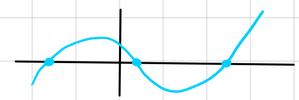
CS70 – Spring 2026

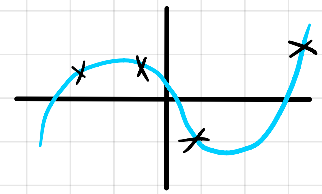
Lecture 11: Feb. 24

Last Lecture

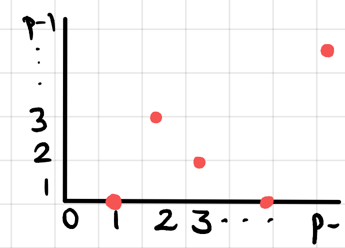
- Polynomials of degree d: $p(x) = a_d x^d + a_{d-1} x^{d-1} + \dots + a_1 x + a_0$
- Property 1: Any such poly. has $\leq d$ roots



- Property 2: Any $d+1$ points (x_i, y_i) define a unique polynomial of degree $\leq d$



- Polynomials mod p for prime p (i.e., over $GF[p]$) satisfy the same properties



- Application to secret sharing

Today: Application to Error Correcting Codes

intruder

$\downarrow$

message $\boxed{m_1 \dots m_n}$ $\rightarrow$ codeword $\boxed{c_1 \dots c_m}$ $\rightsquigarrow$ unreliable channel $\rightsquigarrow$ received msg $\boxed{r_1 \dots r_m}$

The intruder can cause two types of errors:

- erasures: up to k of the c_i get dropped (& we know which)
- general: up to k of the c_i get corrupted (& we don't know which)

Goal: Design $\boxed{c_1 \dots c_m}$ s.t. Bob can recover $\boxed{m_1 \dots m_n}$ from $\boxed{r_1 \dots r_m}$

Redundancy: We will need $m > n$

Erasures

$m_1 \dots m_n$ $\xrightarrow{\;c_1 \dots c_{n+k}\;}$ $\cancel{c_1}\; c_2\; c_3\; \cancel{c_4}\; c_5 \dots c_{n+k}$

$\le k$ packets dropped

$\Rightarrow\ \ge n$ packets received

Goal: Reconstruct $m_1 \dots m_n$ from any k of $c_1 \dots c_{n+k}$

- Let $q > n+k$ be a large prime s.t. packets are integers mod q (e.g. $q > 2^{32}$ for 32-bit packets)

Let $p(x)$ be the unique degree-$(n-1)$ poly. $(\mathrm{mod}\ q)$ through the points (i, m_i) $\quad 1 \le i \le n$

Send codeword $\boxed{c_1\, c_2 \dots c_{n+k}}$ where $\boxed{c_j = p(j) \quad 1 \le j \le n+k}$

Can reconstruct $p(x)$ given any n of the c_i $\longrightarrow$ get the original packets $m_i = p(i)$ $\quad 1 \le i \le n$

Example: Message 6605; $n=4$, $k=2$ (≤ 2 packets dropped)

Take $q=11$

1. Find degree-3 poly. through $(1,6)$, $(2,6)$, $(3,0)$, $(4,5)$

$$\Delta_1(x) = \frac{(x-2)(x-3)(x-4)}{(1-2)(1-3)(1-4)} = (-6)^{-1}.(\ )(\ )(\ ) = 9(x-2)(x-3)(x-4)$$

$$\Delta_2(x) = \frac{(x-1)(x-3)(x-4)}{(2-1)(2-3)(2-4)} = 2^{-1}.(\ )(\ )(\ ) = 6(x-1)(x-3)(x-4)$$

$$\Delta_3(x) = \frac{\quad}{\quad}$$

$$\Delta_4(x) = \frac{(x-1)(x-2)(x-4)}{(3-1)(3-2)(3-4)} = 6^{-1}(\ )(\ )(\ ) = 2(x-1)(x-3)(x-4)$$

