Linear Block Codes

Telecommunications Laboratory

Alex Balatsoukas-Stimming

Technical University of Crete

October 23rd, 2008

Outline

1. Motivation
2. Linear Binary Codes
   - Hard decision (algebraic) decoding
   - Soft decision decoding
   - Error Probability
   - Systematic Codes
   - Error detecting and error correcting capabilities of a code
   - Standard array and syndrome decoding.
3. Hamming codes
   - Simulation results

Motivation

- To achieve a given QoS (usually expressed as the bit error rate) using uncoded modulation, we require a certain SNR.
- Bandwidth limited channel
  1. Use higher order constellations, for example 8-PSK instead of 2-PSK.
- Power limited channel
  1. We can add redundancy (keeping symbol energy constant).
  2. The modulator is forced to work at a higher rate to achieve the same information bit rate, increasing bandwidth occupation.
- The difference between the SNR required for the uncoded and the coded system to achieve the same BER is called the *coding gain*.

Error correcting strategies

- There are two error correcting strategies:
    - Forward error correction (FEC)
    - Automatic repeat request (ARQ)
        1. Stop-and-wait ARQ (e.g. ABP)
        2. Continuous ARQ (e.g. SRP, Go-Back-N)
- ARQ can only be used if there is a feedback channel.
- When the transmission rate is high, retransmissions happen often, thus introducing delay into the communication.
- For one way channels we can only use FEC.

Linear Binary Codes

Linear Binary Codes

- If $\mathcal{C}$ has the form:

$$\mathcal{C} = \mathbb{F}_2^k \mathbf{G}$$

  where $\mathbf{G}$ is a $k \times n$ binary matrix with $n \geq k$ and rank k, called the generator matrix of $\mathcal{C}$, then $\mathcal{C}$ is called an (n, k, d) linear binary code.
- The code words of a linear code have the form $\mathbf{uG}$ where $\mathbf{u}$ is any binary k-tuple of binary source digits.
- For any $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}$ it can be shown that $\mathbf{c}_1 + \mathbf{c}_2 \in \mathcal{C}$, as follows:

$$\mathbf{c}_1 + \mathbf{c}_2 = \mathbf{u}_1\mathbf{G} + \mathbf{u}_2\mathbf{G} = (\mathbf{u}_1 + \mathbf{u}_2)\mathbf{G} = \mathbf{uG} \in \mathcal{C}$$

- The ratio $r = \frac{k}{n}$ is called the rate of the code.

Linear Binary Codes

- An alternative definition of a linear code is through the concept of an $(n-k) \times n$ parity-check matrix $\mathbf{H}$. A code $\mathcal{C}$ is linear if:

$$\mathbf{Hc} = \mathbf{0} \quad \forall \mathbf{c} \in \mathcal{C}$$

- We define $\mathbf{s} = \mathbf{H}\hat{\mathbf{c}}$ as the syndrome of the received binary codeword $\hat{\mathbf{c}}$ which is the received vector $\hat{\mathbf{x}} \in \mathbb{R}^n$ after hard decisions have been made on each of its components.
- If $\mathbf{s} \neq \mathbf{0}$ then we know that an error has occured.

Encoding Example

- Consider the following $k \times n$ generator matrix ($k = 3, n = 4$):

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

- Each one of the $2^k = 8$ code words have the form $\mathbf{uG}$
- For example, for $\mathbf{u}_1 = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix}$ we get the codeword:

$$\mathbf{c}_1 = \mathbf{u}_1\mathbf{G} = \begin{bmatrix} 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix}$$

Hard decision (algebraic) decoding

- In algebraic decoding, 'hard' decisions are made on each component of the received signal $\mathbf{y}$ forming the vector

$$\mathbf{x}' = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_n)$$

  e.g. for BPSK we have:

$$\hat{x}_i = \text{sign}(y_i)$$

- If the vector $\mathbf{x}'$ is a codeword of $\mathcal{C}$, then the decoder selects $\hat{\mathbf{x}} = \mathbf{x}'$, else the structure of the code is exploited to correct them.
- The method is suboptimal because we discard potentially useful information before using it.

Soft decision decoding

- In soft decision decoding, a Maximum Likelihood (or MAP if codewords are not equally likely) estimation is performed on the whole received vector.

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathcal{C}} p(\mathbf{y}|\mathbf{x}) \text{ (ML)}$$

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}\in\mathcal{C}} p(\mathbf{x}|\mathbf{y}) \text{ (MAP)}$$

- Considerable improvement in performance (usually around 3dB), but more complex implementation.

