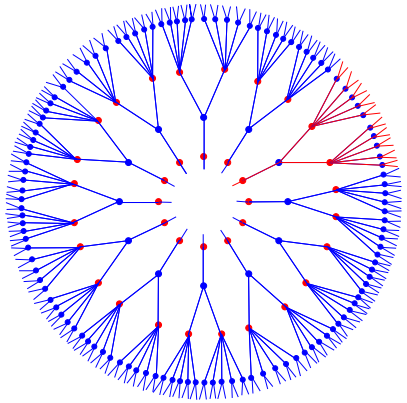
An Introduction to Algorithmic Coding Theory



M. Amin Shokrollahi

Bell Laboratories

Lucent Technologies
Bell Labs Innovations

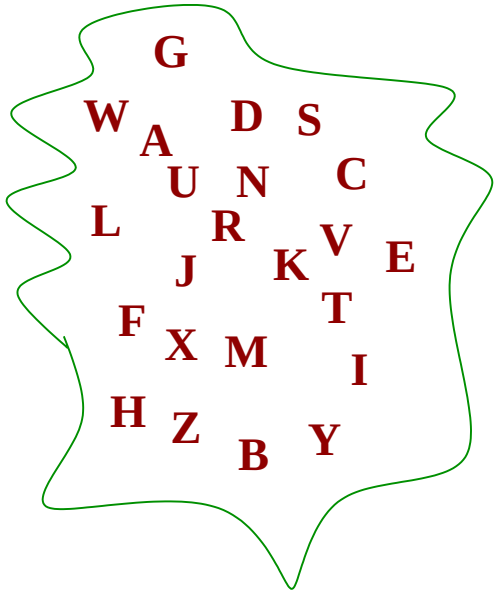
Part 1: Codes

A puzzle

What do the following problems have in common?

Problem 1: Information Transmission

MESSAGE

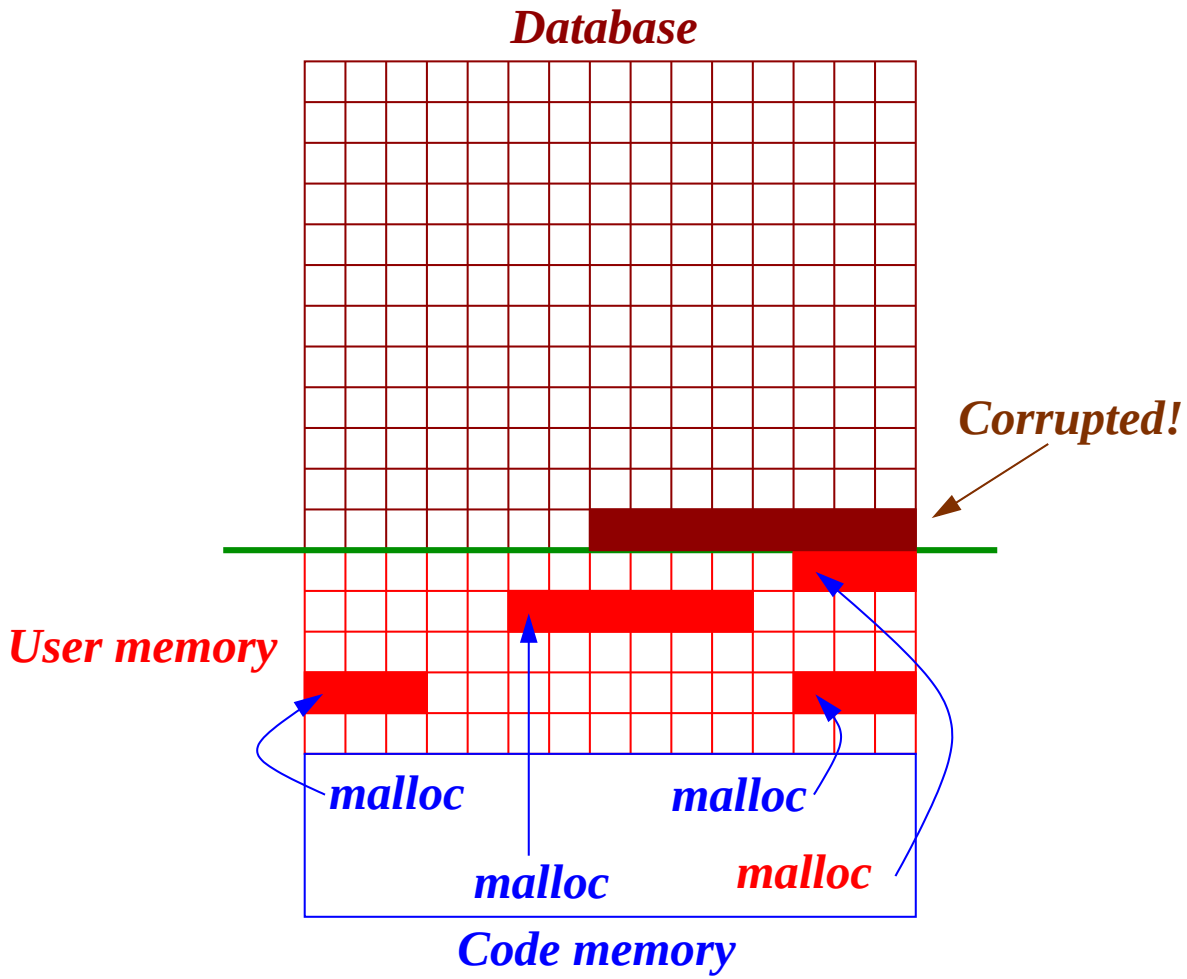


MASSAGE ??

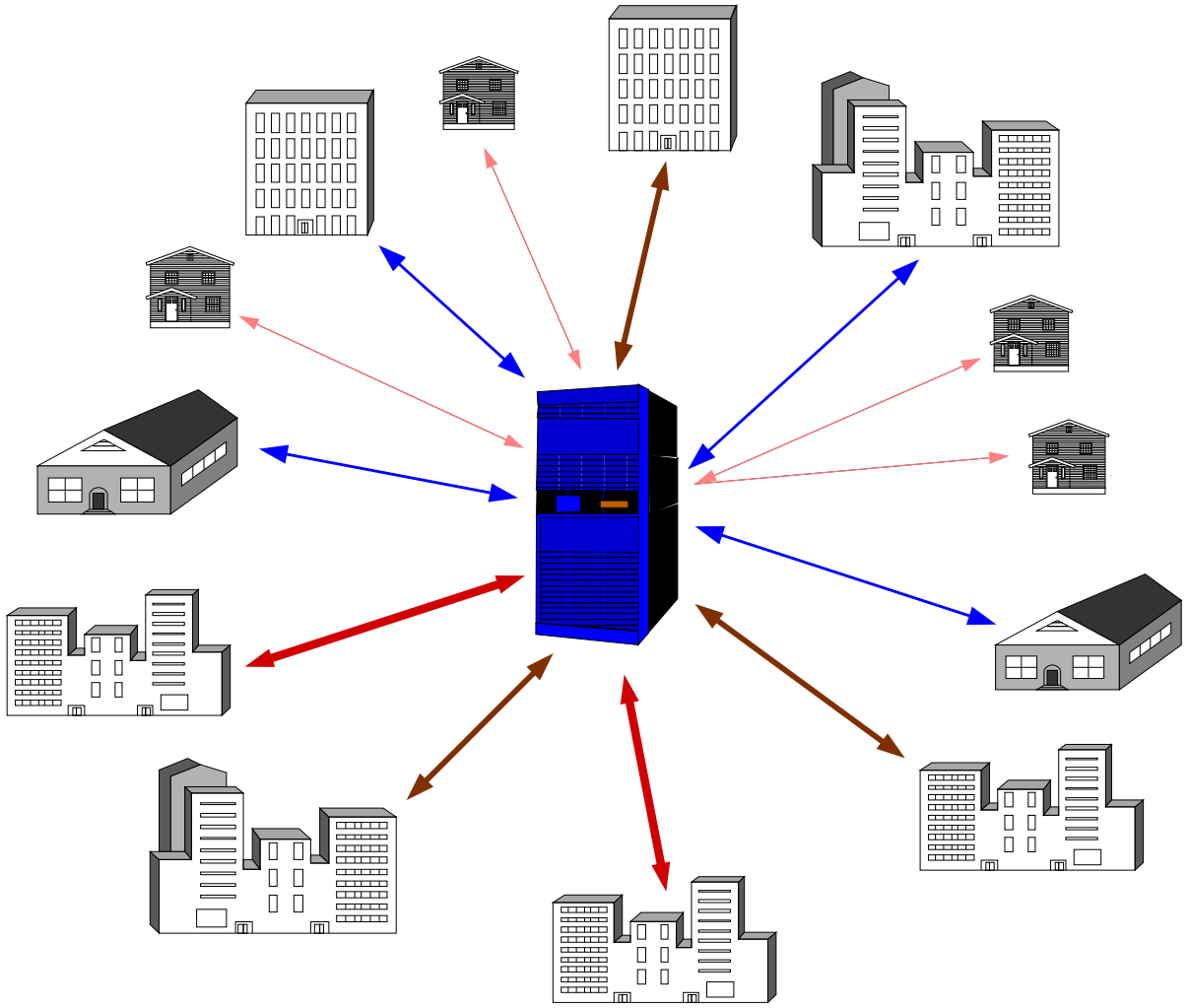
Problem 2: Football Pool Problem

Bayern München	:	1. FC Kaiserslautern	0
Homburg	:	Hannover 96	1
1860 München	:	FC Karlsruhe	2
1. FC Köln	:	Hertha BSC	0
Wolfsburg	:	Stuttgart	0
Bremen	:	Unterhaching	2
Frankfurt	:	Rostock	0
Bielefeld	:	Duisburg	1
Feiburg	:	Schalke 04	2

Problem 3: In-Memory Database Systems



Problem 4: Bulk Data Distribution



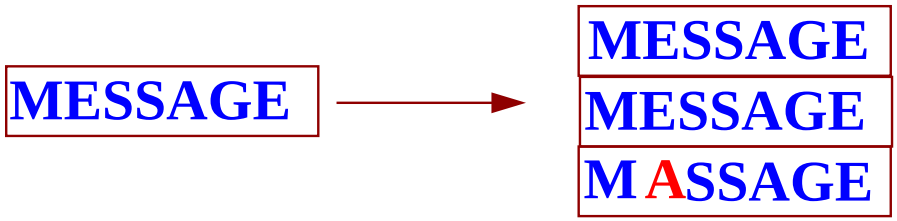
What do they have in common?

They all can be solved using

(algorithmic) coding theory

Basic Idea of Coding

Adding redundancy to be able to correct!



Objectives:

- Add as little redundancy as possible;
- correct as many errors as possible.

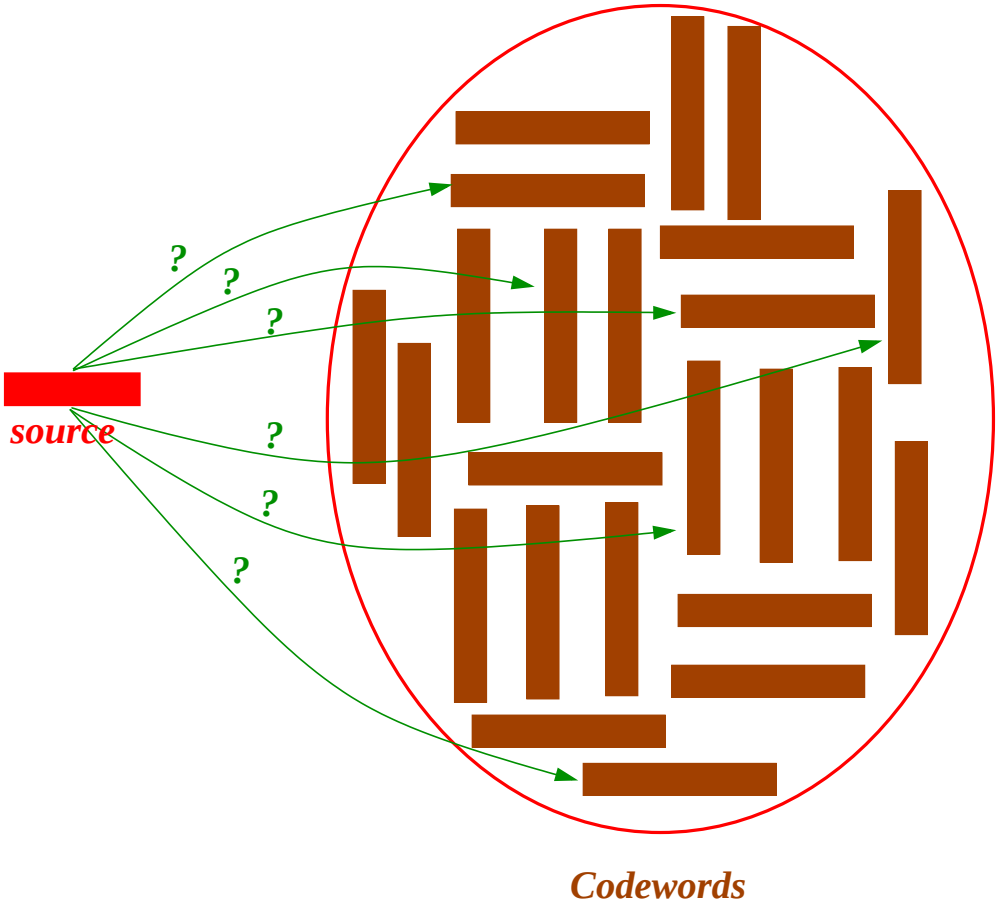
Codes

A code of block-length n over the alphabet $GF(q)$ is a set of vectors in $GF(q)^n$.

