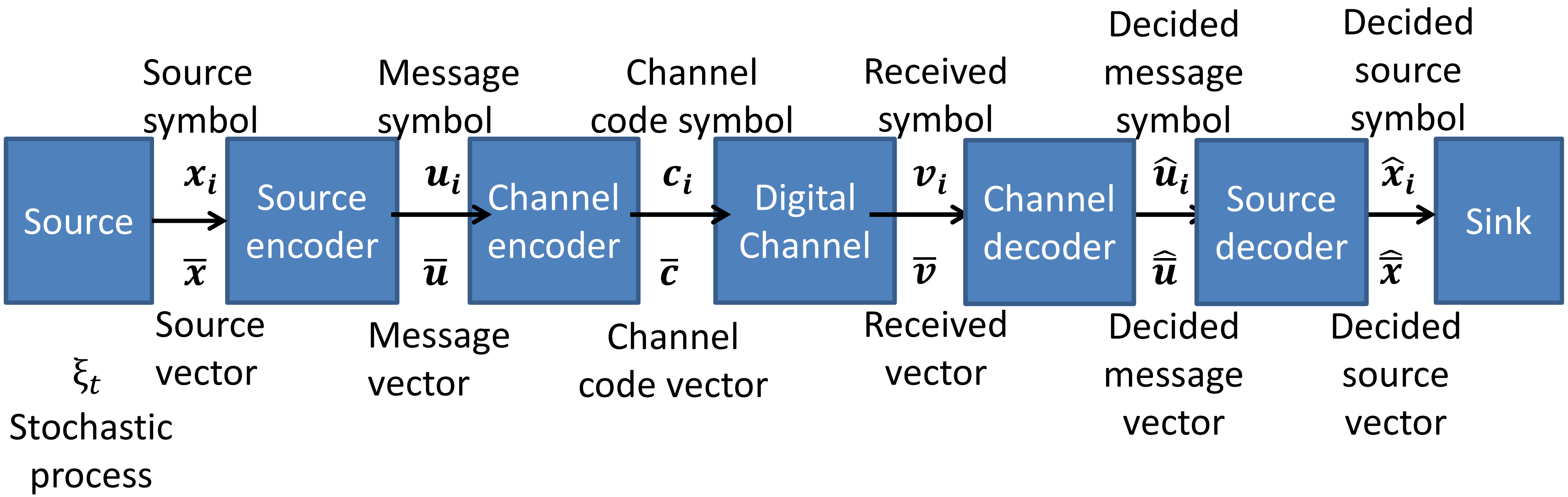
Űrkommunikáció
Space Communication
2023/6.

Channel Coding
Error correction coding



Channel encoding rule

$$\Omega(\overline{\boldsymbol{u}}) = \overline{\boldsymbol{c}}$$

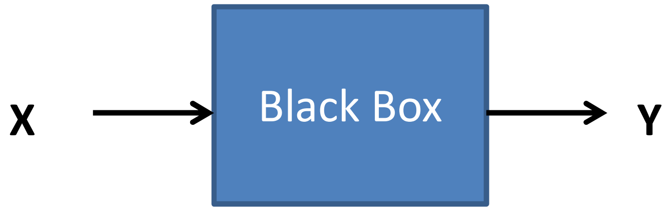
Decoding in 2 steps

$\mathrm{D}(\overline{\boldsymbol{v}})=\hat{\overline{\boldsymbol{c}}}$ 1. Decision

$\Omega^{-1}(\hat{\overline{\boldsymbol{c}}}\,) = \hat{\overline{\boldsymbol{u}}}$ 2. Invers operation

Digital transmission channel

Input X and output Y are discrete random variables



How many information can we gather about X by observing Y?

- a-posteriori Entropy
- Mutual Information

Definition: a-posteriori Entropy [bit/symbol, Shannon/symbol]

$$H(X|Y) = E\{I(x|y)\} = \sum_{x} \sum_{y} p(x,y) ld \frac{1}{p(x|y)}$$

Digital transmission channel

Definition: Mutual Information of two random events

$$\mathrm{I}(x_i; y_j) = ld\ \frac{p(x_i \mid y_j)}{p(x_i)} = ld\ \frac{p(y_j \mid x_i)}{p(y_j)}\ [bit, Shannon]$$

Using Bayes's theorem:

$$ld\ \frac{p(x_i \mid y_j)}{p(x_i)} = ld\ \frac{p(x_i \mid y_j)p(y_j)}{p(x_i)p(y_j)} = ld\ \frac{p(x_i,y_j)}{p(x_i)p(y_j)} = ld\ \frac{p(y_j \mid x_i)p(x_i)}{p(x_i)p(y_j)} = ld\ \frac{p(y_j \mid x_i)}{p(y_j)}$$

Definition: Average mutual information $[bit/symbol], [\,Shannon/symbol]$

$$\mathrm{I}(X\,; Y) = E\{\mathrm{I}(x_i; y_j)\} = \sum_x \sum_y p(x_i, y_j)\mathrm{I}(x_i; y_j) =$$

$$= \sum_x \sum_y p(x_i, y_j) ld \frac{p(x_i,y_j)}{p(x_i)p(y_j)} = D(p(x,y) \mid\mid p(x) \cdot p(y)) =$$

$$= \sum_x \sum_y p(x_i, y_j) ld \frac{p(x_i \mid y_j)}{p(x_i)} = \sum_x \sum_y p(x_i, y_j) \left[ld \frac{1}{p(x_i)} - ld \frac{1}{p(x_i \mid y_j)} \right] =$$

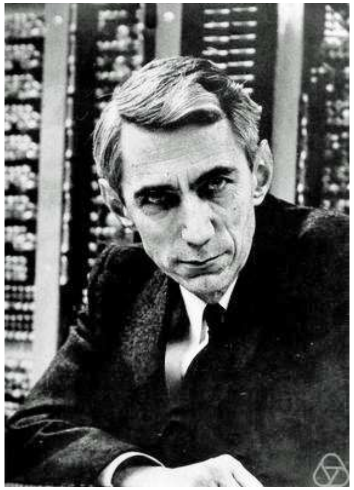
$$= \sum_x \sum_y p(x_i, y_j) ld \frac{1}{p(x_i)} - \sum_x \sum_y p(x_i, y_j) ld \frac{1}{p(x_i \mid y_j)} =$$

$$= \qquad \mathrm{H(X)} \qquad - \qquad \mathrm{H(X \mid Y)}$$

Channel Capacity

$$I(X\,;Y) = H(X) - H(X\,|Y) = H(Y) - H(Y\mid X) = D(p(x,y)\,||\;p(x)\cdot p(y))$$

