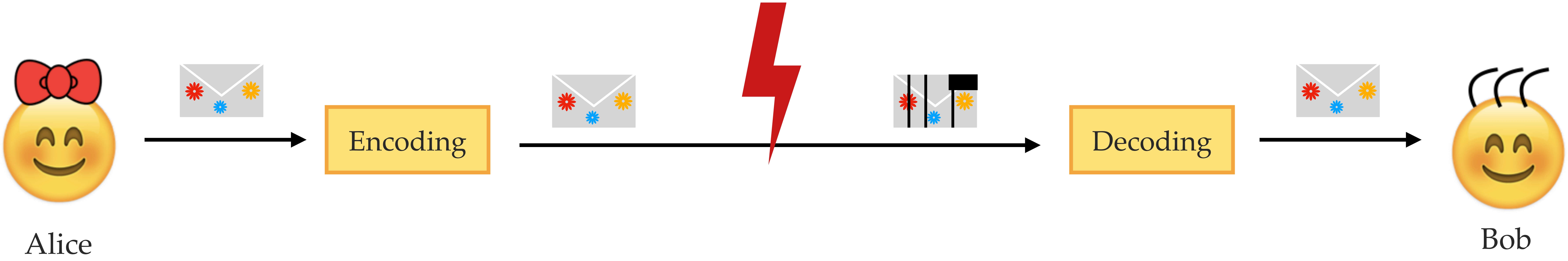
Code-based cryptography II

Monika Trimoska

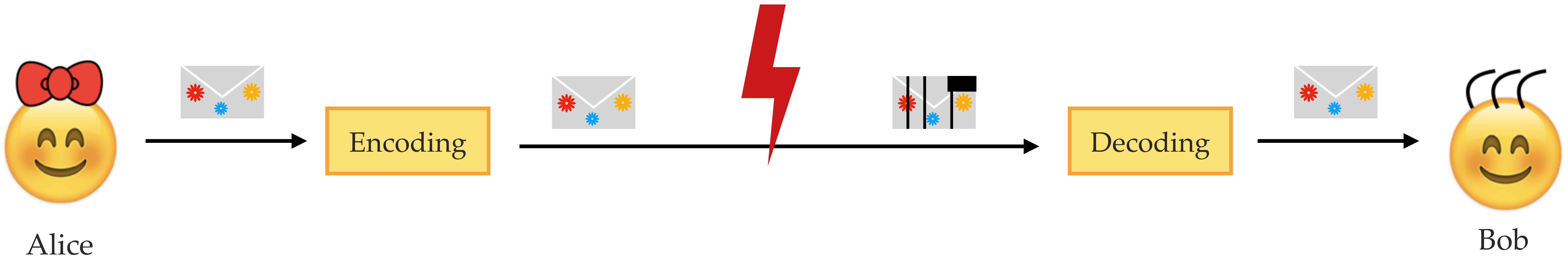
Selected Areas in Cryptology - Part 1
Spring, 2024

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 $\mathscr{C}$ over $\mathbb{F}_q$ is a k-dimensional subspace of $\mathbb{F}_q^n$.

Linear codes (recall)

Linear code

An $[n, k]$ linear code $\mathscr{C}$ 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 $\mathscr{C}$ 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 $\mathscr{C}$ 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 $\mathscr{C}$ 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.

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

Linear codes (recall)

Linear code

An $[n, k]$ linear code $\mathscr{C}$ 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.

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

Generator matrix

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

$$\mathscr{C} = \{\mathbf{x}\mathbf{G} \,|\, \mathbf{x} \in \mathbb{F}_q^k\}.$$

Binary linear codes

Binary linear code

An $[n, k]$ binary linear code $\mathscr{C}$ 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.

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

Generator matrix

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

$$\mathscr{C} = \{\mathbf{x}\mathbf{G} \,|\, \mathbf{x} \in \mathbb{F}_2^k\}.$$

Binary linear codes

Binary linear code

An $[n, k]$ binary linear code $\mathscr{C}$ 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 $\mathscr{C}$ 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. $\mathrm{wt}(\mathbf{e}) = t$.

→ Decoding: Given $\mathbf{y}$, find $\mathbf{c}$ s.t. $\mathbf{y} = \mathbf{c} + \mathbf{e}$ and $\mathrm{wt}(\mathbf{e}) \leq t$.

Representations of linear codes

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

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

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 $\mathbf{G} \in \mathbb{F}_2^{k \times n}$:

$$\mathscr{C} = \{\mathbf{x}\mathbf{G} \,|\, \mathbf{x} \in \mathbb{F}_2^k\}.$$

Example. $n = 7, k = 4$

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

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

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

*We are omitting the transpose ($\top$) for vectors.

Representations of linear codes

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

$$\mathscr{C} = \{\mathbf{x}\mathbf{G} \,|\, \mathbf{x} \in \mathbb{F}_2^k\}.$$

Example. $n = 7, k = 4$

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

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

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

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 (⊤) 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 **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

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

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

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

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

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

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

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

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

→ 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 $\mathbf{H}$

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

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

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

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

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

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

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

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

→ 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 $\mathbf{H}$:

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

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. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

The failure pattern uniquely identifies the error location.

Example: Hamming code

An error occurs.

Example. $n = 7, k = 4$

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

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 $\mathbf{s} = \mathbf{He}$, find $\mathbf{e}$ such that $\mathrm{wt}(\mathbf{e}) \leq t$.

The syndrome decoding problem

The syndrome decoding problem

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

$$\overset{\mathbf{H}}{\begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}} \quad \overset{\mathbf{e}}{\begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \\ e_7 \end{pmatrix}} = \overset{\mathbf{s}}{\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}}$$

Find $\mathbf{e}$ of minimum weight.

