Lecture 9: Codes for distributed storage

March 14, 2019

Regenerating codes

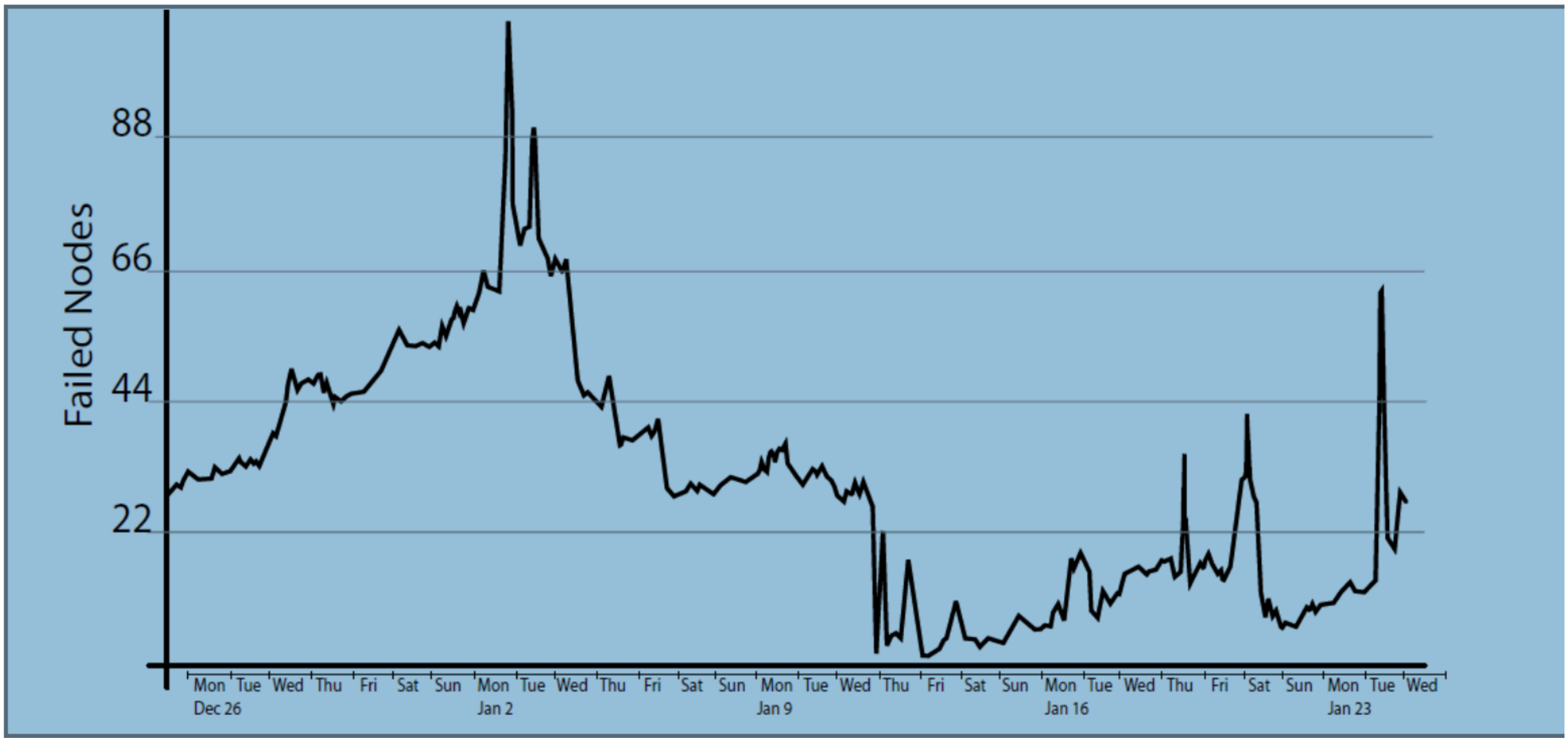
Distributed data storage, problem : node failures

- Example : google center

– 800000 servers, failure rate $= 4\%$ per year
– repair in 2 days
– mean number of failed servers in 2 days $= 175$

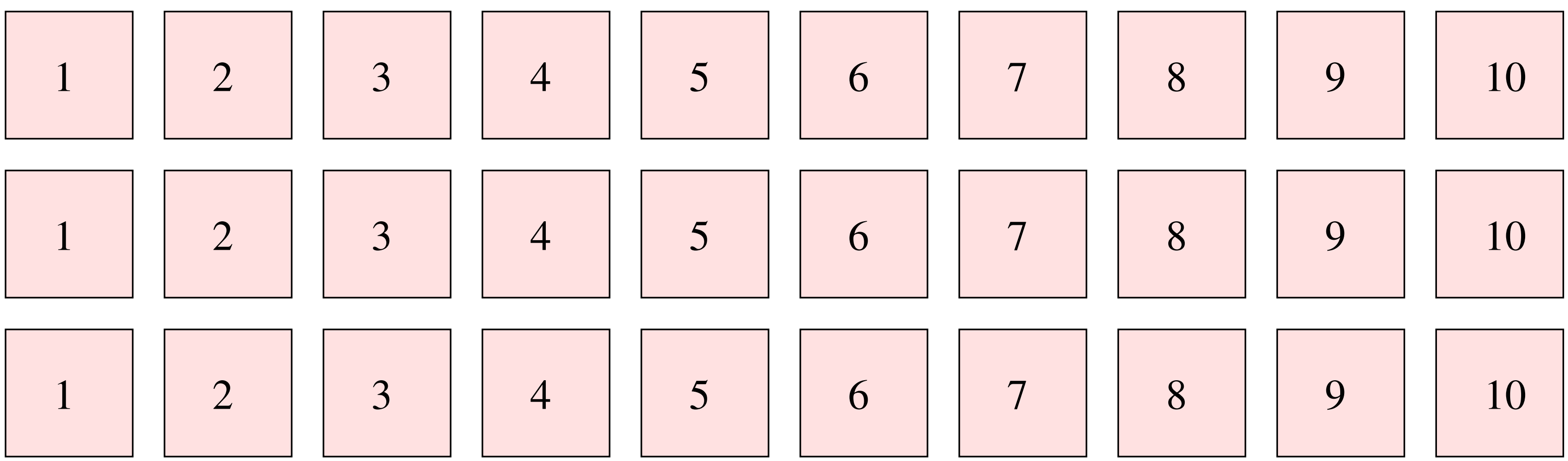
Another example

- # failed nodes over a single month in a 3000 node cluster of Facebook



Example : Hadoop software

- Aim : handling massive amounts of data and computation
- Hadoop Distributed File System : default $3\times$ replication for handling node failures
- 640 MB files : 10 blocks



- Highly inefficient !

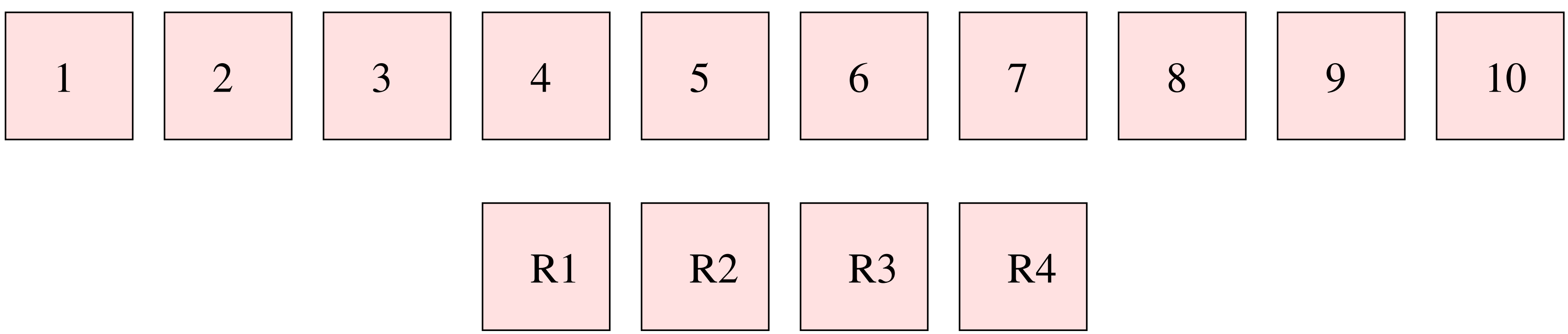
Facebook cluster

- Huge Hadoop cluster

- 2012 : 30 PB (3.10^{16} bytes !) of data and this is growing...
- Thousands of nodes
- Storage efficiency : main driver for cost

HDFS-RAID

- uses a $[n = 14, k = 10, d = 5]$-code to recompute blocks in the source file or redundancy file when they are lost or corrupted.
- Reduces storage overhead from $\times 3$ to $\times 1.4$
- Used for less frequently accessed data
- Can tolerate any loss of 4 blocks