If the set forms a vector space over $GF(q)$, then the code is called linear.

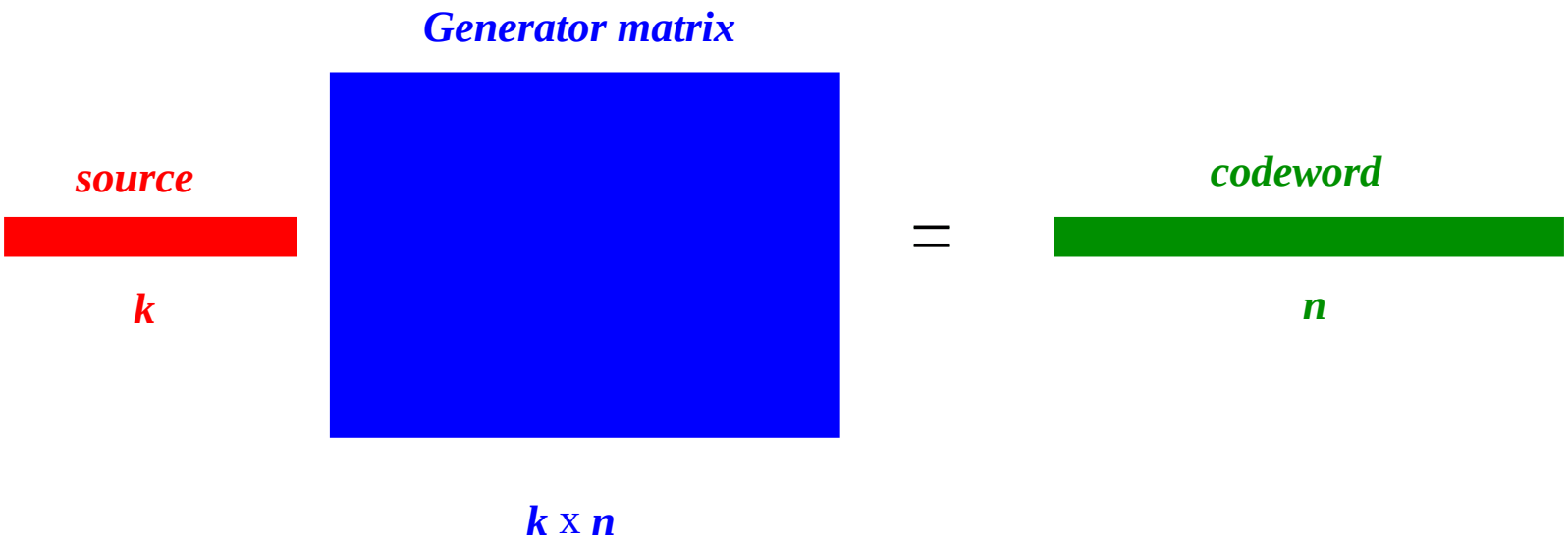
$[n, k]_q$-code

Encoding Problem



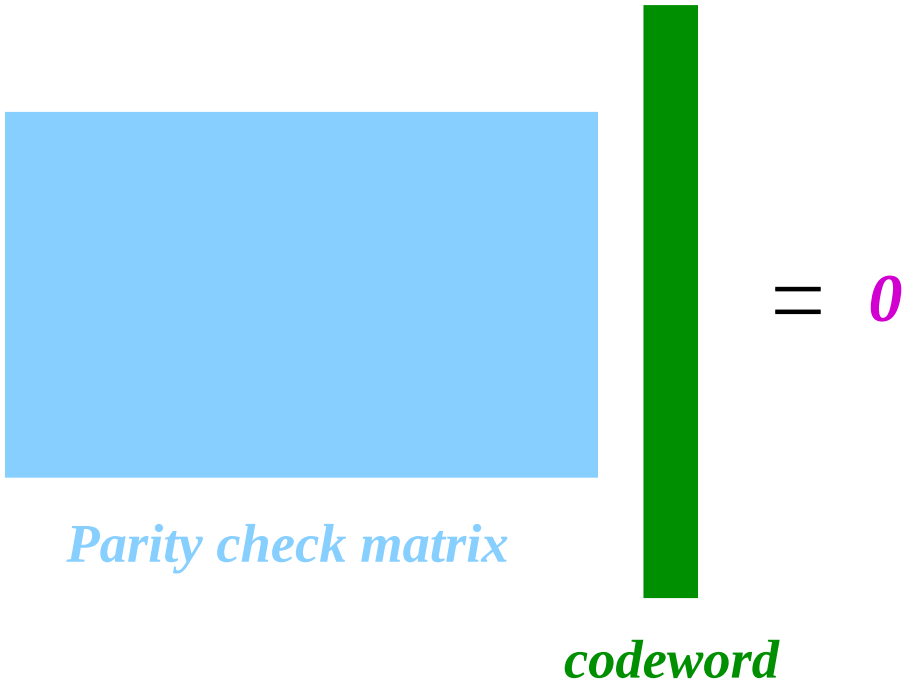
Efficiency!

Linear Codes: Generator Matrix



$O(n^2)$

Linear Codes: Parity Check Matrix



$O(n^2)$ after $O(n^3)$ preprocessing.

The Decoding Problem: Maximum Likelihood Decoding

$$a\ b\ c\ d \mapsto a\ b\ c\ d \mid a\ b\ c\ d \mid a\ b\ c\ d$$

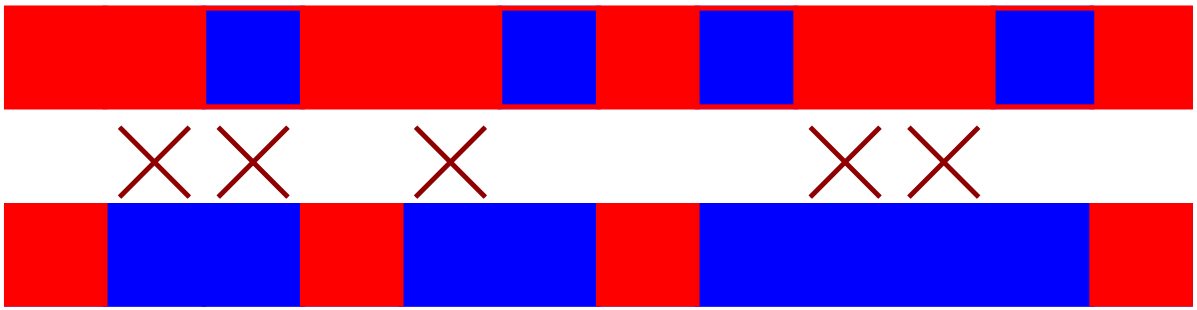
Received: $a\ x\ c\ d \mid a\ b\ z\ d \mid a\ b\ c\ d$.

a	x	c	d	a	b	z	d	a	b	c	d	
a	x	c	d	a	x	c	d	a	x	c	d	3
a	b	z	d	a	b	z	d	a	b	z	d	3
a	b	c	d	a	b	c	d	a	b	c	d	2
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	≥ 3

Hamming Distance

Hamming distance between two vectors of equal dimension is number of positions at which they differ.

$[n, k, d]_q$-code



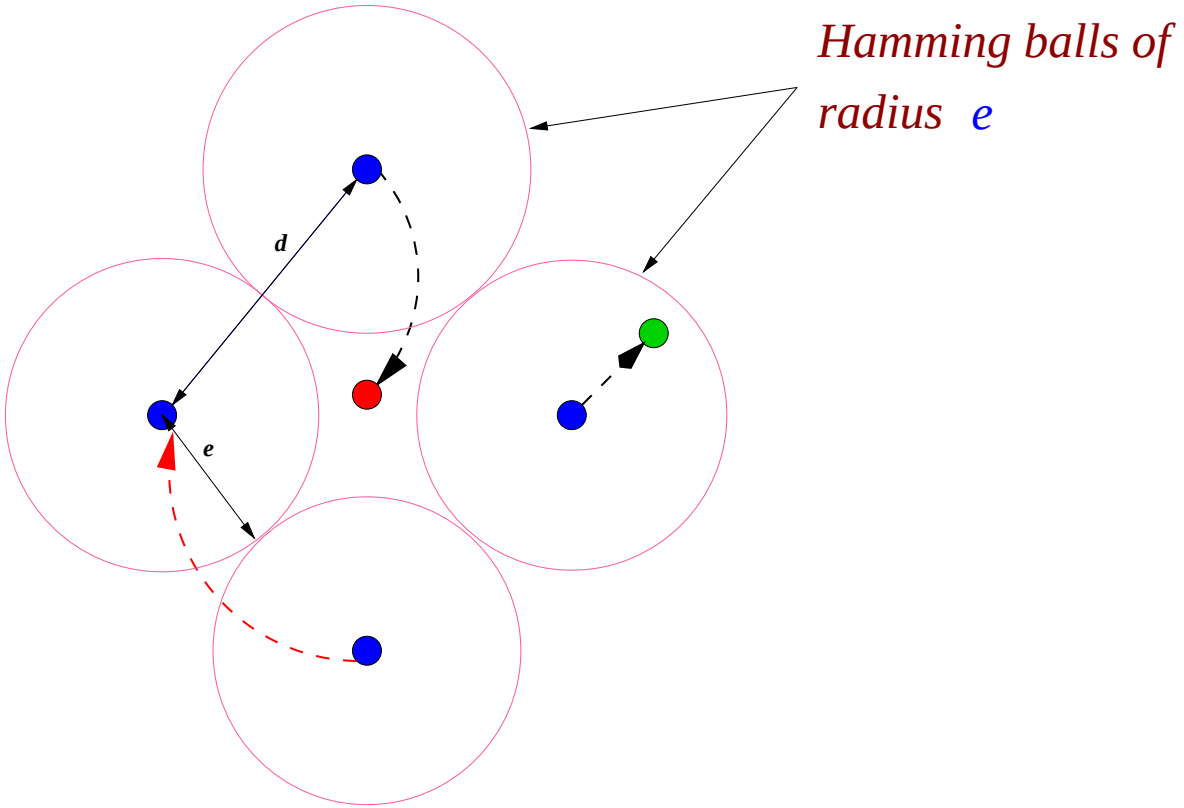
Maximum Likelihood Decoding

Given received word, find a codeword that has least Hamming distance to it.

Intractable in general.

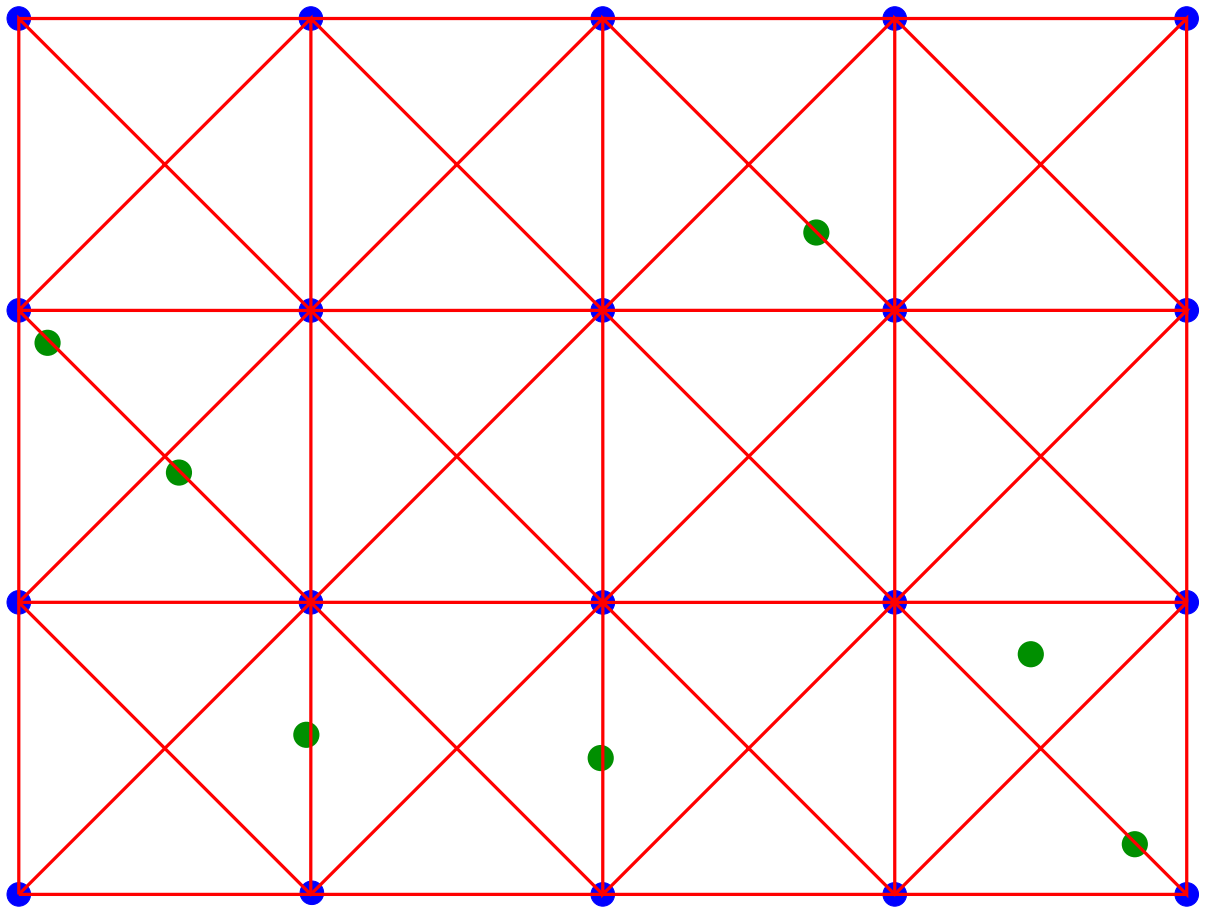
Worst Case Error Correction

$[n, k, d]_q$-code is capable of correcting up to $e = (d-1)/2$ errors in whatever locations!

