EE 605: Error Correcting Codes

Instructor: Saravanan Vijayakumaran

Indian Institute of Technology Bombay

Autumn 2011

Solutions to Assignment 2

---

1. Construct the standard array and syndrome decoding table for the $(7,4)$ linear block code with generator matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

   Solution:

   The standard array is given in Table 1 and the syndrome decoding table is given in Table 2. The parity check matrix which is given by

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

   The syndrome decoding table is obtained by multiplying the coset leaders in the standard array by H^T.

2. A *burst error* of length l is an error pattern which causes l consecutive locations in the transmitted code to be corrupted. Let H be the parity check matrix of a binary linear block code.

   (a) What is the necessary and sufficient condition on the columns of H so that every burst error of length up to t can be detected?

   Solution: We know that the syndrome of a received vector is equal to the syndrome of the error pattern. If $\mathbf{r} = \mathbf{v}+\mathbf{e}$ where $\mathbf{v}$ is the transmitted codeword and $\mathbf{e}$ is the error vector, $\mathbf{r}H^T = \mathbf{e}H^T$. Suppose H has no zero columns, then every error pattern of weight 1 will be detected by a scheme which declares an error detection whenever a nonzero syndrome is observed. Suppose no 2 consecutive columns in H add up to zero, then no error pattern with 2 consecutive ones will result in a zero syndrome. Thus every burst of length two can be detected by a scheme which declares an error detection whenever a nonzero syndrome is observed. Similarly if no $3, 4, ..., t$ consecutive columns in H add up to zero, then every burst error of length upto t can be detected. Thus a sufficient condition for detecting a burst of length upto t is that H has no zero columns and the sum of any $2, 3, \ldots, t$ consecutive columns is not zero.

   We claim that the above sufficient condition is also necessary. Consider an error detection scheme which is capable of detecting burst errors of length upto t.

Any error detection scheme for an (n, k) block code C is a partition of the space $\mathbb{F}_2^n$ into two subsets E and E^c such that an error detection is declared when the received vector $\mathbf{r} \in E$. Since the scheme should not declare an error when the error pattern $\mathbf{e}$ is the all zeros vector i.e. when $\mathbf{r} = \mathbf{v}$, we require $C \subseteq E^c$. Thus an error is not detected whenever $\mathbf{r} \in C$. If a burst error of length upto t results in the received vector which is equal to a codeword then this error will not be detected. So it is necessary that $\mathbf{v} + \mathbf{e} \notin C$ for any $\mathbf{v} \in C$ and any burst error $\mathbf{e}$ of length upto t. Since $\mathbf{v} + \mathbf{e} \notin C$ if and only if $\mathbf{e} \notin C$, it is necessary for every burst error $\mathbf{e}$ of length upto t does not belong to C for the error detection scheme to be able to detect it. Since $\mathbf{e} \notin C \iff \mathbf{e}H^T \neq \mathbf{0}$, it is necessary that $\mathbf{e}H^T$ is not equal to zero for any burst error $\mathbf{e}$ of length up to t. Thus it is necessary that there are no nonzero columns in H and no $2, 3, \ldots, t$ consecutive columns of H add up to zero.

(b) What is the necessary and sufficient condition on the columns of H so that every burst error of length up to t can be corrected?

Solution: Suppose that all the columns of H are nonzero and distinct. Also suppose that the sum of any $2, 3, \ldots, t$ consecutive columns add up to a distinct vector. This is sufficient to correct any burst error of length upto t because each one of these burst errors corresponds will result in a distinct syndrome (because it is a sum of columns). If $\mathbf{r} = \mathbf{v} + \mathbf{e}$ where $\mathbf{v}$ is the transmitted codeword and $\mathbf{e}$ is the error vector, $\mathbf{r}H^T = \mathbf{e}H^T = \mathbf{s}$. A burst error vector $\mathbf{e}$ of length upto t can be identified from the syndrome (due to the uniqueness) and the errors introduced can be corrected by adding $\mathbf{e}$ to $\mathbf{r}$.