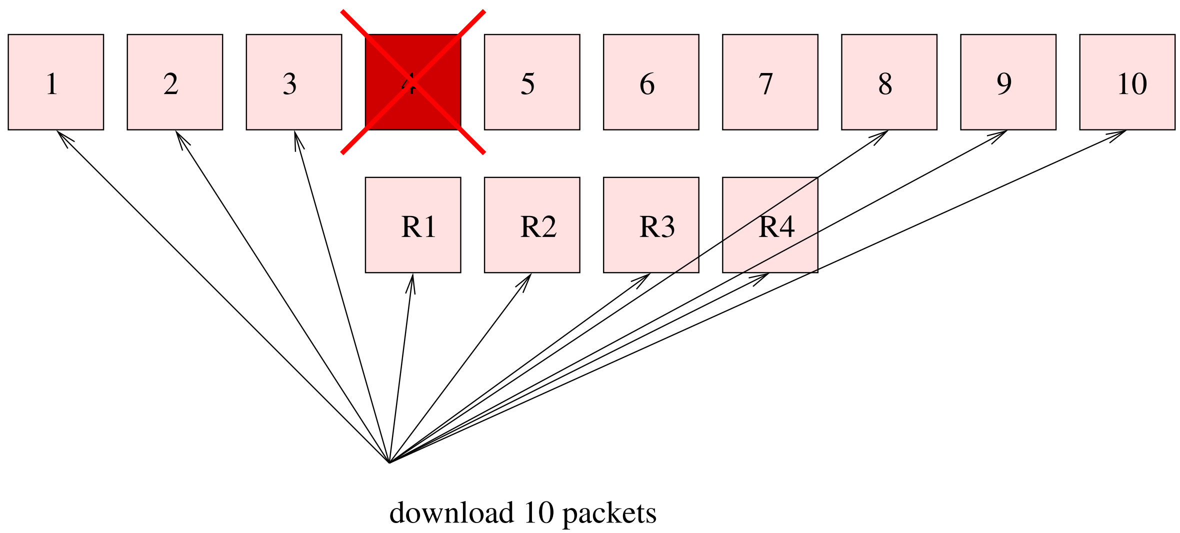


Exercise

1. Prove that the minimum distance d of a linear code of length n and dimension k is at most $n - k + 1$.
2. When $d = n - k + 1$ the code is called a MDS code. Prove that a parity check matrix of such a code is such that any square submatrix of size $n - k$ in a parity-check matrix of such a code is invertible.
3. Show that such a code can tolerate all patterns of $n - k$ erasures and give a method for recovering the whole codeword when there are $n - k$ erasures.

The problem : speed of access

- Good news : can tolerate 4 node failures by looking for 10 good nodes
- What if there is only one node failure ?
- Bad news : still needs 10 good nodes



Exercise

1. Show that any square submatrix of a generator matrix of an MDS code is of full rank
2. Explain why this implies that in order to recover a single erasure in an MDS code of dimension k it is necessary and sufficient to use k other code positions.

Drawback of MDS codes

- Do not tolerate better bandwidth consumption when only a few nodes are down and we want only to recover information from those nodes
- High network traffic in this case
- High disk read

$\Rightarrow$ need for a scalable solution/number of nodes we want to recover

Industry impact

piggyback codes & Hitchhiker (2013-2014)	f, THE APACHE SOFTWARE FOUNDATION, NetApp, hadoop, CISCO
butterfly codes (2013)	WD Western Digital
Ye-Barg codes & Clay codes (2017-18)	ceph

Distributed storage system

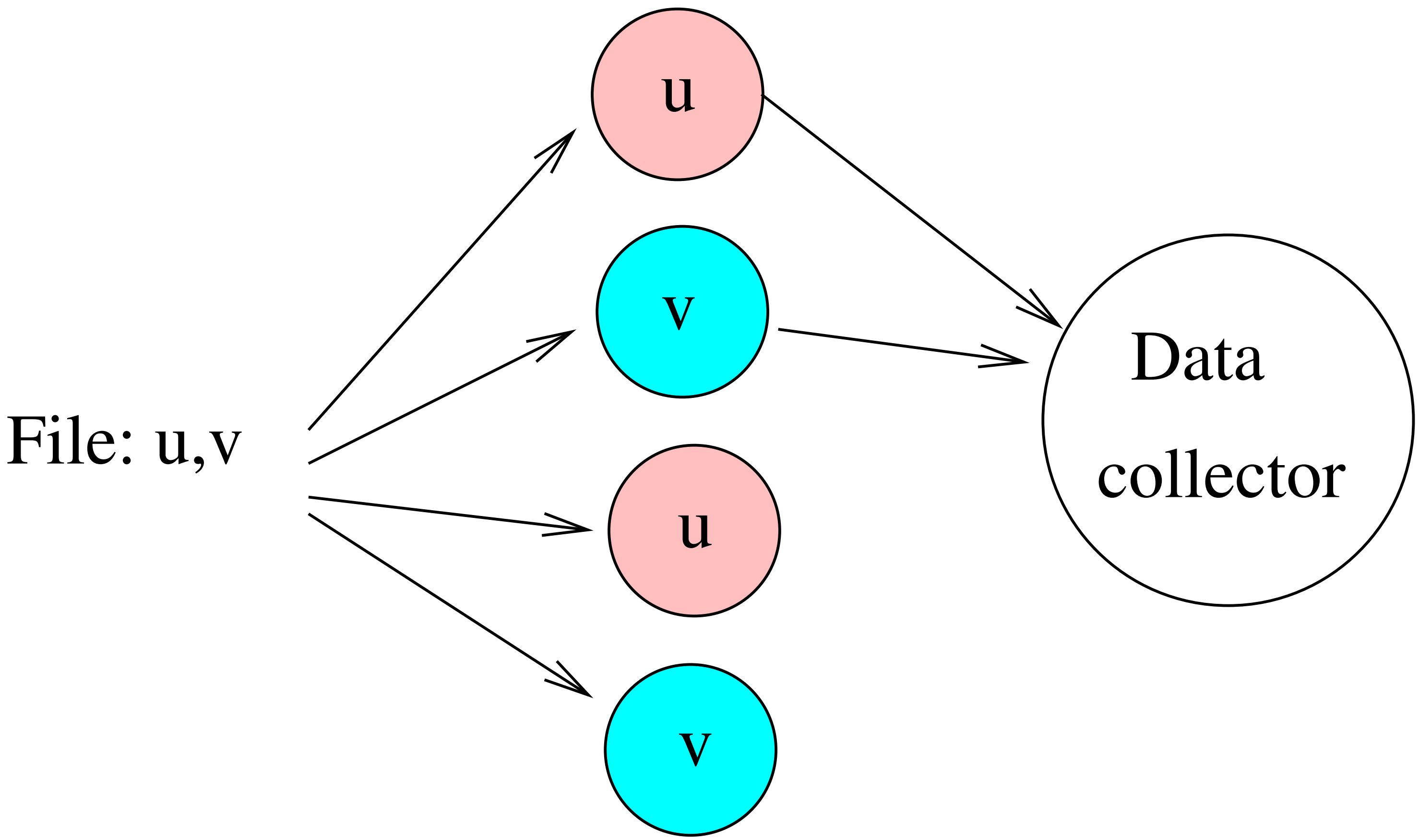
- Distributed storage system(DSS) with n storage nodes
- Block of data of B symbols over a finite alphabet $\mathcal{A}$
- information on some of these symbols is stored in each node of the DSS
- Each storage node is able to store α symbols
- This block of data can be retrieved by a data collector connecting to any k of these nodes
- One of the node goes down and has to be repaired by putting all its information in a new node by connecting to d $(k \leq d \leq n-1)$ nodes that are still working and downloading $\beta \leq \alpha$ bits from each of them, locality$= d$
- bandwidth$\overset{\text{def}}{=} d\beta$