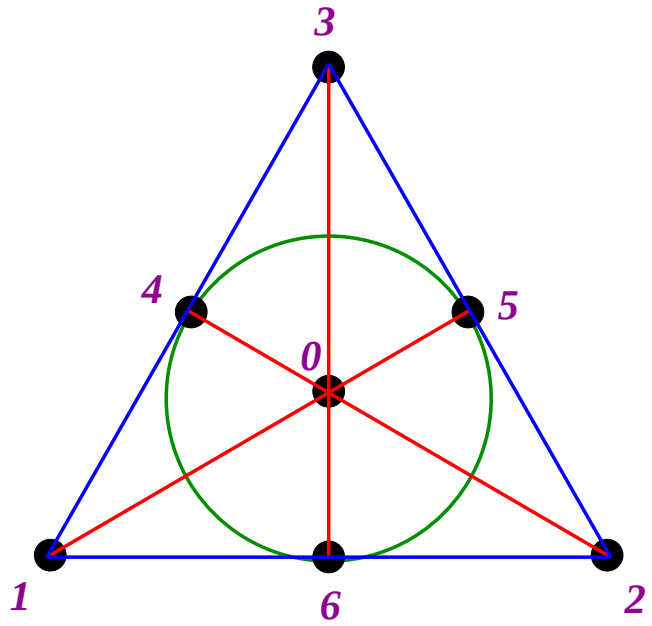


Errors in Known Locations: Erasures

$[n, k, d]_q$-code is capable of correcting up to $d - 1$ errors in whatever locations if the locations are known.



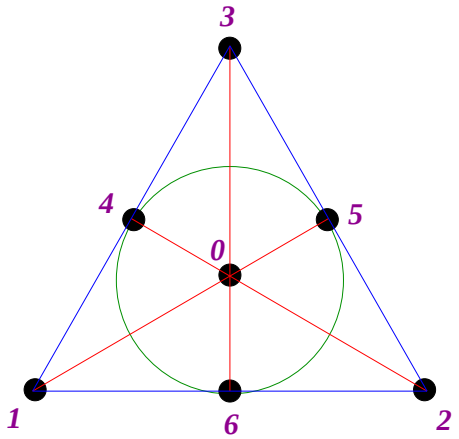
Projective plane over $GF(2)$



0	0	0	0	1	1	1	0
1	1	0	1	0	1	0	0
2	1	1	0	1	0	0	0
3	0	1	1	0	0	1	0
4	0	0	1	1	0	0	1
5	0	1	0	0	1	0	1
6	1	0	0	0	0	1	1

Minimum distance = 4

Hamming Code



$[7,4,3]_2$-code

0	0	0	1	1	1	0	0
1	0	1	0	1	0	0	0
1	1	0	1	0	0	0	0
0	1	1	0	0	1	0	0
0	0	1	1	0	0	1	0
0	1	0	0	1	0	1	0
1	0	0	0	0	1	1	0

0	0	0	0	0	1	1	1
1	1	0	1	1	0	1	1
0	1	1	1	0	1	0	1
1	0	1	0	1	0	0	1
0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	1
1	0	1	0	1	1	1	1

A Solution to the Football Match Problem

$[4, 2, 3]_3$-Hamming Code with generator matrix:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ 1 & 2 \end{pmatrix}$$

Codewords:

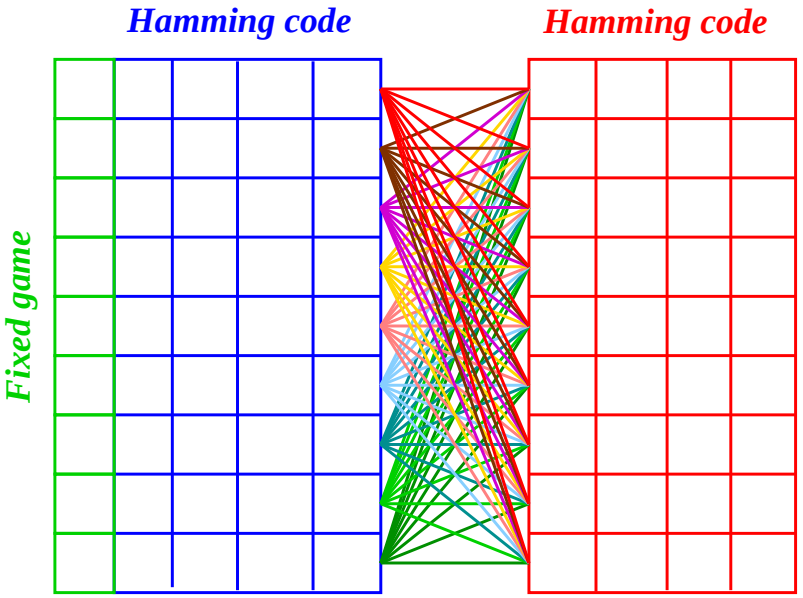
$$\begin{array}{ccccccccc} 0 & 0 & 1 & 2 & 1 & 2 & 2 & 1 & 0 \\ 2 & 1 & 0 & 0 & 1 & 2 & 1 & 2 & 0 \\ 2 & 1 & 1 & 2 & 2 & 1 & 0 & 0 & 0 \\ 2 & 1 & 2 & 1 & 0 & 0 & 2 & 1 & 0 \end{array}$$

A Solution to the Football Match Problem

This Hamming code is perfect: Hamming balls of radius 1 fill the space $GF(3)^4$:

$$3^2 \cdot \sum_{i=0}^{1} \binom{4}{i} 2^i = 9 \cdot (1 + 4 \cdot 2) = 81.$$

Any vector in $GF(3)^4$ has Hamming distance at most 1 to a codeword.



Bounds

How can we prove optimality of codes? (Fix any two of the three parameters n, k, d and maximize the third.)

1. Hamming bound: $\sum_{i=1}^{e} \binom{n}{i}(q-1)^i \leq q^{n-k}$.
   Equality: perfect codes
2. Plotkin bound: $d + k \leq n + 1$.
   Equality: MDS codes.
3. Other more refined bounds....

Perfect Codes

Have been completely classified by van Lint and Tieätväinen.

Essentially: Hamming codes, Golay codes.

Not desirable in communication scenarios.

MDS Codes

Not classified completely.

Open problem: Given dimension k and field size q, determine maximum block length of an MDS-code. (Conjecturally $q+1$ or $q+2$.)

MDS codes are desirable in practice in worst case scenarios if efficient encoding and decoding available.

Prototype: Reed-Solomon codes.

Reed-Solomon Codes: Applications

1. Satellite Communication,
2. Hard disks,
3. Compact Discs, Digital Versatile Disks, Digital Audio Tapes,
4. Wireless Communication,
5. Secret sharing,
6. ...

Reed-Solomon Codes: Definitions

Choose n different elements $x_1, \ldots, x_n$ in $GF(q)$.

Reed-Solomon code is image of the morphism

$$\begin{aligned} GF(q)[x]_{<k} &\mapsto GF(q)^n \\ f &\mapsto (f(x_1), \ldots, f(x_n)) \end{aligned}$$

Block length $= n$.

Dimension? Minimum distance?

Reed-Solomon Codes: Parameters

Theorem. Nonzero polynomial of degree m over a field can have at most n zeros over that field.

Reed-Solomon Codes: Dimension

$$\begin{aligned} GF(q)[x]_{<k} &\mapsto GF(q)^n \\ f &\mapsto (f(x_1), \ldots, f(x_n)) \end{aligned}$$

Kernel: 0 if $k \leq n$ since nonzero polynomial of degree $\leq k-1$ has at most $k-1$ zeros.

Dimension: k (if $k \leq n$).

Reed-Solomon Codes: Minimum Distance

$$\begin{aligned} GF(q)[x]_{<k} &\mapsto GF(q)^n \\ f &\mapsto (f(x_1), \ldots, f(x_n)) \end{aligned}$$

Minimum distance: Maximal number of zeros in a nonzero codeword is $k-1$, since evaluating polynomial of degree $\leq k-1$.

Minimum distance is $\geq n-k+1$, hence equal, hence MDS code!.

Reed-Solomon Codes: Encoding

$f \mapsto (f(x_1), \ldots, f(x_n))$ is easy to compute!

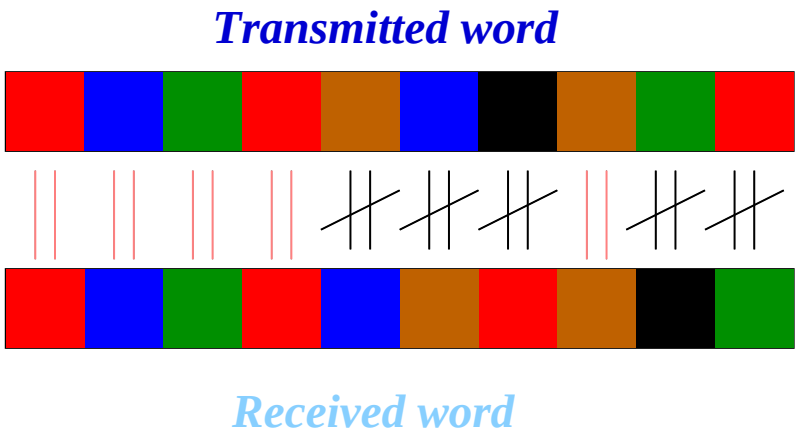
$O(n^2)$ using naive algorithm.

$O(n \log^2(n) \log\log(n)$ using fast algorithms.

Reed-Solomon Codes: Decoding

No efficient maximum likelihood decoding known.

Concentrate on bounded distance decoding.



Number of agreements $\geq (n+k)/2$.

Number of disagreements $\leq (n-k)/2$.

Welch-Berlekamp Algorithm

Transmitted word: $(f(x_1), \ldots, f(x_n))$.

Received word: $(y_1, \ldots, y_n)$.

Number of agreements $\geq (n+k)/2$.

Find f!

Welch-Berlekamp Algorithm

Step 1: Find $g(x) \in GF(q)[x]_{<(n+k)/2}$ and
$h(x) \in GF(q)[x]_{\leq(n-k)/2}$, not both zero, such that

$$\forall\, i = 1, \ldots, n: \quad g(x_i) + y_i h(x_i) = 0.$$

(Solving a system of equations!)

Step 2: Then $f = g/h$!

Welch-Berlekamp Algorithm: Proof

$H(x) := g(x) - f(x)h(x)$.

Degree of $H(x) < (n+k)/2$.

If $y_i = f(x_i)$, then $H(x_i) = 0$.

$H(x)$ has at least $(n+k)/2$ zeros.

$H(x)$ is zero.

$f(x) = g(x)/h(x)$.

Welch-Berlekamp Algorithm: Running time

Step 1: Solving a homogeneous $n \times (n+1)$ system of equations; $O(n^3)$.

Can be reduced to $O(n^2)$ (Welch-Berlekamp, 1983; displacement method (Olshevsky-Shokrollahi, 1999)).

Step 2: Polynomial division; $O(n^2)$.

