20SK: Exercise #6
Channel Coding (ECC)

Jan Přikryl

January 2, 2018

Introduction

Source encoding: *Remove rendunant information.*
Channel coding: *Protect data against transmission errors.*

Medium: *Controlled redundancy (increase message entropy).*

Always block codes
Linear codes

- Repetition code
- Hamming code
- BCH
- LDPC (Gallager, 1963)
- Cyclic codes (Reed-Solomon)

Other approaches:

- Convolution codes
- Turbo codes

Problem 1

1. Assume a repetition code of block length n (n is odd) being used to transmit data over BSC with crossover probability ϵ with simple majority vote decoding.
2. Determine the probability of uncorrected error in a block of length n

Problem 1

1. Assume a repetition code of block length n (n is odd) being used to transmit data over BSC with crossover probability ϵ with simple majority vote decoding.
2. Determine the probability of uncorrected error in a block of length n

   2.1 we will decode an incorrect bit if more than $(n + 1)/2$ bits of the code word are corrupted

Problem 1

1. Assume a repetition code of block length n (n is odd) being used to transmit data over BSC with crossover probability ϵ with simple majority vote decoding.
2. Determine the probability of uncorrected error in a block of length n
    2.1 we will decode an incorrect bit if more than $(n+1)/2$ bits of the code word are corrupted
    2.2 given ϵ, probability that all k bits, $k < n$, will flip is $\epsilon^k(1-\epsilon)^{n-k}$

Problem 1

1. Assume a repetition code of block length n (n is odd) being used to transmit data over BSC with crossover probability ϵ with simple majority vote decoding.
2. Determine the probability of uncorrected error in a block of length n
   2.1 we will decode an incorrect bit if more than $(n+1)/2$ bits of the code word are corrupted
   2.2 given ϵ, probability that all k bits, $k < n$, will flip is $\epsilon^k(1-\epsilon)^{n-k}$
   2.3 given k, the total number of combinations containing k flipped bits a binomic number

Problem 1

1. Assume a repetition code of block length n (n is odd) being used to transmit data over BSC with crossover probability ϵ with simple majority vote decoding.
2. Determine the probability of uncorrected error in a block of length n
   - 2.1 we will decode an incorrect bit if more than $(n+1)/2$ bits of the code word are corrupted
   - 2.2 given ϵ, probability that all k bits, $k < n$, will flip is $\epsilon^k(1-\epsilon)^{n-k}$
   - 2.3 given k, the total number of combinations containing k flipped bits a binomic number
   - 2.4 we have to sum for every possible k from $(n+1)/2$ up to n, hence

$$p(n,\epsilon) = \sum_{k=(n+1)/2}^{n} \binom{n}{k} \epsilon^k (1-\epsilon)^{n-k}.$$

Problem 1 continued

3. Create function `rcbsc(n,ep)` computing probabilities of uncorrected errors for blocks of odd lengths up to n and plotting them using `stem()` into a graph.
4. Q: What would be the optimal length of the repetition code?
5. Q: What would be the code rate and transsmission speed?
6. Q: How does it correspond to Shannon limit theorem? Discuss the difference.

Problem 2

Assume linear code with a generator matrix **G**.

1. Q: Given **G**, what is the length of plain text vector **v**?
2. Q: How are codewords computed?

3. Q: How will you enumerate all codewords?

4. Write function `[C,dmin]=linprop(G)` that computes a binary matrix **C** of all valid codewords for a linear code with generator matrix **G** and determines the minimum code distance d_{min}.

Problem 2

Assume linear code with a generator matrix $\mathbf{G}$.

1. Q: Given $\mathbf{G}$, what is the length of plain text vector $\mathbf{v}$?
2. Q: How are codewords computed?

$$\mathbf{w} = \mathbf{v}\mathbf{G}$$

3. Q: How will you enumerate all codewords?

4. Write function `[C,dmin]=linprop(G)` that computes a binary matrix $\mathbf{C}$ of all valid codewords for a linear code with generator matrix $\mathbf{G}$ and determines the minimum code distance d_{min}.

Problem 2

Assume linear code with a generator matrix $\mathbf{G}$.

1. Q: Given $\mathbf{G}$, what is the length of plain text vector $\mathbf{v}$?
2. Q: How are codewords computed?

$$\mathbf{w} = \mathbf{v}\mathbf{G}$$

3. Q: How will you enumerate all codewords?

$$\mathbf{V} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}, \qquad \mathbf{C} = \mathbf{V}\mathbf{G}$$

4. Write function `[C,dmin]=linprop(G)` that computes a binary matrix $\mathbf{C}$ of all valid codewords for a linear code with generator matrix $\mathbf{G}$ and determines the minimum code distance d_{min}.

Problem 3

1. Hamming encoding and decoding