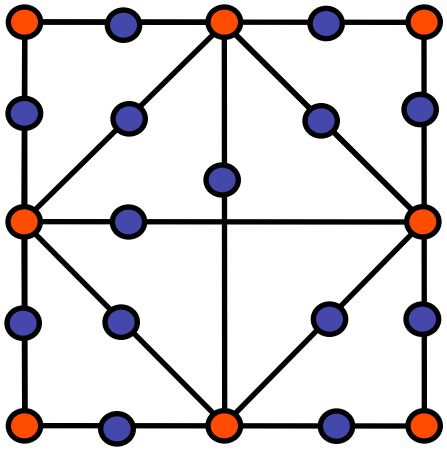
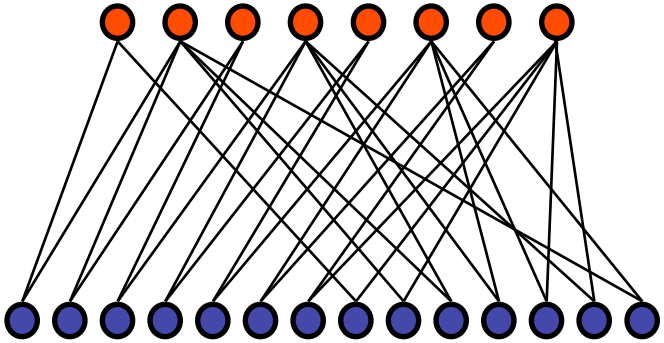
algolma
laboratoire d'algorithmique
laboratoire de mathematiques algorithmique

Codes and Graphs





Amin Shokrollahi
Laboratoire d'algorithmique
Laboratoire de mathematiques algorithmiques
EPFL

Codes

A code $\mathcal{C}$ of blocklength n and dimension k is a k-dimensional subspace of $\mathbf{F}_2^n$

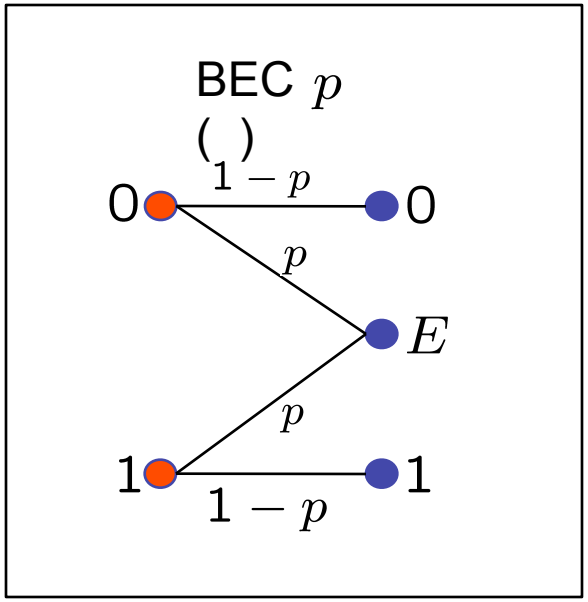
Codes are used for transmission of information on unreliable channels.

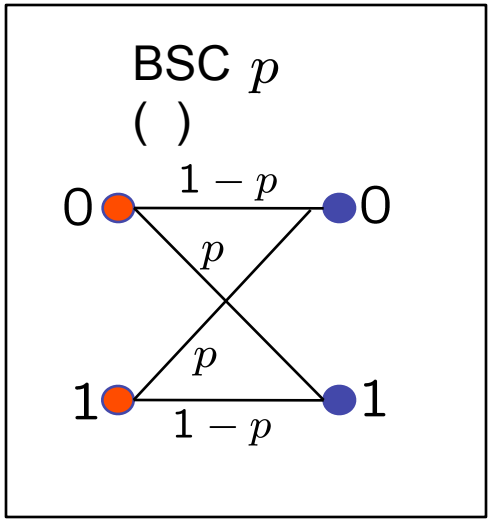
Encoding problem: Given binary string $(x_1, \ldots, x_k)$, encode it to a codeword $(y_1, \ldots, y_n)$.

Codeword is transmitted over a channel, a corrupted version is received.

Decoding problem: Find the best estimate of the original codeword given the received one.

Channels and Capacity





Capacity = $1-p$

Capacity = $1-h(p)$

Shannon, 1949: Reliable communication not possible at rates above capacity.

Reliable communication possible at rates less than capacity with error probability $O(e^{-\gamma n})$. (Random coding!)

Problems

Random codes come arbitrarily close to capacity, but with a decoding algorithm with exponential running time.

Explicit codes that achieve capacity and have polynomial running times?

Forney's concatenated codes achieve capacity with polynomial running time.

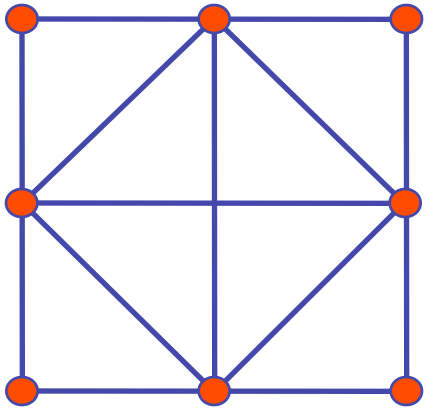
But: If rate is Capacity-ε, then decoding time is polynomial in blocklength n and $2^{1/\varepsilon}$.

