Session 3: An application to coding theory and cryptography

Oriol Serra

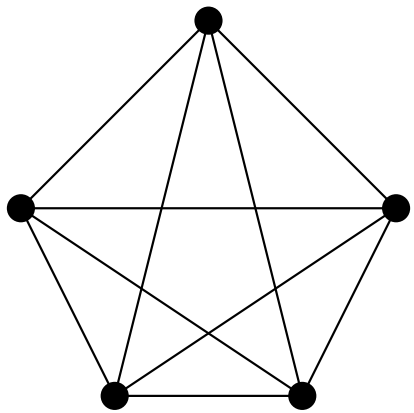
CIMPA-Indonesia School

Overview

Cycle codes of graphs

Cycle codes of graphs

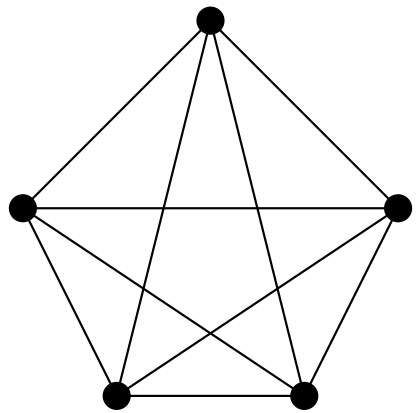
- $G = (V, E)$ undirected connected graph with no loops, no multiple edges.



$G = K_5$

Cycle codes of graphs

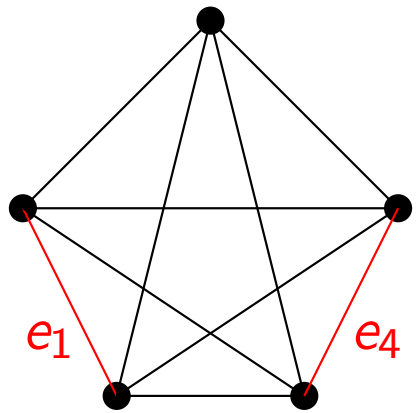
- $G = (V, E)$ undirected connected graph with no loops, no multiple edges.
- $E = \{e_1, e_2, \ldots, e_m\}$



$G = K_5$

Cycle codes of graphs

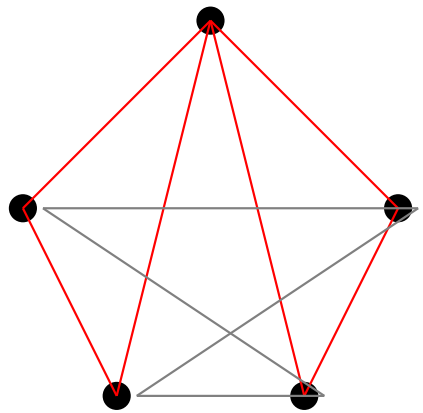
- $G = (V, E)$ undirected connected graph with no loops, no multiple edges.
- $E = \{e_1, e_2, \ldots, e_m\}$
- Subgraph $H \subset G \longleftrightarrow$ Characteristic vectors in $\{0,1\}^m$: $h_i = 1_H(e_i)$.



$$H = (1, 0, 0, 1, 0, 0, 0, 0, 0, 0)$$

Cycle codes of graphs

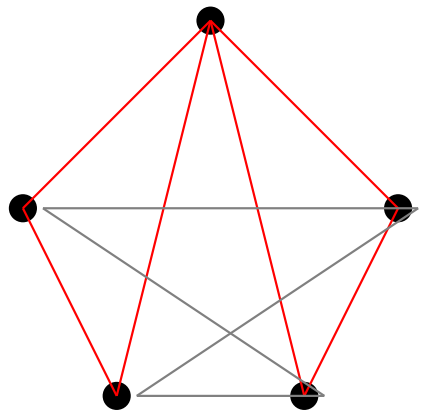
- $G = (V, E)$ undirected connected graph with no loops, no multiple edges.
- $E = \{e_1, e_2, \ldots, e_m\}$
- Subgraph $H \subset G \longleftrightarrow$ Characteristic vectors in $\{0,1\}^m$: $h_i = 1_H(e_i)$.
- Cycle $c \in \{0,1\}^m$: Subgraph with all vertices incident with an even number of edges.



$$c = (1,1,1,1,0,1,1,0,0,0)$$

Cycle codes of graphs

- $G = (V, E)$ undirected connected graph with no loops, no multiple edges.
- $E = \{e_1, e_2, \ldots, e_m\}$
- Subgraph $H \subset G \longleftrightarrow$ Characteristic vectors in $\{0,1\}^m$: $h_i = 1_H(e_i)$.
- Cycle $c \in \{0,1\}^m$: Subgraph with all vertices incident with an even number of edges.
- Cycle space of G: the vector space over $\mathbb{F}_2$ of all cycles. It has dimension $m - n + 1$ (cyclomotic number)



$$c = (1,1,1,1,0,1,1,0,0,0)$$

Cycle codes of graphs

Cycle code of G: the linear binary code $[n, k, d]$ defined by the cycle space of G with

(i) length m (number of edges)

(ii) dimension $k = m - n + 1$ (cyclotomic number)

(iii) minimum distance $d =$ girth of G (length of smallest cycle).

(iv) incidence matrix of $G \longleftrightarrow$ parity-check matrix of the code (low–density parity-check code)

Array codes

Array codes

Binary Code of length $N = ab$ $\longrightarrow$ Array $a \times b$

- (100010001) $\longrightarrow$ $\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Array codes

Binary Code of length $N = ab$ $\longrightarrow$ Array $a \times b$

- $(100010001) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Code on $GF(2^b)$ of length $N' = a$ $\longrightarrow$ Code on $GF(2)$

- $(1, x, x^2) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Array codes

Binary Code of length $N = ab$ $\longrightarrow$ Array $a \times b$

- $(100010001) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Code on $GF(2^b)$ of length $N' = a$ $\longrightarrow$ Code on $GF(2)$

- $(1, x, x^2) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

- Distance between codewords $\longrightarrow$ Distance between columns: correction of column errors or column erasures

Array codes

Binary Code of length $N = ab$ $\longrightarrow$ Array $a \times b$

- $(100010001) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

Code on $GF(2^b)$ of length $N' = a$ $\longrightarrow$ Code on $GF(2)$

- $(1, x, x^2) \longrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

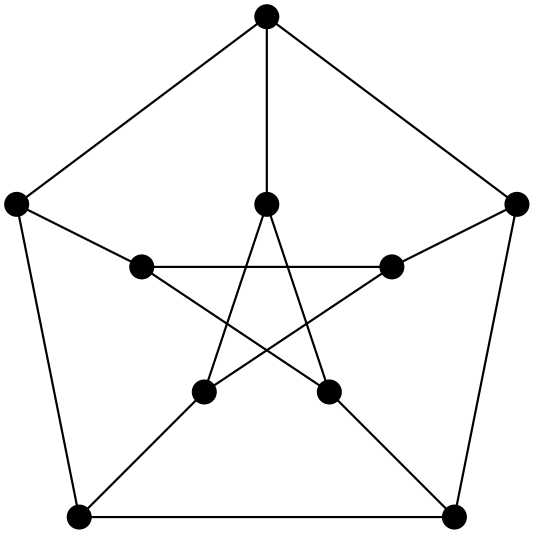
- Distance between codewords $\longrightarrow$ Distance between columns: correction of column errors or column erasures
- Array codes are used to address bursts of errors (as opposite to random errors).
  The are implemented in the standards of CD technology (by using Reed–Solomon codes).

L.B. Vries and K. Odaka, CIRC–The error correcting code for the Compact Disk digital audio system, Digital Audio, B.A. Blesser et al. editors, Audio Engrg. Soc. (1083), 178–186

Array cycle codes

Array cycle codes

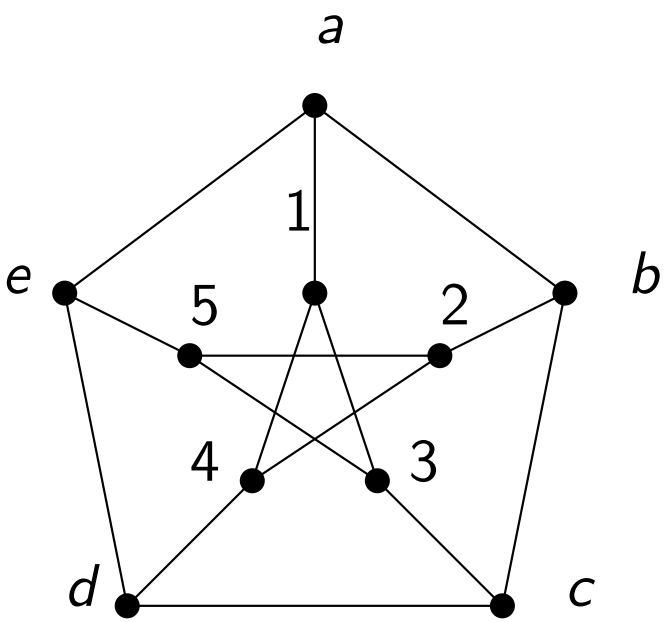
- Graph $G = (V, E)$ with $m = ab$ edges.



