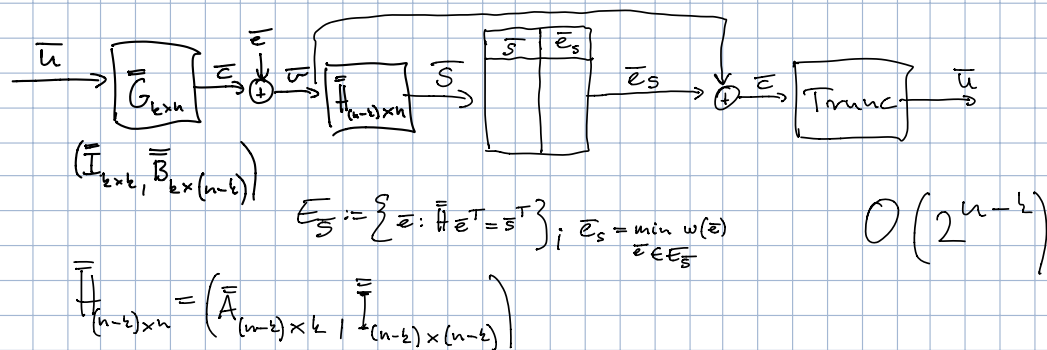
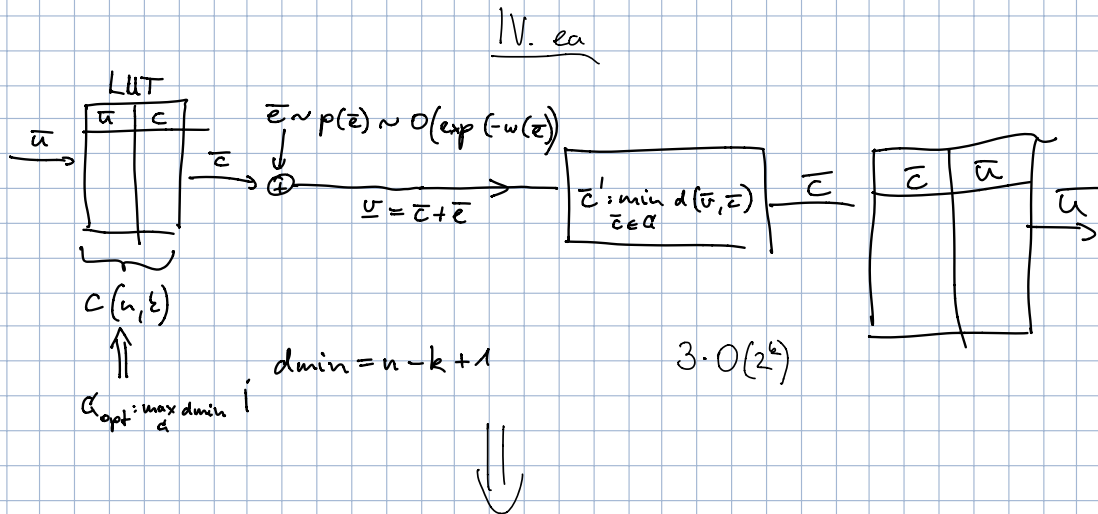




Lemma: If $\bar{c} : \bar{H} \bar{c}^T = \bar{s}^T$ then $\bar{c}' = \bar{c} + c : \bar{H} \bar{c}'^T = \bar{s}^T$
 $\downarrow$ b.i.z.
 $\bar{H} \bar{c}'^T = \bar{H} (\bar{c} + c)^T = \bar{H} \bar{c}^T + \bar{H} c^T = \bar{H} \bar{c}^T = \bar{s}^T$

Pl: $C(5, 2) \quad \bar{G}_{2 \times 5} = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \Rightarrow \bar{H}_{3 \times 5} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$

$\bar{s} = (0 \ 0 \ 1)$

$E_s = \{ (0 \ 0 \ 0 \ 0 \ 1), (0 \ 1 \ 1 \ 1 \ 0), (1 \ 0 \ 1 \ 1 \ 1), (1 \ 1 \ 0 \ 0 \ 0) \}$

$\swarrow + \bar{c} \quad \searrow + \bar{c}$

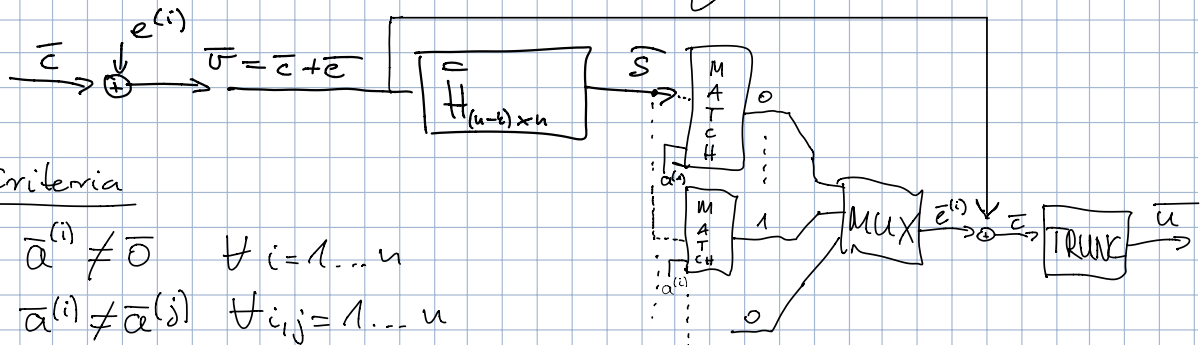
Constructing a code which can correct $\forall$ single error — Hamming code

↳ wired MODEM comm. (only single error)

single error: $\bar{e}^{(i)} = (0 \dots 0 \underset{\substack{\uparrow \\ \text{"i"}}}{1} 0 \dots 0) \stackrel{?}{\rightarrow} \text{"i"}$

$\bar{H} \bar{r}^T = \bar{s}^T \leftarrow \text{computable} \Rightarrow \bar{H} (\bar{c} + \bar{e}^{(i)})^T = \bar{s}^T \Rightarrow \underbrace{\bar{H} \bar{c}^T}_{\substack{\uparrow \\ \text{given}}} + \underbrace{\bar{H} \bar{e}^{(i)T}}_{\substack{\uparrow \\ \text{?}}} = \bar{s}^T \Rightarrow \boxed{\bar{H} \bar{e}^{(i)T} = \bar{s}^T}$
 $\uparrow$ given $\uparrow$ observable

$\left(\begin{array}{c} \bar{a}^{(1)T} \\ \vdots \\ \bar{a}^{(i)T} \\ \vdots \\ \bar{a}^{(n-k)T} \end{array} \right) \cdot \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \stackrel{?}{\rightarrow} \text{"i"} = \begin{pmatrix} a_1^{(i)} \\ a_2^{(i)} \\ \vdots \\ a_{n-k}^{(i)} \end{pmatrix} = \bar{a}^{(i)T} = \bar{s}^T$
 $\dim(\bar{a}^{(i)}) = n-k, \forall i=1 \dots n$
 $\uparrow$ i-th column vector



Criteria

$\bar{a}^{(i)} \neq \bar{0} \quad \forall i=1 \dots n$
 $\bar{a}^{(i)} \neq \bar{a}^{(j)} \quad \forall i, j=1 \dots n, i \neq j$

$2^{n-k} - 1 = n$
 $2^{n-k} = n+1$

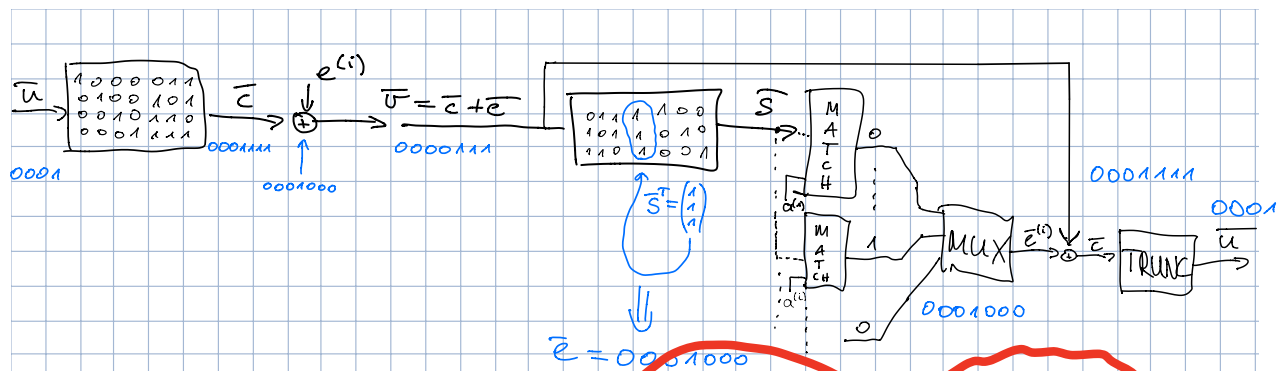
$\sum_{i=0}^1 \binom{n}{i} = 2^{n-k}$ this is a perfect code $\left(\sum_{i=0}^t \binom{n}{i} = 2^{n-k} \right)$
 code

n	k
3	1
7	4
15	11
31	26

$C(7, 4)$

$\bar{H}_{3 \times 7} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$
 $3 \leq i \leq 6 \leq 7$

$\bar{G}_{4 \times 7} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$



$$\left\lfloor \frac{a_{\min} - 1}{2} \right\rfloor = 1$$

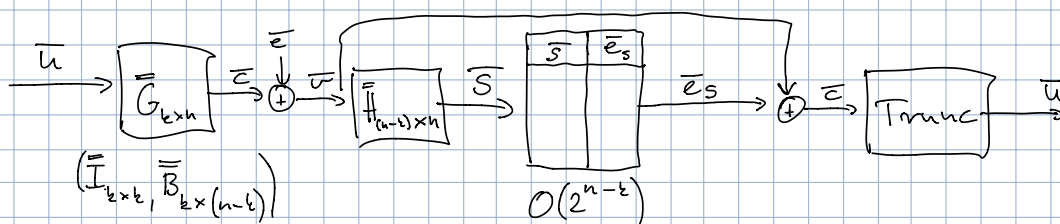
$$d_{\min} \geq 3$$

$$d_{\min} \leq n - k + 1 = 3 - 1 + 1 = 3 \quad \left. \vphantom{d_{\min} \leq n - k + 1} \right\} d_{\min} = 3$$

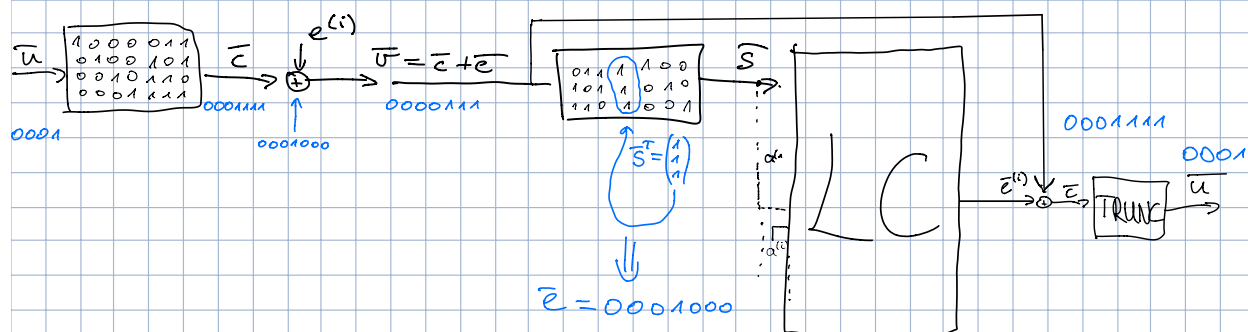
MDS

This is my first
MDS
code

2016. 09. 30.



Hamming-code: can correct $\forall$ single error



Given: $P_b, \gamma(QoS): P_b' \leq 10^{-8}$

received message $\neq$ sent message

$$P_{\text{blockerror}} = P(\hat{u} \neq u) = P(\hat{e} \neq e) = \sum_{i=2}^n P_b^i (1-P_b)^{n-i} \binom{n}{i} = 1 - (1-P_b)^n$$

$\uparrow$
 10^{-8}

$$\Psi(P_b, \gamma) \Rightarrow n, k$$

$$2^{n-k} = n+1$$

PL:

$$P_b = 10^{-2}, \gamma = 5 \rightarrow C(7,4) \Rightarrow \text{data speed drop } \frac{4}{7}$$

How to correct $\forall$ double error

$$e^{(i,j)} = (0 \dots 0 \overset{i}{1} 0 \dots 0 \overset{j}{1} 0 \dots 0) \rightarrow "i,j"$$

$$\begin{array}{c} \bar{e} \\ \downarrow \\ \bar{e}^{(i,j)} \\ \downarrow \\ \oplus \\ \bar{c} \end{array} \rightarrow \bar{s} = \bar{c} + \bar{e}^{(i,j)}$$

$$\bar{H}_{(n-k) \times n} \bar{s} = \bar{s}^T$$

$$\bar{H} \bar{s}^T = \bar{s}^T \Rightarrow \bar{H} (\bar{c} + \bar{e}^{(i,j)})^T = \bar{s}^T$$

given

$$\Rightarrow \bar{H} \bar{c}^T + \bar{H} \bar{e}^{(i,j)T} = \bar{s}^T \quad \left| \begin{array}{l} \bar{H} \bar{e}^{(i,j)T} = \bar{s}^T \\ \uparrow \\ \text{given} \end{array} \right. \quad ???$$

$$\begin{pmatrix} \bar{a}^{(i)T} & \bar{a}^{(j)T} \\ \vdots & \vdots \\ \bar{a}^{(i)T} & \bar{a}^{(j)T} \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} \bar{a}^{(i)T} \\ \vdots \\ \bar{a}^{(i)T} \end{pmatrix} + \begin{pmatrix} \bar{a}^{(j)T} \\ \vdots \\ \bar{a}^{(j)T} \end{pmatrix} = \bar{s}^T$$

Matching

Criteria:

- $\bar{a}^{(i)} = \bar{0} \quad \forall i = 1 \dots n$
 - $\bar{a}^{(i)} \neq \bar{a}^{(j)} \quad \forall i, j = 1, \dots, n \quad i \neq j$
 - $\bar{a}^{(i)} \neq \bar{a}^{(m)} + \bar{a}^{(e)} \quad \forall i, m, e = 1, \dots, n \quad i \neq m \neq e$
 - $\bar{a}^{(i)} + \bar{a}^{(j)} = \bar{a}^{(m)} + \bar{a}^{(e)}$
- single error
- double error

$$\alpha_i \bar{e}^{(i)} + \alpha_j \bar{e}^{(j)} + \alpha_c \bar{a}^{(c)} + \alpha_m \bar{a}^{(m)} = 0 \Rightarrow \text{in } \bar{H}$$

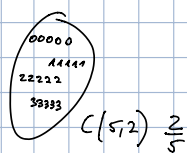
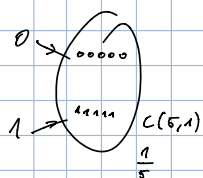
4 column vector have to be linearly independent $\Rightarrow \bar{H}_{4 \times 5} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix} \in C(5,1)$

Lemma: If a $C(n,k)$ linear code can correct " t " # of errors; then $\bar{H}_{(n-k) \times n}$ must have 2^t linearly indep. column vectors!