Definition: Channel capacity $\left[\frac{Shannon}{channel\ use}\right], \left[\frac{bit}{channel\ use}\right]$, *e.g.* [bit/sec]



Forrás: ww.techzibits.com

$$C = \max_{p(x)} I(X\,;Y) =$$
$$= \max_{p(x)} [H(X) - H(X|Y)] =$$
$$= \max_{p(x)} [H(Y) - H(Y|\mathrm{X})] =$$
$$= \max_{p(x)} D\big(p(x_i, y_j)\,||\;p(x_i)\cdot p(y_j)\big)$$

Channel capacity is bounded

$$0 \le C \le \max_{p(x)} [H(X)] = H_0 = ld\ n$$

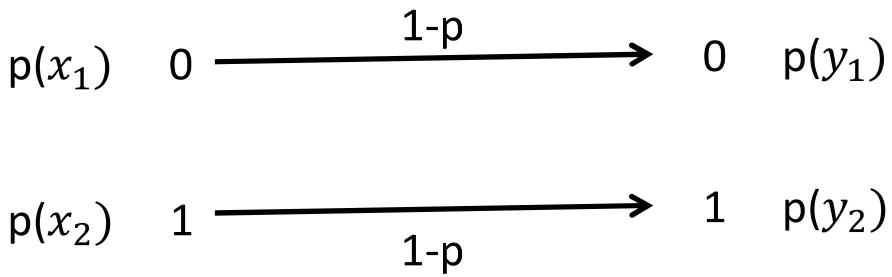
X and Y are independent:
H(X)=H(X|Y)

Error free channel: $y_i = x_i$
p(x_i|y_i) = 1
H(X|Y) =0

Ideal Binary Channel

- Ideal: No parameter, that is only one parameter: error probability *p=0*
- Binary in- and output:

$X = \{x_1, x_2\}$ e.g: $\{0,1\}$ $\quad$ $\mathrm{Y} = \{y_1, y_2\}$ e.g: $\{0,1\}$



- How much is the capacity?

Starting form here:

$$C_{ideal\ binary}(\text{p=0}) = \max_{p(x)} [H(X) - H(X|\mathrm{Y})] = H_0(X) = ld2 = 1\ [\frac{bit}{channel\ use}]$$

Capacity of BSC

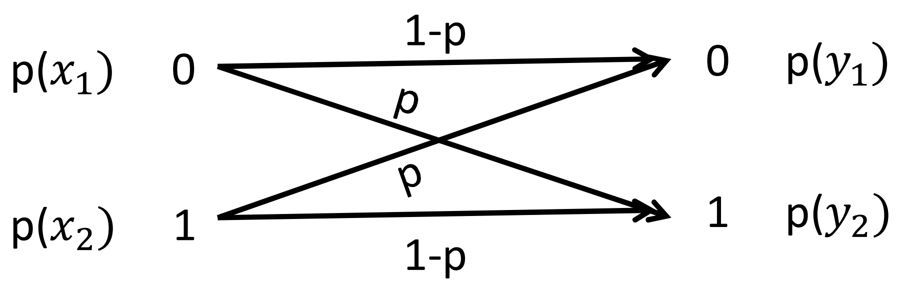
BSC: **B**inary **S**ymmetric **C**hannel

- One parameter: error probability *p*

- Binary in- and output:

$X = \{x_1, x_2\}$ pl.: $\{0,1\}$ $\quad$ $\mathrm{Y} = \{y_1, y_2\}$ pl.: $\{0,1\}$

- Symmetric:



- How much is the capacity?

Starting now form here: $\quad C_{BSC}(\mathrm{p}) = \max_{p(x)} [H(Y) - H(Y|\mathrm{X})]$

Capacity of BSC

C_{BSC}(p) = $\max\limits_{p(x)} [H(Y) - H(Y|\mathrm{X})]$ [bit/channel use]

- H(Y) maximal if Y is uniformly distributed: $\mathrm{p}(y_1) = \mathrm{p}(y_2) = 1/2$

And then H(Y)=1 [bit/binary symbol]

- In the case of BSC the output is uniformly distributed for example when the input is a such:

$\mathrm{p}(x_1) = \mathrm{p}(x_2) = \frac{1}{2}$, then:

$\mathrm{p}(y_1) = \mathrm{p}(x_1) \cdot (1-p) + \mathrm{p}(x_2) \cdot p = \mathrm{p}(x_1) - \mathrm{p}(x_1) \cdot p + (1 - \mathrm{p}(x_1)) \cdot p =$
$= \mathrm{p}(x_1) - 2 \cdot \mathrm{p}(x_1) \cdot p + p = \mathrm{p}(x_1) - p + p = 1/2$

- H(Y|X)=$\sum_x \sum_y p(x,y) ld \frac{1}{p(y|x)} = \sum_x \sum_y p(x) p(y|x) ld \frac{1}{p(y|x)} =$

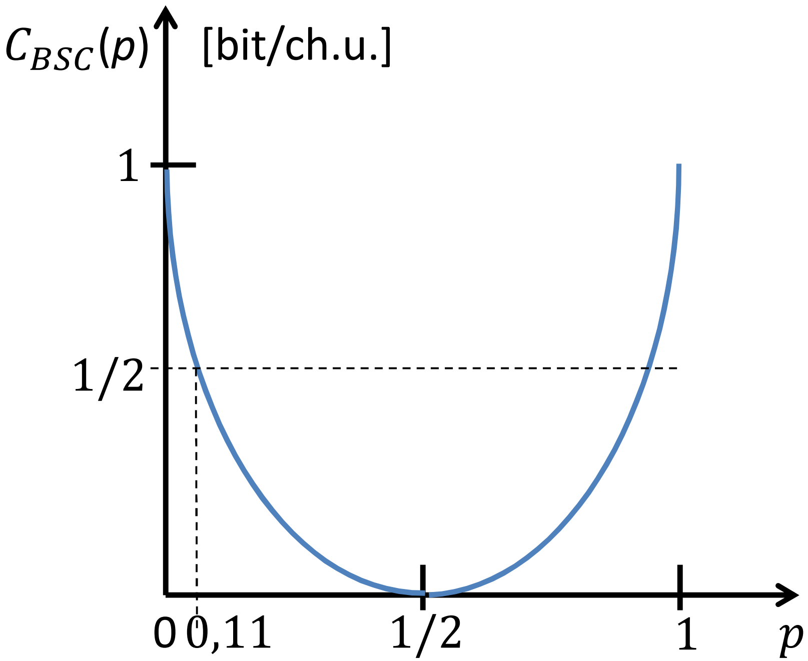
$$= \sum_x p(x) \sum_y p(y|x) ld \frac{1}{p(y|x)}$$

$$\mathrm{p}(x_1) \cdot \left[(1-p) \cdot ld \frac{1}{(1-p)} + p \cdot ld \frac{1}{p}\right] + \mathrm{p}(x_2) \cdot \left[p \cdot ld \frac{1}{p} + (1-p) \cdot ld \frac{1}{(1-p)}\right] =$$

$$\underbrace{\left(\mathrm{p}(x_1) + \mathrm{p}(x_2)\right)}_{1} \cdot \underbrace{\left[(1-p) \cdot ld \frac{1}{(1-p)} + p \cdot ld \frac{1}{p}\right]}_{\text{h(p) binary entropy function}}$$