Information set decoding

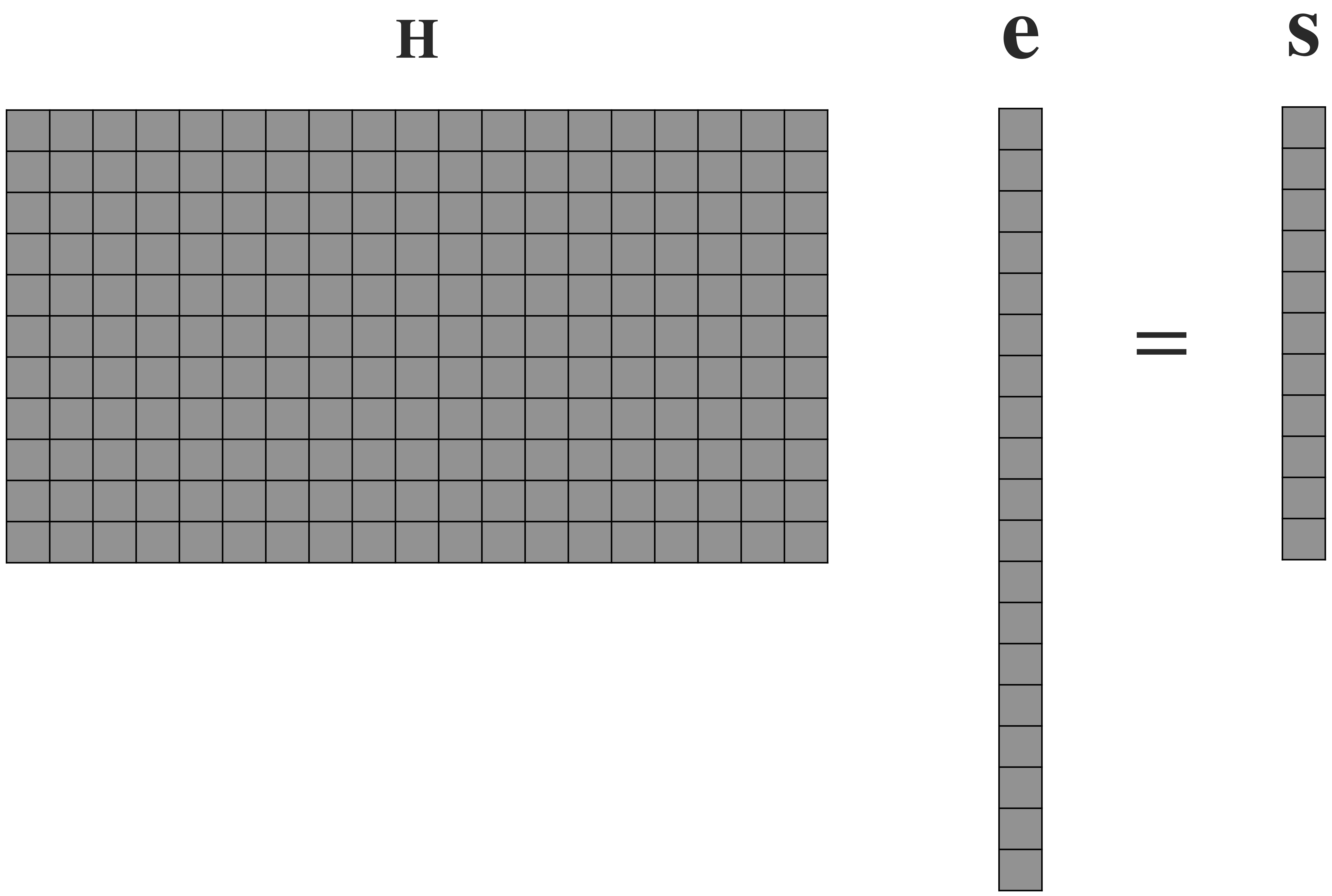
Information set decoding algorithms

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

The syndrome decoding problem

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

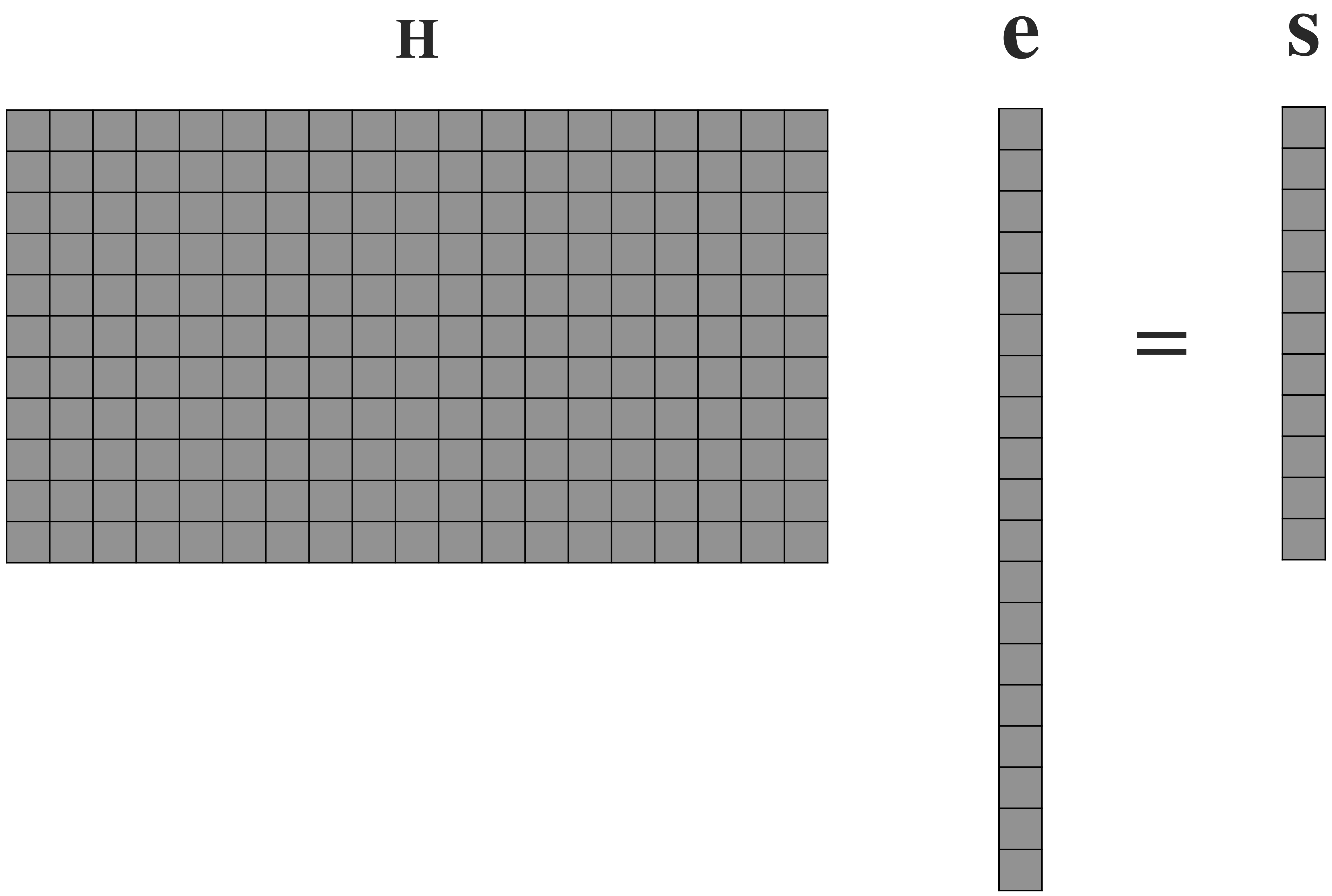
Brute force



H **e** = **s**

Brute force

- ☐ Entry is 0
- ■ Entry is 1
- ▨ Entry is 0 or 1

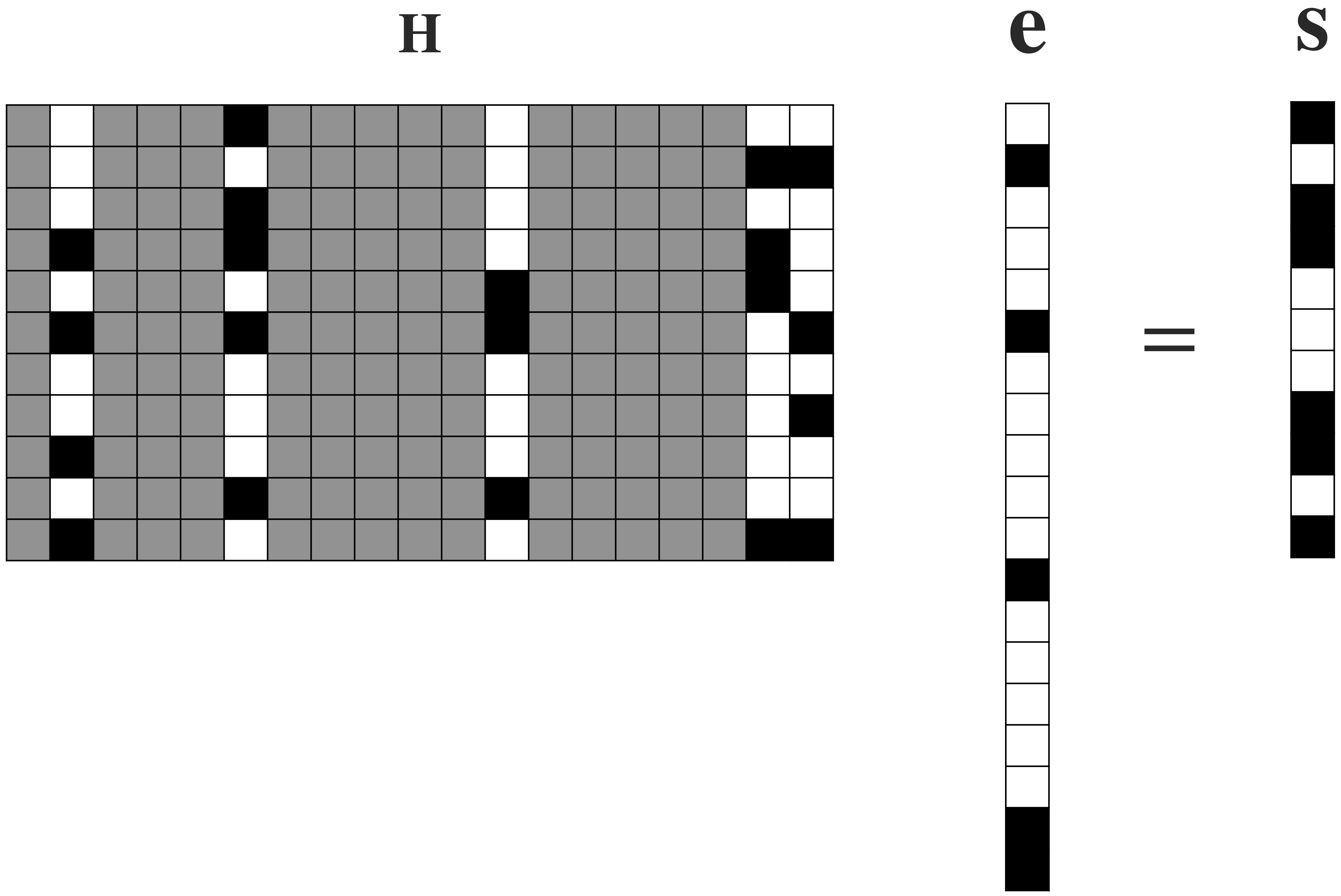


Brute force

☐ Entry is 0

■ Entry is 1

▨ Entry is 0 or 1

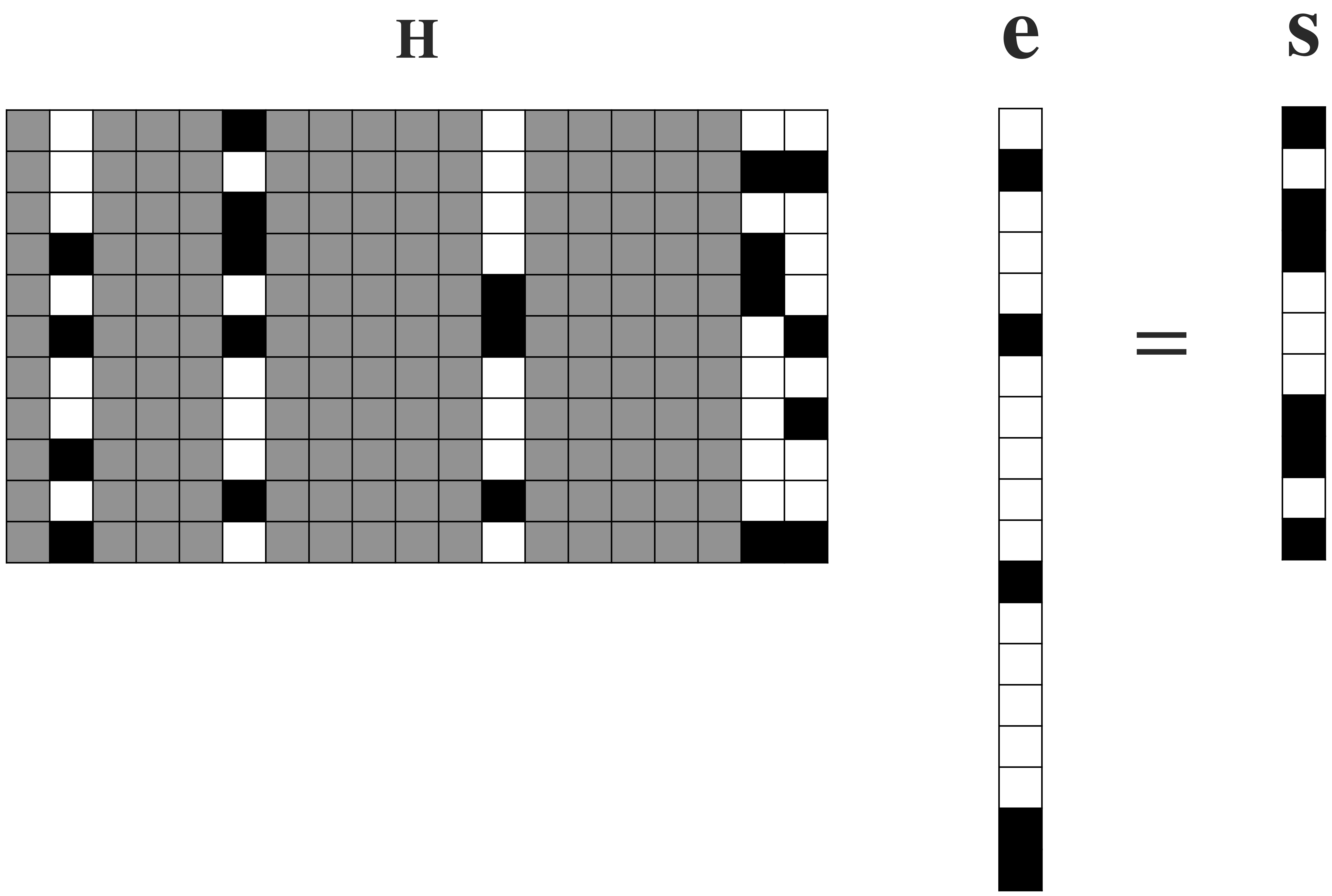


Brute force

□ Entry is 0

■ Entry is 1

▨ Entry is 0 or 1



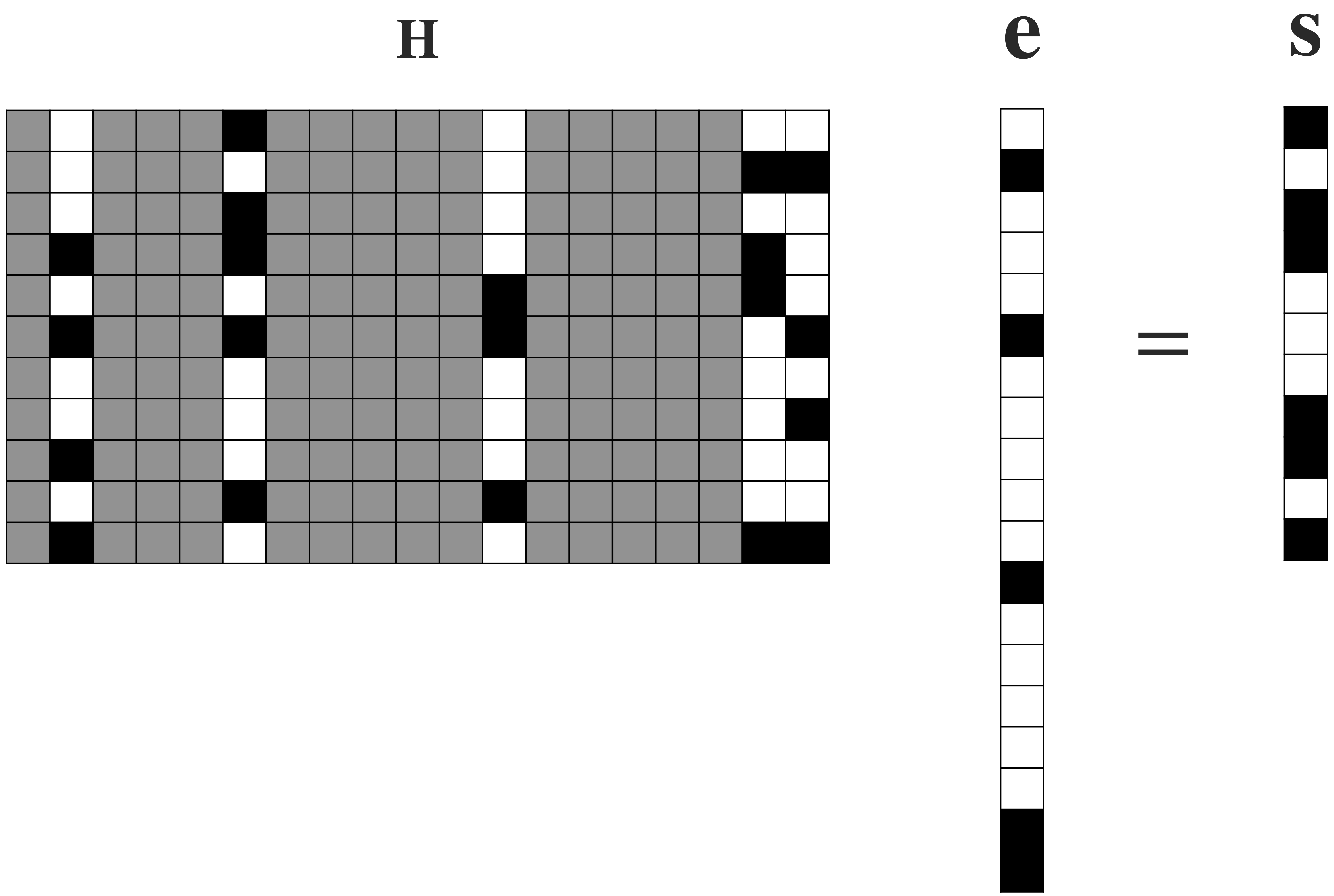
$\mathbf{s}$ is equal to the sum of the columns where e_i is nonzero.

Brute force

☐ Entry is 0

■ Entry is 1

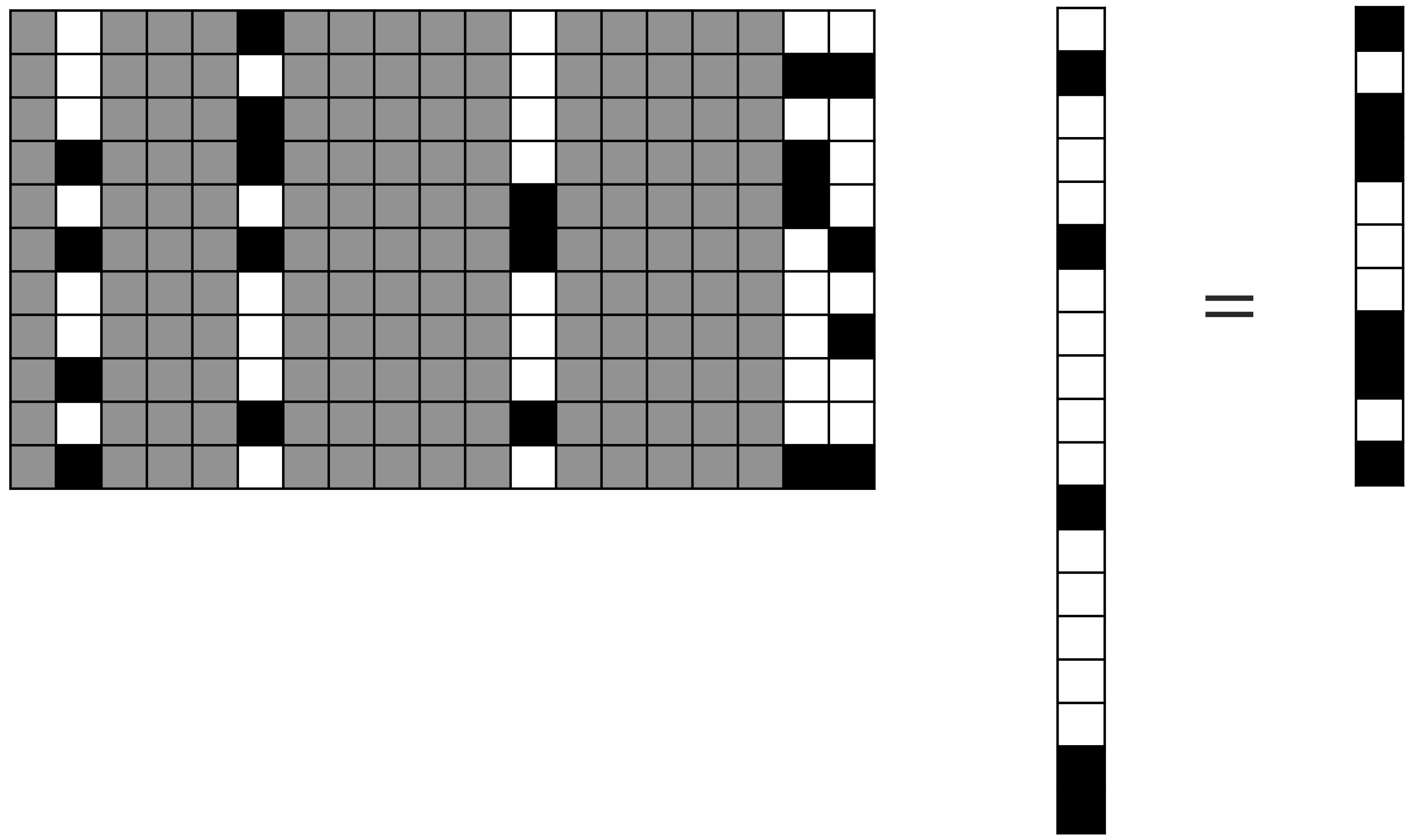
▨ Entry is 0 or 1



s is equal to the sum of the columns where e_i is nonzero.

