Performance of SPC product codes under the erasure channel

Sara D. Cardell[1] Joan-Josep Climent[1] Alberto López Martín[2]

[1] Universitat d'Alacant, Spain
[2] Instituto Nacional de Matemática Pura e Aplicada, Brazil

ALCOMA 2015

Outline

Outline

Erasure channel

Elias, P.: Coding for noisy channels.
In: IRE International Convention Record, part 4, pp. 37–46 (1955)

Properties

- Each sent symbol is either correctly received or considered as erased.
- Each codeword symbol is lost with a fixed independent probability.
- An $[n, k, d]$-code can recover up to $d - 1$ erasures.

Erasure channel

Elias, P.: Coding for noisy channels.
In: IRE International Convention Record, part 4, pp. 37–46 (1955)

Properties

- Each sent symbol is either correctly received or considered as erased.
- Each codeword symbol is lost with a fixed independent probability.
- An $[n, k, d]$-code can recover up to $d - 1$ erasures.

Erasure channel

Elias, P.: Coding for noisy channels.
In: IRE International Convention Record, part 4, pp. 37–46 (1955)

Properties

- Each sent symbol is either correctly received or considered as erased.
- Each codeword symbol is lost with a fixed independent probability.
- An $[n, k, d]$-code can recover up to $d - 1$ erasures.

Erasure channel

Elias, P.: Coding for noisy channels.
In: IRE International Convention Record, part 4, pp. 37–46 (1955)

Properties

- Each sent symbol is either correctly received or considered as erased.
- Each codeword symbol is lost with a fixed independent probability.
- An $[n, k, d]$-code can recover up to $d - 1$ erasures.

Product code

Let $\mathbb{F}_q$ be the Galois field with q elements.

Definition

A **linear product code** $\mathcal{C}$ over $\mathbb{F}_q$ is formed from two other linear codes $\mathcal{C}^-$ and $\mathcal{C}^|$ with parameters $[n^-, k^-, d^-]$ and $[n^|, k^|, d^|]$ over $\mathbb{F}_q$, respectively. The product code $\mathcal{C}$ will have parameters $[n^- n^|, k^- k^|, d^- d^|]$ over $\mathbb{F}_q$.

Over the erasure channel, the product code corrects up to $d^- d^| - 1$ erasures.

Product code

Let $\mathbb{F}_q$ be the Galois field with q elements.

Definition

A **linear product code** $\mathcal{C}$ over $\mathbb{F}_q$ is formed from two other linear codes $\mathcal{C}^{-}$ and $\mathcal{C}^{|}$ with parameters $[n^{-}, k^{-}, d^{-}]$ and $[n^{|}, k^{|}, d^{|}]$ over $\mathbb{F}_q$, respectively. The product code $\mathcal{C}$ will have parameters $[n^{-}n^{|}, k^{-}k^{|}, d^{-}d^{|}]$ over $\mathbb{F}_q$.

Over the erasure channel, the product code corrects up to $d^{-}d^{|} - 1$ erasures.

Product code

Let $\mathbb{F}_q$ be the Galois field with q elements.

Definition

A **linear product code** $\mathcal{C}$ over $\mathbb{F}_q$ is formed from two other linear codes $\mathcal{C}^{-}$ and $\mathcal{C}^{|}$ with parameters $[n^{-}, k^{-}, d^{-}]$ and $[n^{|}, k^{|}, d^{|}]$ over $\mathbb{F}_q$, respectively. The product code $\mathcal{C}$ will have parameters $[n^{-}n^{|}, k^{-}k^{|}, d^{-}d^{|}]$ over $\mathbb{F}_q$.

Over the erasure channel, the product code corrects up to $d^{-}d^{|} - 1$ erasures.

Product code

The codewords of length $n^{-} n^{|}$ can be seen as arrays with size $n^{-} \times n^{|}$ in a way that the columns are codewords of $\mathcal{C}^{|}$ and the rows are codewords of $\mathcal{C}^{-}$.

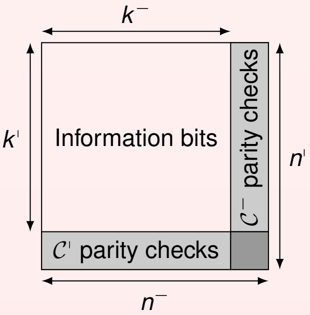


Figure: Codeword of a product code, with systematic encoding

SPC code

Definition

A **single parity-check code** is a linear binary code with parameters $[n, n-1, 2]$.

- The single parity-check (SPC) code is a very popular error detection code, since it is very easy to implement.
- This codes can correct one single erasure over the erasure channel.

SPC code

Definition

A **single parity-check code** is a linear binary code with parameters $[n, n-1, 2]$.

- The single parity-check (SPC) code is a very popular error detection code, since it is very easy to implement.
- This codes can correct one single erasure over the erasure channel.

SPC code

Definition

A **single parity-check code** is a linear binary code with parameters $[n, n-1, 2]$.

- The single parity-check (SPC) code is a very popular error detection code, since it is very easy to implement.
- This codes can correct one single erasure over the erasure channel.