Capacity of BSC

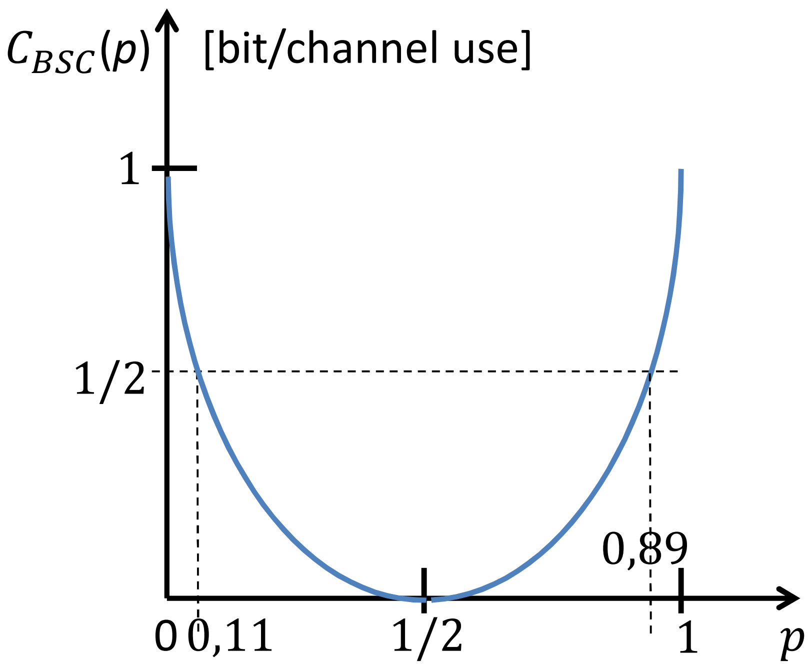
C_{BSC}(p) = $\max_{p(x)} [H(Y) - H(Y|\mathrm{X})] = 1 - h(p)$ [bit/channel use]



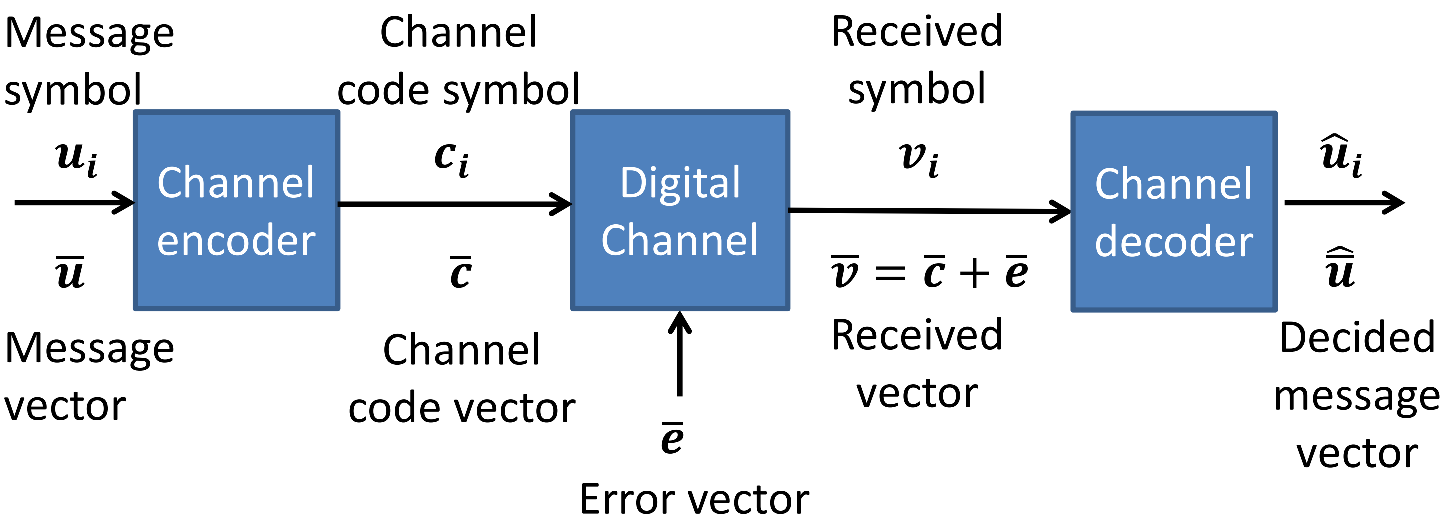
Shannon's Channel coding theorem, 1948

- If H(X) < C then exists Ω (X)=$\tilde{X}$ transformation (coding, modulation, method) so that $P_e \rightarrow 0$ until H ($\tilde{X}$) < C holds.

- An other formulation (case of Block coding):
  A vector (block) of K message symbols extended to a vector of N code symbols
  $P_e \rightarrow 0$, until $\lim_{K\rightarrow\infty} {}^{K}/_{N} < C$ remains valid.



