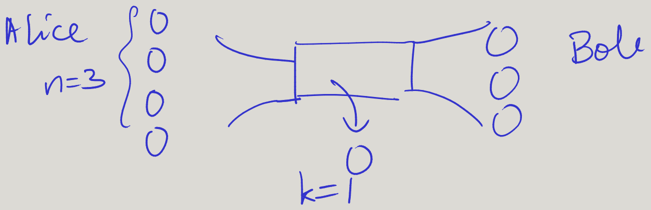
1) Drop packets: Erasure codes

Send n packets, but drops k packets.



1 2 3

1 1 1 2 2 2 3 3 3 $n(k+1) > n+k$

~~$a_0 + a_1 x + a_2 x^2 \mod p$~~

$(x, p(x))$

$(0, p(0))$

$(1, p(1))$

$(2, p(2))$

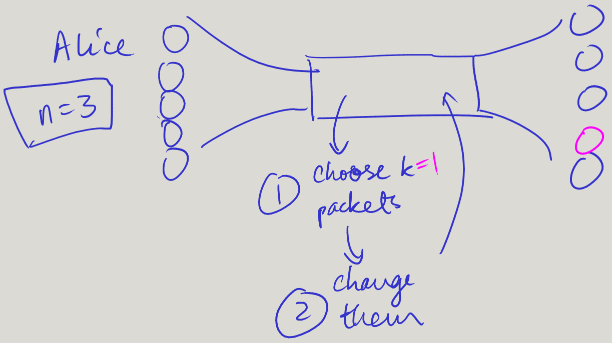
$(3, p(3))$

$(0, P(0))$

$(1, P(1))$

$(2, P(2))$

2) Error-correction



① Find locations of errors

② Disregard the erroneous locations

$E(x)$ deg k

deg 2

$(x_i, p(x_i))$

$(0, p(0))$

~~$(1, p(1))$~~ → $(1, 5)$

$(2, p(2))$

$(3, p(3))$

$(4, p(4))$

$E(x_i) = 0$ if i^{th} packet corrupted

$E(x_1)\, P(x_1) \equiv r_1\, E(x_1)$

$E(x_2)\, P(x_2) \equiv r_2\, E(x_2)$

$E(x_i)\, P(x_i) \equiv r_i\, E(x_i)$

$E(x_{..})\, P_{n+2k}(x_{n+2k}) \equiv r_{n+2k}\, E(x_{n+2k})$

$$Q(x) = \underbrace{P(x)}_{\text{actual poly}}\overbrace{E(x)}^{\text{errloc.}}$$

$$\boxed{P(x) = \frac{Q(x)}{E(x)},}$$ $E(x)$

$n + 2k$, k dropped, no corruption.

CS 70 Discrete Mathematics and Probability Theory
Summer 2020 Course Notes **DIS 3D**

1 Berlekamp-Welch Algorithm — on your own

In this question we will send the message $(m_0, m_1, m_2) = (1, 1, 4)$ of length $n = 3$. We will use an error-correcting code for $k = 1$ general error, doing arithmetic over GF(5).

(a) Construct a polynomial $P(x) \pmod 5$ of degree at most 2, so that

$$P(0) = 1, \qquad P(1) = 1, \qquad P(2) = 4.$$

What is the message $(c_0, c_1, c_2, c_3, c_4)$ that is sent?

(b) Suppose the message is corrupted by changing c_0 to 0. Set up the system of linear equations in the Berlekamp-Welch algorithm to find $Q(x)$ and $E(x)$.

(c) Assume that after solving the equations in part (b) we get $Q(x) = 4x^3 + x^2 + x$ and $E(x) = x$. Show how to recover the original message from Q and E.

2 Secret Veto

In the usual secret-sharing scenario we consider (for instance) a secret vault at the United Nations, which we want to design with the property that any k representatives can pool their information and open it, but any smaller number has no hope of doing so. Assume that the solution in the notes has been implemented, so that the key is some number s, and each member has been assigned a number $f(i) \mod q$ for some degree $k-1$ polynomial f with coefficients in GF(q) and satisfying $f(0) = s$.

(a) A group of $k + \ell$ representatives get together to discuss opening the vault. What will happen if ℓ representatives are opposed to opening the vault and, instead of revealing their true numbers,

Send k pack. & corrupt ℓ packets, Send $k+2\ell$

How many not corrupted $k+\ell$

secretly reveal some *different* numbers from GF(q)? Will the group be able to open the vault? If so, how long will it take? Try every subset of size k.

