ELG 5372 Error Control Coding

Lecture 11: Erasure Decoding, Modifications to Linear Codes, and Introduction to Cyclic Codes

uOttawa

L'Université canadienne
Canada's university

Université d'Ottawa | University of Ottawa

Erasure Decoding

- An erasure is a symbol where the probability of error is high.
  - For example, in BPSK, if the decision variable is close to 0, the certainty of the detection is low. Decoder may declare this symbol to be an erasure.
  - In packet transmissions, codewords may be interleaved over multiple packets. If one packet is not received, then the symbols contained in that packet are "erased".
- When the decoder declares an erasure, then we essentially have a symbol error with a known location.

Erasure decoding (2)

- Consider the all 0 codeword in Hamming (7,4). Assume we receive the following:
  - 000X00X, where X is an erasure.
  - 0000000 and 0001000 0000001 or 0001001 are the possible received vectors.
  - Decoding the first yields 0000000
  - Decoding the second yields 0000000
  - Decoding the third yields 0000000
  - Decoding the fourth yields 0001101
- Comparing all to the non-erased bits of the received word, the fourth has 1 bit different, where the other three are the same. Therefore, the decoder outputs 0000000.

Erasure decoding 3

- We don't need to consider all combinations.
- Only when replacing all erasures with 0 and all erasures with 1 do we get distinct outputs from the decoder.
  - Binary erasure decoding algorithm

  1. Place 0's in all erased coordinates and decode as $\mathbf{c}_0$.
  2. Place 1's in all erased coordinates and decode as $\mathbf{c}_1$
  3. Output codeword for which HD($\mathbf{c}_i$,$\mathbf{r}$) is minimum.

Erasure capability of code

- Consider a linear block code with minimum distance d_{min}.
- A single erased symbol leaves a code with minimum distance at least d_{min}-1.
- Therefore f erased symbols can be filled provided $f < d_{min}$.
  - In previous example, assuming no errors in the non erased bits, only 1 codeword has all zeros in the non-erased bits.
- If there are errors as well as erasures: For a code experiencing f erasures, then the minimum distance for the code left by the non-erased symbols is at least $d_{min}-f$.
- The number of errors that can be corrected is:

$$t_f = \lfloor (d_{\min} - f - 1)/2 \rfloor$$

- Therefore $2e + f < d_{min}$.

Why does binary erasure algorithm work?

- Suppose we have f erasures and e errors, such that $2e+f < d_{min}$.
- Replacing all erasures by 0 introduces e_0 errors into the received codeword, therefore we have $e+e_0$ total errors. Also $e_0 \leq f$.
- Replacing all erasures by 1 introduces e_1 errors into the received codeword, therefore we have $e+e_1$ total errors. Also $e_1 \leq f$ and $e_0+e_1 = f$.
- In the worst case, $e_0 = e_1 = f/2$. therefore both words to be decoded contain $e+f/2$ errors.
- If $e_0 \neq e_1$, then there will be one of the words that has less than $e+f/2$ errors. $2(e+f/2) = 2e+f < d_{min}$. Therefore, there is always one that is below the error correcting capability of the code.

Non Binary Erasure Decoding

- For non binary codes, erasure decoding is more complicated and depends on the structure of the code
- Erasure decoding is popular for decoding of RS codes.
- Erasure decoding of RS codes will be discussed later in the course.

Modifications to Linear Codes

- Extending a code
  - An ($n.k,d$) code is extended by adding an additional redundant coordinate to produce an (n+1,k,d+1) code.
  - For example we can use even parity to extend Hamming (7,4) to an (8,4) code with d_{min} = 4.

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} \quad \mathbf{H} = \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Modifications to Linear Codes 2

- A code is punctured by deleting one of its parity bits
  - A (n,k) code becomes an (n-1, k) code.
  - If the punctured symbol is in a non-zero coordinate of the minimum weight codeword, the minimum distance will also be reduced by 1.
  - Puncturing corresponds to removing a column from the generator matrix.