Binary vectors are poor for being linearly independent

q -ary vectors

$$\Rightarrow u_i, c_i, v_i, e_i \in \{0, 1, \dots, q-1\}$$



Galois Fields

$$t \rightarrow \bar{H}$$

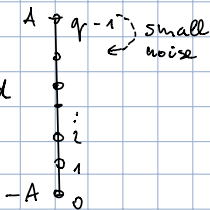
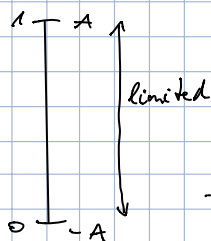
" 2^t " linearly ind. vectors

2016.10.05

Binary - codes $\Rightarrow q$ -ary codes

$$u_i, c_i, v_i \in \{0, 1\}$$

$$u_i, v_i, c_i \in \{0, 1, \dots, q\}$$



$$\text{Data Speed } \log(q) \rightleftharpoons \text{BER}$$

Finite Algebra $\rightarrow$ Galois Field: $GF(q) = \{0, 1, \dots, q-1\}$; "+" ; "*" "

$\forall \alpha, \beta \in GF(q) \rightarrow \alpha + \beta \in GF(q)$

$\alpha + \beta = \beta + \alpha ; (\alpha + \beta) + \gamma = \alpha + (\beta + \gamma)$

$\exists \emptyset \quad \alpha + \emptyset = \alpha \quad \forall \alpha \in GF(q)$

$\forall \alpha \in GF(q) \quad \exists \beta : \alpha + \beta = \emptyset$
 $\beta = \alpha^{-1}_a = (-\alpha)$

$\forall \alpha, \beta \in GF(q) \setminus \{0\} \rightarrow \alpha \cdot \beta \in GF(q) \setminus \{0\}$

$\alpha \cdot \beta = \beta \cdot \alpha ; (\alpha \cdot \beta) \cdot \gamma = \alpha \cdot (\beta \cdot \gamma)$

$\exists 1 : \alpha \cdot 1 = \alpha \quad \forall \alpha \in GF(q) \setminus \{0\}$

$\forall \alpha \in GF(q) \setminus \{0\} \quad \exists \beta : \alpha \cdot \beta = 1$
 $\beta = \alpha^{-1}_m = (1/\alpha)$

$\alpha \cdot (\beta + \gamma) = \alpha \cdot \beta + \alpha \cdot \gamma$

Eg.

q is prime

"+" ; "*" | $\alpha + \beta = u \cdot q + r$
 $r < q$
 $\alpha + \beta \mod q = r$

$\alpha \cdot \beta = u' \cdot q + r'$
 $r' < q$
 $\alpha \cdot \beta \mod q = r'$

$GF(5) \rightarrow 4 + 3 = 7 = 1 \cdot 5 + 2$
 $4 + 3 \mod 5 = 2$

$GF(7) \rightarrow 6 \cdot 3 = 18 = 2 \cdot 7 + 4$
 $6 \cdot 3 \mod 7 = 4$

$3 = 5^{-1}$ over $GF(7)$
 $5 \cdot 3 = 15 = 2 \cdot 7 + 1$

$1 + 4 = 1 \cdot 5 + \emptyset \quad (-1) = 4$
 additive inverse

$5x + 4 = 3$ over $GF(7)$

$5x = 3 + 4^{-1}$
 $5x = 3 + 3$

$5x = 6$
 $x = (5^{-1}) \cdot 6$
 $x = 3 \cdot 6 = 4$
 $x = 4$

$$\forall \alpha \in GF(q) \setminus \{0\} \rightarrow \alpha^{q-1} = 1$$

Proof

$$\alpha_1 \quad \alpha_2 \quad \alpha_{q-1} \quad | \quad \underbrace{\alpha \cdot \alpha_1}_{\alpha_{i_1}} \quad \underbrace{\alpha \cdot \alpha_2}_{\alpha_{i_2}} \quad \underbrace{\alpha \cdot \alpha_{q-1}}_{\alpha_{i_{q-1}}} \quad \dots \quad ?$$

$$\text{Min } m : \alpha^m = 1 \rightarrow m = \text{order}(\alpha)$$

$$\begin{array}{c} \nexists \text{ order}(\alpha) = q-1 \\ \downarrow \\ \text{primitive element} \\ \text{of } GF(q) \end{array}$$

Power table over $GF(7)$

element	1	2	3	4	5	6	ord
1	1	1	1	1	1	1	1
2	2	4	1	2	4	1	3
3	3	2	6	4	5	1	6
4	4	2	1	4	2	1	3
5	5	4	6	2	3	1	6
6	6	1	6	1	6	1	2

$$2^3 = 8 = 1 \cdot 7 + 1$$

primitive element

$$3^3 = 27 = 3 \cdot 7 + 6$$

$$3^3 = 3^1 \cdot 3^2 = 3 \cdot 2 = 6$$

primitive element

$$3^4 = 3^2 \cdot 3^2 = 2 \cdot 2 = 4$$

$$3^5 =$$

$$5^4 = 5^2 \cdot 5^2 = 4 \cdot 4$$

$\forall GF(q)$ has at least 1 primitive element

Polynomials over $GF(q)$

$$a(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_n x^n; a_0, a_1, a_2, \dots, a_n \in GF(q)$$

$$b(x) = b_0 + b_1 x + b_2 x^2 + \dots + b_n x^n + \dots + b_m x^m$$

$$c(x) = a(x) + b(x) = a_0 + b_0 + (a_1 + b_1)x + (a_2 + b_2)x^2 + \dots + (a_m + b_m)x^m$$

$$c(x) = a(x) \cdot b(x) = a_0 \cdot b_0 + (a_0 b_1 + a_1 b_0)x + (a_0 b_2 + a_1 b_1 + a_2 b_0)x^2 + \dots + \left(\sum_{j=0}^{C_i} b_{i-j} \cdot a_j \right) x^i \dots$$

$C_i = \min\{i, \deg(a \cdot b)\}$

$$a(x) \xleftrightarrow{X} \bar{a} = (a_0, a_1, \dots, a_n)$$

polynom	vector
$c(x) = a(x) + b(x)$	$\bar{c} = \bar{a} + \bar{b}$
$c(x) = a(x) \cdot b(x)$	$\bar{c} = \bar{a} * \bar{b}$

Given: $a(x), d(x)$

$$\deg(a(x)) = n > \deg(d(x)) = k$$

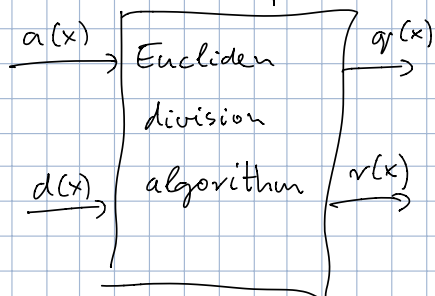
$$\exists q(x), r(x) : a(x) = q(x)d(x) + r(x)$$

quotient $\swarrow$ remainder (unradik.)

$$\deg(q(x)) = n - k$$

$$\deg(r(x)) < k$$

"n-k" steps



Basic law of algebra

$$a(x) \quad \exists x_i : a(x_i) = 0 \quad (\text{root})$$

$$a(x) = b(x)(x - x_i)$$

$$\deg(a(x)) > \deg(b(x))$$

$$a(x) = b(x)(x - x_i) + r \quad \xrightarrow{r=0} \phi$$

$|_{x=x_i} \qquad \qquad |_{x=x_i}$

$$\exists x_j : b(x_j) = \phi$$

$$b(x) = c(x)(x - x_j) \rightarrow$$

$$\rightarrow a(x) = c(x)(x - x_j)(x - x_i)$$

$$\# \text{ of roots} \leq \deg(a(x))$$

2016.10.07.

Binary $\rightarrow$ "q"-ary
- wireless comm.

$$GF(q) : \text{"q" prime} \rightarrow \text{mod } q$$

Reed - Solomon codes (RS)

$$C(n, k) \text{ over } GF(k), \quad n = q - 1, \quad \alpha_0, \alpha_1, \dots, \alpha_{n-1} \in GF(q) \quad \alpha_i \neq 0$$

$\forall i = 0, \dots, n-1$

$$\bar{u} = (u_0, u_1, \dots, u_{k-1}) \xrightarrow{\sum} u(x) = u_0 + u_1 x + u_2 x^2 + \dots + u_{k-1} x^{k-1}; \quad \deg(u(x)) = k-1$$

$$\bar{c} = (c_0, c_1, \dots, c_{n-1})$$

$$\left. \begin{aligned} c_0 &= u(x) \Big|_{x=\alpha_0} = u(\alpha_0) = u_0 + u_1 \alpha_0 + u_2 \alpha_0^2 + \dots + u_{k-1} \alpha_0^{k-1} \\ c_1 &= u(x) \Big|_{x=\alpha_1} = u(\alpha_1) = u_0 + u_1 \alpha_1 + u_2 \alpha_1^2 + \dots + u_{k-1} \alpha_1^{k-1} \\ &\vdots \\ c_{n-1} &= u(x) \Big|_{x=\alpha_{n-1}} = u(\alpha_{n-1}) = u_0 + u_1 \alpha_{n-1} + u_2 \alpha_{n-1}^2 + \dots + u_{k-1} \alpha_{n-1}^{k-1} \end{aligned} \right\} \begin{array}{l} \text{Linear Code} \\ * \end{array}$$

$$*(c_0, c_1, \dots, c_{n-1}) = (u_0, u_1, \dots, u_{k-1}) \cdot \bar{G}_{k \times n} \quad \left| \quad \bar{G}_{k \times n} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_0 & \alpha_1 & \dots & \alpha_{n-1} \\ \alpha_0^2 & \alpha_1^2 & \dots & \alpha_{n-1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_0^{k-1} & \alpha_1^{k-1} & \dots & \alpha_{n-1}^{k-1} \end{pmatrix}$$

$$d_{\min} = w_{\min} = n - \# \text{ of zero components} \geq n - (k-1) = n - k + 1 \quad \left| \quad d_{\min} \leq n - k + 1\right.$$

$d_{\min} = n - k + 1$ **MDS Code**

Standard implementation: $\alpha \in GF(q)$ | $\alpha_0 = \alpha^0 = 1, \alpha_1 = \alpha^1 = \alpha, \alpha_2 = \alpha^2, \dots, \alpha_{n-1} = \alpha^{n-1}$
 $\text{ord}(\alpha) = q-1$

$$\bar{G}_{k \times n} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \alpha^{2(k-1)} & \dots & \alpha^{(k-1)(n-1)} \end{pmatrix}$$

E.g. 4 double errors must be corrected

$$t=2 \quad \left| \quad t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor = 2 \right.$$

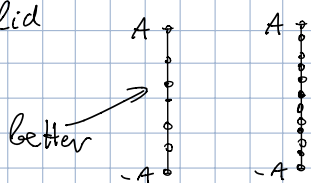
"q" prime $n = q - 1$

$$t = \left\lfloor \frac{n - k + 1 - 1}{2} \right\rfloor = 2 \rightarrow n - k = 2t = 4$$

q	n	k
1	0	—
2	1	—
3	2	—
5	4	0
7	6	2
11	10	6
13	12	8
⋮	⋮	⋮

valid

we choose the smallest



$$\bar{G}_{2 \times 6} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 6 & 4 & 5 \end{pmatrix}$$

$$\forall \bar{c}^T = \bar{0}^T \quad \forall \bar{c} \in C$$

$$\bar{C} = (c_0, c_1, \dots, c_{n-1}) \xrightarrow{X} \begin{cases} c(x) = c_0 + c_1 x + c_2 x^2 + \dots + c_{n-1} x^{n-1} \\ c(x) = c_0 + c_1 x + c_2 x^2 + \dots + c_{n-1} x^{n-1} = 0 \\ c(x^2) = c_0 + c_1 x^2 + c_2 x^4 + \dots + c_{n-1} x^{2(n-1)} = 0 \\ \vdots \\ c(x^{n-k}) = c_0 + c_1 x^{n-k} + c_2 x^{2(n-k)} + \dots + c_{n-1} x^{(n-k)(n-1)} = 0 \end{cases} \Big|_{x=x^i} = \emptyset \quad ; \quad i=1, \dots, n-k$$

$$\begin{matrix} \bar{H} \\ (n-k) \times n \end{matrix} = \begin{pmatrix} 1 & x & x^2 & \dots & x^{n-1} \\ 1 & x^2 & x^4 & \dots & x^{2(n-1)} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x^{n-k} & x^{2(n-k)} & \dots & x^{(n-k)(n-1)} \end{pmatrix} \bar{G}_{2 \times n} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & x & x^2 & \dots & x^{n-1} \\ 1 & x^2 & x^4 & \dots & x^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x^{n-k} & x^{2(n-k)} & \dots & x^{(n-k)(n-1)} \end{pmatrix}$$

$$\left(\bar{H} \bar{c}^T \right)_\ell = \sum_{i=0}^{n-1} x^{ei} c_i = \sum_{i=0}^{n-1} x^{li} \sum_{j=0}^{n-1} x^{ij} u_j = \sum_{j=0}^{n-1} u_j \underbrace{\sum_{i=0}^{n-1} x^{i(j+\ell)}}_{\substack{\text{arithmetic or} \\ \text{geometric sequence} \\ \text{quation} = x^{j+\ell}}} = \sum_{j=0}^{n-1} u_j \frac{x^{n(j+\ell)} - 1}{x^{j+\ell} - 1} = \emptyset$$

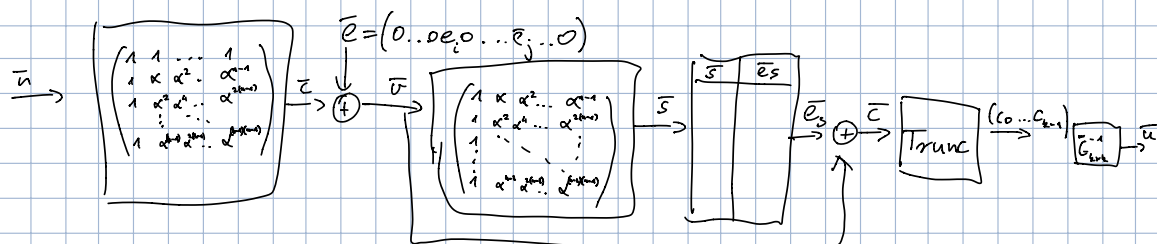
$n = q-1$
 $x^{n(j+\ell)} = (x^{q-1})^{j+\ell} = 1^{j+\ell} = 1$

$t \rightarrow 2t$ # of linearly independent column vectors in $\bar{H}$ $2t = n-k \rightarrow \text{MDS}$

$$\det \begin{pmatrix} x^i & x^{i_2} & \dots & x^{i_{n-k}} \\ x^{2i} & x^{2i_2} & \dots & x^{2i_{n-k}} \\ \vdots & \vdots & \ddots & \vdots \\ x^{(n-k)i} & x^{(n-k)i_2} & \dots & x^{(n-k)i_{n-k}} \end{pmatrix} \neq \emptyset$$

Vandermonde matrix

$(n-k) \times (n-k)$



$$e_{ij} \in \{0, 1, \dots, q-1\}$$

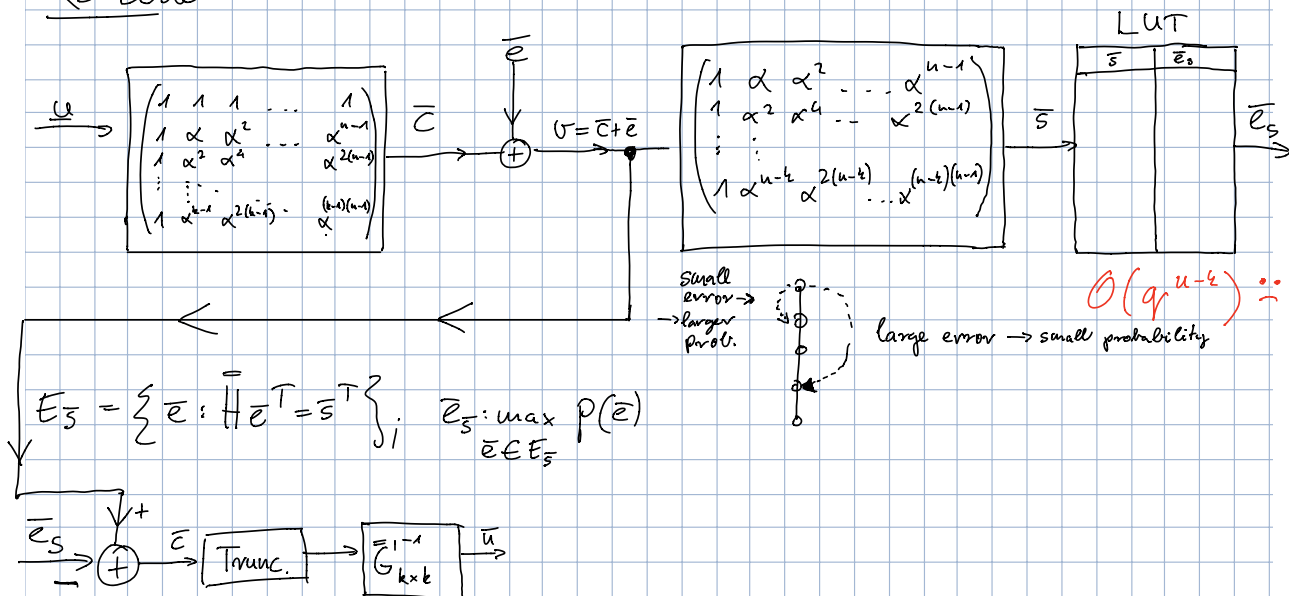
$$\underbrace{(u_0 \dots u_{k-1})}_{\text{unknown}} \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{n-k} & \alpha^{2(n-k)} & \dots & \alpha^{(n-k)(n-1)} \end{pmatrix} = \underbrace{(c_0 \dots c_{n-k})}_{\text{known}} \quad \begin{matrix} \# \text{ of unknowns} = k < \# \text{ of equations} \\ \text{overdetermined} \end{matrix}$$

Optimal
MDS

overdetermined
↓
We drop some of its column
↓
Quadratic Vandermonde matrix
↓
invertible

2016.10.12.