SPC product code

- $\mathcal{C}^{-} = \mathcal{C}^{|}$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^{-} \otimes \mathcal{C}^{|}$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

SPC product code

- $\mathcal{C}^{-} = \mathcal{C}^{|}$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^{-} \otimes \mathcal{C}^{|}$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

SPC product code

- $\mathcal{C}^{-} = \mathcal{C}^{|}$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^{-} \otimes \mathcal{C}^{|}$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

SPC product code

- $\mathcal{C}^{-} = \mathcal{C}^{|}$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^{-} \otimes \mathcal{C}^{|}$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

SPC product code

- $\mathcal{C}^- = \mathcal{C}^|$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^- \otimes \mathcal{C}^|$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

SPC product code

- $\mathcal{C}^- = \mathcal{C}^|$ is a linear binary code with parameters $[n, n-1, 2]$.
- We consider the product code $\mathcal{C} = \mathcal{C}^- \otimes \mathcal{C}^|$.
- The parameters of the product code are $[n^2, (n-1)^2, 4]$.
- The code $\mathcal{C}$ corrects only 3 erasures.

In some special cases this code can correct more than 3 erasures.

Erasure pattern

Definition

An **erasure pattern** of size $m \times m$, with t erasures, where $0 \leq t \leq m^2$ and $1 \leq m \leq n$, is an array of size $m \times m$ where t of the entries correspond to the position of the erasures.

Decoding

- An erasure pattern of size $n \times n$ corresponds to words of size $n \times n$, where the position of the erasures is the unique information we consider.
- Given a received word with t erasures, the decoder will perform iterative row-wise and column-wise decoding to recover the erased bits.
- When a single bit is erased in a row or column, it can be recovered.
- If more than one bit is erased in a row (column), it is skipped.
- Decoding is performed until no further recovery is possible.

Decoding

- An erasure pattern of size $n \times n$ corresponds to words of size $n \times n$, where the position of the erasures is the unique information we consider.
- Given a received word with t erasures, the decoder will perform iterative row-wise and column-wise decoding to recover the erased bits.
- When a single bit is erased in a row or column, it can be recovered.
- If more than one bit is erased in a row (column), it is skipped.
- Decoding is performed until no further recovery is possible.

Decoding

- An erasure pattern of size $n \times n$ corresponds to words of size $n \times n$, where the position of the erasures is the unique information we consider.
- Given a received word with t erasures, the decoder will perform iterative row-wise and column-wise decoding to recover the erased bits.
- When a single bit is erased in a row or column, it can be recovered.
- If more than one bit is erased in a row (column), it is skipped.
- Decoding is performed until no further recovery is possible.

Decoding

- An erasure pattern of size $n \times n$ corresponds to words of size $n \times n$, where the position of the erasures is the unique information we consider.
- Given a received word with t erasures, the decoder will perform iterative row-wise and column-wise decoding to recover the erased bits.
- When a single bit is erased in a row or column, it can be recovered.
- If more than one bit is erased in a row (column), it is skipped.
- Decoding is performed until no further recovery is possible.

Decoding

- An erasure pattern of size $n \times n$ corresponds to words of size $n \times n$, where the position of the erasures is the unique information we consider.
- Given a received word with t erasures, the decoder will perform iterative row-wise and column-wise decoding to recover the erased bits.
- When a single bit is erased in a row or column, it can be recovered.
- If more than one bit is erased in a row (column), it is skipped.
- Decoding is performed until no further recovery is possible.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

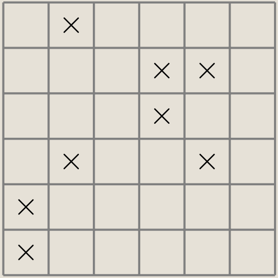
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



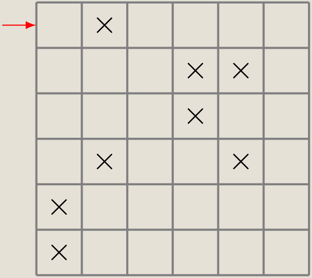
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6,5,2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36,25,4]$.

			×	×	
			×		
	×			×	
×					
×					

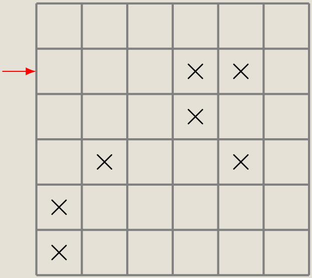
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



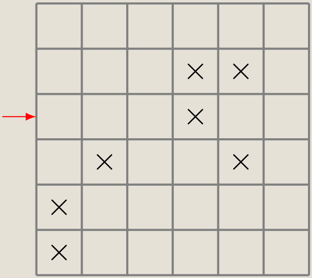
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



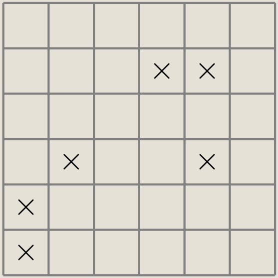
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

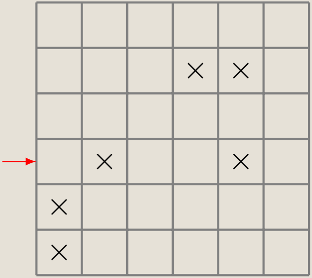


- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



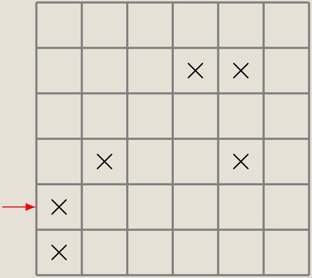
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