Modifications to Linear Codes 3

- Expurgating a code means to produce a new code by deleting some of its codewords
  - (n,k) -> (n, k-1).
  - The results may or may not be a linear block code.
  - The minimum distance cannot decrease, but it may increase.
- Augmenting a code is achieved by adding codewords.
  - (n,k) -> (n,k+1)
  - New code may or may not be linear
  - Distance may decrease.
- A code is shortened by deleting a message symbol:
  - (n,k) -> (n-1, k-1)
- A code is lengthened by adding a message symbol
  - (n,k) -> (n+1, k+1)

Introduction to Cyclic Codes

- For linear block codes, the standard array (or the syndrome lookup) can be used for decoding.
- However, for long codes, the storage and computation time of this method can be prohibitive.
- There is no mechanism by which we can design a generator matrix (or parity check matrix) to achieve a given minimum distance.
- Cyclic codes are based on polynomial operations.

Basic Definitions

- Let $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$ be a codeword.
- Let $\mathbf{c}^R = (c_{n-1}, c_0, c_1, \ldots, c_{n-2})$ be a right cyclic shift of $\mathbf{c}$.
- Let $\mathbf{c}^L = (c_1, c_2, \ldots, c_{n-1}, c_0)$ be a left cyclic shift of $\mathbf{c}$.
- We can show that $\mathbf{c}^L = \mathbf{c}^{RR\ldots R}$ (n-1 times).
- Definition of a cyclic code
  - Let C be a linear $(n.k)$ block code. C is a cyclic code if for every codeword $\mathbf{c}$ in C, then $\mathbf{c}^R$ is also in C.

Example

codeword	HW(**c**)	codeword	HW(**c**)
0000000	0	1000110	3
0001101	3	1001011	4
0010111	4	1010001	3
0011010	3	1011100	4
0100011	3	1100101	4
0101110	4	1101000	3
0110100	3	1110010	4
0111001	4	1111111	7

It is easy to see that this code is a cyclic code

Polynomial representation

- $\mathbf{c} = (c_0, c_1, \ldots, c_{n\text{-}1}) \rightarrow c(x) = c_0+c_1x+\ldots+c_{n\text{-}1}x^{n\text{-}1}$.
- A shift left (not cyclic) is thus $\mathrm{xc(x)} = c_0x+c_1x^2+\ldots+c_{n\text{-}1}x^n$.
- Vectorially, this is represented as $(0,c_0, c_1, c_2, \ldots, c_{n\text{-}1})$
- Let $p(x)$ be a polynomial and let $d(x)$ be a divisor. Then $p(x) = q(x)d(x) + r(x)$, where $q(x)$ is the quotient and $r(x)$ is the remainder.
- $\mathbf{c}^R = (c_{n\text{-}1}, c_0, \ldots\ c_{n\text{-}2}) \rightarrow c^R(x) = c_{n\text{-}1}+c_0x+\ldots+c_{n\text{-}2}x^{n\text{-}1}$.
- Therefore $x\mathrm{c}(x) = c^R(x) + c_{n\text{-}1}x^n - c_{n\text{-}1}$.
- Or $xc(x) = c_{n\text{-}1}(x^n\text{-}1) + c^R(x)$.
- $c^R(x)$ is the remainder when we divide $xc(x)$ by $x^n\text{-}1$.
- $c^R(x) = xc(x)\ \mathrm{mod}(x^n\text{-}1)$.

Example

- $(0001101) = x^3+x^4+x^6$.
- $xc(x) = x^4+x^5+x^7$.
- $(x^7+x^5+x^4) \div (x^7+1) = 1$ remainder $1+x^4+x^5 = (1000110)$.

Rings

- A ring *R* is a set with two binary operations defined on it (+ and •) such that

1. *R* is a commutative group over +. The additive identity is denoted by 0.
2. The • operation (multiplication) is associative (a•b) •c = a•(b•c).
3. The left and right distributive laws apply:
    - a•(b+c) = a•b+a•c
    - (a+b) •c = a•c+b•c
4. The ring is said to be a commutative ring if a•b = b•a for every a and b in *R*.
    - The ring is a ring with identity if there exists a multiplicative identity denoted as 1.
    - Multiplication need not form a group and there may not be a multiplicative inverse

Rings (2)

- Some elements in a ring with identity may have a multiplicative inverse.
- For an a in *R*, if there exist another element such that $a \bullet a^{-1} = 1$, then a is referred to as a unit of *R*.
- Example: Z_4

+	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

•	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

- Although Z4 does not form a group over multiplication, it does satisfy the requirement to be a ring over + and •

Rings (3)

- We can also show that GF($q = p^m$) form rings over + and •.

+	0	1
0	0	1
1	1	0

×	0	1
0	0	0
1	0	1

For example, it is easy to show that GF(2) satisfies the requirements of being ring.