RS code



Design: given " t " errors

$$n = q - 1 \quad n - k = 2t \quad "q" \text{ prime}$$

q	n	k
$\vdots$	$\vdots$	$\vdots$
11	10	4

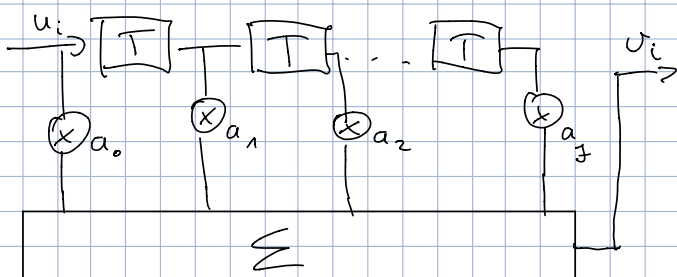
$$GF(q) \ni \alpha$$

Pr: 4 3 error

q	n	k
$\vdots$	$\vdots$	$\vdots$
11	10	4

$C(10, 4)$ over $GF(11)$

Linear Feed Forward Shift Register (LFFSR)



$$i = 0$$

$$v_0 = a_0 u_0$$

$$i = 1$$

$$v_1 = a_0 u_1 + a_1 u_0$$

$$i = 2$$

$$v_2 = a_0 u_2 + a_1 u_1 + a_2 u_0$$

$$\vdots$$

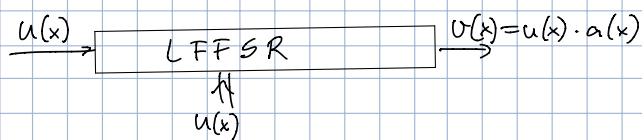
$$\vdots$$

$$\vdots$$

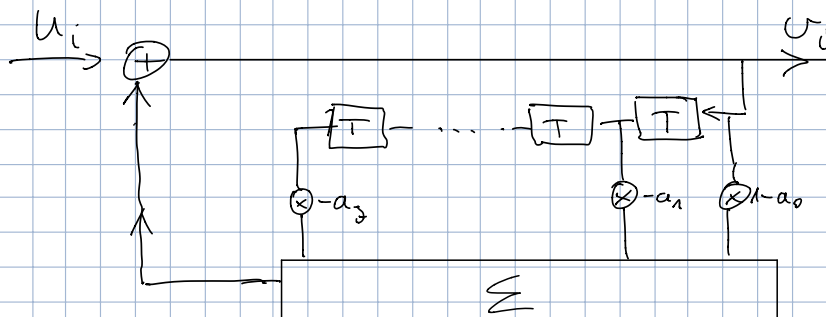
$$i$$

$$v_i = \sum_{j=0}^i a_j u_{i-j}$$

$$\xrightarrow{X} v(x) = a(x) \cdot u(x)$$



Linear Feedback Shift Register (LFSR)



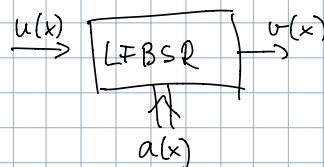
$$v_i = (1-a_0)v_i - a_1v_{i-1} - a_2v_{i-2} - \dots - a_nv_{i-n} + u_i$$

$$\underbrace{v_i - (1-a_0)v_i}_{a_0v_i} + a_1v_{i-1} + a_2v_{i-2} + \dots + a_nv_{i-n} = u_i$$

$$\sum_{j=0}^n a_j v_{i-j} = u_i$$

$$a(x) \cdot v(x) = u(x)$$

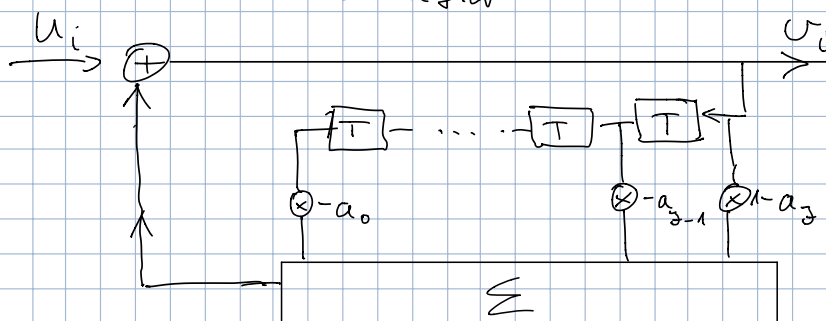
$$v(x) = \frac{u(x)}{a(x)}$$



$$v(x) = \frac{u(x)}{a(x)}$$

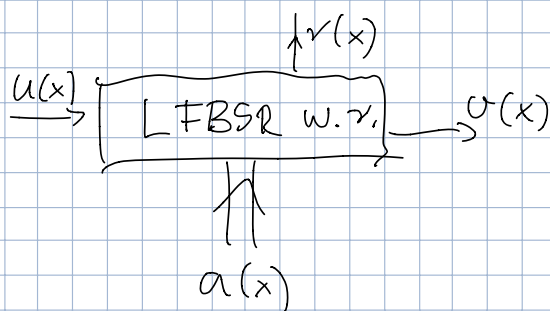
LFSR w.r. (with remainder)

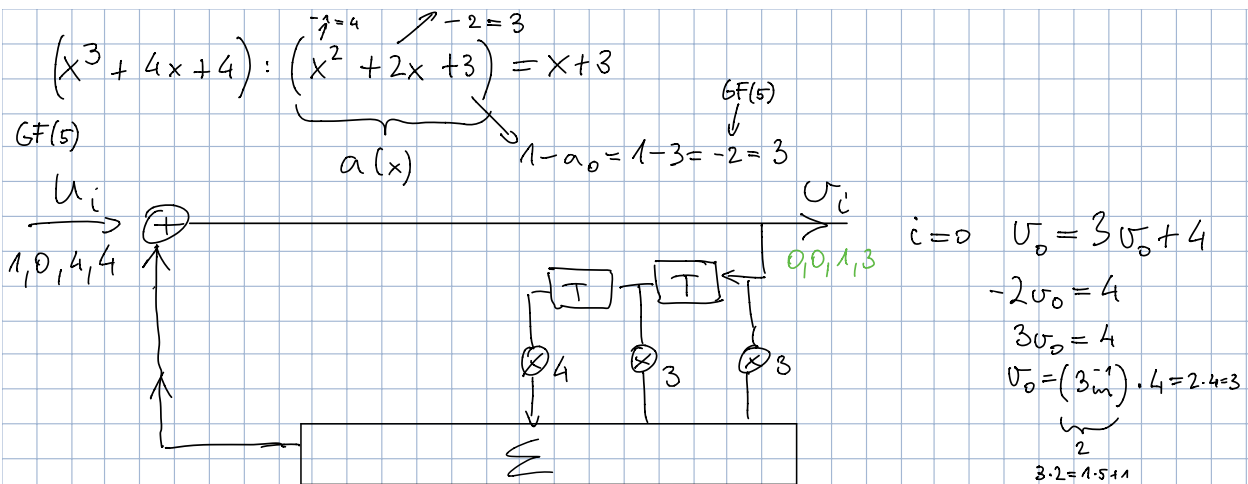
$n-1$ shifter



$$u(x) = v(x) \cdot a(x) + r(x)$$

$$\deg(u(x)) < n \Rightarrow \deg(a(x)) = n$$





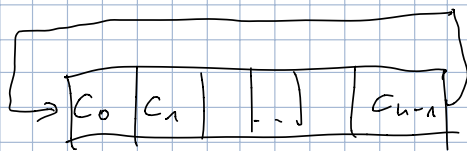
$i=1$
 $u_1 = 3u_1 + 4 + 4 = 1$
 $-2u_1 = 3$
 $u_1 = 2 \cdot 3 = 1$

$i=2$
 $u_2 = 3u_2 + 3 + 2$
 $u_2 = 3u_2 + 0 \rightarrow u_2 = 0$

$i=3$
 $u_3 = 3u_3 + 4 + 1 = 0$
 $u_3 = 3u_3 + 0$

Cyclic Shift Operation

$\vec{c} = (c_0, c_1, \dots, c_{n-2}, c_{n-1}) \rightarrow \vec{c}' = S\vec{c} = (c_{n-1}, c_0, c_1, \dots, c_{n-2})$



$c(x) = c_0 + c_1x + \dots + c_{n-2}x^{n-2} + c_{n-1}x^{n-1}$

$c'(x) = x \cdot c(x) \mod (x^n - 1)$

$c'(x) = c_{n-1} + c_0x + c_1x^2 + \dots + c_{n-2}x^{n-1}$

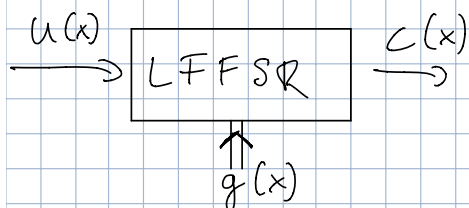
$x \cdot c(x) = c_0x + c_1x^2 + \dots + c_{n-2}x^{n-1} + c_{n-1}x^n$

$x \cdot c(x) = c_{n-1}(x^n - 1) + c'(x) \Rightarrow c'(x) = x \cdot c(x) \mod (x^n - 1)$

$\vec{c} \leftrightarrow c(x);$ if $c \in G \rightarrow \vec{c}' = S\vec{c} \in G$
 (if $c(x) \in G \rightarrow x \cdot c(x) \mod (x^n - 1) \in G$)

if $\vec{c}_1, \vec{c}' \in G \rightarrow \alpha \cdot \vec{c} + \beta \cdot \vec{c}' \in G$

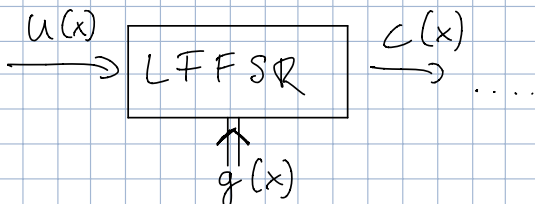
$$C(n, k) \rightarrow \exists g(x) \begin{cases} \rightarrow \deg(g(x)) = n-k \\ \rightarrow g_{n-k} = 1 \\ \rightarrow \forall c(x) \in C: c(x) = u(x) \cdot g(x) \\ \rightarrow g(x) \mid x^n - 1 \end{cases}$$



Cyclic Linear codes

$$\begin{aligned} 1, & \text{ if } c \in C \rightarrow \bar{c} = Sc \in C \\ 2, & \text{ } \alpha, \bar{c}, \bar{c}' \in C \rightarrow \alpha \cdot \bar{c} + \beta \bar{c}' \in C \end{aligned} \quad \rightarrow \quad \exists g(x) : \begin{cases} \rightarrow \deg(g(x)) = n-k & C(n, k) \\ \rightarrow g_{n-k} = 1 \\ \rightarrow \forall c(x): c(x) = u(x)g(x) \\ \rightarrow g(x) \mid x^n - 1 \end{cases}$$

Encoding



RS codes are Cyclic $\rightarrow c(x) \mid x^n - 1$ $\rightarrow c(x) = c_0 + c_1x + \dots + c_{n-1}x^{n-1}$

MDS + coding is implemented on SR

optimal performance + real time

$$\left. \begin{aligned} c(\alpha) &= 0 \\ c(\alpha^2) &= 0 \\ &\vdots \\ c(\alpha^{n-k}) &= 0 \end{aligned} \right\} \alpha, \alpha^2, \dots, \alpha^{n-k} \text{ are roots of } c(x)$$

$$c(x) = \prod_{i=1}^{n-k} (x - \alpha^i) \cdot u(x); \deg(u(x)) = k-1$$

$$c(x) = g(x) \cdot u(x); g(x) = \prod_{i=1}^{n-k} (x - \alpha^i)$$

Példa:

feladat: correct $\forall t=2$ error ; $n=q-1$; q prime; $n-k=2 \cdot 2=4$

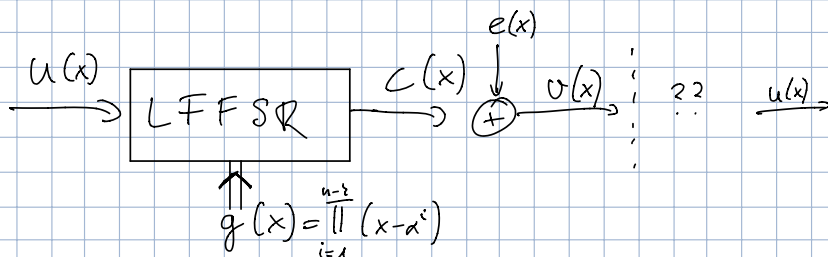
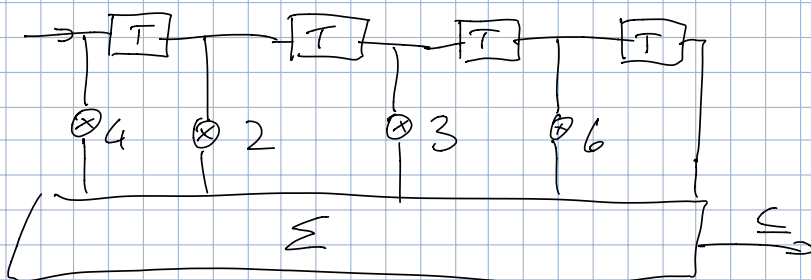
q	n	k
1	—	—
2	—	—
3	—	—
5	—	—
7	6	2
11		

$GF(7) \ni 3$ prim. e.

$$g(x) = \prod_{i=1}^{n-k} (x - \alpha^i) = \prod_{i=1}^4 (x - \alpha^i) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)(x - \alpha^5) = (x + 4)(x + 5)(x + 1)(x + 3)$$

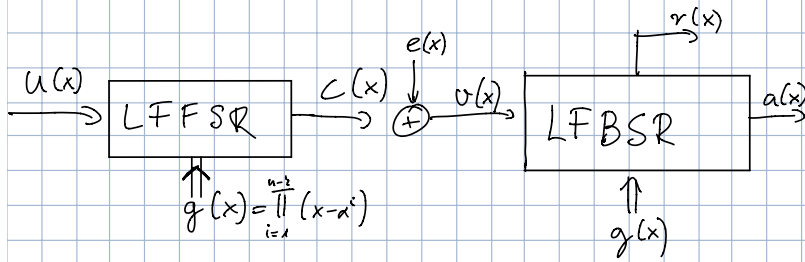
$$g(x) = (x^2 + 2x + 6)(x^2 + 4x + 3) = x^4 + 2x^3 + 6x^2 + 4x^3 + x^2 + 3x + 3x^2 + 6x + 4 =$$

$$= x^4 + 6x^3 + 3x^2 + 2x + 4$$



Known: $v(x) = u(x) \cdot g(x) + e(x)$

↑ observable ↑ known



$$v(x) = a(x)g(x) + r(x)$$

$\uparrow$ given $\downarrow$ comp. $\uparrow$ known. $\downarrow$ comp.