Coding, Construction of codes



Channel encoding rule

$$\Omega(\overline{\boldsymbol{u}}) = \overline{\boldsymbol{c}}$$

Decoding in 2 steps

$\mathrm{D}(\overline{\boldsymbol{v}})=\widehat{\overline{\boldsymbol{c}}}$ 1. Decision

$\Omega^{-1}(\widehat{\overline{\boldsymbol{c}}}\,) = \widehat{\overline{\boldsymbol{u}}}$ 2. Invers operation

Coding rule: Mutually obvious transformation of Message space into Code space

Coding, Construction of codes

Message space: $\overline{U} = \{\overline{u_i}\}$

Message vector: $\overline{u_i} = [u_1, u_2, \dots, u_k, \dots, u_K]$

Message symbol: $u_k = \{0, 1, 2, \dots, r-1\}$

Dimension K, r-ary vector space

r^K possible message vector

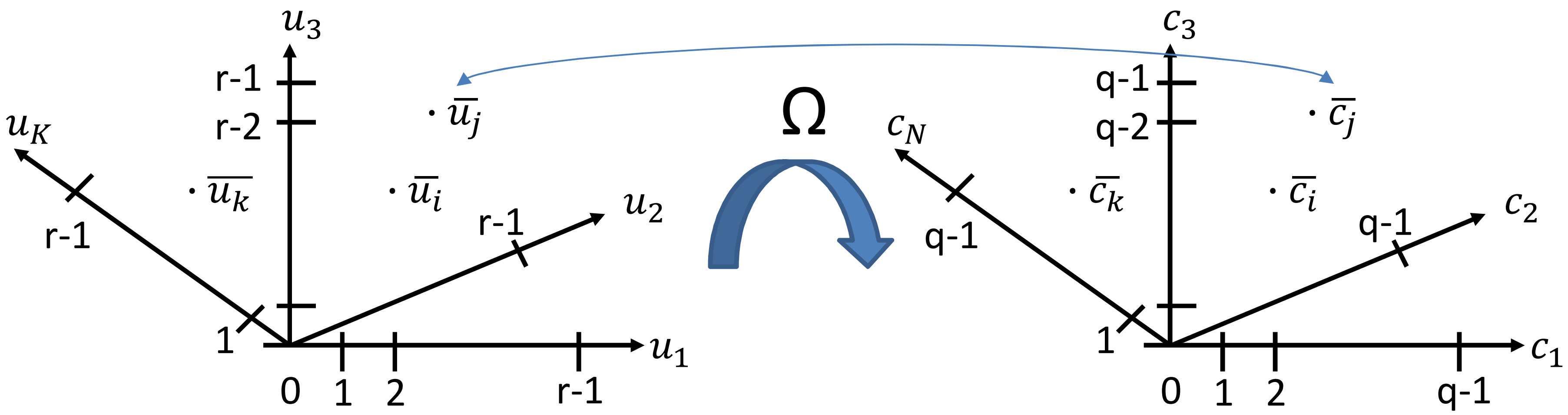
Code space: $\bar{C} = \{\overline{c_i}\}$

Code vector: $\overline{c_i} = [c_1, c_2, \dots, c_n, \dots, c_N]$

Code symbol: $c_n = \{0, 1, 2, \dots, q-1\}$

Dimension N, q-ary vector space

q^N possible code vector



Decoding

Error vector: $\bar{e} = [e_1, e_2, \dots, e_n, \dots, e_N]$

Example: two error events: $\bar{e} = \left[0, 0, \dots, e_i, \dots, e_j, \dots, 0, \dots, 0\right]$ if the events happens at position i and j, and the symbol values of the events are e_i and e_j.

Four unknown (positions and values) should be determined.

Example for „deleted" error type: $\overline{e_{del}} = \left[0, 0, \dots, *_j, \dots, 0, \dots, 0\right]$.

The position is known.

Received vector: $\bar{v} = \bar{c} + \bar{e} = [v_1, v_2, \dots, v_n, \dots, v_N] = [c_1 + e_1, \dots, c_n + e_n, \dots, c_N + e_N]$

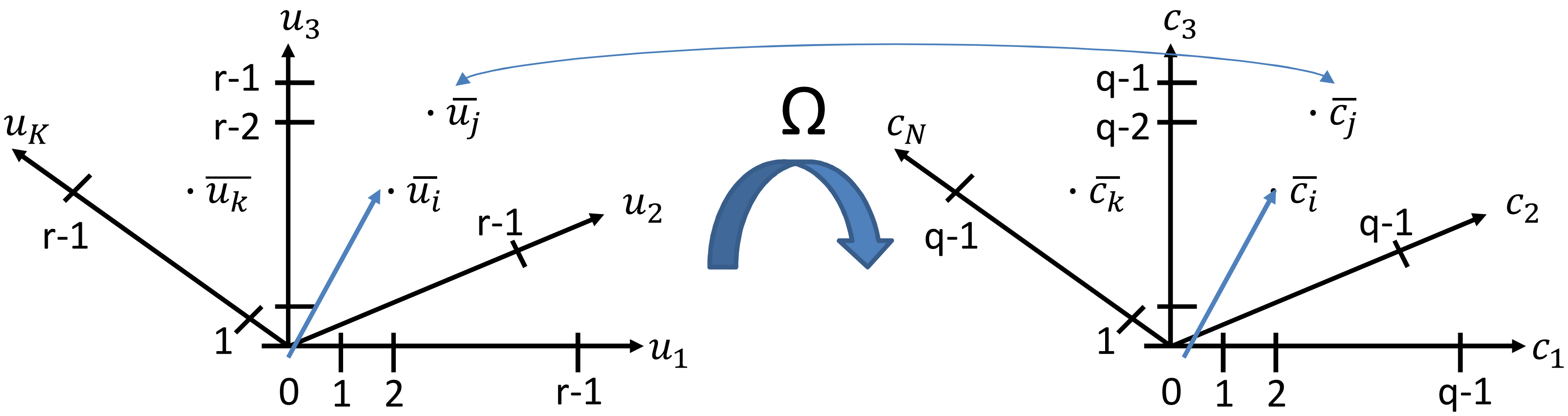
Based on the received vector: $\bar{v}$ Decoding in 2 steps

