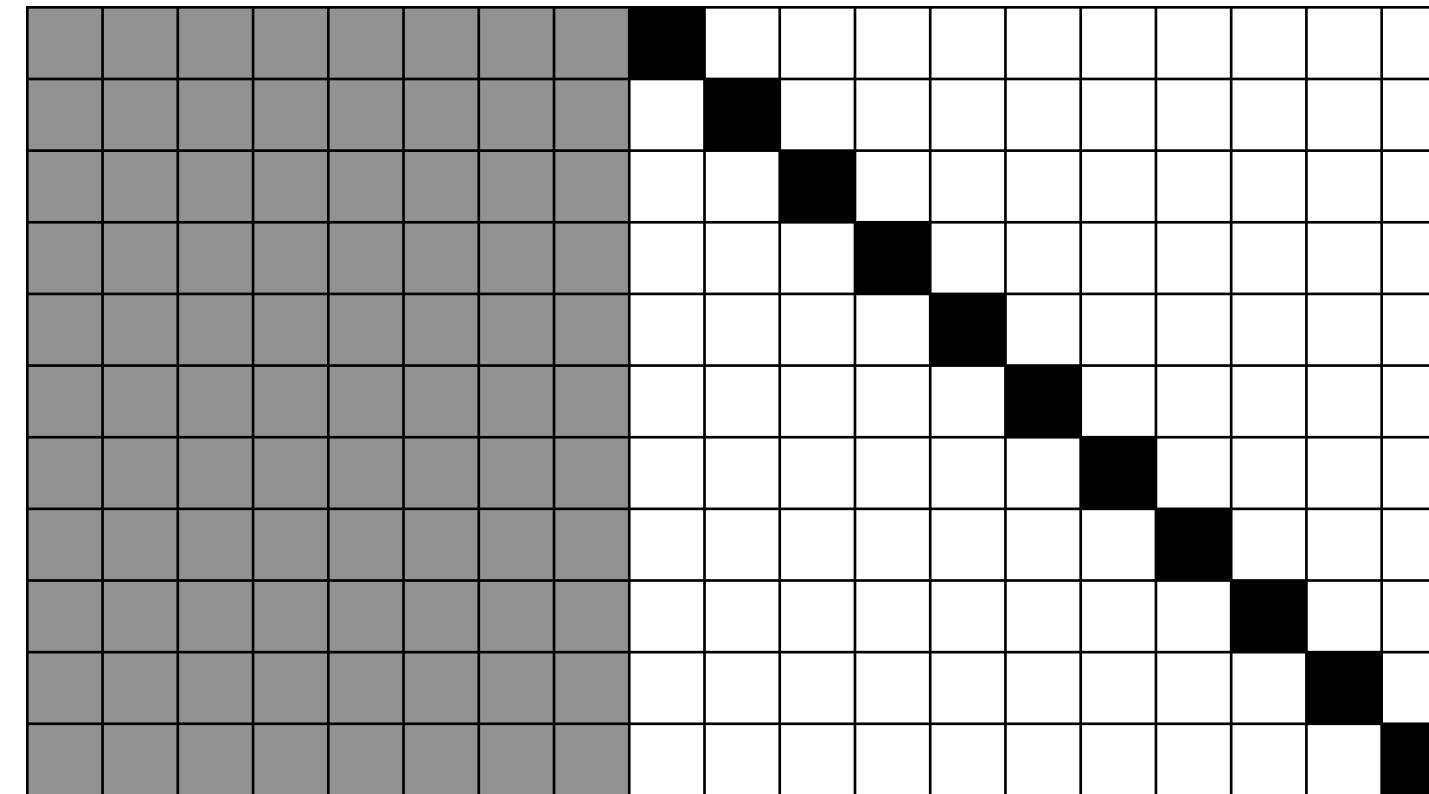
All errors are in
the identity part

Probability that we are in the correct configuration: $\frac{\binom{n-k}{t}}{\binom{n}{t}}$.

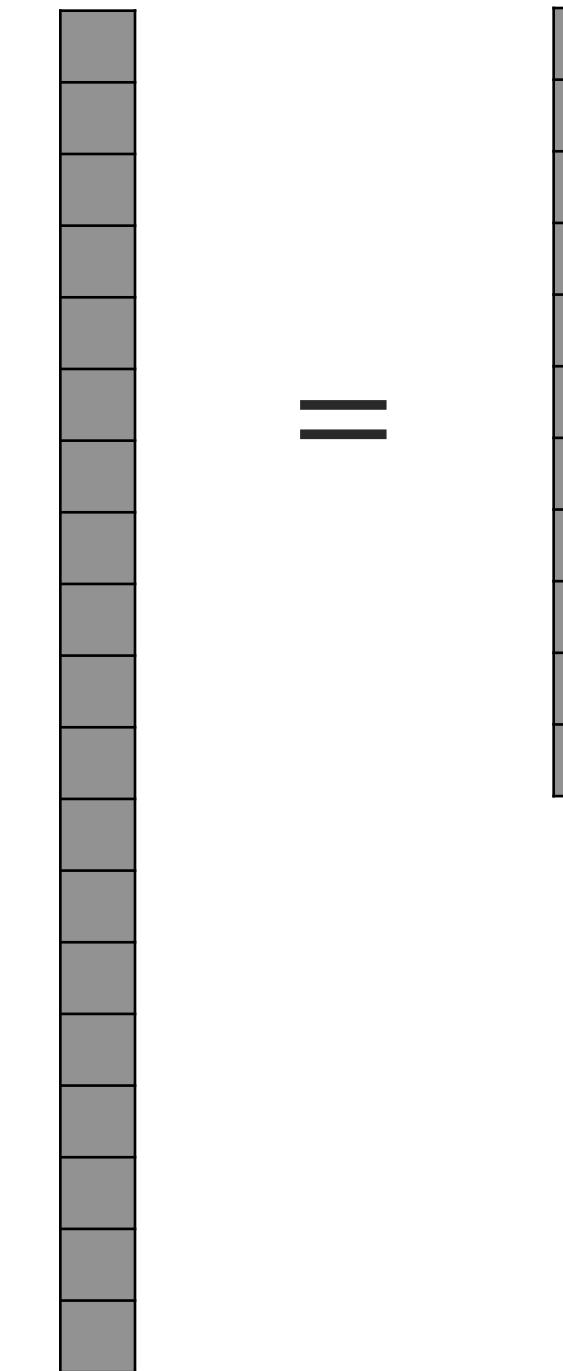
Cost: $\frac{\binom{n}{t}}{\binom{n-k}{t}}$ matrix operations.

Lee-Brickell attack

$$\mathbf{H}' = \mathbf{UHP}$$



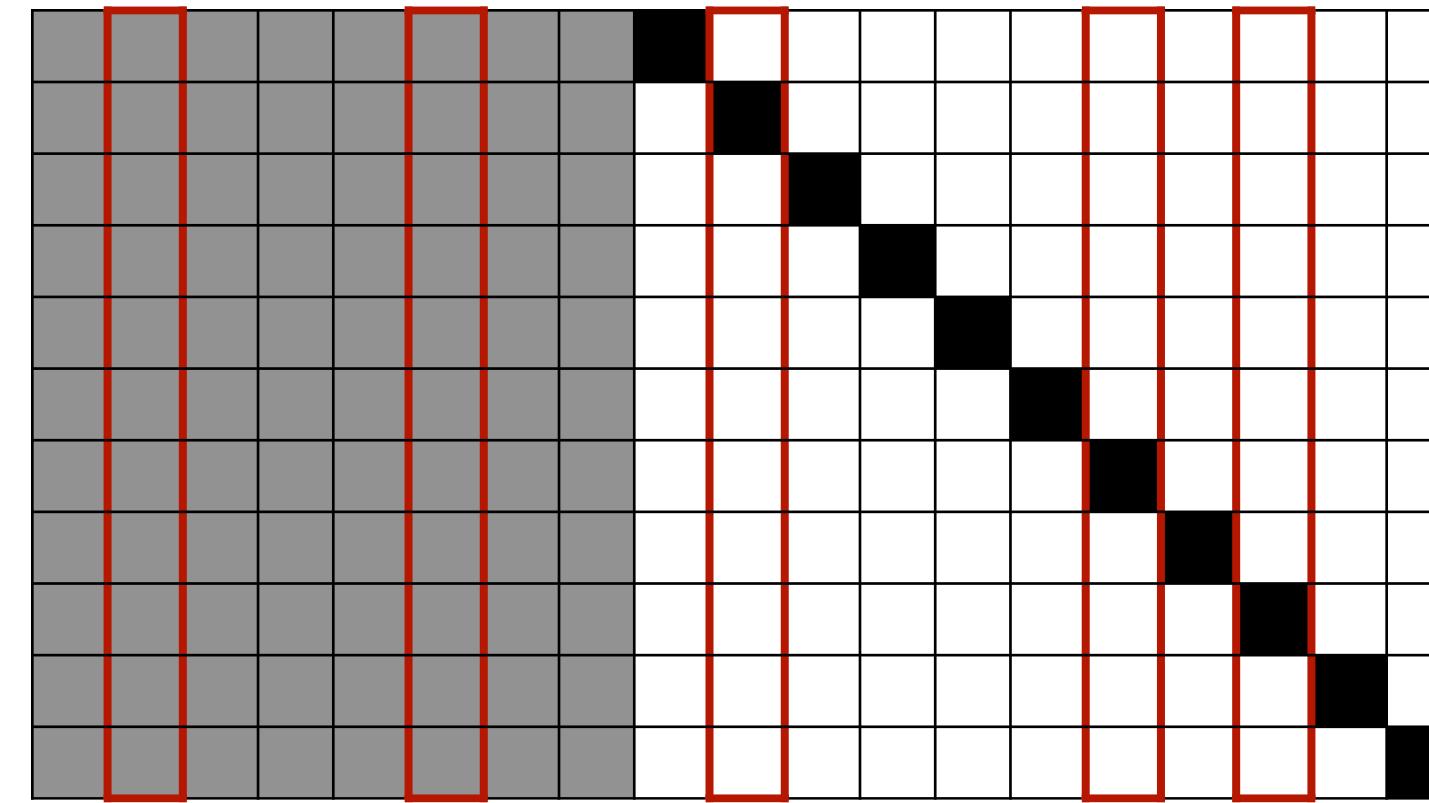
$$\mathbf{e} \quad \mathbf{s}' = \mathbf{Us}$$



→ Permute \mathbf{H} and bring to systematic form.