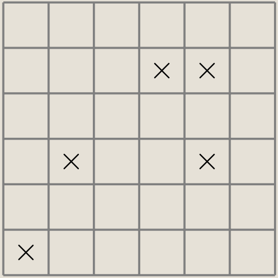
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



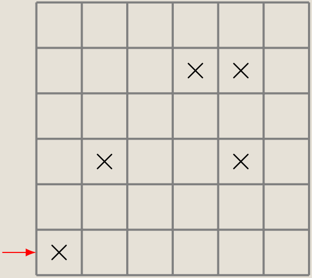
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



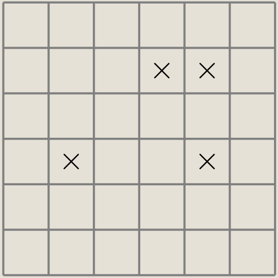
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

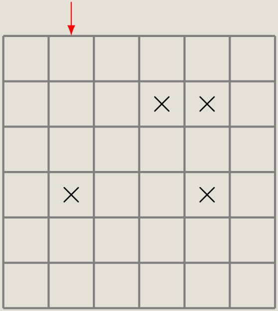


- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



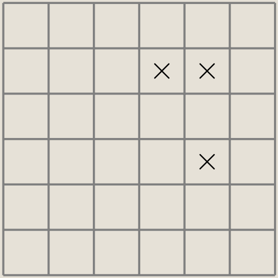
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



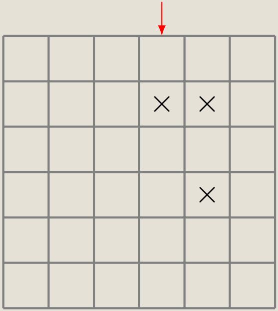
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

				×	
				×	

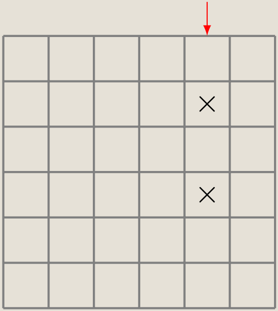
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6,5,2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36,25,4]$.



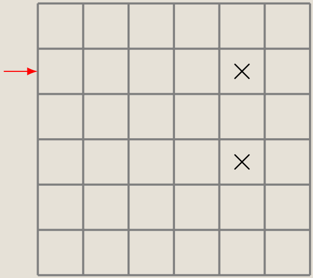
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



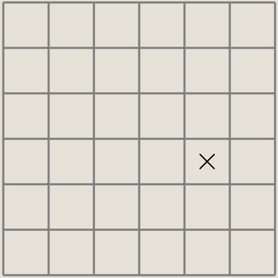
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



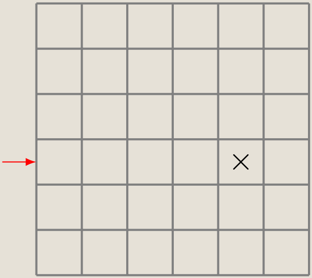
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



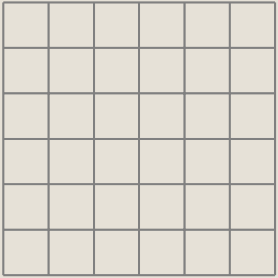
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



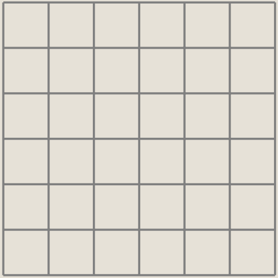
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.



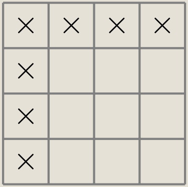
- The code is supposed to correct 3 erasures.
- We have corrected 8 erasures.

Decoding

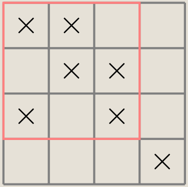
Definition

An erasure pattern of size $m \times m$ is said to be **uncorrectable** if and only if it contains a subpattern of size $l \times l$, $l \leq m$, such that each row and each column have two or more erasures.

Example



(a) Correctable erasure pattern of size 4×4 with 7 erasures



(b) Uncorrectable erasure pattern of size 4×4 with 7 erasures

Erasure pattern

Example

×	×		
	×	×	
		×	×
×			×

Figure: Uncorrectable erasure pattern of size 4×4 with 7 erasures

Erasure pattern

Example

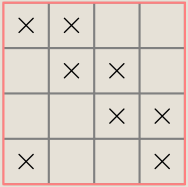


Figure: Uncorrectable erasure pattern of size 4×4 with 7 erasures

Decoding

- Erasure patterns of size $n \times n$ with 3 erasures or less are always correctable.
- Erasure patterns of size $n \times n$ with t erasures, $4 \leq t \leq 2n - 1$ may or may not be correctable.

