Quick recap of last time.

**Erasure Codes:** Reconstructing a message if some parts of it (packets) are lost.

**Idea:** Encode n-packet message as a polynomial with n coefficients

- Send values at n+k points if ≤ k will be lost
- Reconstruct from what you receive.

Today’s topic.

**Problem:** Communicate n packets m₁, …, mₙ on noisy channel that corrupts ≤ k packets.

**Reed-Solomon Code:**
1. Make a polynomial, \( P(x) \) of degree \( n-1 \), that encodes message.
   - \( P(1) = m₁ \), …, \( P(n) = mₙ \).
   - Recall: could encode with packets as coefficients.
2. Send \( P(1), \ldots, P(n+2k) \).

**After noisy channel:** Receive values \( R(1), \ldots, R(n+2k) \).

**Properties:**
1. \( P(i) = R(i) \) for at least \( n+k \) points \( i \),
2. \( P(x) \) is unique degree \( n-1 \) polynomial that contains \( ≥ n+k \) received points.

**Proof:**
1. Easy. Only \( k \) corruptions (by assumption).
2. Degree \( n-1 \) polynomial \( Q(x) \) consistent with \( n+k \) points.
   - \( Q(x) \) agrees with \( R(i) \), \( n+k \) times.
   - \( P(x) \) agrees with \( R(i) \), \( n+k \) times.
   - Total points contained by both: \( 2n+2k \).
   - Total points to choose from: \( n+2k \).
   - Points contained by both: \( ≥ n \).
   - Points - Holes: \( P - H \).

**Error Correction**

**Error Correction:**
Noise Channel: \( \text{corrupts} \ k \) packets. (rather than loss/erasures.)

**Additional Challenge:** Finding which packets are corrupt.

**Example.**

- **Message:** 3, 0, 6.
- **Reed Solomon Code:** \( P(x) = x^2 + x + 1 \) (mod 7) has
  \( P(1) = 3, P(2) = 0, P(3) = 6 \mod 7 \).
- **Send:** \( P(1) = 3, P(2) = 0, P(3) = 6, P(4) = 0, P(5) = 3 \).
- **Receive:** \( R(1) = 3, R(2) = 0, R(3) = 6, R(4) = 0, R(5) = 3 \).
- **Proof:** \( P(i) = R(i) \) for \( n+k = 3+1 = 4 \) points.
Slow solution.

Brute Force:
For each subset of \( n + k \) points
Fit degree \( n - 1 \) polynomial, \( Q(x) \), to \( n \) of them.
Check if consistent with \( n + k \) of the total points.
If yes, output \( Q(x) \).

- For subset of \( n + k \) pts where \( R(i) \neq P(i) \),
  method will reconstruct \( P(x) \)!
- For any subset of \( n + k \) pts,
  1. there is unique degree \( n - 1 \) polynomial \( Q(x) \) that fits \( n \) of them
  2. and where \( Q(x) \) is consistent with \( n + k \) points

Reconstructs \( P(x) \) and only \( P(x) \)!!

---

### Example.

Received \( R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3 \)
Find \( P(x) = p_2 x^2 + p_1 x + p_0 \) that contains \( n + k = 3 + 1 \) points.

All equations...

\[
\begin{align*}
\rho_2 + \rho_1 + \rho_0 &= 3 \pmod{7} \\
4\rho_2 + 2\rho_1 + \rho_0 &= 1 \pmod{7} \\
2\rho_2 + 3\rho_1 + \rho_0 &= 6 \pmod{7} \\
2\rho_2 + 4\rho_1 + \rho_0 &= 0 \pmod{7} \\
1\rho_2 + 5\rho_1 + \rho_0 &= 3 \pmod{7}
\end{align*}
\]

Assume point 1 is wrong and solve...no consistent solution!
Assume point 2 is wrong and solve...consistent solution!

---

### Example.

Received \( R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3 \)
Find \( P(x) = p_2 x^2 + p_1 x + p_0 \) that contains \( n + k = 3 + 1 \) points.

All equations...

\[
\begin{align*}
(1 - a)(p_2 + p_1 + p_0) &= (3)(1 - a) \pmod{7} \\
(2 - a)(4p_2 + 2p_1 + p_0) &= (1)(2 - a) \pmod{7} \\
(3 - a)(2p_2 + 3p_1 + p_0) &= (3)(3 - a) \pmod{7} \\
(4 - a)(2p_2 + 4p_1 + p_0) &= (0)(4 - a) \pmod{7} \\
(5 - a)(4p_2 + 5p_1 + p_0) &= (3)(5 - a) \pmod{7}
\end{align*}
\]

Error locator polynomial: \( (x - 2) \).

Multiply equation \( i \) by \( (i - 2) \). All equations satisfied!
But don’t know error locator polynomial! Do know form: \( (x - e) \).
4 unknowns \( \{p_0, p_1, p_2, e\} \), 5 nonlinear equations.