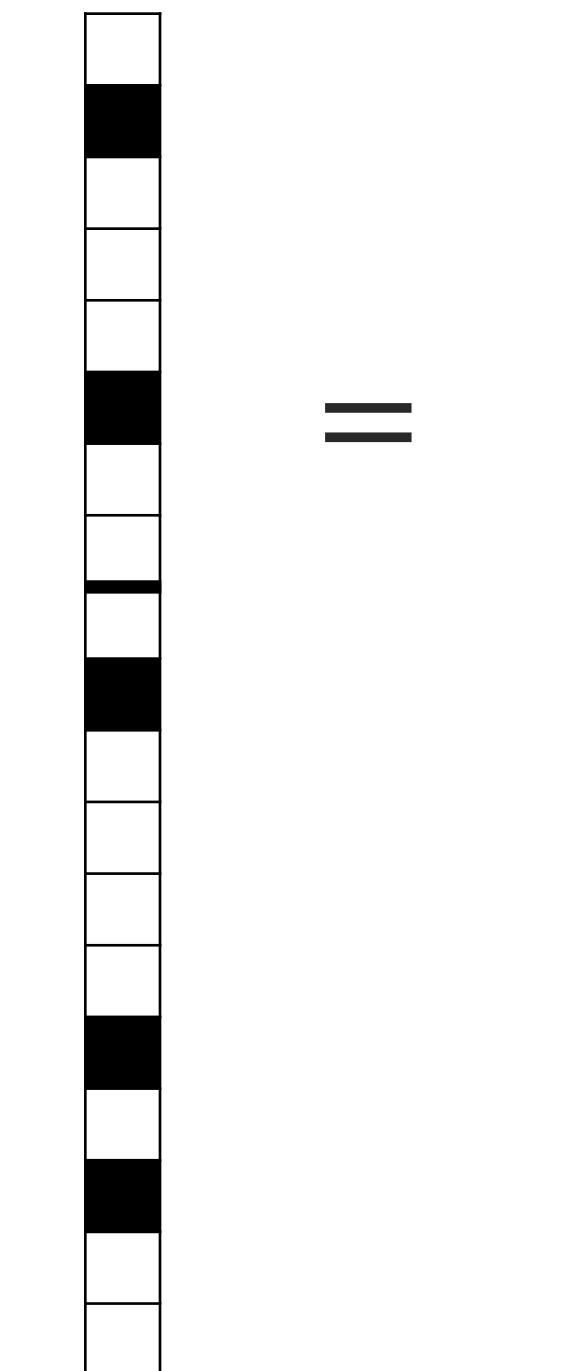
Lee-Brickell attack

$$\mathbf{H}' = \mathbf{UHP}$$



$$\mathbf{e}'$$

$$\mathbf{s}' = \mathbf{Us}$$



→ Permute \mathbf{H} and bring to systematic form.

Suppose that there are $(t - p)$ errors in the identity (right) part and p errors in the left part.

Lee-Brickell attack

$$\mathbf{H}' = \mathbf{UHP}$$
$$\mathbf{e}' \quad \mathbf{H}_2 \quad \mathbf{H}_6 \quad \mathbf{s}' + \mathbf{Qp}$$

→ Permute \mathbf{H} and bring to systematic form.

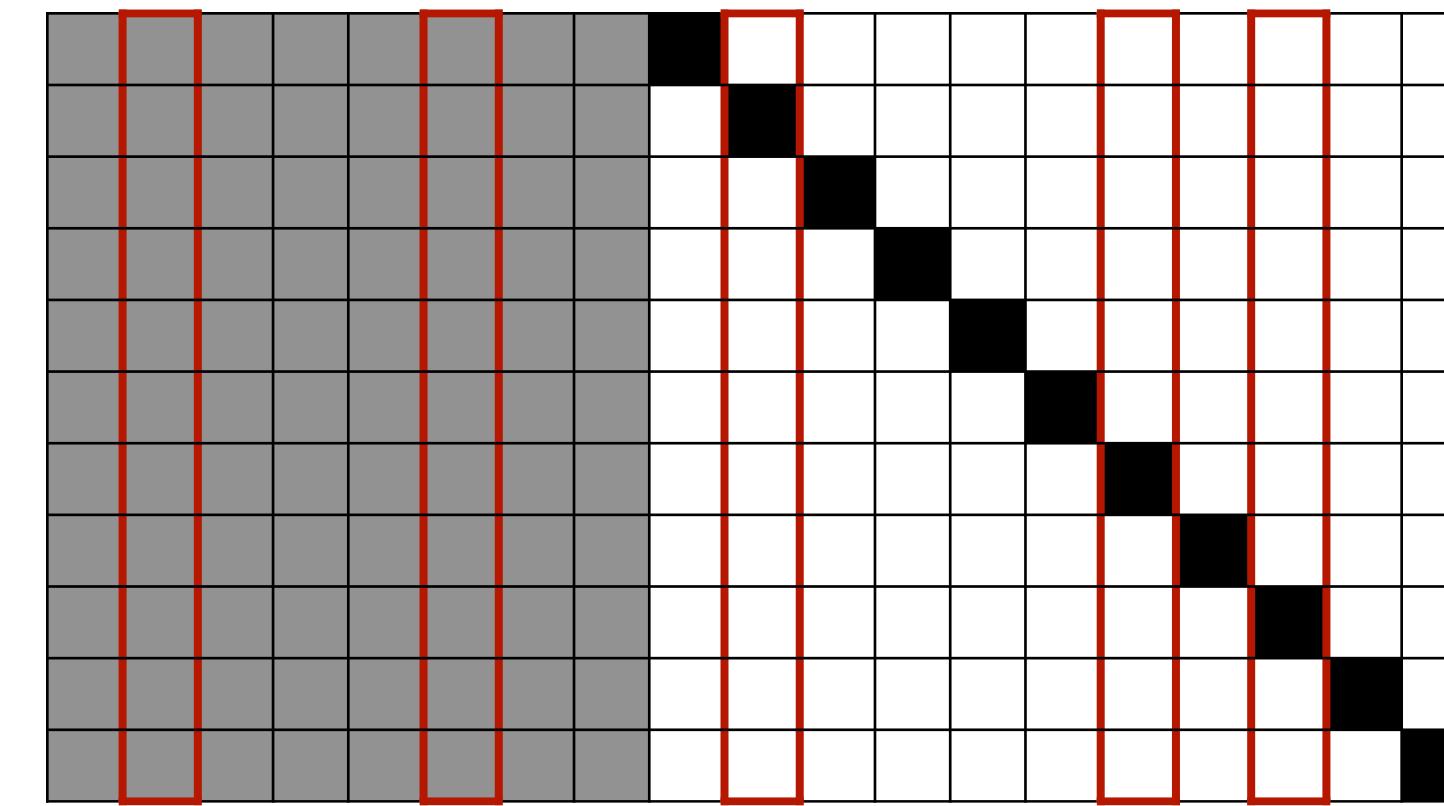
Suppose that there are $(t - p)$ errors in the identity (right) part and p errors in the left part.