Decoding

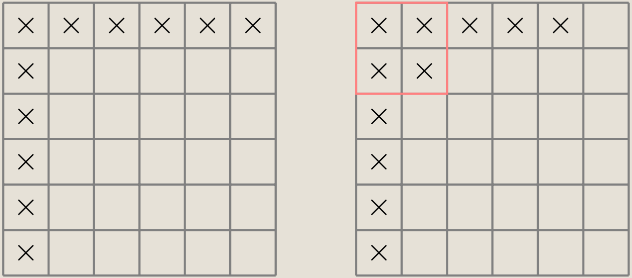
- Erasure patterns of size $n \times n$ with 3 erasures or less are always correctable.
- Erasure patterns of size $n \times n$ with t erasures, $4 \leq t \leq 2n - 1$ may or may not be correctable.

Decoding

Example (cont.)

Consider the SPC code $\widehat{\mathcal{C}}$ with parameters $[6, 5, 2]$.

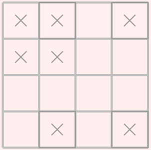
We can construct the binary product code $\mathcal{C} = \widehat{\mathcal{C}} \otimes \widehat{\mathcal{C}}$ with parameters $[36, 25, 4]$.

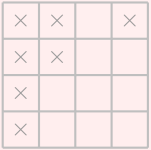


Classification

Definition

An uncorrectable erasure pattern is said to be **strict** if none of the erasures can be corrected. Equally, an uncorrectable erasure pattern is said to be **partial** if it can be partially corrected.

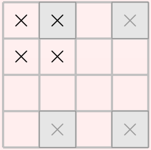




Strict uncorrectable erasure patterns

Lemma

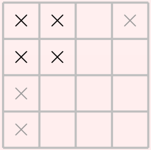
An strict uncorrectable erasure pattern contains two or more erasures in each row and column in error.



Partial uncorrectable erasure pattern

Lemma

A partial uncorrectable erasure pattern always contains an strict uncorrectable erasure pattern.



Idea

Purpose

We would like to count the number of uncorrectable erasure patterns of size $n \times n$ with t erasures, $4 \leq t \leq 2n - 1$.

In this work, we count the number of strict uncorrectable erasure patterns.

Idea

Purpose

We would like to count the number of uncorrectable erasure patterns of size $n \times n$ with t erasures, $4 \leq t \leq 2n - 1$.

In this work, we count the number of strict uncorrectable erasure patterns.

Partition of an integer

Definition

If t is a positive integer, then a **partition** of t is a non-increasing sequence of positive integers $(\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_p)$ such that $\sum_{i=1}^{p} \lambda_i = t$.

We denote by $\mathcal{P}_t$ the set of possible partitions of the integer t.

Example

For example, the set of partitions of 6 is given by

$$\mathcal{P}_6 = \{(6), (5,1), (4,2), (4,1,1), (3,3), (3,2,1), (3,1,1,1), (2,2,2), (2,2,1,1), (2,1,1,1,1), (1,1,1,1,1,1)\}.$$

Partition of an integer

Definition

If t is a positive integer, then a **partition** of t is a non-increasing sequence of positive integers $(\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_p)$ such that $\sum_{i=1}^{p} \lambda_i = t$.

We denote by $\mathcal{P}_t$ the set of possible partitions of the integer t.

Example

For example, the set of partitions of 6 is given by

$$\mathcal{P}_6 = \{(6), (5,1), (4,2), (4,1,1), (3,3), (3,2,1), (3,1,1,1), (2,2,2), (2,2,1,1), (2,1,1,1,1), (1,1,1,1,1,1)\}.$$

Partition of an integer

Definition

If t is a positive integer, then a **partition** of t is a non-increasing sequence of positive integers $(\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_p)$ such that $\sum_{i=1}^{p} \lambda_i = t$.

We denote by $\mathcal{P}_t$ the set of possible partitions of the integer t.

Example

For example, the set of partitions of 6 is given by

$$\mathcal{P}_6 = \{(6), (5,1), (4,2), (4,1,1), (3,3), (3,2,1), (3,1,1,1), (2,2,2), \\ (2,2,1,1), (2,1,1,1,1), (1,1,1,1,1,1)\}.$$

Conjugate partition

Definition

Let us consider a partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ of t. The **conjugate** of λ is defined as the vector $\lambda^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_{p'}^*)$ where

$$\lambda_j^* = |\{i \mid 1 \leq i \leq p, \lambda_i \geq j\}| .$$

Notice that both, λ and λ^*, are partitions of the same integer t.

Conjugate partition

Definition

Let us consider a partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ of t. The **conjugate** of λ is defined as the vector $\lambda^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_{p'}^*)$ where

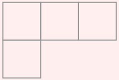
$$\lambda_j^* = |\{i \mid 1 \leq i \leq p, \lambda_i \geq j\}| .$$

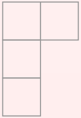
Notice that both, λ and λ^*, are partitions of the same integer t.

Conjugate partition

Example

$\lambda = (3, 1)$ and $\lambda^* = (2, 1, 1)$ are conjugate partitions of $t = 4$.





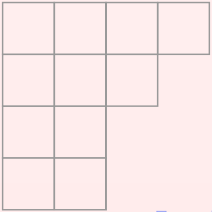
Young diagram and Young tableaux

Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ be two partitions of the same integer t.

Definition

A **Young diagram** of shape λ is an arrangement of t boxes in p rows where there are λ_i boxes in row i, with $i = 1, 2, \ldots, p$, and these boxes are left justified.

$$\lambda = (4, 3, 2, 2)$$



Young diagram and Young tableaux

Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ be two partitions of the same integer t.