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

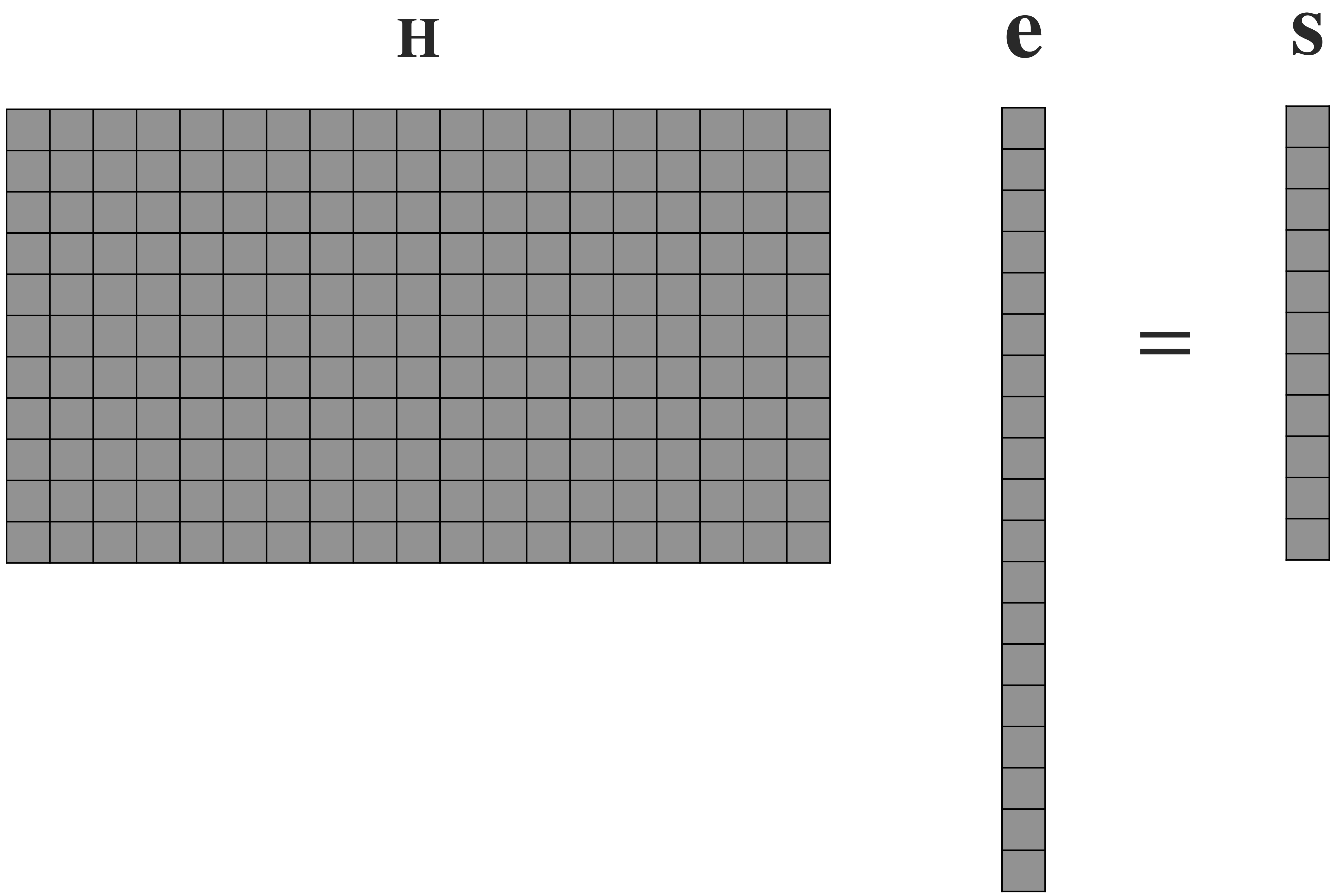
Brute force: complexity



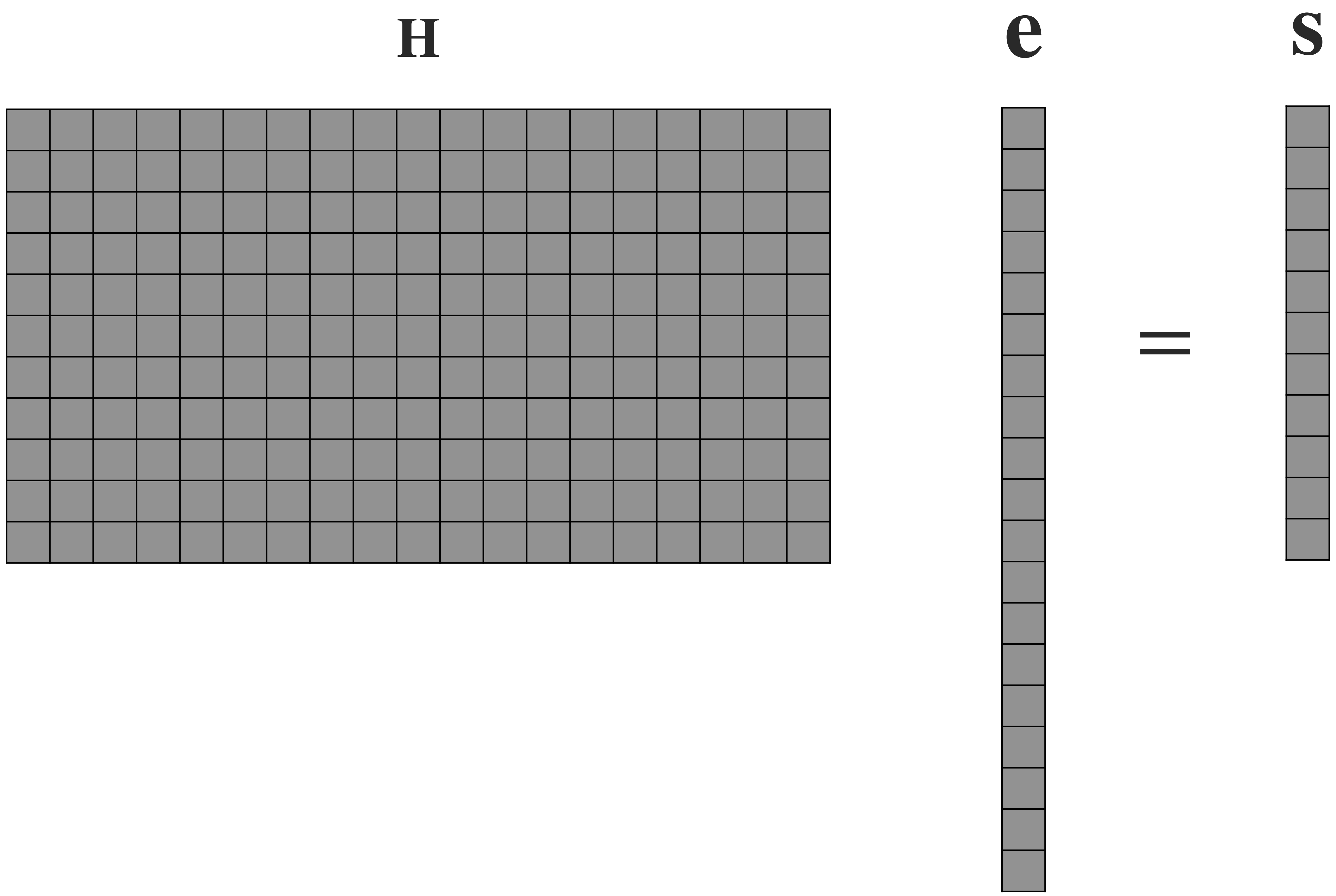
Pick any group of t columns of $\mathbf{H}$, add them and compare with $\mathbf{s}$.

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

Prange’s attack

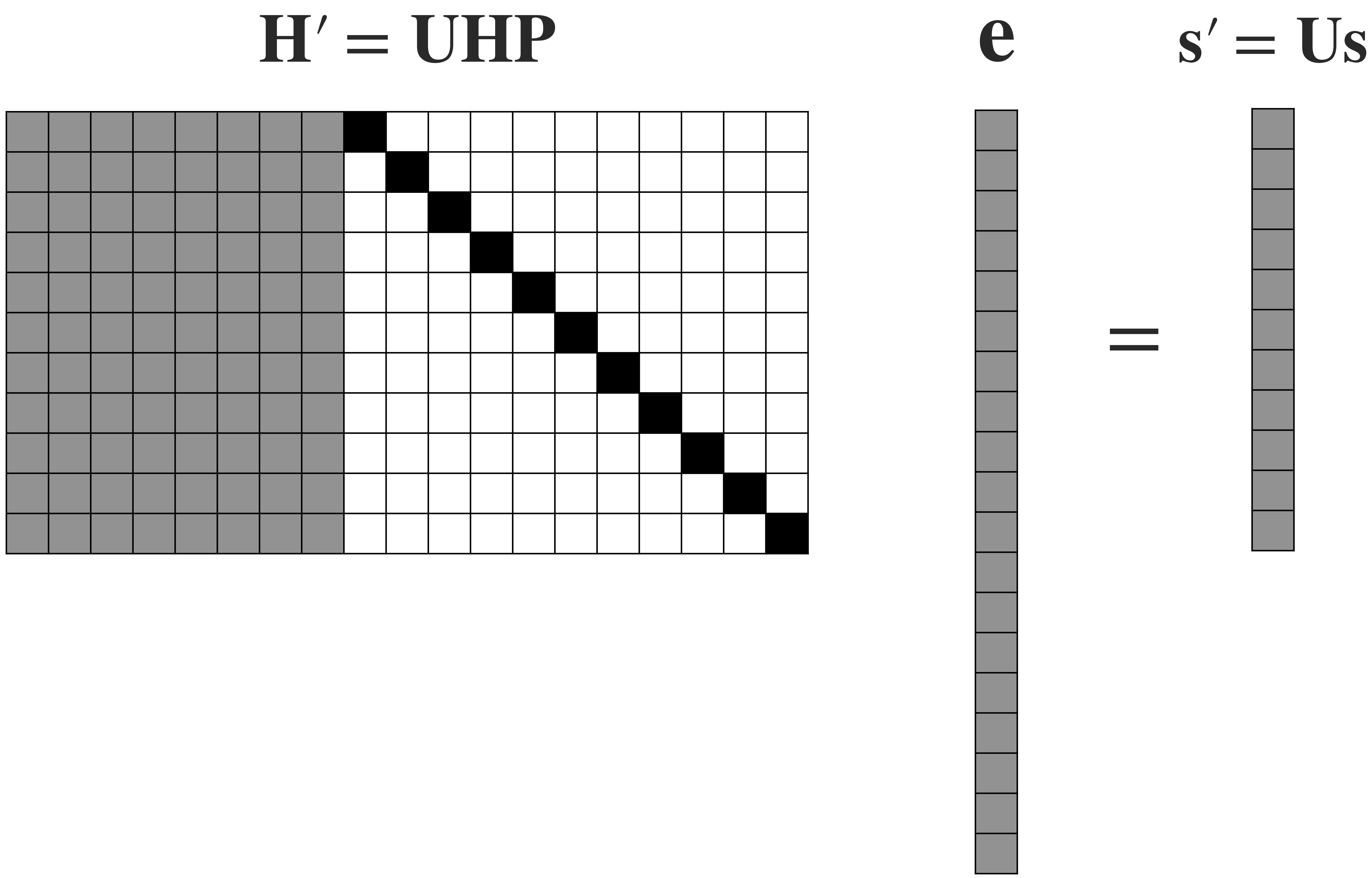


Prange’s attack



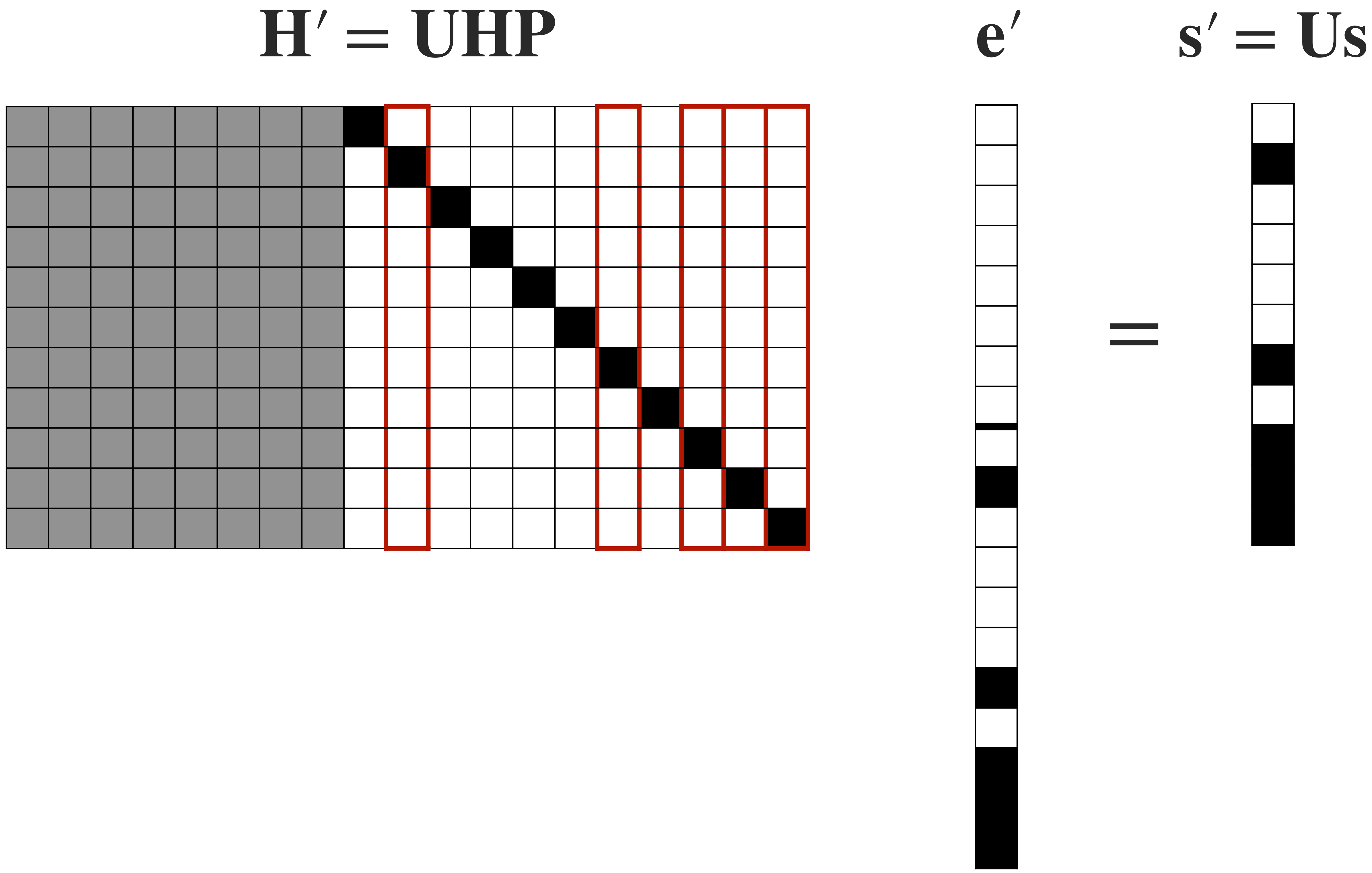
→ Permute **H** and bring to systematic form.