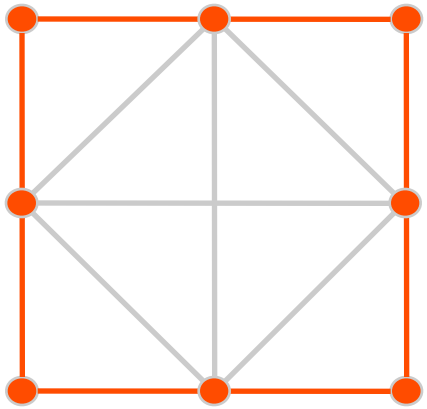
This is not practical. Can we do better?

Summary

	What we have	What we want
Encoding	$O(nk)$	$O(n)$
Decoding	$\mathrm{poly}(n, 2^{1/\varepsilon})$	$\mathrm{poly}(n, 1/\varepsilon)$

Cycle Codes





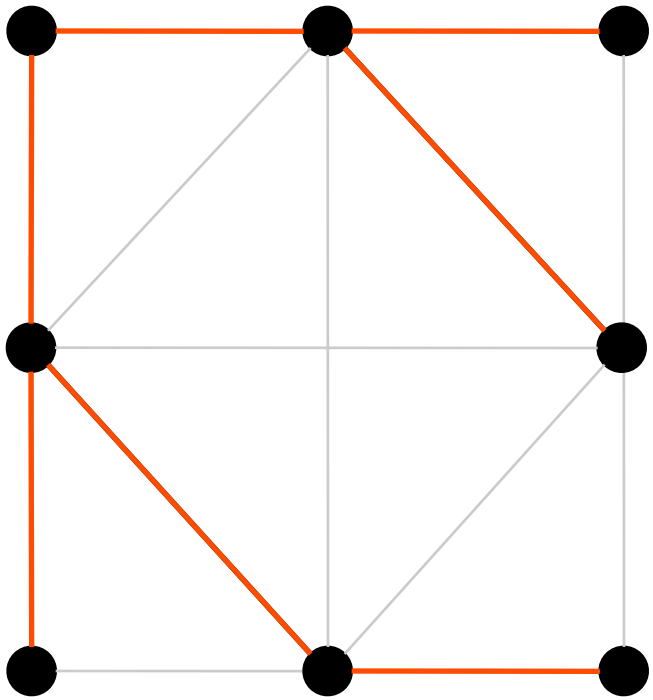
Hakimi and Frank, 1965, Hakimi and Bredeson, 1968.

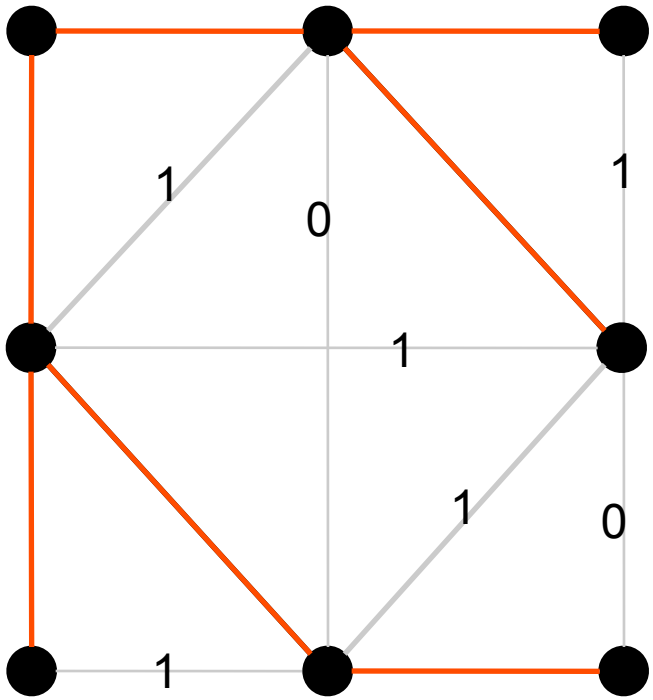
Codewords are cycles.

Codewords are binary assignments to edges such that for every node the sum of the adjacent edges is 0.