Then, \mathbf{s}' is random-looking, but \mathbf{s}' summed with the error columns on the left has weight $t - p$:

$$\text{wt}(\mathbf{s}' + \mathbf{Qp}) = t - p.$$

Lee-Brickell attack

$$\mathbf{H}' = \mathbf{UHP}$$



$$\mathbf{e}'$$

$$+$$

$$\mathbf{H}_2$$

$$+$$

$$\mathbf{H}_6$$

$$=$$

$$\mathbf{s}' + \mathbf{Qp}$$

Let \mathbf{p} be a vector chosen from
 $\{\mathbf{p} \in \mathbb{F}^k \mid \text{wt}(\mathbf{p}) = p\}$

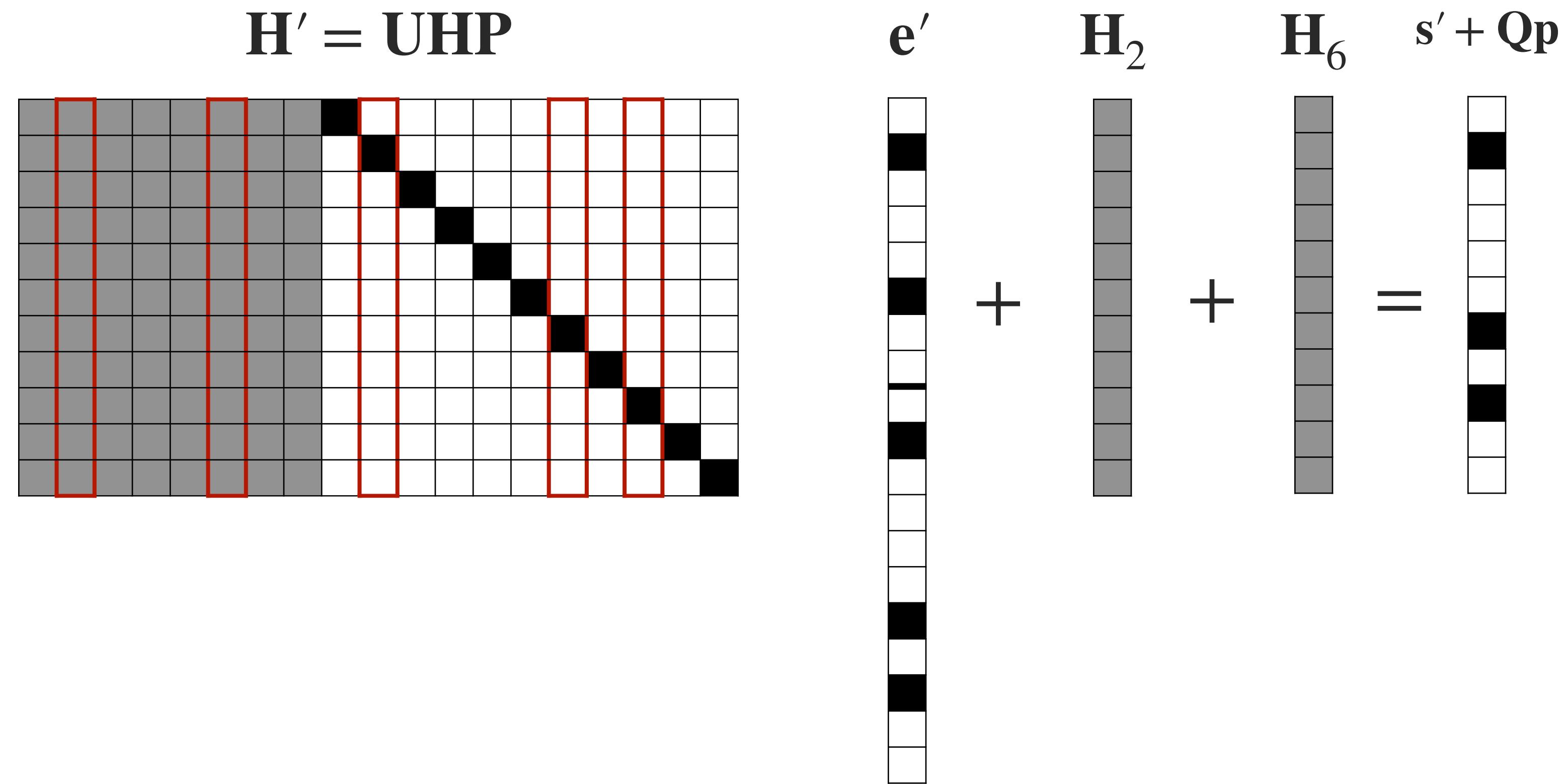
→ Permute \mathbf{H} and bring to systematic form.

Suppose that there are $(t - p)$ errors in the identity (right) part and p errors in the left part.

Then, \mathbf{s}' is random-looking, but \mathbf{s}' summed with the error columns on the left has weight $t - p$:

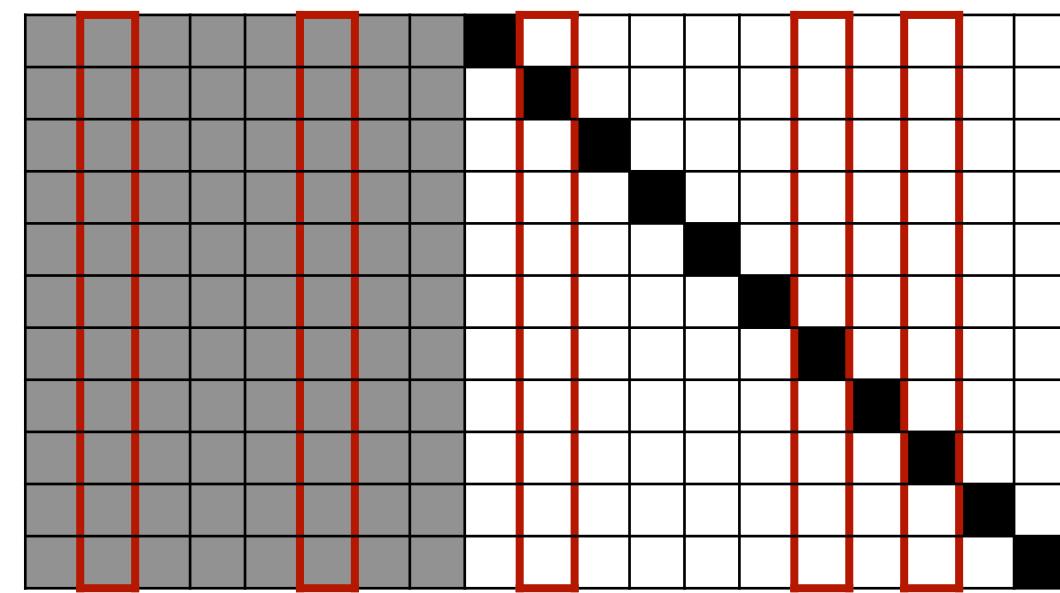
$$\text{wt}(\mathbf{s}' + \mathbf{Qp}) = t - p.$$

Lee-Brickell attack



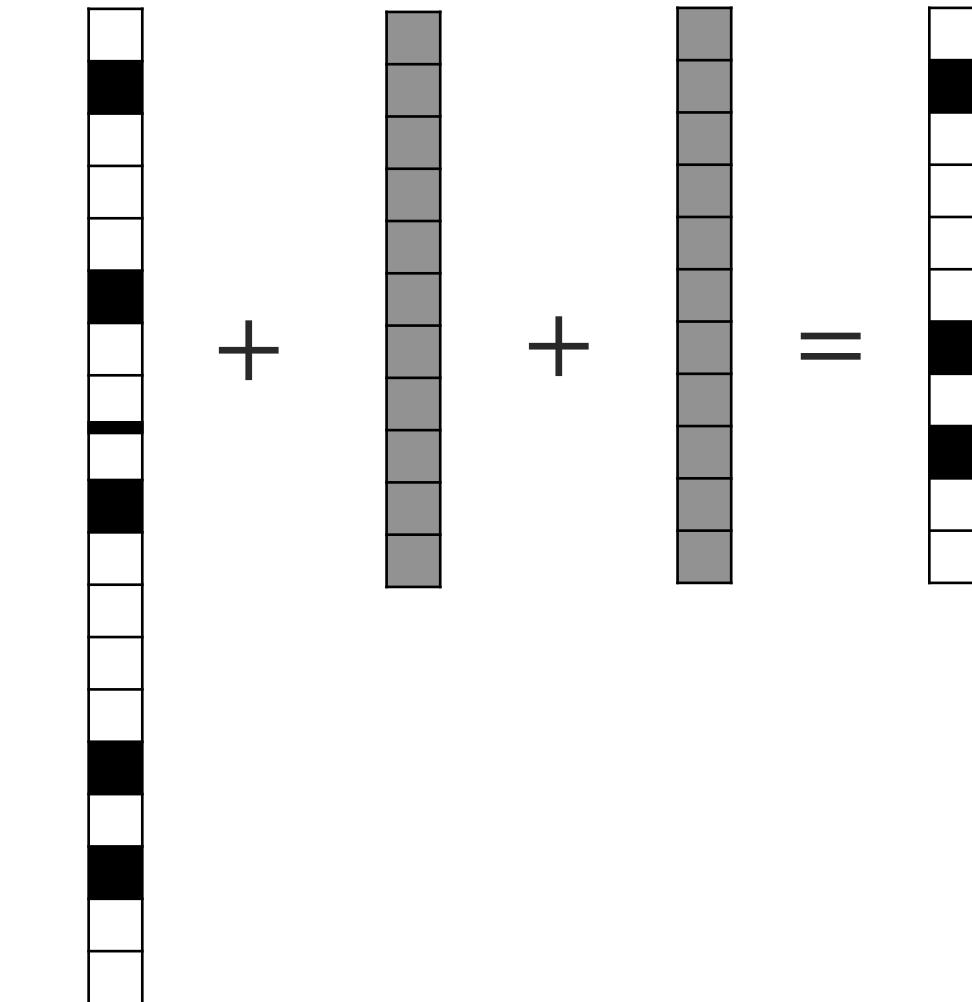
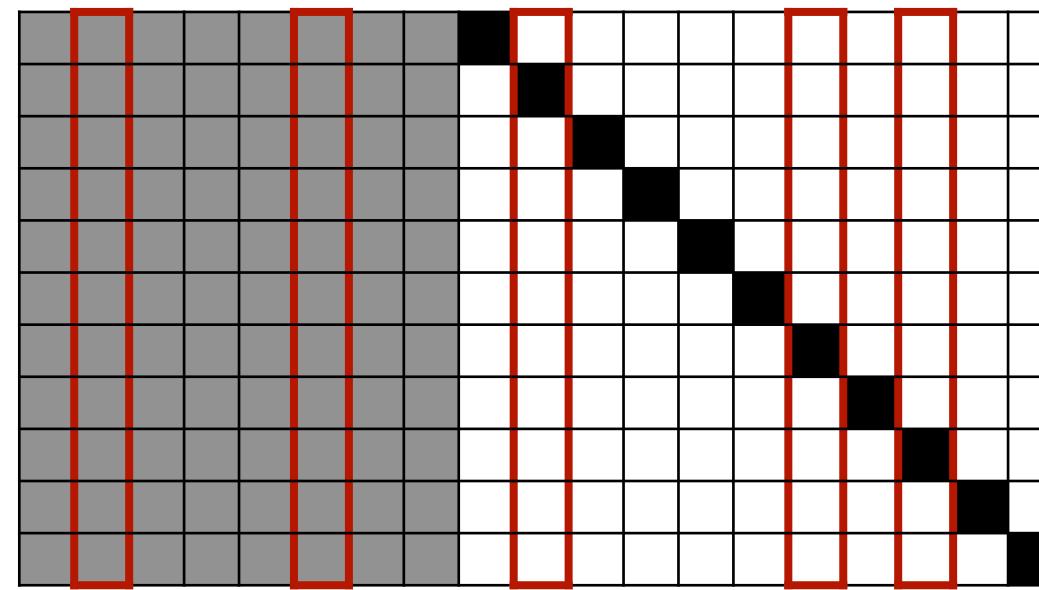
- Permute H and bring to systematic form.
- Pick p of the columns on the left and compute their sum: Qp .
- If $\text{wt}(s' + Qp) = t - p$ then put $e' = p || (s' + Qp)$. Output unpermuted version of e .
- Else, return to the second step to choose another subset of columns from Q , or return to the first step and rerandomize.