Prange’s attack



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

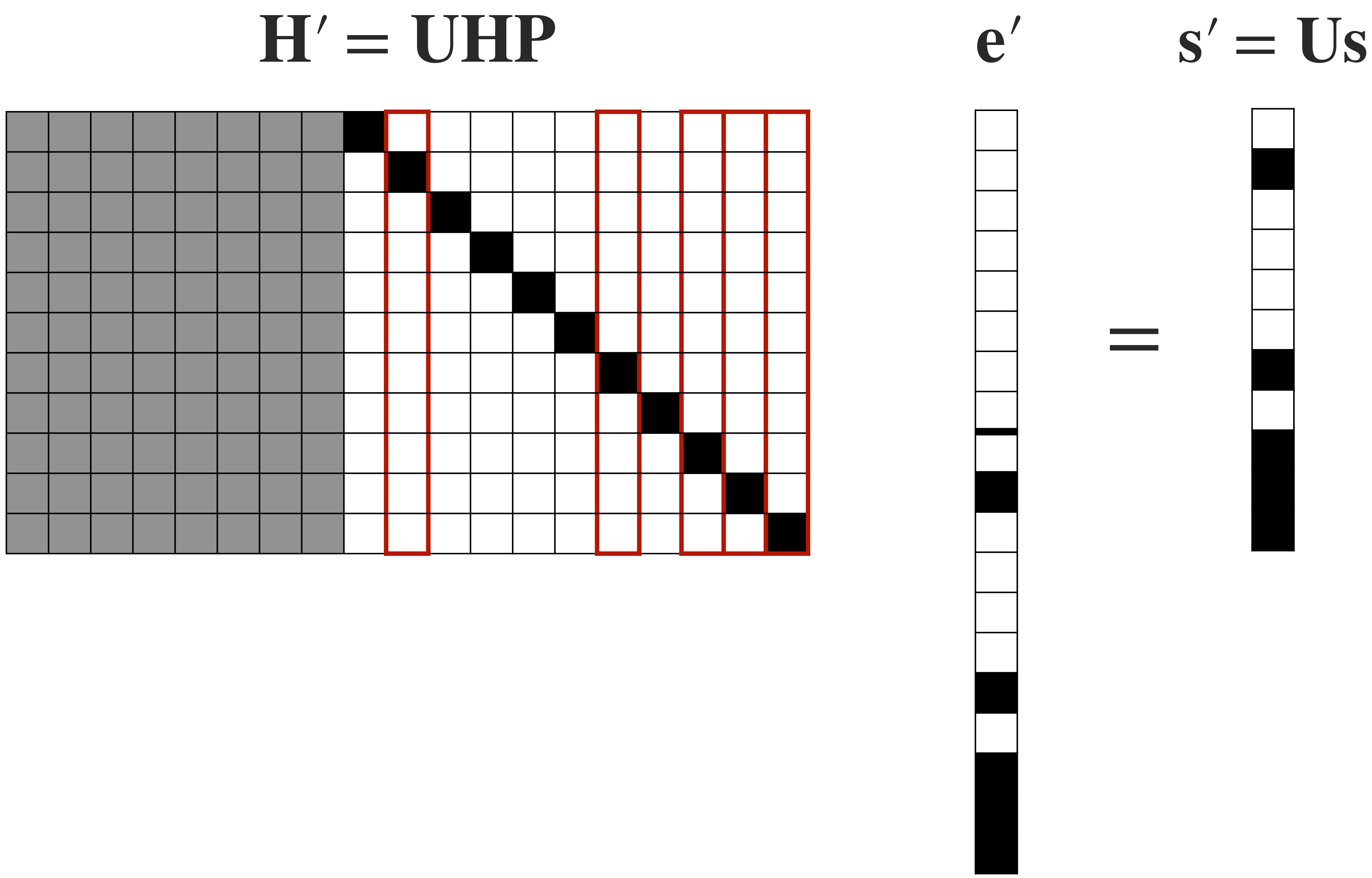
Prange’s attack



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

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

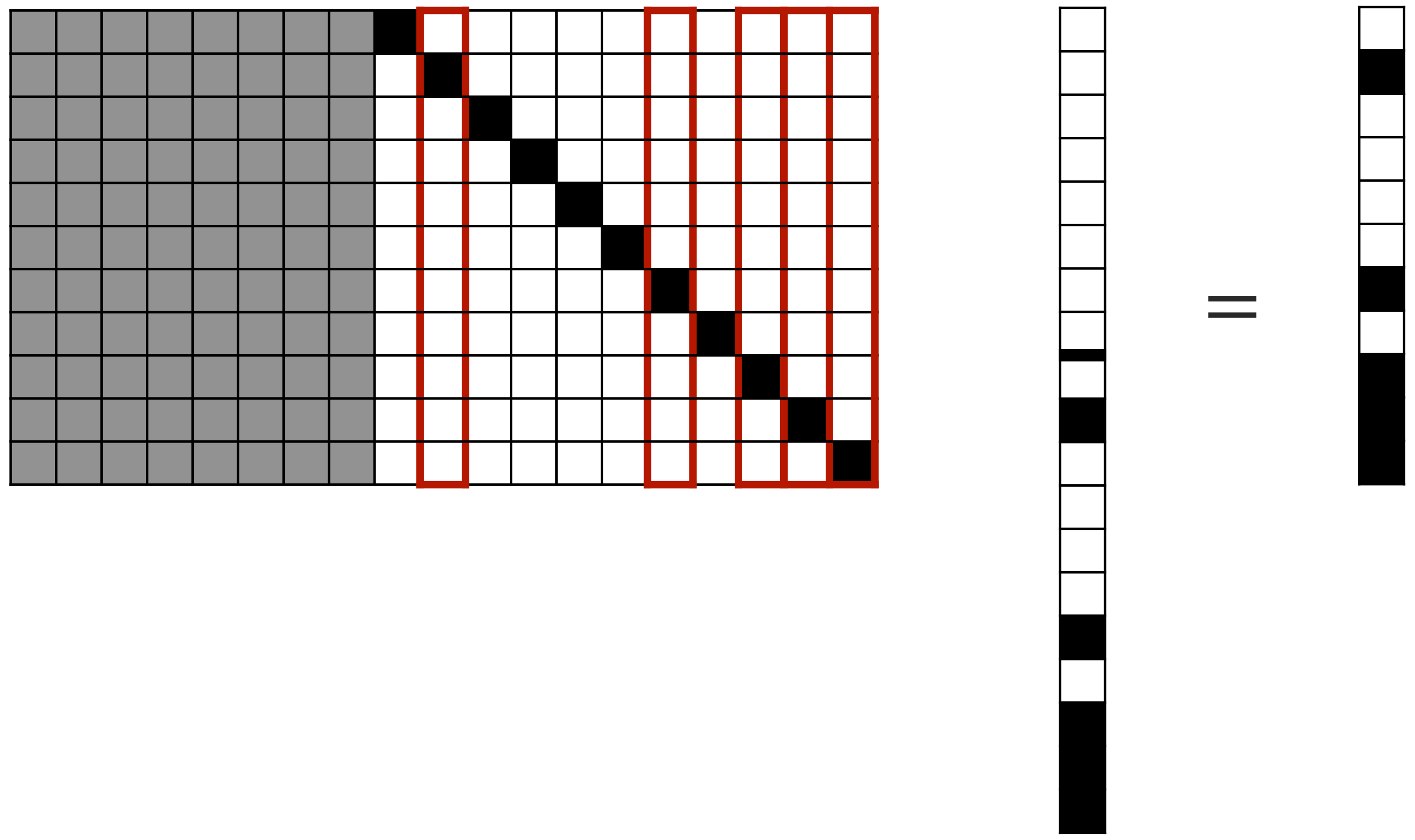
Prange’s attack



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

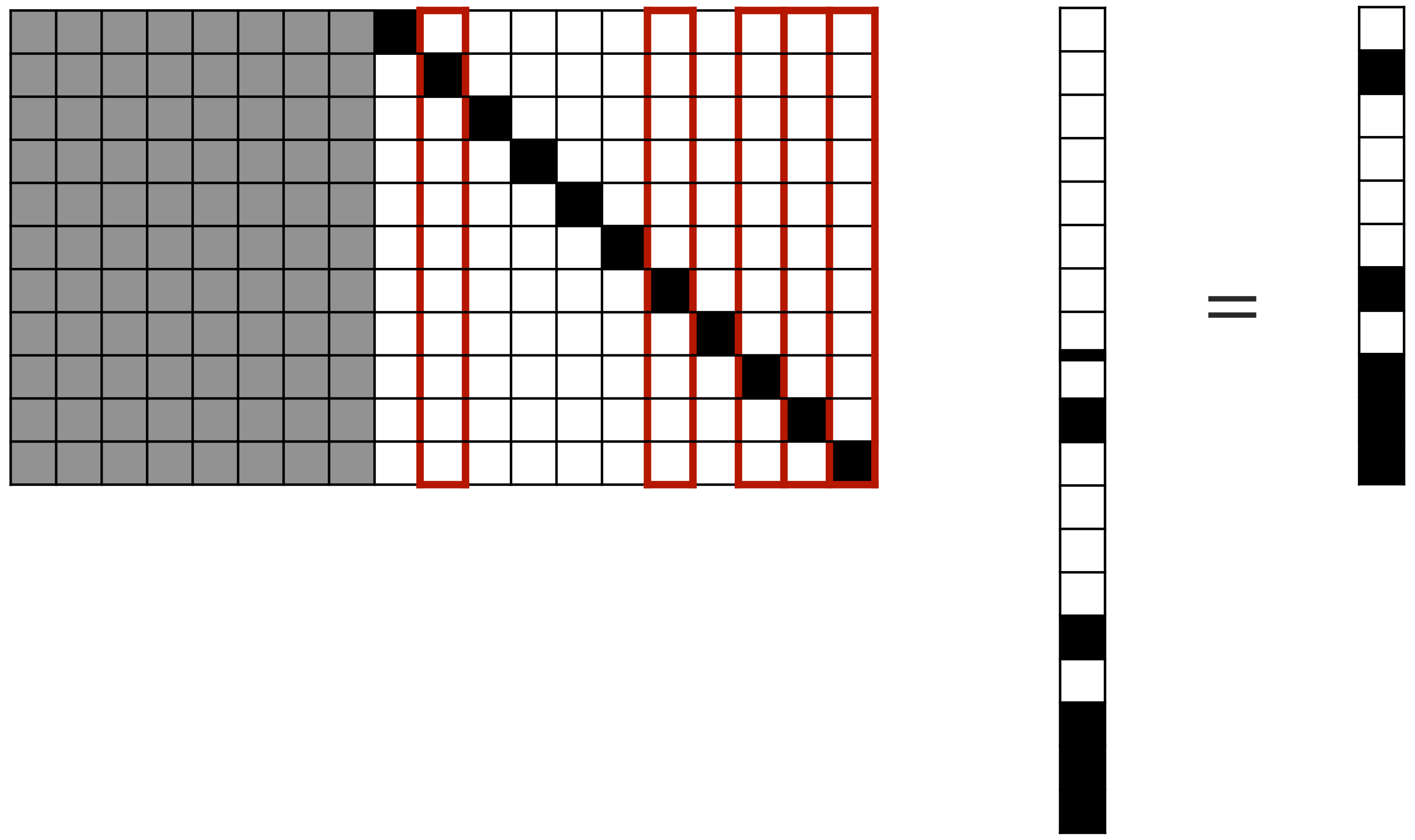
  Suppose that all t errors are in the identity (right) part. Then $\mathbf{e}' = (000\ldots)||\mathbf{Us}$ and $\mathrm{wt}(\mathbf{Us}) = t$.
- If $\mathrm{wt}(\mathbf{Us}) = t$, then output unpermuted version of $\mathbf{e}'$.
- Else, return to the first step and rerandomize: choose a new permutation.

Prange’s attack: complexity



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

Prange's attack: complexity

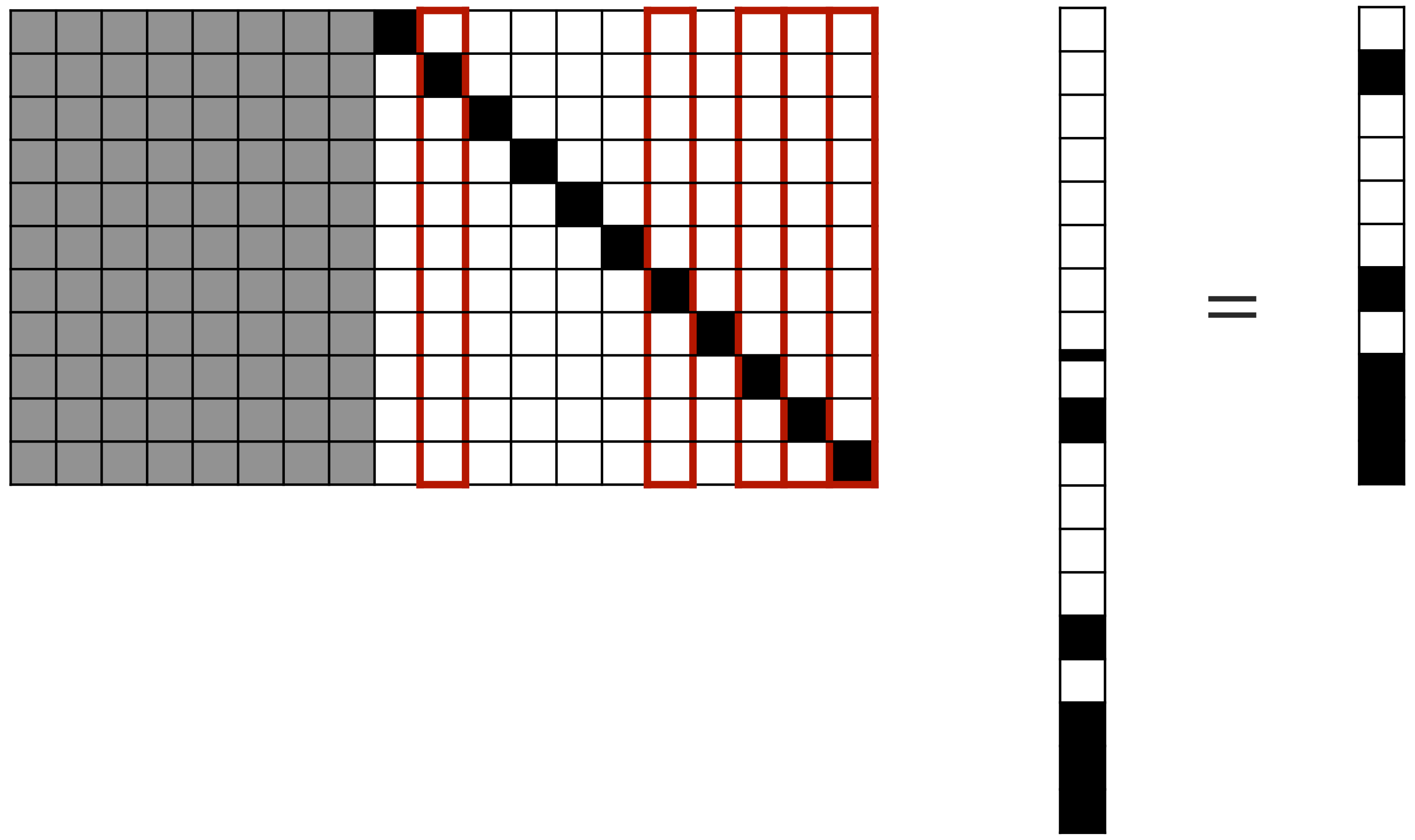


- Permute **H** and bring to systematic form.
- If wt(**Us**) = t, then output unpermuted version of **e**.
- Else, return to the first step and rerandomize: choose a new permutation.

All errors are in the identity part

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

Prange's attack: complexity



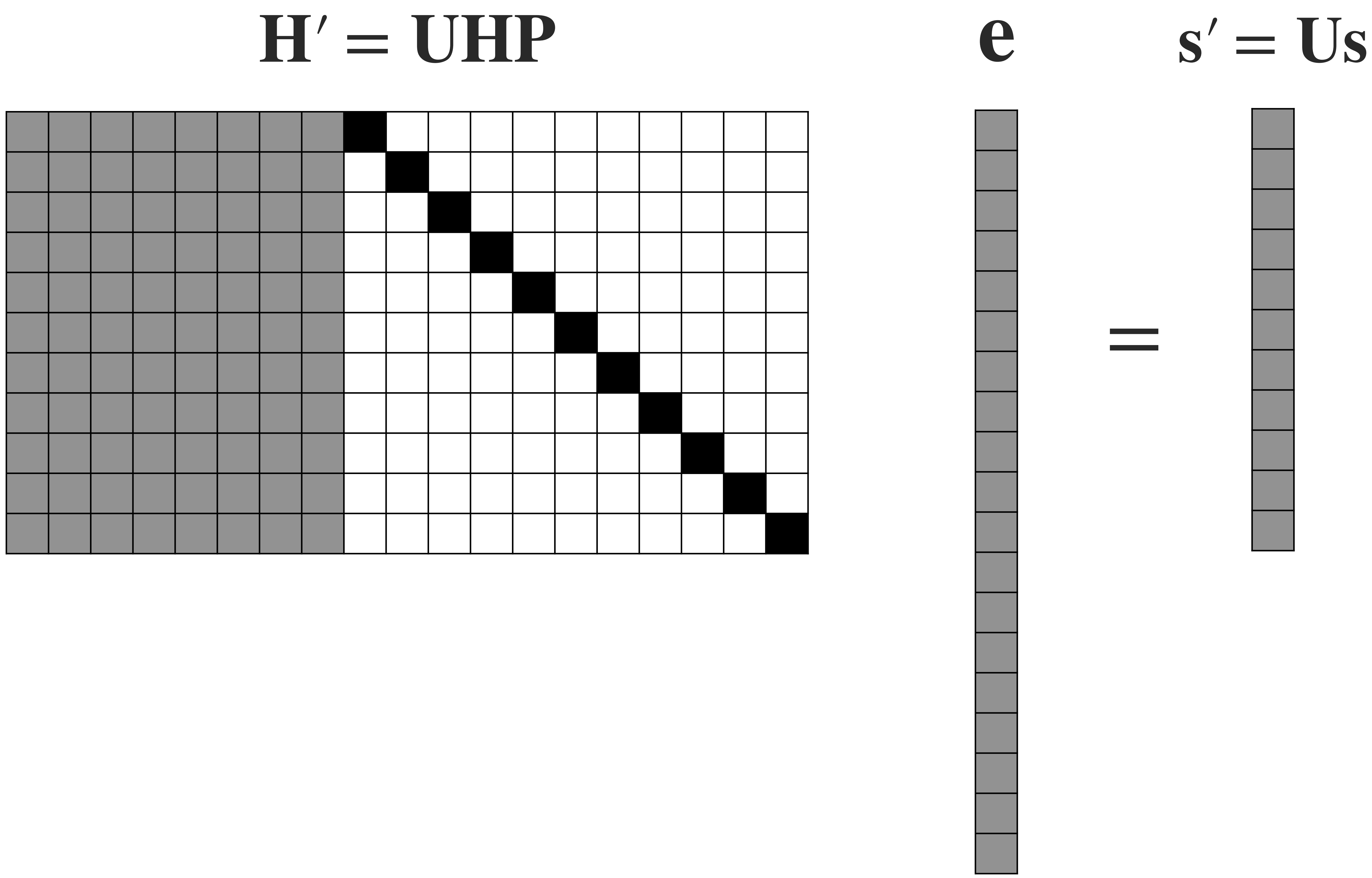
- Permute **H** and bring to systematic form.
- If wt(**Us**) $= t$, then output unpermuted version of **e**.
- Else, return to the first step and rerandomize: choose a new permutation.