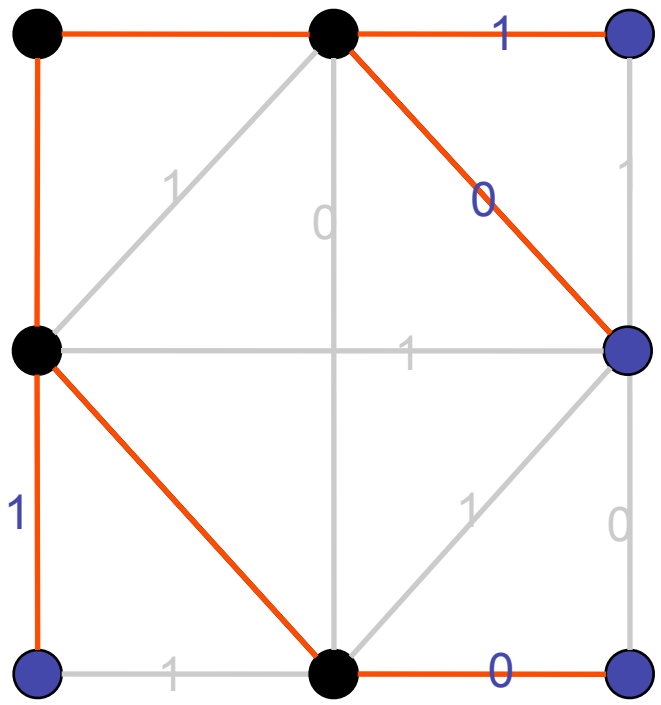
Encoding? Decoding?

Encoding: Spanning Tree (Forest)

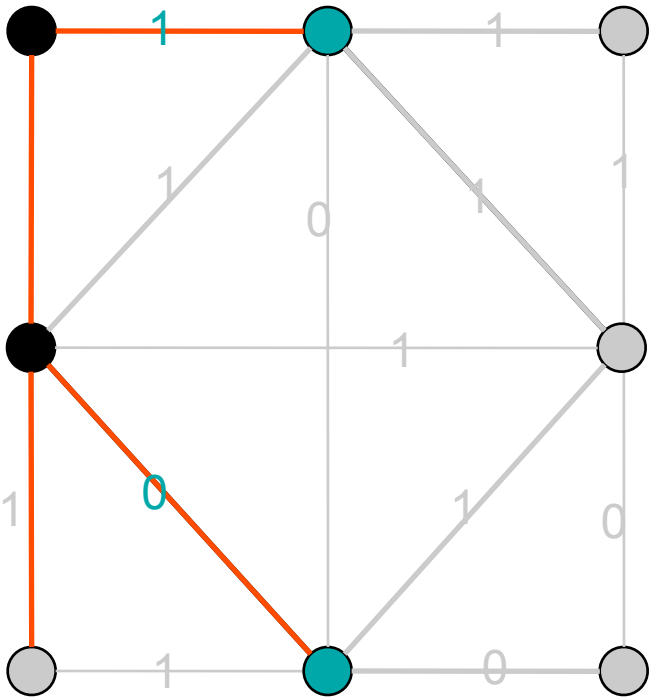




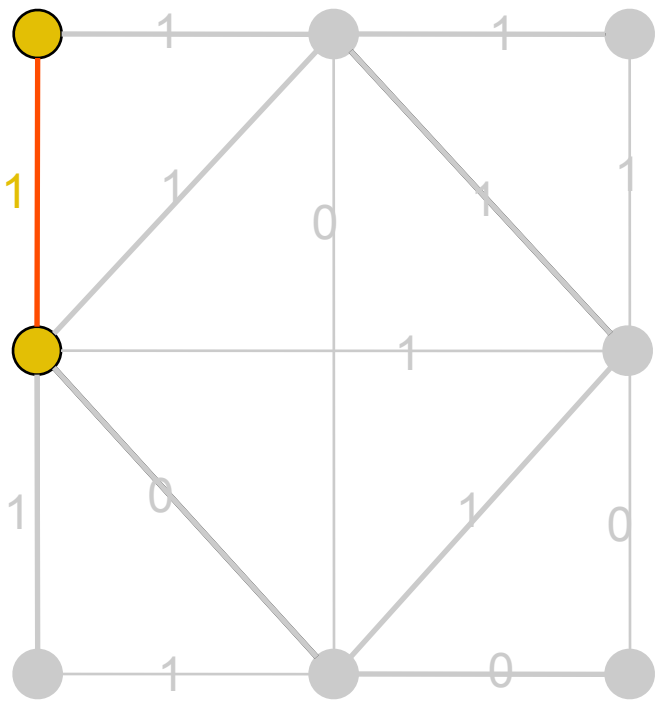
Information positions = Edges outside SP



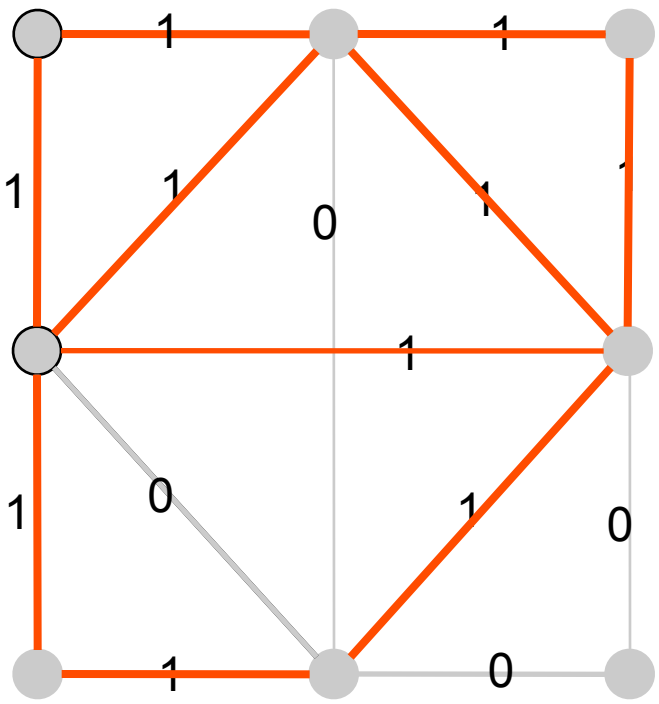
Find values of edges connected to leaves



