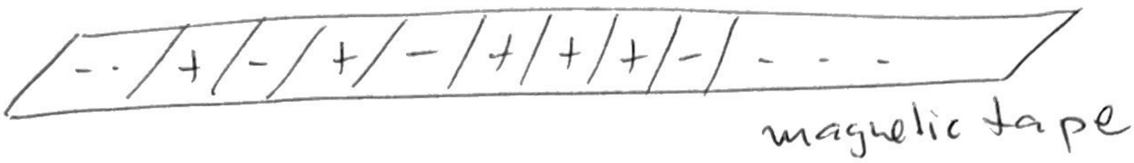
LN $\boxed{XI}$ a

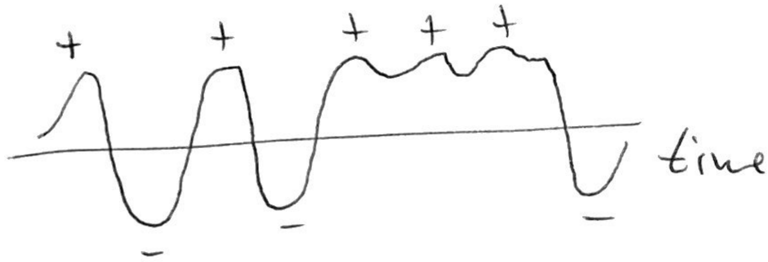
§5.4) data storage



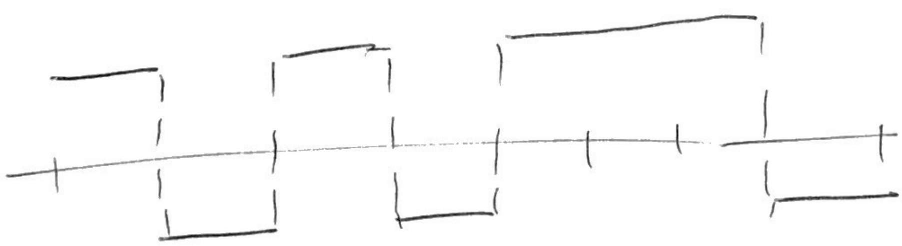
... 1 0 1 0 1 1 1 0 ...

$$\begin{cases} + \sim 1 \\ - \sim 0 \end{cases}$$

Reading: in practice



it is not ideal



Easy to get (reading) errors

One would like to be able to detect whether an error occured and correct the errors.

To do so one included redundancy in the coding of data

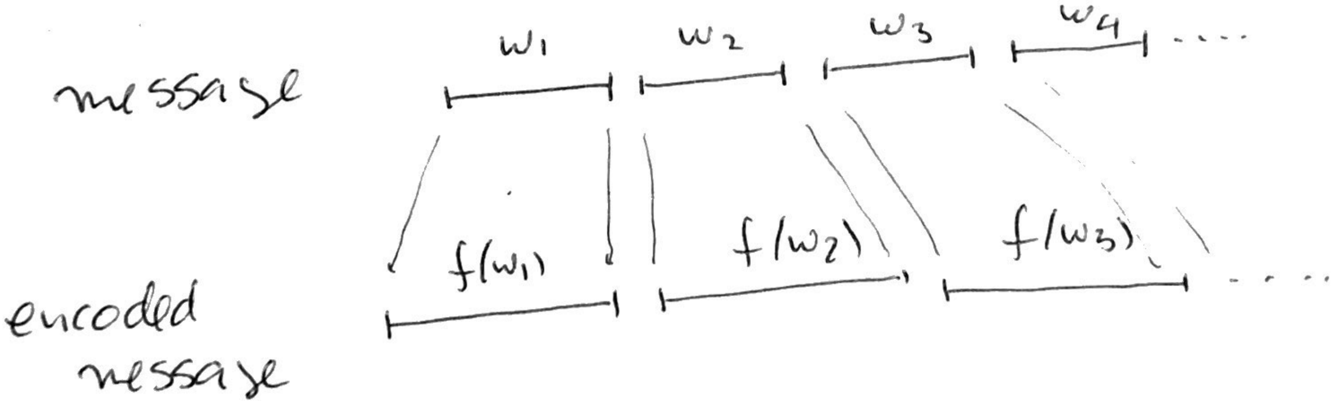
—//—

Letters $\{0,1\} = \mathbb{Z}_2, +, \times$ $\quad B = \mathbb{Z}_2$.