Rings of Polynomials

- Let R be a ring and let $f(x)$ be a polynomial of degree n with coefficients in R. ($a_n \neq 0$).

$$f(x) = \sum_{i=0}^{n} a_i x^i$$

- The symbol x is called an indeterminate.
- The set of all polynomials with indeterminate x and coefficients in R form a ring called a polynomial ring (arithmetic is defined as in R).
  - We denote this as $R[x]$.

Examples

- $Z_4[x]$ contains all polynomials with coefficients from Z_4.
  - $(2+3x) + (1+2x+x^3) = 3+x+x^3$.
  - $(2+3x)(1+2x+x^3) = 2+2x^3+3x+2x^3+3x^4 = 2+3x+3x^4$.
- GF(2)[x] is a ring of polynomials whose coefficients are either 0 or 1 with operations in modulo-2 arithmetic.
  - $(1+x)(1+x) = 1+x^2$.
  - $(1+x+x^3)(1+x^2+x^3)(1+x) = 1+x^7$.
  - $(1+x+x^2)+(x+x^3) = 1+x^2+x^3$.

Quotient Rings

- Consider the ring of polynomials GF(2)[x].
- Let S_0 be the set of all polynomials that are divisible by x^n+1.
  - $S_0 = \{0, x^n+1, x^{n+1}+x, x^{n+1}+x^n+x+1,\ldots\}$
  - For simplicity, let n=3.
  - Therefore $S_0 = \{0, x^3+1, x^4+x, x^4+x^3+x+1,\ldots\}$
- Let S_1 be the set of polynomials for which $f(x) \bmod (x^3+1) = 1$
  - $S_1 = \{1, x^3, x^4+x+1, x^4+x^3+x,\ldots\} = 1+S_0$.
- Let S_2 be the set of polynomials for which $f(x) \bmod (x^3+1) = x$.
  - $S_2 = \{x, x^n+x+1, x^4, x^4+x^3,\ldots\} = x+S_0$.

Quotient Rings (2)

- S_3 = all polynomials mod(x^3+1) = x+1 = x+1+S_0.
- S_4 = all polynomials mod(x^3+1) = x^2 = x^2+S_0.
- S_5 = all polynomials mod(x^3+1) = x^2+1 = x^2+1+S_0.
- S_6 = all polynomials mod(x^3+1) = x^2+x = x^2+x +S_0.
- S_7 = all polynomials mod(x^3+1) = x^2+x+1 = x^2+x+1+S_0.
- We can see that S_0-S_7 form the cosets GF(2)[x] under addition.
- Had we taken n = 4, we would have found 16 cosets, n = 5, 32 cosets etc.

Quotient Rings (3)

+	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_0	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_1	S_1	S_0	S_3	S_2	S_5	S_4	S_7	S_6
S_2	S_2	S_3	S_0	S_1	S_6	S_7	S_4	S_5
S_3	S_3	S_2	S_1	S_0	S_7	S_6	S_5	S_4
S_4	S_4	S_5	S_6	S_7	S_0	S_1	S_2	S_3
S_5	S_5	S_4	S_7	S_6	S_1	S_0	S_3	S_2
S_6	S_6	S_7	S_4	S_5	S_2	S_3	S_0	S_1
S_7	S_7	S_6	S_5	S_4	S_3	S_2	S_1	S_0

Quotient Rings (4)

•	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_0	S_0	S_0	S_0	S_0	S_0	S_0	S_0	S_0
S_1	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_2	S_0	S_2	S_4	S_6	S_1	S_3	S_5	S_7
S_3	S_0	S_3	S_6	S_5	S_5	S_6	S_3	S_0
S_4	S_0	S_4	S_1	S_5	S_2	S_6	S_3	S_7
S_5	S_0	S_5	S_3	S_6	S_6	S_3	S_5	S_0
S_6	S_0	S_6	S_5	S_3	S_3	S_5	S_6	S_0
S_7	S_0	S_7	S_7	S_0	S_7	S_0	S_0	S_7

Quotient Rings (5)

- Let $R = \{S_0, S_1, \ldots S_7\}$.
  - R forms a commutative group under + with S_0 as identity.
  - R-S_0 does not form a group under •.
  - Therefore R is not a field.
  - However, R does form a ring, with S_1 as multiplicative identity.