Find value of edges of reduced degree one



Find value of edges of reduced degree one



Codeword

Since spanning tree has $r - 1$ edges, rate of the code is

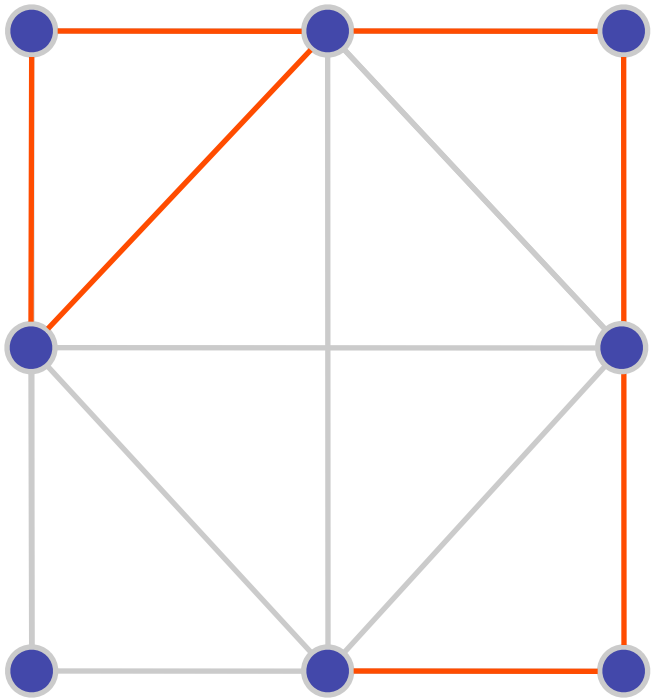
$$1 - \frac{r-1}{n}$$

Encoding is linear time for cycle codes if graph is sparse.

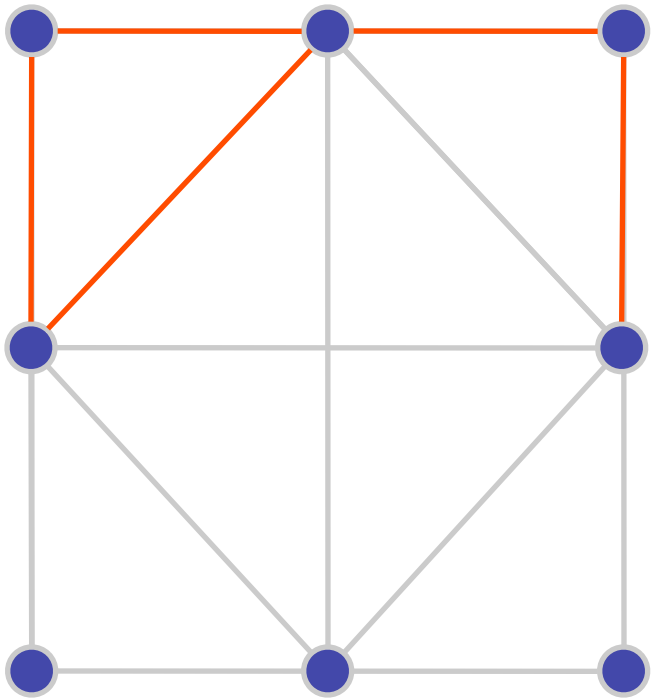
Decoding? Demonstration for binary erasure channel.

Edges are lost independently with probability p.