All errors are in the identity part

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

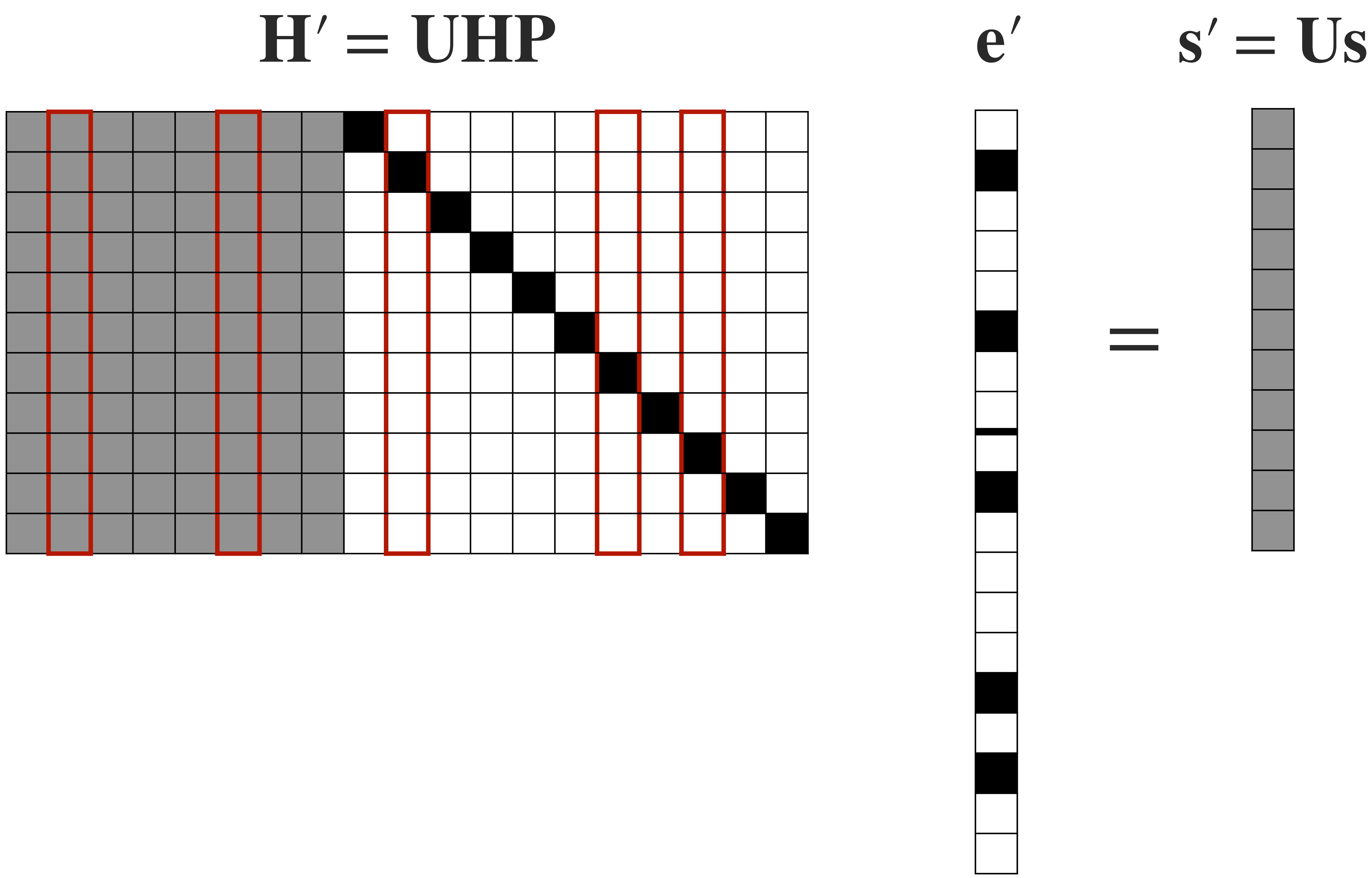
Cost: $\frac{\binom{n}{t}}{\binom{n-k}{t}}$ matrix operations.

Lee-Brickell attack



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

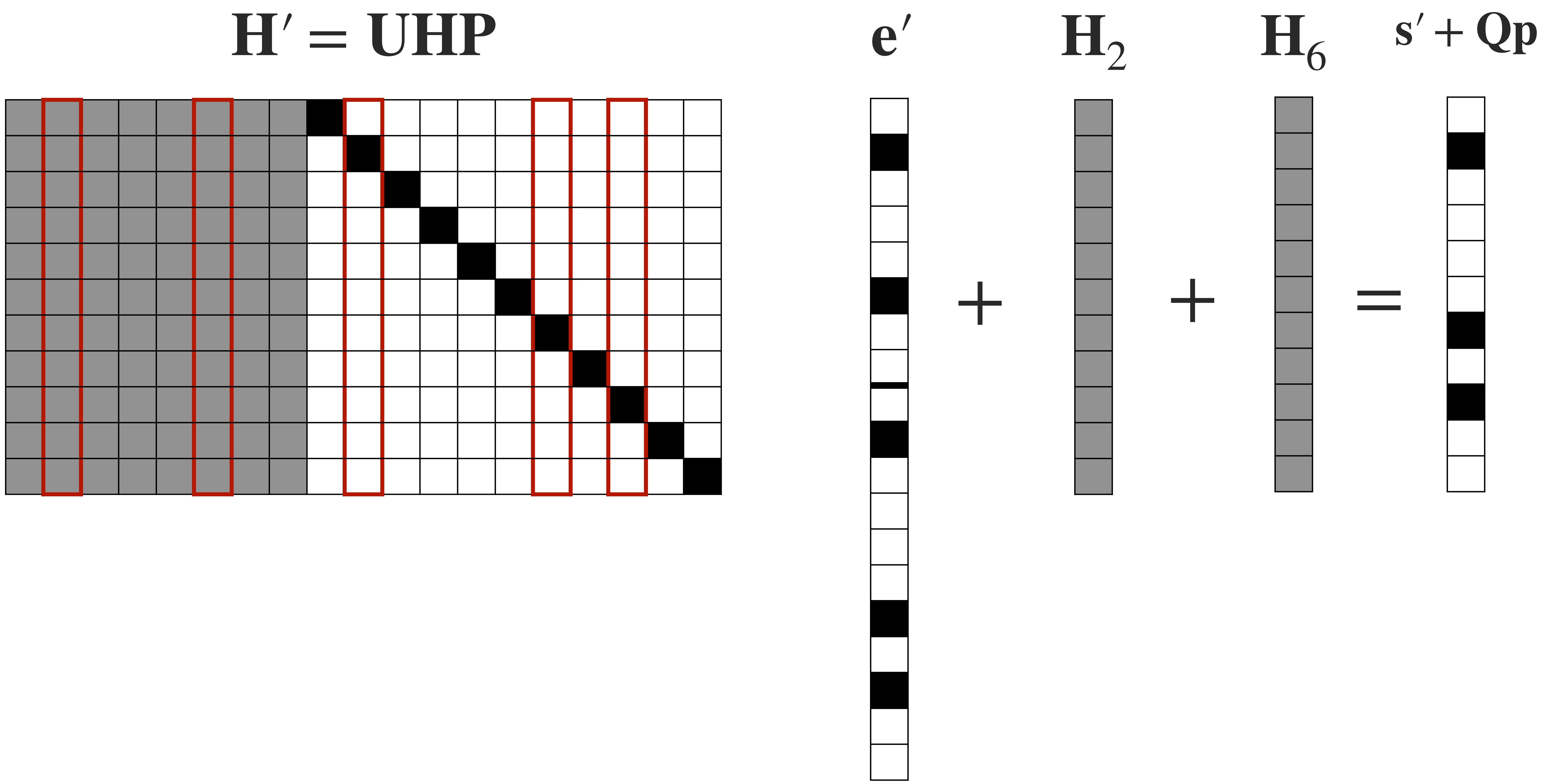
Lee-Brickell attack



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

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

Lee-Brickell attack

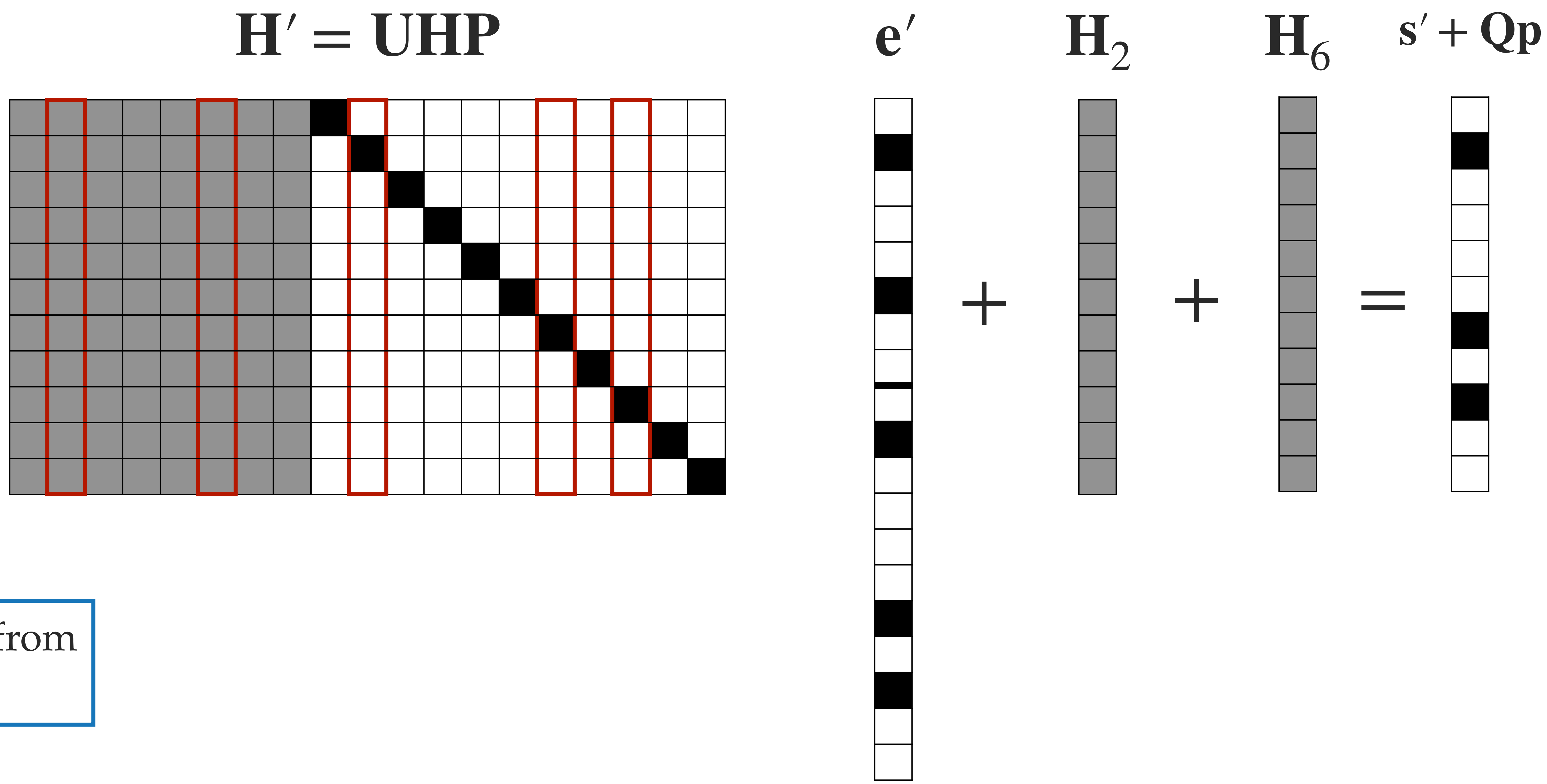


→ Permute **H** and bring to systematic form.

Suppose that there are $(t-p)$ errors are 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 $t-p$: $\mathrm{wt}(\mathbf{s}' + \mathbf{Qp}) = t - p$.

Lee-Brickell attack



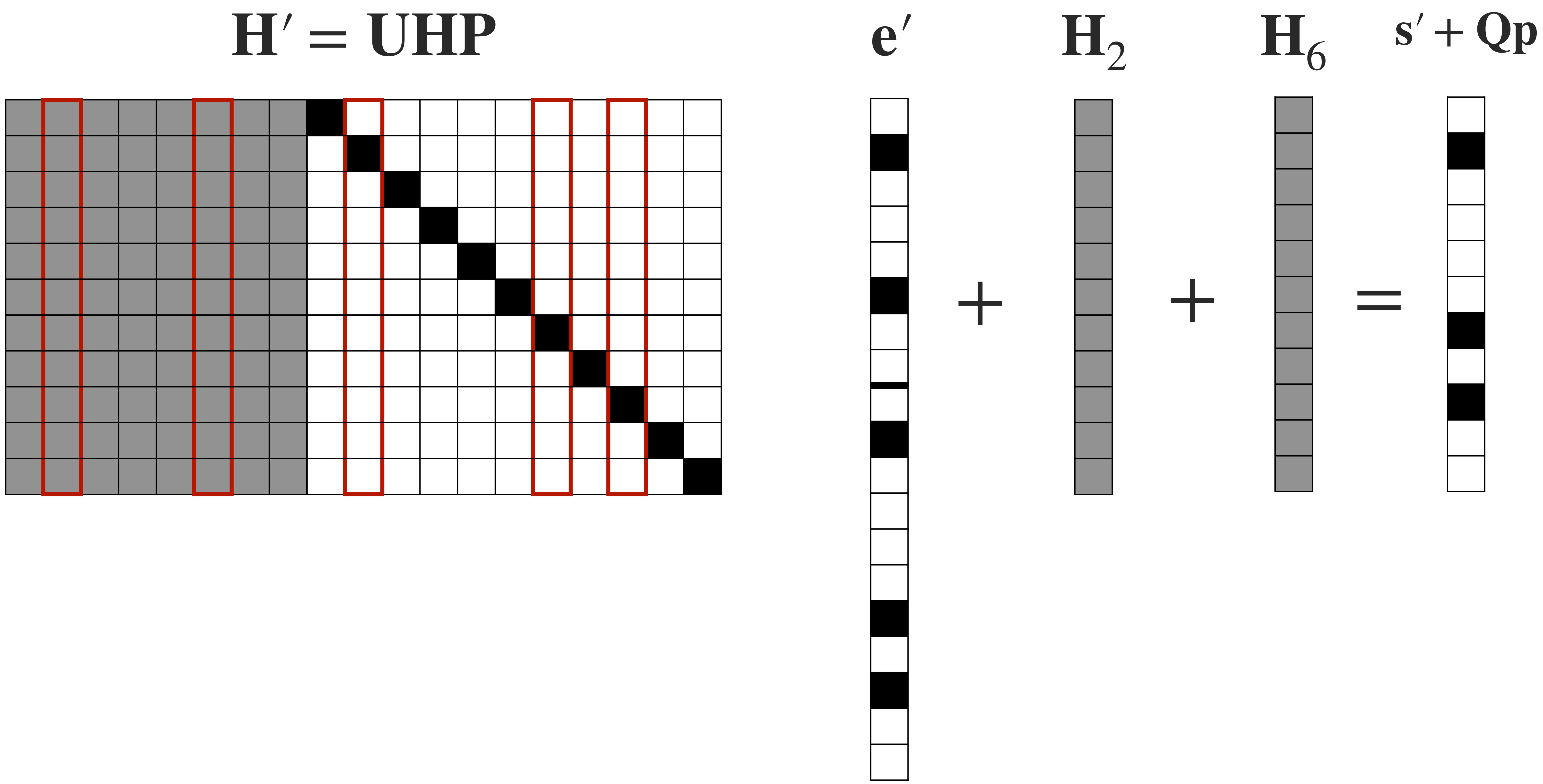
Let $\mathbf{p}$ be a vector chosen from $\{\mathbf{p} \in \mathbb{F}^k \,|\, \mathrm{wt}(\mathbf{p}) = p\}$