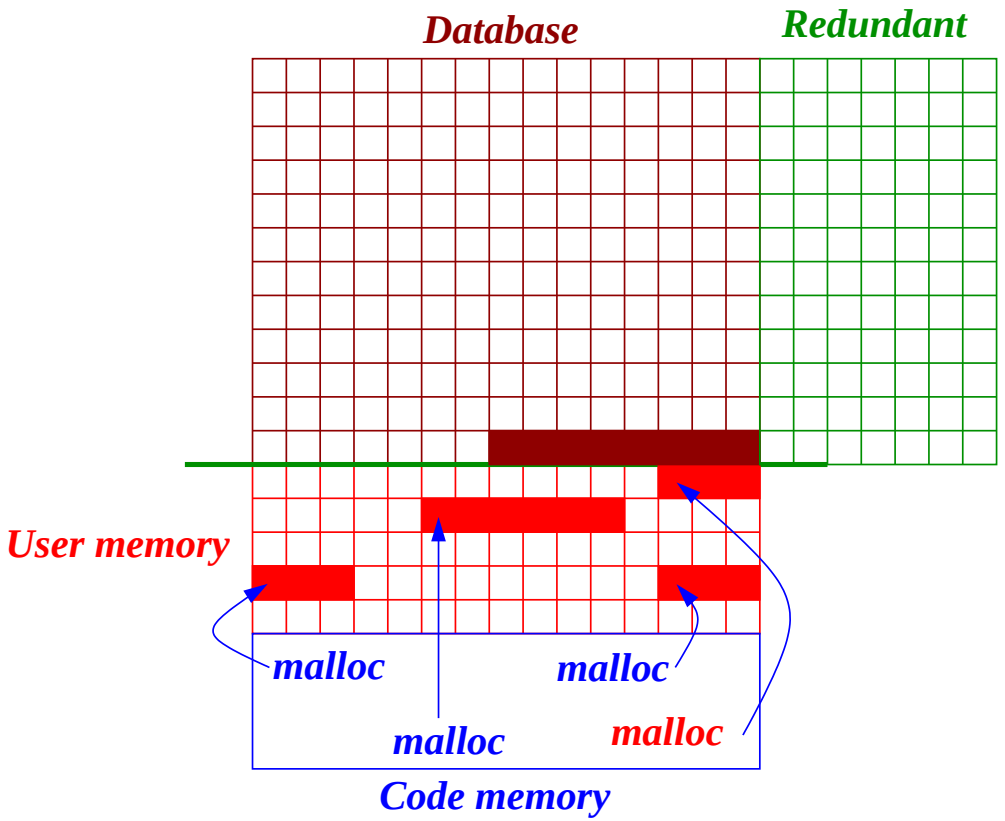
Welch-Berlekamp Algorithm: Generalization

Has been generalized to more than $(n-k)/2$ errors (list-decoding, Sudan, 1997, Guruswami-Sudan, 1999).

Step 2 requires factorization of bivariate polynomials.

Can be done more efficiently (Gao-Shokrollahi 1999, Olshevsky-Shokrollahi 1999).

A Solution to the In-Memory Database Problem



Reed-Solomon Codes: Generalization

Disadvantage of RS-codes: $GF(q)$ must be large to accommodate many points, so long codes impossible.

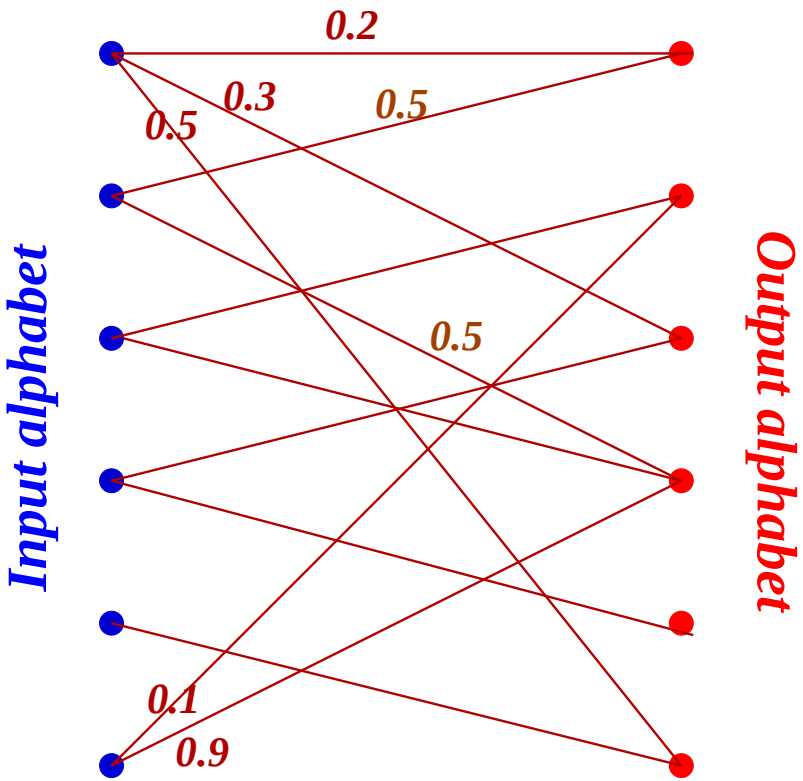
Interpret $GF(q)$ as affine line over itself, and generalize to more complicated algebraic curves.

Lead to best known codes in terms of minimum distance, dimension, block-length.

Above algorithms can be generalized to these Algebraic-geometric codes.

Probabilistic Methods

Channels


0.2
0.3
0.5
0.5
0.5
0.1
0.9
Input alphabet
Output alphabet
Transition probabilities

Entropy and Mutual Information

X and Y discrete random variables on alphabets $\mathcal{X}$ and $\mathcal{Y}$ and distributions $p(x)$ and $q(x)$. $p(x,y)$ their joint distribution.

Entropy $H(X)$ of X

$$H(X) = -\sum_{x\in\mathcal{X}} p(x)\log p(x).$$

Mutual information $I(X;Y)$

$$I(X;Y) = -\sum_{y\in\mathcal{Y}}\sum_{x\in\mathcal{X}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}.$$

Entropy and Mutual Information

- $H(X)$ is the amount of *uncertainty* of random variable X.
- $I(X;Y)$ is the reduction in the uncertainty of X due to the knowledge of Y.

Capacity

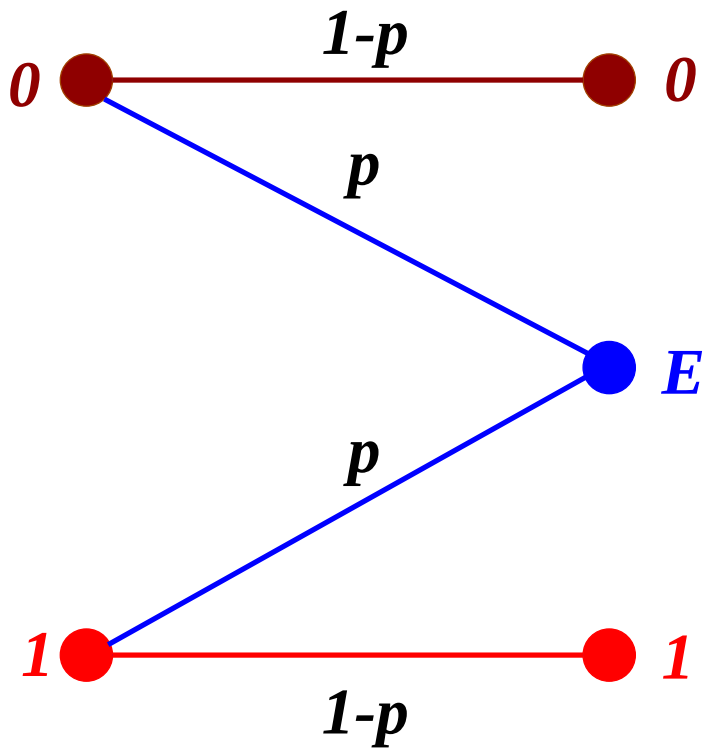
Capacity of a channel with input alphabet $\mathcal{X}$ and output alphabet $\mathcal{Y}$ and probability transition matrix $p(y \mid x)$ is

$$C = \max_{p(x)} I(X;Y),$$

where maximum is over all possible input distributions $p(x)$.

Examples of Capacity: BEC

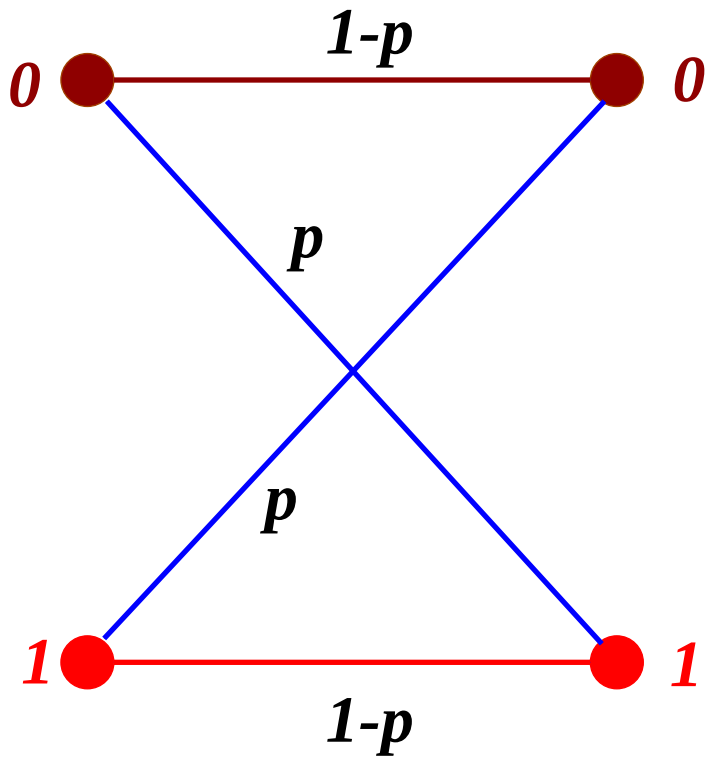
Binary Erasure Channel:



Capacity $= 1 - p$

Examples of Capacity: BSC

Binary Symmetric Channel:



Capacity $= 1 + p\log_2(p) + (1-p)\log_2(1-p)$

Capacity and Communication

Shannon's Coding Theorem, 1948:

$\mathcal{C}$ channel with capacity C. For any rate $R \leq C$ there exists a sequence of codes of rate R such that the probability of error of the Maximum Likelihood Decoding for these codes approaches zero as the block-length approaches infinity.

The condition $R \leq C$ is necessary and sufficient.

Problems

- How to find the sequences of codes?

  (Random codes, Concatenated codes, $\cdots$)

- How to decode efficiently?

  Has been open for almost 50 years.

  Low-Density Parity-Check Codes

Part 2:
Low-Density Parity-Check Codes

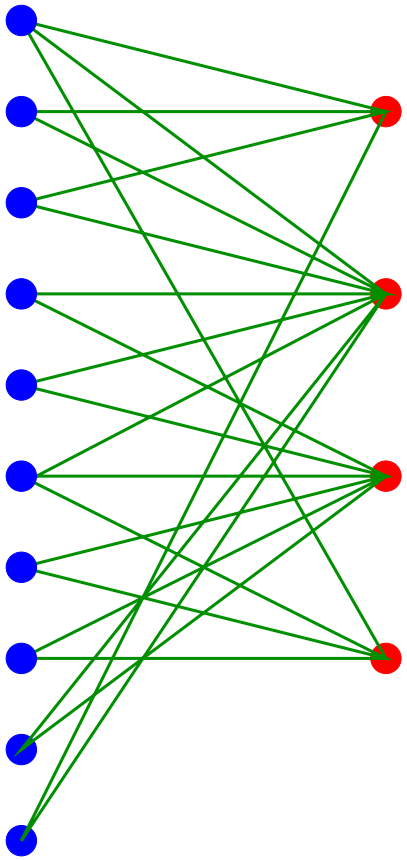
Low-Density Parity Check Codes

Gallager	1963
Zyablov	1971
Zyablov-Pinsker	1976
Tanner	1981
Turbo Codes	1993
Berroux-Glavieux-Thitimajshima	

Sipser-Spielman, Spielman	1995
MacKay-Neal, MacKay	1995
Luby-Mitzenmacher-S-Spielman-Stemann	1997
Luby-Mitzenmacher-S-Spielman	1998
Richardson-Urbanke	1999
Richardson-Shokrollahi-Urbanke	1999

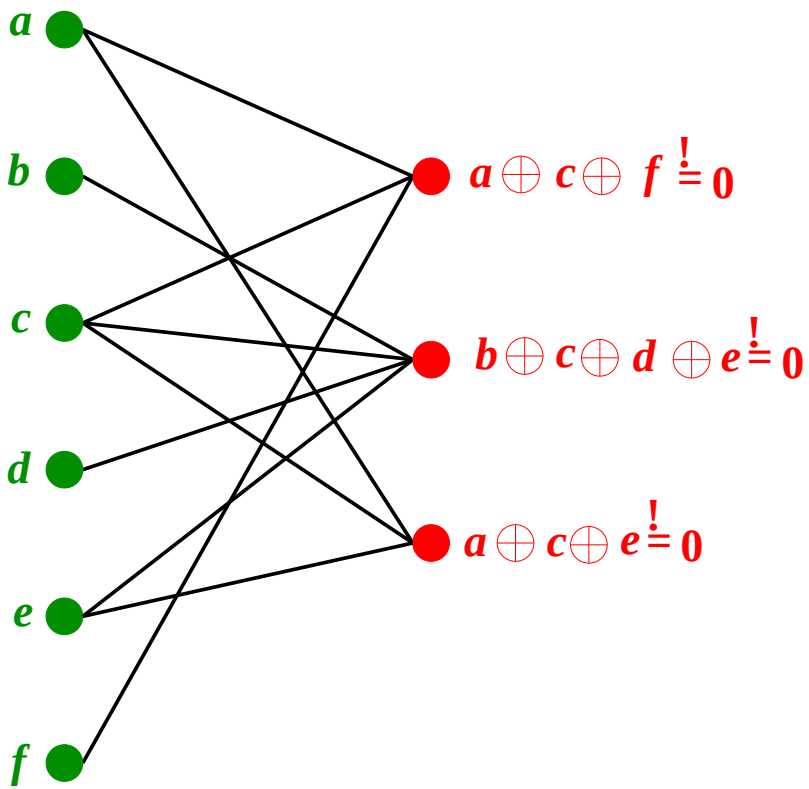
Code Construction

Codes are constructed from sparse bipartite graphs.