Definition

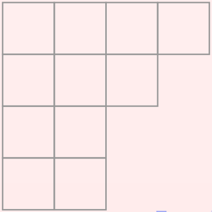
A **Young tableau** of shape λ and content μ is obtained from a Young diagram of shape λ by inserting in each box one of the integers $1, 2, \ldots, q$ in such a way that the following conditions hold:

i) the elements in each row are non-decreasing,

ii) the elements in each column are strictly increasing,

iii) the integer j occurs μ_j times, with $j = 1, 2, \ldots, q$.

$$\lambda = (4, 3, 2, 2)$$



Young diagram and Young tableaux

Let $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ be two partitions of the same integer t.

Definition

A **Young tableau** of shape λ and content μ is obtained from a Young diagram of shape λ by inserting in each box one of the integers $1, 2, \ldots, q$ in such a way that the following conditions hold:

i) the elements in each row are non-decreasing,

ii) the elements in each column are strictly increasing,

iii) the integer j occurs μ_j times, with $j = 1, 2, \ldots, q$.

$\lambda = (4, 3, 2, 2)$
$\mu = (3, 3, 2, 2, 1)$

1	1	1	2
2	2	3	
3	4		
4	5		

Young diagram and Young tableaux

This is not the only option; there are 5 more Young tableaux with these properties.

1	1	1	2
2	2	3	
3	4		
4	5		

1	1	1	2
2	2	4	
3	3		
4	5		

1	1	1	2
2	2	5	
3	3		
4	4		

1	1	1	3
2	2	2	
3	4		
4	5		

1	1	1	4
2	2	2	
3	3		
4	5		

1	1	1	5
2	2	2	
3	3		
4	4		

Kotska numbers

Definition

If $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ are two partitions of the same integer t, then the **Kostka number** denoted by $\kappa_{\lambda,\mu}$, is the number of Young tableaux of shape λ and content μ.

Example

We want to compute $\kappa_{(3,2,1),(3,2,1)}$.

We have to count the number of Young tableaux of shape $\lambda = (3, 2, 1)$ and content $\mu = (3, 2, 1)$.

1	1	1
2	2	
3		

Thus, $\kappa_{(3,2,1),(3,2,1)} = 1$.

Kotska numbers

Definition

If $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ are two partitions of the same integer t, then the **Kostka number** denoted by $\kappa_{\lambda,\mu}$, is the number of Young tableaux of shape λ and content μ.

Example

We want to compute $\kappa_{(3,2,1),(3,2,1)}$.

We have to count the number of Young tableaux of shape $\lambda = (3, 2, 1)$ and content $\mu = (3, 2, 1)$.

1	1	1
2	2	
3		

Thus, $\kappa_{(3,2,1),(3,2,1)} = 1$.

Kotska numbers

Definition

If $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ are two partitions of the same integer t, then the **Kostka number** denoted by $\kappa_{\lambda,\mu}$, is the number of Young tableaux of shape λ and content μ.

Example

We want to compute $\kappa_{(3,2,1),(3,2,1)}$.
We have to count the number of Young tableaux of shape $\lambda = (3, 2, 1)$ and content $\mu = (3, 2, 1)$.

1	1	1
2	2	
3		

Thus, $\kappa_{(3,2,1),(3,2,1)} = 1$.

Kotska numbers

Definition

If $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_p)$ and $\mu = (\mu_1, \mu_2, \ldots, \mu_q)$ are two partitions of the same integer t, then the **Kostka number** denoted by $\kappa_{\lambda,\mu}$, is the number of Young tableaux of shape λ and content μ.

Example

We want to compute $\kappa_{(3,2,1),(3,2,1)}$.
We have to count the number of Young tableaux of shape $\lambda = (3, 2, 1)$ and content $\mu = (3, 2, 1)$.

1	1	1
2	2	
3		

Thus, $\kappa_{(3,2,1),(3,2,1)} = 1$.

Counting binary matrices

- Let A be a binary matrix of size $m_1 \times m_2$.
- Let $R = (r_1, r_2, \ldots, r_{m_1})$ be the vector where r_i is the sum of the elements in row i of matrix A.
- Let $C = (c_1, c_2, \ldots, c_{m_2})$ be the vector where c_j is the sum of the elements in row j of matrix A.
- Note that $r_1 + r_2 + \cdots + r_{m_1} = c_1 + c_2 + \cdots + c_{m_2}$ and call this number t.

Counting binary matrices

R. A. Brualdi, Algorithms for constructing (0,1)-matrices with prescribed row and column sum vectors, Discrete Mathematics 306 (23) (2006) 3054–3062.

Theorem

The number of binary matrices with R and C as the row sum and the column sum, respectively, is given by

$$\sum_{\lambda \in \mathcal{P}_t} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C}.$$

Counting binary matrices

R. A. Brualdi, Algorithms for constructing (0,1)-matrices with prescribed row and column sum vectors, Discrete Mathematics 306 (23) (2006) 3054–3062.

Theorem

The number of binary matrices with R and C as the row sum and the column sum, respectively, is given by

$$\sum_{\lambda \in \mathcal{P}_t} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C}.$$

Outline

Counting patterns

Purpose

We would like to count the number of uncorrectable erasure patterns of size $n \times n$ with t erasures, $4 \leq t \leq 2n - 1$.