1. Decision $\mathrm{D}(\boldsymbol{\bar{v}}) = \boldsymbol{\hat{\bar{c}}}$

2. Invers operation $\Omega^{-1}(\boldsymbol{\hat{\bar{c}}}) = \boldsymbol{\hat{\bar{u}}}$

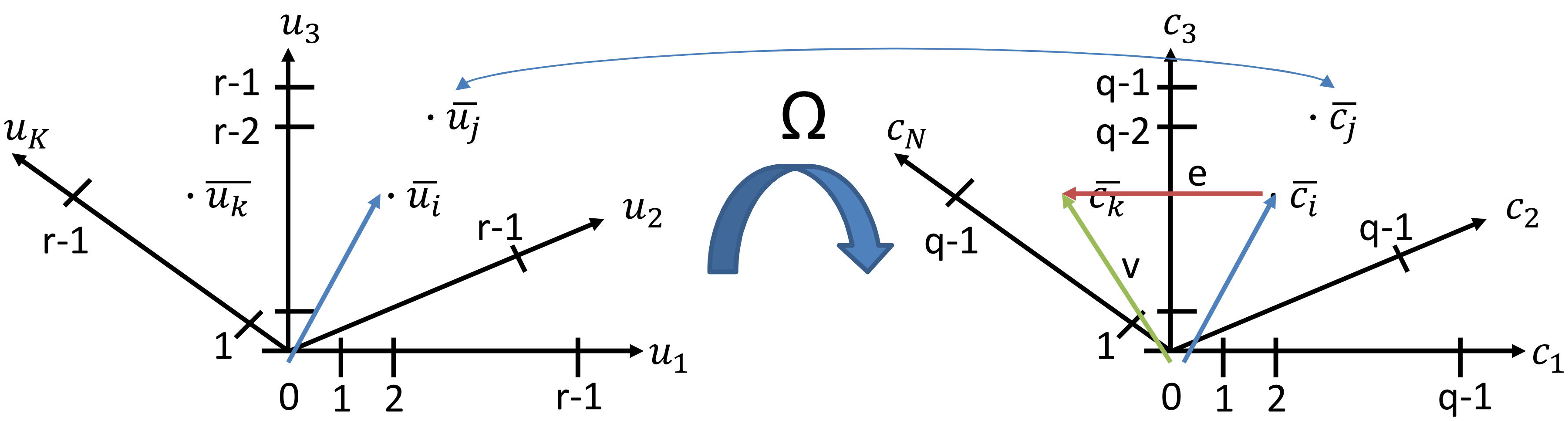
- Trivial: $\bar{v} = \overline{c_i}$
- Unsolvable: $\bar{v} = \overline{c_j} \neq \overline{c_i}$ that we sent
- Solvable with the possibility of wrong decision: $\bar{v} \neq \overline{c_i}\ \forall i$

Trivial



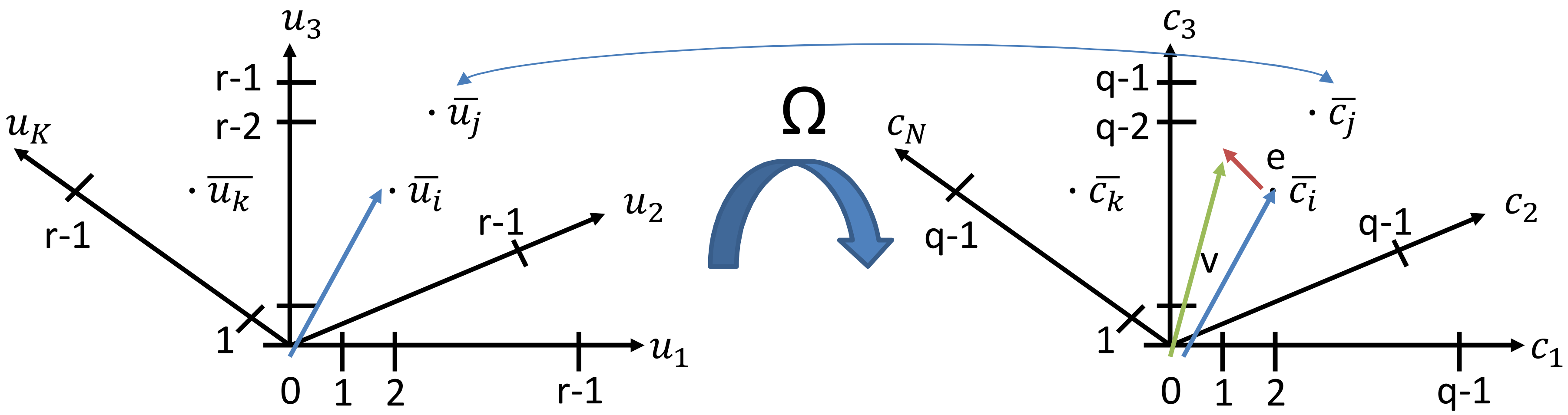
u_3
r-1
r-2
u_K
$\cdot\ \overline{u_j}$
$\cdot\ \overline{u_k}$
$\cdot\ \overline{u_i}$
r-1
r-1
u_2
1
0
1
2
r-1
u_1
Ω
c_3
q-1
q-2
c_N
$\cdot\ \overline{c_j}$
$\cdot\ \overline{c_k}$
$\cdot\ \overline{c_i}$
q-1
q-1
c_2
1
0
1
2
q-1
c_1

Unsolvable



u_3
r-1
r-2
u_K
$\cdot\ \overline{u_j}$
$\cdot\ \overline{u_k}$
$\cdot\ \overline{u_i}$
r-1
r-1
u_2
1
u_1
0
1
2
r-1
Ω
c_3
q-1
q-2
c_N
$\cdot\ \overline{c_j}$
e
$\overline{c_k}$
$\cdot\ \overline{c_i}$
q-1
q-1
c_2
v
1
c_1
0
1
2
q-1

Solvable with the possibility of wrong decision



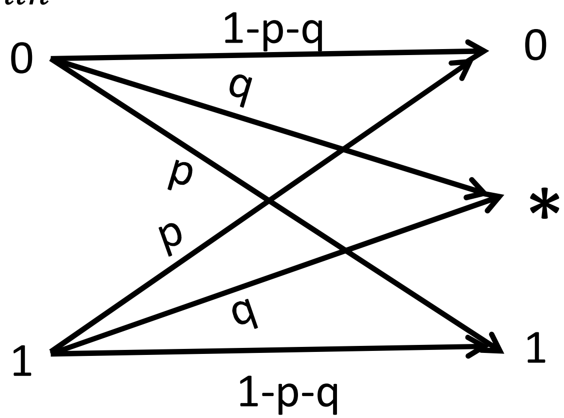
Coding, Construction of codes

Definitions for calculation in vector space:

- Hamming distance: $d(\overline{c_i}, \overline{c_j}) = \sum_{n=1}^{N} \chi\left(c_{i_n} \neq c_{j_n}\right)$
- Code distance: $d_{min} = \min_{i,j \neq i} \{d(\overline{c_i}, \overline{c_j})\}$
- Code weight: $w = \min_{i, \overline{c_i} \neq 0} \{\sum_{n=1}^{N} \chi\left(c_{i_n} \neq 0\right)\}$

Type of errors:

- Number of detectable error: $t_{det} < d_{min}$, $t_{det_{max}} = d_{min} - 1$
- Number of correctable errors: $t_{corr} = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor$
- Number of „deleted” type errors: $t_{del} = d_{min} - 1$