A word: $w \in B^n$ $\quad$ length$(w) = n$

concatenation $w \in B^n$ $v \in B^m$ $\quad wv \in B^{n+m}$.

coding $f: B^m \xrightarrow[1-1]{} B^n$ $\quad n \geq m$



usually the encoded message is longer.

$n > m$

Ex) $f: \mathbb{B}^m \longrightarrow \mathbb{B}^{m+1}$

$f(w) = wx \qquad x = \sum w_i = \# 1$

parity.

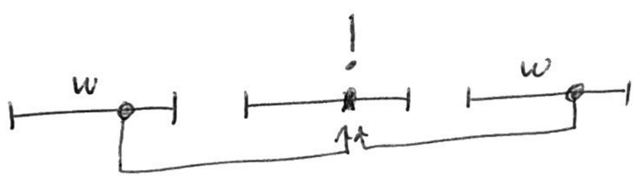
f can detect 1 error.

Ex) $f: \mathbb{B}^m \longrightarrow \mathbb{B}^{3m}$

$f(w) = www$

f can detect 2 error.

correct 1 error



Of course there are more interesting codes

—//—

$w \in \mathbb{B}^m \quad$ weight $(w) = \# 1$ in w.

distance between $v, w \in \mathbb{B}^m$

$d(v,w) = \text{weight}(v-w)$

$= \#$ of places where the words are different.

$$\left.\begin{matrix} 011011 \\ 010001 \end{matrix}\right) \quad d = 2$$

Given a coding $f: \mathcal{B}^m \longrightarrow \mathcal{B}^n$

$f(\mathcal{B}^m)$ codewords.

We like when

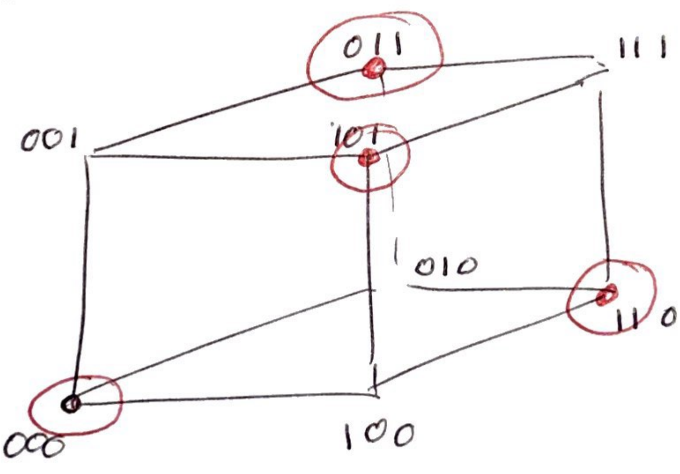
$$\boxed{\min_{v \neq w} d\big(f(v), f(w)\big)}$$

is large.

Ex] $f: \mathcal{B}^2 \longrightarrow \mathcal{B}^3$

$f(w) = wx \qquad x = \sum w_i$ parity.

$\mathcal{B}^3$



$00 \longmapsto 000$

$01 \longmapsto 011$

$10 \longmapsto 101$

$11 \longmapsto 110$

$$d(f(v), f(w)) = 2 \qquad v \neq w$$

Thm $f: B^m \longrightarrow B^n$

f can detect k errors $\Longleftrightarrow$

$$d(f(v), f(w)) \geq k+1$$

Pf) Let $\tilde{w}$ be the received ~~word~~ version of the code word $f(w)$.

To detect an error means

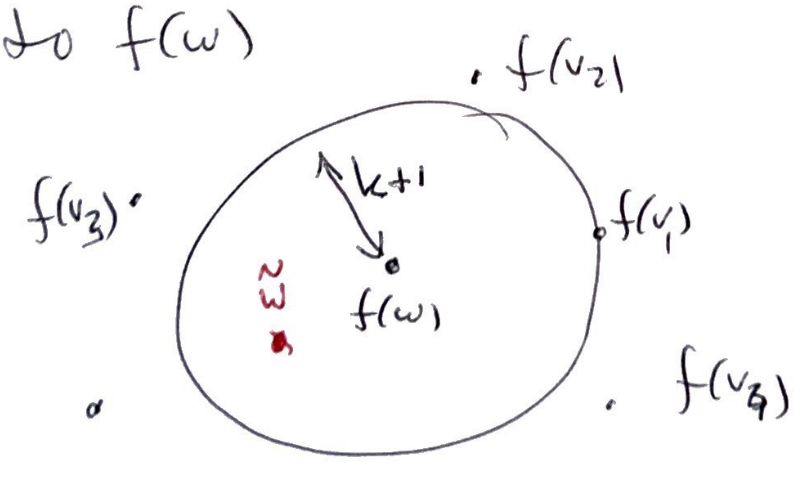
$$\tilde{w} \notin f(B^m)$$

$\tilde{w}$ is not a code word.