$$\Rightarrow p(x) = 6\Delta_1(x) + 6.\Delta_2(x) + 0.\Delta_3(x) + 5.\Delta_4(x)$$

$$= 54(x-2)(x-3)(x-4) + 36(x-1)(x-3)(x-4) + 10(x-1)(x-3)(x-4)$$

$$= \boxed{x^3 + 2x^2 + 9x + 5 \pmod{11}}$$

$$\boxed{p(x) = x^3 + 2x^2 + 9x + 5}$$

2. Compute $k=2$ additional points on $p(x)$:

$$p(5) = 5^3 + 2\cdot 5^2 + 9\cdot 5 + 5 \equiv 5 \pmod{11} \rightarrow (5,5)$$

$$p(6) = 6^3 + 2\cdot 6^2 + 9\cdot 6 + 5 \equiv 6 \pmod{11} \rightarrow (6,6)$$

3. Send the $n+k=6$ packets:
$(1,6)$, $(2,6)$, $(3,0)$, $(4,5)$, $(5,5)$, $(6,6)$

4. Given <u>any 4</u> of these packets, we have 4 points on degree-3 poly. $p(x)$ $\longrightarrow$ can compute $p(x)$ by Lagrange Interpolation

5. Recover original msg. as $p(1)\,p(2)\,p(3)\,p(4)$

<u>Ex</u>: Sp. 2nd & 3rd packets are dropped. Compute $p(x)$ from $(1,6)$, $(4,5)$, $(5,5)$, $(6,6)$ & recover message.

Interpolation Revisited

Given: $d+1$ points $(x_1, y_1), \ldots, (x_{d+1}, y_{d+1})$

Goal: Find the unique degree-d polynomial $p(x)$ s.t. $p(x_i) = y_i$ for $1 \le i \le d+1$

Method 1: Lagrange ✓

Method 2: Solve system of linear equations:

Write $p(x) = a_d x^d + \cdots + a_1 x + a_0$

Equations for the coefficients a_i:

$$\left.\begin{aligned}
a_d x_1^d + \cdots + a_1 x_1 + a_0 &= y_1 \\
a_d x_2^d + \cdots + a_1 x_2 + a_0 &= y_2 \\
&\vdots \\
a_d x_{d+1}^d + \cdots + a_1 x_{d+1} + a_0 &= y_{d+1}
\end{aligned}\right\} \text{$d+1$ equations in $d+1$ unknowns}$$

Example: Find degree-2 polynomial (mod 11) through the points (0,4), (1,2), (2,3)

$$p(x) = a_2 x^2 + a_1 x + a_0$$

Equations:

$$a_2 \cdot 0 + a_1 \cdot 0 + a_0 = 4$$
$$a_2 \cdot 1 + a_1 \cdot 1 + a_0 = 2$$
$$a_2 \cdot 4 + a_1 \cdot 2 + a_0 = 3$$

$$\left.\begin{aligned} a_0 &= 4 \\ a_2 + a_1 + a_0 &= 2 \\ 4a_2 + 2a_1 + a_0 &= 3 \end{aligned}\right\}$$

Solve: $a_0 = 4$

$$\left.\begin{aligned} a_2 + a_1 &= -2 \\ 4a_2 + 2a_1 &= -1 \end{aligned}\right\} \quad 2a_2 = 3$$

$$a_2 = 2^{-1} \cdot 3 \equiv 6 \cdot 3 \pmod{11}$$
$$\equiv 7 \pmod{11}$$

$$a_1 = -2 - a_2 = -2 - 7$$
$$\equiv 2 \pmod{11}$$

$$p(x) = 7x^2 + 2x + 4 \pmod{11}$$

$\cancel{Ex}$: $p(x) = \frac{3}{2} x^2 - \frac{7}{2} x + 4$

$\boxed{Q:}$ How do we know that these equations always have a (unique) solution ?