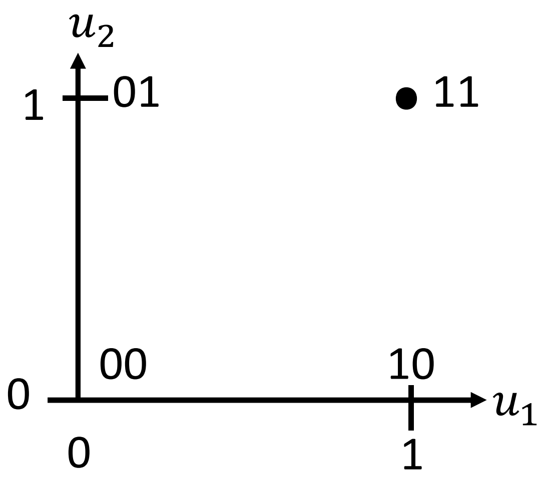


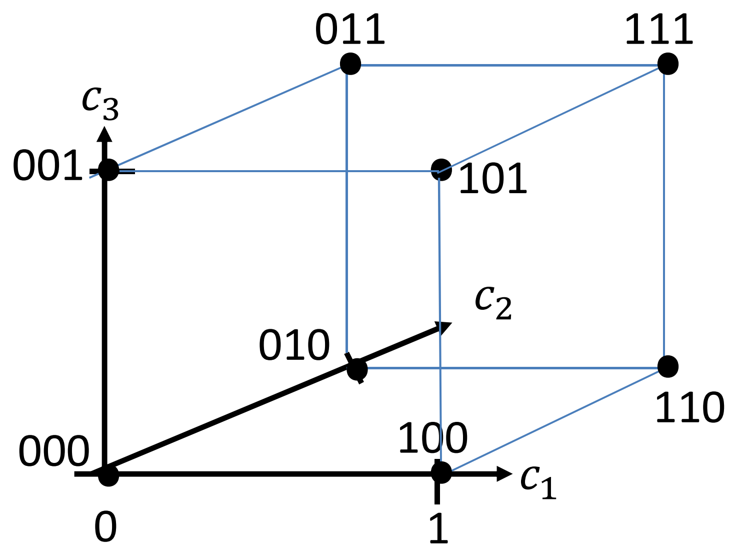
Basic types of Channel coding (error correction coding)

- Block codes (N,K,q): Hamming, Cyclic, Reed-Solomon, etc.
- Convolutional coding (Trellis codes, Viterbi coding/decoding)

Let us start the encoding heuristically

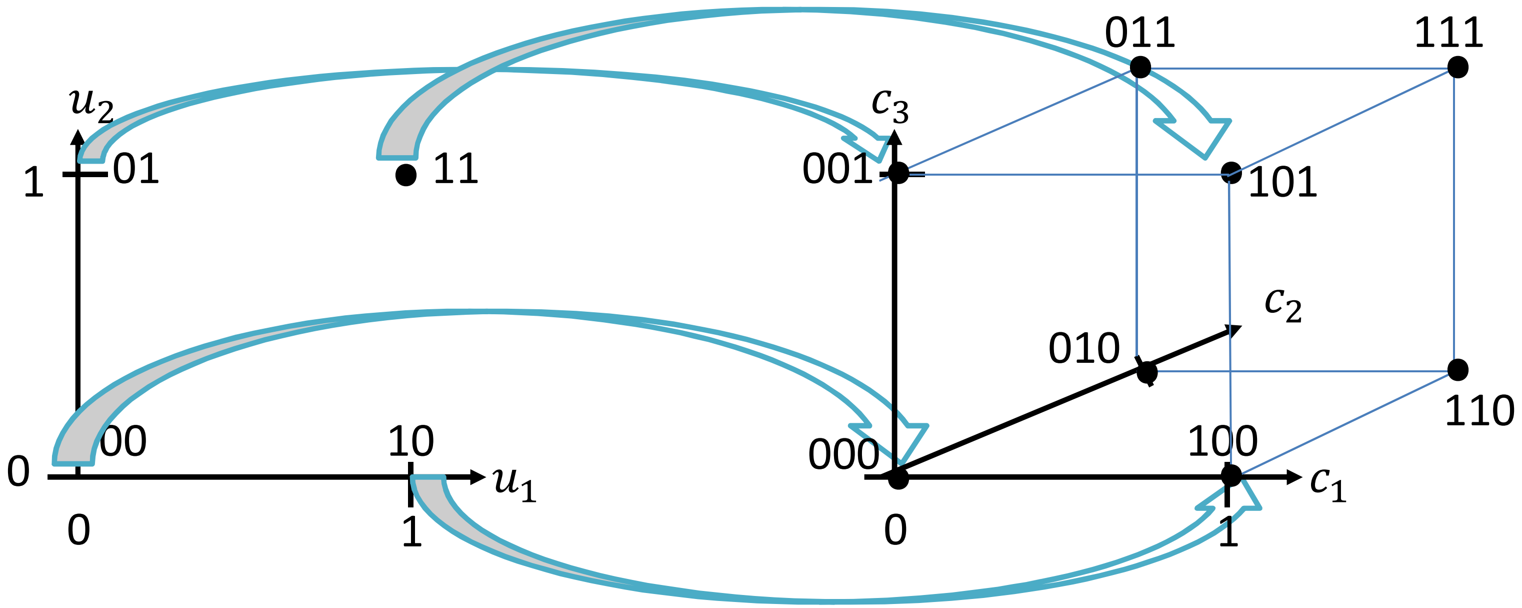
(N=3, K=2, q=r=2) binary message and vector space, +1 redundant binary symbol