Can~~not~~ open: ~~$k-\ell$~~ correct
k correct

Why not use BW? BW needs $k+\ell/2$ correct packets

(b) Repeat part (a) in the event that only $\ell/2$ of the ℓ representatives in opposition reveal different numbers than they were assigned—assume that ℓ is even.

Yes, we can recover using BW $k+\ell/2$ correct packets ✓

3 Berlekamp-Welch Algorithm with Fewer Errors — On your own

In class we derived how the Berlekamp-Welch algorithm can be used to correct k general errors, given $n+2k$ points transmitted. In real life, it is usually difficult to determine the number of errors that will occur. What if we have less than k errors? This is a follow up to the exercise posed in the notes.

Suppose Alice wants to send 1 message to Bob and wants to guard against 1 general error. She decides to encode the message with $P(x) = 4$ (on GF(7)) such that $P(0) = 4$ is the message she want to send. She then sends $P(0), P(1), P(2) = (4,4,4)$ to Bob.

(a) Suppose Bob receives the message $(4,5,4)$. Without performing Gaussian elimination explicitly, find $E(x)$ and $Q(x)$.

(b) Now, suppose there were no general errors and Bob receives the original message $(4,4,4)$. Show that the $Q(x), E(x)$ that you found in part (a) still satisfies $Q(i) = r_iE(i)$ for all $i = 0,1,2$.

(c) Verify that $E(x) = x$, $Q(x) = 4x$ is another possible set of polynomials that satisfies $Q(i) = r_iE(i)$ for all $i = 0,1,2$.

(d) Suppose you're actually trying to decode the received message $(4,4,4)$. Based on what you showed in the previous two parts, what will happen during row reduction when you try to solve for the unknowns?

(e) Prove that in general, no matter what the solution of $Q(x)$ and $E(x)$ are though, the recovered $P(x)$ will always be the same.

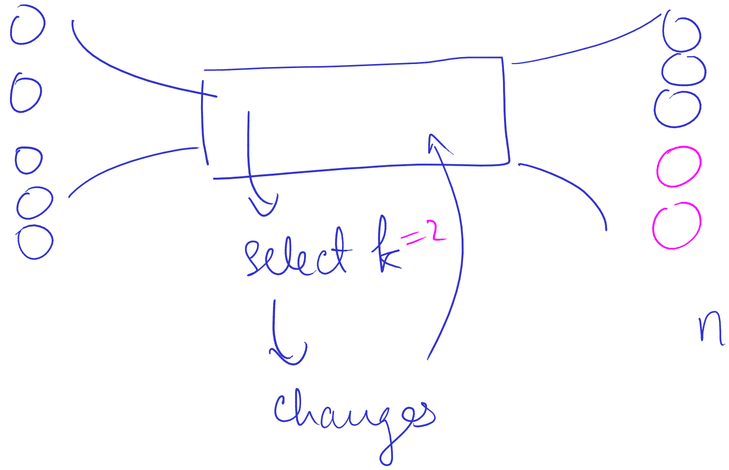
4 Error-Detecting Codes

Suppose Alice wants to transmit a message of n symbols, so that Bob is able to *detect* rather than *correct* any errors that have occurred on the way. That is, Alice wants to find an encoding so that Bob, upon receiving the code, is able to either

(I) tell that there are no errors and decode the message, or

(II) realize that the transmitted code contains at least one error, and throw away the message.

Assuming that we are guaranteed a maximum of k errors, how should Alice extend her message (i.e. by how many symbols should she extend the message, and how should she choose these symbols)? You may assume that we work in GF(p) for very large prime p. Show that your scheme works, and that adding any lesser number of symbols is not good enough.



1 2 3 6

$$n\left\{\begin{matrix}P(0)\\P(1)\\P(2)\end{matrix}\right.$$

$$k\left\{\begin{matrix}P(3)\\P(4)\end{matrix}\right.$$

$n+k$ packets $\longrightarrow$ $n+k$ pts $\longrightarrow$ deg $n-1$ poly

1. Bob selects first n packets.

2. Do Lagrange Interp. $\Longrightarrow P(x)$

3. See if $P(x_i) = r_i \ \forall i$

① If message is not corr. then $P(x_i) = \overset{i^{th}\text{ received packet}}{r_i} \ \forall i \in [0, n+k-1]$

$\deg P(x) = n-1$

② If $P(x_i) = r_i \forall i$ then message is not corr.

corruption $\Rightarrow$ $\exists i : P(x_i) \neq r_i$

1. The first n packets not corrupted.

2. One of first n corrupted. $\Rightarrow g(x) \neq \underline{p(x)}$