division w.r.

$$a(x) = u(x); \quad r(x) = e(x)$$

$$\deg(r(x)) \leq n-k$$

$$\underline{e} = (0 \dots 0 e_{i_1} 0 \dots 0 e_{i_2} 0 \dots 0 e_{i_t} 0 \dots 0)$$

$\downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow$
 $0 \quad i_1 \quad \quad i_2 \quad \quad i_t \quad n-1$

$$e(x) = 0 + 0x + e_{i_1}x^{i_1} + e_{i_2}x^{i_2} + \dots + e_{i_t}x^{i_t}$$

"Low error

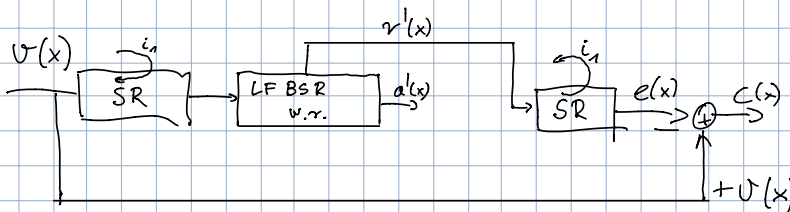
configuration"

$$\rightarrow \begin{array}{c} \text{---} x \text{---} x \text{---} x \text{---} \end{array} \begin{array}{c} n-k-1 \\ i_t \leq n-k \end{array} \quad \deg(e(x)) \leq n-k$$

Assumption:

$$i_t - i_1 \leq n-k$$

"bursty nature"



Problem: We don't know the position i_1

$$e(x) = b(x)g(x) + r(x)$$

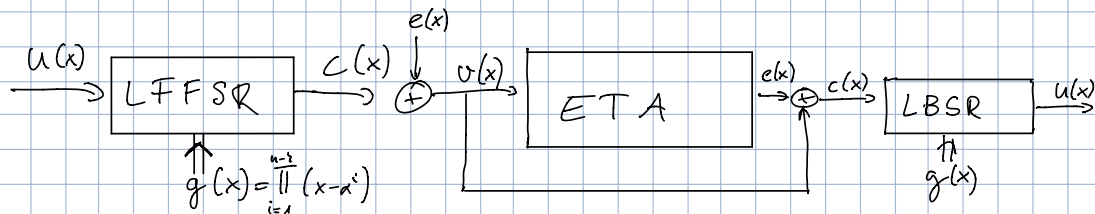
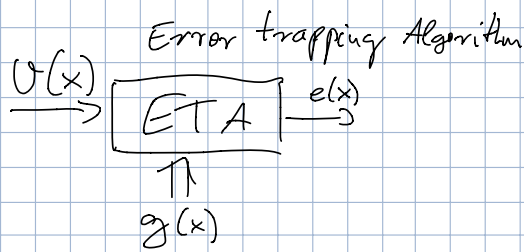
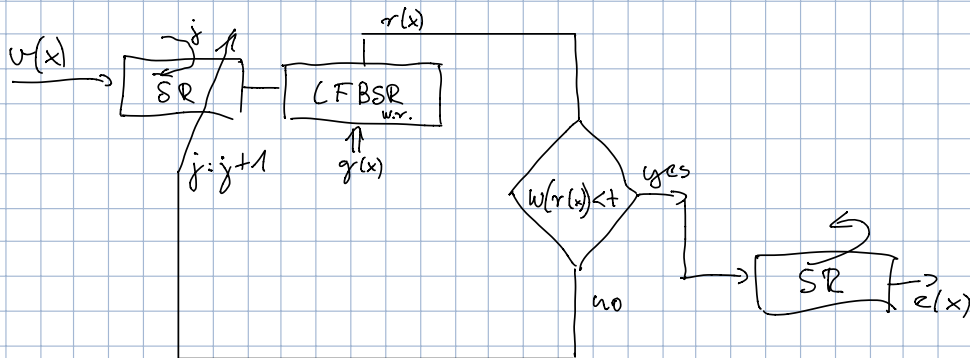
$$\rightarrow v(x) \bmod g(x) = e(x) \bmod g(x)$$

$$v(x) = (u(x) + b(x))g(x) + r(x)$$

$$x^{-j} \cdot e(x) = b'(x) \cdot g(x) + r'(x)$$

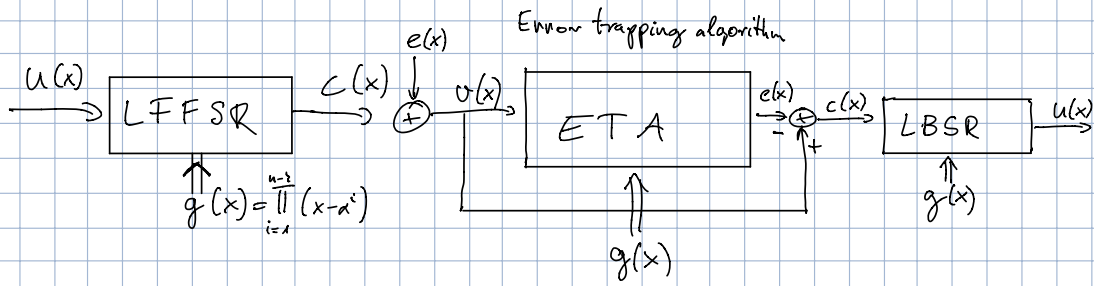
$\therefore$ error term is obtained

$$x^{-j} \cdot e(x) = r'(x) \rightarrow e(x) = x^j \cdot r'(x)$$



$$c(x) = u(x) \cdot g(x) \rightarrow u(x) = \frac{c(x)}{g(x)} \quad \left| \quad g(x) = \prod_{i=1}^{n-1} (x - \alpha^i)$$

2016.10.26.



given "t" $\rightarrow q, n, k, \alpha$

RS codes



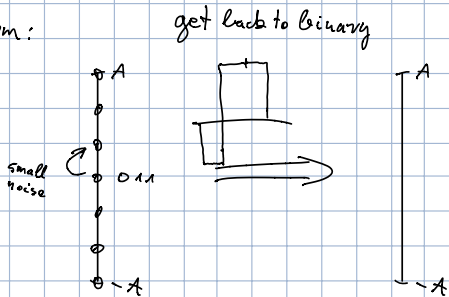
Optimal (MDS)

SR impl.



Real Time system

Problem:

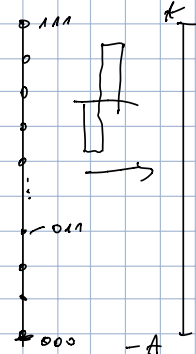


"q" prime

000
001
010
011
111
⋮

data speed
 $\log_2$

$$q = 2^m \quad 2^3$$



$$q = 2^m \rightarrow GF(2^m)$$

ERROR

$$2 \cdot 4 = 0 \pmod 8$$

let nem 0 elem szorzata nem lehetne 0

Algebra over $GF(p^m)$

" p " is prime $\rightarrow p=2$

$P(y)$; $\deg(P(y))=m$

$P(y) \neq P_1(y) \cdot P_2(y)$ irreducible polynomial

$\deg(P_i(y)) < \deg(P(y)) \quad i=1,2$

Pl:

$$P(y) = y^2 + y + 1$$

$$P(y) = y^3 + y + 1$$

$$P(y) = y^4 + y + 1$$

$$P(y) = y^5 + y^2 + 1$$

		operation domain	
elements		p-ary represent	polynomial repr.
s y m b o l i c	0	0 0 ... 0	$0 \cdot y^{m-1} + 0y^{m-2} + \dots + 0 \cdot y^0 = 0$
	1	0 0 ... 1	$0 \cdot y^{m-1} + 0y^{m-2} + \dots + 1 \cdot y^0 = 1$
	$\vdots$	$\vdots$	$\vdots$
	α	$a_{m-1} a_{m-2} \dots a_0$	$a_{m-1} \cdot y^{m-1} + a_{m-2} \cdot y^{m-2} + \dots + a_0 = a(y)$
	β	$b_{m-1} b_{m-2} \dots b_0$	$b_{m-1} \cdot y^{m-1} + b_{m-2} \cdot y^{m-2} + \dots + b_0 = b(y)$
	γ	$c_{m-1} c_{m-2} \dots c_0$	$c_{m-1} \cdot y^{m-1} + c_{m-2} \cdot y^{m-2} + \dots + c_0 = c(y)$
	$\vdots$	$\vdots$	$\vdots$
$p^m - 1$		$p-1 \quad p-1 \dots p-1$	$(p-1) \cdot y^{m-1} + (p-1) \cdot y^{m-2} + \dots + (p-1)$

$$\alpha + \beta \rightarrow a(y) + b(y) = q(y)p(y) + c(y) \rightarrow a(y) + b(y) \bmod p(y) = c(y) \rightarrow c(y) \rightarrow \gamma$$

$$\deg(c(y)) < \deg(p(y)) = m$$

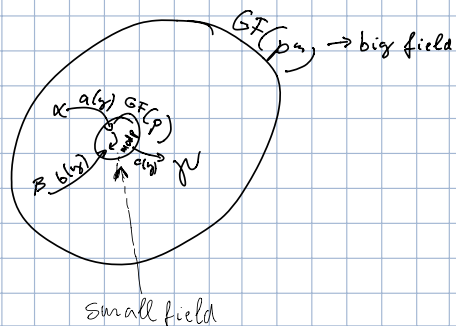
$$\alpha + \beta = \gamma$$

$$\alpha \cdot \beta \rightarrow a(y) \cdot b(y) = q'(y) \cdot p(y) + c'(y) \rightarrow a(y) \cdot b(y) \bmod p(y) = c'(y)$$

$$\downarrow$$

$$\gamma'$$

$$\alpha \cdot \beta = \gamma'$$



Algebra over $GF(2^2)$

$$p(y) = y^2 + y + 1$$

elements	binary rep	polynomial
0	0 0	$0y^1 + 0y^0 = 0$
1	0 1	$0y^1 + 1y^0 = 1$
2	1 0	$1 \cdot y^1 + 0y^0 = y$
3	1 1	$1y^1 + 1y^0 = y+1$

$$1 + 2 = 1 + y = 3$$

$$1 + 3 = 1 + y + 1 = y = 2$$

$$2 + 2 = y + y = (1+1) \cdot y$$

$$2 + 3 = y + y + 1 = (1+1) \cdot y + 1 = 1$$

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

*	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

$$2 \cdot 2 = y \cdot y = y^2 = 1 \cdot (y^2 + y + 1) + y + 1$$

$$2 \cdot 3 = y \cdot (y+1) = y^2 + y = 1 \cdot (y^2 + y + 1) + 1$$

$$3 \cdot 3 = (y+1)(y+1) = y^2 + y + y + 1 = y^2 + 1 = 1(y^2 + y + 1) + y$$

Power Table over $GF(8)$

$$GF(2^3) \quad p(y) = y^3 + y + 1$$

primitive element: 2 $\swarrow$ $GF(2^3)$

elements	binary	polynomial
0	0 0 0	$0y^2 + 0y^1 + 0y = 0$
1	0 0 1	$0y^2 + 0y^1 + 1y = 1$
2	0 1 0	y
3	0 1 1	$y+1$
4	1 0 0	y^2
5	1 0 1	y^2+1
6	1 1 0	y^2+y
7	1 1 1	y^2+y+1

y^0	1	1
y^1	y	2
y^2	y^2	4
y^3	$y+1$	3
y^4	y^2+y	6
y^5	y^2+y+1	7
y^6	y^2+1	5
y^7	1	
y^8	y	
y^9	y^2	

$$y^3 = 1(y^3 + y + 1) + y + 1$$

$$y^4 = y \cdot (y^3 + y + 1) + y^2 + y$$

$$y^5 = (y^2 + 1)(y^3 + y + 1) + y^2 + y + 1$$

$$y^5 + y^3 + y^2 + y^3 + y + 1$$

$$y^6 = (y^2 + y + 1)(y^3 + y + 1) + y^2 + 1$$

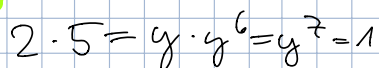
$$6 \cdot 3 = y^4 \cdot y^3 = y^7 = 1$$

$$6 + 3 = y^2 + y + y + 1 = y^2 + 1 = 5$$

Can be implemented on Shift Registers

$$2 \cdot \alpha = y \cdot (a_0 + a_1 y + a_2 y^2) = a_0 y + a_1 y^2 + a_2 y^3 =$$

$$= a_0 y + a_1 y^2 + a_2 (y+1) = a_2 + (a_0 + a_2) y + a_1 y^2$$



content is $\propto$

The diagram illustrates a control system with a feedforward path. An input signal α is split into two branches. One branch goes through a series of integrators (SR blocks) to produce a signal $5 \cdot \alpha$. The other branch goes through a gain block $2 \cdot \alpha$. The outputs of these two branches are combined to produce the final output $5 \cdot \alpha$.

VLSI

$$\mathcal{L}(x) := \mathcal{L}_0 + \mathcal{L}_1 x + \mathcal{L}_2 x^2 + \dots + \mathcal{L}_n x^n \quad \mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2 \dots \mathcal{L}_n, x \in GF(2^n)$$

$$\mathcal{L}(x) := a_0(y) + a_1(y)x + a_2(y)x^2 + \dots + a_n(y)x^n$$

$$\mathcal{L}(x) := y^{i_0} + y^{i_1}x + y^{i_2}x^2 + \dots + y^{i_n}x^n \longrightarrow \text{standard representation}$$

$$L_0 = 7 \Rightarrow a_0(y) = y^2 + y + 1 \rightarrow y^5$$

$$L(x) = 17 + 5x + 6x^2 + 4x^3 + 3x^4$$

$$L(x) = (y^2 + y + 1) + (y^2 + 1)x + (y^2 + y)x^2 + y^2x^3 + (y + 1)x$$

$$L(x) = y^5 + y^6x + y^4x^2 + y^2x^3 + y^3x^4$$

y^0	1
y^1	y
y^2	y^2
y^3	$y + 1$
y^4	$y^2 + y$
y^5	$y^2 + y + 1$
y^6	$y^2 + 1$

RS codes over $GF(2^m)$

$$n = 2^m - 1 \quad k: \quad n - k = 2 \cdot t = 4$$

m	n	k
1	1	-
2	3	-
3	7	3

$C(7,3)$
 $y \in GF(2^n)$

$$g(x) = \prod_{i=1}^{n-1} (x - y^i) = (x+y)(x+y^2)(x+y^3)(x+y^4) = (x^2 + (y+y^2)x + y^3)(x^2 + (y^3+y^4)x + 1) =$$

$$= (e_{1,4,2,3}) / (e_{1,2,6,3}) = \sqrt[4]{2} \cdot \sqrt[4]{3} \cdot \sqrt[4]{2} \cdot \sqrt[4]{3} \cdot \sqrt[4]{2} \cdot \sqrt[4]{3} \cdot \sqrt[4]{2} \cdot \sqrt[4]{3} = \sqrt[4]{2^4 \cdot 3^4} = \sqrt[4]{16 \cdot 81} = \sqrt[4]{1296}$$

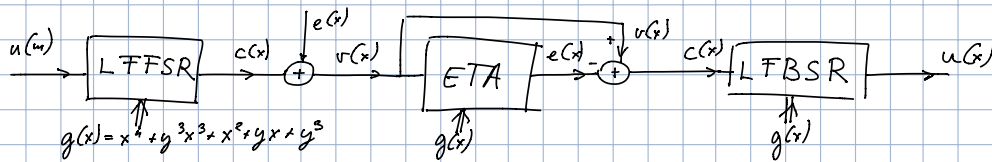
+	0	1	2	3
0	0			
1		0		
2			0	
3				0

$$-3 = 3$$

$$-(y+1) = y+1$$

$$\begin{array}{l} y^0 \\ y^1 \\ y^2 \\ y^3 \\ y^4 \\ y^5 \\ y^6 \\ y^7 \\ \vdots \end{array} \begin{array}{l} 1 \\ y \\ y^2 \\ y+1 \\ y^2+y \\ y^2+y+1 \\ y^2+y+1 \\ 1 \\ y \\ \vdots \end{array}$$