---

### In general..

\[
P(x) = p_{n-1} x^{n-1} + \cdots + p_0 \text{ and receive } R(1), \ldots, R(m = n + 2k).
\]

\[
\begin{align*}
p_{n-1} + \cdots + p_0 &= R(1) \pmod{p} \\
p_{n-1} x^{n-1} + \cdots + p_0 &= R(2) \pmod{p} \\
&\quad \vdots \\
p_{n-1} x^{n-1} + \cdots + p_0 &= R(m) \pmod{p}
\end{align*}
\]

Error! Where???
Could be anywhere!!!...so try everywhere.
Runtime: \( \binom{n+2k}{k} \) possibilities.
Something like \( (n/k)^k \)...Exponential in \( k! \).
How do we find where the bad packets are efficiently??!

---

### The General Case.

\[
\begin{align*}
E(1)(p_{n-1} + \cdots + p_0) &= R(1)E(1) \pmod{p} \\
&\vdots \\
E(i)(p_{n-1} x^{n-1} + \cdots + p_0) &= R(i)E(i) \pmod{p} \\
&\vdots \\
E(m)(p_{n-1} x^{n-1} + \cdots + p_0) &= R(m)E(m) \pmod{p}
\end{align*}
\]

\[
P(x) = p_{n-1} x^{n-1} + \cdots + p_0 \text{ and receive } R(1), \ldots, R(m = n + 2k).
\]

\[
\begin{align*}
P(x) &= p_{n-1} x^{n-1} + \cdots + p_0 \\
P(x) &= \sum_{j=0}^{n-1} a_j x^j
\end{align*}
\]

m = n + 2k satisfied equations, \( n + k \) unknowns. But nonlinear!

Let \( Q(x) = E(x)P(x) = a_{n+k-1} x^{n+k-1} + \cdots + a_0 \).
Rewrite the \( i \)th equation, for all \( i \), as:

\[
Q(i) = R(i)E(i)
\]

Note: this is linear in \( a_i \) and coefficients of \( E(x) \)!
Finding $Q(x)$ and $E(x)$?

- $E(x)$ has degree $k$ ...
  \[ E(x) = x^k + b_{k-1}x^{k-1} + \cdots + b_0 \]
- $Q(x) = P(x)E(x)$ has degree $n + k - 1$ ...
  \[ Q(x) = a_{n+k-1}x^{n+k-1} + a_{n+k-2}x^{n+k-2} + \cdots + a_0 \]

Example: Compute $P(x)$.

\[ Q(x) = x^3 + 6x^2 + 6x + 5, \quad E(x) = x - 2 \]

\[
\begin{array}{c|c}
\hline
1 & 1 & 1 \\
\hline
x - 2 & x^2 + 6x + 5 \\
& x^3 - 2x^2 \\
& 1x^2 + 6x + 5 \\
& 1x^2 - 2x \\
& x + 5 \\
& x - 2 \\
& 0 \\
\hline
\end{array}
\]

\[ P(x) = x^2 + x + 1 \]
Message is $P(1) = 3, P(2) = 0, P(3) = 6.$

Solving for $Q(x)$ and $E(x)$...and $P(x)$

For all points $1, \ldots, n + 2k$,
\[ Q(i) = R(i) \cdot E(i) \quad (\text{mod } p) \]

Gives $n + 2k$ linear equations.
\[
\begin{align*}
a_{n+k-1} + \cdots + a_0 &= R(1)(1 + b_{k-1} + \cdots + b_0) \quad (\text{mod } p) \\
a_{n+k-1}(2)^{n+k-1} + \cdots + a_0 &= R(2)(2^k + b_{k-1}(2^{k-1} + \cdots + b_0) \quad (\text{mod } p) \\
& \quad \vdots \\
a_{n+k-1}(m)^{n+k-1} + \cdots + a_0 &= R(m)((m)^k + b_{k-1}(m^{k-1} + \cdots + b_0) \quad (\text{mod } p)
\end{align*}
\]

.. and $n + 2k$ unknown coefficients of $Q(x)$ and $E(x)$!
Solve for coefficients of $Q(x)$ and $E(x)$.

Once we have those, compute $P(x)$ as $Q(x)/E(x)$.

Error Correction: Berlekamp-Welch

Message: $m_1, \ldots, m_{n}$.  
Sender:
1. Form degree $n - 1$ polynomial $P(x)$ where $P(i) = m_i$.
2. Send $P(1), \ldots, P(n + 2k)$.

Receiver:
1. Receive $R(1), \ldots, R(n + 2k)$.
2. Solve $n + 2k$ equations, $Q(i) = E(i) \cdot R(i)$ to find $Q(x) = E(x) \cdot P(x)$ and $E(x)$.
3. Compute $P(x) = Q(x)/E(x)$.
4. Compute $P(1), \ldots, P(n)$, recover the message.

Example.