Array cycle codes

- Graph $G = (V, E)$ with $m = ab$ edges.
- Partition the edges in columns

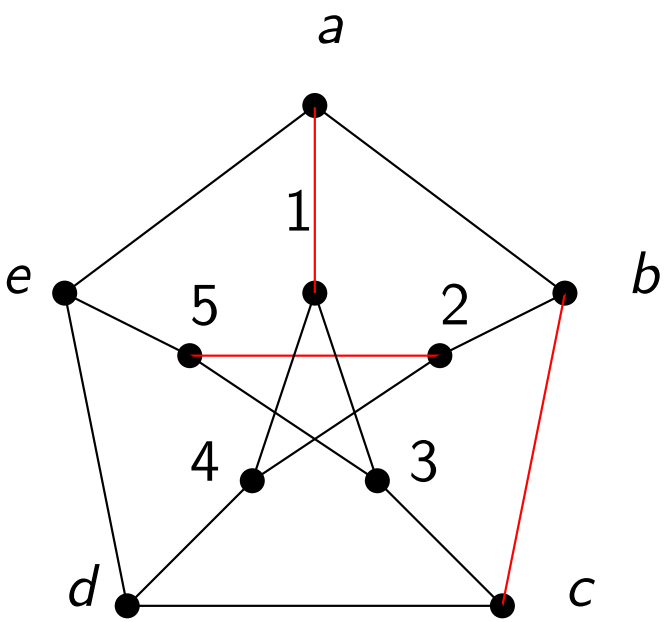
$$\begin{pmatrix} a1 & b2 & c3 & d4 & e5 \\ bc & cd & de & ea & ab \\ 52 & 13 & 24 & 35 & 41 \end{pmatrix}$$



Array cycle codes

- Graph $G = (V, E)$ with $m = ab$ edges.
- Partition the edges in columns $\longrightarrow$ edge-coloring of the graph.

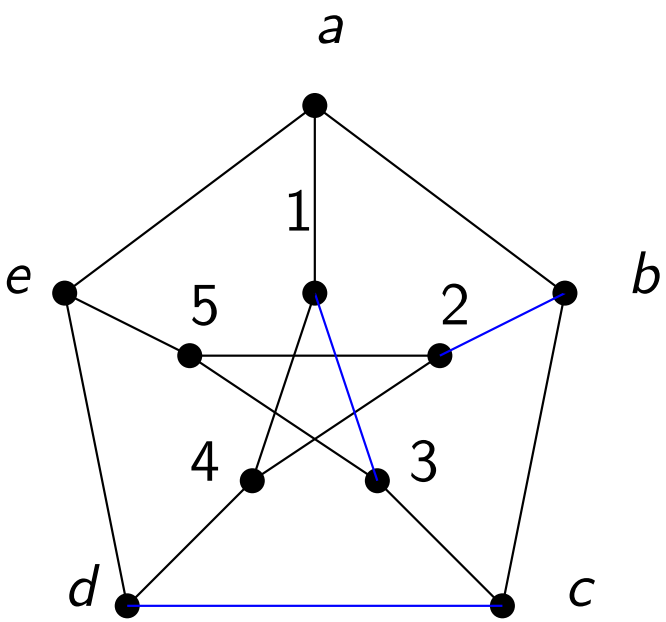
$$\begin{pmatrix} a1 & b2 & c3 & d4 & e5 \\ bc & cd & de & ea & ab \\ 52 & 13 & 24 & 35 & 41 \end{pmatrix}$$



Array cycle codes

- Graph $G = (V, E)$ with $m = ab$ edges.
- Partition the edges in columns $\longrightarrow$ edge-coloring of the graph.

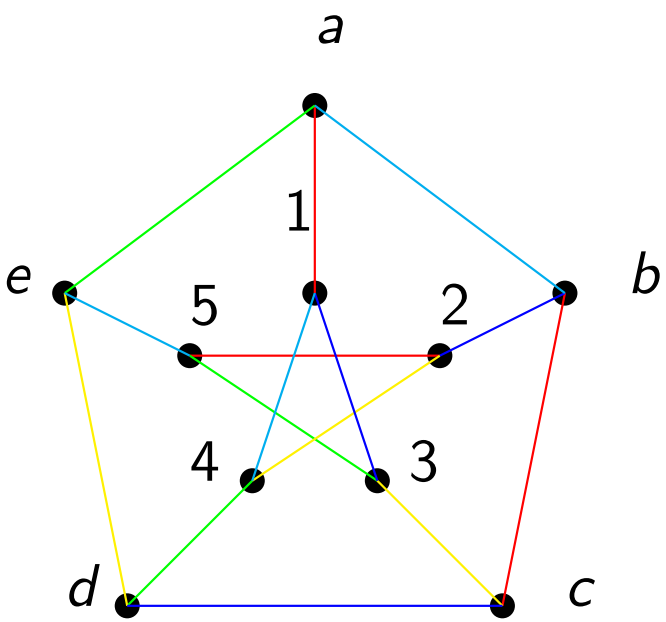
$$\begin{pmatrix} a1 & b2 & c3 & d4 & e5 \\ bc & cd & de & ea & ab \\ 52 & 13 & 24 & 35 & 41 \end{pmatrix}$$



Array cycle codes

- Graph $G = (V, E)$ with $m = ab$ edges.
- Partition the edges in columns $\longrightarrow$ edge-coloring of the graph.

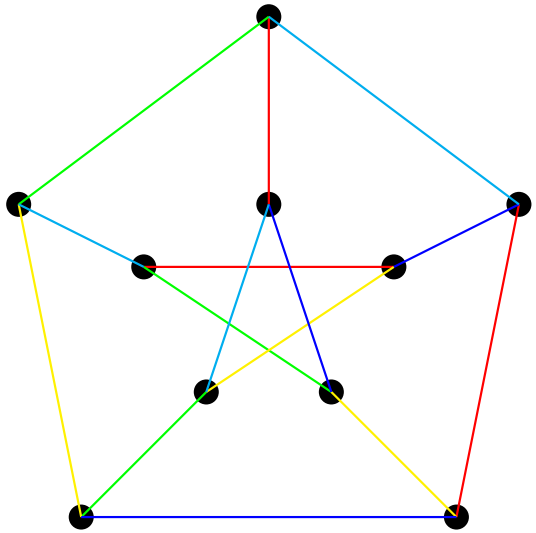
$$\begin{pmatrix} a1 & b2 & c3 & d4 & e5 \\ bc & cd & de & ea & ab \\ 52 & 13 & 24 & 35 & 41 \end{pmatrix}$$



Array cycle codes

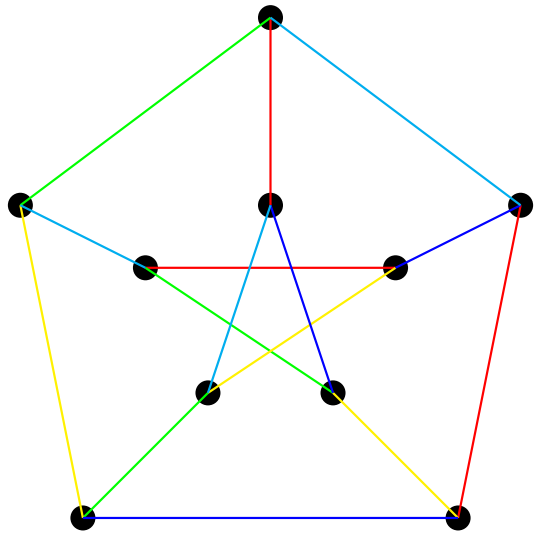
- Graph $G = (V, E)$ with $m = ab$ edges.
- Partition the edges in columns $\longrightarrow$ edge-coloring of the graph.

- Array Cycle code: The graph cycle code turned into an array code.



Array cycle codes: Minimum distance

- The minimum distance of the array cycle code is the minimum number of colors in a cycle.



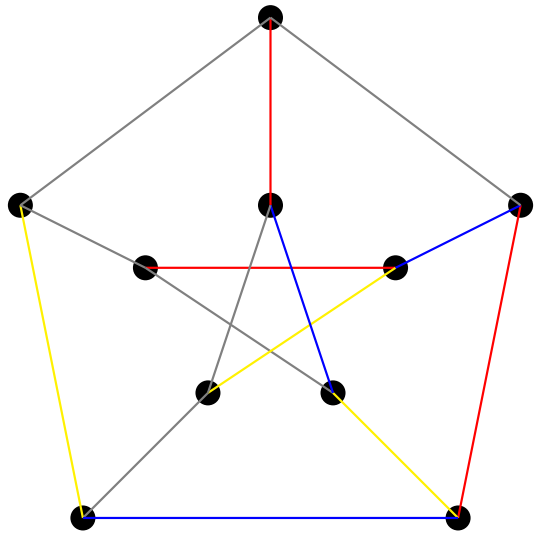
Array cycle codes: Minimum distance

- The minimum distance of the array cycle code is the minimum number of colors in a cycle.
- The Singleton bound for an array code $a \times b$ reads

$$D \leq b - \log_{2^a} |C| + 1.$$

In the example, the code has $|C| = 2^{15-10+1}$ and $D = 4 = 5 - \log_8 |C| + 1$.
It is an MDS (Maximum Distance Separating) code.

This means that every three colors span a spanning tree.



