CS 70, July, 9, 2020

Question of the day

- I want to send a message containing n Packets (numbers).
- The network I am using corrupts k of the Packets.
- we don't know which k Packets are corrupted.
- k is fixed regardless of the length of the message.
- what is the minimum number of Packets I need to send to recover the original message.
- should I send redundant Packets?

Error correcting codes

Today: messaging through an unreliable channel

messages are composed of packets.

Errors:

1. Lost or dropped packets Erasure errors

   Erasure codes, Tolerate packet drops.

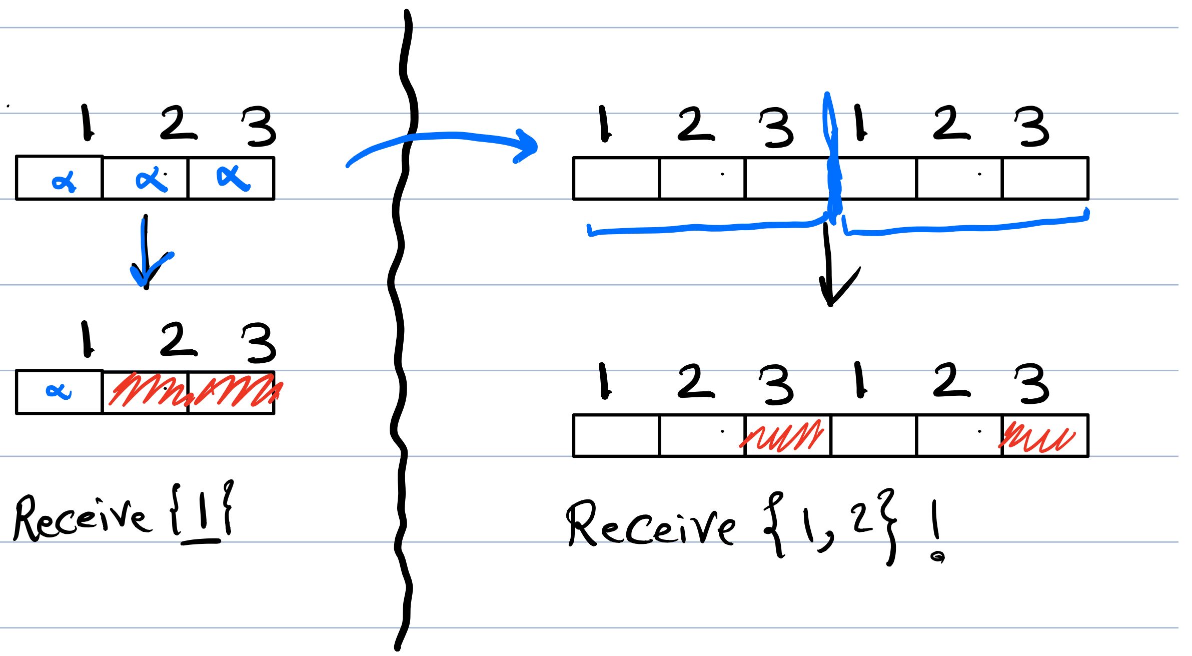
2. Corrupted packets:

   Error correction codes, Tolerate errors in the packet

Error correcting codse: { Algebraic → Polynomials
Combinatorial → Graph Theory

Redundancy

Example:



Erasure Errors:

Original message

1	2	3	4	5
4	1	0	3	4

$\longrightarrow$

Received message

1	2	3	4	5
	1	0		4

$\{(1,4),(2,1),(3,0),(4,3),(5,4)\} \longrightarrow \{(2,1),(3,0),(5,4)\}$

In general:

$\underline{n}$ Packet message, channel that loses $\underline{K}$ Packets

Solution? we can send more Packets!

Redundancy: $\underline{n \times (K+1)}$, need $K+1$ Copies for each Packet

Total Packets: $n \times (K+1)$ ←

Can we do better? Yes! Polynomials

Original message: n Points $\boxed{(1,m_1),(2,m_2),\ldots,(n,m_n)}$

- n Points ⟶ $P(x)$ of degree $n-1$
- Remember: any n Points on $P(x)$ is sufficient to reconstruct $P(x)$.
- Evaluate $P(x)$ on $n+k$ Points.
- The received message has $n+K-K=n$
- Reconstruct $P(x)$ using the n received Packets

The message is: $\underbrace{P(1),\ldots,P(n)}$

Problem: want to send a message with n packets

Channel: loss channel: loses K packets

Question: Can you send $n+K$ packets and recover the message?

Erasure coding scheme: message $= m_1, \ldots, m_n$

Each packet has b bits $\rightarrow 0 \leq m_i \leq 2^b - 1$

Finite Field $GF(P) \rightarrow P \geq 2^b$, $P \geq n+K$.

1. construct $P(x)$ of degree $n-1$ using

$$P(i) = m_i \quad 1 \leq i \leq n$$

2. Send the message $\{P(1), \ldots, P(n+K)\}$

3. Reconstruct $P(x)$ from any received n packets

4. Recover the message $\{P(1), \ldots, P(n)\}$

Error correction:

Noisy channel: corrupts k packets

Challenge: Finding which packets are corrupt.

