

# Today.

Last time:

- Shared (and sort of kept) secrets.

- Main Idea:  $d + 1$  points determine a polynomial.

  - Construct polynomial of degree  $k$ , with  $P(0) = s$ .

  - Give out any  $n$ -points other than  $(0, P(0))$ .

Today: Coding Theory.

- Tolerate packet drops. Erasure Codes.

- Tolerate errors in packets. Error Correction.

# The mathematics.

**Exactly one polynomial of degree  $d$  contains any  $d + 1$  points.**

Assumption: a field, in particular, arithmetic mod  $p$ .

Big Idea:

A polynomial:  $P(x) = a_d x^d + \dots + a_0$  has  $d + 1$  coefficients.

Any set of  $d + 1$  points determines the polynomial.

Stare at the above. What does it mean?

Many representations of a polynomial!

One coefficient representation.

Many, many point,value representations.

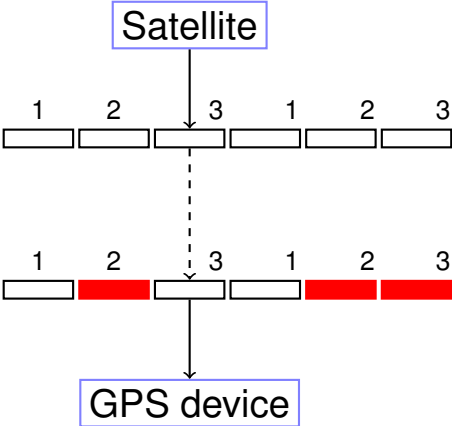
Some details:

Degree  $d$  generally degree “at most”  $d$ .

(example: choose 10 points on a line.)

Arithmetic (mod  $p$ )  $\implies$  work with  $O(\log p)$  bit numbers.

# Erasure Codes.



3 packet message. So send 6!

Lose 3 out 6 packets.

Gets packets 1,1,and 3.

## Solution Idea.

$n$  packet message, channel that loses  $k$  packets.

Must send  $n + k$  packets!

Any  $n$  packets should allow reconstruction of  $n$  packet message.

Any  $n$  point values allow reconstruction of degree  $n - 1$  polynomial.

Seem related?

Use polynomials.

Big Idea View:

Any set of  $n$  points contain information about  $n$  coefficients.  
or even any other set of  $n$  points!!!

“Information” about coefficients smeared across the  $n$  points.

Linear Algebra View:

Representing vector (message) in different basis.  
Many bases!

# The Scheme

**Problem:** Want to send a message with  $n$  packets.

**Channel:** Lossy channel: loses  $k$  packets.

**Question:** Can you send  $n + k$  packets and recover message?

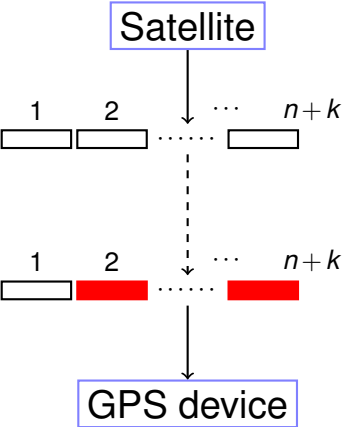
A degree  $n - 1$  polynomial determined by any  $n$  points!

Erasure Coding Scheme: message =  $m_0, m_1, \dots, m_{n-1}$ .

1. Choose prime  $p \approx 2^b$  for packet size  $b$ .
2.  $P(x) = m_{n-1}x^{n-1} + \dots + m_0 \pmod{p}$ .
3. Send  $P(1), \dots, P(n+k)$ .

Any  $n$  of the  $n + k$  packets gives polynomial ...and message!

# Erasure Codes.



$n$  packet message.

So send  $n+k$  points on polynomial.

Lose  $k$  packets.

Any  $n$  packets (points) is enough!

$n$  packet message.

Optimal.

# Polynomials.

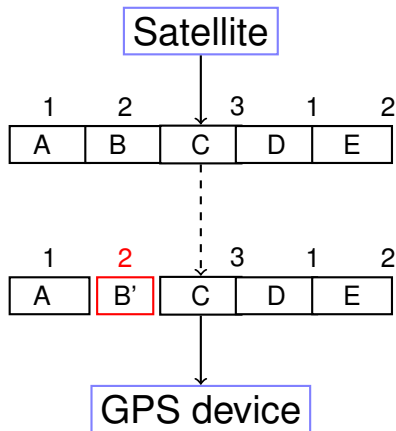
- ▶ ..give Secret Sharing.
- ▶ ..give Erasure Codes.