VLSI



- Optimal
- Real Time
- Binary Transmission

$$\bar{u} = (7, 7, 7) \xrightarrow{?} \bar{c}$$

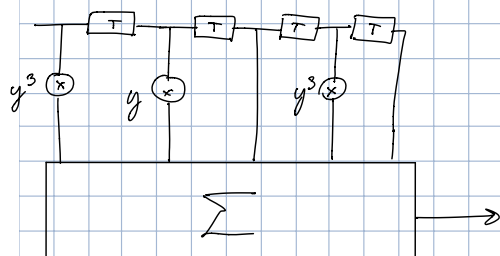
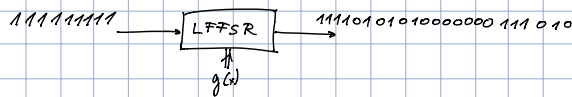
$$u(x) = 7x^2 + 7x + 7 \longrightarrow y^5x^2 + y^5x + y^5$$

$$u(x) = (y^2+y+1)x^2 + (y^2+y+1)x + (y^2+y+1)$$

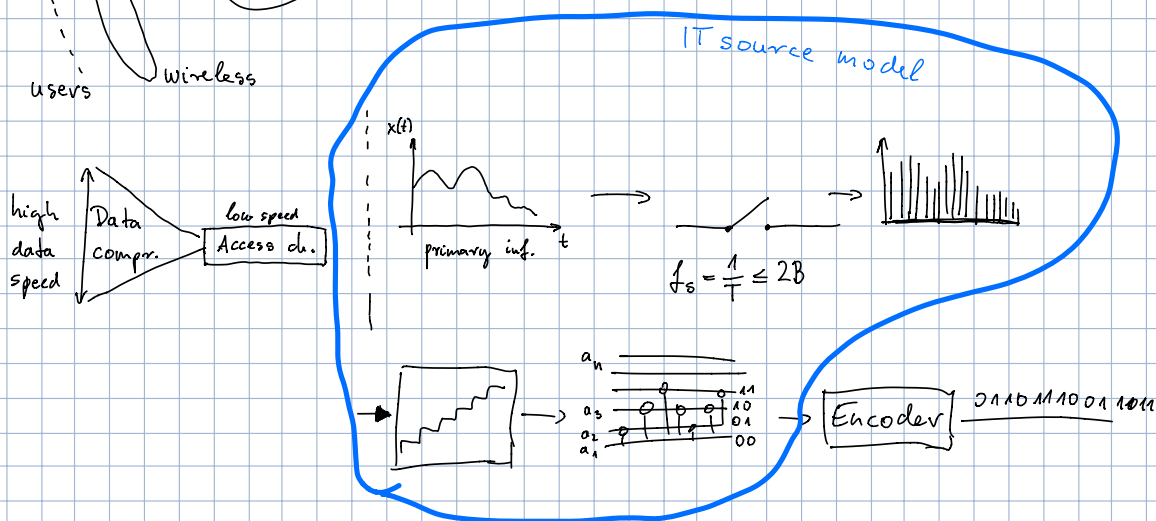
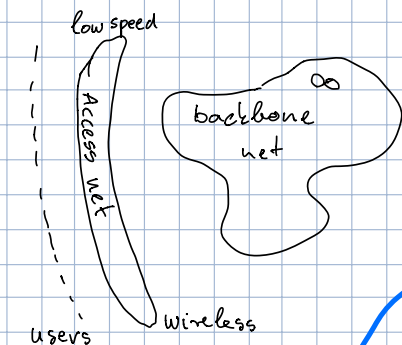
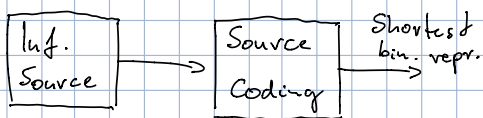
$$\begin{aligned} & (x^4 + y^3x^3 + x^2 + yx + y^5)(y^5x^2 + y^5x + y^5) = y^5x^6 + y^5x^5 + y^5x^4 + y^6x^3 + y^6x^2 \\ & + y^5x^5 + y^5x^4 + y^5x^3 + y^6x^2 + y^6x + y^5x^4 + y^5x^3 + y^5x^2 + y^6x + y^6 = \\ & = y^5x^6 + y^6x^5 + y^6x^4 + 0x^3 + 0x^2 + y^5x + y^6 \end{aligned}$$

$$\bar{c} = (7, 5, 2, 0, 0, 7, 2)$$

$$\begin{aligned} y + y^5 &= y + y^5 + y + 1 = y^6 \\ y^6 + y^5 + y &= y^2 + 1 + y^5 + y + 1 = 0 \\ y + y^6 &= y + y^6 + 1 \end{aligned}$$



Data Compression - Source coding

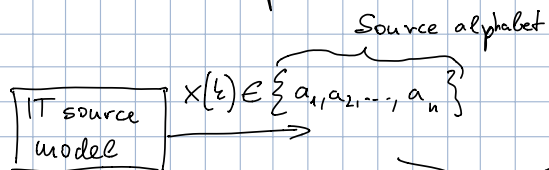
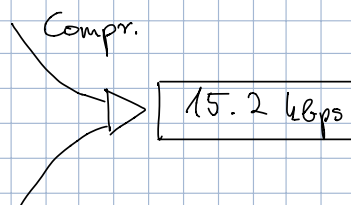


$$\text{Data speed} = f_s \cdot n \text{ [bps]}$$

Speech $8 \text{ kHz} \cdot 8 \text{ bit} = 64 \text{ kbps}$

Music $44.1 \text{ kHz} \cdot 16 \text{ bit} = 705.6 \text{ kbps}$

Video 84 Mbps



Memoryless model
Stationary

Source coding LUT

a_1	10
a_2	011
a_3	11
$\vdots$	$\vdots$
a_n	1111

111010010
Shortest binary sequence

C_{opt} :

minimize L

Model:

$$p_1, p_2, \dots, p_n \rightarrow p(x)$$

$$a_1, a_2, \dots, a_n \rightarrow x$$

$$c_1, c_2, \dots, c_n \rightarrow c(x)$$

$$l_1, l_2, \dots, l_n \rightarrow l(x) \quad \text{length}$$

Average code length

$$L: E(l(x)) = \sum_x p(x) \cdot l(x)$$

Data speed

$$f_s \cdot L$$

Intuitive idea:

If $p(x)$ is large then $l(x)$ small

If $p(x)$ is small then $l(x)$ large

Questions

1, Unique decoding?

How can we separate the codewords (there is no space char)

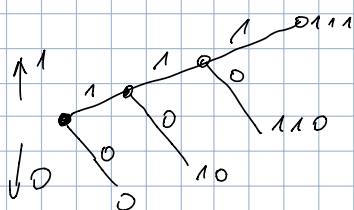
2, $l(x)$?

I need a coffee very much

I need a coffee very much

Prefix free codes $c_i \neq c_j$

01 01110



Shortest codeword

$$\sum_x 2^{-l(x)} \leq 1$$

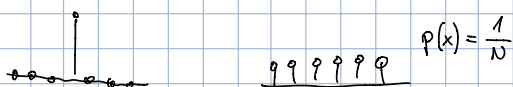
Kraft-McMillan inequality

$$I(x) = \lg \frac{1}{p(x)}$$

↓
random variable

$$H(x) = \mathbb{E} I(x) = \sum_x p(x) \lg \frac{1}{p(x)} \leftarrow \text{entropy}$$

$$0 \leq H(x) \leq \lg N$$



I-divergence:

$$D(p \parallel q) := \sum_x p(x) \lg \frac{p(x)}{q(x)}$$

$\nwarrow$ $\swarrow$
 $p(x)$ $q(x)$

$$D(p \parallel q) \geq 0$$

$$L := \mathbb{E}(\ell(x)) = \sum_x p(x) \cdot \ell(x)$$

$$-D(p \parallel r) = \sum_x p(x) \lg \frac{r(x)}{p(x)} \leq \sum_x p(x) \left(\frac{r(x)}{p(x)} - 1 \right) = \sum_x r(x) - \sum_x p(x) = 1 - 1 = 0$$

$$D(p \parallel q) = \sum_x p(x) \lg \frac{p(x)}{\frac{1}{N}} = \sum_x p(x) \lg (p(x) \cdot N) = \sum_x p(x) \left\{ \lg p(x) + \lg N \right\} =$$

$\nwarrow$ $\swarrow$
 $p(x)$ $q(x) = \frac{1}{N}$

$$= \sum_x p(x) \lg p(x) + \lg N \sum_x p(x) = -H(x) + \lg N \geq 0$$

$$H(x) \leq \lg N$$

$$p(x) = \frac{1}{N} \rightarrow H(x) = \lg N$$

Source coding theorem

$$L \geq H(x)$$

Ha az elosztás nem egyenletes elosztású,
akkor tömöríthető az adat

proof: $p(x)$ source distr.

$$q(x) = \frac{2^{-\ell(x)}}{\sum_y 2^{-\ell(y)}}$$

↑
artificial

$$0 \leq q(x) \leq 1 \quad \forall x$$

$$\sum_x q(x) = 1$$

$$D(p \parallel q) = \sum_x p(x) \log \frac{p(x)}{q(x)} = \sum_x p(x) \log \frac{p(x)}{2^{-\ell(x)} / \sum_y 2^{-\ell(y)}}$$

$$= \sum_x p(x) \log \left\{ p(x) \cdot 2^{\ell(x)} \cdot \underbrace{\sum_y 2^{-\ell(y)}}_{\leq 1} \right\} \leq \sum_x p(x) \log \{ p(x) \cdot 2^{\ell(x)} \} =$$

$$= \sum_x p(x) \left\{ \log p(x) + \log 2^{\ell(x)} \right\} = \sum_x p(x) \log p(x) + \sum_x p(x) \underbrace{\log 2^{\ell(x)}}_{\ell(x)} = -H(x) + L \geq 0$$

$$H(x) \leq L$$

Objective is to approach $H(x)$
as much as possible

$$L := \sum_x p(x) \ell(x); \quad H(x) = \sum_x p(x) \log \frac{1}{p(x)}$$

Shannon-Fano coding

$$\ell(x) := \left\lceil \log \frac{1}{p(x)} \right\rceil$$

$$\sum_x 2^{-\ell(x)} \leq 1$$

$$\sum_x 2^{-\left\lceil \log \frac{1}{p(x)} \right\rceil}$$

$$\log \frac{1}{p(x)} \leq \left\lceil \log \frac{1}{p(x)} \right\rceil \leq \log \frac{1}{p(x)} + 1$$

$$H(X) \leq L = \sum_x p(x) \left\lceil \log \frac{1}{p(x)} \right\rceil \leq \sum_x p(x) \left(\log \frac{1}{p(x)} + 1 \right) = H(X) + 1$$

algorithm:

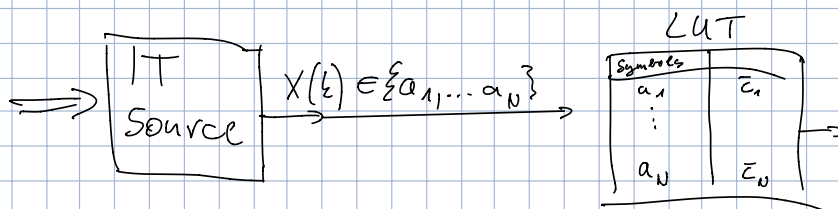
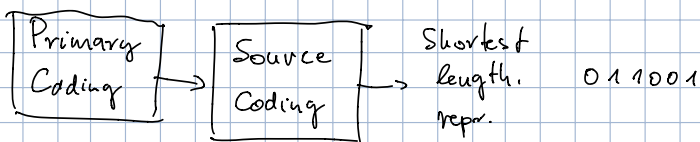
Given $p(x)$

$$1, \ell(x) = \left\lceil \log \frac{1}{p(x)} \right\rceil \quad x \in \{a_1, \dots, a_N\}$$

2, Binary tree $\rightarrow C$

3, LUT

Source Coding



$P(X(k) = a_i)$ Memoryless Channel

$$p_1 \ p_2 \dots p_N \rightarrow p(x)$$

$$a_1 \ a_2 \dots a_N \rightarrow x$$

$$\bar{c}_1 \ \bar{c}_2 \dots \bar{c}_N \rightarrow \bar{c}(x)$$

$$\ell_1 \ \ell_2 \dots \ell_N \rightarrow \ell(x)$$

$$L := E(\ell(x)) = \sum_x \ell(x) p(x)$$

$$\text{Data speed} = f_s \cdot L \quad \Rightarrow \quad C_{\text{opt}} : \min_C L$$

$$\sum_x 2^{-\ell(x)} \leq 1$$

- 1, $\ell(x)$
- 2, Binary tree

$$H(x) = \mathbb{E} \ell(x) = \sum_x p(x) \ell\left(\frac{1}{p(x)}\right)$$

$$0 \leq H(x) \leq \ell N$$

Source Coding Theorem

$$L \geq H(x)$$

How to approach entropy as close as possible?

Shannon-Fano coding

$$\ell(x) = \left\lceil \ell\left(\frac{1}{p(x)}\right) \right\rceil ; \quad \sum_x 2^{-\ell(x)} \leq 1$$

$$H(x) \leq L^{SF} \leq H(x) + 1$$

algorithm:

Given $p(x)$

$$1, \ell(x) = \left\lceil \ell\left(\frac{1}{p(x)}\right) \right\rceil \quad x \in \{a_1, \dots, a_N\}$$

2, Binary tree $\rightarrow G$

3, LUT

SF kódok nem optimalizáltak!

kódhosszak nem a legközelebbiek

Huffman kódol

optimális feltevések

$$p_1 > p_2 > p_3 > \dots > p_{n-1} > p_n$$

1, if $p_1 > p_2$

$$\Downarrow$$
$$l_1 < l_2$$

2, $l_{n-1} = l_n$

$$H(X) \leq L \leq H(X) + \varepsilon$$

Given an " ε " $\rightarrow$ how to develop a code

$$H(X, Y) = H(X) + H(Y) \text{ if } X \text{ and } Y \text{ are independent random variables}$$

$$\begin{aligned} \sum_x \sum_y p(x, y) \log \frac{1}{p(x, y)} &= \sum_x \sum_y p(x) \cdot p(y) \log \frac{1}{p(x)p(y)} = \sum_x \sum_y p(x)p(y) \left\{ \log \frac{1}{p(x)} + \log \frac{1}{p(y)} \right\} \\ &= \left(\sum_x p(x) \log \frac{1}{p(x)} \right) \left(\sum_y p(y) \right) + \left(\sum_y p(y) \log \frac{1}{p(y)} \right) \left(\sum_x p(x) \right) = H(X) + H(Y) \end{aligned}$$

$$H(\underbrace{X_1, \dots, X_k}_{\text{independent, identically distributed}}) = \sum_{i=1}^k H(X_i) = k \cdot H(X_1)$$

independent, identically
distributed

Block coding

$$\bar{X} \rightarrow x(k), x(k-1), \dots, x(k-K+1) \mid Y$$

$$N^k \begin{cases} a_1 & a_1 & \dots & a_1 \\ a_1 & a_1 & \dots & a_2 \\ \vdots & & & \\ a_N & a_N & \dots & a_N \end{cases} \begin{matrix} y_1 \\ y_2 \\ \vdots \\ y_{N^k} \end{matrix}$$

$$\lambda = \frac{L}{K}$$

IT Source Y

Shannon-Fano coding on Y

$$H(Y) \leq L \leq H(Y) + 1$$

$$K \cdot H(X) \leq L \leq K \cdot H(X) + 1$$

$$H(X) \leq \lambda \leq H(X) + \frac{1}{K}$$

given:

$$\varepsilon \rightarrow K = \left\lceil \frac{1}{\varepsilon} \right\rceil$$

Dataspeed: $f_s \cdot \lambda$

$$\text{Dataspeed } O\left(\frac{1}{K}\right) \Leftrightarrow O(N^k)$$

Source Coding without the knowledge of $p(x)$?