Construction heuristically

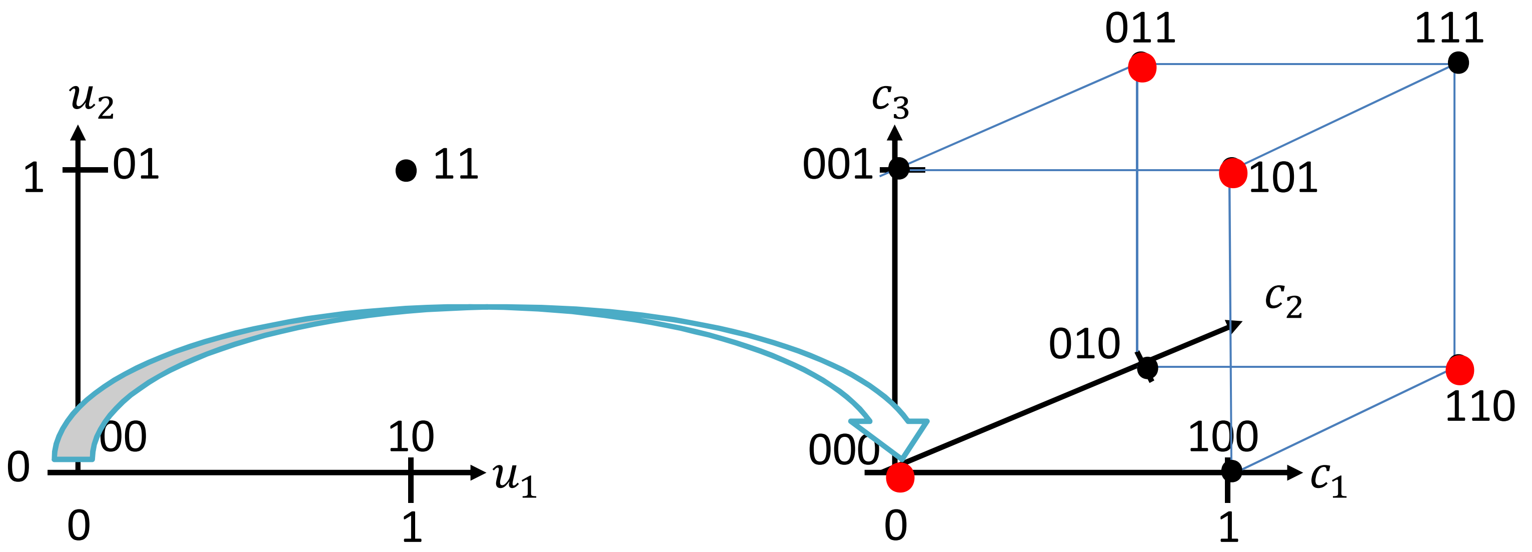
(N=3, K=2, q=r=2) binary message and vector space, +1 redundant binary symbol



Although mutually obvious transformation, not appropriate, because the distances remaining the same

Construction heuristically

(N=3, K=2, q=r=2) binary message and vector space, +1 redundant binary symbol

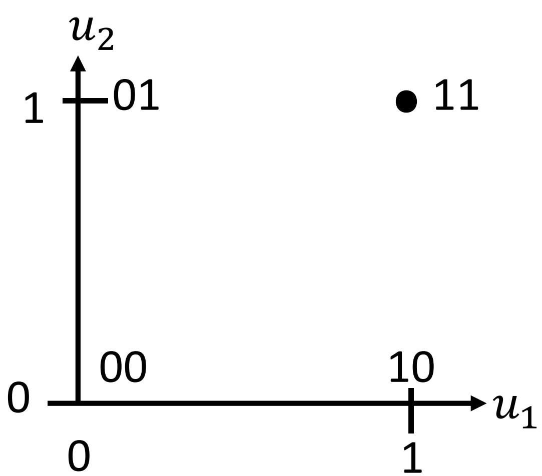


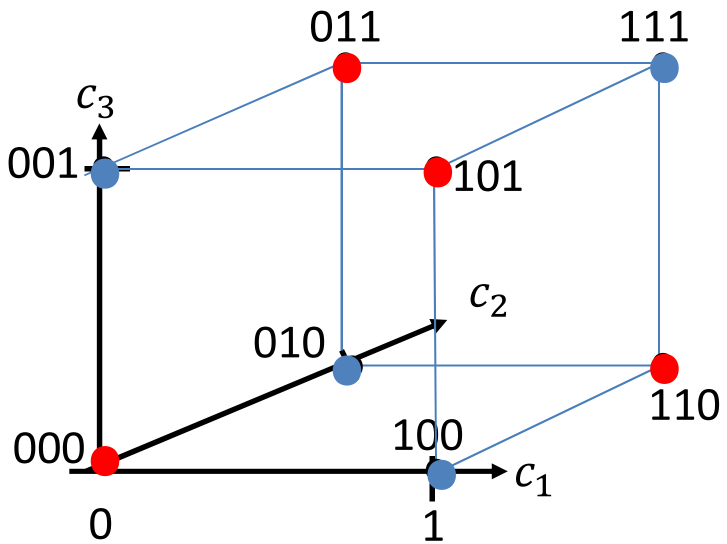
Mutually obvious transformation, appropriate, because increasing distances, dmin=2
(Parity check coding, cyclic coding heuristically!)

Message vector		Code vector		
0	0	0	0	0
1	0	1	0	1
0	1	0	1	1
1	1	1	1	0

Construction heuristically

(N=3, K=2, q=r=2)   binary message and vector space, +1 redundant binary symbol





Vector set represented by blue dots could detect one error because mutually obvious and increasing distances, dmin=2, however, they are not a linear subspace (see later) of the vector space, therefore not appropriate for calculations.
(cyclic but not parity check vector set!)

Example: correcting one „delete" error, red also

Another transformation	Message vector	Code vector	
	0 0	1 0 0	$c=$1 0 0
	1 0	0 0 1	$v=$* 0 0
	0 1	0 1 0	$v=$1 * 0
	1 1	1 1 1	$v=$1 0 *

Algebraic code construction rules

Singleton bound for (N,K,q=r) block codes: $M \leq q^{N-d_{min}+1}$

The number of possible code vectors (therefore message vectors) related to the code attributes d_{min}, N and q.

Proof:

K dimensional q-ary space: $M \leq q^K$, max. $M = q^K$ => $d_{min} = 1$,

Extended to N dimensional: K->N => q^{N-K} times more point, max. $d_{min} = N - K + 1$

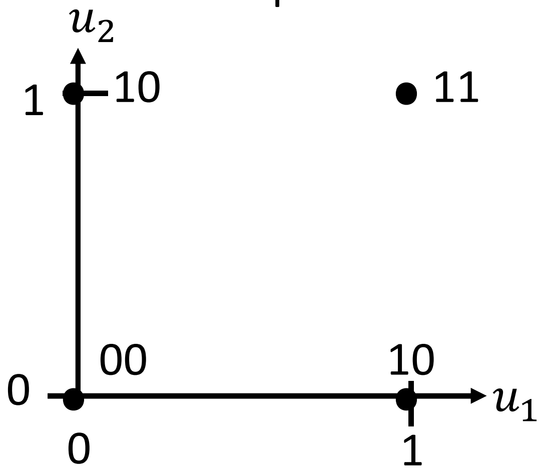
d_{min} ≤N-K+1, $K \leq N - d_{min} + 1$, $M \leq q^K \leq q^{N-d_{min}+1}$