## **Error Correction:**

Noisy Channel: **corrupts**  $k$  packets. (rather than **loses**.)

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

# Error Correction



Which one was corrupted?

3 packet message. Send 5.

Corrupts 1 packets.



# The Scheme.

**Problem:** Communicate  $n$  packets  $m_1, \dots, m_n$  on noisy channel that corrupts  $\leq k$  packets.

## Reed-Solomon Code:

1. Make a polynomial,  $P(x)$  of degree  $n - 1$ , that encodes message.
  - ▶  $P(1) = m_1, \dots, P(n) = m_n$ .
  - ▶ **Comment:** could encode with packets as coefficients.
2. Send  $P(1), \dots, P(n + 2k)$ .

**After noisy channel:** Recieve values  $R(1), \dots, R(n + 2k)$ .

## Properties:

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

## Properties: proof.

$P(x)$ : degree  $n-1$  polynomial.

Send  $P(1), \dots, P(n+2k)$

Receive  $R(1), \dots, R(n+2k)$

At most  $k$   $i$ 's where  $P(i) \neq R(i)$ .

### Properties:

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

### Proof:

(1) Sure. Only  $k$  corruptions.

(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:  $2n+2k$ .  $P$  Pigeons.

Total points to choose from :  $n+2k$ .  $H$  Holes.

Points contained by both :  $\geq n$ .  $\geq P-H$  Collisions.

$\implies Q(i) = P(i)$  at  $n$  points.

$\implies Q(x) = P(x)$ .



## Argument on example: $n = 3, k = 1$

3 packet message.

Send  $n + 2k = 5$  points on degree 3 polynomial  $P(x)$ .

Receive:  $R(1), R(2), R(3), R(4), R(5)$ .

Only one  $i$ , where  $R(i) \neq P(i)$ .

$P(x)$  contains 4 of the points  $R(1), \dots, R(5)$ .

Another degree 3 polynomial,  $Q(x)$   
contains 4 of the points  $R(1), \dots, R(5)$ .

$P(x)$  and  $Q(x)$  have 3 points in common.

Since:  $P(x)$  contains 4,  $Q(x)$  contains 4.

There are only 5. So they agree on  $8 - 5 = 3$ .

P Q      P      P Q      P Q      Q

Degree 3  $\implies P(x) = Q(x)$

## Example.

Message: 3, 0, 6.

Reed Solomon Code:  $P(x) = x^2 + x + 1 \pmod{7}$  has  
 $P(1) = 3, P(2) = 0, P(3) = 6$  modulo 7.

Send:  $P(1) = 3, P(2) = 0, P(3) = 6, P(4) = 0, P(5) = 3$ .

(Aside: Message in plain text!)

Receive  $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$ .

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

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  $P(x) = p_2x^2 + p_1x + p_0$  that contains  $n + k = 3 + 1$  points.

All equations..

$$p_2 + p_1 + p_0 \equiv 3 \pmod{7}$$

$$4p_2 + 2p_1 + p_0 \equiv 1 \pmod{7}$$

$$2p_2 + 3p_1 + p_0 \equiv 6 \pmod{7}$$

$$2p_2 + 4p_1 + p_0 \equiv 0 \pmod{7}$$

$$4p_2 + 5p_1 + p_0 \equiv 3 \pmod{7}$$

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

Assume point 2 is wrong and solve...consistent solution!

In general..

$P(x) = p_{n-1}x^{n-1} + \dots p_0$  and receive  $R(1), \dots R(m = n + 2k)$ .

$$\begin{aligned} p_{n-1} + \dots p_0 &\equiv R(1) \pmod{p} \\ p_{n-1}2^{n-1} + \dots p_0 &\equiv R(2) \pmod{p} \end{aligned}$$

.

$$p_{n-1}i^{n-1} + \dots p_0 \equiv R(i) \pmod{p}$$

.

$$p_{n-1}(m)^{n-1} + \dots p_0 \equiv R(m) \pmod{p}$$

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?!?!?!?

## Ditty...

Oh where, Oh where  
has my little dog gone?  
Oh where, oh where can he be  
With his ears cut short  
And his tail cut long  
Oh where, oh where can he be?

Oh where, Oh where  
have my packets gone.. wrong?  
Oh where, oh where do they not fit.  
With the polynomial well put  
But the channel a bit wrong  
Where, oh where do we look?