Non-parametric data compression

Distribution-free data compression

2016.11.16

By what dataspeed can we communicate channel reliably?

$$H(Y|X) = \sum_x p(x) \sum_y p(y|x) \log \frac{1}{p(y|x)} = \sum_x \sum_y \underbrace{p(x) p(y|x)}_{p(x,y)} \log \frac{1}{p(y|x)} = \sum_x \sum_y p(x,y) \log \frac{1}{p(y|x)}$$

$$H(X, Y) = H(X) + H(Y)$$

$$\sum_x \sum_y p(x,y) \log \frac{1}{p(x,y)} = \sum_x \sum_y p(x,y) \log \frac{1}{p(y|x) \cdot p(x)} = \sum_x \sum_y p(x,y) \left\{ \log \frac{1}{p(x)} + \log \frac{1}{p(y|x)} \right\} =$$

$$= \sum_x \underbrace{\sum_y p(x,y)}_{p(x)} \ln \frac{1}{p(x)} + \sum_x \sum_y p(x,y) \ln \frac{1}{p(y|x)} = H(X) + H(Y|X)$$

Joint Information

$$I(X, Y) = D(p(x,y) \| p(x)p(y)) = \sum_x \sum_y p(x,y) \ln \frac{p(x,y)}{p(x) \cdot p(y)} =$$

$$= \sum_x \sum_y p(x,y) \ln \frac{p(x,y)}{p(x) \cdot p(y)} = \sum_x \sum_y p(x,y) \ln \frac{p(y|x) \cdot p(x)}{p(x) \cdot p(y)} =$$

$$= \sum_x \sum_y p(x,y) \cdot \ln p(x|y) + \underbrace{\sum_y \sum_x p(x,y)}_{p(y)} \ln \frac{1}{p(y)} = H(Y) - H(Y|X) = H(X) - H(X|Y)$$

Typical set:

$$\lim_{n \rightarrow \infty} \frac{1}{n} \ln \frac{1}{p(x_1 \dots x_n)} = H(X)$$

$$\frac{1}{n} \ln \frac{1}{p(x_1 \dots x_n)} = \frac{1}{n} \ln \left(\prod_{i=1}^n \frac{1}{p(x_i)} \right) = \frac{1}{n} \sum_{i=1}^n \left(\ln \frac{1}{p(x_i)} \right) = \frac{1}{n} \sum_{i=1}^n I(x_i) \Rightarrow \mathbb{E}(I(x)) = \sum_x p(x) \ln \frac{1}{p(x)} = H(X)$$

$$\mathcal{A} := \left\{ (x_1 \dots x_n) : 2^{-nH(X) - \varepsilon} \leq p(x_1 \dots x_n) \leq 2^{-nH(X) + \varepsilon} \right\}$$

Properties:

$$1, P(\mathcal{A}) \geq 1 - \varepsilon$$

$$2, |\mathcal{A}| \leq 2^{nH(X) + \varepsilon}$$

$$3, |\mathcal{A}| \geq (1 - \varepsilon) 2^{nH(X) - \varepsilon}$$

Conclusion

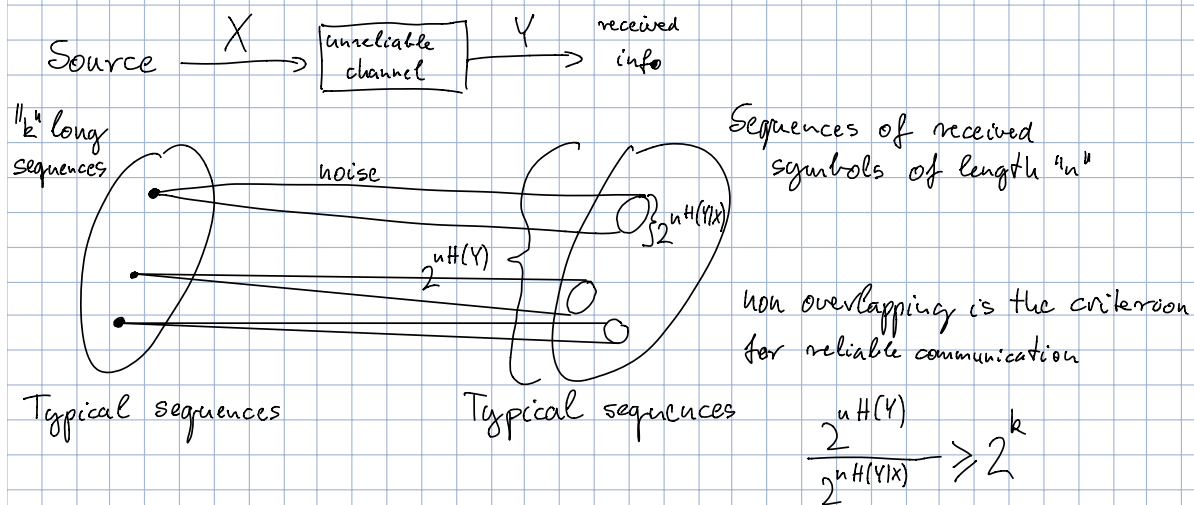
$$|A| \approx 2^{nH(x)}$$

$$P(A) \approx 1$$

$$P(x_1 \dots x_n) \approx 2^{-nH(x)}$$

A source is emitting
"only" typical sequences
which occur with
uniform distribution

How to communicate (at what data speed) reliably over an unreliable channel?



$$2^{n\{H(Y) - H(Y|X)\}} > 2^k$$

$$\frac{k}{n} \leq \max_{p(x)} H(Y) - H(Y|X)$$

$$\frac{k}{n} \leq \max_{p(x)} I(X, Y)$$

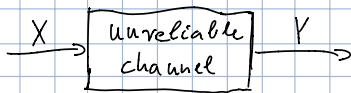
C : channel capacity

data rate $\leq$ channel capacity

$$\frac{k}{n} = R \leq C$$

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

2016.11.23



$$R := \frac{k}{n}$$

↑
data rate

k : length of information sequence
 n : length of encoded inf. of the sequence

Channel coding theorem

$$R \leq C$$

$$C = \max_{p(x)} I(X, Y) = \max_{p(x)} H(Y) - H(Y|X)$$

How to calculate C ?

Characterization of the channel

$$X \in \{x_1, \dots, x_n\}, \quad Y \in \{y_1, \dots, y_m\}$$

$$P_{ij} = P(Y = y_i | X = x_j) \rightarrow P_{yx} = P(Y = y | X = x)$$

↓
 $\bar{P}$

$$\sum_y P_{yx} = 1$$

"Symmetric channel"

The column vectors of $\bar{P}$ are permutations of each other!

$$\begin{pmatrix} 0,1 & 0,2 & 0,7 \\ 0,2 & 0,1 & 0,2 \\ 0,7 & 0,7 & 0,1 \end{pmatrix}$$

$\bar{P} \neq \bar{P}^T$ → Entropy-column

$$\sum_y P_{yx} \log \frac{1}{P_{yx}} = \sum_y P(Y = y | X = x) \cdot \log \frac{1}{P(Y = y | X = x)} = H(\bar{v})$$

$$C = \max_{p(x)} H(Y) - H(Y|X)$$

$$\uparrow \quad \vdots \quad \sum_x \sum_y p(x,y) \frac{1}{p(y|x)} = \sum_x p(x) \sum_y p(y|x) \ln \frac{1}{p(y|x)} = \sum_x p(x) \cdot \underbrace{H(\bar{r})}_1$$



Symmetric channel

$$C = \ln N - H(\bar{r})$$

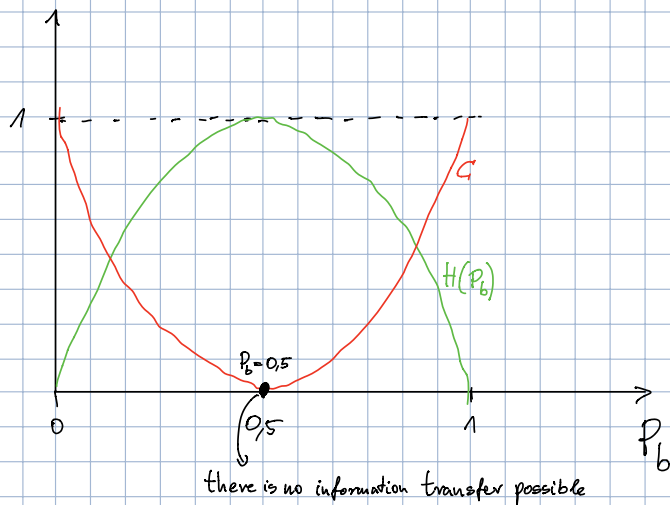
Application - channel capacity of BSC

transmitting 1 receiving 0

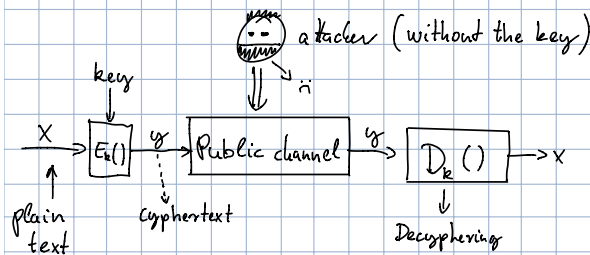
$$\bar{P} = \begin{pmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{pmatrix} = \begin{pmatrix} 1-p_b & p_b \\ p_b & 1-p_b \end{pmatrix} \rightarrow \text{Symmetric}$$

korábbi zöld keresés eredményét felhasználva

$$C = \ln N - H(\bar{r}) = \ln 2 - (1-p_b) \ln \frac{1}{1-p_b} - p_b \ln \frac{1}{p_b} = 1 - H(p_b)$$



CRYPTO

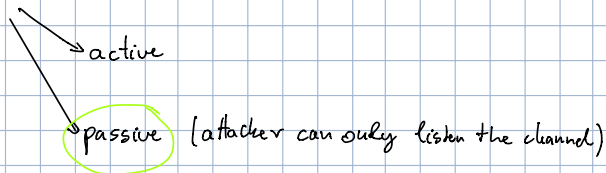


Construct: $E_k(x) = y$?

$D_k(y) = x$ is easy if " k " is known and almost impossible without the " k " parameter !

Hardly invertible functions !

Type of attacks



Passive attack

- cyphertext attacks: y_k $k=1, \dots, K$

- plain-cyphertext attacks: (x_k, y_k) $k=1, \dots, K$
 ↳ wants to get the key

- governed plain - cyphertext pairs:

(x_k, y_k) , $k=1, \dots, K$

↳ don't know the key but can generate y_k from x_k ? ? ?
 . . .

DPGGFF

COFFEE

eltolds $x+1$

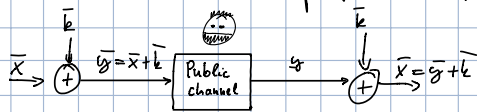
DMURITE $\Rightarrow$ MIDTERM

1 2 3 4 5 6 7
 3 7 1 6 2 4 5

$O(N!)$

Problem with these, that we have to pass the key to the receiver on a very reliable channel

One Time Pad (OTP) - perfect encryption



$$P(\bar{Y} = \bar{y} | \bar{X} = \bar{x}) = P(\bar{Y} = \bar{x} + \bar{k} | \bar{X} = \bar{x}) = P(\bar{K} = \bar{y} + \bar{x}) = \frac{1}{2^n}$$

$$P(\bar{Y} = \bar{y}) = \sum_{\bar{x}} P(\bar{Y} = \bar{y} | \bar{X} = \bar{x}) P(\bar{X} = \bar{x}) = \frac{1}{2^n} \sum_{\bar{x}} P(\bar{X} = \bar{x}) = \frac{1}{2^n} \Rightarrow P(\bar{Y}) = P(\bar{Y} | \bar{X})$$

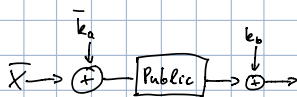
$\bar{X}$ and $\bar{Y}$ are statistically independent

Commutative key cryptography

$$E_{k_b}(E_{k_a}(x)) = E_{k_a}(E_{k_b}(x))$$

Given the plain text x

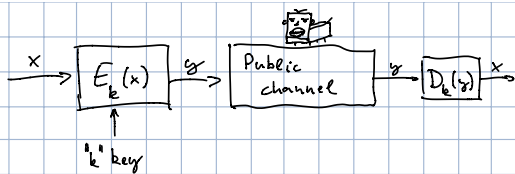
- 1, $y_a = E_{k_a}(x)$ } transmission
- 2, $E_{k_b}(y_a) = E_{k_b}(E_{k_a}(x))$ } transmission back to the sender
- 3, $D_{k_a}(y_b) = D_{k_a}(E_{k_b}(E_{k_a}(x))) = D_{k_a}(E_{k_b}(E_{k_a}(x))) = E_{k_b}(x) = y_c$ } transmission
- 4, $D_{k_b}(y_c) = D_{k_b}(E_{k_b}(x)) = x$



$$\bar{y}_a = \bar{x} + \bar{k}_a ; \bar{y}_b = \bar{x} + \bar{k}_a + \bar{k}_b ; \bar{y}_c = \bar{x} + \bar{k}_b$$

$$\bar{y}_a + \bar{y}_b + \bar{y}_c = \bar{x}$$

2016.11.25.



Commutative key cryptography

$$E_{k_b}(E_{k_a}(x)) = E_{k_a}(E_{k_b}(x))$$

$$\begin{array}{c} \bar{x} \xrightarrow{\oplus \bar{k}_a} \bar{y} \\ (\bar{x} + \bar{k}_a) + \bar{k}_b = (\bar{x} + \bar{k}_b) + \bar{k}_a \end{array}$$

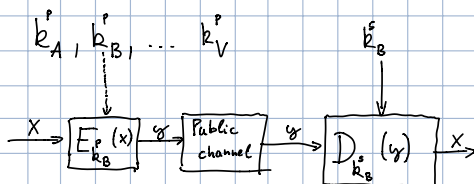
$$(x^{k_a})^{k_b} = x^{k_a \cdot k_b} = (x^{k_b})^{k_a} \quad \leftarrow \text{actually used unbreakable}$$

Concern

How to transmit the message only once, but no key transfer is needed

Public key cryptography

Key Repository - public keys



Objective

Construct algorithms for which $E_{k_b^p}(x); D_{k_b^s}(y)$ is very easy if k_b^p, k_b^s are known

If k_b^s is unknown then $D_{k_b^s}(y)$ is almost impossible

Recap of # theory
↑ number

(a, m) largest common divider - legnagyobb közös osztó

$$\exists s, t: (a, m) = sa + tm \quad s, t \in \mathbb{N}$$

if $(a, m) = 1$ then $\exists \bar{a}^{-1} : b \cdot a = 1 \pmod{m}$

$$1 = s \cdot a + t \cdot m ; b \cdot a = q \cdot m + 1 \Rightarrow b \cdot a - q \cdot m = 1$$