To prove the necessity of the above condition, consider any error correction scheme capable of correcting burst errors of length upto t. Any error correction scheme for an (n, k) block code is a partition of the space $\mathbb{F}_2^n$ into 2^k subsets $A_i, i = 1, 2, \ldots, 2^k$ each of which corresponds to a codeword. If $\mathbf{r} \in A_i$, we decode it as the codeword $\mathbf{v}_i$. Since the scheme is capable of correcting burst errors of length upto t, it is necessary for $\mathbf{v}_i + \mathbf{e}$ to be in A_i where $\mathbf{e}$ is a burst error of length upto t. Since the A_i's are disjoint, we have $\mathbf{v}_i + \mathbf{e}_1 \neq \mathbf{v}_j + \mathbf{e}_2$ where $i \neq j$ and $\mathbf{e}_1, \mathbf{e}_2$ are burst errors of length upto t. Multiplying both sides by H^T, we get $\mathbf{e}_1 H^T \neq \mathbf{e}_2 H^T$ for any two distinct burst errors of length upto t. If we set $\mathbf{e}_1$ to be a weight one error pattern and $\mathbf{e}_2 = \mathbf{0}$, we see that $\mathbf{e}_1 H^T \neq \mathbf{0}$ which means that all the columns of H are necessarily nonzero. Setting both $\mathbf{e}_1$ and $\mathbf{e}_2$ to be weight one error patterns we see that all the columns of H are necessarily distinct. When $\mathbf{e}$ is an error pattern of weight two or more, $\mathbf{e}H^T$ corresponds to a sum of columns in H which needs to be necessarily distinct for different values of $\mathbf{e}$.

3. Let C be (n, k) linear code. Let T be a set of coordinates of the codewords i.e. $T \subseteq \{1, 2, \ldots, n\}$. Let C^T be the code obtained by puncturing C on the coordinates in T and let C_T be the code obtained by shortening C on the coordinates in T. Prove that

(a) $(C^\perp)_T = (C^T)^\perp$

Solution: Let $u \in (C^\perp)_T$. Then u is obtained by shortening a vector v in $C^\perp$ on the coordinates in T. So v is zero in the coordinates specified by T. Since

$v \in C^{\perp}$, we have $\sum_{i=1}^{n} v_i w_i = 0$ for all $w \in C$ which implies $\sum_{i \in T^c} v_i w_i = 0$ (as $v_i = 0$ for $i \in T$). This implies that the vector x obtained by puncturing v on T is perpendicular to all the codewords in C^T. But $u = x$ as it is the vector obtained by puncturing v on T. Thus $u \in (C^T)^{\perp}$ and we have shown that $(C^{\perp})_T \subseteq (C^T)^{\perp}$.

Let $u \in (C^T)^{\perp}$. Then u is perpendicular to all the codewords $v \in C^T$. Each v is obtained by puncturing a codeword w in C on the coordinates in T. We can extend the vector u to another vector x of length n such that x has zeros in the coordinates in T and is equal to u in the other coordinates. Now x is perpendicular to all the codewords w in C obtained by extending each $v \in C^T$ (because u and v are perpendicular on T and x,w are extended versions of them with x being zero in the new coordinates). Thus $x \in C^{\perp}$. Since x has zeros on T and u can be obtained by puncturing x on T we conclude that $u \in (C^{\perp})_T$. We have shown that $(C^T)^{\perp} \subseteq (C^{\perp})_T$.

Since each set is the subset of the other they have to be equal.

(b) $(C^{\perp})^T = (C_T)^{\perp}$

Solution: Similar argument as in (a).

4. Let C_1 be a (n_1, k_1) binary linear block code with minimum distance d_1 and let C_2 be a (n_2, k_2) binary linear block code with minimum distance d_2. The direct sum of C_1 and C_2 is defined as

$$C_1 \oplus C_2 = \{(\mathbf{c}_1, \mathbf{c}_2) \| \mathbf{c}_1 \in C_1, \mathbf{c}_2 \in C_2\}.$$

Show that $C_1 \oplus C_2$ is a $(n_1 + n_2, k_1 + k_2)$ linear block code with minimum distance $\min(d_1, d_2)$. Derive the generator matrix of $C_1 \oplus C_2$ in terms of the generator matrices of C_1 and C_2. Derive the parity check matrix of $C_1 \oplus C_2$ in terms of the parity check matrices of C_1 and C_2.