## Where oh where can my bad packets be?

$$\begin{aligned} E(1)(p_{n-1} + \cdots p_0) &\equiv R(1)E(1) \pmod{p} \\ \mathbf{0} \times E(2)(p_{n-1} \mathbf{2}^{n-1} + \cdots p_0) &\equiv \mathbf{R(2)E(2)} \pmod{p} \\ &\vdots \\ E(m)(p_{n-1}(m)^{n-1} + \cdots p_0) &\equiv R(n+2k)E(m) \pmod{p} \end{aligned}$$

**Idea:** Multiply equation  $i$  by 0 if and only if  $P(i) \neq R(i)$ .

All equations satisfied!!!!

But which equations should we multiply by 0? **Where oh where...??**

**We will use a polynomial!!! That we don't know. But can find!**

Errors at points  $e_1, \dots, e_k$ . (In diagram above,  $e_1 = 2$ .)

**Error locator polynomial:**  $E(x) = (x - e_1)(x - e_2) \dots (x - e_k)$ .

$E(i) = 0$  if and only if  $e_j = i$  for some  $j$

Multiply equations by  $E(\cdot)$ . (Above  $E(x) = (x-2)$ .)

All equations satisfied!!

## Example.

Received  $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

Find  $P(x) = p_2x^2 + p_1x + p_0$  that contains  $n + k = 3 + 1$  points.

Plugin points...

$$\begin{aligned}(1 - e)(p_2 + p_1 + p_0) &\equiv (3)(1 - e) \pmod{7} \\(2 - e)(4p_2 + 2p_1 + p_0) &\equiv (1)(2 - e) \pmod{7} \\(3 - e)(2p_2 + 3p_1 + p_0) &\equiv (6)(3 - e) \pmod{7} \\(4 - e)(2p_2 + 4p_1 + p_0) &\equiv (0)(4 - e) \pmod{7} \\(5 - e)(4p_2 + 5p_1 + p_0) &\equiv (3)(5 - e) \pmod{7}\end{aligned}$$

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 ( $p_0, p_1, p_2$  and  $e$ ), 5 **nonlinear** equations.

..turn their heads each day,

$$\begin{aligned} E(1)(p_{n-1} + \cdots p_0) &\equiv R(1)E(1) \pmod{p} \\ &\vdots \\ E(i)(p_{n-1}i^{n-1} + \cdots p_0) &\equiv R(i)E(i) \pmod{p} \\ &\vdots \\ E(m)(p_{n-1}(n+2k)^{n-1} + \cdots p_0) &\equiv R(m)E(m) \pmod{p} \end{aligned}$$

...so satisfied, I'm on my way.

$m = n + 2k$  satisfied equations,  $n + k$  unknowns. **But nonlinear!**

Let  $Q(x) = E(x)P(x) = a_{n+k-1}x^{n+k-1} + \cdots a_0$ .

Equations:

$$Q(i) = R(i)E(i).$$

and linear in  $a_i$  and coefficients of  $E(x)$ !

## Finding $Q(x)$ and $E(x)$ ?

- ▶  $E(x)$  has degree  $k$  ...

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

$\implies k$  (unknown) coefficients. Leading coefficient is 1.

- ▶  $Q(x) = P(x)E(x)$  has degree  $n+k-1$  ...

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

$\implies n+k$  (unknown) coefficients.

Number of unknown coefficients:  $n+2k$ .

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

For all points  $1, \dots, i, n+2k = m$ ,

$$Q(i) = R(i)E(i) \pmod{p}$$

Gives  $n+2k$  linear equations.

$$a_{n+k-1} + \dots a_0 \equiv R(1)(1 + b_{k-1} \dots b_0) \pmod{p}$$

$$a_{n+k-1}(2)^{n+k-1} + \dots a_0 \equiv R(2)((2)^k + b_{k-1}(2)^{k-1} \dots b_0) \pmod{p}$$

$\vdots$

$$a_{n+k-1}(m)^{n+k-1} + \dots a_0 \equiv R(m)((m)^k + b_{k-1}(m)^{k-1} \dots b_0) \pmod{p}$$

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

Solve for coefficients of  $Q(x)$  and  $E(x)$ .

$$\text{Find } P(x) = Q(x)/E(x).$$

## Example.

Received  $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

$$Q(x) = E(x)P(x) = a_3x^3 + a_2x^2 + a_1x + a_0$$

$$E(x) = x - b_0$$

$$Q(i) = R(i)E(i).$$

$$a_3 + a_2 + a_1 + a_0 \equiv 3(1 - b_0) \pmod{7}$$

$$a_3 + 4a_2 + 2a_1 + a_0 \equiv 1(2 - b_0) \pmod{7}$$

$$6a_3 + 2a_2 + 3a_1 + a_0 \equiv 6(3 - b_0) \pmod{7}$$

$$a_3 + 2a_2 + 4a_1 + a_0 \equiv 0(4 - b_0) \pmod{7}$$

$$6a_3 + 4a_2 + 5a_1 + a_0 \equiv 3(5 - b_0) \pmod{7}$$

$a_3 = 1, a_2 = 6, a_1 = 6, a_0 = 5$  and  $b_0 = 2$ .

$$Q(x) = x^3 + 6x^2 + 6x + 5.$$

$$E(x) = x - 2.$$



# Error Correction: Berlekamp-Welsh

Message:  $m_1, \dots, m_n$ .