Uncorrectable patterns with four erasures

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

The only uncorrectable erasure pattern of 4 erasures is formed by a square:

The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Uncorrectable patterns with four erasures

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

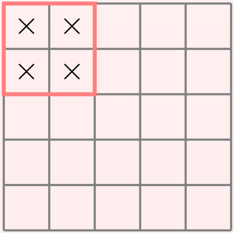
The only uncorrectable erasure pattern of 4 erasures is formed by a square:

The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Uncorrectable patterns with four erasures

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

The only uncorrectable erasure pattern of 4 erasures is formed by a square:

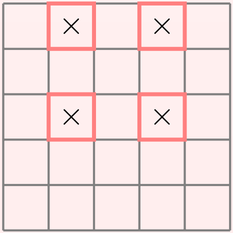


The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Uncorrectable patterns with four erasures

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

The only uncorrectable erasure pattern of 4 erasures is formed by a square:

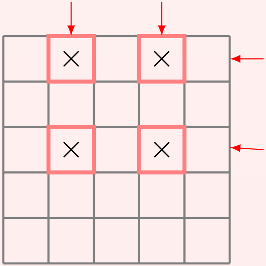


The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Uncorrectable patterns with four erasures

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

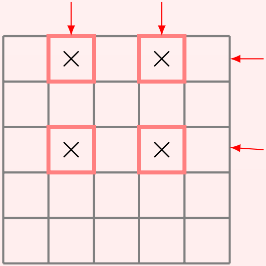
The only uncorrectable erasure pattern of 4 erasures is formed by a square:



The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Assume we have a codeword of size $n \times n$ and that 4 erasures have occurred.

The only uncorrectable erasure pattern of 4 erasures is formed by a square:



The total number of uncorrectable erasure patterns with 4 erasures is $\binom{n}{2}^2$.

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:

The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2 \binom{n^2-4}{1}$.

Uncorrectable patterns with five erasures

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

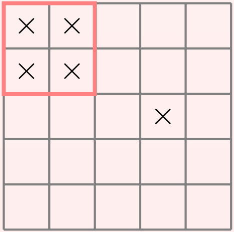
The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:

The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2 \binom{n^2-4}{1}$.

Uncorrectable patterns with five erasures

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:

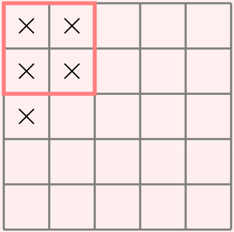


The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2 \binom{n^2-4}{1}$.

Uncorrectable patterns with five erasures

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:

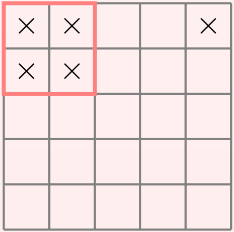


The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2 \binom{n^2-4}{1}$.

Uncorrectable patterns with five erasures

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:

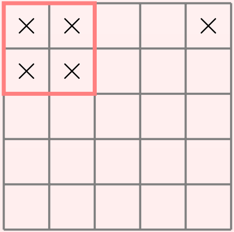


The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2\binom{n^2-4}{1}$.

Uncorrectable patterns with five erasures

Assume we have a codeword of size $n \times n$ and that 5 erasures have occurred.

The only uncorrectable erasure pattern of 5 erasures is formed by a square (uncorrectable pattern of size 2×2) and one extra erasure:



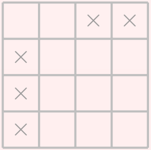
The total number of uncorrectable erasure patterns with 5 erasures is $\binom{n}{2}^2 \binom{n^2-4}{1}$.

Uncorrectable patterns with six erasures

The total number of uncorrectable erasure patterns with 6 erasures is $\binom{n}{2}^2\binom{n^2-4}{2} - 4\binom{n}{2}\binom{n}{3} + 6\binom{n}{3}^2$.

Erasure patterns/ binary matrices

Let us represent an erasure pattern of size $n \times n$ by a binary matrix of size $n \times n$ where there is 1 in the erasure positions and 0 otherwise.

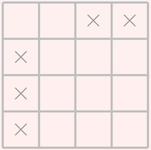


$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

In this work, our purpose is to count the number of strict uncorrectable erasure patterns of size $n \times n$ with t erasures. Equivalently, we want to find all matrices of size $n \times n$ with t ones (and 2 or more ones in each non-zero row and non-zero column).

Erasure patterns/ binary matrices

Let us represent an erasure pattern of size $n \times n$ by a binary matrix of size $n \times n$ where there is 1 in the erasure positions and 0 otherwise.



$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

In this work, our purpose is to count the number of strict uncorrectable erasure patterns of size $n \times n$ with t erasures. Equivalently, we want to find all matrices of size $n \times n$ with t ones (and 2 or more ones in each non-zero row and non-zero column).

Uncorrectable strict erasure patterns

Notation

- The partitions that we will consider will have all length n.
- If a partition has length $r < n$, it will be filled in with $n - r$ zeros.
- For example, $(6, 1)$ is a partition of 7 with length 2, but if we are considering partitions of length 4, we write $(6, 1, 0, 0)$.

Uncorrectable strict erasure patterns

Notation

- The partitions that we will consider will have all length n.
- If a partition has length $r < n$, it will be filled in with $n - r$ zeros.
- For example, $(6, 1)$ is a partition of 7 with length 2, but if we are considering partitions of length 4, we write $(6, 1, 0, 0)$.

