CS70 - Lecture 11 Notes

Name: Felix Su   SID: 25794773

Spring 2016   GSI: Gerald Zhang

Secret Sharing

Minimality

- Use mod p space where p is prime
- $p > n$ where n is the amount of shares you want to hand out
- $p > 2^b$ where b is the number of bits you want in your secret
- Uses **Theorem**(There is always a prime between n and $2n$). This strategy chooses a p that is within 1 bit of secret size (minimality).

Runtime

- Polynomial in terms of k, n, and $\log p$
- Evaluate $k-1$ degree polynomials n times as a system of linear equations, using $\log p$-bit numbers
- Reconstruct secret by solving system of k equations using $\log p$-bit arithmetic.

Counting

- m^{d+1}: $d+1$ coefficients must be $\in \{0, ..., m-1\}$
- m^{d+1}: $d+1$ points with y-values that must be $\in \{0, ..., m-1\}$

Erasure Codes

Solution

- n packet message, loses k packets in channel
- must send $n+k$ packets
- Use n point values to construct an $n-1$ degree polynomial

Erasure Coding Scheme:

1. n packet message: $m_0, m_1, ..., m_{n-1}$
2. Choose prime $p \approx 2^b$ for mod space where each packet has b bits
3. $p > n+k$
4. $P(x) = m_{n-1}x^{n-1} + ... + m_0 \pmod p$
5. Send, $P(1), ..., P(n+k)$

Any n of the $n+k$ packets gives polynomial and the entire message (all coefficients or y-values)

Erasure Coding Example:
Sending
Send message 1, 4, 4 (3 packets, 2 bits)
Make $P(x)$: $P(1) = 1, P(2) = 4, P(3) = 4$
Try mod5 because 5 is the closest prime to $2^b = 4$, but only gives 5 possible shares, so work mod7
Use Lagrange Interpolation
$P(x) = 2x^2 + 4x + 2 \bmod 7$
Send $(0, P(0))(1, P(1))...(6, P(7))$: 6 points
Receieving
Retrieve $P(x)$ using Lagrange or system of linear equations
Need to know which x-value the correct packets correspond to

Error Correction

- Need to recover information sent AND which packets are corrupted
- Send $n + 2k$ packets because if k errors exist, multiple original messages are possible if $< n + 2k$ packets sent.

Reed-Solomon Code:

1. Encoding polynomial $P(x)$ of degree n-1
   - $P(1) = m_1, ..., P(n) = m_n$
   - Can encode with packets as coefficients (check HW6)
2. Use **Lagrange Interpolation** to get $P(x)$
3. Send $(P1), ..., P(n + 2k)$
4. After noisy channel, receive $R(1), ..., R(n + 2k)$
5. $P(i) = R(i)$ for at least $n + k$ points i; $P(i) \neq R(i)$ for k points
6. Do not know where errors occurred
7. $P(x) =$ unique degree $n - 1$ polynomial

Error Locator Polynomial: $\boldsymbol{E(x) = (x - e_1)(x - e_2) \cdots (x - e_k)}$

- Errors at points $e_1, ...e_k$; E(i) = 0 iff $e_j = i$ for some j; $E(x)$ has degree k
- Idea: Multiply equation i by $E(x) = (x - i)$ iff $P(i) \neq R(i)$, but this creates $n + 2k$ **non-linear** equations with n_k unknowns.
- **Solution:** Let $Q(x) = E(x)P(x) = a_{n+k-1}x^{n+k-1} + \cdots + a_0$
  - Now you have $n + 2k$ linear equations $Q(i) = R(i)E(i)$
  - **Find** $E(x)$ **and** $Q(x)$
    * $E(x) = x^k + b_{k-1}x^{k-1} \vdots b_0$ w/ k unknown coefficients
    * $Q(x) = a_{n+k-1}x^{n+k-1} + \cdots + a_0$ w/ $n + k$ unknown coefficients
    * Solve for coefficients of $Q(x)$ and $E(x)$; Total Unknowns: $n + 2k$
  - $\boldsymbol{P(x) = Q(x)/E(x)}$

Brute force: BAD

- Remove every possible combination of k received packets one at a time and form a degree $n + k - 1$ polynomial with remaining $n + k$ points. First consistent solution gives the corrupted packet.
- Runtime: $(n/k)^k$: exponential in k with $\binom{n+2k}{k}$ possibilities

RS Code Example:
Problem:

- Message 3,0,6 : tolerate $k = 1$ errors (send $n + 2k = 5$ packets)
- Lagrange Encoding $P(x) = x^2 + x + 1 \pmod 7$
- Send: $P(1) = 3, P(2) = 0, P(3) = 6, P(4) = 0, P(5) = 3$
- Receive: $R(1) = 3, R(2) = 1, R(3) = 6, R(4) = 0, R(5) = 3$

Solution: Berklekamp-Welsh Algorithm

- $Q(x) = E(x)P(x) = a_3x^3 + a_2x^2 + a_1x + a_0$
- $E(x) = x - b_0$
- $Q(i) = R(i)E(i)$

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

- Gaussian Elimnation: $a_3 = 1, a_2 = 6, a_1 = 6, a_0 = 5$; $b_0 = 2$
- $Q(x) = x^3 + 6x^2 + 6x + 5$
- $E(x) = x - 2$
- Polynomial Long Division: $P(x) = Q(x)/E(x) = x^2 + x + 1 \pmod 7$

$$\begin{array}{r|l}
 & x^2 + 8x + 22 \\
\hline
x - 2 & x^3 + 6x^2 + 6x + 5 \\
 & -x^3 + 2x^2 \\
\hline
 & 8x^2 + 6x \\
 & -8x^2 + 16x \\
\hline
 & 22x + 5 \\
 & -22x + 44 \\
\hline
 & 49
\end{array}$$

- **Message = 3,0,6**
- RS Code: $P(x) = x^2 + x + 1 \pmod 7$ where $P(1) = 3, P(2) = 0, P(3) = 6$