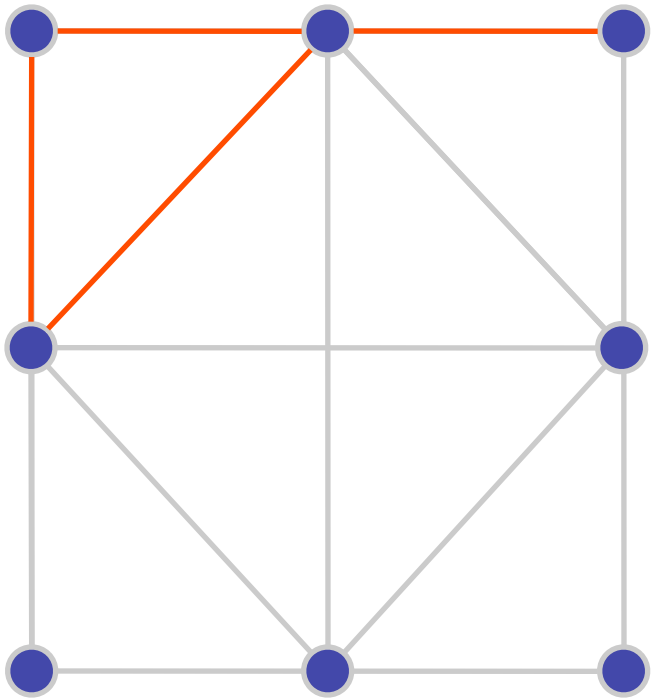
Decoding on the BEC



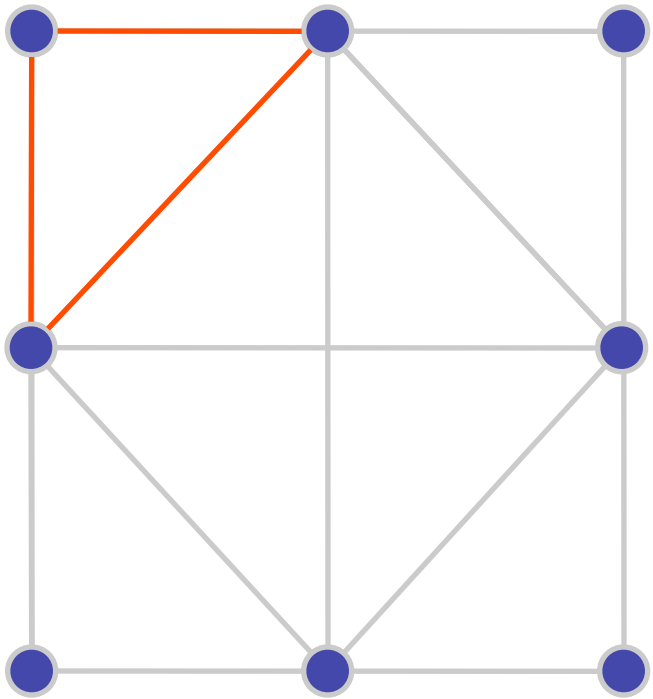
Decoding on the BEC



Decoding on the BEC



Decoding on the BEC



Decoding cannot be continued at this point!

How many errors can we correct?

Decoding is linear time for sparse graphs.

Decoding is successful if and only if the erased edges form a forest.

Obstruction to decoding is existence of a 2-core.

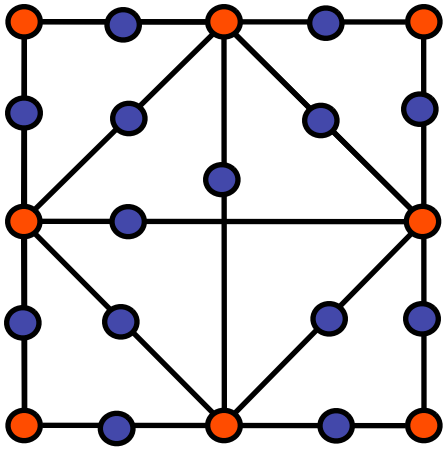
Decoding error is below constant only if all 2-cores are of size $o(n)$

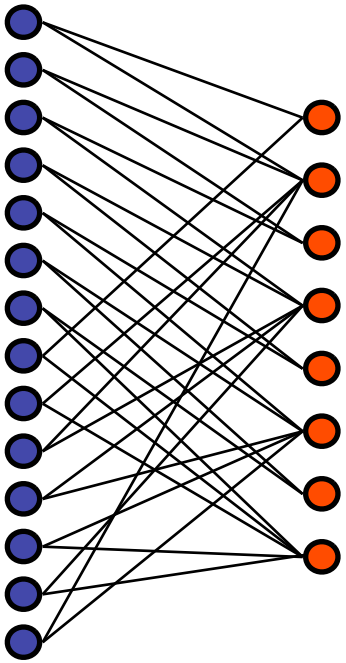
For random graphs this happens iff average degree is smaller than 1.

Decoding not successful if erasure probability is above $\frac{r}{2n}$

This is only half the capacity! Similar results hold for other channels.