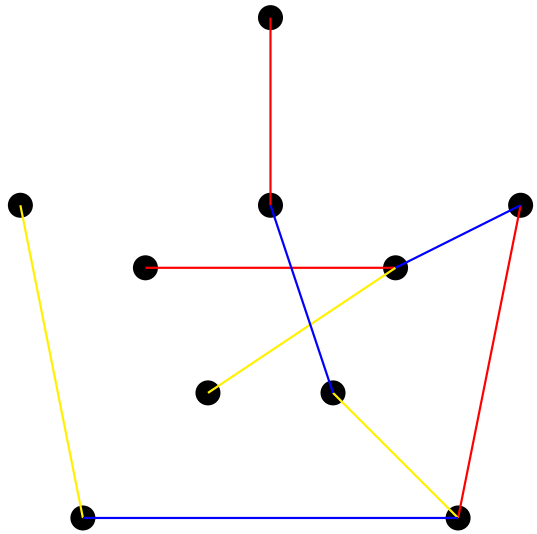
Array cycle codes: Minimum distance

- The minimum distance of the array cycle code is the minimum number of colors in a cycle.
- The Singleton bound for an array code $a \times b$ reads

$$D \leq b - \log_{2^a} |C| + 1.$$

In the example, the code has $|C| = 2^{15-10+1}$ and $D = 4 = 5 - \log_8 |C| + 1$.
It is an MDS (Maximum Distance Separating) code.

This means that every three colors span a spanning tree.

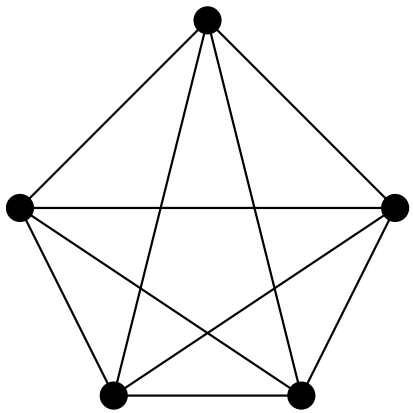


B–codes

- MDS Array cycle codes with $D = 3$.
  Edge colored graph such that every two colors make a spanning tree

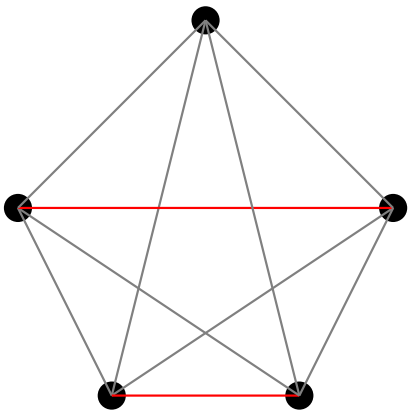
B–codes

- MDS Array cycle codes with $D = 3$.
  Edge colored graph such that every two colors make a spanning tree
- Largest length $\longleftrightarrow$ maximum number of edges.
  Complete graphs



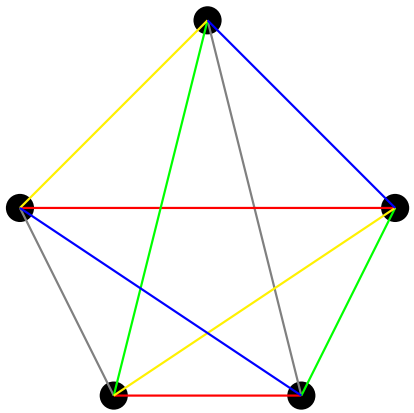
B–codes

- MDS Array cycle codes with $D = 3$.
  Edge colored graph such that every two colors make a spanning tree
- Largest length $\longleftrightarrow$ maximum number of edges.
  Complete graphs
- Every two colors make an acyclic graph $\longleftrightarrow$ every color is a matching (K_n has triangles).



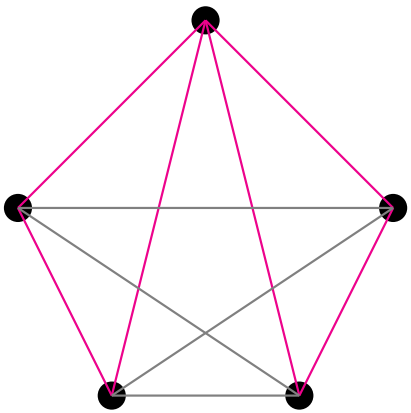
B–codes

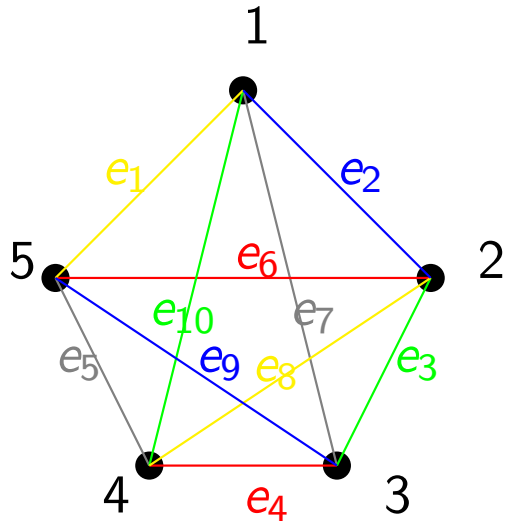
- MDS Array cycle codes with $D = 3$.
  Edge colored graph such that every two colors make a spanning tree
- Largest length $\longleftrightarrow$ maximum number of edges.
  Complete graphs
- Every two colors make an acyclic graph $\longleftrightarrow$ every color is a matching (K_n has triangles).
- They provide MDS array cycle codes with $a = (n-1)/2$, $b = n$, $\mathbb{F}_2$–dimension $n - 2a$ and $D = 3$.
  Known in the literature as B–codes.



Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$



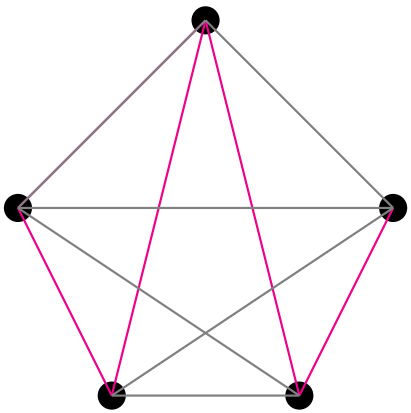


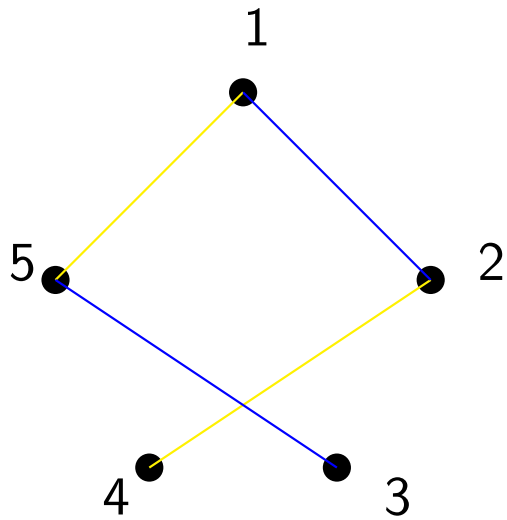
Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

- Two columns have been erased.

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

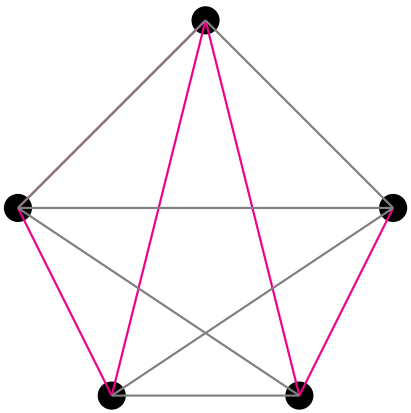


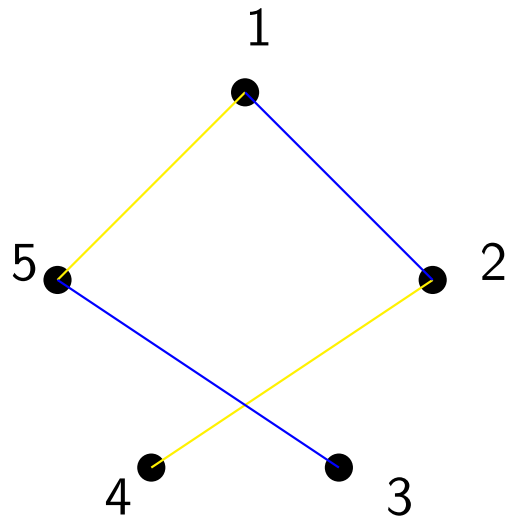


Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

- Two columns have been erased.
  $\begin{pmatrix} * & * & 1 & 0 & 1 \\ * & 0 & 1 & 0 & 1 \end{pmatrix}$
- Parity check of endvertex 3: edge e_9 is not in the word.