Lee-Brickell attack: complexity



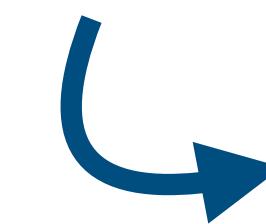
$$\begin{array}{c} \text{[Vertical vector]} \\ + \\ \text{[Vertical vector]} \\ + \\ \text{[Vertical vector]} \\ = \\ \text{[Vertical vector]} \end{array}$$

Lee-Brickell attack: complexity



$t - p$ errors are in
the identity part

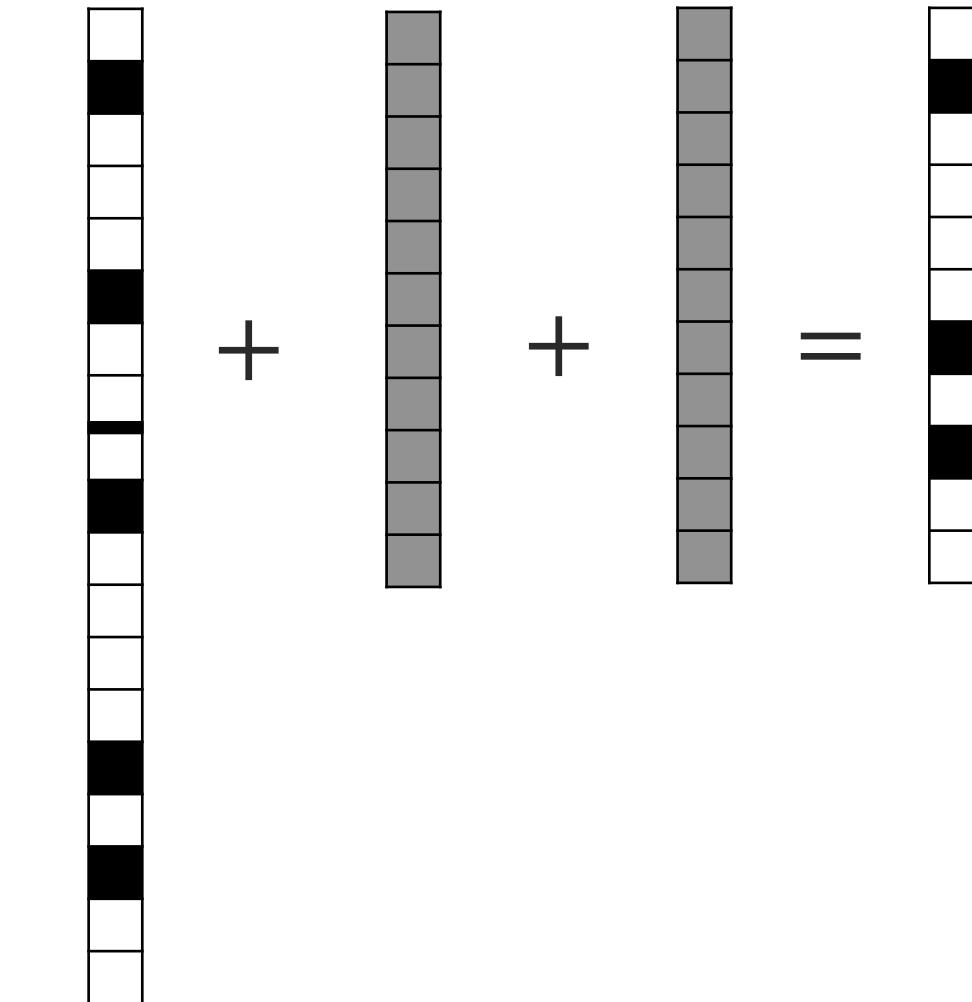
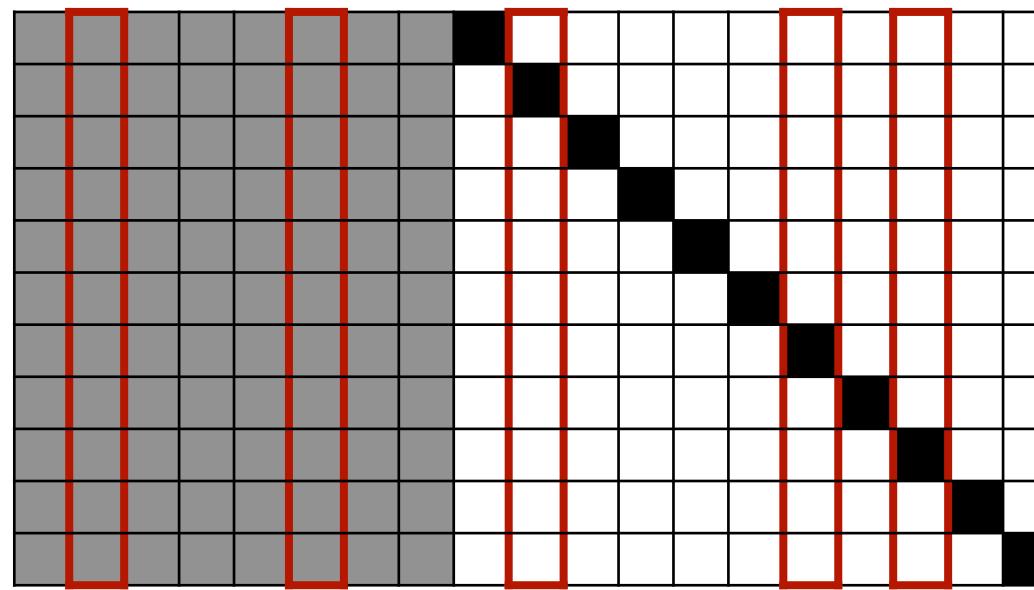
p errors are in the
left part



Probability that we are in the correct configuration:

$$\frac{\binom{n-k}{t-p} \binom{k}{p}}{\binom{n}{t}}.$$

Lee-Brickell attack: complexity



$t - p$ errors are in the identity part

p errors are in the left part

Probability that we are in the correct configuration:
$$\frac{\binom{n-k}{t-p} \binom{k}{p}}{\binom{n}{t}}.$$

Cost:
$$\frac{\binom{n}{t}}{\binom{n-k}{t-p} \binom{k}{p}}$$
 matrix operations + $\binom{k}{p}$ column additions.