LDPC Codes

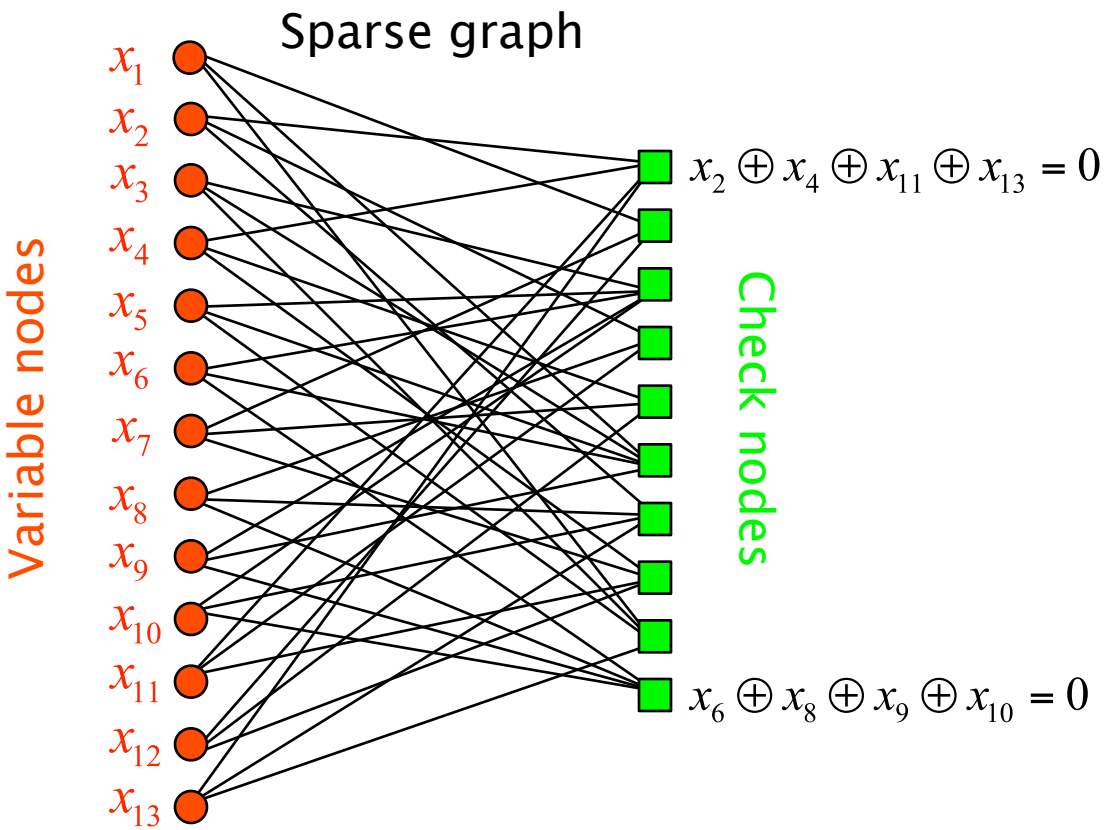




Study general bipartite graphs!

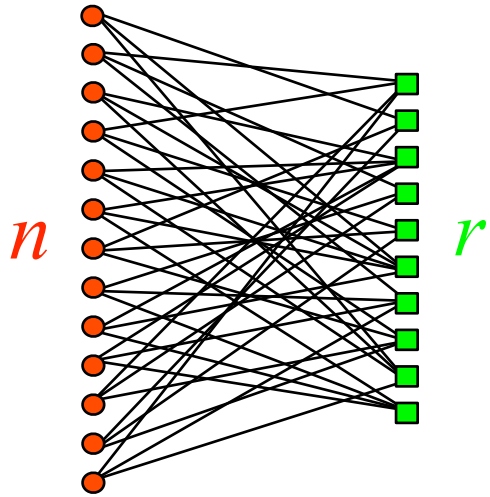
LDPC codes were invented in the early 1960's by Gallager.

Definition


Sparse graph
Variable nodes
Check nodes
x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
x_9
x_{10}
x_{11}
x_{12}
x_{13}
$x_2 \oplus x_4 \oplus x_{11} \oplus x_{13} = 0$
$x_6 \oplus x_8 \oplus x_9 \oplus x_{10} = 0$

Structural Properties

Average variable node degree a_v



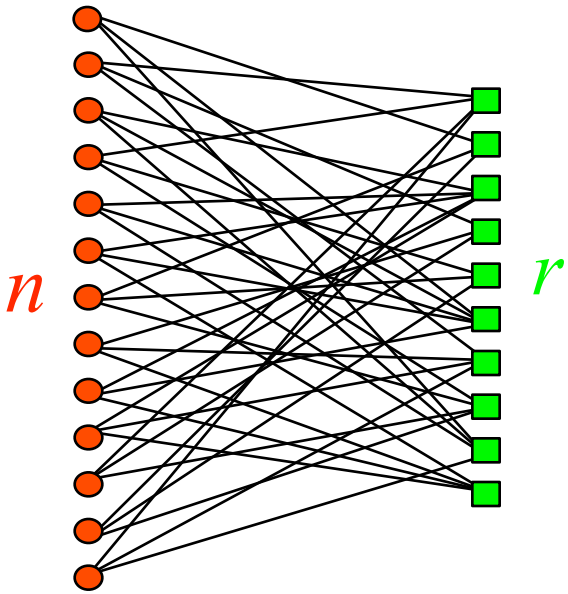
Average check node degree a_c

$$\frac{r}{n} = \frac{a_v}{a_c}$$

Block-Length and Dimension

Block length = n

Rate $\geq 1 - \frac{r}{n}$



Decoding Algorithms

Want to infer the values of the variable nodes from the received values.

Bit-flipping algorithm (Zyablov-Pinsker, Tanner, Sipser-Spielman, Zemor, Barg-Zemor, ….):

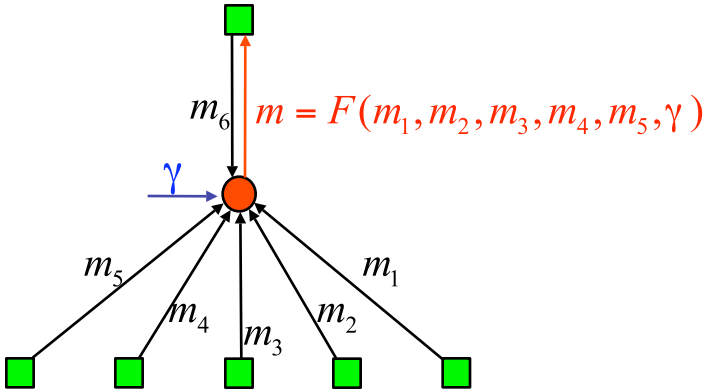
Check bits are satisfied if the sum of adjacent variable nodes is zero, unsatisfied otherwise.