$$\Rightarrow \boxed{\text{minimize bandwidth} \quad d\beta}$$

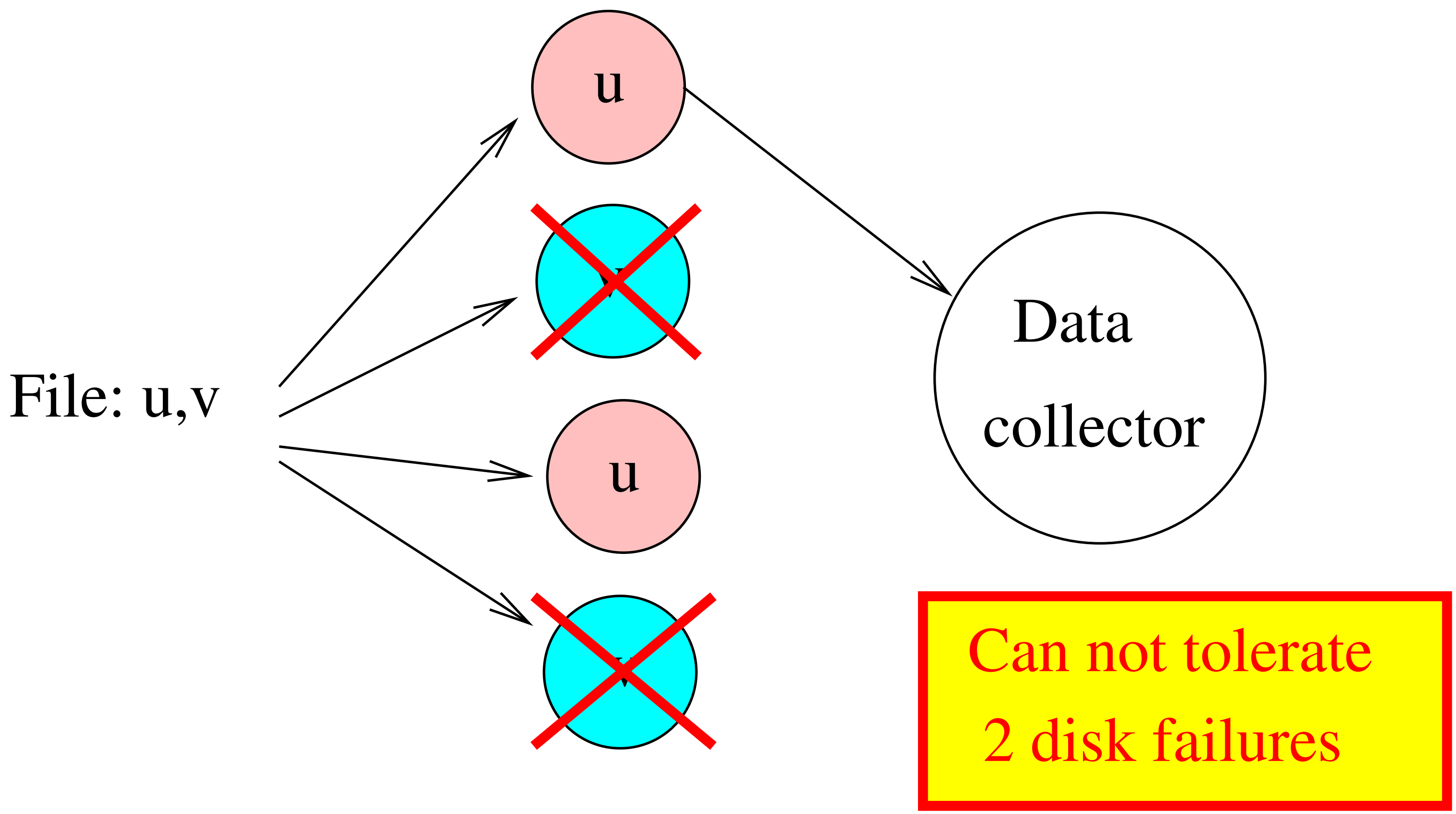
Three examples

	repetition code	Reed-Solomon code	Regenerating code
storage efficiency	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
reliability	tolerate 1 disk failure	tolerate any 2 disk failures	tolerate any 2 disk failures
repair bandwidth	1G	2G	1.5G
locality	1	2	3