Received $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$.
\[ Q(x) = E(x) \cdot P(x) = a_3x^3 + a_2x^2 + a_1x + a_0 \]
\[ E(x) = x - b_0 \]
\[ Q(i) = R(i) \cdot E(i) \]
\[ \begin{align*}
a_3 + a_2 + a_1 + a_0 &= 3(1 - b_0) \quad (\text{mod } 7) \\
a_3 + 4a_2 + 2a_1 + a_0 &= 1(2 - b_0) \quad (\text{mod } 7) \\
6a_3 + 2a_2 + 3a_1 + a_0 &= 6(3 - b_0) \quad (\text{mod } 7) \\
a_3 + 2a_2 + 4a_1 + a_0 &= 0(4 - b_0) \quad (\text{mod } 7) \\
6a_3 + 4a_2 + 5a_1 + a_0 &= 3(5 - b_0) \quad (\text{mod } 7)
\end{align*} \]
\[ a_3 = 1, a_2 = 6, a_1 = 6, a_0 = 5 \text{ and } b_0 = 2. \]

\[ Q(x) = x^3 + 6x^2 + 6x + 5. \]
\[ E(x) = x - 2. \]

A key question.

Is there one and only one $P(x)$ from Berlekamp-Welch procedure?

Existence: there is a $P(x)$ and $E(x)$ that satisfy equations.

Example:

\[ R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3 \]
\[ Q(x) = E(x) \cdot P(x) = a_3x^3 + a_2x^2 + a_1x + a_0 \]
\[ E(x) = x - b_0 \]
\[ Q(i) = R(i) \cdot E(i) \]
\[ \begin{align*}
a_3 + a_2 + a_1 + a_0 &= 3(1 - b_0) \quad (\text{mod } 7) \\
a_3 + 4a_2 + 2a_1 + a_0 &= 1(2 - b_0) \quad (\text{mod } 7) \\
6a_3 + 2a_2 + 3a_1 + a_0 &= 6(3 - b_0) \quad (\text{mod } 7) \\
a_3 + 2a_2 + 4a_1 + a_0 &= 0(4 - b_0) \quad (\text{mod } 7) \\
6a_3 + 4a_2 + 5a_1 + a_0 &= 3(5 - b_0) \quad (\text{mod } 7)
\end{align*} \]
\[ a_3 = 1, a_2 = 6, a_1 = 6, a_0 = 5 \text{ and } b_0 = 2. \]

\[ Q(x) = x^3 + 6x^2 + 6x + 5. \]
\[ E(x) = x - 2. \]
Unique solution for $P(x)$?

Uniqueness: any solution $Q'(x)$ and $E'(x)$ have
\[ \frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)} = P(x). \] (1)

Proof:
We claim
\[ Q'(x)E(x) = Q(x)E'(x) \] on $n + 2k$ values of $x$. (2)

Equation 2 implies 1:
\[ Q'(x)E(x) \text{ and } Q(x)E'(x) \text{ are degree } n + 2k - 1 \]
and agree on $n + 2k$ points
\[ \Rightarrow Q'(x)E(x) = Q(x)E'(x). \]
Cross divide.

Revisiting last bit.

Claim: $Q'(x)E(x) = Q(x)E'(x)$ on $n + 2k$ values of $x$.

Proof: Construction implies that
\[ Q(i) = R(i)E(i) \]
\[ Q'(i) = R(i)E'(i) \]
for $i \in \{1, \ldots, n + 2k\}$.

If $E(i) = 0$, then $Q(i) = 0$. If $E'(i) = 0$, then $Q'(i) = 0$.
\[ \Rightarrow Q(i)E'(i) = Q'(i)E(i) \Rightarrow Q(i)E'(i) = Q'(i)E(i) \text{ holds when } E(i) \text{ or } E'(i) \text{ are zero.} \]

When $E'(i)$ and $E(i)$ are not zero
\[ \frac{Q(i)}{E'(i)} = \frac{Q'(i)}{E(i)} \Rightarrow R(i). \]
Cross multiplying gives equality in fact for these points.

Points to polynomials, have to deal with zeros!

Berlekamp-Welch algorithm decodes correctly when at most $k$ errors!

Summary. Error Correction.

Communicate $n$ packets, with $k$ erasures.

How many packets? $n + k$
How to encode? With polynomial, $P(x)$.
Of degree? $n - 1$
Recover? Reconstruct $P(x)$ with any $n$ points!

Communicate $n$ packets, with $k$ errors.

How many packets? $n + 2k$
How to encode? With polynomial, $P(x)$. Of degree? $n - 1$.
Recover?
Reconstruct error polynomial, $E(x)$, and $P(x)$!

Reconstruct $E(x)$ and $Q(x) = E(x)P(x)$. Linear Equations.
Polynomial division! $P(x) = Q(x)/E(x)$!
Reed-Solomon codes. Berlekamp-Welch Decoding. Perfection!