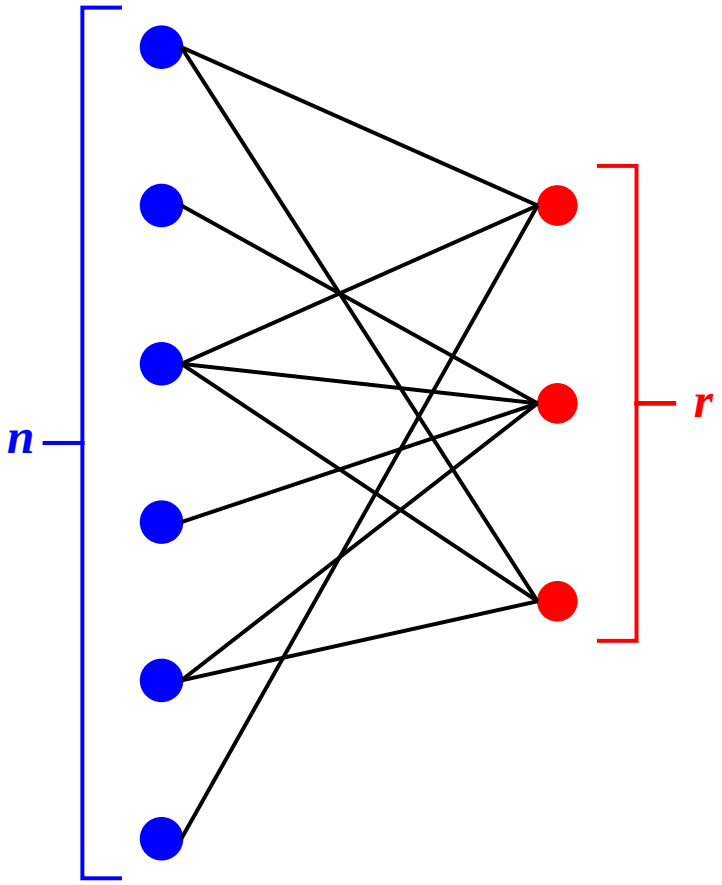
Code Construction

Any binary linear code has a graphical representation.



Not any code can be represented by a sparse graph.

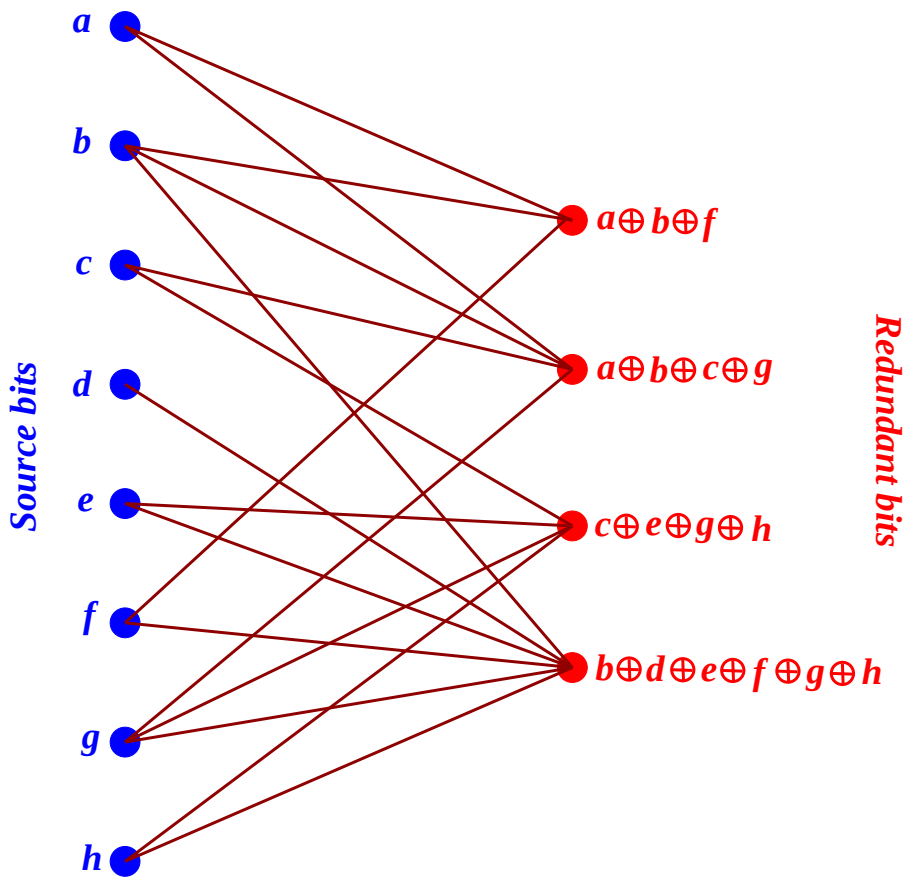
Parameters



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

$$Rate \geq 1 - \frac{\text{average left degree}}{\text{average right degree}}$$

Dual Construction

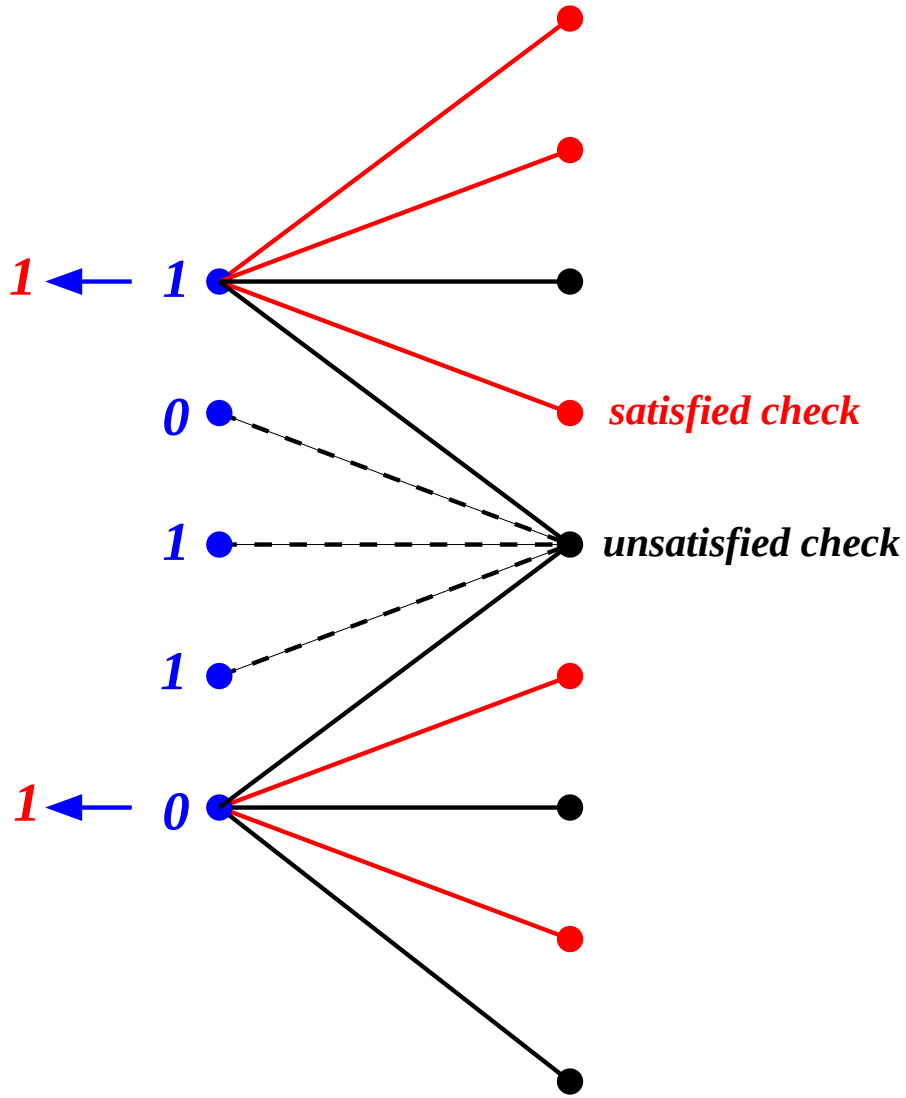


Encoding time is proportional to number of edges.

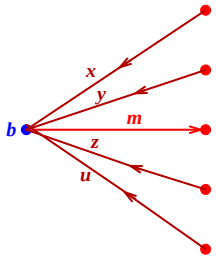
Algorithmic Issues

- Encoding?
  - Is linear time for the dual construction
  - Is quadratic time (after preprocessing) for the Gallager construction. More later!
- Decoding?
  - Depends on the channel,
  - Depends on the fraction of errors.

Decoding on a BSC: Flipping

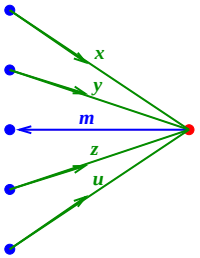


Decoding on a BSC: Gallager Algorithm A (Message passing)



$$m = \begin{cases} x & \text{if } x=y=z=u \\ b & \text{else} \end{cases}$$

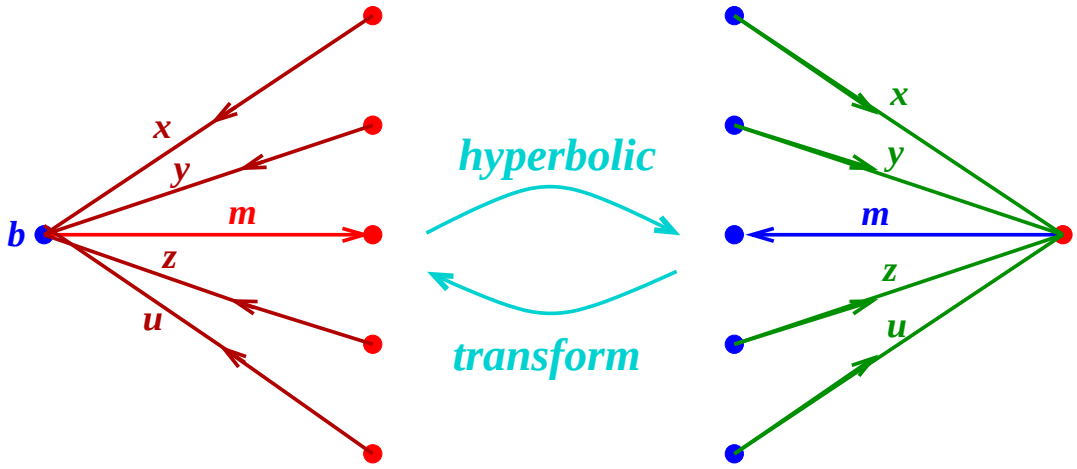
MESSAGE



$m = x \oplus y \oplus z \oplus u$

CHECK

Decoding on a BSC: Belief Propagation



$m = x+y+z+u+b$

$m = x * y * z * u$

$(a,b)*(c,d) := (a+c,\ b+d \bmod 2)$

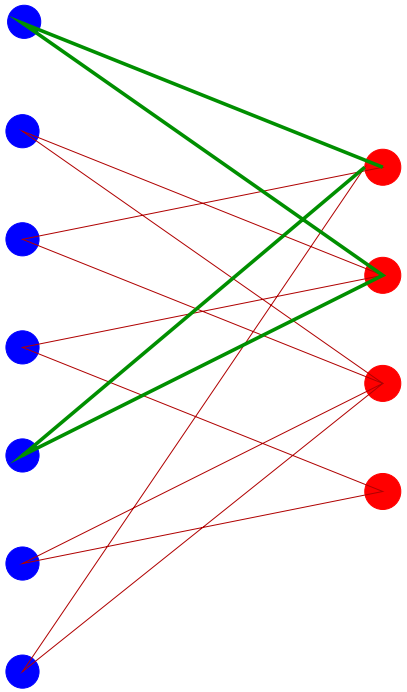
MESSAGE

CHECK

Messages in log-likelihood ratios.

Optimality of Belief Propagation

Belief propagation is bit-optimal if graph has no loops.



