DIS 5B

ERROR-CORRECTING CODES

motivation:

we want to send a message, but our communication channel is not 100% reliable

1. erasure errors: packets can be dropped
2. general errors: packets can be corrupted

ECCs allow us to incorporate redundancy into our encoded message

this way, the receiver can recover the original message even if errors have occurred!

SENDING MESSAGES

our message: $m = (\underbrace{m_1, m_2, m_3, \ldots m_n}_{n \text{ packets}})$

can be encoded with the polynomial $P(x)$ (deg $n-1$)

$P(1) = m_1, \quad P(2) = m_2, \quad \ldots \quad P(n) = m_n$

(think: how would you create this polynomial?)

	P(1)	P(2)	P(3)	P(4)	P(5)	P(6)	P(7)
original:	0	1	5	4	2	2	1
2 erasures:		1	5		2	2	1
2 corruptions:	0	3	6	4	2	2	1

ERASURE ERRORS

problem: k packets are dropped

solution: send k additional points (n+k total)

$$\underbrace{P(1)\ P(2)\ \dots\ P(n)}_{\text{message}}\ \underbrace{P(n+1)\ P(n+2)\ \dots\ P(n+k)}_{k\text{ additional packets}}$$

the receiver can then perform interpolation on the n packets they receive to find P(x).

to recover the message, evaluate P(1) ... P(n)

GENERAL ERRORS

problem: k packets are corrupted

solution: send 2k additional points (n+2k total)

the receiver uses Berlekamp-Welch to identify the errors and correct them.

BERLEKAMP - WELCH (overview)

define the error-locator polynomial $E(x)$ with roots at the indices of corruption.

$$E(x) = (x-e_1)(x-e_2)\dots(x-e_k)$$

let $Q(x) = P(x)E(x)$.

then: $Q(i) = P(i)E(i) = r_i E(i), \quad 1 \le i \le n+2k$

where r_i = ith received packet

we set up $n+2k$ equations $Q(i) = r_i E(i)$ and solve for the coefficients of $Q(x)$ and $E(x)$.

then, $P(x) = Q(x) / E(x)$.