Error Correcting Codes: Combinatorics, Algorithms and Applications **(Fall 2007)**

Lecture 7: Family of Codes

Sep 12, 2007

Lecturer: Atri Rudra *Scribe: Yang Wang & Atri Rudra*

In the previous lecture, we were going to see which codes are perfect codes. Interestingly, the only perfect codes are the following:

- The Hamming codes which we studied in the last couple of lectures,
- The trivial $[n, 1, n]_2$ codes for odd n (which have 0^n and 1^n as the only codewords),
- Two codes due to Golay [1].

The above result was proved by van Lint [3] and Tietavainen [2].

In today's lecture, we will look at an efficient decoding algorithm for the Hamming code and look at some new codes that are related to the Hamming codes.

1 Family of codes

Till now, we have mostly studied specific codes, that is, codes with *fixed* block lengths and dimension. The only exception was the "family" of $[2^r - 1, 2^r - r - 1, 3]_2$ Hamming codes (for $r \geq 2$). The notion of family of codes is defined as following:

Definition 1.1 (Family of codes)**.** *$C = \{C_i\}_{i\geq 1}$ is a family of codes where C_i is a $[n_i, k_i, b_i]_q$ code for each i (and we assume $n_{i+1} > n_i$). The rate of C is defined as*

$$R(C) = \lim_{i\to\infty} \left\{ \frac{k_i}{n_i} \right\}.$$

The relative distance of C is defined as

$$\delta(C) = \lim_{i\to\infty} \left\{ \frac{d_i}{n_i} \right\}.$$

For example, C_H the family of Hamming code is a family of codes with $n_i = 2^i - 1, k_i = 2^i - i - 1, d_i = 3$ and $R(C_H) = 1, \delta(C_H) = 0$. We will mostly work with family of codes from now on. This is necessary as we will study the asymptotic behavior of algorithms for codes, which does not make sense for a fixed code. For example, when we say we say that a decoding algorithm for a code C takes $O(n^2)$ time, we would be implicitly assuming that C is a family of codes and that the algorithm has an $O(n^2)$ running time when the block length is large enough. From now on, unless mentioned otherwise, whenever we talk about a code, we will be implicitly assuming that we are talking about a family of codes.

Finally, note that the motivating question is to study the optimal tradeoff between R and δ.

2 Efficient Decoding of Hamming codes

We have shown that Hamming code has distance of 3 and can thus correct one error. However, this is a *combinatorial* result and does not give us an efficient algorithm. One obvious candidate for decoding is the MLD functions. Unfortunately, the only implementation of MLD that we know will take time $2^{O(n)}$, where n is the block length of the Hamming code. However, we can do much better. The following is a very natural algorithm, which was proposed by Nathan in class (where below $C_{H,r}$ is the $[2^r-1, 2^r-r-1, 3]_2$ Hamming code):

Algorithm 2.1. *Given the received word* $\mathbf{y}$*, first check if* $\mathbf{y} \in C_{H,r}$*. If the answer is yes, we are done. Otherwise, flip the bits of* $\mathbf{y}$ *one at a time and check if the resulting vector* $\mathbf{y}' \in C_{H,r}$*.*

It is easy to check that the above algorithm can correct up to 1 error. If each of the checks $\mathbf{y}' \in C_{H,r}$ can be done in $T(n)$ time, then the time complexity of the proposed algorithm will be $O(nT(n))$. Note that since $C_{H,r}$ is a linear code we have an obvious candidate for checking if any vector $\mathbf{y} \in C_{H,r}$– just check if $\mathbf{y} \cdot H_r = \mathbf{0}$, where recall H_r is the parity check matrix of $C_{H,r}$. Thus, the check involves a matrix-vector multiplication, which can be done in $O(n^2)$. Thus, the proposed algorithm has running time $O(n^3)$.

Remark 2.2. *Note that the above algorithm can be generalized to work for any (binary) linear code with distance* $2t+1$ *(and hence, can correct up to* t *errors): go through all the* $\binom{n}{t}$ *possible error locations and flip all bits under consideration and check if the resulting vector is in the code or not. This will have a running time complexity of* $O(n^{t+2})$ *(as* $\binom{n}{t} \le n^t$*). Thus, the algorithm will have polynomial running time for codes with constant distance (though the running time would not be practical even for moderate values of* t*).*

However, it turns out that for Hamming codes there exists a decoding algorithm with an $O(n^2)$ running time. To see this first note that if the received word $\mathbf{y}$ has no errors then $\mathbf{y} \cdot H_r = \mathbf{0}$. If not, $\mathbf{y} = \mathbf{c} + \mathbf{e_i}$, where $\mathbf{c} \in C$ and $\mathbf{e_i}$ which is the unit vector with the only nonzero element at the i-th position. Thus, if H_r^i stands for the i-th column of H_r,

$$\mathbf{y}H_r = \mathbf{c}H_r + \mathbf{e_i}H_r = \mathbf{e_i}H_r = H_r^i.$$

In other words, $\mathbf{y} \cdot H_r$ gives the *location* of the error. Thus, we have the following algorithm: compute $\mathbf{b} = \mathbf{y} \cdot H_r$. If $\mathbf{b} = \mathbf{0}$, then no error occurred, other wise flip the bit position whose binary representation is $\mathbf{b}$. Since the algorithm computes just one matrix vector multiplication, the modified algorithm above runs in $O(n^2)$ time.