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.

$P(1) = m_1, \dots, P(n) = m_n$

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

Recieved values: $r_1, r_2, \dots, r_{n+2k}$

Properties:

(1) $P(i) = r_i$ for at least $n+k$ points

(2) $P(x)$ is a degree $n-1$ polynomial that contains $\geq n+k$ of received point. $P(x)$ is unique.

Why is $P(x)$ unique?

Proof: Assume $Q(x)$ is a degree $n-1$ polynomial where $Q(i) = r_i$ for $\geq n+k$ out of $n+2k$

$\begin{cases} Q(i) = r_i \text{ for } n+k \text{ times} \\ P(i) = r_i \text{ for } n+k \text{ times} \end{cases} \Rightarrow$ total Points contained by Q and P $\Rightarrow \underline{2n+2k}$

- Total number of Points to choose from: $\underline{n+2k}$

- At least at n Points $Q(i) = P(i) = r_i$ } $Q(x) = P(x)$
  $Q(x)$ and $P(x)$ are degree $\underline{n-1}$

Brute Force Algorithm:

- For each subset of $n+k$ Points
  Fit degree $n-1$ Polynomial, $Q(x)$ n of $n+k$ Points

- Check if Consistent with $n+k$ of the total Points

- If yes Out Put $Q(x)$

For a subset of $n+k$ Points where $\underline{r_i = P(i)}$
method will reconstruct $P(x)$.

- $Q(x)$: Unique degree $n-1$ that fits n Points
- $Q(x)$: Consistent with $n+k$ Points

$$P(x) = Q(x)$$

Example: $n=3, K=1 \Rightarrow n+2K=5$

Received $r_1=3, r_2=1, r_3=6, r_4=0, r_5=3$.

Find $P(X)=a_2X^2+a_1X+a_0$ that contains

$n+K=3+1=4$ Points.

$$\begin{cases} a_2+a_1+a_0=3 \;\times \\ \boxed{\begin{array}{l} 4a_2+2a_1+a_0=1 \\ 2a_2+3a_1+a_0=6 \\ 2a_2+4a_1+a_0=0 \end{array}} \\ 4a_2+5a_1+a_0=3 \end{cases} \pmod 7$$

Assume Point 1 is wrong and solve → no consistent solution

Assume Point 2 is wrong and solve → Contains the solution → exercise!

In general:

$\boxed{P(x)=a_{n-1}x^{n-1}+\dots+P_0}$ with $r_1, \dots, r_m = n+2K$

$\boxed{P(x)=\sum_{i=0}^{n-1} a_i x^i}$

$\boxed{P \geq n+2K}$

$$\left.\begin{array}{l} P(1)=\sum_{i=0}^{n-1} a_i \equiv r_1 \\ P(2)=\sum_{i=0}^{n-1} 2^i a_i \equiv r_2 \\ \vdots \qquad\qquad \vdots \\ P(n+2K)=\sum_{i=0}^{n-1} m^i a_i \equiv r_m \end{array}\right\} \bmod P$$

→ K of these equations are not correct

How to find the error?

Try all combinations: number of ways to choose $n+k$ out of $n+2k$ $\binom{n+2k}{n+k}$

exponential!

counting

How to find the packets efficiently?

$$\left.\begin{aligned} P(1) &= \sum_{i=0}^{n-1} a_i \equiv r_1 \\ P(2) &= \sum_{i=0}^{n-1} 2^i a_i \equiv r_2 \\ &\vdots \\ P(n+2k) &= \sum_{i=0}^{n-1} m^i a_i \equiv r_m \end{aligned}\right\} \bmod p$$

→ k of them are not satisfied.

Idea: multiply equation i by 0 iff $\boxed{P(i) \neq r_i}$

⇒ All equations are satisfied

which one to multiply by 0? we don't know this!

we will use another Polynomial

Assume errors are at $e_1, e_2, \dots, e_k \Rightarrow \{1, 2, \dots, n+2k\}$

Define Error locator Polynomial:

$$E(x) = (x-e_1) \cdots (x-e_k)$$

$$E(e_i) = 0 \qquad i = 1, \dots, k$$

So

$$\begin{cases} E(1)P(1) = \sum_{i=0}^{n-1} a_i \equiv r_1 E(1) \\ E(2)P(2) = \boxed{\sum_{i=0}^{n-1} 2^i a_i \equiv r_2} E(2) \mod P \\ \vdots \qquad \vdots \\ E(n+2k)P(n+2k) \sum_{i=0}^{n-1} m^i a_i \equiv r_m \, E(n+2k) \end{cases}$$

$$\left\{ \boxed{P(x) = \sum_{i=0}^{n} a_i x^i} \right.$$

$E(x) = (x-e_1)\cdots(x-e_k) = x^k + b_{k-1}x^{k-1} + \dots + b_0$ → k unknowns

$P(x)$ n unknowns.