$\downarrow$
 s

$\downarrow$
 t

Fermat theorem:

p prime and $\exists c$ where $p \nmid c$ p doesn't divide c

$$c^{p-1} = 1 \pmod{p}$$

Generalized Fermat theorem

$$\text{If } (c; p_1 \cdot p_2) = 1$$

primes

← azt jelenti, hogy se p_1 se p_2 nem osztója c -nek

$$c^{(p_1-1)(p_2-1)} = 1 \pmod{p_1 \cdot p_2}$$

Euler theorem

$$p_1, \dots, p_n$$

$$(c; p_1 \cdot p_2 \cdot \dots \cdot p_n) = 1$$

$$\Phi(m) = (p_1 - 1)(p_2 - 1) \dots (p_n - 1)$$

$$c^{\Phi(m)} = 1 \pmod{p_1 \cdot p_2 \cdot \dots \cdot p_n}$$

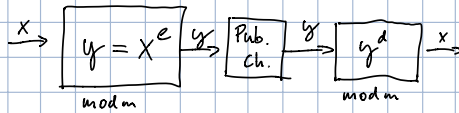
RSA algorithm

pick two very large prime #s: $p_1, p_2 \rightarrow m = p_1 \cdot p_2 ; \Phi(m) = (p_1 - 1)(p_2 - 1)$

$$(e, \Phi(m)) = 1 ; d = \bar{e} \pmod{\Phi(m)} \rightarrow d \cdot e = q \cdot \Phi(m) + 1$$

$$k^{(p)} = (m, e)$$

$$k^{(s)} = (d, p_1, p_2)$$



Wisszafejtés

$$d \cdot e = q \cdot \Phi(m) + 1$$

we only know m

$$m = p_1 \cdot p_2 \rightarrow \text{prime factorize } (O(\infty))$$

$$\Phi(m) = (p_1 - 1)(p_2 - 1)$$

Uniqueness:

$$(x, m) = 1$$

$$y^d = (x^e)^d = x^{de} = x^{q \cdot \Phi(m) + 1} = x^{q \cdot \Phi(m)} \cdot x = (x^{\Phi(m)})^q \cdot x = 1^q \cdot x = x$$

$$(x, m) > 1$$

Common modulus

p_1, p_2 the same for the whole user population

$$(e_A, \Phi(m)) = 1 \rightarrow d_A = e_A^{-1} \bmod \Phi(m)$$

$$(e_B, \Phi(m)) = 1 \rightarrow d_B = e_B^{-1} \bmod \Phi(m)$$

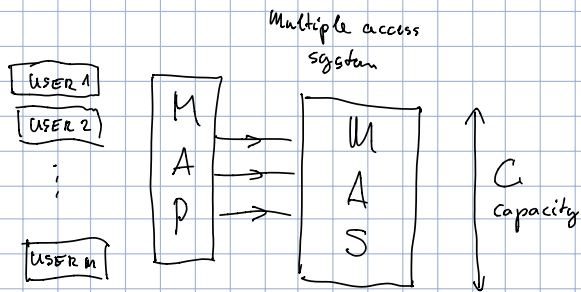
$\vdots$

$$(e_V, \Phi(m)) = 1 \rightarrow d_V = e_V^{-1} \bmod \Phi(m)$$

Breaking the system

$$(e_A, e_B) = 1 \quad \Phi(m) \rightarrow \exists s, t: s \cdot e_A + t \cdot e_B = 1$$

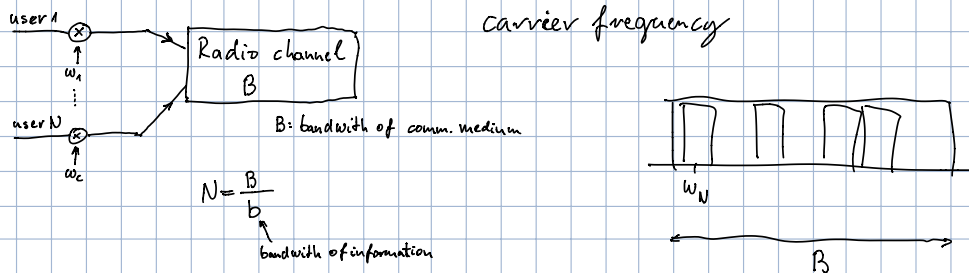
MULTIUSER COMMUNICATION



MAP: max utilization - max kichasnerktsig

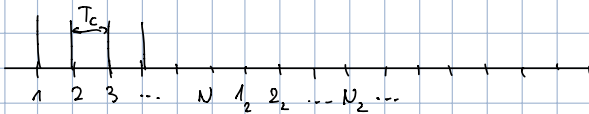
condition: multiple user interference $\leq \epsilon$

1G solution (first generation) FDMA - Frequency

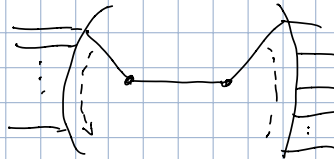


2G TDMA - Time division multiple acc.

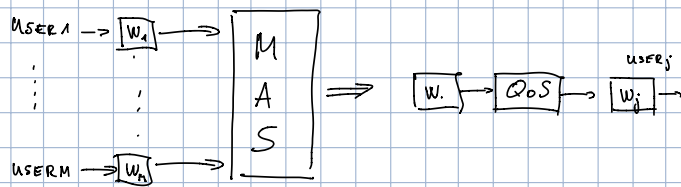
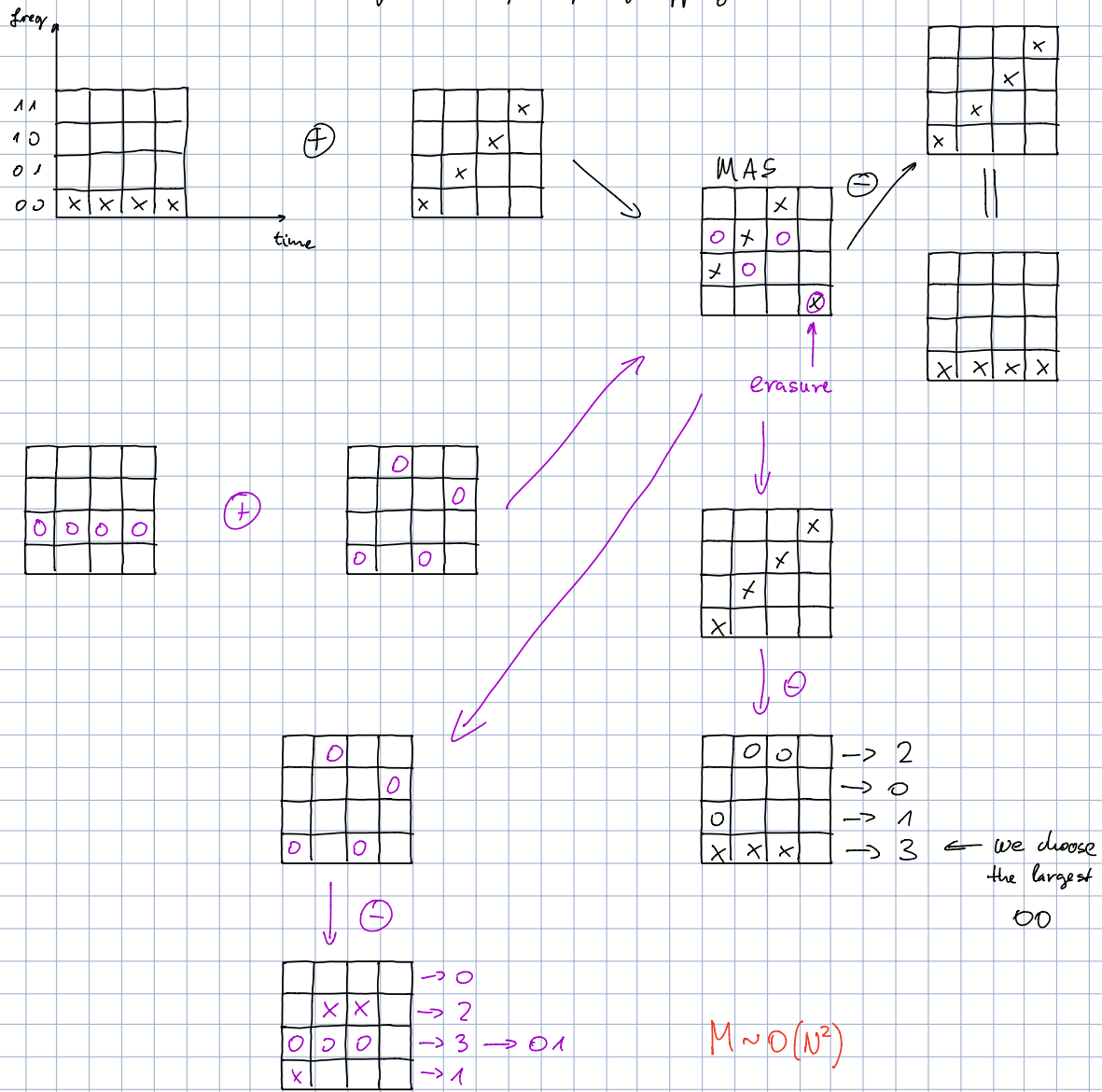
$$T_f = \frac{1}{8k\theta_2} \text{ (beszednel)}$$



$$N = \frac{T_f}{T_c} = \frac{f_c}{f_f}$$



3G Code Division Multiple Access / Frequency Hopping

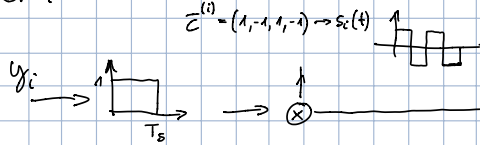


W: window
 - time
 - frequency
 - code
 → CDMA / FH - Freq. hopping
 → CDMA / DS - direct sequence

2016. 11. 30.

USER POPULATION: $1, \dots, M$; $y_i \in \{-1, 1\}$, $s_i(t)$; $i = 1 \dots M$
 $\uparrow$
signature

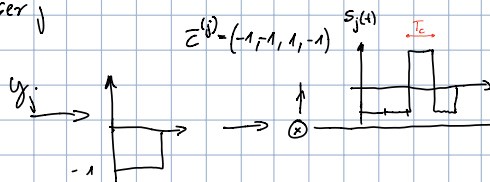
User i



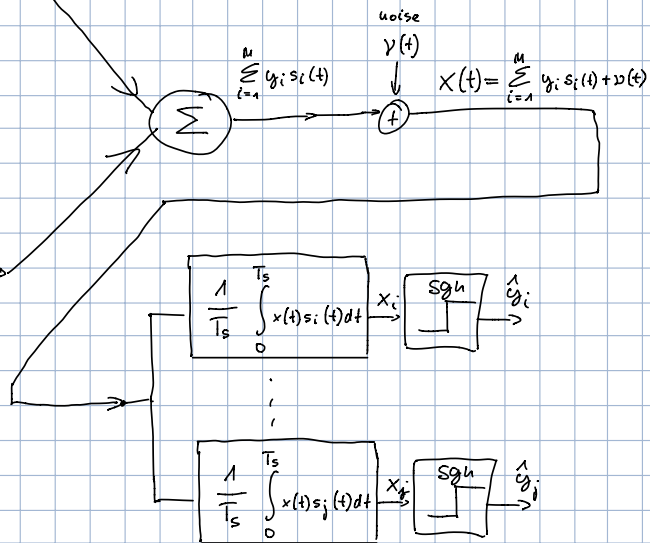
T_s : symbol time

$\frac{1}{T_s}$: data speed

User j



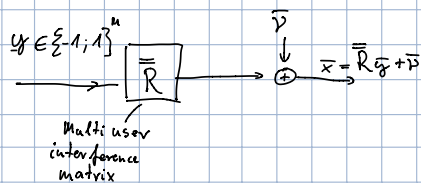
$$N := \frac{T_s}{T_c} = \dim(\mathbf{z}) = \frac{\text{service}}{\text{technology}}$$



$$X_l := \frac{1}{T_s} \int_0^{T_s} x(t) s_l(t) dt = \frac{1}{T_s} \int_0^{T_s} \left(\sum_{i=1}^M y_i s_i(t) v(t) \right) \cdot s_l(t) dt = \sum_{i=1}^M y_i \frac{1}{T_s} \int_0^{T_s} s_l(t) s_i(t) dt + \frac{1}{T_s} \int_0^{T_s} v(t) s_l(t) dt$$

$$R := \frac{1}{T_s} \int_0^{T_s} s_l(t) \cdot s_i(t) dt = \frac{1}{N} \sum_{m=1}^N c_m^{(l)} \cdot c_m^{(i)} = \frac{1}{N} \mathbf{c}^{(l)T} \mathbf{c}^{(i)}$$

$$v_l := \frac{1}{T_s} \int_0^{T_s} s_l(t) v(t) dt; \quad X_l = \sum_{i=1}^M R_{li} \cdot y_i + v_l \quad l = 1, \dots, N$$



$$X_e = R_{ee} y_e + \underbrace{\sum_{i \neq e} R_{ei} y_i}_{\text{Multi User Interference}} + v_e$$

MUI: Multi User Interference

$$\hat{y}_e = \text{sgn} \left\{ R_{ee} y_e + \sum_{i \neq e} R_{ei} y_i + v_e \right\}$$



We have to set $R_{ei} = 0 = \frac{1}{N} \sum_{n=1}^N c_n^{(e)} \cdot c_n^{(i)} = \frac{1}{N} \bar{c}^{(e)T} \bar{c}^{(i)}$

$$M \leq N = \frac{T_s}{T_c}$$

felhasznált széma kódvektorok hossza
 ortogonális vektorok száma miatt

$\frac{4}{6}$ (high speed service)
 1000 mobile comm. (voice)

Walsh - Hadamard construct
making $\{-1, 1\}$ orthogonal vectors

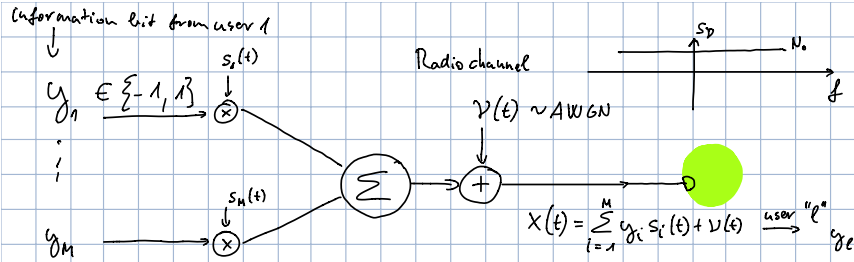
$$\bar{H}_{2 \times 2} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$H_{4 \times 4} = \begin{pmatrix} H_{2 \times 2} & H_{2 \times 2} \\ H_{2 \times 2} & -H_{2 \times 2} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix}$$

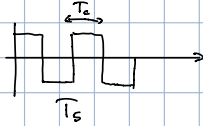
$$\bar{H}_{(N+1) \times (N+1)} = \begin{pmatrix} \bar{H}_{N \times N} & \bar{H}_{N \times N} \\ \bar{H}_{N \times N} & \bar{H}_{N \times N} \end{pmatrix} \quad M \leq 2^L$$

??
 $M > N$ Challenge of b3G

2016.12.07.



$$\bar{c}^{(i)} = (1, -1, 1, -1)$$



$$\frac{1}{T_s} \text{ data speed}$$

$$N = \frac{T_s}{T_c} = \frac{\text{service}}{\text{HW}}$$

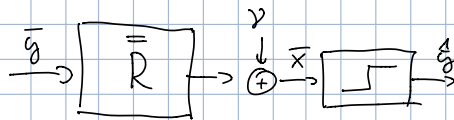
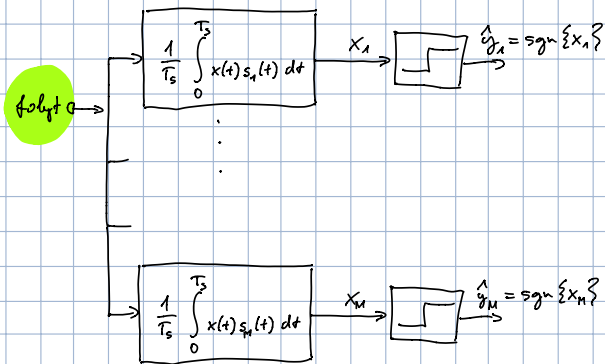
more precise form:

$$x(t) = \sum_{i=1}^M y_i \overset{\text{attenuation factor}}{a_i} s_i(t - \overset{\text{delay}}{T_i}) + v(t)$$

most precise form:

$$x(t) = \sum_{i=1}^M y_i h_i(t) * s_i(t) + v(t)$$

$$\int_{-\infty}^{\infty} h_i(t-\tau) s_i(\tau) d\tau$$



$$\bar{x} = \bar{R} \bar{y} + \bar{v}; \quad R_{ei} = \frac{1}{T_s} \int_0^T s_e(t) s_i(t) dt = \frac{1}{N} \sum_{m=1}^N c_m^{(e)} \cdot c_m^{(i)} = \frac{1}{N} \bar{c}^{(e)T} \bar{c}^{(i)}$$

$$R_{ee} = 1 \quad \forall e$$

$$x_e = \sum_{i=1}^M R_{ei} y_i + v_e = R_{ee} y_e + \sum_{i \neq e} R_{ei} y_i + v_e = y_e + \underbrace{\sum_{i \neq e} R_{ei} y_i}_{\text{Multi User Interference (MUI)}} + v_e$$

$$R_{\ell i} = 0 \quad \forall i = 1, \dots, M \quad i \neq \ell$$

$\bar{c}^{(\ell)}$, $\ell = 1, \dots, M$ - orthogonal vector set

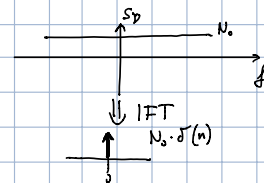
$$\dim(\bar{c}^{(\ell)}) = N \geq M$$

$$N = \frac{T_s}{T_c} = \begin{cases} 1000 \\ 4 \div 6 \end{cases}$$

$$M \stackrel{?}{>} N$$

$$\bar{y}, \bar{y}_\ell = \frac{1}{T_s} \int_0^{T_s} \bar{y}(t) \cdot s_\ell(t) dt, \quad \ell = 1, \dots, M \rightarrow \bar{y} \sim \mathcal{N}(\bar{y}, \bar{R})$$