2.1 A Digression

Finally, we come back to a claim that was made a few lectures back. It was claimed that the $[7,4,3]_2$ Hamming code has G_3 and H_3 as it generator matrix and parity check matrix respectively, where

$$G_3 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} H_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

It can be verified that G_3 and H_3 have full rank and $G_3 H_3^T = 0$. Given these observations, the following lemma proves the claim.

Lemma 2.3. *Given matrix G of dimension $k \times n$ that is the generator matrix of code C_1 and has full row rank and matrix H of dimension $(n-k) \times n$ that is parity check matrix of code C_2 and has full column rank and $GH^T = 0$, then $C_1 = C_2$.*

Proof. We first prove that $C_1 \subseteq C_2$. Given any $\mathbf{c} \in C_1$, $\exists \mathbf{x}$ such that $\mathbf{c} = \mathbf{x}G$. Then,

$$\mathbf{c}H^T = \mathbf{x}GH^T = 0,$$

which implies that $\mathbf{c} \in C_2$, as desired.

To complete the proof note that as H has full rank, its null space (or C_2) has dimension $n - (n-k) = k$ (this follows from a well known fact from linear algebra). Now as G has full rank, the dimension of C_1 is also k. Thus, as $C_1 \subseteq C_2$, it has to be the case that $C_1 = C_2$.[1] $\square$

3 Dual of a Linear Code

Till now, we have thought of the parity check matrix as defining a code via its null space. However, what happens if we think of the parity check matrix as a generator matrix? The following definition addresses this question.

Definition 3.1 (Dual of a code)**.** *Let H be the parity check matrix of C, then the code generated by H is called the dual of C and is denoted by $C^{\perp}$.*

It is obvious from the definition that $dim(C^{\perp}) = n - dim(C)$. The first example that might come to mind is $C^{\perp}_{H,r}$, which is also known as the *Simplex code* (we will denote it by $C_{Sim,r}$). Adding an all 0's column to H_r and using the resulting matrix as a generating matrix, we will get the *Hadamard* code (we will denote it by C_{Had}, r). We claim that $C_{Sim,r}$ and $C_{Had,r}$ are $[2^r - 1, r, 2^{r-1}]_2$ and $[2^r, r, 2^{r-1}]_2$ codes respectively. The claimed block length and dimension follow from the definition of the codes, while the distance follows from the following result.

Proposition 3.2. *$C_{Sim,r}$ and $C_{Had,r}$ both have a distance of 2^{r-1}.*

Proof. We first show the result for $C_{Had,r}$. In fact, we will show something stronger: every codeword in $C_{Had,r}$ has weight exactly 2^{r-1} (the claimed distance follows from this as the Hadamard code is a linear code). Consider a message $\mathbf{x} \neq 0$ that its ith entry is $x_i = 1$. $\mathbf{x}$ is encoded as

$$\mathbf{c} = (x_1, x_2, \ldots, x_r)(H_r^0, H_r^1, \ldots, H_r^{2^r-1}),$$

where H_r^j is the binary representation of $0 \leq j \leq 2^r - 1$ (that is, it contains all the vectors in $\{0,1\}^r$). Further note that the jth bit of the codeword is $\mathbf{x}H_r^j$. Group all the columns of the

[1] If not, $C_1 \subset C_2$ which implies that that $|C_2| \geq |C_1| + 1$. The latter is not possible if both C_1 and C_2 (as linear subspaces) have the same dimension.

generating matrix into pairs $(\mathbf{u}, \mathbf{v})$ such that $\mathbf{v} = \mathbf{u} + \mathbf{e}_i$ (i.e. $\mathbf{v}$ and $\mathbf{u}$ are the same except in the ith position). Notice that this partitions all the columns in 2^{r-1} disjoint pairs. Then,

$$\mathbf{xv} = \mathbf{x}(\mathbf{u} + \mathbf{e_i}) = \mathbf{xu} + \mathbf{xe_i} = \mathbf{xu} + x_i = \mathbf{xu} + 1.$$

Thus we have that exactly one of $\mathbf{xv}, \mathbf{xu}$ is 1. As the choice of the pair $(\mathbf{v}, \mathbf{u})$ was arbitrary, we proved that for any non-zero codeword $\mathbf{c} \in C_{Had}$, $wt(\mathbf{c}) = 2^{r-1}$.

For the simplex code, we observe that all codewords of $C_{Had,3}$ are obtained by padding a 0 to the codewords in $C_{Sim,r}$, which implies that all non-zero codewords in $C_{Sim,r}$ also have a weight of 2^{r-1}. $\square$

We remark that the family of Hamming code have a rate of 1 and a (relative) distance of 0 while the family of Simplex/Hadamard codes have a rate of 0 and a relative distance of $1/2$. Notice that both code families either have rate or relative distance equal to 0. Given this, the following question is natural.

Question 3.3. *Does there exists a code family C such that $R(C) > 0$ and $\delta(C) > 0$ hold* simultaneously*?*

Note that the above is a special case of the general question that we are interested in:

Question 3.4. *What is the optimal tradeoff between $R(C)$ and $\delta(C)$ that can be achieved by some code family C?*

References

[1] M. J. E. Golay. Notes on digital coding. *Proceedings of the IRE*, 37:657, 1949.

[2] Aimo Tietavainen. On the nonexistence theorems for perfect error-correcting codes. *SIAM Journal of Applied Mathematics*, 24(1):88–96, 1973.

[3] Jacobus H. van Lint. Nonexistence theorems for perfect error-correcting codes. In *Proceedings of the Symposium on Computers in Algebra and Number Theory*, pages 89–95, 1970.