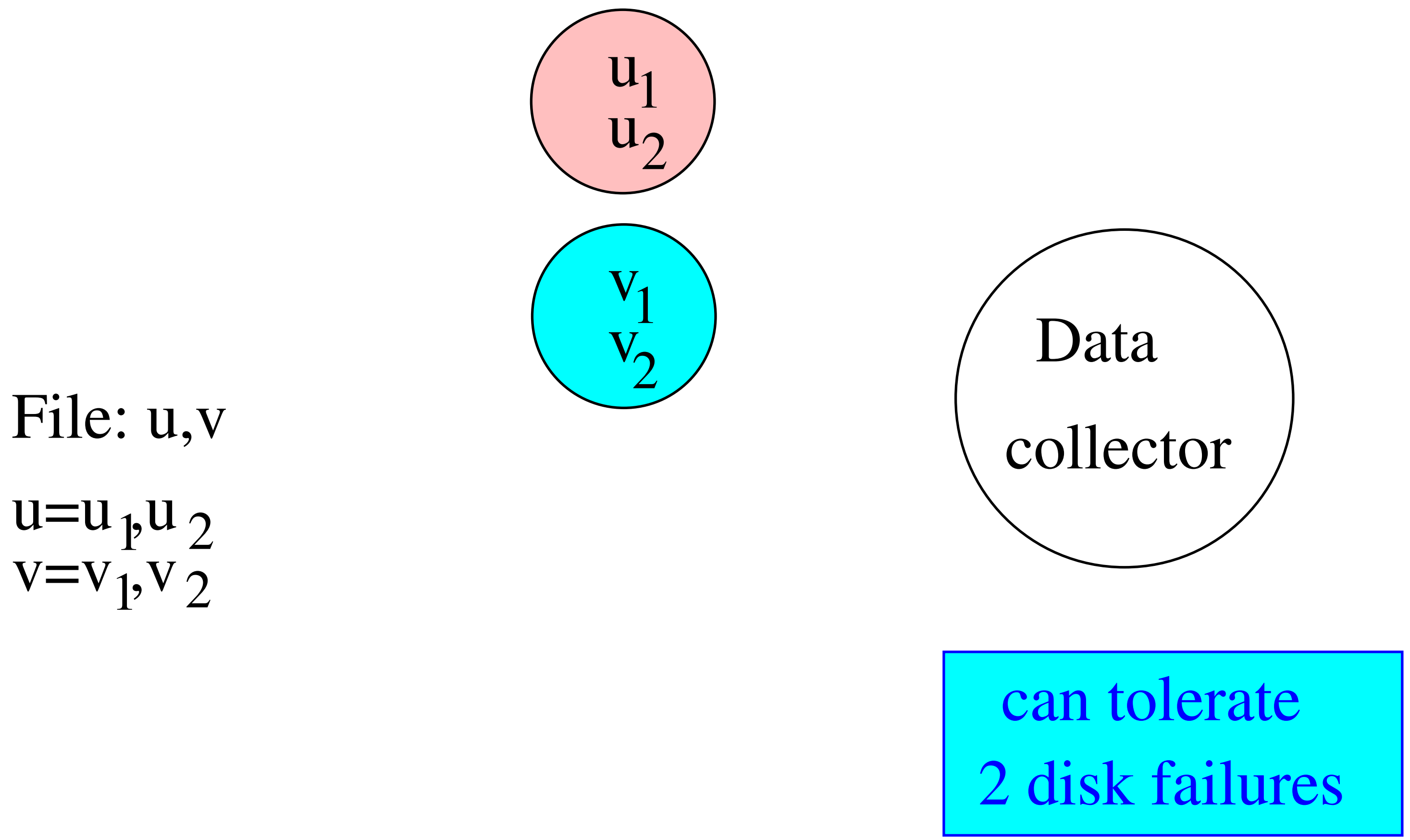
Example 1 : $2\times$ repetition scheme



Problem


File: u,v
u
v
u
v
Data
collector
Can not tolerate
2 disk failures

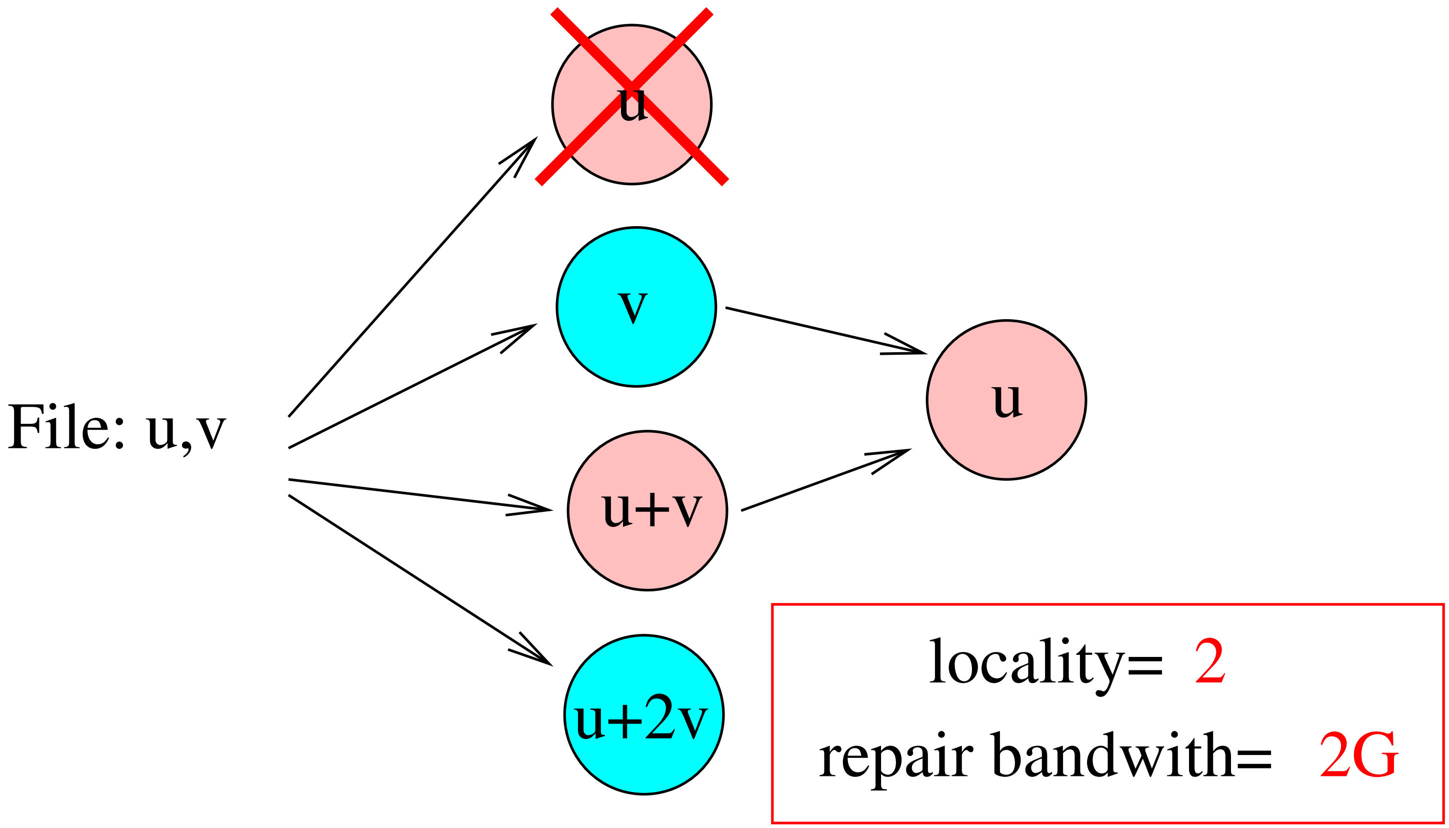
Example 2 : Reed-Solomon code



Exercise

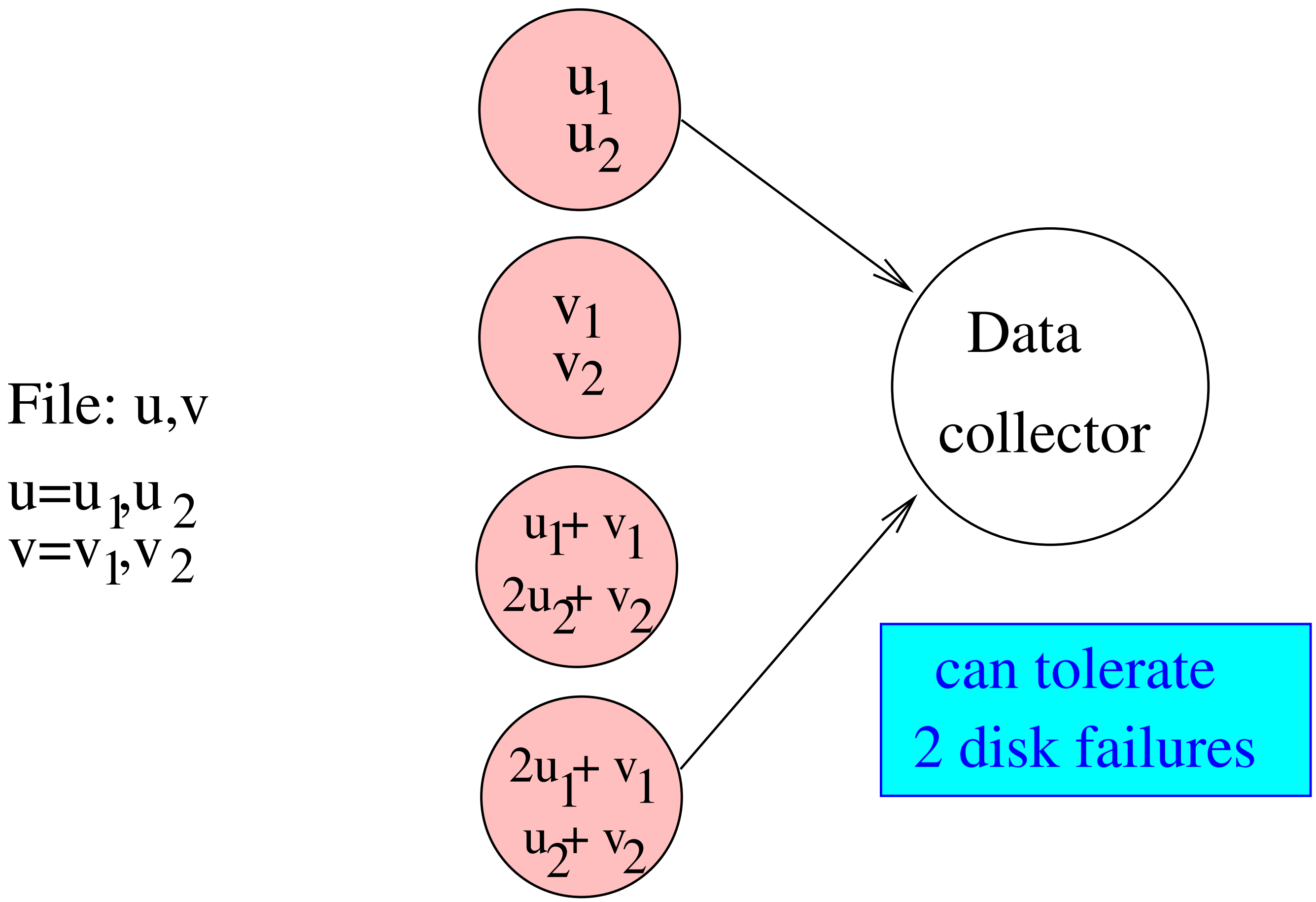
1. Show that this corresponds to a linear encoding scheme and give the generator matrix of this scheme
2. Show that the corresponding code is an MDS code

Problem



- repair : decoding the whole file

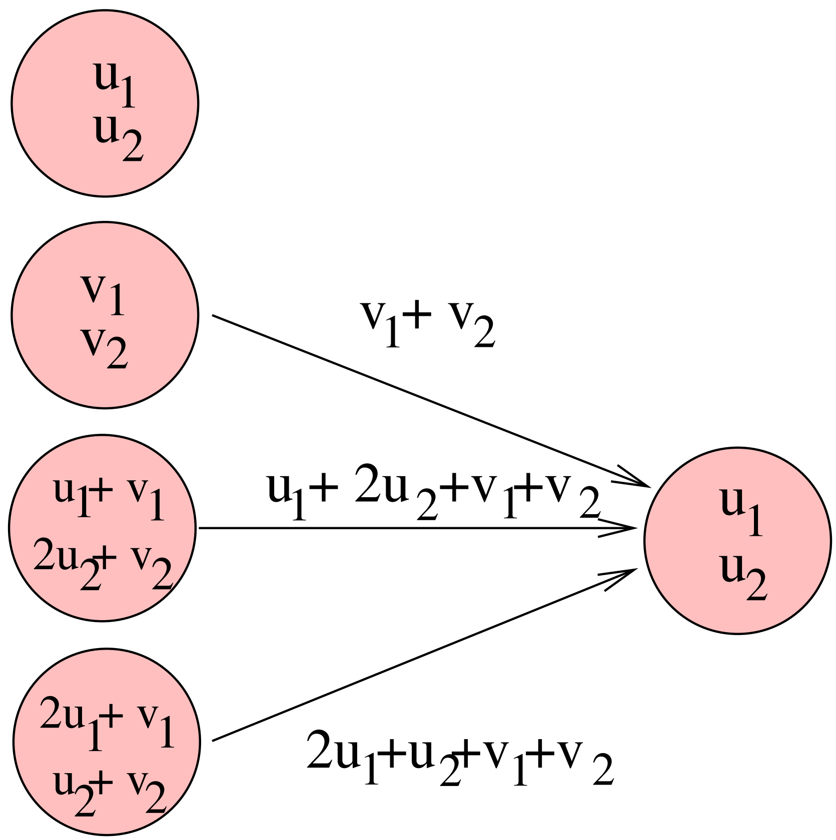
Network code



Repairing a node

File: u,v

$u = u_1, u_2$
$v = v_1, v_2$



locality= 3
repair bandwidth= 1.5G

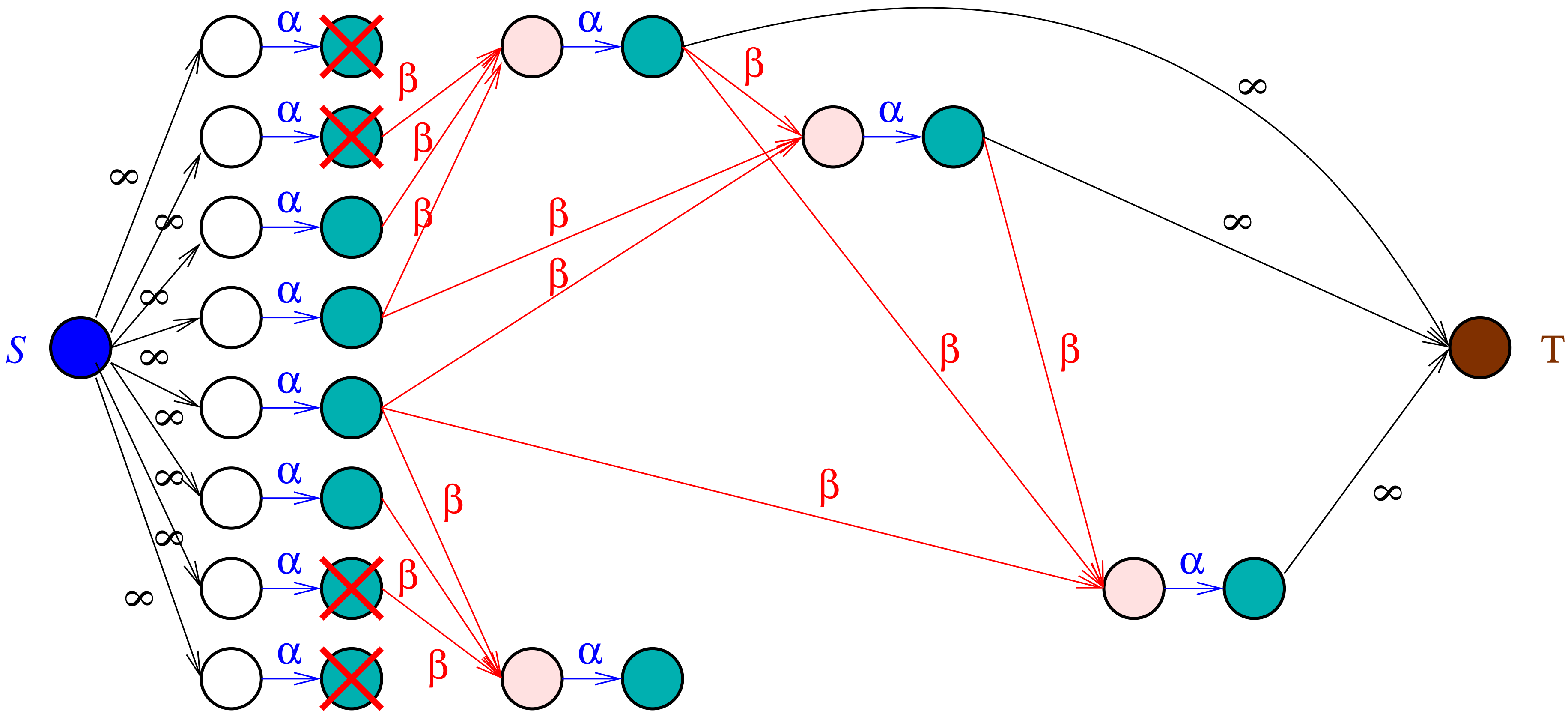
The fundamental limit

Theorem 1. *If there exists a code and a recovery procedure meeting these constraints we have*

$$B \leq \sum_{i=0}^{d-1} \min\left((d-i)\beta, \alpha\right) \tag{1}$$

Definition[regenerating code] A code is said to be regenerating iff its parameters meet the bound (1)