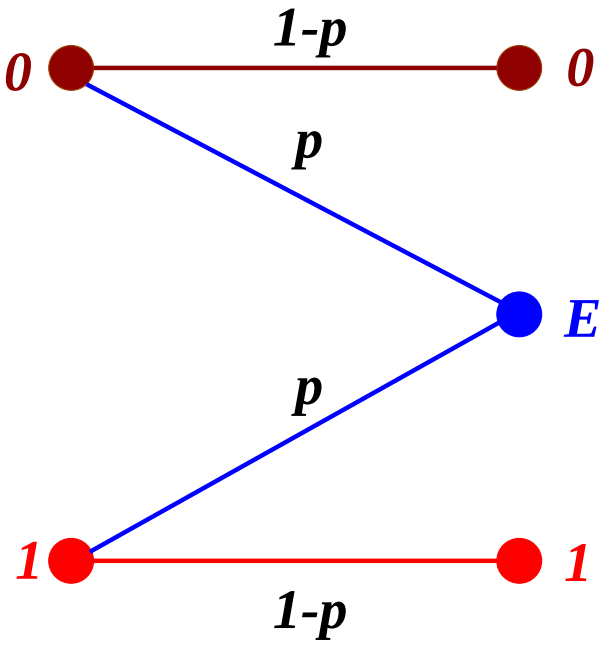
Maximizes the probability

$$P(c_m = b \mid y) = \sum_{c \in C} P(c \mid y).$$

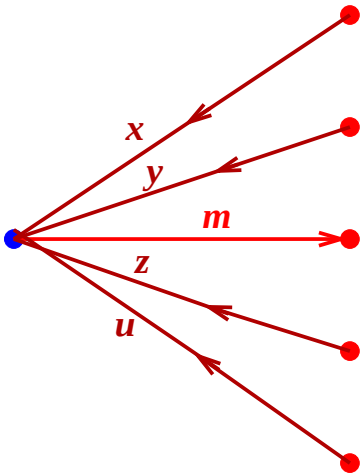
Performance on a (3,6)-graph

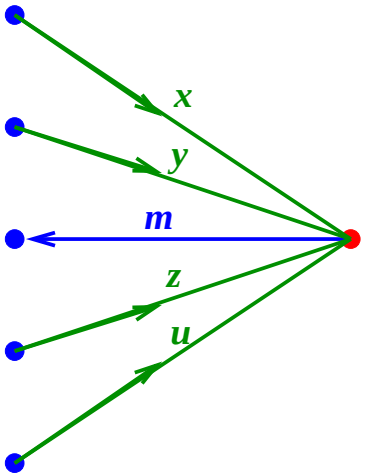
Shannon limit:	11%	
Flipping algorithm:	1%?	
Gallager A:	4%	
Gallager B:	4%	(6.27%)
Erasure decoder:	7%	
Belief propagation:	8.7%	(10.8%)

The Binary Erasure Channel (BEC)



Decoding on a BEC: Luby-Mitzenmacher-Shokrollahi-Spielman-Stemann





$$m = \begin{cases} 1 & \text{if } x \vee y \vee z \vee u = 1 \\ 0 & \text{else} \end{cases}$$

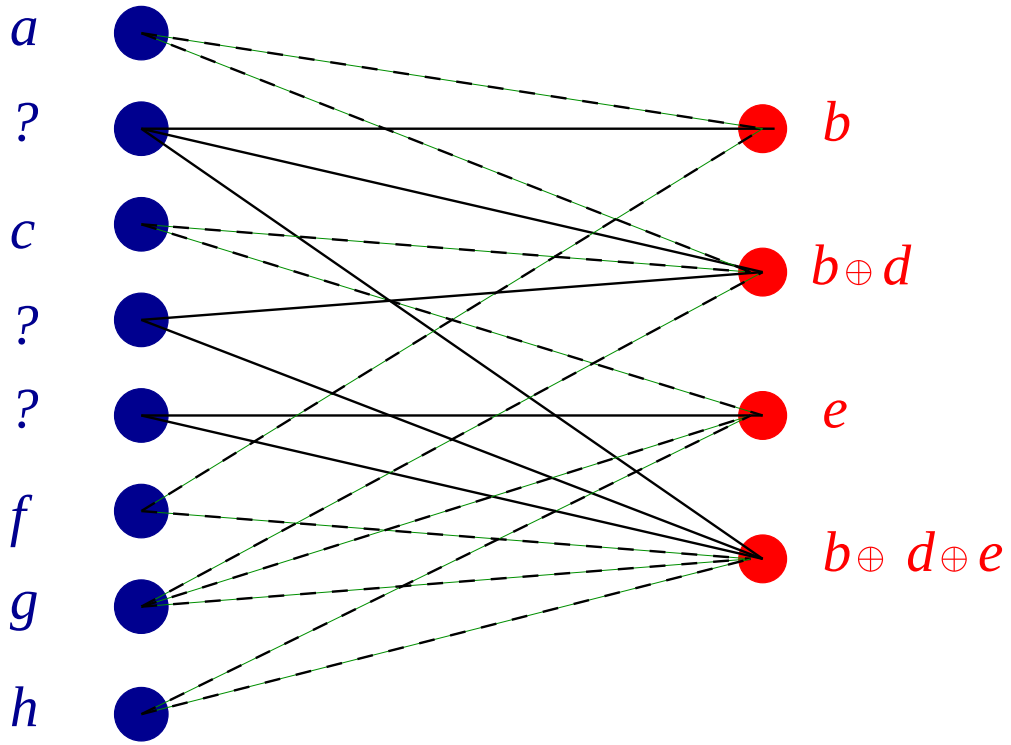
$$m = \begin{cases} 1 & \text{if } x = y = z = u = 1 \\ 0 & \text{else} \end{cases}$$

MESSAGE

CHECK

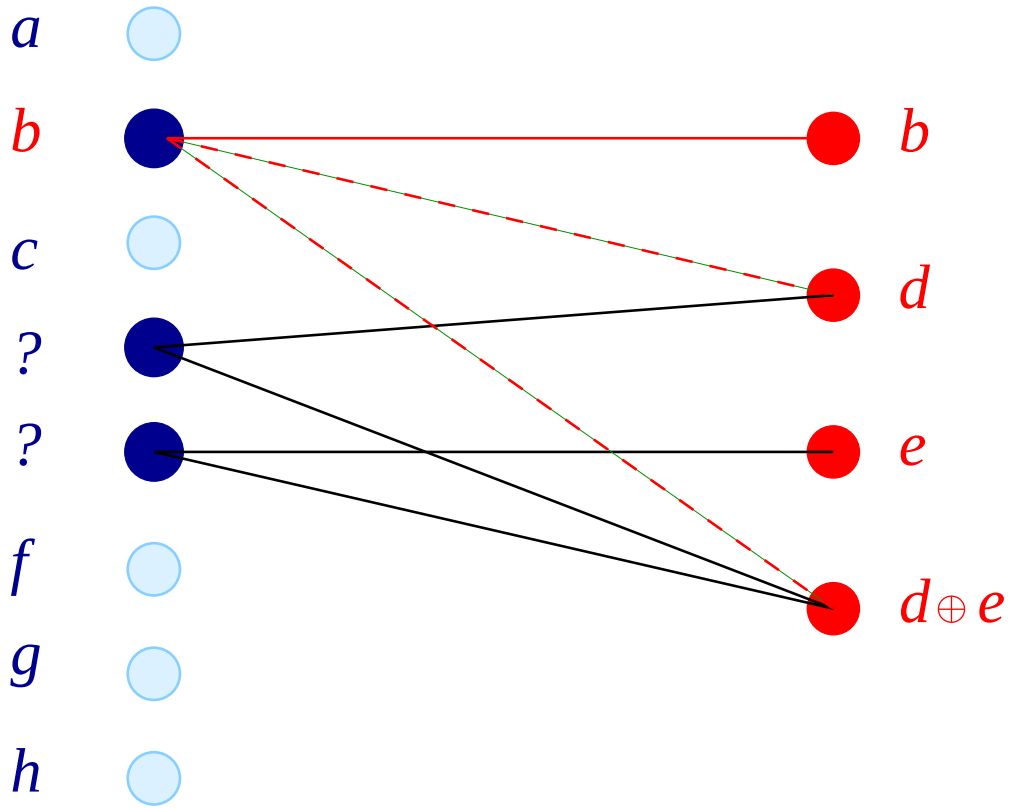
Decoding on a BEC

Phase 1: Direct recovery

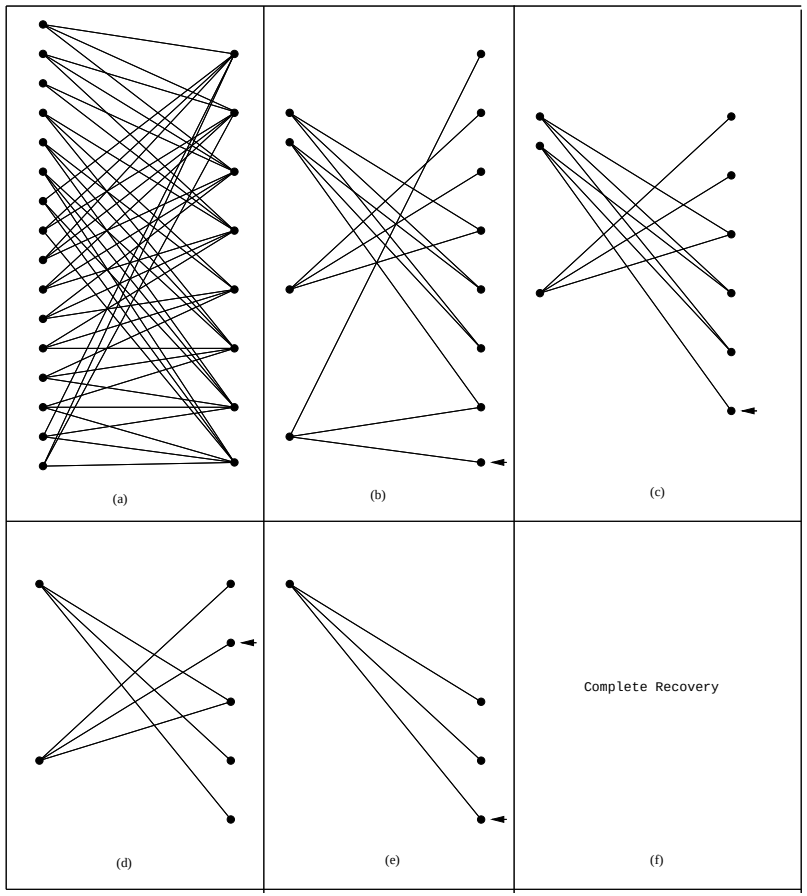


Decoding on a BEC

Phase 2: Substitution



Example



The (inverse) problem

Have: fast decoding algorithms.

Want: design codes that can correct many errors using these algorithms.

Focus on the BEC in the following.

Experiments

Choose regular graphs.

An (d, k)-regular graph has rate at least $1 - d/k$. Can correct at most an d/k-fraction of erasures.

Choose a random (d, k)-graph.

p_0 := maximum fraction of erasures the algorithm can correct.

d	k	d/k	p_0
3	6	0.5	0.429
4	8	0.5	0.383
5	10	0.5	0.341
3	9	0.33	0.282
4	12	0.33	0.2572

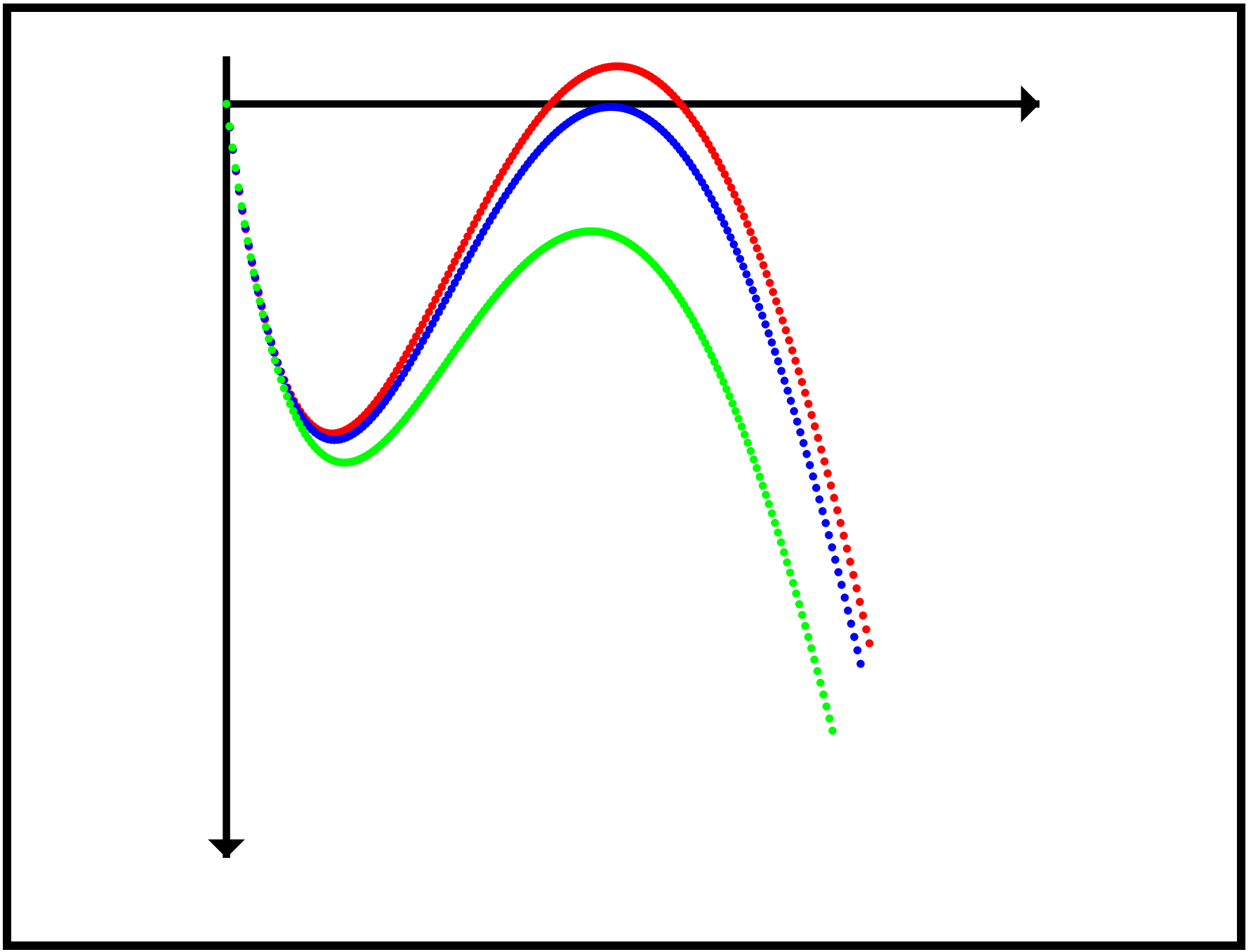
What are these numbers?