Solution: By definition, $C_1 \oplus C_2$ is a nonempty subset of $\mathbb{F}_2^{n_1+n_2}$. To show that it is a linear code of dimension $k_1 + k_2$, we have to first show that it is a subspace of $\mathbb{F}_2^{n_1+n_2}$ over $\mathbb{F}_2$. Consider any two elements $\mathbf{x}$ and $\mathbf{y}$ in $C_1 \oplus C_2$. Then $\mathbf{x} = (x_1, x_2)$ and $\mathbf{y} = (y_1, y_2)$ for some $x_1, y_1 \in C_1$ and $x_2, y_2 \in C_2$. Their sum $\mathbf{x} + \mathbf{y}$ is equal to $(x_1 + x_2, y_1 + y_2)$. This sum belongs to $C_1 \oplus C_2$ because $x_1 + x_2 \in C_1$ and $y_1 + y_2 \in C_2$ (as C_1 and C_2 are linear codes). For any $a \in \mathbb{F}_2$ and $\mathbf{x} \in C_1 \oplus C_2$, $a\mathbf{x} = \mathbf{x}$ or $a\mathbf{x} = (0_{n_1}, 0_{n_2})$ where the former happens when $a = 1$ and the latter happens when $a = 0$. Here 0_{n_i} is a n_i-tuple of zeros. In both cases $a\mathbf{x} \in C_1 \oplus C_2$. Thus $C_1 \oplus C_2$ satisfies the two conditions required for a nonempty subset to be subspace. Hence $C_1 \oplus C_2$ is a linear code.

To show that the dimension of $C_1 \oplus C_2$ is $k_1 + k_2$, consider bases for C_1 and C_2. Let $A = \{a_1, a_2, \ldots, a_{k_1}\}$ be a basis for C_1 and let $B = \{b_1, b_2, \ldots, b_{k_2}\}$ be a basis for C_2. We claim that the set $D = \{(a_1, 0_{n_2}), (a_2, 0_{n_2}), \ldots, (a_{k_1}, 0_{n_2}), (0_{n_1}, b_1), (0_{n_1}, b_2), \ldots, (0_{n_1}, b_{k_2}\}$ is a basis for $C_1 \oplus C_2$. Each element in this set is in $C_1 \oplus C_2$ because $a_i, 0_{n_1} \in C_1$ and $b_i, 0_{n_2} \in C_2$. First we check that this set spans $C_1 \oplus C_2$. Consider any element in $C_1 \oplus C_2$. It is of the form (x, y) where $x \in C_1$ and $y \in C_2$. Then $x = \sum_{i=1}^{k_1} \alpha_i a_i$ where $\alpha_i \in \mathbb{F}_2$ and $y = \sum_{i=1}^{k_2} \beta_i b_i$ where $\beta_i \in \mathbb{F}_2$ because the a_i's form a basis for C_1

and the b_i's form a basis for C_2. We can write $(x, 0_{n_2})$ as

$$(x, 0_{n_2}) = \sum_{i=1}^{k_1} \alpha_i (a_i, 0_{n_2})$$

and $(0_{n_1}, y)$ as

$$(0_{n_1}, y) = \sum_{i=1}^{k_2} \beta_i (0_{n_1}, b_i).$$

Combining these two equations we get

$$(x, y) = \sum_{i=1}^{k_1} \alpha_i (a_i, 0_{n_2}) + \sum_{j=1}^{k_2} \beta_j (0_{n_1}, b_j).$$

Thus D spans $C_1 \oplus C_2$. Now we need to show that the elements of D are linearly independent. Consider a linear combination of the vectors in D which is equal to zero.

$$\sum_{i=1}^{k_1} \alpha_i (a_i, 0_{n_2}) + \sum_{j=1}^{k_2} \beta_j (0_{n_1}, b_j) = (0_{n_1}, 0_{n_2}) \Rightarrow \sum_{i=1}^{k_1} \alpha_i a_i = 0_{n_1} \text{ and } \sum_{j=1}^{k_2} \beta_j b_j = 0_{n_2}$$

Since the elements of A and B form a basis of C_1 and C_2 respectively, they are linearly independent and the above equation gives us $\alpha_i = 0$ and $\beta_i = 0$. Thus the elements of D are linearly independent. Since they also span $C_1 \oplus C_2$, they form a basis for this space. Since the number of elements in D is $k_1 + k_2$, the dimension of $C_1 \oplus C_2$ is $k_1 + k_2$.