Tool 1 : The information flow graph



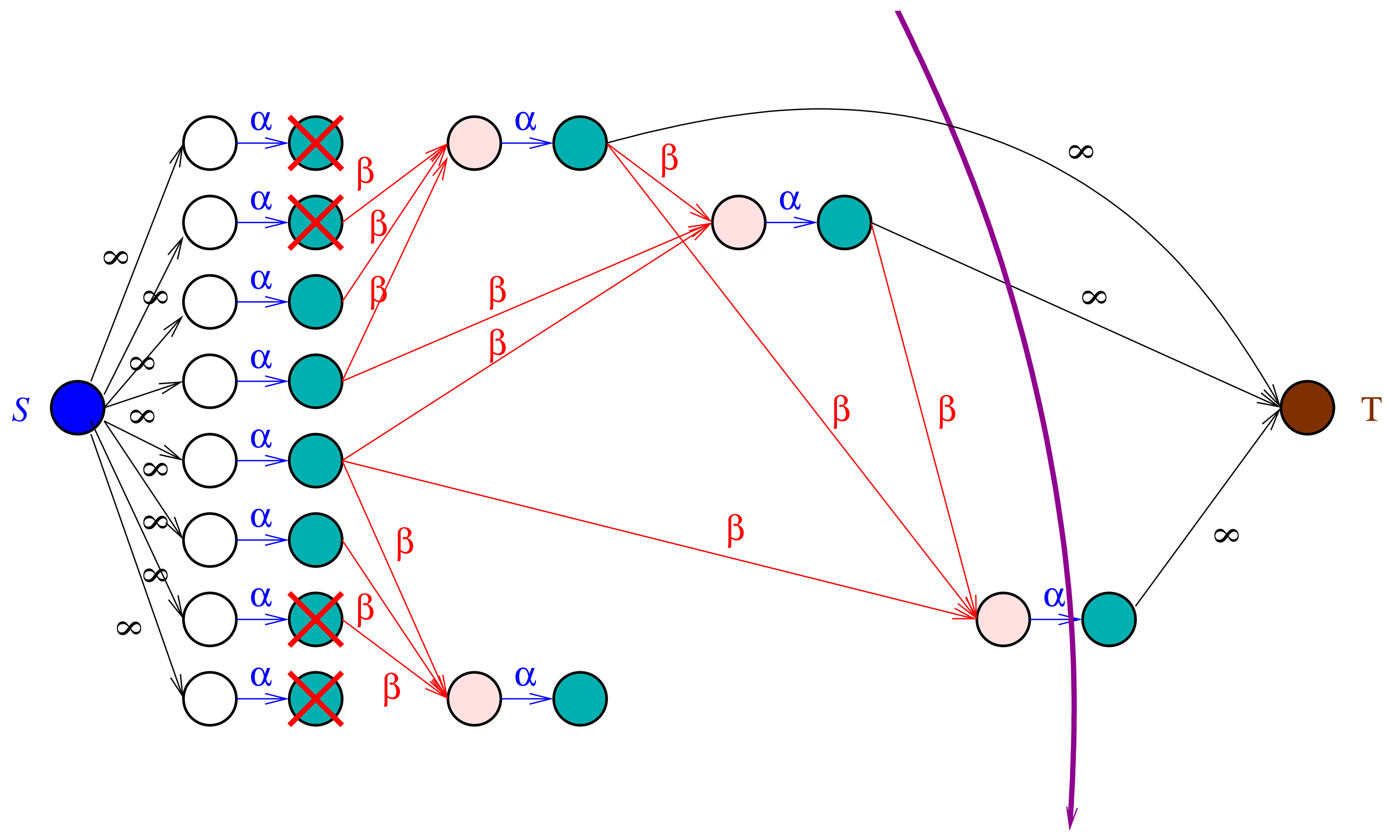
Tool 2 : min cut bound

Lemma 1.