A Theorem

Luby, Mitzenmacher, Shokrollahi, Spielman, Stemann, 1997:

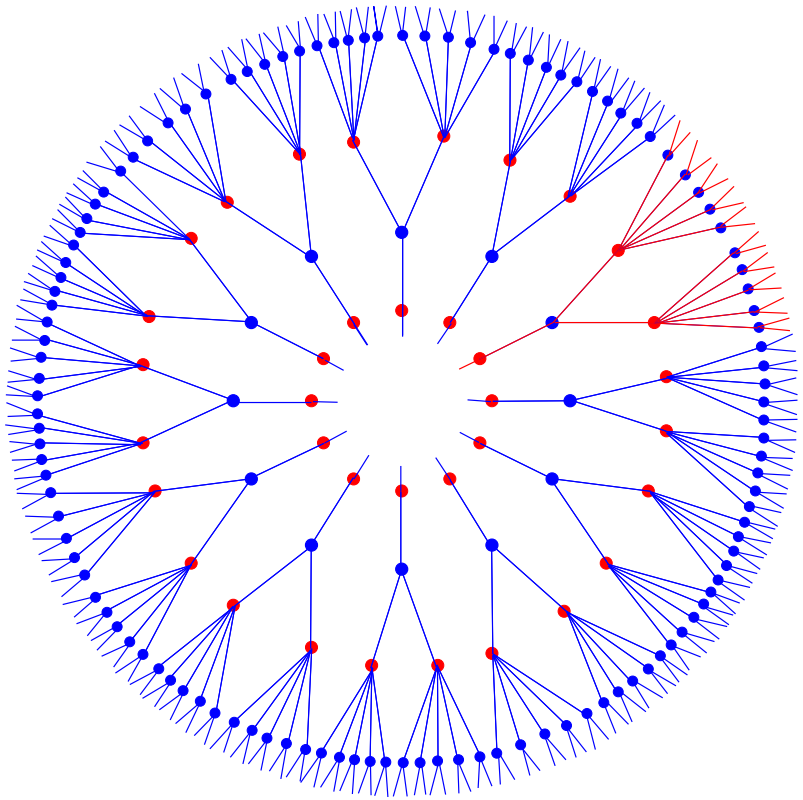
A randomly chosen (d, k)-graph can correct a p_0-fraction of erasures with high probability if and only if

$$p_0 \cdot (1 - (1 - x)^{k-1})^{d-1} < x \quad \text{for } x \in (0, p_0).$$



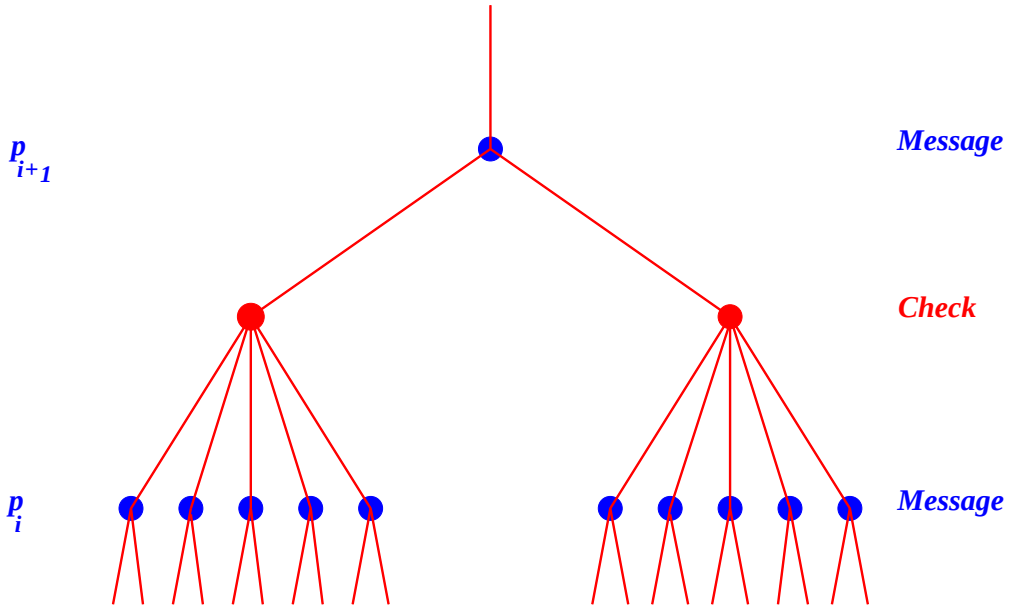
Analysis: $(3, 6)$-graphs

Expand neighborhoods of message nodes.



Analysis: $(3, 6)$-graphs

p_i probability that message node is still erased after ith iteration.



$$p_{i+1} = p_0(1 - (1 - p_i)^5)^2.$$

Successful Decoding

Condition:

$$p_0(1-(1-p_i)^5)^2 < p_i$$

Analysis: $(3, 6)$-graphs

Making arguments exact:

- Neighborhood is tree-like: high probability, standard argument.
- Above argument works for expected fraction of erasures at ℓth round.

  Real value is sharply concentrated around expected value p_ℓ: Edge exposure martingale, Azuma's inequality.

The General Case

Let λ_i and ρ_i be the fraction of edges of degree i on the left and the right hand side, respectively.

Let $\lambda(x) := \sum_i \lambda_i x^{i-1}$ and $\rho(x) := \sum_i \rho_i x^{i-1}$.

Condition for successful decoding for erasure probability p_0 is then

$$p_0 \lambda\left(1 - \rho(1-x)\right) < x$$

for all $x \in (0, p_0)$.

Belief propagation

Richardson-Urbanke, 1999:

f_ℓ: density of the probability distribution of the messages passed from the check nodes to the message nodes at round ℓ of the algorithm.

P_0: density of the error distribution (in log-likelihood representation).

Consider (d, k) regular graph.

$$\Gamma\left(f_{\ell+1}\right) = \left(\Gamma\left(P_0 \otimes {f_\ell}^{\otimes(k-1)}\right)\right)^{\otimes(d-1)},$$

where Γ is a hyperbolic change of measure function,

$$\Gamma(f)(y) := f(\ln \coth y/2)/\sinh(y),$$

and $\otimes$ denotes convolution.

We want f_ℓ to converge to a Delta function at ∞.

Gives rise to high-dimensional optimization algorithms.

Achieving capacity

Want to design codes that can recover from a fraction of $1 - R$ of erasures (asymptotically).

Want to have λ and ρ so that

$$p_0 \lambda(1 - \rho(1 - x)) < x$$

for all $x \in (0, p_0)$, and p_0 arbitrarily close to

$$1 - R = \frac{\int_0^1 \rho(x)\mathrm{d}x}{\int_0^1 \lambda(x)\mathrm{d}x}.$$

Tornado codes

Extremely irregular graphs provide for any rate R sequences of codes which come arbitrarily close to the capacity of the erasure channel!

Degree structure?

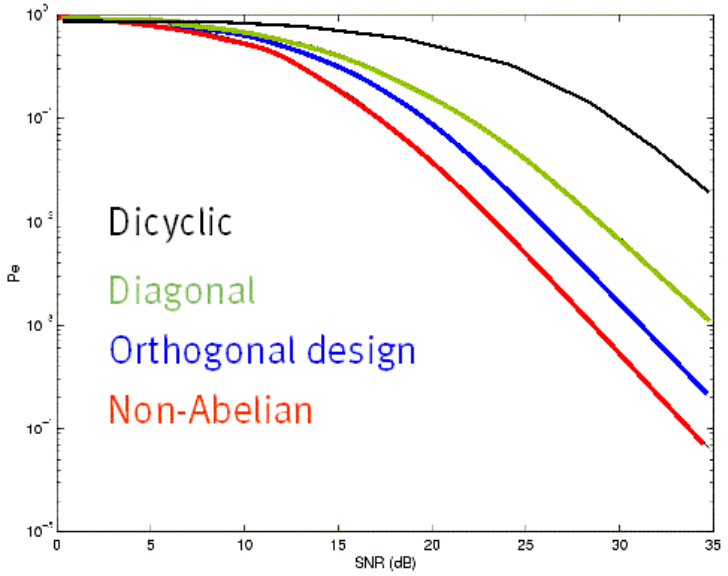
Choose design parameter D.

$$\lambda(x) := \frac{1}{H(D)}\left(x + \frac{x^2}{2} + \cdots + \frac{x^D}{D}\right)$$

$$\rho(x) := \exp\left(\mu(x-1)\right),$$

where $H(D) = 1 + 1/2 + \cdots + 1/D$ and $\mu = H(D)/\left(1 - 1/(D+1)\right)$.

Tornado Codes: Left Degree Distribution



Right regular codes

Shokrollahi, 1999:

Graphs that are regular on the right.

Degrees on the left are related to the Taylor expansion of

$$(1-x)^{1/m}.$$

These are the only known examples of LDPC codes that achieve capacity on a nontrivial channel using a linear time decoding algorithm.

Other channels?

f density function.

$\lambda(f) := \sum_i \lambda_i f^{\otimes(i-1)}$.

$\rho(f) := \sum_i \rho_i f^{\otimes(i-1)}$.

$$\Gamma\left(f_{\ell+1}\right) = \rho\left(\Gamma\left(P_0 \otimes \lambda(f_\ell)\right)\right).$$

Want P_0 such that $f_\ell \to \Delta_\infty$.

Conditions on the density functions

Richardson-Shokrollahi-Urbanke, 1999:

- Consistency: if the channel is *"symmetric"*, then the density functions f_ℓ satisfy $f(x) = f(-x)\mathrm{e}^x$.
- Fixed point theorem: If $P_{\mathrm{err}}(f_i) = P_{\mathrm{err}}(f_j)$ for $i < j$, then $f_i = f_j$ is a fixed point of the iteration.

Conditions on the density functions

- Stability: let

$$r := -\lim_{n\to\infty} \frac{1}{n} \log P_{\text{err}}(P_0^{\otimes n}).$$

  Then for $\lambda_2 \rho'(1) > e^r$ we have $P_{\text{err}}(f_\ell) > \epsilon$ for some fixed ϵ and all ℓ.

  If $\lambda_2 \rho'(1) < e^r$, then the fixed point Δ_∞ is stable.

$$P_{\text{err}}(f) := \int_{-\infty}^{0} f(x) \mathrm{d}x$$

is the error probability.

Stability

- Erasure channel with erasure probability p_0:

$$\lambda_2 \rho'(1) \leq \frac{1}{p_0}.$$

- BSC channel: with probability p:

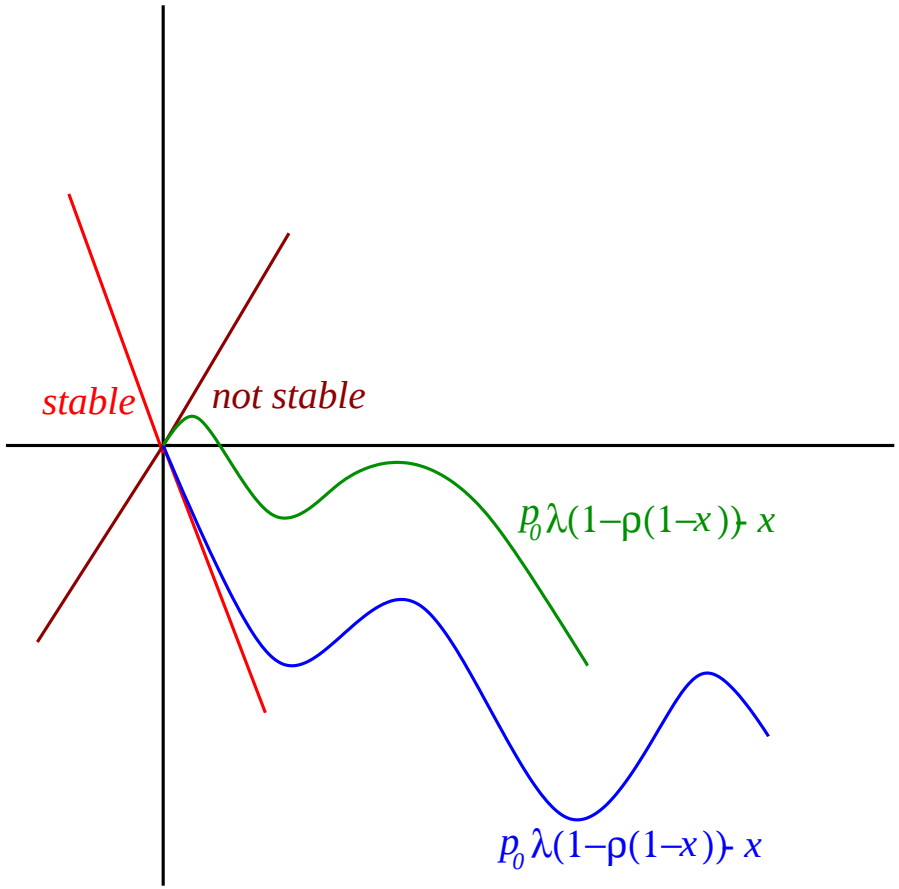
$$\lambda_2 \rho'(1) \leq \frac{1}{2\sqrt{p(1-p)}}.$$

- AWGN channel: with variance σ^2:

$$\lambda_2 \rho'(1) \leq \mathrm{e}^{-\frac{1}{2\sigma^2}}.$$

Stability for the Erasure Channel

Shokrollahi, 1999:



Flatness: Higher Stability Conditions

Shokrollahi, 2000:

$(\lambda_m(x), \rho_m(x))$ capacity achieving sequence of degree distributions.