Variable nodes flip their received value if the number of their unsatisfied neighbors is larger than the number of satisfied neighbors.

Message Passing

Algorithm proceeds in rounds. At every round messages are passed along the edges from variable nodes to check nodes, and then from check nodes back to variable nodes.

A message passed from a variable node v to a check node c may take into account the received value, and all the values received from adjacent check nodes other than c in the previous round. (Same for check and variable nodes.)

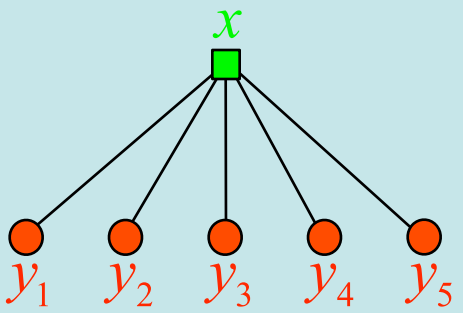


Message Passing Rationale

$\text{Prob}[y_i = 1] = p, y_i$ independent

$\text{Prob}[x = 1] = q$

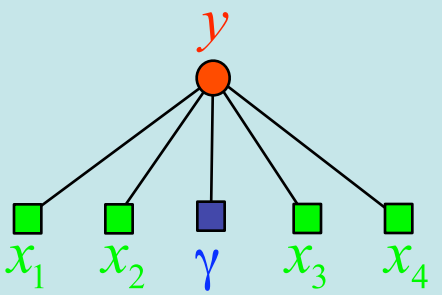
$\Rightarrow (1-2q) = (1-2p)^5$



$x_1, x_2, x_3, x_4, \gamma$ independent observations of y

$\text{Prob}[x_i = 1 \mid y] = q, \ \text{Prob}[\gamma = 1 \mid y] = u$

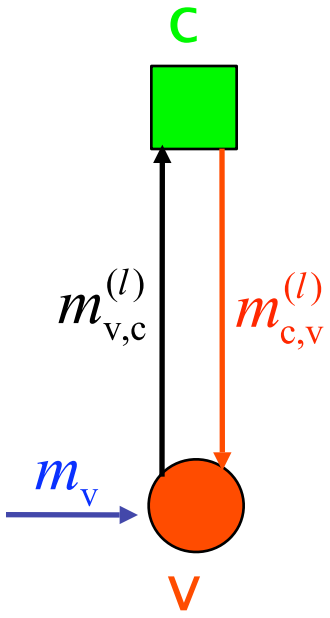
$$\frac{\text{Prob}[y = 1 \mid \text{observations}]}{\text{Prob}[y = 0 \mid \text{observations}]} = \frac{q^4 u}{(1-q)^4 (1-u)}$$



Belief-Propagation Algorithm

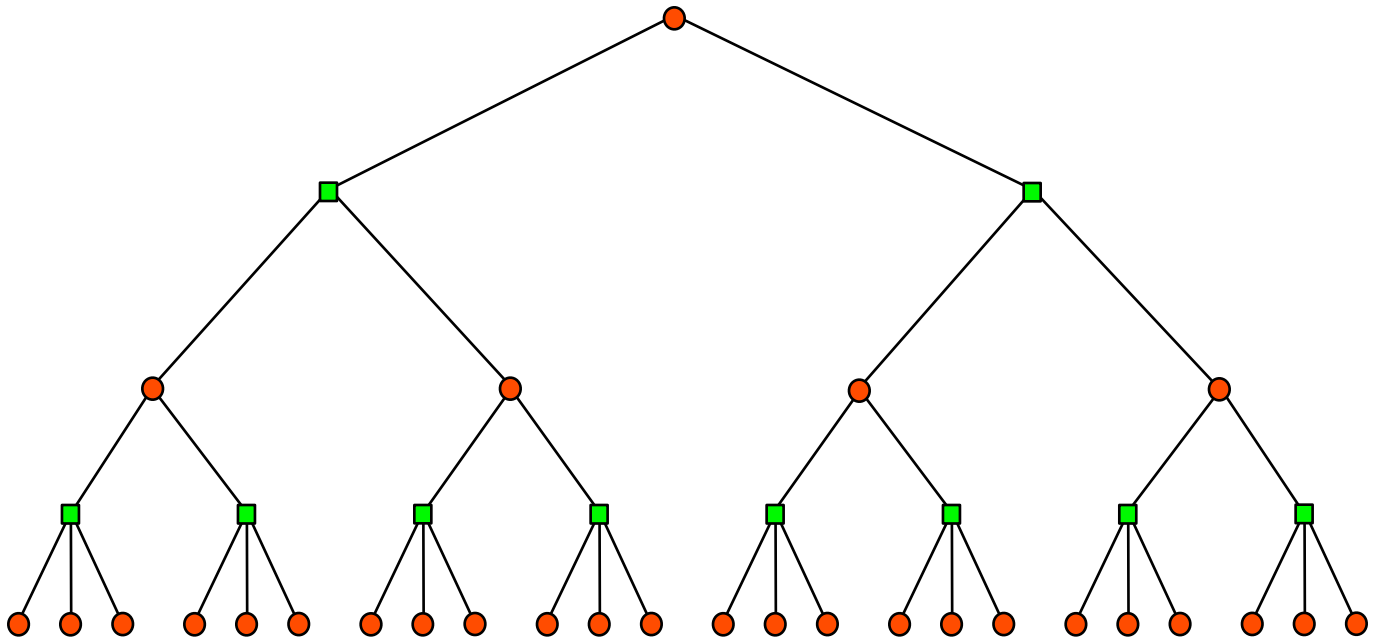
$$m_{v,c}^{(l)} = \begin{cases} m_v, & l = 0 \\ m_v + \sum_{c' \in C_v \setminus \{c\}} m_{c',v}^{(l-1)}, & l \geq 1 \end{cases}$$