$$B \leq \min_G \mathit{MinCut}(T)$$

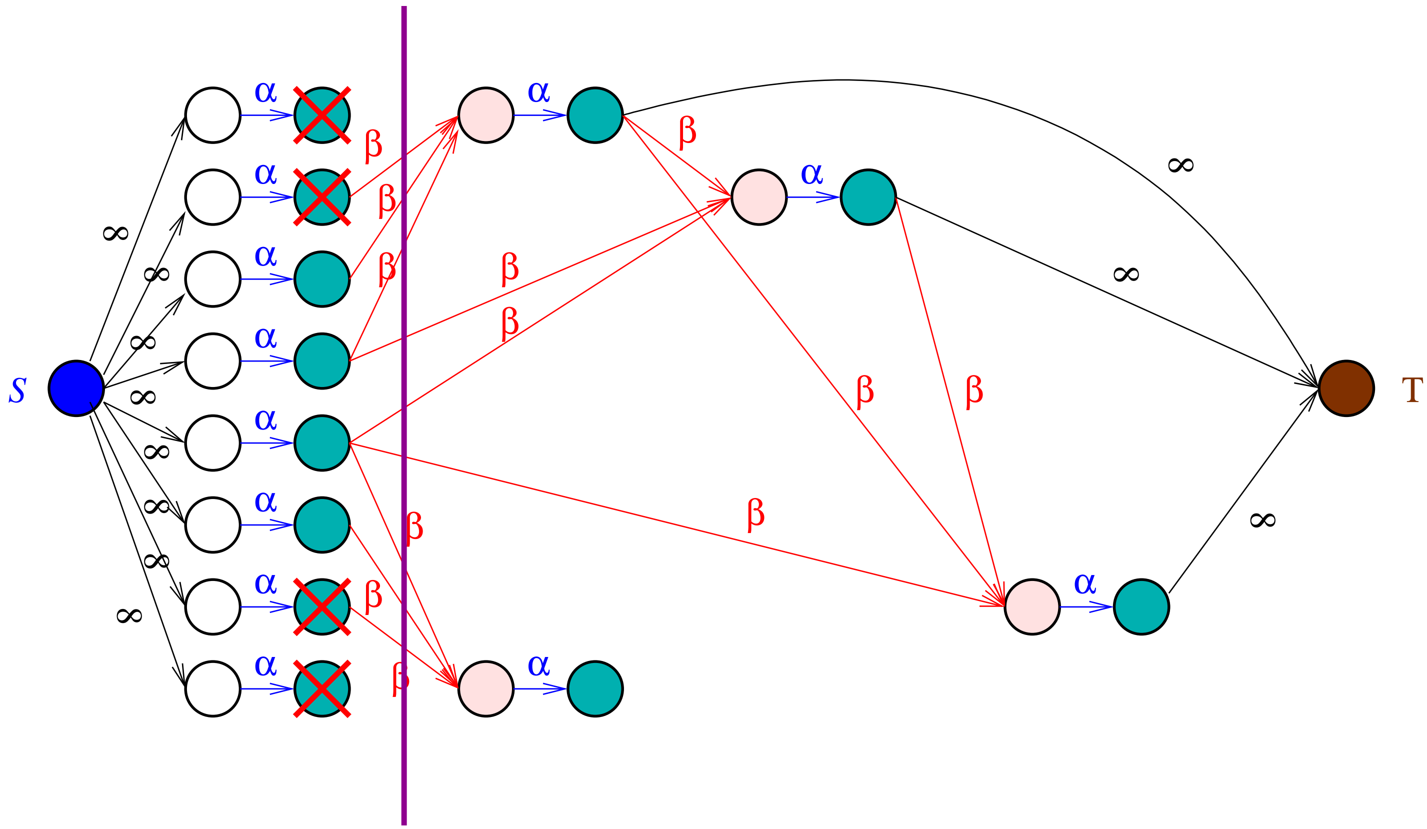
Example of a cut

$$\text{Cut}(T) = \infty + \infty + \alpha = \infty$$



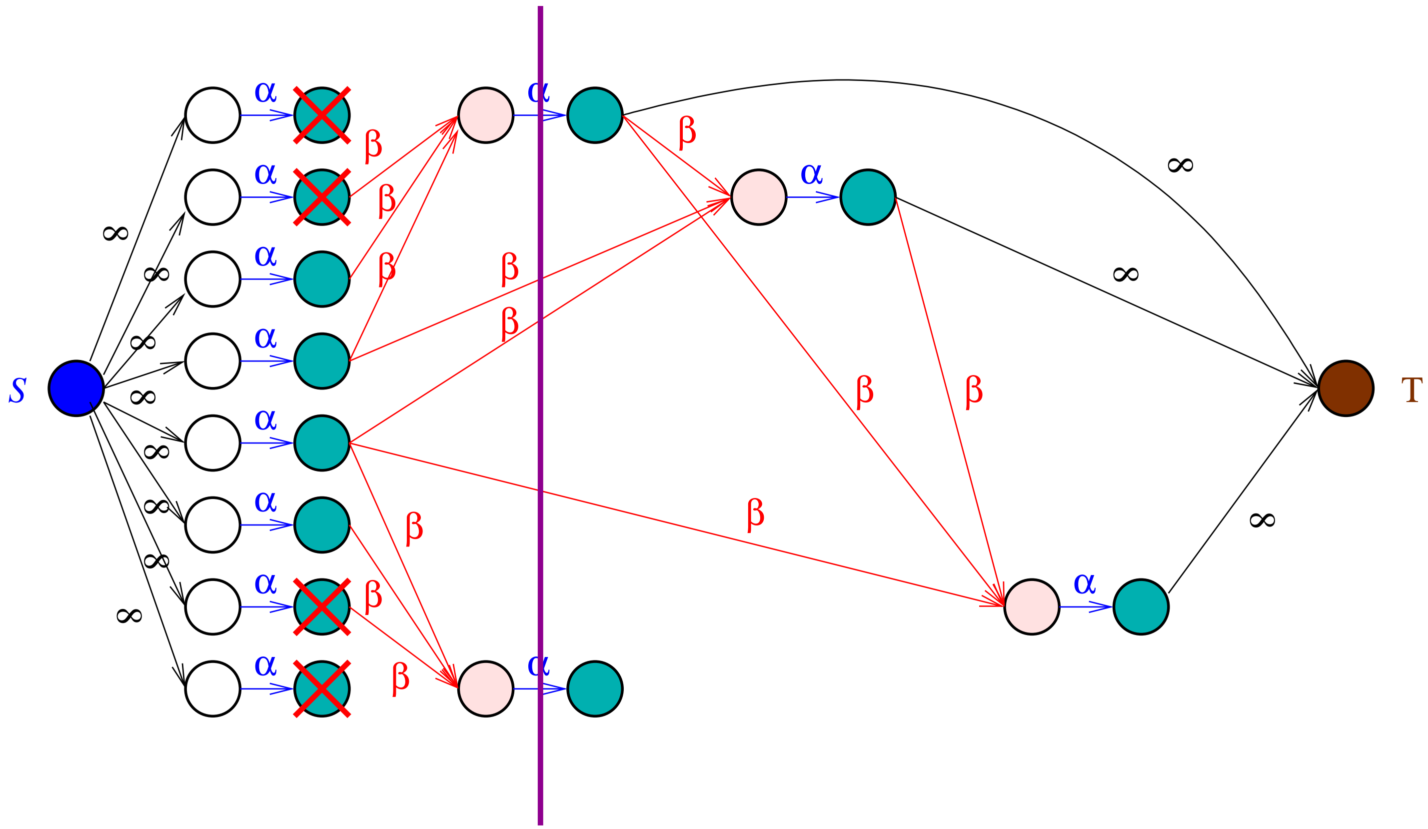
Example of a cut

$$\mathsf{Cut}(T) = 9\beta$$



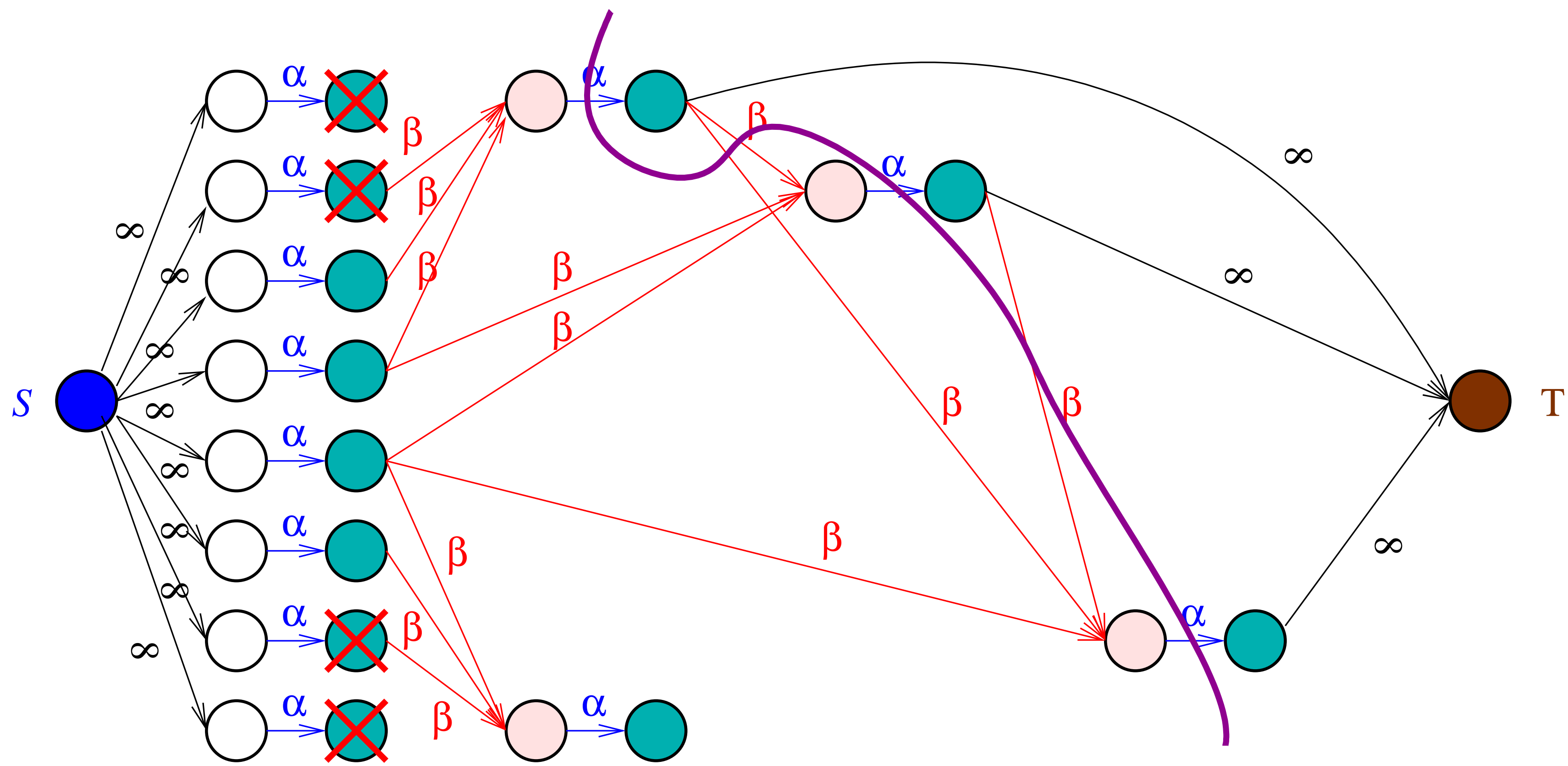
Example of a cut

$$\text{Cut}(T) = 3\beta + 2\alpha$$



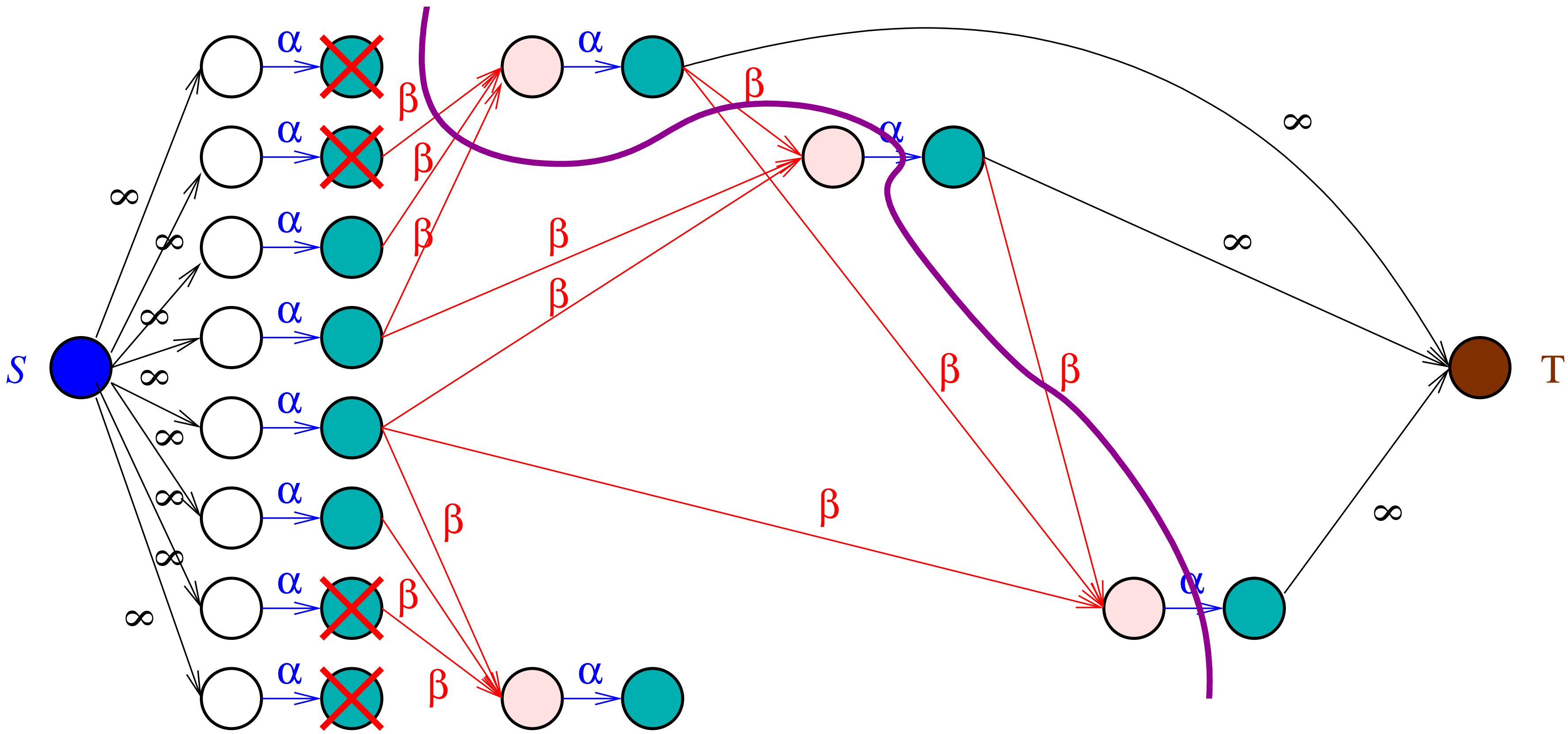
Example of a cut

$$\text{Cut}(T) = 3\alpha$$



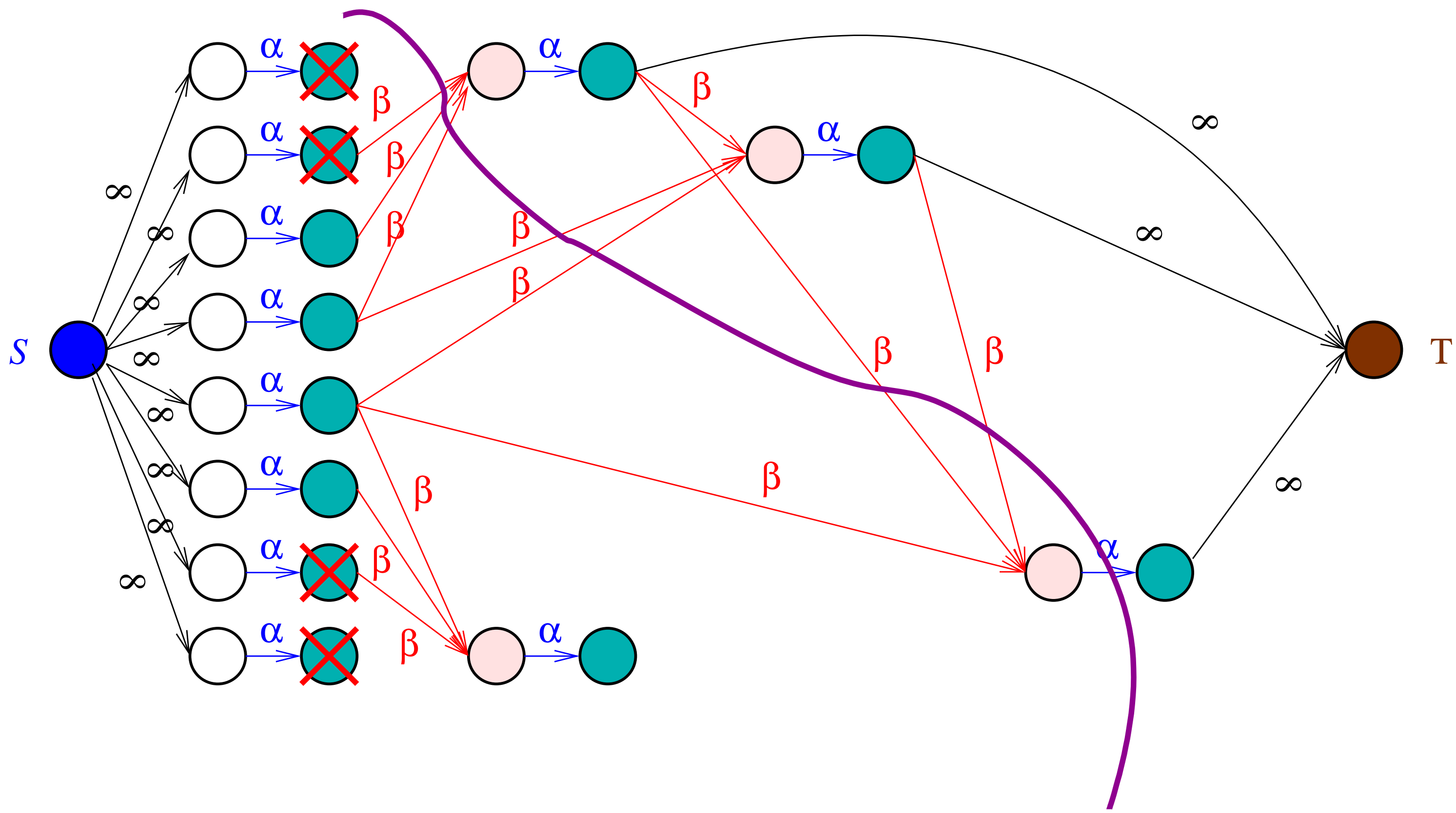
Example of a cut

$$\text{Cut}(T) = 3\beta + 2\alpha$$



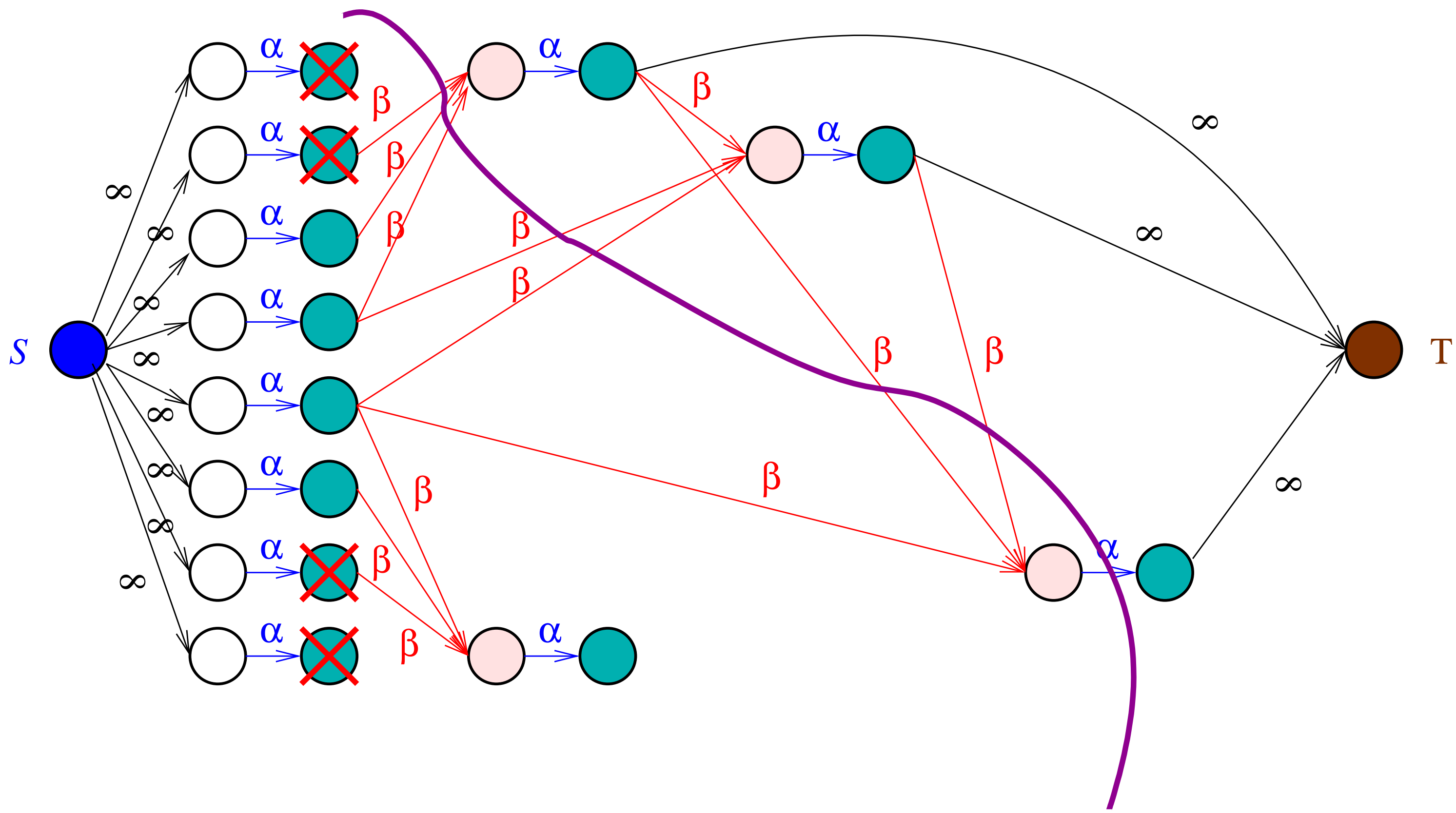