$P(x)E(x)$
$\Rightarrow a_i \; b_i$

we have $n+2k$ equations and $n+k$ unknowns!

$\Rightarrow n+2k$ (nonlinear equation!)

Scary!

Define: $Q(x) = E(x)P(x) = a'_{n+k-1}x^{n+k-1} + \dots + a'_0$

Equations: $\boxed{Q(i) = r_i E(i)}$ → $n+k$

linear a'_i

Q : $n+k$ unknowns

E : k unknowns

$\Rightarrow \begin{cases} n+2k \text{ equations} \\ n+2k \text{ unknowns} \end{cases}$

To summarize

$$Q(1) = \sum_{i=0}^{n+k-1} a'_i \equiv r_1 \Big(1 + \sum_{j=0}^{k-1} b_j\Big) \quad E(1)$$

$$Q(2) = \sum_{i=0}^{n+k-1} 2^i a_i \equiv r_2 \Big(1 + \sum_{j=0}^{k-1} 2^{k-1-j} b_j\Big) \quad E(2)$$

(mod P)

$$\vdots \qquad \vdots \qquad \vdots$$

$$Q(n+2k) = \sum_{i=0}^{n+k-1} m^i a'_i \equiv r_m \Big(1 + \sum_{j=0}^{k-1} m^{k-1-j} b_j\Big) \quad E(n+2k)$$

$m = n_{+2k}$

Example: $r_1 = 3,\ r_2 = 1,\ r_3 = 6,\ r_4 = 0,\ r_5 = 3$, $n = 3,\ k = 1$

$$Q(x) = E(x)P(x) = a_3 x^3 + a_2 x^2 + a_1 x + a_0 \to 4$$

$$E(x) = x - b_0 \to 1$$

Then $Q(i) = r_i E(i)$

$$\begin{cases} a_3 + a_2 + a_1 + a_0 \equiv 3(1 - b_0) \\ a_3 + 4a_2 + 2a_1 + a_0 \equiv 1(2 - b_0) \\ 6a_3 + 2a_2 + 3a_1 + a_0 \equiv 6(3 - b_0) \\ a_3 + 2a_2 + 4a_1 + a_0 \equiv 0(4 - b_0) \\ 6a_3 + 4a_2 + 5a_1 + a_0 \equiv 3(5 - b_0) \end{cases} \pmod 7$$

$a_3 = 1,\ a_2 = 6,\ a_1 = 6,\ a_0 = 5,\ \boxed{b_0 = 2}$

So $\begin{cases} Q(x) = x^3 + 6x^2 + 6x + 5 \\ E(x) = x - 2 \end{cases}$ Long division

$$Q(x) = P(x)E(x) \Rightarrow P(x) = \frac{Q(x)}{E(x)} \Rightarrow \boxed{P(x) = x^2 + x + 1}$$

Error Correction: Berlekamp-welch

Message: $m_1, \dots, m_n$

Sender:

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

Receiver :

1. Receive: $r_1, \dots, r_{n+2k}$

2. Solve $n+2k$ equations $Q(i) = E(i) R(i)$ $\quad r_i$
to find $Q(x) = E(x) P(x)$ and $E(x)$

3. Compute $P(x) = Q(x) / E(x)$

4. Compute $P(1), \dots, P(n)$

The solution always exist. since the solution is constructed this way.

Question: what if the $n+2k$ equations not independent?

(when there are less than k errors)

Assum there is another solution $Q'(x), E'(x)$

Do we have $\frac{Q'(x)}{E'(x)} = \frac{Q(x)}{E(x)} = P(x)$?

We have $\begin{cases} E'(i) \quad Q(i) = r_i E(i) \\ E(i) \quad Q'(i) = r_i E'(i) \end{cases}$ $\quad 1 \le i \le n+2k$

$$\Rightarrow \begin{cases} Q(i) E'(i) = r_i E(i) E'(i) \\ Q'(i) E(i) = r_i E'(i) E(i) \end{cases} \quad 1 \le i \le n+2k$$

$\Rightarrow Q(i)E'(i) = Q'(i)E(i) \quad 1 \le i \le n+2k$

$\begin{cases} Q(x)E'(x) \rightarrow \text{are equal at } n+2k \\ \underbrace{Q'(x)}_{n+k-1}\underbrace{E(x)}_{k} \rightarrow \text{are degree } n+2k-1 \end{cases}$

$\Rightarrow Q(x)E'(x) = Q'(x)E(x)$

divide by $E(x)E'(x)$

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

Summary:

Any $d+1$ points $\rightarrow$ a unique degree d polynomial

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

Recover information.

Erasure tolerance $n+k$, can lose any k

Secret sharing: n people, any k recover

Recover from corruptions:

- Send more information: $n+2k$
- k errors, $n+k$ are correct
- only one degree $n-1$ polynomial consistent
- can fix k bad equations by multiplying by error polynomial.
- A polynomial times a polynomial is a polynomial,
- $n+2k$ coefficients in all, $n+2k$ correct equations