$\boxed{A:}$ Equations can be written in matrix form:

$$(a_0\ a_1 \ldots a_d) \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ x_1 & x_2 & x_3 & \cdots & x_{d+1} \\ x_1^2 & x_2^2 & x_3^2 & \cdots & x_{d+1}^2 \\ \vdots & \vdots & \vdots & & \vdots \\ x_1^d & x_2^d & x_3^d & & x_{d+1}^d \end{pmatrix} = (y_1\ y_2 \ldots y_{d+1})$$

$$a \qquad M \qquad = \qquad y$$

Matrix M is a <u>VanderMonde</u> matrix, which is always invertible if all the x_i are distinct

Hence $\exists$ unique solution $\boxed{a = yM^{-1}}$

Back to ECCs : General Errors

$m_1 \ldots m_n \xrightarrow{c_1 \ldots c_{n+2k}}$ $c_1\, c_2\, c_3\, c_4 \ldots c_{n+2k}$, received as $r_1\, r_2\, r_3\, r_4 \ldots r_{n+2k}$

$\leq k$ packets corrupted
$\geq n+k$ packets uncorrupted

$\Big\}$ BUT we don't know which!

- Let $q > n+2k$ be prime

Let $p(x)$ be the unique degree-$(n-1)$ poly. (mod q) through the points (i, m_i) $1 \leq i \leq n$

Send codeword $c_1 c_2 \ldots c_{n+2k}$ where $c_j = p(j)$ $1 \leq j \leq n+2k$

Can reconstruct $p(x)$ given the $n+2k$ packets $r_1, r_2, \ldots, r_{n+2k}$ (up to k of which are corrupted/bad) NOT obvious how!

Erasures vs. General Errors

Erasures: codeword: $n+k$ points on poly. $p(x)$

k points deleted

$\Rightarrow$ can still reconstruct p from remaining n pts.

General Errors: codeword: $n+2k$ points on poly. $p(x)$

k points corrupted ("bad": $r_i \neq c_i$)

$\Rightarrow$ have $n+k$ "good" points (+ k bad pts)

BUT we don't know which are good!

$\boxed{\text{Q: Can we still reconstruct } p\text{?}}$

General Errors: codeword: $n+2k$ points on poly. $p(x)$
k points corrupted ("bad")
$\Rightarrow$ have $n+k$ "good" points (+ k bad pts)

Q: Can we still reconstruct p ?

Good news: We have $n+k$ good points, which is k more than we need!

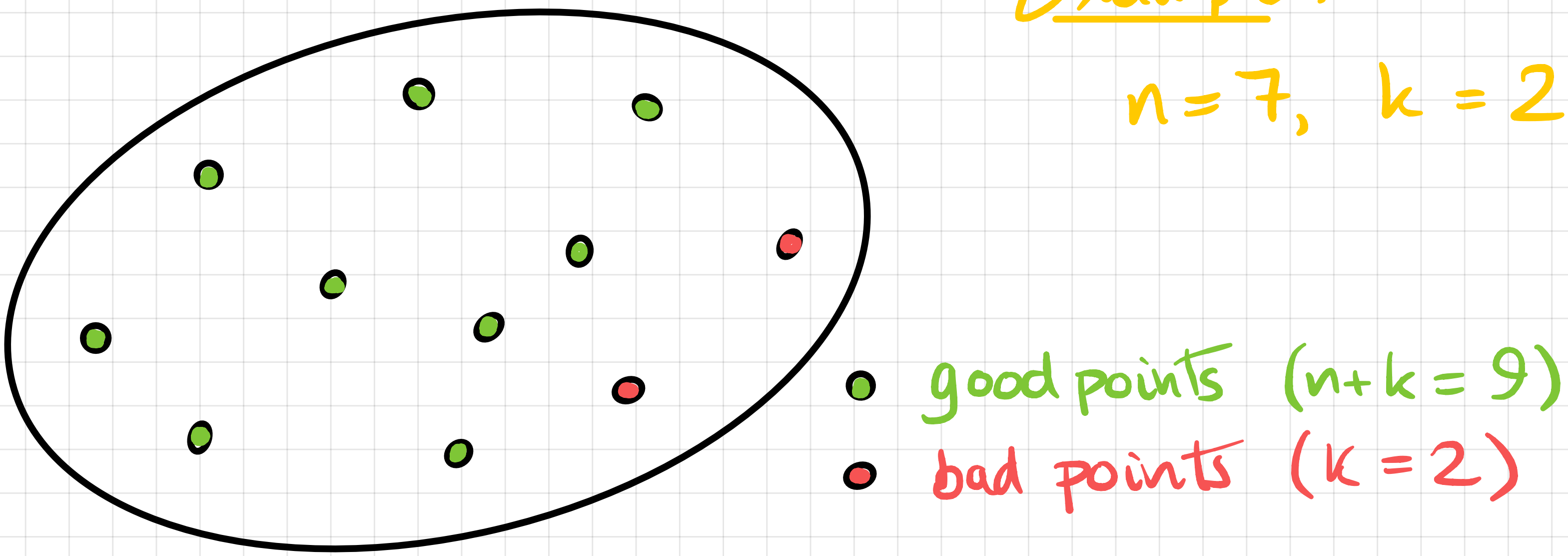
Bad news: We don't know which are the good points!