$$m_{c,v}^{(l)} = \ln \frac{1 + \prod_{v' \in V_c \setminus \{v\}} \tanh(m_{v',c}^{(l)} / 2)}{1 - \prod_{v' \in V_c \setminus \{v\}} \tanh(m_{v',c}^{(l)} / 2)}$$

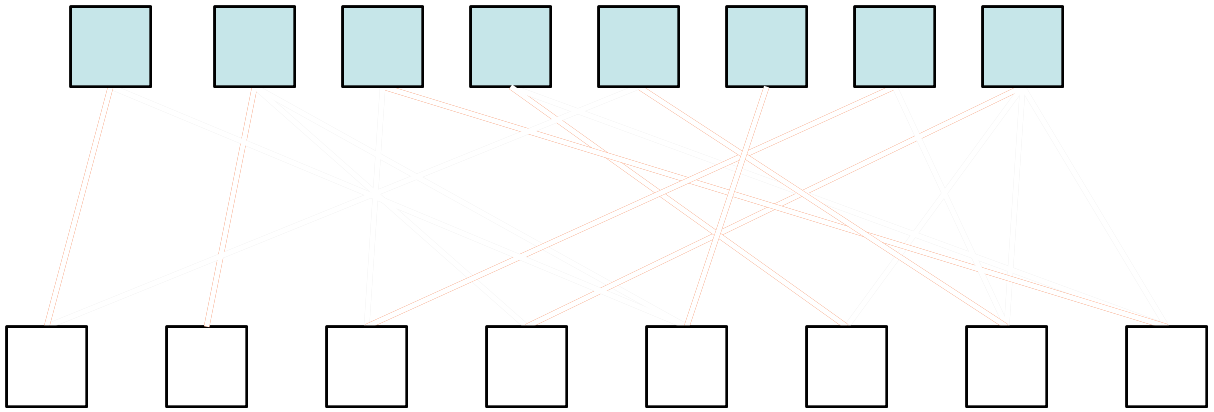


BP and Trees

The BP algorithm is (by construction) exact on trees, i.e., it gives the maximum information for the root that can be obtained from all the observations.



Example: BP on the Erasure Channel



Complexity

In every round the BP decoder performs a constant number of operations per edge.

Running the BP decoder for a constant number of rounds leads to a linear time algorithm if the underlying graph is sparse.

In the binary erasure channel the BP decoder is linear time if it succeeds.

Encoding

Naïve encoding can be done in time $O(n^2)$ after a pre-processing step with cost $O(n^3)$.

Using a cascade of low-density generator codes leads to linear time encoding. (Luby et al.)

Use of Repeat-accumulate codes leads to linear time encoding (The CalTech group.)

For certain graph structures a clever application of the erasure decoder can lead to linear time algorithms (Richardson and Urbanke). Interestingly, algorithm does not specialize to spanning tree algorithm for cycle codes.

Analysis

Take random bi-regular bipartite graphs, and simulate behavior:

(2,4)	33%
(3,6)	43%
(4,8)	39%
(5,10)	34%

Should be able to tolerate close to 50%. Regular graphs are not good. Need irregular degree structure.

Analysis

Luby, Mitzenmacher, Shokrollahi, Spielman, 1997:

Let λ and ρ be probability distributions on set $\{1, \ldots, D\}$

$$\lambda_i := \text{Prob}[\lambda = i] \qquad \rho_i := \text{Prob}[\rho = i]$$

$$\lambda(x) = \sum_i \lambda_i x^{i-1} \qquad \rho(x) = \sum_i \rho_i x^{i-1}$$

Let graph be chosen at random subject to the following constraints:
Fraction of edges of left (right) degree i is λ_i (ρ_i).

Then code can decode a p -fraction of erasures iff

$$p\lambda(1 - \rho(1 - x)) < x, \; x \in (0, p).$$