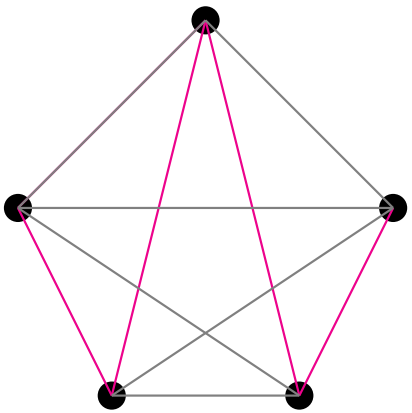


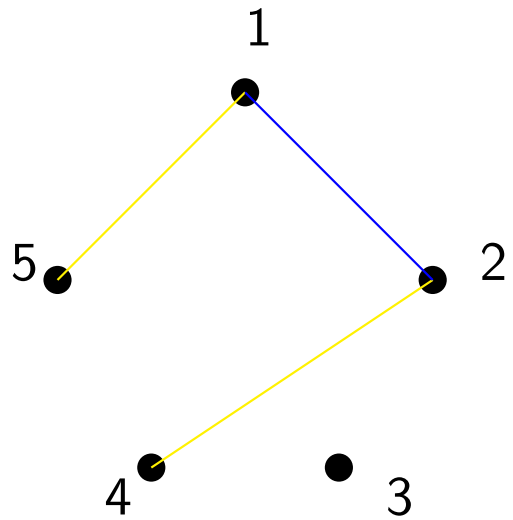


Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

- Two columns have been erased.
  $\begin{pmatrix} * & * & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$
- Parity check of endvertex 3: edge e_9 is not in the word.
- Parity check of endvertex 4: edge e_8 is not in the word.

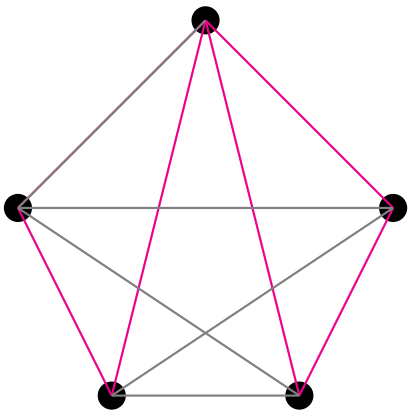


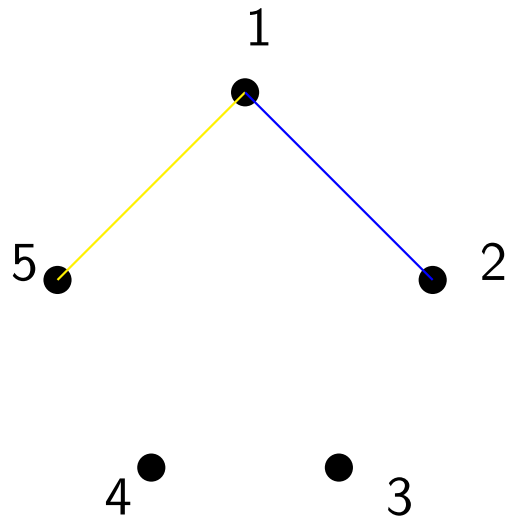


Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

- Two columns have been erased.
  $\begin{pmatrix} * & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$
- Parity check of endvertex 3: edge e_9 is not in the word.
- Parity check of endvertex 4: edge e_8 is not in the word.
- Parity check of endvertex 2: edge e_2 is in the word.

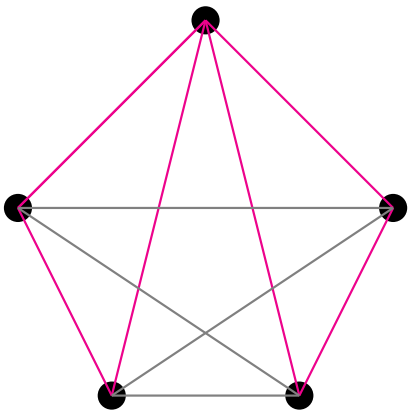


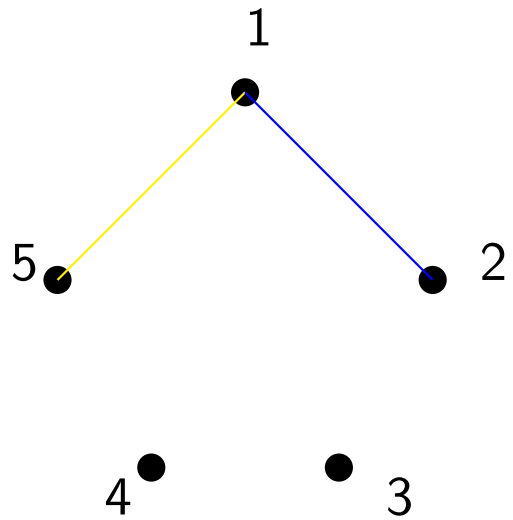


Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

- Two columns have been erased.
  $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$
- Parity check of endvertex 3: edge e_9 is not in the word.
- Parity check of endvertex 4: edge e_8 is not in the word.
- Parity check of endvertex 2: edge e_2 is in the word.
- Parity check of endvertex 5: edge e_1 is in the word.

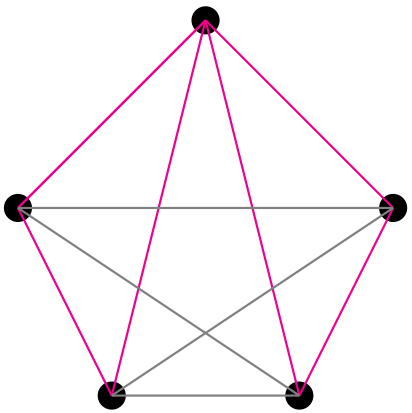


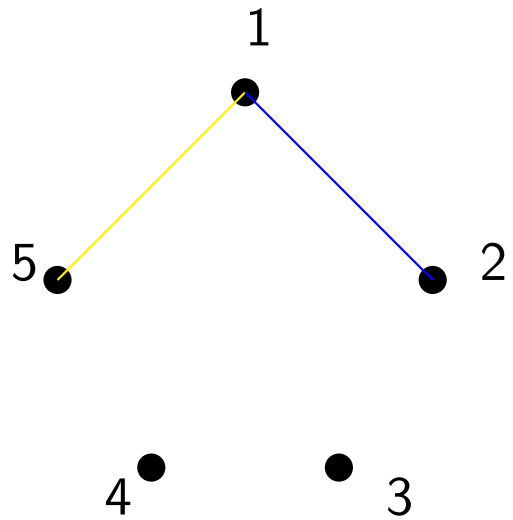


Correction algorithms: Column erasure correction

The sent codeword is $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$

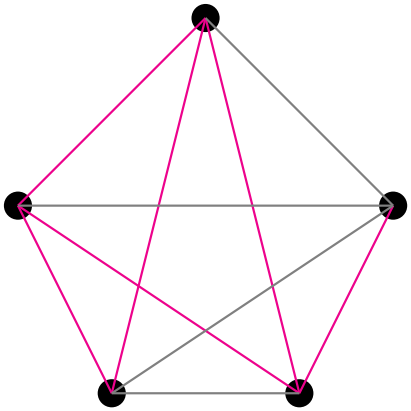
- Two columns have been erased.
  $\begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$
- Parity check of endvertex 3: edge e_9 is not in the word.
- Parity check of endvertex 4: edge e_8 is not in the word.
- Parity check of endvertex 2: edge e_2 is in the word.
- Parity check of endvertex 5: edge e_1 is in the word.
- The algorithm is linear in n.

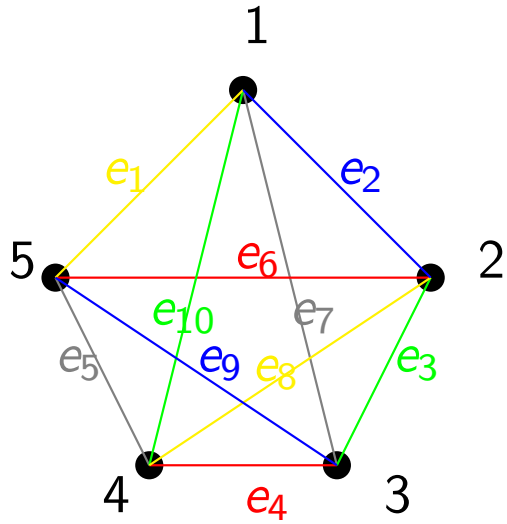




Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

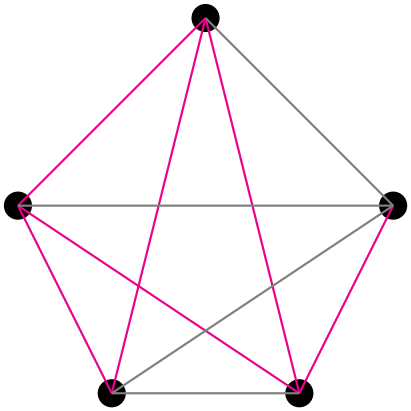


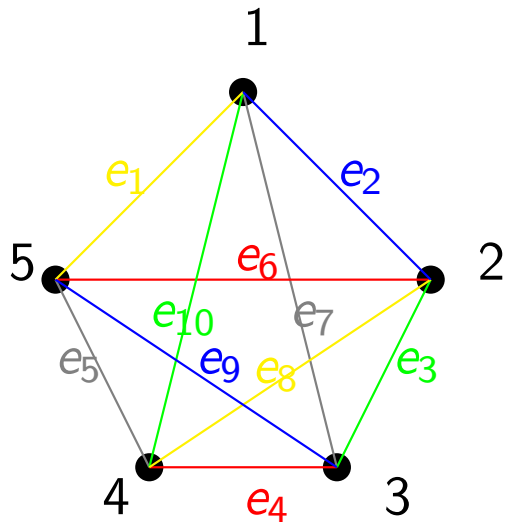


Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

- All errors are located in a single column.