## Sender:

1. Form degree  $n - 1$  polynomial  $P(x)$  where  $P(i) = m_i$ .
2. Send  $P(1), \dots, P(n + 2k)$ .

## Receiver:

1. Receive  $R(1), \dots, R(n + 2k)$ .
2. Solve  $n + 2k$  equations,  $Q(i) = E(i)R(i)$  to find  $Q(x) = E(x)P(x)$  and  $E(x)$ .
3. Compute  $P(x) = Q(x)/E(x)$ .
4. Compute  $P(1), \dots, P(n)$ .



## Check your understanding.

You have error locator polynomial!

Where oh where have my packets gone **wrong**?

Factor? Sure.

Check all values? Sure.

Efficiency? Sure. Only  $n+2k$  values.

See where it is 0.

Hmmm...

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

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

## Unique solution for $P(x)$

**Uniqueness:** any solution  $Q'(x)$  and  $E'(x)$  have

$$\frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)} = P(x). \quad (1)$$

**Proof:**

We claim

$$Q'(x)E(x) = Q(x)E'(x) \text{ on } n+2k \text{ values of } x. \quad (2)$$

Equation 2 implies 1:

$Q'(x)E(x)$  and  $Q(x)E'(x)$  are degree  $n+2k-1$   
and agree on  $n+2k$  points

$E(x)$  and  $E'(x)$  have at most  $k$  zeros each.

Can cross divide at  $n$  points.

$$\implies \frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)} \text{ equal on } n \text{ points.}$$

Both degree  $\leq n \implies$  Same polynomial!



## Last bit.

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

**Proof:** Construction implies that

$$Q(i) = R(i)E(i)$$

$$Q'(i) = R(i)E'(i)$$

for  $i \in \{1, \dots, n+2k\}$ .

If  $E(i) = 0$ , then  $Q(i) = 0$ . If  $E'(i) = 0$ , then  $Q'(i) = 0$ .

$\implies Q(i)E'(i) = Q'(i)E(i)$  holds when  $E(i)$  or  $E'(i)$  are zero.

When  $E'(i)$  and  $E(i)$  are not zero

$$\frac{Q'(i)}{E'(i)} = \frac{Q(i)}{E(i)} = R(i).$$

Cross multiplying gives equality in fact for these points. □

Points to polynomials, have to deal with zeros!

Example: dealing with  $\frac{x-2}{x-2}$  at  $x = 2$ .

Yaay!!

Berlekamp-Welsh algorithm decodes correctly when  $k$  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?  $n + 2k$

Why?

$k$  changes to make diff. messages overlap

How to encode? With polynomial,  $P(x)$ . Of degree?  $n - 1$ .

Recover?

Reconstruct error polynomial,  $E(x)$ , and  $P(x)$ !

**Nonlinear equations.**

Reconstruct  $E(x)$  and  $Q(x) = E(x)P(x)$ . Linear Equations.

Polynomial division!  $P(x) = Q(x)/E(x)$ !

Reed-Solomon codes. Welsh-Berlekamp Decoding. Perfection!

## Summary:ideas.

Any  $d + 1$  points correspond to one polynomial of degree  $\leq d$ .

Any  $d + 1$  points give you back the polynomial.

Can give out  $n \gg d + 1$  points, and any  $d + 1$  gives full information.

Recover Information:

Erasur tolerance  $n + k$ , can lose any  $k$ .

Secret Sharing:  $n$  pieces, any  $k$  recovers.

Recover from Corruptions:

Send more information:  $n + 2k$

$k$  errors,  $n + k$  are correct

$\implies$  and only one degree  $n - 1$  polynomial consistent.

(Use pigeonhole principle.)

Efficiency:

Can fix  $k$  bad equations by multiplying by error polynomial of degree  $k$ .

A polynomial times a polynomial is a polynomial!

$n + 2k$  coefficients in all,  $n + 2k$  correct equations.