Leon's attack

$$\mathbf{H}' = \mathbf{UHP}$$

$$\mathbf{e}' \quad \mathbf{H}_2 \quad \mathbf{H}_6 \quad \mathbf{s}' + \mathbf{Qp}$$

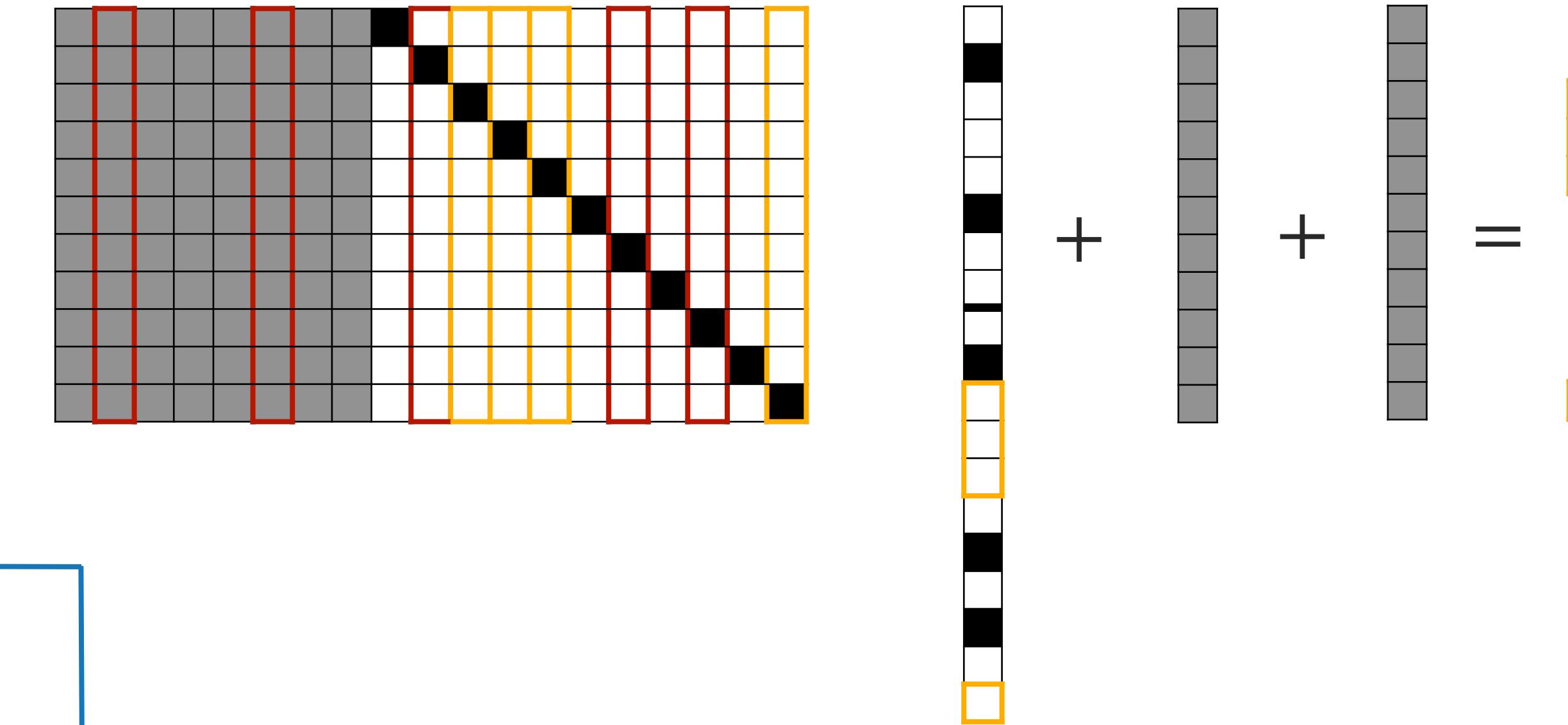
$$\begin{array}{c} \mathbf{e}' \\ + \\ \mathbf{H}_2 \\ + \\ \mathbf{H}_6 \\ = \\ \mathbf{s}' + \mathbf{Qp} \end{array}$$



Since $\mathbf{s}' + \mathbf{Qp}$ should be of low weight, we check instead if an arbitrary subset of l rows are all zero.

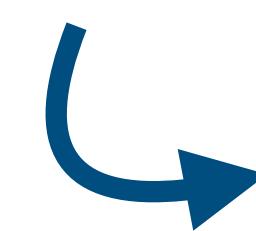
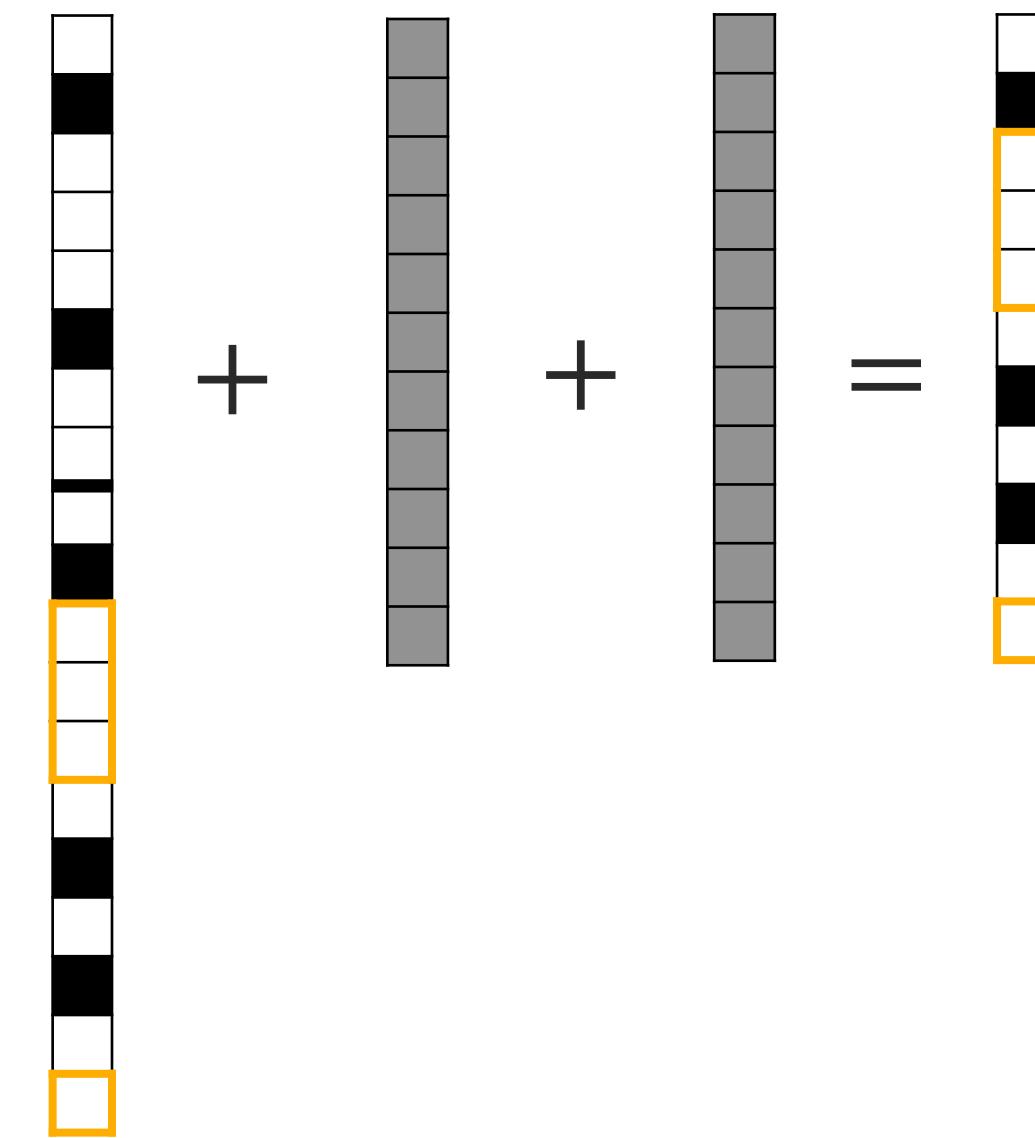
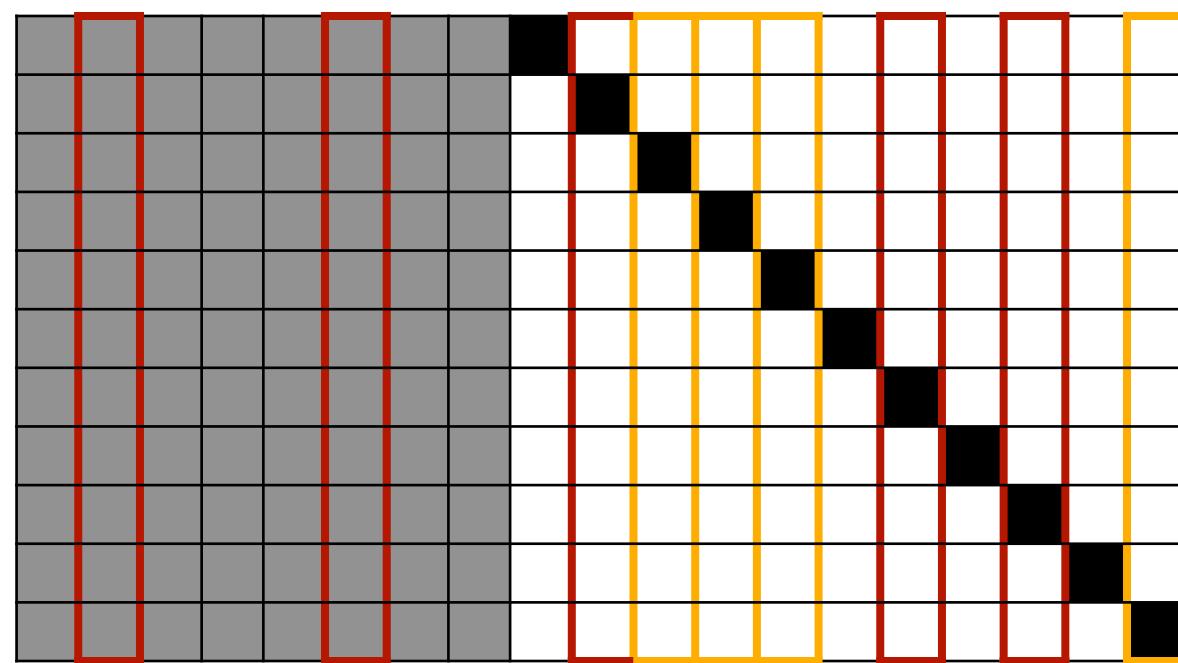
Leon's attack

\mathbf{H}_L denotes the matrix consisting of the rows of \mathbf{H} indexed by L



- Pick a subset L of l rows: \mathbf{H}_L .
- Permute \mathbf{H} and bring to systematic form (then $\mathbf{H}_L = (\mathbf{Q}_L \ \mathbf{I}_L)$).
- Pick p of the columns on the left and compute their sum: $\mathbf{Q}_L \mathbf{p}$.
- If $\text{wt}(\mathbf{s}'_L + \mathbf{Q}_L \mathbf{p}) = 0$
 - If $\text{wt}(\mathbf{s}' + \mathbf{Q} \mathbf{p}) = t - p$ then put $\mathbf{e}' = \mathbf{p} \parallel (\mathbf{s}' + \mathbf{Q} \mathbf{p})$. Output unpermuted version of \mathbf{e} .
 - Else, return to the third step to choose another subset of columns from \mathbf{Q} , or return to the second step and rerandomize.
- Else, return to the third step to choose another subset of columns from \mathbf{Q} , or return to the second step and rerandomize.