Example of a cut

$$\text{Cut}(T) = 5\beta + \alpha$$



Example of a cut

$$\mathsf{Cut}(T) = 6\beta$$



An example

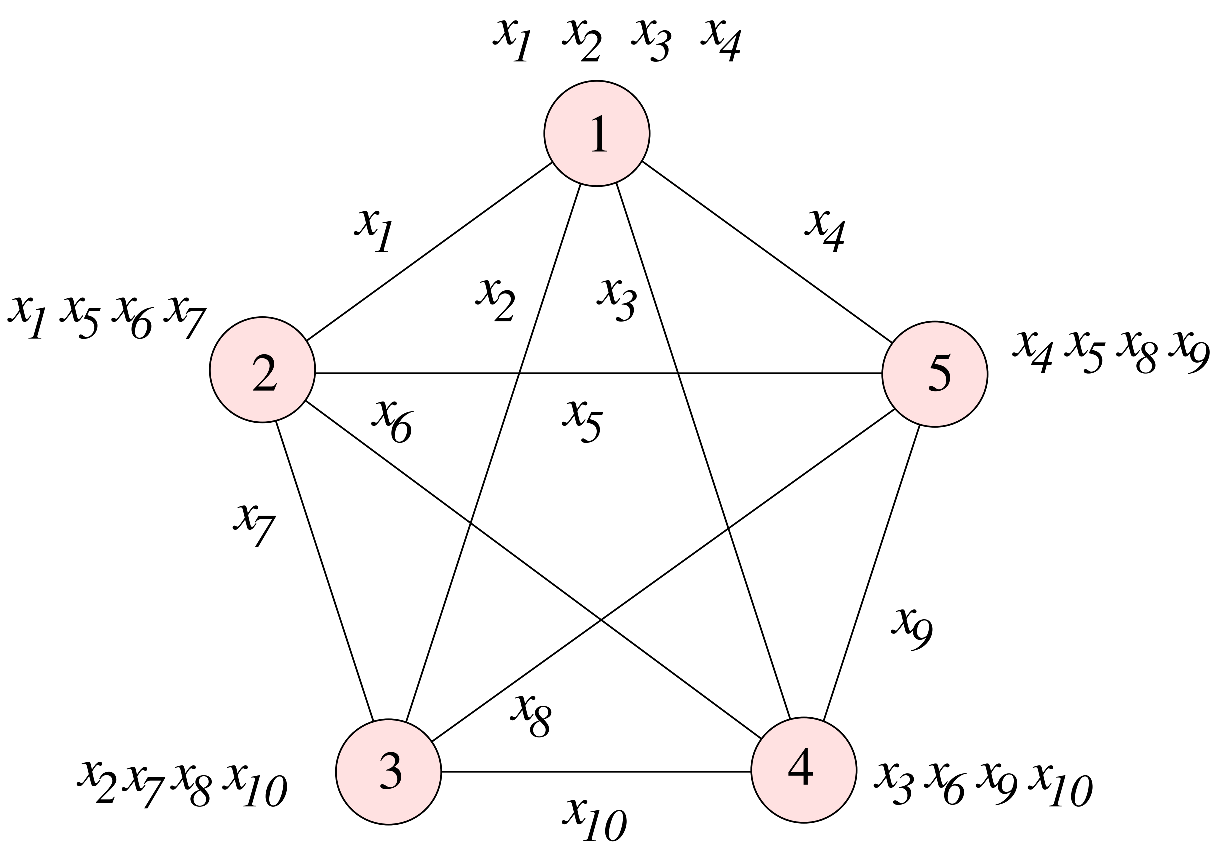
- $n = 10$, $k = 9$, $e = 4$, $d = 3$, $\alpha = 4$, $\beta = 1$
- Corresponding MDS code : single parity-check code $\mathcal{C}$:

$$x_1 \cdots x_{10} \in \mathcal{C} \Leftrightarrow \sum_{i=1}^{10} x_i = 0$$

- File $u_1 \cdots u_9$ encoded into $u_1 \cdots u_{10}$ with $u_{10} = -\sum_{i=1}^{9} u_i$
- Complete graph on 5 vertices=nodes
- each edge carries an x_i
- each node gets the 4 x_i's attached to its 4 incident edges