AWGN Kovariancia matrix



$$K_{\ell i} = E(y_\ell y_i) = E\left(\frac{1}{T_s} \int_0^{T_s} y(t) \cdot s_\ell(t) dt \cdot \frac{1}{T_s} \int_0^{T_s} y(\tau) \cdot s_i(\tau) d\tau\right) = E\left(\frac{1}{T_s^2} \int_0^{T_s} \int_0^{T_s} y(t) y(\tau) s_\ell(t) s_i(\tau) dt d\tau\right) =$$

$$= \frac{1}{T_s^2} \int_0^{T_s} \int_0^{T_s} E(y(t) y(\tau)) s_\ell(t) s_i(\tau) dt d\tau = \frac{1}{T_s^2} \int_0^T R_y(t-\tau) s_\ell(t) s_i(\tau) dt d\tau = \frac{N_0}{T_s} \cdot \frac{1}{T_s} \int_0^{T_s} s_\ell(t) \cdot s_i(t) dt = N_0' \cdot R_{\ell i}$$

$\bar{R}$ is non diagonal because $M > N$

$$p_y(\bar{u}) = \frac{1}{\sqrt{(2\pi)^M \det(N_0' \bar{R})}} e^{-\frac{1}{2} \bar{u}^T N_0'^{-1} \bar{R}^{-1} \bar{u}}$$

Receive $\bar{x}$

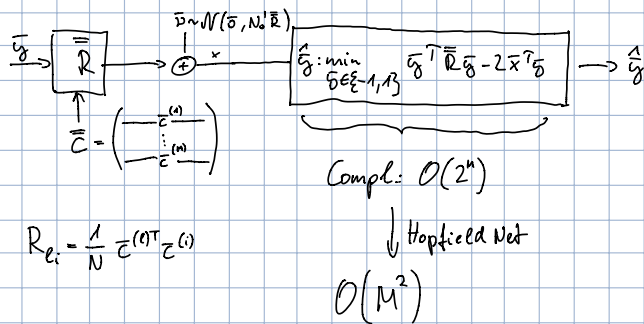
$$\hat{\bar{y}} = \max_{\bar{y} \in \{-1, 1\}^M} p(\bar{y} | \bar{x}) \sim \max_{\bar{y} \in \{-1, 1\}^M} \frac{p(\bar{x} | \bar{y})^{\frac{1}{2N}}}{p(\bar{x})} \sim \max_{\bar{y} \in \{-1, 1\}^M} p(\bar{x} | \bar{y})$$

$$\hat{\bar{y}} = \max_{\bar{y} \in \{-1, 1\}^M} \frac{1}{\sqrt{(2\pi)^M \det(N_0' \bar{R})}} \cdot e^{-\frac{1}{2} (\bar{x} - \bar{R} \bar{y})^T N_0'^{-1} \bar{R}^{-1} (\bar{x} - \bar{R} \bar{y})} \sim \max_{\bar{y} \in \{-1, 1\}^M} -\frac{1}{2} (\bar{x} - \bar{R} \bar{y})^T N_0'^{-1} \bar{R}^{-1} (\bar{x} - \bar{R} \bar{y}) \sim$$

$$\sim \min_{\bar{y} \in \{-1, 1\}^M} (\bar{x} - \bar{R} \bar{y})^T \bar{R}^{-1} (\bar{x} - \bar{R} \bar{y}) \sim \min_{\bar{y} \in \{-1, 1\}^M} \underbrace{\bar{x}^T \bar{R}^{-1} \bar{x}}_{\text{I}} - \underbrace{\bar{y}^T \bar{R}^T \bar{R}^{-1} \bar{x}}_{\text{I}} - \underbrace{\bar{x}^T \bar{R}^{-1} \bar{R} \bar{y}}_{\text{I}} + \underbrace{\bar{y}^T \bar{R}^T \bar{R}^{-1} \bar{R} \bar{y}}_{\text{I}} \sim$$

$$\sim \min_{\bar{y} \in \{-1, 1\}^M} \bar{y}^T \bar{R} \bar{y} - 2 \bar{x}^T \bar{y} \quad \text{Multi-User Detection Problem}$$

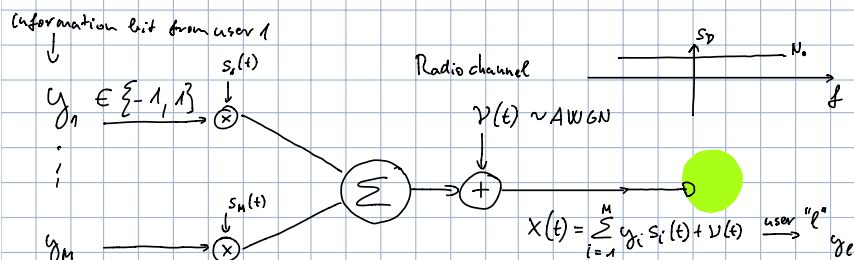
b36 beyond third generation



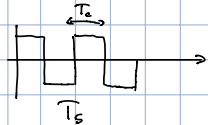
$$R_{ij} = \frac{1}{N} \bar{c}^{(i)T} \bar{c}^{(j)}$$

2016.12.09.

CDMA/DS



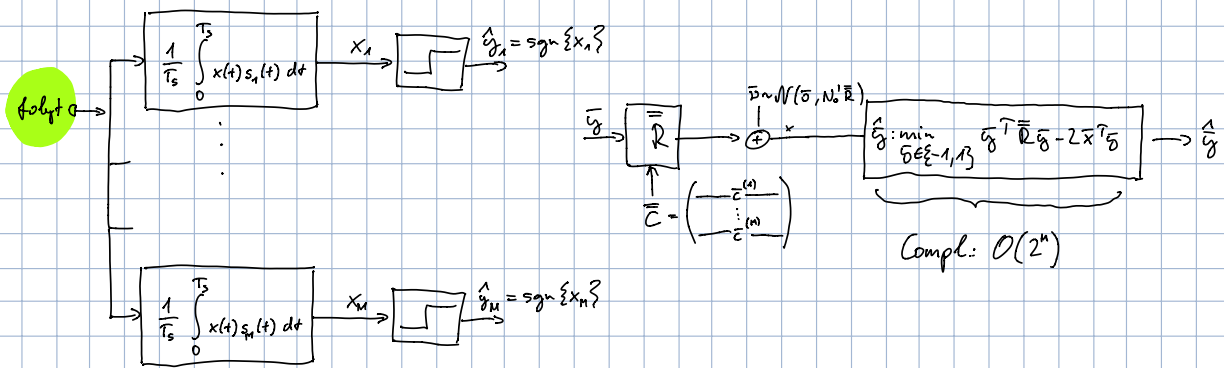
$$\bar{c}^{(i)} = (1, -1, 1, -1)$$



$\frac{1}{T_s}$ data speed

$$N = \frac{T_s}{T_c} = \frac{\text{service}}{\text{HW}}$$

length of the code vector



How to

binary matrix $\bar{C}_{M \times N} = \begin{pmatrix} \bar{c}^{(1)} \\ \bar{c}^{(2)} \\ \vdots \\ \bar{c}^{(M)} \end{pmatrix};$

bit error probability:

$$P_b = \Psi(\bar{C});$$

$$P_b = g(\bar{C})$$

$$\bar{C}_{opt} : \min_{\bar{C}} P_b$$

$\Rightarrow$ optimization problem

$$x_c = \sum_{i=1}^M R_{ci} y_i + v_c = R_{cc} y_c + \sum_{i \neq c} R_{ci} y_i + v_c = y_c + \sum_{i \neq c} R_{ci} y_i + v_c$$

sign / threshold detector:

$$\hat{y}_c = \text{sgn}\{x_c\}$$

$$P_b = P(\hat{y}_c \neq y_c) = P(\hat{y}_c = -1 | y_c = -1) \cdot \frac{1}{2} + P(\hat{y}_c = 1 | y_c = 1) \cdot \frac{1}{2} = \frac{1}{2} [P(\text{sign}\{-1 + \sum_{i \neq c} R_{ci} y_i + v_c\} = -1) + P(\text{sign}\{1 + \sum_{i \neq c} R_{ci} y_i + v_c\} = -1)] =$$

$$= \frac{1}{2} [P(-1 + \sum_{i \neq c} R_{ci} y_i + v_c > 0) + P(1 + \sum_{i \neq c} R_{ci} y_i + v_c < 0)] = \frac{1}{2} \frac{1}{2^{M-1}} \sum_{\bar{z} \in \{-1, 1\}^{M-1}} P(-1 + \sum_{i \neq c} R_{ci} z_i + v_c > 0) + P(1 + \sum_{i \neq c} R_{ci} z_i + v_c < 0)] =$$

$$\bar{y}' = (y_1, \dots, y_{c-1}, y_{c+1}, \dots, y_M)$$

$$P(\bar{y}' = \bar{z}) = \frac{1}{2^{M-1}}$$

$$\bar{z} \in \{-1, 1\}^{M-1}$$

$$= \frac{1}{2^M} \sum_{\bar{z} \in \{-1, 1\}^{M-1}} \{P(v_c > 1 - \sum_{i \neq c} R_{ci} z_i) + P(v_c < -1 - \sum_{i \neq c} R_{ci} z_i)\} =$$

Gaussian random variable

according to symmetry

$$= \frac{1}{2^M} \sum_{\bar{z} \in \{-1, 1\}^{M-1}} \left\{ \Phi\left(\frac{-1 + \sum_{i \neq c} R_{ci} z_i}{\sqrt{N_b}}\right) + \Phi\left(\frac{-1 - \sum_{i \neq c} R_{ci} z_i}{\sqrt{N_b}}\right) \right\}$$

$$P_b = \frac{1}{2^M} \sum_{\bar{z} \in \{-1, 1\}^{M-1}} \left\{ \Phi\left(\frac{-1 + \sum_{i \neq c} \bar{c}^{(i)*} \bar{c}^{(c)} \cdot z_i}{\sqrt{N_b}}\right) + \Phi\left(\frac{-1 - \sum_{i \neq c} \bar{c}^{(i)*} \bar{c}^{(c)} \cdot z_i}{\sqrt{N_b}}\right) \right\} \Rightarrow \bar{C}_{opt} : \min_{\bar{C}} P_b$$

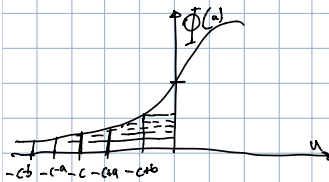
$$\bar{C} \rightarrow O(2^{M \cdot N} \cdot 2^{M-1}) \sim O(2^{M(M-1) \cdot N})$$

Lemma: $0 < a < b < c$

$$\Phi\left(\frac{-c+b}{\sigma}\right) + \Phi\left(\frac{-c-b}{\sigma}\right) > \Phi\left(\frac{-c+a}{\sigma}\right) + \Phi\left(\frac{-c-a}{\sigma}\right)$$

rearrangement:

$$\Phi\left(\frac{-c+b}{\sigma}\right) - \Phi\left(\frac{-c+a}{\sigma}\right) > \Phi\left(\frac{-c-a}{\sigma}\right) - \Phi\left(\frac{-c-b}{\sigma}\right)$$



$$b := \sum_{i \neq c} |R_{ci}| \geq \sum_{i \neq c} R_{ci} z_i$$

$$a := \sum_{i \neq c} R_{ci} z_i$$

$$c := 1$$



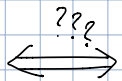
poly: $\leq \frac{1}{2^M} \sum_{\bar{z} \in \{-1, 1\}^{M-1}} \left\{ \Phi\left(\frac{-1 + \sum_{i \neq c} |R_{ci}|}{\sqrt{N_b}}\right) + \Phi\left(\frac{-1 - \sum_{i \neq c} |R_{ci}|}{\sqrt{N_b}}\right) \right\}$

$$P_b \leq \frac{1}{2} \left\{ \Phi\left(\frac{-1 + \sum_{i \neq c} |\bar{c}^{(i)*} \bar{c}^{(c)}|}{\sqrt{N_b}}\right) + \Phi\left(\frac{-1 - \sum_{i \neq c} |\bar{c}^{(i)*} \bar{c}^{(c)}|}{\sqrt{N_b}}\right) \right\}$$



Resources:

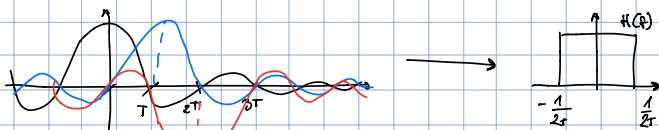
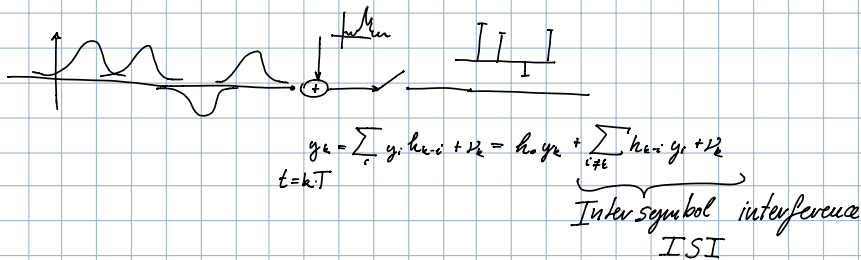
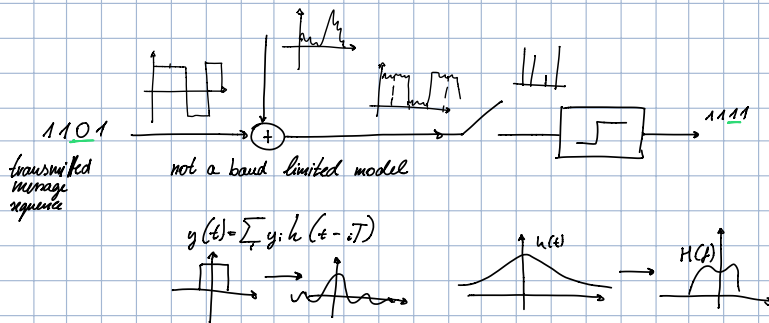
- bandwidth
- transmission power



QoS requirement:

- Data speed
- P_b

QoS $\stackrel{??}{=} \Psi(\text{Resources})$



clock synchronization circuitry - nehéz pontosan időzíteni

$$h_\ell = S_{\ell 0} = \begin{cases} 1 & \ell = 0 \\ 0 & \ell \neq 0 \end{cases}$$

$$\text{DFT } H(e^{j\omega T}) = \sum_\ell h_\ell e^{-j\omega T \ell} = 1$$

$$H(e^{j\omega T}) = \sum_\ell H(f + \frac{\ell}{T}) \rightarrow \text{must be constant} \Rightarrow \text{no ISI}$$