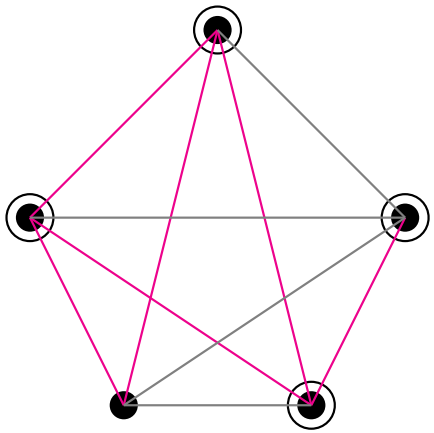


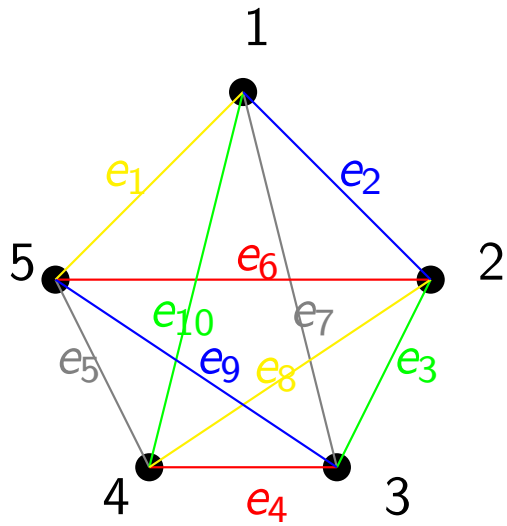


Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

- All errors are located in a single column.
- Find the vertices with unsatisfied parity check vertices.

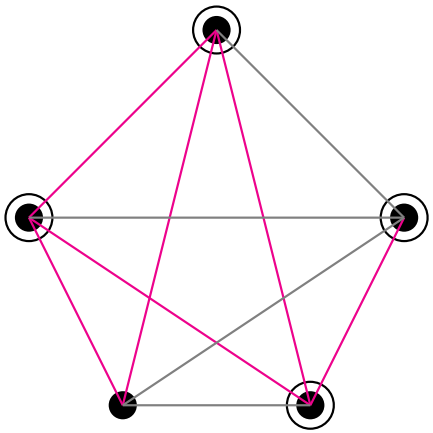


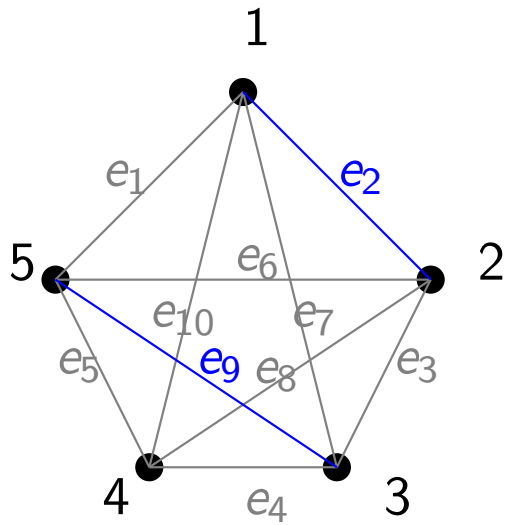


Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

- All errors are located in a single column.
- Find the vertices with unsatisfied parity check vertices.
- Test the color which covers the selected vertices.

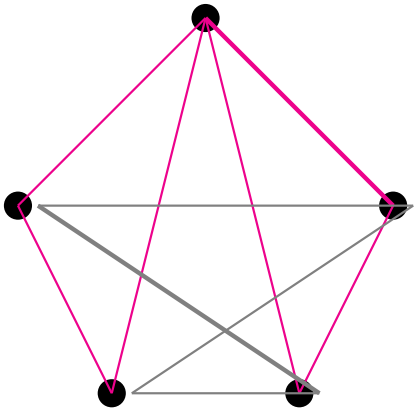


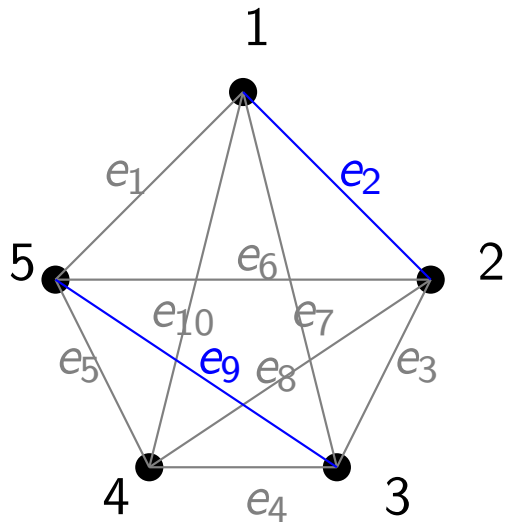


Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

- All errors are located in a single column.
- Find the vertices with unsatisfied parity check vertices.
- Test the color which covers the selected vertices.
- Exchange the bit of the edges covering these selected vertices.

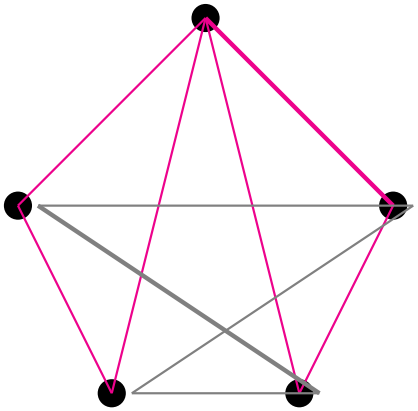


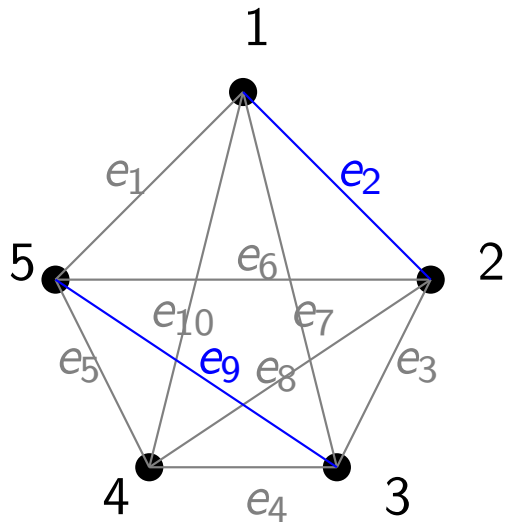


Correction algorithms: Errors in one column

The received codeword is $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$

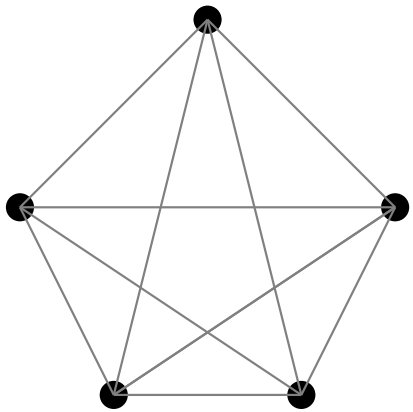
- All errors are located in a single column.
- Find the vertices with unsatisfied parity check vertices.
- Test the color which covers the selected vertices.
- Exchange the bit of the edges covering these selected vertices.
- The algorithm is linear in n.





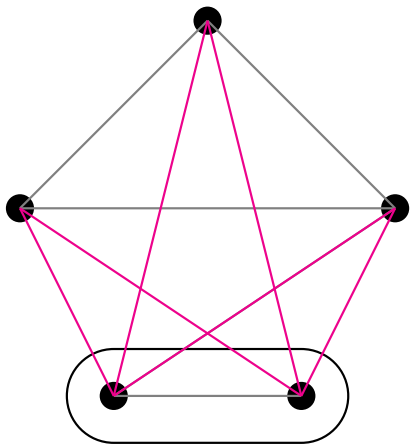
The dual B–codes

- The dual of a MDS code is again MDS.



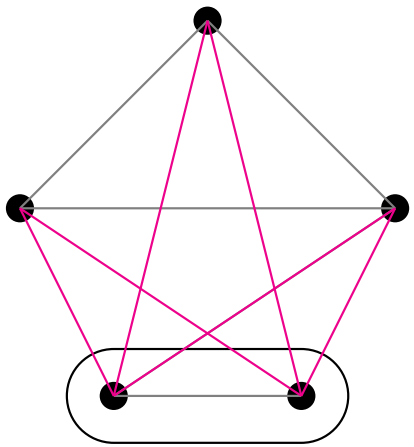
The dual B–codes

- The dual of a MDS code is again MDS.
- The dual of a B-code is the array cocycle code of K_n. Codewords are sets of edges joining a set with its complement.