Hard decision vs. soft decision decoding example (1/2)

- Assume that we have a $(3,1)$ repetition code, that is:

$$\mathbf{x} = (x_1, x_2, x_3) \text{ where } x_2 = x_3 = x_1$$

- The codewords of this code (in the signal space) are:

$$\mathbf{c}_1 = (-1, -1, -1) \text{ and } \mathbf{c}_2 = (+1, +1, +1)$$

- Assume now that transmitted signal is $\mathbf{x} = (+1, +1, +1)$ and the corresponding received vector is $\mathbf{y} = (+0.8, -0.1, -0.2)$

Hard decision vs. soft decision decoding example (2/2)

- Using hard decision decoding, we decide -1 if the majority of the demodulated signals is -1, and $+1$ otherwise.
- The demodulated vector corresponding to the received vector $\mathbf{y}$ is $\hat{\mathbf{y}} = (1, -1, -1)$. Using the majority rule, we decide that $\hat{\mathbf{y}} = \mathbf{c}_1 = (-1, -1, -1)$, thus making a decoding error.
- Using soft decision decoding, we will choose the codeword with the least Euclidean distance from the received vector:

$$d_E^2(\mathbf{y}, \mathbf{c}_1) = (0.8 - 1)^2 + (-0.1 - 1)^2 + (-0.2 - 1)^2 = 2.69$$

$$d_E^2(\mathbf{y}, \mathbf{c}_2) = (0.8 + 1)^2 + (-0.1 + 1)^2 + (-0.2 + 1)^2 = 4.69$$

- So, we correctly choose $\hat{\mathbf{y}} = \mathbf{c}_1 = (-1, -1, -1)$.

Error Probability (1/2)

- Recall that:

$$P(e|\mathbf{x}) \leq \sum_{\hat{\mathbf{x}} \neq \mathbf{x}} e^{-||\mathbf{x}-\hat{\mathbf{x}}||^2/4N_o}$$

- For the simple case of the binary elemental constellation $\mathcal{X} = \{-x, +x\}$, we have:

$$\begin{aligned} d_E^2(\mathbf{c}, \mathbf{c}') &= \sum_i (c_i - c_i')^2 \\ &= \sum_{c_i \neq c_i'} 4x^2 \\ &= 4x^2 d_H(\mathbf{c}, \mathbf{c}') \\ &= 4\mathcal{E} d_H(\mathbf{c}, \mathbf{c}') \end{aligned}$$

Error Probability (2/2)

- Because of the linearity of the code, we have that $\mathbf{c}' + \mathbf{c}'' = \mathbf{c} \in \mathcal{C}$ so, the Hamming distance between $\mathbf{c}$ and $\hat{\mathbf{c}}$ is:

$$d_H(\mathbf{c}, \hat{\mathbf{c}}) = w(\mathbf{c} + \hat{\mathbf{c}}) = w(\mathbf{c}')$$

- So, for the error probability of a single codeword, we have:

$$P(e|\mathbf{c}) \leq \sum_{\hat{\mathbf{c}} \neq \mathbf{c}} e^{-d_H(\mathbf{c}, \hat{\mathbf{c}})\mathcal{E}/N_o} = \sum_{\mathbf{c}^* \neq \mathbf{0}} e^{-w(\mathbf{c}^*)\mathcal{E}/N_o}$$

- The value of the above summation does not depend on $\mathbf{c}$, and hence:

$$P(e|\mathbf{c}) = P(e)$$

Systematic Codes

- A linear code is called systematic if its generator matrix has the form

$$\mathbf{G} = [\mathbf{I}_k \vdots \mathbf{P}]$$

  where $\mathbf{P}$ is a $k \times (n-k)$ matrix.

- The words of these codes have the form

$$\mathbf{c} = \mathbf{uG} = [\mathbf{u} \vdots \mathbf{uP}]$$

- The $(n-k) \times n$ parity check matrix of a systematic code can be constructed as follows

$$\mathbf{H} = [\mathbf{P}^T \vdots \mathbf{I}_{n-k}]$$

Systematic Code Example

- We observe that the generator matrix from the previous example can be written in the form

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} = \mathbf{G} = \left[\begin{array}{c|c} & 1 \\ \mathbf{I}_k & 1 \\ & 1 \end{array} \right] = [\mathbf{I}_k \vdots \mathbf{P}]$$

where $\mathbf{P} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$, so the code is in its systematic form.

- The parity check matrix $\mathbf{H}$ of the above code can be written as

$$\mathbf{H} = [\mathbf{P}^T \vdots \mathbf{I}_{n-k}] = \begin{bmatrix} 1 & 1 & 1 & \vdots & 1 \end{bmatrix}$$

Error detecting capabilities of a code

- The received codeword can be written as $\mathbf{r} = \mathbf{c} + \mathbf{e}$, where $\mathbf{c}$ is the transmitted codeword and $\mathbf{e}$ is called the error pattern.
- A code with a minimum distance $d_{\min}$ is capable of detecting all error patterns of $d_{\min} - 1$ or less errors.
- For error patterns of $d_{\min}$ or more errors, there exists at least one pattern which transforms the transmitted codeword into another valid codeword, so the code is not capable of detecting all of them.
- It can however detect a large fraction of them. If $\mathbf{e} \in \mathcal{C}$, then (because of the linearity of the code) $\mathbf{r} = \mathbf{c} + \mathbf{e} \in \mathcal{C}$. So, there exist $2^k - 1$ error patterns of more than $d_{\min}$ errors which are undetectable, leaving a total of $2^n - 2^k + 1$ detectable error patterns.