If $\tilde{w}$ has up to k errors. then

$$d(\tilde{w}, f(w)) \leq k.$$

So all other code words in $f(B^m)$ has to have distance at least $k+1$ to $f(w)$

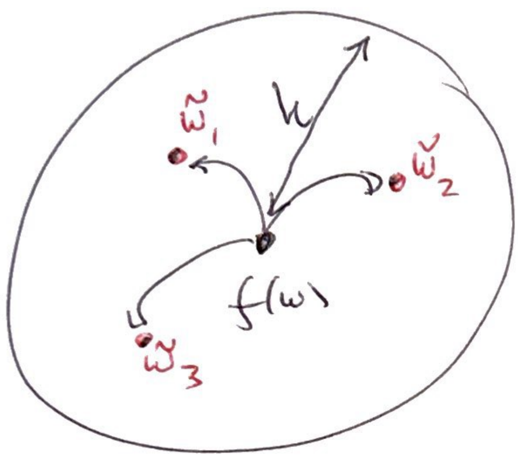


$\tilde{w} \notin f(B^m) \setminus f(w)$

$\square$

If $\tilde{\omega}$ is the received version of $f(w)$ with at most k errors

then $d(\tilde{\omega}, f(w)) \leq k$



$\tilde{\omega}_i$ are words with at most k errors

How to use the code?

Given $\tilde{\omega}$ find the unique closest $f(w) \in f(B^m)$.

Thm ~~f~~ $f: B^m \to B^n$

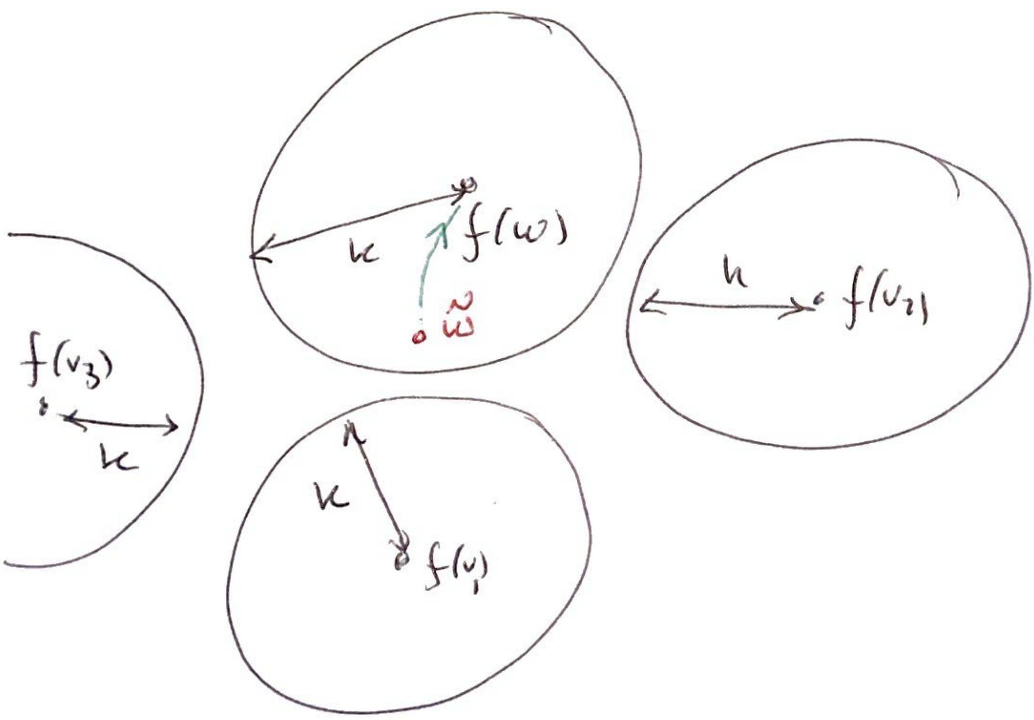
f can correct up to k errors $\iff$

$$\min d(f(v), f(w)) \geq 2k+1$$

Pf) To be able to correct k errors.