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

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

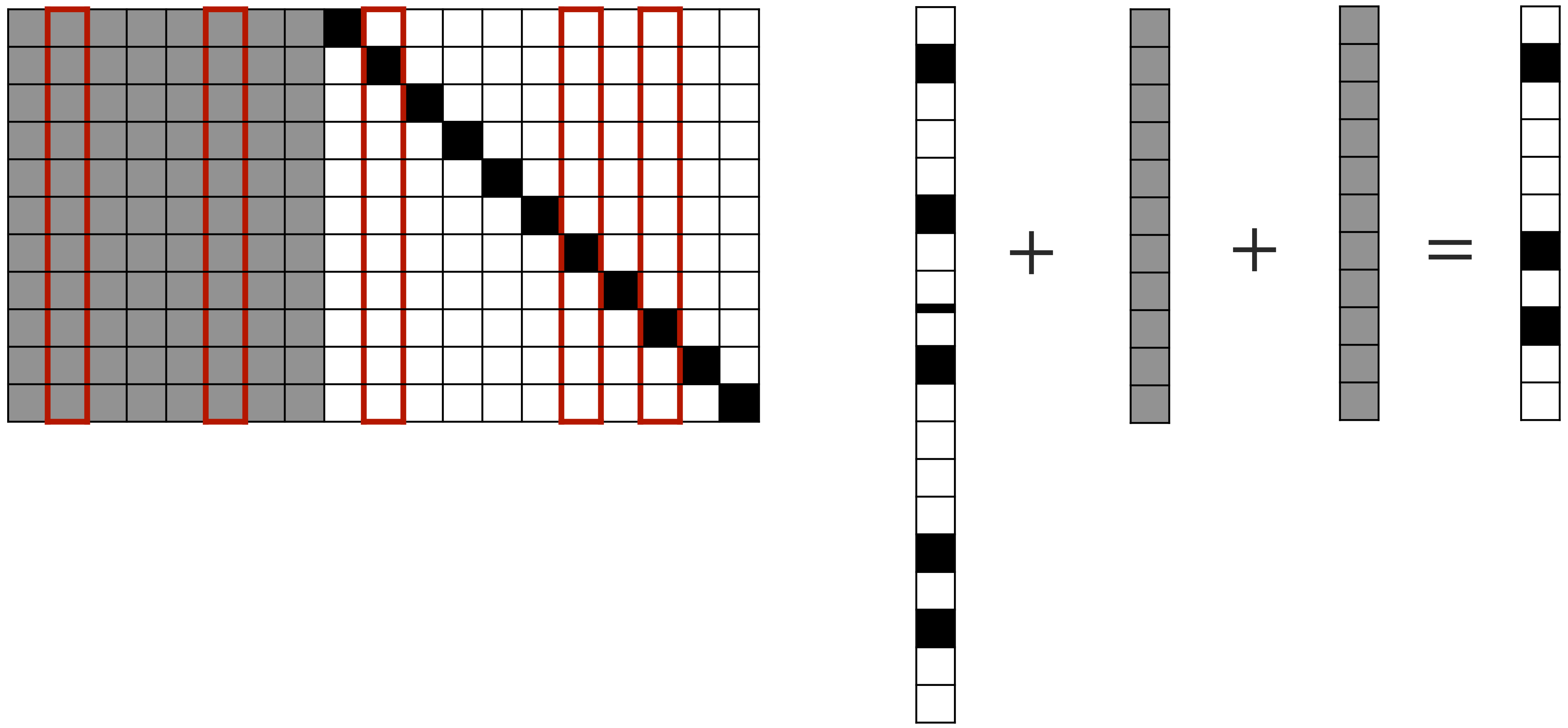
Then, $\mathbf{s}'$ is random-looking, but $\mathbf{s}'$ summed with the error columns on the left has weight $t - p$: $\mathrm{wt}(\mathbf{s}' + \mathbf{Qp}) = t - p$.

Lee-Brickell attack

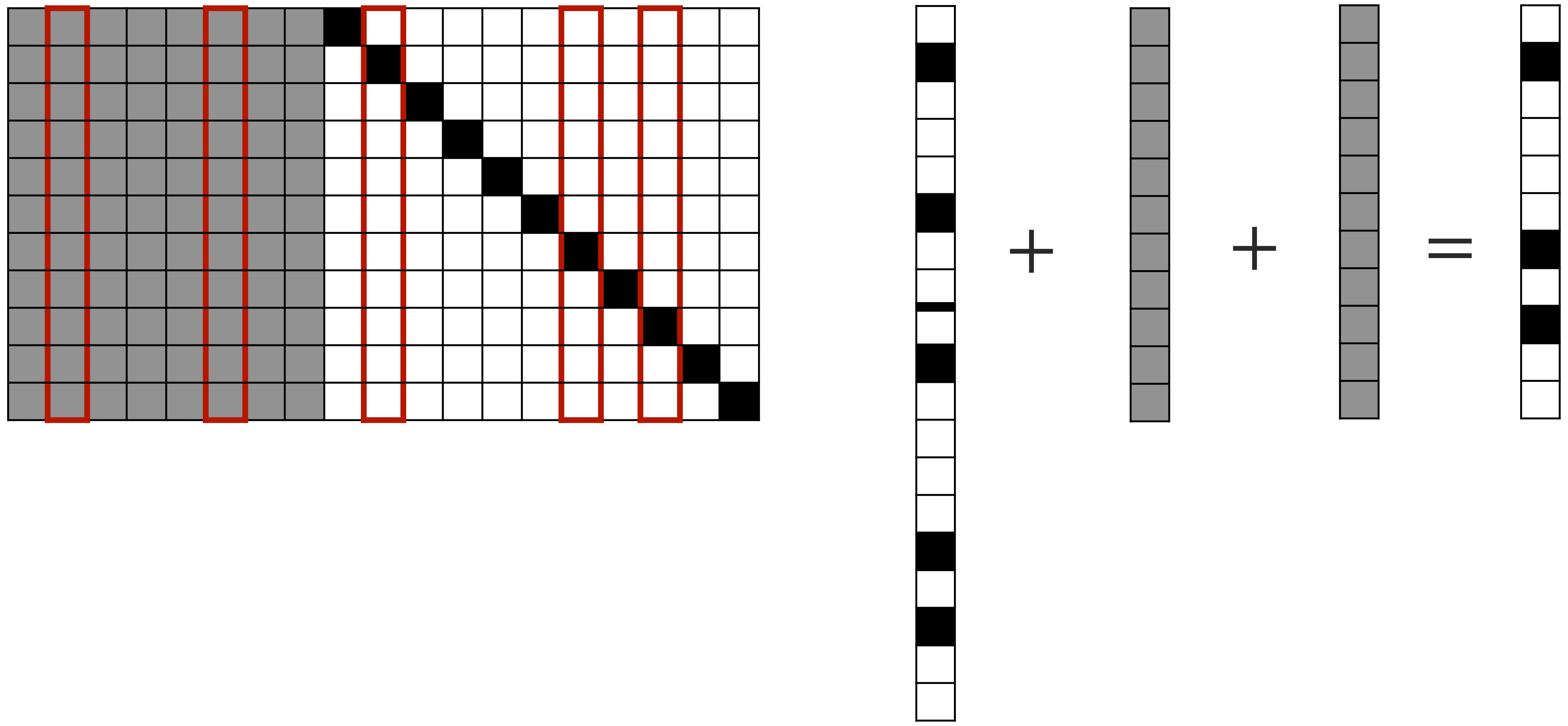


- Permute **H** and bring to systematic form.
- Pick p of the columns on the left and compute their sum: $\mathbf{Qp}$.
- If $\mathrm{wt}(\mathbf{s}' + \mathbf{Qp}) = t - p$ then put $\mathbf{e}' = \mathbf{p}\,||\,(\mathbf{s}' + \mathbf{Qp})$. Output unpermuted version of $\mathbf{e}$.
- Else, return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