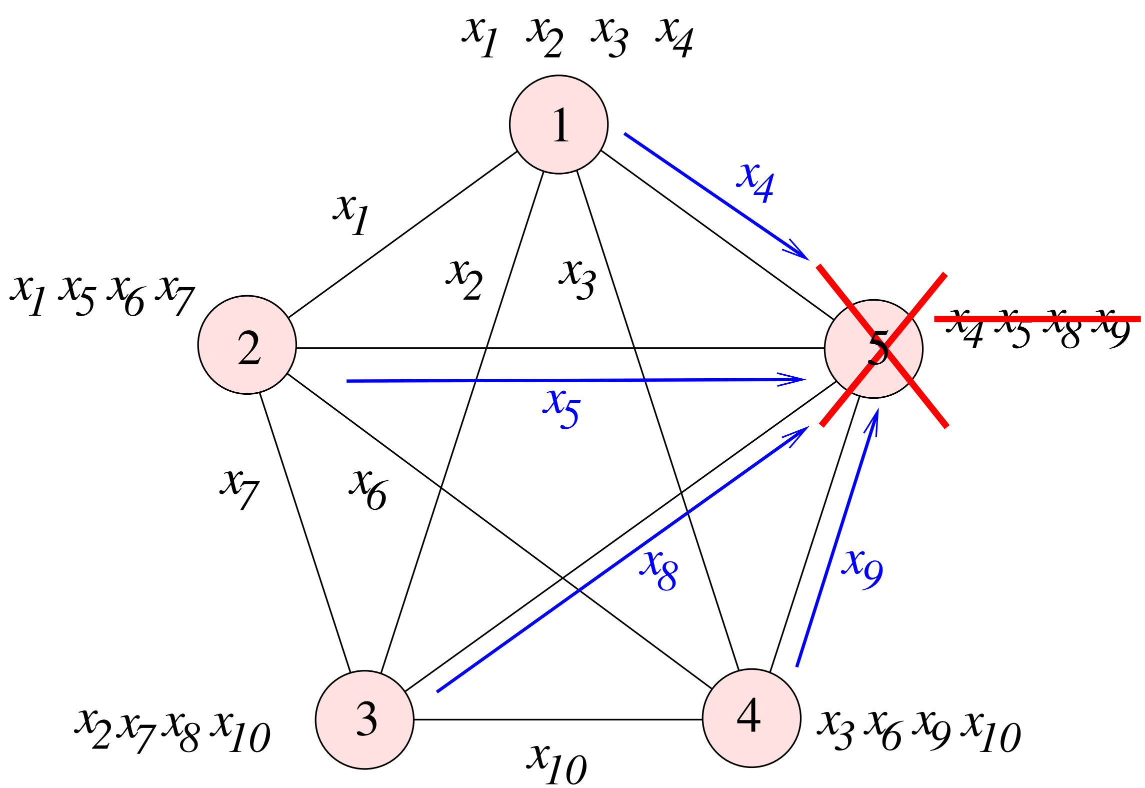
each symbol x_i is replicated twice

Example


x_1 x_2 x_3 x_4
1
x_1
x_4
x_2
x_3
x_1 x_5 x_6 x_7
2
5
x_4 x_5 x_8 x_9
x_6
x_5
x_7
x_9
x_8
x_2 x_7 x_8 x_{10}
3
4
x_3 x_6 x_9 x_{10}
x_{10}

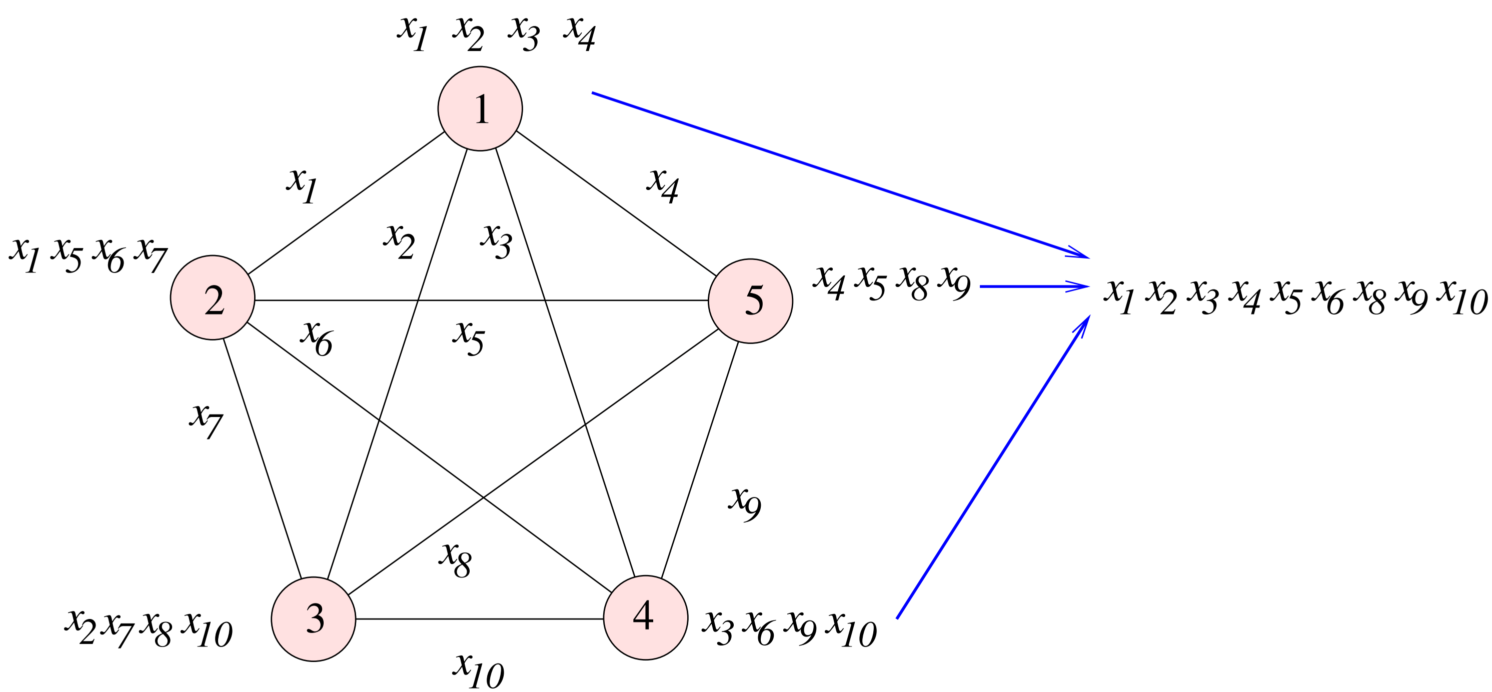
Repairing a node

- if a node fails, request lost symbols from adjacent nodes
- bandwidth=4



Recovering the whole file

- any node carries 4 symbols
- any 2 nodes carry 7 different symbols
- any 3 nodes carry 9 different symbols



$$x_7 = -x_1 - x_2 - x_3 - x_4 - x_5 - x_6 - x_8 - x_9 - x_{10}$$