Strategy: Find a poly p that goes through some $n+k$ of the points

Among those $n+k$ points, at least $n+k-k=n$ are good — so p must be the correct poly!

Example:

$n = 7, \quad k = 2$

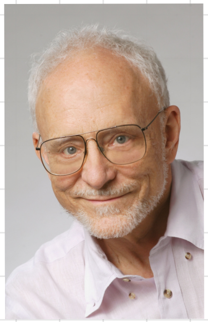


Any subset of $n+k = 9$ points must include at least $n = 7$ good points!

So: enough to find a poly P of degree $n-1 = 6$ that goes through any $n+k = 9$ points

Goal: Given $n+2k$ points (i, r_i), find a degree-$(n-1)$ polynomial that goes through $\geq n+k$ of them

Very Clever Trick [Berlekamp & Welch]



Elwyn Berlekamp

Lloyd Welch

Use another polynomial - the "error-locator polynomial" — to code up the positions of the errors / bad points

Define $\boxed{E(x) := (x-e_1)(x-e_2)\ldots(x-e_k)}$

where $e_1, e_2, \ldots, e_k \in \{1, 2, \ldots, n+2k\}$ are the positions of the errors (i.e., those i for which $(i, r_i) \neq (i, c_i)$)

NOTE: We don't (yet) know these values e_i !

Define $\boxed{E(x) := (x-e_1)(x-e_2)\ldots(x-e_k)}$

Key Equations:

$$\boxed{P(i)\,E(i) = r_i\,E(i) \quad \text{for } 1 \le i \le n+2k}$$

Proof:

Case (i): i is good point – i.e., $P(i) = r_i$

$$P(i)E(i) = P(i)E(i)$$ ✓

Case (ii): i is a bad point – i.e., $P(i) \ne r_i$

$$\underbrace{P(i)E(i)}_{=0} = r_i \underbrace{E(i)}_{=0}$$ ✓

$$\boxed{P(i)E(i) = r_i E(i) \quad \text{for } 1 \le i \le n+2k}$$

Define new polynomial $Q(x) = P(x)E(x)$

Then Q has degree $n-1+k = n+k-1$

And equations become

$$\boxed{Q(i) = r_i E(i) \qquad 1 \le i \le n+2k}$$

Write out:

$$Q(x) = a_{n+k-1} x^{n+k-1} + a_{n+k-2} x^{n+k-2} + \dots + a_1 x + a_0$$

$$E(x) = x^k + b_{k-1} x^{k-1} + \dots + b_1 x + b_0$$

Plugging in the $n+2k$ points (i, r_i) gives:

$n+2k$ equations in $n+2k$ unknowns

$\longrightarrow$ solve for the a_i and the b_i !