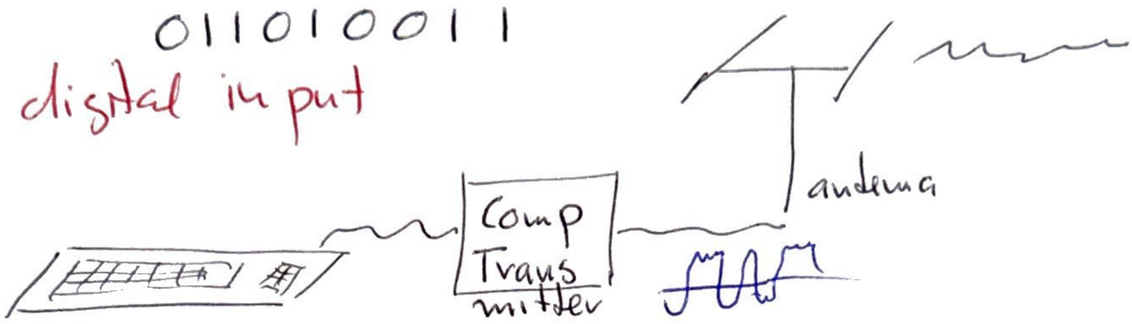
There needs to be only one unique code word at distance at most k.

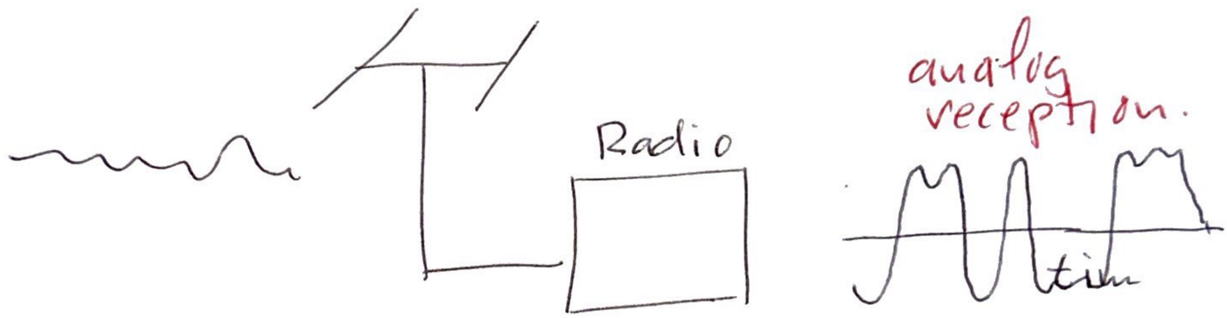
$\tilde{w}$ received version of $f(w)$



you can recover $f(w) \leftarrow \tilde{w}$

You want to send the message





Not something ideal

One should ~~except~~ expect errors ....

And one should be able to

detect

and

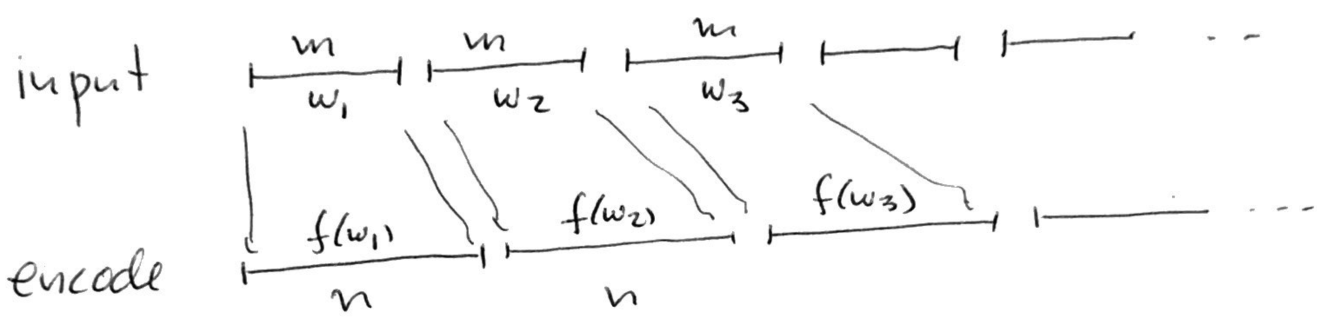
correct

errors.