The dual B–codes

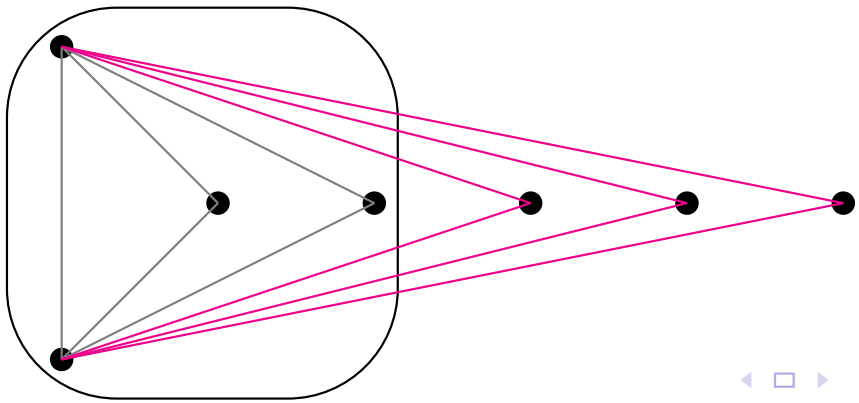
- The dual of a MDS code is again MDS.
- The dual of a B-code is the array cocycle code of K_n. Codewords are sets of edges joining a set with its complement.
- The minimum distance is $D = n - 1$ (the number of matchings).



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

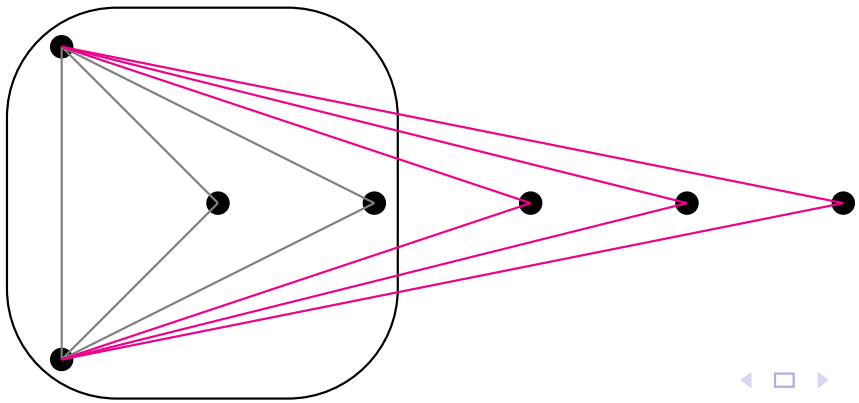
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

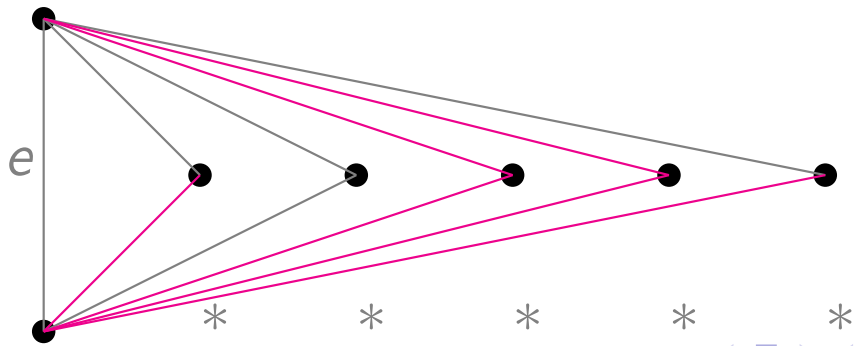
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.
- The edges incident to a vertex determine the cocycle.



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

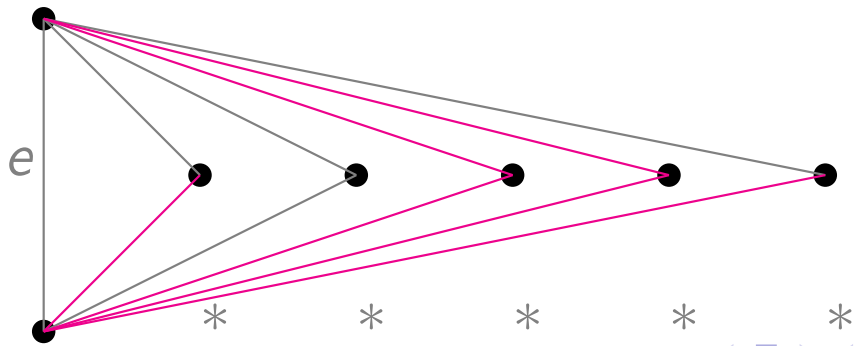
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.
- The edges incident to a vertex determine the cocycle.
- Choose one edge e.



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

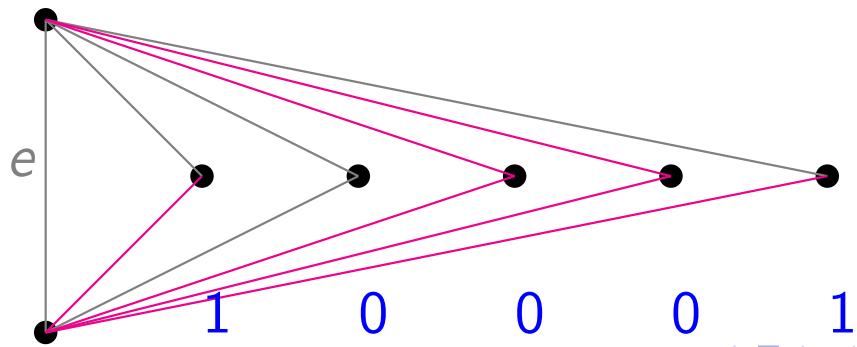
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.
- The edges incident to a vertex determine the cocycle.
- Choose one edge e.
- Consider the $(n-2)$ triangles supported by e.
  Errors affect at most $(n-2)/4$ matchings $\longrightarrow$ at most $(n-2)/2$ triangles.



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

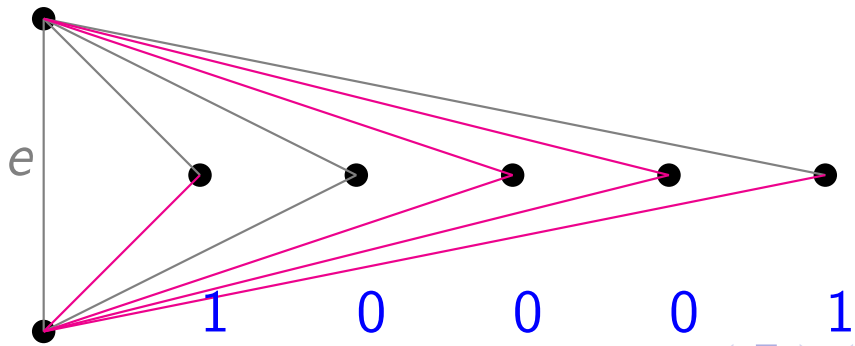
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.
- The edges incident to a vertex determine the cocycle.
- Choose one edge e.
- Consider the $(n-2)$ triangles supported by e.
  Errors affect at most $(n-2)/4$ matchings $\longrightarrow$ at most $(n-2)/2$ triangles.
- For each triangle flip the value of e if the parity is not satisfied. Use majority decision.



The dual B–codes: Correction algorithm

Correction of patterns of less than $(n-2)/4$ column errors.

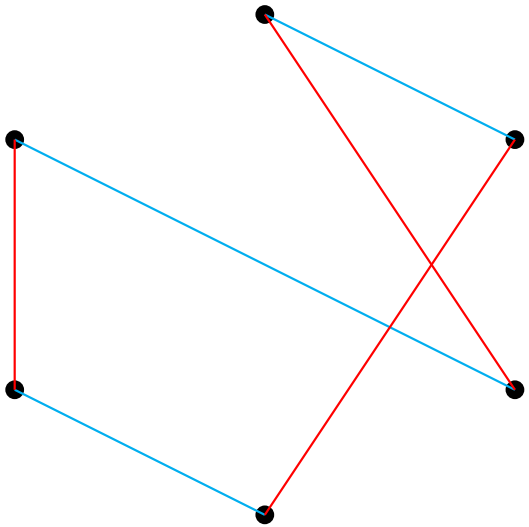
- A cocycle intersects a triangle in 0 or 2 edges: Each triangle provides a parity check condition.
- The edges incident to a vertex determine the cocycle.
- Choose one edge e.
- Consider the $(n-2)$ triangles supported by e.
  Errors affect at most $(n-2)/4$ matchings $\longrightarrow$ at most $(n-2)/2$ triangles.
- For each triangle flip the value of e if the parity is not satisfied. Use majority decision.
- Repeat for each edge incident to one endpoint of e.
- The algorithm is linear in $N = n(n-1)/2$ (the length of the code).

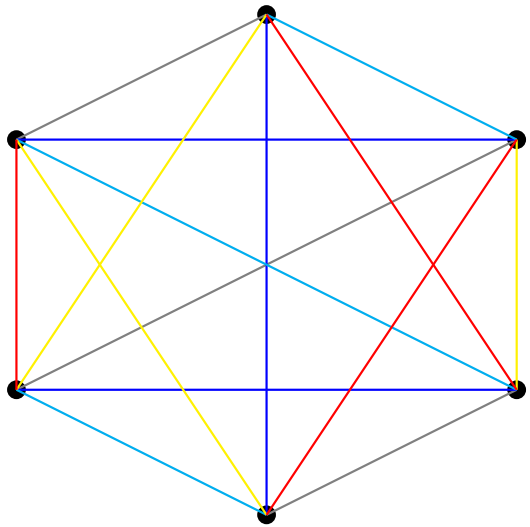


B–codes and the Perfect One-Factorization Conjecture

Conjecture (Kotzig, 1963)

For every n there is a proper coloring of the edges of K_{2n} with $2n-1$ colors such that every two colors induce a Hamiltonian cycle. Equivalently, there is a coloring of K_{2n-1} with $2n-1$ colors such that every two colors induce a Hamiltonian path.