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6^3 = \{(2, 2, 2), (3, 3, 0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{(6), (5,1), (4,2), (4,1,1), (3,3), (3,2,1), (3,1,1,1), (2,2,2), (2,2,1,1), (2,1,1,1,1), (1,1,1,1,1,1)\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{(6), (5,1), (4,2), (4,1,1), (3,3), (3,2,1), \cancel{(3,1,1,1)}, (2,2,2), \cancel{(2,2,1,1)}, \cancel{(2,1,1,1,1)}, \cancel{(1,1,1,1,1,1)}\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \dots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{(6,0,0), (5,1,0), (4,2,0), (4,1,1), (3,3,0), (3,2,1), \xcancel{(3,1,1,1)}, (2,2,2), \xcancel{(2,2,1,1)}, \xcancel{(2,1,1,1,1)}, \xcancel{(1,1,1,1,1,1)}\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{(6,0,0), \cancel{(5,1,0)}, (4,2,0), \cancel{(4,1,1)}, (3,3,0), \cancel{(3,2,1)}, \cancel{(3,1,1,1)}, (2,2,2), \cancel{(2,2,1,1)}, \cancel{(2,1,1,1,1)}, \cancel{(1,1,1,1,1,1)}\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{(6,0,0), (5,1,0), (4,2,0), (4,1,1), (3,3,0), (3,2,1), (3,1,1,1), (2,2,2), (2,2,1,1), (2,1,1,1,1), (1,1,1,1,1,1)\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Definition

Given two positive integers n and t, the set $\mathcal{P}_t^n$ is a subset of the set of partitions of t of length n defined as

$$\mathcal{P}_t^n = \{\lambda \in \mathcal{P}_t \mid \lambda_i \neq 1, \lambda_i \leq n, i = 1, 2, \ldots, n\}.$$

Example

Consider $t = 6$ and $n = 3$.

$$\mathcal{P}_6 = \{\cancel{(6,0,0)}, \cancel{(5,1,0)}, \cancel{(4,2,0)}, \cancel{(4,1,1)}, (3,3,0), \cancel{(3,2,1)}, \cancel{(3,1,1,1)}, (2,2,2), \cancel{(2,2,1,1)}, \cancel{(2,1,1,1,1)}, \cancel{(1,1,1,1,1,1)}\}.$$

$$\mathcal{P}_6^3 = \{(2,2,2), (3,3,0)\}$$

Notation

Lemma

For a partition $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$ of a positive integer t and length n, the number of possible combinations of the elements in λ is given by

$$\delta_\lambda = \frac{n!}{\eta_1!\eta_2!\cdots\eta_n!},$$

where $\eta_i = |\{j \mid \lambda_j = i\}|$, for $i = 1, 2, \ldots, n$.

Example

$\lambda = (3, 1, 1, 0) \in \mathcal{P}_5^4$

$$\delta_\lambda = \frac{4!}{1!2!1!} = 12.$$

Notation

Lemma

For a partition $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_n)$ of a positive integer t and length n, the number of possible combinations of the elements in λ is given by

$$\delta_\lambda = \frac{n!}{\eta_1!\eta_2!\cdots\eta_n!},$$

where $\eta_i = |\{j \mid \lambda_j = i\}|$, for $i = 1, 2, \dots, n$.

Example

$\lambda = (3, 1, 1, 0) \in \mathcal{P}_5^4$

$$\delta_\lambda = \frac{4!}{1!2!1!} = 12.$$

Main result

Theorem

The number of strict uncorrectable erasure patterns of size $n \times n$ with t erasures is given by

$$\sum_{R,C \in \mathcal{P}_t^n} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_t} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C}. \qquad (1)$$

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\mathcal{P}_4 = \{(4), (3,1), (2,2), (2,1,1), (1,1,1,1)\}$$

$$\mathcal{P}_4^3 = \{(2,2,0)\} \longrightarrow C = R = (2,2,0)$$

$$\delta_R = \delta_C = \frac{3!}{2!1!} = 3$$

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C}$$

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\mathcal{P}_4 = \{(4), (3,1), (2,2), (2,1,1), (1,1,1,1)\}$$

$$\mathcal{P}_4^3 = \{(2,2,0)\} \longrightarrow C = R = (2,2,0)$$

$\delta_R = \delta_C = \frac{3!}{2!1!} = 3$

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C}$$

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\mathcal{P}_4 = \{(4), (3,1), (2,2), (2,1,1), (1,1,1,1)\}$$

$$\mathcal{P}_4^3 = \{(2,2,0)\} \longrightarrow C = R = (2,2,0)$$

$$\delta_R = \delta_C = \frac{3!}{2!1!} = 3$$

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C}$$

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\mathcal{P}_4 = \{(4), (3,1), (2,2), (2,1,1), (1,1,1,1)\}$$

$$\mathcal{P}_4^3 = \{(2,2,0)\} \longrightarrow C = R = (2,2,0)$$

$$\delta_R = \delta_C = \frac{3!}{2!1!} = 3$$

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C}$$

Main result

Example

λ	$\kappa_{\lambda,R}$	λ^*	$\kappa_{\lambda^*,C}$	$\kappa_{\lambda,R} \cdot \kappa_{\lambda^*,C}$
4	1	1, 1, 1, 1	0	0
3, 1	1	2, 1, 1	0	0
2, 2	1	2, 2	1	1
2, 1, 1	0	3, 1	1	0
1, 1, 1, 1	0	4	1	0
				1