Lee-Brickell attack: complexity



Lee-Brickell attack: complexity

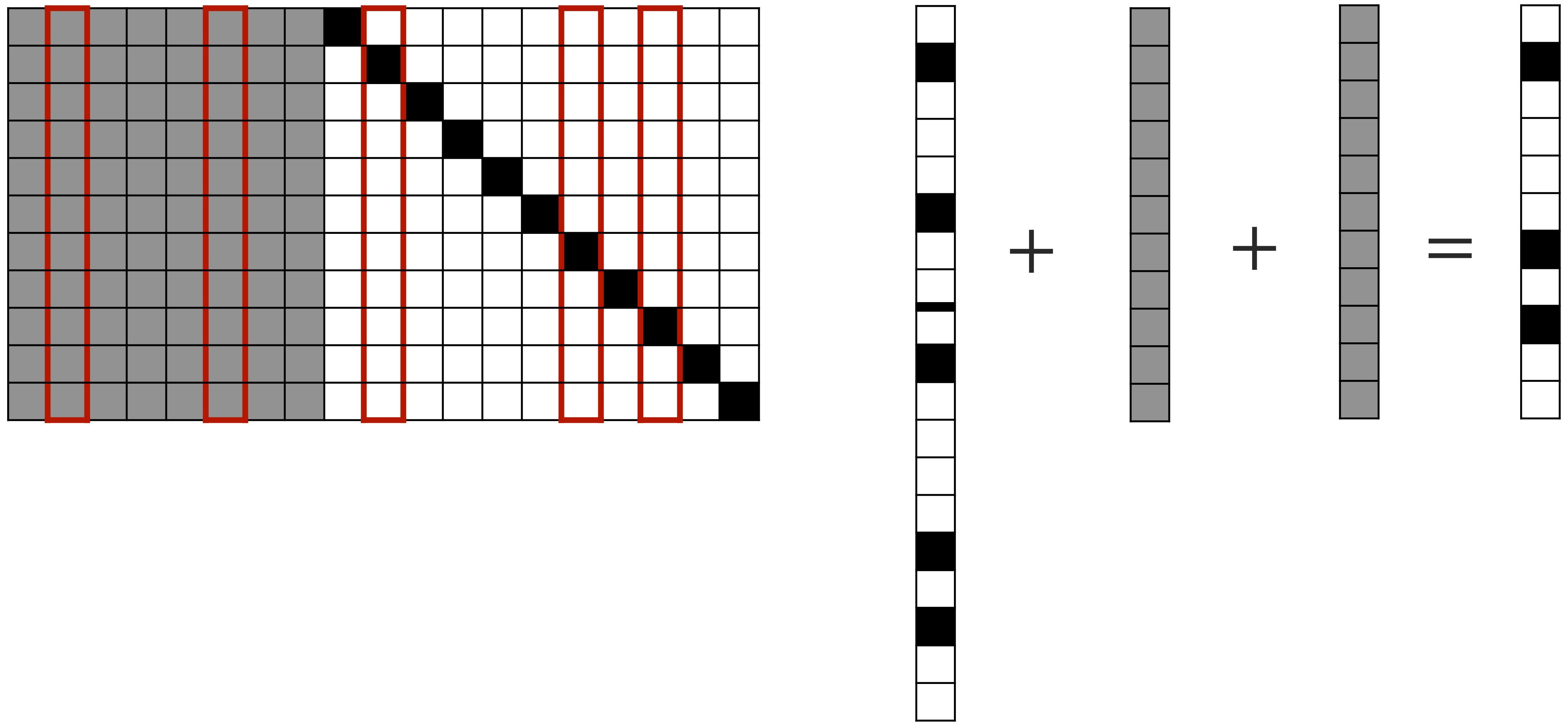


$t - p$ errors are in the identity part

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



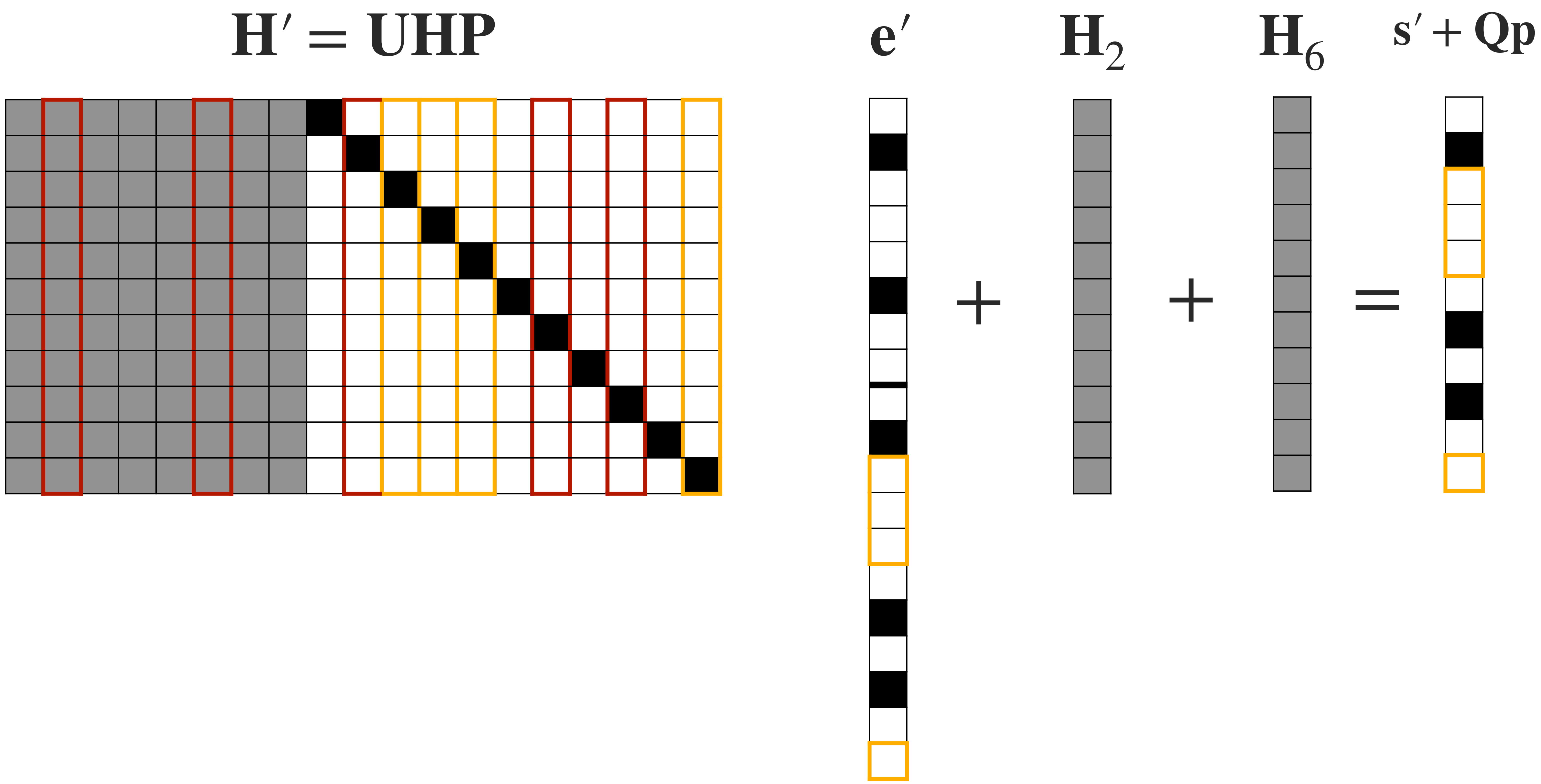
$t - p$ errors are in the identity part

p errors are in the left part

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

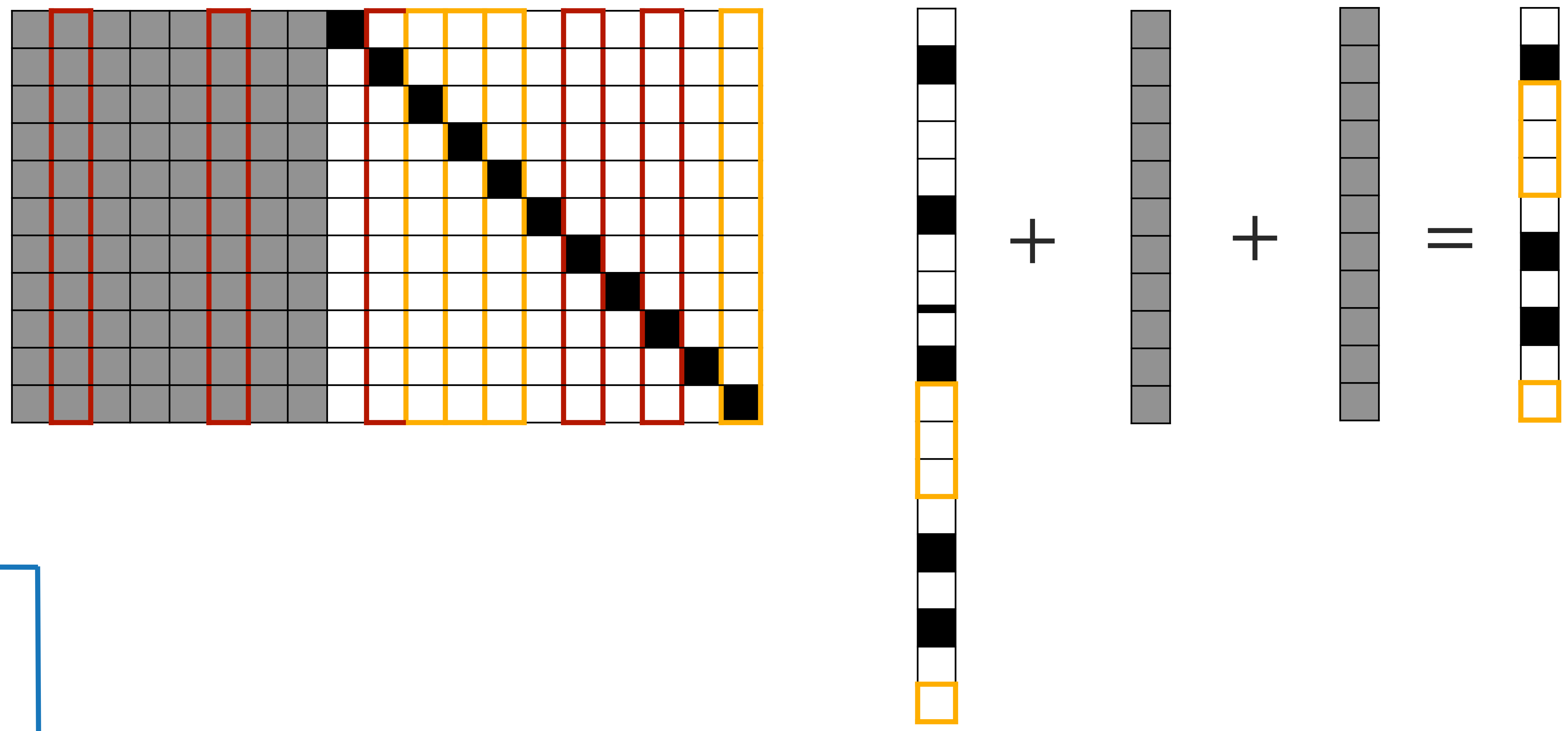
Cost: $$\frac{\binom{n}{t}}{\binom{n-k}{t-p}\binom{k}{p}}$$ matrix operations + $\binom{k}{p}$ column additions.

Leon's attack



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

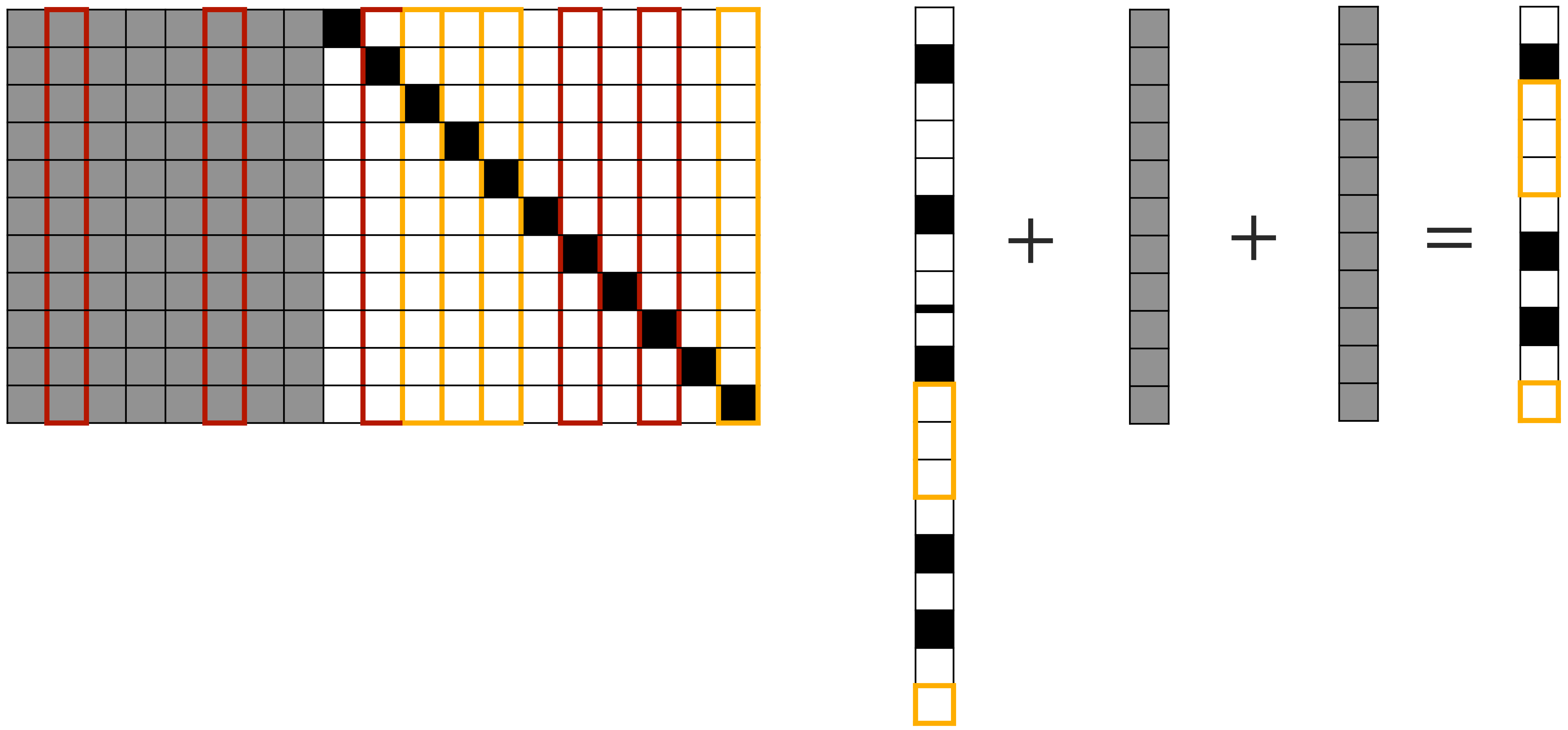
Leon's attack



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

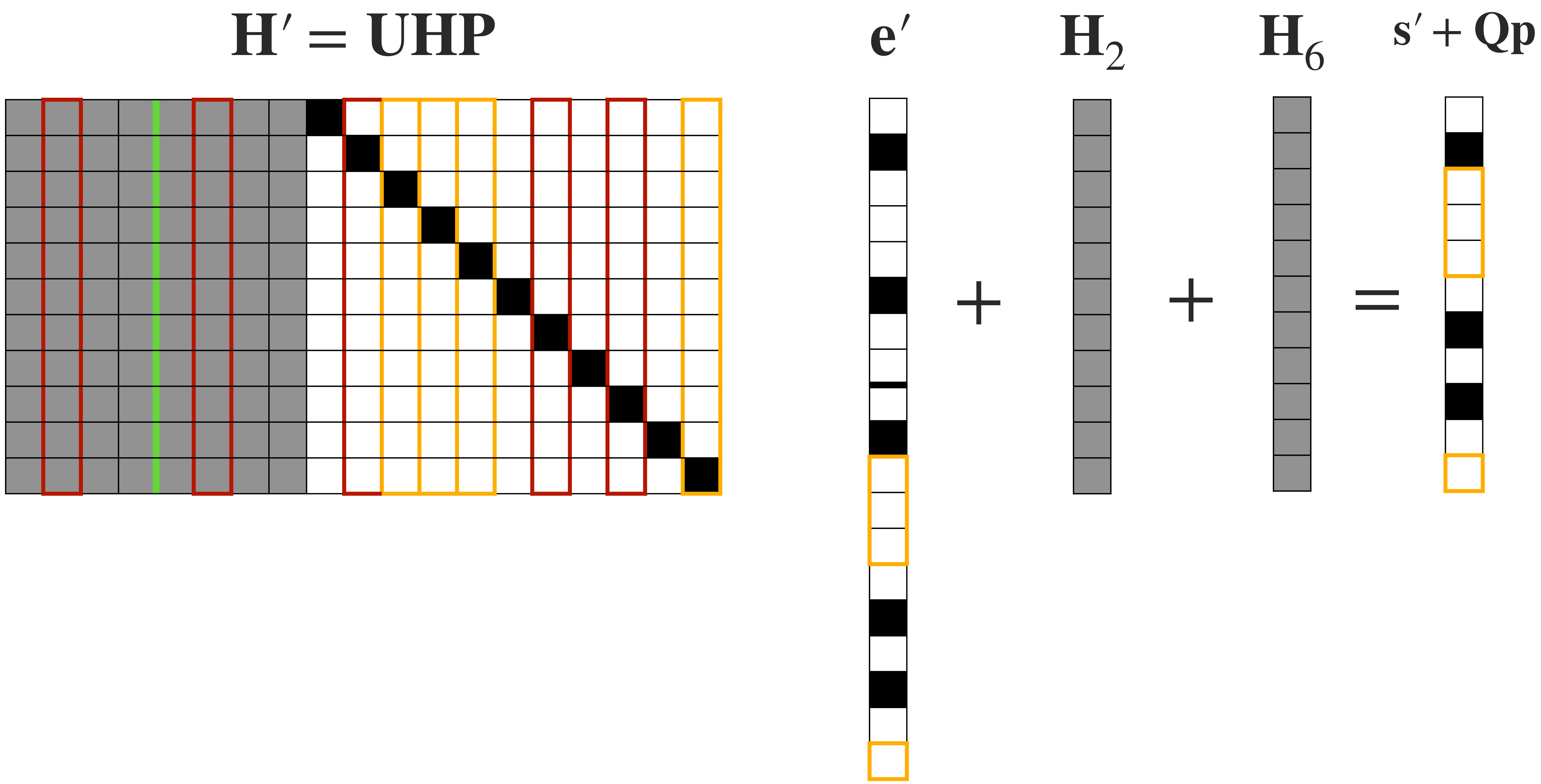
- Pick a subset L of l rows: $\mathbf{H}_L$.
- Permute $\mathbf{H}$ and bring to systematic form (then $\mathbf{H}_L = \begin{pmatrix} \mathbf{Q}_L & \mathbf{I}_L \end{pmatrix}$).
- Pick p of the columns on the left and compute their sum: $\mathbf{Q}_L\mathbf{p}$.
- If $\mathrm{wt}(\mathbf{s}'_L + \mathbf{Q}_L\mathbf{p}) = 0$
  - If $\mathrm{wt}(\mathbf{s}' + \mathbf{Q}\mathbf{p}) = t - p$ then put $\mathbf{e}' = \mathbf{p} || (\mathbf{s}' + \mathbf{Q}\mathbf{p})$. 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

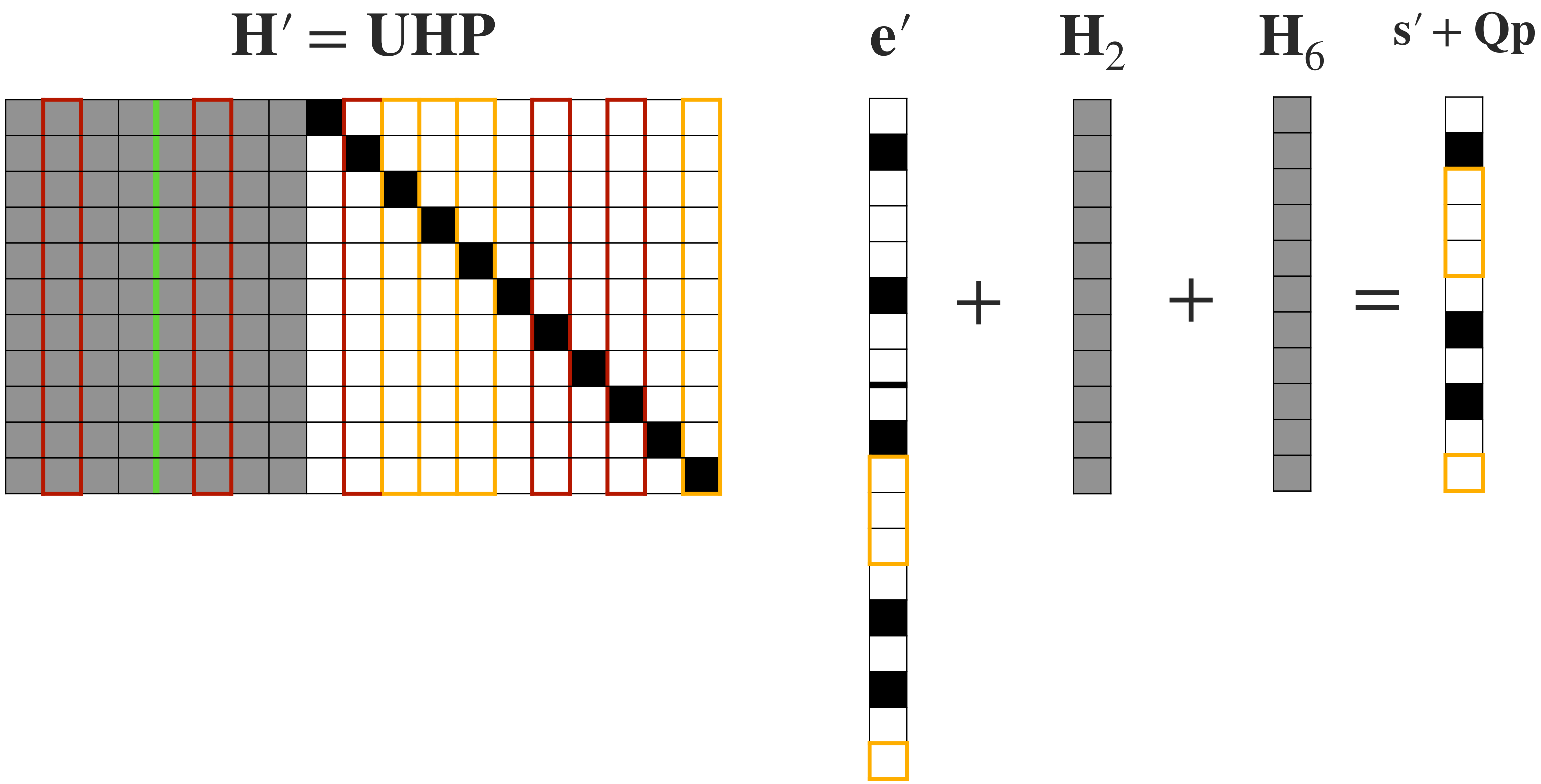


Probability that we are in the correct configuration: $\dfrac{\dbinom{n-k-l}{t-p}\dbinom{k}{p}}{\dbinom{n}{t}}$.