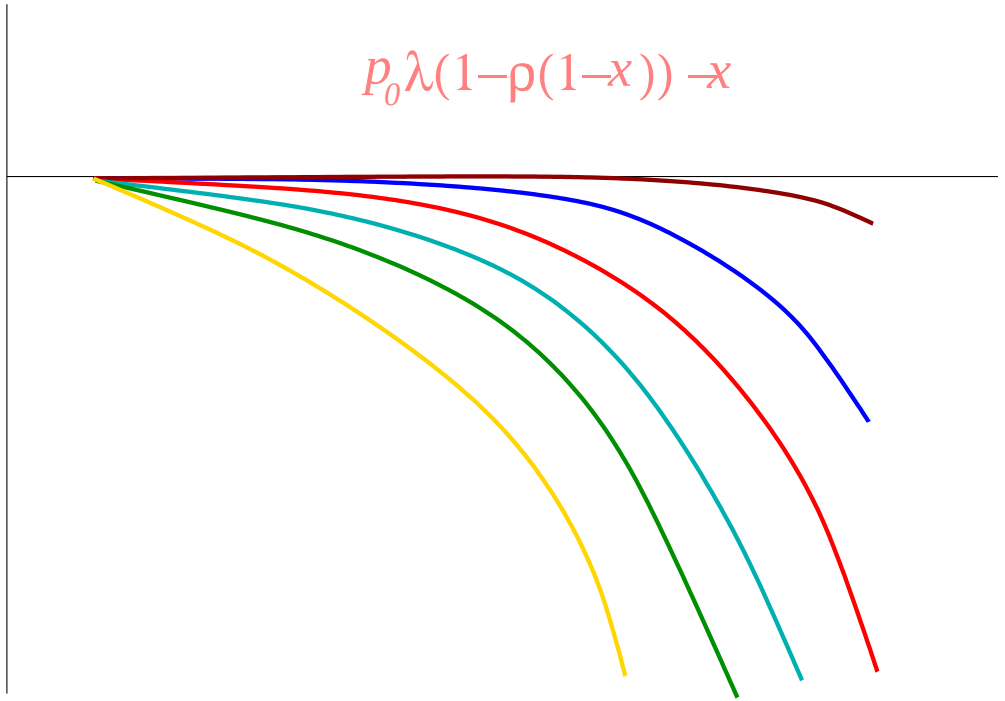
Then:

$$(1 - R)\lambda_m(1 - \rho_m(1 - x)) - x$$

converges uniformly to the zero-function on the interval $[0, 1 - R]$.

No equivalent known for other channels.

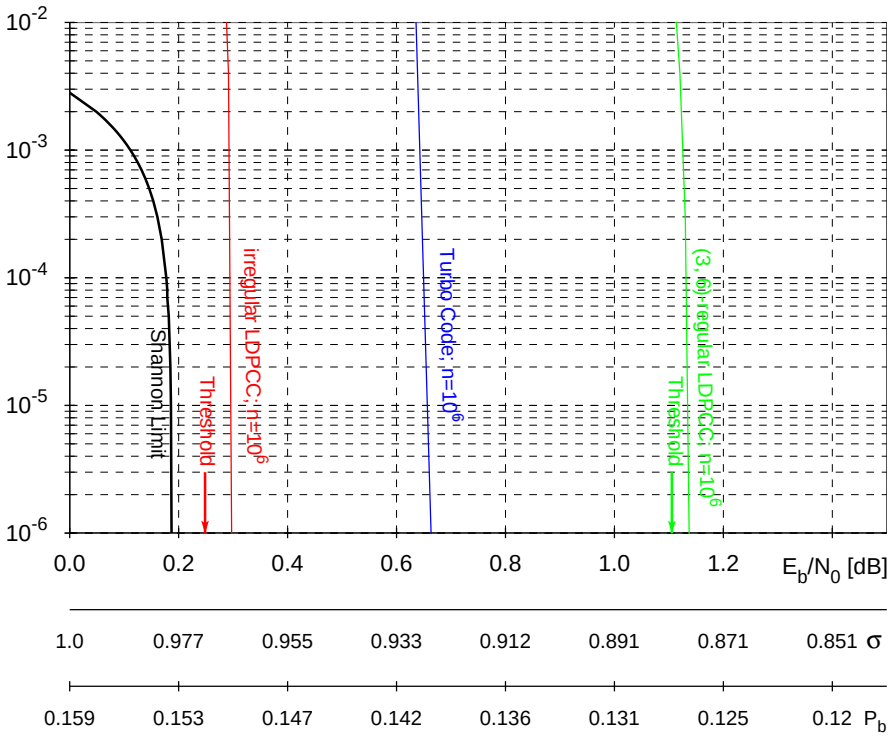
Flatness: Higher Stability Conditions



Capacity achieving

No sequences of c.a. degree distributions for channels other than the erasure channel known.

Conjecture: They exist!



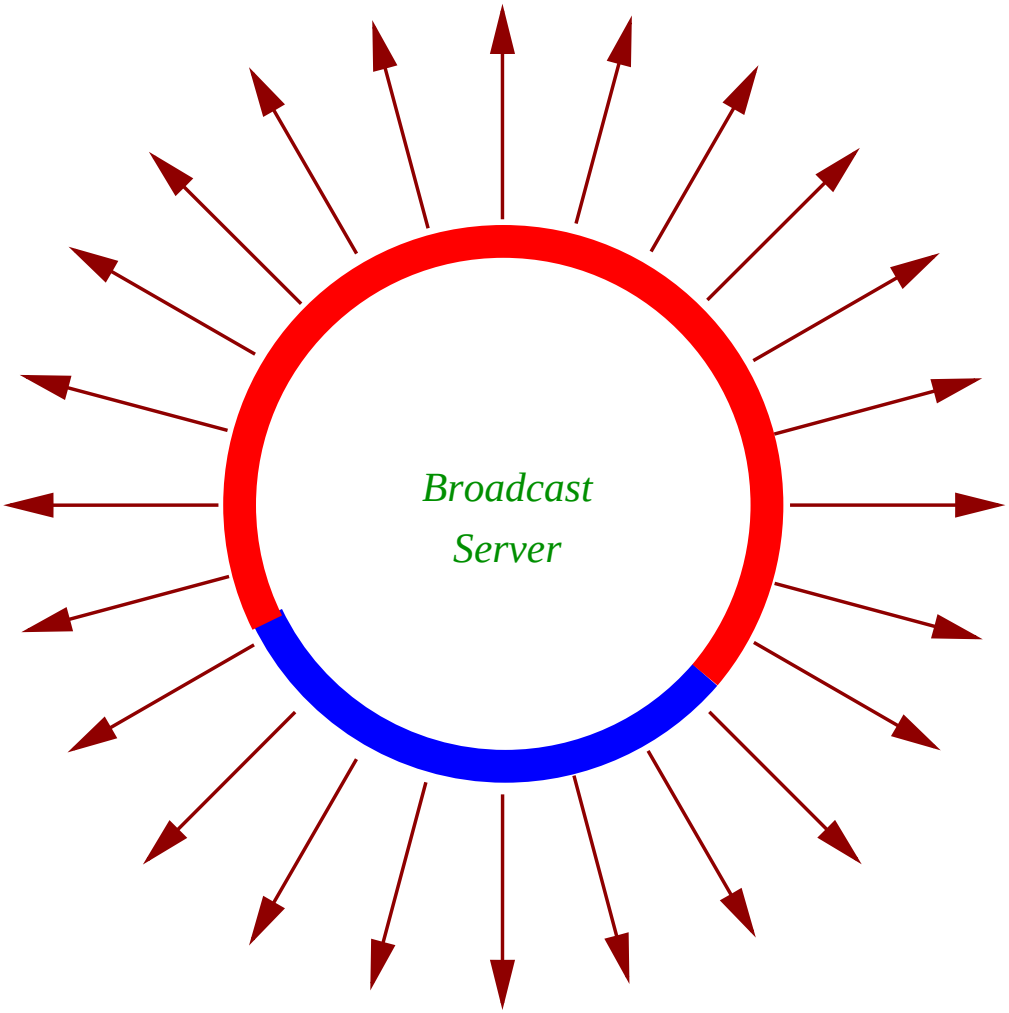
Applications to computer networks

Distribution of bulk data to a large number of clients.

Want

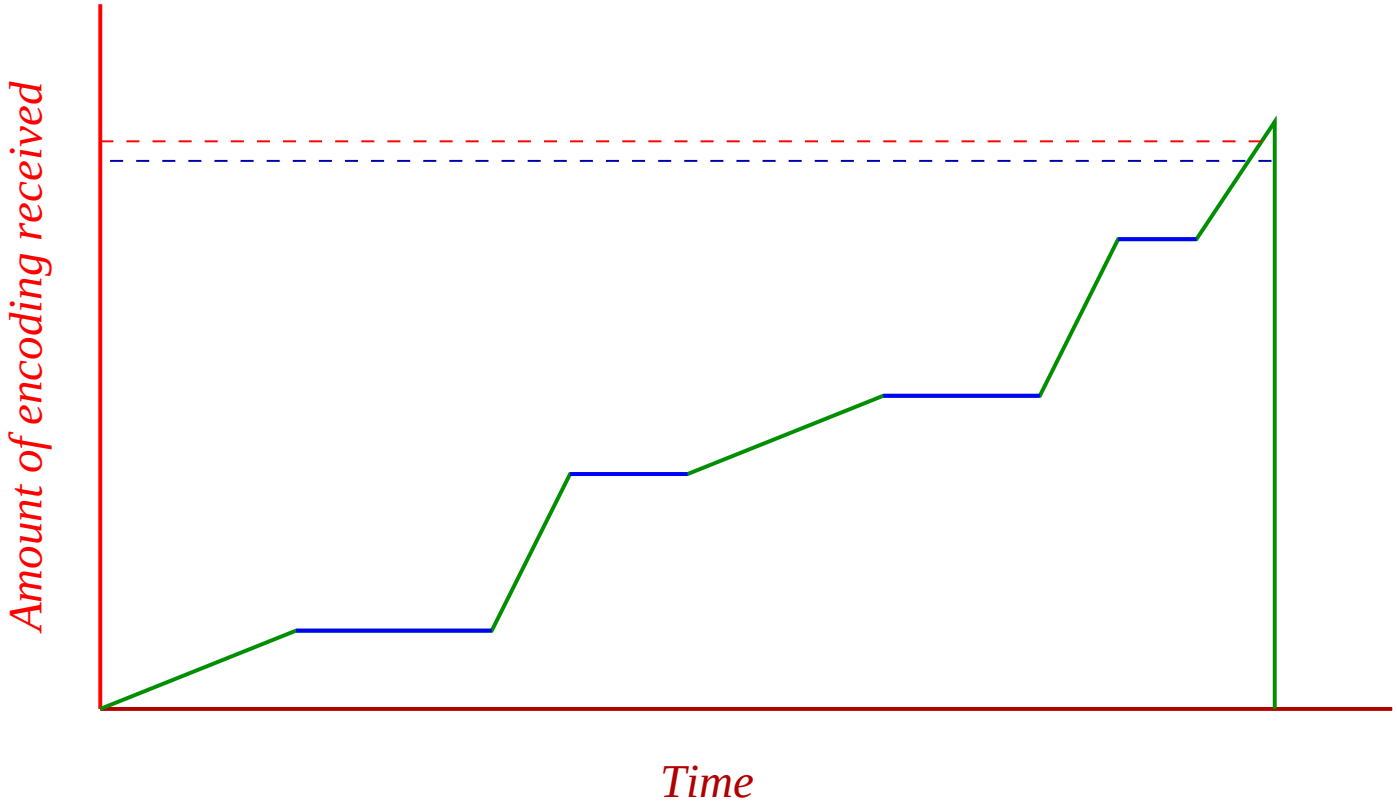
- fully reliable,
- low network overhead,
- support vast number of receivers with heterogeneous characteristics
- users want to access data at times of their choosing and these access times overlap.

A Solution



A Solution

Client joins multicast group until enough of the encoding has been received, and then decodes to obtain original data.



Digital Fountain, http://www.dfountain.com.

Open problems

Asymptotic theory

1. Classification of capacity achieving sequences for the erasure channel.
2. Capacity achieving sequences for other channels.
3. Exponentially small error probabilities for the decoder (instead of polynomially small).

Explicit constructions

1. Constructions using finite geometries.
2. Construction using Reed-Solomon-Codes.
3. Algebraic constructions.

Short codes

Graphs with loops.

Algorithmic issues

1. Design and analysis of new decoding algorithms.
2. Design of new encoders.

Applications

Packet based wireless networks.

Randomness

Use of randomness in other areas: random convolutional codes?.