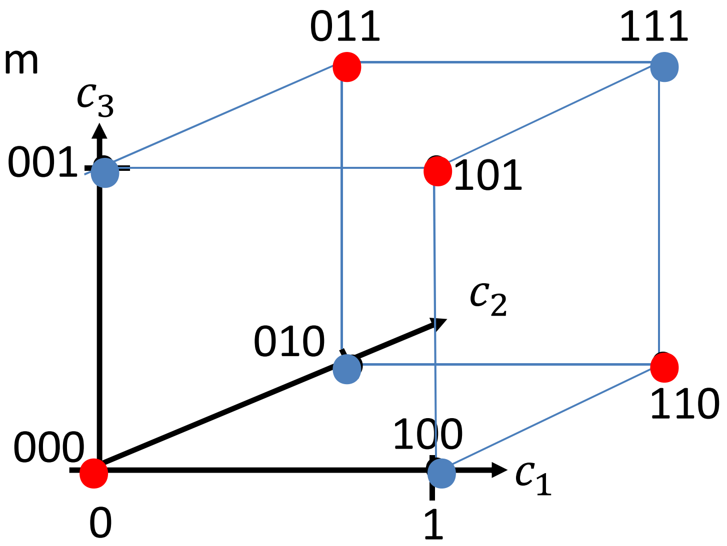
MDS code (**M**aximum **D**istance **S**eparable): $M = q^{N-d_{min}+1}$

Or equivalently: $K = N - d_{min} + 1$ or $d_{min} = N - K + 1$

MDS example : (N=3,K=2,q=2), $d_{min} = 2$

$M = q^{N-d_{min}+1} = 4$ possible messages maximum





Code construction rules cont.

Hamming bound (sphere-packing bound): t_{corr} required, (N,K,q=r)?

Determine the number of points in a decision subspace around a valid code vector; the size of decision subspace,
needed for correction of $t_{corr} = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor$ errors

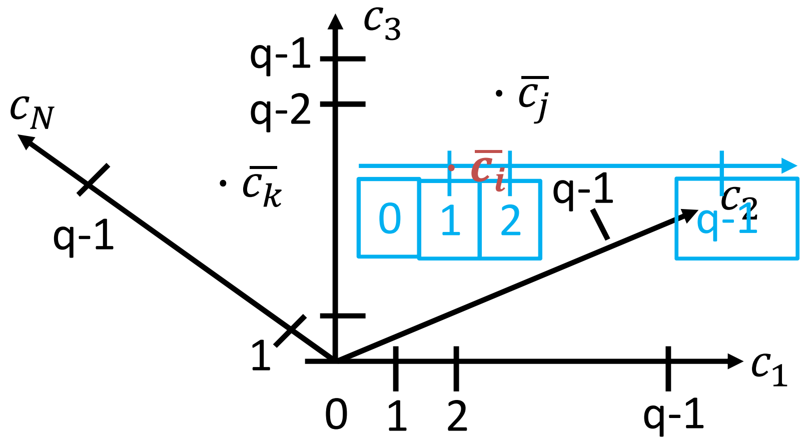
$$1 + N \cdot (q-1) + \binom{N}{2} \cdot (q-1)^2 + \cdots + \binom{N}{t_{corr}} \cdot (q-1)^{t_{corr}} = \sum_{i=0}^{t_{corr}} \binom{N}{i} \cdot (q-1)^i$$

max. $M = q^K$

$q^K \cdot \sum_{i=0}^{t_{corr}} \binom{N}{i} \cdot (q-1)^i \leq q^N$;

Hamming bound:

$$\sum_{i=0}^{t_{corr}} \binom{N}{i} \cdot (q-1)^i \leq q^{N-K}$$



Binary case: $\sum_{i=0}^{t_{corr}} \binom{N}{i} \leq 2^{N-K}$

Perfect code:

$$\sum_{i=0}^{t_{corr}} \binom{N}{i} \cdot (q-1)^i = q^{N-K}$$

Code construction rules cont.

Hamming bound (sphere-packing bound): t_{corr} required, (N,K,q=r)?

HOWEVER: Not only the size but also the form of the decision subspaces counts.

Example: Perfect; $t_{corr} = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor = 2; q = 2;$

$$1 + N \cdot (q-1) + \binom{N}{2} \cdot (q-1)^2 = q^{N-K}$$

$$1 + N + \frac{N \cdot (N-1)}{2} = 2^{N-K}$$

N=90 and K=78 solves the equation, however, doesn't exists in a 90 dimensional space 302231454903657293676544 (=2^{78}) portion of disjoint decision subspaces so that every one vector of the space (2^{90} piece) is part of one and only one decision subspace.
The Hamming bound is just a bound for the size.

=>

=>

Examples: Hamming bound, perfect code

Hamming code (N,K,q): Perfect code, that capable to correct maximum one error, and detect max. two errors. In the practice mostly binary, however non-binary Hamming codes also exists.

$t_{corr} = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor = 1;$ => $d_{min} \geq 3$

For $t_{corr} = 1$ the Hamming bound is performed perfectly with the (N,K,q) set $\forall\, m \geq 2$

$$\left(N = \frac{q^m - 1}{q-1},\ K = N - m,\ q\right) \qquad \text{(m=1 => K=0 !!!)}$$

Proof:

$$1 + N \cdot (q-1) = q^{N-K} = q^m\,; \qquad N \cdot (q-1) = q^m - 1\,; \qquad N = \frac{q^m - 1}{q-1}$$

MDS => $d_{min} = N - K + 1 = 3$

	q=2			q=3			q=5		
m	N	K	$R_c = K/N$	N	K	$R_c = K/N$	N	K	$R_c = K/N$
2	3	1	1/3	4	2	1/2	6	4	2/3
3	7	4	0,57	13	10	0,77	31	28	0,9
4	15	11	0,73	40	36	0,9	156	152	0,97
5	31	26	0,84	121	116	0,96	⋮	⋮	⋮

Example: MDS, perfect code

Hamming (N=3, K=1, q=r=2) code

MDS: $d_{min} = N - K + 1 = 3; \quad t_{corr} = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor = 1$

Perfect: $1 + N \cdot (q-1) = q^{N-K} = 1 + 3 = 2^2 = 4$ elements in a decision subspace

Message: 0 or 1

Code vector: 000 or 111

That is a simple repetition coding!