Stern's attack

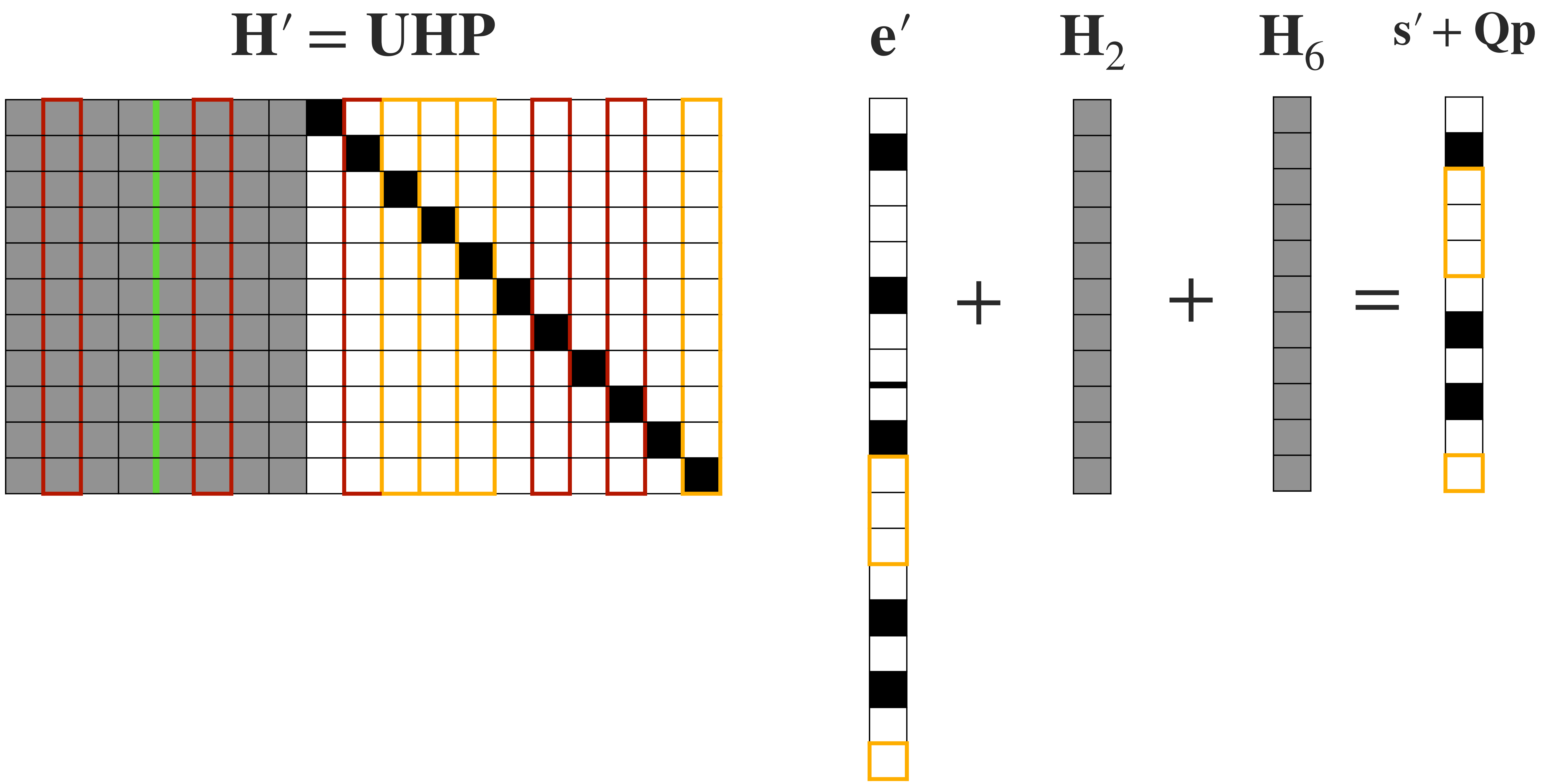


Stern’s attack



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 $\mathbf{Q}$.

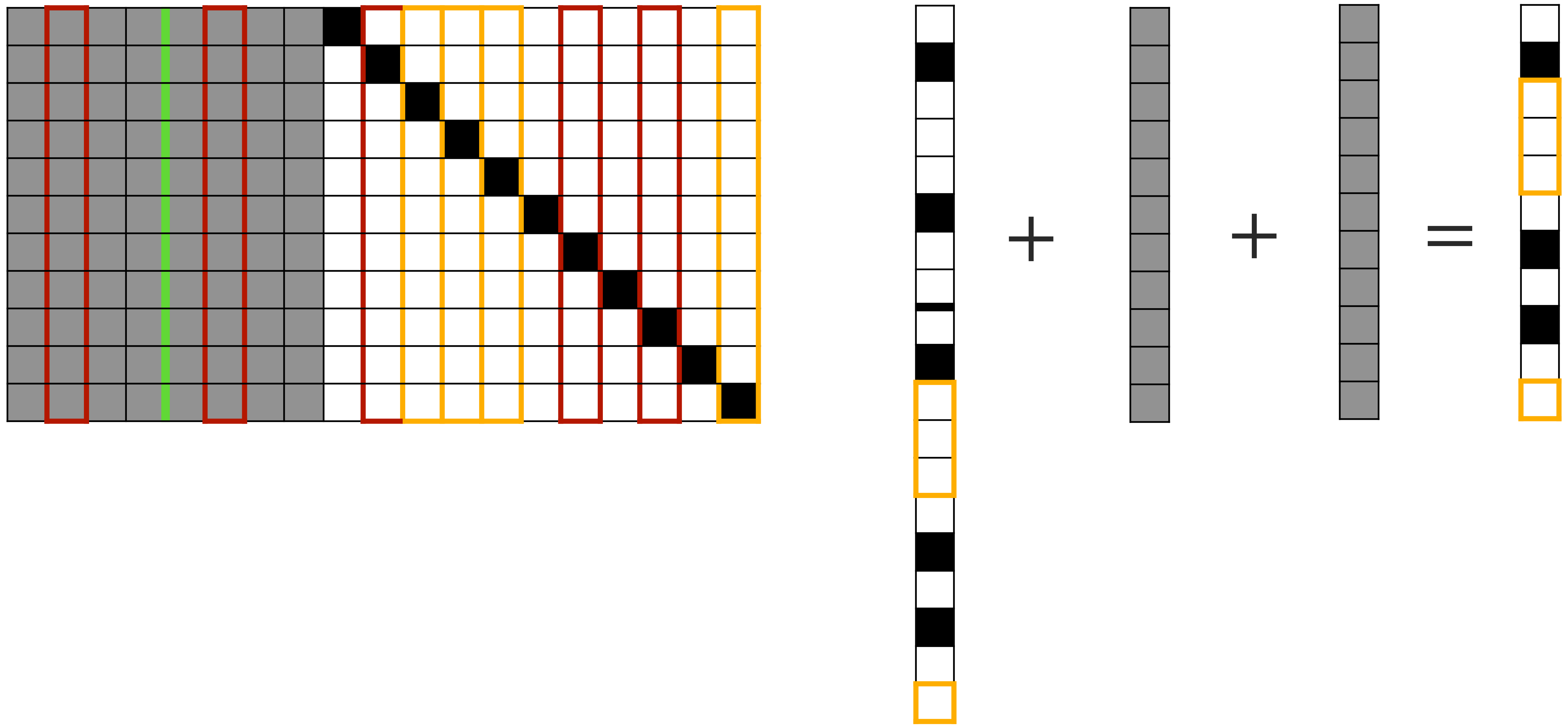
Stern’s attack



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.

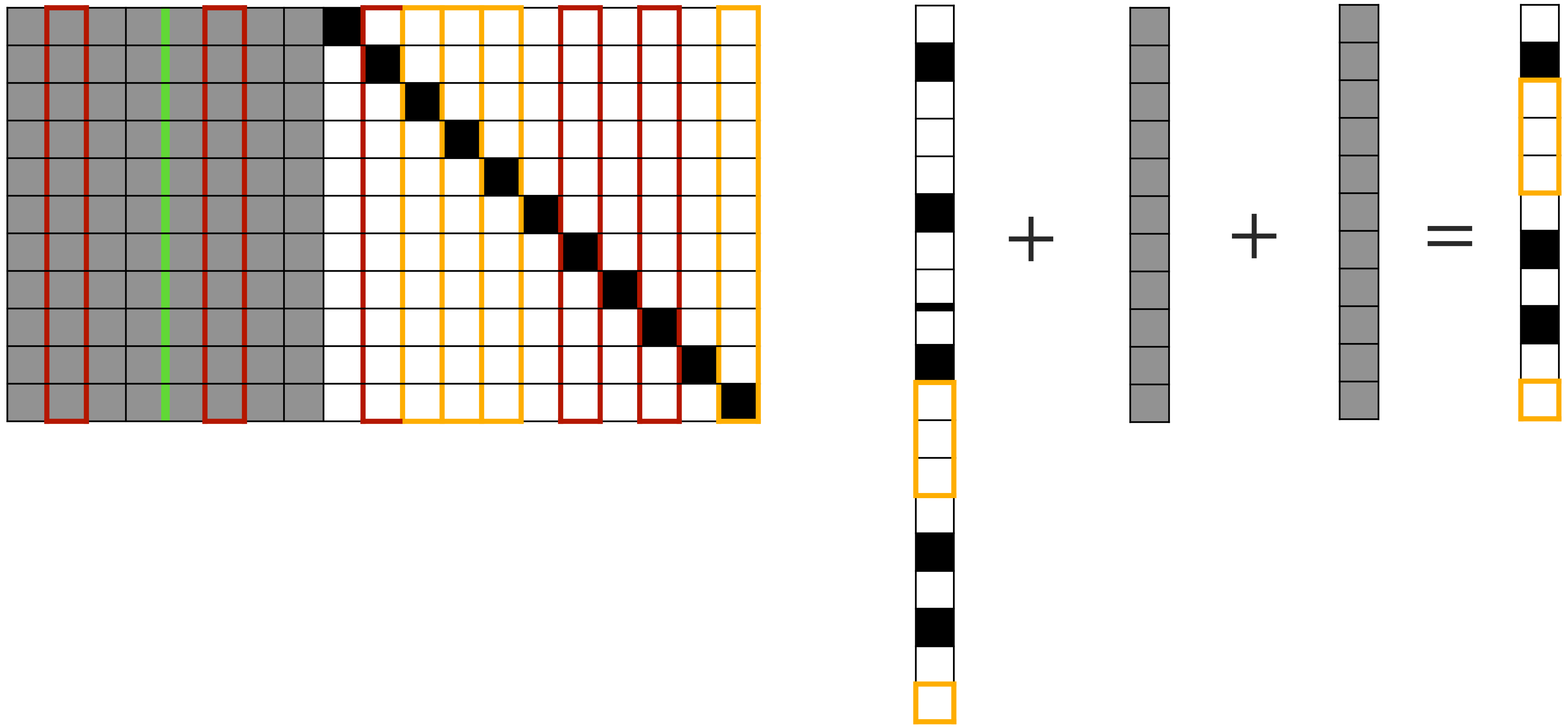
Stern's attack



- Pick a subset L of l rows: $\mathbf{H}_L$.
- Permute $\mathbf{H}$ and bring to systematic form (then $\mathbf{H}_L = \begin{pmatrix}\mathbf{Q}_L & \mathbf{I}_L\end{pmatrix}$).
- Split $\mathbf{Q}$ into two disjoint parts: $\mathbf{Q} = (\mathbf{A} \quad \mathbf{B})$.
- 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 $\mathrm{wt}(\mathbf{s}' + \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b}) = t - p$ then put $\mathbf{e}' = \mathbf{a}\,||\,\mathbf{b}\,||\,(\mathbf{s}' + \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b})$. Output unpermuted version of $\mathbf{e}$.
- Else return to the second step and rerandomize.

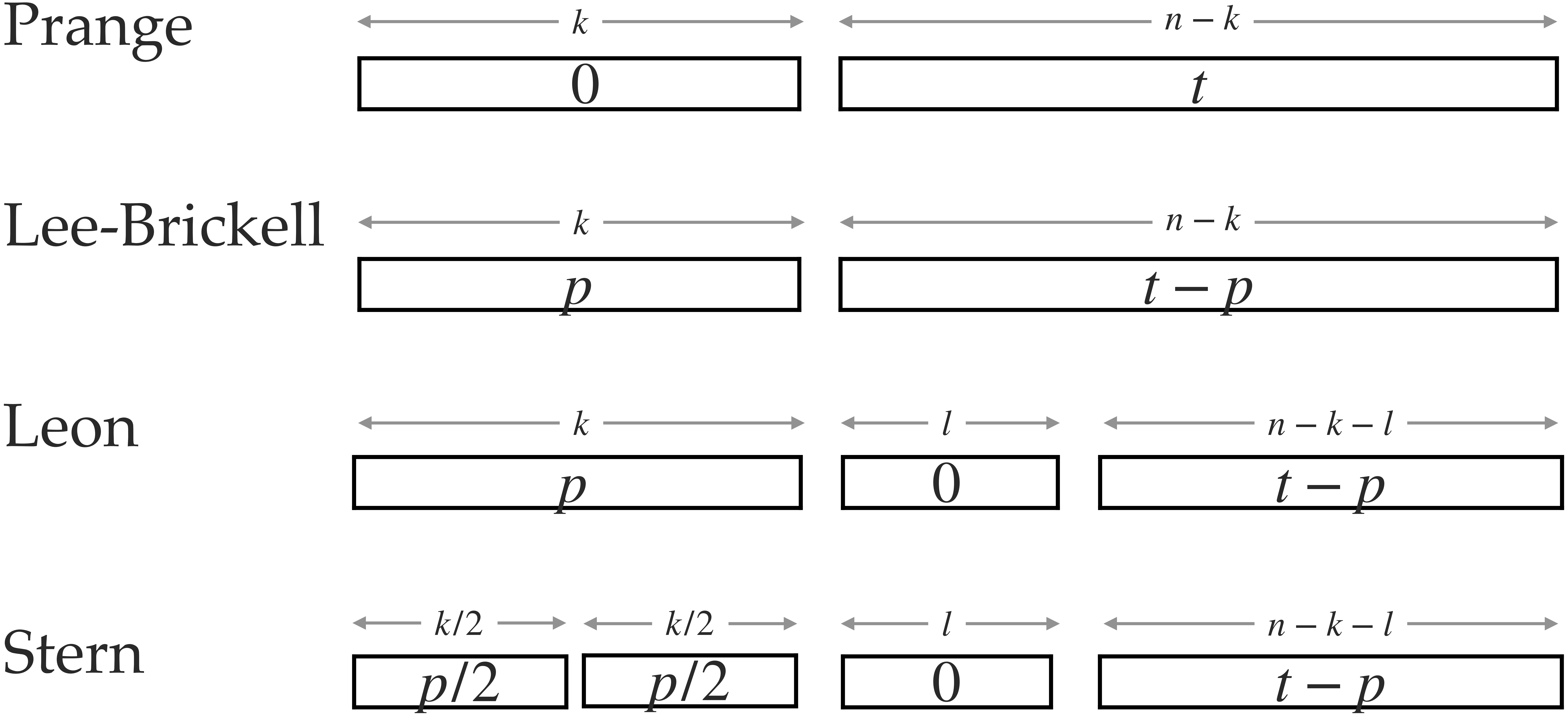
a and **b** are vectors chosen from $W = \{\mathbf{w} \in \mathbb{F}_2^{k/2} \,|\, \mathrm{wt}(\mathbf{w}) = \frac{p}{2}\}$

Stern’s attack: complexity



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

ISD algorithms summary



Prange	k 0	$n-k$ t		
Lee-Brickell	k p	$n-k$ $t-p$		
Leon	k p	l 0	$n-k-l$ $t-p$	
Stern	$k/2$ $p/2$	$k/2$ $p/2$	l 0	$n-k-l$ $t-p$

Next time:

MPC-in-the-Head construction