<u>Example</u>: Message = 820 $\quad n=3,\ k=1$ error

$q = 11$

1. Construct unique degree-2 poly. (mod 11) through $(1,8), (2,2), (3,0)$

$$\Delta_1(x) = \frac{(x-2)(x-3)}{(1-2)(1-3)} = 6(x-2)(x-3)$$

$$\Delta_2(x) = \frac{(x-1)(x-3)}{(2-1)(2-3)} = -(x-1)(x-3)$$

$$\Delta_3(x) = \underline{\qquad}$$

$$\Rightarrow P(x) = 8\cdot\Delta_1(x) + 2\cdot\Delta_2(x) + 0\cdot\Delta_3(x)$$

$$= \ldots$$

$$= \boxed{2x^2 - x + 7} \pmod{11}$$

<u>CHECK</u>: $P(1) = 8;\ P(2) = 2;\ p(3) = 0$ (all mod 11)

$$\boxed{P(x) = 2x^2 - x + 7}$$

2. Compute $2k = 2$ additional points:

$$P(4) = 2 \quad ; \quad P(5) = 8$$

3. Send $n + 2k$ points as codeword:

$$(1,8), (2,2), (3,0), (4,2), (5,8)$$

Sp first character is corrupted & we receive $(1,1)$

4. Write down five equations in five unknowns:

$$Q(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$$

$$E(x) = x + b_0$$

$$\left.\begin{array}{l} Q(i) = r_i \, E(i) \\ 1 \le i \le 5 \end{array}\right\} \quad \begin{array}{l} Q(1) = 1.E(1) \\ Q(2) = 2.E(2) \\ Q(3) = 0.E(3) \end{array} \quad \begin{array}{l} Q(4) = 2.E(4) \\ Q(5) = 8.E(5) \end{array}$$

4. Write down five equations in five unknowns:

$$Q(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0$$

$$E(x) = x + b_0$$

$$\left.\begin{array}{l} Q(i) = r_i E(i) \\ 1 \le i \le 5 \end{array}\right\}$$

$$Q(1) = 1.E(1) \qquad Q(4) = 2.E(4)$$
$$Q(2) = 2.E(2) \qquad Q(5) = 8.E(5)$$
$$Q(3) = 0.E(3)$$

$$\left.\begin{array}{rcl} a_3 + a_2 + a_1 + a_0 &=& 1 + b_0 \\ 8a_3 + 4a_2 + 2a_1 + a_0 &=& 2(2+b_0) \\ 27a_3 + 9a_2 + 3a_1 + a_0 &=& 0 \\ 64a_3 + 16a_2 + 4a_1 + a_0 &=& 2(4+b_0) \\ 125a_3 + 25a_2 + 5a_1 + a_0 &=& 8(5+b_0) \end{array}\right\} \bmod 11$$

$$\left.\begin{aligned}
a_3 + a_2 + a_1 + a_0 &= 1 + b_0 \\
8a_3 + 4a_2 + 2a_1 + a_0 &= 2(2 + b_0) \\
27a_3 + 9a_2 + 3a_1 + a_0 &= 0 \\
64a_3 + 16a_2 + 4a_1 + a_0 &= 2(4 + b_0) \\
125a_3 + 25a_2 + 5a_1 + a_0 &= 8(5 + b_0)
\end{aligned}\right\} \bmod 11$$

5. Solve these equations (mod 11) to get:

$$a_3 = 2, \quad a_2 = 8, \quad a_1 = 8, \quad a_0 = 4, \quad b_0 = -1$$

Hence we have

$$Q(x) = 2x^3 + 8x^2 + 8x + 4$$

$$E(x) = x - 1$$

So error is in position 1 (1st packet)

And $P(x) = \dfrac{Q(x)}{E(x)} = \dfrac{2x^3 + 8x^2 + 8x + 4}{x - 1} = \boxed{2x^2 - x + 7}$

6. Recover original msg: $\boxed{P(1)\,P(2)\,P(3) = 820}$