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

×	×	
×	×	

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

×		×
×		×

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

	×	×
	×	×

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9$$

×	×	
×	×	

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

×		×
×		×

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

	×	×
	×	×

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\ \kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\ \kappa_{\lambda^*,C} = 9$$

×	×	
×	×	

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C\in\mathcal{P}_4^3} \delta_R\delta_C \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9 \sum_{\lambda\in\mathcal{P}_4} \kappa_{\lambda,R}\,\kappa_{\lambda^*,C} = 9$$

×		×
×		×

Main result

Example

Consider $n = 3$ and $t = 4$.

$$\sum_{R,C \in \mathcal{P}_4^3} \delta_R \delta_C \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9 \sum_{\lambda \in \mathcal{P}_4} \kappa_{\lambda,R}\, \kappa_{\lambda^*,C} = 9$$

	×	×
	×	×

Outline

Conclusions

So far...

- We know the number of uncorrectable erasure patterns of size $n \times n$ with t erasures when $t = 4, 5, 6, 7, 8$.
- Using Kotska numbers and Young tableaux, we can compute the number of strict uncorrectable erasure patterns of size $n \times n$.

Future work

- An erasure pattern of size $n \times n$ can be represented by a bipartite graph with $2n$ vertices (n in each vertex class).
- For $9 \leq t \leq 2n - 1$, the number of uncorrectable patterns can be computed taking partitions of $2n$ and considering connected components of the corresponding graph in each case.

Conclusions

So far...

- We know the number of uncorrectable erasure patterns of size $n \times n$ with t erasures when $t = 4, 5, 6, 7, 8$.
- Using Kotska numbers and Young tableaux, we can compute the number of strict uncorrectable erasure patterns of size $n \times n$.

Future work

- An erasure pattern of size $n \times n$ can be represented by a bipartite graph with $2n$ vertices (n in each vertex class).
- For $9 \leq t \leq 2n - 1$, the number of uncorrectable patterns can be computed taking partitions of $2n$ and considering connected components of the corresponding graph in each case.

Conclusions

So far...

- We know the number of uncorrectable erasure patterns of size $n \times n$ with t erasures when $t = 4, 5, 6, 7, 8$.
- Using Kotska numbers and Young tableaux, we can compute the number of strict uncorrectable erasure patterns of size $n \times n$.

Future work

- An erasure pattern of size $n \times n$ can be represented by a bipartite graph with $2n$ vertices (n in each vertex class).
- For $9 \leq t \leq 2n - 1$, the number of uncorrectable patterns can be computed taking partitions of $2n$ and considering connected components of the corresponding graph in each case.

Conclusions

So far...

- We know the number of uncorrectable erasure patterns of size $n \times n$ with t erasures when $t = 4, 5, 6, 7, 8$.
- Using Kotska numbers and Young tableaux, we can compute the number of strict uncorrectable erasure patterns of size $n \times n$.

Future work

- An erasure pattern of size $n \times n$ can be represented by a bipartite graph with $2n$ vertices (n in each vertex class).
- For $9 \leq t \leq 2n - 1$, the number of uncorrectable patterns can be computed taking partitions of $2n$ and considering connected components of the corresponding graph in each case.

Conclusions

So far...

- We know the number of uncorrectable erasure patterns of size $n \times n$ with t erasures when $t = 4, 5, 6, 7, 8$.
- Using Kotska numbers and Young tableaux, we can compute the number of strict uncorrectable erasure patterns of size $n \times n$.

Future work

- An erasure pattern of size $n \times n$ can be represented by a bipartite graph with $2n$ vertices (n in each vertex class).
- For $9 \leq t \leq 2n - 1$, the number of uncorrectable patterns can be computed taking partitions of $2n$ and considering connected components of the corresponding graph in each case.


Thank You

Performance of SPC product codes under the erasure channel

Sara D. Cardell[1] Joan-Josep Climent[1] Alberto López Martín[2]

[1] Universitat d'Alacant, Spain
[2] Instituto Nacional de Matemática Pura e Aplicada, Brazil

ALCOMA 2015

References

P. Elias, Coding for noisy channels, in: IRE International Convention Record, pt. 4, 1955, pp. 37–46.

M. A. Kousa, A novel approach for evaluating the performance of SPC product codes under erasure decoding 50 (1) (2002) 7–11.

M. A. Kousa, A. H. Mugaibel, Cell loss recovery using two-dimensional erasure correction for ATM networks, in: Proceedings of the Seventh International Conference on Telecommunication Systems, 1999, pp. 85–89.

J. M. Simmons, R. G. Gallager, Design of error detection scheme for class C service in ATM, IEEE/ACM Transactions on Networking 2 (1) (1994) 80–88.

D. M. Rankin, T. A. Gulliver, Single parity check product codes 49 (8) (2001) 1354–1362.

R. A. Brualdi, Algorithms for constructing (0,1)-matrices with prescribed row and column sum vectors, Discrete Mathematics 306 (23) (2006) 3054–3062.

D. E. Knuth, Permutations, matrices, and generalized young tableaux, Pacific Journal of Mathematics 34 (3) (1970) 707–727.