Main ~~Hany~~ idea: include redundancy

encode the input ($\mathbb{B} = \mathbb{Z}_2$)

$$f: \mathbb{B}^m \longrightarrow \mathbb{B}^n \qquad n > m$$



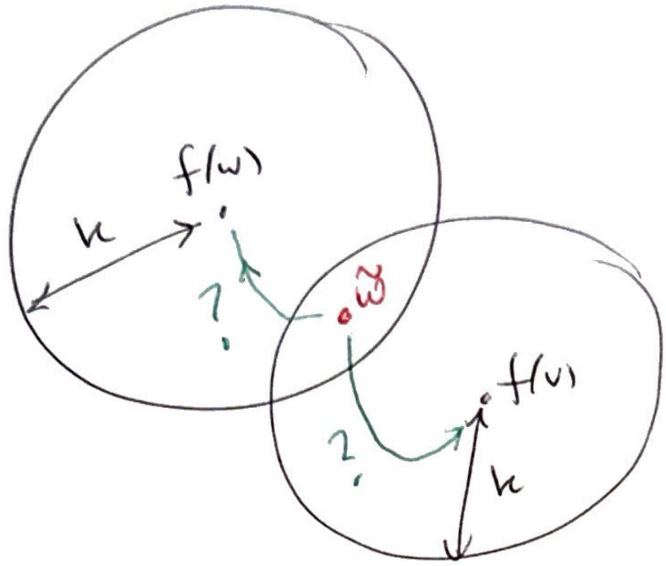
A good code $f: \mathbb{B}^m \to \mathbb{B}^n$ has

$$\boxed{D = \boxed{\min_{v \neq w} d\big(f(w), f(v)\big)}}$$

large.

If code has $D = 2k+1$ it can correct k errors and detect $D-1$ errors

~~code~~



distance $d(f(w), f(v)) \leq 2k$ you don't know what $\tilde{w}$ belongs to. $\square$.

—— // ——

LNXIb

How to construct codings?

One way is:

$f: B^m \to B^n$ is a linear coding (or group code) if

$$f(B^m) \subset B^n$$

is a subgroup.

<u>Thm</u> $f: B^m \to B^n$ group code

$$\min_{v \neq w} d(f(v), f(w)) = \min_{v \neq 0} \operatorname{weight}(f(v)).$$

Pf: without proof. Easy. Like for linear maps

$$d(Ax, Ay) = \|A(x-y)\| \qquad \square.$$

A group code can be constructed with a generating matrix G. Its entries are in the field $\mathbb{Z}_2 = B$.

$$G = m\updownarrow \left(\begin{array}{c|c} \underbrace{\begin{matrix} 1 & & 0 \\ & \ddots & \\ 0 & & 1 \end{matrix}}_{m} & \underbrace{\boxed{}}_{n-m} \end{array} \right)$$

$f_G: B^m \to B^n \qquad w \mapsto wG.$

<u>Rmk</u> $f_G(w) = w\underbrace{\boxed{\cdots\cdots}}_{\text{check symbols}}$.

Ex) $G = \left(\begin{array}{cc|ccc} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{array}\right)$

$$f_G \begin{cases} 01 \longmapsto 0101 \\ 10 \longmapsto 1011 \\ 11 \longmapsto 1110 \end{cases} \qquad \min_{v \neq 0} \underbrace{|f(v)|}_{\text{weight}} = 2.$$

f_G can detect 1 error.

Ex) $G = \left(\begin{array}{cc|ccc} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{array}\right)$

$$f_G \begin{cases} 01 \longmapsto 01011 \\ 10 \longmapsto 10110 \\ 11 \longmapsto 11101 \end{cases} \qquad \min_{v \neq 0} |f(v)| = 3$$

f_G can detect 2 error.

f_G can correct 1 error.

Ex) $G = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{array}\right)$

$$f_G \begin{cases} 001 \longmapsto 001111 \\ 010 \longmapsto 010101 \\ 011 \longmapsto 011010 \\ 100 \longmapsto 100111 \\ 101 \longmapsto 101000 \\ 110 \longmapsto 110010 \\ 111 \longmapsto 111101 \end{cases}$$

$\min_{v \neq 0} |f(v)| = 2$

f_G can only detect 1 error.

no correction.

Ex) $G = \left(\begin{array}{ccc|ccccccc} 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array}\right)$

$$f_G \begin{cases} 001 \longmapsto 0010111001 \\ 010 \longmapsto 0101011010 \\ 011 \longmapsto 0111100011 \\ 100 \longmapsto 1001101100 \\ 101 \longmapsto 1011100100 \\ 110 \longmapsto 1100110110 \\ 111 \longmapsto 1110001111 \end{cases}$$

$\min_{v \neq 0} |f(v)| = 5 \begin{cases} \text{detect 4 errors} \\ \text{correct 2 errors.} \end{cases}$