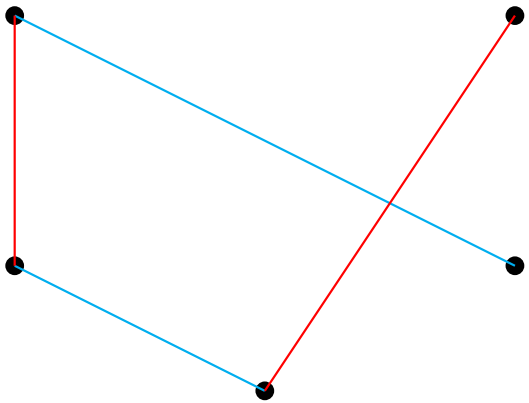


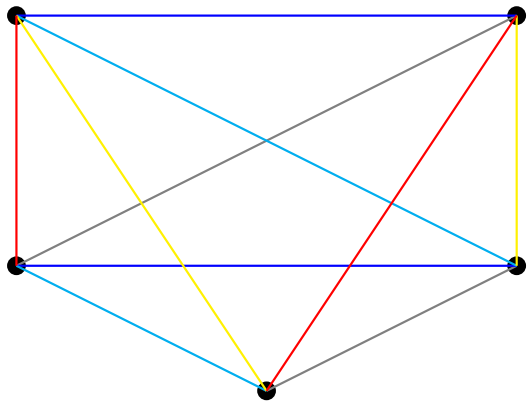


B–codes and the Perfect One-Factorization Conjecture

Conjecture (Kotzig, 1963)

For every n there is a proper coloring of the edges of K_{2n} with $2n-1$ colors such that every two colors induce a Hamiltonian cycle. Equivalently, there is a coloring of K_{2n-1} with $2n-1$ colors such that every two colors induce a Hamiltonian path.



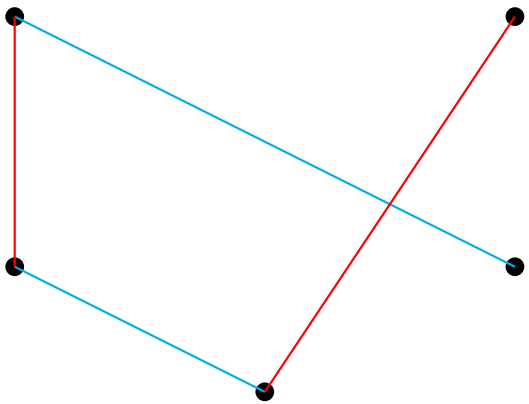


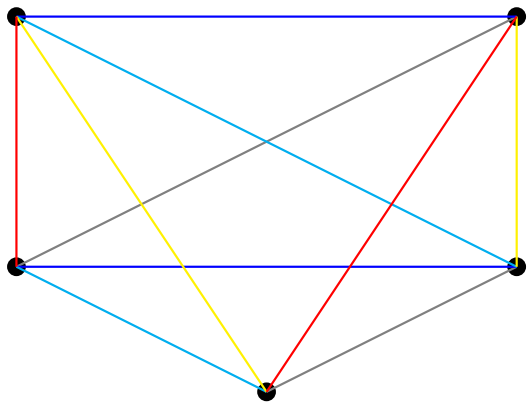
B–codes and the Perfect One-Factorization Conjecture

Conjecture (Kotzig, 1963)

For every n there is a proper coloring of the edges of K_{2n} with $2n-1$ colors such that every two colors induce a Hamiltonian cycle. Equivalently, there is a coloring of K_{2n-1} with $2n-1$ colors such that every two colors induce a Hamiltonian path.

- Easy for $2n = p+1$, and $n = p$, p prime.



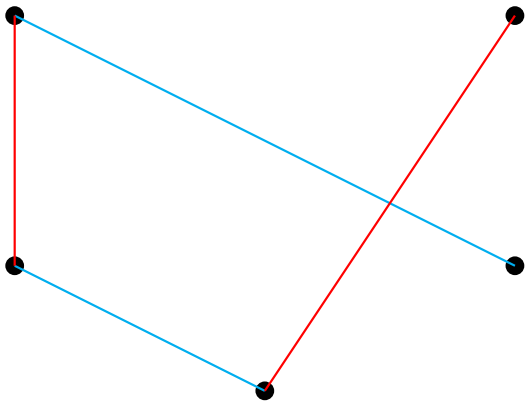


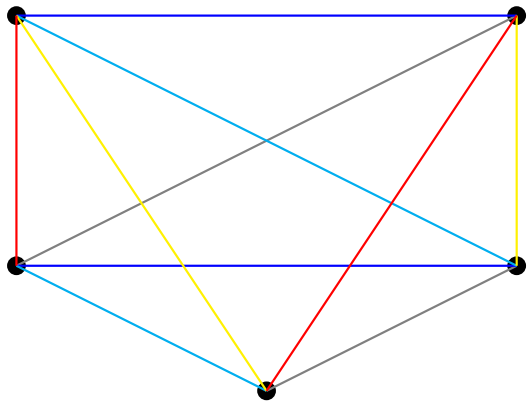
B–codes and the Perfect One-Factorization Conjecture

Conjecture (Kotzig, 1963)

For every n there is a proper coloring of the edges of K_{2n} with $2n-1$ colors such that every two colors induce a Hamiltonian cycle. Equivalently, there is a coloring of K_{2n-1} with $2n-1$ colors such that every two colors induce a Hamiltonian path.

- Easy for $2n = p+1$, and $n = p$, p prime.
- Achieved for all $n \leq 100$.



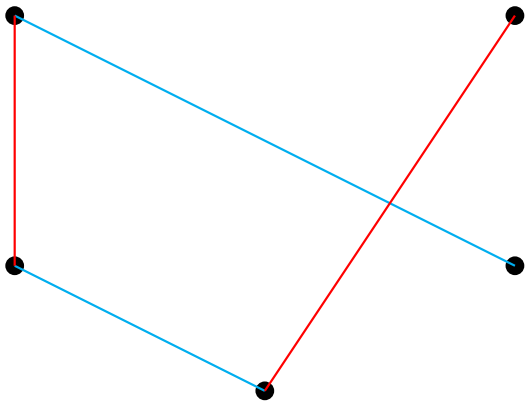


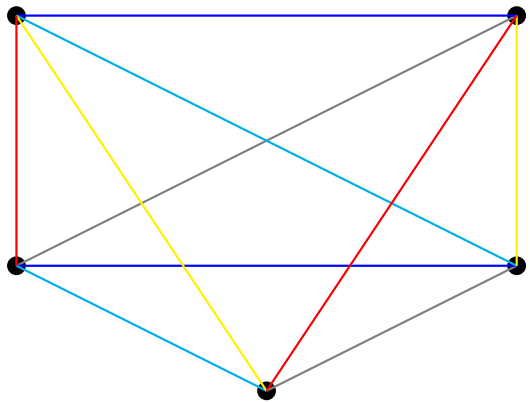
B–codes and the Perfect One-Factorization Conjecture

Conjecture (Kotzig, 1963)

For every n there is a proper coloring of the edges of K_{2n} with $2n-1$ colors such that every two colors induce a Hamiltonian cycle. Equivalently, there is a coloring of K_{2n-1} with $2n-1$ colors such that every two colors induce a Hamiltonian path.

- Easy for $2n = p+1$, and $n = p$, p prime.
- Achieved for all $n \leq 100$.





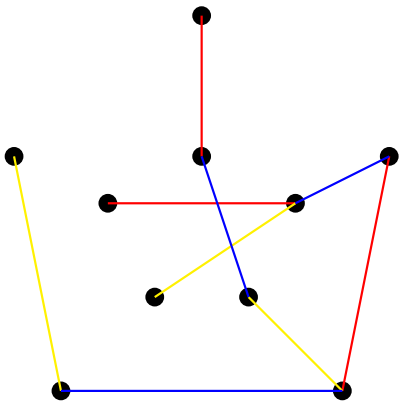
More examples of MDS array cycle codes

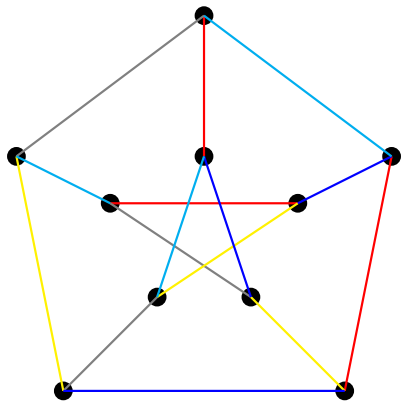
For a graph G the above correction algorithms work if G admits an edge–coloring such that every $D-1$ colors make a spanning tree.

More examples of MDS array cycle codes

For a graph G the above correction algorithms work if G admits an edge–coloring such that every $D-1$ colors make a spanning tree.

- The Petersen graph provides an MDS array cycle code with $D = 4$. Unfortunately $D = 4$ implies $n \leq 10$.