The basis D also gives us the structure of the generator matrix of $C_1 \oplus C_2$. If G_1 is the generator matrix of C_1, then it has the a_i's in the set A as its rows and if G_2 is the generator matrix of C_2 then it has the b_i's in the set B as its rows. The generator matrix for $C_1 \oplus C_2$ has the elements in the set D as its rows. Thus it is given by

$$G = \begin{bmatrix} G_1 & \mathbf{0}_{k_1 \times n_2} \\ \mathbf{0}_{k_2 \times n_1} & G_2 \end{bmatrix}$$

If H_1 is the parity check matrix of C_1 and H_2 is the parity check matrix of C_2, the parity check matrix of $C_1 \oplus C_2$ is given by

$$H = \begin{bmatrix} H_1 & \mathbf{0}_{(n_1-k_1) \times n_2} \\ \mathbf{0}_{(n_2-k_2) \times n_1} & H_2 \end{bmatrix}$$

This can be verified by writing $\mathbf{v} \cdot H^T = \mathbf{0}$ where $v \in \mathbb{F}_2^{n_1+n_2}$.

Since $C_1 \oplus C_2$ is a linear code, its minimum distance is equal to the minimum weight of its nonzero codewords. Let x be a nonzero codeword of minimum weight in C_1 i.e. $d_1 = w_H(x)$. Let y be a nonzero codeword of minimum weight in C_2 i.e. $d_2 = w_H(y)$. Then $(x, 0_{n_2}) \in C_1$ and $(0_{n_1}, y) \in C_2$ because $0_{n_1} \in C_1$ and $0_{n_2} \in C_2$. Since

$w_H((x, 0_{n_2})) = d_1$ and $w_H((0_{n_1}, y)) = d_2$, the minimum distance d_{min} of $C_1 \oplus C_2$ satisfies the following inequality

$$d_{min} \leq \min(d_1, d_2).$$

Let $\mathbf{z}$ be a nonzero codeword in $C_1 \oplus C_2$. Then $\mathbf{z} = (u, v)$ where $u \in C_1$, $v \in C_2$ and both u and v acannot be zero codewords. The Hamming weight of $\mathbf{z}$ is $w_H(\mathbf{z}) = w_H(u) + w_H(v)$. Since at least one of u and v is nonzero and $w_H(u) \geq d_1$ when $u \neq 0_{n_1}$, $w_H(v) \geq d_2$ when $v \neq 0_{n_2}$, we have

$$w_H(\mathbf{z}) \geq \min(d_1, d_2).$$

Since $\mathbf{z}$ was an arbitrary nonzero codeword, we have

$$d_{min} \geq \min(d_1, d_2).$$

Thus $d_{min} = \min(d_1, d_2)$.

0000000	1000011	0100101	0010110	0001111	1100110	1010101	1001100	0110011	0101010	0011001	1110000	1101001	1011010	0111100	1111111
0000001	1000010	0100100	0010111	0001110	1100111	1010100	1001101	0110010	0101011	0011000	1110001	1101000	1011011	0111101	1111110
0000010	1000001	0100111	0010100	0001101	1100100	1010111	1001110	0110001	0101000	0011011	1110010	1101011	1011000	0111110	1111101
0000100	1000111	0100001	0010010	0001011	1100010	1010001	1001000	0110111	0101110	0011101	1110100	1101101	1011110	0111000	1111011
0001000	1001011	0101101	0011110	0000111	1101110	1011101	1000100	0111011	0100010	0010001	1111000	1100001	1010010	0110100	1110111
0010000	1010011	0110101	0000110	0011111	1110110	1000101	1011100	0100011	0111010	0001001	1100000	1111001	1001010	0101100	1101111
0100000	1100011	0000101	0110110	0101111	1000110	1110101	1101100	0010011	0001010	0111001	1010000	1001001	1111010	0011100	1011111
1000000	0000011	1100101	1010110	1001111	0100110	0010101	0001100	1110011	1101010	1011001	0110000	0101001	0011010	1111100	0111111

Table 1: Standard array for the code given in Exercise 1

Coset leader	Syndrome
0000000	000
0000001	001
0000010	010
0000100	100
0001000	111
0010000	110
0100000	101
1000000	011

Table 2: Syndrome table for the code given in Exercise 1