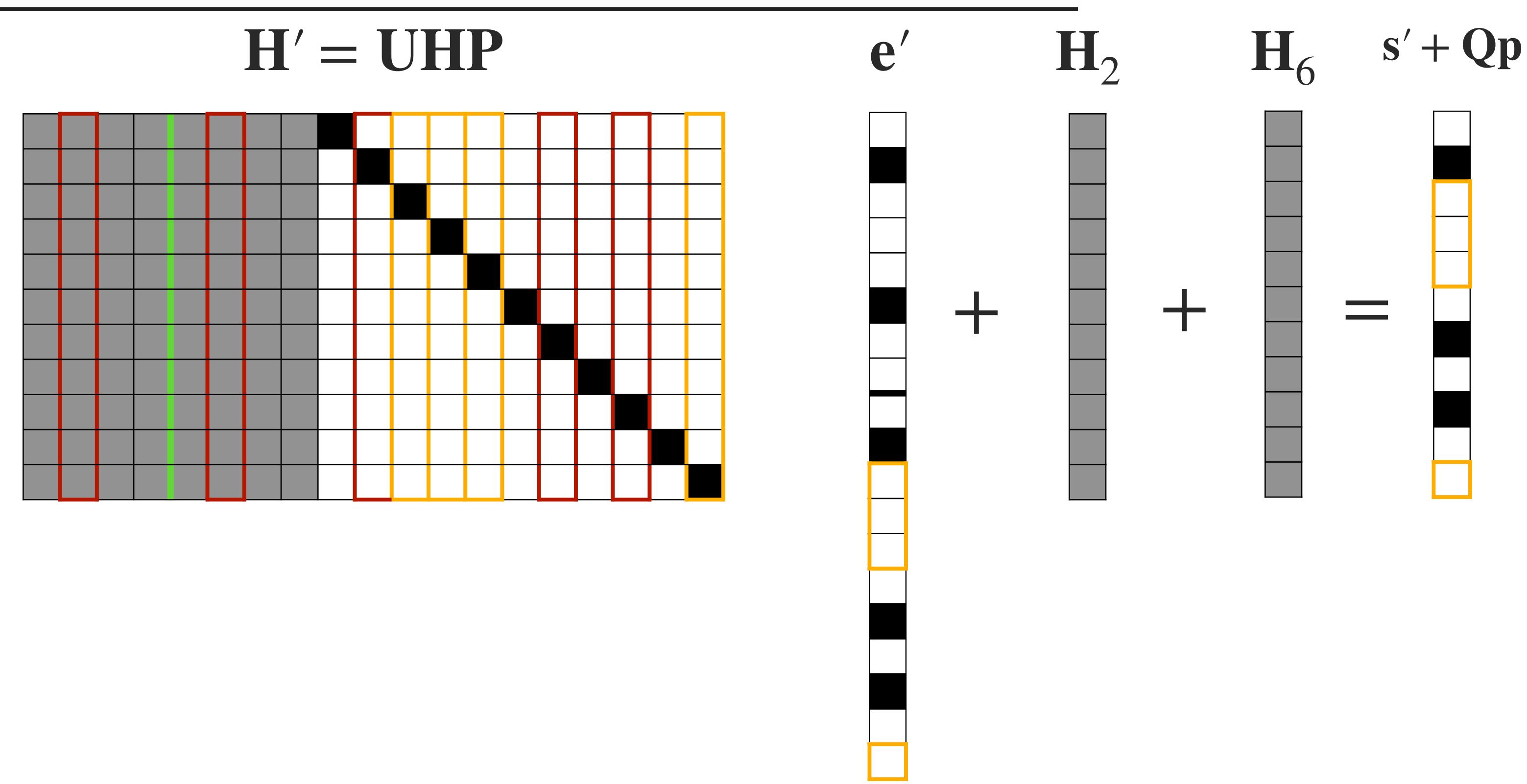
Leon's attack: complexity



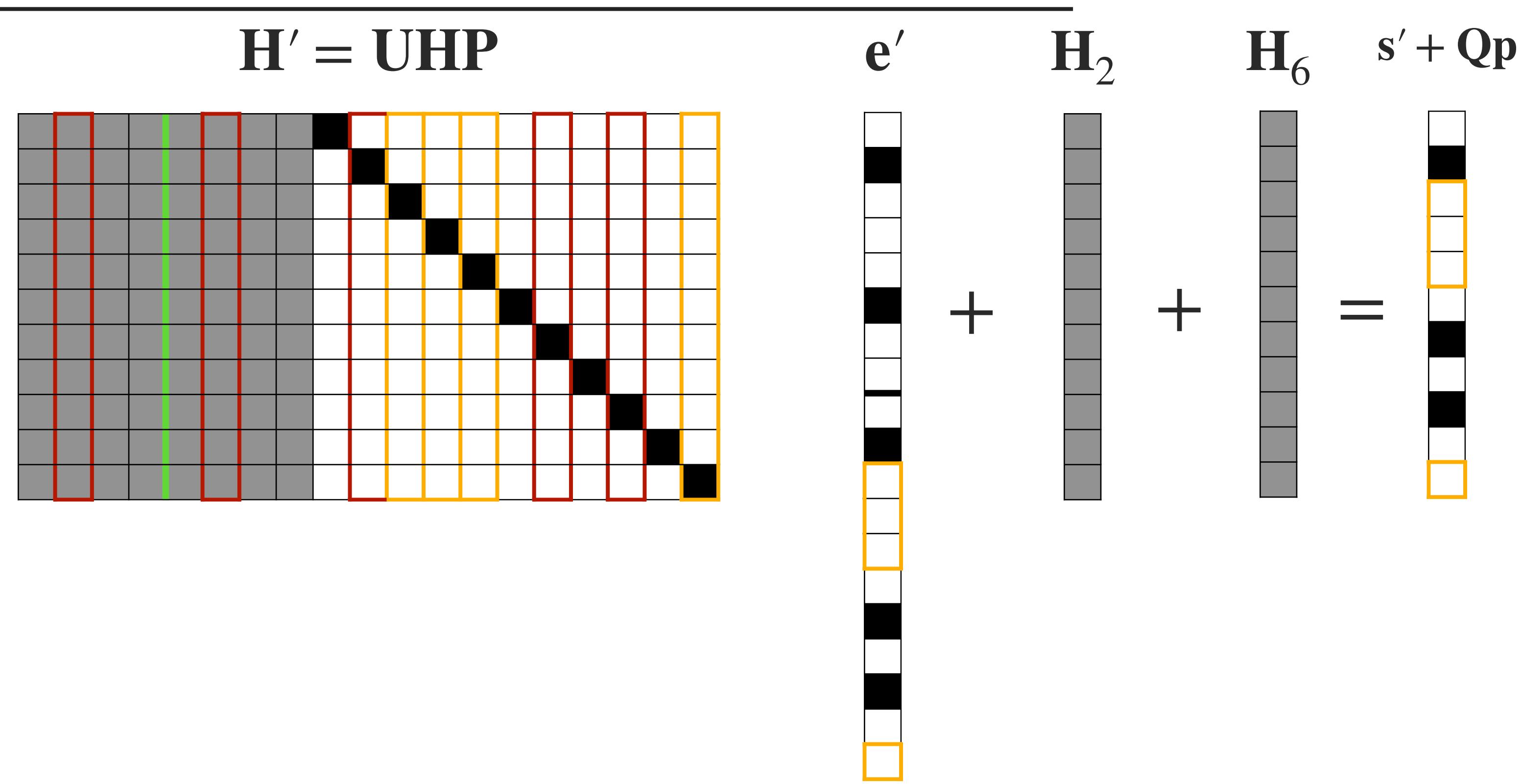
Probability that we are in the correct configuration:

$$\frac{\binom{n-k-l}{t-p} \binom{k}{p}}{\binom{n}{t}}.$$

Stern's attack

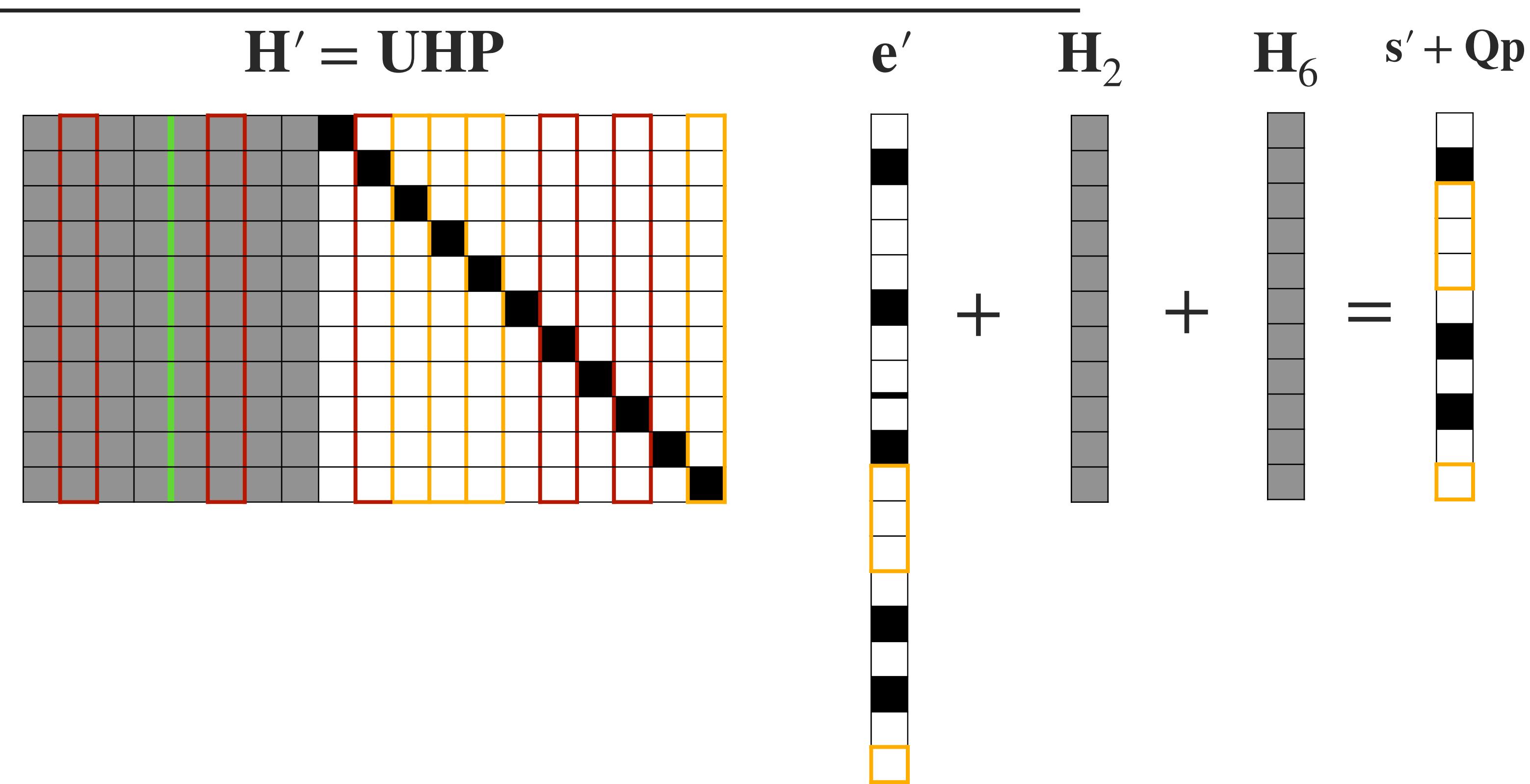


Stern's attack



Suppose that there are exactly $\frac{p}{2}$ errors in the first half of \mathbf{Q} and exactly $\frac{p}{2}$ errors in the first half of \mathbf{Q}' .

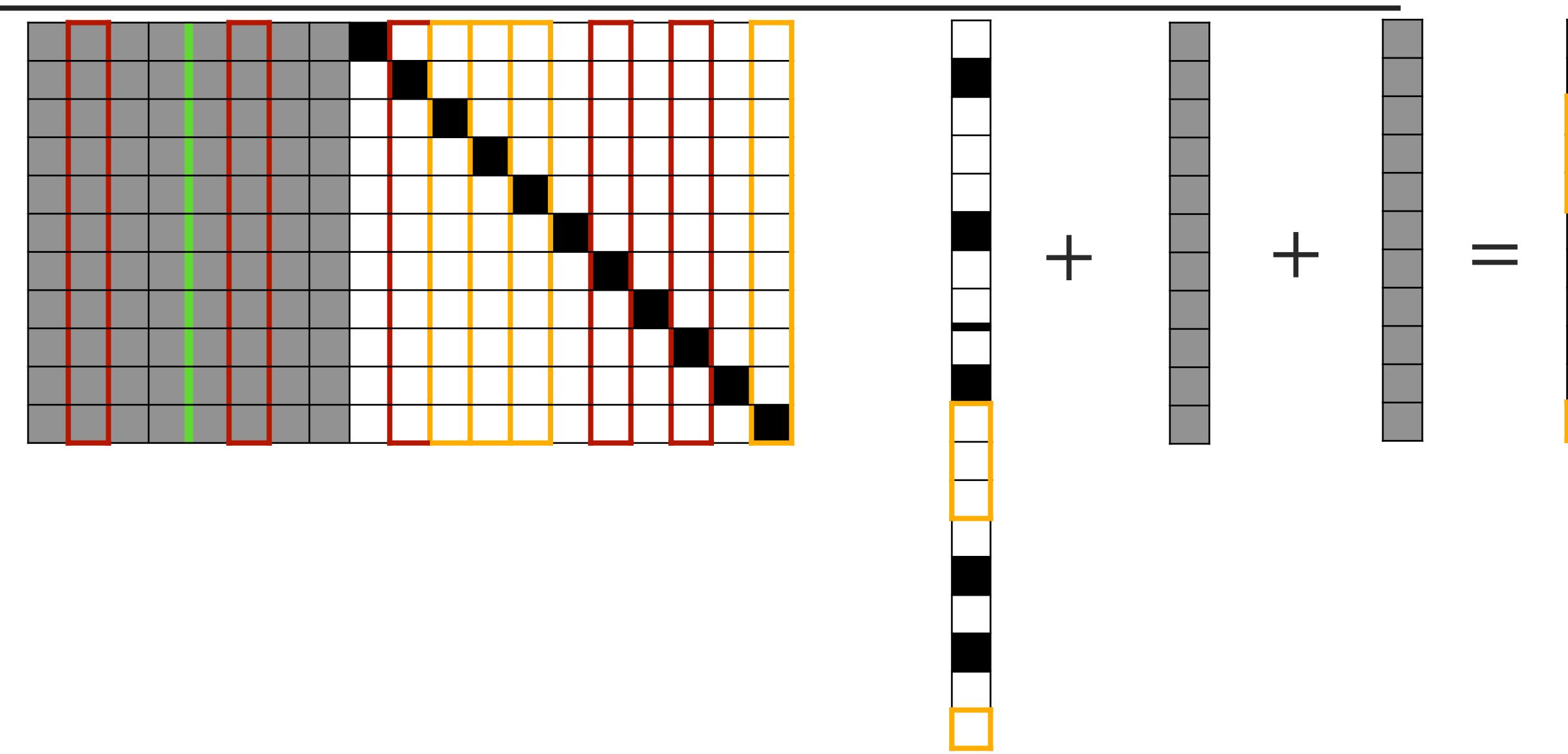
Stern's attack



Suppose that there are exactly $\frac{p}{2}$ errors in the first half of \mathbf{Q} and exactly $\frac{p}{2}$ errors in the first half of \mathbf{Q}' .

Instead of looking for an all zero subset of rows, we are looking for a **collision**.

