Error-correcting bit flips

data $\longrightarrow$ noisy medium $\xrightarrow{\text{Corrupted data}}$ Decoder $\longrightarrow$ Recovers original data

- Storage device
- Communication medium.

n-bits

$x_1 \cdots x_n$ data $\xrightarrow[\text{erased}]{\text{1 bit gets}}$ $x_1 x_2 \cdot x_{i-1} \, ? \, x_{i+1} \cdot x_n$

(as in replaced by ?) (ith bit missing)

Impossible if $(x_1 \cdots x_n)$ can be arbitrary.

Error-correcting code: Judicious redundancy built into $(x_1 \cdots x_n)$ "codeword" that allows to combat effects of noise.

Restrict $(x_1 \cdots x_n)$ to have even # 1s.

$$C^{\text{(one erasure)}} = \{ (x_1 \cdots x_n) \in \{0,1\}^n \mid x_1 + x_2 + \cdots x_n \equiv 0 \pmod 2 \}$$

$$(\Leftrightarrow \; x_n = x_1 \oplus x_2 \oplus \cdots \oplus x_{n-1})$$

Parity check code. (x_n is the parity bit)

Code has one redundant bit (single parity check)

Defn: Code (over bits) : $C \subseteq \{0,1\}^n$

(001100 . . .) → code where each bit repeated twice

Also corrects one erasure.

But has $\frac{n}{2}$ bits of redundancy.

Goal of coding theory: Find codes of small (optimal?) redundancy for various noise models.

Redundancy $:= n - \log_2 |C|$

$= 1$ for parity check code.

Exercise: For correcting one erasure, 1 redundant bit is smallest possible. (optimal)

Correcting bit flips

1 bit gets flipped, don't know which position

Parity check code doesn't work:

Receive

100000 ← 000000, 110000, 101000, 100100, 100010, 100001

n possible codewords

"$x_1 \oplus x_2 \oplus \cdots \oplus x_n = 0$"

"Check equation" should give more information.

Assume we know value of

$$s(x) = x_1 + 2x_2 + 3x_3 + \cdots + n x_n$$

Check eqn: "$s(x) = a$" for some $a \in \mathbb{Z}$.

$$x \in \{0,1\}^n \xrightarrow[\text{flipped}]{\text{1-bit}} y \in \{0,1\}^n$$

Can figure out x from y & $s(x)$.

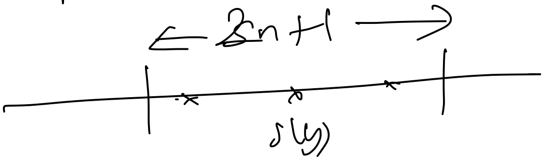
Compute: $s(y) = y_1 + 2y_2 + \cdots + n x_n$

$$s(x) - s(y) = \begin{cases} j & \text{if } x_j = 1,\ y_j = 0 \\ -j & \text{if } x_j = 0,\ y_j = 1 \end{cases}$$

$\therefore$ $|s(x) - s(y)|$ tells location of bit flip.

- $|s(x) - s(y)| \leq n$

So suffices to know $s(x)$ mod $2n+1$



$\forall a \in \{0, 1, .., 2n\}$

$$C_a = \{(x_1, x_2 \ldots x_n) \in \{0,1\}^n \mid s(x) = x_1 + 2x_2 + \cdots + nx_n \equiv a \ (\text{mod}(2n+1))\}$$

is a code that can correct 1-bit flip.

__Note:__ $\exists\, a$ s.t. $|C_a| \geq \dfrac{2^n}{(2n+1)}$ (pigeonhole principle)

Redundancy of such C_a is $\leq \log_2(2n+1)$

$\leq \log_2 n + O(1)$

$\downarrow$

(optimal upto $O(1)$ additive term)

"__Hamming codes__"

$$\sum_{i=1}^{n} i x_i \equiv a \pmod{(2n+1)}$$

$i \longrightarrow \vec{v_i}$ $\vec{v_i}$ is the binary representation of i

$$\vec{v_i} \in \{0,1\}^m, \quad m = \lceil \log_2 n \rceil$$

$$C_{Hamming} = \left\{ (x_1 x_2 \cdots x_n) \in \{0,1\}^n \;\middle|\; \sum_{i=1}^{n} x_i \vec{v_i} = \vec{0} \right\}$$

Optimal single bit flip correcting code!

$$\underset{\{0,1\}^n}{x} \xrightarrow[\text{flip}]{\text{1-bit}} \underset{\{0,1\}^n}{y}$$

If x_p was flipped,
$y_p = x_p + 1 \pmod 2$
(note: don't know p)

$$\sum_{i=1}^{n} y_i \vec{v_i} = \sum_{\substack{i=1 \\ i \neq p}}^{n} x_i \vec{v_i} + (x_p + 1)\vec{v_p}$$

$$= \sum_{i=1}^{n} x_i \vec{v_i} + \vec{v_p}$$

$$= \vec{0} + \vec{v_p} = \vec{v_p}$$

(by check eqn)

Thus $\sum_{i=1}^{n} y_i v_i$ gives binary representation of the location of the bit flip