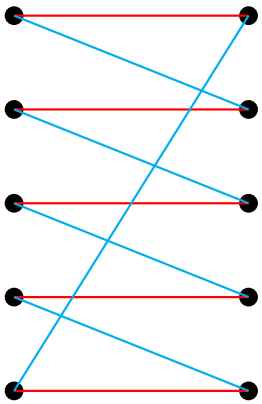


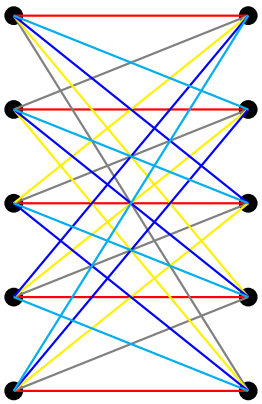


More examples of MDS array cycle codes

For a graph G the above correction algorithms work if G admits an edge–coloring such that every $D-1$ colors make a spanning tree.

- The Petersen graph provides an MDS array cycle code with $D = 4$. Unfortunately $D = 4$ implies $n \leq 10$.
- For $D = 3$ the complete bipartite graphs $K_{n-1,n}$ are conjectured to admit such an edge–coloring.
  Equivalently $K_{n,n}$ admits a coloring such that two colors span a Hamiltonian cycle.
  Known to be the case for $n = p$, $n = 2p-1$, $n = p^2$, and small values of n.

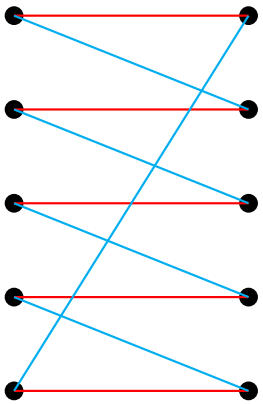


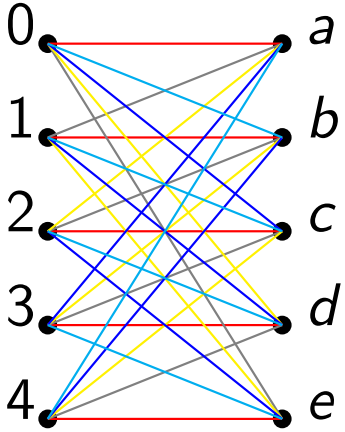


More examples of MDS array cycle codes

For a graph G the above correction algorithms work if G admits an edge–coloring such that every $D-1$ colors make a spanning tree.

- The Petersen graph provides an MDS array cycle code with $D = 4$. Unfortunately $D = 4$ implies $n \leq 10$.
- For $D = 3$ the complete bipartite graphs $K_{n-1,n}$ are conjectured to admit such an edge–coloring.
  Equivalently $K_{n,n}$ admits a coloring such that two colors span a Hamiltonian cycle.
  Known to be the case for $n = p$, $n = 2p-1$, $n = p^2$, and small values of n.



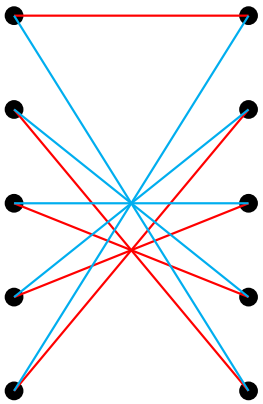


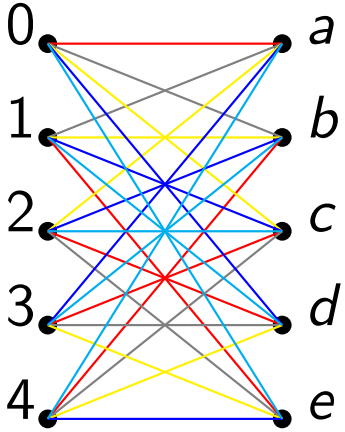
$$\begin{bmatrix} 0a & 2d & 4b & 1e & 3c \\ 3d & 0b & 2e & 4c & 1a \\ 1b & 3e & 0c & 2a & 4d \\ 4e & 1c & 3a & 0d & 2b \\ 2c & 4a & 1d & 3b & 0e \end{bmatrix}$$

More examples of MDS array cycle codes

For a graph G the above correction algorithms work if G admits an edge–coloring such that every $D-1$ colors make a spanning tree.

- The Petersen graph provides an MDS array cycle code with $D = 4$. Unfortunately $D = 4$ implies $n \leq 10$.
- For $D = 3$ the complete bipartite graphs $K_{n-1,n}$ are conjectured to admit such an edge–coloring.
  Equivalently $K_{n,n}$ admits a coloring such that two colors span a Hamiltonian cycle.
  Known to be the case for $n = p$, $n = 2p-1$, $n = p^2$, and small values of n.





$$\begin{bmatrix} 0a & 2d & 4b & 1e & 3c \\ 3d & 0b & 2e & 4c & 1a \\ 1b & 3e & 0c & 2a & 4d \\ 4e & 1c & 3a & 0d & 2b \\ 2c & 4a & 1d & 3b & 0e \end{bmatrix}$$

Array Double Cycle Codes

A graph G with n^2 edges has an array double coloring if the edges can be placed in a square array $n \times n$ such that each pair of rows and each pair of columns induce a spanning tree.

Array Double Cycle Codes

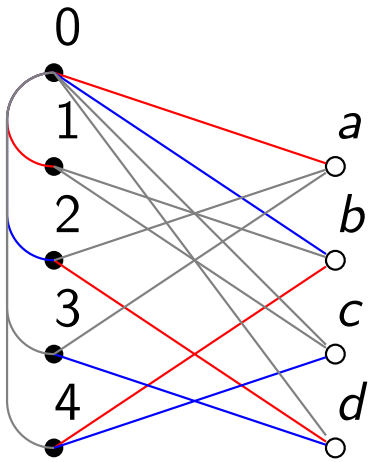
A graph G with n^2 edges has an array double coloring if the edges can be placed in a square array $n \times n$ such that each pair of rows and each pair of columns induce a spanning tree.

- A graph admitting an array double coloring provides an MDS array code with $D = 3$ which allows for column and row errors/erasures. The correction algorithms are as above.

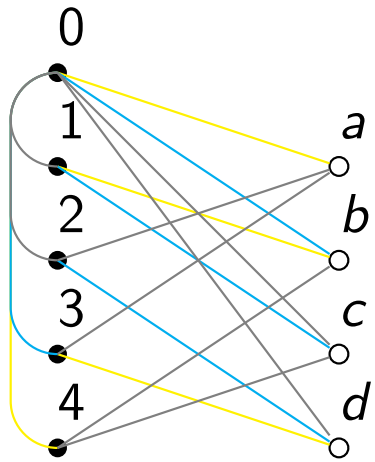
Array Double Cycle Codes

A graph G with n^2 edges has an array double coloring if the edges can be placed in a square array $n \times n$ such that each pair of rows and each pair of columns induce a spanning tree.

- A graph admitting an array double coloring provides an MDS array code with $D = 3$ which allows for column and row errors/erasures. The correction algorithms are as above.
- For every prime p there is a graph G with $(p-1)^2$ edges and $2p-1$ vertices which admits a double array coloring.



$$\begin{pmatrix} 0a & 3d & 1b & 04 \\ 2d & 0b & 03 & 1c \\ 4b & 02 & 0c & 3a \\ 01 & 4c & 2b & 0d \end{pmatrix}$$



An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

- Declare a column to be the secret and distribute the remaining columns to $n-1$ participants.

An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

- Declare a column to be the secret and distribute the remaining columns to $n-1$ participants.
- If $n-2$ participants meet, the two–columns erasure algorithm allows them to recover (efficiently) the secret.

An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

- Declare a column to be the secret and distribute the remaining columns to $n-1$ participants.
- If $n-2$ participants meet, the two–columns erasure algorithm allows them to recover (efficiently) the secret.
- If only $n-3$ participants meet, every arbitrary filling of the secret column allows the algorithm to recover a codeword.

An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

- Declare a column to be the secret and distribute the remaining columns to $n-1$ participants.
- If $n-2$ participants meet, the two–columns erasure algorithm allows them to recover (efficiently) the secret.
- If only $n-3$ participants meet, every arbitrary filling of the secret column allows the algorithm to recover a codeword.

An application to Secret Sharing Schemes

- Choose a codeword at random from an MDS Array Cycle Code on K_n.

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

- Declare a column to be the secret and distribute the remaining columns to $n-1$ participants.
- If $n-2$ participants meet, the two–columns erasure algorithm allows them to recover (efficiently) the secret.
- If only $n-3$ participants meet, every arbitrary filling of the secret column allows the algorithm to recover a codeword.
  Every set of less than $n-2$ participants has 0–information of the secret: all values are equiprobable

Summary

- Array codes are designed for column erasures/errors (runs of errors).
- Graphs provide MDS array codes with minimum distance $D = 3$ (and their duals) with simple and efficient algorithms for erasure/error corrections.
- Array cycle codes graphs are available in sizes $((n-1)/2, n)$ and $(n-1, n-1)$ for $n = p, 2p-1, \ldots$ (connected to P1F Conjecture).
- Array double cycle codes of size $(p-1, p-1)$ are available and allow for row/column erasure/error correction.
- The codes provide an application to cryptographic schemes.

Th*nk you