Stern's attack

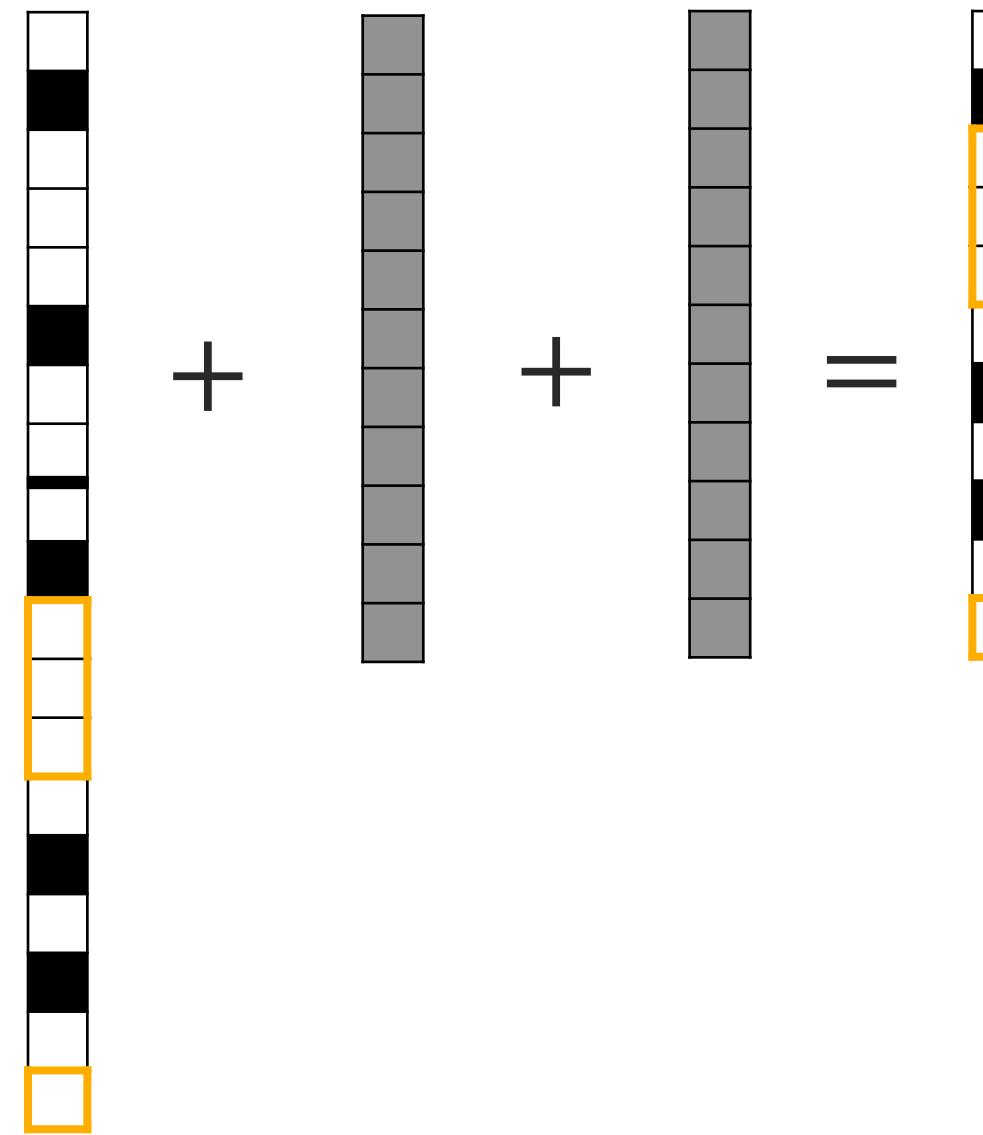
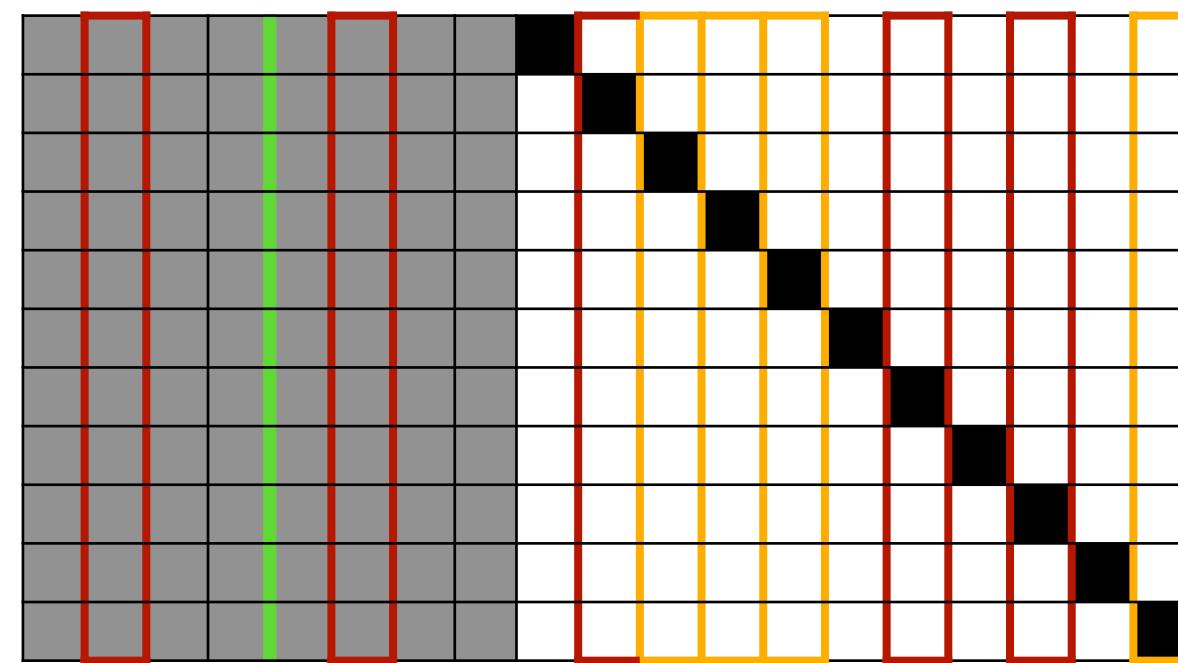


- Pick a subset L of l rows: \mathbf{H}_L .
- Permute \mathbf{H} and bring to systematic form (then $\mathbf{H}_L = (\mathbf{Q}_L \ \mathbf{I}_L)$).
- Split \mathbf{Q} into two disjoint parts: $\mathbf{Q} = (\mathbf{A} \ \mathbf{B})$.
- Build a list of vectors $(\mathbf{s}'_L + \mathbf{A}_L \mathbf{a})$ for all (many) \mathbf{a} .
- For all (many) \mathbf{b} :

 - If $\mathbf{B}_L \mathbf{b}$ collides with (is equal to) any of the vectors in the list built in the fourth step
 - If $\text{wt}(\mathbf{s}' + \mathbf{Aa} + \mathbf{Bb}) = t - p$ then put $\mathbf{e}' = \mathbf{a} \parallel \mathbf{b} \parallel (\mathbf{s}' + \mathbf{Aa} + \mathbf{Bb})$. Output unpermuted version of \mathbf{e} .
 - Else return to the second step and rerandomize.

\mathbf{a} and \mathbf{b} are vectors chosen from
 $W = \{\mathbf{w} \in \mathbb{F}_2^{k/2} \mid \text{wt}(\mathbf{w}) = \frac{p}{2}\}$

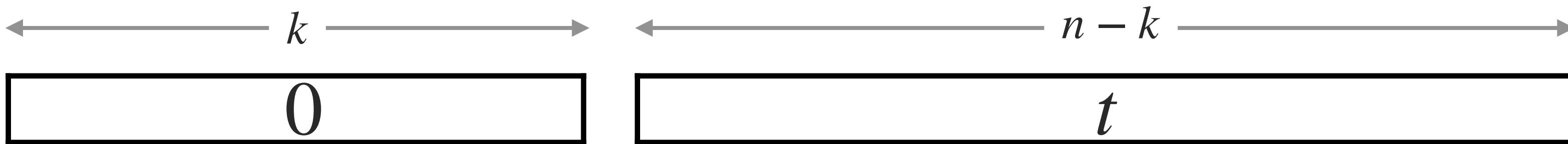
Stern's attack: complexity



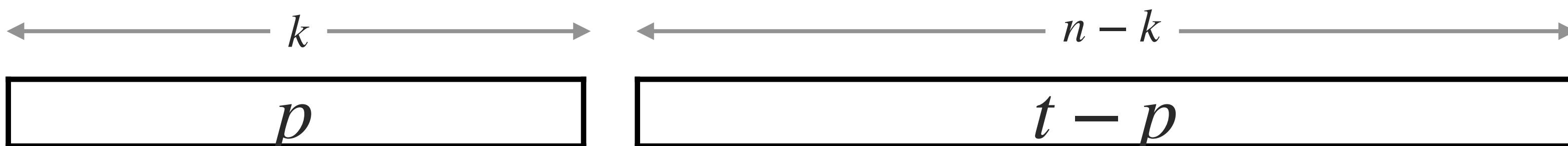
Probability that we are in the correct configuration:
$$\frac{\binom{n-k-l}{t-p} \left(\frac{k}{2}\right)^2 \left(\frac{p}{2}\right)^2}{\binom{n}{t}}.$$

ISD algorithms summary

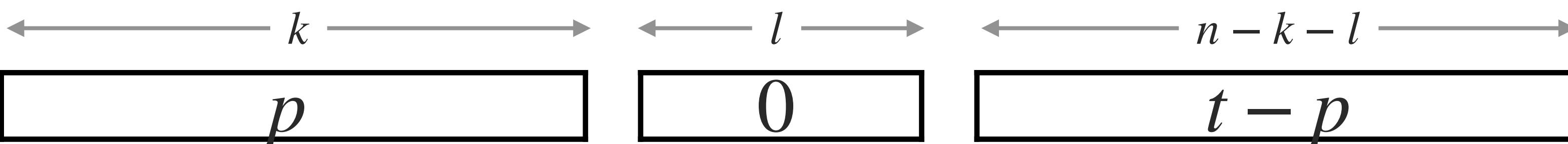
Prange



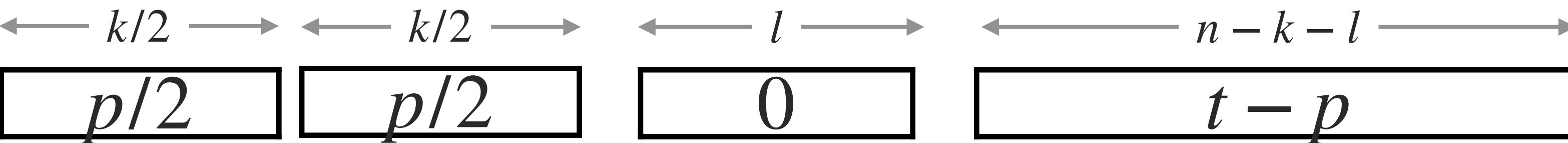
Lee-Brickell



Leon



Stern



Next time:

MPC-in-the-Head construction