Error correcting capabilities of a code (1/2)

- Let t be a positive integer such that

$$2t + 1 \leq d_{\min} \leq 2t + 2$$

- Let $\mathbf{c}$ and $\mathbf{r}$ be the transmitted and the received codeword respectively.
- Let $\mathbf{w} \in \{\mathcal{C} - \{\mathbf{c}, \mathbf{r}\}\}$
- Since the Hamming distance satisfies the triangle inequality, we get

$$d_H(\mathbf{c}, \mathbf{r}) + d_H(\mathbf{r}, \mathbf{w}) \geq d_H(\mathbf{c}, \mathbf{w})$$

- Since $\mathbf{c}$ and $\mathbf{w}$ are codewords of $\mathcal{C}$, we have that

$$d_H(\mathbf{c}, \mathbf{w}) \geq d_{\min} \geq 2t + 1$$

Error correcting capabilities of a code (2/2)

- Suppose that $d_H(\mathbf{c}, \mathbf{r}) = t'$.
- From the above we get that

$$d_H(\mathbf{r}, \mathbf{w}) \geq 2t + 1 - t'$$

- If $t' \leq t$, then

$$d_H(\mathbf{r}, \mathbf{w}) > t$$

- The above tells us that if an error pattern of t or less errors occurs, the received codeword $\mathbf{r}$ is closer to the transmitted codeword $\mathbf{c}$ than to any other codeword $\mathbf{w}$ in $\mathcal{C}$

Standard array

- An array containing all 2^n binary n-tuples which is constructed as follows:

$$\begin{array}{cccc}
\mathbf{c}_1 = \mathbf{0} & \mathbf{c}_2 & \ldots & \mathbf{c}_{2^k} \\
\mathbf{e}_1 & \mathbf{c}_2 + \mathbf{e}_1 & \ldots & \mathbf{c}_{2^k} + \mathbf{e}_1 \\
\mathbf{e}_2 & \mathbf{c}_2 + \mathbf{e}_2 & \ldots & \mathbf{c}_{2^k} + \mathbf{e}_2 \\
\vdots & & & \vdots \\
\mathbf{e}_{2^{n-k}} & \mathbf{c}_2 + \mathbf{e}_{2^{n-k}} & \ldots & \mathbf{c}_{2^k} + \mathbf{e}_{2^{n-k}}
\end{array}$$

  where $\mathbf{c}_i \in \mathcal{C}$ and $\mathbf{e}_i$ are all 2^{n-k} possible error patterns.
- The first column consists of elements called *coset leaders*.

Syndrome decoding (1/2)

- Recall that the syndrome of a received vector is defined as:

$$\mathbf{s} = \mathbf{r}\mathbf{H}^T$$

and that for every codeword $\mathbf{c} \in \mathcal{S}$ it holds that:

$$\mathbf{s} = \mathbf{c}\mathbf{H}^T = \mathbf{0}$$

- All elements of a row of the standard array have the same syndrome:

$$(\mathbf{e}_1 + \mathbf{c}_i)\mathbf{H}^T = \mathbf{e}_1\mathbf{H}^T + \mathbf{c}_i\mathbf{H}^T = \mathbf{e}_1\mathbf{H}^T$$

Syndrome decoding (2/2)

- By computing the syndrome of the received codeword, we can estimate which error pattern occured, namely the error pattern which has the same syndrome as the received vector.
- It is optimal to choose the most likely error patterns as the coset leaders.
- In the case of the AWGN with BPSK modulation, the most likely error patterns for large enough SNR are those with minimum weight.
- After estimating the error pattern, we can correct the error as follows:

$$\hat{\mathbf{c}} = \mathbf{r} + \mathbf{e}_i = (\mathbf{c} + \mathbf{e}_i) + \mathbf{e}_i = \mathbf{c}$$

Hamming codes

Hamming codes

- For any positive integer $m \geq 2$, there exists a Hamming code with the following parameters:
  1. Code length: $n = 2^m - 1$
  2. Number of information symbols: $k = 2^m - m - 1$
  3. Number of parity symbols: $m = n - k$
  4. Error correcting capability: $t = 1 \quad (d_{\text{min}} = 3)$
- Different code lengths can be chosen to achieve a wide variety of rates and performances.
- The parity check matrix $\mathbf{H}$ of a Hamming code consists of all nonzero m-tuples as its columns.

A Hamming code example

- For example, let $m = 3$. We get:
  1. $m = 3$ parity symbols
  2. $n = 2^m - 1 = 2^3 - 1 = 7$ codeword length
  3. $k = 2^m - m - 1 = 2^3 - 3 - 1 = 4$ information symbols

  which is a $(7, 4, 1)$ linear code.
- The parity check matrix $\mathbf{H}$ of this code is:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix} = \left[\begin{array}{c|cccc} & 1 & 0 & 1 & 1 \\ \mathbf{I}_m & 1 & 1 & 1 & 0 \\ & 0 & 1 & 1 & 1 \end{array} \right] = [\mathbf{I}_m \vdots \mathbf{P}^T]$$

- The generator matrix for this Hamming code can be constructed as follows:

$$\mathbf{G} = [\mathbf{I}_k \vdots \mathbf{P}]$$

Simulation results